11-1-2008

# Multi-object decision making for supplier selection in outsourcing

Pushpen K. Mohile

# MULTI-OBJECTIVE DECISION MAKING FOR SUPPLIER SELECTION IN OUTSOURCING


**Pushpen K. Mohile**

B.E. (Production Engineering)

University of Mumbai


Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in the Department of Industrial and Systems Engineering in the Kate Gleason College of Engineering of the Rochester Institute of Technology

November 2008

KATE GLEASON COLLEGE OF ENGINEERING
ROCHESTER INSTITUTE OF TECHNOLOGY
ROCHESTER, NEW YORK

## CERTFICATE OF APPROVAL

---

## MASTER OF SCIENCE DEGREE THESIS

---

The M.S. Degree thesis of Pushpen K. Mohile
has been examined and approved by the thesis
committee as satisfactory for the thesis
requirement for the Master of Science degree.

------------------------------------------------------
Dr. Sudhakar Paidy

------------------------------------------------------
Dr. Panchapakesan Venkataraman

------------------------------------------------------
Dr. Marcos Esterman

**Dedicated to**

**My Beloved Late Poppa**

*Kirit S. Mohile*

**My Mommy**

*Nilam K. Mohile*

**My Sister**

*Pallavi K. Mohile*

# ABSTRACT

A supply chain is the system of organizations, people, technology, activities, information, and resources involved in moving a product or service from supplier to customer. The main and basic challenges in the supply chain are to plan a strategy to manage the resources and meet the demands, to select the suppliers that will deliver the goods and services that are required to build the product, to manufacture the product, to deliver the product to the customers and to make an arrangement for return of the products through customers if there is any fault in the product for servicing. This thesis study concentrates on supplier selection problem. Recently, outsourcing has become the prime part of the company. The activities which are not core to the business or not feasible to manufacture in-house are being outsourced to suitable suppliers. The major hurdle in outsourcing is to select a suitable supplier. The right supplier will lead to the fulfillment of the company's needs and will help increase the financial stability as well as the reputation of the company in the market. The selection of suppliers depends on number of criteria and it is possible that one supplier satisfies some of the selected criteria, a company is looking in to, but not the remaining others and the other supplier may satisfy the other ones but not the first ones. The challenge is to optimize selection process based on critical criteria and select the best supplier(s). The supplier selection problems are multi-objective problems and no single methodology appears to be dominant in solving supplier selection problem. This thesis study has attempted to advance the art of supply chain management by developing a heuristic methodology "Integrated Evolutionary Goal Trade-off (IEGT) Method" which simplifies the task of supplier selection and reduces the tediousness as well as the degree of error by directly involving the Decision Maker into the selection process. The "IEGT" method is highly efficient and it implements the procedure or steps of the Posteriori Articulation method in which after the solutions are presented to the Decision Maker, it incorporates combination of different methods like Evolutionary Algorithm Method, Goal Programming Method and STEM Method to reach the final optimal supplier(s).

**ACKNOWLEDGEMENT**

The successful completion of this thesis work would not have been possible without the help and guidance from a number of individuals.

First and foremost, I would like to thank Dr. Sudhakar Paidy for his significant support and care during my tenure of thesis study under him. I would like to thank him for all his guidance and advice which helped me develop my skills and knowledge in the areas of Supply Chain and Operations Research.

My special thanks to my two committee members, Dr. Marcos Esterman and Dr. Panchapakesan Venkataraman for their helpful comments and encouragement.

I would also like to thank all the staff members of the Industrial and Systems Engineering Department for their direct or indirect support during my tenure of thesis study.

Finally, I would like to thank all my friends for their undue help in good and bad times throughout my tenure of thesis study.

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# CHAPTER 1

# INTRODUCTION

# 1. INTRODUCTION

A supply chain is the system of organizations, people, technology, activities, information, and resources involved in moving a product or service from supplier to customer (Wikipedia). Supply chains can be defined as

> *real world systems that transform raw materials and resources into end products that are consumed by the consumers. Supply chains encompass a series of steps that add value through time, place, and material transformation. Each manufacturer or distributor has some subset of the supply chain that it must manage and run profitably and efficiently to survive and grow.* (Pinto)

The main and basic challenges in the supply chain are to plan a strategy to manage the resources and meet the demands, to select the suppliers that will deliver the goods and services that are required to build the product, to manufacture the product, to deliver the product to the customers and to make an arrangement for return of the product for servicing through customers, if there is any fault in the product.

This thesis study concentrates mainly on the supplier selection problem. Selecting suitable suppliers for purchasing the raw materials and to delegate the non-core operations from internal production is an important part of the operation. Delegating the work from internal production to external entity specializing in the management of that operation is called outsourcing. Currently, outsourcing is the prime part of a company. Outsourcing is defined as purchasing ongoing services and parts from an outside company that a company currently provides, or most organizations normally provide for themselves (Wadhwa and Ravindran 3725). Outsourcing the activity also means that the work is distributed and hence the time-to-market the final product can be reduced. The challenge to the company is the selection of suppliers. The decision of selecting the right supplier is prone to errors. The right supplier is the one who will meet and complement the organization’s needs from its corporate culture to long-term

future needs. "Some suppliers that meet some selection criteria may fail in some other criteria" (Wadhwa and Ravindran 3726). For example, the supplier selected may meet the "price" criteria but the company might have to compensate on the quality of the product and lead time. Selection of suppliers depends on various different criteria. Some criteria are quantitative such as "price of the product," "lead-time for delivery," "transportation cost," etc., whereas some like "reputation of the supplier," "cultural barrier," "risk," etc., are qualitative. No single methodology appears to be dominant in solving the supplier selection problem. In this study multi-objective decision making methodologies are applied to select the suppliers by optimizing various criteria (objectives) and a heuristic methodology is developed to find a suitable solution (final supplier(s)).

The basics of the supply chain and the extended supply chain which includes the challenges faced in supply chain management are discussed in Chapter 2, "Supply Chain Management." Various areas of supply chain management and multi-objective optimization were reviewed and are presented in Chapter 3, "Literature Review." Following the literature search, a supplier selection problem was selected for the thesis and is stated in Chapter 4, "Problem Statement." Supplier selection depends on various criteria. Hence, criteria used for the supplier selection in the real world industries and by various authors, were studied and some of the important criteria were selected, as discussed in Chapter 5, "Criteria Selection & Justification."

In order to analyze all the criteria and select the right suppliers, multi-attribute decision making methods were studied and presented by an ad-hoc example in Chapter 6, "Multi-Attribute Decision Making" and multi-objective decision making methods such as weighted objective method, goal programming method, evolutionary programming method, and stem method, were studied and are discussed in detail in Chapter 7, "Multi-Objective Decision Making Methods and Examples." Multi-attribute decision making is not a part of this thesis study as it is such a vast topic of study. This thesis concentrates on multi-objective decision making methods. Using the terminology and the methods, a representative supplier selection problem is

synthesized and stated along with its data, objectives, and the constraints in Chapter 8, “A Representative Supplier Selection Problem.”

A heuristic multi-objective optimization technique for supplier selection is developed and presented in Chapter 9, “A Heuristic Multi-Objective Methodology for Supplier Selection Problem.” A summary of the entire thesis is presented in Chapter 10, “Conclusion” along with recommendations for areas of expansion to the study.

# CHAPTER 2

# SUPPLY CHAIN MANAGEMENT

# 2. SUPPLY CHAIN MANAGEMENT

## 2.1 Overview of Supply Chain Management

A supply chain is a series of links and shared processes that exist between suppliers and customers.

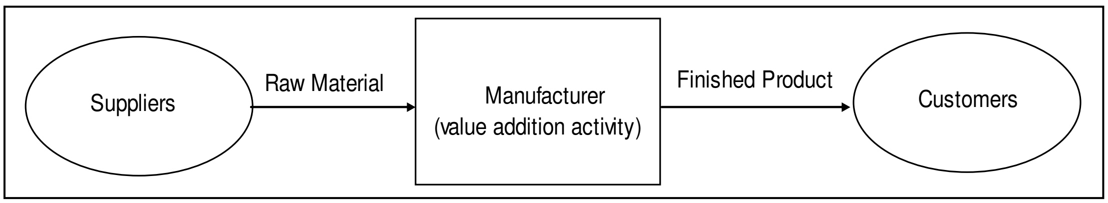


Figure
2.1: Basic Supply Chain Process

A supply chain is a network of facilities and distribution options that performs the tasks of procurement of materials, transformation of these materials into intermediate and finished products, and distribution of these finished products to the consumers (Ganeshan and Harrison).

Supply chain management is the act of optimizing all activities throughout the supply chain, so that products and services are supplied to the consumers in the right quantity, to the right location, at the right time and at optimal cost (Clarkston).

## 2.2 Challenges in Supply Chain Management

These links and processes involve all activities from acquisition of raw material to delivery of the finished goods to the end user / consumer. Raw material enters into a manufacturing organization via a supply system and is transformed into finished goods. The finished goods are then supplied to consumers through a distribution system. Generally, several companies are linked together in this process, each adding value to the product as it moves through the supply chain (Clarkston) (Figure 2.2).

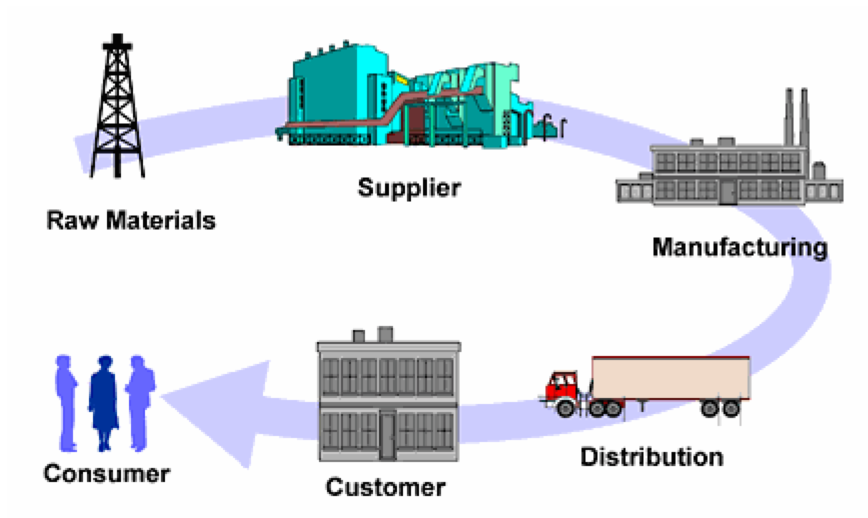


Figure 2.2: Series of links and shared processes between suppliers and customers

**Source:** Strahan, Bruce, and Art Van Bodegraven. *Logistics vs. The Supply Chain*. *The Progress Group: White Papers*. The Progress Group, Inc., 2004. Web. 10 Mar 2007 <http://theprogressgroup.com/publications/wp2_logs.html>.

Within an organization, if a product passes from one department to another department, the receiver department is an internal customer to the delivery department. On the other extreme, an inter-organizational structure may involve links

to many suppliers and consumers resulting in a complex extended supply chain (Figures 2.3).

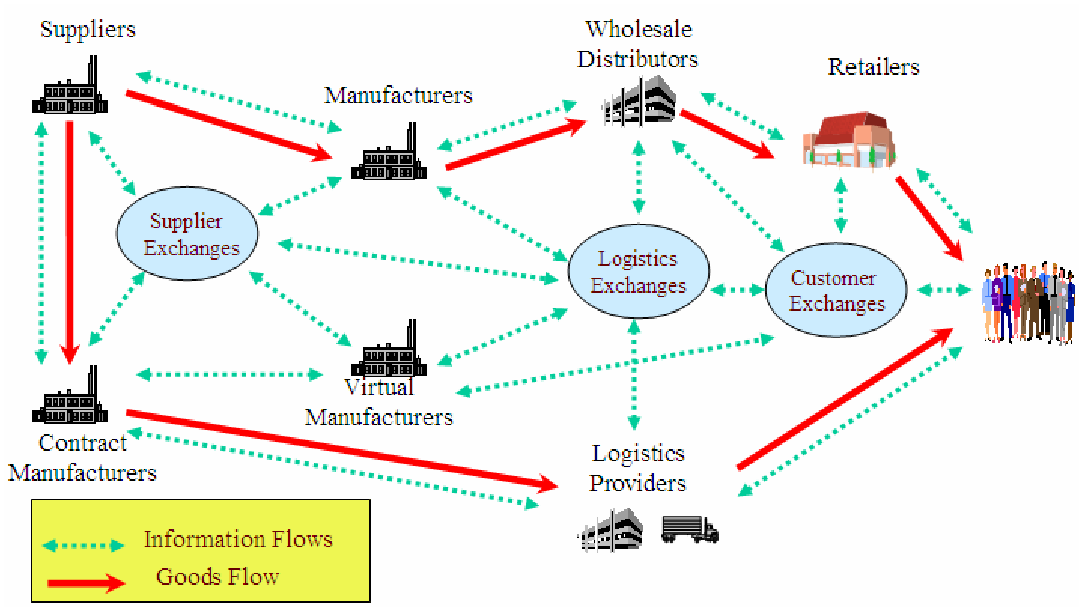


Figure 2.3: An Extended Supply Chain

**Source:** Bauer, Michael-CSC Consulting and AMR Research. "E-Business: The Strategic Impact on Supply Chain and Logistics." *e-Business:_The Best Document Search Engine!*. N.p. 2001. Web. 10 June 2008 <http://www.seeeach.com/doc/90560_e-Business:>.

The following are the five basic tasks for Supply Chain Management:

1. Plan: A strategy needs to be decided for managing resources, meeting demands and production.

2. Source: Choose suppliers that will deliver goods and services you need to create your own product, pricing scheme, delivery and payment processes.

3. Make: Manufacture or create the product.

4. Deliver: Coordinate receipt of orders from customers, develop a network of warehouses, and pick carriers to get products to customers and set-up an invoicing system to receive payments.

5. Return: Create a network to receive defective and excess products back from customers and support customers who have problems with the delivered products.

Challenges faced while managing the supply chain:

Traditional supply chain management assumed that information needs to be shared only with the next point of contact in a supply chain. But this resulted in a Bullwhip effect which means small changes in consumer demand can result in large variations in orders placed upstream (Figure 2.4).

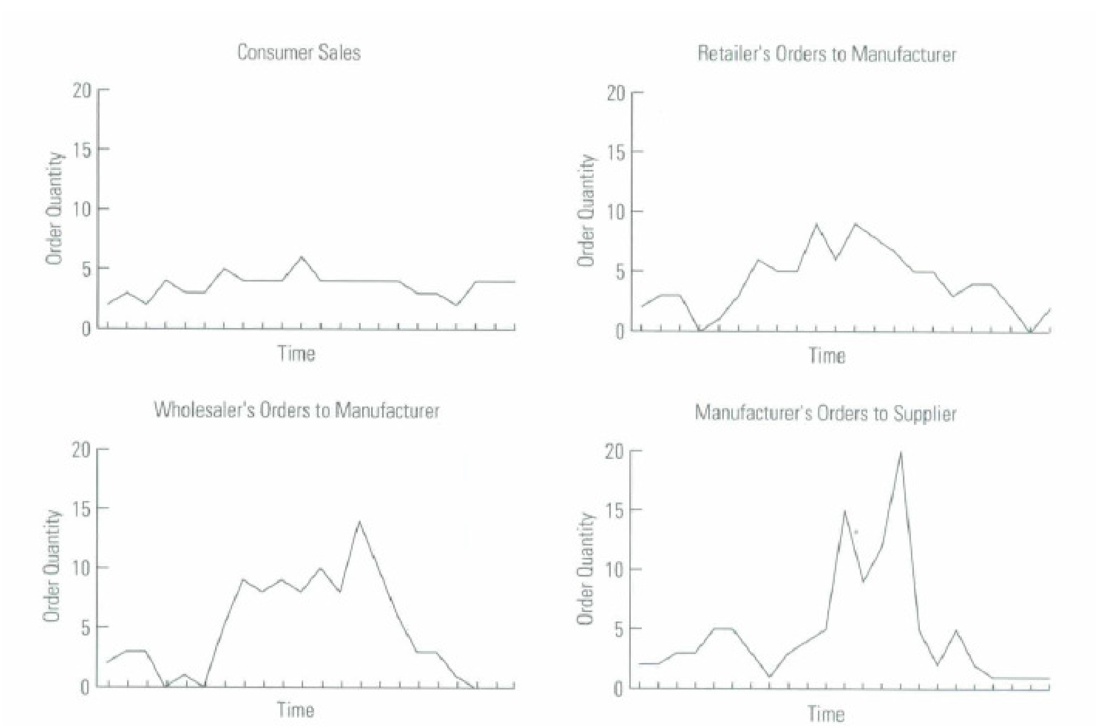


Figure 2.4: Bullwhip effect

**Source:** Lee, Hau L., V. Padmanabhan, and Seungjin Whang. "The Bullwhip Effect in Supply Chains." *Sloan Management Review* 38.3 (1997): 94. Print.

The **Bullwhip Effect** (or *Whiplash Effect*) is an observed phenomenon in forecast-driven distribution channels. Because customer demand is rarely perfectly stable, businesses must forecast demand, in order to properly position inventory and other resources. Forecasts are based on statistics, and they are rarely perfect or accurate. Because forecast errors are a given, companies often carry an inventory buffer called “safety stock.” Moving up the supply chain from end-consumer to raw materials supplier, each supply chain participant has greater observed variation in demand and thus greater need for safety stock. In periods of rising demand, down-stream participants will increase their orders. In periods of falling demand, orders will fall or stop in order to reduce inventory. The effect is that variations are amplified as one moves upstream, further from the customer, in the supply chain (Wikipedia). This can be avoided by keeping the transparency of the data among the whole supply chain.

## 2.3 Supplier Selection in Supply Chain Management

As can be seen from Figure 2.3, contract manufacturers and virtual manufacturers (suppliers) are also an important part of the supply chain system. Selecting suitable suppliers for purchasing the raw materials and delegating the non-core operations from internal production is an important part of the operation. Delegating the work from internal production to external entity specializing in the management of that operation is called outsourcing.

# CHAPTER 3

# LITERATURE REVIEW

# 3. LITERATURE REVIEW

## 3.1 Overview

Initially, the literature search focused on the general topic “Multi-Objective Optimization.” As mentioned by recent authors, “supply chain problems are complex and difficult to solve” (Pinto) and since multi-objective optimization methods are capable of solving such complex problems, the literature search later concentrated on “Application of Multi-Objective methods in Supply Chain area.” Through the wide literature review, it seemed that “Application of Multi-Objective methods in Supplier Selection” was in its infancy state. Hence, the literature search narrowed down to a topic, “Application of Multi-Objective methods in Supplier Selection.”

This thesis study concentrated on developing one of the methods to deal with the supplier selection problem as it is a vast topic and the research work done in this area is less.

### 3.2 Description

The literature search in this research study focused on two major areas; “multi-objective methods” and “supplier selection in outsourcing.” The diversity of problems solved using the multi-objective methodology provides a backdrop for addressing the most challenging problem of the supply chain management: Supplier Selection.

In many large organizations, millions of dollars are spent on outsourcing (Wadhwa and Ravindran 3725). The activities which are not core to the business or not feasible to manufacture in-house are being outsourced to suitable industries. According to a recent survey carried out by Accenture, 80% of the companies’ surveyed use some form of outsourcing and a majority of these companies are spending close to 45% of their total budget on outsourcing (Wadhwa and Ravindran 3725). The major hurdle in outsourcing is to select a suitable supplier. The selection of suppliers depends on the number of criteria. The challenge is to optimize the criteria and select the best supplier(s) out of that.

### 3.2.1 Multi-objective methods application in any field of study

Initially general applications of multi-objective methods were studied. There are numerous topics/ areas under which multi-objective optimization studies have been done.

1. Varshney and Rao in their paper "Multi-Objective Crop Planning" use the linear goal programming method to optimize three objectives "to maximize irrigated crop area, to maximize net benefits and to maximize crop production" with " total land water and crop area fertilizer" as its constraints and "hectares of land used per crop" as the decision variable.

2. Huang, Tian and Zuo in their paper "Multi-Objective Optimization of Three-Stage Spur Gear Reduction" use the interactive physical programming method to optimize three objectives "to minimize volume, to minimize surface fatigue and to maximize load carrying capacity" with "tooth bending fatigue failure, shaft torsional stress, face width interference and tooth number" as its constraints and "core hardness, face width, tooth numbers and diameter of the shaft" as the decision variables.

3. Oliveira, Volpi and Sanquetta in their paper "Multi-Objective programming in Brazilian Forest" use the goal programming method to optimize six objectives "to maximize wood harvest, to maximize number of tourists, to maximize the pasture (creation of buffalos), to minimize number of employees, to maximize the diversity of flora and to maximize the diversity of fauna" with "total land and forest area" as its constraints and "area used for timber, ervamate, leaves, pasture and tourism" as the decision variables.

4. Oduguwa et al. in their paper "Multi-Objective Optimization of protein ligand in drug discovery" use the evolutionary algorithm (C++) to optimize three objectives "to maximize internal energy of the compound, to maximize

protein-compound couple's Van Der Waal's & electrostatic energy of interaction and to maximize shape complementarities" with "computer specifications, population size and number of generations" as its constraints and "docking configurations (output complex of drug)" as the decision variables.

5. Weber, Charles and Lisa in their paper "Supplier Selection using Multi-Objective Programming" use the decision support system approach (compromise) to optimize three objectives "to minimize price, to maximize quality and to minimize late orders" with "number of units required, number of suppliers required to fulfill demand, quantity of late deliveries and quality restriction" as its constraints and "quantities ordered" as the decision variable.

6. Vergidis, Tiwari and Majeed in their paper "Business Process Improvement using Multi-Objective Optimization" use the interactive programming, evolutionary algorithm method to optimize two objectives "to minimize cost and to maximize duration of business process" with "cost, resource, duration of activity and information source" as its constraints and "resource made available" as the decision variable.

### 3.2.2 Multi-objective methods application in Supply Chain field

Further, the search concentrated on the area of “multi-objective optimization in supply chain networks”:

1. Belgasmi, Said and Ghedira in their paper “Evolutionary Multi-Objective Optimization of the Multi-Location Transshipment” use the strength pareto evolutionary algorithm (SPEA2) method to optimize three objectives “to minimize total expected cost, to maximize fill rate and to minimize expected transshipment lead time” with “quantities shipped cannot be more than those available at store and quantities shipped cannot be more than unmet demand at store” as its constraints and “order quantities” as the decision variable.

2. Pinto in her paper “Supply Chain Optimization using Multi-Objective Evolutionary Algorithm” use the non-dominated sorting genetic algorithm-II (NSGA-II) method to optimize five objectives “to minimize total operating cost, to minimize manufacturing cost, to maximize profit, to maximize revenue and to minimize transportation cost” with “plant capacity, supplier capacity, inventory balancing constraints for respective components and total cost” as its constraints and “number of components shipped from plants to supplier, number of products shipped from plants to customer zones and inventory of components at plants” as the decision variables.

3. Mumford in her research “Multi-Objective Optimization for Green Logistics” uses “building multi-objective optimization decision support tools for strategic and operational SCM, with a special focus on environmental issues” as its objective.

4. Sabri and Beamon in their paper “Multi-Objective Approach to Simultaneous Strategic and Operational Planning in Supply Chain Design” use an algorithm (developed by them) to optimize three objectives “to minimize total fixed

cost, to minimize variable cost and to minimize volume flexibility" with "supplier capacity, production requirement, plant capacity, production capacity, throughput capacity, each customer zone assigned to one distribution center, products shipped are equal to products available at plants and total shipments are equal to demand requirements" as its constraints and "quantity of products produced in plants, products shipped from plants to distribution center, products shipped from suppliers to plants and cost" as the decision variables.

