Sādhanā (2022)47:254
https://doi.org/10.1007/s12046-022-02011-0

Priority fractional rationing (PFR) policy and a hybrid metaheuristic for managing stock in divergent supply chains

KURIAN JOHN[1,*], BRIJESH PAUL[1], CHANDRASEKHARAN RAJENDRAN[2] and HANS ZIEGLER[3]

[1]Department of Mechanical Engineering, Mar Athanasius College of Engineering, Kothamangalam 686666, India
[2]Department of Management Studies, Indian Institute of Technology Madras, Chennai 600036, India
[3]Department of Production, Operations and Logistics Management, Faculty of Business Administration and Economics, University of Passau, 94032 Passau, Germany
e-mail: kurianjpk2@gmail.com; brijeshpaul@gmail.com; craj@iitm.ac.in; hans.ziegler@uni-passau.de

MS received 12 April 2022; revised 29 August 2022; accepted 26 September 2022

Abstract. A distributor catering to demands of multiple retailers is considered in this paper and stock-management in this divergent supply chain is achieved through the deployment of periodic review base-stock (i.e. (*R*, *S*) policy) policy at every member. In the model of the supply chain considered in this study, in every time-period, an attempt is made by the distributor to first transport backlogged-demands from the downstream members till the distributor's previous instance-of-review even before considering demands from downstream members in recent time-periods. This practice of the distributor attempting first to satisfy backlogged-demands till its last instance of review will ensure that the shipment will reach the retailer, contemplating whom the order was made by the distributor. A Mixed Integer Linear Programming (MILP)-based mathematical formulation of the supply chain (with the objective of minimizing the Total Supply Chain Cost (TSCC)), to obtain optimum policy (*R*, *S*) parameters at each member of the supply chain, and inherently performing the allocation and rationing of stock over a finite planning horizon, is proposed through this paper. A new heuristic allocation and rationing mechanism for the distributor to distribute stock among retailers during the occurrence of shortage named as *Priority Fractional Rationing* (PFR) policy is also introduced in this study. A heuristic methodology which is a hybrid of Genetic Algorithm and Particle Swarm Optimization algorithm (*HGA-PSO*) combined with PFR policy is proposed through this study after analyzing the computational difficulty encountered and lack of tractability of stock-allocation and stock-rationing mechanism while the MILP-based mathematical formulation is solved to optimality. A local search technique as part of the Particle Swarm Optimization (PSO) algorithm, similar to mutation operation in Genetic Algorithm (GA) is introduced due to the observation of inferior results during pilot studies. The performance evaluation studies of the *HGA-PSO* with that of stand-alone GA, stand-alone PSO with new local search and the exact solution obtained by solving the MILP-based mathematical formulation is presented. The results indicate the superior performance of the hybrid algorithm in comparison with stand-alone GA and PSO.

Keywords. Mixed integer linear programming (MILP); hybrid genetic algorithm-particle swarm optimization; allocation; rationing; (*R*, *S*) policy; divergent supply chain.

1. Introduction and review of literature

Firms face the problem of rationing/allocating the resource(s) in addition to the decision regarding stock-levels to be maintained at various installations in a divergent supply chain. Allocation and rationing decisions impact the management of stock in divergent supply chains, especially in the case of scarce product(s) or product(s) having high monetary value. Inventory rationing policies are categorized by Zhang [1] into the following categories: the first is to formulate the rationing problem as a linear programming problem after receiving the demand; the second is to use rules which are predetermined when a shortage arises. In this study, the first approach is used while formulating the problem as a Mixed Integer Linear Program (MILP), and the latter approach is used while solving the problem using metaheuristic algorithms. By solving the MILP-based mathematical model proposed

*For correspondence
Published online: 25 November 2022

through this study, exact solutions to the problem of stock management though (*R*, *S*) policy, allocation and rationing of stock in the divergent supply chain can be obtained. Due to the computational difficulty and lack of tractability of stock-allocation and stock-rationing mechanism observed while solving the mathematical formulation, heuristic rationing rules, termed as *Priority Fractional Rationing* (PFR) policy, combined with a metaheuristic algorithm, namely one of: (1) hybrid of Genetic Algorithm and Particle Swarm Optimization algorithm (*HGA-PSO*); (2) Genetic Algorithm (*GA*); and (3) Particle Swarm Optimization (*PSO*), are proposed. The PFR policy is tractable and can be used in real life, whereas the solution of the *MILP*-based mathematical model is not tractable and it is a lower bound on the objective of minimizing Total Supply Chain Cost (TSCC) (refer John *et al* [2]), but it provides a point of reference (i.e., a bound) for evaluating the performance of heuristic solution methodologies.

Rationing policies for divergent supply chains were developed by many authors and it dates back to the Clark and Scarf [3] model. There are many research publications in this area from then on, and a brief summary of it is given in Table 1.

In the present work, PFR policy for allocating and rationing the stock is inspired partially by *Priority Rationing* (PR) policy (refer Lagodimos [4] and Zhang [1]) and partially by *Fractional Rationing* (FR) policy (refer Paul and Rajendran [12]). The policy is termed as *Priority Fractional Rationing* policy as the first part of the policy is developed on the basis of a priority list and the second part is developed on the basis of rationing-fractions. Lagodimos [4] and Zhang [1] proposed a PR rule in which, the sequence of satisfying orders is on the basis of a list. John *et al* [16] proposed an allocation-rationing mechanism for divergent supply chains operating with lost sales, and solved the mathematical model for 40 time units; and proposed the use of LP relaxation of the original model due to computational limitations of the exact solution methodologies. Unlike these studies, the first part of the PFR policy proposed in the present study is to allocate stock to satisfy backlogged orders from retailers till the distributor's previous instance of review. The sequence in which the distributor performs allocation in this study is on the basis of a predetermined list which depends on backlog cost rates of downstream members. The quantity that is due to retailers till the distributor's previous instance of review is transported to retailers in order of reducing backlog cost-rates, until either the respective retailer is fully satisfied or the product in stock with the distributor is exhausted. Stock allocated to each retailer is on the basis of each retailer's outstanding order(s) and is dynamic. In the work by Paul and Rajendran [12], the first part of the policy is non-existent in both exact as well as heuristic solution methodology.

The second part of the PFR policy is dynamic in nature and does not follow a priority list. The quantity and the sequence in which an order is satisfied vary from period to period, and is on the basis of the magnitude of orders (from downstream members) after the distributor's previous instance of review. The second part of the policy is active only during the occurrence of shortage of stock in the current period for the distributor.

While optimally solving the model using a solver, the look-ahead mechanism, i.e., the ability to foresee demands, aids the MILP solver in making decision regarding stock to be rationed to each downstream member, and inherently makes allocation and rationing decision. In studies available in the literature which proposes heuristic solution techniques to solve the problem of stock rationing, rationing fractions are calculated on the basis of the on-order-inventory of retailers up to the current time period, and quantity to be rationed are calculated using these fractions. In supply chain models discussed in literature, the distributor does not distinguish between: orders from retailer(s) that is backlogged till the distributor's previous instance ofreview; unsatisfied demand after the distributor's previous instance of review till the current instance; and the current period's order(s). Contrary to studies available in literature, in the present work, while solving the mathematical model optimally using a solver, the rationingquantity is decided by the solver making use of its 'look-ahead' mechanism, i.e., ability to foresee the demand and inherently perform calculations; whereas, when the model is solved by metaheuristic algorithms, heuristic rationing-fractions are calculated by making use of the unsatisfied orders after the distributor's previous instance of review, and rationing of stock is performed on the basis of these fractions. It is observable from the literature that the PR policy is used for rationing the entire stock; while in this work, only a part of rationing is performed using priority list.

The motivation behind developing such a stock-rationing policy is the realisation that most of the existing models treat all orders with equal priority while performing stock rationing, i.e., unsatisfied orders and the current day's demand(s) are treated alike by upstream members. In real life it is reasonable to prioritize orders placed ahead in time, thus maintaining a requisite level of service. The model proposed in this work tries to satisfy the unsatisfied orders from retailers, corresponding to which the distributor has already placed order(s) to its upstream member, or the model prioritises those demands from retailers till the distributor's previous instance of review rather than demands from retailers after the distributor's previous instance of review. Demands which are unsatisfied after the distributor's previous instance of review are treated on par with the current period demand. No studies available (to our knowledge) have attempted to model the system along these lines, and have proposed exact and heuristic solution methodologies for the same. This is an important contribution of the present study.

Table 1. Summary of relevant literature on inventory rationing.

