Variable Neighborhood Search for Flowshop Scheduling Problem with Sequence Dependent Setup Times

by

Güneş Yılmaz

A Thesis Submitted to the
Graduate School of Sciences and Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Master of Science
in
Industrial Engineering

Koç University

August, 2014

Koç University

Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a master's thesis by

Güneş Yılmaz

and have found that it is complete and satisfactory in all respects,

and that any and all revisions required by the final

examining committee have been made.

Committee Members:

____

Prof. Dr. Ceyda Oğuz (Advisor)

____

Prof. Dr. Selçuk Karabatı

____

Assoc. Prof. Dr. F. Sibel Salman

Date: ______________________________

ABSTRACT

In production facilities, most of the end product is processed on a set of machines to be formed. Increase in diversity of the product raises the scheduling problem on a work shop environment for processing the multiple jobs on a set of machines. In this study, we examine the flowshop scheduling problem with sequence dependent setup times (FS-SDST). In regular flowshop problems, the setup time of the jobs or the machines are considered as negligible or independent from the sequence of the jobs. However, in many applications, some setup operations such as cleaning, changing or adjusting the machine tools are required for the machine before processing the following job in the sequence. In the thesis, we study two FS-SDST problems: $F|s_{ijl}$, $prmu|C_{max}$ and $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$. In the $F|s_{ijl}$, $prmu|C_{max}$ problem, we aim to schedule the jobs to be processed on the machines when the objective is to minimize the maximum completion time which is called makespan. In the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem, we study energy-aware FS-SDST problem, in which the aim is to schedule the jobs to be processed on all machines. However, in the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem the objective is to minimize both the total completion time and the total energy consumption.

We propose a Variable Neighborhood Search (VNS) algorithm for these two FS-SDST problems. We examine the performance of the VNS algorithm by using the well-known benchmark set and compare our results with the most powerful metaheuristics from the literature, when the objective is to minimize the makespan. Since the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem is studied for the first time in the literature, we generate data set for the energy-related parameters. Then, we compare the results with a well-known NEH constructive heuristic. This comparison indicates how we improve the NEH solutions by proposed VNS algorithm. We conclude that the proposed algorithm is a robust algorithm for an FS-SDST problem for these two different objectives and we analyze the strengths and weaknesses of the proposed VNS algorithm.

ÖZETÇE

Üretim tesislerinde çoğu ürünler farklı makineler tarafından işlendikten sonra son halini alırlar. Ürünlerdeki çeşitliliğin artması ile birlikte işlerin makinelerdeki işlenme sırasını belirlemek için değişik atölye tiplerinde çizelgeleme problemi ön plana çıkmıştır. Bu çalışmada sıraya bağlı hazırlık süreleri de göz önüne alınarak akış tipi çizelgeleme (ATÇ) problemi incelenmiştir. Geleneksel ATÇ problemlerinde, işlerin veya makinelerin bir sonraki operasyon için hazırlanma süreleri ihmal edilmiştir veya iş sıralamasından bağımsız olarak ele alınmıştır. Ancak, çoğu uygulamalarda, makinenin bir sonraki işi işlemeden önce, makinenin temizlenmesi, makine parçalarının değişimi veya ayarlanması gibi hazırlıkların yapılması gerekmektedir. Bu tezde, sıraya bağlı hazırlık süreleri de göz önüne alınarak farklı amaç fonksiyonları olan iki ATÇ problemi çalışılmıştır. İlk problemde amaç tüm işlerin işlenme sırasını, maksimum tamamlanma zamanını en küçükleyecek şekilde belirlemektir. İkinci problemde sıraya bağlı hazırlık sürelerine ek olarak enerji tüketiminin de göz önüne alındığı akış tipi çizelgeleme (ATÇ) problemi çalışılmıştır. Bu problemde amaç tüm işlerin işlenme sırasını, toplam tamamlanma zamanı ve toplam harcanan enerjiyi küçükleyecek şeklide belirlemektir.

İncelenen ATÇ problemleri için değişken komşuluklu arama (DKA) algoritması önerilmiştir. Bu algoritmanın performansı, ilk problem için, yazında bulunan diğer güçlü sezgisel algoritmalarla karşılaştırılmıştır. İkinci problem yazında ilk defa çalışıldığı için DKA'nın performansı, NEH çözüm kurucu sezgisel algoritma ile kıyaslanmıştır. Bu karşılaştırma ile DKA'nın, NEH algoritmasıyla oluşturulan çözümü ne kadar geliştirdiği gösterilmiştir. Sonuç olarak, sıraya bağlı hazırlık süreli ATÇ problemi için önerilen DKA algoritmasının, farklı amaç fonksiyonlarıyla uyumu gözlenmiştir. Buna ek olarak DKA'nın karşılaştırılan diğer sezgisel yöntemlere göre güçlü ve zayıf yönleri analiz edilmiştir.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my advisor Prof. Ceyda Oğuz for her encouragement, support of my M.Sc. study and motivation. She provided valuable suggestions and without her supervision, this thesis would not have been possible.

I would like to thank the rest of my thesis committee Prof. Selçuk Karabatı and Assoc. Prof. F. Sibel Salman for their insightful comments and suggestions.

I would like to acknowledge financial support from TUBITAK during my M.Sc. study.

I would like to thank my fellow office mates Aysu, Ayşegül, Duygu, Işıl for their valuable and warm friendship; and special thanks to Özgür and Fatih for their willingness to help and their suggestions. I also would like to thank my flatmate Serena for her patience and love.

I would like to thank Mümin who encouraged me during my thesis study. With love and incredible patience, he was always there when I need.

A special thanks to my parents Ayten and Turan, and my brother Hakan for their endless support through my entire life and unconditional love. To my love and family, I dedicate this thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

APD	Average Percentage Deviation
FJSRA	Fictitious Job Setup Ranking Algorithm
FS-SDST	Flowshop Scheduling with Sequence Dependent Setup Times
GA	Genetic Algorithm
GRASP	Greedy Randomized Adaptive Search Procedure
HACO	Hybrid Ant Colony Optimization
IG	Iterated Greedy
IG_LS	Iterated Greedy Algorithm with Local Search Procedure
LB	Lower Bound
LP	Linear Programming
MA	Memetic Algorithm
MA_LS	Memetic Algorithm with Modified Local Search
MILP	Mixed Integer Linear Programming
NEH	Nawaz-Ed-Ham
NEH_RMB	Modified Nawaz-Ed-Ham Heuristic by Rios-Mercado and Bard
PACO	Ant Colony Optimization
PD	Percentage Deviation
RZ	Rajendran and Ziegler
SA	Simulated Annealing
SDST	Sequence Dependent Setup Times
TS	Tabu Search
TSP	Traveling Salesman Problem
VND	Variable Neighborhood Descent
VNS	Variable Neighborhood Search

LIST OF APPENDIX TABLES

Chapter 1

INTRODUCTION

In the manufacturing plants, many products pass through a series of operations to reach the end users. Using the same set of machines for processing different jobs may lead to idle times for the machines, or increase the need for inventory between stations or end products. To process varied products in the shop environment, manufacturers use scheduling models to plan production. With controlling the flow of the jobs, companies can shorten the delivery times, reduce in-process inventory or even decrease the energy consumption during the operations, which increases the resource utilization and decrease the cost of company.

In flowshop scheduling, each job is processed on a set of machines in series. The machine sequence is important for the jobs, since the output of one of the machine will be the input for the following machines. Each job follows the same order of machines, but the operations differ from type of the jobs. For instance, one of the machines can be a painting machine in the shop environment and according to the job, the machine paints the product in a different color. In regular flowshop scheduling problems, the preparation time of the machines for the following job in the sequence are considered as negligible or independent from the sequence of the jobs. However, in many applications, ignoring setup times may increase the operational costs. Hence, the researchers have studied flowshop scheduling with sequence dependent setup times (FS-SDST) for many years. Setup operations for the machine before processing the following job in the sequence may include cleaning, changing or adjusting the machine tools, positioning work in process material, setting the required jigs, heat treatment or changing the color for painting. Considering setup times leads to significant savings such as increase in production speed, faster changeovers, smoother flow and hence, reduces the operational costs. Figure 1.1 illustrates the schematic view of the FS-SDST problem.

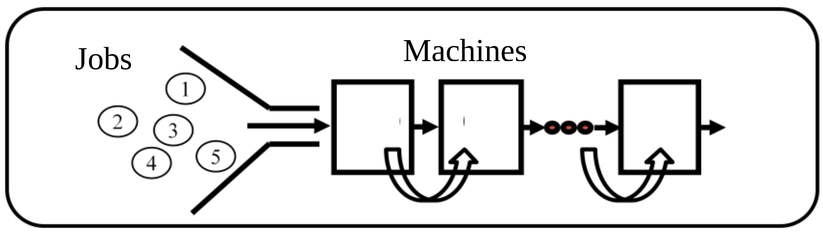

Figure 1.1 Flowshop scheduling problem with sequence dependent setup times

In recent years, sustainable production planning has also been attracted attention by the researchers for scheduling problems. The reason of the energy-saving practices is increase in energy consumption globally with rise in population. Government regulations and global competition force the manufacturers pay attention to the energy-aware scheduling. Adopting the sustainability practices has benefits for environmental and economic aspects.

In this thesis, we examine two FS-SDST problems with different objective functions. In the first problem, each job is characterized with a processing time on each machine and setup times according to the predecessor and successor jobs. We aim to schedule the jobs to be processed on all machines when the objective is to minimize the makespan. This problem is denoted in the literature by $F|s_{ijl}, prmu|C_{max}$ with the three-field notation (Pinedo, 2002). In this thesis, we also use this notation for our first problem. In the second problem, we study energy-aware FS–SDST problem. In manufacturing plants, it is observed that some of the machines are standing idle for a long time and the energy consumption of these machines may be significantly high. Hence, we use the strategy that when the machine is kept idle for a long time, instead of keeping the machine idle, turning off and on the machine can consume lower energy. Hence, for the second problem, we consider additional characteristics for the machines. All machines consume energy during processing a part, idle periods, turning off/on the machine and setup operations. While considering the energy consumption, we desire to minimize the completion time of the jobs. When the energy consumption is considered in the scheduling problem in addition to the traditional objectives such as makespan, these two objectives should be in contrast, which makes the problem more difficult. The strategy (that when the machine is kept idle for a long time, instead of keeping the machine idle, we can turn off and on the machine) implies to lower the total idle time on the machine. This aim is parallel with the makespan objective for the flowshop scheduling. Hence, to create a trade-off between objectives, we use the total completion time objective for the second problem. The motivational example of adding energy objective to the FS-SDST problem with the total completion time objective will be presented and discussed in Section 3.1.2 in detail. Hence, in our second problem, the aim is to schedule the jobs to be processed on all machines and decide about the status of the machine between scheduled jobs so as either to keep the machine idle or to turn off and turn on the machine, when the objective is to minimize both the total completion time and the total energy consumption. We denote this problem by $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ and in this thesis, we use this notation for the second problem.

We propose a robust Variable Neighborhood Search (VNS) algorithm for these two FS-SDST problems. The proposed VNS algorithm uses two neighborhood structures and a local search procedure systematically. We examine the performance of the VNS algorithm for the $F|s_{ijl}, prmu|C_{max}$ problem by using the well-known benchmark set and compare our results with the most powerful metaheuristics from the literature. Since the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem is studied for the first time in the literature, we present a mathematical model for this problem. We generate a set of instances for the energy-related parameters. We solve small-sized problems via the mixed integer linear programming (MILP) model presented in Section 3.2.2 and compare the solutions obtained from the proposed VNS algorithm with the optimal solutions. For large-sized problems, we compare the VNS results with the well-known NEH constructive heuristic results. This comparison indicates how we improve the NEH solutions by proposed VNS algorithm.

The chapters of the thesis are structured as follows. In Chapter 2, we review the studies related with the $F|s_{ijl}, prmu|C_{max}$ and $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problems. Additionally, we survey the most relevant studies with these two problems which are solved by VNS algorithm. In Chapter 3, we give the definition of two FS-SDST problems. We present the mixed integer linear programming (MILP) models for the $F|s_{ijl}, prmu|C_{max}$ problem which is proposed by Stafford and Tseng (2001) and modified version of this mathematical model to the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem. In Chapter 4, we present the implementation of the VNS algorithm. In Chapter 5, we give the computational results and the analysis of these results for two FS-SDST problems. Finally, in Chapter 6, we give conclusions and the important remarks for future research.

Chapter 2

LITERATURE SURVEY

In this chapter, we survey the studies related to the flowshop scheduling problem with sequence dependent setup times when different objectives are considered as in our study. Moreover, we investigate the VNS algorithm and the studies in which the authors use VNS algorithm as their solution methodology.

2.1 Flowshop Scheduling Problem with Sequence Dependent Setup Times

The flowshop scheduling problem with sequence dependent setup times (FS-SDST) is a well-known problem in the literature and several studies exist with different objectives. In this study, we consider two FS–SDST problems when the objectives are to minimize makespan, referred to as $F|s_{ijl}, prmu|C_{max}$, and to minimize both the total completion time and the energy consumption, referred to as $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$. For the problem $F|s_{ijl}, prmu|C_{max}$, there are studies regarding both exact algorithm methods and heuristic algorithms. However, to the best of our knowledge, the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem is studied for the first time in the literature. Hence, we review most related articles with energy-aware scheduling.

2.1.1 $F|s_{ijl}, prmu|C_{max}$ Problem

The article written by Srikar and Ghosh (1986) is one of the fundamental studies on the FS-SDST problem when the objective is to minimize the makespan. In this article, they propose a mixed integer linear programming (MILP) model for the $F|s_{ijl}$, $prmu|C_{max}$ problem with unique binary variable for sequencing. They emphasize that minimizing the makespan in a single machine scheduling with sequence dependent setup times (SDST) implies to minimize the setup times, which resembles the traveling salesman problem (TSP). Different from the traditional TSP-based binary variable for sequencing which takes the value of 1 if job *j* is scheduled immediately before job *l*, they define the binary variable for sequencing which takes the value of 1 if job *j* is scheduled anytime before job *l*. Proposed binary variable decreases the number of variables. They also solved some small-sized problems up to six jobs and six machines with mixed integer linear programming model. Stafford and Tseng (1990) report some corrections on the MILP model for the FS-SDST problem developed by Srikar and Ghosh (1986). The corrections are made on the calculation of mean flow time and the order of index of the sequence dependent setup time parameter in one of the constraint sets. They also solve the problem up to seven jobs to five machines with integer programming. Additionally, they propose three more MILP models with the decision variable defined by Srikar and Ghosh (1986) for the flowshop problems with different characteristics. Tseng and Stafford (2001) also propose two new MILP models for the FS-SDST problem. First model is based on the assignment problem where the binary variable takes the value of 1 if job *j* is scheduled in position *k*. In the constraint sets, they use equality constraints with using two decision variables: idle time on machines and idle time on jobs. In the second model, they use the same binary variable for sequencing the jobs, but they change their constraint set by using inequalities such as the model proposed by Srikar and Ghosh (1986). They solve different size of problems up to seven jobs to seven machines with integer programming and they compare the computation times with the model proposed by Srikar and Ghosh (1986). They claim that their two MILP models are solved optimally in less CPU time than the model proposed by Srikar and Ghosh (1986). In addition to the integer programming model, branch and bound technique is also used for solving the FS-SDST problem. Rios-Mercado and Bard (1999a) propose branch and bound algorithm for the FS-SDST problem considering makespan objective.

They implement lower and upper bound procedures, and dominance rules. Same authors also study branch and cut algorithm (1998a, 2003). In these articles, they consider two MILP models. One of them is TSP based mathematical model and second one is the model proposed by Srikar and Ghosh (1986). They relax the integrality constraints of the models and generate powerful valid inequalities for the polyhedron, which leads to obtaining better results compared to branch and bound technique.

The flowshop scheduling problem with sequence dependent setup times is shown as strongly NP-hard by Gupta and Darrow (1986), when the objective function is makespan. They show that even when one of the machines has sequence dependent setup times in the two-machine flowshop problem, it is still strongly NP-hard. Since the FS-SDST problem is solvable up to ten jobs and few machines optimally, the authors have proposed some heuristic methods for the $F|s_{ijl}$, $prmu|C_{max}$ problem. Ruiz et al. (2005) propose a genetic algorithm and a memetic algorithm in which they improve the genetic algorithm with local search. To compare the quality of the solutions obtained from the proposed algorithms, they adapt several heuristic methods which are proposed for regular flowshop problem. Additionally, they compare the results with alternative methods which have already been proposed for the FS-SDST problem when the objective is makespan. To compare the results in a fair platform, Ruiz et al. code each algorithm in the same computer and use same benchmark sets. For instance sets, they use the data set generated by Taillard (1993) for the regular flowshop problems. In this set, the sizes of instances are combination of 20 jobs to 500 jobs and 5 machines to 20 machines. Ruiz et al. generated four groups of sequence dependent setup times (SDST) values for each instances. This article is a comprehensive article, since in the experimental evaluation, twelve heuristic algorithms are used. The powerful heuristics for the regular flowshop problem which are modified to the FS-SDST problem are the genetic algorithm of Reeves (1995), simulated annealing of Osman and Potts (1989), iterated local search procedure of Stützle (1998), tabu search of Widmer and Herzt (1989) and genetic algorithm by Aldowaisan and Allahverdi (2003). Ruiz et al. modify the calculation of the makespan value of these algorithms by adding the sequence dependent setup times. Moreover, some of the adapted heuristics originally initialize their algorithm with a constructive heuristic, Nawaz-Ed-Ham (1983) (NEH) algorithm, which uses a local search procedure based on the insertion neighborhood. Ruiz et al. (2005) replace this NEH heuristic with the NEH_RMB heuristic modified by Rios-Mercado and Bard (1998b) for the FS-SDST problem. The other metaheuristics, which have been already proposed for the FS-SDST, are NEH_RMB and greedy randomized adaptive search procedure (GRASP) of Rios-Mercado and Bard (1998b), the Total and Setup heuristics of Simons (1992), the TSP based heuristic of Rios-Mercado and Bard (1999b) and the saving index algorithm, which is based on the selection of job that has maximum time savings, of Das et al. (1995). Lastly, Ruiz et al. (2005) propose a simple heuristic which generates random solutions and takes the best one. In computational experiments, Ruiz et al. (2005) observe that the proposed memetic algorithm dominates all other 13 algorithms. The reasons of the power of this memetic algorithm are the new crossover operations which are created for the $F|s_{ijl}, prmu|C_{max}$ problem specifically and hybridization of the proposed genetic algorithm with the local search procedure based on node insertion neighborhood.

Gajpal et al. (2006) propose an ant colony algorithm for the FS-SDST problem to minimize the makespan. They improve the ant colony algorithm, which was proposed for the regular flowshop problem by Rajendran and Ziegler (2004), by changing the initialization and local search procedures. In the ant colony algorithm, initially one solution is constructed and it is improved by local search. In the next step, the pheromone trail is updated and these procedures continue until the stopping condition is met. Gajpal et al. (2006) compare their results with the GRASP algorithm of Rios-Mercado and Bard (1998b) and saving index algorithm of Das et al. (1995). They generate the instance set randomly to compare the results of these algorithms. They observe that the proposed ant colony algorithm gives better results compared to other heuristics. However, since they use a different benchmark set, the comparison of this algorithm with the other alternative heuristics in the literature is not possible.

Ruiz and Stützle (2008) presented two new iterated greedy (IG) heuristics for the FS-SDST problem with two different objectives: makespan and weighted tardiness. The same author has already proposed the iterated greedy algorithm for the regular flowshop problem and the algorithm works efficiently for this flowshop problem (2007). Hence, they extended the algorithm to the FS-SDST problem. In the article, one of the proposed algorithms was based on simple IG algorithm; the other one was hybridization of the IG algorithm with a descent local search based on insertion neighborhood. IG algorithm starts with an initial solution and it has destruction and construction phases basically. In destruction phase, some of the components of the solutions are removed and each removed components are added to the partial solution one by one to obtain the best permutation. In the local search procedure, each component (job) in the sequence, which is obtained after construction phase, is removed from the sequence and inserted into another position which gives a lower makespan value. Ruiz and Stützle (2008) compared their computational results with the alternative heuristics. Since Ruiz et al. (2005) conducted a comprehensive comparison between 14 algorithms; Ruiz and Stützle (2008) choose the two best algorithms among them to compare their results: genetic and memetic algorithm of Ruiz et al. (2005). Ruiz and Stützle (2008) improve the memetic algorithm by the local search procedure which they propose for the IG algorithm. Additionally, they extend the ant colony algorithm, which is proposed by Rajendran and Ziegler (2004) for the regular flowshop, to the FS-SDST problem. Ruiz and Stützle (2008) test their algorithms and alternative methods with the four instance sets generated by Ruiz et al. (2005). The results indicate that the IG algorithm proposed with the descent local search procedure outperforms the other alternative heuristics. Moreover, they emphasize that for the $F|s_{ijl}, prmu|C_{max}$ problem the local search algorithms, especially based on node insertion neighborhood, have an important role to improve the quality of the solutions.

After the article of Ruiz and Stützle (2008), some authors have published studies on the FS-SDST problem when the objective is makespan. However, since they generate their own instance sets to compare their results with other alternative heuristics, the proposed iterated greedy heuristic with descent local search procedure of Ruiz and Stützle (2008) is taken as the state-of-the-art method for the instance set of Ruiz et al. (2005). On the other hand, some of the studies on the FS-SDST problem after aim to develop the existed type of heuristics, not to improve the solutions for the well-known benchmark set. Mirabi (2011) develops a new ant colony optimization technique for the $F|s_{ijl}, prmu|C_{max}$ problem, since Ruiz and Stützle (2008) indicates in their study that the memetic algorithm which is proposed by Ruiz et al. (2005) outperforms the ant colony algorithm of Rajendran and Ziegler (2004). The proposed ant colony algorithm is compared with the genetic and memetic algorithms of Ruiz et al. (2005) and a tabu search algorithm which is proposed by Eksioglu et al. (2008). In the computational experiments, the author generates instance sets different from the sets generated by Ruiz et al. (2005) and concludes that the proposed ant colony algorithm improves the results. Vanchipura and Sridharan (2013) propose two constructive heuristic, since only one strong constructive heuristic is proposed for the $F|s_{ijl}$, $prmu|C_{max}$ problem in the literature so far, which is the NEH_RMB heuristic modified by Rios-Mercado and Bard (1998b) to the FS-SDST problem. Proposed two constructive heuristics are setup ranking algorithm (SRA) based on only setup times and fictitious job setup ranking algorithm (FJSRA) which is related to the job pairs that has minimum setup times. Vanchipura and Sridharan (2013) generate the instance sets to compare these three heuristics. They observe that FJSRA algorithm outperforms the NEH_RMB heuristic (Rios-Mercado and Bard, 1998b) on the large instance sets. When the number of jobs increases, the importance of the setup time also increases. Since the FJSRA heuristic is based on the setup times, it gives better results compared to NEH_RMB (Rios-Mercado and Bard, 1998b).

In the literature, the FS-SDST problem is studied mostly with a simple objective function which is makespan, since the $F|s_{ijl}, prmu|C_{max}$ problem is a difficult problem. To the best of our knowledge, the FS-SDST problem when the objective is to minimize the total completion time has not been studied before, but there are several related works to this problem. On the other hand, in this thesis, an energy-related objective function is added to the FS-SDST problem with total completion time objective, denoted by $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$, for the first time in the literature. In the next section, we review the most related works with the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem.

2.1.2 $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ Problem

The FS-SDST problem when the objective is to minimize the total completion time is formulated similar to the FS-SDST problem when the objective is to minimize makespan. In a mixed integer linear programming model, the constraint sets of the FS-SDST problem with total completion time objective are same, when the completion time is defined as a decision variable. The difference is only seen in the objective function. Hence, Srikar and Ghosh (1986), Stafford and Tseng (1990) and Tseng and Stafford (2001) model the FS-SDST problem considering both makespan and total completion time as the objective function. In these articles, the objective is defined as mean flow instead of total completion time. Since the release dates are assumed to be zero in these studies, the formulation of mean flow and total completion time are the same.

In the literature, the FS-SDST problem with the total completion time objective has not been studied before, to the best of our knowledge. Allahverdi et al. (2008) has a survey of the scheduling problems with setup times. In this article, it is observed that even though there are many studies on the FS-SDST problem with makespan objective, there is not article yet with the total completion time objective, except the one with no-wait constraint in a flowshop. Allahverdi and Aldowaisan (2001) study a two-machine no-wait FS-SDST problem when the objective is to minimize the total completion time. They obtain the optimal solutions for the problem under some assumptions and propose dominance rules. They also develop a simple heuristic which uses the insertion technique repeatedly. After the literature survey of Allahverdi et al. (2008) on scheduling problems with setup time, Salmasi et al. (2010) publish an article for the FS-SDST problem when the objective is to minimize the total flow time without any release dates (so equivalent to the total completion time). However, they consider that the setup times depend on the group of jobs, not on individual jobs. They develop a mathematical programming model for this problem. They also propose two heuristic algorithms, a tabu search (TS) and hybrid ant colony optimization (HACO) algorithms. Since there were no studies published on this problem, they compare the results of these heuristics with each other and a lower bound developed by branch and price technique. They observe that the proposed HACO algorithm works better than the TS algorithm.

In recent years, researchers have tended to study energy aware scheduling problems. They approach the green manufacturing in three main categories: machine level, product level and manufacturing system level (2012, 2013). At the machine level, the researchers focus on reducing energy consumption in the system by designing more energy-efficient machines. Similarly, at the product level, the researchers study designing the products to minimize embodied product energy. However, the energy consumption can also be decreased by managerial decisions without redesigning the machine or the product, which is studied at the manufacturing system level. In the literature, there are several studies on energy aware scheduling problems at the manufacturing system level with different energy and scheduling objectives. Nolde and Morari (2010) study minimizing peak and off-peak energy consumption on machines in a steel plant. Similarly, Bruzzone et al. (2012) propose a mathematical model for flexible flowshop when the objective is to minimize peak of power. In this study, they also consider multiple objectives in terms of minimization of tardiness and makespan. Fang et al. (2013) study the flowshop scheduling problem with peak power consumption constraint. They consider two objectives for the flowshop problem, which are to minimize the makespan and the peak power consumption. They handle these multiple objectives by fixing the upper bound for the peak power and use the makespan objective in the mathematical model.

Different from the objective of minimization the peak power, Yildirim and Mouzon (2008) study a single machine scheduling problem to minimize the total tardiness and total energy consumption considering the power consumption during the idle time. Since some of the machines in the manufacturing plants, which are kept idle between two consecutive jobs for a long time, consume significant amount of energy; they propose to turn off the machine in that period and turn on when the following job is ready to be processed on that machine. They aim to minimize the total energy consumption by deciding to keep the machine idle or to turn off and on the machine in the idle period. In that paper, they propose a mathematical model for this problem and a greedy random adaptive search procedure (GRASP) to obtain set of solutions. The same authors, Yildirim and Mouzon (2012) consider the same machine environment and energy objective with a different scheduling objective, which is to minimize the total completion time. They propose a mathematical model to schedule the jobs on a single machine to minimize the total completion time and total energy consumption by deciding whether to keep the machine idle or to turn off and on the machine when the idle time occurs on that machine between two consecutive jobs. They propose a multi-objective genetic algorithm for this problem. To increase the efficiency of the proposed genetic algorithm, they improve a dominance rule and a heuristic to obtain the Pareto front. In both articles, the authors aim to find both the sequence of the jobs to be processed on the machine and the starting times for each job.

Minimization of production rate is also used as energy objective in the literature. Gutowski et al. (2006) analyze the energy consumption of the manufacturing processes such as milling machine, considering different production rates. Zanoni et al. (2014) also study minimizing the energy consumption in two stage production system by controlling the production rate. They analyze the production system as different cases depending on machine power strategies with continuous and interrupted batch production. Different from Yildirim and Mouzon (2008, 2012), Zanoni et al. (2014) used the machine power as a parameter within different cases rather than a decision variable. Since there are two production stages, they create the cases for the idle time on the machine between two consecutive jobs as such: keep two machines idle between two consecutive processes, keep the first machine idle but turn on/off the second machine, turn on/off the first machine but keep the second machine idle and lastly, turn on/off both two machines. For given cases, they aim to minimize both the cost of storing the product and energy cost during the production of a product (related to production rate) and the idle state of the machine.

Additionally, Mashaei and Lennartson (2013) study a pallet-constrained flowshop problem when the objective is to minimize the energy consumption while providing the desired throughput for the plant. The used strategy to reduce the energy is similar with the strategy which is used by Yildirim and Mouzon (2008, 2012). Mashaei and Lennartson (2013) control the idle machines by turning them off and on in a closed-loop flow shop plant. In the article, they propose a nonlinear mathematical model and develop a simple heuristic to solve the closed-loop flowshop problem.

2.2 Variable Neighborhood Search

Mladenovic and Hansen (1997) design a new metaheuristic approach called variable neighborhood search (VNS) for combinatorial optimization problems. Developed VNS heuristic contains the local search procedure that enforces to find the local optimum for one neighborhood structure, which leads to intensification and increase the quality of the solution in that neighborhood structure. In the existing heuristic algorithms such as simulated annealing, tabu search, genetic algorithm etc., some methods are used in the algorithm to avoid being stuck in the local optimum in the search space, which makes the algorithms more complicated. However, the motivation for the design of the VNS algorithm is to escape from the local optimum by changing the neighborhood structure systematically.

This new metaheuristic has been applied in many different areas such as industrial applications, location problems, data mining, scheduling. Hansen et al (2009) review the algorithm characteristics and the application areas of the VNS. Several authors design a VNS algorithm or a hybrid algorithm with VNS to solve the flowshop scheduling problem with various characteristics and objective. Costa et al. (2011) propose a VNS algorithm for a regular flowshop problem when the objective is to minimize makespan. Moreover, Tasgetiren et al. (2007) and Zobolas et al. (2009) use VNS algorithm to construct a hybrid algorithm with other metaheuristics, particle swarm optimization and genetic algorithms respectively, for the regular flowshop problem. For the FS-SDST problem, the VNS algorithm has not been proposed, yet. However, in the literature, there are some related works with VNS application to the FS-SDST.

Naderi et al. (2008) propose a variable neighborhood search for a hybrid flexible flowshop with SDST where the objective is to minimize the total completion time. They use three different insertion neighborhood structures and in each neighborhood they use variable neighborhood descent (VND) framework. In the first neighborhood structure, each job is removed from the current sequence one by one and inserted into another position which gives the lower objective function value among all possible positions. In the second neighborhood structure, they choose all two pairs of jobs from the sequence and insert other positions randomly, since the insertion of all combination of two jobs into all possible positions takes a long time. In the last structure, they remove randomly three different jobs from the sequence and insert into three other positions. For all neighborhood structures, search continues until there is no improvements. Naderi et al. (2008) evaluate the quality of the proposed VNS algorithm with the other alternative metaheuristics and some dispatching rules. They conclude that the proposed algorithm gives better results compared to the alternative algorithms.

Vanchipura et al. (2014) study the FS-SDST problem where the objective is to minimize makespan, $F|s_{ijl}$, $prmu|C_{max}$, as in our study. They improve two existing constructive heuristics with the variable neighborhood descent (VND) algorithm. These two constructive heuristics are NEH_RMB heuristic which is modified by Rios-Mercado and Bard (1998b) and fictitious job setup ranking algorithm (FJSRA) by Vanchipura and Sridharan (2013). In this study, Vanchipura et al. (2014) use these constructive heuristics to obtain the initial solution and the solutions are improved with node insertion neighborhood search. These two algorithms are tested with the 960 instance sets which are generated by Vanchipura et al. (2014). The results show that the initial solutions affect the quality of the solutions after applying the proposed VND algorithm. They also analyze the relative performance improvement of the solutions which are obtained by the constructive heuristics and the improved version with VND, separately for NEH_RMB (Rios-Mercado and Bard, 1998b) and FJSRA (Vanchipura and Sridharan, 2013).

Chapter 3

FLOWSHOP SCHEDULING PROBLEM WITH SEQUENCE DEPENDENT SETUP TIMES

In this chapter, we study two flowshop problems where separate sequence dependent setup times are explicitly considered. In the following sections, we define these problems in detail with their respective assumptions and present mathematical models for both problems.

3.1 Problem Definition

In the flowshop environment, there are m machines in series and n jobs. A job j, j=1, ... , n, has to be processed on each machine i, i=1, ... , m, and each job has to follow the same route on the machines. In other words, each job should be processed by all machines from 1 up to m in this order. The aim of the flowshop scheduling problem is to schedule these n jobs to be processed on m machines by minimizing a given objective function. In our study, the order of jobs which are processed on a machine is same for every machine, which is called a permutation flowshop problem in the literature. Processing an operation of job j on machine i requires a certain time which is denoted by p_{ij}, i=1, ... , m and j=1, ... , n. Moreover, we consider a sequence dependent setup time between two adjacent jobs in the sequence and the required time for this setup is denoted by s_{ijl} when job l is processed immediately after job j, on machine i, i=1, ... , m, j=1, ... , n, l=1, ... , n and $l \neq j$. Figure 3.1 shows the Gantt chart for a FS-SDST problem with 4-jobs and 3-machines. In the figure, M1, M2 and M3 refer the machines; J1, J2, J3 and J4 refer the jobs. Empty boxes show the processing times p_{ij} and boxes with diagonal lines show the setup times s_{ijl}, which are illustrated on the chart. For the given problem in Figure 3.1, the permutation of jobs is J3, J2, J4 and J1.

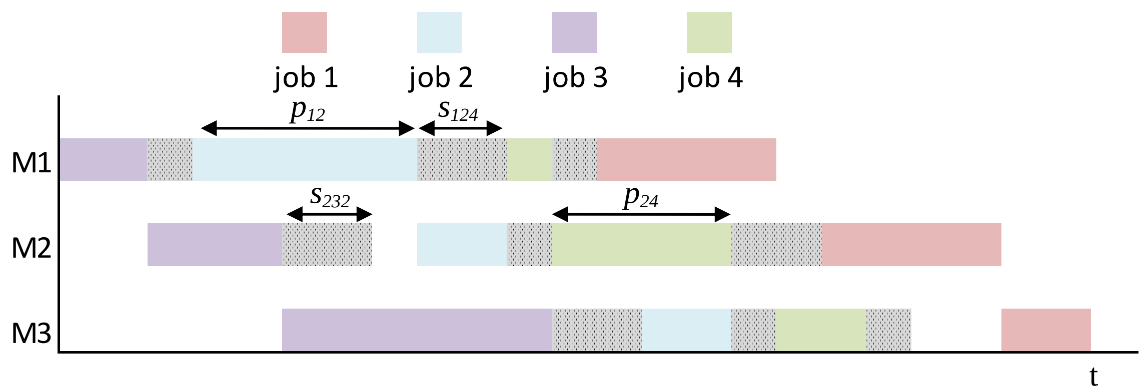

Figure 3.1 Gantt chart for schedule of 4-jobs on 3-machines

For other characteristics of the FS-SDST problem, we use the following assumptions.

