# A Goal Programming Approach to the Remanufacturing Supply Chain Model

Elif Kongar and Surendra M. Gupta*
Laboratory for Responsible Manufacturing
334 SN, Department of MIME
Northeastern University
360 Huntington Avenue
Boston, MA  02115.

## ABSTRACT

The current trend of depletion of natural resources due to an ever-increasing number of consumer goods manufactured has led to an increase in the quantity of used and outdated products discarded. From an environmental point of view, it is not only desirable to disassemble, reuse, remanufacture and/or recycle the discarded products, in many cases it can also be economically justified. This situation being the motive, in recent years there have been several studies reported on disassembly, remanufacturing and/or recycling environments. Since "environmentally conscious manufacturing" is a relatively new concept that brings new costs and profits into consideration, its analysis cannot be provided by readily available techniques. This paper presents a quantitative methodology to determine the allowable tolerance limits of planned/unplanned inventory in a remanufacturing supply chain environment based on the decision-maker's unique preferences. To this end, an integer goal-programming model that provides a unique solution for the allowable inventory level is presented. The objective of the supply-chain model is to determine the number of a variety of components to be kept in the inventory while economically fulfilling the demand of a multitude of components, and yet have an environmentally benign policy of minimizing waste generation. A numerical example is presented to illustrate the methodology.

**Keywords:** Disassembly Process Plan, Inventory, Goal Programming, Recycling, Remanufacturing, Reuse.

## 1. INTRODUCTION

The awareness of the *environmentally conscious manufacturing* concept has led companies to look at their products and manufacturing processes in a whole new light. Companies have begun to look for ways of minimizing waste generation and preserve natural resources as well design their products for the environment[11]. The current trend among the customers is to favor "green" products, which has further motivated companies to produce *environmentally friendly* products. However, rapid technological improvements have induced a change in customers' behavior. Today's customers require new products even though the current ones are still capable of performing all the required tasks. This phenomenon, especially with the electronics products, has lead to incredibly short products' lives which, in turn, lead to environmental detriment because of the ensuing frequent turnover. Thus, the importance of end-of-life (EOL) processing of products cannot be overemphasized.

Among the desirable alternatives for EOL processing of products are remanufacturing, reusing and recycling. Although disposal and incineration are also possible EOL alternatives, they should be kept to a minimum. In order to remanufacture, reuse or recycle, often the product has to be disassembled first. *Disassembly* is the process of systematic removal of desirable constituents from the original assembly so that there is no impairment to any useful component. Disassembly can be *partial* (product not fully disassembled) or *complete* (product fully disassembled). In addition, disassembly can be *destructive* (focusing on materials rather than components recovery) or *non-destructive* (focusing on components rather than materials recovery). Since the process of disassembly is complex as well as labor-intensive, it tends to be very expensive. Thus, obtaining an efficient disassembly schedule is crucial for the economical justification of disassembly.

*Correspondence: e-mail: gupta@neu.edu; URL: http://www.coe.neu.edu/~smgupta/
Phone: (617)-373-4846;  Fax: (617)-373-2921

In this paper, we limit ourselves to partial and non-destructive disassembly in order to retrieve components and/or subassemblies to be reused in remanufacturing of a product. We present a preemptive integer goal programming model for the disassembly-to-order process so as to achieve various economical, physical and environmental goals that are simultaneously satisfied, based on the prescribed aspiration levels set forth by the decision-maker.

## 2. LITERATURE REVIEW

Several studies have recently emerged in the literature that address disassembly and environmentally conscious manufacturing. These studies can broadly be classified into three categories, viz., disassembly scheduling, disassembly process planning and mathematical modeling techniques to optimize the financial and environmental aspects of disassembly.

Gupta and Taleb[3] proposed an algorithm for scheduling the disassembly of a discrete, well-defined product structure. The principle surrounding the disassembly scheduling of a product into components is somewhat similar to Material Requirements Planning (MRP). The algorithm determines the disassembly schedule for the components such that the demands for those components are satisfied. In their subsequent papers, Taleb *et al.*[10] and Taleb and Gupta[9] improved the methodology to include components/materials commonality as well as the disassembly of multiple product structures. Recently, Veerakamolmal and Gupta[13, 14] proposed methods that provide solutions for component recovery planning. The authors determined the number and type of products to disassemble in order to satisfy the demand for a set of components while minimizing the disassembly and disposal costs. Lye *et al.*[7] proposed an algorithm to determine the minimum total servicing cost for a product network based on Floyd's shortest path algorithm. Although, the study takes the precedence relationships and faulty components into account and provides the user with multiple solution methods, it does not address the problem of component commonality and partial disassembly.

Veerakamolmal *et al.*[12] applied planning and sequencing techniques to create an efficient disassembly plan by taking advantage of the product modularity, which minimizes the total processing time and thus the cost of disassembly. Gungor and Gupta[1] presented a methodology for generating a near optimum disassembly sequence plan.

Several authors have applied mathematical programming in the area of disassembly and recycling. Isaacs and Gupta[6] investigated the impact of automobile design on disposal strategies by using goal programming to solve the problem. Hoshino *et al.*[4] used a goal programming model to analyze the profitability and recycling rates for manufacturing systems. See Moyer and Gupta[8], and Gungor and Gupta[2], for additional literature review on disassembly and product recovery.

## 3. PROBLEM STATEMENT AND FORMULATION

This paper examines a disassembly-to-order system so as to fulfill the demand for used components and/or subassemblies in order to remanufacture products using these disassembled items. A variety of products are obtained from the last users and/or collectors and disassembled to meet the demand. Any demanded item that is not used in the current period may be sold to the remanufacturer who stores it in inventory for use in the subsequent periods provided there is space to do so and the shelf lives of the items have not expired. Remaining items are either recycled or disposed of. Note that each product may have multiple components of the same type and might exhibit component commonality within and between product structures.

