



Configuration analysis to support product redesign for end-of-life disassembly

SHIVAKUMAR VISWANATHAN[†] and VENKAT ALLADA^{†*}

End-of-life product disassembly is an important process that makes the parts of a product available for different material and part recycling processes at end of its useful life. However, the efficiency of the disassembly process greatly affects the economics of meeting the environmental goals set for the product. An important determinant of the efficiency of disassembly is the product configuration. Therefore, it is essential for the designer to assess these implications of the configuration while designing a product for end-of-life disassembly. In this paper, a formal model, called the Configuration-Value (CV) model, is proposed to evaluate and analyse the effect of configuration on disassembly. The model focuses on the rate of value extraction during the disassembly process. The model is used to identify the critical bottlenecks in the configuration, to help the designer to identify the design changes that need to be made to improve the product 'disassemblability'. An example is presented to demonstrate the application of the proposed model.

1. Introduction

Product end-of-life (EOL) disassembly is a process of growing importance due to the increasing requirement of companies to take environmental factors into consideration over the entire product life cycle. The EOL disassembly process transforms the product at the end of its useful life from its fully assembled state to the specific part and subassembly states required for various recycling processes (remanufacture, material recycling, etc.) and also for waste treatment and energy recovery processes (Navinchandra 1993, Alting and Legarth 1995).

The disassembly process, however, is not economically adiabatic and is associated with significant investments of labour and resources. Improving the product's 'disassemblability' by design or design for disassembly (DfD) is a powerful approach towards reducing these associated costs and hence increasing the potential economic value salvageable from the product. The importance of DfD has been receiving a lot of attention from researchers and several reviews of the research literature in DfD and related domains of design for recycling and remanufacture have been published (Boothroyd and Alting 1992, Wittenberg 1992, Jovane *et al.* 1993, Keoleian and Menery 1994, Zhang *et al.* 1997, Gungor and Gupta 1999). Therefore, at this time the characteristics of DfD will not be dwelled upon and reader must refer to references for further details on the general concerns involved in DfD.

Revision received February 2000.

[†] 205, Engineering Management Department, Sustainable Design Laboratory, University of Missouri-Rolla, Rolla, MO 65401, USA.

* To whom correspondence should be addressed. e-mail: allada@umr.edu

The present paper is concerned with the implications of the product configuration the EOL disassembly process. The product configuration is the relative spatial and logical arrangement of the different parts of the product assembly with respect to each other. The most fundamental, and probably the most studied (Baldwin *et al.* 1991, Homem De Mello and Sanderson 1991, Wilson and Latombe 1994), effect of product configuration on disassembly is that it determines what part or parts can be removed at a given stage in the disassembly process. However, studies of these effects have primarily focused on its implications for assembly planning.

EOL disassembly presents a scenario that is far from being a mere reversal of the assembly process. The most important difference is that EOL disassembly is driven by the objective to maximize the value (both from an environmental and an economic standpoint) extracted from the fully assembled product at the end of its useful life, while minimizing the cost of the disassembly. As a result, it may not be necessary to disassemble every part of the product. Further, there exists great flexibility in the specific disassembly operations used that could even involve a wanton destruction of certain parts and fasteners to facilitate rapid disassembly. Viewed from the perspective of meeting this objective the configuration begins to mean more than the mere spatial arrangement of parts. For example, consider the two assemblies in figure 1. Both assemblies are identical geometrically but the corresponding parts in each assembly differ in their individual attribute magnitudes (Q_i). If the part attribute is the EOL 'value' of the parts then these two assemblies are very different in their overall structure due to the difference in the way that they satisfy the objective function. Therefore, we wish to argue that this complex association formed between the product configuration and the 'value' attributes of each part is a structure of great relevance from the EOL disassembly standpoint.

In this paper, a formal framework to support this argument is presented as is a method of extracting design information based on this framework to aid in the configuration redesign to support DfD.

1.1. *Related work*

Simon (1991) and Simon and Dowie (1993) proposed a number of indices for assessing the overall recyclability of a given product. Among these indices, parallelism was identified as an important part of the product configuration attribute affecting disassembly.

Marks *et al.* (1993) recognized the importance of the effect of product structure on different product life-cycle concerns. The LINKER representation has been developed that allows extensive reasoning about the product structure.

Jovane *et al.* (1997) emphasized the importance of a thorough analysis of the product structure in order to determine the weak points and redesign modifications with significant potential for improvement. Based on this emphasis, a set of indices using fuzzy logic constructs is proposed to identify the disassembly operations to be improved by redesign depending on the specific context of their cost, revenue and location.

Feldmann and Meedt (1997) developed software that determines the optimal EOL destinations for the different parts of a product. Of interest are the indices developed for product structure analysis, namely the *extraction factor* to identify the components having high potential profit but with associated with high disassembly times, and the *separation potential* to identify the fasteners that have a high disassembly time and block parts having high revenues.

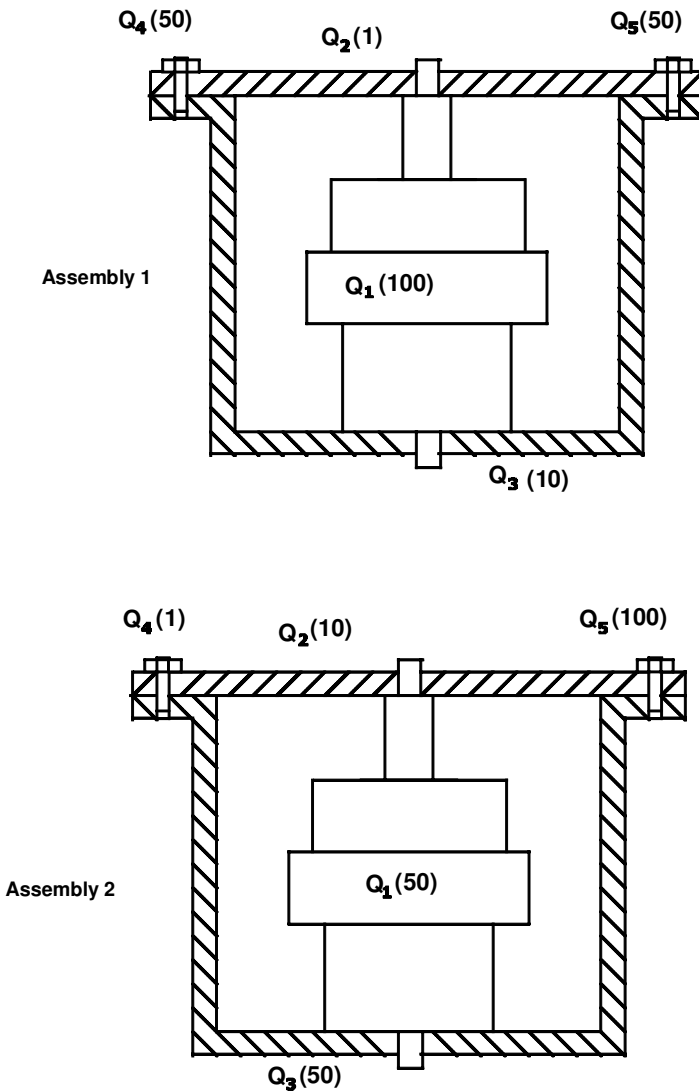


Figure 1. Comparison of configurations.

Luttrupp (1997) proposed some elementary classifications of product structures based on their disassembly characteristics. Though the paper does not make any attempt at formalization, the intuitive idea of disassembly structures presented in this work has been a major motivation for the present paper.

The work of Rosen (1996) has come to the authors' attention due to a number of similarities in the methods used for formalization. The paper has the development of a formal foundation for configuration design for the life-cycle as a central theme. Lattice theory and combinatoric techniques are used to develop the Product Module Reasoning System (PMRS) to incorporate life cycles concerns related to the development of modular product architectures.

