2023년 8월
석사학위 논문

Design and Implementation of Closed-Loop Supply Chain Model Considering Supplier Disruption Risk

- 공급자 붕괴위험을 고려한 폐쇄루프 공급망 모델의 설계와 이행

조선대학교 대학원
경영학과
TSERENGOTOV NOMIN ERDENE

Design and Implementation of Closed-Loop Supply Chain Model Considering Supplier Disruption Risk

- 공급자 붕괴위험을 고려한 폐쇄루프 공급망 모델의 설계와 이행

2023년 8월 25일

조선대학교 대학원
경영학과
TSERENGOTOV NOMIN ERDENE

Design and Implementation of Closed-Loop Supply Chain Model Considering Supplier Disruption Risk

- 공급자 붕괴위험을 고려한 폐쇄루프 공급망 모델의 설계와 이행

지도교수 윤영수

이 논문을 경영학 석사학위신청 논문으로 제출함

2023년 4월

조선대학교 대학원
경영학과
TSERENGOTOV NOMIN ERDENE

TSERENGOTOV NOMIN ERDENE의

석사학위논문을 인준함

위원장 조선대학교 교 수 함주연 (인)

위 원 조선대학교 교 수 김 광 (인)

위 원 조선대학교 교 수 윤영수 (인)

2023년 5월

조선대학교 대학원

Table of Contents

List of Tables

List of Figures

국문 초록

공급자 붕괴위험을 고려한 폐쇄루프 공급망 모델의 설계와 이행

Tserengotov Nomin Erdene
Advisor: Prof. YoungSu Yun, Ph.D.
Department of Business Administration,
Graduate School of Chosun University

다단계 폐쇄 루프 공급망(Closed-loop supply chain: CLSC) 모델은 일반적으로 순방향 물류(Forward logistic: FL) 와 역 물류(Reverse logistic: RL) 각 단계의 설비로 구성된다. CLSC 모델은 다양한 요인에 의해 붕괴될 수도 있다. 예를 들어, 우크라이나와 러시아 사이의 전쟁은 특정 제품과 그 구성품의 공급 경로에 영향을 미칠 수 있다. CLSC 모델에서 이러한 예측 불가능한 상황이 발생하면 글로벌 공급망에 위험이 발생할 수 있다. 따라서 다양한 위험을 효과적으로 관리할 수 있는 CLSC 모델이 필수적이다.

본 논문에서는 공급업체 붕괴위험을 고려한 폐쇄 루프 공급망 (Closed-loop supply chain with disruption risk: CLSC-DR) 모델을 제안한다. CLSC-DR 모델에서는 주 부품 공급업체와 주 경로의 붕괴 위험이 고려된다. 대부분의 기존 연구들은 단순한 형태의 공급망 네트워크에서 부품 공급자 붕괴와 경로 붕괴에 초점을 맞추고 있기 때문에, CLSC-DR 모델에서 부품 공급자 붕괴와 경로 붕괴를 고려하는 것이 보다 현실적이고 효과적이다. 주 부품 공급업체 및 주경로의 붕괴 위험에 대처하기 위해, 백업 부품 공급업체 및 백업 경로가 CLSC-DR 모델에서 고려된다.

최근 몇 년 동안, 사람들은 오프라인 상점보다 온라인 쇼핑에 더 관심을 갖게 되었으며, 이러한 경향은 더 많은 시간을 절약하고 더 안전하다는 장점을 가진다. 따라서 본 연구에서는 정상 배송(Normal delivery: NDL)과 직접 배송(Direct delivery: DDL)을 CLSC-DR 모델에 함께 고려하였다.

본 논문에서 제안하는 CLSC-DR 모델은 수리적 공식화로 표현되며, 혼합형 메타

휴리스틱(Hybrid meta-heuristic)인 GA-VNS-TLBO 접근법을 사용하여 이행한다. GA-VNS-TLBO 접근법은 유전 알고리즘(Genetic algorithm: GA), 가변 이웃 검색(Variable neighbourhood search: VNS) 및 교육 및 학습 기반 최적화(Teaching and learning-based optimization: TLBO)를 함께 사용한 것이다.

수치 실험에서 다양한 크기의 CLSC-DR 모델을 설정한 후 GA-VNS-TLBO 접근법을 적용하여 CLSC-DR 모델을 해결한다. GA-VNS-TLBO 접근법의 수행도는 기존의 몇몇 메타 휴리스틱 접근법(단일 메타 휴리스틱 접근법: GA, VNS 및 TLBO, 혼합 메타 휴리스틱 접근법: 다양한 GA-VNS, GA-TLBO들)의 수행도와 비교된다. 수치실험결과는 GA-VNS-TLBO 접근법이 기존의 메타 휴리스틱 접근법들 보다 그 유연성 및 효율성 측면에서 더 우수하는 것을 보여준다.

그러나, 수치실험에서 사용된 데이터가 랜덤하게 발생되었기 때문에, 실제 현장에서 얻어진 좀 더 현실적인 데이터를 사용한 연구가 필요할 것이며, 이것은 미래의 연구분야로 남겨 둔다.

Abstract

Design and Implementation of Closed-Loop Supply Chain Model Considering Supplier Disruption Risk

Tserengotov Nomin Erdene

Advisor: Prof. YoungSu Yun, Ph.D.

Department of Business Administration,

Graduate School of Chosun University

A multi-stage closed-loop supply chain (CLSC) model is generally composed of facilities at each stage of forward logistics (FL) and reverse logistics (RL). The CLSC model may be disrupted by various factors. For instance, the conflict between Ukraine and Russia could affect the supply routes of certain products and components. When such unpredictable situation occurs in the CLSC model, it can create a risk to global supply chain (SC). Therefore, the CLSC model that can manage various disruption risks effectively is imperative.

In this paper, the CLSC with supplier disruption risk (CLSC-DR) model is proposed. In the CLSC-DR model, the disruption risks of main part suppliers and main routes are considered. Since most conventional studies have focused on part supplier disruption and route disruption in simple SC networks, the consideration of part supplier disruption and route disruption in the CLSC-DR model can make it more realistic and effective. To this purpose, the CLSC-DR model considers backup part suppliers and backup routes to cope with the disruption risks of main part suppliers and main routes.

In recent years, peoples have become more interested in online shopping than in stores, which has the advantage of saving more time and being safer. Therefore, in this study, normal delivery (NDL) as well as direct delivery (DDL) are also considered in the CLSC-DR model.

The CLSC-DR model proposed in this paper is expressed as a mathematical formulation and is implemented using a hybrid meta-heuristic approach, called the GA-VNS-TLBO approach. The GA-VNS-TLBO approach is a combination of genetic algorithms (GA), variable neighborhood search (VNS), and teaching and learning-based optimization (TLBO).

After setting up the CLSC-DR models with various sizes in numerical experiment, the GA-VNS-

TLBO approach is applied to solve the CLSC-DR models. The performance of the GA-VNS-TLBO approach is compared to that of some existing meta-heuristic approaches (GA, VNS and TLBO as a single meta-heuristic approach, and various GA-VNS and GA-TLBO as a hybrid meta-heuristic approach). The results of the numerical experiment show that the GA-VNS-TLBO approach better than conventional meta-heuristic approaches in terms of resilience and efficiency.

However, since the data used in numerical experiment are randomly generated, a study using more realistic data obtained by real world will be carried out, which will be left to my future study.

1. Introduction

1.1 Background and Objective of This Study

In a time when technology and internet environment are rapidly developing, it is more effective for enterprises to use supply chain (SC) model. The SC model consists of forward logistics (FL), which is responsible for the production and delivery of products. However, closed-loop supply chain (CLSC) is required in consideration of reverse logistics (RL) including the process of recycled products due to environmental pollution and lack of raw materials.

The various facilities utilized in the CLSC model are divided into two categories: FL and RL. In the FL, the finished goods are supplied to the customers by various facilities, such as manufacturers, distributors, and retailers. Whereas, in the RL, returned products from customers are recovered at recovery centers after collecting and checking at collection center, and the rests are handled at disposal centers. But, there exists a disruption risk in the SC and CLSC models for various reasons.

There are many conventional studies (Tang, 2006; Chopra and Sodhi, 2004; Waters, 2007; Baghalian et al., 2013; Chuluunsukh et al., 2021) that describe various disruption risks occurred in the SC and CLSC models. In general, there are two types of disruption risks: human-made disruption risk and natural-made disruption risk. Labor dispute, supplier bankruptcy, war and terrorism are considered as human-made disruption risk (Chopra and Sodhi., 2004). The most recent example was when Coronavirus broke out in China in 2020. Because of the pandemic, the production at the countries that buy raw materials from China suffered. Conversely, natural factors can create a disruption risk without relying on humans. Baghalian et al. (2013) and Chuluunsukh et al. (2021) mentioned real-world examples on natural-made disruption risks. In 1999, an earthquake in Taiwan caused Apple to cancel customer orders, and in 2001, an earthquake in Japan caused severe production losses for Toyota (Baghalian et al., 2013).

In a competitive business environment, it is more effective to consider disruption risks as well as various distribution channels. The distribution channel is the path that products are delivered to facilities or customers in an efficient and organized manners. In several conventional studies (Lin et al., 2009; Yun et al., 2018; Yun et al., 2020), distribution channels were classified into three types: Normal delivery (NDL), Direct delivery (DDL), and Direct shipping (DSP), which are considered in the SC and CLSC models. The NDL is the general distribution channel for distributing products from a facility to the next. The DSP is a distribution channel that products are directly delivered from manufacturers to customers without intermediate stages. The DDL delivers products from distribution centers (DCs) to customers without going through retailers.

However, some conventional studies (Jabbarzadeh et al., 2018; Chuluunsukh et al., 2021; Subramanian et al., 2013) did not consider the various distribution channels under the situation that disruption risks occur in the SC or CLSC models. Therefore, in this paper, a CLSC with supplier disruption risk (CLSC-DR) model is proposed. For various distribution channels, NDL and DDL are used in it.

Conventional studies on the SC or CLSC models with disruption risks are summarized as follows. Chuluunsukh et al. (2021) suggested a SC model which consists of supplier groups and manufacturer. The supplier groups have one main supplier and multiple backup suppliers. One backup supplier among the multiple ones will deliver parts to the manufacturer when main supplier is disrupted. The parts are supplied by the backup route of the main supplier when the main route of the main supplier is disrupted. Experimental results showed that increasing the number of the backup routes of the main supplier and the number of suppliers can reduce the overall cost for operating the SC model. Jabbarzadeh et al. (2018) designed a CLSC model with the FL consisting of suppliers, production centers, and first customers, as well as the RL consisting of collection centers, disposal centers, and secondary markets. They considered the disruption risks at suppliers, production center and collection centers, and shown that considering disruption risks when planning a CLSC model can save significant costs.

Complicated network problems including the SC or CLSC models are known as NP-complete (Savaskan, 2004; Gen et al., 2018; Yun et al., 2020). There have been many studies using meta-heuristic approaches to solve these complicated network problems and ensured the efficiency of the SC or CLSC models. Single meta-heuristic approaches such as Genetic algorithm (GA), Cuckoo search (CS), Variable neighborhood search (VNS), Particle swarm optimization (PSO) and Tabu search (TS) have shown to be more effective than other conventional approaches (Savaskan, 2004; Gen and Cheng, 2000; Gen et al., 2018; Yun et al., 2018, 2020). Recently, hybrid meta-heuristics approaches which combine two or more single meta-heuristic approaches have been developed and applied to the complicated network problems. Many conventional studies have proved that applying single or hybrid meta-heuristic approaches to the complicated network problems is an efficient approach (Lin et al., 2009; Zhang et al., 2012; Soleimani and Kannan, 2015; Xinyu and Liang, 2016).

Conventional studies with hybrid meta-heuristic approaches for the CLSC models are briefly summarized as follows. Soleimani and Kannan (2015) proposed a CLSC model which considers various distribution channels. They used a hybrid meta-heuristic approach that combines the GA and PSO and proved that the approach is more effective than other conventional approaches. Yun (2020) suggested a sustainable CLSC model for mobile phone. The sustainable CLSC model can be considered as a multi-objective optimization problem, and a hybrid GA (HGA) approach which combines the GA and CS was used to solve the sustainable CLSC models with various scales in numerical experiments. He demonstrates that the HGA approach outperforms conventional approaches and that the sustainable CLSC model using various distribution channels is more effective than the sustainable CLSC model using a single distribution channel.

In general, each meta-heuristic approach has its advantages and weaknesses, so it is imperative to use a combination of meta-heuristic approaches that can overcome these weaknesses. For example, the GA creates population diversity and elite populations due to the randomness in the GA implementation, but some poor individuals can be generated in the population. These poor individuals can be eliminated, and the more respective individuals can be maintained in the population by applying the teaching and learning based-optimization (TLBO) to the GA loop (Rabeh et al., 2019). Rabeh et al. (2019) demonstrated that the combination of the GA and TLBO approaches is more effective than the GA alone or the TLBO alone. As another meta-heuristic approach, the variable neighborhood search (VNS) approach is to seek global optimal solution by identifying the optimality of its descent stage. It can also be used to get rid of a valley and transform a neighborhood (Chen et al., 2020). Qiuhua et al. (2015) considered a hybrid meta-heuristic approach combining the VNS with TLBO. In this approach, the TLBO is used for global search, whereas the VNS is used for local search and strengthens the solution obtained by the global search, which can achieve the appropriate balance between exploitation and exploration. By combining the TLBO with VNS, the opportunity to find optimal or near-optimal solutions can be increased. As described above, many studies (Chen et al., 2020; Qiuhua et al., 2015; Dib et al., 2015; Gen et al., 2018; Yun et al., 2020) showed that using a hybrid meta-heuristic approach is more efficient than using a single-meta-heuristic approach.

Therefore, in this paper, the GA-VNS-TLBO approach as a hybrid meta-heuristic one is proposed. The proposed GA-VNS-TLBO approach is composed of the learning capability of the TLBO approach, the global search capability of the GA approach, and the local search capability of the VNS approach. The proposed GA-VNS-TLBO approach is applied to the CLSC-DR model with various scales and its performance is compared with those of conventional single and hybrid meta-heuristic approaches.

