Expert Systems with Applications 37 (2010) 6919–6930

Contents lists available at ScienceDirect

Expert Systems with Applications

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

# Adopting co-evolution and constraint-satisfaction concept on genetic algorithms to solve supply chain network design problems

Chang Ying-Hua *

Department of Information Management, TamKang University, 151 Ying-Chuan Road, Tamsui, Taipei County 25137, Taiwan, ROC

ARTICLE INFO

*Keywords:*
Supply chain network design
Genetic algorithms
Co-evolution concept
Constraint-satisfaction concept
Optimization

ABSTRACT

With the rapid globalization of markets, integrating supply chain technology has become increasingly complex. That is, most supply chains are no longer limited to a particular region. Because the numbers of branch nodes of supply chains have increased, products and raw materials vary and resource constraints differ. Thus, integrating planning mechanisms should include the capacity to respond to change. In the past, mathematical programming and a general heuristics algorithm were used to solve globalized supply chain network design problems. When mathematical programming is used to solve a problem and the number of decision variables is too high or constraint conditions are too complex, computation time is long, resulting in low efficiency, and can easily become trapped in partial optimum solution. When a general heuristics algorithm is used and the number of variables and constraints is too high, the degree of complexity increases. This usually results in an inability of people to think about resource constraints of enterprises and obtain an optimum solution.

Therefore, this study uses genetic algorithms with optimum search features. This work combines the co-evolutionary mode, which is in accordance with various criteria and evolves dynamically, and constraint-satisfaction mode capacity to narrow the search space, which helps in finding rapidly a solution that, solves supply chain integration network design problems. Additionally, via mathematical programming, a simple genetic algorithm, co-evolutionary genetic algorithm, constraint-satisfaction genetic algorithm and co-evolutionary constraint genetic algorithm are used to compare the experiments result and processing time to confirm the performance of the proposed method.



## 1. Introduction

With the rapid development of a global economy, information and manufacturing technology, the needs of consumers, production and sales have changed. Enterprises must now think globally. Independent operation modes are no longer suitable for markets undergoing upheavals. The integration of industries and the division of work has become the best solution for enterprise survival. To create an efficient global supply chain, resources, the supply chain and all factories must be tightly integrated. Additionally, firms must respond to customer requirements efficiently, offer high-quality products, reduce operational costs and increase customer satisfaction.

Because most supply chain network designs have multiple layers, members, periods and products, and a comparative resource constraint exists between different layers. For example, a factory may have constraints on its productivity, and a distribution center may have limited storage capacity. These problems typically increase as the number of supply chain layers increase, the time period increases, and the number of products and purchase orders increase. These cause the network search space and time required to obtain a solution to increase markedly. Thus, the supply chain network design problem is an NP-complete problem (Ibaraki & Katoh, 1988). Many studies adopted mathematical programming or a heuristics algorithm to solve such problems. For example, Robinson and Satterfield (1998) established a mixed-integer programming solution for a two-echelon, multiple products storage problem. Hou and Chang (2003) use the evolution principle to solve a production and distribution problem. When faced with a simple supply chain system, mathematical programming can easily find the optimum solution; however, mathematical programming is not suitable for large mixed-integer programming problems. By adopting a single point random search, the number of decision variables increases, which extends the time required to find the solution and the method can easily become trapped in a local optimum solution (Wu, 2002). When adopting a general heuristics algorithm, such as simulated annealing, the search for the optimum solution cannot be reached completely and efficiently (Altiparmak, Gen, & Lin, 2005; Wang, 2002).

* Tel.: +886 911 231536; fax: +886 2 26209737.
*E-mail address:* yhchang@mail.tku.edu.tw

0957-4174/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2010.03.030

This study uses the features of genetic algorithms (parallel searching and can evolve in an environment) combined with the co-evolutionary mode (considering multiple criteria, search speed and avoiding converge before the right time) and the constraint-satisfaction mode (narrow the search space accelerate the speed at which the optimum solution is acquired) to establish supply chain system for all manufacturing branches. The minimum cost of a supply chain network is adopted as the most suitable plan for developing an algorithm for a supply chain network distribution. The proposed method is a rapid way of finding a solution and obtaining optimum search capacity. To verify the accuracy and efficacy of the proposed algorithm, LINGO software was used in mathematical programming mode. The experimental result was compared with the algorithmic result to determine the difference in cost and calculation time. This comparison can verify the efficiency of the proposed algorithm such that a flexible and efficient supply chain integration network mode can be obtained. This result can function as a reference for manufacturers in daily operations and for management decisions.

## 2. Literature review

This study focuses on supply chain integration and its network design problem. Related literature is discussed in the following sequence: the supply chain integration model; genetic algorithms; co-evolutionary mode; and, constraint-satisfaction mode.