- All jobs and machines are available at time zero.
- Processing times and setup times are deterministic and known in advance.
- A machine can process only one job at a time; and a job is processed only on one machine at a time.
- There are no due dates for the jobs.
- Interrupting a job while processing is not allowed.
- There is no precedence relation among the jobs.

The general definition and assumptions presented for the FS-SDST problem are valid for both of the problems which are studied in this thesis. However, the objective functions of $F|s_{ijl}, prmu|C_{max}$ and $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$, and specific characteristics of these two problems are different and the details are explained in the following subsections.

3.1.1 The $F|s_{ijl}, prmu|C_{max}$ Problem

In the $F|s_{ijl}, prmu|C_{max}$ problem, each job has a processing time p_{ij} on each machine and setup times s_{ijl} according to the predecessor and successor jobs, i=1, ... , m, j=1, ... , n, l=1, ... , n and $l \neq j$. The completion time C_{ik} shows the time when the operation of the job in position k is finished on machine i from time zero, i=1, ... , m, k=1, ... , n. In the $F|s_{ijl}$, $prmu|C_{max}$ problem, we aim to schedule the jobs to be processed on all machines when the objective is to minimize the maximum completion time C_{mn}. Namely, the completion time of the job which is scheduled in the last position n in the order, on the last machine m gives the value of the maximum completion time, which is also called makespan in the literature.

3.1.2 The $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ Problem

In the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, we consider the energy consumption of the operations while scheduling these operations in a flowshop environment. As a result, we consider additional characteristics for the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem. As a characteristic of a machine, it consumes energy while processing a part and during the time period when it is kept idle (not processing a part). Turning off and on a machine also consumes a fixed amount of energy. Additionally, while turning a machine off and on, a certain amount of time is required. When a machine is kept idle for a long time, instead of keeping the machine idle, turning off and on the machine can consume lower energy; but the amount of idle time on that machine should be sufficient for the time required for turning off/on the machine. This decision mechanism is also used in the articles of Mouozon and Yildirim (2008) and Yildirim and Mouzon (2012) to reduce the total energy consumption on a single machine.

In the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, idle time on a machine denotes the period of time between the end of processing of a job on that machine and the start of preparing the machine for the following job (beginning of the sequence dependent setup time). It is assumed that the requirement for a setup operation on a machine, which is whether this machine should be switched off or on during the setup, is not considered. The decision for the status of the machine is independent from the setup operation. For setup operations such as cleaning or changing the equipment, machines may also consume energy. Hence, we consider setup energy which differs from machine to machine and we assume that the consumed energy will be in direct proportion to setup time.

In the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, we aim to schedule the jobs to be processed on all machines and decide about the status of the machine between scheduled jobs so as either to keep the machine idle or to turn off and turn on the machine. The objective of the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem is to minimize both the total completion time and the total energy consumption where the total energy consumption can be due to processing a job, keeping the machine idle, turning off/on the machine or setup.

To observe the impact of the new objective, which is minimization of the energy consumption in addition to the total completion time, three different scenarios are discussed based on an instance with three machines and seven jobs. The data of the instance is given in Table 3.1 and three scenarios are given in Figures 3.2, 3.3 and 3.4. We generate the processing time uniformly in [1, 99], setup time in [1, 49], idle cost in [15,25], setup cost in [35,45], on/off cost in [350,450] and required time for turning off/on the machine in [1,20]. We explain the selection of the data ranges in Section 5.1.2 in detail. We use the weighted sum parameters w_1 and w_2, which will be explained in Section 3.2.2, to combine the objectives the total production cost and the total energy cost. In the given instance, we set w_1 and w_2 to 0.5, when we consider the total energy cost. In Figures 3.2, 3.3 and 3.4, each job is shown with a different color and setup times are shown with gray. Moreover, length of the colored block shows the amount of processing time.

Table 3.1 Data of the instance

p_{ij}	J1	J2	J3	J4	J5	J6	J7
M1	12	91	17	77	26	85	55
M2	72	78	51	17	48	50	23
M3	53	93	36	85	25	97	72

s_{1jl}	J1	J2	J3	J4	J5	J6	J7
J1	0	39	37	31	18	24	30
J2	21	0	13	17	10	29	10
J3	12	26	0	45	49	11	28
J4	21	4	40	0	34	33	15
J5	9	15	1	12	0	29	33
J6	25	11	21	32	18	0	28
J7	5	10	12	20	26	13	0

s_{2jl}	J1	J2	J3	J4	J5	J6	J7
J1	0	7	19	39	34	33	23
J2	19	0	29	40	33	27	38
J3	30	40	0	23	28	12	24
J4	16	16	7	0	27	24	9
J5	5	19	20	29	0	5	16
J6	39	21	24	17	20	0	6
J7	13	12	23	39	48	32	0

s_{3jl}	J1	J2	J3	J4	J5	J6	J7
J1	0	20	30	37	10	11	24
J2	29	0	36	43	49	4	20
J3	8	5	0	12	32	4	35
J4	37	23	27	0	8	22	18
J5	3	34	20	22	0	24	10
J6	33	13	38	37	42	0	10
J7	30	30	9	39	31	43	0

Energy Cost	Idle Cost	Setup Cost	ON/OFF Cost	ON/OFF Time
M1	20	45	412	13
M2	17	41	433	17
M3	21	45	357	15

Figure 3.2 Gantt chart for the FS-SDST while minimizing the total production cost

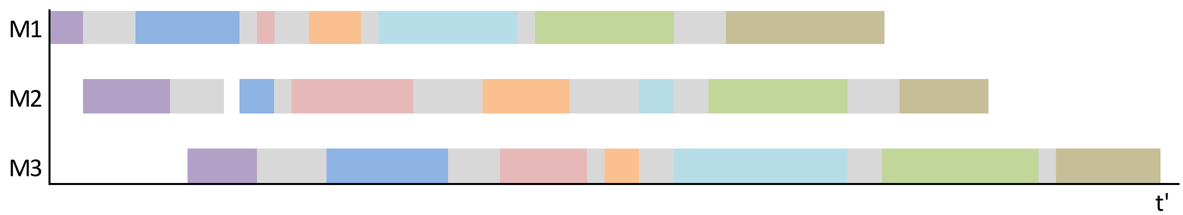

Figure 3.3 Gantt chart for the FS-SDST while minimizing the total production cost and the total energy cost, without setup cost

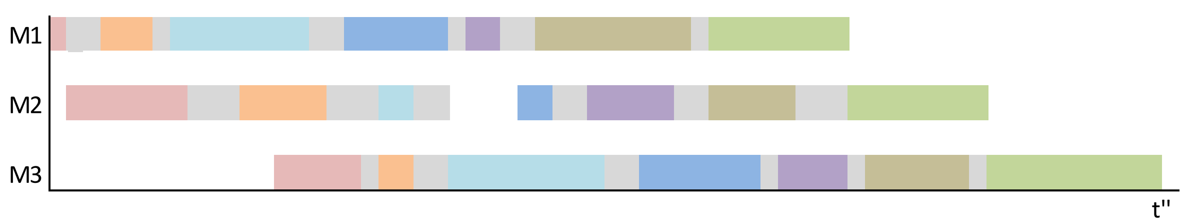

Figure 3.4 Gantt chart for the FS-SDST while minimizing the total production cost and the total energy cost

In the first scenario seen in Figure 3.2, the objective is only to minimize the total production cost. On the other hand, in the second scenario, energy cost is added to the objective function. Energy cost due to setups on machines is assumed as zero or negligible. In the first scenario seen in Figure 3.2, the optimal sequence is [1, 5, 7, 3, 4, 2, 6]; however, in the second scenario seen in Figure 3.3, the optimal sequence is changed to [3, 7, 1, 5, 4, 2, 6]. As a result, adding the energy cost to the objective function affects the optimal sequence explicitly. We observe that even though the value of the total production cost is increased, a lower value of the objective function (total cost) is obtained due to the energy cost. It is observed that, the objective function of this scenario forces idle times to be reduced in the schedule, since idle times increase the energy cost. However, this results in exploiting the sequence dependent setup times, in the sense that longer sequence dependent setup times may be used in order to decrease the idle time, which may increase the completion time of some jobs. Hence there is a trade-off between the cost of idle time (in turn the cost of energy) and the cost of completion time.

In the third scenario, it is assumed that during the setup, machines consume energy in direct proportion with the amount of setup times. Hence, the setup cost is added to the calculation of the total energy cost. In the third scenario seen in Figure 3.4, the optimal sequence is changed to [1, 5, 4, 7, 3, 6, 2]. When this scenario is compared with the second scenario, it is observed that the total production cost and the energy cost due to idle period or turning off/on the machine increase but the energy cost due to setup decreases. On the other hand, since the setup cost decreases, it affects the total objective function value (decreases the total energy cost). As a result, it leads to a different optimal schedule. The production cost, energy cost due to idle/on-off the machine and setup, and the total cost values of the three scenarios are given in Table 3.2.

Table 3.2 Production cost and energy cost values for three scenarios

Scenarios	w_1 (%)	w_2 (%)	Production Cost	Energy Cost		Total Cost
				Idle/On-off	Setup	
Scenario 1	1.0	0.0	12845	1384	15104	12845
Scenario 2	0.5	0.5	12895	170	15763	6532.5
Scenario 3	0.5	0.5	13200	433	12223	12928

Consequently, when minimization of energy consumption is concerned in addition to minimize the total completion time, the optimal sequence changes. This is the motivation to develop a solution procedure for the large-sized instances of the problem. This can be a useful application for the manufacturing companies who desire to decrease not only the total completion time of the jobs but also the energy consumption.

3.2 MILP Formulation for the FS-SDST

In the following subsections, the mixed integer linear programming models for two problems, which are studied in this thesis, are presented.

3.2.1 MILP Formulation for the $F|s_{ijl}, prmu|C_{max}$ Problem

The FS-SDST problem is a well-known problem in the literature. Since the makespan objective is the simplest objective function to study the FS-SDST problem, there are several mixed integer linear programming (MILP) models proposed for the $F|s_{ijl}$, $prmu|C_{max}$ problem, which we reviewed in Section 2.1.1. In this section, we present a MILP model for the $F|s_{ijl}, prmu|C_{max}$ problem which is proposed by Stafford and Tseng (2001). Stafford and Tseng (2001) propose two MILP models which are deriven from assignment problem where the binary variable takes the value of 1 if job j is scheduled in position k, j=1, ... , n and k=1, ... , n. The model presented in this study for the FS-SDST problem is called TS2 model by Stafford and Tseng (2001). As a small modification of the existing model, the initial setup times are assumed to be zero in this thesis, thus initial setup time parameter is eliminated from the model.

The system parameters, decision variables and the mathematical model are presented below.

Sets and parameters:

n number of jobs to be processed,

m number of machines,

p_{ij} processing time of job j on machine i, i=1, ... , m, j=1, ... , n,

s_{ijl} sequence dependent setup time for job l on machine i when job j precedes job l immediately in the sequence, i=1, ... , m, j=1, ... , n, l=1, ... , n and $l \neq j$.

Decision variables:

$$z_{jk} = \begin{cases} 1, \ \ if\ job\ j\ is\ in\ position\ k\ in\ the\ sequence, j = 1, \dots, n, k = 1, \dots, n\ . \\ \ \ 0, otherwise, \end{cases}$$

$$q_{jkl} = \begin{cases} 1, \text{if job } j \text{ is in position } k \text{ in the sequence and is immediately followed by job } l \\ \quad j = 1, \dots, n, k = 1, \dots, n, l = 1, \dots, n \text{ and } l \neq j. \\ 0, \text{otherwise,} \end{cases}$$

C_{ik} completion time of the job in position k of the sequence on machine i, $i = 1, \dots, m, k = 1, \dots, n$.

Objective function:

$$Minimize: C_{MAX} = C_{mn} \tag{3.1}$$

Constraint sets:

$$\sum_{k=1}^{n} z_{jk} = 1 \quad (j = 1, \dots, n) \tag{3.2}$$

$$\sum_{j=1}^{n} z_{jk} = 1 \quad (k = 1, \dots, n) \tag{3.3}$$

$$z_{jk} = \sum_{l=1}^{n} q_{jkl} \qquad (j = 1, \dots, n, k = 1, \dots, n) \tag{3.4}$$

$$z_{jk} = \sum_{l=1}^{n} q_{l,(k-1),j} \quad (j = 1, \dots, n, k = 2, \dots, n) \tag{3.5}$$

$$z_{j1} = \sum_{l=1}^{n} q_{lnj} \qquad (\text{i} = 1, \dots, n) \tag{3.6}$$

$$C_{ik} + \sum_{j=1}^{n} \sum_{l=1}^{n} s_{ijl}\, q_{jkl} + \sum_{i=1}^{n} p_{ij} z_{j,k+1} \leq C_{i,k+1}$$
$$(i = 1, \dots, m,\ k = 1, \dots, n-1) \tag{3.7}$$

$$C_{ik} + \sum_{i=1}^{n} p_{i+1,j} z_{jk} \leq C_{i+1,k} \quad (i = 1, \dots, m-1,\ k = 1, \dots, n) \tag{3.8}$$

$$\sum_{i=1}^{n} p_{ij} z_{j1} \leq C_{i1} \quad (i = 1, \dots, m) \tag{3.9}$$

Decision variables z_{jk} is defined based on an assignment problem; job j is assigned to the position k, j=1, ... , n and k=1, ... , n. Additionally, q_{jkl} is defined for indicating two adjacent jobs, which is used with the sequence dependent setup time in constraint sets, j=1, ... , n, k=1, ... , n, l=1, ... , n and $l \neq j$.

Constraint set (3.1) demonstrates that the objective function which is to minimize the maximum completion time, C_{mn}. Constraint sets (3.2) and (3.3) ensure that each job is assigned to only one position; conversely, one position is assigned to only one job, respectively. Constraint sets (3.4), (3.5) and (3.6) link the two decision variables z_{jk} and q_{jkl}, j=1, ... , n, k=1, ... , n, l=1, ... , n and $l \neq j$. When job j is processed in position k; only one job (job l) follows job j, j=1, ... , n, k=1, ... , n, l=1, ... , n and $l \neq j$. On the other hand, when job j is in position k, there is only one job, job l, in position k-1, j=1, ... , n, k=2, ... , n, l=1, ... , n and $l \neq j$. Constraint set (3.6) demonstrates the special case of the equation set (3.5) when a job is in the first position. In this situation, the job in the previous position represents the job in position n (last position) in the sequence. Constraint set (3.7) guarantees that the completion time of the job in position k+1 is equal or greater than the summation of the completion time of the job in position k, the setup time between two adjacent jobs in positions k and k+1 and the processing time of the job in position k+1, k=1, ... , n-1. On the other hand, constraint set (3.8) guarantees that the completion time of job j on a machine is equal or greater than the summation of the processing time of job j on that machine and the completion time of job j on the previous machine, j=1, ... , n. Constraint set (3.9) indicates the special case when the job in the first position is processed. The completion time of that job on machine i should be equal or greater than the processing time of that job on machine i, i=1, ... , m. We modified only this constraint from the original model which is proposed by Stafford and Tseng (2001). We eliminate the initial setup parameter from this constraint.

3.2.2 MILP Formulation for the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ Problem

For the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, we use the same managerial decision for the minimization of energy consumption as Mouozon and Yildirim (2008) and Yildirim and Mouzon (2012) use, which is that when the machine is kept idle for a long time, instead of keeping the machine idle, we can turn off and on the machine. However these authors consider the energy objective on a single machine scheduling problem, the parameter definitions and the proposed mathematical model in (2008) and (2012) are used and modified for the energy part of the FS-SDST problem.

The MILP model of the $F|s_{ijl}, prmu|C_{max}$ problem contains fundamental constraints for the FS-SDST problem. Since the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem has more characteristics in addition to the $F|s_{ijl}, prmu|C_{max}$ problem; sets and parameters, decision variables and constraint sets of the $F|s_{ijl}, prmu|C_{max}$ problem are used directly in the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem. Below, the additional system parameters, decision variables and constraint sets are presented.

Sets and parameters:

e_i cost of turning off/on machine i, i=1, ... , m,

d_i cost per time unit for idle period of machine i, i=1, ... , m,

t_i required time for turning off/on machine i, i=1, ... , m,

b_i breakeven duration for machine i, i=1, ... , m,

u_i cost per unit time for setup period of machine$i$, i=1, ... , m,

α cost per unit time of work in process inventory for all jobs

M big number.

Decision variables:

x_{ik} idle time on machine i before processing the job in position k, $i = 1, \ldots, m, k = 1, \ldots, n,$

$$y_{ik} = \begin{cases} x_{i,k+1} * d_i - e_i, & \text{energy cost when } x_{ik} > b_i, \\ & i = 1, \ldots, m, k = 1, \ldots, n-1, \\ 0, & \text{otherwise,} \end{cases}$$

$$v_{ik} = \begin{cases} 1, \text{if } x_{ik} < b_i, i = 1, \ldots, m, k = 1, \ldots, n, \\ 0, \text{otherwise.} \end{cases}$$

Objective function:

$Minimize$: $w_1(\alpha \sum_{k=1}^{n} C_{mk}) +$

$$w_2\left(\sum_{i=1}^{m} \sum_{k=1}^{n-1} (x_{i,k+1}\, d_i - y_{i,k+1}\ ) + \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{n-1} \sum_{l=1}^{n} s_{ijl} q_{jkl} u_i\right) \quad (3.10)$$

Constraint sets:

$$C_{ik} + \sum_{j=1}^{n} \sum_{l=1}^{n} s_{ijl}\, q_{jkl} + \sum_{j=1}^{n} p_{ij} z_{j,k+1} + x_{i,k+1} = C_{i,k+1}$$
$$(i = 1, \ldots, m,\ k = 1, \ldots, n-1) \quad (3.7')$$

$$x_{ik} \le b_i + M\left(1 - v_{i,k+1}\right) \quad (i = 1, \ldots, m, k = 1, \ldots, n-1) \quad (3.11)$$

$$b_i \le x_{i,k+1} + M v_{i,k+1} \quad (i = 1, \ldots, m, k = 1, \ldots, n-1) \quad (3.12)$$

$$y_{i,k+1} \le x_{i,k+1} d_i - e_i + M v_{i,k+1} \quad (i = 1, \ldots, m, k = 1, \ldots, n-1) \quad (3.13)$$

$$x_{i,k+1} d_i - e_i - M v_{i,k+1} \le y_{i,k+1} \quad (r = 1, \ldots, m, j = 1, \ldots, n-1) \quad (3.14)$$

$$y_{i,k+1} \le M(1 - v_{i,k+1}) \quad (i = 1, \ldots, m, k = 1, \ldots, n-1) \quad (3.15)$$

$$-M(1 - v_{i,k+1}) \le y_{i,k+1} \quad (i = 1, \ldots, m, k = 1, \ldots, n-1) \quad (3.16)$$

The constraint set (3.10) shows the multiobjective function. In the $F|s_{ijl}$, $prmu|\sum(C_j + Energy_j)$ problem, we have two objectives which are to minimize the total completion time and to minimize the total energy cost during keeping the machine idle, turning off/on operation when it consumes less energy instead of keeping the machine idle, and setup energy. Since the energy consumption during processing any job is fixed, it is not included into the function. In order to combine these two objectives into a single objective, we need to convert the total completion time to cost. For conversion, we multiply the total completion time value with a parameter α, which denotes the cost per unit time of work-in-process inventory. Hence, we basically multiply the total completion time with the α value and we consider this cost as a total production cost for this flowshop problem. In constraint set (3.10), we combine these two objectives into a single objective by weighted sum method. In this method, two objectives are scaled into a single objective function with weighted sum parameters w_1 and w_2. According to the relative importance, multiplied weights are set, which will be discussed in Section 5.1.2. Consequently, the constraint set (3.10) minimizes the total cost of the system which includes the total production cost due to total completion time and total energy cost.

Constraint sets (3.2)-(3.9) which are presented in Section 3.2.1 are also used for the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem except for constraint set (3.7). For the energy part of the problem, since we examine the idle times of the machines, we define a decision variable x_{ik} , which denotes the idle time on machine i before processing the job in position k, i=1, ... , m, k=1, ... , n. As a consequence, the constraint set (3.7) is modified by inserting the new decision variable x_{ik}, i=1, ... , m, k=1, ... , n, and converted into an equality constraint. Hence, instead of equation set (3.7), we use the modified constraint set (3.7'). Constraint sets (3.11) to (3.16) are developed for the energy part of the problem. Constraint sets (3.11) and (3.12) determine the values of decision variable v_{ik} which indicates whether the idle time on machine i before processing the job in position k is greater than breakeven duration for this machine or not, i=1, ... , m, k=1, ... , n. Breakeven duration is the least amount of duration when turning off and on the machine between two consecutive jobs is more favorable than keeping the machine idle. Namely, breakeven duration shows the critical time for turning off/on the machine. This parameter is calculated by the Equation set (3.17) shown below. To calculate the breakeven duration, we use the cost of turning off/on the machine, cost per time unit for idle period of machine and the least required time for turning off/on the machine. When the amount of idle time on machine i before processing the job in position k is greater than the breakeven duration; then v_{ik} will be zero, which means that turning off and on the machine consumes less energy, i=1, ... , m, k=1, ... , n. According to constraint sets (3.13), (3.14), (3.15) and (3.16), the decision variable y_{ik}, i=1, ... , m, k=1, ... , n, is calculated, which indicates the difference of the energy cost between keeping the machine idle and turning off/on the machine. Otherwise (when v_{ik} is one), y_{ik} value, i=1, ... , m, k=1, ... , n, will be zero; which means that keeping the machine idle is more favorable.

$$b_i = \max\left\{\frac{e_i}{d_i},\ t_i\right\} \quad (i = 1, \dots, m) \tag{3.17}$$

Chapter 4

A SOLUTION APPROACH

The $F|s_{ijl}, prmu|C_{max}$ problem is shown as strongly NP-hard by Gupta and Darrow (1986), when the objective function is makespan. The $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem is also NP-hard since it is more complex than the $F|s_{ijl}, prmu|C_{max}$ problem. Due to the complexity of these problems, it is not expected to solve these problems with exact methods in a reasonable time. Hence we propose a robust heuristic algorithm which is used for both of these two problems. When we survey the literature, we observe that the hybrid heuristic algorithms with a local search procedure have better performance for the FS-SDST problem. Hence, we are motivated to propose a VNS algorithm. Moreover, to the best of our knowledge, the VNS algorithm has not been applied to the FS-SDST problem in the literature.

In the following sections, the basic steps of the VNS algorithm and our implementation of this algorithm are given in detail.

4.1 Variable Neighborhood Search

Variable neighborhood search (VNS) is a metaheuristic algorithm which is used to solve various combinatorial optimization problems. This metaheuristic was proposed by Mladenovic and Hansen (1997). The VNS algorithm searches for the best solution in the multiple neighborhood structures and uses local search systematically. Initially, the number of neighborhoods and the type of neighborhood structures should be determined for the algorithm. The steps of the basic VNS algorithm are given in Figure 4.1. Moreover, Figure

4.2 illustrates the scheme of the basic VNS algorithm. General VNS algorithm starts with a solution, generates another solution in the neighbor of this solution and performs a local search on this solution aiming for a better solution. As it can be seen from Figure 4.1 and 4.2, VNS contains '*shaking*', '*local search*' and '*move or not*' steps. In the '*shaking*' step, one candidate solution is selected from the neighborhood of the current solution. Then, the local search procedure is applied to this candidate solution. In the last step, the objective function value of the solution found at the end of the local search is compared with the incumbent solution. If the solution is improved, then this solution is accepted as the current solution and the algorithm continues with the first step in the first neighborhood structure. Otherwise, the current solution is not updated and the algorithm continues with the first step but in the next neighborhood structure. These steps will be repeated until the stopping criterion is met.

Initialization Select the set of neighborhood structures N_k, for $k = 1, \ldots, k_{max}$, that will be used in the search; find an initial solution x; choose a stopping condition;

Repeat the following sequence until the stopping condition is met:

(1)Set $k \leftarrow 1$;

(2)Repeat the following steps until $k = k_{max}$:

(a) *Shaking* Generate a point x' at random from the k^{th} neighborhood of x ($x' \in N_k(x)$);

(b) *Local search* Apply some local search method with x' as initial solution; denote with x'' the so obtained local optimum;

(c) *Move or not* If this local optimum is better than the incumbent, move there ($x \leftarrow x''$), and continue the search with N_1 ($k \leftarrow 1$); otherwise, set $k \leftarrow k+1$;

Figure 4.1 Steps of the basic VNS (Mladenovic and Hansen, 1997)

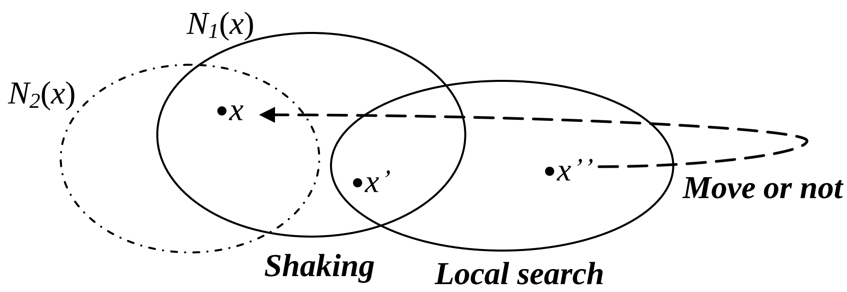

Figure 4.2 Scheme of the basic VNS

In the following section, we explain our design and implementation of the VNS algorithm for the FS-SDST problem in detail.

4.2 Implementation of the VNS Algorithm

For the VNS algorithm, firstly we determine the representation of the solution. Then, as decisions for the initialization of the VNS algorithm, we develop initial solution procedures, set of neighborhood structures and the local search procedures with respect to the properties of the flowshop problem with sequence dependent setup times. Finally, we present the stopping criterion for our VNS. Additionally, we use different acceptance criteria for the last step (‘*move or not*’) in the VNS algorithm. These decisions are explained in detail in the following subsections. The same design desicions are used for both the $F|s_{ijl}, prmu|C_{max}$ problem and the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, since the basic characteristics of the problems are the same.

4.2.1 Solution Representation

Since we consider a permutation flowshop environment, it is sufficient to consider one sequence for all machines in order to represent a solution to the problem. Hence, we represent a solution in the VNS algorithm as a permutation of the jobs [$j_1, j_2, \ldots, j_n$]. In this representation, j_n indicates that corresponding job is processed in the n^{th} order in the sequence. As an example, to indicate the sequence of 10 jobs (n=10) processed on m machines, an array of one to ten (1 x n) dimension such that [3 5 9 2 6 8 4 1 7 10] is used. This array indicates that job 3 is processed in the 1st position on all m machines; job 5 in the 2nd position; job 9 in the 3rd position; job 2 in the 4th position and the other jobs in the given array are interpreted in the same pattern.

4.2.2 Initial Solution

In order to observe whether our implementation of VNS is sensitive to the initial solution or not, we consider two different procedures to create the initial solution in the proposed VNS algorithm. The first procedure generates a random sequence of jobs. The second procedure uses the extended version of a well-known construction heuristic, Nawaz-Enscore-Ham (NEH) heuristic (Rios-Mercado and Bard, 1998b). Original NEH heuristic was proposed for flowshop problems with makespan objective by Nawaz et al. (1983). This method is based on inserting a job into all possible positions of the partial scheduled solution; and constructing the solution by adding the job into the best position that gives a better objective function value. Since this method's complexity is O(mn^3), Taillard (1990) proposed speed-ups to decrease the computational complexity to O(mn^2) for the m machine, n job flowshop scheduling problem with makespan objective. Rios-Mercado and Bard (1998b) extended this NEH heuristic (NEH_RMB) with proposed acceleration to the FS-SDST problems.

NEH_RMB heuristic (Rios-Mercado and Bard, 1998b) takes a set of unscheduled jobs and constructs a feasible solution S. Each job is taken from a set which contains unscheduled jobs under a priority rule. We use largest processing time (LPT) rule which was suggested by Nawaz et al. (1983). In LPT rule, jobs are ordered from the largest to the smallest total processing time on all m machines. This order creates the job list P. At each iteration of the NEH_RMB algorithm (Rios-Mercado and Bard, 1998b), the first job on the list P, job j, is removed and inserted into all possible positions in the partial schedule S. Rios-Mercado and Bard (1998b) defines a greedy function $\psi(k)$ which takes the value of makespan of the new schedule after inserting job j into the position k. Rios-Mercado and Bard (1998b) also refer this greedy function $\psi(k)$ as partial makespan since it computes the makespan values of partial schedule S. According to calculation of $\psi(k)$ function (partial makespan) for every position k=1, ... , |S+1|, the lowest makespan value is found and job j is inserted into that corresponding position, denoted by k^*, in the partial schedule S. This procedure continues until there is no job in the list P, meanwhile the constructed set S contains all n jobs. Figure 4.3 gives the pseudocode of the NEH_RMB procedure (Rios-Mercado and Bard, 1998b).

Procedure NEH_RMB

Input: Set P of unscheduled jobs.

Output: Feasible schedule S.

Step 0. Set $S = \emptyset$

Step 1. Sort the jobs in P to form an LPT priority list

Step 2. while $|P| > 0$ do

Step 2a. Remove j, the first job from P

Step 2b. Compute $\psi(k)$ for every position $k = 1, \ldots, |S+1|$

Step 2c. Find $k^* = \text{argmin}_j\{\psi(k)\}$

Step 2d. Insert job j at position k^* in S

Step 3. Output S

Step 4. Stop

Figure 4.3 Pseudocode of NEH_RMB procedure (Rios-Mercado and Bard, 1998b)

Rios-Mercado and Bard (1998b) use Taillard's speed-ups to compute partial makespan when job j is inserted into each position in the sequence, given in Step 2b of Figure 4.3. Instead of computing the completion times for every position, Taillard (1990) proposed to calculate the earliest completion times and tails of a partial solution S; and then to calculate the relative completion times and partial makespan values after the insertion of job j into schedule S. Earliest completion time measures the duration between the starting time of the operations and the end of processing of job j, j=1, ... , n, on machine i, i=1, ... , m. The earliest completion times are computed recursively from the first job in the sequence on the first machine to the last job in the schedule S on the last machine, m. Tail calculates the duration between the starting time of job j on the machine i and the end of operations. Tails are also computed recursively; but from the last job of the scheduled sequence on machine m to the first job in the schedule S on the first machine. The earliest completion times and tails are calculated once until one job, job j, is added to the schedule S, which eliminates the redundant computations. The relative completion time and values of the partial makespan $\psi(k)$ are calculated when job j (from set P) is inserted in position k. To compute the relative completion times and values of the partial makespan, we use the earliest completion times, the tails, the processing time of job j and the related sequence dependent setup times; instead of computing the whole completion times when job j is inserted in every position in the schedule S. Figure 4.4 presents the pseudocode for the procedure of computing values of partial makespan (Rios-Mercado and Bard, 1998b).

For the $F|s_{ijl}, prmu|C_{max}$ problem, the presented NEH_RMB algorithm (Rios-Mercado and Bard, 1998b) with Taillard's speed ups is used as explained above. However for the $F|s_{ijl},prmu|\sum(C_j+Energy_j)$ problem, the greedy function $\psi(j)$ calculates the total cost of the system which includes the total production cost due to total completion time and total energy cost, instead of makespan. Moreover, since the Taillard's speed ups are proposed specifically for the flowshop problem with makespan objective, we do not use any speed up procedures for the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem.

Procedure Makespan ( )

Input: Partial schedule S=(1,2, …, k-1) and job j to be inserted.

Output: Vector $\psi(k)$ with the value of the makespan when job j is inserted in the k-th position of the schedule S.

Step 1. Compute the earliest completion times

Step 2. Compute the tails

Step 3. Compute the relative completion times

Step 4. Compute values of partial makespan $\psi(k)$

Step 5. Output vector $\psi(k)$

Step 6. Stop

Figure 4.4 Pseudocode of procedure for computing partial makespan (Rios-Mercado and Bard, 1998b)

4.2.3 Neighborhood Structure

Selecting the neighborhood structures, the sequence of the selected neighbors and the local search procedure play an important role in the performance of the VNS algorithm. Inherently, search space of neighborhoods affects the quality of the algorithm. A large neighborhood contains the global optimum solution with higher possibility compared to a small neighborhood. However, a large neighborhood visits to any solution in the search space with lower probability.

In the preliminary tests, we try various neighborhood structures with different combinations in order to obtain better solutions. We present these preliminary results in Section 5.2. As neighborhood structure, we employ several neighborhood structures: swap, adjacent swap, node insertion, 2-opt, maximum setup one-job insertion, maximum setup two-job insertion and minimum setup two-job insertion. Moreover, we use several local search procedures based on different neighborhoods. In the following, we explain these neighborhood structures in detail and in Section 4.2.4, we will explain the implementation of these neighborhood structures in local search procedures.

Swap: We employed swap operator as one of the neighborhood structure for VNS algorithm. In this operator, we generate two different values in job set *n*. These two values represent job positions whose corresponding jobs will be swapped. For instance, if the current solution is [3 5 9 2 6 8 1 4 7 10], for *n*=10, and the generated random values are 4 and 7; then the new solution becomes [3 5 9 1 6 8 2 4 7 10]. Before the swap move, job in the fourth position in the sequence is 2, and job in the seventh position is 1. After the swap move, jobs in the fourth and the seventh positions in the sequence are 1 and 2, respectively, while other jobs remain in the same position.

Adjacent Swap: In adjacent swap move, two adjacent jobs interchange their positions in the sequence. To determine adjacent jobs, we generate one random number which represents a random position in the current solution. We swap the corresponding job of the selected position with the job which will be processed in the following position. For instance, the current solution is [3 5 9 2 6 8 1 4 7 10], and the generated random value is 4. Then, we interchange job 2 which corresponds to position 4 and job 6 which will be following job processed in the sequence. In consequence, we obtain the new solution as [3 5 9 6 2 8 1 4 7 10].

Node Insertion: In the node insertion operation, we generate one value randomly to select a job which will be removed from the sequence and to be inserted into another position. For inserting the removed job, we can select the position randomly or we can find the best position which gives the lowest objective function value. The second procedure makes the shaking part intensified; on the other hand increases time spent in the neighborhood. For example, if the current solution is [3 5 9 2 6 8 1 4 7 10] and the randomly generated value is 5 which shows the position in the sequence, corresponding job will be 6. We remove job 6 from the sequence, insert this job into every position excluding 5^{th} position (which is the current solution) and calculate the objective function value for the new sequence. After the computation of the objective function value, we determine the lowest fitness value and insert job 6 into the corresponding position. In the example, if this corresponding position is 2, the new sequence will be [3 6 5 9 2 8 1 4 7 10]. We also use this neighborhood structure for two jobs. Instead of removing one job in the sequence, we eliminate two jobs randomly. Then, we reinsert these two jobs in the sequence under the explained procedures.