1.2 Implementation Procedure of This Study

The purpose of this study is to propose an efficient CLSC model, called the CLSC-DR model, where disruption risk in supplier and two distribution channels (NDL and DDL) are considered. The proposed CLSC-DR model is represented as a mathematical formulation and implemented using the GA-VNS-TLBO approach.

First, after examining the conventional studies that consider the CLSC model, the characteristics of these studies are analyzed. However, most of the existing conventional CLSC models do not consider both disruption risks and distribution channels. To cope with these weaknesses, the CLSC-DR model with various distribution channels is proposed in this paper.

Second, the material flow of the proposed CLSC-DR model is presented in the form of a network. The proposed CLSC-DR model is represented as a mathematical formulation. In the mathematical formulation, the total cost which is consisted of transportation cost, fixed cost and handling cost is minimized for objective function. Various constraints such as transportation amount constraint, facility usage constraints, etc. are used for optimizing the objective function.

Third, as a hybrid meta-heuristic approach for implementing CLSC-DR models, a GA-VNS-TLBO approach for implementing mathematical formulation of CLSC-DR models is proposed.

Fourth, in numerical experiments, the CLSC-DR model with various scales is used to compare the performance of the GA-VNS-TLBO approach with those of conventional single and hybrid meta-heuristic approaches.

Fifth, through the above research purpose and methodology, the following results can be concluded.

a) Most of conventional CLSC models do not consider disruption risk and various distribution channels. Therefore, this study proves the superiority of the CLSC-DR model by considering various distribution channels.
b) By the comparative analysis between the GA-VNS-TLBO approach and conventional meta-

heuristic approaches and, the former's superiority is proved.

c) In future study, more realistic data are collected to improve the practical applicability of the GA-VNS-TLBO approach.

2. Conventional Studies on CLSC Model with Disruption Risk

Many conventional studies (Xiao and Yu, 2006; Trkman et al., 2009; Wilson, 2007; Wang et al., 2012; Gedik et al., 2014; Jabbarzadeh et al., 2018; Ma et al., 2016; Badejo et al., 2022) which consider the CLSC model including SC ones have concentrated either on the disruption of facility or on the disruption of route. In reality, various scenarios that both the facilities and routes are disrupted occur in the CLSC model simultaneously. In general, a CLSC model has various facilities (suppliers, manufacturers, DCs, retailers, etc.) in its each stage, and if one of these entities is unavailable due to a supplier disruption, the other route connecting the manufacturers is also disrupted, resulting in the whole network being disrupted. Therefore, the efficiency of the CLSC model can be improved by considering the supplier and route disruptions simultaneously. Some literatures (Oke et al., 2009; Tang, 2006; Kleindorfer et al., 2005; Azaron et al., 2021; Baghalian et al., 2013; Poudel et al., 2016; Ramshani et al., 2019; An et al., 2015; Aghamohamadi-Bosjin et al., 2022; Chuluunsukh et al., 2021) have considered these two factors in their SC or CLSC models.

As mentioned above, simultaneous disruption of supplier and route can disrupt the entire CLSC model, so alternative (or backup) suppliers or routes should be considered to avoid them. Wang et al. (2012) and Gedik et al. (2014) considered backup routes to cope with main route disruption, but they excluded the probabilistic disruptions for the main route. On the other hand, Jabbarzadeh et al. (2018) and Badejo et al. (2022) considered backup suppliers to cope with main suppliers, but only Badejo et al (2022) considered probabilistic disruptions to main suppliers. Differing from above mentioned studies, Aghamohamadi et al. (2022) and Chuluunsukh et al. (2021) considered a backup supplier and route as a result of the main supplier and route disruptions. They also offered a mathematical model to represent the CLSC models with probabilistic disruptions in the main supplier and route.

Looking at existing studies, various approaches have been used. For example, it can be divided as conventional approaches and meta-heuristic approaches. Conventional studies (Azaron et al., 2021; Wilson, 2007; Wang et al., 2012; Gedik et al., 2014; Baghalian et al., 2013; Poudel et al., 2016; Jabbarzadeh et al., 2018; Ma et al., 2016; An et al., 2015) used various conventional approaches such as a multi-objective two stage stochastic, simulation, heuristics-NPM (Nested Partition Approach), piecewise linearization, lagrangian relaxation. On the other hand, Ramshani et al. (2019), Aghamohamadi et al. (2022) and Chuluunsukh et al. (2021) used the meta-heuristic approaches such as Tabu search (TS), Route subset selector (RSS), population-based multi objective particle swarm optimization - social engineering optimizer (HPSO-SEO), GA-VNS.

The major chracteristics of conventional studies including our proposed study are summarized in Table 2.1.

Table 2.1 Summary of Conventional Studies on Disruption Risk in SC or CLSC Models

	Disruption		Alternative		Probabilistic Disruption		Math. Model	Distribution channel		Reverse logistic	Approach
	Main		Backup								
	Facility	Route	Facility	Route	Facility	Route		Normal delivery	Direct delivery		
Oke et al (2009)	■	■									Conceptual Study
Tang (2006)	■	■									Conceptual Study
Kleindorfer et al (2005)	■	■									Conceptual Study
Azaron et al (2021)	■	■									A multi-objective two stage stochastic
Xiao and Yu (2006)	■										Conceptual Study
Trkman et al (2009)	■										Conceptual Study
Wilson (2007)		■									Simulation
Wang et al. (2012)		■		■			■				Heuristics-NPM
Gedik et al. (2014)		■		■			■				MIP
Baghalian et al. (2013)	■	■	■				■				Piecewise linearisation
Poudel et al (2016)	■	■		■		■	■				BDA
Jabbarzadeh et al (2018)	■		■				■			■	Lagrangian relaxation
Ramshani et al (2019)	■	■	■	■	■		■				Metaheuristics-TS, RSS
Ruimin et al (2016)	■						■			■	LP-metrics approach
An et al. (2015)	■	■			■		■				Lagrangian relaxation
Badejo et al (2022)	■		■		■		■				Two-stage stochastic model
Aghamohamadi et al. (2022)	■	■	■	■	■	■	■			■	Metaheuristics-HPSO-SEO
Chuluunsukh et al. (2021)	■	■	■	■	■	■	■				Metaheuristics-GA+VNS
This paper	■	■	■	■	■	■	■	■	■	■	Metaheuristics-GA+VNS+TLBO

* RSS: Route Subset Selector
* BDA: Benders decomposition algorithm
* NPM: nested partition approach
* HPSO-SEO: Population-based Multi Objective Particle Swarm Optimization (PMOPSO) + Social Engineering Optimizer (SEO)
* MIP: two-stage mixed integer programming

Among the conventional studies mentioned in Table 2.1, some featured studies are detailly analyzed. Chuluunsukh et al. (2021) proposed a SC model, which considers the various risks associated with the operation of facilities and routes. The SC model is composed of supplier groups and manufacturer. Each supplier group has its own main and backup routes. These routes are utilized by one main supplier and two backup suppliers. The material flows for the SC model is shown in Figure 2.1. The four types of parts (i.e., part type 1, 2, 3, and 4) that are sent to the manufacturer are prepared in four supplier groups, that is, part type 1 is prepared at the main supplier of supplier group 1, the part type 2 at the main supplier of supplier group 2, the part type 3 at the main supplier of supplier group 3, and the part type 4 at the main supplier of supplier group 4. Each of these groups has its main supplier and two backup suppliers. If the main supplier or main route in supplier group 1 gets completely disrupted with a 100% probability, then one of the two backup suppliers will send the part type 1 to the manufacturer. On the other hand, if the main supplier or main route gets partially disrupted with a 50% probability, then one of the two backup suppliers will send the remaining half of the order to the manufacturer. This SC model was represented using a mathematical formulation, where the objective function is to minimize the total cost which is consist of the sum of fixed cost, transportation cost, and handling cost. The hybrid meta-heuristic (pGA-VNS) approach using the VNS and GA was applied for implementing the mathematical formulation. In numerical experiments, the pGA-VNS approach showed to be more efficient than some conventional meta-heuristic approaches such as GA and VNS.

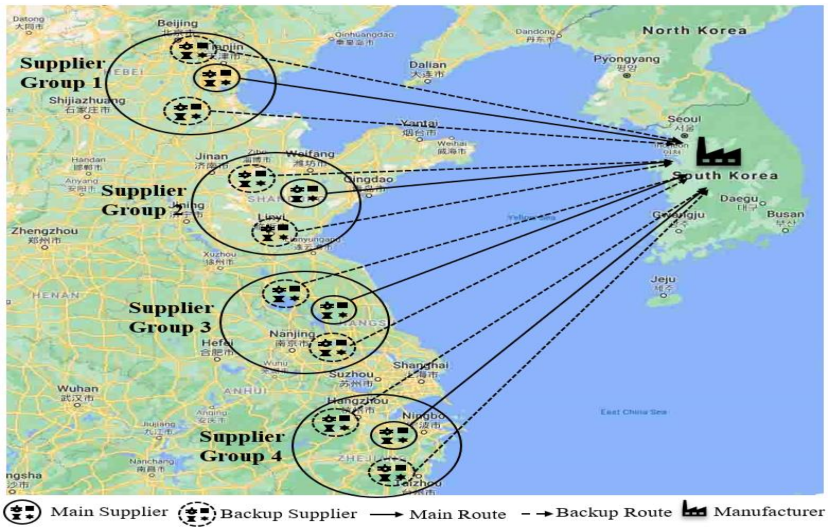

Fig 2.1 Material Flow of the SC model (Chuluunsukh et al., 2021)

Azaron et al. (2021) constructed a small SC model including one supplier base, four manufacturing sites, three markets and four potential locations where warehouses could be built. The conceptual material flow is shown in Figure 2.2. It was assumed that two types of products are distributed at the markets. There were three limited resources that are used to produce these products. The objective of this study was to minimize the travel times and maximize the expected value of the SC model under uncertainty situation by applying a multi-objective two stage programming approach.

In the SC model, the optimal locations of retailers and warehouses as well as the production levels and shipping quantities at various manufacturing sites and warehouses were determined.

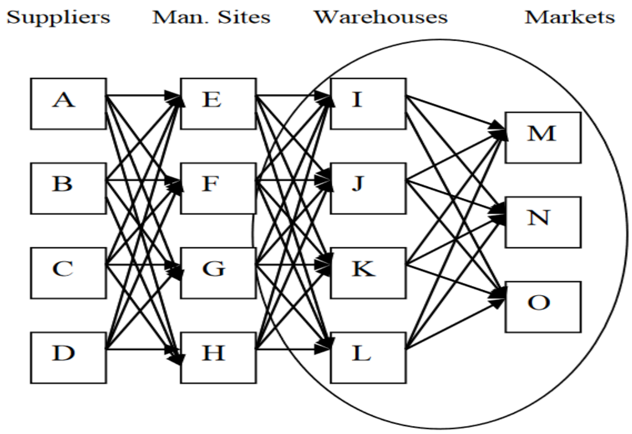

Fig 2.2 Material Flow of the SC model (Azaron et al., 2021)

Jabbarzadeh et al. (2018) presented a stochastic robust optimization model that can be used to design a CLSC model with disruption risks. The CLSC model is consist of suppliers, production centers, and first customers in the FL and collection centers, disposal centers, and secondary markets in the RL. They considered a disruption risk at suppliers, production center and collection centers, and proved that considering disruption risk when planning a CLSC model can save significant costs. The use of a stochastic robust optimization model can help minimizing the total cost of the CLSC model in different scenarios. It can also help the implementation of the CLSC model in coping with the effects of disruptions. The conceptual material flow for the CLSC model is shown in Figure 2.3. The CLSC model was represented a mathematical formulation and implemented using a stochastic robust optimization model. The Lagrangian relaxation approach as a stochastic robust optimization model was used to improve the efficiency of the CLSC model. Real-world data was used in the CLSC model for glass industry and then the efficiency of the use of Lagrangian relaxation approach was analyzed. The experimental results showed that use of lateral transshipment in the CLSC model can help reducing the overall cost.

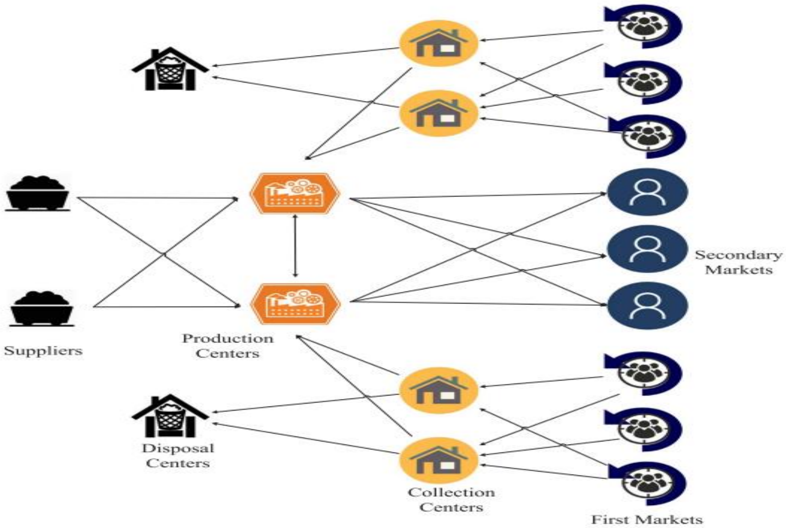

Fig 2.3 Material Flow of the CLSC model (Jabbarzadeh et al., 2018)

Ma et al. (2016) developed a CLSC model which includes plants, collection centers, demand zones and disposal facilities. New products can be manufactured and returned products can be remanufactured at the plants. Products are shipped from the plants to the demand zones, while the returned products from demand zones are sent to the collection centers. The conceptual material flow for the CLSC model is shown in Figure 2.4. The assumptions used in the CLSC model are as follows: (1) It is designed to provide a single period, (2) All of the products returned from the demand zones are collected at the collection centers, and (3) The locations of the demand zones with fixed capacities are fixed and plants being known in advance. Two objective functions were used for the implementation of the CLSC model. First objective function as an economic factor is to minimize the total cost which is the sum of the fixed costs, transportation costs, and production costs. Second one as an environmental factor is to minimize the environmental cost. These two objective functions including some constraints was represented as a multi-objective mixed integer programming model. A LP- metrics approach was used to solve the model.