### *2.1. Supply chain integration mode*

Supply chain networks obtain raw materials, supplies, manufactures and distributes product to customers. All these planning, organizing and controlling from the supply point to the demand point include cash flows, information flows and materials circulation (Maloni & Benton, 1997). By combining strategies to achieve integration of ideality and reality of different enterprises to form a reticulation supply chain network. Because different enterprises in a supply chain may have different or conflicting targets, and supply chains must adjust to changes, integration of a supply chain is not easily achieved (Hayashi, Koguro, & Murakami, 2005; Liu & Hsieh, 2005; Su, 2000; Sue, 1999; Vaidyanathan & Hasan, 2001; Viswanathan & Piplani, 2001). Thus, supply chain members must be coordinated and cooperate to achieve a common target. Additionally, a dynamic integration mode must be established that can respond effectively to changes in global markets.

As supply chain management is a complex decision problem, many researchers adopted mathematical programming or a heuristics algorithm to solve this problem. For instance, Robinson and Satterfield (1998) established a mixed-integer programming model that maximizes profit and is designed to find the solution for a single period and single product in a supply chain network. Koray and Marc (1999) proposed a multiple products mixed-integer programming mode for minimum cost that included manufacturing cost, storage cost and transport cost. They used Bender's decomposition of integer programming to find the solution. Melachrinoudis et al. (2000) used multiple objective approaches to find a solution. Ross (2000) employed a performance-based strategic resource allocation to solve supply network design problems. Syam (2002) applied Lagrangian relaxation and simulated annealing to a supply chain network with multiple levels and settings; minimize cost was the target. Syarif, Yun, and Gen (2002) used genetic algorithms and a spanning tree to find the solution for supply chain network with three levels and single product with the aim of minimizing cost. Altiparmak, Gen, Lin, and Karaoglan (2007) employed steady-state genetic algorithms to obtain the solution for the design of a supply chain network with multiple products and phases, and compare their algorithm with linear programming, Lagrangian relaxation and simulated annealing. This study aims to minimize purchasing cost, transportation cost, fixed cost and manufacturing cost. Based on the above literature, some studies take maximize profit as their objective, whereas most supply chain network models take minimize cost as their measure of performance and consider many costs, including purchasing cost, transportation cost and storage cost Altiparmak et al. (2007) proposed a supply chain network model that covered almost all of these costs. Thus, this study adopts Altiparmak's model as the norm in the mathematical mode.

### *2.2. Genetic algorithms*

Genetic algorithms first developed by Holland (1975), use computer programs to simulate the evolutionary process with the chromosome as the solution to the solved problem. Based on the environmental adaptation of chromosomes, researchers identified a fitness value such that a researcher could determine whether a chromosome would survive to the next generation. The evolutionary process continues until the target has been met. via self-adaptation and an iteration threshold, the algorithm has the ability to evolve to the optimum solution for a problem.

The parallel search feature is hidden in genetic algorithms. This feature is suitable for handling multiple space dimensions, and non-linear complex NP-hard problems (Goldberg, 1989). Genetic algorithms is faster than the simple point search function of traditional algorithms and has a wide application in such areas as stock market analysis (Szeto & Fong, 2000), vehicle routing problems (Ting & Huang, 2005), production allocation problems (Hou and Chang, 2002), tourism itinerary planning (Christophe & Hugues, 2007), communications network design (Chou, Premkumar, & Chu, 2001), weather prediction (Wong, Yip, & Li, 2008), location–allocation problems (Jaramillo, Bhadury, & Batta, 2002; Zhou & Liu, 2003), personnel training (Juang, Lin, & Kao, 2007), and bus network optimization (Bielli, Caramia, & Carotenuto, 2002). However, genetic algorithms have two shortcomings. If the fitness value is set improperly, convergence can occur prematurely. The second is that it is not suitable for highly non-linear problems (Hillis, 1992).

### *2.3. Co-evolutionary mode*

To overcome these shortcomings, Hillis developed the co-evolutionary scheme in 1992. In this scheme a living being (chromosome) and environment (multiple criteria) interact and co-evolve. The genes are improved continuously for survival and the environment changes with the living being. For instance, when an eagle hunts a rabbit, the rabbit must run fast to survive, and the eagle must also fly fast to catch the rabbit. This means that both the chromosome and multiple criteria constraint conditions must evolve. Restated, as the chromosome must match the fitness value, criteria constraint conditions must be evaluated by a fitness function. The evaluation criteria for next evaluation are selected and based on the degree of fitness of criteria. In this way, the shortcomings of traditional genetic algorithms can be overcome and search speed accelerated.

In the operation mode, all chromosomes must be evaluated to determine their degree of match with the multiple criteria constraint conditions. The ease with which these constraint conditions are met is directly related to the size of the fitness values of the criteria. Conversely, the fitness value can be large such that the criteria have the opportunity to be the evaluation function for the next evaluation. Fig. 1 presents this co-evolutionary mode. After this process, the evolutionary direction of a chromosome will be that for which conditions are not easily met. Thus, the near optimum