2-opt: In the 2-opt neighborhood, we select two adjacent job pairs, namely two edges from the sequence. We remove these selected two edges and reconnect in the other way to the partial sequence. For example, if the current solution is [3 5 9 2 6 8 1 4 7 10] and we select [9 2] and [4 7] adjacent job pairs in the sequence, the new solution becomes [3 5 9 4 1 8 6 2 7 10]. The edges between the jobs [9 2] and [4 7] are removed and the partial sequence between job 9 and job 7 (which are [2 6 8 1 4]) is reconnected with the main sequence in opposite way.

Maximum Setup One-job Insertion: Since the problem is flowshop problem with sequence dependent setup times, we develop three setup time dependent neighborhood structures. For these neighborhood structures, we use the setup times between two adjacent jobs only on the first machine. In the preliminary tests, we use also the total setup times between each job pairs on all machines. However, using the setup times on the first machine gives us better results. In the maximum setup one-job insertion, according to the current solution, we determine the largest setup time between every adjacent job on the first machine. Then, we remove the job right after the largest setup time and insert it to another position in the sequence randomly. Random insertion leads to diversification after the selection of the removed job. For instance if the current solution is [3 5 9 2 6 8 1 4 7 10] and the largest setup time on the first machine is between job 4 and job 7, then we remove job 7 from the sequence. We generate a random number from the set (1, ... , *n*, in this example *n* is 10), say 3; then we insert job 7 into position 3. So, the new sequence will be [3 5 7 9 2 6 8 1 4 10].

Maximum Setup Two-job Insertion: The second neighborhood structure based on setup times is similar to the maximum setup one-job insertion. Firstly, we determine the greatest setup time between adjacent jobs in the current solution on the first machine. Different from the previous one, two adjacent jobs which have the largest setup time are removed from the sequence and are inserted in a different position in the sequence. For example, if the current solution is [3 5 9 2 6 8 1 4 7 10] and the largest setup time on the first machine is between job 4 and job 7, then we remove both job 4 and job 7 from the sequence. After generating a random number, say 5, we insert jobs 4 and 7 adjacently into positions 5 and 6. So, the new solution becomes [3 5 9 2 4 7 6 8 1 10].

Minimum Setup Two-job Insertion: The third neighborhood structure we developed based on setup times is similar to the maximum setup two-job insertion. The difference in this neighborhood is that we determine the lowest setup time between two adjacent jobs. Then, we eliminate these two jobs from the sequence and insert into another position generated randomly. In the previous example, if the minimum setup time is observed in between jobs 3 and 5, and random number is 9; then we insert jobs 3 and 5 into positions 9 and 10 respectively. Consequently, the new solution will be [9 2 6 8 1 4 7 10 3 5].

Neighborhood structures based on setup times which contains maximum setup one-job insertion, maximum setup two-job insertion and minimum setup two-job insertion In the preliminary tests,

4.2.4 Local Search Procedure

Local search takes the solution, *x*, which is obtained at the '*shaking*' step of the VNS algorithm, as an initial solution and we search a new solution in the neighborhood of initial solution *x*, *N*(*x*). The local search procedure may consist of one or multiple neighborhood structures, which we give the details of these structures below. On the other hand, to improve the solution *x* in *N*(*x*), best improvement (steepest descent) and first improvement (random descent) strategies are tested for the local search procedure. In the steepest descent strategy, we search the whole neighborhood and accept the solution x' which gives the best improvement to the objective function, also called local minimum of the neighborhood. On the other hand, random descent strategy selects solutions from the neighborhood randomly and accepts the first solution x' that improves the objective function.

For the local search procedure based on one neighborhood structure, we employ several neighbors: swap, adjacent swap, node insertion, 2-opt and maximum setup two-job insertion. The details of these neighborhood structures are given in Section 4.2.3. For the local search procedure based on multiple neighborhood structures, we test the same neighborhood structures used for the local search procedure based on one neighbor. To use the multiple neighborhoods systematically in the local search, we implement the variable neighborhood descent (VND) algorithm, which is proposed by Mladenovic and Hansen (1997). Different from the VNS algorithm shown in Figure 4.1, there is no '*shaking*' phase in the VND algorithm, since it finds steepest descent solution in the selected search space. The neighborhood structures of the VND algorithm may be different from the VNS algorithm. The initial solution for the VND comes from the '*shaking*' phase of the VNS algorithm. Then the best improved solution is searched in the first neighborhood structure in the VND algorithm. Then, similarly in the '*move or not*' phase, the new solution is decided to be accepted or not after comparing with the incumbent of the VND algorithm. According to the acceptance, the neighborhood structure is changed. For this VND algorithm, the stopping condition is the improvement of the solution according to the incumbent value of the main VNS algorithm. When there is no improvement any more, then the local search phase of the VNS (here it is VND) is completed.

After the preliminary tests, which we provide the detailed analysis of these local search procedures in Section 5.2.2, we observe that the local search procedure based on node insertion neighborhood with steepest descent strategy gives best results among the local search procedures based on other neighborhood structures and strategies. Moreover, Ruiz and Stützle (2007, 2008) also use the local search procedure based on node insertion neighborhood with steepest descent strategy for the $F|s_{ijl}, prmu|C_{max}$ problem in their state-of-the-art algorithm and they emphasize that the node insertion neighborhood structure is highly effective for the local search procedure for the FS-SDST problems.

```
Function LocalSearch_NodeInsertion(x)
    improve=true;
    while (improve=true) do
        improve=false;
        for i=1 to n do
            remove job h from sequence x randomly without repetition
            x'= best sequence obtained by inserting job h in all possible positions in x;
            if F(x') < F(x) then
                x = x';
                improve=true;
            endif
        endfor
    endwhile
    return x
end
```

Figure 4.5 Local search based on node insertion with steepest descent strategy (Ruiz and Stützle, 2007, 2008)

Figure 4.5 shows the pseudocode of this local search procedure (Ruiz and Stützle, 2007, 2008). In the local search procedure based on node insertion, one job is selected from the sequence of the starting solution, x; removed and inserted in all possible positions except for its original one. While we are computing the makespan values for each corresponding position for the $F|s_{ijl}, prmu|C_{max}$ problem, we use Taillard's speed-ups. The position which gives the lowest objective function value will be the new position for this selected job and we obtain new sequence, x'. If the objective function value of the new sequence, $F(x')$, is lower than the starting sequence, $F(x)$, then we update the solution and continue with this new sequence; otherwise we continue with the starting (original) sequence. We select another job from the sequence and do the same procedure as explained before, for all jobs in the sequence. We call this explained procedure until we do not improve the objective function value anymore (until we obtain local optimum).

4.2.5 Acceptance Criterion

After obtaining a sequence from the local search, the corresponding objective function value is compared with the incumbent value which is the lowest objective function value found until that time. According to that comparison, we decide whether this new solution is accepted or not as an incumbent solution; and to move to the following neighborhood structure to search better solutions. In the acceptance phase, we employ two different criteria: a simple acceptance criterion and a simulated annealing-like acceptance criterion. In the first one, we accept the new sequence if its objective function value is lower than the incumbent value. However, we observe that sometimes the local search gets stuck at a solution in earlier stages of the algorithm. Hence to prevent this situation, we accept some solutions which have higher objective function values than the incumbent value in the second acceptance criterion. This strategy, which provides diversification to the algorithm, leads us to search solutions in different parts of the solution space and prevents stagnation. We use the following well-known equation for the simulated annealing-like acceptance criterion:

$$rand \leq \exp\left\{\frac{F(incumbent)-F(x)}{temperature}\right\} \tag{4.1}$$

In Equation 4.1, *rand* is a random number between 0 and 1. If x represents the new sequence, then $F(x)$ shows the objective function value for the new sequence. Similarly, $F(incumbent)$ indicates the objective function value of the incumbent solution. Thus, Equation 4.1 gives the probability of accepting a worse solution. We use the following equation for the temperature which was proposed by Osman and Potts (1989) and was adopted by Ruiz and Stützle (2007, 2008).

$$temperature = \lambda \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} p_{ij}}{n \times m \times 10} \qquad (4.2)$$

The value of the parameter *temperature* depends on the instance since it uses the total processing time of all jobs on the all machines, the number of jobs and the number of machines, as seen in Equation 4.2. In this equation, λ is a parameter which needs to be tuned. In Section 5.3.1, the experimental results for tuning λ are shown.

We present the pseudocode of the proposed VNS algorithm in Figure 4.6.

Notation

k_{max}	maximum number of neighborhood structure
x_o	the initial solution
x^*	the incumbent solution
x	the current solution
x'	the solution generated from $N_k(x)$
x''	the local optimum solution from x'
$N_k(x)$	the kth neighborhood of solution x
$F(x)$	the objective function value of x
rand	random number generated between 0 and 1
temperature	a constant value calculated in Equation 4.2

```
x_o = NEH_RMB;                                          %Initialization
x = x_o ;
x*= x_o;
F(x*) =F(x_o);
k=1;
while termination criterion is not met do
    while k ≤ k_max do
        Generate x' from N_k(x)                         %Shaking
        x''= LocalSearch_NodeInsertion(x');             %Local search
        if F(x'') < F(x*)        then                   %Acceptance & move or not
            x=x'';
            x*=x;
            k=1;
        else if (rand ≤ exp{-(F(x'')-F(x*))/temperature}) then
            x=x'';
            k=1;
        else
            k=k+1;
        endif
    endwhile
endwhile
```

Figure 4.6 Pseudocode of proposed VNS algorithm

Chapter 5

COMPUTATIONAL STUDIES

In this chapter, we conduct computational studies to analyze the performance of the VNS algorithm, which is proposed for the $F|s_{ijl}$, $prmu|C_{max}$ and the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problems. Firstly, in Section 5.1, we describe the data sets that we use for the $F|s_{ijl}$, $prmu|C_{max}$ problem and we generate additional data sets for the energy-related parameters for the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem. In Section 5.2, we present our test results to make the decisions for our implementation of the VNS algorithm. In Section 5.3, we give the results of the performance of the proposed VNS algorithm. Lastly, we analyze these results for the $F|s_{ijl}$, $prmu|C_{max}$ problem and the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem in Section 5.4.

5.1 Data Sets

5.1.1 Data Set for the $F|s_{ijl}$, $prmu|C_{max}$ Problem

We conduct the computational studies for the $F|s_{ijl}$, $prmu|C_{max}$ problem with the data set generated by Ruiz et al. (2005). This instance set is based on the well-known data set proposed by Taillard (1993) for the regular flowshop problem. Taillard’s set has 12 instance groups with different number of jobs and machines. Nine of these instance group sizes are the combination of 20, 50, 100 jobs and 5, 10, 20 machines. For the larger instances, there are three more groups: 200 jobs 10 machines, 200 jobs 20 machines and 500 jobs 20 machines. In each group, there are 10 instances, so there are totally 120 instances in the data set. For these instances, the processing time of jobs on each machine is generated uniformly between 1 and 99. To modify this data set to the FS-SDST problem, Ruiz et al. (2005) generated setup time values as four groups according to the 10%, 50%, 100% and 125% of the processing time range [1,99]. Hence, the setup times are generated uniformly in [1,9], [1,49], [1,99], [1,124] and called SDST10, SDST50, SDST100, SDST125, respectively. In total, there are 480 instances with different setup time groups and we use all 480 instances to test our proposed VNS algorithm for the $F|s_{ijl}, prmu|C_{max}$ problem.

5.1.2 Data Set for the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ Problem

For the energy-aware FS-SDST problem, we need the following parameters for computational experiments: processing times of each job on each machine, sequence dependent setup times, setup cost, idle cost, cost for turning off/on machine and the minimum required time for turning off/on the machine. For the processing and setup time, we will use the benchmark set generated by Ruiz et al. (2005) for the FS-SDST problem as explained in Section 5.1.1. For the smaller instances, to compare the solutions obtained from the proposed VNS algorithm with the optimal solutions, we generate the processing time uniformly in [1, 99] and setup time in [1, 9], [1, 49], [1, 99], [1, 124]. Since there are few studies about energy-aware scheduling, there are no known data sets for energy-related parameters. As a result, we generated the energy-related data sets for the computational experiments. For data generation, firstly we created random 30 instances which have job number in [20,200] and machine number in [5, 20]. The processing times were generated uniformly in the range of [1, 99]. Setup times were generated uniformly in the ranges of [1, 9], [1, 49], [1, 99], and [1, 124] with random selection of setup time groups. The parameter α, which is used to convert the total completion time into the cost in the multiobjective function, was selected as 5. We utilize the total weighted completion time articles for cost conversion. Belouadah et al. (1992) generate weight data for the single machine scheduling problem to minimize the total weighted completion time in their study. They choose the range as [1, 10] for weight where the processing time is generated uniformly in [1, 100] range. Since we have the similar processing time range and we assume that all jobs have the same α value, we took the mean value of [1, 10] range, as 5, and used this constant value for all jobs. Created 30 instances were run with the proposed VNS algorithm where the objective is only the total production cost (single objective function). According to the cost values, three different range groups were determined for the energy-related parameters: low, medium and high. In Table 5.1, the data ranges for the different groups are presented. The required time for turning off/on the machine was generated uniformly in the range of [1, 20] for all groups.

Table 5.1 Data ranges of different groups for energy-related parameters

Group	**Idle Cost**	**Setup Cost**	**ON/OFF Cost**
Low	[1,5]	[1,10]	[10,100]
Medium	[15,25]	[35,45]	[350,450]
High	[30,40]	[70,80]	[700,800]

On the other hand, we set the weighted sum parameters w_1 and w_2 which we defined in Section 3.2.2, to combine the objectives the total production cost and the total energy cost into a single objective function by weighted sum method. For these weighted sum parameters w_1 and w_2, three sets were used: (0.1, 0,9), (0.5, 0.5), (0.9, 0.1) and (1.0, 0.0). Since we add the energy objective to the traditional flowshop problem, we use (1.0, 0.0) values for the weighted sum parameters to evaluate the effect of the energy cost to the objective function. As a result, for one instance from Ruiz et al. (2005) data set (with processing and setup times), three instances were generated due to different range groups for the energy related parameters. Moreover, each new instance was run four times since there are four different values for the weighted sum parameters.

5.2 Selection of the VNS Decisions

As decisions of the VNS algorithm, we presented initial solution procedures, set of neighborhood structures, local search procedures, stopping and acceptance criteria in Section 4.2. In this section, we test these decisions to select the best combination for the proposed VNS algorithm. Firstly, we describe the termination criterion for the algorithm. While using this criterion, we show the results of the preliminary tests for different initialization procedures, neighbors, local search procedures and acceptance criteria. Then, we tune the acceptance criterion parameter in the last subsection. For the preliminary tests, only the $F|s_{ijl}$, $prmu|C_{max}$ problem is used, since it is the simplest version of the FS-SDST problem. Moreover, the best known solutions of the Taillard's instance set for the $F|s_{ijl}$, $prmu|C_{max}$ problem are published, which allow us to measure the performance of the test results.

5.2.1 Termination Criterion

For the $F|s_{ijl}$, $prmu|C_{max}$ problem, we set the stopping condition based on the CPU times since we compare our results with the state-of-the-art and other methods, which are implemented with the same stopping criterion by Ruiz and Stützle (2008). We set the CPU time as (n x m/2) x t milliseconds as proposed by Ruiz and Stützle (2008). Hence, the termination criterion depends on the size of the instances. Ruiz and Stützle (2008) set t as 30, 60 and 90 in their article but we only use t = 90 in our study.

For the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem, since we do not compare our result with a benchmark, we give longer CPU time as a termination criterion. One reason for this is that the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem is more complex than the $F|s_{ijl}$, $prmu|C_{max}$ problem. Moreover, for the $F|s_{ijl}$, $prmu|C_{max}$ problem, we use Taillard's speed up methods to calculate the makespan value after insertion operation as explained in Section 4.2.2.

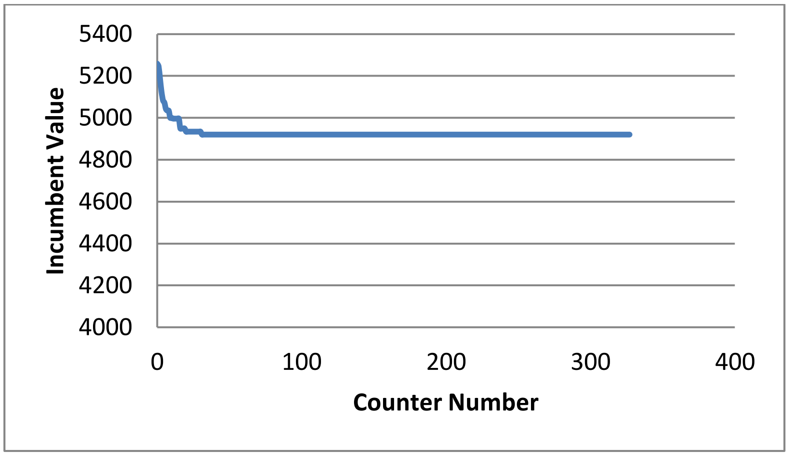

Figure 5.1 Convergence graph of a 50x10 instance with SDST100 for $F|s_{ijl}, prmu|C_{max}$

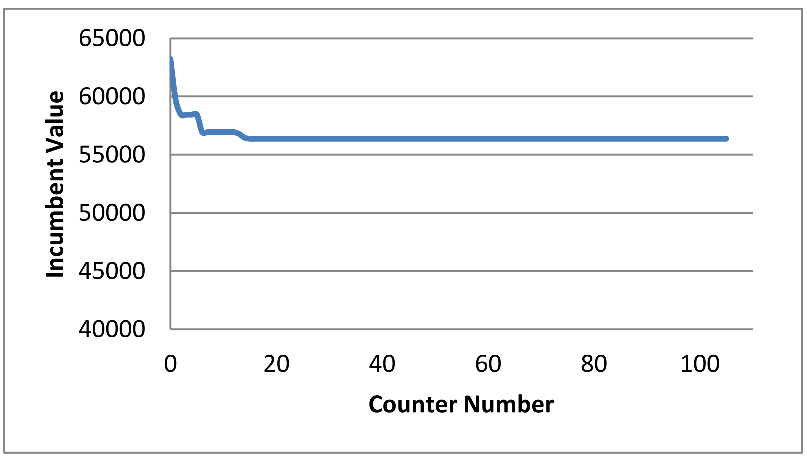

Figure 5.2 Convergence graph of a 20x5 instance with SDST10 and high group energy data set for $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$, when w_1=0.5, w_2=0.5

However, this speed up procedure cannot be applied to the minimization of total completion time objective. As a result, one iteration takes much more time compared to the $F|s_{ijl}, prmu|C_{max}$ problem. To obtain a good solution in a reasonable time period depending on the instance size, we use the same stopping condition for the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, but we set t as 180 due to the difficulty of the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem.

Since we use the time as termination criterion, we also analyze whether the solution converges or not. As an example, Figure 5.1 and 5.2 show the convergence graphs of the $F|s_{ijl}$, $prmu|C_{max}$ and the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problems, respectively, for the proposed VNS algorithm.

5.2.2 Initialization, Neighbors, Local Search and Acceptance

The most important parts of designing a VNS algorithm are to select neighborhood structures, the sequence of these neighborhoods and the local search procedure. In the preliminary tests, we tried various neighborhood structures and local search procedures with different combinations in order to obtain better solutions. As neighborhood structures, we used adjacent swap, swap, node insertion as one job (Insertion (1)) and adjacent two-job (Insertion (2)), maximum setup insertion as one-job (MaxSetup(1)) and two-job (MaxSetup(2)), minimum setup two-job insertion (MinSetup(2)) and 2-opt. These neighborhood structures were explained in detail in Section 4.2.3. In addition to the neighborhood structures, we also conducted preliminary tests about initialization procedure and acceptance criterion. As the initial solution, we used random initial solution and NEH_RMB constructive heuristic (Rios-Mercado and Bard, 1998b). On the other hand, we used simple and simulated annealing-like (SA-like) acceptance criteria. Since we have too many decisions for designing a VNS algorithm, instead of combining each decision, we conducted our preliminary tests as following. Firstly, we kept the neighborhood structures as node insertion (Insertion(1)) and swap operations which are commonly used for flowshop and TSP based problems in the literature. We observed the effect of the initial solution, acceptance criterion and the neighborhood structure in the local search. The quality of the solutions was evaluated by average percentage deviation from the best known solutions, which is explained in detail in Section 5.3.3. For the first step of the preliminary test, we used small and medium size instance sets among the well-known data set from Section 5.1.1.

Table 5.2 Average percentage deviation of the VNS solutions from the best known solutions for different initial solutions, local search procedures and acceptance criteria

Initial Solution	Local Search	Acceptance Criterion	SDST 10	SDST 50	SDST 100	SDST 125	Avg.
NEH_RMB	Adjacent Swap	Simple acc.	2.74	3.78	4.23	5.76	4.13
NEH_RMB	Insertion(1)	Simple acc.	1.77	3.41	3.96	4.98	3.53
NEH_RMB	Adjacent Swap	SA-like	1.94	3.43	4.01	5.33	3.68
NEH_RMB	Insertion(1)	SA-like	**1.43**	**2.97**	**3.61**	**4.52**	**3.13**
Random	Insertion(1)	SA-like	1.89	3.05	3.92	4.61	3.37

In Table 5.2, we observe that for each SDST groups, starting the VNS algorithm with NEH_RMB (Rios-Mercado and Bard, 1998b) gives better results compared to random initialization. Since we used the stopping condition based on CPU time, improving an initial solution with higher quality, such as NEH_RMB (Rios-Mercado and Bard, 1998b) solution, is faster than a random solution. To start the algorithm with a good solution may cause early convergence in heuristic algorithms in general; however the structure of the VNS algorithm with systematic change in neighborhood structures leads to diversification and prevents the convergence in early iterations. On the other hand, with simulated annealing-like acceptance criterion, we allow more diversification to the algorithm. Table 5.2 shows that using SA-like acceptance criterion gives better results for the $F|s_{ijl}$, $prmu|C_{max}$ problem. We used the NEH_RMB constructive heuristic (Rios-Mercado and Bard, 1998b) and SA-like acceptance criterion for the proposed VNS algorithm. In the local search procedure, we used best improvement strategy in the neighborhood structure. At the beginning of the preliminary tests, we used a simple neighborhood structure, that is adjacent swap, and a more complex one, that is node insertion. The VNS algorithm spends most of its computational time on the local search procedure to find the best solution in the specific neighbor. It is observed that evaluating the quality of each solution in the search space of the adjacent swap operator consumes less time than the insertion operation, which leads more iterations for the main loop of VNS algorithm under a CPU time based stopping condition. Although searching the best solution in node insertion neighbor takes longer time and leads to less iterations for the main loop, Table 5.2 shows that it gives better results. Namely, searching the best solution in the larger search space increases the quality of the solution. In Table 5.2, we observe that the adjacent swap operator is a weak neighborhood structure for a local search procedure compared to the insertion neighborhood operator.

Table 5.3 Average percentage deviation of the VNS solutions from the best known solutions for different neighborhood structures

Neighborhood Structure	SDST10	SDST50	SDST100	SDST125	Avg.
Swap - MaxSetup(1)	0.56	1.01	1.46	1.68	1.18
MaxSetup(1) - Swap	0.63	0.97	1.43	1.80	1.21
Swap - MaxSetup(2)	0.61	1.10	1.56	1.65	1.23
MaxSetup(2) - Swap	**0.52**	1.02	1.36	1.66	**1.14**
Swap - MinSetup(2)	0.56	1.02	1.48	1.80	1.21
MinSetup(2) - Swap	0.59	1.13	1.56	1.73	1.25
Swap – Insertion(1) - MaxSetup(1)	0.60	1.02	1.61	1.77	1.25
Swap - MaxSetup(1) – Insertion(1)	0.66	1.18	1.61	1.78	1.31
Insertion(1) - Swap - MaxSetup(1)	0.65	1.10	1.58	1.72	1.26
Insertion(1) - MaxSetup(1) - Swap	0.55	**0.89**	1.61	1.60	1.16
MaxSetup(1) - Swap - Insertion(1)	0.58	1.10	**1.34**	1.77	1.20
MaxSetup(1) - Insertion(1) - Swap	0.59	1.09	1.48	1.75	1.23
Swap - Insertion(1) - MaxSetup(2)	0.66	0.97	1.51	1.76	1.22
Swap - MaxSetup(2) - Insertion(1)	0.56	1.02	1.53	1.68	1.20
Insertion(1) - Swap - MaxSetup(2)	0.58	1.13	1.54	1.84	1.27
Insertion(1) - MaxSetup(2) - Swap	0.56	1.09	1.40	1.69	1.18
MaxSetup(2) - Swap - Insertion(1)	0.60	1.02	1.56	1.56	1.18
MaxSetup(2) - Insertion(1) - Swap	0.61	1.00	1.52	1.79	1.23
Swap - Insertion(1) - MinSetup(2)	0.58	1.04	1.46	1.72	1.20
Swap - MinSetup(2) - Insertion(1)	0.61	1.01	1.42	1.63	1.17
Insertion(1) - Swap - MinSetup(2)	0.61	1.08	1.62	1.77	1.27
Insertion(1) - MinSetup(2) - Swap	0.61	1.08	1.55	1.64	1.22
MinSetup(2) - Swap - Insertion(1)	0.59	1.04	1.49	1.69	1.20
MinSetup(2) - Insertion(1) - Swap	0.56	1.10	1.53	**1.55**	1.19

In the second step of our preliminary test, the aim is to select the neighborhood structures among swap, node insertion, maximum setup insertion as one-job and two-job, and minimum setup two-job insertion. Moreover, the sequence of these neighborhoods need to be determined. For this step, we fixed the initial procedure as NEH_RMB constructive heuristic (Rios-Mercado and Bard, 1998b), acceptance criterion as SA-like and local search procedure based on node insertion neighborhood. First we used two neighborhood structures. Since we have the local search procedure based on insertion, for two neighborhood structures, we combined the swap operator with each setup-dependent neighborhood structures. Then, we added the node insertion neighborhood as the third neighborhood structure to the algorithm. We used different instance sets from the first group of experiments as given in Table 5.2. We observe from the average percentage deviations given in Table 5.3 that there are no dominant neighborhood structures. However, using the maximum setup one-job insertion and the swap neighborhood structures in that order gives better results, on the average, among other neighborhood combinations.

Table 5.4 Average percentage deviation of the VNS solutions from the best known solutions for different neighborhood structures and local search procedures

Neighborhood Structure	Local Search	SDST10	SDST50	SDST100	SDST125	Avg.
MaxSetup(2) - Swap	Insertion(1)	**0.52**	**1.02**	**1.36**	**1.66**	**1.14**
MaxSetup(2) - Swap	(*) Insertion(1)	2.34	3.97	6.07	6.74	4.78
MaxSetup(2) - Swap	VND (Swap - MaxSetup(2))	1.69	3.12	4.62	5.16	3.65
MaxSetup(2) - Swap	VND (Insertion(1) - MaxSetup(2))	1.49	2.79	3.89	4.52	3.17
AdjacentSwap - Swap	AdjacentSwap - Insertion(1)	0.76	1.37	1.94	2.10	1.54
Insertion(2) - Swap	Insertion(2) - Insertion(1)	0.99	1.86	2.61	2.82	2.07
2-opt - Swap	2-opt - Insertion(1)	0.84	1.64	2.04	2.45	1.74

(*) With first improvement strategy

In the last step of the preliminary tests, we kept the initialization procedure and the acceptance criterion the same as discussed above. However, to improve our solution further, we used different techniques. First row in Table 5.4 shows the design of the VNS algorithm, which gives the best result so far (from Table 5.3) and will be referred to as the best combination. In the following three tests, we kept the neighborhood structures of the main algorithm the same but used different local search techniques. Firstly, we used the same neighborhood for the local search but instead of taking the best solution in the solution space (best improvement), we used the first improvement strategy. Moreover, we used variable neighborhood descent (VND) algorithm with two neighborhood structures in the local search procedure in our tests. We also tested using different neighbors in the local search procedures for each neighborhood structure in the main part of the algorithm. In the last three test results given in Table 5.4, we used different neighbors instead of MaxSetup(2) from the best combination of the given VNS design and we used the same neighborhood structure with the shaking part of the algorithm for the local search procedure. In other words, when a solution was generated from the adjacent swap neighbors randomly, the neighborhood structure would also be the same in the local search phase. Except from the current solution and the solution that was obtained from '*shaking*' part, the best improvement strategy was used to find the candidate solution. After SA-like acceptance, this candidate solution was accepted as the current solution or not. When the current solution was not accepted and the neighborhood structure was changed to the swap operation, in the local search procedure the node insertion neighbor was used. The algorithm continues until the stopping condition is met. From the results given in Table 5.4, we observe that we could not improve the solutions of the VNS algorithm referred as best combination further.

Consequently, we set the decisions of the proposed VNS algorithm as follows: NEH_RMB constructive heuristic (Rios-Mercado and Bard, 1998b) for the initial solution, SA-like acceptance criterion for '*move or not*' phase of the algorithm, maximum setup one-job insertion and swap neighborhood structures in that order for the neighborhood structures of the '*shaking*' phase, and node insertion neighborhood for the local search procedure.

5.2.3 Tuning the Parameter of Acceptance Criterion

In the simulated annealing-like acceptance criterion, we used the equation (4.2) for the temperature which was explained in Section 4.2.4. In this equation, there is a constant parameter, λ, which needs to be tuned. To design the VNS algorithm in Section 5.2.2, we took this parameter as 0.5, which is the same value as in Ruiz and Stützle (2008). However, we conducted a test to set this parameter after all other parameters and decisions of the proposed VNS algorithm were established. We used a small number of instance set to adjust the parameter λ and we again used the average percentage deviation to evaluate the results. In this test, we considered the values from 0.1 to 1.0 with the slot of 0.1 for λ. Table 5.5 shows the results according to the given λ values. In this table, we observe that when the λ value is tuned to 0.7, it gives the best results among other λ values, on the average. As a result, we set the λ value to 0.7 in our proposed VNS algorithm.

Table 5.5 Average percentage deviation of the VNS solutions from the best known solutions for different λ values in simulated annealing-like acceptance criterion

λ values	SDST10	SDST50	SDST100	SDST125	Average
0.1	0.14	0.38	0.44	0.59	0.39
0.2	0.17	0.29	0.57	0.61	0.41
0.3	0.14	0.23	0.45	0.55	0.34
0.4	0.12	0.33	0.56	0.62	0.41
0.5	0.11	0.31	0.43	0.49	0.33
0.6	0.10	0.27	0.39	0.59	0.34
0.7	**0.09**	**0.19**	0.44	**0.37**	**0.28**
0.8	0.13	0.30	**0.29**	0.43	0.29
0.9	0.11	0.25	0.37	0.43	0.29
1.0	0.12	0.32	0.41	0.57	0.36

5.3 Results of the Computational Study

In this section, we present the results of the computational experiments we conducted with the proposed VNS algorithm. We compared the results of the $F|s_{ijl}, prmu|C_{max}$ problem with the benchmarks. For the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, we solved small-sized problems via the mixed integer linear programming (MILP) model presented in Section 3.2.2 and compared the solutions obtained from the proposed VNS algorithm with the optimal solutions. For large-sized problems, we compared the VNS results with the well-known NEH constructive heuristic results.

5.3.1 Computational Platform

We code the VNS algorithm in Microsoft Visual Studio 2010 with C++ programming language. For solving the integer linear programming model, we used the IBM ILOG CPLEX Optimization Studio 12.5 Version. The computational experiments were conducted on a computer with an Intel(R) Core(TM) i5-2520M CPU at 2.50 GHz processor with 4.00 GB RAM.

5.3.2 Benchmarks

For the $F|s_{ijl}, prmu|C_{max}$ problem, to analyze the quality of the proposed VNS, we compared our results with the results of alternative heuristics and of the state-of-the-art algorithm. The iterated greedy algorithm with local search procedure (IG_LS) (Ruiz and Stützle, 2008) is state-of-the-art algorithm for the FS-SDST problem when the objective is makespan. IG_LS was also compared with some effective heuristics. Two of them are genetic algorithm (GA) and memetic algorithm (MA), which are developed by Ruiz et al. (2005). Moreover, Ruiz and Stützle (2008) modify an ant colony optimization algorithm (PACO) proposed by Rajendran and Ziegler (2004), which also gives good quality of solutions. Ruiz and Stützle (2008) use a stopping condition, which is based on CPU time, to compare the performance of these heuristic algorithms. Since we have the average deviation percentage values for these heuristics from the article (2008), we used these benchmark data to compare the performance of our VNS algorithm.

Genetic Algorithm (GA): Genetic algorithm is an evolutionary algorithm, which starts with a set of solutions called population, and after some genetic operations this population evolves. Ruiz et al. (2005) initialize some of the individuals in the population with a modified version of the NEH_RMB constructive heuristic (Rios-Mercado and Bard, 1998b) and generate other individuals randomly. They use classical selection mechanisms, which are roulette wheel and tournament selection. As crossover operation, they propose new operations based on that the common jobs or blocks of two parents are transferred to the children. As mutation operation, Ruiz et al. (2005) use simple node insertion operation by selecting a job randomly and inserting it to a random position in the sequence. After these operations, they replace the population by new individuals if their makespan values are lower than that of the parents in the population. In addition to this classical structure of genetic algorithm, Ruiz et al. (2005) implement a restart scheme to avoid the local optimum. Finally, to compare the performance of the GA (Ruiz et al., 2005) Ruiz and Stützle (2008) used CPU time as the stopping condition.

Memetic Algorithm (MA): To improve the performance of the proposed GA, Ruiz et al. (2005) apply a local search procedure, which is based on node insertion neighborhood, to the solutions, after the crossover and mutation operation. Since this procedure takes too long time in the proposed GA, they implement this local search procedure to the individuals in the population with an “enhancement probability”. Namely, they apply the local search operation to some of the individuals. They also do not conduct a full node insertion operation. The iteration number of the local search procedure is limited by a parameter. Ruiz et al. (2005) calibrate the parameters of the local search procedure to balance the quality of the solution and the computational time spent. This proposed hybrid GA is called as memetic algorithm (MA).

