Multi-Level Programming Approach to a Closed-Loop Supply Chain Network Design

Sima Ghayebloo, Department of Industrial Engineering, K. N. Toosi University of Technology, Tehran, Iran

Mohammad Jafar Tarok, Department of Industrial Engineering, K. N. Toosi University of Technology, Tehran, Iran

Mostafa Abedzadeh, Department of Industrial Engineering, K. N. Toosi University of Technology, Tehran, Iran

Claver Diallo, Department of Industrial Engineering, Dalhousie University, Halifax, Canada

ABSTRACT

In this paper, the authors address the problem of network design for a closed-loop supply chain. The problem is formulated as a mixed zero-one bi-level optimization model, with the manufacturer as the leader who minimizes his costs at the upper level, and a forwarding agent dealt with as the follower. The leader decides on the locations of the facilities, and the forwarding agent builds the forward and reverse transportation plans so as to minimize the total transportation cost. A genetic algorithm solution method is used to obtain the Stackelberg solution. Furthermore, the algorithm uses penalty functions to handle the constraints. The solution algorithm is implemented in Matlab, utilizing LINGO 11.0 (2008) to solve each lower level problem instance. Finally, the accuracy of the model is tested on a set of numerical experiments.

Keywords: *Closed-Loop Supply Chain, Multi-Level Programming, Network Design, Reverse Logistics, Stackelberg Solution*

1. INTRODUCTION

The managing of end-of-life products is called reverse logistics (Gupta, 2007). Reverse logistics (RL) is the process of planning, implementing, and controlling the flow of unused materials from markets or usage areas to a point of recovery or proper disposal. More precisely, RL is the process of moving goods from their typical final destination for the purpose of capturing value, or proper disposal (Rogers, 1998). Fleischmann et al. (Fleischmann, 2000) state that product recovery not only reverses the product stream with the consequence that there are many supply sources and few demand points, but that the design is severely complicated by the high uncertainty levels in many factors.

DOI: 10.4018/ijsds.2013100104

In many cases when an existing supply network is already in operation, the design of a recovery network may actually translate into the redesign of the network to include the recovery activities. The key design issues are to determine the number of layers in the network; the number and location of warehouses and redistribution plants and recovery plants and their capacities; select the recovery technologies to implement; choose between an open-loop and a closed-loop system, and define which functions have to be carried out in the reverse logistics channel. Possible functions in the reverse logistics channel are: collection, cleaning and testing, sorting, transportation and processing (Pohlen, 1992).

Many decision making problems require compromises among the objectives of several interacting stakeholders or entities, and they are often arranged in a hierarchical structure with independent and sometimes conflicting objectives that cannot be weighted and aggregated into a single objective to yield a solution accepted by both parties. Supply chain network design in both forward and reverse networks is an example of that kind of problems. Several methods have been proposed to solve this type of problems by collectively determining a solution that accounts for the choices of all the distinct decision makers. The multi-level programming approach is one such method and its basic concept is presented next.

Multi-level programming (MLP) is an approach to solve decentralized planning problems with multiple executors in a hierarchical organization. It explicitly assigns each agent a unique objective and a set of decision variables as well as a set of common constraints that affects all agent (Lee & Shih, 2000). One of the important characteristics of Multi-Level Programming Problems (MLPP) is that a planner at a certain level of hierarchy may have his/her objective function and decision space partially determined by other levels. Further, the control instruments used by each planner to improve his/her own objective function can affect the policies at other levels. These instruments may include the allocation and use of resources at lower levels and the advantages obtained from other levels (Anuradha Gaur, 2008). MLP problems share the following common features:

1. The system possesses interacting decision making units within a hierarchical structure.
2. Each subordinate level performs its policies after acquiring a complete knowledge of the decisions made at the superior levels.
3. Each unit maximizes its net benefits independently of other units but may be influenced by actions and reactions of those units.
4. The external effect on a decision maker's problem can be reflected in both the objective function as well as on the set of feasible decisions.

The bi-level programming as a special form of the multi-level programming, initiated by Von Stackelberg (Stackelberg, 1952) is mainly developed for solving decentralized management problems with decision makers in a hierarchical organization, with the upper level termed the leader and the lower level called the follower (Bard, 1998). In a bi-level decision making, the control of decision factors is partitioned amongst the leader and follower who seek to optimize their individual objective functions, and the corresponding decision do not control but affect the decision of the other level or party (Aiyoshi, 1981). The leader attempts to optimize his or her objective function but he or she must anticipate all possible responses from the follower (Lai, 1996). The follower observes the leader's decision and then responds to it in a way that is personally optimal. Because the set of feasible choices available to either decision maker is independent, the leader's decision affects both the follower's payoff and allowable actions, and vice versa (Gao, 2010).

Wen and Yang (1990) deal with a two-level mixed zero-one programming problem with zero-one decision variables for the leader and real-valued decision variables at the follower level, and they propose an exact method and a heuristic method to obtain the Stackelberg solutions. Their exact method also utilizes the