



# Modified differential evolution and heuristic algorithms for dump tippler machine allocation in a typical sugar mill in Thailand

Chuleeporn Kusoncum<sup>1</sup> · Kanchana Sethanan<sup>2</sup> · Richard F. Hartl<sup>1</sup> · Thitipong Jamrus<sup>1</sup>

Received: 8 November 2018 / Revised: 21 June 2020 / Accepted: 20 August 2020 /  
Published online: 10 September 2020  
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

## Abstract

This paper focuses on a computational tool for scheduling and sequencing sugarcane vehicles for dump tippler machines at the mill yard of a sugar mill, which can be represented as an NP-hard problem for parallel capacitated machines with machine restrictions, job grouping, and sequencing independent setup time. This research aims to determine the optimal sequencing of jobs, i.e. minimizing the makespan by considering machine restrictions, capacitated machines, group size, number of jobs, and the constraints of the sugar mill. In the considered problem machines alternatively operate, that distinguishes it from a general parallel machine problem. The mixed integer linear programming model is developed for solving the small-scale problem instances. Large-scale instances are handled by four heuristics, and four differential evolution (DE) metaheuristics. In order to improve the computational results, solution quality and computation time were considered. In addition, modified DE algorithms were used in encoding operation (initial solution), mutation and local search operation. The computational results revealed that the modified DE algorithms had higher relative improvement on the makespan. Furthermore, this decision-making support tool was implemented as a prototype in the sector of cane and sugar industry in Thailand and extended to other similar industries.

**Keywords** Mill yard management system · Dump tippler machine · Heuristic algorithm · Differential evolution · Scheduling · Sequencing

---

✉ Kanchana Sethanan  
skanch@kku.ac.th

<sup>1</sup> Research Unit on System Modeling for Industry, Industrial Engineering Department, Faculty of Engineering, Khon Kaen University, Khon Kaen 40002, Thailand

<sup>2</sup> Department of Business Administration, Faculty of Business, Economics and Statistics, University of Vienna, Morgenstern-Platz 1, 1090 Vienna, Austria

## 1 Introduction

Thailand is an agro-industrial country and the cane and sugar industry is extremely important to the country's economy. Thailand was ranked as the world's second-largest exporter of sugar after Brazil (USDA 2017). In 2016/2017 crop production, Thailand had 10.98 million rai of sugarcane cultivation area (1 Rai = 0.16 ha) producing approximately 92.98 million tonnes of sugarcane, with the average sugarcane yield being 9.43 tonnes/rai, and 10.029 million tonnes of sugar in total. This industry has a large labor market involving stakeholders such as growers, sugar mills, and third party logistics. There are 55 large sugar mills with crushing capacities of 12,000–55,000 tonnes/day located in different regions of Thailand. Hence this industry plays an important role in distributing income to the local areas. Moreover, improved efficiency and productivity are required in most industries due to higher competition in the business world of today (Office of Global Analysis 2015). Similarly improvement of mill yard operation in a sugar mill can lead to improved efficiency and productivity, and thus competitiveness of the sugar mill (Office of the Cane and Sugar Board 2017).

As sugarcane is a preeminent crop for Thailand's economy, there is continued effort to increase the mill capacities by improving utilization of machines. Currently, the Thai sugar industry has faced with problems arising from underutilization of machines. The sugar industry is thus looking for solutions to improve mill capacity by improving operations within the mill. In a typical sugar mill in Thailand scheduling and sequencing of sugarcane vehicles for dump tippler machines has been found to be a source of inefficiency. It is therefore proposed in this study to optimize scheduling and sequencing of sugarcane vehicles for dump tippler machines to minimize the makespan of the operation. The operation is considered as a very important internal logistic process connected with the inbound logistics process. Presently, there is inefficient scheduling at the dump tippler machines, which is the first operation of the sugarcane preparation process in most sugar mills in Thailand. For this system, the fully loaded sugarcane vehicle is moved on to the dump platform and whole platform is raised to about a 45° angle. At this point the sugarcane rolls off onto the conveyor. The platform and vehicle are then lowered and the process repeated (Baikow 2013). Normally, scheduling of sugarcane vehicles for dump tippler machines can cause a long delay in the system due to the fact that the vehicles have different dimensions, different wheel sizes, different wheel spacings and different capacities. Therefore, the sugarcane vehicles require longer yard time, resulting in reduction of tippler-dump machine capacities.

In Thailand, sugarcane is transported to the mills by, in increasing capacity, trucks, trailers, or tractors, depending on field sizes and the distance between sugarcane fields and the mill. After the sugarcane vehicle arrives at the mill yard, it is weighed prior to processing. At the mill, the sugarcane is mechanically unloaded from the vehicle. The sugarcane is then prepared for crushing in the sugarcane preparation process. This involves cutting and shredding cane to prepare it for juice extraction. It is usually considered that the best extraction results are linked

with the efficiency of the preparatory devices used to slice or shred the cane. The sugarcane preparation process consists of 3 stations: dump tippler system, shredding, and milling (Jenkins 2013). At the sugarcane preparation process, sugarcanes are delivered by the dump tipplers with driven moving floor bodies (or conveyors), and excessive soil and rocks are removed. At this point, the sugarcane is then passed through the sugarcane leveler to ensure consistent depth of sugarcane to feed into a hammer mill shredder which opens about 90% of the juice cells in the cane. This improves the efficiency of removal of the sugar from the fiber in subsequent processing.

Normally, the sugar industry tailors harvesting, delivery and crushing operations to ensure that the time between cutting and crushing does not exceed a specified time duration, i.e., within 24 h. It aims for the lowest possible delay to minimize cane deterioration, sugarcane weight loss, and consequent sugar loss, since these losses will decrease the sugarcane price (Tinnongwatthana 2013). Hence, inbound logistics connected with the sugarcane preparation process which is the first process of the internal logistics of the mill play a crucial role in sustaining the recoverable sugar and also maintaining the maximum milling capacity. It must be very efficient in order to avoid sugar loss and keep the maximum mill capacity by managing in such a way that the sugarcane vehicle leaves the sugar mill as quickly as possible. Presently, the tippler-dump station usually affects the duration of yard time needed by the sugarcane vehicles (resulting in a reduction of tippler-dump machine capacity) due to: (1) the capacities of dump tipplers are different in order to support different vehicle capacities; (2) some dump tipplers are restricted to certain vehicle capacities. This means dump tippler features are restricted depending on vehicle bin dimensions, wheel size, wheel spacing, and capacity; (3) arrival rates of vehicles are different and (4) vehicles running in parallel are operated on a subset of selected dump tipplers, in order to make a consistent depth of sugarcane to feed into the hammer mill shredder to open the juice cells in the sugarcane as much as possible. In this study, a group of vehicles (i.e., two vehicles) are operated only on selected dumps (i.e., two dumps) simultaneously, while the next group of vehicles will be operated on by the remaining dump tipplers. This means two groups of two dump tipplers are alternatively operated.

The challenge for the sugar mill is, however, determining the most suitable sequence for scheduling sugarcane vehicles to dump tipplers in order to successfully reduce the time in the system, while satisfying the sugar mill's maximum crushing capacity. Since the arrival rate of sugarcane vehicles varies, this study attempts to determine the optimal sequence of sugarcane vehicles for dump tipplers. This problem is identified as scheduling  $M$  parallel capacitated machines in a tippler-dump system where the dump tipplers are considered as capacitated machines with restricted assignment to be scheduled to satisfy the conditions of the sugarcane vehicles under consideration. The objective function of the model is to minimize time in the system for each sugarcane vehicle. On the other hand, the number of sugarcane vehicles leaving the tippler-dump system is maximized.