Memetic Algorithm with Modified Local Search (MA_LS): Ruiz and Stützle (2008) replace the applied local search procedure of MA with the descent local search procedure and call it MA_LS. In the descent local search algorithm, Ruiz and Stützle (2008) use the same neighborhood structure, which is node insertion. They remove each job in the sequence one by one and insert into the best position that gives a lower makespan value, among all possible positions. After this procedure is finished for all jobs, if any solution obtained from the local search procedure gives any improvement compared to the current solution, the whole process is repeated, until there is no improvements on the solutions. This procedure leads to a local optimum. Although it takes longer computational time, Ruiz and Stützle (2008) test this modified version of the memetic algorithm (MA_LS) in their study, with the given CPU time.

Ant Colony Optimization Algorithm (PACO): Ant colony optimization algorithm is a population based algorithm but Rajendran and Ziegler (2004) use only one ant for each iteration in their study. They initialize the pheromone trails by well-known NEH heuristic. Then, one ant constructs a full solution by adding components one by one iteratively. After construction, the solution is improved by a local search based on the job-index procedure proposed by Rajendran and Ziegler (2004), which is indeed the same local search procedure used by Ruiz and Stützle (2008). Then, the ant gives feedback about the components; hence accordingly the parameters are updated. The procedure continues until the stopping condition is met. Rajendran and Ziegler (2004) propose this heuristic for regular flowshop problem when the objective is to minimize makespan. Ruiz and Stützle (2008) modify this heuristic to the FS-SDST and change the initialization heuristic to NEH_RMB (Rios-Mercado and Bard, 1998b). Since Ruiz and Stützle (2008) use this heuristic to compare the performance of their proposed algorithm, they use CPU time as the stopping condition for the modified algorithm and call it as PACO.

Iterated Greedy (IG): Iterated greedy algorithm starts with an initial solution and has destruction, construction and acceptance phases in the main loop. Ruiz and Stützle (2008) initialize their proposed IG with well-known constructive heuristic NEH_RMB

(Rios-Mercado and Bard, 1998b). In the destruction phase, some of the components of the solution are removed from the sequence and in the construction phase, each removed components are added to the partial solution one by one to obtain the best permutation, which gives a lower makespan value. The reconstructed solution is compared with the incumbent solution in the acceptance step. As acceptance condition, they use simulated annealing-like acceptance criterion. In this criterion, different from the simple acceptance criterion, which accepts only better solutions, they also accept some of the candidate solutions that have worse fitness value as a new solution. The main loop continues with this new reconstructed solution. This procedure is repeated until the stopping condition is met, which is based on the CPU time.

Iterated Greedy with Local Search (IG_LS): Ruiz and Stützle (2008) improve their proposed IG algorithm with the descent local search procedure based on the node insertion neighborhood, which is explained in the MA_LS. Ruiz and Stützle (2008) apply this local search to the initial solution, which is constructed by NEH_RMB heuristic (Rios-Mercado and Bard, 1998b). Moreover, after the destruction and the construction phases of the IG, the local search is also applied to the reconstructed solution. Lastly, in the acceptance phase, with simulated annealing-like acceptance criterion, the solution obtained from the local search is compared with the incumbent. Among the other benchmarks, this heuristic gives the best results when the stopping condition is the CPU time. The proposed iterated greedy algorithm with local search is the state-of-the-art for the FS-SDST problem when the objective is makespan ($F|s_{ijl}, prmu|C_{max}$) for the published instance set of Ruiz et al. (2005).

Since the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem is studied for the first time in the literature, there are not any benchmarks. To analyze the performance of the proposed VNS algorithm for the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, we solved small-sized problems with the MILP model, which was presented in Section 3.2.2. We compared the solutions of the proposed VNS algorithm with the optimal solutions. For large-sized problems, we compared the VNS results with the results obtained from the well-known NEH constructive heuristic. The NEH algorithm is actually used as an initial solution for the proposed VNS algorithm. Hence, this comparison indicates how much we improve the NEH solutions by the proposed VNS algorithm.

For the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem, we also relaxed some of the binary variables of the MILP model to obtain lower bounds (LB) on the optimal solution to the problem and compared the VNS results with these LBs. In the MILP model, there are three binary variables which are z_{jk} for assigning a job j to a position k, q_{jkl} for indicating two adjacent jobs, job j in position k in the sequence and immediately followed by job l, and v_{ik} for checking whether the idle time is greater than the breakeven duration on machine i before the job in position k. The variable v_{ik} is a positive continuous energy-related variable and it is added to the objective function as negative. Hence, when we relax this binary variable to a continuous variable in the range of [0, 1], the variable v_{ik} takes a high value, which makes the objective function value negative. It should be noted that relaxing other binary variables, that is, z_{jk} and q_{jkl}, does not give a better relaxation.

5.3.3 Performance Measures

To evaluate the results of the proposed VNS algorithm, we need performance measures. For the $F|s_{ijl}$, $prmu|C_{max}$ problem, we use percentage deviation (PD) of the solution obtained by a heuristic algorithm from the best known solution for each instance. The best known solution for each instance can be found at the website http://soa.iti.es/rruiz (Ruiz, 2008). Equation 5.1 shows the calculation of the PD.

$$\mathrm{PD} = \frac{\mathrm{Solution^H} - \mathrm{BestKnown}}{\mathrm{BestKnown}} \text{ x } 100 \qquad (5.1)$$

In Equation 5.1, $\mathrm{Solution^H}$ is the objective function value obtained from heuristic algorithm H for one instance. BestKnown is the best objective function value of that instance known so far. Since there are 480 instances to test for the $F|s_{ijl}$, $prmu|C_{max}$ problem, the results are shown as average percentage deviation (APD). For one instance size group such as 20 jobs to 5 machines, we take the average of the PD results.

For the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem, there are no benchmark sets and best known solutions to compare our result with. However, we find the optimal solutions for the small instances, and we use the NEH heuristic results for the large instances in our comparison. Optimal and NEH results are compared with the proposed VNS solutions and as the performance measure, we also use percentage deviation (PD) method. For the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem, in Equation 5.1, H = VNS, whereas BestKnown is either the optimal solution for small-sized instances or the NEH solution for large-sized instances.

5.3.4 Results for Proposed VNS algorithm

We conducted computational experiments for the $F|s_{ijl}$, $prmu|C_{max}$ and $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problems with the instance sets explained in Section 5.1 and present our results in this section. For the $F|s_{ijl}$, $prmu|C_{max}$ problem, we compare our results with the benchmarks which are given in Section 5.4. Tables 5.6-5.9 indicate the results of the $F|s_{ijl}$, $prmu|C_{max}$ problem as the maximum, the minimum and the average percentage deviation (PD) of the objective function values of the VNS solutions from that of the best known solutions for each instance size. Results are tabulated in the consecutive tables for each SDST group. Bold numbers show our average results and that of the heuristics, which give at least as good as our results. Tables 5.10 and 5.11 indicate the maximum, the minimum and the average percentage deviation (PD) of the objective function values of the VNS solutions from the NEH heuristic according to the SDST groups for the $F|s_{ijl}$, $prmu|C_{max}$ problem, which we will analyze these results to evaluate the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem. For the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem, Tables 5.12 and 5.13 show the average PD of the objective function values of the VNS solutions from the optimal solutions. Tables 5.14-5.17 indicate the average PD of the objective function values of the VNS results from the NEH heuristic results. For detail analysis, we tabulated the maximum, the minimum and the average PD of both total production cost and total energy cost values obtained from VNS algorithm, separately, for the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem in Table A1-36 in Appendix.

Table 5.6 Performance of the benchmarks and the VNS algorithm for the $F|s_{ijl}$, $prmu|C_{max}$ problem when SDST10

SDST10	**GA**	**MA**	**MA_LS**	**PACO**	**IG_RS**	**IG_RS_LS**	**VNS**		
	Avg. (%)	**Avg. (%)**	**Avg. (%)**	**Avg. (%)**	**Avg. (%)**	**Avg. (%)**	**Max.(%)**	**Min.(%)**	**Avg.(%)**
20x5	0.41	0.70	**0.08**	0.18	0.14	**0.04**	*0.42*	*0.00*	**0.08**
20x10	0.56	0.36	**0.13**	0.22	0.24	**0.04**	*0.49*	*0.00*	**0.17**
20x20	0.39	0.56	0.10	0.12	0.19	**0.04**	*0.21*	*0.00*	**0.08**
50x5	0.92	0.77	**0.30**	**0.42**	0.84	**0.27**	*0.84*	*0.41*	**0.58**
50x10	2.01	1.26	**0.81**	1.06	1.43	**0.53**	*1.84*	*0.50*	**1.03**
50x20	2.10	1.28	**0.82**	**1.01**	1.54	**0.60**	*1.54*	*0.62*	**1.18**
100x5	1.03	0.63	**0.31**	0.76	1.34	**0.33**	*0.81*	*0.31*	**0.51**
100x10	1.33	**0.90**	**0.48**	**0.77**	1.32	**0.38**	*1.26*	*0.59*	**0.95**
100x20	1.83	**1.06**	**0.82**	**1.12**	1.47	**0.54**	*1.63*	*0.68*	**1.32**
200x10	1.32	**0.65**	**0.48**	0.85	1.33	**0.32**	*1.16*	*0.27*	**0.65**
200x20	1.71	**0.87**	**0.76**	0.95	1.12	**0.38**	*1.19*	*0.59*	**0.93**
500x20	1.27	0.48	**0.43**	0.61	0.82	**0.21**	*0.69*	*0.23*	**0.43**
Average	1.24	0.79	**0.46**	0.67	0.98	**0.31**	*1.01*	*0.35*	**0.66**

Table 5.7 Performance of the benchmarks and the VNS algorithm for the $F|s_{ijl}, prmu|C_{max}$ problem when SDST50

SDST50	GA	MA	MA_LS	PACO	IG_RS	IG_RS_LS	VNS		
	Avg. (%)	**Avg. (%)**	**Avg. (%)**	**Avg. (%)**	**Avg. (%)**	**Avg. (%)**	**Max.(%)**	**Min.(%)**	**Avg.(%)**
20x5	1.15	1.50	**0.30**	0.51	0.58	**0.10**	*0.84*	*0.00*	**0.39**
20x10	1.17	0.77	**0.32**	0.44	0.58	**0.19**	*0.84*	*0.00*	**0.36**
20x20	0.49	0.78	**0.16**	0.25	0.37	**0.07**	*0.37*	*0.00*	**0.16**
50x5	3.43	2.18	**1.13**	1.98	2.42	**1.04**	*2.71*	*0.66*	**1.90**
50x10	3.01	**1.68**	**1.08**	**1.62**	2.12	**0.92**	*2.39*	*1.52*	**1.92**
50x20	2.43	1.69	**0.89**	**1.28**	2.03	**0.82**	*2.43*	*0.66*	**1.41**
100x5	3.98	**2.34**	**1.38**	3.95	**2.33**	**1.09**	*3.52*	*1.47*	**2.52**
100x10	3.07	**1.52**	**1.21**	3.10	2.13	**0.88**	*2.39*	*0.99*	**1.72**
100x20	2.51	**1.54**	**1.03**	2.45	1.82	**0.81**	*2.60*	*0.92*	**1.74**
200x10	3.49	**1.35**	**1.21**	3.37	1.90	**0.63**	*1.98*	*0.85*	**1.48**
200x20	2.67	**1.19**	**1.02**	2.64	1.51	**0.53**	*1.53*	*0.80*	**1.24**
500x20	2.07	0.76	0.79	2.00	1.28	**0.31**	*1.02*	*0.44*	**0.72**
Average	2.46	1.44	**0.88**	1.97	1.59	**0.62**	*1.89*	*0.69*	**1.30**

Table 5.8 Performance of the benchmarks and the VNS algorithm for the $F|s_{ijl}, prmu|C_{max}$ problem when SDST100

SDST100	GA	MA	MA_LS	PACO	IG_RS	IG_RS_LS	VNS		
	Avg. (%)	**Avg. (%)**	**Avg. (%)**	**Avg. (%)**	**Avg. (%)**	**Avg. (%)**	**Max.(%)**	**Min.(%)**	**Avg.(%)**
20x5	1.82	1.43	**0.39**	0.61	1.24	**0.17**	*1.69*	*0.00*	**0.57**
20x10	1.27	1.09	**0.29**	0.48	1.03	**0.18**	*1.27*	*0.00*	**0.42**
20x20	0.94	1.14	**0.17**	0.48	0.74	**0.17**	*0.89*	*0.00*	**0.31**
50x5	5.26	3.02	**1.99**	3.31	3.70	**1.82**	*4.71*	*1.74*	**2.72**
50x10	4.18	2.55	**1.50**	2.49	2.99	**1.30**	*3.33*	*0.80*	**2.20**
50x20	3.11	**1.77**	**1.18**	1.98	2.40	**1.11**	*2.74*	*0.95*	**1.96**
100x5	6.00	**3.04**	**2.16**	6.65	3.48	**1.63**	*4.64*	*2.24*	**3.28**
100x10	4.15	**2.45**	**1.61**	4.89	2.77	**1.02**	*3.50*	*1.94*	**2.75**
100x20	3.49	2.39	**1.53**	3.91	2.46	**1.05**	*3.07*	*1.23*	**2.16**
200x10	4.71	2.19	**1.77**	5.53	2.49	**0.92**	*3.12*	*1.56*	**2.17**
200x20	3.48	1.68	**1.40**	3.82	1.92	**0.76**	*2.30*	*1.18*	**1.66**
500x20	2.64	1.16	1.14	2.75	1.50	**0.46**	*1.67*	*0.20*	**0.91**
Average	3.42	1.99	**1.26**	3.08	2.23	**0.88**	*2.75*	*0.99*	**1.76**

Table 5.9 Performance of the benchmarks and the VNS algorithm for the $F|s_{ijl}, prmu|C_{max}$ problem when SDST125

SDST125	GA	MA	MA_LS	PACO	IG_RS	IG_RS_LS	VNS		
	Avg. (%)	Avg. (%)	Avg. (%)	Avg. (%)	Avg. (%)	Avg. (%)	Max.(%)	Min.(%)	Avg.(%)
20x5	1.90	1.40	**0.32**	0.65	1.24	**0.30**	*1.84*	*0.00*	**0.40**
20x10	1.52	1.24	**0.37**	**0.56**	1.44	**0.36**	*1.24*	*0.00*	**0.64**
20x20	0.95	1.21	**0.24**	**0.39**	0.81	**0.19**	*1.42*	*0.00*	**0.41**
50x5	5.63	**3.48**	**1.97**	**3.67**	4.00	**2.01**	*5.82*	*1.65*	**3.80**
50x10	4.59	3.35	**1.50**	2.96	3.47	**1.54**	*4.02*	*0.62*	**2.62**
50x20	3.25	**1.63**	**1.26**	**2.06**	2.59	**1.18**	*3.43*	*0.93*	2.06
100x5	6.82	**3.65**	**2.52**	7.75	4.14	**1.91**	*6.07*	*2.56*	**4.08**
100x10	4.80	**2.84**	**1.94**	5.61	3.26	**1.34**	*4.21*	*1.46*	**2.89**
100x20	3.50	**2.16**	**1.50**	4.15	2.60	**1.00**	*3.04*	*1.23*	**2.23**
200x10	5.37	2.63	**2.14**	6.20	2.94	**1.17**	*3.49*	*1.62*	**2.61**
200x20	3.69	1.69	1.49	4.16	2.24	**0.76**	*1.90*	*0.84*	**1.35**
500x20	2.83	1.36	1.23	3.02	1.64	**0.52**	*1.21*	*0.55*	**0.93**
Average	3.74	2.22	**1.37**	3.43	2.53	**1.02**	*3.14*	*0.96*	**2.00**

Table 5.10 Percentage deviation of the VNS results from the NEH solutions for the $F|s_{ijl}, prmu|C_{max}$ problem when SDST10 and SDST50

SDST10	Max. (%)	Min.(%)	Avg.(%)	*SDST50*	Max.(%)	Min.(%)	Avg.(%)
20x5	-4.90	-1.68	-3.47	20x5	-9.20	-3.20	-6.23
20x10	-5.40	-2.96	-3.99	20x10	-8.61	-4.35	-5.85
20x20	-7.28	-1.95	-3.70	20x20	-5.94	-2.88	-4.39
50x5	-3.63	-1.76	-2.54	50x5	-8.69	-4.45	-6.04
50x10	-6.15	-3.33	-4.17	50x10	-6.76	-2.60	-5.17
50x20	-5.92	-2.69	-3.94	50x20	-5.95	-3.37	-4.36
100x5	-3.18	-2.38	-2.69	100x5	-7.27	-4.11	-5.35
100x10	-3.65	-2.09	-2.66	100x10	-5.60	-3.65	-4.68
100x20	-4.08	-2.29	-2.99	100x20	-4.44	-2.84	-3.65
200x10	-2.49	-1.74	-2.12	200x10	-4.05	-2.94	-3.66
200x20	-2.78	-1.77	-2.22	200x20	-3.14	-2.35	-2.73
500x20	-1.61	-0.90	-1.28	500x20	-1.95	-1.61	-1.78
Average	-4.26	-2.13	-2.98	Average	-5.97	-3.20	-4.49

Table 5.11 Percentage deviation of the VNS results from the NEH solutions for the $F|s_{ijl}$, $prmu|C_{max}$ problem when SDST100 and SDST125

SDST100	**Max.(%)**	**Min.(%)**	**Avg.(%)**	*SDST125*	**Max.(%)**	**Min.(%)**	**Avg.(%)**
20x5	-11.36	-6.25	-9.07	20x5	-13.32	-4.33	-9.12
20x10	-7.83	-5.34	-6.37	20x10	-9.11	-3.39	-6.68
20x20	-6.42	-3.70	-4.78	20x20	-6.70	-1.93	-4.75
50x5	-10.50	-5.66	-8.18	50x5	-10.46	-6.01	-9.07
50x10	-8.91	-5.44	-6.83	50x10	-10.45	-4.74	-7.10
50x20	-6.74	-4.20	-5.28	50x20	-8.29	-3.98	-6.51
100x5	-8.37	-6.02	-7.13	100x5	-9.17	-7.07	-8.21
100x10	-6.67	-3.53	-5.54	100x10	-8.54	-3.39	-5.71
100x20	-5.67	-3.33	-4.40	100x20	-5.84	-2.83	-4.60
200x10	-6.10	-3.46	-4.77	200x10	-5.98	-4.59	-5.10
200x20	-4.48	-2.53	-3.36	200x20	-4.30	-2.97	-3.75
500x20	-3.08	-1.92	-2.50	500x20	-3.06	-2.37	-2.71
Average	-7.18	-4.28	-5.68	Average	-7.94	-3.97	-6.11

Table 5.12 Percentage deviation of the VNS results from the optimal solutions for the small-sized instances of $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	10x2 Avg. (%)	10x3 Avg. (%)	10x4 Avg. (%)	10x5 Avg. (%)	15x2 Avg. (%)	15x3 Avg. (%)
SDST10	Low	0.1	0.9	0.70	0.54	1.62	2.76	0.31	2.18
		0.5	0.5	0.00	0.00	0.08	0.19	0.00	0.11
		0.9	0.1	0.00	0.00	0.00	0.02	0.00	0.08
		1.0	0.0	0.00	0.00	0.00	0.00	0.06	0.05
	Medium	0.1	0.9	2.53	8.87	14.59	18.31	5.03	6.67
		0.5	0.5	0.88	0.79	1.72	3.11	0.28	0.94
		0.9	0.1	0.00	0.02	0.00	0.30	0.04	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.10
	High	0.1	0.9	4.21	12.31	19.79	25.88	7.21	7.68
		0.5	0.5	2.06	2.08	3.48	4.78	1.02	1.79
		0.9	0.1	0.00	0.01	0.05	0.41	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.04
SDST50	Low	0.1	0.9	0.04	0.50	1.81	1.98	1.91	1.35
		0.5	0.5	0.08	0.04	0.10	0.19	0.37	0.68
		0.9	0.1	0.01	0.00	0.02	0.02	0.44	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.01	0.00
	Medium	0.1	0.9	2.72	5.26	9.27	10.06	3.73	3.52
		0.5	0.5	0.25	0.61	1.75	1.94	0.19	1.02
		0.9	0.1	0.00	0.00	0.02	0.16	0.66	0.19
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.00
	High	0.1	0.9	4.10	5.84	10.90	11.71	5.64	6.27
		0.5	0.5	0.44	1.48	3.43	3.23	1.24	2.09
		0.9	0.1	0.00	0.00	0.10	0.23	0.11	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.62	0.00

Table 5.13 Percentage deviation of the VNS results from the optimal solutions for the small-sized instances of $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	10x2 Avg. (%)	10x3 Avg. (%)	10x4 Avg. (%)	10x5 Avg. (%)	15x2 Avg. (%)	15x3 Avg. (%)
SDST100	Low	0.1	0.9	0.43	1.15	0.81	0.99	0.94	0.34
		0.5	0.5	0.01	0.06	0.09	0.06	0.00	0.36
		0.9	0.1	0.00	0.01	0.01	0.01	0.00	0.10
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.06
	Medium	0.1	0.9	2.05	3.70	3.82	5.44	3.44	3.97
		0.5	0.5	0.13	0.52	0.59	1.04	2.62	1.58
		0.9	0.1	0.00	0.05	0.11	0.04	0.73	0.16
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.08
	High	0.1	0.9	2.26	4.70	4.26	6.11	3.70	5.89
		0.5	0.5	0.70	1.25	1.30	2.39	0.82	1.44
		0.9	0.1	0.00	0.08	0.19	0.07	0.00	0.20
		1.0	0.0	0.00	0.00	0.00	0.00	0.84	0.20
SDST125	Low	0.1	0.9	0.48	0.33	1.63	1.19	0.24	1.10
		0.5	0.5	0.01	0.61	0.06	0.16	0.00	0.43
		0.9	0.1	0.00	0.00	0.01	0.02	1.24	0.38
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.16
	Medium	0.1	0.9	0.70	3.50	4.43	3.82	4.03	3.57
		0.5	0.5	0.05	1.13	0.71	0.97	2.30	0.75
		0.9	0.1	0.00	0.00	0.06	0.18	0.00	0.46
		1.0	0.0	0.00	0.00	0.00	0.00	0.94	0.01
	High	0.1	0.9	1.56	4.02	4.32	4.58	4.24	2.60
		0.5	0.5	0.13	1.27	1.69	1.56	2.67	0.99
		0.9	0.1	0.00	0.04	0.15	0.26	0.08	0.21
		1.0	0.0	0.00	0.00	0.00	0.00	0.38	0.01

Table 5.14 Percentage deviation of the VNS results from the NEH solutions for the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem with 20 and 50 jobs to 5, 10 and 20 machines for SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	20x5 Avg. (%)	20x10 Avg. (%)	20x20 Avg. (%)	50x5 Avg. (%)	50x10 Avg. (%)	50x20 Avg. (%)
SDST10	Low	0.1	0.9	-7.12	-6.15	-4.49	-6.66	-4.93	-4.17
		0.5	0.5	-4.34	-4.64	-3.00	-6.89	-4.42	-3.26
		0.9	0.1	-4.46	-4.38	-3.25	-5.90	-4.29	-4.19
		1.0	0.0	-4.51	-4.20	-3.24	-5.58	-4.33	-4.05
	Medium	0.1	0.9	-12.12	-8.74	-7.29	-8.78	-7.09	-5.47
		0.5	0.5	-6.29	-6.18	-4.54	-6.65	-4.83	-3.63
		0.9	0.1	-4.51	-4.32	-3.07	-5.96	-4.50	-3.49
		1.0	0.0	-4.47	-4.26	-3.18	-5.60	-4.01	-4.03
	High	0.1	0.9	-11.39	-8.52	-6.95	-8.09	-7.77	-6.05
		0.5	0.5	-8.16	-7.82	-4.91	-7.26	-5.29	-4.44
		0.9	0.1	-4.52	-4.43	-3.32	-6.03	-4.55	-3.64
		1.0	0.0	-4.43	-4.18	-3.25	-5.87	-4.17	-4.06
SDST50	Low	0.1	0.9	-9.83	-6.60	-3.78	-6.95	-5.21	-3.88
		0.5	0.5	-6.03	-4.52	-3.29	-6.53	-4.22	-3.32
		0.9	0.1	-5.98	-4.93	-3.01	-6.20	-3.97	-3.82
		1.0	0.0	-5.91	-5.29	-3.34	-6.29	-4.43	-3.27
	Medium	0.1	0.9	-10.17	-6.48	-5.16	-8.82	-5.50	-4.08
		0.5	0.5	-7.78	-5.88	-4.05	-7.08	-4.33	-3.15
		0.9	0.1	-6.40	-4.68	-3.56	-6.91	-4.42	-3.14
		1.0	0.0	-6.08	-5.40	-3.29	-6.10	-4.20	-3.71
	High	0.1	0.9	-11.07	-7.21	-5.16	-9.35	-6.40	-3.78
		0.5	0.5	-10.24	-6.67	-4.06	-7.29	-5.83	-3.63
		0.9	0.1	-5.49	-4.54	-3.68	-6.70	-4.22	-2.92
		1.0	0.0	-6.13	-5.33	-3.15	-6.04	-4.17	-3.50

Table 5.15 Percentage deviation of the VNS results from the NEH solutions for the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem with 20 and 50 jobs to 5, 10 and 20 machines for SDST100 and SDST125

Setup Group	En. Data Group	w1 (%)	w2 (%)	20x5 Avg. (%)	20x10 Avg. (%)	20x20 Avg. (%)	50x5 Avg. (%)	50x10 Avg. (%)	50x20 Avg. (%)
SDST100	Low	0.1	0.9	-9.07	-6.12	-4.07	-7.74	-5.08	-3.64
		0.5	0.5	-7.34	-5.52	-3.22	-7.66	-5.32	-3.53
		0.9	0.1	-8.21	-6.30	-4.20	-7.34	-4.49	-3.87
		1.0	0.0	-7.98	-5.45	-4.38	-6.94	-5.15	-3.54
	Medium	0.1	0.9	-9.13	-5.99	-4.39	-6.55	-4.66	-3.72
		0.5	0.5	-9.08	-5.27	-3.83	-7.29	-5.38	-3.07
		0.9	0.1	-7.98	-4.76	-3.63	-8.25	-4.48	-3.53
		1.0	0.0	-8.09	-5.95	-4.32	-7.92	-5.16	-3.68
	High	0.1	0.9	-11.88	-6.26	-4.75	-8.89	-5.40	-4.19
		0.5	0.5	-9.72	-5.20	-3.78	-6.87	-4.95	-3.45
		0.9	0.1	-8.47	-5.37	-3.48	-8.00	-4.96	-3.79
		1.0	0.0	-7.97	-5.91	-4.29	-6.87	-4.64	-4.02
SDST125	Low	0.1	0.9	-10.47	-6.22	-3.93	-7.48	-4.84	-3.26
		0.5	0.5	-8.48	-6.50	-4.02	-7.93	-5.06	-3.07
		0.9	0.1	-8.53	-6.75	-4.51	-9.06	-4.84	-3.84
		1.0	0.0	-7.84	-7.12	-4.34	-7.45	-5.60	-3.70
	Medium	0.1	0.9	-12.30	-5.79	-3.86	-7.42	-6.33	-3.47
		0.5	0.5	-9.79	-4.94	-3.90	-7.92	-5.54	-3.12
		0.9	0.1	-8.61	-4.52	-3.21	-7.96	-5.65	-3.29
		1.0	0.0	-8.09	-7.33	-4.50	-7.87	-6.29	-3.35
	High	0.1	0.9	-12.57	-5.93	-4.21	-9.07	-5.61	-3.54
		0.5	0.5	-10.92	-5.84	-3.93	-8.12	-4.68	-3.35
		0.9	0.1	-9.68	-5.43	-4.16	-7.20	-4.71	-3.52
		1.0	0.0	-7.96	-7.08	-4.38	-6.72	-5.11	-3.50

Table 5.16 Percentage deviation of the VNS results from the NEH solutions for the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem with 100 jobs to 5, 10 and 20 machines, 200 jobs to 10 and machines and 500 jobs to 20 machines for SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	100x5 Avg. (%)	100x10 Avg. (%)	100x20 Avg. (%)	200x10 Avg. (%)	200x20 Avg. (%)	500x20 Avg. (%)
SDST10	Low	0.1	0.9	-5.21	-2.97	-2.49	-2.18	-1.42	-0.30
		0.5	0.5	-5.74	-3.29	-2.50	-2.46	-1.39	-0.42
		0.9	0.1	-5.75	-3.10	-2.24	-2.10	-1.85	-0.45
		1.0	0.0	-4.41	-3.42	-2.16	-2.45	-1.20	-0.53
	Medium	0.1	0.9	-6.30	-3.97	-3.21	-2.17	-2.03	-0.26
		0.5	0.5	-6.42	-3.23	-2.82	-2.14	-1.69	-0.44
		0.9	0.1	-5.72	-3.36	-1.97	-2.05	-1.45	-0.27
		1.0	0.0	-4.29	-3.46	-1.74	-2.33	-1.41	-0.51
	High	0.1	0.9	-7.05	-4.50	-3.26	-2.12	-1.75	-0.46
		0.5	0.5	-5.85	-2.84	-2.32	-1.98	-1.34	-0.41
		0.9	0.1	-4.96	-3.35	-2.48	-2.34	-1.60	-0.34
		1.0	0.0	-4.68	-3.63	-2.40	-2.25	-1.50	-0.40
SDST50	Low	0.1	0.9	-5.38	-2.77	-1.91	-1.69	-0.66	-0.28
		0.5	0.5	-5.61	-2.88	-1.97	-1.94	-1.39	-0.38
		0.9	0.1	-5.05	-2.68	-2.27	-1.97	-1.17	-0.34
		1.0	0.0	-4.95	-3.44	-1.92	-2.09	-1.27	-0.33
	Medium	0.1	0.9	-7.10	-3.66	-2.70	-2.49	-1.35	-0.16
		0.5	0.5	-5.24	-2.73	-2.13	-1.75	-1.36	-0.20
		0.9	0.1	-5.59	-2.71	-2.14	-2.08	-1.35	-0.38
		1.0	0.0	-4.08	-3.65	-1.65	-2.17	-1.17	-0.34
	High	0.1	0.9	-6.98	-3.92	-2.73	-2.00	-1.25	-0.15
		0.5	0.5	-6.08	-2.70	-2.29	-1.94	-1.03	-0.27
		0.9	0.1	-5.49	-2.62	-1.97	-1.43	-1.36	-0.24
		1.0	0.0	-4.65	-3.23	-2.35	-2.09	-1.01	-0.24

Table 5.17 Percentage deviation of the VNS results from the NEH solutions for the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem with 100 jobs to 5, 10 and 20 machines, 200 jobs to 10 and machines and 500 jobs to 20 machines for SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	100x5 Avg. (%)	100x10 Avg. (%)	100x20 Avg. (%)	200x10 Avg. (%)	200x20 Avg. (%)	500x20 Avg. (%)
SDST100	Low	0.1	0.9	-6.09	-2.90	-1.88	-1.09	-1.25	-0.15
		0.5	0.5	-6.00	-3.20	-2.30	-0.96	-0.92	-0.10
		0.9	0.1	-6.66	-3.24	-2.18	-1.31	-1.48	-0.25
		1.0	0.0	-5.15	-3.75	-2.56	-2.19	-1.31	-0.31
	Medium	0.1	0.9	-6.50	-3.43	-2.25	-1.36	-1.19	-0.09
		0.5	0.5	-6.03	-2.64	-1.87	-1.04	-1.01	-0.17
		0.9	0.1	-6.32	-3.34	-2.09	-0.99	-1.01	-0.34
		1.0	0.0	-5.81	-3.99	-2.03	-2.10	-1.40	-0.38
	High	0.1	0.9	-5.68	-3.10	-2.35	-1.21	-1.14	-0.18
		0.5	0.5	-5.47	-3.50	-2.72	-1.14	-0.91	-0.19
		0.9	0.1	-6.00	-3.09	-2.31	-0.74	-0.89	-0.24
		1.0	0.0	-5.41	-4.00	-2.35	-2.34	-1.06	-0.39
SDST125	Low	0.1	0.9	-6.42	-3.39	-2.03	-0.30	-0.84	-0.02
		0.5	0.5	-5.90	-3.55	-1.99	-0.38	-1.15	-0.13
		0.9	0.1	-6.62	-3.93	-2.19	-0.65	-1.56	-0.26
		1.0	0.0	-5.13	-3.40	-2.30	-2.43	-1.29	-0.55
	Medium	0.1	0.9	-6.11	-3.42	-2.30	-0.37	-1.21	-0.18
		0.5	0.5	-5.86	-2.80	-2.08	-0.28	-1.46	-0.24
		0.9	0.1	-6.92	-3.17	-2.35	-0.60	-1.29	-0.26
		1.0	0.0	-5.36	-3.06	-2.35	-2.53	-1.42	-0.22
	High	0.1	0.9	-6.79	-3.27	-2.40	-0.21	-1.48	-0.35
		0.5	0.5	-5.79	-2.91	-1.99	-0.36	-1.21	-0.11
		0.9	0.1	-6.71	-3.47	-1.86	-0.44	-1.05	-0.05
		1.0	0.0	-5.16	-3.61	-2.03	-2.51	-1.44	-0.42

5.3.5 Trade-off Between the Total Production Cost and the Total Energy Cost

To observe the impact of the energy cost objective in addition to the total production cost for the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem, we tabulate the production cost, the energy cost and the total cost values separately in Table 5.18. We used ten instances with 50 jobs and 10 machines. We select the sequence dependent setup time values as SDST50 and for the energy-related parameters, we select medium range group. We will discuss the results in Section 5.4.2.