Policy name	Name of author(s), year of publication	Brief discussion
Fair share (FS) policy and its variations	Clark and Scarf [3]	Authors proposed the FS policy which aims at equalizing end stock-point's stock-out probability in *N*-echelon serial and divergent systems
	Lagodimos [4]	The author proposed exact models and approximations for service measures in the case of non-identical retailers
	Lagodimos and Koukoumialos [5]	Authors proposed the *Augmented Fair Share* rationing (AFS) using fixed fractions to distribute shortage to end stock-points
Consistent Appropriate Share (CAS) policy and its variations	de Kok [6]	The author proposed CAS policy which aims at maintaining the fixed fraction of projected net inventory of each end stock-point over system-wide projected net inventory. The Author developed exact and heuristic algorithms to determine system parameters and specific fill rates
	Verrijdt and de Kok [7]	Authors extended the analysis to general divergent supply chains with stocks only at the end stock-points and refined the heuristic proposed earlier
Balanced Stock (BS) policy	van der Heijden [8]	The author proposed the BS rule which tries to ration the system wide shortage in a way so that the rationing-fractions minimize average imbalance
	de Kok and Fransoo [9]	Authors proposed an exact solution for the BS rule
Priority Rationing (PR) policy	Lagodimos [4]	The author proposed the PR policy which uses a priority list created using: random rule (RAN); or MIN rule to satisfy the demands of successor stock-points, until the stock is exhausted
	Zhang [1]	The author derived a formula to find near-optimal purchasing quantity for the single-period multi-customer problem when inventory is rationed using randomly generated list
Restricted Time Remembering (RTR) policy	Melchiors [10]	The author assumed the lead-time to be divided into few intervals of time and restricted the critical levels of the RTR policy to be constant over each interval of time
Modified Echelon Stock Rationing (MESR) policy	Huang and Iravani [11]	Authors modeled the production and stock rationing policy as a Markov decision process (MDP) and analyzed the optimal policy characteristics
Fractional Rationing (FR) policy	Paul and Rajendran [12]	Authors proposed heuristic rules to calculate allocation fractions for inventory rationing. They developed both exact as well as metaheuristic methodologies to solve the problem of inventory optimization in a two-stage divergent supply chain
Threshold policy	Kouki and Larsen [13]	Authors considered a spare parts inventory system deploying a base-stock policy and investigate the performance of two rationing policies. First policy is the reservation policy and second being a threshold-based policy
	Vicil [14]	Author considered two models, one being the case where the high-priority customer class can be backordered and the low-priority customer class having no patience and hence leading to loss of sale and; second one being the vice-versa
	Wan and Cao [15]	Author proposed a threshold inventory rationing policy. First come first serve basis is used to treat demand if inventory is above threshold level, if not, the priority demands are satisfied
Priority Fractional Rationing Policy (PFR)	The PFR policy is proposed to determine stock-policy parameters and to perform stock-allocation and stock-rationing in divergent supply chains. The proposed mechanism for allocation and rationing of stock would ensure the shipment of the quantity delivered to the distributor contemplating the recipient to the corresponding retailer	

Several studies available in literature treated all orders, from all downstream members with equal importance while performing stock rationing. The model in this study is a realistic approach to the problem of stock rationing, and also minimizes the effect of the 'look-ahead' mechanism of solver, i.e., ability to foresee demands, thus making rationing decisions to minimize the cost to the maximum extent possible. When the distributor places an order to its upstream member (i.e. manufacturer), it takes into account: all orders received up to the current time period from the downstream members; and quantity ordered till the distributor's previous instance of review. Each order from the distributor reckons with the downstream member's orders after the distributor's previous instance of review, or in other words, distributor places each order to the manufacturer contemplating the recipient. In this work, a MILP-based mathematical formulation to obtain optimal order-policy parameters for stockcontrol and calculation of stock allocation and stock rationing quantity, with the aim of minimizing the TSCC comprising installation-specific stocking cost, backlog cost and cost of ordering is proposed.

Many researchers have time and again brought out the computational complexity and difficulty experienced during the optimal determination of parameters of stock control policy and deciding on the quantity to be rationed (van der Heijden [8]; Shang and Song [17]; and Raghavan and Roy [18]). This ascertains the need for approximate methods to solve the issue of stock optimization and rationing in divergent chains. It is evident from the literature that though these methods could provide only approximate or near optimal solutions, they reduce computational effort drastically. The ease of use and modelling along with very low computational effort makes these algorithms attractive to practicing managers. A brief discussion on the research attempts in the area of hybrid metaheuristic algorithms is presented in the following paragraph.

Several authors have made use of metaheuristic algorithms like *GA*, Simulated Annealing (*SA*) and *PSO* for solving complex problems in stock management, but the use of a hybrid algorithm of *GA* and *PSO* with local search, to the best of our knowledge is minimal. For global optimization of multimodal functions, Kao and Zahara [19] proposed a hybrid method by combining *GA* and *PSO*. With the objective of minimizing surface mount set up time, an evolution-based clustering algorithm which is a hybrid of *GA* and *PSO* was proposed by Kuo and Lin [20]. Diabat [21] addressed the issue of VMI in a supply network consisting of two echelons, and the author tried to find the optimal sales quantity which maximizes the profit optimally and proposed a hybrid of *GA* and SA to overcome the problem of solving the model to optimality. Garg [22] proposed a hybrid of PSO and GA to solve constrained optimization problems and stated that the proposed method could produce results which are superior to existing algorithms while solving engineering problems. Kexin and Qiu [23] proposed a hybrid of GA and PSO to predict properties of crude oil and claimed that the newly proposed hybrid is more accurate and time saving than classical GA or PSO. Moradi *et al* [24] proposed a hybrid PSO-GA technique to optimize buckling load of stiffened laminated composite panels. Authors also proposed a new inertia weight for PSO through their study. Demand side management to reduce electricity bill for customers is modelled by Roy and Das [25] as a multi-objective optimization problem and proposed a hybrid of GA and PSO to solve the problem.

A supply chain model where the distributor distinguishes between: orders from retailer(s) that is backlogged till the distributor's previous instance of review; unsatisfied demand after the distributor's previous instance of review till the current instance; and the current period's order(s) is non-existent in literature. MILP-based mathematical formulation to manage stock while attempting to minimize the cost of the supply chain in the scenario discussed above is also unavailable in literature to the best of our knowledge. Even though several authors have proposed several policies to allocate or ration stock, a policy such as PFR which first performs allocation on the basis of a priority list and then rations the stock remaining with the distributor, thus ensuring that the quantity delivered to the distributor contemplating the recipient is delivered to the corresponding retailer is also non-existent in literature. Several studies in the area of inventory management proposed the use of evolutionary algorithms for solving problems as optimally solving it is computationally complex. However, the use of an algorithm which is a hybrid of GA and PSO with a local search similar to mutation operation in GA embedded to the PSO algorithm and making use of the PFR policy to allocate and ration stock is not available in the literature.

On the basis of the review of the literature, objectives of the present study are set as follows:

- to develop a mathematical formulation (MILP) to obtain optimal solutions to the research question of managing stock, stock allocation and stock rationing in divergent supply chains operating with (*R*, *S*) policy, by minimizing the TSCC comprising installation-specific stocking cost, shortage cost and cost of ordering over a finite planning horizon.
- on account of the computational difficulty encountered while trying to obtain optimal solutions and lack of tractability of mathematical formulations, metaheuristic algorithms coupled with a heuristic mechanism for allocation and rationing of stock, namely the PFR policy is proposed to determine stock-policy parameters and to perform stock-allocation and stock-rationing in divergent supply chains. The proposed mechanism for allocation and rationing of stock would ensure the shipment of the quantity delivered to the distributor contemplating the recipient to the corresponding retailer.

- to propose and compare the performance in terms of cost of: (1) the hybrid metaheuristic, i.e., *HGA-PSO*; (2) stand-alone *GA*; and (3) stand-alone *PSO*, with the optimal solution obtained by solving the mathematical formulation which inherently performs allocation and rationing of stock.

2. Model and assumptions

This research work is concerning a supply chains having one distributor, serving *I* retailers, and retailers serving customers and this section details assumptions of the model. The model is a subject of study over a finite planning horizon, and a day is assumed to be a unit-of-time. Discount policy of any type is not permitted for any installations in this study. All installations operate with sufficient capacity over the entire planning horizon. Every installation of the chain deploys an (*R*, *S*) policy for managing stock and incurs member-specific cost rates. The chain is assumed to operate with a single product and it flows from the distributor to retailers. An order from any retailer is partially/completely fulfilled on the basis of the stock available with the distributor; if the stock available with the distributor is insufficient to fulfill demand in the current period, the unfulfilled demand is backlogged, until the distributor receives shipment in future periods.

The distributor places orders to a manufacturer, and the manufacturer is assumed to be capable of manufacturing sufficient quantity to satisfy any order from the distributor after the stipulated lead-time. The model is inappropriate for those product(s) having shelf-life lower than the finite planning horizon for which the study is performed, and the quality of the product is assumed to be uniform throughout this planning horizon. Transshipments among retailers are disallowed in this study. Orders from customers are deterministic, dynamic and discrete and these are sampled and are known *a priori* over the finite planning horizon. However, the *a priori* knowledge of customer order does not happen in real life. The mathematical formulation proposed in this study to solve the problem makes use of this *a priori* knowledge of demand whereas while deploying the metaheuristic, these demands are available only on period basis. Hence solutions obtained by solving the mathematical model of the problem to optimality indicates a lower bound or the best case scenario whereas the solution from the metaheuristics is real life. Communication of orders from downstream members to upstream members is instantaneous, and the time taken for processing information is minimal and is assumed to be nil.

The day *t* is incremented by a unit of time, i.e., a day, and the same order of events repeats itself over the entire planning horizon. The TSCC is the sum of every member's stocking, backlog and costs incurred in placing an order over *T*, that is the planning horizon. ve sequence of events is presented in section 3. Each of the set of events is accounted for in the mathematical model, for example, the first event in the sequence of events, i.e. with respect to the receipt of material and updating of information pertaining to stock is accounted for in expressions (4) and (5).