As can be seen, though the importance of the product configuration for disassembly has been recognized extensively the approaches proposed for analysis have

tended to be procedural in nature rather than to model formally the effect of the configuration on disassembly. The present paper aims at filling this gap by proposing a novel formalization to model the configuration effects.

2. Rationale

Each product segment (be it a single part or an aggregation) has an effective EOL value by virtue of its EOL destination—reuse, material recycling, remanufacture or disposal. The EOL value of the i th segment (V_i) is given by the sum of the primary value (v_i) plus the savings accrued due to reclamation (s_i), if any (equation 1).

$$V_i(\$) = v_i(\$) + s_i(\$). \quad (1)$$

The primary value is the maximum reclamation value possible for the segment assuming *no further disassembly*, while considering the processing and transportation costs involved in making the segment available for economic use, as given by equation (2). If the EOL destination as determined by v_i is not disposal, the savings accrued (s_i), is equal to the disposal cost (i.e. the absolute magnitude of v_i^{disposal}).

$$v_i = \max(v_i^{\text{reuse}}, v_i^{\text{remanufacture}}, v_i^{\text{recycle}}, v_i^{\text{disposal}}, v_i^{\text{other}}). \quad (2)$$

The extent to which the environmental goals set by a company are realized is eventually controlled by economic considerations. Therefore, the optimal depth of disassembly is determined based on a trade-off between the values of the disassembled segments and the associated disassembly costs. Though many different procedures have been proposed to determine this optimal disassembly depth (Navinchandra 1993, Zussman *et al.* 1994, 1998, Johnson and Wang 1995, Harjula *et al.* 1996, Penev and De Ron 1996, Jovane *et al.* 1997, Lambert 1997, Johnson and Wang 1998), the generic objective function used to obtain the optimal depth can be expressed as:

$$U_k = \frac{\sum_{i=1}^k V_i + {}^k V_0 - \sum_{i=1}^k C_i}{\sum_{i=1}^k C_i}, \quad (3)$$

where

- U_k are the returns per unit cost after the k th disassembly operation,
- V_i is the value of the product segment disassembled in the i th disassembly operation,
- ${}^k V_0$ is the value of the undisassembled segment remaining after the k th disassembly operation,
- C_i is the cost of the i th disassembly operation, given by a function of the labour rate, time taken to perform the disassembly operation and resource consumption.

The optimal disassembly depth is the state of the state of the assembly when this objective function is maximized, i.e. when U_{\max} is obtained. But the caveats associated with such an optimal depth are:

- U_{\max} though optimal for the product may be too low to be really profitable;

- the optimal depth may not coincide with the depth required actually to be environmentally beneficial (Harjula *et al.*, 1996).

Hence, the objective of DfD is to increase the magnitude of U_{\max} while ensuring that the optimal depth (based on U_{\max}) matches the desired environmentally beneficial depth.

From equation (3), it can be seen that U_{\max} can be increased by a combination of any or all of the following.

- (1) Reducing the cost of each disassembly operation.
- (2) Increasing the value of the product segments (Pnueli and Zussman 1997).
- (3) Rearrangement of the value and cost elements in the sequence.

This last method is where the effect of the configuration is significantly felt. For example, consider two disassembly sequences, A and B , for the same product. Sequence A is such that the segments are salvaged in the descending order of their values and in the ascending order in sequence B . If one were to assume that the cost of all the disassembly operations in A and B are equal, one would see that there is a significant difference in the way U varies over the sequence. Sequence A has a higher U over most of the sequence as compared with sequence B making it more desirable than B . But the existence of a sequence like A in a product is directly dependent on its configuration (Johnson and Wang 1995). This distribution of value associated with a sequence shall be referred to, from this point on, as the *value precedence* (VP). Analysis of the patterns by which the value of the segments are unravelled in the VPs due to the effect of the configuration constitutes the basic rationale of the Configuration Value (CV) model proposed in this paper.

3. Background

3.1. Terminology

Dividend – the entity on which the disassembly operation is performed.

Quotient – the entity that results due to the performance of a disassembly operation on the dividend. The quotient can be further categorized as defined below.

Atomic quotient – requires no further disassembly according to the disassembly strategy. Therefore an atomic quotient could be a single part, fastener, subassembly, clump, cluster or a part fraction formed due to destructive disassembly. The atomic quotients are referred to simply as quotients throughout the paper unless otherwise mentioned.

Metadividend – requires further disassembly after removal from the quotient. Hence, the metadividend becomes the dividend later on in the disassembly process.

The above terms are explained using the AND/OR graph (Homem De Mello and Sanderson 1991) in figure 2. The node $\{a, b, c\}$ represents the initial dividend. On completion of the first disassembly operation, shown by bold arrows connected by an arc, the quotients $\{b\}$ and $\{a, c\}$ are obtained. Since $\{a, c\}$ is to be disassembled further, it is by definition, a metadividend; while $\{b\}$ is an atomic quotient. The metadividend $\{a, c\}$ becomes the dividend for the next disassembly operation resulting in the formation of the atomic quotients $\{a\}$ and $\{c\}$.

3.2. Assumptions

The basic assumptions made are as follows.

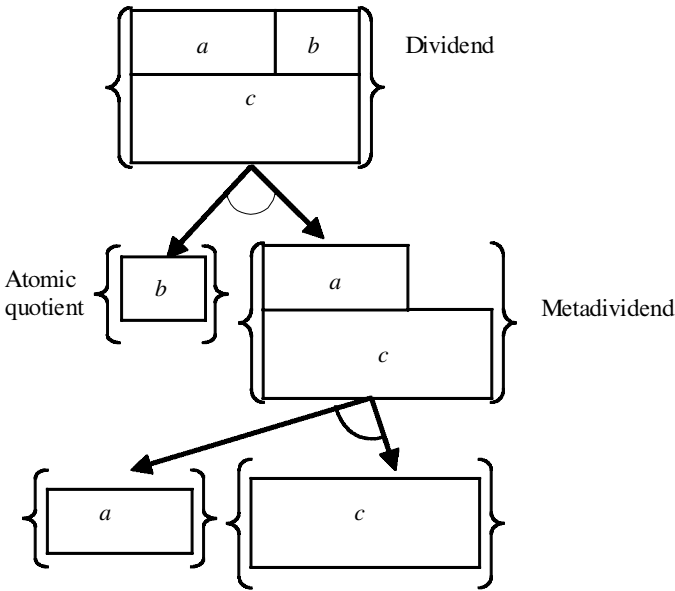


Figure 2. Disassembly terminology using AND/OR graph.

- (1) The complete description of the configuration and the values of the different quotients are available *a priori*.
- (2) Only non-destructive disassembly is considered here.
- (3) After any disassembly operation, a maximum of two quotients is formed. Of these two quotients, only one can be a metadividend. Though the requirement that the disassembly be sequential (i.e. no component parallel) and that the components be 1 – disassemblable (i.e. components can be disassembled with one motion) (Shyamsundar *et al.* 1998) are not crucial to the proposed model necessary, in this paper it is assumed that it is so.
- (4) In developing the CV model, it is assumed that some preliminary identification of the groups of parts that could be removed together and hence be treated as a single atomic quotient has already been made. This is a reasonable assumption as the identification of parts having compatible materials that could be removed as a clump, or identification of functional subassemblies that need not be disassembled further, are dependent more on specific part attributes rather than on configuration relations. It is also recognized that the eventual disassembly depth may result in segments that contain more than one of these starting quotients.
- (5) Disassembly is performed by a single agent, i.e. situations where multiple disassembly agents can work on the product simultaneously are not considered.

Other assumptions made will be discussed as they are employed.

3.3. Partially ordered sets

Central to the development of the CV model is the use of partially ordered sets or posets. For the standard books on the subject, see Neggers and Kim (1998). In the

interest of clarity a brief introduction to the poset concepts used in the present paper is provided below.