To solve this problem, a 0–1 mixed integer programming model was first developed to solve small-scale problems. To solve realistic-scale problems, an effective differential evolution algorithm (DE) and heuristics based on job pairings were

proposed. An effective initial solution and modified mutation operation of DE were improved by generating two algorithms in the encoding method, and it used a different equation to transform a target vector to a mutant vector in two sub-groups of the population. Then, the modified DE was formulated to enhance the search ability of DE using a SWAP local search. Heuristics based on job pairings proposed in this study were: (1) current practice algorithm; (2) cyclic algorithm; (3) cyclic algorithm based on WSPT rule; (4) matching jobs with the same amount of processing time algorithm. All of the proposed heuristics on job pairing were developed to obtain near optimal conditions for solving the realistic-scale problem using the least computational time. To show its effectiveness, its numerical experimental results will be compared with those of the current practice of a sugar mill selected as a case study and also with the traditional DE. The next section provides a review of relevant literature. The mathematical model is described in Sect. 3 and the methodology of our proposed methods is presented in Sect. 4. Section 5 outlines computational results and comparison with the case study. Finally, a summary of the main findings is given in Sect. 6.

## 2 Literature review

Our review of the literature reveals that several researchers have focused on inbound logistics activities of the cane and sugar industry, cultivation, harvest, transportation, and mill yard management, such as Higgins et al. (1998), Muchow et al. (1998), Astika et al. (2001), Sartori et al. (2001), and Sethanan and Neungmatcha (2014). These researchers have studied and developed the inbound logistics management systems by considering cost, time, quality, and services in an attempt to supply sugarcane to meet the demands of sugar mills efficiently. Optimization and simulation techniques have widely been used to solve sugarcane supply chain problems.

However, very few studies have been conducted for mill yard operation, even though it is a crucial process and usually impacts the inbound logistics performance of the sugar mill. Additionally, the methodology used to solve the problem is generally by simulation. For example, Arjona et al. (2001) applied a simulated model for analysis of a harvesting and transportation system of a sugarcane plantation to increase machine utilization, which involved a reduction in the number of machinery needed without increasing time in the system. Later, Higgins and Davies (2005) also used a simulated model for capacity planning in sugarcane transport, by considering the number of locomotives and shifts, the number of bins, and delays of harvesting operations resulting from harvesters waiting for bin deliveries. In addition, Iannoni and Morabito (2006) applied discrete simulation techniques to study the sugar mill reception area, operation processes, and policies, aiming to increase the amount of sugarcane unloaded. Recently, Bocanegra-Herrera and Vidal-Holguín (2016) have also developed a simulation model as a decision support system for sugarcane supply, which considered inbound logistics, harvesting, transportation, and unloading of sugarcane at the mill yard, since they might contribute to a significant proportion of the total cost. These researchers considered two important factors including environmental conditions (e.g. the rainy period) and the effects of policy change (different

layouts of the unloading areas such as chain net system, slide dump system, vehicle dump system, and tippler dump system), and studied the impact of these factors on the amount of unloaded sugarcane.

Based on the literature review of the present study, there has been no effort to study the dump tippler process, even though the dump tippler process is known to be the bottleneck of the mill yard operation in the sugarcane preparation process of the sugar industry. Normally, scheduling and sequencing of the dump tipplers can incur high inbound logistics cost, since not only it can cause long waiting time of vehicles at the mill yard, but it can also affect unloaded sugarcane quantity and weight and sweetness of the sugarcane.

In many sugar producing countries in Asia, including Thailand, sugarcane vehicles are scheduled and sequenced into the dump tipplers because the principal means of sugarcane transportation in these countries is land transport by vehicles with different sizes. In the leading countries, especially in Brazil and Australia, trains or large-scale vehicles are used to deliver sugarcane to the sugar mill (Martinez et al. 2006; Masoud et al. 2015). The limitation of transportation laws which regulate vehicles' payloads for different vehicle types in Thailand and some local practices have caused a large number of sugarcane vehicles waiting in the mill yard during the crushing period. Therefore, the main thrust of this research is on the dump tippler process of the mill yard, in order to optimize scheduling and sequencing of sugarcane vehicles from the inner yard to the dump tippler machines for minimizing the completion time of sugarcane vehicles.

This problem can be formulated as the parallel capacitated machines with machine restriction and jobs with restricted assignment (PCMGS). Using the three field notation of Pinedo (1995), this problem can be stated as a deterministic PCMGS,  $R_m|p_{i,j}, s_{i,j}, M_{i,j,t}, Grouping|C_{max}$  problem which, in our case, investigated sugarcane vehicle scheduling for dump tippler machines with the objective to minimize the makespan. In practice, the dump-tippler machines are divided into two groups with equal numbers of machines, in order to achieve a consistent depth of sugarcane to feed into a hammer mill shredder to open the juice cells in the sugarcane as much as possible. Then, during each time period, the sugarcane vehicles are assigned to operate on each group of the dump tippler machines. Hence, in this paper, at any time, all machines (i.e., dump tippler machines) cannot be operated independently. They are separated in two groups of two dump tippler machines to alternately operate.

The parallel machine scheduling problems (PMSP) have widely been studied in the past decades. Various features may be included such as types of machines (i.e., identical, uniform, and unrelated machines), machine setup time, capacitated machines, machine restriction, and job grouping. Machine setup time, especially dependent setup time, is an essential factor for production scheduling in all manufacturing systems. In PMSP, the features of sequence-dependent setup time have been studied by various researchers (see Nait Tahar et al. 2006; Yalaoui and Chu 2003; Silva and Magalhaes 2006). For scheduling unrelated parallel machines, Dolgui et al. (2009) studied multi-product lot-sizing and scheduling on unrelated parallel machines to minimize the makespan (Dolgui et al. 2009). Recently, Moonsri et al. (2015) studied scheduling of unrelated parallel machines with sequence-dependent

setup time and machine eligibility in a Hard-Disk Drive (HDD) assembly process for the Head Gimbal Assembly (Moonsri et al. 2015).

However, very little PMSP research has been conducted on different machine capacity. For instance, Yanyan et al. (2011) studied scheduling parallel multiple capacitated machines with sequence-dependent constraint for identical reheating furnace production in the steel industry, which aimed to minimize the total processing time for energy consumption reduction and quality improvement of the product. Recently, Fox and Korupolu (2013) studied weighted flowtime on capacitated machines in a single machine and single-dimension problem.

For studies of machine eligibility restriction in parallel machines, various research has been studied for several applications, such as in semiconductor manufacturing (Centeno and Armacost 1997), block erection of the shipyard (Hu et al. 2010), and automobile gear manufacturing (Gokhale and Mathirajan 2012), etc. In general, a group of jobs processed together is called a batch that means all jobs of the same batch are completed together when the last job in this batch has finished its processing (Webster and Baker 1995). Similar to the work of Potts and Kovalyov (2000), job grouping will be considered in the present study. In another study by Sharma et al. (2010), FCFS- job grouping approach is used to maximize resource utilization for scheduling of jobs in grid computing application. Recently, Yuan et al. (2011) applied job grouping approach in scheduling of containers at automated seaport container terminals to improve the makespan. The experimental results of their study showed that the job grouping approach could effectively improve the makespan and reduce the total straddle carrier waiting time (Yuan et al. 2011).