We present an integer goal-programming model to determine the number and type of products to be disassembled in order to fulfill a set of demand constraints while satisfying the predetermined cost/profit, physical and environmental goals. The cost functions included in the model are the total disassembly cost (*TPC*), the total recycling cost (*TRC*), the total inventory cost (*TIC*) and the total disposal cost (*TDC*). The revenue functions involved are the total resale revenue (*TRR*) and the total recycling revenue (*TCR*).

Since disassembly is a labor-intensive process, it is time consuming and costly. It is therefore imperative that we minimize the disassembly time (and hence cost). For starters, only the components demanded are disassembled unless other items have to be disassembled in order to reach the desired components. It is assumed that when a component is disassembled, only that component gets separated from the product and the residual partially disassembled product remains intact.

Figure 1 exhibits a partial product network diagram illustrating the disassembly process. Directed arcs represent the possible disassembly sequences. The shaded item *j* is the demanded component/subassembly. *j′* is the component/subassembly which is connected to the root and in order to reach *j*, *j′* and all the intermediate components or subassemblies (in this case all *j″*s) have to be disassembled. This disassembly path is the least cost path obtained by any shortest path algorithm. The cost to disassemble component or subassembly *j* can be derived as follows (refer to Table 1 for the nomenclature used in the paper):

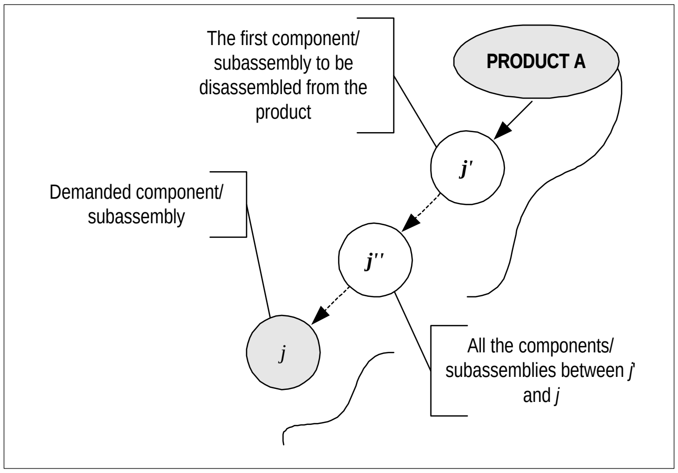
The first component/ subassembly to be disassembled from the product
PRODUCT A
j'
Demanded component/ subassembly
j''
j
All the components/ subassemblies between j' and j

**Figure 1.** Partial Product Network for Disassembly Process

The unit disassembly cost and the number of units that are disassembled directly effect the total cost of disassembling *j′* from product *i*. Thus,

$$TPC_{ij'} \quad = \qquad [PC_{ij'} \,.\, (a_{j'} \,.\, X_{ij'} + b_{j'} \,.\, R_{ij'} + g_{j'} \,.\, \text{¡}_{ij'} + d_{j'} \,.\, W_{ij'})]/(Q_{ij'}). \tag{1}$$

Similarly,

$$TPC_{j'j''} \quad = \qquad [PC_{j'j''} \,.\, (a_{j''} \,.\, X_{ij''} + b_{j''} \,.\, R_{ij''} + g_{j''} \,.\, \text{¡}_{ij''} + d_{j''} \,.\, W_{ij''})]/(Q_{j'j''}). \tag{2}$$

And

$$TPC_{j''j} \quad = \qquad [PC_{j'j''} \,.\, (a_{j} \,.\, X_{ij} + b_{j} \,.\, R_{ij} + g_{j} \,.\, \text{¡}_{ij} + d_{j} \,.\, W_{ij})]/(Q_{j''j}). \tag{3}$$

Thus, the total disassembly cost can be expressed as:

$$TPC_{ij} \quad = \qquad TPC_{ij'} + TPC_{j'j''} + TPC_{j''j} \tag{4}$$

Therefore, from equations (1), (2) and (3)

$$\begin{aligned} TPC_{ij} = {} & [PC_{ij'} \,.\, (a_{j'} \,.\, X_{ij'} + b_{j'} \,.\, R_{ij'} + g_{j'} \,.\, \text{¡}_{ij'} + d_{j'} \,.\, W_{ij'})]/(Q_{ij'}) + \\ & [PC_{j'j''} \,.\, (a_{j''} \,.\, X_{ij''} + b_{j''} \,.\, R_{ij''} + g_{j''} \,.\, \text{¡}_{ij''} + d_{j''} \,.\, W_{ij''})]/(Q_{j'j''}) + \\ & [PC_{j'j''} \,.\, (a_{j} \,.\, X_{ij} + b_{j} \,.\, R_{ij} + g_{j} \,.\, \text{¡}_{ij} + d_{j} \,.\, W_{ij})]/(Q_{j''j}). \end{aligned} \tag{5}$$