A partially ordered set or poset, represented by (P, \leq) , is a system consisting of a non-empty set P and a binary relation \leq in P such that the following conditions (also called the weak inclusion conditions) are satisfied for all $x, y, z \in P$:

- (1) $x \leq x$
- (2) If $x \leq y$ and $y \leq x$ then $x = y$;
- (3) If $x \leq y$ and $y \leq z$ then $x \leq z$.

The relation \leq is a partial order in the set P , and P is said to be partially ordered by the relation \leq .

Two distinct elements x and y in a poset (P, \leq) are *comparable* if either $x \leq y$ or $y \leq x$, and *incomparable* otherwise. Two incomparable elements, x and y , are denoted by $x\theta y$. A poset (P, \leq) is called a *chain* if every pair of distinct elements in P are comparable.

Consider a poset (P, \leq) where $x, y \in P$. Then y is said to *cover* x , if $x < y$ and there is no $z \in P$ such that $x < z < y$. This is denoted by $x \cup y$.

A point x of the poset (P, \leq) is called a *maximal* element if there is no point $y \in P$ with $x < y$ in P . Similarly a point x of the poset (P, \leq) is called a *minimal* element if there is no point $y \in P$ with $y < x$ in P .

Partially ordered sets can be conveniently represented pictorially using a *Hasse* diagram. A Hasse diagram can be drawn for a given poset (P, \leq) using the following rules. Each element in P is represented using a distinct point such that whenever $x < y$, the point y is drawn higher than the point x . If $x \cup y$ then the points representing x and y are connected by a straight line.

Consider two posets, (P, \leq) and (Q, \leq) . A mapping $f: P \rightarrow Q$ is called *order preserving* if for any x and $y \in P$, $x \leq y$ implies that $f(x) \leq f(y)$. A poset (S, \leq) is a *subposet* of (P, \leq) if $S \subseteq P$ and if the inclusion mapping $i: S \rightarrow P$ is order preserving. If the condition $x < z < y$ is true on (P, \leq) and if $x, y \in S$ implies that $z \in S$, where S is a subset of P , then S is called a *convex subposet*.

4. Configuration–Value model

4.1. Sequence space

Let Q be the assembly of m quotients that needs to be disassembled. A disassembly sequence on Q can be expressed as a poset, $\mathbf{S} = (Q, <)$, where $x < y$ implies that atomic quotient y is removed prior to atomic quotient x , for any x and $y \in Q$. \mathbf{S} is a chain due to the assumption that only one atomic quotient is removed at any stage.

The set, Ω_Q , is defined as the set of all sequences possible on Q , i.e. $\Omega_Q = \{\mathbf{S}_1, \mathbf{S}_2, \mathbf{S}_3, \dots, \mathbf{S}_n\}$ where \mathbf{S}_i denotes the i th sequence and n is the total number of feasible sequences. Therefore, $\Omega_Q \subseteq \Omega$, where Ω is the set of all sequences, feasible or infeasible, possible for the same set of m quotients where the disassembly is total. Therefore $\Omega = \{\mathbf{S}_1, \mathbf{S}_2, \mathbf{S}_3, \dots, \mathbf{S}_k\}$ where $k = \text{factorial}(m)$ or $m!$

The distance, between any two sequences $\delta(\mathbf{S}_i, \mathbf{S}_j)$ can be defined as the cardinality of the difference $(\mathbf{S}_i - \mathbf{S}_j)$. Therefore, if $\mathbf{S}_i = a < b < c < d$ and $\mathbf{S}_j = a < c < b < d$ then $\delta(\mathbf{S}_i, \mathbf{S}_j) = 1$ as they differ by one ordered pair $b < c$. It can also be seen that $\delta(\mathbf{S}_i, \mathbf{S}_j) = \delta(\mathbf{S}_j, \mathbf{S}_i)$.

The set, Ω , can be conceptualized as an undirected graph where an edge exists between any two sequences, \mathbf{S}_i and \mathbf{S}_j if $\delta(\mathbf{S}_i, \mathbf{S}_j) = 1$, where $\mathbf{S}_i, \mathbf{S}_j \in \Omega$. Every vertex

of this graph therefore has $(m - 1)$ neighbours where m is the number of quotients in the assembly, Q . The graph so constructed, $G(\Omega)$, consists of single connected component. This implies that there exists at least one chain between any two sequences, S_i and S_j , where $S_i, S_j \in \Omega$ (*chain* here is with reference to its graph theoretic meaning and not the poset definition). Of all the chains that exist between S_i and S_j in Ω , there exists at least one *monotonic* chain, i.e. where $\delta(S_i, S_x) + \delta(S_j, S_x) = \delta(S_i, S_j), \forall S_x$ on the chain.

Ideally, Ω_Q for an assembly is the set of all possible sequences under the assumption that the assembly is free floating, i.e. there exist no constraints on the directions in which the quotients can be removed. When such constraints exist they are directly reflected in the Ω_Q . For example, the disassembly of the pen under different constraints (figure 3) can result in very different Ω_Q s. In figure 3, hatched lines indicate fixing constraints on the quotient in question.

4.2. *Petition concept*

From the discussion above it was seen that the maximization of the extracted value is the primary objective of EOL disassembly. Therefore, a disassembly sequence where the highest valued quotients can be extracted early in the disassembly is very desirable. Hence, an ideal sequence would be one where the quotients could be removed in the descending order of their value. This VP is called the *desired value precedence* (DVP) or S_{dvp} . S_{dvp} belongs to Ω but may not belong to Ω_Q .

If no two quotients have the same value then S_{dvp} is unique. For the sake of convenience, we will consider this to be true for the rest of the paper. As S_{dvp} is the

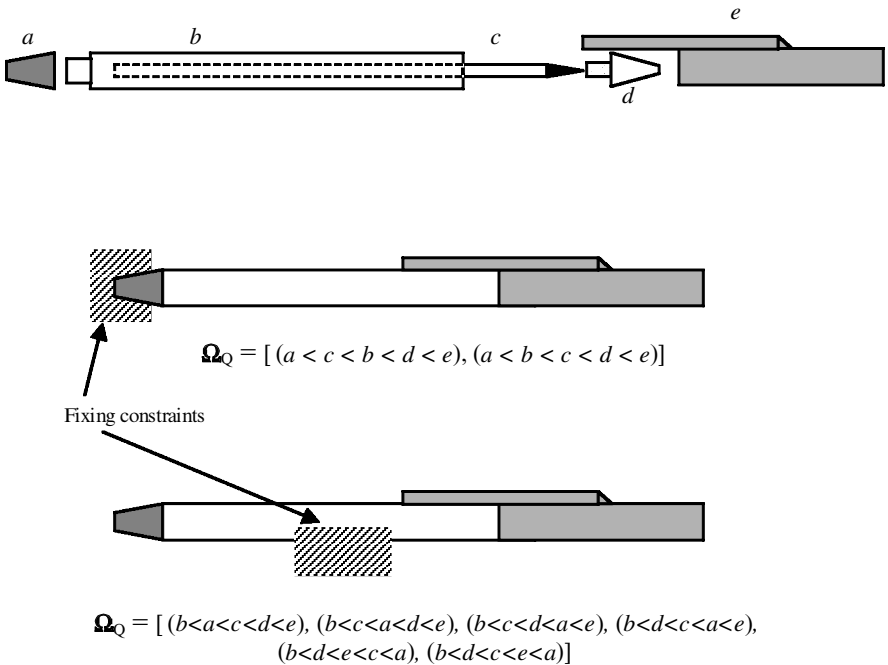


Figure 3. Effect of constraints on Ω_Q . (a) Pen assembly; (b) constraint on quotient a; (c) constraint on quotient b.

ideal sequence, any sequence, S_x , where $\delta(S_{dvp}, S_x) > 0$ is therefore non-ideal. Hence, farther a sequence is from the S_{dvp} lower is its suitability for value extraction during EOL disassembly. However, if two sequences S_x and S_y ($S_x \neq S_y$) have $\delta(S_x, S_{dvp}) = \delta(S_y, S_{dvp})$, it does not follow that they are non-ideal to the same extent.