Table 5.18 Production cost and energy cost values for the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem with 50 jobs to 10 machines

Ins.	w_1 (%)	w_2 (%)	Production Cost	Energy Cost	Total Cost	Ins.	w_1 (%)	w_2 (%)	Production Cost	Energy Cost	Total Cost
Instance 1	0.1	0.9	674930	345480	378425	Instance 6	0.1	0.9	643125	341330	371510
	0.5	0.5	603775	378529	491152		0.5	0.5	605255	364258	484757
	0.9	0.1	586990	403152	568606		0.9	0.1	586465	418091	569628
	1.0	0.0	583905	424416	583905		1.0	0.0	579805	436164	579805
Instance 2	0.1	0.9	657915	341087	372770	Instance 7	0.1	0.9	664260	359073	389592
	0.5	0.5	577540	372449	474995		0.5	0.5	604065	393594	498830
	0.9	0.1	558030	404851	542712		0.9	0.1	590415	433339	574707
	1.0	0.0	566540	429647	566540		1.0	0.0	582425	444452	582425
Instance 3	0.1	0.9	618830	356234	382494	Instance 8	0.1	0.9	662295	357041	387566
	0.5	0.5	578935	368908	473922		0.5	0.5	588120	386021	487071
	0.9	0.1	546450	412006	533006		0.9	0.1	582050	418120	565657
	1.0	0.0	550485	432083	550485		1.0	0.0	580355	443575	580355
Instance 4	0.1	0.9	636100	344286	373467	Instance 9	0.1	0.9	642030	337323	367794
	0.5	0.5	591845	366356	479101		0.5	0.5	590530	347636	469083
	0.9	0.1	585215	403684	567062		0.9	0.1	578595	402438	560979
	1.0	0.0	584265	447127	584265		1.0	0.0	574725	405950	574725
Instance 5	0.1	0.9	669090	356331	387607	Instance 10	0.1	0.9	674330	340492	373876
	0.5	0.5	594445	389206	491826		0.5	0.5	600990	365933	483462
	0.9	0.1	576565	429527	561861		0.9	0.1	592380	411946	574337
	1.0	0.0	590395	453894	590395		1.0	0.0	593140	424523	593140

5.4 Analysis of the Results

In this section, we summarize the results of the computational studies which are tabulated in Tables 5.6-5.17 and analyze them separately for the $F|s_{ijl}$, $prmu|C_{max}$ and $F|s_{ijl}$,

$prmu|\sum(C_j+Energy_j)$ problems. Our main discussion will be on the average percentage deviation (PD) values of objective functions for both two problems. In addition, we will also discuss the percentage deviation (PD) of the total production cost and total energy cost values from the optimal and NEH values for the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, which are tabulated in Table A1-36 in Appendix.

5.4.1 Analysis of the Results for the $F|s_{ijl}, prmu|C_{max}$ Problem

We observe from the results given in Tables 5.6-5.9 that the proposed VNS algorithm outperforms the genetic algorithm (GA) for each SDST group and each instance size. The reason of the low performance of the GA can be due to combining several solutions with the crossover operation, which may lead too much diversification. With the local search procedure of the proposed VNS algorithm, we explore all solutions in a smaller search space dictated by the neighborhood structure used, which leads to intensification and an increase in the quality of the solution.

The advanced genetic algorithm with local search procedure, called memetic algorithm (MA), improves the results for most of the instance sizes for each SDST group, when we compare with the results of GA. For larger instance sizes, some results of MA give better results than the VNS algorithm. The reason for the better performance of MA in some of the instances can be due to the local search strategy of MA. Ruiz et al. (2005) use this local search procedure with limited iteration number, which decreases the consumed time in local search and increase the iteration number in genetic algorithm part. In Table 5.7, half of the instance size groups outperform the proposed VNS algorithm when the setup time is SDST50. However, on the average, for each SDST groups, proposed VNS algorithm shows a better performance compared to MA as can be seen in Table 5.6-5.9.

When Ruiz and Stützle (2008) replace the applied local search procedure in MA with the descent local search procedure, called MA_LS, they improve the solution quality according to MA. From Tables 5.6-5.9, it is observed that the average percentage deviation

(APD) of the MA_LS results from the best known solutions are lower than the APD of the proposed VNS algorithm for each SDST groups. However, for the largest instance 500x20, the VNS algorithm gives better results than the MA_LS algorithm in SDST50, SDST100 and SDST125 group of setup times, as can be seen in Tables 5.7, 5.8 and 5.9, respectively. Even though the local search procedure is the same in both VNS and MA_LS algorithms, MA_LS shows a better performance overall than the proposed VNS algorithm. One of the reasons may be that MA_LS is an evolutionary algorithm and the solutions are evolved through the iterations. On the other, in shaking part of the VNS, the solution is generated randomly, and then this solution is improved with the local search procedure. Hence, there are two different strategies used in MA_LS and VNS algorithms: intensification and diversification, respectively. For the $F|s_{ijl}, prmu|C_{max}$ problem, the computational results show that MA with descent local search (MA_LS) works better than VNS. Additionally, since the stopping condition is CPU time, coding the algorithm has an important role. Using an additional speed-up in MA_LS may affect the performance of the algorithm.

The PACO algorithm has a lower APD value than the proposed VNS for the instance sizes 50x5, 50x20, 100x10 and 100x20, as it can be observed in Table 5.6. In Table 5.7, there are only two instance groups, 50x10 and 50x20, where PACO algorithm performs better results than the VNS algorithm. Table 5.9 shows that for the largest SDST group, 20x10, 20x20, 50x5 and 50x20, PACO algorithm achieves lower APD value than VNS algorithm. However, for each SDST group, the proposed VNS algorithm has a better performance on the average. Since PACO also uses the same local search procedure with MA_LS, IG_LS and the proposed VNS, PACO is outperformed by all of these three algorithms on the average. The reason of the weak performance of the PACO may be due to its feedback mechanism, which is based on the job position while constructing a solution. This information does not carry the successor or predecessor relationship between the adjacent jobs. However, this flowshop problem is affected by the sequence dependent setup times (SDST). On the other hand, proposed VNS has a setup-dependent neighborhood structure. As a result, after the same local search part, VNS shows a better performance than the PACO.

The VNS algorithm outperforms the IG algorithm in each instance size of each SDST group, except one instance, as shown in Tables 5.6-5.9. In Table 5.7, only the APD of the IG for instance 100x5 has a lower value than the VNS algorithm. The structure of the IG is based on node insertion. Some of the components of the solution are removed from the sequence and inserted into the best position that gives a lower makespan value. The local search procedure of the proposed VNS algorithm is also based on node insertion. However, different from IG algorithm, instead of selecting some of the jobs in the sequence we remove each job in the sequence one by one and insert into the best position that gives a lower makespan value, among all possible positions.

The hybrid version of IG with the local search procedure, IG_LS, is the state-of-the art for the $F|s_{ijl}$, $prmu|C_{max}$ problem. IG_LS algorithm outperforms the proposed VNS algorithm in each SDST group and instance as can be seen in Tables 5.6-5.9. It is observed that adding the local search procedure to the IG algorithm improves the quality of the algorithm significantly. In general, IG_LS is based on the node insertion operation in both the main part and the local search part of the algorithm. The reason of the power of the IG_LS algorithm may be that Ruiz and Stützle (2008) use less diversification techniques in their algorithm compared to the proposed VNS algorithm. Ruiz and Stützle (2008) only select the jobs from the sequence randomly, but they place these jobs in a logical way. On the other hand, in shaking part of the VNS, we move to a random solution in the neighborhood and considering this solution as the initial solution, we explore the best solution in its neighborhood in the local search part. Moreover, as we discuss in MA_LS and VNS comparison, coding the algorithm has an important role, since the stopping condition is CPU time. Ruiz and Stützle (2008) use this local search procedure with MA and IG, and we observe that MA_LS and IG_LS algorithms have better performance than VNS algorithm. Ruiz and Stützle (2008) may use an additional speed-up in their local search procedure and it may lead to having more iterations, which may increase the quality of the solutions.

To conclude, we can infer the following conclusions for the $F|s_{ijl}$, $prmu|C_{max}$ problem from the results given in Tables 5.6-5.9:

- The node insertion neighborhood structure is a powerful move operation for the $F|s_{ijl}$, $prmu|C_{max}$ problem. Especially, using this neighborhood in a local search procedure improves the performance of an existing algorithm significantly, as indicated by the results of MA_LS and IG_LS.
- When the same local search procedure is used in two different heuristic algorithms, then the structure of the main algorithm becomes important, which determines the initial solution for the local search. We observe that while constructing or improving a solution in the heuristic, we should consider the successor and predecessor relationship between jobs in the algorithm. Namely, the problem-specific operations may result in a better performance as seen in PACO and VNS comparison. On the other hand, using more intensified strategies such as evolutionary algorithms or iterated greedy algorithm instead of generating random solutions, such as the shaking part of the VNS algorithm, may lead the solution to converge in a better value, since the stopping condition is CPU time (See the analysis of VNS with MA_LS and IG_LS).
- In the proposed VNS algorithm, we observe that the solution is mostly improved by the local search phase and most of the CPU time is consumed in that phase. We utilize the Taillard's speed up techniques in the computation of the local search phase in our proposed VNS algorithm as explained in Section 4.2.2. Ruiz and Stützle [22] do not declare their speed up procedure in their article but the results imply that they also use some powerful speed up techniques. Since the stopping criterion is CPU time, the speed ups and the coding skills play an important role in the heuristic algorithms.

Tables 5.10 and 5.11 present the maximum, the minimum and the average percentage deviation (PD) of the VNS results from the NEH heuristic for the $F|s_{ijl}$, $prmu|C_{max}$ problem according to the SDST groups. Since there are no benchmark data for the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem, we compare the VNS results with the NEH heuristic results for the large-sized instances. To analyze these APD values for the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem in a fair manner, we also obtained the APD values for the $F|s_{ijl}$, $prmu|C_{max}$ problem to observe the improvement. The proposed VNS algorithm has already been initialized by the solution which is constructed by NEH heuristic. Hence, the result of the proposed VNS algorithm will give at least the same value of the NEH heuristic but expectedly a lower objective function value. As a result, the PD values in Tables 5.10 and 5.11 are negative, which indicate that there are improvements in the solution quality. In Table 5.10 and 5.11, the maximum and the minimum percentage values are considered as absolute values. These tables indicate how much we improve the NEH solutions by the proposed VNS algorithm. We observe that when the SDST values increase the absolute values of the average percentage deviation (APD) also increase. It means that the proposed VNS algorithm improves the solutions more when there are higher setup time values. The reason can be that the proposed VNS has a setup-dependent neighborhood structure. When the setup time value increases, the improvement gives a higher deviation. We will analyze the results given in Tables 5.10 and 5.11 to evaluate the performance of the VNS algorithm for the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem in the following subsection.

5.4.2 Analysis of the Results for the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ Problem

We study the $F|s_{ijl}$, $prmu|\sum(C_j+Energy_j)$ problem for the first time in the literature. Hence, firstly we conduct the preliminary experiments for small-sized instances. To analyze the performance of the VNS heuristic, we compare the heuristic results with the optimal solutions which are obtained by CPLEX. Tables 5.12 and 5.13 display the percentage deviation (PD) of VNS results from the optimal solutions for the $F|s_{ijl}$,

$prmu|\sum(C_j+Energy_j)$ problem. The results are tabulated according to the instance size, setup groups, energy-related data groups and weighted sum parameters. We find the optimal solutions for the problem up to 10 jobs with 5 machines and 15 jobs with 3 machines in reasonable time. We also conduct some experiments on 20 jobs with 2 machines and 15 jobs with 4 machines. However, we observe that when the weighted sum parameter of the total production cost w_1 (due to the total completion time) is 0.1 and the weighted sum parameter of the total energy cost w_2 is 0.9, these problems cannot be solved optimally within one hour. We limited the time for CPLEX as one hour in our study, since we aim to have this comparison as a preliminary work for the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem.

In Tables 5.12 and 5.13, we expect to find the optimal solutions by the proposed VNS algorithm, since the size of the problems is small. However, we observe that we reach the optimal solutions with the proposed VNS algorithm mostly in instances 10x2 and 15x2. On the other hand, except from some of instances, we obtain close results to the optimal solutions. The reason of the deviation is that we consider the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem without inserted idle time during scheduling in the VNS algorithm. Our solution representation is permutation of the jobs in the algorithm and we schedule the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem according to the this permutation. Since we minimize a non-regular objective function in the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, to find the global optimum we should use the inserted idle time. However, since it leads to high time consumption in the proposed VNS algorithm, we study this problem without inserted idle time in the proposed heuristic.

In Tables 5.12 and 5.13, it is observed that the average percentage deviation of the proposed VNS algorithm solution from the optimal solution is higher when the data of the energy-related parameter is high, for each instance size and setup time group. Moreover, for each instance group, when the weighted sum parameter of the total production cost w_1 (due to the total completion time) is higher than the weighted sum parameter of the total energy cost w_2, the results of the VNS heuristic have closer values to the optimal solutions.

Additionally, these problems are solved in CPLEX in shorter time than the problems which have higher w_2 value. Namely, when the multi-objective problem is closer to the total production cost problem, we obtain better results with the VNS algorithm in a shorter time. On the other hand, when we increase the importance of the energy objective, the problem gets harder. These observations are expected because when the effect of the energy-related objective increases in the objective function value by the energy data values or the weighted sum parameter, the problem gets more complex. The reason is that when the objective function is only to minimize the total production cost, which is a regular objective function, the proposed VNS algorithm performs better solutions. On the other hand, when we increase the importance of the energy objective in the objective function, which is a non-regular objective function, inserting idle time into the schedule may give better results.

In Tables 5.14-5.17, the percentage deviation values are negative similar to the results given in Tables 5.10 and 5.11, which means that there is an improvement over the NEH solutions. The values are tabulated as their absolute values under the maximum and the minimum PD columns. Tables indicate that when the machine number increases for specific number of jobs, the absolute value of the average percentage deviation decreases for each setup and energy groups, which implies that there is less improvement on the NEH solutions. This decrease is also observed in Tables 5.10 and 5.11 which shows the PD of VNS results from the NEH solutions for the $F|s_{ijl}, prmu|C_{max}$ problem, except the SDST10 group. The reason of this reduction may be the stopping condition of the proposed heuristic. The stopping condition is based on CPU time and it depends on the instance size. Although for larger instances we run the algorithm in longer time, we observe that the total number of iterations reduces faster due to complexity of the problem.

On the average, the improvement of the NEH solution is almost same for the large-sized instances. However, for 20 jobs and 50 jobs instances, we observe that when the weighted sum parameter of the total production cost w_1 is 0.1 and the weighted sum parameter of the total energy cost w_2 is 0.9, we can improve the NEH solution by the VNS algorithm more. One of the reasons for this behavior can be that the change in the sequence even by one job affects the energy cost more in the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, which also leads to a decrease in the total cost when we give higher weight to the energy cost objective.

We observe that for small-sized problems, the improvement of the NEH solution by the proposed VNS heuristic is greater for the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when we compare with the $F|s_{ijl}, prmu|C_{max}$ problem. However, for large-sized instances, the value of the APD decrease dramatically in the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem.

In Table 5.18, we observe that when we add the energy cost to the objective function or increase the value of the weighted sum parameter of the total energy cost w_2, the total energy cost value decreases as we expected. On the other hand, when the value of the weighted sum parameter of the total production cost w_1 increases, the value of the total production cost decreases except some instances. For the instances 2, 3, 5 and 10 seen in Table 5.18, when we change the weighted sum parameters w_1=0.9 and w_2=0.1 to w_1=1.0 and w_2=0.0, the total production cost also increases. The reason may be that when we set the weighted sum parameters different from 0, the structure of the problem changes. Hence, unexpected results can be observed such as instances 2, 3, 5 and 10.

We can conclude with the following observations for the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem from the results shown in Tables 5.14 -5.18:

- The proposed VNS algorithm improves the solutions effectively for the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem but when the instance size gets larger, the performance of the algorithm decreases. The most important reason is that we do not use any speed up algorithm for calculation of the objective function value in the local search procedure. When a job is removed from the sequence and inserted into another position among all possible positions, Taillard's speed up works effectively for the $F|s_{ijl}, prmu|C_{max}$ problem. Since the stopping condition is based on CPU time, for large-sized problems, the VNS algorithm iterates for a small number of iterations to improve the solution due to the complexity of the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem. For 500 jobs with 20 machines, we use

15 minutes as termination criterion according to the equation (n x m/2) x t milliseconds as proposed by Ruiz and Stützle (2008) and when t is 180. Allowing more time to the proposed VNS algorithm will increase the quality of the solution but it may not be desirable due to high time consumption.

- When we consider the energy cost into the objective function, we obtain significant energy saving.

In the Appendix, for the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem we tabulated also the maximum, the minimum and the average percentage deviation (PD) of the total production cost, the total energy cost and the total cost values of the VNS solutions from the optimal and NEH heuristic solutions, according to the size of the instances. Tables A1-12 show the PD values of the VNS algorithm from the optimal solutions for the small-sized problem. Tables A13-36 show the PD values of the VNS algorithm from the NEH solutions for the large-sized problem.

For the small-sized problems seen in Table A1-12, we observe high PD values when the weighted sum parameter of the total production cost w_1 is 0.1 and the weighted sum parameter of the total energy cost w_2 is 0.9. In general, the average PDs of the total production cost values are negative, which means that with VNS algorithm, we find better schedule according to the total production cost. On the other hand, the average PDs of the total energy cost values are positive, which means that VNS has worse performance for minimizing total energy cost. Namely, when we have the average PD of the VNS results from the optimal solution, we obtain a better schedule for the total production cost objective and a worse schedule for the total energy cost objective. In addition, when the instance sizes get larger, the absolute values of the PD values get higher. The reason of the deviation is that we consider the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem without inserted idle time during scheduling in the VNS algorithm. When we increase the value of the weighted sum parameter of the energy cost objective, we obtain some worse results, since our objective acts more like non-regular objective.

For the large-sized problems seen in Table A13-36, we observe that we improve the NEH solutions with the proposed VNS algorithm by improving the total production cost and the total energy cost almost equally. Namely, both the total production cost and the toal energy cost values are improved by VNS algorithm. For some of the instances, some positive PD values are observed. It means that the total cost is still improved by the proposed VNS algorithm; however the improvement is done by worsening one of the objective function value (which has positive PD value).

Chapter 6

CONCLUSIONS AND FUTURE RESEARCH

6.1 Conclusions

In this thesis, we studied two flowshop scheduling problems with sequence dependent setup times (FS-SDST): $F|s_{ijl}, prmu|C_{max}$ and $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$. In the $F|s_{ijl}, prmu|C_{max}$ problem, each job is characterized with a processing time on each machine and setup times according to the predecessor and successor jobs. We aimed to schedule the jobs to be processed on all machines to minimize the makespan. In the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, we studied energy-aware FS-SDST problem. We used the strategy that when the machine is kept idle for a long time, instead of keeping the machine idle, turning off and on the machine can consume lower energy. For the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, we considered that all machines consume energy during processing a part, idle period, turning off/on the machine and setup operations. While considering the energy consumption, we desired to minimize the completion time of the jobs to minimize the in-process inventory. In the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, the aim is to schedule the jobs to be processed on all machines and to decide about the status of the machine between scheduled jobs so as either to keep the machine idle or to turn off and turn on the machine, when the objective is to minimize both the total completion time and the total energy consumption.

We proposed a Variable Neighborhood Search (VNS) algorithm for these two FS-SDST problems. The proposed VNS algorithm uses two neighborhood structures and a local search procedure systematically. After the preliminary tests, we decided to start the algorithm with NEH heuristic initialization. As the neighborhood structures, we used the maximum setup one-job insertion and the swap neighborhood structures in that order for the '*shaking*' phase of the VNS algorithm. To improve the quality of the solution, we applied the local search procedure based on the node insertion neighborhood with the steepest descent strategy. We observe that the node insertion neighborhood structure is a powerful move operation for the FS-SDST problem. We also used the simulated annealing-like acceptance criterion for '*move or not*' phase of the algorithm, which provides diversification to the proposed VNS.

We analyzed the performance of the VNS algorithm for the $F|s_{ijl}, prmu|C_{max}$ problem by using the well-known instance set. We compared our results with the genetic algorithm (GA), memetic algorithm (MA), MA with modified local search (MA_LS), ant colony optimization algorithm (PACO), iterated greedy (IG) and IG with local search (IG_LS) from the literature. While the VNS algorithm gives better results in comparison to GA, MA, PACO and IG algorithms, the MA_LS and IG_LS outperform the proposed VNS algorithm. For the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, since it is studied for the first time in the literature, we generated a set of instances for the energy-related parameters. We solved small-sized problems with MILP model and proposed VNS algorithm, and compared the results to analyze the performance of VNS algorithm. For large-sized problem instances, we compared the VNS results with the well-known NEH constructive heuristic. This comparison indicates how much we improve the NEH solutions by the proposed VNS algorithm. We observed that the proposed VNS algorithm is a robust algorithm for the FS-SDST problem for different objectives. The proposed VNS algorithm gives competitive and acceptable results for the $F|s_{ijl}, prmu|C_{max}$ problem and the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem.

6.2 Future Research

Since the VNS algorithm has some decisions to make, the performance of the algorithm still can be improved by changing the some of the decisions made in the algorithm. New neighborhood structures can be proposed or the number of neighborhood structures can be increased. The solution quality is mostly improved by the local search phase compared to neighborhoods, hence another local search procedure can be proposed. For the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem, the algorithm can be improved by inserting idle time to the schedule. Moreover for the calculation of the total completion time objective, some speed up algorithms can be developed for future research.

BIBLIOGRAPHY

T. Aldowaisan and A. Allahverdi. New heuristics for no-wait flowshops to minimize makespan. *Computers and Operations Research*, 30:1219-1231, 2003.

A. Allahverdi, C. T. Ng, T. C. E. Cheng, and M. Y. Kovalyov. A survey of scheduling problems with setup times or costs. *European Journal of Operational Research*, 187:985-1032, 2008.

A. Allahverdi and T. Aldowaisan. Minimizing total completion time in a no-wait flowshop with sequence-dependent additive changeover times. *Journal of the Operational Research Society*, 52:449-462, 2001.

H. Belouadah, M. E. Posner, and C. N. Potts. Scheduling with release dates on a single machine to minimize total weighted completion time. *Discrete Applied Mathematics*, 36:213-231, 1992.

A. A. G. Bruzzone, D. Anghinolfi, M. Paolucci, and F. Tonelli. Energy-aware scheduling for improving manufacturing process sustainability: A mathematical model for flexible flow shops. *CIRP Annals- Manufacturing Technology*, 61: 459-462, 2012.

W. E. Costa, M. C. Goldbarg, and E. G. Goldbarg. New VNS heuristic for total flowtime flowshop scheduling problem. *Technical Report*, UFRN-DIMAp-2011-103-RT, Relatorio Tecnico, 2011.

M. Dai, D. Tang, A. Giret, M. A. Salido, and W. D. Li. Energy-efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm. *Robotics and Computer-Integrated Manufacturing*, 29:418-429, 2012.

S. R. Das, J. N. D. Gupta, and B. M. Khumawala. A savings index heuristic algorithm for flowshop scheduling with sequence dependent set-up times. *Journal of the Operational Research Society*, 46:1365-1373, 1995.

B. Ekşioğlu, S. D. Ekşioğlu, and P. Jain. A tabu search algorithm for the flowshop scheduling problem with changing neighborhoods. *Computers & Industrial Engineering*, 54:1-11, 2008.

K. Fang, N. A. Uhan, F. Zhao, and J. W. Sutherland. Flow shop scheduling with peak power consumption constraints. *Annals of Operations Research*, 206:115-145, 2013.

Y. Gajpal, C. Rajendran, and H. Ziegler. An ant colony algorithm for scheduling in flowshops with sequence-dependent setup times of jobs. *International Journal of Advanced Manufacturing Technology*, 30:416-424, 2006.

J. N. D. Gupta and W. P. Darrow. The two-machine sequence dependent flowshop scheduling problem. *European Journal of Operational Research*, 24: 439-446, 1986.

T. Gutowski, J. Dhamus, and A. Thiriez. Electrical energy requirements for manufacturing processes. *In: 13th CIRP International Conference on Life Cycle Engineering*, Leuven, 2006.

P. Hansen, N. Mladenovic, and J. A. M. Perez. Variable neighborhood search: methods and applications. *Annals of Operations Research*, 175:367-407, 2009.

H. Luo, B. Du, G. Q. Huang, H. Chen, and X. Li. Hybrid flow shop scheduling considering machine electricity consumption cost. *International Journal of Production Economics*, 146:423-439, 2013.

M. Mashaei and B. Lennartson. Energy reduction in a pallet-constrained flow shop through on-off control of idle machines. *IEEE Transactions on Automation Science and Engineering*, 10(1):45-56, 2013.

M. Mirabi. Ant colony optimization technique for the sequence-dependent flowshop scheduling problem. *International Journal of Advanced Manufacturing Technology*, 55:317-326, 2011.

N. Mladenovic and P. Hansen. Variable neighborhood search. *Computers & Operations Research*, 24(11):1097-1100, 1997.

G. Mouzon and M. B. Yildirim. A framework to minimize total energy consumption and total tardiness on a single machine. *International Journal of Sustainable Engineering*, 1(2):105-116, 2008.

B. Naderi, M. Khalili, M. T. Taghavifard, and V. Roshanaei. A variable neighborhood search for hybrid flexible flowshops with setup times minimizing total completion time. *Journal of Applied Sciences,* 8(16):2843-2850, 2008.

M. Nawaz, E. E. Enscore, and I. Ham. A heuristic algorithm for the M-machine, N-job flowshop sequencing problem. *Omega, International Journal of Management Science*, 11(1):91-95, 1983.

K. Nolde and M. Morari. Electrical load tracking scheduling of a steel plant. *Computers & Chemical Engineering*, 34:1899-1903, 2010.

I. H. Osman and C. N. Potts. Simulated annealing for permutation flowshop scheduling. *Omega, International Journal of Management Science*, 17(6):551-557, 1989.

M. Pinedo. Scheduling: theory, algorithms and systems. *Prentice-Hall,* Upper Saddle, NJ, Second ed., 2002.

C. Rajendran and H. Ziegler. An efficient heuristic for scheduling in a flowshop to minimize total weighted flowtime of jobs. *European Journal of Operational Research*, 103(1):129-138, 1997.

C. Rajendran and H. Ziegler. Ant-colony algorithms for permutation flowshop scheduling to minimize makespan/flowtime of jobs. *European Journal of Operational Research,* 155:426-438, 2004.

C. R. Reeves. A genetic algorithm for flowshop sequencing. *Computers and Operations Research*, 22(1):5-13, 1995.

R. Z. Rios-Mercado and J. F. Bard. Computational experience with a branch-and-cut algorithm for flowshop scheduling with setups. *Computers & Operations Research,* 25(5):351-366, 1998a.

R. Z. Rios-Mercado and J. F. Bard. Heuristics for the flow line problem with setup costs. *European Journal of Operational Research*, 110(1):76-98, 1998b.

R. Z. Rios-Mercado and J. F. Bard. A branch-and-bound algorithm for permutation flow shops with sequence dependent setup times. *IIE Transactions,* 31(8):721-731, 1999a.

R. Z. Rios-Mercado and J. F. Bard. An enhanced TSP-based heuristic for makespan minimization in a flow shop with setup times. *Journal of Heuristics*, 5:53-70, 1999b.

R. Z. Rios-Mercado and J. F. Bard. The flow shop scheduling polyhedron with setup times. *Journal of Combinatorial Optimization*, 7(3):291-318, 2003.

R. Ruiz. http://soa.iti.es/rruiz. 2008.

R. Ruiz, C. Maroto, and J. Alcaraz. Solving the flowshop scheduling problem with sequence dependent setup times using advanced metaheuristics. *European Journal of Operational Research*, 165(1): 34-54, 2005.

R. Ruiz and T. Stützle. A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem. *European Journal of Operational Research*, 177(3):2033-2049, 2007.

R. Ruiz and T. Stützle. An Iterated Greedy heuristic for the sequence dependent setup times flowshop problem with makespan and weighted tardiness objectives. *European Journal of Operational Research*, 187(3):1143-1159, 2008.

N. Salmasi, R. Logendran, and M. R. Skandari. Total flow time minimization in a flowshop sequence-dependent group scheduling problem. *Computers & Operations Research*, 37:199-212, 2010.

J. V. Simons Jr. Heuristics in flow shop scheduling with sequence dependent setup times. *Omega, The International Journal of Management Science*, 20 (2):215-225, 1992.

B. N. Srikar and S. Ghosh. A MILP model for the N-job, M-stage flowshop with sequence dependent set-up times. *International Journal of Production Research*, 24(6):1459-1474, 1986.

E. F. Stafford and F. T. Tseng. On the Srikar-Ghosh MILP model for the Nxm SDST flowshop problem. *International Journal of Production Research*, 28(10):1817-1830, 1990.

T. Stützle. Applying iterated local search to the permutation flow shop problem. *Technical Report*, TU Darmstadt, AIDA-98-04, FG Intellektik, 1998.

E. Taillard. Benchmarks for basic scheduling problems. *European Journal of Operational Research*, 64(2):278-285, 1993.

E. Taillard. Some efficient heuristic methods for the flow-shop sequencing problem. *European Journal of Operational Research*, 47(1):65-74, 1990.

M. F. Tasgetiren, Y. C. Liang, M. Sevkli, and G. Gencyilmaz. A particle swarm optimization algorithm for makespan and total flowtime minimization in the permutation flowshop sequencing problem. *European Journal of Operational Research*, 177:1930-1947, 2007.

F. T. Tseng and E. F. Stafford. Two MILP models for the N x M SDST flowshop sequencing problem. *International Journal of Production Research*, 39(8):1777-1809, 2001.

R. Vanchipura and R. Sridharan. Development and analysis of constructive heuristic algorithms for flow shop scheduling problems with sequence-dependent setup times. *International Journal of Advanced Manufacturing Technology*, 67:1337-1353, 2013.

R. Vanchipura, R. Sridharan, and A. S. Babu. Improvement of constructive heuristics using variable neighborhood descent for scheduling a flowshop with sequence dependent setup time. *Journal of Manufacturing Systems*, 33:65-75, 2014.

M. Widmer and A. Herzt. A new heuristic method for the flow shop sequencing problem. *European Journal of Operational Research*, 41:186-193, 1989.

M. B. Yildirim and G. Mouzon. Single-machine sustainable production planning to minimize total energy consumption and total completion time using a multiple objective genetic algorithm. *IEEE Transactions on Engineering Management*, 59:585-597, 2012.

S. Zanoni, L. Bettoni, and C. H. Glock. Energy implications in a two-stage production system with controllable production rates. *International Journal of Production Economics*, 149:164-171, 2014.

G. I. Zobolas, C. D. Tarantilis, and G. Ioannou. Minimizing makespan in permutation flow shop scheduling problems using a hybrid metaheuristic algorithm. *Computers & Operations Research*, 36:1249-1267, 2009.

APPENDIX

This Appendix presents the maximum, the minimum and the average percentage deviation (PD) of the total production cost, the total energy cost and the total cost values of the VNS solutions from the optimal and NEH heuristic solutions, according to the size of the instances for the $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem.