3. Mathematical formulation

Two solution methodologies are proposed in this study to obtain the order-policy parameters, i.e.: base-stock level; and review-period of the members of the chain which minimize the TSCC. When to order? How much to order? Are the two questions which any stock management system will have to address, and in this study, there are additional questions of how much stock to allocate and ration to retailers? The first solution technique is an exact solution to the problem using a mathematical programming model which is accurate but computationally intensive. The second is a metaheuristic approach where *PSO*, *GA*, and *HGA-PSO* algorithms are used to determine the member-specific base-stock level and review-period, and they make use of PFR, the heuristic rationing rule to carry out allocation and rationing of stock during shortage. The mathematical model is discussed in this section.

The exact solution to the problem of optimizing of stock, allocation and rationing of stock for the supply chain is obtained by solving the MILP-based mathematical programming model proposed in this section. The mathematical programming model minimizes the TSCC, and can be solved using solvers to give optimal solutions to stock-control policy parameters for every installation of the supply chain. The model is developed based on the sequence of steps mentioned in the previous section (refer figure 1), and a deterministic simulation is carried over the specified run-length. The mathematical programming formulation is given below.

3.1 *Formulation*

Minimize

$$TSCC = \sum_{t=1}^{T} \left\{ \left[\left(h^D \times EI_t^D \right) + \left(O^D \times \delta_t^D \right) \right] + \sum_{i=1}^{I} \left[\left(h_i^R \times EI_{i,t}^R \right) + \left(b_i^R \times B_{i,t}^R \right) + \left(O_i^R \times \delta_{i,t}^R \right) \right] \right\} \quad (1)$$

subject to:

{

[

$$\sum_{r=1}^{\max\_r_i^R} \Delta_{i,r}^R = 1 \quad (2)$$

$$\delta_{i,t}^R - (\Delta_{i,1}^R + \sum_{\substack{r=2 \\ t \bmod r=0}}^{Max\_r_i^R} \Delta_{i,r}^R) = 0 \quad (3)$$

$$BOI_{i,t}^R = EOI_{i,t-1}^R - QSS_{i,t-LT_i^R}^D \quad (4)$$

$$BI_{i,t}^R = EI_{i,t-1}^R + QSS_{i,t-LT_i^R}^D \quad (5)$$

$$EI_{i,t}^R - B_{i,t}^R = BI_{i,t}^R - B_{i,t-1}^R - Dem_{i,t}^R \quad (6)$$

$$QS_{i,t}^R = BI_{i,t}^R - EI_{i,t}^R \quad (7)$$

$$SD_{i,t}^R = SD_{i,t-1}^R + Dem_{i,t}^R \quad (8)$$

$$NOQ_{i,t}^R + OQ_{i,t}^R = SD_{i,t}^R - SOQ_{i,t-1}^R \quad (9)$$

$$OQ_{i,t}^R \leq M^R \times \delta_{i,t}^R \quad (10)$$

$$NOQ_{i,t}^R \leq M^R \times (1 - \delta_{i,t}^R) \quad (11)$$

$$SOQ_{i,t}^R = SOQ_{i,t-1}^R + OQ_{i,t}^R \quad (12)$$

$$EOI_{i,t}^R = BOI_{i,t}^R + OQ_{i,t}^R \quad (13)$$

$], i = 1, 2, ..., I;$

[

$$\sum_{r=1}^{\max\_r^D} \Delta_r^D = 1 \quad (14)$$

$$\delta_t^D - (\Delta_1^D + \sum_{\substack{r=2 \\ t \bmod r=0}}^{Max\_r^D} \Delta_r^D) = 0 \quad (15)$$

$$BOI_t^D = EOI_{t-1}^D - QS_{D,t-LT^D}^M \quad (16)$$

$$BI_t^D = EI_{t-1}^D + QS_{D,t-LT^D}^M \quad (17)$$

$$IBI_{0,t}^D = BI_t^D \quad (18)$$

(

$$A\_Q_{i,t}^D - NA\_Q_{i,t}^D = S\_RP_{i,t-1}^D - SQS_{i,t-1}^D \quad (19)$$

$$A\_Q_{i,t}^D \leq M^D \times \lambda_{i,t}^D \quad (20)$$

$$NA\_Q_{i,t}^D \leq M^D \times (1 - \lambda_{i,t}^D) \quad (21)$$

$$IBI_{i,t}^D - NIBI_{i,t}^D = IBI_{i-1,t}^D - A\_Q_{i,t}^D \quad (22)$$

$$A\_QS_{i,t}^D = IBI_{i-1,t}^D - IBI_{i,t}^D \quad (23)$$

$$IBI_{i,t}^D \leq M^D \times \gamma_{i,t}^D \quad (24)$$

$$NIBI_{i,t}^D \leq M^D \times (1 - \gamma_{i,t}^D) \quad (25)$$

) $\forall i \in \Omega^D$ and retailers are to be contemplated in the sequence they are listed in the set Ω^D. Where Ω^D is an ordered set and, retailers are indexed in a fixed manner such that $b_i^R \geq b_{i+1}^R$.

$$Dem_t^D = \sum_{i=1}^{I} OQ_{i,t}^R \quad (26)$$

$$EI_t^D - B_t^D = BI_t^D - B_{t-1}^D - Dem_t^D \quad (27)$$

$$EI_t^D \leq M^D \times \alpha_t^D \quad (28)$$

$$B_t^D \leq M^D \times (1 - \alpha_t^D) \quad (29)$$

$$SD_t^D = SD_{t-1}^D + Dem_t^D \quad (30)$$

$$NOQ_t^D + OQ_{M,t}^D = SD_t^D - SOQ_{M,t-1}^D \quad (31)$$

$$OQ_{M,t}^D \leq M^D \times \delta_t^D \quad (32)$$

$$NOQ_t^D \leq M^D \times (1 - \delta_t^D) \quad (33)$$

$$SOQ_{M,t}^D = SOQ_{M,t-1}^D + OQ_{M,t}^D \quad (34)$$

$$EOI_t^D = BOI_t^D + OQ_{M,t}^D \quad (35)$$

(

$$R\_QS^D_{i,t} \leq SOQ^R_{i,t} - (SQS^D_{i,t-1} + A\_QS^D_{i,t}) \quad (36)$$

$$QSS^D_{i,t} = R\_QS^D_{i,t} + A\_QS^D_{i,t} \quad (37)$$

$$SQS^D_{i,t} = SQS^D_{i,t-1} + QSS^D_{i,t} \quad (38)$$

$$S\_RP^D_{i,t} \leq S\_RP^D_{i,t-1} + (M^D \times \delta^D_t) \quad (39)$$

$$S\_RP^D_{i,t} \geq S\_RP^D_{i,t-1} - (M^D \times \delta^D_t) \quad (40)$$

$$S\_RP^D_{i,t} \leq SOQ^R_{i,t} + (M^D \times (1 - \delta^D_t)) \quad (41)$$

$$S\_RP^D_{i,t} \geq SOQ^R_{i,t} - (M^D \times (1 - \delta^D_t)) \quad (42)$$

$) \; i = 1, 2, ..., I;$

$$BI^D_t - EI^D_t = \sum_{i=1}^{I} QSS^D_{i,t} \quad (43)$$

$$\sum_{i=1}^{I} R\_QS^D_{i,t} \leq IBI^D_{|\Omega^D|,t} \quad (44)$$

]

$$QS^M_{D,t} = OQ^D_{M,t} \quad (45)$$

$\}, \; t = 1, 2, ..., T;$

with the preliminary settings:

$S\_RP^D_{i,0} = 0$ $\quad i = 1, 2, ..., I.$ (46)

$EOI^R_{i,0} = B^R_{i,0} = SD^R_{i,0} = 0$ $\quad i = 1, 2, ..., I.$ (47)

$SOQ^R_{i,0} = 0$ $\quad i = 1, 2, ..., I.$ (48)

$QSS^D_{i,t-LT^R_i} = 0$ $\quad i = 1, 2, ..., I; t = 1, 2, ..., LT^R_i.$ (49)

$EI^R_{i,0} = S^R_i$ $\quad i = 1, 2, ..., I.$ (50)

$SQS^D_{i,0} = 0$ $\quad i = 1, 2, ..., I.$ (51)

$EOI^D_0 = B^D_0 = SD^D_0 = 0$ (52)

$SOQ^D_{M,0} = 0$ (53)

$QS^M_{D,t-LT^D} = 0$ $\quad t = 1, 2, ..., LT^D.$ (54)

$EI^D_0 = S^D$ (55)

$\delta^R_{i,t} \in (0, 1)$ $\quad i = 1, 2, ..., I; t = 1, 2, ..., T.$ (56)

$\delta^D_t \in (0, 1); \alpha^D_t \in (0, 1)$ $\quad t = 1, 2, ..., T.$ (57)

$\lambda^D_{i,t} \in (0, 1); \gamma^D_{i,t} \in (0, 1)$ $\quad i = 1, 2, ..., I; t = 1, 2, ..., T.$ (58)

all variables are $\geq 0.$ (59)

Figure 2 is a depiction of the run-length in comparison to the solution gap and the time taken while the mathematical programming model is solved during pilot study. With increase in run-length, it is observed that the time taken to solve the mathematical programming model increases exponentially, and this justifies the use of heuristics to solve stock management problems over large planning horizon. The problem instance is solved using a computer with the following specifications: - Bit: 32; processor: Intel(R) Core (TM) i5-2400 CPU @ 3.10 GHz; and RAM: 2 GB. The problem setting used to obtain the results of the pilot study is demand setting-A, explained in detail in section 5.1. The customer demand is sampled *a priori* over various planning horizons, and are given as input to the MILP to perform a deterministic simulation of the model. The inability of the solver to solve the problem instance optimally for run lengths greater than or equal to sixty-time units is observable from figure 2.