The problems within the research work mentioned above are:

1. In the weighted objective method, the decision maker needs to give the prior information about the importance of each objective to the analyst. The analyst can give weight to the objectives based on the information provided to him by the decision maker and optimize the resulting single objective.

2. Similarly, in the goal programming method, the decision maker needs to give the prior information to the analyst about the order in which the goals need to be achieved. The resulting problem is solved via a series of single objective problems (Hwang et al. 56).

3. A considerable number of evolutionary algorithms have been proposed in the last few years. Comparative studies have shown that amongst all the evolutionary algorithms created, the Strength Pareto Evolutionary Algorithm (SPEA2) and the Non-dominated Sorting Genetic Algorithm (NSGA-II) show the best performance (Belgasmi et al. 11). But in higher dimensional spaces, SPEA2 seems to have advantages over NSGA-II (Belgasmi et al. 11).

4. However, the research into multi-objective optimization techniques for supply chain management is still in its infancy (Mumford).

### 3.2.3 Multi-objective methods application in "Supplier Selection"

As one can see from the literature search, numerous articles have been written and researched on the topics "general multi-objective optimization technique" and "multi-objective optimization in supply chain management" but less research has been done on the topic "multi-objective optimization in supplier selection."

Hence, the search narrowed down to "multi-objective methods used for supplier selection in outsourcing."

1. Thaver and Wilcock in the problem "Identification of Overseas Vendor Selection Criteria used by Canadian Apparel Buyers" use a nine-point scale ranking system to optimize criteria "price, quality, willingness to negotiate prices, lead time for delivery, time for quotation, communication system, quantity required, technical expertise, merchandise fashionability, financial position, export quota, long-term commitment, economic stability in country, registered to quality program, processing EDI, registered to ISO 9000" and select the suppliers.

The subject identified in this paper is one in which there are too many questions and very few answers.

2. "The Outsourcing Institute's Annual Survey of Outsourcing End Users," article states, "price, commitment to quality, additional value-added capability, scope of resources, location, existing relationship, cultural match, reputation, flexible contract terms" as criteria for supplier selection.

3. Wadhwa and Ravindran in their paper "Vendor Selection in Outsourcing" use weighted objective method, goal programming method and compromise programming method to optimize three objectives "minimize price, minimize lead time and maximize quality" with "capacity, demand, number of suppliers

and linear” as its constraints and “product quantity, binary variables for supplier and price level” as the decision variables.

Wadhwa and Ravindran have made suggestions in their paper that risk quantification and global supplier selection is gaining much importance in the real business world, which can be an extension to their work.

In recent years there has been an increased demand and popularity for solving the multi-objective problems in supplier selection. Thus the focus of the study would be to optimize the criteria “price of the goods or services, quality of the goods or services, lead time for delivery, additional value-added capability of the supplier, scope of the resources, location of the supplier, existing relationship, reputation of the supplier, cultural barrier and risk” using the goal programming method, the evolutionary programming method and the STEM method.

# CHAPTER 4

# PROBLEM STATEMENT

## 4. PROBLEM STATEMENT

Outsourcing is defined as purchasing ongoing services and parts from an outside company that a company currently provides, or most organizations normally provide for themselves (Wadhwa and Ravindran 3725). Outsourcing is "paying another company to provide you with a service or product that you would otherwise have your own employees conduct" (Anthony). Many large organizations are outsourcing those activities which are not either cost efficient if done in-house or are not core to their businesses (Wadhwa and Ravindran 3725). Outsourcing the activity also means that the work is distributed and hence the time-to-market the final product can be reduced.

Recently, outsourcing has become the prime focus of the company (Wadhwa and Ravindran 3725). Cost reduction is not the only criteria for outsourcing but the ability to focus on core competencies is also an important criteria. Many companies are now evaluating, supply chain procurement and the logistics as the candidates for outsourcing. Various reasons for outsourcing are (Wadhwa and Ravindran 3726):

> In many cases the third party can provide procurement services more efficiently. Outsourcing can provide access to specialized technology and operational platforms.

1. Outsourcing can help reduce the staffing levels.
2. The advancement in technologies has made procurement a very specialized service.

In the Outsourcing Institute's Annual Survey of Outsourcing End Users (The Outsourcing Institute Membership), the reasons for outsourcing are stated as:

1. Reduce and control operating costs
2. Improve company focus

3. Gain access to world-class facilities
4. Free internal resources for other purposes
5. Accelerate the reengineering benefits
6. Make capital funds available
7. Share risks
8. Cash Infusion

A survey carried out by the Aberdeen group found that more than 83% of the organizations engaged in outsourcing achieved significant decrease in the purchasing cost, more than 73% of the industries found reduction in transaction cost and 60% were able to reduce sourcing and procurement cycles (Wadhwa and Ravindran 3726).

Once the decision to outsource has been taken by the company, the next most important activity or challenge to the company is the selection of suppliers. The decision for selecting the right supplier is prone to errors. The right supplier will lead to the fulfillment of the company's needs and the long-term relationship (Wadhwa and Ravindran 3726). The right supplier will also help increase the financial stability as well as the reputation of the company in the market. Selection of the right supplier is a very difficult task. It is possible that some suppliers may satisfy four criteria from a set of nine selected criteria but not satisfy the remaining five. Some suppliers may satisfy the other five criteria but not the first four. Study has shown that these criteria vary from product to product and also by the presence of quality programs within the business (Thaver and Wilcock 56).

Thus, supplier selection problems are multi-objective problems and not single objective problems. Some criteria are quantitative whereas some are qualitative. No single methodology can address all the issues of the supplier selection problem and provide the optimal solution. For example, a multi-objective method such as evolutionary programming is the best way to create a number of non-dominated solutions within a few minutes that would take hours for any other method. Once a number of solutions are created, to reduce it to a smaller set it is best to involve the

decision maker and set the goals for each objective, thereby reducing the set of solutions. This process is called the goal programming method. The results after the application of the goal programming method may not be satisfactory to the decision maker, hence the STEM method can be used to further filter the results and improve the non-satisfactory results. Thus, a combination of methods would definitely prove to be a good methodology to tackle the multiple criteria of the supplier selection problem.  In this study various criteria would be considered and different multi-criteria methodologies would be studied against more prevalent criteria (supplier selection) and a heuristic methodology would be developed to find a suitable solution.

# CHAPTER 5

# CRITERIA SELECTION AND JUSTIFICATION

## 5. CRITERIA SELECTION AND JUSTIFICATION

Various criteria used by different researchers in their research against the supplier selection problem were studied to reach the final set of criteria to be used in this study. Thaver and Wilcock in their paper "Identification of overseas supplier selection criteria used by Canadian apparel buyers" use the following criteria and a nine-point scale ranking system for the selection of the suppliers (62):

b. Prices
c. Quality
d. Willingness to negotiate prices
e. Lead times for delivery
f. Time for quotation, communication system
g. Quantity required, technical expertise
h. Merchandise fashionability
i. Financial position
j. Export quota
k. Long-term commitment
l. Economic stability in country
m. Registered to quality program
n. Processing EDI
o. Registered to ISO 9000

"The Outsourcing Institute's Annual Survey of Outsourcing End Users" article states the following criteria for supplier selection:

b. Price
c. Commitment to quality
d. Additional value-added capability
e. Scope of resources, location
f. Existing relationship

g. Cultural match
h. Reputation
i. Flexible contract terms

Wadhwa and Ravindran in their paper “Vendor selection in outsourcing” use weighted objective, goal programming and compromise programming methods to optimize three objectives (3729):

a. Minimize price
b. Minimize lead time
c. Maximize quality

The criteria to be considered in this thesis for the supplier selection problem and the reasons for their selection are stated below:

The criteria for selection of suppliers depend on the type of product or service to be outsourced. It will not be the same for all the purposes (Thaver and Wilcock 58). In general, the cost of the service outsourced is the main and primary criteria which every company tries to negotiate. In today’s world, a major portion of the company sales is incurred in outsourcing (Wadhwa and Ravindran 3726). For the purpose of competitiveness, it is important that companies keep their purchasing cost to a minimum. Hence, the first criterion to be considered is “**price of the goods or services**” acquired. Its unit of measure is US Dollars.

The next criterion is the “**lead time for delivery**.” The time a supplier takes to deliver a single order is very important in a company decision to consider that supplier or not. The company can calculate the ideal time required to manufacture and supply certain products and then compare it with the lead times given by different suppliers to see if they qualify on this criterion. Its unit of measure is number of days.

The third important criterion is the “**quality of the products**” being supplied by the suppliers. This can further decide the level of reliability of the suppliers. If the quality is good, one can always keep the customers happy. It is measured quantitatively as “percent rejections” of the parts supplied.

The “**transportation cost**” is also a determinant criterion to be considered. The suppliers can be located anywhere, locally or else overseas. Since there will be supplier visits and audits conducted by the companies, transportation cost measured in US Dollars is the fourth important criterion to be considered.

The fifth criterion to be considered is the “**scope of the resources**” which means, the company’s access to the set of resources required to deliver a particular product or service that is outsourced. In this criterion, the range and the power of a particular supplier to access the required resources within minimal time is measured.

The next few important criteria to be considered are “**reputation of the supplier**” in the current market, “**cultural barrier**,” “**risk**;” a particular company has such as receiving the reliable delivery and services etc., and “**existing relationship of the company**” with the supplier.

The tenth important criterion is the “**additional value-added capability**” which means, the capability of the suppliers to provide additional value to the services they deliver.

Thus the criteria to be considered in this thesis study are:

a. Price of the goods or services
b. Lead time for delivery
c. Quality of the goods or service
d. Transportation Cost
e. Scope of the resources
f. Reputation of the supplier

g. Cultural barrier
h. Risk
i. Existing relationship
j. Additional value-added capability

The first four criteria – “price of the goods or services,” “lead time for delivery,” “quality of the goods or service,” and “transportation cost” – are quantitative ones which can be optimized using multi-objective decision making methods. The other six criteria – “scope of the resources,” “reputation of the supplier,” “cultural barrier,” “risk,” “existing relationship,” and “additional value-added capability” – are non-quantitative criteria (qualitative criteria); rather, they are the attributes in the supply chain problem and hence, can be optimized using multi-attribute decision making methods.

A brief summary of the two methodologies “Multi-Attribute Decision Making” and “Multi-Objective Decision Making” is presented in the upcoming chapters to understand the application of these methods on the optimization of the multiple criteria of supplier selection. Since, the thesis study concentrates on multi-objective decision making a representative supplier selection problem is synthesized using the glossary and the mathematical forms to illustrate the selection of right suppliers based on four important quantitative criteria. A heuristic methodology developed in this thesis, is explained and its application in selecting the right suppliers is illustrated in detail with the help of a representative supplier selection problem.

# CHAPTER 6

# MULTI-ATTRIBUTE DECISION MAKING

# 6. MULTI-ATTRIBUTE DECISION MAKING

## 6.1 Introduction

In the study of decision making in complex situations, terms like "multi-objectives," "multi-attribute," "multi-criteria," "multi-dimensional" are used interchangeably to describe decision making situations (Hwang et al. 12). Multiple attribute decision problems involve the selection of the best alternative from a pool of pre-selected alternatives described in terms of their attributes. In other words, this method is used for selecting an alternative from a small, explicit list of alternatives (Hwang et al. 303).

The MADM methods can be classified as follows (Hwang et al. 304):

1. Methods for full dimensional approach:
   1.1 Dominance
   1.2 Disjunctive and conjunctive constraints

2. Methods for single dimensional approach
   2.1 Maximin
   2.2 Maximax
   2.3 Lexicography
   2.4 Elimination of aspects
   2.5 Effective index

3. Methods for single dimensional approach – with utility theory
   3.1 Unidimensional utility theory
   3.2 Additive utility model
   3.3 Additive expected utility model
   3.4 Quasi-additive utility model
   3.5 Hierarchy utility model

4. Methods for intermediate dimensional approach

   4.1 Trade-offs

   4.2 Non-metric multi-dimensional scaling

## 6.2 MADM application in this thesis study

As stated earlier, the criteria considered for supplier selection in the study are:

a. Price of the goods or services
b. Lead time for delivery
c. Quality of the goods or service
d. Transportation Cost
e. Scope of the resources
f. Reputation of the supplier
g. Cultural barrier
h. Risk
i. Existing relationship
j. Additional value-added capability

Out of the above mentioned criteria, “scope of the resources,” “reputation of the supplier,” “cultural barrier,” “risk,” “existing relationship,” and “additional value-added capability” are non-quantitative criteria. These criteria are the attributes in the problem and hence, can be solved using one of the multi-attribute decision making methods.

For example, consider an ad-hoc method in which the criteria are measured in terms of a 1-10 scale with “10” being the highest score for a particular supplier and “1” being the lowest so as to be on the same terms or units as the other criteria (objectives) which will help optimize the objectives together.

Table 6.1

| Rating on 1-10 scale | | | | | |
|---|---|---|---|---|---|
| **Criteria** | **Suppliers** | | | | |
| | **Supplier 1** | **Supplier 2** | **Supplier 3** | **Supplier 4** | **Supplier 5** |
| **Scope of the resources** | **8** | **9** | **7** | **5** | **8** |

| | | | | | |
|---|---|---|---|---|---|
| **Reputation of the supplier** | **9** | **10** | **6** | **6** | **9** |
| **Cultural barrier** | **10** | **10** | **10** | **8** | **6** |
| **Risk** | **10** | **9** | **9** | **7** | **8** |
| **Existing relationship** | **9** | **6** | **2** | **8** | **4** |
| **Additional value-added capability** | **7** | **8** | **6** | **4** | **5** |

The decision maker will specify certain target values for each objective (criterion) to be achieved. For example, the decision maker will specify that all the suppliers scoring above "7" in the criteria "scope of the resources," "reputation of the supplier," and "cultural barrier" and above "5" in the criteria "risk," "existing relationship," and "additional value-added capability," must be selected.

The automatic screening of the suppliers can be done based on the targets specified by the decision maker for the attributes, "scope of the resources," "reputation of the supplier," "cultural barrier," "risk," "existing relationship," and "additional value-added capability," the supplier list can be narrowed to a smaller set.

Based on the scores indicated in the table and the target values given by the decision maker, the qualified suppliers are "Supplier 1" and "Supplier 2."

Similarly, one of the methods from MADM can be applied to arrive at the optimal decision for these criteria. This thesis study concentrates mainly on multi-objective decision making and not on multi-attribute decision making which is again a vast field of study. Hence, these criteria are not within the scope of the study in this thesis.

The first four objectives "price," "lead time," "quality," and "transportation cost" can be solved using multi-objective decision making methods, which is the focus of this thesis study. Thus, these are the four objectives being considered in the upcoming chapters of the thesis study for supplier selection.

# CHAPTER 7

# MULTI-OBJECTIVE DECISION MAKING METHODS AND EXAMPLES

# 7. MULTI-OBJECTIVE DECISION MAKING METHODS AND EXAMPLES

## 7.1 Introduction

Once the decision to outsource has been made by the company, the next most important activity or challenge to the company is the selection of suppliers. The right supplier will lead to the fulfillment of the company's needs and a long-term relationship (Wadhwa and Ravindran 3726). It will help increase the financial stability as well as the reputation of the company in the market. Selection of the right supplier is a difficult task. It is possible that some suppliers may satisfy four criteria from a set of nine selected criteria but not satisfy the remaining five and some suppliers may satisfy the other five criteria but not the first four. Studies have shown that these criteria vary from product to product and also by the presence of quality programs within the business (Thaver and Wilcock 56). Thus, this highlights the fact that supplier selection problems are multi-objective problems and not single objective problems. Some criteria are quantitative whereas some are qualitative. No single methodology appears to be dominant in solving the supplier selection problem.

The need to resolve conflicting and multiple objectives in the current supply chain scenarios such as supplier selection requires additional research focusing on the multi-objective methods that will lead to suitable optimization models. For a company to stay competitive in the marketplace, it has to adopt different business strategies. Use of multi-objective optimization techniques to solve the multiple objectives of supply chain scenarios would lead to proper treatment of all critical objectives.

The optimal decision for a supplier selection problem cannot be reached, by optimizing the objectives separately. At the same time it is not possible to optimize all the objectives at once. Thus the traditional methods cannot be used to optimize the supply chain objectives (Pinto). The results will also be misleading in the dynamic environment. In solving a multiple objective problem, a non-dominated (Pareto) set

of solutions will be generated and presented to the decision maker in order to reach the final solution. A non-dominated solution is the one in which no objective function can be improved without degrading simultaneously at least one of the other objective functions. A solution can not be chosen as a better solution from a set of non-dominated solutions mathematically. The preference information from the decision maker (DM) is needed to reduce the set of non-dominated solutions as well as in arriving at the final solution. Hwang et al. presented the taxonomy of numerous multi-objective models that use the preference information given by the decision maker to the analyst at a particular stage (8):

1. No articulation of preference information
   - No need for any information from the decision maker to the analyst once the objectives and the constraints are set-up.
   - Decision maker will accept solution obtained from the method.
   - The advantage is the decision maker is not disturbed by the analyst which may be preferred by the decision maker.
   - The disadvantage is that analyst needs to make many assumptions about the decision maker's preferences which is difficult to do with even the best and knowledgeable analyst.

2. A priori articulation of preference information
   - Preference information is given to the analyst by the decision maker before he solves the problem.
   - The information may be given in two ways, the decision maker will give some judgment about specific objective preference levels or he will rank the objectives in order of their importance.

3. Progressive articulation of preference information (Interactive Methods)
   - These methods are known as interactive methods.
   - At each iteration of solution, preference information is expected from the decision maker. The decision maker may not provide any priori information in

these cases but he gives the preference information on a local level to a particular solution(s) presented.

- After the limited number of interactions with the decision maker, these methods lead him to the final/ preferred solution.
- The disadvantage is that the decision maker is asked to be involved frequently as compared to other methods.

4. A Posteriori articulation of preference information (Non-dominated Solutions Generation Methods)
   - In this method, the analyst presents the subset of the complete set of non-dominated solutions to the decision maker and the decision maker then implicitly uses the trade-off information in order to select the preferred solution.
   - The disadvantage is that there are too many solutions presented to the decision maker amongst which he needs to select one. Hence these methods are combined with other methods such as interactive methods.

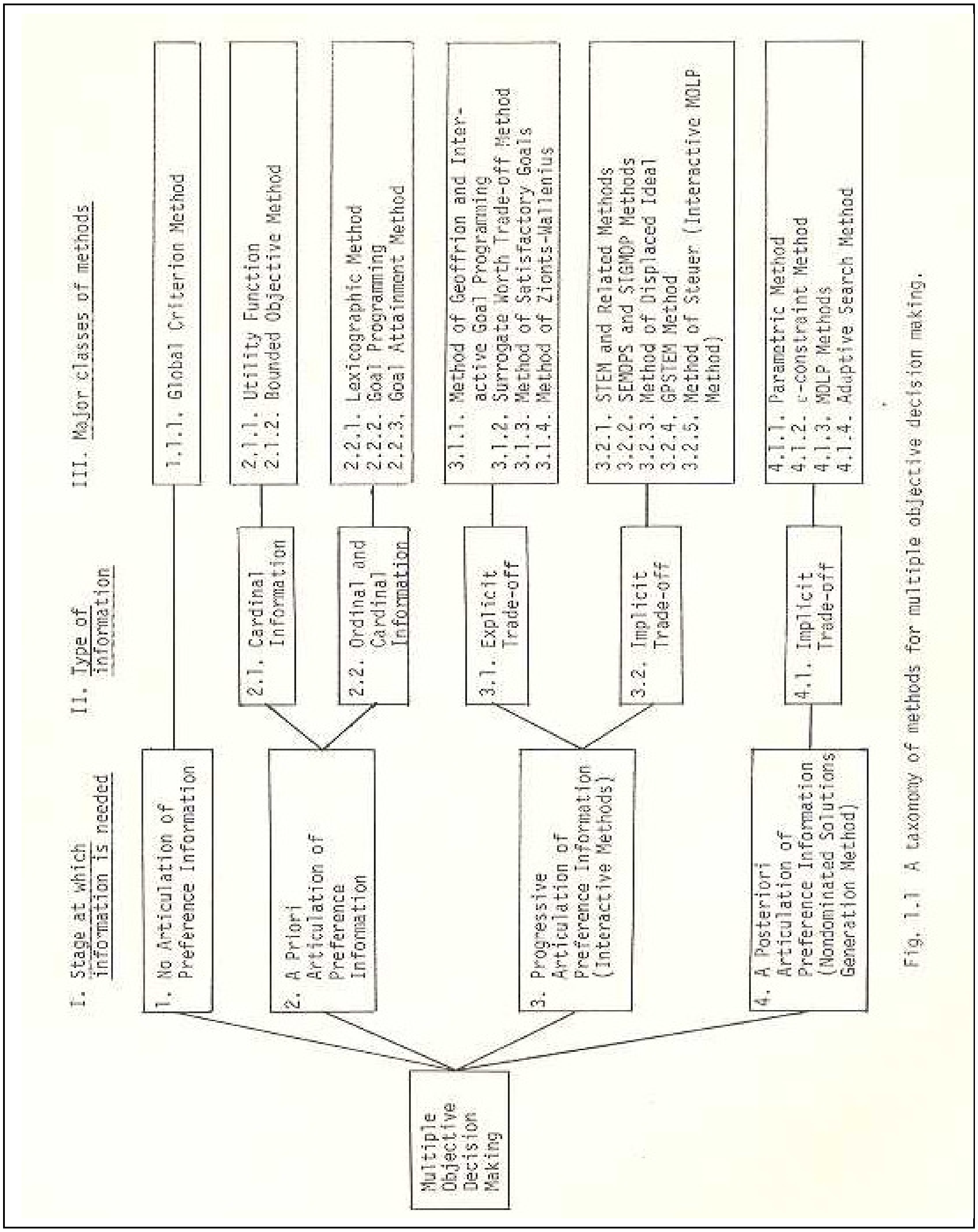


Figure 7.1: Taxonomy of Multi-objective Decision Making Methods

**Source:** Hwang, C. L., S. R. Paidy, K. Yoon, and A. S. M. Masud. *Multiobjective Decision Making – Methods and Applications*. New York: Springer-Verlag Berlin Heidelberg, 1979. Print.

In this study various criteria would be considered and different multi-criteria methodologies would be studied against more prevalent criteria (supplier selection) and a heuristic methodology would be developed to find a suitable solution. An optimization model would be developed that is best suited for the procurement of the services from the suppliers which can be both local as well as overseas. This model further can also be applied to the procurement of the raw material from the suppliers. The development process will study and contrast various optimization methods being used in the previous research in a variety of problems and use an innovative methodology to solve the supplier selection problem. The methods include the weighted objective method, the goal programming method, the evolutionary algorithm method and the STEM method.

The different multi-objective methods used in this study to solve the supplier selection problem are:

1. Weighted Objective Method (Hwang et al. 32):

   This priori articulation method scalarizes a set of objectives into a single objective by pre-multiplying each objective with the user-supplied weight. This method is one of the simplest to optimize a multi-objective problem. For example, if there are two objectives such as minimizing the cost and maximizing the production of a particular product in a company manufacturing two types of products, one would apply weights to these two objectives as indicated by the decision maker and optimize the problem easily. If the cost to be minimized is of high importance to the company as compared to the production of the specific product, the decision maker would give more weight to the cost variable than the product variable.