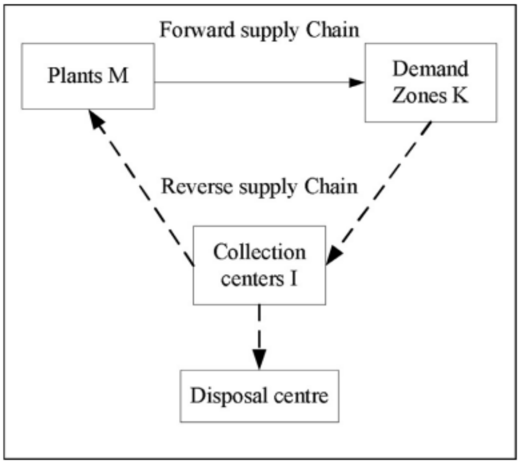

Fig 2.4 Material Flow of the CLSC model (Ma et al., 2016)

As conventional studies mentioned and analysed above, we can summarize some weakness as follows:

- The studies (Oke et al., 2009; Tang, 2006; Kleindorfer et al., 2005; Xiao and Yu, 2006; Trkman et al., 2009) does not provided a comprehensive analysis such as the efficiency analysis of the SC model. Instead, their studies ware merely conceptual ones.
- Some studies (Oke et al., 2009; Tang, 2006; Kleindorfer et al., 2005; Azaron et al., 2021; Xiao and Yu, 2006; Trkman et al., 2009; Wilson, 2007; Ma et al., 2016) did not consider backup suppliers or backup routes that can replace the main supplier or main routes, if there was a disruption risk. In addition, most studies (Oke et al., 2009; Tang, 2006; Kleindorfer et al., 2005; Azaron et al., 2021; Xiao and Yu, 2006; Trkman et al., 2009; Wilson, 2007; Wang et al., 2012; Ma et al., 2016; Gedik et al., 2014; Baghalian et al., 2013; Jabbarzadeh et al., 2018) did not consider probabilistic disruption risk.
- All studies did not use various distribution channels in their SC or CLSC models.
- Many studies (Azaron et al., 2021; Wilson, 2007; Wang et al., 2012; Gedik et al., 2014; Baghalian et al., 2013; Poudel et al., 2016; Jabbarzadeh et al., 2018; Ma et al., 2016; An et al., 2015) did not consider the use of meta-heuristic approaches, although the use of various meta-heuristic approaches is more efficient then the use of conventional approaches.

To cope with these weakness summrized above, the following two solutions should be considered.

First, additional or backup suppliers and routes are needed, if a main supplier or route is disrupted with a probabilistic disruption risk. Second, the consideration of various distribution channels can ensure that products are delivered to customer as quickly as possible. We will use the hybrid meta-heuristic approach to solve these complex problems. In this paper, we present CLSC model that consider the two solutions mentioned earlier. Our proposed model efficiently incorporates both of these approaches to enhance the effectiveness.

3. Design of the Proposed CLSC-DR Model

The network structure of the proposed CLSC-DR model is shown in Figure 3.1. Part supplier groups, module manufacturers, manufacturers, DCs, retailers, and costumers are all included in the FL.

Each part supplier group has one main supplier and two backup suppliers, where main and backup suppliers can send a part type to module manufacturer or manufacturer using the main route of main supplier and the backup routes of backup suppliers. For example, the main supplier at part supplier group 1 sends a part type 1 to module manufacturer using its main route. However, if the main supplier or its main route is disrupted, then one of the backup suppliers will send the part type 1 to the module manufacturer using its backup route. A similar situation is also shown in the part supplier group 2, that is, the main supplier at part supplier group 2 sends a part type 2 to module manufacturer using its main route. However, if the main supplier or its main route is disrupted, then one of the backup suppliers will send the part type 2 to the module manufacturer using its backup route. For part supplier group 3, the main supplier sends a part type 3 to the manufacturer using its main route. However, if the main supplier or its main route is disrupted, then one of the two backup suppliers sends the part type 3 to the manufacturer using its backup route.

The module manufacturer uses part type 1 and 2 to produce a module and send it to the manufacturer. The manufacturer uses the module and part type 3 to produce a product. The product is delivered to customers through DCs and retailers.

The RL consists of collection center, recovery center, secondary customer and disposal center. The product returned by customer is collected and sorted at the collection center. Unrecoverable and recoverable parts are obtained after collecting and sorting the returned product. The unrecoverable parts are sent to the disposal center, and the recoverable parts are sent to the recovery center. At the recovery center, the quality and function of the recoverable parts are recovered and then they are classified into three types (recovered products, modules, and parts). The recovered products are sent to the secondary customer, the recovered modules to the manufacturer, and the recovered parts to the module manufacturer.

As already mentioned in Section 1, various distribution channels such as the NDL and DDL can be used in the CLSC models to improve their transportation efficiency. Therefore, both the NDL and the DDL are used in the proposed CLSC-DR model. The NDL in the proposed CLSC-DR model begins with the supplier. The products produced by the manufacturer are then transported to the retailer through the DC before being delivered to the customers. And the collection center, recovery center, secondary customer, and disposal center are also connected with the NDL. The DDL means that products at the DC are delivered directly to the customers without involving any retailers in the process.

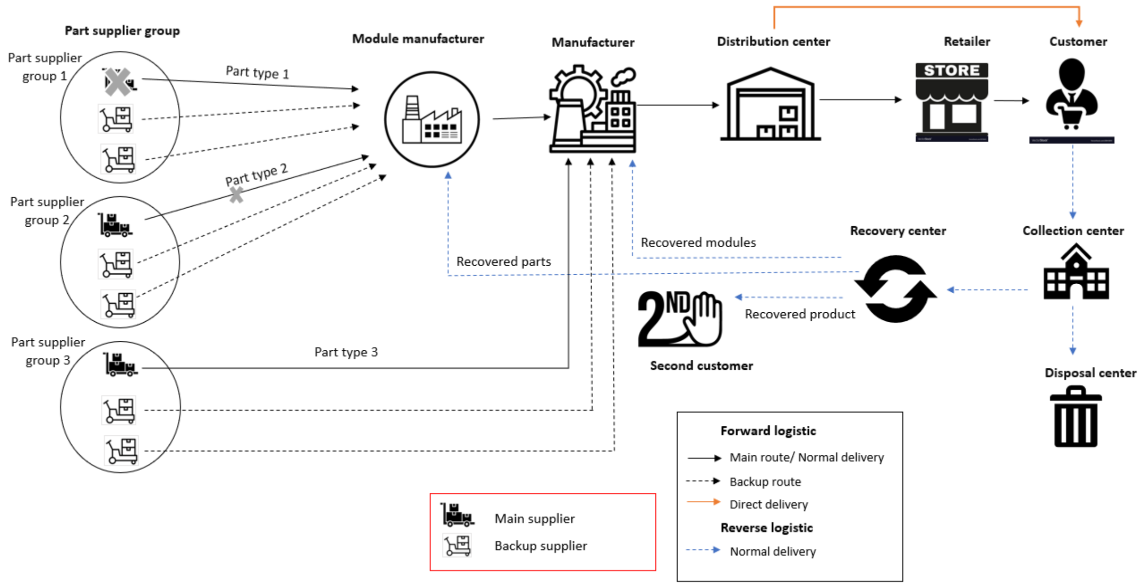

Fig 3.1 A conceptual flow of the proposed CLSC-DR model

The differences between conventional studies and the proposed CLSC-DR model are as follows.

- In the proposed CLSC-DR model, backup suppliers and their routes can be used, when a main supplier or its route is disrupted. These alternative considerations using main or backup supplier and main or backup route have not been treated in most of conventional studies.
- For various transportation types, the proposed CLSC-DR model uses the NDL and DDL simultaneously, which can improve the efficiency in operating or managing the proposed CLSC-DR model. However, most of conventional studies have not considered various transportation types.

4. Mathematical Formulation

The following assumptions are used for representing the proposed CLSC-DR model.

- The production of a single product is only considered.
- The numbers of facility at each stage are already known. Among them, only one facility of the part suppliers, module manufacturers, manufacturers, distribution centers, retailers, collection centers and recovery centers should be opened at each stage, whereas, all facilities of the customers, second customers, and disposal centers are always opened.
- One main supplier and more than one backup supplier at each part supplier group are considered.
- One main supplier at each part supplier group is opened, when there is no disruption at each part supplier group, while, one of the backup suppliers is opened, when the main supplier at a randomly selected supplier group is disrupted. As a same meaning, the main route of one main supplier at each part supplier group is opened, when there is no disruption at each part supplier group, while, the backup route of one of the backup suppliers is opened, when the main route of the main supplier at a randomly selected supplier group is disrupted.
- Fixed costs for operating the facilities which can be opened at each stage are different and already known.
- Unit handling costs of the facilities which can be opened at each stage are already known and are identical at the same stage.
- Unit transportation costs between each facility of each stage are already known and are different.
- Eighty percent (80%) of the products returned from customer are collected at collection center.
- The quality of the recovered products, recoverable modules, and recoverable parts at the recovery center are identical with those of new products, modules and parts.

The following defines index set, parameters, and decision variables.

- **Index Set**

s : index of main supplier, $s \in S$

s': index of backup supplier, $s' \in S'$

g : index of part supplier group, $g \in G$

t : index of main route, $t \in T$

t' : index of backup route, $t' \in T'$

o : index of module manufacturer, $o \in O$

m : index of manufacturer, $m \in M$

d : index of distribution center, $d \in D$

r : index of retailer, $r \in R$

c : index of customer, $c \in C$

l : index of collection center, $l \in L$

w : index of disposal center, $w \in W$

e : index of recovery center, $e \in E$

u : index of second customer, $u \in U$

- **Parameters**

F_{gs}: fixed cost at main supplier s of part supplier group g

$F_{gs'}$: fixed cost at backup supplier s' of part supplier group g

F_o: fixed cost at module manufacturer o

F_m: fixed cost at manufacturer m

F_d: fixed cost at distribution center d

F_r: fixed cost at retailer r

F_l: fixed cost at collection center l

F_e: fixed cost at recovery center e

H_{gs}: unit handling cost at main supplier s of supplier group g

$H_{gs'}$: unit handling cost at backup supplier s' of supplier group g

H_o: unit handling at module manufacturer o

H_m: unit handling at manufacturer m

H_d: unit handling at distribution center d

H_r: unit handling at retailer r

H_l: unit handling at collection center l

H_e: unit handling at recovery center *e*

T_{gsot}: unit transportation cost from main supplier *s* of part supplier group *g* to module manufacturer *o* using main route *t*

$T_{gs'ot'}$: unit transportation cost from backup supplier s' of part supplier group g to module manufacturer *o* using backup route t'

T_{gsmt}: unit transportation cost from main supplier *s* of part supplier group *g* to manufacturer *m* using main route *t*

$T_{gs'mt'}$: unit transportation cost from backup supplier s' of part supplier group *g* to manufacturer *m* using backup route t'

T_{om}: unit transportation cost from module manufacturer *o* to manufacturer *m*

T_{md}: unit transportation cost from manufacturer *m* to distribution center *d*

T_{dr}: unit transportation cost from distribution center *d* to retailer *r*

T_{dc}: unit transportation cost from distribution center *d* to customer *c*

T_{rc}: unit transportation cost from retailer *r* to customer *c*

T_{cl}: unit transportation cost from customer *c* to collection center *l*

T_{lw}: unit transportation cost from collection center *l* to disposal center *w*

T_{le}: unit transportation cost from collection center *l* to recovery center *e*

T_{eo}: unit transportation cost from recovery center *e* to module manufacturer *o*

T_{em}: unit transportation cost from recovery center *e* to manufacturer *m*

T_{eu}: unit transportation cost from recovery center *e* to second customer *u*

q_{gsot}: transporting quantity from main supplier *s* of part supplier *g* to module manufacturer *o* using main route *t*

$q_{gs'ot'}$: transporting quantity from backup supplier s' of part supplier group g to module manufacturer *o* using backup route t'

q_{gsmt}: transporting quantity from main supplier *s* of part supplier group *g* to manufacturer *m* using main route *t*

$q_{gs'mt'}$: transporting quantity from backup supplier s' of part supplier group g to manufacturer *m* using backup route t'

q_{om}: transporting quantity from module manufacturer *o* to manufacturer *m*

q_{md}: transporting quantity from manufacturer *m* to distribution center *d*

q_{dr}: transporting quantity from distribution center *d* to retailer *r*

q_{dc}: transporting quantity from distribution center *d* to customer *c*

q_{cl}: transporting quantity from customer *c* to collection center *l*

q_{lw}: transporting quantity from collection center *l* to disposal center *w*

q_{le}: transporting quantity from collection center *l* to recovery center *e*

q_{eo}: transporting quantity from recovery center *e* to module manufacturer *o*

q_{em}: transporting quantity from recovery center *e* to manufacturer *m*

q_{eu}: transporting quantity from recovery center *e* to second customer *u*

cap_o: capacity of module manufacturer *o*

cap_m: capacity of manufacturer *m*

cap_d: capacity of distribution center *d*

cap_r: capacity of retailer *r*

cap_c: capacity of customer *c*

cap_l: capacity of collection center *l*

cap_e: capacity of recovery center *e*

cap_w: capacity of disposal center *w*

cap_u: capacity of second customer *u*

- **Decision variable**

j_{gs}: takes the value 1 if main supplier *s* at part supplier group *g* is available and 0 otherwise.

$j_{gs'}$: takes the value 1 if backup supplier s' at part supplier group *g* is available and 0 otherwise.

k_{gst}: takes the value 1 if main route *t* of main supplier *s* at part group *g* is available and 0 otherwise.