Due to the limitation in executing the mathematical model, and optimally solving it over large time horizon, metaheuristic algorithms along with the heuristic rationing rule is proposed in the next section.

4. Heuristic algorithms: *PSO*; *GA*; *HGA-PSO*; and *Priority Fractional Rationing* policy

4.1 *PSO; GA; HGA-PSO algorithms*

GA and *PSO* algorithms are search techniques that have produced excellent results for wide range of problems. The use of *GA* in the area of stock management is well established and several authors proposed many variants of *GA* to suit the respective problem instances and for improving existing results (Zhou *et al* [26], Daniel and Rajendran [27], etc.). *PSO* algorithms are widely used to solve many of the supply chain management problems, but the use of *PSO* specifically in the area of managing stock, to the best of knowledge is found to be limited. Hybrid algorithms are in use in many areas, and discussion is presented in the literature review section. The hybrid algorithm proposed in

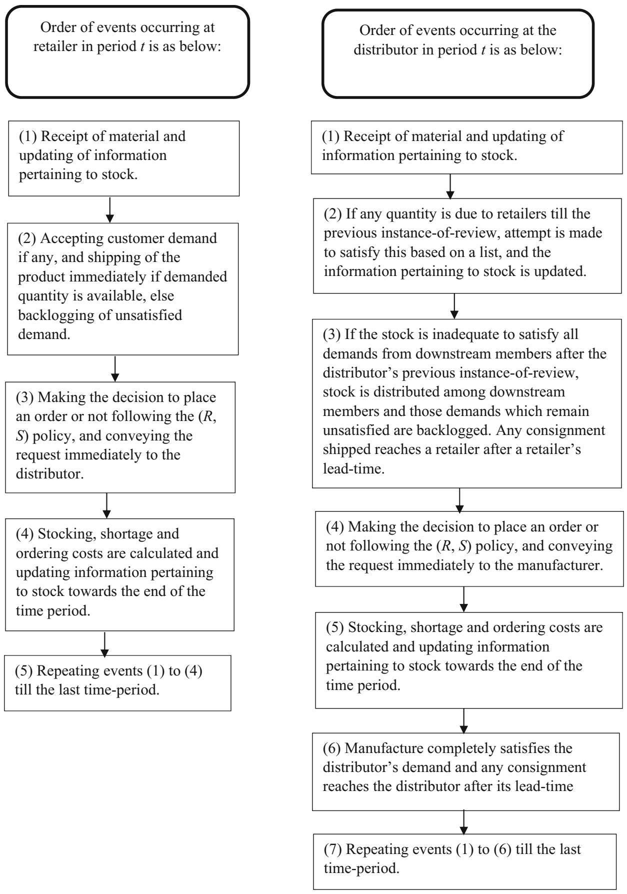

Figure 1. Sequence of events at the distributor and retailers.

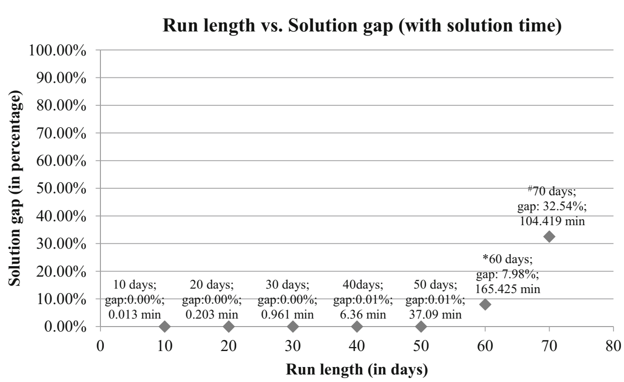

Figure 2. Graphical representation of relation between the run length and the solution gap along with the time taken to solve the mathematical model.

this work is a combination of *GA* and *PSO* algorithms and is used to determine the base-stock level and review-period of installations, and a heuristic-rule is proposed to perform rationing of stock available with the distributor. The performance of the system is evaluated with respect to a set of base-stock levels and review-periods on the basis of TSCC

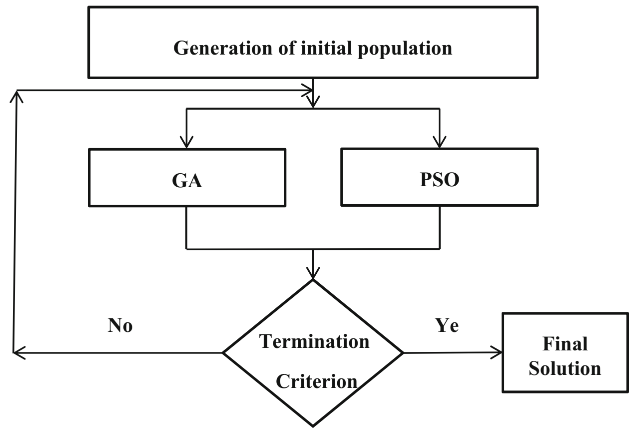

Figure 3. Schematic representation of the Hybrid of GA and PSO algorithm.

through the deterministic simulation of the model over the finite planning horizon. The performance of *GA* and *PSO* algorithms are compared with the hybrid algorithm and results are discussed in the succeeding sections. The heuristic solution methodology is practical, easy to implement, understand, and solves the problem with minimal computational effort for any run-length desired. The schematic representation of the hybrid algorithm is shown in Figure 3.

4.1.1 *Measure of performance (PSO, GA, and HGA-PSO)* The performance of all algorithms is measured for the same run-length and the measure of performance is same as the objective of the MILP-based mathematical programming model and is TSCC which is computed by the following expression (same as expression (1)):

$$TSCC = \sum_{t=1}^{T} \Big\{ \big[\big(h^D \times EI_t^D\big) + \big(O^D \times \delta_t^D\big) \big] + \sum_{i=1}^{I} \Big[\Big(h_i^R \times EI_{i,t}^R\Big) + \Big(b_i^R \times B_{i,t}^R\Big) + \Big(O_i^R \times \delta_{i,t}^R\Big) \Big] \Big\}$$

4.1.2 *Design of the chromosome/particle (PSO, GA, and HGA-PSO)* The design of the chromosome/particle is common for all three algorithms. The chromosome/particle is coded with a set of base-stock levels and review periods using the phenotype representation and is shown in Figure 4. The size of the chromosome/particle depends on the total number of installations and it will be two times (I+1), where I is the maximum number of retailers in the model.

S^D to S_I^R in Figure 4, represent the levels of stock at the distributor and retailers from 1 to I respectively and, RP^D to RP_I^R in Figure 4, represent review-periods of the distributor and retailers from 1 to I respectively.

S^D	S_1^R	S_2^R	.	.	S_{I-1}^R	S_I^R	RP^D	RP_1^R	RP_2^R	.	.	RP_{I-1}^R	RP_I^R

Figure 4. Representation of a chromosome/particle.

4.1.3 *Initial population (PSO, GA, and HGA-PSO)* For all algorithms in this study, this step is common. The initial population size (n) is taken as five times the size of a chromosome/particle. For generating the initial population, base-stock levels are generated randomly within the respective base-stock level limits of the distributor and retailers.

4.1.4 *Separation of the initial population (HGA-PSO)* This step is performed only for the hybrid algorithm, both stand-alone *GA* and stand-alone *PSO* algorithms do not carry out this step. The measure of performance (i.e., TSCC) for each chromosome/particle in the initial population is found out using simulation of the supply chain model. Once TSCC of all chromosomes/particles are obtained, chromosomes/particles in the initial population are sorted in the descending order of their TSCC values. Once the sorting process is complete, the population is divided into two halves. The first half of the sorted population becomes the initial population of *GA* in the hybrid algorithm, and the second half of the sorted population becomes the initial population of *PSO* algorithm in the hybrid algorithm.

4.1.5 *Crossover operations (GA and HGA-PSO)* This step is performed for: the first half of n parent chromosomes/particles of the sorted population, i.e., $(n/2)$ chromosomes/particles fed to the *GA* part of the *HGA-PSO* algorithm; and for n parent chromosomes/particles of the stand-alone *GA*. The gene-wise crossover operator is used in this study for both algorithms (Daniel and Rajendran [27]). The fitness function value f_i for the ith chromosome/particle is computed using Equation (60).

$$f_i = 1/(1 + TSCC_i) \tag{60}$$

Chromosomes/particles equal to the number of genes are selected for crossover using the roulette-wheel procedure. The probability of selecting the ith chromosome/particle for crossover is obtained by computing the following expression. Generate $(n/2)$ off-spring for *HGA-PSO* and n off-spring for *GA* using this method.

$$\left\{f_i \Big/ \sum_{i'}^{n'} f_{i'}\right\} \text{ where } n' = n \text{ for } GA \text{ and } n' = n/2 \text{ for } HGA-PSO \text{ algorithm.} \tag{61}$$

4.1.6 *PSO operations (PSO and HGA-PSO)* This step is performed for: the latter half of the sorted n parent chromosomes/particles of the initial population, i.e., $(n/2)$ parent chromosomes/particles fed to the *PSO* part of the *HGA-PSO* algorithm; and for n parent chromosomes/particles fed to the stand-alone *PSO* algorithm. Social behaviors such as schooling of fish and flocking by birds, etc. has inspired the development of *PSO* which searches for optima on the basis of swarm intelligence. Potential solutions called particles search through the solution space, and are capable of making individual explorations while being at the influence of the swarm. Each chromosome/particle in the swarm is assumed to have gene/position-vector $(x_{i,j})$ and velocity-vector $(v_{i,j})$. In this work, the global-best, i.e., the chromosome/particle with the best solution (minimum TSCC) is termed as $g_{i,j}^{best}$ and each chromosome's/particle's personal-best attained so far is termed as $p_{i,j}^{best}$. Each chromosome/particle will have knowledge of its individual best $(p_{i,j}^{best})$ solution and the swarm's best $(g_{i,j}^{best})$ solution. By best, the reference is to the chromosome/set-of-position-vectors with the smallest value for the measure of performance. Each chromosome/particle in the swarm will try to vary its genes/position-vectors (by accelerating) within the prescribed limits to improve its previous best solutions.