In order to quantify better the extent of departure of a sequence from S_{dvp} , the concept of a petition is introduced. Every ordered pair in the sequence S that differs from the S_{dvp} is called a *petition*. The name arises from the perception of the high valued quotient ‘petitioning’ the lower valued quotient to ‘get behind’ in the sequence. The petitions over the sequence S can be defined as a poset, $T_S = (Q, <_\psi)_S$ where $b <_\psi a$ implies that:

- $b < a$; and
- quotient $b(V_b)$ is greater than quotient $a(V_a)$, where $a, b \in Q$.

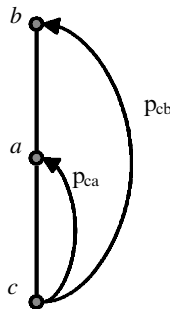
Each petition is assigned a value, p_{ab} , ($a <_\psi b, a$ and $b \in Q$):

$$p_{ab} = T_1 e^{T_2}, \tag{4}$$

where

- T_1 $(V_a - V_b)/(V_{max} - V_{min})$,
- T_2 $(V_a - V_{min})/(V_{max} - V_{min})$,
- V_{max} maximum value of a quotient in S ,
- V_{min} minimum value of a quotient in S ,
- $V_{max} \neq V_{min}$.

The first term (T_1) is the measure of the value difference that caused the petition. The term (e^{T_2}) magnifies this value difference as a function of the values of the involved quotients. Hence, if petitions p_{ij} and p_{ab} have T_1 equal, then $p_{ij} > p_{ab}$ would be true only if $V_i > V_a$. The net petition magnitude, P_S , is the sum of the petition magnitudes over T for the sequence S . Therefore, for any two sequences, S_x and S_y , if $P_x > P_y$ then S_x is non-ideal to a greater extent than S_y . An intuitively useful way of representing the relation between the sequence and the petition poset is the *petition diagram*. The petition diagram is an enhancement of the Hasse diagram for the sequence obtained by drawing an arc from the quotient b to a , if $b <_\psi a$, where $a, b \in Q$ (figure 4).



$$(V_b < V_a; V_a < V_c)$$

Figure 4. Example of a petition diagram.

Finally, to summarize, the application of the petition concept allows the mapping of the VP of any sequence S to a single real valued number P_S .

4.3. Configuration redesign

The DVP though ideal from the value perspective, does not consider the disassembly costs involved. Therefore, the *optimal value precedence* (OVP) or S_{ovp} is defined as the sequence in Ω_Q having the highest magnitude of U_{avg} , i.e. the average of U over the entire sequence. The reason for using U_{avg} is that the depth of disassembly (obtained using U_{max}) cannot serve as the baseline for analysis as it could keep varying after every modification in the design of the assembly.

S_{ovp} may not be equal to S_{dvp} . Therefore, the idealized objective for configuration redesign can be stated as the modification of the configuration of Q so as to make $S_{ovp} = S_{dvp}$. However, actual design constraints may prevent the achievement of this idealized objective, so an addendum to the objective would be—to the extent possible.

The sequence having the lowest petition magnitude in Ω_Q is not necessarily the S_{ovp} . This sequence in Ω_Q having the lowest petition magnitude is called the *best value precedence* (BVP) or S_{bvp} . When $S_{dvp} \in \Omega_Q$, $S_{bvp} = S_{dvp}$ and when $S_{dvp} \notin \Omega_Q$, S_{bvp} is such that there exists no other sequence S_i in Ω_Q such that $\delta(S_i, S_{dvp}) < \delta(S_{bvp}, S_{dvp})$.

The three sequences, S_{dvp} , S_{bvp} and S_{ovp} together serve to characterize any given configuration. S_{dvp} can be considered to define a global reference point while S_{bvp} defines a local reference point (embedded in the global system), for S_{ovp} . By this interpretation, the objective to make $S_{ovp} = S_{dvp}$ could be approached in two stages.

- (1) Increase the similarity between S_{bvp} and S_{dvp} .
- (2) Increase the similarity between S_{ovp} and S_{bvp} .

The achievement of these objectives could take different forms. Let D be a design action that acts on the assembly Q converting it into its modified form, Q' , i.e. $D: Q \rightarrow Q'$.

- (1) If the only change due to D is in S_{ovp} , while Ω_Q , S_{dvp} , S_{bvp} and S_{ovp} remain unchanged then it implies a modification in the effect of the cost elements. However, the configuration and its effect on the extraction of value remain unchanged. Such a change is called a *passive* modification. If the distance $\delta(S_{ovp}, S_{dvp})$ in Q' is less than in Q and ${}^Q U_{avg} < {}^{Q'} U_{avg}$ then it is a *passive* improvement in the configuration.
- (2) If S_{dvp} changes, but Ω_Q remains unchanged, then it implies a modification in the values of the quotients while leaving the configuration of the assembly unaffected. The variation in S_{dvp} would be accompanied by a change in S_{bvp} and S_{ovp} . Though this could lead to an improvement in the disassembly characteristics, this kind of modification is passive, as there is no change in the configuration.
- (3) If the design action results in a modification in Ω_Q then it is called an *active* modification. This could be due to (a) a change in the number of quotients (i.e. a modification in Ω) or a (b) change in the spatial relationship between the quotients with Ω remaining unchanged.

An actual design action can be a combination of passive and active modifications. However, in this paper the focus is restricted to *active* modification. In the next

section, we shall describe the use of the CV model to identify bottleneck spatial relationships in order to aid active modification of the second type.

5. CV model application

5.1. Conditional and non-conditional precedence relations

To remove a quotient a set of precedence constraints may need to be satisfied. For example, for the pen in figure 3a, in order to remove quotient c , the disassembly of either (a or (e and d)) must precede that of c .

As can be seen above the constraints that need to be satisfied are not necessarily unique. The specific disassembly sequence determines the particular precedence constraints that are satisfied. These sequence-specific precedence requirements are referred to as *conditional precedence* (cp) relations. For example, consider the sequence $b < c < d < e < a$. Quotient a is the first quotient to be removed and has no precedence constraints. Now, in order to remove quotient c , either (a or (e and d)) must precede it. Of these constraints, the removal of a is the first precedence constraint to be satisfied. Therefore quotient a is said to be a conditional precedent of c . Similarly if the sequence were to be $b < c < a < d < e$ then e and d would be the cps of d . A cp poset, $S_c = (Q, \not\prec)$ is defined to be such that $x \not\prec y$ implies that $x < y$ and y is the cp of x where $x, y \in Q$. The cp poset is in effect a mapping $c: S \rightarrow S_c$ such that $x < y$ implies that $c(x) < c(y)$ or $c(x)\theta c(y)$.

Petitions can be defined on this cp poset. The cp petition poset for the sequence S is defined as, $C_S = (Q, \not\prec_\psi)_S$, where $b \not\prec_\psi a$ ($a, b \in Q$) implies that:

- $b \not\prec a$
- $V_a < V_b$.

Also, $C_S \subseteq T_S$.

The objective of active modification centered on the S_{bvp} is to make $S_{bvp} = S_{dvp}$ or at least to reduce P_{bvp} . The petitions in the total petition poset of the BVP, i.e. T_{bvp} , are the obvious targets for improvement. So the question to be asked is why a given petition, p_{ab} exists or why quotient b precedes quotient a in S_{bvp} . This may be so due to a cp relation but a petition could also arise due to an ncp relation.