$k_{gs't'}$: takes the value 1 if backup route t' of backup supplier s' at part supplier group *g* is opened and 0 otherwise.

j_o: takes the value of 1 if module manufacturer *o* is opened and 0 otherwise

j_m: takes the value of 1 if manufacturer *m* is opened and 0 otherwise

j_d: takes the value of 1 if distribution center *d* is opened and 0 otherwise

j_r: takes the value of 1 if retailer *r* is opened and 0 otherwise

j_l: takes the value of 1 if collection center *l* is opened and 0 otherwise

j_e: takes the value of 1 if recovery center *e* is opened and 0 otherwise

The objective function is to minimize the total cost (*TC*) which consists of total transportation cost (*TT*), total handling cost (*TH*) and total fixed cost (*TF*) as follows:

$$Min\ TC = TF + TH + TT \tag{1}$$

The *TF* consists of the sum of the costs of establishing and opening part supplier groups, module manufacturers, manufacturers, distribution centers, retailers, collection centers, recovery centers. For example, the fixed cost of the main supplier at part supplier groups is calculated by the fixed cost (F_{gs}) of the main supplier and whether it is opened (j_{gs}). This is expressed in the following formula.

$$\begin{aligned} TF = & \textstyle\sum_g \sum_s F_{gs} * j_{gs} + \sum_g \sum_{s'} F_{gs'} * j_{gs'} + \\ & \textstyle\sum_o (F_o * j_o) + \sum_m (F_m * j_m) + \sum_d (F_d * j_d) + \\ & \textstyle\sum_r (F_r * j_r) + \sum_l (F_l * j_l) + \sum_e (F_e * j_e) \end{aligned} \tag{2}$$

The *TH* is determined by the handling capacity generated by each stage including part supplier groups, module manufacturers, manufacturers, distribution centers, retailers, collection centers and recovery centers, as well as whether each one is opened or not. For example, the handling cost incurred by the main supplier is calculated as the handling cost per part unit in the main supplier (H_{gs}), the handling quantity (q_{gsot}), and whether the main supplier is opened or not (j_{gs}). This is expressed as the following formula.

$$\begin{aligned} TH = & \textstyle\sum_g \sum_s (H_{gs} * q_{gsot} * j_{gs}) + \sum_g \sum_{s'} (H_{gs'} * q_{gs'ot'} * j_{gs'}) + \\ & \textstyle\sum_o (H_o * cap_o * j_o) + \sum_m (H_m * cap_m * j_m) + \\ & \textstyle\sum_d (H_d * cap_d * j_d) + \sum_r (H_r * cap_r * j_r) + \\ & \textstyle\sum_l (H_l * cap_l * j_l) + \sum_e (H_e * cap_e * j_e) \end{aligned} \tag{3}$$

The *TT* is incurred when all products or parts are transported or delivered between the facilities at each stage. For example, the cost of supplying a part from the main supplier to the module manufacturer is expressed in terms of the unit transportation cost (T_{gsot}), transporting quantity supplied from the main supplier to the module manufacturer (q_{gsot}), whether j_o, j_{gs} , and k_{gst} is opened or not. This is expressed as the following formula.

$$\begin{aligned} TT = & \sum_g \sum_s \sum_o \sum_t T_{gsot} * q_{gsot} * j_{gs} * j_o * k_{gst} + \\ & \sum_g \sum_{s'} \sum_o \sum_{t'} T_{gs'ot'} * q_{gs'ot'} * j_{gs'} * j_o * k_{gs't'} + \\ & \sum_g \sum_s \sum_m \sum_t T_{gsmt} * q_{gsmt} * j_{gs} * j_m * k_{gst} + \\ & \sum_g \sum_{s'} \sum_m \sum_{t'} T_{gs'mt'} * q_{gs'mt'} * j_{gs'} * j_m * k_{gs't'} + \\ & \sum_o \sum_m (T_{om} * q_{om} * j_o * j_m) + \sum_m \sum_d (T_{md} * q_{md} * j_m * j_d) + \\ & \sum_d \sum_r (T_{dr} * q_{dr} * j_d * j_r) + \sum_d \sum_c (T_{dc} * q_{dc} * j_d) + \\ & \sum_r \sum_c (T_{rc} * q_{rc} * j_r) + \sum_c \sum_l (T_{cl} * q_{cl} * j_l) + \sum_l \sum_w (T_{lw} * q_{lw} * j_l) + \\ & \sum_l \sum_e (T_{le} * q_{le} * j_l * j_e) + \sum_e \sum_o (T_{eo} * q_{eo} * j_e * j_o) + \\ & \sum_e \sum_m (T_{em} * q_{em} * j_e * j_m) + \sum_e \sum_u (T_{eu} * q_{eu} * j_e) \end{aligned} \tag{4}$$

Constraints for optimizing the objective function are as follows.

$$\begin{aligned} & (\sum_g \sum_s \sum_o \sum_t q_{gsot} * j_{gs} * j_o * k_{gs} + \\ & \sum_g \sum_{s'} \sum_o \sum_{t'} q_{gs'ot'} * j_{gs'} * j_o * k_{gs'} + \\ & \sum_e \sum_m q_{em} * j_e * j_m) - \sum_o cap_o * j_o \leq 0 \end{aligned} \tag{5}$$

$$\begin{aligned} & (\sum_g \sum_s \sum_m \sum_t q_{gsmt} * j_{gs} * j_m * k_{gs} + \\ & \sum_g \sum_{s'} \sum_m \sum_{t'} q_{gs'mt'} * j_{gs'} * j_m * k_{gs'} + \\ & \sum_e \sum_m q_{em} * j_e * j_m) - \sum_m cap_m * j_m \leq 0 \end{aligned} \tag{6}$$

$$\sum_m \sum_d q_{md} * j_m * j_d) - \sum_d cap_d * j_d \leq 0 \tag{7}$$

$$\sum_d \sum_r q_{dr} * j_d * j_r) - \sum_r cap_r * j_r \leq 0 \tag{8}$$

$$(\sum_d \sum_c q_{dc} * j_d + \sum_r \sum_c q_{rc} * j_r) - \sum_c cap_c \leq 0 \tag{9}$$

$$\sum_c \sum_l q_{cl} * j_l - \sum_l cap_l * j_l \leq 0 \tag{10}$$

$$\sum_l \sum_w q_{lw} * j_l * j_w - \sum_w cap_w * j_w \leq 0 \tag{11}$$

$$\sum_l \sum_e q_{le} * j_l * j_e - \sum_e cap_e * j_e \leq 0 \tag{12}$$

$$\sum_e \sum_u q_{eu} * j_e - \sum_u cap_u * j_u \leq 0 \tag{13}$$

In the equations (5) to (13), there is a quantity limitation for transportation between each stage. For example, the total number of the parts sent by the main supplier (q_{gsmt}), backup supplier ($q_{gs'mt'}$) and recovery center (q_{em}) must be less than or equal to the number of parts processed by the module manufacturer (cap_o).

$$\sum_s j_{gs} + \sum_{s'} j_{gs'} = 1, \qquad \forall g \tag{14}$$

$$\sum_s \sum_t k_{gst} + \sum_s \sum_t k_{gs't'} = 1, \qquad \forall g \tag{15}$$

$$\sum_o j_o = 1 \tag{16}$$

$$\sum_m j_m = 1 \tag{17}$$

$$\sum_d j_d = 1 \tag{18}$$

$$\sum_r j_r = 1 \tag{19}$$

$$\sum_l j_l = 1 \tag{20}$$

$$\sum_e j_e = 1 \tag{21}$$

In the equations (14) to (21), only one facility should be opened at each stage.

$$j_{gs} = \{0,1\}, \qquad \forall s \in S,\ \forall g \in G \tag{22}$$

$$j_{gs'} = \{0,1\}, \qquad \forall s' \in S',\ \forall g \in G \tag{23}$$

$$k_{gst} = \{0,1\}, \qquad \forall s \in S,\ \forall g \in G,\ \forall t \in T \tag{24}$$

$$k_{gs't'} = \{0,1\}, \qquad \forall s' \in S',\ \forall g \in G,\ \forall t' \in T' \tag{25}$$

$$j_o = \{0,1\}, \qquad \forall o \in O \tag{26}$$

$$j_m = \{0,1\}, \qquad \forall m \in M \tag{27}$$

$$j_d = \{0,1\}, \qquad \forall d \in D \tag{28}$$

$$j_r = \{0,1\}, \qquad \forall r \in R \tag{29}$$

$$j_l = \{0,1\}, \qquad \forall l \in L \tag{30}$$

$$j_e = \{0,1\}, \qquad \forall e \in E \tag{31}$$

In the equations (22) to (31), each decision variable should take a value of 0 or 1.

5. Proposed GA-VNS-TLBO Approach

5.1 Background of Meta-heuristic Approaches

Meta-heuristic approaches can help in solving optimization problems. Numerous meta-heuristic approaches have been consistently developed since the 1960s. Most of meta-heuristic approaches have been developed inspired by natural phenomena, and the information obtained from past explorations is used for the next generation. They can be distinguished as several types as follows.

First, evolutionary algorithms, evolutionary strategies, and genetic algorithms (GA) are developed to mimic the evolutionary processes of nature. Second, particle swarm optimization (PSO), ant colony optimization (ACO), and cuckoo search (CS) algorithms are used to mimic the behavior of living organisms. Third, Tabu search (TS) and simulated annealing (SA) algorithms mimics natural and social phenomena. Fourth, hill climbing (HC), variable neighborhood search (VNS), and iterated local search (ILS) algorithms are used to improve the solution by exploring its neighbors through systematic iterations (Kim 2017). Including the algorithms mentioned above, the relationship between various meta-heuristic approaches is explained in Figure 5.1.

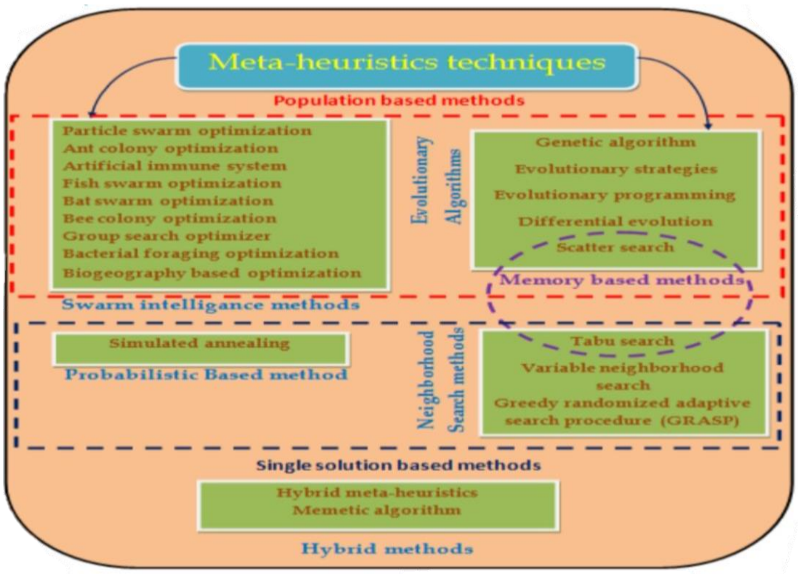

Fig 5.1 Relationship between meta-heuristic approaches

5.2 Structure of the GA-VNS-TLBO Approach

The proposed GA-VNS-TLBO approach is a hybrid meta-heuristic approach that combines the benefits of three different approaches. This means that the proposed GA-VNS-TLBO approach can be constructed by combining the global search capabilities of the GA approach, the local search capabilities of the VNS, and the learning capabilities of the TLBO approach. First, we explore the features of GA, VNS, and TLBO concerning their suitability for hybridization.

5.2.1 GA Approach

The GA approach was first developed by the Holland (1975) and based on phenomena in the course of natural evolution and stochastic optimization techniques. Since it has been improved by the studies such as Goldberg (1985) and Gen (1997), it is still actively working. Most of meta-heuristic approaches use approaches to obtain optimal solutions by initiating a search from one solution and improving it. The GA approach differs from other approaches in that it uses a population of different solutions and makes further improvements to find the best solution.

The procedure of the GA approach first uses the individual (or chromosome) to express the problem considered, resulting in the production of an initial population (P). After genetic manipulation by applying crossover and mutation to the initial population, the offspring (O) is produced. Fitness evaluation is performed on the produced offspring to select a new population that satisfies constraints. This process is repeated until the total number of generations is reached to a pre-defined maximum number of generations. (Gen and Cheng, 1997). The general procedure for implementing the GA is shown in Figure 5.2.

The GA approach with global search ability is one of the approaches to effectively solve large NP-Hard problems that conventional approaches cannot solve. It is also an approach to explore global optimization in a complex search space, and has been applied to a variety of applications to demonstrate efficiency. Dehghanian (2009) study used a GA approach to optimize the Sustainable Supply Chain Network problem. A study by Kannan (2010) designed a CLSC model for battery recycling and optimized it using a GA approach. A study by Yun (2013) proposed a GA approach to evaluate the reverse logistics network.

```
procedure: GA-approach
input: GA parameters
begin
    generation number t ←0;
    initialize population P(t);
    evaluate P(t);
    while (not termination condition) do
        create offspring C(t) from P(t) by crossover operator;
        create C(t) from P(t) by mutation operator;
        evaluate C(t);
        select P(t+1) from P(t) and C(t) by selection operator;
        t ← t + 1
     end
output: a best solution
end
```

Fig 5.2 Pseudo code of the GA approach

However, GA shows a lack of proper memory and learning capability for superior individuals generated during evolution and a relatively slow convergence process compared to other meta-heuristics approaches. Depending on how to set the parameter used, the performance is greatly affected. Due to the above disadvantages, it can be dropped into the local optimum. To address this problem, recently, hybrid meta-heuristics approaches that combine two or more meta-heuristics approaches have been developed and applied to complex network problems including the CLSC model. Many conventional studies have demonstrated that applying a hybrid meta-heuristic approach is more effective than applying a single meta-heuristic approach to complex network problems (Lin et al., 2009; Zhang et al., 2012; Xinyu and Liang, 2016). Soleimani et al. (2015) proposed an HGA approach that combines PSO and GA approaches as a way to solve large-scale CLSC models, demonstrating that the HGA approach outperforms the general GA approaches. A study by Li, et al. (2016) proposed an HGA approach that mixes TS and GA approaches for efficient work scheduling problems. Therefore, in this study, a hybrid meta-heuristic approach for optimizing the CLSC-DR model is proposed.