Table A1 Percentage deviation of the VNS results from the optimal solutions for the 10x2 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	Low	0.1	0.9	2.45	-0.72	0.43	11.70	0.00	3.35	2.23	0.00	0.70
		0.5	0.5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Medium	0.1	0.9	0.00	-5.00	-2.36	24.58	0.00	8.57	8.49	0.00	2.53
		0.5	0.5	0.00	-2.17	-0.83	21.46	0.00	9.10	1.99	0.00	0.88
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	High	0.1	0.9	0.00	-3.88	-1.76	25.58	0.00	7.79	14.52	0.00	4.21
		0.5	0.5	0.00	-1.76	-0.50	15.42	0.00	9.81	4.01	0.00	2.06
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SDST50	Low	0.1	0.9	0.00	-0.62	-0.22	0.94	0.00	0.46	0.09	0.00	0.04
		0.5	0.5	0.48	0.00	0.12	0.00	-2.26	-0.56	0.33	0.00	0.08
		0.9	0.1	0.00	0.00	0.00	5.00	0.00	1.25	0.04	0.00	0.01
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Medium	0.1	0.9	0.00	-22.97	-10.95	11.42	0.00	6.39	6.61	0.00	2.72
		0.5	0.5	0.00	-0.93	-0.28	3.89	0.00	1.07	1.01	0.00	0.25
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	High	0.1	0.9	0.00	-22.97	-11.70	10.91	0.00	6.36	8.23	0.00	4.10
		0.5	0.5	0.00	-1.85	-0.51	4.44	0.00	1.20	1.69	0.00	0.44
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table A2 Percentage deviation of the VNS results from the optimal solutions for the 10x2 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	Min.	Avg.	PD (%) of En. Cost Max.	Min.	Avg.	PD (%) of Total Cost Max.	Min.	Avg.
SDST100	Low	0.1	0.9	0.00	-2.62	-0.66	6.59	0.00	1.65	1.74	0.00	0.43
		0.5	0.5	0.00	-0.03	-0.01	0.53	0.00	0.13	0.03	0.00	0.01
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Medium	0.1	0.9	0.00	-9.50	-4.69	7.28	0.00	3.17	5.15	0.00	2.05
		0.5	0.5	0.00	-2.62	-0.66	4.80	0.00	1.20	0.50	0.00	0.13
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	High	0.1	0.9	0.00	-18.56	-7.39	5.99	0.00	3.18	3.72	0.00	2.26
		0.5	0.5	0.00	-8.74	-2.84	7.48	0.00	3.00	1.60	0.00	0.70
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SDST125	Low	0.1	0.9	0.00	-2.27	-0.69	3.51	0.00	1.38	1.14	0.00	0.48
		0.5	0.5	0.00	0.00	0.00	0.14	0.00	0.03	0.02	0.00	0.01
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Medium	0.1	0.9	0.00	-18.04	-7.19	4.96	0.00	1.82	2.01	0.00	0.70
		0.5	0.5	0.00	-1.91	-0.48	2.07	0.00	0.52	0.18	0.00	0.05
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	High	0.1	0.9	0.00	-10.74	-4.85	5.39	0.00	2.05	4.38	0.00	1.56
		0.5	0.5	9.89	0.00	2.47	0.00	-3.77	-0.94	0.51	0.00	0.13
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table A3 Percentage deviation of the VNS results from the optimal solutions for the 10x3 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	**Low**	0.1	0.9	5.88	-1.98	0.97	14.38	-15.27	-0.22	1.42	0.00	0.54
		0.5	0.5	0.00	0.00	0.00	0.48	0.00	0.12	0.02	0.00	0.00
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	-23.43	-5.90	0.00	0.00	0.00
	Medium	0.1	0.9	5.03	-14.98	-3.98	50.98	0.69	18.26	21.48	0.95	8.87
		0.5	0.5	3.31	-1.62	0.06	13.84	-6.29	3.82	1.67	0.00	0.79
		0.9	0.1	0.00	-0.24	-0.06	7.67	0.00	1.92	0.09	0.00	0.02
		1.0	0.0	0.00	0.00	0.00	0.00	-8.41	-2.10	0.00	0.00	0.00
	High	0.1	0.9	-4.75	-18.04	-12.08	51.35	5.36	21.83	31.31	1.34	12.31
		0.5	0.5	5.45	-4.44	0.65	27.31	-4.37	6.22	4.70	0.00	2.08
		0.9	0.1	0.00	0.00	0.00	0.67	0.00	0.17	0.05	0.00	0.01
		1.0	0.0	0.00	0.00	0.00	0.00	-20.10	-6.75	0.00	0.00	0.00
SDST50	**Low**	0.1	0.9	0.00	-4.88	-1.70	6.07	0.26	3.23	0.92	0.12	0.50
		0.5	0.5	0.00	-0.12	-0.03	2.51	0.00	0.81	0.09	0.00	0.04
		0.9	0.1	0.00	0.00	0.00	0.58	0.00	0.22	0.01	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	-2.81	-0.88	0.00	0.00	0.00
	Medium	0.1	0.9	0.00	-13.65	-5.08	15.69	0.00	7.37	10.30	0.00	5.26
		0.5	0.5	3.33	0.00	1.06	1.26	-2.14	0.07	0.92	0.00	0.61
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	High	0.1	0.9	0.00	-24.20	-9.91	16.67	0.00	7.72	11.89	0.00	5.84
		0.5	0.5	0.00	-9.03	-3.83	13.96	0.00	5.79	2.99	0.00	1.48
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	-1.72	-0.43	0.00	0.00	0.00

Table A4 Percentage deviation of the VNS results from the optimal solutions for the 10x3 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST100	**Low**	0.1	0.9	0.00	-5.98	-2.56	7.51	0.30	4.01	2.58	0.17	1.15
		0.5	0.5	0.00	0.00	0.00	1.29	0.00	0.39	0.19	0.00	0.06
		0.9	0.1	0.00	0.00	0.00	1.29	0.00	0.69	0.02	0.00	0.01
		1.0	0.0	0.00	0.00	0.00	0.00	-0.12	-0.03	0.00	0.00	0.00
	Medium	0.1	0.9	16.26	-14.99	-2.77	11.50	0.78	4.60	8.26	1.58	3.70
		0.5	0.5	-0.81	-1.98	-1.41	3.80	1.05	2.19	1.12	0.14	0.52
		0.9	0.1	0.00	0.00	0.00	1.46	0.00	0.36	0.22	0.00	0.05
		1.0	0.0	0.00	0.00	0.00	0.80	-0.14	0.17	0.00	0.00	0.00
	High	0.1	0.9	-3.50	-14.99	-8.68	12.88	2.05	5.68	10.79	1.78	4.70
		0.5	0.5	-1.25	-4.84	-2.56	6.04	1.34	3.04	2.57	0.55	1.25
		0.9	0.1	0.00	0.00	0.00	1.33	0.00	0.33	0.32	0.00	0.08
		1.0	0.0	0.00	0.00	0.00	0.00	-0.13	-0.03	0.00	0.00	0.00
SDST125	**Low**	0.1	0.9	0.27	-3.69	-0.86	3.71	-0.04	1.03	0.84	0.00	0.33
		0.5	0.5	3.03	0.00	0.76	0.00	-0.39	-0.10	2.45	0.00	0.61
		0.9	0.1	0.00	0.00	0.00	0.29	0.00	0.07	0.01	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	-0.76	-0.22	0.00	0.00	0.00
	Medium	0.1	0.9	-3.86	-10.83	-7.34	6.06	2.55	4.50	4.81	2.03	3.50
		0.5	0.5	1.76	-1.39	-0.43	5.23	0.07	2.33	2.12	0.45	1.13
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	-0.68	-0.17	0.00	0.00	0.00
	High	0.1	0.9	-3.86	-20.65	-12.45	6.82	2.65	4.95	5.47	2.39	4.02
		0.5	0.5	5.98	-2.62	0.33	2.86	-1.36	1.53	1.99	0.81	1.27
		0.9	0.1	0.29	0.00	0.07	0.00	-0.35	-0.09	0.14	0.00	0.04
		1.0	0.0	0.00	0.00	0.00	0.00	-0.54	-0.13	0.00	0.00	0.00

Table A5 Percentage deviation of the VNS results from the optimal solutions for the 10x4 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	Min.	Avg.	PD (%) of En. Cost Max.	Min.	Avg.	PD (%) of Total Cost Max.	Min.	Avg.
SDST10	**Low**	0.1	0.9	0.00	-4.77	-1.58	20.41	0.18	10.94	3.08	0.03	1.62
		0.5	0.5	0.54	-0.14	0.07	3.22	-10.85	-0.82	0.10	0.06	0.08
		0.9	0.1	0.00	0.00	0.00	2.88	0.00	0.87	0.01	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	-0.40	-0.10	0.00	0.00	0.00
	Medium	0.1	0.9	-0.82	-22.24	-8.45	54.12	2.19	28.93	24.14	1.05	14.59
		0.5	0.5	0.00	-2.87	-0.90	12.55	0.00	8.84	3.07	0.00	1.72
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	-3.40	-1.14	0.00	0.00	0.00
	High	0.1	0.9	0.71	-22.24	-9.90	56.18	7.52	30.03	36.02	2.16	19.79
		0.5	0.5	-0.24	-6.03	-3.85	29.82	1.32	18.05	6.52	0.17	3.48
		0.9	0.1	0.00	-0.18	-0.11	3.14	0.00	1.67	0.12	0.00	0.05
		1.0	0.0	0.00	0.00	0.00	0.00	-0.86	-0.21	0.00	0.00	0.00
SDST50	**Low**	0.1	0.9	9.55	-2.47	1.18	5.52	-0.78	2.62	3.55	0.73	1.81
		0.5	0.5	0.00	-0.12	-0.04	1.61	0.00	0.85	0.25	0.00	0.10
		0.9	0.1	0.00	0.00	0.00	3.13	0.00	0.93	0.08	0.00	0.02
		1.0	0.0	0.00	0.00	0.00	0.79	-0.64	0.08	0.00	0.00	0.00
	Medium	0.1	0.9	-9.05	-25.00	-16.89	18.17	6.80	13.63	12.14	4.56	9.27
		0.5	0.5	-0.63	-3.27	-1.90	10.30	3.20	5.93	3.00	0.88	1.75
		0.9	0.1	0.00	0.00	0.00	0.73	0.00	0.18	0.09	0.00	0.02
		1.0	0.0	0.00	0.00	0.00	0.57	-0.74	-0.13	0.00	0.00	0.00
	High	0.1	0.9	-9.05	-29.48	-18.61	17.41	6.33	13.58	14.33	5.12	10.90
		0.5	0.5	-2.16	-6.38	-4.23	11.32	4.48	8.03	5.26	1.74	3.43
		0.9	0.1	0.00	-0.12	-0.03	0.94	0.00	0.63	0.17	0.00	0.10
		1.0	0.0	0.00	-1.08	-0.27	1.44	-1.60	-0.34	0.00	-1.08	-0.27

Table A6 Percentage deviation of the VNS results from the optimal solutions for the 10x4 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST100	Low	0.1	0.9	-0.13	-3.14	-1.33	3.90	1.63	2.28	1.68	0.34	0.81
		0.5	0.5	0.00	0.00	0.00	1.50	0.00	0.55	0.24	0.00	0.09
		0.9	0.1	0.00	0.00	0.00	1.50	0.13	0.59	0.03	0.00	0.01
		1.0	0.0	0.00	0.00	0.00	-0.14	-1.98	-0.67	0.00	0.00	0.00
	Medium	0.1	0.9	-6.11	-14.29	-12.04	7.65	2.54	5.27	5.80	1.69	3.82
		0.5	0.5	-0.03	-0.99	-0.47	2.20	0.04	1.31	1.18	0.01	0.59
		0.9	0.1	0.00	-0.02	0.00	1.17	0.00	0.73	0.16	0.00	0.11
		1.0	0.0	0.00	0.00	0.00	-0.14	-1.93	-0.77	0.00	0.00	0.00
	High	0.1	0.9	-13.54	-15.62	-14.30	8.28	3.00	5.24	6.97	2.06	4.26
		0.5	0.5	-0.03	-5.14	-2.81	5.33	0.08	2.89	2.52	0.05	1.30
		0.9	0.1	0.00	-0.20	-0.06	1.11	0.72	0.91	0.26	0.08	0.19
		1.0	0.0	0.00	0.00	0.00	-0.03	-1.41	-0.64	0.00	0.00	0.00
SDST125	Low	0.1	0.9	-0.19	-9.20	-3.60	11.05	2.45	5.45	1.92	1.50	1.63
		0.5	0.5	0.00	-0.35	-0.14	2.20	0.00	1.08	0.22	0.00	0.06
		0.9	0.1	0.00	0.00	0.00	0.63	0.00	0.19	0.02	0.00	0.01
		1.0	0.0	0.00	0.00	0.00	0.14	-0.55	-0.10	0.00	0.00	0.00
	Medium	0.1	0.9	-4.62	-29.97	-14.28	13.01	1.29	6.31	8.44	0.88	4.43
		0.5	0.5	-0.05	-4.00	-2.27	4.33	0.47	2.39	1.63	0.27	0.71
		0.9	0.1	0.85	-1.70	-0.21	6.38	-3.70	0.90	0.16	0.00	0.06
		1.0	0.0	0.00	0.00	0.00	0.30	-0.12	0.06	0.00	0.00	0.00
	High	0.1	0.9	-10.90	-29.97	-20.42	8.18	2.25	5.60	6.09	1.27	4.32
		0.5	0.5	-0.05	-8.46	-4.24	5.77	0.46	3.58	2.38	0.34	1.69
		0.9	0.1	1.75	-0.16	0.40	1.38	-4.22	-0.59	0.28	0.00	0.15
		1.0	0.0	0.00	0.00	0.00	0.25	-0.33	-0.10	0.00	0.00	0.00

Table A7 Percentage deviation of the VNS results from the optimal solutions for the 10x5 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	**Low**	0.1	0.9	0.32	-5.44	-2.16	27.58	5.85	16.14	4.02	1.41	2.76
		0.5	0.5	0.00	-0.10	-0.03	8.05	2.95	4.37	0.31	0.07	0.19
		0.9	0.1	0.00	0.00	0.00	4.31	1.23	2.75	0.03	0.01	0.02
		1.0	0.0	0.00	0.00	0.00	1.08	-5.18	-2.07	0.00	0.00	0.00
	Medium	0.1	0.9	-6.85	-21.72	-14.78	61.91	26.35	38.21	30.90	10.88	18.31
		0.5	0.5	4.24	-2.30	0.34	26.64	0.03	10.72	5.66	0.67	3.11
		0.9	0.1	0.00	-0.02	-0.01	10.04	3.09	5.91	0.50	0.17	0.30
		1.0	0.0	0.00	0.00	0.00	-0.64	-11.85	-5.51	0.00	0.00	0.00
	High	0.1	0.9	-6.85	-30.58	-17.27	75.81	20.86	41.56	45.43	14.60	25.88
		0.5	0.5	-0.14	-4.26	-2.01	37.43	3.30	17.89	9.65	1.11	4.78
		0.9	0.1	0.71	0.00	0.18	8.87	-4.28	2.75	0.77	0.27	0.41
		1.0	0.0	0.00	0.00	0.00	-0.40	-5.57	-3.09	0.00	0.00	0.00
SDST50	**Low**	0.1	0.9	0.00	-2.81	-1.41	9.57	2.15	4.88	4.38	0.46	1.98
		0.5	0.5	0.00	-1.34	-0.34	9.53	0.79	3.50	0.32	0.10	0.19
		0.9	0.1	0.00	0.00	0.00	1.67	0.27	1.17	0.03	0.01	0.02
		1.0	0.0	0.00	0.00	0.00	0.20	-2.26	-0.86	0.00	0.00	0.00
	Medium	0.1	0.9	-8.58	-26.02	-17.99	20.45	7.79	14.60	14.42	4.88	10.06
		0.5	0.5	0.00	-4.76	-2.90	10.56	4.45	6.86	2.71	0.52	1.94
		0.9	0.1	0.00	0.00	0.00	1.63	0.80	1.24	0.20	0.10	0.16
		1.0	0.0	0.00	0.00	0.00	0.10	-1.81	-0.99	0.00	0.00	0.00
	High	0.1	0.9	-9.76	-23.53	-17.05	18.82	7.42	14.18	15.78	5.91	11.71
		0.5	0.5	6.50	-12.66	-4.82	15.77	-0.40	8.16	4.64	1.77	3.23
		0.9	0.1	0.00	-1.17	-0.29	5.73	0.59	2.19	0.29	0.12	0.23
		1.0	0.0	0.00	0.00	0.00	-0.21	-1.99	-1.15	0.00	0.00	0.00

Table A8 Percentage deviation of the VNS results from the optimal solutions for the 10x5 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w1 (%)	w2 (%)	PD (%) of Prod. Cost Max.	Min.	Avg.	PD (%) of En. Cost Max.	Min.	Avg.	PD (%) of Total Cost Max.	Min.	Avg.
SDST100	Low	0.1	0.9	12.25	-1.73	2.18	3.33	-3.20	0.83	1.80	0.48	0.99
		0.5	0.5	0.00	-0.06	-0.02	0.49	0.13	0.31	0.11	0.03	0.06
		0.9	0.1	0.00	-0.02	0.00	0.88	0.13	0.42	0.03	0.00	0.01
		1.0	0.0	0.00	0.00	0.00	-0.06	-1.06	-0.54	0.00	0.00	0.00
	Medium	0.1	0.9	-0.53	-16.14	-10.95	10.00	3.02	7.02	7.77	2.81	5.44
		0.5	0.5	-0.88	-9.27	-4.28	8.25	2.15	4.40	1.31	0.77	1.04
		0.9	0.1	0.00	-0.09	-0.02	0.56	0.00	0.30	0.11	0.00	0.04
		1.0	0.0	0.00	0.00	0.00	0.06	-1.07	-0.36	0.00	0.00	0.00
	High	0.1	0.9	-0.53	-22.98	-13.23	10.67	2.03	7.21	9.39	1.95	6.11
		0.5	0.5	-0.88	-13.32	-5.49	9.21	2.18	5.16	3.55	1.50	2.39
		0.9	0.1	0.00	-0.09	-0.02	0.54	0.00	0.31	0.16	0.00	0.07
		1.0	0.0	0.00	0.00	0.00	0.59	-1.08	-0.17	0.00	0.00	0.00
SDST125	Low	0.1	0.9	-0.68	-4.62	-2.91	6.90	1.91	3.88	2.41	0.42	1.19
		0.5	0.5	2.36	-0.03	0.58	0.93	-7.41	-1.50	0.35	0.00	0.16
		0.9	0.1	343.54	0.00	85.88	1.11	-96.87	-23.55	0.03	0.00	0.02
		1.0	0.0	0.00	0.00	0.00	0.81	-1.62	-0.46	0.00	0.00	0.00
	Medium	0.1	0.9	-7.74	-19.36	-15.09	7.51	2.32	5.20	5.53	1.69	3.82
		0.5	0.5	-0.51	-8.58	-3.25	6.52	1.44	2.96	1.72	0.71	0.97
		0.9	0.1	0.00	0.00	0.00	1.09	0.55	0.84	0.23	0.11	0.18
		1.0	0.0	0.00	0.00	0.00	0.99	-0.64	-0.05	0.00	0.00	0.00
	High	0.1	0.9	-13.71	-31.35	-20.55	7.61	3.29	5.63	5.99	2.69	4.58
		0.5	0.5	-1.25	-9.11	-4.33	4.56	1.79	3.27	2.76	0.95	1.56
		0.9	0.1	0.00	-0.14	-0.03	1.55	0.03	0.86	0.44	0.01	0.26
		1.0	0.0	0.00	0.00	0.00	1.10	-0.78	-0.22	0.00	0.00	0.00

Table A9 Percentage deviation of the VNS results from the optimal solutions for the 15x2 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	**Low**	0.1	0.9	0.75	-0.62	0.03	3.66	0.00	1.78	1.04	0.00	0.31
		0.5	0.5	0.02	0.00	0.00	0.00	-0.77	-0.19	0.00	0.00	0.00
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.23	0.00	0.06	1.10	0.00	0.28	0.23	0.00	0.06
	Medium	0.1	0.9	0.00	-11.03	-4.14	42.65	0.00	16.68	15.65	0.00	5.03
		0.5	0.5	0.00	-1.67	-0.79	26.20	0.00	9.05	1.11	0.00	0.28
		0.9	0.1	0.07	0.00	0.02	4.72	-1.33	0.85	0.10	0.00	0.04
		1.0	0.0	0.00	0.00	0.00	0.00	-14.56	-3.64	0.00	0.00	0.00
	High	0.1	0.9	4.71	-11.56	-5.33	46.72	-1.48	18.97	17.08	1.10	7.21
		0.5	0.5	0.84	-1.72	-0.38	24.03	0.00	7.40	2.73	0.00	1.02
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	2.71	0.00	0.68	0.00	0.00	0.00
SDST50	**Low**	0.1	0.9	3.57	-0.33	1.33	14.16	-7.54	1.50	4.48	0.22	1.91
		0.5	0.5	0.95	-0.16	0.20	19.22	0.00	8.17	1.30	0.00	0.37
		0.9	0.1	0.98	0.00	0.43	35.70	-3.52	8.04	1.04	0.00	0.44
		1.0	0.0	0.04	0.00	0.01	2.94	0.00	0.74	0.04	0.00	0.01
	Medium	0.1	0.9	0.00	-7.56	-4.05	13.01	0.00	6.96	7.68	0.00	3.73
		0.5	0.5	49.40	-1.71	11.65	5.21	-49.40	-9.60	0.42	0.00	0.19
		0.9	0.1	1.20	-0.46	0.18	29.17	0.00	12.33	1.94	0.00	0.66
		1.0	0.0	0.01	0.00	0.00	0.00	-1.88	-0.47	0.01	0.00	0.00
	High	0.1	0.9	4.04	-8.33	-1.15	17.44	0.00	7.27	15.09	0.00	5.64
		0.5	0.5	9.40	-3.30	1.35	7.94	-7.92	1.22	2.60	0.38	1.24
		0.9	0.1	0.00	-0.16	-0.04	8.18	0.00	2.05	0.44	0.00	0.11
		1.0	0.0	1.84	0.00	0.62	37.36	-0.22	13.22	1.84	0.00	0.62

Table A10 Percentage deviation of the VNS results from the optimal solutions for the 15x2 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST100	**Low**	0.1	0.9	0.28	-2.68	-0.97	6.18	1.62	4.04	2.81	0.04	0.94
		0.5	0.5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Medium	0.1	0.9	5.98	-10.05	-1.56	11.50	0.00	4.70	7.18	0.00	3.44
		0.5	0.5	6.72	-4.33	-0.02	18.09	0.00	7.88	6.98	0.00	2.62
		0.9	0.1	3.36	0.00	0.84	0.00	-2.45	-0.61	2.92	0.00	0.73
		1.0	0.0	0.00	0.00	0.00	0.00	-0.30	-0.08	0.00	0.00	0.00
	High	0.1	0.9	3.11	-10.05	-4.84	11.82	0.79	4.87	9.06	0.79	3.70
		0.5	0.5	3.13	-0.88	0.54	2.06	0.00	1.05	2.58	0.00	0.82
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	3.36	0.00	0.84	0.00	-2.83	-0.75	3.36	0.00	0.84
SDST125	**Low**	0.1	0.9	1.91	-0.60	0.24	1.38	-0.61	0.36	0.67	0.00	0.24
		0.5	0.5	0.00	0.00	0.00	0.10	0.00	0.03	0.01	0.00	0.00
		0.9	0.1	3.81	0.00	1.28	9.15	-15.64	-3.61	3.89	0.00	1.24
		1.0	0.0	0.00	0.00	0.00	0.00	-0.19	-0.05	0.00	0.00	0.00
	Medium	0.1	0.9	15.37	0.00	6.44	7.00	0.00	3.63	8.14	0.00	4.03
		0.5	0.5	5.22	-0.36	1.87	6.42	0.04	3.22	5.73	0.29	2.30
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	2.26	0.00	0.94	2.08	-12.12	-3.97	2.26	0.00	0.94
	High	0.1	0.9	11.99	-11.71	0.76	7.57	0.00	4.65	7.93	0.00	4.24
		0.5	0.5	4.55	-1.22	1.35	7.16	0.66	3.91	6.05	0.94	2.67
		0.9	0.1	0.00	-0.93	-0.23	8.17	0.00	2.04	0.33	0.00	0.08
		1.0	0.0	1.13	0.00	0.38	1.76	-12.60	-5.69	1.13	0.00	0.38

Table A11 Percentage deviation of the VNS results from the optimal solutions for the 15x3 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	Low	0.1	0.9	1.91	-3.18	-0.55	21.50	4.70	12.88	2.46	1.70	2.18
		0.5	0.5	0.56	0.00	0.14	0.43	-3.09	-0.67	0.41	0.00	0.11
		0.9	0.1	0.24	0.00	0.06	14.12	0.00	3.93	0.30	0.00	0.08
		1.0	0.0	0.19	0.00	0.05	11.74	-9.86	0.37	0.19	0.00	0.05
	Medium	0.1	0.9	33.40	-10.97	7.82	27.95	-11.85	9.09	12.50	2.37	6.67
		0.5	0.5	2.62	-4.52	-0.69	31.65	-6.73	9.03	1.25	0.22	0.94
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.22	0.00	0.10	9.59	-3.64	3.61	0.22	0.00	0.10
	High	0.1	0.9	1.40	-10.67	-3.56	32.51	6.05	14.42	18.08	3.52	7.68
		0.5	0.5	2.92	-3.42	-1.07	21.66	-1.20	11.83	3.19	1.08	1.79
		0.9	0.1	1.77	0.00	0.44	0.00	-20.00	-5.00	0.00	0.00	0.00
		1.0	0.0	0.16	0.00	0.04	11.77	-2.94	3.03	0.16	0.00	0.04
SDST50	Low	0.1	0.9	-0.22	-10.86	-4.16	13.32	1.54	8.34	3.07	0.26	1.35
		0.5	0.5	3.28	-0.48	0.70	11.97	-9.54	0.83	2.00	0.03	0.68
		0.9	0.1	0.00	0.00	0.00	0.78	0.22	0.39	0.01	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	-0.09	-7.16	-2.34	0.00	0.00	0.00
	Medium	0.1	0.9	3.29	-0.94	1.36	6.92	1.68	4.09	6.22	1.67	3.52
		0.5	0.5	0.00	-2.51	-1.10	7.65	1.79	4.57	2.00	0.14	1.02
		0.9	0.1	1.30	0.00	0.32	0.34	-7.23	-1.72	0.73	0.00	0.19
		1.0	0.0	0.00	0.00	0.00	0.00	-1.48	-0.72	0.00	0.00	0.00
	High	0.1	0.9	15.58	-9.50	-1.48	9.90	3.90	7.48	8.04	5.01	6.27
		0.5	0.5	1.85	-2.51	-0.74	7.47	1.82	5.00	4.60	0.49	2.09
		0.9	0.1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1.0	0.0	0.00	0.00	0.00	0.05	-2.81	-0.69	0.00	0.00	0.00

Table A12 Percentage deviation of the VNS results from the optimal solutions for the 15x3 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w1 (%)	w2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST100	Low	0.1	0.9	-0.11	-1.79	-0.81	2.96	0.38	1.50	0.88	0.09	0.34
		0.5	0.5	0.56	-0.67	-0.04	6.65	0.38	3.62	0.59	0.02	0.36
		0.9	0.1	0.42	0.00	0.13	4.26	-9.41	-1.24	0.22	0.00	0.10
		1.0	0.0	0.23	0.00	0.06	0.44	-3.57	-0.84	0.23	0.00	0.06
	Medium	0.1	0.9	3.84	-12.44	-4.69	8.83	0.61	5.48	6.79	1.00	3.97
		0.5	0.5	2.94	-0.87	0.38	8.97	-0.15	3.04	4.09	0.10	1.58
		0.9	0.1	0.00	-0.23	-0.06	8.28	0.00	2.07	0.63	0.00	0.16
		1.0	0.0	0.33	0.00	0.08	3.16	-0.61	0.60	0.33	0.00	0.08
	High	0.1	0.9	5.73	-18.73	-9.78	10.08	5.07	7.45	9.72	2.97	5.89
		0.5	0.5	-1.54	-2.96	-1.97	4.32	3.04	3.72	1.71	1.13	1.44
		0.9	0.1	0.82	0.00	0.20	0.51	0.00	0.20	0.77	0.00	0.20
		1.0	0.0	0.35	0.00	0.20	1.93	0.00	1.30	0.35	0.00	0.20
SDST125	Low	0.1	0.9	2.29	-0.12	1.57	2.88	-0.38	0.66	2.53	0.16	1.10
		0.5	0.5	0.35	0.00	0.09	11.30	0.00	3.21	1.21	0.00	0.43
		0.9	0.1	1.37	0.00	0.48	0.01	-13.64	-4.89	1.07	0.00	0.38
		1.0	0.0	0.62	0.00	0.16	1.00	-6.81	-1.45	0.62	0.00	0.16
	Medium	0.1	0.9	4.17	-1.82	0.49	5.52	2.32	3.96	5.05	2.53	3.57
		0.5	0.5	4.79	-5.45	-0.64	6.59	-0.99	2.07	1.93	0.06	0.75
		0.9	0.1	1.06	0.00	0.49	1.60	-0.68	0.23	1.00	0.00	0.46
		1.0	0.0	0.05	0.00	0.01	0.47	-2.97	-0.64	0.05	0.00	0.01
	High	0.1	0.9	-3.91	-10.34	-6.47	6.42	1.31	3.28	5.20	0.74	2.60
		0.5	0.5	2.61	-1.45	0.76	3.77	-0.76	1.09	3.11	0.00	0.99
		0.9	0.1	0.69	0.00	0.25	1.31	-1.07	0.09	0.58	0.00	0.21
		1.0	0.0	0.05	0.00	0.01	0.02	-2.89	-0.73	0.05	0.00	0.01

Table A13 Percentage deviation of the VNS results from the NEH solutions for the 20x5 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	Low	0.1	0.9	-2.48	-9.37	-6.00	-0.91	-22.33	-10.85	-2.16	-10.66	-7.12
		0.5	0.5	-1.90	-6.79	-4.37	6.56	-13.51	-2.63	-2.43	-6.30	-4.34
		0.9	0.1	-2.43	-6.47	-4.47	7.55	-13.98	-2.35	-2.40	-6.45	-4.46
		1.0	0.0	-2.34	-7.15	-4.51	10.52	-12.01	-2.75	-2.34	-7.15	-4.51
	Medium	0.1	0.9	-1.10	-19.15	-7.44	-3.16	-21.36	-14.89	-8.42	-15.82	-12.12
		0.5	0.5	-2.99	-10.75	-5.88	4.80	-19.41	-7.59	-2.59	-10.15	-6.29
		0.9	0.1	-2.62	-6.76	-4.51	13.12	-12.85	-3.11	-2.33	-6.71	-4.51
		1.0	0.0	-2.43	-7.15	-4.47	10.63	-9.64	-2.19	-2.43	-7.15	-4.47
	High	0.1	0.9	-1.19	-12.10	-5.54	-4.45	-18.21	-13.41	-6.69	-14.79	-11.39
		0.5	0.5	-1.45	-13.84	-5.99	-0.47	-25.51	-12.68	-4.83	-10.96	-8.16
		0.9	0.1	-2.15	-7.52	-4.68	16.25	-12.27	-0.90	-2.30	-7.23	-4.52
		1.0	0.0	-2.40	-6.91	-4.43	9.05	-15.63	-3.11	-2.40	-6.91	-4.43
SDST50	Low	0.1	0.9	2.01	-14.83	-7.29	-8.17	-20.03	-12.87	-6.46	-12.45	-9.83
		0.5	0.5	-0.93	-8.98	-5.45	-0.10	-17.25	-10.87	-1.68	-9.37	-6.03
		0.9	0.1	-1.45	-8.45	-5.91	4.57	-22.65	-10.90	-1.60	-8.44	-5.98
		1.0	0.0	-1.45	-8.76	-5.91	-0.38	-23.40	-13.16	-1.45	-8.76	-5.91
	Medium	0.1	0.9	2.96	-11.79	-3.61	-2.31	-17.91	-11.32	-3.94	-15.01	-10.17
		0.5	0.5	-0.10	-10.13	-5.56	-3.62	-17.25	-10.76	-5.31	-11.40	-7.78
		0.9	0.1	-0.89	-9.44	-5.83	-5.44	-18.96	-12.22	-1.78	-9.94	-6.40
		1.0	0.0	-1.45	-8.80	-6.08	-4.15	-27.64	-13.90	-1.45	-8.80	-6.08
	High	0.1	0.9	5.40	-13.49	-2.91	-7.50	-15.21	-11.84	-7.66	-14.25	-11.07
		0.5	0.5	2.03	-11.41	-7.24	-0.75	-16.28	-12.32	-5.62	-13.26	-10.24
		0.9	0.1	-1.74	-8.44	-4.81	-0.31	-19.51	-9.15	-2.69	-8.01	-5.49
		1.0	0.0	-1.45	-9.30	-6.13	-1.65	-29.17	-12.21	-1.45	-9.30	-6.13

Table A14 Percentage deviation of the VNS results from the NEH solutions for the 20x5 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST100	Low	0.1	0.9	0.04	-13.28	-7.21	-4.59	-17.54	-10.21	-2.84	-12.93	-9.07
		0.5	0.5	-3.45	-10.27	-6.79	-0.63	-21.82	-10.40	-3.64	-11.50	-7.34
		0.9	0.1	-5.55	-11.50	-8.14	0.11	-18.38	-11.55	-5.62	-11.53	-8.21
		1.0	0.0	-5.46	-10.96	-7.98	-9.85	-25.46	-14.50	-5.46	-10.96	-7.98
	Medium	0.1	0.9	7.47	-13.66	-3.49	-1.41	-16.36	-9.72	-1.50	-15.68	-9.13
		0.5	0.5	-4.19	-12.46	-8.59	-3.30	-16.51	-9.47	-6.22	-10.87	-9.08
		0.9	0.1	-3.70	-11.85	-7.88	-1.04	-13.65	-8.72	-3.39	-12.01	-7.98
		1.0	0.0	-5.55	-11.49	-8.09	-3.09	-24.02	-10.50	-5.55	-11.49	-8.09
	High	0.1	0.9	3.83	-10.49	-3.42	-5.74	-17.49	-12.36	-5.50	-16.46	-11.88
		0.5	0.5	3.12	-14.61	-6.74	-5.12	-18.87	-11.11	-5.76	-14.81	-9.72
		0.9	0.1	-0.57	-15.08	-8.04	-6.97	-16.49	-10.09	-1.87	-14.60	-8.47
		1.0	0.0	-4.43	-11.44	-7.97	-6.06	-23.58	-11.36	-4.43	-11.44	-7.97
SDST125	Low	0.1	0.9	-1.54	-17.55	-7.77	-8.13	-17.92	-12.72	-5.98	-15.82	-10.47
		0.5	0.5	-3.68	-15.73	-7.98	-3.96	-20.81	-11.14	-4.03	-14.78	-8.48
		0.9	0.1	-4.54	-14.55	-8.49	-0.63	-22.50	-10.35	-4.57	-14.67	-8.53
		1.0	0.0	-4.44	-14.07	-7.84	-0.52	-26.12	-12.08	-4.44	-14.07	-7.84
	Medium	0.1	0.9	4.47	-9.96	-4.95	-6.46	-17.01	-12.99	-6.30	-15.96	-12.30
		0.5	0.5	-4.22	-14.17	-9.54	-3.36	-14.99	-9.99	-5.49	-12.65	-9.79
		0.9	0.1	-4.72	-12.29	-8.23	-4.93	-25.37	-10.95	-4.82	-14.28	-8.61
		1.0	0.0	-4.45	-14.07	-8.09	1.30	-21.37	-11.72	-4.45	-14.07	-8.09
	High	0.1	0.9	10.39	-18.04	-6.27	-7.02	-18.79	-12.87	-6.74	-17.97	-12.57
		0.5	0.5	-2.90	-10.56	-7.64	-7.82	-16.57	-12.33	-6.31	-14.78	-10.92
		0.9	0.1	-3.06	-14.11	-9.00	-5.27	-20.79	-12.14	-5.18	-13.55	-9.68
		1.0	0.0	-4.49	-14.07	-7.96	-1.29	-21.42	-11.09	-4.49	-14.07	-7.96

Table A15 Percentage deviation of the VNS results from the NEH solutions for the 20x10 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	**Low**	0.1	0.9	2.35	-10.23	-3.66	-5.01	-19.47	-10.85	-3.82	-9.84	-6.15
		0.5	0.5	-2.84	-7.37	-5.04	8.27	-8.02	1.23	-2.78	-6.89	-4.64
		0.9	0.1	-3.22	-7.62	-4.44	9.52	-5.69	2.66	-3.23	-7.61	-4.38
		1.0	0.0	-2.27	-7.38	-4.20	11.87	-1.64	4.45	-2.27	-7.38	-4.20
	Medium	0.1	0.9	3.32	-12.59	-2.47	-5.99	-14.29	-10.88	-3.64	-11.39	-8.74
		0.5	0.5	-1.82	-10.58	-5.67	0.67	-14.84	-7.42	-3.30	-8.54	-6.18
		0.9	0.1	-3.50	-6.73	-4.53	8.82	-6.47	-0.62	-2.79	-6.65	-4.32
		1.0	0.0	-2.09	-7.40	-4.26	9.48	-3.32	3.90	-2.09	-7.40	-4.26
	High	0.1	0.9	3.19	-13.65	-2.15	-6.13	-13.69	-9.68	-5.15	-12.02	-8.52
		0.5	0.5	0.94	-15.04	-7.24	0.62	-20.01	-8.47	-4.06	-12.08	-7.82
		0.9	0.1	-2.41	-7.49	-4.86	4.22	-6.54	-0.36	-1.83	-6.50	-4.43
		1.0	0.0	-2.11	-7.29	-4.18	10.70	-2.51	4.36	-2.11	-7.29	-4.18
SDST50	**Low**	0.1	0.9	2.40	-13.27	-6.07	-0.53	-12.25	-6.98	-2.01	-12.28	-6.60
		0.5	0.5	-2.53	-6.83	-4.39	3.83	-15.15	-5.16	-2.76	-6.28	-4.52
		0.9	0.1	-2.17	-7.07	-4.93	5.23	-12.26	-4.95	-2.30	-6.95	-4.93
		1.0	0.0	-3.90	-7.20	-5.29	5.02	-9.41	-3.26	-3.90	-7.20	-5.29
	Medium	0.1	0.9	3.28	-9.57	-3.79	-2.79	-10.54	-6.74	-3.13	-10.39	-6.48
		0.5	0.5	-1.11	-10.67	-6.08	-1.46	-9.96	-5.70	-4.44	-7.54	-5.88
		0.9	0.1	-1.28	-7.01	-4.73	2.78	-13.17	-4.24	-1.80	-6.10	-4.68
		1.0	0.0	-3.80	-7.20	-5.40	-0.08	-11.13	-5.32	-3.80	-7.20	-5.40
	High	0.1	0.9	4.43	-8.25	-2.16	-2.63	-10.48	-7.48	-2.75	-10.03	-7.21
		0.5	0.5	2.00	-14.84	-6.26	-3.97	-12.64	-6.72	-5.17	-9.33	-6.67
		0.9	0.1	-1.04	-7.41	-4.45	-0.14	-10.17	-4.80	-2.40	-6.36	-4.54
		1.0	0.0	-3.94	-7.54	-5.33	-2.42	-6.51	-4.33	-3.94	-7.54	-5.33