A proportion of the petitions exist between quotients that have no cp relations but are the 'price' paid in order to obtain the lowest overall feasible P . The occurrence of such instances also noted by Scheuring (1995) and Navinchandra (1993). These are called *non-conditional precedence* (ncp) petitions.

The ncp petition poset, $N = (N, <_\psi)_S$ is a subset of T such that:

- $N \subseteq Q$
- $x <_\psi y$ is true for all $(x, y) \in (N, <_\psi)_S$
- $x \not\prec_\psi y$ is *not* true for any $(x, y) \in (N, <_\psi)_S$.

Therefore for any sequence S , $N_S \cup C_S = T_S$.

It will shortly be shown that ncp petitions are entailed by certain cp petitions in the sequence. The removal of these entailing cp relations would cause the ncp petitions to disappear. Typically, product structure analysis methods proposed by different authors look only at direct precedence effects. However, the presence of these ncp petitions reveals that prioritization based purely on precedence criteria is potentially misleading due to the presence of entailments which may make certain pre-

cedence relations more important than immediately apparent. In order to prove this assertion, the following structures are defined.

A poset called the span, $\mathbf{Z}_{ij} = (Z_{ij}, \not\subseteq)_{\mathbf{S}}$ is defined over the sequence \mathbf{S} :

- \mathbf{Z}_{ij} is a convex subposet of (Q, R) , where the subscripts i and j are the minimal and maximal elements of \mathbf{Z}_{ij} respectively and $Z_{ij} \subseteq Q$;
- $(Z_{ij}, \not\subseteq_{\psi})_{\mathbf{S}}$ is a subposet of $(Q, \not\subseteq_{\psi})_{\mathbf{S}}$;
- for all $x \in Z_{ij}$ there exists a $y \in Z_{ij}$, such that either $(x \not\subseteq_{\psi} y)$ or $(y \not\subseteq_{\psi} x)$ exists;
- for any $x, y \in Q$, if $(x \not\subseteq_{\psi} y)$ exists and if $x \in Z_{ij}$ then $y \in Z_{ij}$.

From this definition, it can be seen that $(Z_{ij}, \not\subseteq_{\psi})_{\mathbf{S}}$ is a connected component of $(Q, \not\subseteq_{\psi})_{\mathbf{S}}$. The concept of a subsposet is also handy. A subsposet \mathbf{z}_{ij} is a subposet of the span \mathbf{Z}_{ij} and is defined by the first three conditions listed above. The smallest valid subsposet has a cardinality of two and occurs when the subsposet \mathbf{z}_{xy} is such that x covers y . This is called a *minimal* subsposet (${}^m\mathbf{z}_{xy}$).

If $x \not\subseteq_{\psi} y$, where $x, y \in Z_{ij}$, then for any $z \in Q$, if $x \not\subseteq z \not\subseteq y$ then either $x \not\subseteq_{\psi} z$ or $z \not\subseteq_{\psi} y$ exists. This property is called *ψ -convexity*. As a result of this, if \mathbf{Z}_{ij} is not a minimal span then it necessarily contains or nests at least one subsposet. The nesting pattern between a span and the subsposets can also be expressed as a partial order using the \subset operator in the same way as the $<$ operator.

Lemma 1: *If a sequence poset \mathbf{S} has a minimal subsposet \mathbf{z}_{xy} such that $(x, y) \in (N, <_{\psi})_{\mathbf{S}}$ then $\mathbf{S} \neq \mathbf{S}_{\text{bvp}}$.*

Proof: Consider a configuration having a sequence $\mathbf{S} = \mathbf{S}_{\text{bvp}}$ such that $x <_{\psi} y \in (N, <_{\psi})_{\mathbf{S}}$ and \mathbf{z}_{xy} is a minimal subsposet. Since x and y have an ncp relation, a sequence \mathbf{S}' , that is identical to \mathbf{S} in all respects except that $y < x$, can belong to Ω_Q . As $x <_{\psi} y$ is true, $y <_{\psi} x$ cannot exist because V_x is greater than V_y . Therefore, $\mathbf{P}_{\mathbf{S}'} = \mathbf{P}_{\text{bvp}} - p_{xy}$. But the existence would violate the definition that the \mathbf{S}_{bvp} has the lowest petition magnitude associated with it. Therefore, we can conclude that a sequence \mathbf{S} having a minimal subsposet \mathbf{s}_{xy} such that $(x, y) \in (N, <_{\psi})_{\mathbf{S}}$ cannot be the \mathbf{S}_{bvp} . □

Lemma 2: *If a sequence poset \mathbf{S} has a span \mathbf{Z}_{ij} such that $(x, y) \in (N, <_{\psi})_{\mathbf{S}}$ for every $x <_{\psi} y (x, y \in Z_{ij})$ then $\mathbf{S} \neq \mathbf{S}_{\text{bvp}}$*

Proof: Consider configuration having a sequence $\mathbf{S} = \mathbf{S}_{\text{bvp}}$ with a span \mathbf{Z}_{ij} such that $(x, y) \in (N, <_{\psi})_{\mathbf{S}}$ for every $x <_{\psi} y (x, y \in Z_{ij})$. By definition, all spans are ψ -convex. Therefore \mathbf{Z}_{ij} necessarily has at least one minimal subsposet \mathbf{z}_{ab} where $a <_{\psi} b$ and $a \cup b, a$ and $b \in Z_{ij}$. From the definition of \mathbf{Z}_{ij} , $(a, b) \in (N, <_{\psi})_{\mathbf{S}}$. But if this were true then it would violate Lemma 1. Therefore such a sequence \mathbf{S} cannot be the \mathbf{S}_{bvp} . □

The lemmas show that an ncp span always nests at least one cp span. This means that the existence of cp petitions is a necessary precondition for the existence of ncp petition relations. In the present model, all the cp petitions nested in an ncp petition are considered to be equally ‘responsible’ for the ncp petition. In order to quantify the dependence of the ncp petition on the cp petitions, a dependency factor is defined.

Let p_{ij} be an ncp petition, i.e. $(i, j) \in N$. By Lemma 2, the span corresponding to this petition nests at least one cp petition. Therefore, D_{ij} is defined as the set of all cp petitions nested by p_{ij} which implies that $\forall p_{ab} \in \mathbf{D}_{ij}, \mathbf{Z}_{ab} \subset \mathbf{Z}_{ij}$ and $(a, b) \in C_{\mathbf{S}}$.

Each element, p_{ab} , of \mathbf{D}_{ij} has associated with it a dependency factor,

$$\xi_{ij} = (p_{ij}/|\mathbf{D}_{ij}|), \tag{5}$$

where $|\mathbf{D}_{ij}|$ = cardinality of \mathbf{D}_{ij} .

The dependency factor, ξ , is essentially a fraction of the magnitude of the ncp petition assigned to each of the cp petitions it nests, i.e. the elements of \mathbf{D}_{ij} . As a result, the *net dependence* (μ_{ab}) on a cp petition, p_{ab} is the sum of all the dependency factors associated with it and is given by:

$$\mu_{ab} = \sum_{i=1}^k m_i \xi_i, \tag{6}$$

where

k is the number of ncp petitions in the sequence,

$$m_i = \begin{cases} 1 & \text{if } \mathbf{Z}_{ab} \subset (\text{span of } i\text{th ncp petition}) \\ 0 & \text{otherwise,} \end{cases}$$

ξ_i is the dependency factor corresponding to the i th ncp petition.

5.2. Bottleneck determination

A weak point or *bottleneck* is typically a condition, situation, process or an object that retards free flow or progress. By this definition, every quotient that is involved in a petition is a bottleneck because it bears a portion of the responsibility for the existence of the petition. But, using the petitions themselves as the sole indicators of bottlenecks is not sufficient. Hence, some quotient based metrics are defined.