The PSMS with the objective to minimize the makespan in a classic PMSP, denoted as  $P||C_{max}$  (Garey and Johnson 1978),  $P_m|M_j|C_{max}$  (Leung and Li 2008; Lee et al. 2011),  $R_m|S_{ijk}|C_{max}$  (Rabadi et al. 2006), has been shown to be a strongly NP-hard problem. This research which is a special case of  $P||C_{max}$  and stated as  $R_m|p_{i,j}, s_{i,j}, M_{i,j,t}, Grouping|C_{max}$ , is more complex than the  $P_m|M_j|C_{max}$  problem and remains NP-hard. Therefore, this observation makes a strong case for the application of heuristics and meta-heuristic methods to be suitable for dealing with the problem. Storn and Price (1997) were the researchers who invented the DE technique which has been successful in solving a variety of problems. DE is one of most powerful random search methodologies because of its uncomplicated computational technique (Kachitvichyanukul 2012), and quick convergence (Wu and Che 2018) and has been applied in several research areas and industries. For example, the simple assembly line balancing problem (Pitakaso and Sethanan 2016), the general assignment problem (Sethanan and Pitakaso 2016b), the vehicle routing problem in the poultry industry (Dechampaï et al. 2017), scheduling raw milk transportation in dairy factories (Sethanan and Pitakaso 2016a), scheduling and sequencing problems in manufacturing industry (see Nearchou 2006; Weaver et al. 2012; Chakravarthy and Karatza 2013).

In this research, the adapted heuristics and metaheuristic techniques of traditional DE and modified DE were improved to solve large-scale instances in the scheduling and sequencing of sugarcane vehicles for the dump tippler machines of the mill yard, which is a case of a PCMGS problem with the objective to minimize the makespan ( $R_m|p_{i,j}, s_{i,j}, M_{i,j,t}, Grouping|C_{max}$ ). The proposed algorithms were investigated by

comparing their results to the constructive heuristic technique based on sugar mill current practice. The main advantage of the metaheuristic techniques was obtaining near-optimal solutions with a reasonable computing time. The traditional and modified DE techniques were developed as a new approach to optimize the dump tippler process. In other words, the proposed techniques may be considered as hybrid techniques aiming to improve the quality of the DE solutions, reduce the CPU time for obtaining solutions, and solve large-scale instances.

### 3 Model description

#### 3.1 Problem statement and assumptions

The problem of this investigation is identified as parallel capacitated machines with machine restriction, job grouping, and sequencing independent setup times  $(R_m|p_{i,j}, s_{i,j}, M_{i,j,t}, Grouping|C_{max})$ . The  $m$  dump tipplers are considered as machines with different capacity and machine restrictions. Once the sugarcane vehicles arrive at the mill yard, they are weighed prior to being mechanically unloaded of the sugarcane from the vehicle to the dump tipplers. Each dump tippler  $i$  ( $i=1, 2, \dots, M$ ) has different capacity  $c_i$  in order to support  $N$  sugarcane vehicles with different capacities  $q_j$  ( $j=1, 2, \dots, N$ ). Additionally, some dump tipplers are restricted to some particular vehicle loading capacities. This means a dump tippler's features are machine restrictions depending on vehicle bin dimensions, wheel sizes, wheel spacing, and capacity. Due to area limitation of the mill, dump tipplers are designed in a group technology machine discipline. This means two dump tipplers (i.e., D1 and D2) are installed on one side and other two dump tipplers (D3 and D4) are installed on the opposite side, instead of installing the four dump tipplers in parallel on one side (see Figs. 1, 2). In order to ensure a consistent depth of sugarcane to feed into a hammer mill shredder, the dump tipplers which are installed on a diagonal must be operated simultaneously. For example, when dump tipplers D1 and D4 simultaneously start and finish their operations, dump tipplers D3 and D2 will start their operations simultaneously

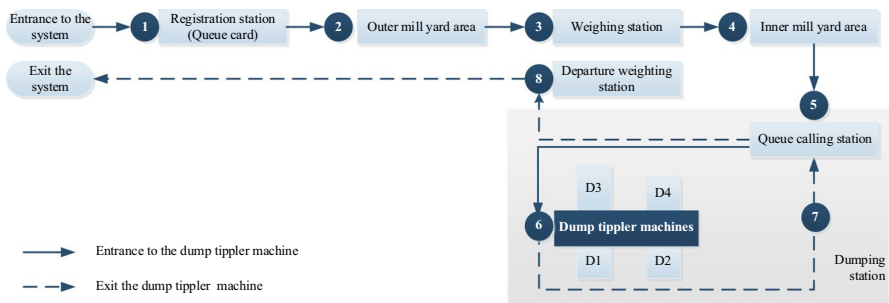
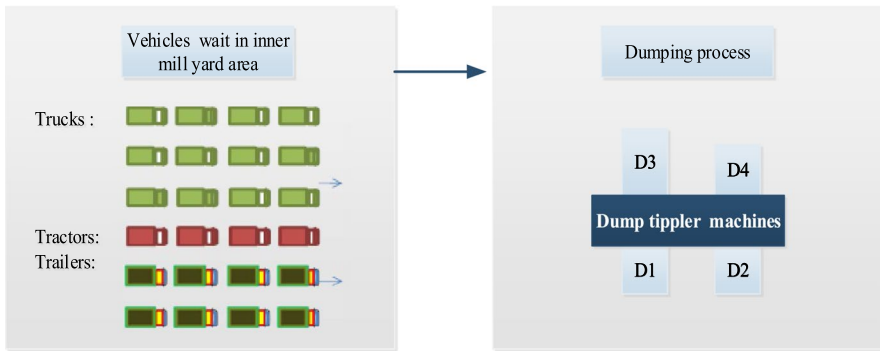


Fig. 1 The layout of the mill yard at the sugar mill



**Fig. 2** The characteristics of the problem at dump tippler machines

right after the latest completion time of the sugarcane vehicle at D1 or D4. This process is repeated until there are no more sugarcane vehicles in the process.

To formulate this model, we assume that:

1. The sugarcane loaded vehicles are considered as the  $N$  jobs with sugarcane quantity  $q_j$  assigned to the  $M$  dump tippers with capacities  $c_i$ .
2. One job is assigned to only one dump tippler and each job cannot be split to other dump tippers.
3. The scheduling and sequencing considers the machine restrictions  $M_{i,j,t}$  and the capacitated machine  $c_i$ ; (1) three dump tippler machines (D1, D2, and D4) have the capacity of 50 tonnes/dump which can take sugarcane from both trucks and trailers, and (2) the other dump tippler (D3) has the capacity of 80 tonnes/dump which can take sugarcane from trucks, trailers, and tractors.
4. The dump tippers are considered as unrelated parallel machines with different processing times of job  $p_{i,j}$  at the dump tippler station. Additionally, the Sequence Dependent Setup Time ( $s_{i,j}$ ) of vehicle  $j$  depends on both type of vehicle and dump tippler.
5. Due to the mill area limitation, dump tippers are designed in a group technology machine discipline. This means two dump tippers (i.e. D1 and D2) are installed on one side and the other two dump tippers (D3 and D4) are installed on the opposite side.
6. The scheduling strategy of vehicles loaded with sugarcane to the dump tippers is on a first come first served (FCFS) principal.