2. Goal Programming Method (Hwang et al. 56):

   In this priori articulation method, goals (target values) are set for the multiple objectives that are ranked according to the priorities in which they need to be achieved. The importance of these goals and the order in which they need to be

achieved are articulated by the decision maker. For example, suppose in a doll manufacturing company, the company manufactures two types of dolls ‘A’ and ‘B’. For each doll ‘A’ sold, the company makes $ 0.4 profit and for each doll ‘B’ sold, the company makes $ 0.3 profit. Doll ‘A’ requires twice the time to manufacture as compared to that of doll ‘B’. Two objectives the company has are to maximize the profit and maximize the production of product ‘A’. The raw material available for each day’s production of dolls ‘A’ and ‘B’ is limited to only 400. After calculation, the company finds out that the maximum number of doll ‘A’ it can produce is 250 and 0 of doll ‘B’, whereas if one customer asks for 300 dolls of type ‘A’, it falls short of the raw material. In such a situation, the decision maker must specify the priority of his goals. He may specify his first goal, to avoid the over usage of the raw materials, second to satisfy the closest customer by producing as many number of product ‘A’ as possible and the last priority is to maximize the profit as much as possible.

3. Evolutionary Algorithm Method:

   There are many different types of evolutionary algorithms such as genetic algorithms, evolutionary strategies, genetic programming and evolutionary programming (Pinto). The basic working principle/logic of genetic algorithm is as explained below (Pinto):

   - Create random population of ‘n’ individuals.
   - These solutions are then compared and evaluated against the fitness function.
   - Create new members for the next population using the reproductive operators: crossover and mutation.
   - If the non-dominated set falls below the pre-specified level then there is no need for increase in population size.

   An evolutionary algorithm can be used as a posteriori articulation method.

4. STEM Method (Hwang et al. 170):

   The STEM method falls under the “Progressive articulation of Preference Information” category. The trade-off information is implicitly specified to the analyst by the decision maker. Trade-off information is the ratio of change in one objective function to the change in another objective function. In the STEM method, multiple phases of computation and decision making are interactive. Hence it allows the decision maker to recognize good solutions and the relative importance of the objectives. A pay-off table (usually a set of solutions in which one of the objectives is at its optimum) is constructed before the first interactive cycle and the best solution is found from it using a min-max strategy where the objective is to minimize the maximum deviation of an objective from its optimal solution. This step is analogous to the Global Criterion method where no articulation of preference information is needed. The cycle keeps repeating, at the $m^{th}$ cycle, the feasible solution is presented to the decision maker, which is the closest solution to the optimal value. The decision maker then compares this value to the ideal solution, if more iterations are required, the decision maker needs to relax some satisfactory objectives in order to improve unsatisfactory ones. The process cycle continues until the decision maker is satisfied with the solution.

These four methods present almost all aspects of preference information articulation and will be used as building blocks for the heuristic method proposed in this study.

### 7.2 Weighted Objective Method

Weighing objectives to obtain an efficient or pareto-optimal solution is one of the oldest multi-objective solution techniques (Wadhwa and Ravindran 3730). This method scales a set of objectives into a single objective by pre-multiplying each objective with the user supplied weight. It is the simplest way to apply if there are a number of objectives to be optimized. Weighted means are used by the statisticians to compensate for the presence of bias. It is used to give some elements or objectives more weight than others.

It has been shown that line passing through the point of tangency of indifference curve and non-dominated solution set is a source of the optimal values for the weight (Hwang et al. 32), i.e. slope of the tangent is proportional to the ratio of the optimal values of the weights. Thus if the optimal weights can be determined, the multi-objective problem will ensure the most satisfactory solution. However, in practice the weights often are the decision maker's subjective estimate of the importance of different objectives and not necessarily the optimal values (Hwang et al. 32).

Similar to the two sided coin concept, this method also has its own advantages as well as disadvantages. The advantage of this method is that it is easier to get the weights from the decision maker, who may believe these values are correctly known before the actual solution. The disadvantage of this approach is that the weights depend on the achievement level of the objective functions and relative achievement of the objective functions compared to the achievement levels of the other objective functions (Hwang et al. 32).

For example, if there are two objectives such as minimizing the cost and maximizing the production of a particular product in a company, manufacturing two types of products, one would apply weights to these two objectives as indicated by the decision maker and optimize the problem easily. If the cost to be minimized is of high importance to the company as compared to the production of the specific product, the

decision maker would give more weight to the cost function rather than the product function.

Maximize: $w_1F_1 + w_2F_2$

Subject to: $g_i(x) \leq 0 \quad \forall \quad i = 1, 2, 3 \ldots\ldots\ldots n$

where $w_1$, $w_2$ are the weights on each of the objectives $F_1$ and $F_2$ respectively. The optimal solution to the weighted problem is a non-inferior solution to the multi-objective problem as long as all the weights are positive. The weights can be systematically varied to generate several efficient solutions. The weighting method is generally used to approximate the efficient set; it is not a good method for finding an exact representation of the efficient solution.

**Example:**

The following example by Wadhwa, Ravindran is illustrated and used in the paper "Supplier Selection in Outsourcing." This is a supply chain problem in which buyers have to select the suppliers based on the various criteria that buyers decide upon. Often in the supplier selection problem, buyers have a dilemma due to the volume discounts offered by the suppliers, which depend on the volume of the order placed. The criteria considered in this problem for the selection of the potential suppliers are:

**Price:** Total cost of the purchasing of the parts from the suppliers consists of two factors, Fixed Cost and Variable Cost.

**Lead Time:** Lead time is the summation of the product of lead time for each part and the quantity of the parts being ordered. Lead time is measured in terms of days.

**Quality:** This means the quality of the parts supplied by various suppliers which is measured by keeping a log of the number of rejections of

the parts received by the quality department of the company. It is measured in terms of percent of rejections.

The principal set of indices used to denote the various entities such as customers, parts, suppliers, etc. are shown in Table 7.1:

Table 7.1

| INDEX | ENTITY | QUANTITY |
|---|---|---|
| i | Part | 2 |
| j | Buyer | 2 |
| k | Supplier | 2 |
| m | Incremental Price Break | 2 |

Parameters used in the problem are:

- $P_{ikm}$: Cost of acquiring unit of part *i* from supplier *k* at price level *m*, $\forall$ i, k, m = 0, 1
- $F_k$: Fixed Cost associated with each supplier, $\forall$ k = 0, 1

  $F_1 = 3500$

  $F_2 = 3600$
- $d_{ij}$: Demand of part *i* by buyer *j*, $\forall$ i & j = 0, 1

  $d_{00} = 150$

  $d_{01} = 175$

  $d_{10} = 200$

  $d_{11} = 180$
- $q_{ik}$: Quality that supplier maintains for part *i,*

  $q_{00} = 0.03$

  $q_{01} = 0.09$

  $q_{10} = 0.06$

  $q_{11} = 0.02$

- $l_{ijk}$: Lead time of supplier *k* to produce and supply part *i* to buyer *j,*

  $l_{000} = 15$

$l_{010} = 17$

$l_{001} = 19$

$l_{011} = 18$

$l_{100} = 24$

$l_{110} = 21$

$l_{101} = 11$

$l_{111} = 12$

- $CAP_k$: Production capacity of supplier *k* for part *i,*

  $CAP_0 = 300 \quad \forall \quad i = 0$

  $CAP_0 = 350 \quad \forall \quad i = 1$

  $CAP_1 = 280 \quad \forall \quad i = 0$

  $CAP_1 = 360 \quad \forall \quad i = 1$

- $b_{ikm}$: Quantity at which price break occurs for part *i* given by supplier *k* for buyer *j,*

  $b_{000} = 85 \quad \forall \quad j = 0, 1$

  $b_{010} = 95 \quad \forall \quad j = 0, 1$

  $b_{001} = 180 \quad \forall \quad j = 0, 1$

  $b_{011} = 190 \quad \forall \quad j = 0, 1$

  $b_{100} = 120 \quad \forall \quad j = 0, 1$

  $b_{110} = 125 \quad \forall \quad j = 0, 1$

  $b_{101} = 210 \quad \forall \quad j = 0, 1$

  $b_{111} = 205 \quad \forall \quad j = 0, 1$

- $L_{ij}$: Lead time that buyer *j* requires for part *i,*

  $L_{00} = 18$

  $L_{01} = 20$

  $L_{10} = 26$

  $L_{11} = 13$

- $Q_j$: Quality level that buyer *j* requires all suppliers to maintain,

  $Q_0 = 0.095$

  $Q_1 = 0.09$

- N: Maximum number of suppliers that can be selected

Decision variables used in the model are:

- $X_{ijkm}$: Number of units of part *i* supplied by supplier *k* to buyer *j* at price level *m*
- $Z_k$: It is a binary variable which denotes whether a supplier is selected or not
- $Y_{ijkm}$: Also a binary variable which denotes which price level is selected

The problem consists of three objectives:

1. To minimize the total purchasing cost:

   Total Variable Cost: $\sum_i \sum_j \sum_k \sum_m P_{ikm} \cdot X_{ijkm}$

   Fixed Cost: $\sum_k F_k \, . \, Z_k$

   Hence, MIN $\sum_i \sum_j \sum_k \sum_m P_{ikm} \cdot X_{ijkm} + \sum_k F_k \, . \, Z_k$

2. To minimize the lead time:

   MIN $\sum_i \sum_j \sum_k \sum_m l_{ijk} \cdot X_{ijkm}$

3. To minimize the number of part rejections:

   MIN $\sum_i \sum_j \sum_k \sum_m q_{ik} \cdot X_{ijkm}$

Under the weighted objective method, the above problem is transformed into the following single objective optimization problem:

$$\text{MIN} \quad w_1 \left( \sum_i \sum_j \sum_k \sum_m P_{ikm} \cdot X_{ijkm} + \sum_k F_k \, . \, Z_k \right) + w_2 \left( \sum_i \sum_j \sum_k \sum_m l_{ijk} \cdot X_{ijkm} \right) +$$

$$w_3 \left( \sum_i \sum_j \sum_k \sum_m q_{ik} \cdot X_{ijkm} \right)$$

where $w_1$, $w_2$ and $w_3$ are weights

The problem has the following constraints:

1. Capacity Constraint: Each supplier *k* has maximum capacity $CAP_k$,

$\sum_i \sum_j \sum_m X_{ijkm} \leq (CAP_k) Z_k \quad \forall \quad k = 0, 1$

2. Demand Constraint: The demand of buyer *j* for part *i* has to be satisfied.

   $\sum_k \sum_m X_{ijkm} = d_{ij} \quad \forall \quad i,j$

3. Maximum number of selected suppliers: Maximum number of selected suppliers should be less than the number of participating suppliers

   $\sum_k Z_k \leq N$

4. Linearizing constraints:

   $X_{ijkm} \leqslant (b_{ikm} - b_{ikm-1}) * Y_{ijkm} \quad \forall i, j, k, \; 1 \leqslant m \leqslant m_k,$

   $X_{ijkm} \geqslant (b_{ikm} - b_{ikm-1}) * Y_{ijkm+1} \quad \forall i, j, k, \; 1 \leqslant m \leqslant m_k - 1.$

**Solution:**

The solution according to the paper "Supplier Selection in Outsourcing" by Wadhwa and Ravindran (2007) is as follows,

1. Price = USD $93070
2. Lead Time = 11890 days
3. Quality = 35.85 %

### 7.3 Goal Programming Method

Goals are the objectives or targets desired by the decision maker expressed in terms of a specific state in space and time. Thus, while objectives give the desired direction, goals give a desired target level to achieve.

Goal Programming was originally proposed by Charnes and Cooper for a linear model, which was further developed by Ijiri, Lee and Ignizio (Hwang et al. 56). The method requires the decision maker, to set the goals (targets) for the multiple objectives that are ranked according to the priorities in which they need to be achieved. The importance of these goals and the order in which they need to be achieved are articulated by the decision maker. A preferred solution is thus defined as the one which minimizes the deviations from the set goals. Given a portfolio of properly established goals, one tries to achieve them as closely as possible (Wadhwa and Ravindran 3731).

Goal programming is a three step approach as follows (Wadhwa and Ravindran 3731):

**Step 1:** Get the goals (targets) from the decision maker to be achieved for each objective. Goals are not constraints. Hence some may not be achievable.

For example, for objective function $F_i$ whose target value is $B_i$; the goal constraint is written as,

$$F_i(x) + d_i^- - d_i^+ = B_i$$

Where, $d_i^-$ = underachievement of goal

$d_i^+$ = overachievement of goal

**Step 2:** Get decision maker's preference on achieving the goals. The preference information can be provided in one of three possible ways:

**a. Ordinal:** Objectives are ranked according to preference of order by the decision maker.

**b. Cardinal:** Specific weights are specified by the decision maker for each objective according to the preference order.

**c. Hybrid:** This consists of ranking and weights both being specified by the decision maker.

**Step 3:** Find an optimal solution that will be as close as possible to the stated goals, in the specified preference order.

The goal programming problem may be the linear integer goal programming problem or the non-linear integer goal programming problem. The deviations are to be minimized as much as possible. A lower ranking achievement function cannot be satisfied for the detriment function (Hwang et al. 57). The same problem can be solved using the iterative goal programming method. Using the method of linear approximation of non-linear functions, the non-linear goal programming problem can be solved by linear goal programming (Hwang et al. 57).

In this thesis the iterative goal programming method approach is used. The goal programming advantage is that the decision maker can give rankings instead of specifying weights to each objective function. The goal programming method has been widely used in many multi-objective decision making problems (Hwang et al. 58).

For example, suppose in a doll manufacturing company, the company manufactures two types of dolls 'A' and 'B'. For each doll 'A' sold, the company makes \$ 0.4 profit and for each doll 'B' sold, the company makes \$ 0.3 profit. Doll 'A' requires twice the time to manufacture as compared to that of doll 'B'. Two objectives the

company has are to maximize the profit and maximize the production of product ‘A’. The raw material available for each day’s production of dolls ‘A’ and ‘B’ is limited to only 400. After calculation, the company finds out that the maximum number of doll ‘A’ it can produce is 250 and 0 of doll ‘B’, whereas if one customer asks for 300 dolls of type ‘A’, it falls short of the raw material. In such a situation, the decision maker must specify the priority of his goals. He may specify his first goal, to avoid the over usage of the raw materials, second to satisfy the closest customer by producing as many number of product ‘A’ as possible and the last priority is to maximize the profit as much as possible.

**Example:**

The following example is illustrated and used in the paper “Vendor Selection in Outsourcing” by Wadhwa and Ravindran. This is a supply chain problem in which buyers have to select the suppliers based on the various criteria that buyers decide upon. Often in the supplier selection problem, buyers are in the dilemma due to the volume discounts offered by the suppliers, which depend on the volume of the order placed. The criteria considered in this problem for the selection of the potential suppliers are:

**Price:** Total cost of the purchasing of the parts from the suppliers consists of two factors, Fixed Cost and Variable Cost.

**Lead Time:** Lead time is the summation of the product of lead time for each part and the quantity of the parts being ordered. Lead time is measured in terms of days.

**Quality:** This means the quality of the parts supplied by various suppliers which is measured by keeping a log of the number of rejections of the parts received by the quality department of the company. It is measured in terms of percent of rejections.

**Step 1:** The decision maker has specified some goals for the three objectives. The problem is solved in the ideal condition and the ideal solution for the three objectives, namely, minimizing price, lead time and quality is found. The target values or the goals are set to 90% of their ideal values. The target values for price, lead time and quality are 102933, 12867 and 21.8 respectively.

**Step 2:** The order in which the objectives are prioritized is shown below:

a. Minimum Cost
b. Minimum Lead Time
c. Minimum Percent of Rejections

**Step 3:** Analysis:

The principal set of indices used to denote the various entities such as customers, parts, suppliers, etc. are shown in Table 7.2:

Table 7.2

| INDEX | ENTITY | QUANTITY |
|---|---|---|
| i | Part | 2 |
| j | Buyer | 2 |
| k | Supplier | 2 |
| m | Incremental Price Break | 2 |

Parameters used in the problem are:

- $P_{ikm}$: Cost of acquiring unit of part *i* from supplier *k* at price level *m*, $\forall$ i, k, m = 0, 1
- $F_k$: Fixed Cost associated with each supplier, $\forall$ k = 0, 1
  $F_1 = 3500$
  $F_2 = 3600$
- $d_{ij}$: Demand of part *i* by buyer *j*, $\forall$ i & j = 0, 1
  $d_{00} = 150$

$d_{01} = 175$

$d_{10} = 200$

$d_{11} = 180$

- $q_{ik}$: Quality that supplier maintains for part *i,*

  $q_{00} = 0.03$

  $q_{01} = 0.09$

  $q_{10} = 0.06$

  $q_{11} = 0.02$

- $l_{ijk}$: Lead time of supplier *k* to produce and supply part *i* to buyer *j,*

  $l_{000} = 15$

  $l_{010} = 17$

  $l_{001} = 19$

  $l_{011} = 18$

  $l_{100} = 24$

  $l_{110} = 21$

  $l_{101} = 11$

  $l_{111} = 12$

- $CAP_k$: Production capacity of supplier *k* for part *i,*

  $CAP_0 = 300 \quad \forall \quad i = 0$

  $CAP_0 = 350 \quad \forall \quad i = 1$

  $CAP_1 = 280 \quad \forall \quad i = 0$

  $CAP_1 = 360 \quad \forall \quad i = 1$

- $b_{ikm}$: Quantity at which price break occurs for part *i* given by supplier *k* for buyer *j,*

  $b_{000} = 85 \quad \forall \quad j = 0, 1$

  $b_{010} = 95 \quad \forall \quad j = 0, 1$

  $b_{001} = 180 \quad \forall \quad j = 0, 1$

  $b_{011} = 190 \quad \forall \quad j = 0, 1$

  $b_{100} = 120 \quad \forall \quad j = 0, 1$

  $b_{110} = 125 \quad \forall \quad j = 0, 1$

  $b_{101} = 210 \quad \forall \quad j = 0, 1$

$b_{111} = 205 \quad \forall \quad j = 0, 1$

- $L_{ij}$: Lead time that buyer *j* requires for part *i,*

  $L_{00} = 18$

  $L_{01} = 20$

  $L_{10} = 26$

  $L_{11} = 13$

- $Q_j$: Quality level that buyer *j* requires all suppliers to maintain,

  $Q_0 = 0.095$

  $Q_1 = 0.09$

- N: Maximum number of suppliers that can be selected

Decision variables used in the model are:

- $X_{ijkm}$: Number of units of part *i* supplied by supplier *k* to buyer *j* at price level *m*
- $Z_k$: It is a binary variable which denotes whether a supplier is selected or not
- $Y_{ijkm}$: Also a binary variable which denotes which price level is selected

The problem consists of three objectives:

1. To minimize the total purchasing cost:

   Total Variable Cost: $\sum_i \sum_j \sum_k \sum_m P_{ikm} \cdot X_{ijkm}$

   Fixed Cost: $\sum_k F_k \,.\, Z_k$

   Hence, MIN $\sum_i \sum_j \sum_k \sum_m P_{ikm} \cdot X_{ijkm} + \sum_k F_k \,.\, Z_k$

2. To minimize the lead time:

   MIN $\sum_i \sum_j \sum_k \sum_m l_{ijk} \cdot X_{ijkm}$

3. To minimize the number of part rejections:

   $\text{MIN} \sum_i \sum_j \sum_k \sum_m q_{ik} \cdot X_{ijkm}$

The problem has the following constraints:

1. Capacity Constraint: Each supplier *k* has maximum capacity $CAP_k$,

   $\sum_i \sum_j \sum_m X_{ijkm} \leq (CAP_k)\, Z_k \qquad \forall \qquad k = 0, 1$

2. Demand Constraint: The demand of buyer *j* for part *i* has to be satisfied.

   $\sum_k \sum_m X_{ijkm} = d_{ij} \qquad \forall \qquad i,j$

3. Maximum number of selected suppliers: Maximum number of selected suppliers should be less than the number of participating suppliers

   $\sum_k Z_k \leq N$

4. Linearizing constraints:

   $X_{ijkm} \leqslant (b_{ikm} - b_{ikm-1}) * Y_{ijkm} \quad \forall i, j, k, \;\; 1 \leqslant m \leqslant m_k,$

   $X_{ijkm} \geqslant (b_{ikm} - b_{ikm-1}) * Y_{ijkm+1} \quad \forall i, j, k, \;\; 1 \leqslant m \leqslant m_k - 1.$

**Iteration 1:**

MIN $d_1^+$

SUBJECT TO

$\sum_i \sum_j \sum_m X_{ijkm} \leq (CAP_k)\, Z_k$

$\sum_k \sum_m X_{ijkm} = d_{ij}$

$\sum_k Z_k \leq N$

$\sum_i \sum_j \sum_k \sum_m P_{ikm} \cdot X_{ijkm} + \sum_k F_k \,.\, Z_k + d_1^- - d_1^+ = 102933$

**Iteration 2:**

MIN $d_2^+$

SUBJECT TO

$\sum_i \sum_j \sum_k \sum_m P_{ikm} \cdot X_{ijkm} + \sum_k F_k \cdot Z_k \leq 102933$

$\sum_i \sum_j \sum_m X_{ijkm} \leq (CAP_k)\, Z_k$

$\sum_k \sum_m X_{ijkm} = d_{ij}$

$\sum_k Z_k \leq N$

$\sum_i \sum_j \sum_k \sum_m l_{ijk} \cdot X_{ijkm} + d_2^- - d_2^+ = 12867$

**Iteration 3:**

MIN $d_3^+$

SUBJECT TO

$\sum_i \sum_j \sum_k \sum_m P_{ikm} \cdot X_{ijkm} + \sum_k F_k \cdot Z_k \leq 102933$

$\sum_i \sum_j \sum_k \sum_m l_{ijk} \cdot X_{ijkm} \leq 12867$

$\sum_i \sum_j \sum_m X_{ijkm} \leq (CAP_k)\, Z_k$

$\sum_k \sum_m X_{ijkm} = d_{ij}$

$\sum_k Z_k \leq N$

$\sum_i \sum_j \sum_k \sum_m q_{ik} \cdot X_{ijkm} + d_3^- - d_3^+ = 21.8$

**Solution:**

The solution according to the paper “Supplier Selection in Outsourcing” by Wadhwa and Ravindran (2007) is as follows,

1. Price = USD $93040
2. Lead Time = 12155 days
3. Quality = 33.55 %

### 7.4 Evolutionary Algorithm Method

Evolutionary algorithms are optimization algorithms that use the Darwinian theory of natural selection as the basis for optimization (Pinto). Evolution, which is a result of natural selection, is an optimization method which has had the luxury of having many years to complete its optimization or at least reach some kind of stable equilibrium. Evolutionary algorithms mimicking this behavior were first thought of for use in optimization by Prof. John H. Holland of the University of Michigan at Ann Arbor (Pinto).

The evolutionary algorithm mimics nature's evolutionary principles to drive its search toward an optimal solution. Since a number of individuals are processed for each generation, the outcome of an evolutionary algorithm is also a population of solutions. If the optimization problem has a single optimum, all evolutionary algorithm population individuals can be expected to converge to that optimum. This ability to find multiple optimal solutions in one single simulation run makes evolutionary algorithms suitable in solving multi-objective optimization problems. Evolutionary algorithms are reported to give robust results (Pinto).