Thus, in general, the total cost of disassembling all the demanded components and/or subassemblies from all products:

$$TPC = \sum_{i} \sum_{j} TPC_{ij} \tag{6}$$

**Table 1.** Notation

| | |
|---|---|
| $AS$ | Total allowable storage space (unit$^2$); |
| $CR_j$ | Recycling (e.g. incentives, material) revenue per unit time ($/unit time); |
| $D_j$ | Vector representing the total demand for $P_j$ (unit); |
| $DC_{ij}$ | Components, subassemblies and products disposal cost; |
| $DR_j$ | Upper limit of recycling for component or subassembly *j*; |
| $h$ | Number for hard constraints in the model; |
| $h_i$ | Holding cost of product *i* ($/unit); |
| $h_{ij}$ | Holding cost of $P_{ij}$ ($/unit); |
| $i$ | Index used for products; |
| $I_i$ | Row vector of *i* one's; |
| $I_{ii}$ | Identity matrix of rank *i*; |
| $j$ | Index used for components and subassemblies; |
| $m$ | Total number of components in the problem space; |
| $ND$ | Number of disposed components, subassemblies and products (unit); |
| $NI$ | Number of components, subassemblies and products sent to inventory (unit); |
| $NRC$ | Number of recycled components and subassemblies (unit); |
| $OH_i$ | Number of products in the beginning of period (unit); |
| $OH_{ij}$ | Number of $P_j$s from all products in the beginning of period (unit); |
| $PC_{jj\text{¢}}$ | Disassembly cost ($/unit); |
| $PR$ | Total resale profit ($); |
| $PRC$ | Total recycling profit ($); |
| $RL_j$ | Upper limit for recycling (unit); |
| $q$ | Total number of products in the problem space (unit); |
| $Q_{ij}$ | Multiplicity matrix representing the number of each type of component and subassembly $P_j$ ; |
| $R_{ij}$ | Matrix of $P_j$ obtained from product *i* that will require recycle; |
| $RC_j$ | Recycling (e.g. tooling, labor) cost per unit time ($/unit time); |
| $RV_j$ | Resale value of component *j* ($/unit); |
| $S_i$ | Vector representing the supply of product *i* from all sources; |
| $SL_{ij}$ | Shelf life of $P_{ij}$ (unit time); |
| $TC_i$ | Cost of acquisition and transportation for product *i* ($/unit); |
| $TCR$ | Total recycling revenue ($) |
| $TDC$ | Total disposal cost ($); |
| $TIC$ | Total inventory cost ($); |
| $TOTAL$ | Total profit ($); |
| $TPC$ | Total processing cost ($); |
| $TRC$ | Total recycling cost ($); |
| $TRR$ | Total resale revenue ($); |
| $TS$ | Total storage space occupied by all components, subassemblies and products (unit$^2$); |
| $u$ | Integer goal programming objective value; |
| $V_i$ | Storage space needed for product *i*; |
| $V_{ij}$ | Storage space needed for $P_j$; |
| $W_{ij}$ | Matrix of $P_j$ obtained from product *i* that will require disposal; |
| $X_{ij}$ | Matrix of $P_j$ used to fulfill the total demand for components and subassemblies; |
| $Y_i$ | Vector representing the number of each of product *i* in the batch to be disassembled; |
| $i\ _{ij}$ | Matrix of $P_j$ obtained from product *i* that will require storage; |
| $a_j$ | Binary variable for component/subassembly *j* that is resold (1 if resold, 0 otherwise); |
| $b_j$ | Binary variable for component/subassembly *j* that is recycled (1 if recycled, 0 otherwise); |
| $g_j$ | Binary variable for component/subassembly *j* that is stored (1 if stored, 0 otherwise); |
| $q_j$ | Binary variable for component/subassembly *j* that is disposed of (1 if disposed, 0 otherwise). |

Note that the number of demanded components or subassemblies must be equal to the number of components retrieved in order to reach *j* (Figure 1). Similar reasoning also applies to the subsequent items (*j′* or *j″*). Thus:

$$(a_{j''} \,.\, X_{ij''} + b_{j''} \,.\, R_{ij''} + g_{j''} \,.\, ¡_{ij''} + d_{j''} \,.\, W_{ij''}) = (a_j \,.\, X_{ij} + b_j \,.\, R_{ij} + g_j \,.\, ¡_{ij} + d_j \,.\, W_{ij})/(Q_{j''j}). \qquad (7)$$

$$(a_{j'} \,.\, X_{ij'} + b_{j'} \,.\, R_{ij'} + g_{j'} \,.\, ¡_{ij'} + d_{j'} \,.\, W_{ij'}) = (a_{j''} \,.\, X_{ij''} + b_{j''} \,.\, R_{ij''} + g_{j''} \,.\, ¡_{ij''} + d_{j''} \,.\, W_{ij''})]/(Q_{j'j''}). \qquad (8)$$

### 3.1. Preemptive Integer Goal Programming Model

The preemptive integer goal programming (GP) is performed in several steps in order to fulfill the prescribed goals. The first goal always aims to satisfy the hard constraints. Subsequent steps try to address the other goals, one at a time, in order of their relative importance. Examples of the goals that we would be of interest to us include maximizing the recycling revenue (*TCR*), minimizing the total disposal cost (*TDC*), minimizing the total inventory cost (*TIC*) and maximizing the profit from resale (*PR*). Other goals of interest include minimizing the number of items stored (*NI*), maximizing the number of recycled items (*NRC*) and minimizing the number of disposed items (*ND*). Different scenarios could be performed based on the decision-maker's preferences. For each goal, the desire to overachieve (minimize $n_i$) or underachieve (minimize $g_i$), or satisfy the target value exactly (minimize $n_i + g_i$) is specified[5]. We formulate the problem described in the previous section as follows:

Find $\{X_{ij}\}$ so as to

$$\text{Lexicographically minimize } u = \{(\sum_{q+1}^{h} n_k + \sum_{1}^{h} g_k), (n_{h+1}, g_{h+1}), (n_{h+2}, g_{h+2})\} \qquad (9)$$

*Goal 1 Constraints (hard constraints):*

The number of products in the batch to be disassembled must not exceed the number of available products. Thus,

$$\{Y_i\} + \{¡_i\} + \{W_i\} + n_k - g_k = \{S_i\}; \text{ for all } i \text{ and } j \qquad (10)$$