In this section, a 0–1 mixed integer programming model is developed to schedule all sugarcane vehicles (job  $j$ ) to the various dump tippler machines with the objective of minimizing the makespan. Parameters and decision variables used in formulating the model are defined in Sect. 3.2. The model is presented in Sect. 3.3 with a brief explanation of each constraint.



### 3.2 Parameter definitions

#### Indices

$i$	Machine index $i \in \{1, 2, 3, \dots, M\}$
$j$	Job index $j \in \{1, 2, 3, \dots, N\}$
$t$	Time period $t \in \{1, 2, 3, \dots, O\}$

#### Parameters

$N$	Number of jobs
$M$	Number of machines
$O$	Number of time periods $(O = \frac{N}{2})$
$q_j$	Sugarcane quantity of job $j$ (unit: tonnes)
$c_i$	Capacity of machine $i$ (unit: tonnes)
$P_{i,j}$	Processing time on machine $i$ when processing job $j$ (unit: minutes)
$s_{i,j}$	Setup time on machine $i$ when processing job $j$ (unit: minutes)
$M_{i,j,t}$	Machine $i$ performs job $j$ at time period $t$

#### Decision variables

$C_{MAX}$	Makespan of jobs (unit: minutes)
$C_{i,j,t}$	Completion time on machine $i$ when processing job $j$ at time period $t$ (unit: minutes)
$CM_t$	Maximum completion time at time period $t$ (unit: minutes)
$ST_{i,j,t}$	Starting time of job $j$ on machine $i$ at time period $t$ (unit: minutes)
$X_{i,j,t}$	= 1, if job $j$ is assigned to machine $i$ at time period $t$ = 0; otherwise

### 3.3 Model formulation

In this section, a mathematical model is formulated for the makespan minimization. Parameters and decision variables used in formulating the model are defined. The 0–1 mixed integer programming formulation is presented below.

$$\text{Minimize } Z = C_{MAX} \quad (1)$$

$$C_{MAX} \geq C_{i,j,t} \quad \forall i, j, t \quad (2)$$

$$CM_t \geq C_{i,j,t} \quad \forall i, j, t \quad (3)$$

$$\sum_{i=1}^M \sum_{t=1}^O X_{i,j,t} = 1 \quad \forall j \quad (4)$$

$$\sum_{j=1}^N X_{i,j,t} \leq 1 \quad \forall i, t \quad (5)$$

$$M_{i,j,t} - X_{i,j,t} \geq 0 \quad \forall i, j, t \quad (6)$$

$$q_j * X_{i,j,t} \leq c_i \quad \forall i, j \quad (7)$$

$$C_{i,j,t} \geq [(p_{i,j} + s_{i,j}) * X_{i,j,t}] + ST_{i,j,t} \quad \forall i, j, t \quad (8)$$

$$ST_{i,j,t} \geq CM_{t-1} \quad \forall i, j, t > 1 \quad (9)$$

$$C_{i,j,t} \geq 0 \quad \forall i, j, t \quad (10)$$

$$ST_{i,j,t} \geq 0 \quad \forall i, j, t \quad (11)$$

$$X_{i,j,t} \in \{0, 1\} \quad \forall i, j, t \quad (12)$$

The objective function is to minimize the makespan (unit: minutes) by Eq. (1). Constraint (2) requires the makespan to be greater than the completion time period on machine  $i$  when processing job  $j$  at time period  $t$ . Constraint (3) ensures that the maximized completion time at period time  $t$  is equal to or greater than the completion time on machine  $i$  when processing job  $j$  at time period  $t$ . Constraint (4) ensures that every job is assigned to exactly one machine at a time period. Constraint (5) ensures ensure that a machine in a time period  $t$  cannot do more than one job. Constraint (6) is a machine restriction as detailed in the problem statement and assumptions section. Constraint (7) ensures that the sugarcane quantity of job  $j$ , processed on machine  $i$ , must not exceed the capacity of that dump tippler machine. Constraint (8) ensures that the completion time period on machine  $i$  when processing job  $j$  at time period  $t$  of the jobs is equal to or greater than the sum of processing time, setup time and starting time of the job. Constraint (9) ensures that the starting time of machine  $i$  is assigned job  $j$  at time period  $t$  equal to or greater than the maximum completion time at previous time period  $t - 1$ . Constraints (10) to (12) represent the types of variables with the basic restrictions on the decision and binary variables.

The mathematical model was developed on a standard personal computer with Intel® Core™ i7-4500U processor (2.40 GHz clock) and 4.00 GB RAM as the modeller and Lingo/CPLEX version 13.0 software as the solver. However, the mathematical model was suitable for solving small-scale problems but not large-scale problems due to excessive computational time. Therefore, heuristics or metaheuristics were needed and developed to obtain near optimal results for solving the realistic-scale problem.

## 4 Heuristics and metaheuristics development

When the size of the problems becomes too large and too complicated to be solved by Lingo/CPLEX, heuristics and metaheuristics are employed. In this paper, the development of four heuristics and four DE was proposed for sequencing sugarcane vehicles to the dump tippers, since they are highly efficient and effective for solving complicated and large-scale problems, particularly the NP class, in terms of computational time and solution quality. Details of the heuristics and metaheuristics are presented below.

### 4.1 Constructive heuristic development

In this section, four heuristic algorithms are developed to solve the problem with the objective to minimize the makespan. Since two dump tippler machines are operated simultaneously, two vehicles (jobs) are selected to work simultaneously. Since, in this problem, it is very complicated because it involves machine restrictions, capacitated machines, and amounts of cane in each vehicle. Furthermore, the machines are working in parallel and simultaneously. The key success factor in solving this type of problem is finding different ideas for each algorithm in order to meet the objective. Details of each algorithm are presented below.

#### 4.1.1 H2: Current practice

This heuristic was developed based on the current practice of the sugar mill as the case study. This heuristic sequences the vehicles based on a First Come First Served (FCFS) basis. However, the machine restriction is also considered. This means a vehicle will be first checked if it follows/meets the FCFS; if yes, then it is checked that it meets the machine restriction (i.e., machine capacity) that it can dump its cane to any available dump tippler. If yes, the vehicle is completely scheduled. On the other hand, it may have to wait until it meets the dump tippler restrictions.

Since there are various vehicles waiting at the outer mill yard (i.e., 300 vehicles), in this algorithm a group of vehicles (i.e., 10, 20, 30 vehicles) is selected as a group based on the FCFS policy. For example, 12 vehicles (i.e., Job 1–Job 12 randomly selected by the FCFS policy) can be selected to the dump tippler machines under machine sequencing and machine restrictions. Details of the heuristic are presented in Fig. 3.