Particles have position-vectors similar to genes in a chromosome, which is randomly generated between limits and limits are explained in subsection 4.1.3. For the first generation, the velocity-vector corresponding to each chromosome/particle is generated randomly between their upper and lower limit, which is set as +/− 4 in the current study (refer to Kennedy *et al* [28]). The $g_{i,j}^{best}$ and the $p_{i,j}^{best}$ for each swarm are found out using simulation. In each subsequent iteration or generation, the $g_{i,j}^{best}$ and $p_{i,j}^{best}$ values of the swarm together with chromosome's/particle's velocity and genes/position-vectors are updated. Updating of velocity-vectors of the gene/position-vector is done using Equation (62) given below. All parameters are finalized after pilot studies, and details are not produced to save space.

Terminology:

$v_{i,j}^{new}$ = New velocity of the ith chromosome/particle, jth gene/position-vector

$v_{i,j}^{old}$ = Old velocity of the ith chromosome/particle, jth gene/position-vector

$x_{i,j}^{old}$ = Old value of the ith chromosome/particle, jth gene/position-vector

$x_{i,j}^{new}$ = New value of the ith chromosome/particle, jth gene/position-vector

ψ = Inertia weight

c_1, c_2 = Acceleration constants
rand () = Uniform random number

$$v_{i,j}^{new} = \left(\psi \times v_{i,j}^{old}\right) + \left(c_1 \times rand() \times \left(p_{i,j}^{best} - x_{i,j}^{old}\right)\right) + \left(c_2 \times rand() \times \left(g_{i,j}^{best} - x_{i,j}^{old}\right)\right) \quad (62)$$

Where ψ is the inertia weight that controls exploration and exploitation of the search that is usually set to a uniform random number between 0.5 and 1. In this work the value of ψ is varied randomly as well as dynamically. The dynamic variation in ψ is brought about by multiplying the randomly generated inertia weight with a value (in this work it is set as 0.925), only in generations of the algorithm where the reminder of dividing the generation number with ten is zero. Acceleration constants (c_1 and c_2) in this work is set to 2 and, *rand*() is a uniform random number. After completing the update of velocity, updating the position is carried out using Equation (63).

$$x_{i,j}^{new} = x_{i,j}^{old} + v_{i,j}^{new} \quad (63)$$

Once the velocity and gene/position-vector update is complete, the $g_{i,j}^{best}$ and $p_{i,j}^{best}$ are updated. $n/2$ off-spring chromosomes/particles of the *HGA-PSO* algorithm and *n* off-spring chromosomes/particles of the *PSO* algorithm are ready for the next step.

4.1.7 *Mutation operations (PSO, GA, and HGA-PSO)* This step is performed on *n/2* off-spring chromosomes/particles after crossover operations, and on *n/2* off-spring chromosomes/particles after *PSO* operations in the *HGA-PSO* algorithm; and *n* off-spring chromosomes/particles after crossover operations in the stand-alone *GA* algorithm; and *n* off-spring chromosomes/particles after *PSO* operations in the stand-alone *PSO* algorithm. The rate of mutation in this study is fixed as 0.10 and, the mutation of a gene/position-vector is performed using the following Equation.

$$x_{i,j}^{new} = x_{i,j}^{old} \times (1 - p) + x_{i,j}^{old} \times 2 \times rand() \times p \quad (64)$$

where *rand*() is a uniform random number and *p* is a value in the range of 0 to 1, and in this study, the value of *p* for off-springs of the *HGA-PSO* algorithm is fixed as 0.05 for the first $n^*/2$ off-springs ($n^* = n/2$), and as 0.10 for the next $3n^*/10$ off-springs, and 0.20 for the last $n^*/5$. *p* is fixed as 0.05 for the first *n*/2 off-springs (out of *n*), *p* is 0.10 for the next 3*n*/10 off-springs (out of *n*), and *p* is 0.20 for the rest of *n*/5 off-springs (out of *n*) of stand-alone *GA* and *PSO* algorithms. Parameters of the *HGA-PSO*, *GA*, and *PSO* algorithms are finalized after carrying out pilot studies, and details are not produced to save space.

4.1.8 *Selection of chromosomes/particles for the next generation (PSO, GA, and HGA-PSO) HGA-PSO* algorithm: $n/2$ off-spring obtained after crossover and mutation operations are combined with $n/2$ chromosomes/particles of the initial population (parents) and the best $n/2$ (in terms of TSCC) is passed on to the next generation; $n/2$ off-spring chromosomes/particles after the *PSO* operations and mutation operations is passed on to the next generation. The combined *n* chromosomes/particles from the *GA* part of the algorithm and the *PSO* part of the algorithm make up the initial population for the next generation. The process of sorting on the basis of the measure of performance is carried out, and chromosomes/particles are supplied to the *GA* and *PSO* algorithms as explained earlier and the cycle continues until the termination criterion is reached.

GA: *n* off-spring generated by crossover and mutation operations along with the initial population (*n*) is evaluated for the measure of performance, which is TSCC. 2*n* chromosomes/particles are sorted in the decreasing order of TSCC value and best *n* chromosomes/particles are passed on to the next generation for the process of generating off-spring (by crossover and mutation operators), and this cycle continues until the termination criterion of *GA* is satisfied.

PSO algorithm: *n* offspring chromosomes/particles, after updating gene/position-vector and mutation operations, forms the initial population for the next generation and the cycle continues until termination criterion of *PSO* algorithm is reached.

4.2 *Priority fractional rationing (PFR) policy*

The metaheuristic algorithms discussed in the previous section are used to generate the base-stock levels and the review periods whereas heuristic allocation and rationing rule (PFR) is used to distribute stock to retailers. The quantity backlogged by the distributor till its previous instance of review are allocated using the priority allocation rule and the quantity backlogged after the last instance of review along with the demand of the present period is rationed (if the distributor is devoid of sufficient stock) using the fractional rationing rule. The rationing policy in this study is a combination of: allocation on the basis of priority rule; and rationing on the basis of fraction rule, and hence the policy is termed as *Priority Fractional Rationing* policy (PFR).

A discussion on how the policy is different in comparison to the existing policies is essential and is presented in this section. Lagodimos [4] proposed the *Priority Rationing* (PR) *Rule* and the objective of the PR rule is to reduce the number of end stock points that are unsatisfied. Whereas PFR tries to minimize the system wide shortage cost and do not attempt to minimize the number of end stock points which are unsatisfied. The logic of using a list rendering priority to certain downstream members is the only idea that is borrowed from the PR rule proposed by Lagodimos [4]. The functioning of the PFR rule is different to the functioning of the PR rule as stock allocated by the rule proposed is dynamic and depends on the demand quantity backlogged by the distributor till its previous instance of review. The *Fractional Rationing* rule proposed by Paul and Rajendran [12] made use of an expression developed on the basis of the on-order inventory of retailers to obtain rationing-fractions pertaining to each retailer, and performed the rationing of stock on the basis of these fractions and the distributor's stock. In this study, sum of: demands backlogged by the distributor to each retailer after the distributor's previous instance-of-review (up-to the previous period); and the current period's demands are used to obtain the rationing-fraction. On the basis of these rationing-fractions and the distributor's stock (available after allocation) the quantity (rationed) to be shipped to each retailer in the current period is calculated. Hence it is different from what is available in literature, and this rationing policy reduces the look-ahead capacity of the solver, and hence is realistic. The reason it is realistic is because, the model in this study tries to satisfy orders from downstream members in the order of occurrence. The model does not substitute an order placed by a retailer at an earlier period for an order placed by another retailer in a later period. Stockallocation and rationing mechanism in this study respects the order of occurrence of retailer demands while trying to minimize the TSCC. A study in this direction which gives importance to the order of occurrence of demand and trying to minimize the cost is, to the best of our knowledge, nonexistent in literature.

The rationing rule proposed in this study tries to have the same probability for stock-out for retailers when retailers operate with identical cost-rates and is similar to the FS rule (see table 1) in this scenario. CAS rule (see table 1) rations inventory on the basis of the ratio of safety-stock at a retailer to the total safety-stock at all retailers. Paul and Rajendran [12] explained that an attempt by authors in developing a rule based on CAS, making use of the ratio of base-stock at retailers to calculate fractions for rationing did not provide encouraging results. Hence an investigation in this direction is ruled out in the present work. Both *Balanced Stock* (BS; an improvement over CAS) and *Linear Rationing* (LR) class of rationing rules tries to reduce imbalances at every retailer. In this study, as the entire stock at the distributor is rationed to downstream members, the question of negative allocation fractions and hence imbalances does not arise. Thus, a model which respects the order of occurrence of demands, and a policy combining allocation on the basis of priority rule and rationing on the basis of fractional rule is a step forward in the area of stock-rationing and contributes significantly to the literature.