Inflow (F_i)—sum of all the cp petitions magnitudes associated with the i th quotient such that:

$$F_i = \sum_{n=1}^k p_{ni} \quad (p_{ni} \in \mathbf{C}_S), \tag{7}$$

where k is the number of cp petitions in sequence $\mathbf{S} = |\mathbf{C}_S|$.

Outflow (E_i)—sum of all the cp petition magnitudes associated with the i th quotient such that:

$$E_i = \sum_{n=1}^k p_{in} \quad (p_{in} \in \mathbf{C}_S), \tag{8}$$

where k is the number of cp petitions in sequence $\mathbf{S} = |\mathbf{C}_S|$.

Net Flow (NF_i)—sum of the inflow and outflow at the quotient i ,

$$NF_i = |F_i| + |E_i|. \tag{9}$$

Inflow intensity (f_{ij})—of a petition, p_{ij} , is defined as the ratio of its magnitude to the net inflow (F_j) of quotient j :

$$f_{ij} = \frac{p_{ij}}{E_j}. \tag{10}$$

Outflow intensity (e_{ij})—of the petition, p_{ij} , is defined as the ratio of its magnitude with respect to the net outflow (E_i) of quotient i :

$$e_{ij} = \frac{P_{ij}}{E_i} \tag{11}$$

5.2.1. *Critical bottlenecks*

The critical bottlenecks in this scenario are the smallest group of quotients, which by virtue of their position in the sequence and their precedence relations account for all the cp petitions in the sequence. So, $\forall p_{ij} \in C_s$, either i or $j \in \varphi$, where φ is the set of critical bottleneck quotients. The critical bottleneck set can be obtained by successively identifying the quotients with the largest associated net flow.

6. **Case study**

The configuration effects are a function of the topological relationships between the quotients rather than their geometric features. Therefore, it is very convenient to use block assemblies to test the model as they can be used to represent the topological relationships of any ‘real world’ assembly.

In this section, such a case is presented to validate the CV model. Consider the planar block assembly, A , shown in figure 5. The constraints are that the quotients can only be removed along the $X+$, $X-$ and $Y+$ directions, without changing the position of the assembly. The S_{bvp} for this assembly is determined to be $8 < 6 < 7 < 2 < 4 < 1 < 5 < 9 < 3$. The petition diagram for S_{bvp} is shown in figure 6.

The net petition magnitude, $P_{bvp} = 15.172$. This indicates that S_{bvp} departs significantly from S_{dvp} and the design requires modification to become suitable for disassembly. The total petition poset, T_{bvp} , is obtained and the magnitudes of the petitions are shown below in a matrix form:

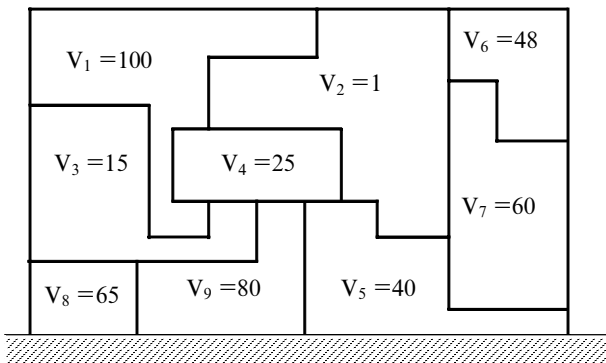


Figure 5. Assembly for a case study.

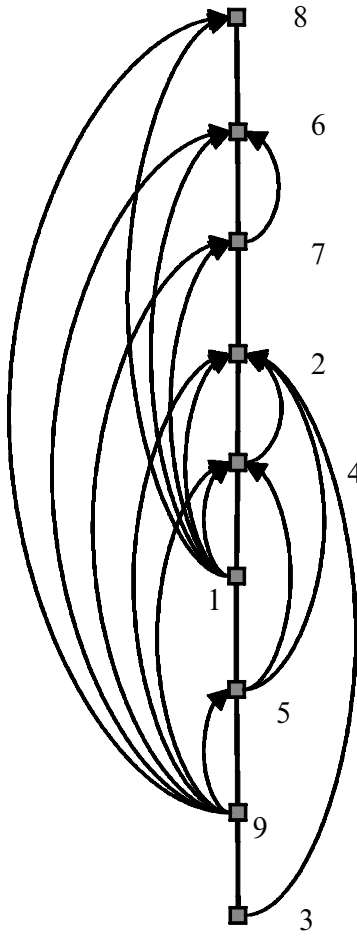


Figure 6. Petition diagram for S_{bvp} of an example assembly.

$$\mathbf{T}_{bvp} = \begin{bmatrix} X & 2.72 & 0 & 2.06 & 0 & 1.43 & 1.09 & 0.96 & 0 \\ 0 & X & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.16 & X & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.31 & 0 & X & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.58 & 0 & 0.22 & X & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & X & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.22 & X & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & X & 0 \\ 0 & 1.77 & 0 & 1.23 & 0.89 & 0.72 & 0.45 & 0.33 & X \end{bmatrix},$$

where $(\text{element})_{ij} = p_{ij} (\forall i, j \leq 9)$.

The cp petition poset, C_{bvp} is obtained next and the petition diagram for C_{bvp} is shown in figure 7.

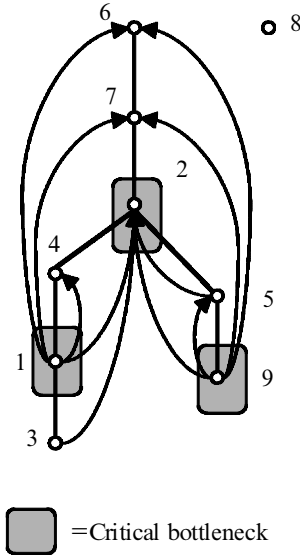


Figure 7. Petition diagram for the cp petition poset, C_{bvp} .

The magnitude of the cp petitions and the different quotient metrics are shown below in the matrix form:

$$C_{bvp} = \begin{bmatrix} X & 2.72 & 0 & 2.06 & 0 & 1.43 & 1.09 & 0 & 0 & \dots & 7.30 \\ 0 & X & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0.16 & X & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0.16 \\ 0 & 0.31 & 0 & X & 0 & 0 & 0 & 0 & 0 & \dots & 0.31 \\ 0 & 0.58 & 0 & 0 & X & 0 & 0 & 0 & 0 & \dots & 0.58 \\ 0 & 0 & 0 & 0 & 0 & X & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.22 & X & 0 & 0 & \dots & 0.22 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & X & 0 & \dots & 0 \\ 0 & 1.77 & 0 & 0 & 0.89 & 0.72 & 0.45 & 0 & X & \dots & 3.84 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 5.55 & 0 & 2.06 & 0.89 & 2.36 & 1.54 & 0 & 0 & \dots & 12.40 \end{bmatrix},$$

where

- (element) $_{ij}$ is the cp petition from quotient i to quotient j ($\forall i, j \leq 9$)
- (element) $_{lj}$ is the inflow for quotient j , F_j where $l = 10$,
- (element) $_{il}$ is the outflow for quotient i , E_i where $l = 10$.

The last row of the matrix gives the inflow (F_i) of each quotient obtained by adding all the elements in column i . The last column of the matrix gives the outflow (E_i) of each quotient obtained by adding all the elements in row i . The net flow ($NF_i = F_i + E_i$) for the quotients is:

$$NF = [7.30 \quad 5.55 \quad 0.16 \quad 2.37 \quad 1.48 \quad 2.36 \quad 1.76 \quad 0 \quad 3.84].$$

The ncp petitions in the assembly are $\{p_{98}, p_{18}, p_{94}, p_{54}\}$. The nesting pattern of the ncp petitions is shown in figure 8. The convention used to draw this nesting pattern (as described below) is similar to that used for the Hasse diagrams.