#### 4.1.2 H3: Cyclic algorithm

Since, in this case study, two dump tippler machines, are operated simultaneously, the key idea is to find two vehicles to operate simultaneously. In this algorithm, cyclic scheduling is applied. Firstly, two vehicles with the lowest processing time are selected. Then, the next vehicles with the next highest processing time are

```

1. procedure: Algorithm H2: Current practice algorithm
2. input: Processing time  $p_{ij}$ , Setup time  $s_{ij}$ , The set of jobs  $\{1, 2, 3, \dots, N\}$ , Machine restriction  $M_{ijt}$ , The set of machines  $\{1, 2, 3, \dots, M\}$ ,
Machine sequencing  $ms$ , Group size number  $g$ 
3. output: Scheduling and sequencing of job and makespan
4. begin
5.   for Machine  $i = 1: M$ 
6.     Input machine sequencing  $ms$ 
7.   end
8.   for group size number  $g = 1: G$ 
9.     Job  $j \leftarrow 0$ ;
10.    While Job  $j \leq N$  do
12.      Assigns job to machine based on FCFS by following machine sequencing  $ms$  and machine restriction  $M_{ijt}$ 
13.    end
14.     $j \leftarrow j + 1$ 
15.  end
16. end;

```

**Fig. 3** The procedure of the current practice algorithm (Algorithm H2)

```

1. procedure: Algorithm H3: Cyclic algorithm
2. input: Processing time  $p_{ij}$ , Setup time  $s_{ij}$ , The set of jobs  $\{1, 2, 3, \dots, N\}$ , Machine restriction  $M_{ijt}$ , The set of machines  $\{1, 2, 3, \dots, M\}$ ,
Machine sequencing  $ms$ , Group size number  $g$ 
3. output: Scheduling and sequencing of job and makespan
4. begin
5.   for Machine  $i = 1: M$ 
6.     Input machine sequencing  $ms$ 
7.   end
8.   for group size number  $g = 1: G$ 
9.     Job  $j \leftarrow 0$ ;
10.    While Job  $j \leq N$  do
11.      Assigns both jobs to machines based on equal total processing and setup time
      (i.e.,  $[(p_{ij} + s_{ij}) - (p_{ij} + s_{ij})] = 0$ ) by following machine restriction  $M_{ijt}$ ,
      one job+1 must be assigned to one machine only.
12.    end
       $j \leftarrow j + 2$ 
13.    While Job  $j \leq N$  do
14.      Assigns both jobs to machines based on the lowest difference of the sum of total processing and setup time
      (i.e.,  $\text{Min} [(p_{ij} + s_{ij}) - (p_{ij} + s_{ij})]$ ) by following machine restriction  $M_{ijt}$ 
15.      Assigns both jobs to machines based on the highest difference of the sum of total processing and setup time
      (i.e.,  $\text{Max} [(p_{ij} + s_{ij}) - (p_{ij} + s_{ij})]$ ) by following machine restriction  $M_{ijt}$ 
16.    end
       $j \leftarrow j + 2$ 
17.  end
18. end
19. end;

```

**Fig. 4** The procedure of the cyclic algorithm (Algorithm H3)

selected. The process is repeated until all vehicles are scheduled. Details of the heuristic are presented in Fig. 4.

#### 4.1.3 H4: Cyclic algorithm based on WSPT rule

This algorithm applies the weighted shortest processing time (WSPT) rule to the scheduled vehicles. This means this rule will select the vehicles with the lowest ratio of the total processing time required and the amount of cane contained in that vehicle. Details of the heuristic are presented in Fig. 5.

#### 4.1.4 H5: Matching jobs with the equal of processing time algorithm

Since the processing time to dump sugarcane to the dump tippler machine for each vehicle depends on the amount of sugarcane, this algorithm attempts to reduce the waiting time of one of the vehicles by selecting two vehicles with the same amount of sugarcane. Hence, the two vehicles with the lowest difference will be selected. In case there are more than one pair of jobs with the lowest difference in amount of sugarcane, the priority is given to the pair of vehicles with maximum sugarcane amount. Details of the heuristic are presented in Fig. 6.

The Gantt chart of the proposed heuristics is shown in Fig. 7.

```

1. procedure: Algorithm H4: Cyclic algorithm based on WSPT rule
2. input: Processing time  $p_{ij}$ , Setup time  $s_{ij}$ , The set of jobs  $\{1, 2, 3, \dots, N\}$ , Machine restriction  $M_{ijt}$ , The set of machines  $\{1, 2, 3, \dots, M\}$ , Machine sequencing  $ms$ , Group size number  $g$ 
3. output: Scheduling and sequencing of job and makespan
4. begin
5.   for Machine  $i = 1; M$ 
6.     Input machine sequencing  $ms$ 
7.   end
8.   for group size number  $g = 1; G$ 
9.     Job  $j \leftarrow 0$ ;
10.    While Job  $j \leq N$  do
11.      Assigns both jobs to machines based on equal total processing and setup time
        (i.e.,  $[(p_{ij} + s_{ij}) - (p_{ij} + s_{ij})] = 0$ ) by following machine restriction  $M_{ijt}$ 
12.    end
13.     $j \leftarrow j + 2$ 
14.    While Job  $j \leq N$  do
15.      Assigns both jobs to machines based on the weighted shortest processing time (WSPT) rule
        (i.e.,  $\text{Max} [(p_{ij} + s_{ij}) - (p_{ij} + s_{ij})]$ ) by following machine restriction  $M_{ijt}$ 
16.    end
17.     $j \leftarrow j + 2$ 
18.  end
19. end;

```

Fig. 5 The procedure of the cyclic algorithm (Algorithm H4)

```

1. procedure: Algorithm H5: Matching jobs with the same amount processing time algorithm
2. input: Processing time  $p_{ij}$ , Setup time  $s_{ij}$ , The set of jobs  $\{1, 2, 3, \dots, N\}$ , Machine restriction  $M_{ijt}$ , The set of machines  $\{1, 2, 3, \dots, M\}$ ,
Machine sequencing  $ms$ , Group size number  $g$ 
3. output: Scheduling and sequencing of job and makespan
4. begin
5.   for Machine  $i = 1: M$ 
6.     Input machine sequencing  $ms$ 
7.   end
8.   for group size number  $g = 1: G$ 
9.     Job  $j \leftarrow 0$ ;
10.    While Job  $j \leq N$  do
11.      Assigns both jobs to machines based on minimum difference of the sum of total processing and setup time
      (i.e.,  $\text{Min} [(p_{ij} + s_{ij}) - (p_{ij} + s_{ij})]$ ) by following machine restriction  $M_{ijt}$ 
12.    end
13.     $j \leftarrow j + 2$ 
14.  end
15. end;

```

**Fig. 6** The procedure of matching jobs with the same amount of processing time algorithm (Algorithm H5)

## 4.2 Metaheuristics development

The DE has been successfully applied to solve large-scale problems in several research areas. We proposed (1) the traditional differential evolution and (2) the modified differential evolution for solving the sequence of sugarcane vehicles for dump tippler machines scheduling. The DE operations; the heuristic based encoding and decoding, generation of initial solution, mutation, recombination, selection, and local search operations are presented in Table 1.

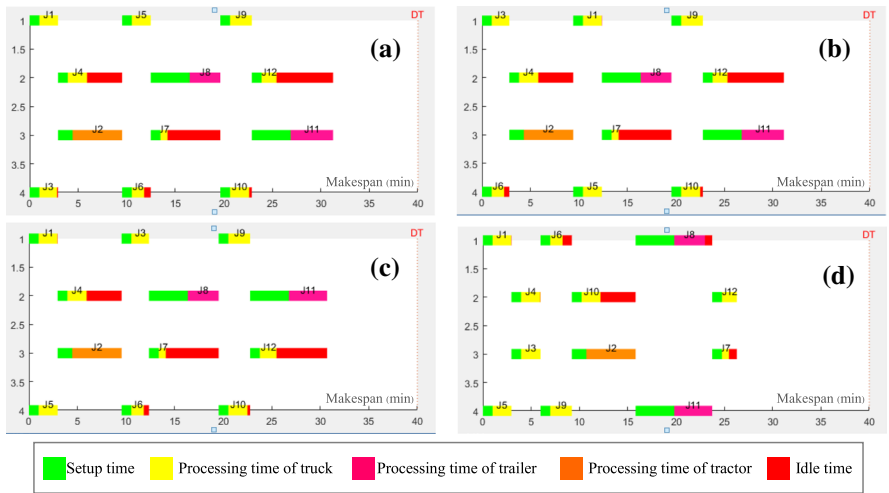
### 4.2.1 Traditional differential evolution (Traditional DE)

The details of the traditional DE are presented in DE6 and DE7 algorithms as shown in Fig. 8.