Table A16 Percentage deviation of the VNS results from the NEH solutions for the 20x10 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST100	Low	0.1	0.9	3.04	-8.84	-3.97	-4.42	-9.85	-7.02	-3.46	-8.77	-6.12
		0.5	0.5	-1.16	-7.74	-5.31	0.10	-14.25	-5.72	-2.47	-8.21	-5.52
		0.9	0.1	-4.39	-7.95	-6.31	0.68	-11.32	-5.35	-4.48	-8.02	-6.30
		1.0	0.0	-3.49	-8.01	-5.45	-3.56	-16.29	-7.81	-3.49	-8.01	-5.45
	Medium	0.1	0.9	4.41	-8.65	-2.06	-2.16	-8.58	-6.22	-2.19	-8.01	-5.99
		0.5	0.5	3.55	-12.65	-4.94	-1.09	-9.83	-5.39	-2.56	-9.13	-5.27
		0.9	0.1	-1.95	-6.45	-4.92	1.68	-11.46	-4.01	-2.18	-6.26	-4.76
		1.0	0.0	-3.62	-8.34	-5.95	-3.07	-13.19	-8.02	-3.62	-8.34	-5.95
	High	0.1	0.9	6.60	-6.87	-1.52	-3.86	-8.80	-6.42	-3.73	-8.73	-6.26
		0.5	0.5	2.82	-9.30	-3.41	-2.21	-8.85	-5.69	-3.51	-7.86	-5.20
		0.9	0.1	0.14	-10.72	-5.29	1.29	-10.76	-5.43	-1.36	-8.06	-5.37
		1.0	0.0	-3.65	-8.34	-5.91	-2.90	-13.03	-8.32	-3.65	-8.34	-5.91
SDST125	Low	0.1	0.9	0.05	-11.64	-6.26	-2.37	-8.74	-6.24	-3.59	-9.33	-6.22
		0.5	0.5	-2.99	-9.75	-6.74	-1.26	-9.33	-5.67	-3.49	-9.39	-6.50
		0.9	0.1	-2.87	-10.87	-6.68	3.34	-15.93	-8.06	-2.68	-10.73	-6.75
		1.0	0.0	-4.07	-9.39	-7.12	-1.05	-12.85	-7.14	-4.07	-9.39	-7.12
	Medium	0.1	0.9	4.84	-11.15	-3.88	-2.53	-8.86	-5.89	-2.64	-8.82	-5.79
		0.5	0.5	0.99	-12.04	-5.52	0.12	-8.34	-4.61	-2.86	-7.38	-4.94
		0.9	0.1	-0.63	-7.16	-4.93	1.90	-8.26	-2.96	-1.07	-7.39	-4.52
		1.0	0.0	-4.25	-10.87	-7.33	0.24	-11.94	-6.11	-4.25	-10.87	-7.33
	High	0.1	0.9	4.06	-15.89	-4.48	-1.11	-8.86	-5.97	-1.12	-8.77	-5.93
		0.5	0.5	0.60	-11.71	-4.22	-2.17	-9.09	-6.24	-2.74	-7.81	-5.84
		0.9	0.1	-2.98	-8.92	-5.27	-4.15	-8.85	-5.76	-3.76	-7.83	-5.43
		1.0	0.0	-4.31	-9.29	-7.08	-1.77	-9.62	-6.83	-4.31	-9.29	-7.08

Table A17 Percentage deviation of the VNS results from the NEH solutions for the 20x20 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	Low	0.1	0.9	3.60	-11.54	-4.22	7.20	-13.73	-4.35	-2.94	-6.33	-4.49
		0.5	0.5	-1.78	-6.11	-3.70	11.11	-4.93	3.96	-1.36	-5.03	-3.00
		0.9	0.1	-1.35	-5.32	-3.33	12.21	-5.64	3.47	-1.27	-5.32	-3.25
		1.0	0.0	-1.35	-4.63	-3.24	9.60	-3.45	2.39	-1.35	-4.63	-3.24
	Medium	0.1	0.9	3.42	-3.21	0.01	-6.63	-10.91	-8.85	-5.61	-8.73	-7.29
		0.5	0.5	1.03	-7.46	-3.59	-2.35	-13.53	-6.01	-3.18	-5.64	-4.54
		0.9	0.1	-1.13	-5.25	-3.35	5.48	-6.18	0.12	-1.59	-4.91	-3.07
		1.0	0.0	-1.35	-4.54	-3.18	8.39	-0.64	4.44	-1.35	-4.54	-3.18
	High	0.1	0.9	3.93	-3.04	0.14	-4.00	-13.45	-7.75	-3.74	-12.28	-6.95
		0.5	0.5	0.16	-9.18	-3.64	-2.90	-8.86	-6.15	-3.36	-6.68	-4.91
		0.9	0.1	-2.30	-4.86	-3.66	5.73	-7.41	-0.92	-2.20	-4.15	-3.32
		1.0	0.0	-1.35	-4.54	-3.25	10.11	-1.09	4.09	-1.35	-4.54	-3.25
SDST50	Low	0.1	0.9	1.04	-10.44	-3.20	-0.03	-5.99	-4.03	-2.37	-5.38	-3.78
		0.5	0.5	-0.57	-5.60	-3.25	0.11	-5.35	-3.27	-1.71	-5.43	-3.29
		0.9	0.1	-1.32	-5.59	-3.06	1.96	-7.49	-1.36	-1.37	-5.44	-3.01
		1.0	0.0	-1.94	-5.31	-3.34	4.69	-7.08	0.08	-1.94	-5.31	-3.34
	Medium	0.1	0.9	1.77	-5.18	-0.66	-2.76	-8.98	-5.47	-2.48	-8.60	-5.16
		0.5	0.5	-1.38	-11.31	-5.24	0.67	-6.35	-3.30	-2.63	-5.16	-4.05
		0.9	0.1	-1.29	-7.71	-3.43	-1.79	-7.00	-4.08	-1.91	-7.43	-3.56
		1.0	0.0	-1.96	-5.54	-3.29	1.65	-5.17	-1.66	-1.96	-5.54	-3.29
	High	0.1	0.9	3.81	-2.59	0.69	-2.44	-7.90	-5.37	-2.22	-7.57	-5.16
		0.5	0.5	-0.35	-6.12	-2.45	-2.55	-7.40	-4.55	-2.15	-6.03	-4.06
		0.9	0.1	-0.73	-6.75	-4.40	0.42	-5.39	-1.85	-1.01	-5.71	-3.68
		1.0	0.0	-1.73	-5.36	-3.15	2.84	-5.04	-1.41	-1.73	-5.36	-3.15

Table A18 Percentage deviation of the VNS results from the NEH solutions for the 20x20 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST100	**Low**	0.1	0.9	1.81	-9.62	-3.14	-2.46	-6.85	-4.27	-1.95	-6.27	-4.07
		0.5	0.5	-2.41	-4.32	-3.33	-0.25	-6.15	-2.90	-1.99	-4.30	-3.22
		0.9	0.1	-2.67	-6.78	-4.30	3.23	-6.40	-2.11	-2.58	-6.68	-4.20
		1.0	0.0	-2.27	-6.86	-4.38	-1.27	-8.35	-4.36	-2.27	-6.86	-4.38
	Medium	0.1	0.9	3.45	-7.57	-1.03	-2.32	-6.42	-4.52	-2.15	-6.40	-4.39
		0.5	0.5	1.47	-6.47	-2.95	-2.14	-6.36	-4.12	-2.54	-5.58	-3.83
		0.9	0.1	-0.45	-6.53	-3.93	0.34	-6.18	-2.74	-1.99	-5.25	-3.63
		1.0	0.0	-1.93	-6.97	-4.32	-0.67	-10.12	-3.57	-1.93	-6.97	-4.32
	High	0.1	0.9	3.52	-5.82	0.00	-3.58	-7.24	-4.85	-3.62	-7.04	-4.75
		0.5	0.5	2.67	-8.70	-4.62	-1.26	-6.24	-3.61	-1.58	-5.08	-3.78
		0.9	0.1	0.64	-8.90	-4.11	1.34	-4.85	-2.41	-1.27	-5.27	-3.48
		1.0	0.0	-2.54	-6.83	-4.29	0.71	-10.97	-4.36	-2.54	-6.83	-4.29
SDST125	**Low**	0.1	0.9	0.70	-9.52	-2.98	-1.28	-7.10	-4.18	-1.52	-6.05	-3.93
		0.5	0.5	-2.84	-7.70	-4.35	2.36	-7.50	-3.35	-2.85	-4.78	-4.02
		0.9	0.1	-2.22	-7.26	-4.49	-1.35	-9.55	-4.66	-2.47	-6.95	-4.51
		1.0	0.0	-2.46	-7.22	-4.34	-0.57	-9.97	-4.60	-2.46	-7.22	-4.34
	Medium	0.1	0.9	4.24	-9.11	-1.69	-2.07	-5.88	-3.94	-1.93	-5.76	-3.86
		0.5	0.5	1.29	-7.61	-3.16	-1.10	-6.36	-4.08	-2.50	-5.56	-3.90
		0.9	0.1	-0.84	-7.15	-3.64	1.21	-4.84	-2.12	-1.16	-5.19	-3.21
		1.0	0.0	-2.44	-7.04	-4.50	1.44	-8.39	-4.20	-2.44	-7.04	-4.50
	High	0.1	0.9	8.13	-8.61	0.46	-2.03	-6.49	-4.29	-1.85	-6.40	-4.21
		0.5	0.5	0.23	-7.95	-3.18	-1.75	-5.82	-4.05	-1.54	-5.51	-3.93
		0.9	0.1	-0.19	-8.81	-4.38	0.96	-6.94	-3.78	-1.41	-5.59	-4.16
		1.0	0.0	-2.70	-7.22	-4.38	0.00	-9.78	-4.29	-2.70	-7.22	-4.38

Table A19 Percentage deviation of the VNS results from the NEH solutions for the 50x5 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	**Low**	0.1	0.9	-2.19	-9.59	-6.89	1.81	-12.65	-4.75	-3.04	-8.77	-6.66
		0.5	0.5	-4.79	-9.99	-6.90	3.78	-15.96	-6.41	-4.97	-9.90	-6.89
		0.9	0.1	-4.27	-8.24	-5.90	-1.59	-16.86	-7.95	-4.26	-8.25	-5.90
		1.0	0.0	-4.34	-8.46	-5.58	0.44	-10.55	-6.08	-4.34	-8.46	-5.58
	Medium	0.1	0.9	-2.93	-15.31	-9.38	-3.39	-13.44	-7.90	-5.83	-13.92	-8.78
		0.5	0.5	-4.65	-8.84	-6.47	4.43	-15.98	-8.20	-5.04	-7.98	-6.65
		0.9	0.1	-3.61	-7.96	-5.97	0.83	-9.92	-5.69	-3.59	-7.97	-5.96
		1.0	0.0	-4.58	-8.29	-5.60	6.01	-17.01	-4.94	-4.58	-8.29	-5.60
	High	0.1	0.9	-3.08	-13.81	-6.77	-4.68	-13.81	-9.01	-5.08	-9.47	-8.09
		0.5	0.5	-4.17	-9.80	-7.24	-1.13	-16.14	-7.10	-5.17	-8.88	-7.26
		0.9	0.1	-4.14	-8.10	-6.05	0.50	-10.45	-5.46	-4.24	-7.93	-6.03
		1.0	0.0	-4.49	-8.02	-5.87	4.68	-14.25	-5.94	-4.49	-8.02	-5.87
SDST50	**Low**	0.1	0.9	-4.59	-8.33	-6.38	-3.26	-14.02	-8.31	-4.97	-8.83	-6.95
		0.5	0.5	-3.35	-7.56	-6.26	-7.20	-18.00	-11.61	-3.69	-7.57	-6.53
		0.9	0.1	-4.05	-8.73	-6.19	-1.47	-17.29	-8.76	-4.07	-8.77	-6.20
		1.0	0.0	-4.90	-7.32	-6.29	-2.44	-15.09	-8.64	-4.90	-7.32	-6.29
	Medium	0.1	0.9	-1.11	-9.19	-5.29	-4.07	-14.22	-10.16	-5.02	-11.12	-8.82
		0.5	0.5	-2.79	-9.68	-6.55	2.02	-12.89	-8.51	-3.27	-9.48	-7.08
		0.9	0.1	-4.17	-9.62	-6.75	-5.26	-14.79	-10.54	-4.21	-9.59	-6.91
		1.0	0.0	-4.03	-7.55	-6.10	3.55	-14.45	-7.32	-4.03	-7.55	-6.10
	High	0.1	0.9	0.38	-8.23	-3.64	-7.22	-13.18	-10.50	-6.99	-11.44	-9.35
		0.5	0.5	-3.78	-8.04	-6.04	-5.91	-12.15	-9.28	-5.26	-9.48	-7.29
		0.9	0.1	-4.77	-9.07	-6.43	-5.64	-15.17	-10.11	-5.16	-9.52	-6.70
		1.0	0.0	-4.35	-8.15	-6.04	-4.63	-13.62	-8.85	-4.35	-8.15	-6.04

Table A20 Percentage deviation of the VNS results from the NEH solutions for the 50x5 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST100	Low	0.1	0.9	-2.46	-12.16	-6.93	-3.29	-15.61	-8.79	-4.70	-10.65	-7.74
		0.5	0.5	-5.60	-9.92	-7.45	-2.79	-15.52	-9.78	-6.02	-9.57	-7.66
		0.9	0.1	-4.19	-9.69	-7.32	-6.66	-14.56	-9.96	-4.21	-9.73	-7.34
		1.0	0.0	-3.32	-11.07	-6.94	-2.96	-11.94	-8.80	-3.32	-11.07	-6.94
	Medium	0.1	0.9	1.35	-9.97	-4.65	-3.83	-10.10	-7.01	-3.55	-9.69	-6.55
		0.5	0.5	-2.55	-9.21	-6.73	-1.94	-13.00	-8.45	-4.89	-10.40	-7.29
		0.9	0.1	-5.38	-11.16	-8.15	-4.31	-18.79	-9.89	-5.31	-11.19	-8.25
		1.0	0.0	-6.02	-9.74	-7.92	-1.59	-17.00	-10.63	-6.02	-9.74	-7.92
	High	0.1	0.9	2.51	-5.50	-2.72	-5.54	-15.35	-9.67	-5.36	-14.21	-8.89
		0.5	0.5	-3.77	-7.90	-5.89	-1.06	-11.34	-7.88	-4.35	-9.52	-6.87
		0.9	0.1	-4.37	-10.93	-7.77	-8.02	-15.48	-10.00	-4.87	-11.38	-8.00
		1.0	0.0	-4.56	-9.11	-6.87	-2.12	-15.92	-9.10	-4.56	-9.11	-6.87
SDST125	Low	0.1	0.9	-2.32	-12.04	-7.43	-3.31	-16.15	-7.67	-2.77	-12.96	-7.48
		0.5	0.5	-5.59	-9.47	-7.76	-4.99	-20.30	-10.20	-5.62	-10.57	-7.93
		0.9	0.1	-6.57	-11.46	-9.05	-1.43	-15.54	-9.73	-6.59	-11.48	-9.06
		1.0	0.0	-4.37	-9.88	-7.45	-5.41	-14.34	-9.51	-4.37	-9.88	-7.45
	Medium	0.1	0.9	1.03	-11.49	-3.90	-3.74	-15.69	-8.13	-4.35	-14.16	-7.42
		0.5	0.5	-4.33	-9.39	-6.37	-7.66	-14.54	-10.57	-5.89	-9.69	-7.92
		0.9	0.1	-4.40	-10.69	-7.86	-3.79	-15.02	-9.31	-4.39	-10.87	-7.96
		1.0	0.0	-4.32	-11.79	-7.87	-0.70	-14.80	-8.85	-4.32	-11.79	-7.87
	High	0.1	0.9	1.80	-8.46	-3.38	-6.92	-12.63	-9.71	-6.55	-11.64	-9.07
		0.5	0.5	-3.60	-12.70	-6.79	-2.93	-12.98	-9.38	-4.28	-10.79	-8.12
		0.9	0.1	-3.80	-9.48	-7.01	-5.40	-14.92	-8.58	-3.97	-9.30	-7.20
		1.0	0.0	-3.33	-10.14	-6.72	-2.62	-12.87	-8.64	-3.33	-10.14	-6.72

Table A21 Percentage deviation of the VNS results from the NEH solutions for the 50x10 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	Low	0.1	0.9	-3.61	-7.41	-5.05	3.10	-10.81	-4.54	-3.59	-6.51	-4.93
		0.5	0.5	-1.87	-6.12	-4.44	3.83	-9.60	-3.87	-1.96	-5.91	-4.42
		0.9	0.1	-2.32	-6.69	-4.30	3.52	-7.29	-1.76	-2.31	-6.66	-4.29
		1.0	0.0	-3.16	-6.12	-4.33	2.56	-4.83	-1.12	-3.16	-6.12	-4.33
	Medium	0.1	0.9	-1.20	-5.97	-2.77	-5.89	-14.11	-9.97	-4.10	-9.46	-7.09
		0.5	0.5	-2.97	-7.61	-4.89	1.79	-10.89	-4.45	-3.53	-5.91	-4.83
		0.9	0.1	-3.90	-5.85	-4.57	4.19	-10.26	-2.17	-3.75	-5.80	-4.50
		1.0	0.0	-2.23	-6.54	-4.01	6.17	-5.68	-1.33	-2.23	-6.54	-4.01
	High	0.1	0.9	3.26	-6.95	-2.69	-5.64	-15.61	-9.58	-4.70	-12.73	-7.77
		0.5	0.5	-2.63	-7.83	-5.07	-0.12	-9.14	-5.81	-4.00	-6.95	-5.29
		0.9	0.1	-3.24	-5.95	-4.61	11.28	-10.35	-3.19	-3.13	-5.62	-4.55
		1.0	0.0	-2.66	-6.48	-4.17	2.56	-10.77	-3.47	-2.66	-6.48	-4.17
SDST50	Low	0.1	0.9	-1.76	-8.88	-5.17	-2.07	-9.38	-5.35	-2.71	-8.03	-5.21
		0.5	0.5	-2.57	-5.50	-4.16	-0.67	-10.13	-4.87	-2.39	-5.25	-4.22
		0.9	0.1	-2.12	-4.75	-3.95	0.31	-13.00	-5.84	-2.15	-4.82	-3.97
		1.0	0.0	-2.63	-5.57	-4.43	-0.10	-10.00	-5.40	-2.63	-5.57	-4.43
	Medium	0.1	0.9	-0.94	-10.04	-4.11	-1.31	-9.68	-5.76	-2.91	-9.26	-5.50
		0.5	0.5	-2.45	-6.99	-3.71	-1.63	-8.59	-5.31	-2.59	-6.61	-4.33
		0.9	0.1	-2.98	-5.52	-4.33	-1.34	-8.49	-5.44	-3.02	-5.75	-4.42
		1.0	0.0	-2.81	-5.39	-4.20	0.01	-8.10	-4.16	-2.81	-5.39	-4.20
	High	0.1	0.9	0.88	-4.38	-1.40	-4.19	-10.16	-6.94	-4.04	-9.32	-6.40
		0.5	0.5	-3.13	-8.56	-5.65	-3.77	-8.41	-5.98	-4.80	-7.90	-5.83
		0.9	0.1	-3.10	-6.06	-4.15	0.77	-9.20	-4.62	-3.12	-5.96	-4.22
		1.0	0.0	-2.60	-6.03	-4.17	-0.96	-6.37	-3.49	-2.60	-6.03	-4.17

Table A22 Percentage deviation of the VNS results from the NEH solutions for the 50x10 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST100	Low	0.1	0.9	-3.33	-8.17	-5.06	-2.20	-8.22	-5.09	-3.32	-6.64	-5.08
		0.5	0.5	-2.17	-6.73	-5.15	-3.70	-10.72	-6.63	-2.45	-7.02	-5.32
		0.9	0.1	-3.35	-5.90	-4.50	3.20	-9.75	-3.50	-3.43	-5.89	-4.49
		1.0	0.0	-3.23	-7.17	-5.15	-1.52	-9.94	-5.55	-3.23	-7.17	-5.15
	Medium	0.1	0.9	4.75	-6.27	-1.15	-3.10	-8.26	-5.10	-3.38	-6.87	-4.66
		0.5	0.5	-2.79	-7.41	-5.23	-3.50	-7.79	-5.54	-4.28	-6.25	-5.38
		0.9	0.1	-3.34	-5.42	-4.39	1.88	-10.65	-5.13	-2.97	-5.65	-4.48
		1.0	0.0	-4.19	-6.66	-5.16	-0.96	-13.93	-7.00	-4.19	-6.66	-5.16
	High	0.1	0.9	2.87	-7.80	-2.49	-3.16	-8.39	-5.60	-3.26	-7.69	-5.40
		0.5	0.5	0.37	-6.26	-3.55	-1.95	-7.40	-5.71	-2.74	-6.52	-4.95
		0.9	0.1	-3.12	-8.71	-4.98	-2.28	-8.69	-4.81	-3.28	-7.90	-4.96
		1.0	0.0	-2.74	-6.90	-4.64	-1.64	-11.36	-6.40	-2.74	-6.90	-4.64
SDST125	Low	0.1	0.9	-0.79	-7.70	-3.73	-2.80	-7.86	-5.55	-3.61	-5.99	-4.84
		0.5	0.5	-3.25	-6.65	-5.07	4.10	-9.94	-5.07	-2.23	-6.95	-5.06
		0.9	0.1	-3.08	-7.14	-4.84	0.26	-9.58	-4.78	-3.05	-7.15	-4.84
		1.0	0.0	-3.63	-6.98	-5.60	-2.89	-9.59	-4.86	-3.63	-6.98	-5.60
	Medium	0.1	0.9	-1.36	-7.61	-4.10	-4.97	-8.19	-6.58	-4.69	-7.60	-6.33
		0.5	0.5	-2.35	-6.06	-4.18	-2.24	-10.07	-6.73	-3.62	-6.60	-5.54
		0.9	0.1	-4.08	-7.77	-5.63	1.71	-11.17	-5.74	-3.92	-7.78	-5.65
		1.0	0.0	-4.50	-8.01	-6.29	1.32	-9.10	-6.34	-4.50	-8.01	-6.29
	High	0.1	0.9	9.06	-13.25	-2.32	-3.34	-10.60	-5.79	-3.38	-9.98	-5.61
		0.5	0.5	-1.42	-7.23	-3.54	-2.38	-8.25	-5.26	-2.05	-6.87	-4.68
		0.9	0.1	-2.64	-6.24	-4.77	-0.61	-7.65	-4.47	-3.08	-6.45	-4.71
		1.0	0.0	-3.69	-8.02	-5.11	1.16	-12.05	-4.92	-3.69	-8.02	-5.11

Table A23 Percentage deviation of the VNS results from the NEH solutions for the 50x20 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	**Low**	0.1	0.9	-1.42	-6.34	-4.03	-0.06	-10.80	-4.38	-3.21	-5.41	-4.17
		0.5	0.5	-2.11	-5.45	-3.44	7.62	-3.83	-0.14	-2.14	-5.14	-3.26
		0.9	0.1	-2.46	-5.33	-4.21	9.69	-7.43	-0.66	-2.50	-5.32	-4.19
		1.0	0.0	-2.52	-5.18	-4.05	12.86	-6.68	-1.24	-2.52	-5.18	-4.05
	Medium	0.1	0.9	0.36	-4.00	-1.34	-3.70	-10.06	-6.99	-2.58	-7.77	-5.47
		0.5	0.5	-2.04	-4.58	-3.15	-0.48	-7.32	-4.86	-2.77	-4.97	-3.63
		0.9	0.1	-2.21	-5.40	-3.70	11.49	-3.11	1.07	-2.22	-5.02	-3.49
		1.0	0.0	-2.50	-5.36	-4.03	8.06	-4.47	-0.46	-2.50	-5.36	-4.03
	High	0.1	0.9	0.48	-5.70	-1.63	-4.32	-8.51	-6.94	-4.40	-7.69	-6.05
		0.5	0.5	-2.21	-10.24	-4.26	-0.04	-8.79	-4.67	-2.72	-6.29	-4.44
		0.9	0.1	-2.46	-6.68	-3.92	3.98	-3.04	-0.25	-2.42	-5.88	-3.64
		1.0	0.0	-2.96	-5.54	-4.06	8.29	-4.09	0.65	-2.96	-5.54	-4.06
SDST50	**Low**	0.1	0.9	-0.84	-4.96	-2.96	-2.13	-7.26	-4.56	-1.99	-4.67	-3.88
		0.5	0.5	-1.93	-5.37	-3.45	-0.10	-5.93	-2.57	-1.93	-5.05	-3.32
		0.9	0.1	-2.15	-5.85	-3.86	1.65	-4.64	-1.92	-2.12	-5.78	-3.82
		1.0	0.0	-1.40	-4.77	-3.27	0.23	-7.70	-2.98	-1.40	-4.77	-3.27
	Medium	0.1	0.9	0.88	-4.48	-1.98	-3.43	-5.54	-4.32	-3.34	-5.08	-4.08
		0.5	0.5	-1.22	-4.74	-3.34	-0.59	-4.81	-2.96	-2.57	-3.76	-3.15
		0.9	0.1	-1.28	-5.51	-3.19	0.70	-6.91	-2.69	-1.61	-5.07	-3.14
		1.0	0.0	-2.19	-5.12	-3.71	0.21	-6.34	-3.38	-2.19	-5.12	-3.71
	High	0.1	0.9	-0.16	-3.83	-2.20	-1.59	-6.29	-3.88	-1.65	-5.97	-3.78
		0.5	0.5	0.37	-8.06	-3.81	0.31	-7.06	-3.48	-1.22	-4.86	-3.63
		0.9	0.1	-1.81	-5.14	-3.22	0.85	-4.34	-1.68	-2.12	-4.16	-2.92
		1.0	0.0	-1.43	-4.86	-3.50	-0.15	-9.22	-3.36	-1.43	-4.86	-3.50

Table A24 Percentage deviation of the VNS results from the NEH solutions for the 50x20 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	Min.	Avg.	PD (%) of En. Cost Max.	Min.	Avg.	PD (%) of Total Cost Max.	Min.	Avg.
SDST100	Low	0.1	0.9	0.24	-4.74	-2.27	-3.03	-6.03	-4.36	-2.93	-5.03	-3.64
		0.5	0.5	-2.34	-4.93	-3.58	-0.90	-5.48	-3.35	-2.39	-4.68	-3.53
		0.9	0.1	-3.01	-5.14	-3.89	-0.43	-6.41	-3.00	-2.97	-5.05	-3.87
		1.0	0.0	-2.12	-5.31	-3.54	1.55	-7.30	-3.57	-2.12	-5.31	-3.54
	Medium	0.1	0.9	1.42	-2.22	-0.63	-2.95	-5.40	-3.94	-2.79	-5.07	-3.72
		0.5	0.5	-0.74	-4.87	-2.60	-1.58	-4.70	-3.35	-1.78	-4.38	-3.07
		0.9	0.1	-1.29	-4.58	-3.46	-1.29	-6.14	-3.89	-2.04	-4.63	-3.53
		1.0	0.0	-2.07	-4.88	-3.68	0.19	-6.54	-2.22	-2.07	-4.88	-3.68
	High	0.1	0.9	2.61	-4.50	-1.67	-3.00	-5.52	-4.29	-2.89	-5.49	-4.19
		0.5	0.5	0.36	-5.14	-3.19	-2.27	-4.94	-3.52	-2.20	-3.92	-3.45
		0.9	0.1	-2.43	-6.21	-4.18	0.11	-4.71	-2.72	-2.41	-5.21	-3.79
		1.0	0.0	-2.46	-5.42	-4.02	-0.04	-6.23	-3.20	-2.46	-5.42	-4.02
SDST125	Low	0.1	0.9	-0.47	-5.37	-3.14	-1.92	-4.31	-3.31	-2.07	-4.61	-3.26
		0.5	0.5	-1.82	-5.47	-3.30	1.60	-5.83	-2.19	-1.76	-4.84	-3.07
		0.9	0.1	-1.66	-6.31	-3.81	-1.32	-9.14	-4.74	-1.68	-6.33	-3.84
		1.0	0.0	-1.69	-6.00	-3.70	-2.74	-8.00	-4.91	-1.69	-6.00	-3.70
	Medium	0.1	0.9	2.96	-7.63	-1.67	-2.52	-4.99	-3.59	-2.39	-4.60	-3.47
		0.5	0.5	-1.03	-5.56	-3.00	-1.02	-5.42	-3.18	-1.35	-4.81	-3.12
		0.9	0.1	-2.53	-4.18	-3.51	1.25	-5.70	-2.29	-2.33	-4.26	-3.29
		1.0	0.0	-2.20	-5.93	-3.35	-0.67	-5.78	-3.27	-2.20	-5.93	-3.35
	High	0.1	0.9	-0.18	-8.29	-2.30	-2.02	-4.55	-3.58	-1.99	-4.49	-3.54
		0.5	0.5	0.98	-7.26	-2.27	-1.44	-5.23	-3.66	-1.65	-5.08	-3.35
		0.9	0.1	-2.70	-5.80	-3.72	-1.24	-3.82	-3.03	-2.61	-4.99	-3.52
		1.0	0.0	-2.25	-5.62	-3.50	-0.86	-6.81	-4.09	-2.25	-5.62	-3.50

Table A25 Percentage deviation of the VNS results from the NEH solutions for the 100x5 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	**Low**	0.1	0.9	-4.08	-6.88	-5.09	-2.97	-11.66	-6.67	-4.01	-6.75	-5.21
		0.5	0.5	-4.20	-7.20	-5.74	1.48	-9.87	-4.82	-4.14	-7.18	-5.74
		0.9	0.1	-4.65	-9.87	-5.75	-1.95	-11.97	-6.70	-4.65	-9.87	-5.75
		1.0	0.0	-2.55	-5.87	-4.41	0.82	-15.41	-6.66	-2.54	-5.87	-4.41
	Medium	0.1	0.9	-2.22	-7.98	-4.88	-6.80	-15.08	-9.49	-3.95	-9.72	-6.30
		0.5	0.5	-5.29	-7.99	-6.41	-2.71	-12.47	-6.67	-5.27	-7.83	-6.42
		0.9	0.1	-4.16	-7.00	-5.70	-2.44	-15.35	-7.56	-4.19	-7.01	-5.72
		1.0	0.0	-2.69	-5.34	-4.29	-2.33	-12.42	-6.06	-2.69	-5.34	-4.29
	High	0.1	0.9	-3.79	-10.74	-6.27	-4.12	-15.42	-7.97	-5.47	-9.78	-7.05
		0.5	0.5	-3.17	-8.53	-5.76	0.47	-12.11	-6.58	-3.37	-8.56	-5.85
		0.9	0.1	-3.72	-6.05	-4.92	-3.11	-12.12	-7.19	-3.71	-6.02	-4.96
		1.0	0.0	-3.52	-5.89	-4.68	-3.27	-13.08	-6.98	-3.52	-5.89	-4.68
SDST50	**Low**	0.1	0.9	-3.27	-7.18	-5.00	-3.59	-12.47	-6.97	-3.79	-7.18	-5.38
		0.5	0.5	-3.33	-6.88	-5.50	-4.03	-14.33	-9.82	-3.41	-7.02	-5.61
		0.9	0.1	-3.06	-6.81	-5.03	-8.78	-13.26	-10.78	-3.09	-6.82	-5.05
		1.0	0.0	-3.10	-7.45	-4.95	-4.37	-12.42	-7.97	-3.10	-7.45	-4.95
	Medium	0.1	0.9	-2.06	-7.70	-4.46	-5.27	-12.88	-8.96	-5.12	-9.85	-7.10
		0.5	0.5	-3.12	-7.62	-4.81	-3.28	-12.88	-7.59	-3.51	-7.64	-5.24
		0.9	0.1	-2.50	-8.30	-5.52	-3.31	-13.16	-8.76	-2.53	-8.41	-5.59
		1.0	0.0	-1.21	-5.60	-4.08	-0.75	-9.87	-6.51	-1.21	-5.60	-4.08
	High	0.1	0.9	-0.30	-5.83	-3.35	-5.07	-12.78	-8.37	-3.89	-10.82	-6.98
		0.5	0.5	-3.56	-8.65	-5.26	-5.48	-14.30	-8.57	-4.47	-8.33	-6.08
		0.9	0.1	-4.05	-6.54	-5.32	-8.07	-12.14	-9.78	-4.21	-6.69	-5.49
		1.0	0.0	-2.39	-6.24	-4.65	2.18	-11.22	-7.26	-2.39	-6.24	-4.65

Table A26 Percentage deviation of the VNS results from the NEH solutions for the 100x5 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST100	Low	0.1	0.9	-4.28	-9.48	-5.87	-3.81	-13.73	-6.87	-4.30	-10.38	-6.09
		0.5	0.5	-3.51	-8.36	-5.91	-4.40	-13.75	-8.28	-3.66	-8.43	-6.00
		0.9	0.1	-4.58	-9.65	-6.64	-4.64	-16.87	-10.67	-4.61	-9.70	-6.66
		1.0	0.0	-2.81	-7.76	-5.15	-1.27	-9.89	-5.65	-2.81	-7.76	-5.15
	Medium	0.1	0.9	-0.95	-5.68	-3.86	-5.76	-11.40	-7.72	-4.65	-9.14	-6.50
		0.5	0.5	-3.70	-7.17	-5.56	-5.09	-9.95	-7.80	-4.00	-7.60	-6.03
		0.9	0.1	-4.22	-8.35	-6.26	-4.76	-11.48	-8.22	-4.24	-8.42	-6.32
		1.0	0.0	-4.23	-8.15	-5.81	-4.82	-11.23	-7.84	-4.23	-8.15	-5.81
	High	0.1	0.9	-1.15	-6.23	-3.59	-4.68	-9.46	-6.19	-4.21	-8.06	-5.68
		0.5	0.5	-2.30	-6.92	-4.71	-3.07	-11.56	-6.97	-3.22	-7.88	-5.47
		0.9	0.1	-3.77	-7.76	-5.90	-3.89	-11.60	-7.59	-3.78	-7.83	-6.00
		1.0	0.0	-3.38	-6.14	-5.41	-3.30	-11.57	-7.94	-3.38	-6.14	-5.41
SDST125	Low	0.1	0.9	-4.25	-8.59	-6.34	-2.21	-8.60	-6.68	-5.27	-8.47	-6.42
		0.5	0.5	-2.56	-10.75	-5.82	-1.36	-12.57	-7.36	-2.52	-10.81	-5.90
		0.9	0.1	-3.76	-11.77	-6.61	-2.70	-18.63	-8.76	-3.76	-11.80	-6.62
		1.0	0.0	-2.71	-6.81	-5.13	-2.88	-11.61	-7.25	-2.71	-6.81	-5.13
	Medium	0.1	0.9	0.30	-6.63	-3.61	-3.15	-11.68	-7.10	-2.76	-9.13	-6.11
		0.5	0.5	-3.26	-8.38	-5.46	-3.52	-10.69	-7.17	-3.48	-8.90	-5.86
		0.9	0.1	-5.38	-8.59	-6.83	-6.25	-11.97	-9.40	-5.47	-8.51	-6.92
		1.0	0.0	-3.06	-6.99	-5.36	-2.82	-10.06	-7.22	-3.06	-6.99	-5.36
	High	0.1	0.9	-1.43	-5.50	-3.57	-5.19	-10.15	-7.48	-4.90	-8.90	-6.79
		0.5	0.5	-2.87	-7.20	-4.98	-4.75	-11.24	-7.24	-3.70	-8.26	-5.79
		0.9	0.1	-5.00	-8.06	-6.69	-2.40	-11.32	-6.86	-5.15	-8.02	-6.71
		1.0	0.0	-2.58	-6.95	-5.16	-5.95	-8.86	-7.12	-2.58	-6.95	-5.16