The number and type of the components and/or subassemblies that are resold, recycled, stored and disposed of must be equal to the number and type of the components/subassemblies, which are retrieved from the products. Thus, by accounting for multiplicity:

$$(a_{ij} \,.\, \{X_{ij}\}) + (b_{ij} \,.\, \{R_{ij}\}) + (g_{ij} \,.\, \{¡_{ij}\}) + (q_{ij} \,.\, \{W_{ij}\}) + n_k - g_k = \{(Y_i \cdot I_{ii}) \cdot Q_{ij}\}; \qquad (11)$$
$$\text{for all } i \text{ and all } j \exists D_j > 0 \text{ and } P_j \in LS^S(Root_i)$$

The demand for every type of component must be met. Thus:

$$\{I_i \cdot X_{ij}\} + n_k - g_k = \{D_j\}; \text{ for all } j \exists D_j > 0 \text{ and } P_j \in LS^S(Root_i) \qquad (12)$$

The recycled components and/or subassemblies are subject to an upper limit. Thus:

$$\{I_i \cdot R_{ij}\} + n_k - g_k = \{DR_j\}; \text{ for all } i \text{ and } j \qquad (13)$$

The number of items in inventory is restricted by the allowable space for the storage. Thus,

$$TS + n_k - g_k = AS \qquad (14)$$

Where,

$$TS = \sum_{i} \sum_{\substack{P_j \in LS^S(Root_i) \\ j \exists D_j \geq 0 \\ j \exists h_j < DW_j \\ \text{and} \\ CR_j \geq RC_j}} (V_{ij} .\{\Upsilon_{ij}\}) + \sum_{i} \sum_{\substack{P_j \notin LS^S(Root_i) \\ j \exists D_j = 0 \\ j \exists h_j < DW_j \\ \text{and} \\ CR_j \geq RC_j}} (V_{ij} .\{\Upsilon_{ij}\}) + \sum_{i \in (OH_i - Y_i)} (V_i .\{\Upsilon_i\}) \qquad (15)$$

All the variables must be non-negative integers. Thus,

$$\{Y_i\}, \{X_{ij}\}, \{R_{ij}\}, \{\text{¡}_{ij}\}, \{W_{ij}\}, \{W_{ij}\}, \{n_k\}, \{g_k\} \geq 0 \text{ and integer; for all } i, j \text{ and } k. \qquad (16)$$

In addition to the hard constraints presented above, the model also includes additional constraints based on four more goals. In this paper we considered two different sets of goals. The first set of goals focuses on the cost and revenue functions and addresses *PRC*, *PR*, *TDC* and *TIC*. The second set of goals mostly stresses environmental functions and addresses *NRC*, *ND*, *NI* and *TOTAL*. These goals and associated objective functions are listed below.

| ***GP Set I*** | ***Goal Constraint*** | | | ***Associated Objective Function*** | |
|---|---|---|---|---|---|
| Priority 1 | $PRC + n_{h+1} + g_{h+1}$ | = | $PRC^*$ | min $n_{h+1}$ | (17) |
| Priority 2 | $PR + n_{h+2} + g_{h+2}$ | = | $PR^*$ | min $n_{h+2}$ | (18) |
| Priority 3 | $TDC + n_{h+3} + g_{h+3}$ | = | $TDC^*$ | min $g_{h+3}$ | (19) |
| Priority 4 | $TIC + n_{h+4} + g_{h+4}$ | = | $TIC^*$ | min $g_{h+4}$ | (20) |

| ***GP Set II*** | ***Goal Constraint*** | | | ***Associated Objective Function*** | |
|---|---|---|---|---|---|
| Priority 1 | $NRC + n_{h+1} + g_{h+1}$ | = | $NRC^*$ | min $n_{h+1}$ | (21) |
| Priority 2 | $ND + n_{h+2} + g_{h+2}$ | = | $ND^*$ | min $n_{h+2}$ | (22) |
| Priority 3 | $NI + n_{h+3} + g_{h+3}$ | = | $NI^*$ | min $g_{h+3}$ | (23) |
| Priority 4 | $TOTAL + n_{h+4} + g_{h+4}$ | = | $TOTAL^*$ | min $n_{h+4}$ | (24) |

*PRC* is the profit gained from the recycling process and can be expressed as:

$$PRC = TCR - TRC . \qquad (25)$$

*TCR* is the revenue of recycling process. This function is associated with the unit recycling revenue ($CR_j$) and the number of recycled units ($R_{ij}$).

$$TCR = \sum_{i} \sum_{\substack{j \exists D_j \geq 0 \\ P_j \in LS^S(Root_i) \\ and \\ CR_j \geq RC_j}} (CR_j \cdot \{R_{ij}\}) + \sum_{i} \sum_{\substack{j \exists D_j = 0 \\ P_j \notin LS^S(Root_i) \\ and \\ CR_j \geq RC_j}} (CR_j .\{R_{ij}\}) \qquad (26)$$

*TRC* is defined as the cost of recycling process. This cost function is effected by the unit recycling cost ($RC_j$) and the number of recycled units ($R_{ij}$). $RC_j$ is considered as the manpower and tooling cost for the operation. The items which are subject to recycling are selected among the components and/or subassemblies which are already disassembled and associated with higher or equal unit recycling revenue value compared with the unit recycling cost ($RC_j$). The second term on the right hand side of the equation stands for the partial subassemblies which are subject to recycling with or without destructive and complete disassembly. *TRC* can be expressed as:

$$TRC = \sum_{i} \sum_{\substack{j \exists D_j \geq 0 \\ P_j \in LS^S(Root_i) \\ and \\ CR_j \geq RC_j}} (RC_j \cdot \{R_{ij}\}) + \sum_{i} \sum_{\substack{j \exists D_j = 0 \\ P_j \notin LS^S(Root_i) \\ and \\ CR_j \geq RC_j}} (RC_j .\{R_{ij}\}) \qquad (27)$$