**4.2.1.1 The heuristics-based encoding and decoding method** This method considers a problem to assign 12 sugarcane vehicles to 4 dump tippler machines and set two groups of sugarcane vehicles with 6 vehicles each based on the FCFS basis. Information on the operation time and set up time of each vehicle is given in Tables 4 and 5.

Generally, each dump tippler machine has a limited capacity. In this example, the maximum capacities for dump tippler machines D1, D2, D3, and D4 are 50, 50, 80, and 50 tonnes, respectively. To optimize the sequence of the sugarcane vehicles assigned to the dump tippler machines for which the dump tippler machines sequencing is  $\{D1-D4, D3-D2\}$ , a vector encoding of DE to represent the problem is designed with  $1 \times N$  dimensions, where  $N$  is the number of

Example	Job 1	Job 2	Job 3	Job 4	Job 5	Job 6	Job 7	Job 8	Job 9	Job 10	Job 11	Job 12
	M 1	M 4	M 3	M 2	M 1	M 4	M 3	M 2	M 1	M 4	M 3	M 2
Algorithm H2	Job 1	Job 3	Job 2	Job 4	Job 5	Job 6	Job 7	Job 8	Job 9	Job10	Job11	Job12
	M 1	M 4	M 3	M 2	M 1	M 4	M 3	M 2	M 1	M 4	M 3	M 2
Algorithm H3	Group 1						Group 2					
	Job 3	Job 6	Job 2	Job 4	Job 1	Job 5	Job 7	Job 8	Job 9	Job 10	Job 11	Job 12
	M 1	M 4	M 3	M 2	M 1	M 4	M 3	M 2	M 1	M 4	M 3	M 2
Algorithm H4	Group 1						Group 2					
	Job 1	Job 5	Job 2	Job 4	Job 3	Job 6	Job 7	Job 8	Job 9	Job 10	Job 12	Job 11
	M 1	M 4	M 3	M 2	M 1	M 4	M 3	M 2	M 1	M 4	M 3	M 2
Algorithm H5	Group 1						Group 2					
	Job 1	Job 5	Job 3	Job 4	Job 6	Job 9	Job 2	Job 10	Job 8	Job 11	Job 7	Job 12
	M 1	M 4	M 3	M 2	M 1	M 4	M 3	M 2	M 1	M 4	M 3	M 2



**Fig. 7** Gantt chart of the proposed heuristics results **a** Algorithm H2 (makespan = 31.2532 min), **b** Algorithm H3 (makespan = 31.1442 min), **c** Algorithm H4 (makespan = 30.7078 min), and **d** Algorithm H5 (makespan = 26.2877 min)

**Table 1** Lists of characteristics of the DE algorithms

Attribute no.	Process	DE-6	DE-7	DE-8	DE-9
1	Initial solution	√ (Randomly)	√ (algorithm H5)	√ (Randomly)	√ (algorithm H5)
2	Mutation	√	√	√ (2 sub-group)	√ (2 sub-group)
3	Recombination	√	√	√	√
4	Local search			√	√
5	Fitness Function	√	√	√	√
6	Selection	√	√	√	√

```

1. Procedure: Traditional differential evolution algorithms
2. Input: PCMG problem data, DE parameters (CR, F, NP, I)
3. Output: Optimal solution
4. Begin:
5.   Encoding step: randomly generate a set of target vectors  $n$  ( $n=1\dots NP$ );
   //  $n$  is the index of the vector representation;  $NP$  is a predefined number of population
6.   while termination condition is not satisfied do
7.     for  $n = 1$  to  $NP$  //  $NP$  is the predefined number of population
8.       Solution Initialization (1) Random (for applying DE6) and (2) Heuristic initial solution (for applying DE7)
9.       Decoding and Evaluation
10.      Mutation Operation (1 mutation equation)
11.      Recombination Operation
12.      Selection Operation
13.    end
14.  end
15. end;

```

Fig. 8 The procedure of traditional DE (DE6 and DE7 algorithm)

Vector	Group 1						Group 2					
	Job 1	Job 2	Job 3	Job 4	Job 5	Job 6	Job 7	Job 8	Job 9	Job10	Job11	Job12
	0.344	0.017	0.783	0.321	0.817	0.427	0.936	0.131	0.904	0.109	0.385	0.787
Vector(ROV)	Group 1						Group 2					
	Job 2	Job 4	Job 1	Job 6	Job 3	Job 5	Job10	Job 8	Job11	Job12	Job 9	Job 7
	0.017	0.321	0.344	0.427	0.783	0.817	0.109	0.131	0.385	0.787	0.904	0.936

Fig. 9 An example of decoding performed by sorting the rank order value (ROV Ascending for each group)

sugarcane vehicles as shown in Fig. 9. The numbers of the rows are group numbers with sugarcane vehicle labels, and are randomly generated in the first iteration of the proposed algorithm. Encoding for the initial solution phase consists of two methods; (1) Randomly and (2) by Algorithm H5, that can be generated by using the following procedure as detailed in Sect. 4.1.4. Using the algorithm H5 to generate the initial solution, the random number of vectors is randomly generated in ascending order of each group as shown in Fig. 9. In the next iteration this number will be changed by mutation and recombination operations.

To decode the vector shown in Fig. 9, the solutions of the sequence of the sugarcane vehicles to dump tippler machine scheduling can be obtained by using the rank order value (ROV) in each group, which is the procedure used to sort the position values of a vector in ascending order in each group. From the ranking of the values in the position value, the order of the sugarcane vehicle is obtained. The result of ROV is shown in Fig. 9, which after applying ROV, the order of this vector is {Group 1; 2, 4, 1, 6, 3, 5 and Group 2; 10, 8, 11, 12, 9, 7}. The sequence each group will be used to assign the sugarcane vehicles into the dump tippler machines according to the steps which are described below.



(1) Select the order of the sugarcane vehicles that are being assigned to a set of dump tippler machines and name it as “order”. (2) Assign all sugarcane vehicles to exactly one dump tippler machine. The sugarcane vehicle will be assigned to a machine with the machine sequencing. If the sugarcane vehicle type does not match the dump tippler machine to process, then skip to consider the next sugarcane vehicle. If the vehicle matches with the dump tippler machine, assign the vehicle to the machine. (3) Continue assigning sugarcane vehicles until all vehicles are assigned to exactly one dump tippler machine.

The next step is decoding which is the allocation of sugarcane vehicles for the dump tipplers by following the sequence of dump tippler machine and machine restrictions. Figure 10 shows the results as  $M_1=4-3-11$ ,  $M_2=6-8-7$ ,  $M_3=2-10-9$  and  $M_4=1-5-12$ .