5.2.2 VNS Approach

Mladenovic and Hansen (1997) proposed the VNS approach. The concept of VNS approach is that it continuously seeks out a better solution by exploring a set of pre-defined areas. It can either randomly or systematically explore these neighborhoods. The first step in creating set for neighborhood structures is to define a set composed of related sets of neighborhoods. From there, each iteration of the algorithm will perform three steps: movement, shaking, and local search. The initial solution of each step is generated randomly. During the shaking step, a random neighbor solution is generated. The local search step is applied to the neighbor`s neighbor solution. If the neighbor's neighbor solution is better than the initial solution, the neighbor's neighbor solution becomes the current solution, and the search continues from the current solution. If the neighbor's neighbor solution is not better than the initial solution, we will move to the next neighbor to create a new solution for this neighbor and try to improve it. The general procedure for implementing the VNS approach is as shown in Figure 5.3.

```
procedure: VNS-approach
input: a set of neighborhood structures N_l, l = 1,2,..., l_max
begin
    S= generate initial solution ();
    repeat
    l = 1 ;
        while (l ≤ l_max)
            S'= Shaking (S, N_l)
            S'*= Local search (S')
            if f(S'*) < f(S)
              S ← S'*
              l=1 ;
            else
              l=l+1 ;
        end
    until stopping condition are met ;
output : The best solution ;
end
```

Fig 5.3 Pseudo code of the VNS approach (Hosseinabadi et al., 2016)

The VNS approach possesses the capability to investigate and exploit diverse regions of the search space across multiple neighborhoods. Angelo et al. (2015) designed a general variable neighborhood search (GVNS) approach as a meta-heuristic approach for solving the multi-product dynamic lot sizing problem in a CLSC model. They demonstrated that the GVNS approach can successfully solve large problems. However, as discussed earlier, it can be seen that optimization problem-solving approach using single meta-heuristic approach has disadvantages. This proves that using hybrid meta-heuristic approaches is more efficient than using single meta-heuristic approach to overcome this weakness.

Zhai et al. (2016) designed hybrid heuristic algorithms by integrating GA, VNS, and fuzzy simulations (FS) to solve the hub position problem. Along with convergence analysis, the calculation results showed that VNS-based GA approach achieves better performance than standard GA approach. Devika et al. (2014) developed new hybrid meta-heuristic approach based on adaptive imperialist competitive algorithms and VNS to solve the CLSC models. In evaluating the effectiveness and robustness of these algorithms, they were compared the proposed approach with conventional algorithms. The outcomes revealed that the suggested approach outperforms other methods, yielding superior solutions.

5.2.3 TLBO approach

The TLBO approach was first developed by Rao et al. (2011). It uses a population-based approach to model a classroom environment and perform optimization on a given objective. It has two phases: the learner phase and the teacher phase. The latter involves the teacher interacting with the students. At this phase, teachers are committed to providing knowledge to learners and improving average student outcomes. The learner phase simulates the learning process of a student through interaction. Interacting and discussing with other students can help learners gain knowledge. A student will learn new information, if the other student has more knowledge than him (or her). The general procedure for implementing the TLBO approach is as shown in Figure 5.4.

The TLBO approach does not require any algorithm-specific parameters. That is the advantage of using the TLBO approach in many fields of research. In a study by Rajesh (2020), the TLBO approach was used to effectively solve the supply chain model. However, most studies (Babazadeh et al. 2017; Rabeh et al. 2019) using the TLBO approach have been more effective when used in conjunction with the GA approach than when using the TLBO approach alone.

GA approach can maintain group diversity due to its randomness in the process of generating population and calculating evolution, but it can include inferior individuals within the generated population. TLBO can be applied to these individuals to improve their fitness value, thereby maintaining the best solution (Rabeh et al. 2019). Babazadeh et al. (2017) suggested a capacitated three-stage SC network using the GA-TLBO approach. The experimental outcomes demonstrated that the hybrid approach combining GA and TLBO outperforms the alternative approaches.

```
procedure: TLBO-approach
input: control parameters: max_Gen, pop_size
begin
randomly generate initial solution P(t);
calculate fitness value in P(t);
select Teacher value (X_best) and calculate the mean of the class (X_mean) in P(t);
  while ( t > max_Gen; not stop condition) do
      for i = 1 to pop_size
        T_f = round(1 + rand);
        X_new = X_i + rand*(X_best - T_f X_mean);
        if f(X_new) < f(X_i) then
            update X_i ← X_new ;
        end
        randomly select the other solution (X_p) in all solutions;
        if f(X_i) < f(X_p) then
          X_new = X_i + rand(X_i - X_p) ;
        else
          X_new = X_i - rand(X_i - X_p);
        end
        update X_i ← X_new
        select X_best and calculate X_mean;
      next
      t ← t + 1
  end
output: X_best
end
```

Fig. 5.4 Pseudo code of the TLBO approach (Rao et al., 2011)

5.3 Implementation of the Proposed GA-VNS-TLBO Approach

In this study, we propose a GA-VNS-TLBO approach as a combination strategy using the GA, VNS, and TLBO approach to optimize the CLSC-DR model. The procedure for applying the GA-VNS-TLBO approach is as follows.

First, the initial population is randomly generated. Secondly, for the GA loop, 50% sub-populations with superior fitness values are used. On the other hand, for the VNS loop, 50% sub-populations with inferior fitness are used. The TLBO loop utilizes 50% of the sub-populations with superior fitness values from the population generated through the VNS and GA loops. After applying GA, VNS, and TLBO loop, a new population is produced, and then the elitist selection scheme (Gen et al. 1997) is applied to this new population and parent one to produce new parent population for next iterations. This procedure is repeated until a pre-defined maximum number of iterations is reached.

Detailed implementation procedure of the GA-VNS-TLBO algorithm is showed in Figures 5.5 and 5.6.

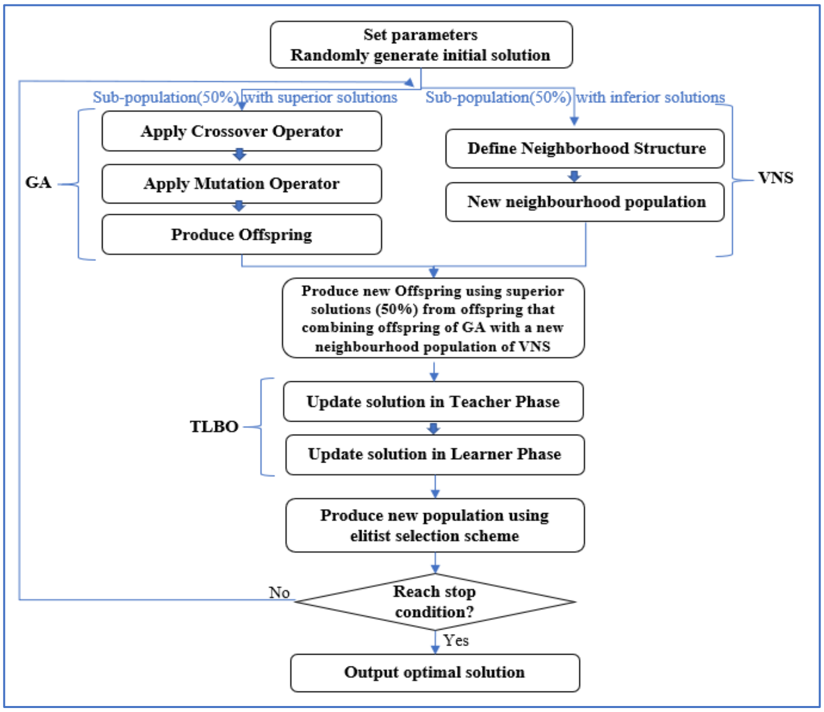

Fig 5.5 A conceptual flow chart of the GA-VNS-TLBO approach

```
procedure: GA-VNS-TLBO approach
input: GA parameters, VNS parameters, TLBO parameters
begin:
  t ← 1;
  randomly generate initial population I(t);
  evaluate I(t) and keep the best solution I_best in I(t);
  while (not terminating condition) do
    create P(t) using superior solutions (50%) from I(t);
    create V(t) using inferior solutions (50%) from I(t);
    create O(t) from P(t) by crossover routine and mutation routine; // GA loop
    evaluate O(t) and keep the best solution Best_GA in O(t);
    running VNS procedure using V(t); // VNS loop
    create N(t) from V(t);
    take best solution Best_VNS;
    create offspring using O(t) and N(t);
    create T(t) using superior solutions (50%) from offspring;
    running TLBO procedure using T(t); // TLBO loop
    take best solution Best_X from T(t);
    Best_G = argmin{I_best, Best_GA, Best_VNS, Best_X}
    reproduce I(t+1) from O(t), N(t), and T(t) by elitist selection scheme;
    t ← t+1;
  end while
output: Best_G
end
```

Fig 5.6 Pseudo code of the GA-VNS-TLBO approach

The procedure for implementing the VNS approach and TLBO approach used for GA-VNS-TLBO approach is as shown in Figures 5.7 and 5.8.

procedure: VNS-approach
input: *V(t)* population
begin
Evaluate fitness *V(t);*
while (**not** stop condition)
$k \leftarrow 1$
while (**not** stop condition)
generate random neighbor *E(k)* from *the* k^{th} neighborhood *N(k)* of *V(t)*;
take best solution *Best_VNS from E(k);*
if *f*(*Best_VNS*) < *f*(*N*(*k*)) **then**
update *N(k)* ← *Best_VNS*
produce new population *N(t)* from *V(t)* and *N(k)*;
$k \leftarrow 1$
else
$k \leftarrow k + 1$
end
end
$t \leftarrow t+1$
end
output: *Best_VNS*
end

Fig 5.7 Pseudo code of the VNS approach

procedure: TLBO-approach
input: *T(t)* population, *pop_size*: size of *T(t)* population
begin
evaluate fitness *T(t)*;
select Teacher value (X_{best}) and calculate the mean of the class (X_{mean}) in *T(t)*;
while (**not** stop condition)
for $i = 1$ **to** *pop_size*
T_f = round(1 + rand);
$X_{new} = X_i$ + rand*(X_{best} - $T_f X_{mean}$);
if $f(X_{new}) < f(X_i)$ **then**
update $X_i \leftarrow X_{new}$
end
randomly select the other solution (X_p) in all solutions;
if $f(X_i) < f(X_p)$ **then**
$X_{new} = X_i$ + rand(X_i - X_p) ;
else
$X_{new} = X_i$ - rand(X_i - X_p);
end
update $X_i \leftarrow X_{new}$
select *Best_X* and calculate X_{mean};
next
$t \leftarrow t+1$
end
output: *Best_X*
end

Fig. 5.8 Pseudo code of the TLBO approach

6. Numerical experiments

In numerical experiments, the mathematical formulation of the CLSC-DR model suggested in Section 3 is implemented using five scales as shown in Table 6.1. For example, for the CLSC-DR model, each supplier group has one main supplier and four backup suppliers, and three module manufacturers, manufacturers, distribution centers, retailers, collection centers, and recovery centers are considered for Scale 1, of which only one facility is opened at each stage and not all remaining facilities. And only one facility is considered and opened for a customer, a second customer, and a disposal center, respectively. Data on transportation cost per unit, fixed cost, and handling cost were randomly generated through Excel. (Saffari et al., 2015; Talaei et al., 2016) as shown in Table 6.2.

Table 6.1 Five Scales for CLSC-DR Model

Scales	No.of Supplier Group	No.of Main supplier	No.of Backup supplier	No.of Main route	No.of Backup route	No.of Module manufacturer	No.of Manufacturer	No.of Distribution center	No.of Retailer	No.of Collection center	No.of Recovery center	No.of Customer	No.of Second customer	No.of Disposal center
1	3	1	4	1	4	3	3	3	3	3	3	1	1	1
2	4	1	9	1	9	6	6	6	6	6	6	1	1	1
3	5	1	14	1	14	9	9	9	9	9	9	1	1	1
4	6	1	19	1	19	12	12	12	12	12	12	1	1	1
5	7	1	24	1	24	15	15	15	15	15	15	1	1	1

Table 6.2 Setting for Parameter Values

Parameters	Values	Parameters	Values
F_{gs}	$U[1000,1500]$	T_{gsot}	$U[2, 4]$
$F_{gs'}$	$U[1500,2000]$	$T_{gs'ot'}$	$U[3, 5]$
F_o	$U[2200,2500]$	T_{gsmt}	$U[3, 5]$
F_m	$U[2000,2500]$	$T_{gs'mt'}$	$U[5, 7]$
F_d	$U[1800,2000]$	T_{om}	$U[3, 5]$
F_r	$U[1300,1500]$	T_{md}	$U[6, 8]$
F_l	$U[1500,1900]$	T_{dr}	$U[4, 6]$
F_e	$U[1800,2100]$	T_{dc}	$U[2, 4]$
H_{gs}	$U[15, 20]$	T_{rc}	$U[1, 3]$
$H_{gs'}$	$U[20, 25]$	T_{cl}	$U[1, 3]$
H_o	$U[18, 20]$	T_{lw}	$U[1, 3]$
H_m	$U[25, 30]$	T_{le}	$U[4, 6]$
H_d	$U[20, 25]$	T_{eo}	$U[5, 7]$
H_r	$U[28, 35]$	T_{em}	$U[4, 6]$
H_l	$U[25, 30]$	T_{eu}	$U[1, 3]$
H_e	$U[25, 30]$		

6.1 Comparison of Proposed GA-VNS-TLBO Approach and Conventional Approaches

The five conventional meta-heuristics approaches for comparing the GA-VNS-TLBO approach comparison are shown in Table 6.3.

