Computers & Industrial Engineering 124 (2018) 545–559

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

Large-scale reverse supply chain network design: An accelerated Benders decomposition algorithm

Ahmed Alshamsi[a], Ali Diabat[b,c,*]

[a] Department of Technical Services, GASCO Habshan and Bab Plant, ADNOC Gas Processing, Abu Dhabi National Oil Company, Abu Dhabi, United Arab Emirates
[b] Division of Engineering, New York University Abu Dhabi, Saadiyat Island, 129188, Abu Dhabi, United Arab Emirates
[c] Department of Civil and Urban Engineering, Tandon School of Engineering, New York University, Brooklyn, NY 11201, United States of America

ARTICLE INFO

Keywords:
Reverse logistics
Benders decomposition
Acceleration techniques
Pareto-optimal cut
Large-scale optimization

ABSTRACT

Reverse logistics (RL) is a term that captures any process that involves the movement of goods or services from their destination back to their source. RL is increasingly gaining recognition as an essential component in the design process of supply chains. In this paper, we formulate a mixed-integer linear programming (MILP) model that aims to determine the optimal locations and capacities of various nodes such as inspection centers and remanufacturing facilities and to help decision makers find optimal transportation decisions and to determine the number and type of transportation units required in the network. We develop an exact method to solve large-scale real-sized instances of this problem. We initially attempt to solve the problem using the traditional Benders Decomposition (BD) technique which fails to solve the problem in reasonable computational times. We improve the traditional BD technique by adding several accelerating methods such as trust-region, logistics constraints, Pareto-optimal cuts, restructuring of the problem, and continuous relaxation of the integer variables to increase the convergence rate and to reduce the total number of cuts required in the master problem. In almost all the instances, we were able to reach optimal solutions. For the remaining instances, we succeed in solving the largest problem with an optimality gap of 0.5% and within a reasonable running time. The paper highlights and evaluates the performance and effectiveness of the different acceleration techniques of our improved BD algorithm along with the computational results.

1. Introduction

Reverse logistics describes a sequence of operations that starts at the consumer level and ends at the manufacturer; this sequence is in contrast to the traditional forward approach of the classical supply chain. Recycling, reuse, and remanufacturing of products are all activities of RL networks, and these actions are increasingly prevalent due to growing environmental and socio-economic concerns (Alshamsi & Diabat, 2017). The recycling process can be defined as the process of making new products from old ones. Remanufacturing, on the other hand, is defined as an industrial process in which worn-out products are restored to like-new conditions (Lund, 1984). The American Reverse Logistics Executive Council defines reverse logistics as "the process of planning, implementing, and controlling the efficient, cost–effective flow of raw material, in-process inventory, finished goods and related information from the point of consumption to the point of origin for the purpose of recapturing value or proper disposal" (Rogers & Tibben-Lembke, 1999).

As a result of the significant growth in global population, limited resources available to support this growth, and ubiquitous environmental pollution due to produced waste, adopting green growth strategies has become a critical requirement. Sustainable development, green growth, green supply chains, and reverse logistics (RL) network design are interconnected concepts. More specifically, the 'green' term involves addressing the influence and relationships between any specific process (economic, growth, production, supply chain, etc.) and the natural environment (Srivastava, 2007). However, sustainable development is a broader term that covers and contains the remaining focus areas. According to Choucri, no agreement exists for a specific meaning of sustainable development; further, a range of definitions surround the term sustainability. One definition of sustainable development is that it is "the process of meeting the needs of current and future generations without undermining the resilience of the life-supporting properties of nature and the integrity and security of the social systems" (Choucri, 2007).

Reverse logistics is vital to protecting the environment, to ensuring

* Corresponding author.
E-mail address: Diabat@nyu.edu (A. Diabat).

https://doi.org/10.1016/j.cie.2018.05.057
Received 7 December 2017; Received in revised form 7 May 2018; Accepted 31 May 2018
Available online 20 July 2018
0360-8352/

the sustainability of resources, and to increasing the success of businesses that offer logistics and support services. The idea of reverse logistics enhances the return on investments for a business entity because a business can reap second returns on investment through recycling and remanufacturing, even when equipment appears to be out of order (Alshamsi & Diabat, 2015). Reverse logistics also increases the likelihood of a favorable public perception towards a business entity as embracing and supporting the conservation of the environment. This is likely to increase customer rating and elevates the level of customer loyalty. Competition among business entities is also enhanced through reverse logistics as firms seek to increase profit margins and gain any competitive edge possible (Alshamsi & Diabat, 2017).

The expanding scope of RL in large-scale case studies leads to complex problems that are difficult to solve in terms of computational processing time (CPU) and the quality of solutions available using commercial software. Mixed-integer-linear programming (MILP) is an increasingly attractive option for tackling such problems, especially given the constant improvements being made to algorithms in this domain as well as rapidly improving processor speeds. According to Granville and Pereira, to establish the groundwork for the expansion of the Benders decomposition method, a trial was initially done with linear integer programs. The results yielded mixed-integer-linear programming that produces a solution using Benders decomposition in two steps. The first step extends the BD to give a solution of pure integers, and the second step includes computational improvements for greater efficiency. Computational outcomes of the BD give promising partitioning techniques for integer programming and illustrate that the process can be accelerated by more than 90% in some instances over standard algorithms (Granville & Pereira, 1985).