**4.2.1.2 Generation of initial vector operation** The target vector  $X_{ji,G}$  is generated to be equal to population size  $NP$  by using random generation (for applying DE6 and DE8) and algorithm H5 (for applying DE7 and DE9) as shown in Figs. 7 and 9. Each target vector will be transformed into a mutant vector and trial vector using mutation and recombination processes.

**4.2.1.3 Mutation operation** Equation (13) is used in the mutation operation for expanding the search space. The target vector  $X_{ji,G}$  will be transformed to a mutant vector  $V_{ji,G+1}$  by randomly selecting three target vectors  $X_{r1,G}$ ,  $X_{r2,G}$ ,  $X_{r3,G}$  and operating on these selected vectors according to the formula in Eq. (13). The scaling factor  $F$  is a constant from  $[0, 2]$ . In this paper, it is set to 2.0 by experiment.

$$V_{ji,G} = X_{r1,G} + F(X_{r2,G} - X_{r3,G}) \tag{13}$$

**4.2.1.4 Recombination operation** A trial vector  $U_{ji,G+1}$  is obtained from the recombination operation using Eq. (14), which selects the position value of the trial vector  $U_{ji,G+1}$  from the position value of target vector  $X_{ji,G}$  or the position value of mutant vector  $V_{ji,G+1}$  by comparing the vector  $\text{randb}(j)$  for that position with the  $CR$  value. If the identified position value random vector is less than or equal to the  $CR$  value, the position value of trial vector  $U_{ji,G+1}$  will select the position value of mutant vector  $V_{ji,G}$ . Conversely, if the identified position value of the random vector is more than the  $CR$  value, the position value of trial vector  $U_{ji,G+1}$  will select the position value of the target vector  $X_{ji,G}$  as in formula Eq. (14).

	Group 1					Group 2						
Sequencing of job	Job 4	Job 1	Job 2	Job 6	Job 3	Job 5	Job10	Job 8	Job11	Job12	Job 9	Job 7
Sequencing of machine	M 1	M 4	M 3	M 2	M 1	M 4	M 3	M 2	M 1	M 4	M 3	M 2

**Fig. 10** The results for allocating sugarcane vehicles into the dump tippler by following machine sequencing and machine restrictions

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} & \text{if } (\text{randb}(j) \leq CR) \\ X_{ji,G} & \text{if } (\text{randb}(j) > CR) \end{cases} \quad (14)$$

where  $\text{rand}_{ji}U[0, 1]$ ,  $\text{Irاند}$  is a random integer from  $[1, 2, \dots, D]$  and where  $D$  is the dimension of the vector (number of sugarcane vehicles).  $\text{Irاند}$  ensures that  $V_{ji,G+1} \neq X_{ji,G}$ .

**4.2.1.5 Selection operation** The fitness function of target vector  $X_{ji,G}$  will be compared with the fitness function of trial vector  $U_{ji,G}$  for selecting the lowest function value in this problem to be a target vector of next generation  $X_{ji,G+1}$  using Eq. (15).

$$X_{ji,G+1} = \begin{cases} U_{ji,G} & \text{if } (f(U_{ji,G}) < f(X_{ji,G})) \\ X_{ji,G} & \text{otherwise} \end{cases} \quad (15)$$

## 4.2.2 Modified differential evolution (Modified DE)

In order to improve the sequence of sugarcane vehicle to the dump tippler machine scheduling by using a modified DE (DE8 and DE9), the initial solution, mutation, recombination and selection operations are applied, which is similar to the operations for traditional DE (DE6 and DE7). However, the mutation operation for expanding the search space uses two equations for transforming the mutant vector and a local search will be added.

The details of job scheduling of the modified DE are presented in Algorithm DE 8 and DE 9 as shown in Fig. 11.

```

1. Procedure: Modified differential evolution algorithms
2. Input: PCMG problem data, DE parameters (CR, F, NP, I)
3. Output: Optimal solution
4. Begin:
5.   Encoding step: randomly generate a set of target vectors  $n (n=1 \dots NP)$ ;
   //  $n$  is the index of the vector representation;  $NP$  is a predefined number of population
6.   while termination condition is not satisfied do
7.     for  $n = 1$  to  $NP // NP$  is the predefined number of population
8.       Solution Initialization (1) Random (for applying DE8) and (2) Heuristic initial solution (for applying DE9)
9.       Decoding and Evaluation
10.      Mutation Operation (2 mutation equations; 2 sub-group)
11.      Recombination Operation
12.      Selection Operation
13.     end
14.   end
15. end;

```

**Fig. 11** The procedure of the modified DE (Algorithm DE8 and DE9)

**4.2.2.1 The heuristics based encoding and decoding method** The heuristics based encoding and decoding methods for the modified DE are similar to the operations for traditional DE, with two methods, (1) Randomly and (2) using Algorithm H5 for encoding as in traditional DE.

**4.2.2.2 Generation of initial vector operation** The target vector  $X_{ji,G}$  is generated to be equal to population size  $NP$  by using random generation (for applying with DE8) and algorithm H5 (for applying with DE9) as shown in Figs. 7 and 9. The population  $NP$  will be divided into two sub-groups. For example,  $NP = 300$  will be divided into two sub-groups and each sub-group of the population will be transformed to be a mutant vector  $V_{ji,G+1}$  with different equations as in the next step (Li and Zhang 2011).

**4.2.2.3 Mutation operation** The mutant vector is determined by applied mutant operations. Each sub-group of the target vector  $X_{ji,G}$  will be randomly selected  $X_{r1,G}$ ,  $X_{r2,G}$ ,  $X_{r3,G}$  to transform these vectors to be a mutant vector  $V_{ji,G+1}$  by using the formulas as in Eqs. (16) and (17), respectively. For example, from Eq. (16), if the fitness function value of random target vector  $f(X_{r1})$  is less than the fitness function value of random target vector  $f(X_{r2})$ , the mutant vector will be determined from  $X_{r1} + F(X_{r2} - X_{r3})$ . Conversely, the mutant vector will be determined from  $X_{r2} + F(X_{r1} - X_{r3})$ . Equation (17) similarly determines the mutant vector by considering  $f(X_{r2})$  and  $f(X_{r3})$ . The proposed algorithm emphasises expanding the search space for searching for the mutant vector which affects the next iteration solution of differential evolution (Chiang et al. 2010)

$$V_{ji,G+1} = \begin{cases} X_{r1} + F(X_{r2} - X_{r3}); & f(X_{r1}) < f(X_{r2}) \\ X_{r2} + F(X_{r1} - X_{r3}); & \text{otherwise} \end{cases} \quad (16)$$

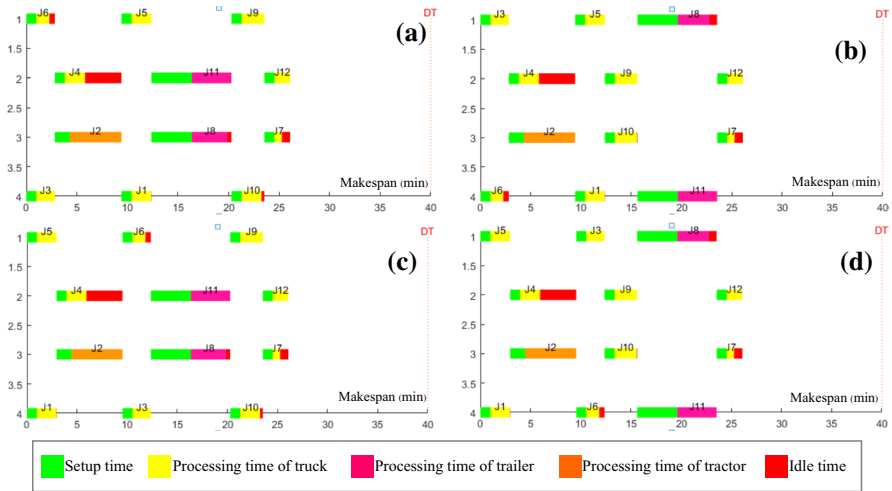
$$V_{ji,G+1} = \begin{cases} X_{r1} + F(X_{r2} - X_{r3}); & f(X_{r1}) < f(X_{r3}) \\ X_{r3} + F(X_{r2} - X_{r1}); & \text{otherwise} \end{cases} \quad (17)$$

**4.2.2.4 Recombination operation** Equation (14) is used to determine the trial vector  $U_{ji,G+1}$  which is similar to the operation for traditional DE.