Table 6.3 Approaches used for Comparison

Approach	Description
GA	Single meta-heuristic approach by (Gen & Cheng, 2000)
VNS	Single meta-heuristic approach by (Mladenović & Hansen, 1997)
TLBO	Single meta-heuristic approach by (Rao, 2011)
GA-VNS	Hybrid meta-heuristic approach by (Dib et al., 2015)
GA-TLBO	Hybrid meta-heuristic approach by (Gucyetmez et al., 2016)
GA-VNS-TLBO	Proposed Hybrid meta-heuristic approach in this study

In Table 6.3, the parameter settings for the GA, VNS, TLBO, GA-VNS, GA-TLBO and GA-VNS-TLBO approaches are as follows: A total number of generations is 100, population size is 20, crossover rate is 0.5, and mutation rate is 0.3. These parameter values were obtained after the fine-tuning procedure of each approach. The 10 independent trials were used to eliminate the randomness of each approach. The computer environment in which the numerical experiment was run is an IBM-compatible PC 1.1 GHz Processor (Intel Celeron N4020 CPU), and 4GB RAM, which was programmed using MATLAB R2022a.

As measures of performance, the best solution (BS), average solution (AS), and average CPU time (CPU) were used to compare the performance of each approach as shown in Table 6.4. The computation results of each approach are shown in Tables 6.5 to 6.9, when a part supplier group among all ones is randomly selected, and then either the main supplier or its main route of the selected part supplier group is disrupted for each scale shown in Table 6.1.

Table 6.5 summarizes the results of the Scale 1 and shows the significant differences between all approaches. For instance, the differences among the GA, VNS, TLBO, GA-VNS, and GA-TLBO approaches are 4.24%, 4.15%, 3.29%, 3.57%, and 1.78% respectively in terms of the BS, when compared with the GA-VNS-TLBO approach, which means that the GA-VNS-TLBO approach has significantly better performance than the GA, VNS, TLBO, GA-VNS, and GA-TLBO approaches. A similar result is also evident in terms of the AS, with differences observed among the GA, VNS, TLBO, GA-VNS, and GA-TLBO approaches of 3.37%, 3.00%, 1.88%, 2.53%, and 0.97%, respectively, compared with the GA-VNS-TLBO approach. Notably, the GA-VNS-TLBO approach exhibits slightly superior performance compared with the GA-TLBO approach, while significantly outperforming the other approaches. However, it should be noted that the GA-VNS-TLBO approach was the slowest in terms of CPU time, while the GA approach was the fastest.

Table 6.4 Measures for Comparing the Performances of Each Approach

Measure	Description
BS	Best solution in all trials
AS	Values averaged over all trials
CPU	CPU time averaged over all trials

Table 6.5 Experimental Result using Scale 1

Approach	BS	AS	CPU	Gap 1(%)	Gap 2(%)
GA	318102	325190.3	**<u>11.1</u>**	4.24%	3.37%
VNS	317826	324008.9	11.2	4.15%	3.00%
TLBO	315179	320494.4	11.5	3.29%	1.88%
GA-VNS	316054	322541.9	12.6	3.57%	2.53%
GA-TLBO	310584	317643.2	12.4	1.78%	0.97%
GA-VNS-TLBO	**<u>305149</u>**	**<u>314584.5</u>**	12.8		

*** The best value at each performance are bold and underlined**
*** Gap 1(%): Difference when compared the performances of GA, VNS, TLBO, GA-VNS and GA-TLBO with that of GA-VNS-TLBO in terms of the BS**
*** Gap 2(%): Difference when compared the performances of GA, VNS, TLBO, GA-VNS and GA- TLBO with that of GA-VNS-TLBO in terms of the AS**

According to the results presented in Table 6.6 using Scale 2, the variation in terms of the BS among the GA, VNS, TLBO, GA-VNS, and GA-TLBO approaches is 3.10%, 4.23%, 2.95%, 4.86%, and 0.56%, respectively, when compared with the GA-VNS-TLBO approach. Notably, the GA-VNS-TLBO approach demonstrates significantly superior performance compared with the GA, VNS, TLBO, and GA-VNS approaches. Furthermore, it slightly outperforms the GA-TLBO approach. Similarly, the computation result in terms of the AS reveals that the performance, indicated by the differences, among the GA, VNS, TLBO, GA-VNS, and GA-TLBO approaches is 3.37%, 4.36%, 2.70%, 4.11%, and 0.17%, respectively, in comparison to the GA-VNS-TLBO approach. The results consistently demonstrate that the GA-VNS-TLBO approach generally outperforms the other approaches. In terms of CPU time, the GA and GA-TLBO approaches achieve the fastest execution, while the GA-VNS-TLBO approach exhibits slightly slower performance.

Table 6.6 Experimental result using Scale 2

Approach	BS	AS	CPU	Gap 1(%)	Gap 2(%)
GA	332940	341219.3	**15.5**	3.10%	3.37%
VNS	336588	344504.1	16.2	4.23%	4.36%
TLBO	332461	339018.9	19.3	2.95%	2.70%
GA-VNS	338630	343660.2	18.8	4.86%	4.11%
GA-TLBO	324751	330659.9	**15.5**	0.56%	0.17%
GA-VNS-TLBO	**322935**	**330096.2**	15.6		

*** The best value at each performance are bold and underlined**

*** Gap 1(%): Difference when compared the performances of GA, VNS, TLBO, GA-VNS and GA-TLBO with that of GA-VNS-TLBO in terms of the BS**

*** Gap 2(%): Difference when compared the performances of GA, VNS, TLBO, GA-VNS and GA- TLBO with that of GA-VNS-TLBO in terms of the AS**

Table 6.7 Experimental result using Scale 3

Approach	BS	AS	CPU	Gap 1(%)	Gap 2(%)
GA	352743	359175.8	15.8	5.33%	4.45%
VNS	359800	364594.5	19.2	7.43%	6.02%
TLBO	351554	357223.8	18.2	4.97%	3.88%
GA-VNS	354792	361354.3	20.2	5.94%	5.08%
GA-TLBO	339794	345321.7	**15.1**	1.46%	0.42%
GA-VNS-TLBO	**334906**	**343882.7**	15.6		

*** The best value at each performance are bold and underlined**

*** Gap 1(%): Difference when compared the performances of GA, VNS, TLBO, GA-VNS and GA-TLBO with that of GA-VNS-TLBO in terms of the BS**

*** Gap 2(%): Difference when compared the performances of GA, VNS, TLBO, GA-VNS and GA- TLBO with that of GA-VNS-TLBO in terms of the AS**

According to the results presented in Table 6.7 using Scale 3, the differences in terms of the BS among the GA, VNS, TLBO, GA-VNS, and GA-TLBO approaches are 5.33%, 7.43%, 4.97%, 5.94%, and 1.46%, respectively, when compared with the GA-VNS-TLBO approach. It is evident that the GA-VNS-TLBO approach demonstrates better performance in terms of the BS compared with the other approaches.

The computation result in terms of the AS also reveals the differences among the approaches. Specifically, the differences among the GA, VNS, TLBO, GA-VNS, and GA-TLBO approaches are 4.45%, 6.02%, 3.88%, 5.08%, and 0.42%, respectively, when compared with the GA-VNS-TLBO approach. In comparison with the other approaches, the GA-VNS-TLBO approach exhibits slightly better performance than the GA-TLBO approach, while significantly outperforming the other approaches. However, it should be noted that the GA-TLBO approach achieves the fastest CPU time, surpassing the GA-VNS-TLBO approach in terms of speed.

Table 6.8 Experimental result using Scale 4

Approach	BS	AS	CPU	Gap 1(%)	Gap 2(%)
GA	374152	380951.1	16.2	5.08%	4.75%
VNS	383055	388119.3	23.6	7.58%	6.72%
TLBO	366358	376110.7	19.4	2.89%	3.42%
GA-VNS	372806	380167.3	19.9	4.70%	4.53%
GA-TLBO	358521	364284.5	**<u>16.0</u>**	0.69%	0.16%
GA-VNS-TLBO	**<u>356066</u>**	**<u>363686.2</u>**	16.3		

*** The best value at each performance are bold and underlined**

*** Gap 1(%): Difference when compared the performances of GA, VNS, TLBO, GA-VNS and GA-TLBO with that of GA-VNS-TLBO in terms of the BS**

*** Gap 2(%): Difference when compared the performances of GA, VNS, TLBO, GA-VNS and GA- TLBO with that of GA-VNS-TLBO in terms of the AS**

As shown in Table 6.8, the differences in the GA, VNS, TLBO, GA-VNS, GA-TLBO approaches compared with the GA-VNS-TLBO approach are 5.08%, 7.58%, 2.89%, 4.70%, and 0.69%, respectively. In terms of the AS, the differences between the five different approaches compared with the GA-VNS-TLBO approach are as follows: 4.75% for the GA approach, 6.72% for the VNS approach, 3.42% for the TLBO approach, 4.53% for the VNS approach, and 0.16% for the GA-TLBO approach, where the GA-VNS-TLBO approach shows to be slightly better performance than the GA-TLBO approach, while it shows considerably better performance than the others. The GA-TLBO approach exhibits the quickest CPU time, whereas the slowest CPU time is observed in the GA-VNS-TLBO approach.

Table 6.9 Experimental result using Scale 5

Approach	BS	AS	CPU	Gap 1(%)	Gap 2(%)
GA	398508	402193.4	**<u>17.0</u>**	6.78%	6.09%
VNS	404080	409351.9	15.3	8.27%	7.98%
TLBO	391685	396819.1	20.5	4.95%	4.68%
GA-VNS	393456	402595.9	**<u>15.0</u>**	5.43%	6.20%
GA-TLBO	376683	382436.1	16.1	0.93%	0.88%
GA-VNS-TLBO	**<u>373201</u>**	**<u>379090.5</u>**	19.2		

*** The best value at each performance are bold and underlined**

*** Gap 1(%): Difference when compared the performances of GA, VNS, TLBO, GA-VNS and GA-TLBO with that of GA-VNS-TLBO in terms of the BS**

*** Gap 2(%): Difference when compared the performances of GA, VNS, TLBO, GA-VNS and GA- TLBO with that of GA-VNS-TLBO in terms of the AS**

The results obtained using Scale 5 demonstrate that the differences in terms of the BS among the GA, VNS, TLBO, GA-VNS, and GA-TLBO approaches are 6.78%, 8.27%, 4.95%, 5.43%, and 0.93%, respectively, when compared with the GA-VNS-TLBO approach. Especially, in the difference between the GA-TLBO and GA-VNS-TLBO approaches, the former exhibits slightly better performance than the latter, while significantly outperforming the other approaches.

Regarding in terms of the AS, the differences among the GA, VNS, TLBO, GA-VNS, and GA-TLBO approaches are 6.09%, 7.98%, 4.68%, 6.20%, and 0.88%, respectively, when compared with the GA-VNS-TLBO approach. The performance of the GA-VNS-TLBO approach surpasses that of the GA, VNS, TLBO, GA-VNS, and GA-TLBO approaches significantly. Additionally, the GA-VNS-TLBO approach performs slightly better than the GA-TLBO approach. In terms of CPU time, the GA-VNS approach is faster than the GA-VNS-TLBO approach.

The convergence behaviors of various approaches are shown in Figures 6.1 to 6.5. They show various changes in the behaviors of each approach as the number of iterations approaches to about 100. In Figure 6.1, it shows that the GA-VNS-TLBO approach is more effective than the other approaches in the initial search processes, while the other approaches show different results, but overall performances are lower than the GA-VNS-TLBO approach at all stages.

The GA-VNS-TLBO approach can quickly improve optimization in the early stages, while other approaches (GA, VNS, TLBO, GA-VNS, and GA-TLBO) show different convergence behaviors in the early stages, but are less powerful than the GA-VNS-TLBO approach in the later stages, as shown in Figure 6.2. In Fig. 6.3, the various competing approaches show their convergence behaviors in the early stages. The GA-VNS-TLBO approach is more likely to exhibit rapid convergence than the other approaches. The convergence behavior exhibited in Fig. 6.4 is similar to those of other approaches, except that the VNS approach has the faster convergence rate. On the other hand, the GA-VNS-TLBO approach has demonstrated good results during the later stages.

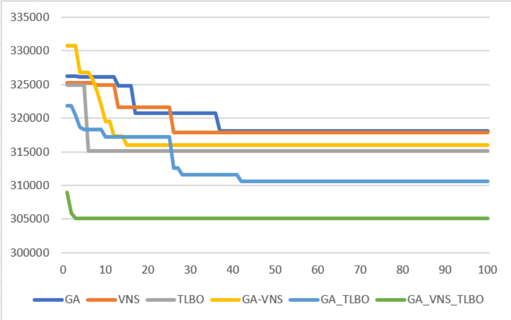

Fig 6.1 Convergence behaviors of each approach for scale 1

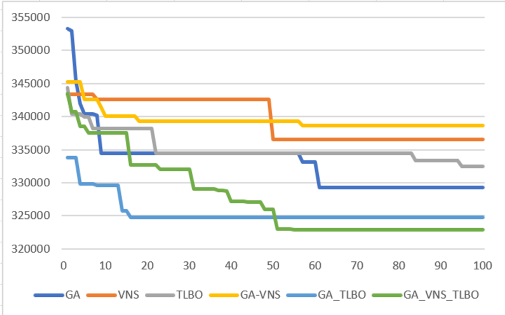

Fig 6.2 Convergence behaviors of each approach for scale 2

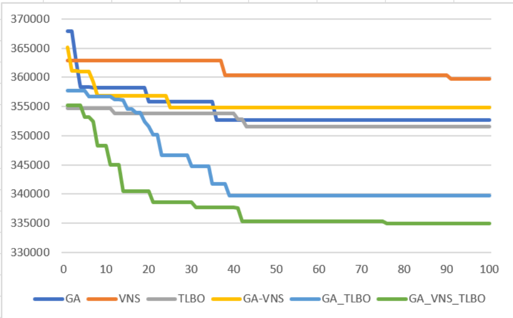

Fig 6.3 Convergence behaviors of each approach for scale 3

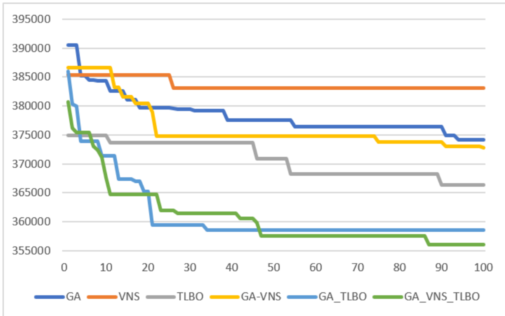

Fig 6.4 Convergence behaviors of each approach for scale 4

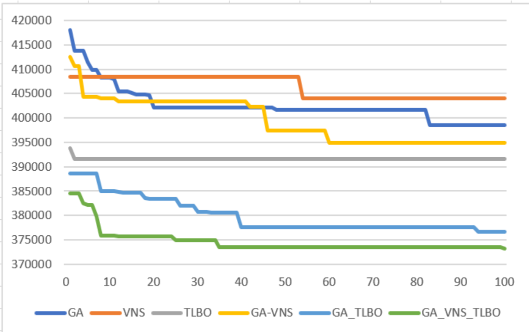

Fig 6.5 Convergence behaviors of each approach for scale 5

The GA-VNS-TLBO approach in Fig 6.5 can improve optimization rapidly in the early stages, whereas the other approaches (GA, GA-VNS, and GA-TLBO), except for the VNS and TLBO approaches, show different convergence behaviors at the early stages, but are inferior to the GA-VNS-TLBO approach over the entire optimization process.