The Benders decomposition algorithm was first projected for use in a class of mixed-integer-linear programming problems. If the integer variables are fixed, the outcome of the problem is a continuous linear programming which can employ standard duality theory to enhance its cuts (Magnanti, Mireault, & Wong, 1986). Numerous enhancements and extensions of the BD algorithm have been made to improve the effectiveness of the algorithm on some problem classes (Magnanti & Wong, 1984). In addition, the algorithm usually provides some foundation for the structuring of heuristics for problems that would otherwise be difficult. The range of optimization problems that could be tackled using the algorithm includes many integer, bi-level, multi-stage, stochastic and nonlinear problems (Floudas & Pardalos, 2009; Rahmaniani, Crainic, Gendreau, & Rei, 2016).

In Alshamsi and Diabat (2017), a genetic algorithm (GA) was developed to solve large-scale MILP problems for different real-world RL case studies from the Gulf Cooperation Council (GCC) area. The performance of GA in Alshamsi and Diabat (2017) was shown to be very efficient in terms of the quality of the solutions and CPU times compared with the exact solution method, but this cannot be guaranteed in general with heuristic algorithms. Our goal is to solve the same problems with provable efficiency by employing decomposition methods. This seems reasonable in our setting since the particular structure of our proposed MILP model lends itself naturally to such approaches. Developing and accelerating Benders decomposition algorithm that is capable of solving large-scale MILP problems of RL faster and with a high rate of efficiency by implementing new ideas is our main contribution in this study.

Different acceleration techniques will be used to increase the convergence rate and to decrease computational times required to solve the same six cases adopted from Alshamsi and Diabat (2017), and these will be discussed in Section 5. Some of these ideas were used in our previous work (Alshamsi & Diabat, 2017) to improve GA.

The paper is structured as follows. Section 2 provides a thorough overview of the relevant literature with particular emphasis on some acceleration techniques used in tandem with BD. Section 3 articulates the problem statement, and Section 4 presents the proposed formulation and parameters. Section 5 delves into the acceleration techniques used in this work. Section 6 summarizes the algorithm's performance and presents some empirical computational results. Finally, Section 7 presents the conclusions as well as recommendations for further research.

2. Literature review

On the environmental side of things, Soysal (2016) proposed a probabilistic mixed-integer-linear programming model for the closed-loop inventory routing problem (CIRP) that accounts for forward and reverse logistics operations, explicit fuel consumption, demand uncertainty, and multiple products. The results showed that the model can make significant savings in total cost. Mohajeri and Fallah (2016) studied the carbon footprint that arises in a closed–loop supply chain. Carbon emission was expressed in environmental constraints. The aim of these constraints was to limit the carbon emission per unit of product supplied with different transportation modes. Moreover, Al Shamsi, Al Raisi, and Aftab (2014) developed a model for the pollution inventory routing problem with perishable goods (PPIRP) to reduce carbon emissions. Their work shed light on the trade-off between emission costs and total costs. The model achieves approximately 61% reduction in carbon emissions and a 23% decrease in empty vehicle trips compared to Inventory Routing Problem with perishable goods (PIRP) model that does not account for the cost due to greenhouse emissions.

Various methods for solving mixed-integer-linear programming (MILP) problems have been developed and used in many papers. For example, a branch-and-bound procedure was studied by Boyce, Farhi, and Weischedel (1973), Boffey and Hinxman (1979), and Gallo (1981). However, this method can efficiently solve only small-sized problems because the computational time required for large-sized problems is prohibitive. On the other hand, heuristic procedures are capable of solving large-scale problems. Boffey and Hinxman (1979), Wong (1984), Le, Diabat, Richard, and Yih (2013), Fu, Diabat, and Tsai (2014), Diabat and Deskoores (2016), and Alshamsi and Diabat (2017) implemented heuristic techniques in their studies. As always, the use of heuristics comes at the cost of the absence of guarantees on the quality of solutions, which motivates the need for exact methods.

Decomposition methods were developed to decompose large complex problems into smaller problems that are easier to solve. The two most prominent examples are the Benders decomposition method from 1962 and the Lagrangian relaxation method from 1970 (Fisher, 1981). Pioneered by Benders in 1962, the BD algorithm's major purpose is solving problems with complicated variables so that, with provisional fixes, the problem would become easier to handle (Jörnsten, 1979). BD is also referred to as an outer linearization and variable partitioning method because it is based on a sequence of steps involving repeated projection, outer linearization, and relaxation (Geoffrion, 1970). The basic idea behind Benders procedures is a partitioning of the given problem into two subproblems: the integer or non-linear problem known as the Benders Master Problem (BMP) and the linear problem known as the Benders subproblem (BSP). Then, to prevent the complex calculation of a complete set of constraints for the feasible region in the first problem, the BMP is relaxed in term of constraints. At each iteration, one of two types of cuts (an optimality cut or a feasibility cut) is produced by using the BSP and adding it to the relaxed BMP. Both BSP and BMP are solved iteratively, and they produce an upper bound (UB) and a lower bound (LB) of the optimal value of the objective function of the original problem in minimizing problem, respectively. If both LB and UB are equal, then we stop the algorithm (Benders, 1962). BD is described in detail in Section 5 and Appendix A.

Much research concentrates on finding better cutting-plane-producing schemes to reduce the processing time (Watson & Rogers, 2006). Balinski (1965) succeeded in accelerating the Benders algorithm by using *Balinski cuts* that are easily generated without running an extra linear programming step at each iteration. Balinski cuts were initially thought to be Pareto-optimal cuts – no other cuts can dominate them.