- All petitions are represented by triangles.
- Petition p_{ij} nests p_{ab} if $Z_{ab} \subset Z_{ij}$.
- If p_{ij} nests p_{ab} then p_{ij} is displayed higher than p_{ab} .
- A straight line connects p_{ij} and p_{ab} , if p_{ij} nests p_{ab} and there exists no other petition p_{cd} such that $Z_{ij} \subset Z_{cd} \subset Z_{ab}$.

Based on the nesting pattern the dependency factors for the ncp petitions are:

$$\xi = \{\xi_{98} = 0.03, \xi_{18} = 0.16, \xi_{94} = 0.615, \xi_{54} = 0.22\}.$$

Using C_S the critical bottlenecks are determined and are given by $\varphi = \{1, 9, 2, 7\}$. The extent of importance of each bottleneck can be determined using the net flow associated with the bottlenecks (without over counting). Figure 9 shows the relative contribution of each of these bottlenecks to the petition magnitude of the cp petition poset, C_S . Also shown is the net dependence (μ) associated with the bottleneck, i.e. the sum of μ for each of the cp petitions associated with the bottleneck.

The net dependence (μ) for the cp petitions is shown below in matrix form:

$$\text{Net dependence } (\mu) = \begin{bmatrix} 0 & 0.19 & 0 & 1.025 & 0 & 0.19 & 0.19 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.19 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.03 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.19 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.03 & 0 & 0 & 0.645 & 0.03 & 0.03 & 0 & 0 \end{bmatrix},$$

where μ_{ij} = net dependence on $p_{ij} (\forall i, j \leq 9)$.

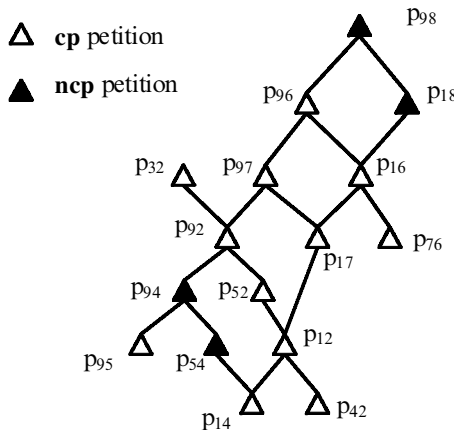


Figure 8. Nesting pattern of cp and ncp petitions.

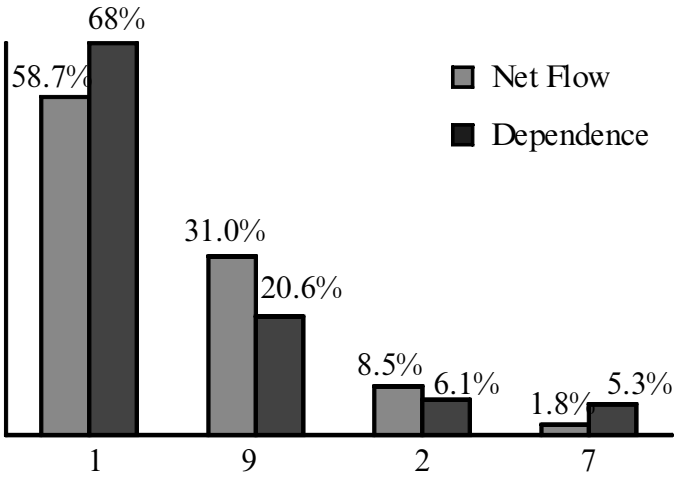


Figure 9. Relative contribution of critical bottlenecks to the net flow and dependence.

The inflow and outflow intensities for the bottlenecks 1, 9 and 2 are shown in table 1.

6.1. Discussion

The results of the analysis show that quotient 1 and its associated relations with {2, 4, 6, 7} constitute the most significant bottleneck. From figure 5, it can be seen that quotient 1 has a very high value and the delay in its early removal is responsible for its status as a critical bottleneck. The relations of quotient 9 with {2, 5, 6, 7} responsible for the second bottleneck are similar to those involved with quotient 1. The delayed removal of the high-valued quotient 9 is again the cause for this bottleneck. This can be inferred from the relative magnitude of the inflow and out-

| C | Inflow intensity | | Outflow intensity | |
|---|------------------|--------|-------------------|-------|
| 1 | | | p ₁₂ | 37.26 |
| | | | p ₁₄ | 28.22 |
| | | | p ₁₆ | 19.59 |
| | | | p ₁₇ | 14.93 |
| 9 | | | p ₉₂ | 46.21 |
| | | | p ₉₅ | 23.24 |
| | | | p ₉₆ | 18.80 |
| | | | p ₉₇ | 11.75 |
| 2 | p ₃₂ | 15.238 | | |
| | p ₄₂ | 29.524 | | |
| | p ₅₂ | 55.238 | | |
| 7 | p ₇₆ | 100% | p ₇₆ | 100% |

C, critical bottlenecks.

Table 1. Inflow and outflow intensities of critical bottlenecks.

flow of these quotients. Bottlenecks having a large inflow with respect to outflow would imply a low-valued quotient inappropriately located in the configuration. Similarly, a bottleneck having a large outflow as compared with the inflow would imply a large-valued quotient removed late in the sequence.

The petitions associated with the critical bottlenecks 1 and 9 also bear the maximum responsibility for the existence of the ncp petitions as can be seen in figure 9. From the nesting pattern in figure 8, the cp petitions responsible for the ncp petitions can be inferred to some extent. The ncp petitions, p_{94} and p_{54} arise out of the subsequence $\dots < 4 < 1 < 5 < 9 < \dots$ instead of possibly $\dots < 5 < 9 < 4 < 1 < \dots$. The reason for the former sequence being present in the Sbv_p is due to the effect of delaying the removal of the high-valued quotient 1. Therefore the ordering $4 \notin 1$ is the basic reason for p_{94} and p_{54} . However, in the case of p_{98} and p_{18} the causal ordering involves most of the sequence.

The CV model only provides an evaluation and it is left to the designer's ingenuity to overcome these bottlenecks depending on the peculiarities of the design and the nature of the design conflicts involved. Though the bottlenecks are ranked in the order of their importance, the choice of bottleneck to be modified and the extent to which its associated relations are to be modified ultimately depends on the designer. In general, a candidate for design modification would be the critical bottleneck involving the largest net flow and having a large net dependence. However, the quotient that needs to bear the modification to undo these critical relations need not be the critical bottleneck or the actual quotients related to it. In the present example the options could be to focus on the elimination of the high-intensity relations such as between quotients 1 and 2 and between 9 and 2 or to eliminate a significant number of cp petitions. By making changes to quotients 1 and 9 (figure 10), the new BVP obtained is $2 < 4 < 5 < 3 < 7 < 6 < 9 < 8 < 1$ having net petition magnitude of only $P8 = 0.685$.

The model implicitly satisfies other typical disassembly guidelines such as minimize number of parts and fasteners. This follows from the CV model that a product

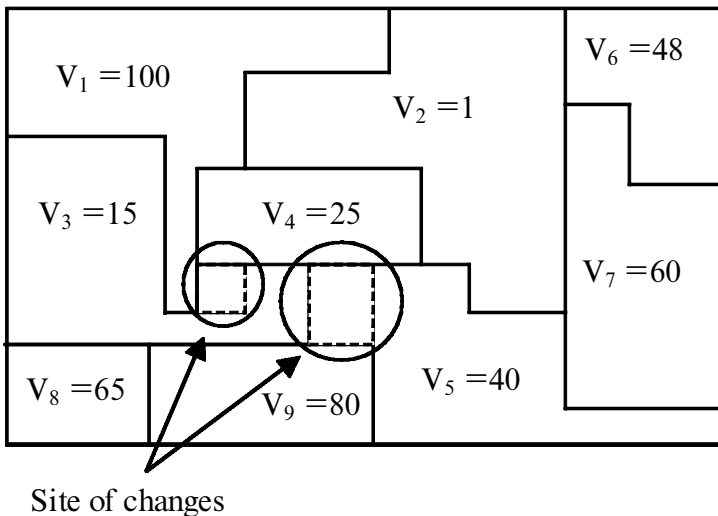


Figure 10. Modified design with an improved configuration for EOL disassembly

with fewer quotients will have lower chances of having a large number of petitions. As fasteners are considered to be quotients they would appear as very low-valued quotients. If they are located high in the sequence then they would attract a large number of petitions thus standing out as bottlenecks. Another guideline, maximize parallelism is predicted by the model. From a cp petition diagram it can be seen that greater the parallelism in the ordering, the lower the number of petitions possible.