*PR* is the profit gained from reselling the demanded components and/or subassemblies and is defined as the difference between the resale revenue and the cost of disassembly, storage and disposal:

$$PR = TRR - TPC - TIC - TDC. \tag{28}$$

*TRR* is directly influenced by $RV_j$ and $TC_i$. $RV_j$ is the resale value of component *j*, and $TC_i$ is the cost per unit of acquiring and transporting product *i* from the distribution centers (or directly from the last users) to the disassembly facility. *TRR* is the difference between the resale revenue and the total cost of product acquisition, and can be formulated as:

$$TRR = \sum_{i} \sum_{\substack{j\exists D_j>0 \\ \text{and} \\ P_j\in LS^S(Root_i)}} (RV_j\cdot\{X_{ij}\}) \quad - \sum_{i} (TC_i\cdot\{Y_i\}) \,. \tag{29}$$

From equation (6), the total cost of disassembling all the demanded components and/or subassemblies from all products can be expressed as:

$$TPC = \sum_{i} \sum_{j} TPC_{ij} \tag{30}$$

*TIC* is the inventory cost based on the unit holding cost ($h_j$) and the number of units stored ($\Upsilon_{ij}$). Since the surplus products are also subject to storage, *TIC* can be formulated as:

$$TIC = \sum_{i} \sum_{\substack{P_j\in LS^S(Root_i) \\ j\exists D_j\geq 0 \\ j\exists h_j<DW_j \\ \text{and} \\ CR_j\geq RC_j}} (h_j\,.\{\Upsilon_{ij}\}) + \sum_{i} \sum_{\substack{P_j\notin LS^S(Root_i) \\ j\exists D_j=0 \\ j\exists h_j<DW_j \\ \text{and} \\ CR_j\geq RC_j}} (h_j\,.\{\Upsilon_{ij}\}) + \sum_{i\in(OH_i-Y_i)} (h_i\,.\{\Upsilon_i\}) \tag{31}$$

*TDC* is the cost of disposal involving the unit disposal cost (*DC*) (i.e., material loss, penalties etc.) and the number of units disposed ($W_{ij}$). Note that the surplus products are also subject to disposal. Thus, *TDC* can be formulated as:

$$TDC = \sum_{i} \sum_{\substack{P_j\in LS^S(Root_i) \\ j\exists D_j\geq 0 \\ j\exists h_j>DW_j \\ \text{and} \\ CR_j<RC_j}} (DW_j\,.\{W_{ij}\}) + \sum_{i} \sum_{\substack{P_j\notin LS^S(Root_i) \\ j\exists D_j=0 \\ j\exists h_j>DW_j \\ \text{and} \\ CR_j<RC_j}} (DW_j\,.\{W_{ij}\}) + \sum_{i\in(OH_i-Y_i)} (DW_i\,.\{W_i\}) \tag{32}$$

Number of recycled items can be expressed as:

$$NRC = \sum_{i} \sum_{\substack{j\exists D_j\geq 0 \\ P_j\in LS^S(Root_i) \\ \text{and} \\ CR_j\geq RC_j}} \{R_{ij}\} + \sum_{\text{i}} \sum_{\substack{j\exists D_j=0 \\ P_j\notin LS^S(Root_i) \\ \text{and} \\ CR_j\geq RC_j}} \{R_{ij}\} \tag{33}$$

Number of disposed items can be can be calculated as:

$$ND = \sum_{i} \sum_{\substack{P_j\in LS^S(Root_i) \\ j\exists D_j\geq 0 \\ j\exists h_j>DW_j \\ and \\ CR_j<RC_j}} \{W_{ij}\} + \sum_{i} \sum_{\substack{P_j\notin LS^S(Root_i) \\ j\exists D_j=0 \\ j\exists h_j>DW_j \\ and \\ CR_j<RC_j}} \{W_{ij}\} + \sum_{i\in(OH_i-Y_i)} \{W_i\} \tag{34}$$

Number of items sent to inventory can be expressed as:

$$NI = \sum_{i} \sum_{\substack{P_j \in LS^S(Root_i) \\ j\exists D_j \geq 0 \\ j\exists h_j < DW_j \\ \text{and} \\ CR_j \geq RC_j}} \{\Upsilon_{ij}\} + \sum_{i} \sum_{\substack{P_j \notin LS^S(Root_i) \\ j\exists D_j = 0 \\ j\exists h_j < DW_j \\ \text{and} \\ CR_j \geq RC_j}} \{\Upsilon_{ij}\} + \sum_{i \in (OH_i - Y_i)} \{\Upsilon_i\} \tag{35}$$

*TOTAL* is the total profit gained from the whole disassembly process and can be expressed as the sum of two profit functions as in below:

$$TOTAL = PRC + PR\,. \tag{36}$$

## 4. NUMERICAL EXAMPLE

We present a numerical example to foster better understanding of the model. Consider two products (A and B) as shown in Figure 2. Table 2 provides the data for the numerical example. Note that the component multiplicity variables $Q_{12}$ and $Q_{25}$ are shown in two columns. This represents the $Q_{ij}$ values for the common item in the same product structure. Additional data includes: $S_i$ = {100, 50}; $TC_i$ = {10, 15}; $AS$ = 500,000 unit$^2$. $DR_j$ = {300,350,300,300,400,200,200}; $h_i$ = {9, 8}; $V_i$ = {9, 13}. On hand inventory is given as $OH_3$ = 100, $OH_4$ = 200, $OH_7$ = 100 for components/subassemblies and $OH_{\text{I}}$ = 100 and $OH_{\text{II}}$ = 50 for products.