The heuristic stock-rationing policy proposed in this study is a combination of two processes. The first one is to allocate the unsatisfied demands till the distributor's previous instance-of-review on the basis of a priority list. The list is based on the shortage cost-rate of retailers, and retailers are arranged in the descending order of their backlog cost-rates, thus capturing the importance of a retailer. Higher the backlog cost-rate associated with a retailer, higher is its importance and a retailer being more important than another is possible in real life. The model keeps track of demands from each retailer till the distributor's previous instance of review. At the start of every time epoch, after updating the stock, the distributor attempts to satisfy the outstanding demands from retailers according to the priority list until either the quantity stocked by the distributor is depleted or the outstanding demands from retailers till the distributor's previous instance of review is satisfied. Stock allocated to a retailer by the distributor as per the order of priority is obtained using Equation (65).

$$A\_QS^D_{i,t} = \min\left\{\left(S\_RP^D_{i,t-1} - SQS^D_{i,t-1}\right), BI^D_t\right\} \tag{65}$$

Stock to be allocated to the first retailer in the priority list (based on shortage cost-rates) is calculated using Equation (65). Once stock-allocated to the first retailer (in the list) is shipped, the information regarding stock with the distributor is updated, and stock to be allocated for the second retailer in the priority list is calculated using Equation (65). This process is continued until the quantity stocked by the distributor is exhausted or stock-allocated is shipped to the corresponding retailer.

The second part of the policy is to disburse the unsatisfied demand after the distributor's previous instance of review, and the demand of the current day. If the distributor has sufficient stock (remaining after allocation) then the demands from retailers are completely satisfied with no backlogging. If the stock available after allocation is not adequate to satisfy demands from retailers, then the rationing fractions are calculated making use of Equation (66):

$RF_{i,t}$: Rationing-fraction of retailer i in time period t, given by

$$RF_{i,t} = \left\{\frac{\left(SOQ^R_{i,t} - \left(SQS^D_{i,t-1} + A\_QS^D_{i,t}\right)\right)}{\sum_{i''}^{I}\left(SOQ^R_{i'',t} - \left(SQS^D_{i'',t-1} + A\_QS^D_{i'',t}\right)\right)}\right\}. \tag{66}$$

Once the rationing fractions corresponding to all retailers are calculated, the distributor calculates the quantity of stock to be rationed to each retailer, and the distributor calculates this making use of Equation (67).

$$R\_QS^D_{i,t} = \min\left\{\left(RF_{i,t} \times \left(BI^D_t - \sum_i^I A\_QS^D_{i,t}\right)\right), \left(SOQ^R_{i,t} - \left(SQS^D_{i,t-1} + A\_QS^D_{i,t}\right)\right)\right\} \quad (67)$$

First, the retailer with the highest $RF_{i,t}$ is identified and $R\_QS^D_{i,t}$ is calculated for this retailer. Once $R\_QS^D_{i,t}$ for the retailer with the highest $RF_{i,t}$ is calculated, the information pertaining to stock with the distributor is updated. Then $RF_{i,t}$ for remaining retailers are recalculated, and the $R\_QS^D_{i,t}$ for the retailer with the highest $RF_{i,t}$ (among set of retailers, excluding the retailer(s) for which $R\_QS^D_{i,t}$ has been estimated) is calculated. This process is continued until the stock with the distributor is exhausted.

The total shipment to any retailer will be the sum of stock-allocated and stock-rationed to the retailer and is calculated by the following Equation.

$$QSS^D_{i,t} = R\_QS^D_{i,t} + A\_QS^D_{i,t} \quad (68)$$

The functioning of the allocation part of stock-rationing policy in the case of exact and heuristic solution methodologies is theoretically similar. Even though they are theoretically similar, stock-allocated on each day can be different in both methods as the base-stock levels and review-periods could be different. The rationing part of the rule is not the same in both the solution methods as the solver used to solve the mathematical formulation has the unique 'look-ahead' capability which enables it to foresee demand over the entire planning horizon, and makes rationing decisions with the objective of minimizing the system wide cost. The heuristic solution methodology based on *HGA-PSO*, *GA* and *PSO* lacks the 'look-ahead' ability, as it is in real life situations, and solves the problem of stock-rationing on a day to day basis making use of the PFR rule proposed in this section. A metaheuristic making simultaneous use of an allocation and rationing rule to distribute stock to downstream members, to the best of knowledge, is the first of its kind, and is a step towards reality.

5. Experiments, results and discussions

5.1 *Experiments*

To evaluate the performance of techniques to solve the problem discussed through this paper, the number of retailers is set as 4, that is $I = 4$ while performing experiments. It is to be noted that the number of retailers can be greater than or less than 4, that is I can be varied according to the situation. The mathematical model and the heuristic algorithms are solved over a planning horizon of 40 time units ($T = 40$) and the heuristic algorithms alone are solved over 400 time units ($T = 400$). The run-length was limited to 40 time units due to the drastic increase in computational time, and at times termination of the solver citing limitations in executing the mathematical model beyond this run-length (see Figure 2). It is to be noted that the heuristic algorithms proposed in this study are capable of solving the problem for any specified time duration. However, the run-length is fixed as 40 time units for making an absolute comparison of the performance of the heuristic algorithms to the solution of the mathematical model and results are reported in section 5.2. All heuristic algorithms are also solved over a period of 400 time units and results are reported in section 5.3. The MILP-based mathematical programming model can be solved by any solver, but in this study an IBM ILOG CPLEX Optimization Studio is used. The settings (demand and cost) used to perform experiments are discussed in this section. In experiments conducted, the review-period of any member is limited to the value of 5, i.e., $Max\_r^R_i = 5; \forall i \leq I$ and $Max\_r^D = 5$. It is found in literature that stocking cost can be used to approximate other costs (see Silver and Peterson [29]). The stocking cost-rate of all retailers is fixed at two monetary units per unit in stock. The shortage cost-rate for any retailer can be set as $b^R_i = k_i \times h^R_i$ where $k_i = 5, 4, 3, 2$ for $i = 1, 2, 3, 4$ respectively for the cost setting. The stocking cost-rate of the distributor is set as one monetary unit per unit-product per period in stock. The shortage cost-rate of the distributor is taken as zero, as actual shortages occur only at the retailer. As supply chains operate to maximize the profitability of the entire chain, assuming any shortage cost-rate for the distributor is duplication of the actual shortage cost-rate incurred by the retailer. The ordering cost of installations are taken as a multiple of: the stocking cost-rate of retailers; the maximum expected demand among retailers; and a multiplier, and can be generalized as $h^R_i \times \max(E(Dem^R_i) \times 2$ among all i. $E(Dem^R_i)$ is the expectation of customer demand of retailer i and is taken as the mean demand from the customer to the particular retailer. Such settings are used with the aim of testing the performance of the proposed solution techniques. The lead-time for all members is taken as one in this study. The customer demand in this work is sampled *a priori* from a uniform distribution between the minimum and maximum values mentioned in table 2. The sampled demands are given as input to the MILP, hence the demands are dynamic and deterministic in all experiments. Given the minimum and maximum values that a random variable can take, the variance is maximum when the random variable is assumed to follow uniform distribution. Three demand settings, three demand streams in all the demand settings, and a cost setting is considered in this study. All the instances are solved using: (1) Cplex optimizer; (2) *HGA-PSO* algorithm; (3) stand-alone *GA*; (4) stand-alone *PSO* algorithm over a run-length of 40 days, and *HGA-PSO* algorithm, stand-alone *GA*, stand-alone *PSO* algorithm over a run-length of

Table 2. Demand setting across retailers.

Installation	A	B	C
	Demand setting (*Minimum demand*, *maximum demand*)		
Retailer 1	(0, 80)	(0, 80)	(0, 20)
Retailer 2	(0, 80)	(0, 60)	(0, 40)
Retailer 3	(0, 80)	(0, 40)	(0, 60)
Retailer 4	(0, 80)	(0, 20)	(0, 80)

Table 3. TSCC for all demand settings and various solution techniques over the time horizon of 40 time units.

Demand stream	Optimal	*HGA-PSO*	*GA*	*PSO*
Demand setting-A				
1	28767	31708	31736	31741
2	28767	29585	30454	29585
3	28286	31320	31339	32659
Demand setting-B				
1	22455	22748	22748	24575
2	22667	23929	24550	26498
3	22792	24349	25590	24605
Demand setting-C				
1	21174	22437	22437	22661
2	23542	23911	23911	24349
3	21380	22011	22011	22085

PSO particle swarm optimization, *GA* genetic algorithm, *HGA-PSO* hybrid of genetic algorithm and particle swarm optimization, *Optimal* optimal solution.

400 days. The value of M^R and M^D are fixed as 10000 during experiments.

In experiments, a supply chain functioning with one distributor and four retailers is considered, hence, number of genes in a chromosome/particle of metaheuristic algorithms is 10 and the value of n is 50. The termination criterion of all three algorithms (*HGA-PSO*, *PSO* and *GA*) is fixed as 500, and all the parameters mentioned in section 5.1 are fixed after carrying out extensive pilot studies.