An important issue in the application of the CV model is its suitability for computer application. The conceptualization of sequence spaces and VPs assumes knowledge of all the sequences. It is also easy to see that as the number of parts increases the number of possible sequences begins to increase very rapidly. However, the application of the CV model centers round the three reference VPs: DVP, BVP and OVP. Despite this, the CV model is dependent on the existence of an efficient sequencing algorithm. So all the problems associated with handling NP-complete problems are credited to the CV model too.

This paper has only addressed a subproblem, namely analysis to support active modification to make S_{bvp} closer to S_{dvp} . The problem of analysing the issues involved in making $S_{ovp} = S_{bvp}$ is something that still remains to be addressed.

7. Conclusion

A formal model called the Configuration Value model to evaluate and identify the bottlenecks of a given product configuration based on value and geometric ordering information is presented. Product configuration is an important aspect of disassembly that has been largely neglected by researchers. Though the configuration has a number of other effects on disassembly, its relation to the extraction of value is fundamental to the very objective of EOL disassembly.

The CV model is well suited for disassembly analysis to support DFD. First, by identifying a global (DVP) and local (BVP) reference point in the design space based on the value precedence relations, the model provides a clear direction for configuration redesign. Second, the model's low reliance on detailed product information favours its application during the early stages of the product design process when configuration changes are relatively easier to make. Third, the quantitative nature of the model makes it reproducible and allows easy computer implementation. Finally, the CV model provides a firm footing to the development of a formal foundation of the disassembly process that would greatly aid in design for disassembly.

Acknowledgements

The work was supported in part by the US National Science Foundation Grant #BES 9727136.

References

- ALTING, L. and LEGARTH, J. B., 1995, Life cycle engineering and design. *Annals of the CIRP*, **44**, 569–579.
- BALDWIN, D. F., ABELL, T. E., LUI, M. M., DE FAZIO, T. L. and WHITNEY, D. E., 1991, An integrated computer aid for generating and evaluating assembly sequences for mechanical products. *IEEE Transactions on Robotics and Automation*, **7**, 78–94.
- BOOTHROYD, G. and ALTING, L., 1992, Design for assembly and disassembly. *Annals of the CIRP*, **41**, 625–636.
- FELDMANN, K. and MEEDT, O., 1997, Determination and evaluation of the optimal end of Life strategy for products based on simulation of disassembly and recycling. In F.-L.

- Krause and G. Seliger (eds), *Life Cycle Networks* (London: Chapman & Hall), pp. 249–250.
- GUNGOR, A. and GUPTA, S. M., 1999, Issues in environmentally conscious manufacturing and product recovery: a survey. *Computers and Industrial Engineering*, **36**, 811–853.
- HARJULA, T., RAPOZA, B., KNIGHT, W. A. and BOOTHROYD, G., 1996, Design for disassembly and the environment. *Annals of the CIRP*, **45**, 109–114.
- HOMEM DE MELLO, L. S. and SANDERSON, A. C., 1991, AND/OR graph representation of assembly plans. *IEEE Transactions on Robotics and Automation*, **6**, 188–199.
- JOHNSON, M. R. and WANG, M. H., 1995, Planning product disassembly for material recovery. *International Journal of Production Research*, **43**, 3119–3142.
- JOHNSON, M. R. and WANG, M. H., 1998, Economical evaluation of disassembly operations for recycling, remanufacturing and reuse. *International Journal of Production Research*, **36**, 3227–3252.
- JOVANE, F., ALTING, L., ARMILLOTTA, A., EVERSHEIM, W., FELDMANN, K. and SELIGER, G., 1993, A key issue in product life cycle: disassembly. *Annals of the CIRP*, **42**, 651–658.
- JOVANE, F., SEMERARO, Q. and ARMILLOTTA, A., 1997, Computer-aided disassembly planning as a support to product redesign. In F.-L. Krause and G. Seliger (eds), *Life Cycle Networks* (London: Chapman & Hall), pp. 388–399.
- KEOLEIAN, G. A. and MENEREY, D., 1994, Sustainable development by design: review of life cycle design and other approaches. *Air and Waste*, **44**, 645–668.
- LAMBERT, A. J. D., 1997, Optimal disassembly of complex products. *International Journal of Production Research*, **35**, 2509–2523.
- LUTTROP, C., 1997, Disassembly structures — an approach to understand product structures from a separation viewpoint. In F.-L. Krause and G. Seliger (eds), *Life Cycle Networks* (London: Chapman & Hall), pp. 413–422.
- MARKS, M. D., EUBANKS, C. F. and ISHII, K., 1993, Life-cycle clumping of product designs for ownership and retirement. In T. K. Hight and L. A. Stauffer, (eds), *Proceedings ASME Design Theory and Methodology* (Albuquerque: ASME), pp. 83–90.
- NAVINCHANDRA, D., 1993, ReStar: a design tool for environmental recovery analysis. *Proceedings of the ICED 1993*, pp. 780–787.
- NEGGERS, J. and KIM, H. S., 1998, *Basic Posets* (Rivers Edge: World Scientific).
- PENEV, K. D. and DE RON, A. J., 1996, Determination of a disassembly strategy. *International Journal of Production Research*, **34**, 495–506.
- PNUELI, Y. and ZUSSMAN, E., 1997, Evaluating the end-of-life value of a product and improving it by redesign. *International Journal of Production Research*, **35**, 921–942.
- ROSEN, D., 1996, Design of modular product architectures in discrete design spaces subject to life cycle issues. *Proceedings of the 1996 ASME Design Engineering Technical Conferences*, 96- DETC/DAC-1485.
- SCHEURING, J. F., 1994, Product design for disassembly, MS thesis, Georgia Institute of Technology.
- SHYAMSUNDAR, N., ASHAI, Z. and GADH, R., 1998, Geometry-based metric formulation and methodology to support virtual design for disassembly. *Engineering Design and Automation*, **4**, 13–26.
- SIMON, M., 1991, Design for dismantling. *Professional Engineering*, **November**, 20–22.
- SIMON, M. and DOWIE, T., 1993, *Quantitative Assessment of Design Recyclability*. Report DDR/TR8 (Manchester: Manchester Metropolitan University) [<http://sun1.mpce.stu.mmu.ac.uk/pages/projects/dfc/pubs/dfc8/report8.htm>]
- WILSON, R. H. and LATOMBE, J., 1994, Geometric reasoning about mechanical assembly. *Artificial Intelligence*, **71**, 371–396.
- WITTENBERG, G., 1992, Life after death for consumer products, design for disassembling. *Assembly Automation*, **12**, 21–25.
- ZUSSMAN, E., KRIWET, A. and SELIGER, G., 1994, Disassembly-oriented assessment methodology to support design for recycling. *Annals of the CIRP*, **43**, 9–14.
- ZUSSMAN, E., ZHOU, M. C. and CAUDILL, R., 1998, Disassembly Petri net approach to modeling and planning disassembly processes of electronic products. *Proceedings of the IEEE International Symposium on Electronics and the Environment*, Oak Brook, IL, pp. 331–336.