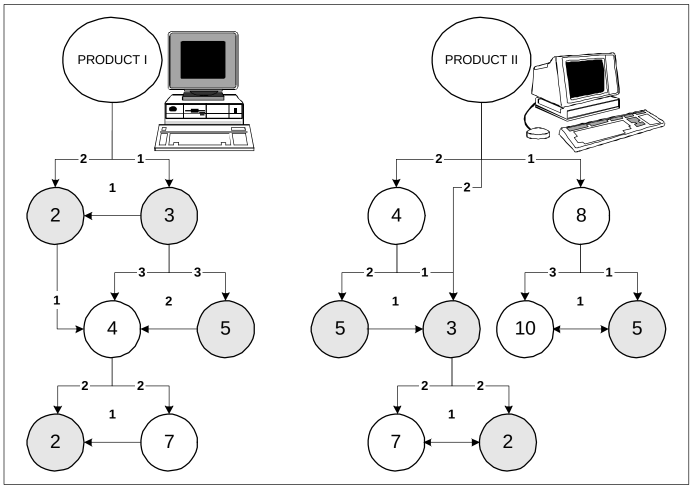

**Figure 2.** Original Product Networks and Disassembly Times

**Table 2.** Data for Numerical Example

| *j* | $Q_{1j}$ | | $Q_{2j}$ | | *Type* | $D_j$ | $RV_j$ | $RC_j$ | $CR_j$ | $DW_j$ | $h_j$ | $V_j$ | $SL_j$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 1 | 6 | 2 | | Component | 500 | 6 | 1 | 3 | 2 | 3 | 2 | 1 |
| 3 | 3 | | 2 | | Subassembly | 100 | 7 | 2 | 1 | 3 | 1 | 2 | 2 |
| 4 | 3 | | 1 | | Subassembly | | 4 | 3 | 1 | 3 | 3 | 3 | 1 |
| 5 | 6 | | 1 | 4 | Component | 100 | 5 | 1 | 4 | 2 | 2 | 2 | 2 |
| 7 | 3 | | 2 | | Component | | 4 | 2 | 1 | 3 | 1 | 1 | 1 |
| 8 | - | | 2 | | Subassembly | | 5 | 2 | 3 | 2 | 3 | 1 | 1 |
| 10 | - | | 2 | | Component | | 3 | 2 | 3 | 1 | 3 | 1 | 1 |

For the numerical example, the binary variables are defined as follows:

$$a_j = \begin{cases} 1, \text{ if } D_j > 0 \\ 0, \text{ elsewhere} \end{cases} \quad b_j = \begin{cases} 1, \text{ if } CR_j \geq RC_j \\ 0, \text{ elsewhere} \end{cases} \quad g_j = \begin{cases} 1, \text{ if } h_j < DC_j \text{ and } CR_j < RC_j \\ 0, \text{ elsewhere} \end{cases} \quad q_j = \begin{cases} 1, \text{ if } h_j > DC_j \text{ and } CR_j < RC_j \\ 0, \text{ elsewhere} \end{cases}$$

In order to benchmark the values for the GP model, the problem is first solved as a linear program (LP), the results of which are presented in Table 3.

**Table 3.** Initial Results of Linear Programming Model

| ***Function*** | ***Variable*** | ***Value*** |
|---|---|---|
| Total Profit | *TOTAL* | 1945 |
| Total Resale Profit | *PR* | 1969 |
| Total Recycling Profit | *PRC* | 376 |
| Total Inventory Cost | *TIC* | 402 |
| Total Resale Revenue | *TRR* | 3480 |
| Total Recycling Cost | *TRC* | 1527 |
| Total Recycling Revenue | *TCR* | 1149 |
| Total Processing Cost | *TPC* | 1282 |
| Total Disposal Cost | *TDC* | 228 |
| Total Space Occupied | *TS* | 654 |
| Number of Recycled Items | *NRC* | 1304 |
| Number of Disposed Items | *ND* | 32 |
| Number of Stored Items | *NI* | 52 |

The LP results also suggested that 72 units of Product I and no units of Product II should be disassembled to satisfy the demand.

Table 4 presents the results for various scenarios of the Goal Programming model for Set I. Note that, for comparison purposes, the numbers in parentheses represent the values obtained in the current step for the goal, which is satisfied in the next step.