The distributions of the best solution (i.e., the BS) obtained after 10 independent runs for each scale are shown in Figure 6.6 to 6.10.

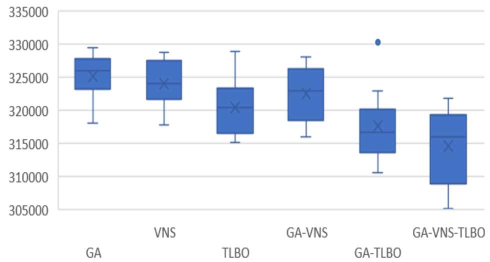

Fig 6.6 Distributions of the best solutions at each approach for Scale 1

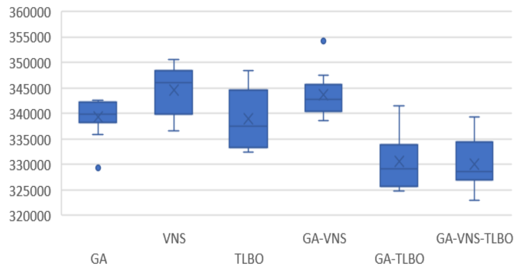

Fig 6.7 Distributions of the best solutions at each approach for Scale 2

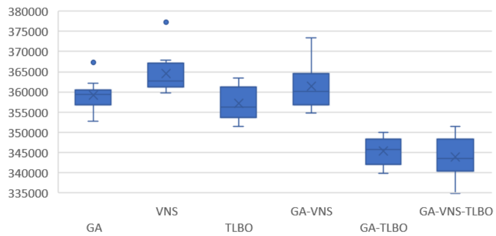

Fig 6.8 Distributions of the best solutions at each approach for Scale 3

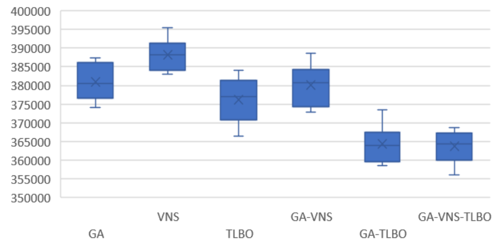

Fig 6.9 Distributions of the best solutions at each approach for Scale 4

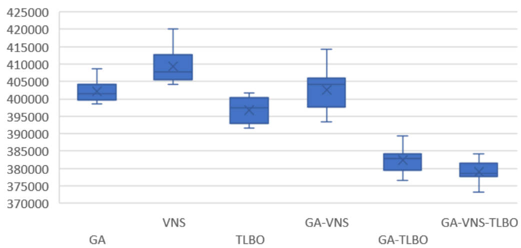

Fig 6.10 Distributions of the best solutions at each approach for Scale 5

Figures 6.6 to 6.10 show that the GA-VNS-TLBO approach is a more compact distribution with a lower average value compared with other approaches such as the GA, VNS, TLBO, GA-VNS, and GA-TLBO. It also has better optimization potential.

The results of the Tables 6.5 to 6.9 and Figures 6.1 to 6.10 provide evidence that the GA-VNS-TLBO approach exhibits superior performance compared with single meta-heuristics approaches (GA, VNS, and TLBO) as well as hybrid meta-heuristics approaches (GA-VNS and GA-TLBO) in terms of the BS and AS. These results highlight the significance of effectively combining single meta-heuristic approaches to achieve successful hybrid meta-heuristic approaches, such as the GA-VNS-TLBO approach. Through the computation results of Tables 6.5 to 6.9 and Figures 6.1 to 6.10, the following conclusions can be reached.

- Among all meta-heuristic approaches using Scales 1 to 5, the GA-VNS-TLBO approach has demonstrated superior performance in terms of the AS and BS compared with the conventional GA, VNS, and TLBO approaches. This shows that the proposed hybrid meta-heuristic approach is more efficient.
- When considering the comparison among all algorithms, it becomes evident that the GA-VNS-TLBO approach outperforms not only the single meta-heuristic approaches (GA, VNS, and TLBO) but also the hybrid meta-heuristic approaches (GA-VNS and GA-TLBO) in terms of the BS and AS. The superiority of the GA-VNS-TLBO approach is clearly observed in the comparison with all other algorithms. The results of the analysis have revealed that the performance of hybrid meta-heuristic approach such as the GA-VNS and GA-TLBO depends on the optimal combination of single meta-heuristic algorithms.

6.2 Sensitivity Analysis of the GA-VNS-TLBO Approach

The GA-VNS-TLBO approach employs the sub-population consisting of the best individuals (50%) for the GA loop, while the remaining sub-population containing the worst individuals is used for the VNS loop. Subsequently, the sub-population (50%) comprising the best individuals from the offspring generated by the GA and VNS loops is utilized in the TLBO loop. Most conventional studies use a whole population for hybrid meta-heuristic approaches. Thus, to compare the results of the studies using whole populations and those using some parts of the whole population in hybrid meta-heuristic approaches, two approaches are used to compare the GA-VNS-TLBO approach.

The first approach, GA-VNS-TLBO1, involves utilizing the entire population (100%) that is randomly generated in the initial stages for the GA and VNS loops. Furthermore, the complete offspring (100%) obtained after the GA and VNS loops are employed in the TLBO loop. The second approach, GA-VNS-TLBO2, adopts a different strategy. It selects a sub-population (50%) consisting of the best individuals for the GA loops, while the remaining sub-population (50%) composed of the worst individuals is utilized in the VNS loop. However, the entire offspring (100%) obtained after the GA and VNS loops are employed in the TLBO loops. Table 6.10 showcases the performance of the GA-VNS-TLBO1, GA-VNS-TLBO2, and GA-VNS-TLBO approaches, comparing their respective outcomes.

Table 6.10 Performance Comparison among the GA-VNS-TLBO1, GA-VNS-TLBO2, and GA-VNS-TLBO approaches

	GA-VNS-TLBO1					**GA-VNS-TLBO2**					**GA-VNS-TLBO**		
Scale	BS	AS	CPU time	Gap 1 (%)	Gap 2 (%)	BS	AS	CPU time	Gap 1 (%)	Gap 2 (%)	BS	AS	CPU time
1	311959	318195	20.4	2.23%	1.15%	307600	317639.3	18.0	0.80%	0.97%	**305149**	**314584.5**	**12.8**
2	326338	329725	21	1.05%	-0.11%	324041	**327625.6**	18.5	0.34%	-0.75%	**322935**	330096.2	**15.6**
3	337568	346446	23.2	0.79%	0.75%	338654	346322	18.9	1.12%	0.71%	**334906**	**343882.7**	**15.6**
4	357590	362790	20.7	0.43%	-0.25%	356085	**362074.3**	18.5	0.01%	-0.44%	**356066**	363686.2	**16.3**
5	376098	382283	22.1	0.78%	0.84%	376787	381728.8	19.8	0.96%	0.70%	**373201**	**379090.5**	**19.2**

* The best value at each performance are bold and underlined

* Gap 1(%): Difference when compared the performances of GA-VNS1 and GA-VNS2 with that of GA-VNS-TLBO in terms of the BS

* Gap 2(%): Difference when compared the performances of GA-VNS1 and GA-VNS2 with that of GA-VNS-TLBO in terms of the AS

Table 6.10 highlights the distinction between the GA-VNS-TLBO and GA-VNS-TLBO1 approaches. Especially, the GA-VNS-TLBO approach demonstrates superior performance in terms of the BS and AS compared to GA-VNS-TLBO1. However, it is important to note that GA-VNS-TLBO1 exhibits a slower search speed in terms of CPU time. A similar situation can be observed when comparing the GA-VNS-TLBO and GA-VNS-TLBO2 approaches. Once again, the former exhibits more efficient performance than the latter in terms of the BS, AS, and CPU time.

In summary, when integrating the GA, VNS, and TLBO loops in the GA-VNS-TLBO approach, utilizing the sub-population (50%) generated in the initial stages for the GA and VNS loops, and subsequently employing the sub-population (50%) obtained from the GA and VNS loops for the

TLBO loop proves to be more effective in reducing the search speed compared with using the entire population (100%) for all the loops or the entire offspring (100%) obtained after the GA and VNS loops are employed in the TLBO loops.

Figs 6.11 to 6.15 show the convergence behaviors of three different approaches (GA-VNS-TLBO, GA-VNS-TLBO1, and GA-VNS-TLBO2) when the number of iterations is reached to 100.

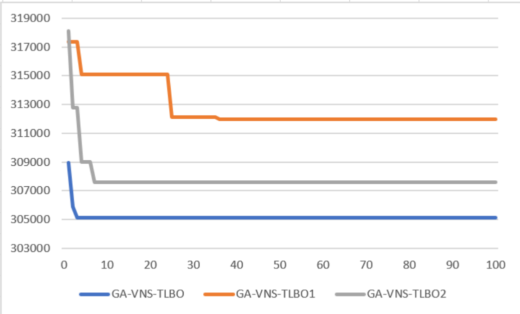

Fig 6.11 Convergence behaviors of each approach for Scale 1

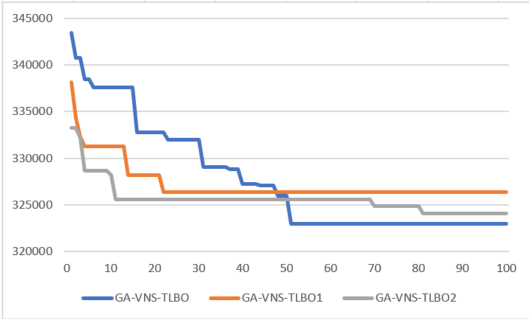

Fig 6.12 Convergence behaviors of each approach for Scale 2

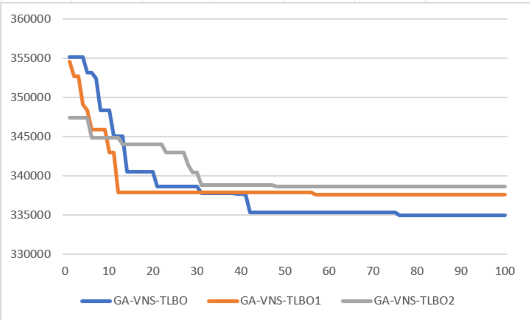

Fig 6.13 Convergence behaviors of each approach for Scale 3

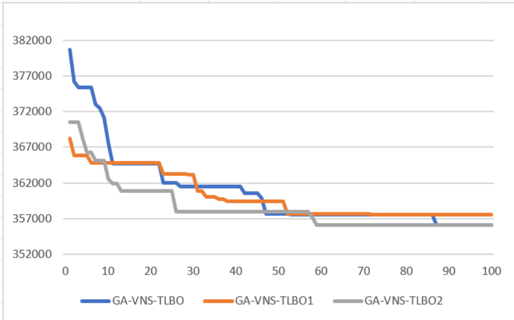

Fig 6.14 Convergence behaviors of each approach for Scale 4

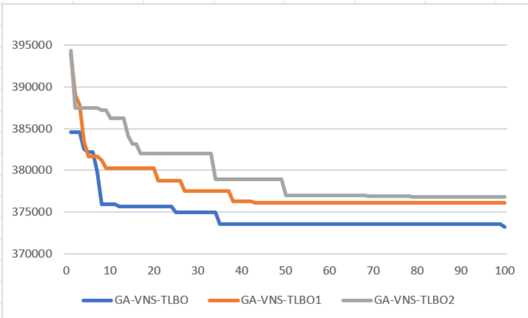

Fig 6.15 Convergence behaviors of each approach for Scale 5

In Figure 6.11, the GA-VNS-TLBO approach shows rapid convergence behaviors in the initial stages, while the other approaches (GA-VNS-TLBO1 and GA-VNS-TLBO2) show different convergence behavior in the initial stages, and overall performance is lower than that of the GA-VNS-TLBO approach in all stages.

On the other hand, in the later stages, the GA-VNS-TLBO approach quickly converges and its performance is better than the others.

Similar convergence behaviors are also shown in Fig 6.12 to 14, where all the competing approaches show various convergence behaviors in their early stages, while the GA-VNS-TLBO approach shows rapid convergence behaviors rather than the others in all stages.

In Fig 6.15, it can be shown that the GA-VNS-TLBO approach has rapid convergence behaviors in the early stages, whereas the other approaches (GA-VNS-TLBO1 and GA-VNS-TLBO2) have slow convergence behaviors at the early stages, and their performances are inferior to the GA-VNS-TLBO approach over the whole iterations.