Table A27 Percentage deviation of the VNS results from the NEH solutions for the 100x10 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	**Low**	0.1	0.9	-1.88	-5.08	-2.84	0.43	-10.18	-3.82	-1.91	-5.66	-2.97
		0.5	0.5	-1.89	-4.66	-3.32	3.38	-8.23	-1.58	-1.88	-4.53	-3.29
		0.9	0.1	-2.35	-3.98	-3.10	0.13	-7.25	-3.13	-2.35	-3.98	-3.10
		1.0	0.0	-2.52	-4.32	-3.42	1.45	-6.72	-2.17	-2.52	-4.32	-3.42
	Medium	0.1	0.9	-0.96	-5.09	-3.32	-1.26	-9.99	-4.77	-1.81	-6.62	-3.97
		0.5	0.5	-1.78	-4.42	-3.27	-0.67	-6.09	-2.88	-1.86	-4.14	-3.23
		0.9	0.1	-1.83	-4.79	-3.36	3.90	-9.15	-3.16	-1.81	-4.82	-3.36
		1.0	0.0	-2.50	-5.10	-3.46	1.43	-6.34	-2.92	-2.50	-5.10	-3.46
	High	0.1	0.9	-1.38	-5.83	-3.03	-2.90	-7.96	-5.43	-3.03	-5.98	-4.50
		0.5	0.5	-1.67	-5.15	-2.88	-0.73	-5.90	-2.57	-1.77	-4.55	-2.84
		0.9	0.1	-2.18	-4.38	-3.36	1.27	-5.71	-3.01	-2.19	-4.39	-3.35
		1.0	0.0	-3.10	-4.91	-3.63	0.52	-6.40	-3.01	-3.10	-4.91	-3.63
SDST50	**Low**	0.1	0.9	-1.05	-4.10	-2.29	-1.21	-6.39	-3.92	-1.37	-3.87	-2.77
		0.5	0.5	-2.35	-3.70	-2.80	0.14	-8.17	-4.26	-2.49	-3.77	-2.88
		0.9	0.1	-1.71	-4.02	-2.67	-1.94	-7.01	-4.00	-1.71	-4.01	-2.68
		1.0	0.0	-2.42	-3.97	-3.44	-1.80	-8.07	-4.70	-2.42	-3.97	-3.44
	Medium	0.1	0.9	-0.75	-5.19	-2.33	-2.65	-5.17	-4.13	-2.51	-5.18	-3.66
		0.5	0.5	-1.27	-4.47	-2.76	-0.99	-3.99	-2.63	-1.65	-3.93	-2.73
		0.9	0.1	-1.44	-3.84	-2.71	1.11	-6.64	-2.73	-1.34	-3.97	-2.71
		1.0	0.0	-2.16	-5.38	-3.65	-4.07	-8.14	-6.56	-2.16	-5.38	-3.65
	High	0.1	0.9	0.37	-4.79	-1.85	-2.54	-7.59	-4.30	-2.23	-6.51	-3.92
		0.5	0.5	-0.82	-3.75	-2.19	-1.16	-6.22	-3.47	-1.40	-3.48	-2.70
		0.9	0.1	-0.96	-3.78	-2.62	-0.90	-4.94	-2.68	-0.95	-3.77	-2.62
		1.0	0.0	-2.36	-4.03	-3.23	-3.15	-6.61	-5.19	-2.36	-4.03	-3.23

Table A28 Percentage deviation of the VNS results from the NEH solutions for the 100x10 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost			PD (%) of En. Cost			PD (%) of Total Cost		
				Max.	Min.	Avg.	Max.	Min.	Avg.	Max.	Min.	Avg.
SDST100	Low	0.1	0.9	-0.44	-3.96	-2.82	-1.38	-5.64	-3.16	-1.23	-3.89	-2.90
		0.5	0.5	-0.49	-5.99	-3.20	0.56	-7.84	-3.23	-0.40	-5.83	-3.20
		0.9	0.1	-2.16	-5.35	-3.22	-3.04	-6.51	-4.53	-2.17	-5.36	-3.24
		1.0	0.0	-1.88	-6.26	-3.75	-2.91	-7.30	-4.46	-1.88	-6.26	-3.75
	Medium	0.1	0.9	1.24	-3.54	-2.06	-2.77	-5.58	-3.73	-2.11	-5.11	-3.43
		0.5	0.5	-1.14	-3.43	-2.48	-0.31	-5.20	-2.91	-1.51	-4.03	-2.64
		0.9	0.1	-2.01	-4.81	-3.35	-1.52	-5.33	-3.21	-2.00	-4.72	-3.34
		1.0	0.0	-1.72	-6.66	-3.99	1.22	-7.14	-3.82	-1.72	-6.66	-3.99
	High	0.1	0.9	0.09	-5.06	-2.17	-1.87	-5.69	-3.21	-1.84	-5.35	-3.10
		0.5	0.5	-0.96	-4.72	-3.21	-1.62	-6.05	-3.78	-1.82	-4.95	-3.50
		0.9	0.1	-1.86	-4.42	-3.10	-0.60	-4.81	-2.99	-1.93	-4.46	-3.09
		1.0	0.0	-2.57	-6.20	-4.00	-2.51	-5.55	-4.36	-2.57	-6.20	-4.00
SDST125	Low	0.1	0.9	-2.21	-4.83	-3.18	-2.80	-4.66	-3.67	-2.81	-4.25	-3.39
		0.5	0.5	-1.22	-5.96	-3.59	-0.64	-5.54	-3.08	-1.16	-5.61	-3.55
		0.9	0.1	-2.92	-7.14	-3.93	-2.13	-8.15	-4.35	-2.94	-7.16	-3.93
		1.0	0.0	-2.25	-5.17	-3.40	-2.49	-7.17	-4.08	-2.25	-5.17	-3.40
	Medium	0.1	0.9	1.05	-4.28	-2.02	-1.92	-5.52	-3.70	-1.84	-4.85	-3.42
		0.5	0.5	-1.68	-3.85	-2.74	-1.17	-4.22	-2.90	-1.64	-3.65	-2.80
		0.9	0.1	-1.81	-5.38	-3.19	0.14	-5.59	-2.78	-1.76	-5.37	-3.17
		1.0	0.0	-2.15	-4.71	-3.06	1.27	-5.66	-3.36	-2.15	-4.71	-3.06
	High	0.1	0.9	1.96	-6.20	-1.50	-1.26	-5.23	-3.45	-1.34	-4.90	-3.27
		0.5	0.5	0.07	-3.94	-2.43	-1.81	-5.65	-3.34	-1.98	-4.85	-2.91
		0.9	0.1	-0.81	-5.31	-3.46	-0.58	-5.07	-3.49	-0.93	-5.25	-3.47
		1.0	0.0	-2.45	-4.62	-3.61	-2.00	-6.51	-4.50	-2.45	-4.62	-3.61

Table A29 Percentage deviation of the VNS results from the NEH solutions for the 100x20 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	**Low**	0.1	0.9	-1.80	-3.10	-2.57	-0.27	-5.80	-2.20	-1.48	-3.11	-2.49
		0.5	0.5	-1.90	-3.77	-2.51	-0.47	-3.23	-2.20	-1.93	-3.72	-2.50
		0.9	0.1	-1.87	-2.63	-2.25	1.35	-2.57	-0.56	-1.87	-2.62	-2.24
		1.0	0.0	-1.88	-2.68	-2.16	3.63	-5.73	-0.94	-1.88	-2.68	-2.16
	Medium	0.1	0.9	-0.25	-3.03	-1.58	-2.95	-5.79	-4.18	-1.92	-4.47	-3.21
		0.5	0.5	-2.41	-4.47	-3.21	2.84	-3.85	-1.05	-2.63	-3.16	-2.82
		0.9	0.1	-1.54	-2.79	-1.99	0.54	-4.26	-1.16	-1.62	-2.72	-1.97
		1.0	0.0	-1.54	-2.25	-1.74	0.99	-4.12	-1.10	-1.54	-2.25	-1.74
	High	0.1	0.9	-0.31	-1.72	-1.17	-2.89	-5.70	-3.95	-2.52	-4.69	-3.26
		0.5	0.5	-0.68	-3.72	-2.20	-0.42	-4.07	-2.61	-1.21	-3.82	-2.32
		0.9	0.1	-1.55	-3.22	-2.55	1.42	-1.95	-1.10	-1.55	-3.00	-2.48
		1.0	0.0	-1.80	-2.86	-2.40	1.32	-2.87	-0.40	-1.80	-2.86	-2.40
SDST50	**Low**	0.1	0.9	-0.51	-2.71	-1.97	0.58	-3.20	-1.84	-0.70	-2.95	-1.91
		0.5	0.5	-1.38	-2.77	-1.98	0.26	-5.41	-1.95	-1.26	-3.00	-1.97
		0.9	0.1	-1.62	-3.09	-2.27	-1.08	-3.48	-2.07	-1.62	-3.09	-2.27
		1.0	0.0	-0.93	-3.66	-1.92	-0.44	-3.53	-1.64	-0.93	-3.66	-1.92
	Medium	0.1	0.9	-0.90	-3.55	-2.20	-0.96	-5.10	-2.79	-1.39	-4.61	-2.70
		0.5	0.5	-1.39	-2.12	-1.67	-1.21	-4.17	-2.80	-1.51	-2.61	-2.13
		0.9	0.1	-1.71	-3.07	-2.21	-0.42	-1.92	-1.37	-1.67	-2.96	-2.14
		1.0	0.0	-1.02	-2.20	-1.65	-0.22	-2.87	-1.40	-1.02	-2.20	-1.65
	High	0.1	0.9	0.23	-1.69	-0.48	-2.12	-4.11	-2.96	-1.91	-3.78	-2.73
		0.5	0.5	-1.44	-3.31	-2.35	-1.52	-2.70	-2.24	-1.80	-2.98	-2.29
		0.9	0.1	-1.56	-2.77	-2.08	-0.05	-2.81	-1.27	-1.55	-2.78	-1.97
		1.0	0.0	-0.88	-3.24	-2.35	-0.46	-3.23	-1.81	-0.88	-3.24	-2.35

Table A30 Percentage deviation of the VNS results from the NEH solutions for the 100x20 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST100	**Low**	0.1	0.9	-1.67	-4.48	-2.62	0.02	-2.83	-1.31	-1.29	-2.40	-1.88
		0.5	0.5	-1.63	-2.91	-2.32	-0.79	-3.56	-2.14	-1.69	-2.87	-2.30
		0.9	0.1	-1.31	-3.27	-2.19	0.09	-2.91	-1.65	-1.28	-3.26	-2.18
		1.0	0.0	-1.85	-3.57	-2.56	-0.99	-4.82	-3.07	-1.85	-3.57	-2.56
	Medium	0.1	0.9	0.55	-1.77	-1.02	-1.08	-4.34	-2.39	-0.90	-4.06	-2.25
		0.5	0.5	-0.72	-2.47	-1.57	-0.61	-3.81	-2.17	-1.24	-2.51	-1.87
		0.9	0.1	-1.76	-2.92	-2.16	-0.68	-1.89	-1.45	-1.76	-2.80	-2.09
		1.0	0.0	-1.62	-2.90	-2.03	-0.75	-5.04	-2.49	-1.62	-2.90	-2.03
	High	0.1	0.9	-0.16	-2.27	-1.46	-1.63	-2.87	-2.40	-1.66	-2.71	-2.35
		0.5	0.5	-0.29	-3.60	-2.28	-1.84	-3.86	-2.96	-2.14	-3.76	-2.72
		0.9	0.1	-1.22	-3.32	-2.24	-1.17	-4.14	-2.60	-1.46	-3.33	-2.31
		1.0	0.0	-1.99	-2.92	-2.35	-1.83	-4.84	-3.01	-1.99	-2.92	-2.35
SDST125	**Low**	0.1	0.9	-1.78	-2.43	-2.20	-1.17	-2.61	-1.91	-1.43	-2.50	-2.03
		0.5	0.5	-1.10	-3.48	-2.10	-0.76	-1.92	-1.28	-1.16	-3.26	-1.99
		0.9	0.1	-1.01	-3.10	-2.20	1.07	-5.01	-1.51	-0.97	-3.13	-2.19
		1.0	0.0	-2.01	-2.51	-2.30	-0.41	-4.64	-2.40	-2.01	-2.51	-2.30
	Medium	0.1	0.9	1.58	-3.46	-0.88	-1.92	-3.11	-2.45	-1.75	-3.14	-2.30
		0.5	0.5	-1.36	-2.58	-2.05	-0.25	-2.98	-2.10	-1.33	-2.50	-2.08
		0.9	0.1	-1.56	-3.97	-2.37	-0.93	-4.61	-2.22	-1.64	-4.05	-2.35
		1.0	0.0	-1.22	-3.19	-2.35	-0.43	-5.23	-2.74	-1.22	-3.19	-2.35
	High	0.1	0.9	0.61	-2.01	-0.56	-1.15	-3.22	-2.51	-1.20	-3.02	-2.40
		0.5	0.5	-0.56	-4.13	-2.25	-1.59	-2.41	-1.86	-1.28	-2.75	-1.99
		0.9	0.1	-0.61	-2.91	-1.96	-0.37	-3.51	-1.47	-0.58	-2.63	-1.86
		1.0	0.0	-1.43	-2.54	-2.03	0.47	-3.54	-2.07	-1.43	-2.54	-2.03

Table A31 Percentage deviation of the VNS results from the NEH solutions for the 200x10 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	**Low**	0.1	0.9	-1.23	-3.28	-2.12	-0.46	-7.23	-3.07	-1.36	-3.22	-2.18
		0.5	0.5	-1.67	-3.28	-2.46	-0.40	-4.25	-2.46	-1.67	-3.27	-2.46
		0.9	0.1	-1.53	-2.59	-2.10	0.49	-6.54	-2.59	-1.53	-2.59	-2.10
		1.0	0.0	-1.94	-3.15	-2.45	1.00	-6.02	-3.06	-1.94	-3.15	-2.45
	Medium	0.1	0.9	0.27	-3.48	-1.30	-1.16	-7.10	-3.96	-0.97	-3.95	-2.17
		0.5	0.5	-1.34	-2.60	-2.10	-1.76	-3.97	-2.72	-1.37	-2.62	-2.14
		0.9	0.1	-1.32	-2.67	-2.05	1.11	-5.14	-2.05	-1.32	-2.64	-2.05
		1.0	0.0	-1.63	-3.13	-2.33	0.27	-6.03	-3.10	-1.63	-3.13	-2.33
	High	0.1	0.9	-0.26	-2.50	-1.45	-1.83	-3.92	-2.90	-1.23	-2.76	-2.12
		0.5	0.5	-0.93	-2.94	-1.92	-0.18	-4.96	-2.50	-1.35	-2.91	-1.98
		0.9	0.1	-1.58	-3.21	-2.33	-0.26	-5.69	-2.99	-1.60	-3.24	-2.34
		1.0	0.0	-1.39	-2.90	-2.25	0.12	-4.93	-2.75	-1.39	-2.90	-2.25
SDST50	**Low**	0.1	0.9	-1.26	-2.44	-1.73	-0.27	-3.26	-1.47	-1.13	-2.61	-1.69
		0.5	0.5	-1.32	-2.94	-1.92	-0.69	-4.27	-2.55	-1.32	-2.88	-1.94
		0.9	0.1	-1.22	-2.85	-1.97	-0.81	-4.95	-2.18	-1.22	-2.85	-1.97
		1.0	0.0	-1.68	-2.39	-2.09	-0.03	-5.10	-2.62	-1.68	-2.39	-2.09
	Medium	0.1	0.9	-1.28	-4.70	-2.22	-2.10	-3.65	-2.66	-1.80	-3.87	-2.49
		0.5	0.5	-1.15	-2.47	-1.81	0.46	-2.98	-1.38	-1.16	-2.56	-1.75
		0.9	0.1	-1.46	-2.99	-2.08	-1.06	-3.51	-2.08	-1.47	-2.98	-2.08
		1.0	0.0	-1.66	-2.93	-2.17	-1.67	-5.01	-3.19	-1.66	-2.93	-2.17
	High	0.1	0.9	0.00	-2.76	-1.39	-0.97	-3.51	-2.21	-1.04	-3.09	-2.00
		0.5	0.5	-0.48	-2.42	-1.82	0.29	-3.47	-2.26	-0.66	-2.59	-1.94
		0.9	0.1	-0.49	-2.50	-1.44	1.30	-3.60	-1.32	-0.40	-2.49	-1.43
		1.0	0.0	-1.80	-2.50	-2.09	-0.62	-3.91	-2.84	-1.80	-2.50	-2.09

Table A32 Percentage deviation of the VNS results from the NEH solutions for the 200x10 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST100	**Low**	0.1	0.9	0.00	-2.41	-1.13	1.37	-2.76	-0.79	0.00	-2.38	-1.09
		0.5	0.5	-0.06	-2.56	-1.03	0.56	-3.11	-0.68	-0.08	-2.48	-0.96
		0.9	0.1	0.00	-3.27	-1.33	1.87	-3.57	-1.00	0.00	-3.28	-1.31
		1.0	0.0	-1.32	-3.08	-2.19	-1.42	-4.86	-3.07	-1.32	-3.08	-2.19
	Medium	0.1	0.9	0.29	-2.63	-1.30	-0.12	-3.84	-1.51	-0.33	-2.63	-1.36
		0.5	0.5	0.00	-2.96	-1.15	1.93	-2.72	-0.78	0.00	-2.62	-1.04
		0.9	0.1	-0.28	-1.99	-1.02	1.25	-2.76	-0.90	-0.17	-2.17	-0.99
		1.0	0.0	-1.02	-3.65	-2.10	-1.10	-5.46	-2.94	-1.02	-3.65	-2.10
	High	0.1	0.9	0.00	-2.61	-1.23	1.44	-3.13	-0.82	0.00	-2.58	-1.21
		0.5	0.5	0.86	-2.47	-0.64	0.00	-2.84	-1.25	0.00	-2.16	-1.14
		0.9	0.1	0.00	-2.11	-0.97	0.57	-2.10	-0.30	0.00	-2.11	-0.74
		1.0	0.0	-1.67	-3.09	-2.34	-1.27	-4.11	-3.08	-1.67	-3.09	-2.34
SDST125	**Low**	0.1	0.9	0.00	-0.94	-0.52	1.33	-0.17	0.26	0.00	-0.62	-0.30
		0.5	0.5	0.00	-0.70	-0.42	1.88	-0.25	0.40	0.00	-0.62	-0.38
		0.9	0.1	0.00	-1.26	-0.65	0.79	-1.47	-0.32	0.00	-1.26	-0.65
		1.0	0.0	-1.48	-3.85	-2.43	-1.39	-6.20	-3.78	-1.48	-3.85	-2.43
	Medium	0.1	0.9	0.58	-1.29	-0.26	0.11	-1.24	-0.41	0.00	-0.94	-0.37
		0.5	0.5	0.07	-0.93	-0.28	0.39	-0.94	-0.27	0.00	-0.87	-0.28
		0.9	0.1	1.23	-0.95	0.05	1.38	-1.18	0.53	-0.14	-1.76	-0.60
		1.0	0.0	-1.06	-3.97	-2.53	-1.92	-5.51	-3.54	-1.06	-3.97	-2.53
	High	0.1	0.9	0.67	-1.40	-0.37	0.23	-0.47	-0.18	-0.04	-0.50	-0.21
		0.5	0.5	0.02	-0.95	-0.40	0.29	-1.40	-0.28	0.00	-0.79	-0.36
		0.9	0.1	1.31	-0.62	0.63	1.58	-0.87	0.69	-0.36	-0.91	-0.44
		1.0	0.0	-0.66	-3.70	-2.51	-2.35	-5.69	-3.83	-0.66	-3.70	-2.51

Table A33 Percentage deviation of the VNS results from the NEH solutions for the 200x20 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST10	**Low**	0.1	0.9	-0.30	-2.13	-1.29	-1.13	-3.06	-2.28	-0.64	-2.00	-1.42
		0.5	0.5	-1.12	-1.57	-1.40	-0.25	-1.76	-0.93	-1.14	-1.55	-1.39
		0.9	0.1	-1.23	-2.15	-1.85	-2.10	-5.27	-3.15	-1.23	-2.15	-1.85
		1.0	0.0	-1.02	-1.34	-1.20	-0.07	-3.27	-2.21	-1.02	-1.34	-1.20
	Medium	0.1	0.9	-0.09	-2.94	-1.65	-1.35	-4.77	-2.44	-1.54	-2.59	-2.03
		0.5	0.5	-1.03	-2.85	-1.75	1.45	-4.01	-1.20	-1.03	-2.37	-1.69
		0.9	0.1	-0.86	-2.51	-1.45	-0.24	-3.48	-1.38	-0.85	-2.53	-1.45
		1.0	0.0	-1.13	-1.75	-1.41	-0.21	-4.29	-2.43	-1.13	-1.75	-1.41
	High	0.1	0.9	-0.51	-2.02	-1.21	-0.86	-3.24	-2.07	-0.97	-2.32	-1.75
		0.5	0.5	-0.64	-1.62	-1.29	0.45	-3.04	-1.56	-0.77	-1.81	-1.34
		0.9	0.1	-0.96	-2.14	-1.57	-1.23	-3.92	-2.36	-0.97	-2.12	-1.60
		1.0	0.0	-0.91	-1.91	-1.50	-0.36	-3.15	-1.42	-0.91	-1.91	-1.50
SDST50	**Low**	0.1	0.9	-0.47	-1.15	-0.78	0.15	-1.01	-0.44	-0.51	-0.88	-0.66
		0.5	0.5	-0.66	-1.73	-1.43	0.31	-1.53	-0.76	-0.65	-1.65	-1.39
		0.9	0.1	-0.67	-1.62	-1.17	0.05	-2.00	-1.05	-0.68	-1.61	-1.17
		1.0	0.0	-0.53	-1.64	-1.27	-1.03	-3.07	-1.85	-0.53	-1.64	-1.27
	Medium	0.1	0.9	-1.34	-1.97	-1.76	-0.71	-1.70	-1.21	-1.01	-1.71	-1.35
		0.5	0.5	-0.80	-2.06	-1.33	-0.91	-1.87	-1.46	-1.05	-1.74	-1.36
		0.9	0.1	-0.90	-2.00	-1.40	0.35	-1.95	-0.32	-0.86	-2.00	-1.35
		1.0	0.0	-0.68	-1.57	-1.17	-0.50	-2.87	-1.68	-0.68	-1.57	-1.17
	High	0.1	0.9	-0.40	-2.05	-1.20	-0.67	-2.14	-1.26	-0.70	-2.13	-1.25
		0.5	0.5	-0.68	-2.07	-1.38	-0.05	-1.42	-0.52	-0.42	-1.42	-1.03
		0.9	0.1	-1.15	-1.72	-1.43	0.12	-1.98	-0.52	-1.13	-1.63	-1.36
		1.0	0.0	-0.42	-1.41	-1.01	0.05	-1.62	-0.91	-0.42	-1.41	-1.01

Table A34 Percentage deviation of the VNS results from the NEH solutions for the 200x20 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST100	Low	0.1	0.9	-0.63	-1.71	-1.06	-0.74	-2.28	-1.55	-0.83	-1.91	-1.25
		0.5	0.5	-0.69	-1.47	-0.96	0.20	-1.08	-0.36	-0.65	-1.45	-0.92
		0.9	0.1	-1.04	-1.88	-1.48	-0.31	-2.31	-1.46	-1.05	-1.88	-1.48
		1.0	0.0	-0.84	-1.89	-1.31	-0.97	-2.93	-1.95	-0.84	-1.89	-1.31
	Medium	0.1	0.9	-0.13	-1.29	-0.73	-0.94	-1.57	-1.29	-0.79	-1.52	-1.19
		0.5	0.5	-0.22	-1.53	-1.08	-0.15	-1.19	-0.88	-0.19	-1.40	-1.01
		0.9	0.1	-0.62	-1.77	-1.05	0.55	-0.90	-0.32	-0.54	-1.70	-1.01
		1.0	0.0	-0.91	-1.94	-1.40	-1.10	-3.03	-2.05	-0.91	-1.94	-1.40
	High	0.1	0.9	0.15	-1.59	-0.66	-0.89	-1.70	-1.19	-0.78	-1.62	-1.14
		0.5	0.5	-0.24	-1.62	-0.85	0.04	-1.61	-0.97	-0.10	-1.54	-0.91
		0.9	0.1	-0.57	-1.39	-0.90	-0.12	-2.10	-0.79	-0.52	-1.27	-0.89
		1.0	0.0	-0.58	-1.50	-1.06	-0.44	-3.04	-1.73	-0.58	-1.50	-1.06
SDST125	Low	0.1	0.9	-0.41	-1.16	-0.83	-0.43	-1.18	-0.84	-0.48	-1.17	-0.84
		0.5	0.5	-0.66	-1.71	-1.26	0.32	-0.16	0.03	-0.59	-1.58	-1.15
		0.9	0.1	-1.18	-2.12	-1.57	-0.45	-2.05	-1.16	-1.18	-2.11	-1.56
		1.0	0.0	-0.74	-1.82	-1.29	-0.58	-3.13	-1.85	-0.74	-1.82	-1.29
	Medium	0.1	0.9	1.01	-1.74	-0.42	-0.61	-1.88	-1.36	-0.67	-1.86	-1.21
		0.5	0.5	-0.94	-1.97	-1.59	-0.89	-1.76	-1.25	-1.14	-1.72	-1.46
		0.9	0.1	-0.97	-1.52	-1.32	-0.21	-1.38	-0.82	-0.93	-1.51	-1.29
		1.0	0.0	-1.04	-1.91	-1.42	-1.05	-2.31	-1.58	-1.04	-1.91	-1.42
	High	0.1	0.9	0.64	-1.44	-0.94	-1.17	-2.04	-1.53	-1.13	-1.97	-1.48
		0.5	0.5	-0.76	-2.11	-1.33	-0.52	-1.80	-1.11	-1.01	-1.37	-1.21
		0.9	0.1	-0.57	-1.54	-1.13	0.64	-1.15	-0.44	-0.52	-1.46	-1.05
		1.0	0.0	-1.17	-1.90	-1.44	-0.90	-1.97	-1.35	-1.17	-1.90	-1.44

Table A35 Percentage deviation of the VNS results from the NEH solutions for the 500x20 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST10 and SDST50

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	Min.	Avg.	PD (%) of En. Cost Max.	Min.	Avg.	PD (%) of Total Cost Max.	Min.	Avg.
SDST10	**Low**	0.1	0.9	-0.07	-0.50	-0.30	0.35	-0.96	-0.32	-0.07	-0.53	-0.30
		0.5	0.5	-0.23	-0.61	-0.42	-0.21	-1.23	-0.69	-0.23	-0.61	-0.42
		0.9	0.1	-0.32	-0.58	-0.45	0.01	-2.07	-0.91	-0.32	-0.59	-0.45
		1.0	0.0	-0.37	-0.66	-0.53	-0.30	-2.17	-1.40	-0.37	-0.66	-0.53
	Medium	0.1	0.9	-0.13	-0.42	-0.24	-0.12	-0.59	-0.32	-0.15	-0.34	-0.26
		0.5	0.5	-0.37	-0.48	-0.42	0.29	-1.81	-0.70	-0.35	-0.53	-0.44
		0.9	0.1	-0.16	-0.41	-0.27	0.07	-1.31	-0.41	-0.16	-0.42	-0.27
		1.0	0.0	-0.40	-0.68	-0.51	-1.20	-2.35	-1.92	-0.40	-0.68	-0.51
	High	0.1	0.9	-0.22	-1.25	-0.54	0.82	-1.08	-0.35	-0.36	-0.58	-0.46
		0.5	0.5	0.00	-0.73	-0.40	0.07	-1.36	-0.60	0.00	-0.78	-0.41
		0.9	0.1	0.00	-0.53	-0.33	0.00	-2.73	-1.16	0.00	-0.56	-0.34
		1.0	0.0	-0.16	-0.63	-0.40	-1.17	-1.55	-1.38	-0.16	-0.63	-0.40
SDST50	**Low**	0.1	0.9	-0.27	-0.40	-0.32	0.38	-0.31	-0.09	-0.27	-0.30	-0.28
		0.5	0.5	-0.34	-0.42	-0.39	0.23	-0.14	0.03	-0.33	-0.41	-0.38
		0.9	0.1	-0.31	-0.38	-0.34	-0.02	-0.76	-0.37	-0.31	-0.38	-0.34
		1.0	0.0	-0.28	-0.46	-0.33	0.57	-0.49	-0.11	-0.28	-0.46	-0.33
	Medium	0.1	0.9	-0.15	-0.32	-0.21	0.05	-0.41	-0.13	-0.04	-0.37	-0.16
		0.5	0.5	-0.14	-0.39	-0.24	0.10	-0.06	0.03	-0.13	-0.32	-0.20
		0.9	0.1	-0.22	-0.48	-0.38	0.42	-0.87	-0.20	-0.21	-0.46	-0.38
		1.0	0.0	-0.14	-0.44	-0.34	-0.18	-0.92	-0.62	-0.14	-0.44	-0.34
	High	0.1	0.9	-0.03	-0.51	-0.33	0.08	-0.26	-0.08	-0.05	-0.31	-0.15
		0.5	0.5	-0.10	-0.45	-0.27	0.01	-0.43	-0.26	-0.13	-0.45	-0.27
		0.9	0.1	-0.14	-0.41	-0.24	0.47	-0.28	-0.05	-0.14	-0.41	-0.24
		1.0	0.0	-0.13	-0.40	-0.24	0.36	-0.26	0.00	-0.13	-0.40	-0.24

Table A36 Percentage deviation of the VNS results from the NEH solutions for the 500x20 sized $F|s_{ijl}, prmu|\sum(C_j+Energy_j)$ problem when SDST100 and SDST125

Setup Group	En. Data Group	w_1 (%)	w_2 (%)	PD (%) of Prod. Cost Max.	PD (%) of Prod. Cost Min.	PD (%) of Prod. Cost Avg.	PD (%) of En. Cost Max.	PD (%) of En. Cost Min.	PD (%) of En. Cost Avg.	PD (%) of Total Cost Max.	PD (%) of Total Cost Min.	PD (%) of Total Cost Avg.
SDST100	**Low**	0.1	0.9	-0.08	-0.52	-0.26	0.35	-0.02	0.21	0.00	-0.33	-0.15
		0.5	0.5	-0.08	-0.16	-0.12	0.53	0.12	0.29	-0.06	-0.15	-0.10
		0.9	0.1	-0.14	-0.41	-0.25	0.35	-0.81	-0.28	-0.14	-0.41	-0.25
		1.0	0.0	-0.23	-0.43	-0.31	0.04	-0.66	-0.21	-0.23	-0.43	-0.31
	Medium	0.1	0.9	0.04	-0.25	-0.13	0.06	-0.31	-0.07	-0.03	-0.29	-0.09
		0.5	0.5	0.00	-0.41	-0.20	0.18	-0.18	-0.02	0.00	-0.30	-0.17
		0.9	0.1	-0.24	-0.43	-0.35	0.16	-0.28	-0.04	-0.24	-0.42	-0.34
		1.0	0.0	-0.22	-0.61	-0.38	0.27	-1.02	-0.40	-0.22	-0.61	-0.38
	High	0.1	0.9	0.00	-0.67	-0.32	0.16	-0.50	-0.14	0.00	-0.45	-0.18
		0.5	0.5	-0.02	-0.32	-0.20	-0.05	-0.20	-0.15	-0.07	-0.28	-0.19
		0.9	0.1	-0.14	-0.35	-0.26	0.46	-0.09	0.14	-0.12	-0.33	-0.24
		1.0	0.0	-0.09	-0.59	-0.39	-0.27	-0.90	-0.58	-0.09	-0.59	-0.39
SDST125	**Low**	0.1	0.9	0.00	-0.25	-0.09	0.82	0.00	0.23	0.00	-0.05	-0.02
		0.5	0.5	-0.04	-0.33	-0.14	0.39	0.01	0.18	-0.03	-0.32	-0.13
		0.9	0.1	-0.12	-0.42	-0.27	0.43	-0.38	0.12	-0.12	-0.42	-0.26
		1.0	0.0	-0.40	-0.70	-0.55	-0.16	-0.74	-0.48	-0.40	-0.70	-0.55
	Medium	0.1	0.9	0.07	-0.38	-0.20	0.01	-0.28	-0.17	-0.07	-0.31	-0.18
		0.5	0.5	-0.22	-0.68	-0.36	0.26	0.06	0.18	-0.12	-0.50	-0.24
		0.9	0.1	-0.16	-0.43	-0.26	0.26	-0.68	-0.11	-0.16	-0.41	-0.26
		1.0	0.0	0.00	-0.37	-0.22	0.53	-0.56	-0.11	0.00	-0.37	-0.22
	High	0.1	0.9	0.04	-0.70	-0.35	0.02	-0.57	-0.35	0.00	-0.58	-0.35
		0.5	0.5	0.00	-0.48	-0.19	0.31	-0.20	0.03	0.00	-0.21	-0.11
		0.9	0.1	0.00	-0.17	-0.06	0.35	0.00	0.12	0.00	-0.14	-0.05
		1.0	0.0	-0.16	-0.65	-0.42	0.04	-0.60	-0.28	-0.16	-0.65	-0.42