**Table 4.** Results for Various Scenarios for Set I.

| **#** | **Step 1*** | **Step 2** | **Step 3** | **Step 4** | **Step 5** | ***TOTAL*** | ***NI*** | ***NRC*** |
|---|---|---|---|---|---|---|---|---|
| 1 | HC | *PRC* = 1180<br>(*PR* = 1020) | *PR* = 1513.33<br>(*TDC* = 1406.67) | *TDC* = 1406.67<br>(*TIC* = 1784) | *TIC* = 1180 | 1513.33 | 810 | 1380 |
| 2 | HC | *PRC* = 1180<br>(*PR* = 1020) | *PR* = 1513.33<br>(*TIC* = 2116) | *TIC* = 1180<br>(*TDC* = 1406.67) | *TDC* = 1406.67 | 1513.33 | 810 | 1380 |
| 3 | HC | *PRC* = 1180<br>(*TIC* = 2200) | *TIC* = 1566<br>(*TDC* =1406.67) | *TDC* = 1406.67<br>(*PR* = 139.33) | *PR* = 1513.333 | 1127.33 | 1003 | 1187 |
| 4 | HC | *PRC* = 1180<br>(*TDC* = 1460) | *TDC* = 1406.67<br>(*TIC* =1290) | *TIC* = 780<br>(*PR* =1.33) | *PR* = 9.33 | 429.33 | 760 | 848 |
| 5 | HC | *TIC* = 2<br>(*PRC* = 2) | *PRC* = 602<br>(*PR* = 990) | *PR* = 1020<br>(*TDC* = 1460) | *TDC* = 1458.33 | 1616.67 | 2 | 1475 |
| 6 | HC | *TIC* = 2<br>(*PRC* = 2) | *PRC* = 602<br>(*TDC* = 1485) | *TDC* = 1458.33<br>(*PR* = 602) | *PR* = 1016.67 | 1616.67 | 2 | 1475 |
| 7 | HC | *TIC* = 2<br>(*PRC* = 2) | *PRC* = 602<br>(*PR* = 990) | *PR* = 1020<br>(*TDC* = 1460) | *TDC* = 1460 | 1620 | 2 | 1480 |
| 8 | HC | *TIC* = 2<br>(*PR* = 0) | *PR* = 1469.33<br>(*TDC* = 1282.67) | *TDC* = 1282.67<br>(*PRC* = 0) | *PRC* = 378 | 1845.33 | 2 | 1304 |

*Hard constraints are satisfied

Similarly, Table 4 presents the results for various scenarios of the Goal Programming model for Set II.

**Table 5.** Results for Various Scenarios for Set II.

| # | Step 1* | Step 2 | Step 3 | Step 4 | Step 5 | ***PRC*** | ***PR*** | ***TIC*** |
|---|---|---|---|---|---|---|---|---|
| 1 | HC | *NRC* = 2440<br>(*ND* = 180) | *ND* = 180<br>(*NI* = 410) | *NI* = 361<br>(*TOTAL* = 460) | *TOTAL* = 462 | 0 | 830 | 368 |
| 2 | HC | *NRC* = 2440<br>(*NI* = 360) | *NI* = 360<br>(*ND* = 180) | *ND* = 180<br>(*TOTAL* = 460) | *TOTAL* = 460 | 0 | 830 | 360 |
| 3 | HC | *NRC* = 2440<br>(*TOTAL* = 460) | *TOTAL* = 520<br>(*ND* = 180) | *ND* = 180<br>(*NI* = 390) | *NI* = 390 | 0 | 1120 | 600 |
| 4 | HC | *TOTAL* = 1945.33<br>(*NRC* = 1304) | *NRC* = 1304<br>(*NI* = 50) | *NI* = 50<br>(*ND* = 32) | *ND* = 32 | 376 | 1969.33 | 400 |
| 5 | HC | *TOTAL* = 1945.33<br>(*NRC*= 1304) | *NRC* = 1304<br>(*ND* = 32) | *ND* = 32<br>(*NI* = 220) | *NI* = 132 | 456 | 1969.33 | 480 |
| 6 | HC | *TOTAL* = 1945.33<br>(*NI* = 50) | *NI* = 40<br>(*NRC* = 1304) | *NRC* = 1304<br>(*ND* = 32) | *ND* = 32 | 376 | 1969.33 | 400 |
| 7 | HC | *NI* = 2<br>(*NRC* = 869) | *NRC* = 1304<br>(*ND* = 374) | *ND* = 32<br>(*TOTAL* = 1262) | *TOTAL* = 1849.33 | 376 | 1489.33 | 16 |
| 8 | HC | *NI* = 2<br>(*TOTAL* = 0) | *TOTAL* = 1848.33<br>(*ND* = 32) | *ND* = 32<br>(*NRC* = 1304) | *NRC* = 1304 | 376 | 1489.33 | 16 |

*Hard constraints are satisfied

As it is shown in Table 4 and Table 5, different values for various performance measurements can be obtained by using goal programming. Depending on the decision-maker's preferences, a most suitable scenario can be chosen among these options. For example, scenario 8 in Set I can be considered for high profit ($1,845.33), low inventory cost ($2) and limited storage space (2 units) conditions. This goal, while recycling 1,304 units, provides $1,469.33 resale profit and $378 recycling profit value. The disposal cost is obtained as $1,282.67.

### 4.1. Analysis of Results

Since goal programming is a useful tool in obtaining feasible solutions under different circumstances, it provides the decision-maker with a number of choices. The various results obtained from the two different sets of integer goal programming scenarios are plotted in Figures 3 and 4.

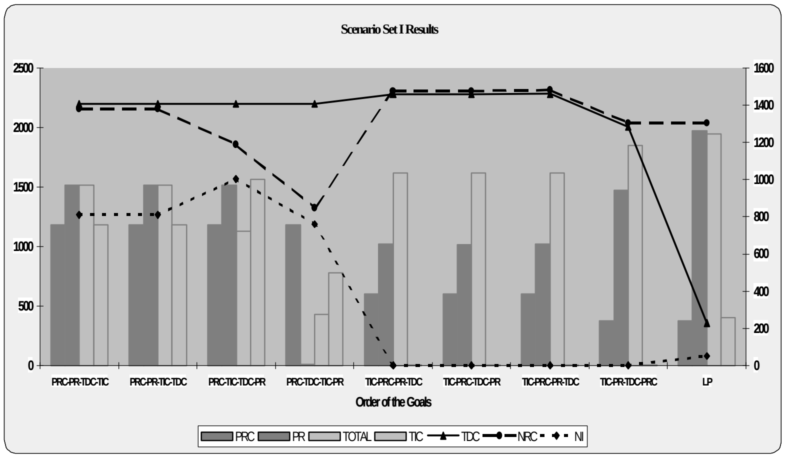

**Figure 3.** Variation of Performance Measures Depending on the Order of Goals in Set I

Figure 3 exhibits the goal programming results involving Set I. When the emphasis is on the economical factors rather than environmental factors, this approach is useful in obtaining the most desired disassembly process plan. For example, say, the company is interested in reducing the inventory cost because of a sudden budget limitation but also desires a higher profit from resale to protect its market share, scenario 8 should be selected as the disassembly plan.