Table 5. TSCC for all demand settings and various metaheuristic solution techniques over the time horizon of 400 time units.

Demand stream	*HGA-PSO*	*GA*	*PSO*
Demand setting-A			
1	319149	319149	343580
2	332389	332389	332389
3	315855	315855	321532
Demand setting-B			
1	242017	244430	251753
2	244800	246400	244800
3	245365	249352	249827
Demand setting-C			
1	230176	236674	232833
2	236357	240762	240762
3	233050	234933	234933

PSO particle swarm optimization, *GA* genetic algorithm, *HGA-PSO* hybrid of genetic algorithm and particle swarm optimization.

5.2 *Results and discussions (40 time units)*

Results obtained by solving the mathematical formulation and the metaheuristic algorithms over the time horizon of 40 time units are presented in table 3.

Results presented are the TSCC of three demand streams each with a run-length of over the time horizon of 40 days for each demand setting. Cost setting represents the case of retailers being dissimilar as the shortage cost-rates of retailers are assumed to be different. A retailer operating with higher shortage cost-rate represents the scenario where the retailer is serving product to critical customers. The value of the same product could vary from customer to customer depending on several factors, and researchers in the past have used shortage cost rate to differentiate between retailers. Möllering and Thonemann [30] used shortage cost-rates to differentiate between retailers. The percentage deviation of TSCC of metaheuristic algorithms from the exact solution is calculated for each demand stream using the expression $\{(TSCC_{heuristic} - TSCC_{exact})/TSCC_{exact}\} \times 100$. From results in table 4, it is observable that the *HGA-PSO* outperforms *GA* algorithm and *PSO* algorithm in several scenarios or performs equally well as *GA* or *PSO*. From this it

Table 4. Percentage deviation of TSCC of metaheuristic algorithms from the optimal solution over the time horizon of 40 time units.

Demand stream	Demand setting-A			Demand setting -B			Demand setting -C		
	HGA-PSO	*GA*	*PSO*	*HGA-PSO*	*GA*	*PSO*	*HGA-PSO*	*GA*	*PSO*
1	**10.22**	10.32	10.34	**1.30**	**1.30**	9.44	**5.96**	**5.96**	7.02
2	**2.35**	5.35	**2.35**	**5.57**	8.31	16.90	**1.57**	**1.57**	3.43
3	**10.73**	10.79	15.46	**6.83**	12.28	7.95	**2.95**	**2.95**	3.30

PSO particle swarm optimization, *GA* genetic algorithm, *HGA-PSO* hybrid of genetic algorithm and particle swarm optimization.

A positive entry in the above table denotes the inferior performance of the corresponding algorithm with respect to the optimal solution.

Bold numbers indicate minimum of values for a demand setting

Table 6. Percentage deviation of TSCC obtained from *GA* and *PSO* algorithms from the *HGA-PSO* solution over the time horizon of 400 time periods.

	Demand setting-A		Demand setting-B		Demand setting-C	
Demand stream	*GA*	*PSO*	*GA*	*PSO*	*GA*	*PSO*
1	**0.00**	7.66	**1.00**	4.02	2.82	**1.15**
2	**0.00**	**0.00**	0.65	**0.00**	**1.86**	**1.86**
3	**0.00**	1.80	**1.62**	1.82	**0.81**	**0.81**

PSO particle swarm optimization, *GA* genetic algorithm, *HGA-PSO* hybrid of genetic algorithm and particle swarm optimization

A positive entry in the above table denotes the inferior performance of the corresponding algorithm with respect to the *HGA-PSO*

Bold numbers indicate minimum of values for a demand setting

can be inferred that the hybrid metaheuristic technique can be employed effectively in problems relating to managing stock in divergent supply chains.

5.3 *Results and discussions (400 time units)*

With the same set of parameters (same as over the time horizon of 40 time units) and the same cost and demand settings, and demand streams, the heuristic algorithms are solved over a period of 400 time units and results are reported in table 5.

The percentage deviation of solutions of *GA* and *PSO* algorithms from the *HGA-PSO* algorithm solution are calculated for each demand stream using the expression $\left\{(TSCC_{GA/PSO} - TSCC_{hGA-PSO})/TSCC_{hGA-PSO}\right\} \times 100$, and are presented in table 6. From results, it is observable that the *HGA-PSO* outperforms the *GA* algorithm and the *PSO* algorithm in several scenarios or performs equally well as *GA* or *PSO* algorithms even for large run lengths, and hence indicates its usage for problems in the area of managing stock.

The CPU time taken to obtain a solution using any metaheuristic method coupled with the heuristic rationing rule does not differ much with demand streams, and it is possible to execute it for large run lengths. Even though the performance of the metaheuristic solution methodologies can never better exact solutions, they are more realistic in approach and are computationally less intensive. By being realistic, it is implied that metaheuristic solution methodologies solve the problem of stock allocation and stock rationing on a day-to-day basis. The PFR policy is tractable in real life, and can be calculated with ease during any attempt to apply the method in real life. This method is applicable in the real-life scenario where demand data is not known with 100% accuracy, and is a helpful tool for managers to make decisions regarding how much to store, when to order and how to ration in the scenario where shortages arise. The solution to the MILP-based mathematical programming model performing rationing inherently can be used as a benchmark to evaluate the performance of the metaheuristic solution techniques which are more realistic, easy to model, computationally less intensive and can handle any amount of difficulty without much effect to the quality of the solution. The computational difficulty along with the solution being intractable does make the MILP-based mathematical model unattractive to real life applications. For example, the mathematical model took approximately 4 hours to solve one of the demand streams and the solver terminated citing execution limitations during attempts to execute the model beyond the run-length of 40 for many demand streams. Even though the MILP-based mathematical model is computationally difficult, it does provide a platform for researchers and practicing managers to test the performance of heuristic solution methods which can capture real-life scenarios effectively and efficiently as the solutions to the mathematical model can be considered as a lower bound to the kind of problems discussed in this paper.

6. Conclusions and scope for future research

This paper addresses the problem of management of stock through (R, S) policy in a two-stage supply chain which is divergent. The supply chain considered in this study prioritizes the sequence of occurrence of orders from retailers to the distributor while clearing backorders, and hence ensures the supply of the product to the right recipient, i.e., the retailer contemplated by the distributor during the placement of an order. A MILP-based mathematical formulation capable of providing optimal results of review period and base-stock level for all members of the supply chain while allocating and rationing stock is proposed in this study. Owing to the lack of tractability and computational difficulty observed while solving the mathematical formulation to optimality, a new heuristic rationing policy termed as *priority fractional rationing* policy is proposed in this study to allocate and ration stock to downstream members when the distributor faces with the shortage of units. The PFR policy is tractable, and can be used in real-life, whereas the solution of the *MILP*-based mathematical model is not tractable and it is a lower bound on the objective of minimizing total cost of the supply chain, but it provides a point of reference (i.e., a bound) for evaluating the performance of heuristic solution methodologies.

A solution methodology which is a hybrid of *GA* and *PSO* algorithm in association with the PFR policy is proposed in this work. The proposed hybrid algorithm along with the heuristic rationing policy provides results which are superior to or equal to stand-alone *GA* and stand-alone *PSO* algorithms for the problem instances that the model has been tested with. The introduction of a local search similar to mutation in GA as part of the PSO algorithm has yielded better results and is recommended for further studies. The proposed hybrid algorithm's performance is compared with the exact solution as well as with solutions obtained using stand-alone *GA* and stand-alone *PSO* and the percentage deviation from optimal solution is promising, but can be improved further. New rules for generating priority list for allocating stock is a thought for the future. New stock-control policies can be attempted in future along with new rationing rules and supply chains with several products, several stages and more than one distributor can be a possible extension of this research.

List of symbols

t Current time period or present day

i Indicator for retailer, and $i = 1, 2, \ldots, I$. Retailer is represented by R, distributor by D and manufacturer by M

Parameters

T	Total duration of time
I	Maximum number of retailers
h_i^R	Stocking cost-rate of the *i*th retailer
b_i^R	Backlog cost-rate of the *i*th retailer
O_i^R	Cost incurred by the *i*th retailer per order
h^D	Stocking cost-rate of the distributor
O^D	Cost incurred by the distributor per order
$Max\_r_i^R$	Upper-limit on review days for the *i*th retailer
Max_r^D	Upper-limit on review days for the distributor
LT_i^R	Lead-time of the *i*th retailer
LT^D	Lead-time of the distributor
M^R	A very big value (retailer)
M^D	A very big value (distributor)