A considerable number of evolutionary algorithms have been proposed in the last few years (Belgasmi et al. 11). Some of them are genetic algorithms, evolutionary strategies, genetic programming and evolutionary programming (Pinto). Genetic algorithms are iterative and require a certain number of iterations to converge to the optimal solution. The basic working principle/logic of a genetic algorithm is explained below (Pinto):

a. Create random population of 'n' individuals.
b. These solutions are then compared and evaluated against the fitness function.
c. Create new members for the next population using the reproductive operators: crossover and mutation.

d. If the non-dominated set falls below the pre-specified level then there is no need for increase in population size.

Crossover combines two or more individuals to create a new individual while mutation is performed on a single parent by mutating one or more parameters. Crossover and mutation are the diversity operators that bring diversity to the present population. Research in multi-objective genetic algorithms came about with the development of the Vector Evaluated Genetic Algorithm (VEGA) and the Multi-Objective Genetic Algorithm (MOGA). Later on the Non-dominated Sorting Genetic Algorithm (NSGA) was presented by Srinivas and Deb in 1994 and in 2002 NSGA-II was being developed by Deb et al.

Evolutionary algorithms are sometimes called genetic algorithms (Day). Basic steps described by Day are:

a. Each objective function is solved and its value is determined.
b. Pairs of individuals are selected to reproduce.
c. Individuals are forced to undergo Crossover.
d. Mutation is performed on the individuals.
e. Fitness test is performed on the children produced.
f. Steps are repeated until the destination is reached.

**Example:**

The following example is illustrated and used in the paper "Supply Chain Optimization using Multi-Objective Evolutionary Algorithms" by Pinto.

Pinto has implemented the Non-dominated Sorting Genetic Algorithm-II in a three stage supply chain problem. The three stages are:

a. Supplier
b. Plant
c. Customer Zone

The decision maker has various objectives to be achieved such as minimizing the total operating cost, manufacturing cost, transportation cost and maximizing the profit, revenue. The problem is formulated in such a way that two objective functions are clubbed together in each set to form 4 sets in total and constraints are selected depending on the set of objective functions used.

Principal set of indices used to denote the various entities such as components, suppliers, plants, etc. are as shown in Table 7.3:

Table 7.3

| INDEX | ENTITY | QUANTITY |
|---|---|---|
| i | Component | 3 |
| j | Supplier | 5 |
| k | Plant | 3 |
| m | Customer Zone | 4 |

Parameters used in the problem are:

- $L_{ij}$: Capacity of supplier *j* for component *i*
- $CS_{ij}$: Cost of making a component *i* by supplier *j*
- $STC_{ijk}$: Transportation Cost of component *i* from supplier *j* to plant *k* / unit
- $U_k$: Capacity of plant *k*
- $LC_k$: Labor Cost of plant *k* / unit
- $MC_k$: Manufacturing Cost of plant *k* / unit
- $IC_k$: Inventory Cost of plant *k* / unit
- $PTC_{kl}$: Plant Transportation Cost from plant *k* to customer zone *l* / unit
- $D_l$: Demand at customer zone *l*
- $SP_l$: Selling price at customer zone *l* / unit
- $S_{ij}$: Binary variable denoting whether component *i* can be supplied by supplier *j* or not.

Decision variables used in the model are:

- $X_{ijk}$: Number of components *i* from supplier *j* to plant *k*
- $Y_{kl}$: Amount of products shipped from plant *k* to customer zone *l*

- $I_{ik}$: Inventory of component *i* at plant *k*

**Objective Functions:**

**Set 1:** MIN Total Operating Cost (TOC)

MIN Manufacturing Cost (MC) / Total Operating Cost (TOC)

**Set 2:** MAX Profit

MIN Manufacturing Cost (MC)

**Set 3:** MAX Revenue

MIN Manufacturing Cost (MC)

**Set 4:** MAX Revenue

MIN Transportation Cost (TC)

$$TC = \sum_i \sum_j \sum_k X_{ijk} \cdot S_{ij} \cdot STC_{ijk} + \sum_k \sum_l Y_{kl} \cdot PTC_{kl}$$

$$\text{Total MC (TMC)} = \sum_k LC_k + MC_k + IC_k$$

$$\text{Supplier Cost (SC)} = \sum_i \sum_j CS_{ij} \cdot S_{ij} \cdot X_{ijk}$$

$$TOC = TC + TMC + SC$$

**Constraints:**

1. Capacity Constraint:
    a. Plant Capacity: $\sum_l Y_{kl} \leq U_k \quad \forall \quad k = 0, 1, 2$
    b. Supplier Capacity: $\sum_k S_{ij} \cdot X_{ijk} = L_{ij} \quad \forall \quad i, j$

2. Inventory Balancing Constraint:

a. Component 1: $\sum_j S_{1j} \cdot X_{1jk} = \sum_l Y_{kl} + I_{1k}$ $\forall$ k = 0, 1, 2

b. Component 2: $\sum_j S_{2j} \cdot X_{2jk} = \sum_l Y_{kl} + I_{2k}$ $\forall$ k = 0, 1, 2

c. Component 3: $\sum_j S_{3j} \cdot X_{3jk} = \sum_l Y_{kl} + I_{3k}$ $\forall$ k = 0, 1, 2

**Solution:**

The solution according to the paper "Supply Chain Optimization using Multi-Objective Evolutionary Algorithms" (Pinto) is as follows,

**Set 1**

| | **Objective Function** | **Corresponding Value** |
|---|---|---|
| **Statistic** | **TOC** | **MC/TOC** |
| Max | 175129.609375 | 0.327840 |
| Min | 51477.074219 | 0.384800 |
| Mean | 107536.784915 | |
| Std. Dev | 25380.061538 | |
| 95% Con | 701.175982 | |
| | **MC/TOC** | **TOC** |
| Max | 0.494218 | 111098.085938 |
| Min | 0.168562 | 87393.148438 |
| Mean | 0.356052 | |
| Std. Dev | 0.051778 | |
| 95% Con | 0.004525 | |

**Set 2**

| | Objective Function | Corresponding Value |
|---|---|---|
| **Statistic** | **Profit** | **MC** |
| Max | 280994.187500 | 129756.703125 |
| Min | 52039.582031 | 37570.750000 |
| Mean | 188021.082475 | |
| Std. Dev | 45594.950157 | |
| 95% Con | 1260.279687 | |
| | **MC** | **Profit** |
| Max | 135679.109375 | 267313.718750 |
| Min | 35442.843750 | 53814.980469 |
| Mean | 87696.929625 | |
| Std. Dev | 18810.017051 | |
| 95% Con | 519.9234197 | |

**Set 3**

| | Objective Function | Corresponding Value |
|---|---|---|
| **Statistic** | **Revenue** | **MC** |
| Max | 464739.875000 | 116392.10156300 |
| Min | 284256.718750 | 69288.17968800 |
| Mean | 386805.191991 | |
| Std. Dev | 39446.818147 | |
| 95% Con | 1052.114310 | |
| | **MC** | **Revenue** |
| Max | 116392.101563 | 464739.87500000 |
| Min | 69288.179688 | 284256.71875000 |
| Mean | 95844.675456 | |
| Std. Dev | 10250.941854 | |
| 95% Con | 273.4102021 | |

**Set 4**

| | Objective Function | Corresponding Value |
|---|---|---|
| **Statistic** | **Revenue** | **TC** |
| Max | 497068.62500000 | 80233.38281300 |
| Min | 308609.46875000 | 47546.85546900 |
| Mean | 436037.48765828 | |
| Std. Dev | 36426.07053919 | |
| 95% Con | 1032.84958016 | |
| | **TC** | **Revenue** |
| Max | 82803.61718800 | 493821.8750000 |
| Min | 47546.85546900 | 308609.4687500 |
| Mean | 66736.92726940 | |
| Std. Dev | 6267.46296005 | |
| 95% Con | 177.71190774 | |

### 7.5 STEM Method

A series of similar methods have been proposed by various authors. The STEM method, progressive orientation procedure, and the method of constraints are for solution of multiple objective linear programming (MOLP) problems (Hwang et al. 170).

The STEM method falls under the "Progressive articulation of Preference Information" category. The trade-off information is implicitly specified to the analyst by the decision maker. Trade-off information is the ratio of change in one objective function to the change in another objective function. In the STEM method, multiple phases of computation and decision making are interactive. Hence it allows the decision maker to recognize good solutions and the relative importance of the objectives. A pay-off table (usually a set of solutions in which one of the objectives is at its optimum) is constructed before the first interactive cycle and the best solution is found from it using a min-max strategy where the objective is to minimize the maximum deviation of an objective from its optimal solution. This step is analogous to the global criterion method where no articulation of preference information is needed. At the $m^{th}$ cycle, the feasible solution is presented to the decision maker, which is the closest solution to the optimal value. The decision maker then compares this value to the ideal solution. If further iterations are required, the decision maker needs to relax some satisfactory objectives in order to improve unsatisfactory ones. The process cycle continues until the decision maker is satisfied with the solution.

The STEM process follows these steps (Hwang et al. 170):

**Step 0:** Construction of a pay-off table:
A pay-off table, for example Table 7.4, is constructed before the first interactive cycle.

Table 7.4

| | f1 | f2 | X1 | X2 |
|---|---|---|---|---|
| **f1** | **130** | **100** | **100** | **300** |
| **f2** | **100** | **250** | **250** | **0** |

Where f1, f2 are the objective functions and X1, X2 are the decision variables. The values corresponding to these variables are the constants which are obtained if a problem containing two objectives is solved.

**Step 1:** Calculation phase:

The feasible solution at the $m^{th}$ cycle, which is the closest solution to the optimal value, is sought in this phase. The distances to the optimal solution are reduced as much as possible. The weights are defined such that the sum of the weights is 1. This means that different solutions obtained from different weighting strategies can be easily compared.

**Step 2:** Decision phase:

The compromise solution is presented to the decision maker. The decision maker then compares this value to the ideal solution. If further iterations are required, the decision maker needs to relax some satisfactory objectives in order to improve unsatisfactory ones. The process cycle continues until the decision maker is satisfied with the solution.

**Example:**

STEM method has not been applied to any supply chain problem as far as the scope of literature search for this thesis study is concerned. The STEM method application is explained in Appendix 2 with the Hardee Toy example illustrated in the book "Multiple Objective Decision Making – Methods and Applications" by Hwang et al.

# CHAPTER 8

# A REPRESENTATIVE SUPPLIER SELECTION PROBLEM

## 8. A REPRESENTATIVE SUPPLIER SELECTION PROBLEM

Following is a supplier selection problem synthesized from the example in “Vendor Selection in Outsourcing” (Wadhwa and Ravindran). This is a problem in which buyers have to select the suppliers based on the various criteria that buyers decide upon. Often in the supplier selection problem, buyers are in a dilemma due to the volume discounts offered by the suppliers, which depend on the volume of the order placed. A total of four criteria are considered in this study, three namely “price,” “lead time,” and “quality” were considered in the example in “Vendor Selection in Outsourcing” by Wadhwa and Ravindran (3729) and one, “transportation cost” was added by us in this supplier selection problem. The criteria considered in this problem for the selection of the potential suppliers are:

**Price:** Total cost of the purchasing of the parts from the suppliers consists of two factors, Fixed Cost and Variable Cost.

**Lead Time:** Lead time is the summation of the product of lead time for each part and the quantity of the parts being ordered. Lead time is measured in terms of days.

**Quality:** This means the quality of the parts supplied by various suppliers, which is measured by keeping a log of the number of rejections of received parts. It is measured in terms of percent of rejections.

**Transportation Cost:** It is the summation of the product of transportation cost/ unit and the quantity of the parts being ordered.

Principal set of indices used to denote the various entities such as customers, parts, suppliers, etc. are as shown in Table 8.1:

Table 8.1

| INDEX | ENTITY | QUANTITY |
|---|---|---|
| i | Part | 2 |
| j | Buyer | 2 |
| k | Supplier | 2 |
| m | Incremental Price Break | 2 |

The incremental price breaks can be multiple, but as suggested in the research (Wadhwa and Ravindran 3732) only 2 price break levels have been considered. The logic incorporated into the program is such that, it eliminates the need for the index 'm' (Price Break).

Parameters used in the program are:

- $P_{ijk}$: Buyer j's cost of acquiring unit of part *i* from supplier *k* at price level *m*, $\forall$ i, j, k = 0, 1

Table 8.2

| Product | Supplier | Limit 1 | Price 1 | Limit 2 | Price 2 |
|---|---|---|---|---|---|
| 0 | 0 | 85 | 125.40 | 180 | 115.00 |
| 0 | 1 | 95 | 128.30 | 190 | 120.00 |
| 1 | 0 | 120 | 150.35 | 210 | 130.00 |
| 1 | 1 | 125 | 148.75 | 205 | 122.00 |

- $F_k$: Fixed Cost associated with each supplier, $\forall$ k = 0, 1

  $F_1 = 3500$

  $F_2 = 3600$

- $d_{ij}$: Demand of part *i* by buyer *j*, $\forall$ i & j = 0, 1

  $d_{00} = 150$

  $d_{01} = 175$

  $d_{10} = 200$

  $d_{11} = 180$

- $q_{ik}$: Quality that supplier maintains for part *i*

  $q_{00} = 0.03$

  $q_{01} = 0.09$

  $q_{10} = 0.06$

  $q_{11} = 0.02$

- $l_{ijk}$: Lead time of supplier *k* to produce and supply part *i* to buyer *j,* denoted as LSB[i][j][k] in the program

  $LSB_{000} = 15$

  $LSB_{010} = 17$

  $LSB_{001} = 19$

  $LSB_{011} = 18$

  $LSB_{100} = 24$

  $LSBl_{110} = 21$

  $LSB_{101} = 11$

  $LSB_{111} = 12$

- $CAP_k$: Production capacity of supplier *k* for part *i,*

  $CAP_0 = 300 \quad \forall \quad i = 0$

  $CAP_0 = 350 \quad \forall \quad i = 1$

  $CAP_1 = 280 \quad \forall \quad i = 0$

  $CAP_1 = 360 \quad \forall \quad i = 1$

- $b_{ikm}$: Quantity at which price break occurs for part *i* given by supplier *k* for buyer *j,*

  $b_{000} = 85 \quad \forall \quad j = 0, 1$

  $b_{010} = 95 \quad \forall \quad j = 0, 1$

  $b_{001} = 180 \quad \forall \quad j = 0, 1$

  $b_{011} = 190 \quad \forall \quad j = 0, 1$

  $b_{100} = 120 \quad \forall \quad j = 0, 1$

  $b_{110} = 125 \quad \forall \quad j = 0, 1$

  $b_{101} = 210 \quad \forall \quad j = 0, 1$

  $b_{111} = 205 \quad \forall \quad j = 0, 1$

- $L_{ij}$: Lead time that buyer *j* requires for part *i,*

  $L_{00} = 18$

  $L_{01} = 20$

  $L_{10} = 26$

  $L_{11} = 13$

- $Q_j$: Quality level that buyer *j* requires all suppliers to maintain,

  $Q_0 = 0.095$

  $Q_1 = 0.09$

- maxN: Maximum number of suppliers that can be selected
- $T_{ijk:}$ Transportation Cost "Tijkm" that buyer j pays for ordering the part i from supplier k at price break level m

Table 8.3

| **Product** | **Buyer** | **Supplier** | **Limit 1** | **T_Cost 1** | **Limit 2** | **T_Cost 2** |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 85 | 30.00 | 180 | 25.00 |
| 0 | 0 | 1 | 95 | 35.00 | 190 | 28.00 |
| 0 | 1 | 0 | 85 | 40.00 | 180 | 37.00 |
| 0 | 1 | 1 | 95 | 32.00 | 190 | 25.00 |
| 1 | 0 | 0 | 120 | 35.00 | 210 | 30.00 |
| 1 | 0 | 1 | 125 | 40.00 | 205 | 33.00 |
| 1 | 1 | 0 | 120 | 45.00 | 210 | 42.00 |
| 1 | 1 | 1 | 125 | 37.00 | 205 | 30.00 |

**Note:** The price and the transportation cost values are arbitrarily chosen and are not taken from the paper.

Decision variables used in the model are:

- $X_{ijk}$: Number of units of part *i* supplied by supplier *k* to buyer *j* at price level *m*
- $Z_k$: It is a binary variable which denotes whether a supplier is selected or not
- $Y_{ijk}$: Also a binary variable which denotes which price level is selected

The problem consists of four objectives:

1. To minimize the total purchasing cost:

   Total Variable Cost: $\sum_i \sum_j \sum_k P_{ijk} \cdot X_{ijk}$

   Fixed Cost: $\sum_k F_k \,.\, Z_k$

   Hence, MIN $\sum_i \sum_j \sum_k P_{ijk} \cdot X_{ijk} + \sum_k F_k \,.\, Z_k$

2. To minimize the lead time:

   MIN $\sum_i \sum_j \sum_k LSB_{ijk} \cdot X_{ijk}$

3. To minimize the number of part rejections:

   MIN $\sum_i \sum_j \sum_k q_{ik} \cdot X_{ijk}$

4. To minimize the transportation cost:

   MIN $\sum_i \sum_j \sum_k T_{ijk} \cdot X_{ijk}$

The problem has the following constraints:

1. Capacity Constraint: Each supplier *k* has maximum capacity $CAP_k$,

   $\sum_i \sum_j \sum_m X_{ijkm} \leq (CAP_k)\, Z_k \qquad \forall \qquad k = 0, 1$

2. Demand Constraint: The demand of buyer *j* for part *i* has to be satisfied.

   $\sum_k \sum_m X_{ijkm} = d_{ij} \qquad \forall \qquad i,j$

3. Maximum number of selected suppliers: Maximum number of selected suppliers should be less than the number of participating suppliers

   $\sum_k Z_k \leq N$

4. Linearizing constraints:

$$X_{ijkm} \leqslant (b_{ikm} - b_{ikm-1}) * Y_{ijkm} \quad \forall i, j, k, \ \ 1 \leqslant m \leqslant m_k,$$

$$X_{ijkm} \geqslant (b_{ikm} - b_{ikm-1}) * Y_{ijkm+1} \quad \forall i, j, k, \ \ 1 \leqslant m \leqslant m_k - 1.$$

# CHAPTER 9

# A HEURISTIC MULTI-OBJECTIVE METHODOLOGY FOR SUPPLIER SELECTION PROBLEM

# 9. A HEURISTIC MULTI-OBJECTIVE METHODOLOGY FOR SUPPLIER SELECTION PROBLEM

## 9.1 Overview

As mentioned earlier, there is no mathematical optimal solution to a multi-objective problem. The final solution to the multi-objective problem is the one which is selected by the decision maker from a small set of non-dominated solutions. A non-dominated solution is the one in which no objective function can be improved without degrading simultaneously at least one of the other objective functions.

The proposed methodology uses the concepts of evolutionary methods (described in earlier chapters) to generate an initial set of non-dominated solutions. Instead of solving numerous single objective problems to yield these non-dominated solutions, we have developed a heuristic algorithm in which feasible solutions are generated with random values to all decision variables. Using the mutation and crossover principles, later in the document, these are converted to non-dominated solutions.

## 9.2 Description & Implementation

The optimization tools like LINDO, TORA, etc. can solve only one linear optimization problem at a time. These optimization tools cannot solve non-linear problems. Even the problems with the linearizing method will be tedious to solve using these tools. No tool exists to solve multi-objective problems as each is a unique methodology. Hence, the tools can be used only when one objective needs to be optimized. A number of values are experimented for the decision variable in multi-objective problem experimentation, although the problem is linear and one objective is optimized at a time. Hence, with LINDO or TORA the problem would have to be solved 'n' number of times. All these problems led to the decision for using C++ program to develop a heuristic algorithm named "Integrated Evolutionary Goal Trade-off (IEGT) Method" to generate a population of solutions, solve non-linear problems with the help of linearizing method, and solve multiple objectives.

The heuristic algorithm is based on generating random solutions that are feasible. In the context of a single objective optimization, such a population of solutions can be used to obtain a near optimal solution. Though, theoretically one cannot guarantee that this will yield an optimal solution, the algorithm offers an efficient way to generate near optimal solutions.

The concept for generating the 'n' number of solutions (population of solutions) is taken from the Evolutionary Algorithm method of optimization. Using a random function in the C++ program, one can generate 'n' number of solutions within a few minutes. The method of generating solutions using C++ program is much more efficient as compared to any other optimization tool.

In the context of multi-objective problems, there exists no algorithm to yield an ideal solution. Most of the solution methodologies depend on single objective optimization problems. As mentioned earlier, the multi-objective problems are solved via generating one or more non-dominated (or superior) solutions and help the decision

maker in choosing the final solution. The proposed algorithm is an attempt to integrate the strengths of many solution methodologies in yielding an efficient, streamlined solution methodology.

It was decided to generate a population of 300,000 feasible solutions, which is a fairly large number of solutions for analysis. It would have taken months to generate this many solutions with any other method or an optimization tool. It is only due to the efficiency of the program that it can generate this many solutions within a few minutes of computing.

These 300,000 feasible solutions (solutions that meet all stated constraints) are generated at which each solution is in a multi-objective space. The algorithm removes all dominated (or inferior) solutions and presents a set of non-dominated solutions. One may recall that a non-dominated solution is one in which no objective can be improved without degrading at least one other objective.

The resulting non-dominated solution set can be too large for a decision maker to analyze systematically. The proposed algorithm uses a solution methodology of goal programming to reduce this set to a more manageable size. An interactive method (STEM) is proposed to yield a final solution utilizing this reduced solution set as a starting point.

### 9.2.1 PHASE I: Evolutionary Approach

The basic working principle/logic of an evolutionary algorithm is as explained below (Pinto):

- Create random population of ‘n’ individuals.
- These solutions are then compared and evaluated against the fitness function.
- Create new members for the next population using the reproductive operators: crossover and mutation.

- If the non-dominated set falls below the pre-specified level then there is no need for increase in population size.

In the "IEGT" method, the "general solution" function is called from the main function in which the solution is generated. In the general solution function, initially a solution corresponding to the product '0' is created and then for product '1' is created. When product 'i=0', supplier 'k=0' and the buyer 'j=0 or 1', total quantity of product '0' supplied by supplier '0' to buyers '0 and 1' is calculated through a random number generation method between 0 and 300, since the supplier '0' capacity for the product '0' is 300. The program logic forces the quantity 'X' to satisfy all the capacity and the demand constraints. The loop is repeated until the constraints are satisfied, and the random number 'X' is generated in such a way that it satisfies all the constraints. The logic is repeated for product '1'.

The "price matrix" function assigns different price values per unit of the product purchased to the variable 'P'. The products falling in the first or second price level are found out using the values of 'X' obtained from the general solution function and the known values of the limits for the price breaks. The price matrix is shown below in Table 9.1:

Table 9.1

| **Product** | **Supplier** | **Limit 1** | **Price 1** | **Limit 2** | **Price 2** |
|---|---|---|---|---|---|
| 0 | 0 | 85 | 125.40 | 180 | 115.00 |
| 0 | 1 | 95 | 128.30 | 190 | 120.00 |
| 1 | 0 | 120 | 150.35 | 210 | 130.00 |
| 1 | 1 | 125 | 148.75 | 205 | 122.00 |

Similarly, the values for the transportation cost per unit of the product shipped are also determined. The transportation cost matrix is shown in Table 9.2:

Table 9.2

| Product | Buyer | Supplier | Limit 1 | T_Cost 1 | Limit 2 | T_Cost 2 |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 85 | 30.00 | 180 | 25.00 |
| 0 | 0 | 1 | 95 | 35.00 | 190 | 28.00 |
| 0 | 1 | 0 | 85 | 40.00 | 180 | 37.00 |
| 0 | 1 | 1 | 95 | 32.00 | 190 | 25.00 |
| 1 | 0 | 0 | 120 | 35.00 | 210 | 30.00 |
| 1 | 0 | 1 | 125 | 40.00 | 205 | 33.00 |
| 1 | 1 | 0 | 120 | 45.00 | 210 | 42.00 |
| 1 | 1 | 1 | 125 | 37.00 | 205 | 30.00 |

**Note:** The price and the transportation cost values are arbitrarily chosen and are not taken from the article "Vendor Selection in Outsourcing" (Wadhwa and Ravindran).