Figure 4 exhibits the goal programming results involving Set II. Under the circumstances where environmental factors rather than economical factors are emphasized by the decision-maker, this approach is useful in obtaining the most desired disassembly process plan. For example, when it is crucial to decrease the amount of waste and a high total profit is also important, scenario six could be considered appropriate, providing a 40 units of storage, which is the smallest inventory level among other similar options (scenarios 4 and 5).

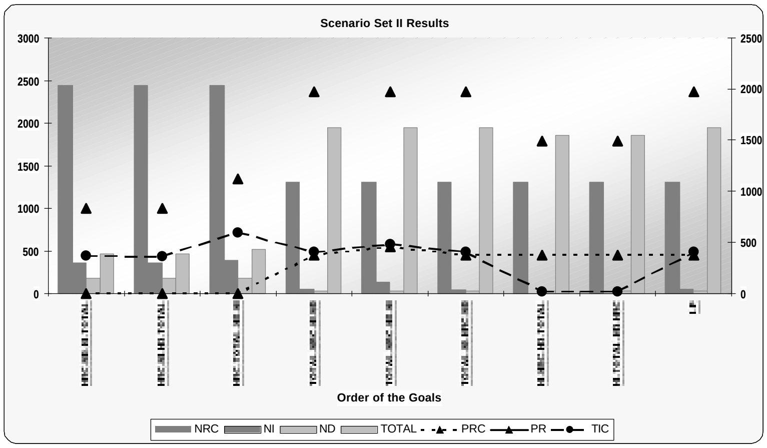

**Figure 4.** Variation of Performance Measures Depending on the Order of Goals in Set II

## 5. CONCLUSIONS

In this paper an integer goal programming model has been presented in order to determine the most desirable disassembly process plan while satisfying various environmental, financial and physical goals. The decision for selecting the most appropriate plan is left to the decision-maker's preference by providing him/her with various feasible options. In the environmentally conscious manufacturing environment it is no longer realistic to use a single objective function since the introduction of restrictive regulations makes the decision procedure more complicated and mostly multi-objective. A multi-objective decision criterion, which is more flexible to changes in decision criteria and governmental regulations, should be used. The model presented in this paper, while fulfilling an acceptable profit level is also capable of satisfying additional goals simultaneously. This goal programming approach is especially appropriate for decision-maker centered cases.

## REFERENCES

1. Gungor A, Gupta S. M., "Disassembly Sequence Planning for Complete Disassembly in Product Recovery", Proceedings of the 1998 Northeast Decision Sciences Inst Conf , 250-252, 1998.
2. Gungor A, Gupta S. M., "Issues in Environmentally Conscious Manufacturing and Product Recovery: A Survey", Computers and Ind Engineering. Vol. 36, No. 4, 811-853, 1999.
3. Gupta S. M., Taleb K. N., "Scheduling Disassembly", Int. J. Prod. Res., 32(8), 1857-1866, 1994.
4. Hoshino T, Yura K and Hitomi K, "Optimization Analysis for Recycle-oriented Manufacturing", Int J Prod Res, 33(8), 2069-2078, 1995.
5. Ignizio J.P., *Linear Programming in Single and Multiple Objective Systems*, Englewood Cliffs, NJ, 1982.
6. Isaacs J. A., Gupta S. M., "Economic Consequences of Increasing Polymer Content on U.S. Automobile Recycling Infrastructure", J. of Industrial Ecology. Vol. 1, No. 4, 19-33, 1997.
7. Lye S. W., Lee S. G., and Khoo M. K., "An Algorithm for Optimising the Servicing of Products with Constrained, Multiple Defects", Int. J. Prod. Res., 38(10), 2069-2185, 2000.
8. Moyer L, Gupta S. M., "Environmental Concerns and Recycling / Disassembly Efforts in the Electronics Industry", J of Electronics Manufacturing. Vol. 7, No. 1, 1-22, 1997.

9. Taleb K. N., Gupta S. M., “Disassembly of Multiple Product Structures, Computers and Industrial Engineering”, Vol. 32, No. 4, 949-961, 1997.
10. Taleb K. N., Gupta S. M., Brennan L, “Disassembly of Complex Products with Parts and Materials Commonality”, Production Planning and Control. Vol. 8, No. 3, 255-269, 1997.
11. Veerakamolmal, P. and Gupta, S. M., "Design for Disassembly, Reuse and Recycling", in *Green Electronics/ Green Bottom Line: Environmentally Responsible Engineering*, Edited by Lee Goldberg, Butterworth-Heinemann; (Newnes), Chapter 5, 69-82, ISBN: 0-7506-9993-0, 2000.
12. Veerakamolmal P, Gupta S. M., McLean, C. R., “Disassembly Process Planning, Proceedings of the Int. Conference on Engineering Design and Automation”, 162-165, 1997.
13. Veerakamolmal P, Gupta S. M., “Optimal Analysis of Lot Size Balancing for Multi-Products Selective Disassembly”, Int J of Flexible Automation and Integrated Manufacturing. Vol. 6, No. 3/4, 245-269, 1998.
14. Veerakamolmal P, Gupta S. M., “Planning Components Recovery from Multiple Products”, Proceedings of the 1998 Northeast Decision Sciences Inst. Conf , 270-272, 1998.