Decision variables

r	Variable taking the possible values of review-period. /*note: r is from $\{1, 2, ..., \max\_r_i^R\}$*/
$\Delta_{i,r}^R$	Binary variable (1 if r is the review-period of the *i*th retailer; else 0). /*note: r is from $\{1, 2, ..., \max\_r_i^R\}$*/
$\delta_{i,t}^R$	Binary variable (1 if the *i*th retailer places an order in period t; else 0)
Δ_r^D	Binary variable (1 if the period of review is r for the distributor; else 0). /*note: r is from $\{1, 2, ..., \max\_r_i^R\}$*/
δ_t^D	Binary variable (1 in the *t*th period if the distributor makes an order; else 0)
$BOI_{i,t}^R$	Quantity that is ordered but yet to be received by the *i*th retailer and, that is calculated after taking in to account the consignment received from the upstream member in the start of the *t*th day
BOI_t^D	Quantity that is ordered but yet to be received by the distributor and, that is calculated after taking in to account the consignment received from the upstream member in the start of the *t*th day
$BI_{i,t}^R$	Quantity that is in stock with the *i*th retailer and, that is calculated after taking in to account the consignment received from the upstream member in the start of the *t*th day
BI_t^D	Quantity that is in stock with the distributor and, that is calculated after taking in to account the consignment received from the upstream member in the start of the *t*th day
$EI_{i,t}^R$	Quantity that is in stock with the *i*th retailer towards the close of the *t*th day
EI_t^D	Quantity that is in stock with the distributor towards the close of the *t*th day
$EOI_{i,t}^R$	Quantity that is ordered but yet to be received by the *i*th retailer towards the close of the *t*th day
EOI_t^D	Quantity that is ordered but yet to be received by the distributor towards the close of the *t*th day
$B_{i,t}^R$	Demand that is backlogged by the *i*th retailer towards the close of the *t*th day
B_t^D	Demand that is backlogged by the distributor towards the close of the *t*th day
$QS_{i,t}^R$	Quantity that is transported by the *i*th retailer to its customers towards the close of the *t*th day
$R\_QS_{i,t}^D$	Quantity that is rationed to the *i*th retailer by the distributor towards the close of the *t*th day
$A\_QS_{i,t}^D$	Quantity that is allocated to the *i*th retailer by the distributor towards the close of the *t*th day
$QSS_{i,t}^D$	Sum of quantity that is rationed and allocated to the *i*th retailer by the distributor for the *t*th day and, towards the close of the *t*th day
$SQS_{i,t}^D$	Total quantity transported to the *i*th retailer by the distributor up-to the *t*th day
$QS_{D,t}^M$	Quantity transported from the manufacturer to the distributor towards the close of the *t*th day
$Dem_{i,t}^R$	Order from customer to the *i*th retailer in the *t*th day
Dem_t^D	Distributor's demand in the *t*th day
$SD_{i,t}^R$	Demand from customers to the *i*th retailer up-to the *t*th day
SD_t^D	Demand from all retailers to the distributor up-to the *t*th day
$OQ_{i,t}^R$	Order by the *i*th retailer in the *t*th day

$OQ^D_{M,t}$	Order by the distributor to manufacturer in the tth day
$NOQ^R_{i,t}$	Dummy variable corresponding to the ith retailer in the tth day that is introduced to avoid infeasibility of the model
NOQ^D_t	Dummy variable corresponding to the distributor in the tth day that is introduced to avoid infeasibility of the model
$SOQ^R_{i,t}$	Orders by the ith retailer up-to the tth day
$SOQ^D_{M,t}$	Orders by the distributor up-to the tth day
$A\_Q^D_{i,t}$	Quantity allocated to the ith retailer in the tth day by the distributor
$NA\_Q^D_{i,t}$	Dummy variable corresponding to the distributor in the tth day that is introduced to avoid infeasibility of the model which could arise due to allocation
$IBI^D_{i,t}$	Interposed-stock of the distributor that is available for allocation to the ith retailer in the tth day
$NIBI^D_{i,t}$	Dummy variable corresponding to the distributor in the tth day that is introduced to avoid infeasibility of the model which could arise due to the updating of stock
$S\_RP^D_{i,t}$	Total quantity ordered by the ith retailer till the distributor's previous instance-of-review before the tth day
$\lambda^D_{i,t}$	Binary variable (1 if the distributor allocates stock to the ith retailer; else 0)
$\gamma^D_{i,t}$	Binary variable (1 if $IBI^D_{i,t}$ positive; else 0)
α^D_t	Binary variable introduced to avoid the simultaneous existence of EI^D_t and B^D_t
S^R_i	Base-stock level of the ith retailer
S^D	Base-stock level of the distributor
$TSCC$	Total cost over all members and for total period

Acknowledgements

This research work has been carried out with the financial support from IIT Madras, University of Passau and DAAD as a part of the joint Ph.D. degree program. The authors are thankful to reviewers and the Editor for their suggestions and comments to improve the earlier version of the manuscript.

References

[1] Zhang J 2003 Managing multi-customer service level requirements with a simple rationing policy. *Operations Research Letters.* 31: 477–482

[2] John K, Rajendran C and Ziegler H 2016 Investigation of order-up-to policy and allocation-rationing mechanism for divergent supply chains with multiple objectives. In: *Multiple Criteria Decision Making in Supply Chain Management, U.S.A.* (ed) Ravindran A R, CRC Press, Boca Raton

[3] Clark A J and Scarf H 1960 Optimal policies for multi-echelon inventory problem. *Management Science.* 6: 475–490

[4] Lagodimos A G 1992 Multi-echelon service models for inventory systems under different rationing policies. *International Journal of Production Research.* 30: 939–958

[5] Lagodimos A G and Koukoumialos S 2008 Service performance of two-echelon supply chains under linear rationing. *International Journal of Production Economics.* 112: 869–884

[6] de Kok A G 1990 Hierarchical production planning for consumer goods. *European Journal of Operational Research.* 45: 55–69

[7] Verrijdt J H C M and de Kok A G 1996 Distribution planning for a divergent depot less two-echelon network under service restrictions. *European Journal of Operational Research.* 89: 341–354

[8] van der Heijden M C 1997 Supply rationing in multi-echelon divergent systems. *European Journal of Operational Research.* 101: 532–549

[9] de Kok A G and Fransoo J C 2003 Planning supply chain operations: Definition and comparison of planning concepts. In: *Supply Chain Management: Design, Coordination and Operation* (eds) de Kok A G and Graves S C, Elsevier, Amsterdam, pp 597–675

[10] Melchiors P 2003 Restricted time-remembering policies for the inventory rationing problem. *International Journal of Production Economics.* 81–82: 461–468

[11] Huang B and Iravani S M R 2007 Optimal production and rationing decisions in supply chains with information sharing. *Operations Research Letters.* 35: 669–676

[12] Paul B and Rajendran C 2011 Rationing mechanisms and inventory control-policy parameters for a divergent supply chain operating with lost sales and costs of review. *Computers and Operations Research.* 38: 1117–1130

[13] Kouki C and Larsen C 2021 Rationing policies in a spare parts inventory system with customers differentiation. *International Journal of Production Research.* 59(20): 6270–6290

[14] Vicil O 2021 Inventory rationing on a one-for-one inventory model for two priority customer classes with backorders and lost sales. *IISE Transactions.* 53(4): 472–495

[15] Wan G and Cao Y 2018 A continuous cost evaluation approach for periodic review inventory systems with threshold rationing policy. *Computers & Industrial Engineering.* 126: 75–87

[16] John K, Rajendran C and Ziegler H 2019 A comparative study on allocation/rationing mechanisms operational with/without backorder clearing in divergent supply chains. *Sadhana.* 44: 231

[17] Shang K H and Song J-S 2003 Newsvendor bounds and heuristic for optimal policies in serial supply chains. *Management Science.* 49: 618–638

[18] Raghavan N R S and Roy D 2005 A stochastic Petri net approach for inventory rationing in multi-echelon supply chains. *Journal of Heuristics.* 11: 421–446

[19] Kao Y and Zahara E 2008 A hybrid genetic algorithm and particle swarm optimization for multimodal functions. *Applied Soft Computing.* 8: 849–857
[20] Kuo R J and Lin L M 2010 Application of a hybrid of genetic algorithm and particle swarm optimization algorithm for order clustering. *Decision Support Systems.* 49: 451–462
[21] Diabat A 2014 Hybrid algorithm for a vendor managed inventory system in a two-echelon supply chain. *European Journal of Operations Research.* 238: 114–121
[22] Garg H 2016 A hybrid PSO-GA algorithm for constrained optimization problems. *Applied Mathematics and Computation.* 274: 292–305
[23] Bi Kexin and Qiu T 2019 An intelligent SVM modeling process for crude oil properties prediction based on a hybrid GA-PSO method. *Chinese Journal of Chemical Engineering.* 27–8: 1888–1894
[24] Moradi S, Vosoughi AR and Anjabin N 2021 Maximum buckling load of stiffened laminated composite panel by an improved hybrid PSO-GA optimization technique. *Thin-Walled Structures.* 160: 107382
[25] Roy C and Das D K 2021 A hybrid genetic algorithm (GA)–particle swarm optimization (PSO) algorithm for demand side management in smart grid considering wind power for cost optimization. *Sādhanā* 46: 101
[26] Zhou G, Min H and Gen M 2002 The balanced allocation of customers to multiple distribution centers in the supply chain network: a genetic algorithm approach. *Computers & Industrial Engineering.* 43: 251–261
[27] Daniel S R J and Rajendran C 2005 A simulation-based genetic algorithm for inventory optimization in a serial supply chain. *International Transactions in Operational Research.* 12: 101–127
[28] Kennedy J, Eberhart R C and Shi Y 2001 Swarm Intelligence. The Morgan Kaufmann Series in Evolutionary Computation. Morgan Kaufmann Publishers, San Francisco
[29] Silver E A and Peterson R 1985 Decision Systems for Inventory Management and Production Planning. 2nd edn. Wiley, New York
[30] Möllering K T and Thonemann U W 2008 An optimal critical level policy for inventory systems with two demand classes. *Naval Research Logistics.* 55: 632–642