The function "objective functions" evaluates the values for all the four objective functions based on the values determined for the variables 'X' (quantity), 'P' (price), 'LSB' (lead time), and 'T' (transportation cost) from the previous defined functions. The values generated for various variables and objective functions in a population of solutions were saved in array variables.

The solutions generated from a population of solutions are compared amongst each other, and the solution in which all the four objectives are inferior to any other is removed. Thus all the inferior solutions were removed which reduced the solution set from 300,000 to 118. The superior solution is also known as the non-dominated solution which means no objective could be improved without sacrificing some other objective. The crossover and the mutation steps used in the evolutionary algorithm method to obtain the non-dominated set is being taken care off in this program by using the comparison of all the values of the solutions.

The set of 118 solutions generated from the program is shown below:

| Solution No. | Original No. | Price | Lead-Time | Quality | Transportation Cost |
|---|---|---|---|---|---|
| 1 | 3461 | 92688.35 | 9923.00 | 24.85 | 21346.00 |
| 2 | 5871 | 92133.50 | 9861.00 | 21.45 | 22191.00 |
| 3 | 6130 | 92226.60 | 9862.00 | 21.87 | 22126.00 |
| 4 | 7412 | 92471.00 | 9925.00 | 26.91 | 20884.00 |
| 5 | 9886 | 92561.85 | 9946.00 | 26.55 | 20973.00 |
| 6 | 11284 | 92670.75 | 9979.00 | 23.61 | 21856.00 |
| 7 | 12539 | 92452.70 | 9924.00 | 22.89 | 21870.00 |
| 8 | 16626 | 92430.30 | 10031.00 | 21.91 | 22005.00 |
| 9 | 16830 | 92013.80 | 9903.00 | 20.91 | 22095.00 |
| 10 | 23670 | 92040.40 | 9867.00 | 21.03 | 22159.00 |
| 11 | 30751 | 92167.10 | 9945.00 | 21.25 | 22056.00 |
| 12 | 32446 | 92563.20 | 10008.00 | 26.89 | 20757.00 |
| 13 | 35363 | 92446.40 | 9963.00 | 22.51 | 21936.00 |
| 14 | 36696 | 91734.50 | 9855.00 | 19.65 | 22659.00 |
| 15 | 39022 | 92253.20 | 9854.00 | 21.99 | 22198.00 |
| 16 | 39954 | 93819.90 | 12294.00 | 36.01 | 19910.00 |
| 17 | 43949 | 92867.90 | 9997.00 | 24.63 | 21409.00 |
| 18 | 44475 | 91791.20 | 9821.00 | 19.73 | 22462.00 |
| 19 | 46760 | 94166.20 | 11892.00 | 34.59 | 19990.00 |
| 20 | 47119 | 93814.75 | 12244.00 | 35.11 | 20035.00 |
| 21 | 48302 | 92603.50 | 10035.00 | 26.65 | 20751.00 |
| 22 | 56301 | 92545.80 | 9879.00 | 23.31 | 22004.00 |
| 23 | 59258 | 93750.85 | 12328.00 | 35.87 | 19970.00 |
| 24 | 59463 | 94622.10 | 11620.00 | 33.73 | 20230.00 |
| 25 | 60477 | 92843.60 | 9941.00 | 24.19 | 21573.00 |
| 26 | 62134 | 92676.40 | 9985.00 | 25.13 | 21107.00 |
| 27 | 64370 | 92747.75 | 9920.00 | 24.19 | 21485.00 |
| 28 | 65105 | 91982.65 | 9938.00 | 20.33 | 22151.00 |
| 29 | 65381 | 92514.40 | 9987.00 | 26.93 | 20748.00 |
| 30 | 70030 | 94114.55 | 11985.00 | 34.73 | 20080.00 |
| 31 | 70605 | 92503.80 | 9974.00 | 26.13 | 20808.00 |
| 32 | 73296 | 93621.65 | 12380.00 | 34.37 | 20045.00 |
| 33 | 73707 | 93653.10 | 12291.00 | 34.71 | 20010.00 |
| 34 | 74302 | 91761.10 | 9842.00 | 19.77 | 22574.00 |
| 35 | 77829 | 92496.85 | 9936.00 | 26.77 | 20893.00 |
| 36 | 80313 | 93550.70 | 12361.00 | 35.59 | 19840.00 |
| 37 | 83729 | 92335.60 | 9995.00 | 28.83 | 20350.00 |
| 38 | 85970 | 92667.15 | 9962.00 | 24.67 | 21185.00 |
| 39 | 87124 | 92776.30 | 9915.00 | 24.73 | 21517.00 |
| 40 | 89049 | 91734.50 | 9921.00 | 19.65 | 22593.00 |
| 41 | 89601 | 92574.15 | 9920.00 | 23.35 | 21915.00 |
| 42 | 95063 | 92012.75 | 9963.00 | 20.29 | 22137.00 |
| 43 | 95228 | 92563.75 | 10011.00 | 26.65 | 20777.00 |
| 44 | 105229 | 92403.00 | 9885.00 | 22.49 | 22124.00 |
| 45 | 106028 | 92647.25 | 9998.00 | 25.93 | 20995.00 |
| 46 | 106064 | 92293.20 | 10001.00 | 28.47 | 20262.00 |
| 47 | 107736 | 92740.20 | 9929.00 | 24.75 | 21433.00 |
| 48 | 109388 | 91912.65 | 9921.00 | 20.19 | 22169.00 |
| 49 | 112229 | 92530.80 | 9974.00 | 27.25 | 20797.00 |
| 50 | 112417 | 92641.85 | 9967.00 | 25.99 | 21084.00 |
| 51 | 114769 | 92753.15 | 9891.00 | 24.13 | 21591.00 |

| | | | | | |
|---|---|---|---|---|---|
| 52 | 115034 | 92441.15 | 9922.00 | 22.75 | 21926.00 |
| 53 | 115895 | 92680.05 | 9979.00 | 25.15 | 21140.00 |
| 54 | 116816 | 93714.95 | 12487.00 | 36.71 | 19760.00 |
| 55 | 117349 | 92776.30 | 9955.00 | 24.73 | 21387.00 |
| 56 | 119598 | 92687.40 | 10013.00 | 25.51 | 21011.00 |
| 57 | 127012 | 92607.75 | 9917.00 | 25.33 | 21202.00 |
| 58 | 130393 | 91894.10 | 9885.00 | 20.37 | 22095.00 |
| 59 | 131777 | 92293.10 | 9911.00 | 22.17 | 21958.00 |
| 60 | 131946 | 91880.80 | 9869.00 | 20.31 | 22223.00 |
| 61 | 131985 | 91734.50 | 9842.00 | 19.65 | 22231.00 |
| 62 | 134271 | 92758.55 | 9942.00 | 24.07 | 21437.00 |
| 63 | 146580 | 92349.70 | 9969.00 | 28.05 | 20483.00 |
| 64 | 148728 | 92703.40 | 10126.00 | 27.67 | 20568.00 |
| 65 | 149124 | 93840.75 | 12084.00 | 34.95 | 19850.00 |
| 66 | 149931 | 92717.85 | 9960.00 | 24.73 | 21284.00 |
| 67 | 150271 | 92403.30 | 10008.00 | 27.87 | 20461.00 |
| 68 | 151990 | 92368.40 | 9960.00 | 28.05 | 20547.00 |
| 69 | 153595 | 92503.40 | 10004.00 | 27.97 | 20628.00 |
| 70 | 157994 | 91761.10 | 9817.00 | 19.77 | 22383.00 |
| 71 | 159372 | 92108.65 | 9905.00 | 21.25 | 22025.00 |
| 72 | 161612 | 92487.00 | 10094.00 | 29.07 | 20259.00 |
| 73 | 165699 | 92253.20 | 9851.00 | 21.99 | 22183.00 |
| 74 | 166855 | 92839.55 | 10093.00 | 26.01 | 20983.00 |
| 75 | 169772 | 93775.20 | 12363.00 | 35.57 | 19920.00 |
| 76 | 174061 | 92561.65 | 9964.00 | 26.05 | 20943.00 |
| 77 | 174797 | 92360.10 | 9992.00 | 28.35 | 20419.00 |
| 78 | 176395 | 92784.60 | 9999.00 | 24.43 | 21268.00 |
| 79 | 176783 | 93786.15 | 12256.00 | 33.91 | 20160.00 |
| 80 | 177825 | 92441.50 | 9976.00 | 27.03 | 20660.00 |
| 81 | 180027 | 92042.15 | 9855.00 | 20.95 | 22221.00 |
| 82 | 180678 | 92567.80 | 9932.00 | 26.25 | 21060.00 |
| 83 | 188612 | 93995.25 | 12022.00 | 34.43 | 20005.00 |
| 84 | 191092 | 92703.40 | 9900.00 | 24.83 | 21442.00 |
| 85 | 192618 | 93699.35 | 12329.00 | 35.59 | 19940.00 |
| 86 | 193498 | 92720.95 | 9995.00 | 24.99 | 21154.00 |
| 87 | 197348 | 92767.05 | 10053.00 | 25.69 | 21002.00 |
| 88 | 200493 | 93661.90 | 12238.00 | 33.91 | 20085.00 |
| 89 | 203625 | 92055.45 | 9841.00 | 21.01 | 22240.00 |
| 90 | 208299 | 94123.45 | 11915.00 | 34.91 | 19985.00 |
| 91 | 211021 | 94180.45 | 12023.00 | 34.25 | 20200.00 |
| 92 | 211353 | 91840.90 | 9830.00 | 20.13 | 22256.00 |
| 93 | 212013 | 91761.10 | 9823.00 | 19.77 | 22512.00 |
| 94 | 215441 | 91844.40 | 9816.00 | 19.97 | 22397.00 |
| 95 | 216104 | 92346.30 | 9888.00 | 22.41 | 22001.00 |
| 96 | 222044 | 91762.85 | 9827.00 | 19.69 | 22522.00 |
| 97 | 222949 | 91791.20 | 9817.00 | 19.73 | 22475.00 |
| 98 | 229435 | 94231.55 | 11834.00 | 34.39 | 20040.00 |
| 99 | 232342 | 91975.65 | 9879.00 | 20.65 | 22127.00 |
| 100 | 232763 | 92357.80 | 9988.00 | 28.67 | 20404.00 |
| 101 | 233891 | 91761.10 | 9870.00 | 19.77 | 22186.00 |
| 102 | 239695 | 92526.20 | 9973.00 | 22.87 | 21893.00 |
| 103 | 241009 | 91734.50 | 9878.00 | 19.65 | 22609.00 |
| 104 | 245662 | 92456.55 | 10062.00 | 28.43 | 20332.00 |
| 105 | 248899 | 91814.30 | 9828.00 | 20.01 | 22266.00 |
| 106 | 252864 | 92511.85 | 10036.00 | 27.85 | 20565.00 |
| 107 | 255342 | 92479.85 | 9997.00 | 27.79 | 20634.00 |
| 108 | 261068 | 92690.45 | 9994.00 | 25.45 | 21102.00 |

| | | | | | |
|---|---|---|---|---|---|
| 109 | 272597 | 92731.15 | 9904.00 | 24.79 | 21489.00 |
| 110 | 273436 | 94392.00 | 11639.00 | 33.67 | 20120.00 |
| 111 | 275617 | 92492.40 | 9996.00 | 27.59 | 20659.00 |
| 112 | 275911 | 92652.65 | 9969.00 | 25.87 | 21101.00 |
| 113 | 276727 | 91920.70 | 9881.00 | 20.49 | 22154.00 |
| 114 | 287601 | 92612.30 | 9934.00 | 23.61 | 21866.00 |
| 115 | 289343 | 92669.65 | 9940.00 | 24.85 | 21256.00 |
| 116 | 290687 | 92633.20 | 9941.00 | 25.61 | 21156.00 |
| 117 | 292093 | 92537.35 | 9985.00 | 27.03 | 20787.00 |
| 118 | 294218 | 91999.45 | 9972.00 | 20.23 | 22159.00 |

**Note:** The time required to generate 300,000 solutions and reduce it to 118 solutions was approximately 6 minutes on an Intel processor running Windows XP.

### 9.2.2 PHASE II: Goal Programming Approach

In the goal programming method, goals (target values) are set for the multiple objectives that are ranked according to the priorities in which they need to be achieved. The importance of these goals and the order in which they need to be achieved are articulated by the decision maker.

In the “IEGT” method application, when these 118 solutions are presented to the decision maker, he would not have insight to decide upon the final optimal solution from such a large set of solutions. Hence, to further reduce the set of solutions, the concept of goal programming (PHASE II) has been applied in the program. As per the concept of goal programming, the analyst should obtain the order in which the objectives are ranked by the decision maker and the target values for each objective function from the decision maker. In order to have the information about the target values, the minimum and maximum of each objective function value from the produced solution set is presented to the decision maker. The decision maker is then, required to assess the target values for each objective function. The sample of program output which consists of minimum, maximum and the target values for each objective suggested by the decision maker is shown below:

| | Minimum | Maximum | Target |
|---|---|---|---|
| Price | 91734.50 | 94622.10 | 93500.00 |
| Lead Time | 9816.00 | 12487.00 | 12000.00 |
| Quality | 19.65 | 36.71 | 30.00 |
| Transportation Cost | 19760.00 | 22659.00 | 22000.00 |

**Note:** As these are observed from a very large population of solutions generated randomly (no bias), the minimum values are treated as the mathematical minimum (or optimal solution).

The process of providing the minimum and maximum value to the decision maker and receiving the target values from the decision maker is carried out interactively in the program. Hence, when the decision maker specifies the target values for the objective functions, the 118 solutions are compared against these target values and the solution in which the target values for all the objective functions are met, is selected and presented to the decision maker. The decision maker is then asked if he is satisfied with the solution set generated. The decision maker may not be satisfied with the number of solutions generated and may want to reduce the solution set to a smaller set. In such a case, the program presents the decision maker with the minimum and maximum value of each objective function from the recently generated solution set, which gives the hint to the decision maker about the amount of variation in the reduced solution set and the limits of each objective function. This is illustrated from a sample of program output shown below:

**Iteration 1:**

| | Minimum | Maximum | Target |
|---|---|---|---|
| Price | 91734.50 | 94622.10 | 93500.00 |
| Lead Time | 9816.00 | 12487.00 | 12000.00 |
| Quality | 19.65 | 36.71 | 30.00 |
| Transportation Cost | 19760.00 | 22659.00 | 22000.00 |

```
Total number of solutions meeting the target values is: 62

Satisfied (Type 'Y' for Yes or 'N' for No): N
```

**Minimum and maximum presented for iteration 2:**

```
                              Minimum      Maximum

Price                         92293.10     92867.90
Lead Time                      9891.00     10126.00
Quality                          22.17        29.07
Transportation Cost           20259.00     21958.00
```

The decision maker then specifies tighter targets and reduces the number of solutions. The process continues until the decision maker is satisfied with the results. This is illustrated in the matrix which consists of the target values specified by the decision maker and the number of solutions meeting these target values, as shown below:

Table 9.3

| Sr. No. | OBJECTIVE FUNCTION TARGETS | | | | Number of Solutions meeting the Target |
|---|---|---|---|---|---|
| | PRICE TARGET | LEAD-TIME TARGET | QUALITY TARGET | TRANSPORTATION COST TARGET | |
| 0 | 94623 | 12487 | 37 | 22659 | 118 |
| 1 | 93500 | 12000 | 30 | 22000 | 62 |
| 2 | 92800 | 12000 | 30 | 22000 | 59 |
| 3 | 92800 | 10000 | 30 | 22000 | 47 |
| 4 | 92800 | 10000 | 28 | 22000 | 42 |
| 5 | 92800 | 10000 | 27 | 22000 | 37 |
| 6 | 92800 | 10000 | 27 | 21500 | 27 |
| 7 | 92700 | 10000 | 27 | 21500 | 18 |
| 8 | 92700 | 10000 | 26 | 21300 | 10 |

Similarly, the process is carried out if the decision maker is not satisfied with the number of solutions and wants to increase the number of solutions which can be done by relaxing the targets as shown below:

Table 9.4

| Sr. No. | OBJECTIVE FUNCTION TARGETS | | | | Number of Solutions meeting the Target |
|---|---|---|---|---|---|
| | PRICE TARGET | LEAD-TIME TARGET | QUALITY TARGET | TRANSPORTATION COST TARGET | |
| 0 | 94623 | 12487 | 37 | 22659 | 118 |
| 1 | 92000 | 12000 | 30 | 22000 | 0 |
| 2 | 92800 | 12000 | 30 | 22000 | 59 |

| | | | | | |
|---|---|---|---|---|---|
| **3** | **92800** | **10000** | **30** | **22000** | **47** |
| **4** | **92800** | **10000** | **28** | **22000** | **42** |
| **5** | **92800** | **10000** | **27** | **22000** | **37** |
| **6** | **92800** | **10000** | **27** | **21500** | **27** |
| **7** | **92700** | **10000** | **27** | **21500** | **18** |
| **8** | **92700** | **10000** | **26** | **21300** | **10** |

**Note:** The price target is too high in iteration 1 to generate the number of solutions, which is relaxed in iteration 2.

### 9.2.3 PHASE III: Progressive Articulation – STEM Approach

The STEM method falls under the "Progressive articulation of Preference Information" category. The trade-off information is implicitly specified to the analyst by the decision maker. The trade-off information is the ratio of change in one objective function to the change in another objective function. In the STEM method, the phases of computation and decision making are interactive. Hence it allows the decision maker to recognize good solutions and relative importance of the objectives. A pay-off table is constructed before the first interactive cycle and the best solution is found from it. At the $m^{th}$ cycle, the feasible solution is presented to the decision maker, which is the closest solution to the optimal value. The decision maker then compares this value to the ideal solution. If further iterations are required, the decision maker needs to relax some satisfactory objectives in order to improve unsatisfactory ones. The process cycle continues until the decision maker is satisfied with the solution.

Now in "IEGT" method example, after being presented with a handful of solutions from PHASE II, if the decision maker is not satisfied with any one of the solutions and wants to improve that solution, the concept of the STEM method (PHASE III) is being incorporated in the program, which is also interactive. The set of solutions obtained from PHASE II are presented to the decision maker. If the decision maker is not happy with any solution for all objective values and is willing to relax some of the objective functions, he may relax the target value for those objective functions and

check if the objective functions with which he is not satisfied are improved or not. The process is continued until the decision maker is satisfied with the solutions generated. The decision maker finally chooses the optimal solution from a smaller set of two to four solutions presented to him. This is illustrated from a sample of program output shown below:

**Solution presented to the decision maker from goal programming method:**

| | Minimum | Maximum | Target |
|---|---|---|---|
| Price | 91734.50 | 94622.10 | 92700.00 |
| Lead Time | 9816.00 | 12487.00 | 10000.00 |
| Quality | 19.65 | 36.71 | 26.00 |
| Transportation Cost | 19760.00 | 22659.00 | 21300.00 |

Solutions meeting the target values:

| Solution No. | Price | Lead-Time | Quality | Transportation Cost |
|---|---|---|---|---|
| 26 | 92676.40 | 9985.00 | 25.13 | 21107.00 |
| 38 | 92667.15 | 9962.00 | 24.67 | 21185.00 |
| 45 | 92647.25 | 9998.00 | 25.93 | 20995.00 |
| 50 | 92641.85 | 9967.00 | 25.99 | 21084.00 |
| 53 | 92680.05 | 9979.00 | 25.15 | 21140.00 |
| 57 | 92607.75 | 9917.00 | 25.33 | 21202.00 |
| 108 | 92690.45 | 9994.00 | 25.45 | 21102.00 |
| 112 | 92652.65 | 9969.00 | 25.87 | 21101.00 |
| 115 | 92669.65 | 9940.00 | 24.85 | 21256.00 |
| 116 | 92633.20 | 9941.00 | 25.61 | 21156.00 |

Total number of solutions meeting the target values is: 10

Satisfied (Type 'Y' for Yes or 'N' for No): N

**The decision maker wants to improve price (Function 1), lead-time (Function 2) or transportation cost (Function 4) by relaxing the quality function (Function 3) from 26 to 28.**

**Note:** The changed target values if compared with the previous targets are shown in bold letters in the following program output samples.

**Iteration 1:**

| | Minimum | Maximum | Target |
|---|---|---|---|
| Price | 92607.75 | 92690.45 | 92700.00 |
| Lead Time | 9917.00 | 9998.00 | 10000.00 |
| Quality | 24.67 | 25.99 | **28.00** |
| Transportation Cost | 20995.00 | 21256.00 | 21300.00 |

Solutions meeting the target values:

| Solution No. | Price | Lead-Time | Quality | Transportation Cost |
|---|---|---|---|---|
| 4 | 92471.00 | 9925.00 | 26.91 | 20884.00 |
| 5 | 92561.85 | 9946.00 | 26.55 | 20973.00 |
| 26 | 92676.40 | 9985.00 | 25.13 | 21107.00 |
| 29 | 92514.40 | 9987.00 | 26.93 | 20748.00 |
| 31 | 92503.80 | 9974.00 | 26.13 | 20808.00 |
| 35 | 92496.85 | 9936.00 | 26.77 | 20893.00 |
| 38 | 92667.15 | 9962.00 | 24.67 | 21185.00 |
| 45 | 92647.25 | 9998.00 | 25.93 | 20995.00 |
| 49 | 92530.80 | 9974.00 | 27.25 | 20797.00 |
| 50 | 92641.85 | 9967.00 | 25.99 | 21084.00 |
| 53 | 92680.05 | 9979.00 | 25.15 | 21140.00 |
| 57 | 92607.75 | 9917.00 | 25.33 | 21202.00 |
| 76 | 92561.65 | 9964.00 | 26.05 | 20943.00 |
| 80 | 92441.50 | 9976.00 | 27.03 | 20660.00 |
| 82 | 92567.80 | 9932.00 | 26.25 | 21060.00 |
| 107 | 92479.85 | 9997.00 | 27.79 | 20634.00 |
| 108 | 92690.45 | 9994.00 | 25.45 | 21102.00 |
| 111 | 92492.40 | 9996.00 | 27.59 | 20659.00 |
| 112 | 92652.65 | 9969.00 | 25.87 | 21101.00 |
| 115 | 92669.65 | 9940.00 | 24.85 | 21256.00 |
| 116 | 92633.20 | 9941.00 | 25.61 | 21156.00 |
| 117 | 92537.35 | 9985.00 | 27.03 | 20787.00 |

Total number of solutions meeting the target values is: 22

**The solution set is expanded by relaxing the objective function "quality" while keeping same targets for the rest of the objective functions. 13 solutions are better for "price" objective (< 92607) while 11 solutions are better for "transportation cost" objective (< 20995) and no solutions are better for the "lead time" objective if the values are compared with the minimum of iteration 1. The decision maker can now, tighten the other targets iteratively.**

**Iteration 2:**

| | Minimum | Maximum | Target |
|---|---|---|---|
| Price | 92441.50 | 92690.45 | **92600.00** |
| Lead Time | 9917.00 | 9998.00 | 10000.00 |
| Quality | 24.67 | 27.79 | 28.00 |
| Transportation Cost | 20634.00 | 21256.00 | **21000.00** |

Solutions meeting the target values:

| Solution No. | Price | Lead-Time | Quality | Transportation Cost |
|---|---|---|---|---|
| 4 | 92471.00 | 9925.00 | 26.91 | 20884.00 |
| 5 | 92561.85 | 9946.00 | 26.55 | 20973.00 |
| 29 | 92514.40 | 9987.00 | 26.93 | 20748.00 |
| 31 | 92503.80 | 9974.00 | 26.13 | 20808.00 |
| 35 | 92496.85 | 9936.00 | 26.77 | 20893.00 |
| 49 | 92530.80 | 9974.00 | 27.25 | 20797.00 |
| 76 | 92561.65 | 9964.00 | 26.05 | 20943.00 |
| 80 | 92441.50 | 9976.00 | 27.03 | 20660.00 |
| 107 | 92479.85 | 9997.00 | 27.79 | 20634.00 |
| 111 | 92492.40 | 9996.00 | 27.59 | 20659.00 |
| 117 | 92537.35 | 9985.00 | 27.03 | 20787.00 |

Total number of solutions meeting the target values is: 11

Satisfied (Type 'Y' for Yes or 'N' for No): N

All of these 11 solutions are better than the best of iteration 1 solutions for the "price" objective. Similarly, these 11 are better solutions for the "transportation cost" objective. These improvements were possible due to the relaxed requirement in the "quality" objective. This solution set can be further reduced in size by re-establishing the targets.