7. Conclusion

In a rapid change of market environment, companies need to design a CLSC model effectively to survive and gain a competitive advantage. When an unforeseen situation occurs in the CLSC model, there exists a disruption risk in it. Therefore, the CLSC model is essential to effectively managing a variety of disruption risks. In this paper, a closed-loop supply chain with supplier disruption risk (CLSC-DR) model has been proposed. In the CLSC-DR model, considering supplier disruption and route disruption is a more realistic and effective approach because most conventional studies have focused on supplier disruptions and route disruptions in a simple SC model. For various distribution channels, normal delivery and direct delivery have been considered in the CLSC-DR model. The normal delivery is the general distribution channel for distributing products from a facility to the next. The direct delivers products from DC to customers without going through retailers.

The CLSC-DR model has been mathematically formulated and implemented using the GA-VNS-TLBO approach. The model aims to minimize the total cost, including transportation costs, fixed costs, and handling costs, at each stage and various constraints to be considered, such as restrictions on transportation quantity between stages and restrictions on centers (or facilities) to be opened, were used together. Objective function is employed to achieve this minimization goal in the model.

The GA-VNS-TLBO approach combines three single meta-heuristic approaches such as GA, VNS and TLBO, that is, the sub-population (50%) with superior fitness values and the sub-population (50%) with inferior fitness values of whole population obtained in initial stage, and the sub-offspring (50%) with superior fitness values in the whole offspring obtained after GA and VNS loops are used for GA, VNS, and TLBO loop, respectively. To demonstrate the superiority of the CLSC-DR model, the performance comparison has performed in following two ways:

First, the five scales of the CLSC-DR model have used to compare the performance of the GA–VNS-TLBO approach with those of the conventional approaches (GA, VNS, TLBO, GA-VNS, and GA-TLBO). The experimental results have shown that the GA-VNS-TLBO approach is more efficient in terms of both BS and AS when compared with the single meta-heuristic approaches (GA, VNS, and TLBO) and the hybrid meta-heuristic approaches (GA-VNS, and GA-TLBO). They have also revealed that the success of hybrid meta-heuristic algorithms depends on their combinations with various single meta-heuristic approaches.

Second, when implementing the GA-VNS-TLBO approach, using the sub-population (50%) with the best individuals for the GA loop, the other sub-population (50%) with the worst individuals for the VNS loop, and the sub-population (50%) with the best individuals in the offspring resulting from the GA and VNS loops is more efficient in locating best solution and average solution and in reducing searching time than using the whole population (100%) or the sub-population (50%) for the GA, VNS and TLBO loops.

In future research, enhancing the practical application of the methodology employed in this study by collecting and using more realistic data is needed. Despite the significantly improved performance in terms of best and average solutions compared to other competing approaches, there is a need to make an effort to decrease the search speed of the GA-VNS-TLBO approach.

References

Aghamohamadi, S., Manavizadeh, N., Rabbani, M. (2020). A hybrid metaheuristic algorithm for a data driven leagile sustainable closed-loop supply chain model under disruption risks, *Scientia Iranica*, 29(3):1685-1704.

An, S., Cui, N., Bai, Y., Xie, W., Chen, M. and Ouyang, Y. (2015). Reliable emergency service facility location under facility disruption, en-route congestion and in-facility queuing, *Transportation Research Part E,* http://dx.doi.org/10.1016/j.tre.2015.07.006

Angelo, S., Ioannis, K. (2015). General variable neighborhood search for the multi-product dynamic lot sizing problem in closed-loop supply chain, *Electronic Notes in Discrete Mathematics*, 47: 69-76.

Azaron, A., Venkatadri, U. and Doost, A. F. (2021). Designing profitable and responsive supply chains under uncertainty, *International Journal of Production Research*, 59(1):213-225.

Babazadeh, R., and Tavakkoli-Moghaddam, R. (2017). A hybrid GA-TLBO Algorithm for Optimizing a Capacitated Three-Stage Supply Chain Network, *International Journal of Industrial Engineering & Production Research*, 28: 151- 161.

Badejo, O. and Ierapetritou, M. (2023). A mathematical modeling approach for supply chain management under disruption and operational uncertainty, *AiChE Journal*. https://doi.org/10.22541/au.166372745.51424242/v1

Baghalian, A., Rezapour, S., and Farahani, R. Z. (2013). Robust supply chain network design with service level against disruptions and demand uncertainties: A real-life case, *European Journal of Operational Research*, 227:199-215.

Chen, Y. K., Weng, S. X., Liu, T. P. (2020). Teaching–learning based optimization (TLBO) with variable neighborhood search to retail shelf-space allocation, *Mathematics*, 8(8):1296. https://doi.org/10.3390/math8081296

Cheng, R. W., Gen, M. and Tsujimura, Y. (1996). A tutorial survey of job-shop scheduling problems using genetic algorithm – I. representation, *Computer & Industrial Engineering*, 30(4):983-997.

Chopra, S. and Sodhi, M. S. (2004). Managing risk to avoid supply chain breakdown, *MIT Sloan Management Review*, 46(1):53–62.

Chuluunsukh, A., Yun, Y. S. (2021). Supply chain network model considering supply disruption in assembly industry: Hybrid genetic algorithm approach, *Journal of the Korea Industrial Information Systems Research*, 26(3):9-22.

Chuluunsukh, A., Yun, Y., and Gen, M. (2021). Applying GA-VNS approach to supply chain network model with facility and route disruptions, *International Journal of Management Science and Engineering Management*, 17(3):151-165.

Dehghanian, F., and Mansour, S. (2009). Designing sustainable recovery network of end-of-life products using genetic algorithm, *Resources, Conservation and Recycling,* 53:559-570.

Devika, K., Jafarian, A., Nourbakhsh, V. (2014). Designing a sustainable closed-loop supply chain network based on triple bottom line approach: A comparison of metaheuristics hybridization techniques, *European Journal of Operational Research*, 235(3):594-615, https://doi.org/10.1016/j.ejor.2013.12.032.

Dib, O., Manier, M.A., Moalic, L., Caminada, A. (2015). Combining VNS with genetic algorithm to solve the one-to-one routing issue in road networks, *Computers & Operations Research*, https://doi.org/10.1016/j.cor.2015.11.010.

Gedik, R., Medal, H., Rainwater, C., Pohl, E. A., and Mason, S.J., (2014). Vulnerability assessment and re-routing of freight trains under disruptions: A coal supply chain network application, *Transportation Research Part E*, 71:45-57.

Gen, M. and Cheng, R. (1997). *Genetic Algorithms and Engineering Design*, John-Wiley & Sons, New York, NY, USA.

Gen, M. and Cheng, R. (2000). *Genetic Algorithms and Engineering Optimization*, John-Wiley & Sons, New York, NY, USA.

Gen, M., Lin, L., Yun, Y. S. and Inoue, H. (2018). Recent advances in hybrid priority-based genetic algorithms for logistics and SCM network design, *Computers and Industrial Engineering,* 115:394–412.

Goldberg, D., Lingle, R. (1985). Alleles, loci and the traveling salesman problem, *Proceedings of the 1st International Conference on Genetic Algorithms and Their Applications*, pp. 154-159.

Gucyetmez, M., Çam, E. (2016). A new hybrid algorithm with genetic-teaching learning optimization (G-TLBO) technique for optimizing of power flow in wind-thermal power systems. *Electrical Engineering*, https://doi.org/10.1007/s00202-015-0357-y.

Holland, J. H. (1975). *Adaptation in natural and artificial systems,* Ann Arbor, University of Michigan Press, AH, USA.

Hosseinabadi, AR., Alavipour, F., Shamshirbnd, Sh., Balas, VE. (2016). *A Novel Meta-Heuristic Combinatory Method for Solving Capacitated Vehicle Location-Routing Problem with Hard Time Windows*, Springer international, Switzerland.

Jabbarzadeh, A., Haughton, M. and Khosrojerdi, A. (2018). Closed-loop supply chain network design under disruption risks: A robust approach with real world application, *Computers and Industrial Engineering*, 116:178-191.

Kannan, G., Sasikumar, P., and Devika, K. (2010). A genetic algorithm approach for solving a closed loop supply chain model: A case of battery recycling, *Applied Mathematical Modelling*, 34:655-670.

Kleindorfer, P. R., Saad, G. H. (2005). Managing disruption risks in supply chains, *Production and Operations Management*, 14(1):53-68. http://dx.doi.org/10.1111/j.1937-5956.2005.tb00009.x.

Lin, L., Gen, M., and Wang, X. (2009). Integrated multistage logistics network design by using hybrid evolutionary algorithm, *Computers and Industrial Engineering*, (56):854–873.

Ma, R., Yao, L., Jin, M., Ren, P., Lv, Z. (2016). Robust environmental closed-loop supply chain design under uncertainty, *Chaos, Solitons & Fractals*, 89:195-202, https://doi.org/10.1016/j.chaos.2015.10.028.

Mladenovic, N. and Hansen, P. (1997). Variable neighborhood search, *Computers and Operations Research*, 24:1097-1100. https://doi.org/10.1016/S0305-0548(97)00031-2.

Neslihan, D., Eren, Ö., Turan, P., and Hadi, G. (2014). A genetic algorithm approach for optimising a closed-loop supply chain network with crisp and fuzzy objectives, *International Journal of Production Research*, 52(12):3637-3664.

Oke, A., Gopalakrishnan, M. (2009). Managing disruptions in supply chains: A case study of a retail supply chain. *International Journal of Production Economics*, 118:168-174. https://doi.org/10.1016/j.ijpe.2008.08.045.

Poudel, S. R., Marufuzzaman, M. and Bian, L. (2016). Designing a reliable bio-fuel supply chain network considering link failure probabilities, *Computers and Industrial Engineering*, 91:85-99.

Rabeh, R., Ferfra, M., Ezbakhe, A. (2019). Secondary frequency control of an islanded microgrid by combined GA-TLBO algorithm, *2019 8th International Conference on Systems and Control (ICSC)*, pp. 194-199.

Rajesh, R. (2020). Network design for resilience in supply chains using novel crazy elitist TLBO.

Neural Computing & Applications, 32:7421–7437. https://doi.org/10.1007/s00521-019-04260-3.

Ramshani, M., Ostrowski, J., Zhang, K., and Li, X. (2019). Two level uncapacitated facility location problem with disruptions, *Computers and Industrial Engineering*, https://doi.org/10.1016/j.cie.2019.106089

Rao, R.V., Savsani, V.J., Vakharia, D.P. (2011). Teaching–learning-based optimization: A novel approach for constrained mechanical design optimization problems, *Computer-Aided Design*, 43(3):303-315. https://doi.org/10.1016/j.cad.2010.12.015.

Saffari, H., Makui, A., Mahmoodian, V., and Pishvaee, M. S. (2015). Multi-objective robust optimization model for social responsible closed-loop supply chain solved by non-dominated sorting genetic algorithm, *Journal of Industrial and Systems Engineering*, 8(3):42-58.

Savaskan, R. C., Bhattacharya, S., and Van Wassenhove, L. V. (2004). Closed-loop supply chain models with product manufacturing, *Management Science*, 50:239–252.

Soleimani, H., Kannan, G. (2015). A hybrid particle swarm optimization and genetic algorithm for closed-loop supply chain network design in large-scale networks, *Applied Mathematical Modelling*, 39(14):3990-4012.

Subramanian, P., Ramkumar, N., Narendran, T.T., Ganesh, K. (2013). PRISM: Priority based Simulated annealing for a closed loop supply chain network design problem, *Applied Soft Computing*, 13(2):1121-1135, https://doi.org/10.1016/j.asoc.2012.10.004.

Talaei, M., Moghaddam, B. F., Pishvaee, M. S., Bozorgi-Amiri, A., and Gholamnejad, S. A. (2016). A robust fuzzy optimization model for carbon-efficient closed-loop supply chain network design problem: a numerical illustration in electronics industry, *Journal of Cleaner Production*, 113:662-673.

Tang, C. S. (2006). Perspectives in supply chain risk management, *International Journal of Production Economics*, 103:451–488.

Trkman, P., and McCormark, K. (2009). Supply chain risk in turbulent environments - A conceptual model for managing supply chain network risk, *International Journal of Production Economics*, 119:247-258.

Wang, X., Ruan, J., and Shi, Y. (2012). A recovery model for combinational disruptions in logistics delivery: Considering the real-world participators, *International Journal of Production Economics*, 140:508-520.

Waters, D. (2007). *Supply Chain Risk Management: Vulnerability and Resilience in Logistics*, Kogan Page limited, London, UK.

Wilson, M. C. (2007). The impact of transportation disruptions on supply chain performance, *Transportation Research Part E,* 43:295-320

Xiao, T. and Yu, G. (2006). Supply chain disruption management and evolutionarily stable strategies of retailers in the quantity-setting duopoly situation with homogeneous goods, *European Journal of Operational Research*, 173:648–668.

Xinyu, L., and Liang, G. (2016). An effective hybrid genetic algorithm and Tabu search for flexible job shop scheduling problem, *International Journal of Production Economics*, 174:93–110.

Yun, Y. S. (2006). Hybrid genetic algorithm with adaptive local search scheme, *Computers and Industrial Engineering*, 51:128–141.

Yun, Y. S., Chuluunsukh, A., and Chen, X. (2018). Hybrid genetic algorithm for optimizing closed-loop supply chain model with direct shipment and delivery, *New Physics: Sae Mulli*, 68(6):683-692.

Yun, Y. S., Chuluunsukh, A., and Gen, M. (2020). Sustainable closed-loop supply chain design problem: A hybrid genetic algorithm approach, *Mathematics*, 8(1):84.

Yun, Y. S., Chung, H. S., and Moon, C. U. (2013). Hybrid genetic algorithm approach for precedence-constrained sequencing problem, *Computers and Industrial Engineering*, 65:137–147.

Zhai, H., Liu, Y., Yang, K. (2016). Modeling two-stage UHL problem with uncertain demands, *Applied Mathematical Modelling*, 40(4):3029-3048, https://doi.org/10.1016/j.apm.2015.09.086.

Zhang, Q., Manier, H., and Manier, M. A. (2012). A genetic algorithm with Tabu search procedure for flexible job shop scheduling with transportation constraints and bounded processing times, *Computers and Operations Research*, 39:1713–1723.