The corresponding values for the price/unit, transportation cost/unit, and number of units ordered for each product, buyer and supplier are also obtained in the program output. The program output for the values of price/unit, transportation cost/unit and number of units corresponding to the 11 solutions obtained in iteration 2 is shown below:

Solution No. **4**
Original No. 7412

| | i = PRODUCT j = BUYER k = SUPPLIER | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 148.0 | 2.0 | 31.0 | 144.0 | 2.0 | 198.0 | 18.0 | 162.0 |

Solution No. **5**
Original No. 9886

| | i = PRODUCT j = BUYER k = SUPPLIER | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 150.0 | 0.0 | 37.0 | 138.0 | 4.0 | 196.0 | 19.0 | 161.0 |

Solution No. **29**
Original No. 65381

| | i = PRODUCT j = BUYER k = SUPPLIER | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 149.0 | 1.0 | 31.0 | 144.0 | 14.0 | 186.0 | 8.0 | 172.0 |

Solution No. **31**
Original No. 70605

| | i = PRODUCT j = BUYER k = SUPPLIER | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 150.0 | 0.0 | 42.0 | 133.0 | 19.0 | 181.0 | 1.0 | 179.0 |

Solution No. **35**
Original No. 77829

| | i = PRODUCT j = BUYER k = SUPPLIER | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 149.0 | 1.0 | 33.0 | 142.0 | 4.0 | 196.0 | 17.0 | 163.0 |

Solution No. **49**
Original No. 112229

| | i = PRODUCT j = BUYER k = SUPPLIER | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 150.0 | 0.0 | 26.0 | 149.0 | 6.0 | 194.0 | 18.0 | 162.0 |

Solution No. **76**
Original No. 174061

| | i = PRODUCT j = BUYER k = SUPPLIER | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 149.0 | 1.0 | 45.0 | 130.0 | 14.0 | 186.0 | 7.0 | 173.0 |

Solution No. **80**
Original No. 177825

| | i = PRODUCT j = BUYER k = SUPPLIER | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 149.0 | 1.0 | 28.0 | 147.0 | 15.0 | 185.0 | 5.0 | 175.0 |

Solution No. **107**
Original No. 255342

| | i = PRODUCT j = BUYER k = SUPPLIER | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 145.0 | 5.0 | 20.0 | 155.0 | 12.0 | 188.0 | 9.0 | 171.0 |

Solution No. **111**
Original No. 275617

| | i = PRODUCT j = BUYER k = SUPPLIER | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 147.0 | 3.0 | 22.0 | 153.0 | 12.0 | 188.0 | 10.0 | 170.0 |

Solution No. **117**
Original No. 292093

i = PRODUCT j = BUYER k = SUPPLIER

| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
|---|---|---|---|---|---|---|---|---|
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 149.0 | 1.0 | 30.0 | 145.0 | 11.0 | 189.0 | 12.0 | 168.0 |

## Iteration 3:

| | Minimum | Maximum | Target |
|---|---|---|---|
| Price | 92441.50 | 92492.40 | **92500.00** |
| Lead Time | 9976.00 | 9997.00 | 10000.00 |
| Quality | 27.03 | 27.79 | 28.00 |
| Transportation Cost | 20634.00 | 20660.00 | 21000.00 |

Solutions meeting the target values:

| Solution No. | Price | Lead-Time | Quality | Transportation Cost |
|---|---|---|---|---|
| 4 | 92471.00 | 9925.00 | 26.91 | 20884.00 |
| 35 | 92496.85 | 9936.00 | 26.77 | 20893.00 |
| 80 | 92441.50 | 9976.00 | 27.03 | 20660.00 |
| 107 | 92479.85 | 9997.00 | 27.79 | 20634.00 |
| 111 | 92492.40 | 9996.00 | 27.59 | 20659.00 |

Total number of solutions meeting the target values is: 5

Satisfied (Type 'Y' for Yes or 'N' for No): N

Solution No. **4**
Original No. 7412

i = PRODUCT j = BUYER k = SUPPLIER

| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
|---|---|---|---|---|---|---|---|---|
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 148.0 | 2.0 | 31.0 | 144.0 | 2.0 | 198.0 | 18.0 | 162.0 |

Solution No. **35**
Original No. 77829

i = PRODUCT j = BUYER k = SUPPLIER

| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
|---|---|---|---|---|---|---|---|---|
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 149.0 | 1.0 | 33.0 | 142.0 | 4.0 | 196.0 | 17.0 | 163.0 |

Solution No. **80**
Original No. 177825

i = PRODUCT j = BUYER k = SUPPLIER

| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
|---|---|---|---|---|---|---|---|---|
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 149.0 | 1.0 | 28.0 | 147.0 | 15.0 | 185.0 | 5.0 | 175.0 |

Solution No. **107**
Original No. 255342

i = PRODUCT j = BUYER k = SUPPLIER

| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
|---|---|---|---|---|---|---|---|---|
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 145.0 | 5.0 | 20.0 | 155.0 | 12.0 | 188.0 | 9.0 | 171.0 |

Solution No. **111**
Original No. 275617

i = PRODUCT j = BUYER k = SUPPLIER

| | ijk 000 | ijk 001 | ijk 010 | ijk 011 | ijk 100 | ijk 101 | ijk 110 | ijk 111 |
|---|---|---|---|---|---|---|---|---|
| Price Level | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Price per unit | 115.0 | 128.3 | 125.4 | 120.0 | 150.35 | 122.0 | 150.35 | 122.0 |
| Transport Cost | 25.0 | 35.0 | 40.0 | 25.0 | 35.0 | 33.0 | 45.0 | 30.0 |
| Units (X) | 147.0 | 3.0 | 22.0 | 153.0 | 12.0 | 188.0 | 10.0 | 170.0 |

**From the above values, it is clearly seen that there is not a considerable improvement in the price, lead-time or transportation cost but there is an improvement in the quality. Hence, the decision maker may improve the quality again.**

**Iteration 4:**

| | Minimum | Maximum | Target |
|---|---|---|---|
| Price | 92441.50 | 92496.85 | 92500.00 |
| Lead Time | 9925.00 | 9997.00 | 10000.00 |
| Quality | 26.77 | 27.79 | **27.00** |
| Transportation Cost | 20634.00 | 20893.00 | 21000.00 |

Solutions meeting the target values:

| Solution No. | Price | Lead-Time | Quality | Transportation Cost |
|---|---|---|---|---|
| 4 | 92471.00 | 9925.00 | 26.91 | 20884.00 |
| 35 | 92496.85 | 9936.00 | 26.77 | 20893.00 |

Total number of solutions meeting the target values is: 2

Satisfied (Type 'Y' for Yes or 'N' for No): Y

```
Solution No. 4
Original No. 7412
                        i = PRODUCT j = BUYER k = SUPPLIER
                    ijk   ijk   ijk   ijk   ijk    ijk   ijk    ijk
                    000   001   010   011   100    101   110    111
Price Level          1     0     0     1     0      1     0      1
Price per unit    115.0 128.3 125.4 120.0 150.35 122.0 150.35 122.0
Transport Cost     25.0  35.0  40.0  25.0  35.0   33.0  45.0   30.0
Units (X)         148.0   2.0  31.0 144.0   2.0  198.0  18.0  162.0
```

| | Solution No. 4 | | | | Original No. 7412 | | | |
|---|---|---|---|---|---|---|---|---|
| | Product 0 | | | | Product 1 | | | |
| | Buyer 0 | Buyer 1 | Total | Capacity of Supplier | Buyer 0 | Buyer 1 | Total | Capacity of Supplier |
| Supplier 0 | 148 | 31 | 179 | 300 | 2 | 18 | 20 | 350 |
| Supplier 1 | 2 | 144 | 146 | 280 | 198 | 162 | 360 | 360 |
| Demand from each buyer | 150 | 175 | | | 200 | 180 | | |

```
Solution No. 35
Original No. 77829
                          i = PRODUCT j = BUYER k = SUPPLIER
                    ijk   ijk   ijk   ijk   ijk    ijk   ijk    ijk
                    000   001   010   011   100    101   110    111
Price Level          1     0     0     1     0      1     0      1
Price per unit    115.0 128.3 125.4 120.0 150.35 122.0 150.35 122.0
Transport Cost     25.0  35.0  40.0  25.0  35.0   33.0  45.0   30.0
Units (X)         149.0   1.0  33.0 142.0   4.0  196.0  17.0  163.0
```

| | Solution No. 35 | | | | Original No. 77829 | | | |
|---|---|---|---|---|---|---|---|---|
| | Product 0 | | | | Product 1 | | | |
| | Buyer 0 | Buyer 1 | Total | Capacity of Supplier | Buyer 0 | Buyer 1 | Total | Capacity of Supplier |
| Supplier 0 | 149 | 33 | 182 | 300 | 4 | 17 | 21 | 350 |
| Supplier 1 | 1 | 142 | 143 | 280 | 196 | 163 | 359 | 360 |
| Demand from each buyer | 150 | 175 | | | 200 | 180 | | |

**The decision maker may select the final optimal solution as "Solution No. 4" since, it has low "price," "lead-time," and "transportation cost" except the "quality" which the decision maker was prepared to relax.**

The C++ program method developed is highly efficient and it implements the procedure or steps of the posteriori articulation method in which after the solutions are presented to the decision maker, it incorporates a combination of different methods like the evolutionary algorithm method, the goal programming method and the STEM method to reach the final optimal solution.

**Proposed supplier selection procedure, applying MADM and MODM methods on qualitative and quantitative criteria, with the help of an arbitrary example:**

If a general "Supplier selection" example is considered in which there are for example, 5000 suppliers for a part being considered for outsourcing and the company needs to select only a couple of suppliers out of these, the process of selection would be as follows:

The decision maker will specify certain target values for each objective (criterion) to be achieved. Since, initially the pool of suppliers interested for providing the service to the company is large, it can be narrowed down with the help of a simple C++ program. This will help the company to save the manual resource as well as time. The automatic screening of the suppliers can be done based on the targets specified by the decision maker for the attributes, "scope of the resources," "reputation of the supplier," "cultural barrier," "risk," "existing relationship," and "additional value-added capability," the supplier list can be narrowed down to "150," for example, with the priorities of these attributes in the same order as specified above.

According to the priorities of the remaining four objectives "price," "lead time," "quality," and "transportation cost," as mentioned above, the list of suppliers will be narrowed down (goal programming method). For example, initially only about 10 suppliers out of 150 qualify the specified target values for first 4 objectives i.e. "price," "lead time," "quality" and "transportation cost."

If the decision maker is not satisfied with the results, he will apply the STEM method according to which he relaxes the target for any one of the four objectives which he believes is less important as compared to the other three to improve the results for the non-satisfactory objectives. Suppose the decision maker relaxes the target value for the fourth criteria, which is "transportation cost" from US $ 20,000 to US $ 25,000 to be considered for the business, the suppliers based on this are increased from 10 to 25, for example. Revisiting the first three main criteria (objectives), "price," "lead time," and "quality," with some more stringent target values with which the decision maker is satisfied, the suppliers are narrowed down to 5.

Finally, with some negotiations with these 5 suppliers, the company can select 2 suppliers from the existing 5.

# CHAPTER 10

# CONCLUSION

## 10. CONCLUSION

Supplier selection is one of the most critical steps in the outsourcing process; the success of outsourcing activity is highly dependent on successful selection of suppliers (Wadhwa and Ravindran 3735). This thesis study has attempted to advance the art of supply chain management by developing a heuristic methodology “Integrated Evolutionary Goal Trade-off (IEGT) Method” which simplifies the task of supplier selection and reduces the tediousness as well as the degree of error by directly involving the decision maker in the selection process.

Initially, the literature search was done on application of multi-objective methods. After the search in general applications, the search was narrowed down to application of multi-objective methods in the supply chain area. Finally, through the extensive literature search it was observed that very little work has been done in application of multi-objective methods in the supplier selection area. Hence, the application of multi-objective methods in supplier selection was selected as the focus of this thesis study.

A number of different criteria considered by different researchers in their study and the criteria considered in the real world were studied, their role in the supplier selection was identified, and finally ten important criteria from the pool of criteria were selected for this thesis study of supplier selection. Six criteria are qualitative whereas four are quantitative in the selected pool of ten criteria. Qualitative criteria can be optimized by one of the multi-attribute decision making methods. An ad-hoc example is presented in Chapter 6, which illustrates one of the ways to optimize the selection of suppliers based on qualitative criteria. The multi-attribute decision making method is a vast field of study. The prime focus of this thesis study is multi-objective decision making method application in supplier selection. Quantitative criteria can be optimized by using multi-objective decision making methods. Hence, only the quantitative criteria were considered whereas the qualitative criteria were

discarded during the application of heuristic methodology "IEGT" in supplier selection problem.

A representative supplier selection problem was synthesized from the example in "Vendor Selection in Outsourcing" (Wadhwa and Ravindran). A heuristic methodology "IEGT" was developed. A C++ program was written to implement the proposed methodology for the representative problem and optimize the four criteria "price," "lead-time," "quality," and "transportation cost." The "IEGT" method consists of three phases. Phase I is based on the evolutionary algorithm approach. In Phase I, population of solutions is generated; they are compared amongst each other and non-dominated (superior) solutions are selected whereas the inferior solutions are discarded. In Phase II which is based on the goal programming method approach, targets for the four objectives are assessed by the decision maker; the selected solutions from Phase I are compared against these target values and the solutions passing the targets for all four objectives are selected for Phase III. Phase III is based on the STEM method (an interactive method) approach. In Phase III, the unsatisfactory objectives are improved by the decision maker and final supplier(s) is/are selected.

Any other single methodology would not be able to optimize all the four criteria together for the supplier selection. The goal programming method which involves the decision maker to set the goals for the various objectives seems to be the perfect method as compared to any other method, to reduce the set of suppliers from a larger group. If the decision maker is not satisfied with the outcome from the goal programming method, the STEM method is the best way to further filter the suppliers and improve the results. The "IEGT" method is composed of the combination of these strong positive points from various multi-objective methods. The "IEGT" method, a global supplier selection method, is capable of optimizing all four quantitative criteria and helps find the right suppliers.

The main advantage of this "IEGT" method is that it saves an ample amount of time as compared to manual selection of suppliers, if a large pool of suppliers is considered for the service or parts/components. After the application of Phase III, if we have 2-3 suppliers on hand, the negotiation skills (manual selection) can certainly be applied after that, to select one out of three. The "IEGT" method builds on the advantages of the popular multi-objective methodologies in assessing the preference information from the decision maker. The proposed algorithm is simple and user-friendly as well as eliminates the tediousness in the supplier selection process. As illustrated with the representative problem, the "IEGT" method will certainly prove helpful for supplier selection in the real world.

There is certainly a wide scope for future extension of this work. Implementation of the "IEGT" method in a real world scenario is one of the extensions to this work. The suppliers can be encouraged to enter the tender information online and then taking all the entered data information as an input to the developed method, an optimal solution of 2-3 suppliers can be achieved. Hence, a collaboration of the online data input method with the "IEGT" methodology would make the model applicable and useful for real world scenarios. "IEGT" methodology is developed to tackle and optimize the supplier selection problem which matches the representative supplier selection mentioned in this thesis. One can apply multi-attribute decision making methods to optimize the six qualitative criteria, develop a code for qualitative criteria optimization and collaborate that code with the code developed in this thesis which optimizes the four quantitative criteria. In such a way, all the selected criteria can be optimized using a single algorithm and would make the supplier selection task simple, easy and efficient.

# REFERENCES

**REFERENCES:**


Anthony, Baldo. "Outsourcing Essentials." 2.2 (2004). Print.

Belgasmi, Nabil, Lamjed Ben Said, and Khaled Ghedira. "Evolutionary Multi-objective Optimization of the Multi-Location Transshipment Problem." *Operational Research: An International Journal* 4.1 (2004): 5-14. Print.

Clarkston Group Inc. "Supply Chain Management Primer." *ITtoolbox SCM Knowledge Base*. Clarkston Group Inc., 2000. Web. 20 Jan. 2007 <http://supplychain.ittoolbox.com/pub/JB051101a.pdf>.

Day, Charles D. "Application of an Evolutionary Algorithm to Multivariate Optimal Allocation in Stratified Sample Designs." Internal Revenue Service, Statistics of Income Division, 2006. Print.

Dempsey, W. A. "Vendor selection and the buying process." *Industrial Marketing Management* 7.3 (1978): 257-67. Print.

Dickson, G. W. "An analysis of vendor selection systems and decisions." *Journal of Purchasing* 2.1 (1966): 5-17. Print.

Duckstein, Lucein, and Serafim Opricovic. "Multi-objective Optimization in River Basin Development." *Water Resources Research* 16.1 (1980): 14-20. Print.

Ganeshan, Ram, and Terry P. Harrison. *An Introduction to Supply Chain Management*. Pennsylvania: Department of Management Sciences and Information Systems, 303 Business Building, Penn State University, University Park, 1995. Print.

Hicks, D. A. "Four-step Methodology for using simulation and optimization techniques in strategic supply chain planning." *Proceedings of the 1999 Winter Simulation Conference* 2.1 (1999): 215-1220. Print.

Huang, Hong-Zhong, Zhi-Gang Tian, and Ming J. Zuo. "Multi-objective Optimization of Three-Stage Spur Gear Reduction Units Using Interactive Physical Programming." *Journal of mechanical science and technology* 19.5 (2005): 1080-1086. Print.

Hwang, C. L., S. R. Paidy, K. Yoon, and A. S. M. Masud. *Multiobjective Decision Making – Methods and Applications*. New York: Springer-Verlag Berlin Heidelberg, 1979. Print.

Mentzer, John T., William DeWitt, James S. Keebler, Soonghong Min, Nancy W. Nix, Carlo D. Smith, and Zach G. Zacharia. "Defining supply chain management." *Journal of Business Logistics* 22.2 (2001): 1-25. Print.

Mumford, Christine. *MOOGL: Multi-objective Optimization for Green Logistics.* Cardiff University, n.d. Web. 30 June 2007 <http://www.cuimrc.cf.ac.uk/Moogl_Home>.

Oduguwa, A., A. Tiwari, S. Fiorentino, and R. Roy. "Multi-objective Optimization of the Protein-Ligand Docking Problem in Drug Discovery." *ACM Library* 2006: 1793-1800. Print.

Oliveira, Fabiane de, Neida Maria Patias Volpi, and Carlos Roberto Sanquetta. "Goal programming in a planning problem." *Applied Mathematics and Computation* 140.1 (2003): 165-178. Print.

Pinto, Errol G. *Supply Chain Optimization using Multi-objective Evolutionary Algorithms*. Pennsylvania State University, 2004. Web. 15 May 2007

<http://water.engr.psu.edu/reed/Education/CE%20563%20Projects/Pinto.pdf>.

Sabri, Ehap H., and Benita M. Beamon. "A Multi-Objective Approach to Simultaneous Strategic and Operational Planning in Supply Chain Design." *Omega* 28.5 (2000): 581-598. Print.

"Survey of Current and Potential Outsourcing End-Users." *The Outsourcing Institute Membership* (1998): 1-6. Print.

Thaver, Ismat, and Anne Wilcock. "Identification of overseas vendor selection criteria used by Canadian apparel buyers." *Journal of Fashion Marketing and Management* 10.1 (2006): 56-70. Print.

Truong, T. H., and F. Azadivar. "Simulation Based Optimization for Supply Chain Configuration Design." *Proceedings of the 2003 Winter Simulation Conference* (2003): 1268-1275. Print.

Varshney, K. G., and Y. P. Rao. "Multi-objective Crop Planning using Goal Programming- A Case Study." *Journal of the Institution of Engineers* 69.2 (1989): 58-65. Print.

Wadhwa, Vijay, and A. Ravi Ravindran. "Vendor selection in outsourcing." *Computers & Operations Research* 34.12 (2007): 3725-3737. Print.

"Bullwhip effect - Wikipedia, the free encyclopedia." Wikipedia, the free encyclopedia. Web. 10 Feb. 2007 <http://en.wikipedia.org/wiki/Bullwhip_effect>.

"Supply Chain - Wikipedia, the free encyclopedia." Wikipedia, the free encyclopedia. Web. 8 Oct. 2006 <http://en.wikipedia.org/wiki/Supply_chain>.

Vergidis, K., A. Tiwari, and B. Majeed. "Business process improvement using multi-objective optimization." *BT Technology Journal* 24.2 (2006): 229-235. Print.

Weber, Charles A., Ellram, and Lisa M. "Supplier selection using multi-objective programming: A decision support system approach." *International Journal of Physical Distribution and Logistics Management* 23.2 (1993): 3-12. Print.

# APPENDIX 1

# LITERATURE SEARCH TABLES

**TOPICS RELATED TO MULTI-OBJECTIVE OPTMIZATION PROBLEMS:**

| Reference | Problem statement | Objectives | Constraints | Method used | Variables |
|---|---|---|---|---|---|
| Varshney and Rao,1989 | Multi-objective Crop Planning | Maximize irrigated crop area<br>Maximize net benefits<br>Maximize crop production | Total Land<br>Water<br>Crop area<br>Fertilizer | Linear Goal Programming | Hectares of Land used per crop |
| Lucein Duckstein and Serafim Opricovic, 1980 | Multi-objective Optimization in River Basin Development | Minimize Total Cost<br>Minimize Water Shortage<br>Maximize Water Quality<br>Maximize Energy<br>Maximize Recreation<br>Minimize Flood Protection<br>Minimize Land Use<br>Maximize Manpower impact<br>Maximize environmental architecture<br>Maximize International cooperation<br>Maximize Development possibility<br>Maximize Sensitivity | | Compromise Programming | Attributes (Water Shortage, water quality, land use, annual cost, etc.) |
| Vergidis, Tiwari and Majeed, 2006 | Business Process Improvement | Minimize Cost, Maximize duration of business process | Cost, resource, duration of activity, information source, | Interactive Programming, evolutionary algorithm | resources made available |

| | | | | | |
|---|---|---|---|---|---|
| Hong-Zhong Huang Zhi-Gang Tian and Ming J. Zuo, 2005 | Multi-objective Optimization of Three-Stage Spur Gear Reduction | Minimize Volume<br>Minimize Surface Fatigue<br>Maximize Load Carrying Capacity | Tooth Bending Fatigue Failure<br>Shaft torsion stress<br>Face width<br>Interference and Tooth number | Interactive Physical Programming | Core Hardness, Face width, Tooth numbers, Diameter of the shaft |
| Fabiane de Oliveira, Neida Maria Patias Volpi, Carlos Roberto Sanquetta, 2003 | Multi-objective programming in Brazilian Forest problem | Maximize Wood Harvest<br>Maximize number of tourists<br>Maximize the pasture (creation of buffalos)<br>Maintain Employees<br>Maximize the diversity of flora<br>Maximize the diversity of fauna | Total Land<br>Forest area | Goal Programming | Area used for timber, ervamate leaves, pasture and tourism |
| Oduguwa, Tiwari, Fiorentino and Roy | Multi-objective Optimization of Protein Ligand in Drug Discovery | Maximize internal energy of the compound, protein-compound couple's Van Der Waal's & electrostatic energy of interaction, Shape complementarities | Computer specs, Population size, number of generations | Evolutionary algorithm (C++) | Docking configurations (o/p-complex of drug) |
| Weber, Charles and Lisa, 1993 | Supplier Selection using Multi-objective Programming | Minimize Price, Maximize quality, minimize late orders | Number of units required, Number of Suppliers required to fulfill demand, quantity of late deliveries | Decision support system approach (compromise) | Quantities ordered |

**TOPICS RELATED TO MULTI-OBJECTIVE OPTIMIZATION IN SUPPLY CHAIN NETWORKS:**

| Reference | Problem statement | Objectives | Constraints | Method used | Variables | Category |
|---|---|---|---|---|---|---|
| Nabil Belgasmi, Lamjed Ben Said and Khaled Ghedira, 2004 | Evolutionary Multi-objective Optimization of the Multi-Location Transshipment | Minimize Total expected cost Maximize expected fill rate Minimize expected transshipment lead time | Quantities shipped cannot be more than available quantities at store, Quantities shipped cannot be more than unmet demand at store. | Strength Pareto Evolutionary Algorithm (SPEA2) | Order Quantities | Tactical Procurement |
| Errol G. Pinto, April 2004 | Supply Chain Optimization using Multi-objective Evolutionary Algorithms | Minimize Total Operating Cost Minimize Manufacturing Cost Maximize Profit Maximize Revenue Minimize Transportation Cost | Plant Capacity, Supplier Capacity, Inventory balancing constraints for respective components, Total Cost constraint. | Non-dominated Sorting Genetic Algorithm-II | Number of components from plant to supplier, products from plants to customer zone and inventory of components at plants. | Operational Distribution |

| | | | | | | |
|---|---|---|---|---|---|---|
| Dr. Christine Mumford, 2007 | Multi-objective Optimization for Green Logistics | Building multi-objective optimization decision support tools for strategic and operational SCM, with a special focus on environmental issues. | “Research into Multi-objective techniques for supply chain management is still in its infancy” | | | |
| Ehap H. Sabri, Benita M. Beamon, 2000 | Multi-objective Approach to Simultaneous Strategic and Operational Planning in Supply Chain Design | Minimize Total fixed & variable costs<br>Minimize volume flexibility | Supplier capacity, Production requirement,<br>Plant capacity, Production capacity, Throughput capacity,<br>Each customer zone assigned to 1 DC, Products shipped = products available at plants, Total Shipments = Demand requirements. | Algorithm | Quantity of product produced at plants, products shipped from plants to Distribution center, products shipped from vendors to plants, Total cost. | Strategic and Operational Distribution |

| | | |
|---|---|---|
| Webpronews.<br>com | Lean Supply Chain Management | 1. Reduce & eliminate waste/non-value added activities to total supply chain flow.<br>2. Waste can be measured in time, inventory & unnecessary costs.<br>3. Supply chain should flow. Any activity that stops the flow should add value & create value.<br>4. Pull-system, continuous improvement, Top management's commitment.<br>5. Realize cause-effect impacts.<br>6. Drive for root causes.<br>7. Be open to changes, to completely redesigned process.<br>8. Lean supply chain can reduce: Time by 10-40%, Inventory by 10-30% and Costs by 10-25% |

# APPENDIX 2

# HARDEE TOY EXAMPLE SOLVED USING STEM METHOD

**STEM METHOD:**

**Problem (Hwang et al. 174):**

The Hardee Toy Company makes two kinds of toy dolls. Doll A is a high quality toy and Doll B is of lower quality. The respective profits are $0.40 and $0.30 per doll. Each Doll A requires twice as much time as Doll B, and if all dolls were of type B, the company could make 500 per day. The supply of material is sufficient for only 400 dolls per day (both A and B combined). The problem assumes that all the dolls for type A and type B the factory can make could be sold, and that the best customer of the company wishes to have as many as possible of the type A doll. The manager realizes that the two objectives: the maximization of profit and the maximum production of Doll A, should be considered in scheduling production.

**Hardee Toy Company:**
**X1 = Number of Product A produced.**
**X2 = Number of Product B produced.**

**To Maximize the Profit:**
Objective Function (f1):
MAX 0.4X1 + 0.3X2

Constraints:
ST X1 + X2 < 400
2X1 + X2 < 500
END

**Solution:**
OBJECTIVE FUNCTION VALUE
1) **130.0000**

| VARIABLE | VALUE | REDUCED COST |
|---|---|---|
| X1 | **100.000000** | 0.000000 |
| X2 | **300.000000** | 0.000000 |

| ROW | SLACK OR SURPLUS | DUAL PRICES |
|---|---|---|
| 2) | 0.000000 | 0.200000 |
| 3) | 0.000000 | 0.100000 |

**To Maximize the Production of Product A:**

Objective Function (f2):

MAX X1

ST X1 + X2 < 400

2X1 + X2 < 500

END

**Solution:**

OBJECTIVE FUNCTION VALUE

1) **250.0000**

| VARIABLE | VALUE | REDUCED COST |
|---|---|---|
| X1 | **250.000000** | 0.000000 |
| X2 | **0.000000** | 0.500000 |

| ROW | SLACK OR SURPLUS | DUAL PRICES |
|---|---|---|
| 2) | 150.000000 | 0.000000 |
| 3) | 0.000000 | 0.500000 |

**Ideal Solution for the problem maximizing these 2 objectives: f1=130 & f2=250 (Point E; X1 = 250 and X2 = 100) which is infeasible.**

**Pay-off table:**

| | f1 | f2 | X1 | X2 | Point in Graph |
|---|---|---|---|---|---|
| **f1** | **130** | **100** | **100** | **300** | **C** |
| **f2** | **100** | **250** | **250** | **0** | **B** |

**Minimizing the deviations of the objectives from their optimal solution:**

**Iteration 0:**

Objective Function:

MIN D

ST X1 + X2 < 400

2X1 + X2 < 500

D + 0.5652X1 > 141.3

D + 0.1739X1 + 0.1304X2 > 56.52

END

**Solution:**

OBJECTIVE FUNCTION VALUE

1) 11.30660

| VARIABLE | VALUE | REDUCED COST |
|---|---|---|
| D | 11.306603 | 0.000000 |
| X1 | **229.995407** | 0.000000 |
| X2 | **40.009178** | 0.000000 |

| ROW | SLACK OR SURPLUS | DUAL PRICES |
|---|---|---|
| 2) | 129.995407 | 0.000000 |
| 3) | 0.000000 | 0.113023 |
| 4) | 0.000000 | -0.133262 |
| 5) | 0.000000 | -0.866738 |
| 6) | 11.306603 | 0.000000 |
| 7) | 229.995407 | 0.000000 |
| 8) | 40.009178 | 0.000000 |

Substituting X1 = 230 & X2 = 40 in objectives f1 & f2, we get: f1 = **104** & f2 = **230**

**Pay-off table:**

| | **f1** | **f2** | **X1** | **X2** | **Point in Graph** |
|---|---|---|---|---|---|
| **f1** | **130** | **100** | **100** | **300** | **C** |
| **f2** | **100** | **250** | **250** | **0** | **B** |
| **Iter. 0** | **104** | **230** | **230** | **40** | **F** |

**The decision maker indicates that the objective f2 is satisfactory at or above 200:**

**Iteration 1:**

MIN D

ST X1 + X2 < 400

2X1 + X2 < 500

X1 > 200

0.4X1 + 0.3X2 > 104

D + 0.4X1 + 0.3X2 > 130

END

**Solution:**

OBJECTIVE FUNCTION VALUE

1) 20.00000

| VARIABLE | VALUE | REDUCED COST |
|---|---|---|
| D | 20.000 | 0.0000 |
| X1 | **200.000** | 0.0000 |
| X2 | **100.000** | 0.0000 |

| ROW | SLACK OR SURPLUS | DUAL PRICES |
|---|---|---|
| 2) | 100.000000 | 0.000000 |
| 3) | 0.000000 | 0.300000 |
| 4) | 0.000000 | -0.200000 |
| 5) | 6.000000 | 0.0000000 |
| 6) | 0.000000 | -1.000000 |
| 7) | 20.000000 | 0.000000 |
| 8) | 200.000000 | 0.000000 |
| 9) | 100.000000 | 0.000000 |

Substituting X1 = 200 & X2 = 100 in objectives f1 & f2, we get: f1 = **110** & f2 = **200**

**Pay-off table:**

| | **f1** | **f2** | **X1** | **X2** | **Point in Graph** |
|---|---|---|---|---|---|
| **f1** | **130** | **100** | **100** | **300** | **C** |
| **f2** | **100** | **250** | **250** | **0** | **B** |
| **Iteration 0** | **104** | **230** | **230** | **40** | **F** |
| **Iteration 1** | **110** | **200** | **200** | **100** | **G** |

**Point G**, solution in iteration 1, is the solution presented to the decision maker. If it is accepted, this solution becomes the **final solution**. Otherwise, no satisfactory solution exists for this problem using this method.

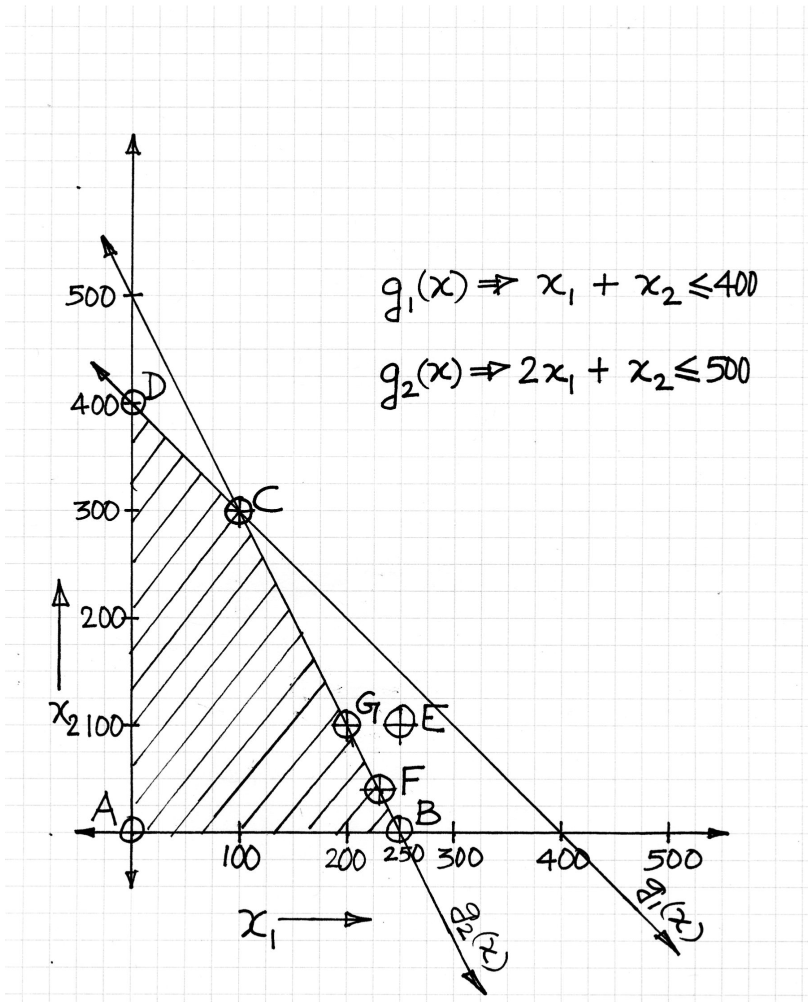
$g_1(x) \Rightarrow x_1 + x_2 \leq 400$
$g_2(x) \Rightarrow 2x_1 + x_2 \leq 500$
500
400
300
200
$x_2$ 100
100
200
250
300
400
500
$x_1$
A
B
C
D
E
F
G
$g_1(x)$
$g_2(x)$

fig.1

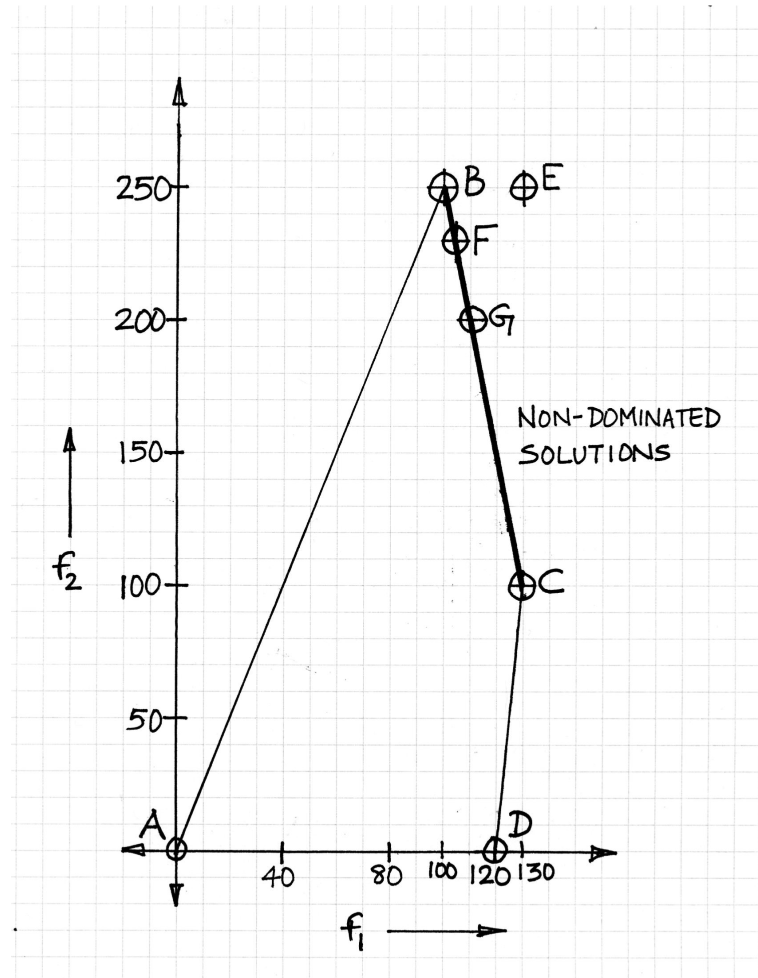
250
200
150
100
50
$f_2$
40
80
100
120
130
$f_1$
A
B
C
D
E
F
G
NON-DOMINATED
SOLUTIONS

fig. 2

# APPENDIX 3

# IMPORTANT TERMS USED IN THESIS

**Non-dominated Solution:**

A non-dominated solution is the one in which no one objective function can be improved without decreasing simultaneously, at least one of the other objective functions.

**e.g.** From Figure 2, consider segment AB, we can see that point B is better than point A since both the objective functions f1 and f2 increase as we move from point A (0, 0) to point B (100, 250). Similarly for segments AD and CD, point D (120, 0) is better than point A (0, 0) and point C (130, 100) is better than point D (120, 0). But when we consider segment BC, neither of the points can be concluded as a better solution because while moving from point B (100, 250) to point C (130, 100), f1 increases but f2 decreases and vice-versa. Hence, any point on **segment BC** is called a **non-dominated solution.**

**Trade-off Information:**

It is the ratio of change in one objective function by the change in another objective function.

e.g.

Two types by which the decision maker gives the information:

1. **Implicit trade-off Information:** The decision maker will not specify the acceptable limit for all the objectives. He will specify only the acceptable limit for one of the objective functions so that the analyst can relax that objective to improve the non-satisfactory one.

   **e.g.** In the above example, when the solution in iteration 0 is presented to the decision maker**,** the decision maker indicates that the objective function f2 is satisfactory at or above 200 which means objective function f2 can be relaxed by 30 to improve objective function f1.

2. **Explicit trade-off Information:** The decision maker will specify the acceptable limit for all the objectives, the acceptable limit to which the

satisfactory ones can be relaxed and the acceptable limit by which the non-satisfactory ones have to be improved. This information is given by a ratio called trade-off ratio.

The trade-off ratio is given by:

$$\text{Trade-off ratio} = \frac{\Delta f1}{\Delta f2}$$

**e.g.** In the above example, when the solution in iteration 0 is presented to the decision maker**,** the decision maker indicates that the objective function f2 is satisfactory at or above 200 which means objective function f2 can be relaxed by 30 and objective f1 is satisfactory at or above 110 which means objective function f1 has to be improved by 6.

# APPENDIX 4

# C++ PROGRAM “IEGT” METHOD

## C++ PROGRAM "IEGT" METHOD

### Phase I

```
#include "mystd.cpp"

//Function Prototypes
int verify_constr( double[], int );
int nrand( int );
int xrand ( int, int );
double urand();
void gen_solution ();
void price_matrix ();
void objective_functions ();

//    i = product j = buyer   k=supplier  m=price level (not used)
//    0                 0                 0
//    0                 0                 1
//    0                 1                 0
//    0                 1                 1
//    1                 0                 0
//    1                 0                 1
//    1                 1                 0
//    1                 1                 1


//Global Variables
double P[2][2][2];            // Pijkm Buyer j's Cost of acquiring unit of part i from supplier k at price
level m

int F0 = 3500;                // Fixed Cost associated with supplier 1
int F1 = 3600;                // Fixed Cost associated with supplier 2
```

```
// Demand "dij" of part i by buyer j
int d00 = 150;
int d01 = 175;
int d10 = 200;
int d11 = 180;

// Quality "qik" that supplier maintains for the part i in terms of 'percent of rejections'
double q[2][2];

// Lead time "lijk" of supplier k to produce and supply part i to buyer j in terms of 'Days'
int LSB[2][2][2];

// Production capacity "CAPki" of supplier k for part i
int CAP00 = 300;
int CAP01 = 350;
int CAP10 = 280;
int CAP11 = 360;

/*
// Quantity "bikm" at which price break occurs for part i given by supplier k for both buyers
int b000 = 85;
int b010 = 95;
int b001 = 180;
int b011 = 190;
int b100 = 120;
int b110 = 125;
int b101 = 210;
int b111 = 205;
*/

// Lead time "Lij" that buyer j requires for part i in terms of 'Days'
int L00 = 18;
int L01 = 20;
int L10 = 26;
int L11 = 13;
```

```
// Quality level "Qj" that buyer j requires the all suppliers to maintain in terms of 'percent of
rejections'
double Q0 = 0.095;
double Q1 = 0.09;

// Transportation Cost "Tijkm" that buyer j pays for ordering the part i from supplier k at price break
level m
double T[2][2][2];

int maxN;                    // Maximum number of suppliers that can be selected

double X[2][2][2];           // Xijkm Number of units of part i supplied by supplier k to buyer j at price
level m

int Z[2];                    // Zk Binary variable which denotes whether a supplier is selected or not

int Y[2][2][2];              // Yijkm Binary variable which denotes which price level is selected

double objfun[4];

const int nsolution=300000;
double solution[nsolution][5]; //solutions generated (i,). i=solution #, j=objecvtive#, j=5 inferior (0,1)

int low, high;

double X0j0, X1j0;

int main()
{
      if (iofiles("data_input.dat","optimized_output.lis") != 0)
            return 1;

      srand(67891); // random number seed
```

```
// Quality "qik" that supplier maintains for the part i in terms of 'percent of rejections'
q[0][0] = 0.03;
q[0][1] = 0.09;
q[1][0] = 0.06;
q[1][1] = 0.02;

// Lead time "lijk" of supplier k to produce and supply part i to buyer j in terms of 'Days'
LSB[0][0][0] = 15;
LSB[0][1][0] = 17;
LSB[0][0][1] = 19;
LSB[0][1][1] = 18;
LSB[1][0][0] = 24;
LSB[1][1][0] = 21;
LSB[1][0][1] = 11;
LSB[1][1][1] = 12;

int check=0;

outFile << fixed;
outFile.precision(2);
outFile.setf(ios::right, ios::adjustfield);

outFile << "\n\n \t\t\t\t\t i = PRODUCT         j = BUYER         k = SUPPLIER" << endl;

outFile << endl << "                    ";

for (int i=0; i<2; i++)
            for (int j=0; j<2; j++)
                  for (int k=0; k<2; k++)
                        {
                              outFile.width(8);
                              outFile << i << j << k;
                        }

      outFile << endl << endl << "General Output with values of all variables and corresponding solutions
of Objective Functions";
```

```
for (int isolution=0; isolution<nsolution; isolution++)
{
/*    feasible = 1;
      while (!feasible)
      {
            // generate one solution
            gen_solution();
            // verify constraints
            feasible = verify_constr();
            if (!feasible) outFile << endl << "Infeasible";
      }
*/
      gen_solution();

      price_matrix ();

      objective_functions ();

      outFile << endl << endl << isolution << "   ";

      outFile << endl << "Price Level         ";
      for (int i=0; i<2; i++)
            for (int j=0; j<2; j++)
                  for (int k=0; k<2; k++)
                        {
                              outFile.width(10);
                              outFile << Y[i][j][k];
                        }
      outFile << endl << "Price per unit      ";
      for (int i=0; i<2; i++)
            for (int j=0; j<2; j++)
                  for (int k=0; k<2; k++)
                        {
                              outFile.width(10);
                              outFile << P[i][j][k];
                        }
```

```
        outFile << endl << "Transportation Cost ";
        for (int i=0; i<2; i++)
              for (int j=0; j<2; j++)
                    for (int k=0; k<2; k++)
                          {
                                outFile.width(10);
                                outFile << T[i][j][k];
                          }

        outFile << endl << "Units               ";
        for (int i=0; i<2; i++)
              for (int j=0; j<2; j++)
                    for (int k=0; k<2; k++)
                          {
                                outFile.width(10);
                                outFile << X[i][j][k];
                          }

        outFile << endl << "Objectives          ";
        for (int i=0; i<4; i++)
              {
                    outFile.width(10);
                    outFile << objfun[i];
              }

  }

  // look for inferior (dominated) solutions
  outFile << endl << "\n\n Solutions indicating the inferior (dominated) ones" << endl;
  for (int count=0; count<nsolution; count++)
  {
        solution[count][4] = -1; // non-dominated solution
        for (int i=0; i<nsolution; i++)
        {
              if (i==count) continue;
              check = 0;
```

```
			for (int j=0; j<4; j++)
				if (solution[i][j] >= solution[count][j]) check=1;

			if (check==0)
			{
				solution[count][4] = i; // inferior solution
				break;
			}
		}
			outFile << "\n solution #=";
			outFile.width(5);
			outFile << count;
			for (int j=0; j<5; j++)
			{
				outFile.width(12);
				outFile << solution[count][j];
			}
			if (solution[count][4]==-1)
			{	cout << endl;
				for (int j=0; j<5; j++)
				{
					cout.width(12);
					cout << solution[count][j];
				}
			}
	}

	outFile << endl << "\n\nNon-dominated solutions" << endl;
	outFile << endl << "sol# \t orig# \t price \t lead-time \t quality \t transpcost" <<endl;
	int solseq=0;
	for (int count=0; count<nsolution; count++)
	{
		if (solution[count][4]==-1)
		{
			outFile << endl;
			outFile.width(7); outFile << ++solseq;
```

```
                    outFile.width(7); outFile << count;
                    for (int j=0; j<4; j++)
                    {
                    outFile.width(12);
                    outFile << solution[count][j];
                    }
            }
        }
}

void gen_solution()
{
            int k0, k1, other_supplier_capacity; // temp variables
            // index i=product, j=buyer, k=supplier

            // Product i=0

            while (1) // infeasible
            {
                  //Supplier k=0. Supplier 0 Capacity for Product 0 is 300

                  // Buyer j = 0 and 1
                  // Total qty of Product 0 supplied by supplier 0 (to buyers 0 and 1)
                  if (urand()<=0.50)
                        {k0=0; k1=1;X0j0 = xrand(0,300); other_supplier_capacity = 280;} // supplier 0
                  else
                        {k0=1; k1=0;X0j0 = xrand(0,280); other_supplier_capacity = 300;} // supplier 1

                  while (1) // infeasible
                  {
                        // if supplier 0 is not used for this product
                        if (urand()<0.01)
                        {     X[0][0][k0] = 0;
                              X0j0 = 0;
                              X[0][1][k0] = 0;
                        }
```

```
            else
            {
                  // demand from buyer 0 is 150

                  // qty supplied to buyer 0

                  if ( X0j0 < 150 )
                        X[0][0][k0] = xrand (0, (int) X0j0);
                  else
                        X[0][0][k0] = xrand (0, 150);

                  // qty supplied to buyer 1
                  X[0][1][k0] = X0j0 - X[0][0][k0];
            }

      // Supplier k0=1

            // buyer demands are complemented by supplier 1
            X[0][0][k1] = 150 - X[0][0][k0];
            X[0][1][k1] = 175 - X[0][1][k0];

            if ( X[0][1][k1] >= 0 ) break; // feasible solution
      }

      // Supplier 1 Capacity for Product 0 is 280
      if ( (X[0][0][k1] + X[0][1][k1]) <= other_supplier_capacity ) break; // feasible solution
}

// Product i=1

while (1)   //    infeasible
{
      // Supplier k=0. Supplier 0 Capacity for Product 1 is 350

      // Buyer j = 0 and 1
      // Total qty of Product 1 supplied by supplier 0 (to buyers 0 and 1)
```

```
if (urand()<=0.50)
      {k0=0; k1=1;X1j0 = xrand(0,350); other_supplier_capacity = 360;} // supplier 0
else
      {k0=1; k1=0;X1j0 = xrand(0,360); other_supplier_capacity = 350;} // supplier 1
// X1j0 = xrand(0,350);

while (1)   //    infeasible
{
      // if supplier k0 is not used for this product
      if ( urand() < 0.01 )
      {
             X[1][0][k0] = 0;
             X1j0 = 0;
             X[1][1][k0] = 0;
      }
      else
      {
            // demand from buyer 0 is 200

            // qty supplied to buyer 0

            if ( X1j0 < 200 )
                  X[1][0][k0] = xrand (0, (int) X1j0);
            else
                  X[1][0][k0] = xrand (0, 200);

            // qty supplied to buyer 1
            X[1][1][k0] = X1j0 - X[1][0][k0];
      }

// Supplier k=1

      // buyer demands are complemented by supplier k1

      X[1][0][k1] = 200 - X[1][0][k0];
      X[1][1][k1] = 180 - X[1][1][k0];
```

```
                        if ( X[1][1][k1] >= 0 ) break;      // feasible solution
                  }

                  // Supplier 1 Capacity for Product 1 is 360
                  if ( (X[1][0][k1] + X[1][1][k1]) <= other_supplier_capacity ) break;    // feasible
                                                                                              solution
            }
}


// Price Breaks for two levels of quantities and the transportation cost associated with the quantity
shipped

void price_matrix ()
{
      // index i=product, j=buyer, k=supplier

      // Price breaks for Product i=0 by Supplier k=0 to Buyer j=0
      if ( X[0][0][0] < 85 )
      {
            P[0][0][0] = 125.40;
            Y[0][0][0] = 0;
            T[0][0][0] = 30;
      }
      else
            if ( X[0][0][0] < 180 )
            {
                  P[0][0][0] = 115.00;
                  Y[0][0][0] = 1;
                  T[0][0][0] = 25;

            }
```

```
// Price breaks for Product i=0 by Supplier k=0 to Buyer j=1
  if ( X[0][1][0] < 85 )
  {
        P[0][1][0] = 125.40;
        Y[0][1][0] = 0;
        T[0][1][0] = 40;

  }
  else
        if ( X[0][1][0] < 180 )
        {
              P[0][1][0] = 115.00;
              Y[0][1][0] = 1;
              T[0][1][0] = 37;
        }

  // Price breaks for Product i=0 by Supplier k=1 to Buyer j=0
  if ( X[0][0][1] < 95 )
  {
        P[0][0][1] = 128.30;
        Y[0][0][1] = 0;
        T[0][0][1] = 35;
  }
  else
        if ( X[0][0][1] < 190 )
        {
              P[0][0][1] = 120.00;
              Y[0][0][1] = 1;
              T[0][0][1] = 28;
        }

  // Price breaks for Product i=0 by Supplier k=1 to Buyer j=1
  if ( X[0][1][1] < 95 )
  {
        P[0][1][1] = 128.30;
        Y[0][1][1] = 0;
```

```
        T[0][1][1] = 32;
}
else
        if ( X[0][1][1] < 190 )
        {
                P[0][1][1] = 120.00;
                Y[0][1][1] = 1;
                T[0][1][1] = 25;
        }

// Price breaks for Product i=1 by Supplier k=0 to Buyer j=0
if ( X[1][0][0] < 120 )
{
        P[1][0][0] = 150.35;
        Y[1][0][0] = 0;
        T[1][0][0] = 35;
}
else
        if ( X[1][0][0] < 210 )
        {
                P[1][0][0] = 130.00;
                Y[1][0][0] = 1;
                T[1][0][0] = 30;
        }

// Price breaks for Product i=1 by Supplier k=0 to Buyer j=1
if ( X[1][1][0] < 120 )
{
        P[1][1][0] = 150.35;
        Y[1][1][0] = 0;
        T[1][1][0] = 45;
}
else
        if ( X[1][1][0] < 210 )
        {
                P[1][1][0] = 130.00;
```

```
            Y[1][1][0] = 1;
            T[1][1][0] = 42;
      }

// Price breaks for Product i=1 by Supplier k=1 to Buyer j=0
if ( X[1][0][1] < 125 )
{
      P[1][0][1] = 148.75;
      Y[1][0][1] = 0;
      T[1][0][1] = 40;
}
else
      if ( X[1][0][1] < 205 )
      {
            P[1][0][1] = 122.00;
            Y[1][0][1] = 1;
            T[1][0][1] = 33;
      }

// Price breaks for Product i=1 by Supplier k=1 to Buyer j=1
if ( X[1][1][1] < 125 )
{
      P[1][1][1] = 148.75;
      Y[1][1][1] = 0;
      T[1][1][1] = 37;
}
else
      if ( X[1][1][1] < 205 )
      {
            P[1][1][1] = 122.00;
            Y[1][1][1] = 1;
            T[1][1][1] = 30;
      }
```

```
/*    // Price breaks for Product i=0  by Supplier k=0 to Buyers j=0 and j=1
      if ( X[0][0][0] || X[0][1][0] < 85 ) P[0][0] = 125.40;
      else if ( X[0][0][0] || X[0][1][0] < 180 ) P[0][0] = 120.80;

      // Price breaks for Product i=0  by Supplier k=1 to Buyers j=0 and j=1
      if ( X[0][0][1] || X[0][1][1] < 95 ) P[0][1] = 128.30;
      else if ( X[0][0][1] || X[0][1][1] < 190 ) P[0][1] = 118.70;

      // Price breaks for Product i=1  by Supplier k=0 to Buyers j=0 and j=1
      if ( X[1][0][0] || X[1][1][0] < 120 ) P[1][0] = 150.35;
      else if ( X[1][0][0] || X[1][1][0] < 210 ) P[1][0] = 146.60;

      // Price breaks for Product i=1  by Supplier k=1 to Buyers j=0 and j=1
      if ( X[1][0][1] || X[1][1][1] < 125 ) P[1][1] = 148.75;
      else if ( X[1][0][0] || X[1][1][0] < 205 ) P[1][1] = 146.35;
*/

}

// Objective Function Solutions
void objective_functions ()
{
      double value1 = 0.0;
      double value2 = 0.0;
      double value3 = 0.0;
      double value4 = 0.0;
      static int count=-1;

      count++; // solution number
      for (int j=0; j<4; j++)
      objfun[j] = 0.0;



      // Generating the value for the first objective function "Total Cost" which is the addition of
Variable and Fixed Costs
```

```
// Variable Cost
for (int i=0; i<2; i++)
            for (int j=0; j<2; j++)
                  for (int k=0; k<2; k++)
                        {
                              value1 = P[i][j][k] * X[i][j][k];
                              objfun[0] = objfun[0] + value1;
                        }

// Adding the Fixed Cost, which is the dependent of the Supplier selected
if ( (X[0][0][0] + X[0][1][0] + X[1][0][0] + X[1][1][0]) > 0 )
      objfun[0] = objfun[0] + F0;
if ((X[0][0][1] + X[0][1][1] + X[1][0][1] + X[1][1][1]) > 0)
            objfun[0] = objfun[0] + F1;

// Generating the value for the second objective function "Lead Time"
for (int i=0; i<2; i++)
            for (int j=0; j<2; j++)
                  for (int k=0; k<2; k++)
                        {
                              value2 = LSB[i][j][k] * X[i][j][k];
                              objfun[1] = objfun[1] + value2;
                        }

// Generating the value for the third objective function "Quality"
for (int i=0; i<2; i++)
            for (int k=0; k<2; k++)
                  for (int j=0; j<2; j++)
                        {
                              value3 = q[i][k] * X[i][j][k];
                              objfun[2] = objfun[2] + value3;
                        }

// Generating the value for the fourth objective function "Transportation Cost"
for (int i=0; i<2; i++)
```

```
                for (int j=0; j<2; j++)

                {
                      for (int k=0; k<2; k++)
                            {
                                  value4 = T[i][j][k] * X[i][j][k];
                                  objfun[3] = objfun[3] + value4;
                            }
                      // objfun[3] *= 0.01;   // convert to $
                }

      // save the solution for comparison
      for (int j=0; j<4; j++)
            solution[count][j] = objfun[j];

}


// RANDOM NUMBER GENERATOR 0 to 1
double urand()
{
      return(rand()/32768.0);
}

// RANDOM NUMBER GENERATOR 0 OR 1

int nrand()
{
      if (rand() < 16383) return 0;
      else return 1;
}

// Random Number Generator low to high

int xrand (int low, int high)
{
```

```
        int Z;

        Z = low + rand()%(high - low + 1);

        return Z;
}

int verify_constr()

{

        // Verifying the Capacity Constraints
/*
        // Supplier k=0 and Product i=0
        if ( (X[0][0][0][0] + X[0][0][0][1] + X[0][1][0][0] + X[0][1][0][1] ) > 300) return 0;

        // Supplier k=0 and Product i=1
        if ( (X[1][0][0][0] + X[1][0][0][1] + X[1][1][0][0] + X[1][1][0][1] ) > 350) return 0;

        // Supplier k=1 and Product i=0
        if ( (X[0][0][1][0] + X[0][0][1][1] + X[0][1][1][0] + X[0][1][1][1] ) > 280) return 0;

        // Supplier k=1 and Product i=1
        if ( (X[1][0][1][0] + X[1][0][1][1] + X[1][1][1][0] + X[1][1][1][1] ) > 360) return 0;
*/


        // Verifying the Demand Constraints
/*
        // Product i=0 and Buyer j=0
        if ( (X[0][0][0] + X[0][0][0][1] + X[0][0][1][0] + X[0][0][1][1] ) != 150) return 0;

        // Product i=0 and Buyer j=1
        if ( (X[0][1][0][0] + X[0][1][0][1] + X[0][1][1][0] + X[0][1][1][1]) != 175) return 0;

        // Product i=1 and Buyer j=0
```

```
    if ( (X[1][0][0][0] + X[1][0][0][1] + X[1][0][1][0] + X[1][0][1][1]) != 200) return 0;

    // Product i=1 and Buyer j=1
    if ( (X[1][1][0][0] + X[1][1][0][1] + X[1][1][1][0] + X[1][1][1][1]) != 180) return 0;


*/
    return 1;

}
```

**Phase II and Phase III**

```
#include "mystd.cpp"


// Function Prototypes
void compare (double [][4], int);
void compare_targets(double [][4], int, char);

// Global Variables
int  i,j, k;
int count, count1;
double Price_min, Lead_Time_min, Quality_min, Transport_cost_min;
double Price_max, Lead_Time_max, Quality_max, Transport_cost_max;
double Target_Price, Target_Lead_Time, Target_Quality, Target_Transport_cost;

int check = 0;
double solution[500][4], solution2[500][4];
char satisfy;


int main()
{

      int seqno, origno;

      if (iofiles("data_input.dat","optimized_output2.lis") != 0)
            return 1;

            count = -1;

            outFile << fixed;
            outFile.precision(2);
            outFile.setf(ios::right, ios::adjustfield);

            cout << fixed;
```

```
cout.precision(2);
cout.setf(ios::right, ios::adjustfield);


// Reading the values from the input file
// Printing the values to the output file
outFile << "Solutions " << endl;
for (int i=0, check=0; check==0 ; i++)
{
      count ++;   // Counting the number of solutions
      outFile << endl;
      inFile >> seqno >> origno;

      outFile.width(7); outFile << seqno;
      outFile.width(7); outFile << origno;

      for (int j=0; j<4; j++)
      {
            inFile >> solution[i][j];
            if ( solution [i][j] == 0)
            {
                  check = 1;
                  break;
            }
            outFile.width(10);
            outFile << solution[i][j];
      }
}


// find out the fmin and fmax
compare (solution, count);

// segregate solutions based on Target values for objectives
compare_targets(solution, count, satisfy);
```

```
}

// Function to find the minimum and maximum values
//   of the Objective Functions
void compare (double solution[][4], int count)
{
      double min ;
      double max ;

      for ( j=0; j < 4; j++)
      {
            min = solution [0][j];
            max = solution [0][j];

            for ( i=1; i < count; i++)
            {
                  if ( solution [i][j] < min)   // Comparing the two values
                        {
                              min = solution [i][j];
                        }

                  if ( solution [i][j] > max)   // Comparing the two values
                        {
                              max = solution [i][j];
                        }

            }

            if ( j==0)
            {
                  Price_min = min;                          // Minimum value of Objective Function 'Price'
                  Price_max = max;                          // Maximum value of Objective Function 'Price'
            }
            else if (j==1)
            {
```

```
                  Lead_Time_min = min;                          // Minimum value of Objective Function 'Lead
Time'
                  Lead_Time_max = max;                          // Maximum value of Objective Function 'Lead
Time'
            }
            else if (j==2)
            {
                  Quality_min = min;                                  // Minimum value of Objective Function
'Quality'
                  Quality_max = max;                                  // Maximum value of Objective Function
'Quality'
            }
            else
            {
                  Transport_cost_min = min;                     // Minimum value of Objective Function
'Transportation Cost'
                  Transport_cost_max = max;                     // Maximum value of Objective Function
'Transportation Cost'
            }

      }

}

void compare_targets(double solution[][4], int count, char satisfy)
{
      satisfy = 'N';
      while ( satisfy != 'Y')       // Not satisfied with the solution
      {
            cout << "\n\nThe minimum value for Function 1 'Price' is: " << Price_min << endl;
            cout << "The maximum value for Function 1 'Price' is: " << Price_max << endl;


            cout << "\n\nThe minimum value for Function 2 'Lead Time' is: " << Lead_Time_min << endl;
            cout << "The maximum value for Function 2 'Lead Time' is: " << Lead_Time_max << endl;
```

```
            cout << "\n\nThe minimum value for Function 3 'Quality' is: " << Quality_min << endl;
            cout << "The maximum value for Function 3 'Quality' is: " << Quality_max << endl;


            cout << "\n\nThe minimum value for Function 4 'Transportation Cost' is: " << Transport_cost_min
<< endl;
            cout << "The maximum value for Function 4 'Transportation Cost' is: " << Transport_cost_max <<
endl;


            cout << endl << endl << "                         "
                  << "    Minimum " << "    Maximum " << endl << endl;
            cout << "Price                    " ;
            cout.width(12); cout << Price_min ;
            cout.width(12);   cout << Price_max << endl;

            cout << "Lead Time                " ;
            cout.width(12);   cout << Lead_Time_min ;
            cout.width(12);   cout << Lead_Time_max << endl;

            cout << "Quality                  " ;
            cout.width(12);   cout << Quality_min ;
            cout.width(12);   cout << Quality_max << endl;

            cout << "Transportation Cost      " ;
            cout.width(12);   cout << Transport_cost_min ;
            cout.width(12); cout << Transport_cost_max << endl;


            cout << "\nPlease specify the Target value for Function 1 'Price': ";
            cin >> Target_Price;

            cout << "\nPlease specify the Target value for Function 2 'Lead Time': ";
            cin >> Target_Lead_Time;
```

```
cout << "\nPlease specify the Target value for Function 3 'Quality': ";
cin >> Target_Quality;

cout << "\nPlease specify the Target value for Function 4 'Transportation Cost': ";
cin >> Target_Transport_cost;


int count1; // Number of solutions that meet the targets
count1 = 0;

outFile << endl << endl << "                         "
      << "    Minimum " << "    Maximum "
      << "     Target " << endl << endl;

outFile << "Price                    " ;
outFile.width(12);      outFile << Price_min ;
outFile.width(12);  outFile << Price_max ;
outFile.width(12);  outFile << Target_Price << endl;

outFile << "Lead Time                " ;
outFile.width(12);  outFile << Lead_Time_min ;
outFile.width(12);  outFile << Lead_Time_max ;
outFile.width(12);  outFile << Target_Lead_Time << endl;

outFile << "Quality                  " ;
outFile.width(12);  outFile << Quality_min ;
outFile.width(12);  outFile << Quality_max ;
outFile.width(12);  outFile << Target_Quality << endl;

outFile << "Transportation Cost      " ;
outFile.width(12);  outFile << Transport_cost_min ;
outFile.width(12);  outFile << Transport_cost_max ;
outFile.width(12);  outFile << Target_Transport_cost << endl;

// print header for the solution set
for (int i=0; i<count; i++)
```

```
		{
			if (solution[i][0] <= Target_Price)
				if (solution[i][1] <= Target_Lead_Time)
					if (solution[i][2] <= Target_Quality)
						if (solution[i][3] <= Target_Transport_cost)
						{
							outFile << endl << "Solutions meeting the target values: " <<
endl;

							cout << endl << "Solutions meeting the target values: " << endl;
							break;
						}
		}


		// solution set that meets targets
		for (int i=0; i<count; i++)
		{
			if (solution[i][0] <= Target_Price)
				if (solution[i][1] <= Target_Lead_Time)
					if (solution[i][2] <= Target_Quality)
						if (solution[i][3] <= Target_Transport_cost)
						{
								count1++;
								outFile.width(5);
								outFile << endl << i+1;
								cout.width(5);
								cout << endl << i+1;

							for (int j=0; j<4; j++)

							{
								outFile.width(12);
								outFile << solution[i][j] ;
								cout.width(12);
								cout << solution[i][j] ;
```

```
                                        // save the current solution into the set
                                        solution2[count1-1][j] = solution[i][j];


                                    }
                                }

                }

                if (count1 > 0)
                {
                    // Finding the minimum and the maximum from
                    // the selected set of solutions
                    compare (solution2, count1);
                }

                cout << endl
                    << "\nTotal number of solutions meeting the target values are: "
                    << count1 << endl;
                outFile << endl
                    << "\nTotal number of solutions meeting the target values are: "
                    << count1 << endl;
                cout << endl
                    << "Are you satisfied with the number of solutions? (Y or N) ";
                cin >> satisfy;
                satisfy = toupper(satisfy);
                outFile << endl << "Satisfied (Type 'Y' for Yes or 'N' for No): " << satisfy << endl;
            }

}
```

# REFERENCES

**REFERENCES:**


Belgasmi, Nabil, Lamjed Ben Said, and Khaled Ghedira. "Evolutionary Multi-objective Optimization of the Multi-Location Transshipment Problem." *Operational Research: An International Journal* 4.1 (2004): 5-14. Print.

Duckstein, Lucein, and Serafim Opricovic. "Multi-objective Optimization in River Basin Development." *Water Resources Research* 16.1 (1980): 14-20. Print.

Huang, Hong-Zhong, Zhi-Gang Tian, and Ming J. Zuo. "Multi-objective Optimization of Three-Stage Spur Gear Reduction Units Using Interactive Physical Programming." *Journal of mechanical science and technology* 19.5 (2005): 1080-1086. Print.

Hwang, C. L., S. R. Paidy, K. Yoon, and A. S. M. Masud. *Multiobjective Decision Making – Methods and Applications*. New York: Springer-Verlag Berlin Heidelberg, 1979. Print.

Mumford, Christine. *MOOGL: Multi-objective Optimization for Green Logistics.* Cuimrc.ac.uk. Cardiff University, n.d. Web. 30 June 2007 <http://www.cuimrc.cf.ac.uk/Moogl_Home>.

Oduguwa, A., A. Tiwari, S. Fiorentino, and R. Roy. "Multi-objective Optimization of the Protein-Ligand Docking Problem in Drug Discovery." *ACM Library* 2006: 1793-1800. Print.

Oliveira, Fabiane de, Neida Maria Patias Volpi, and Carlos Roberto Sanquetta. "Goal programming in a planning problem." *Applied Mathematics and Computation* 140.1 (2003): 165-178. Print.

Pinto, Errol G. *Supply Chain Optimization using Multi-objective Evolutionary Algorithms*. PSU.edu. Pennsylvania State University, 2004. Web. 15 May 2007 <http://www.engr.psu.edu/ce/Divisions/Hydro/Reed/Education/CE%20563%20Projects/Pinto.pdf>.

Sabri, Ehap H., and Benita M. Beamon. "A Multi-Objective Approach to Simultaneous Strategic and Operational Planning in Supply Chain Design." *Omega* 28.5 (2000): 581-598. Print.

Thomas, Craig. *Lean Supply Chain Management*. Webpronews.com. N.p. 2004. Web. 26 March 2007 <http://www.webpronews.com/topnews/2004/06/18/lean-supply-chain-management>.

Varshney, K. G., and Y. P. Rao. "Multi-objective Crop Planning using Goal Programming- A Case Study." *Journal of the Institution of Engineers* 69.2 (1989): 58-65. Print.

Vergidis, K., A. Tiwari, and B. Majeed. "Business process improvement using multi-objective optimization." *BT Technology Journal* 24.2 (2006): 229-235. Print.

Weber, Charles A., Ellram, and Lisa M. "Supplier selection using multi-objective programming: A decision support system approach." *International Journal of Physical Distribution and Logistics Management* 23.2 (1993): 3-12. Print.