**4.2.2.5 Selection operation** The selection operation is similar to the operation for traditional DE.

**4.2.2.6 The local search** The SWAP algorithm is the heuristic developed to attempt to exchange the agents of two tasks. Suppose a sugarcane vehicle  $i$  is currently assigned to a dump tippler  $j$  and another sugarcane vehicle  $i'$  is assigned to another dump tippler  $j'$ . The sugarcane vehicle  $i$  will be re-assigned. The re-assignment will be executed only when the SWAP algorithm generates a better makespan. However, the SWAP will be accepted only when the makespan generated by the new move is less than that of the current sequence.

The Gantt chart of the proposed heuristics is shown in Fig. 12.



**Fig. 12** The Gantt chart of results for allocating sugarcane vehicle into the dump tippler machines **a** traditional DE6 (makespan = 26.0858 min), **b** traditional DE7 (makespan = 26.0858 min), **c** modified DE8 (makespan = 26.0858 min), and **d** modified DE9 (makespan = 26.0858 min)

## 5 Computational experiments

### 5.1 Parameter setting and test problems

The research was motivated by the interest of sugar mills in the central region of Thailand which consists of 18 sugar mills out of the total of 55 sugar mills in Thailand. In this case study, the sugar mill capacity was 12,000 tonnes per day. From the data observed at the sugar mill based on 2016/2017 crop year, the maximum number of sugarcane vehicles in the sugar mill system was 581–927 vehicles per day, with the average of 648 vehicles per day. There are three types of vehicles which are (1) truck (i.e. six-wheel truck or ten-wheel truck) (2) trailer, and (3) tractor (a truck modified from a ten-wheel truck to increase the pay load, generally referred to as a tractor in Thailand’s sugar industry). Once, the vehicles arrive at the mill, they will have to wait at the outer yard area which has a maximum capacity of 600 vehicles. Then the vehicles are weighed at the sugarcane weighing station and wait at the inner yard area which has a maximum parking capacity of 300 vehicles, before moving to a dump tippler to dump the sugarcane into the conveyor. In this case study, due to the limited space, there are four dump tippler machines (i.e., D1–D4) which are normally the bottleneck in the system, leading to long waiting time for the sugarcane vehicles. In order to make a consistent depth of sugarcane to feed into a hammer mill shredder, the dump tippler machines cannot operate independently. This means they are assigned alternately to operate on assigned vehicles. Therefore, the scheduling and sequencing of the sugarcane vehicles into the dump tipplers is required to solve this problem to minimize the makespan. The real problem size is a very large-scale problem with an average of 648 sugarcane vehicles per day depending on the number of

sugarcane grower contracts, sugar mill capacity, and machine break-down of the sugar mill.

In order to test the model, the effectiveness and performance of heuristics and metaheuristics in the scheduling and sequencing of sugarcane vehicles into the dump tippler machines were also validated by comparing the optimal solution from a mathematical model obtained by the Lingo/CPLEX version 13.0 software on windows, the current practice of the sugar mill case study, proposed heuristics, and DE obtained by the MATLAB version R2017a9.2.0.538062. The current practice was performed depending on the skill and experiences of the dump tippler process planner and available data of the mill yard management. The effectiveness and performance of the proposed methods were illustrated by 5 different problem sizes, with each problem size having 5 different problems with 10 runs and also the actual data of a sugar mill was used in this experiment as given in Tables 2, 3, 4, 5 and 6.

In this research, the parameters used in the DE algorithm were obtained from experiment. The parameters and the value settings are as follows: population size,  $NP = 300$ , Scaling factor or mutant factor,  $F = 3$ ,  $CR = 0.8$  and stops at 120 s and 300 s for small and large-scale problems respectively, as shown in Table 7. The function evaluation of DE was set to be equal to 10 runs with population size of 300, so that sufficient function evaluation was allowed to find good solutions. Moreover, the algorithm was run with MATLAB version R2017a9.2.0.538062 on Intel® Core™ i7 processor (2.4-GHz clock).

### 5.2 Performance measurement

In this section, the performance of the proposed heuristics and the DE for solving the PCMGS,  $R_m|p_{i,j}, s_{i,j}, M_{i,j,t}, Grouping|C_{max}$  sequencing problem will be described. Two quantities are investigated; (1) percentage heuristic performance ( $HP$ ) (see Eq. 18) and (2) the percentage relative improvement ( $RI$ ) (see Eq. 19)

$$HP = \frac{Sol_{opt}}{Sol_{alg}} \times 100 \tag{18}$$

where  $HP$  is the proposed heuristic performance (%),  $Sol_{opt}$  is the optimal solution obtained from the mathematical model, and  $Sol_{alg}$  is the solution obtained from the proposed heuristics and DE

$$RI = \frac{Sol_{cur} - Sol_{alg}}{Sol_{cur}} \times 100 \tag{19}$$

**Table 2** Input parameters; dump tippler machine  $i$  (i.e., Machine  $i$ )

Dump tippler machine $i$	$D1$	$D2$	$D3$	$D4$
Capacity of dumping machine ( $c_i$ )	50	50	80	50

D1 and D4/D3 and D2 dump at the same time

**Table 3** Input parameters; vehicle  $j$  (i.e., Job  $j$ )

Vehicle $j$	$j1$	$j2$	$j3$	$j4$	$j5$	$j6$	$j7$	$j8$	$j9$	$j10$	$j11$	$j12$
Sugarcane quantity of vehicle ( $q_j$ )	19.26	45.88	18.17	19.89	19.55	12.77	6.55	31.67	22.43	19.66	39.28	15.45
Vehicle type	1	3	1	1	1	1	1	2	1	1	2	1

Vehicle types: (1) Truck can dump at D1–D4, the average of sugarcane quantity is less than or equal to 25 tonnes/truck. (2) Trailer can dump at D1–D4, the average sugarcane quantity is equal to 25–45 tonnes/trailer. (3) Tractor can dump at only D3, the average sugarcane quantity is more than 45 tonnes/tractor

**Table 4** Input parameters; processing time of vehicle ( $p_{i,j}$ )

Processing time ( $p_{i,j}$ )	$j1$	$j2$	$j3$	$j4$	$j5$	$j6$	$j7$	$j8$	$j9$	$j10$	$j11$	$j12$
<i>D1</i>	1.93	–	1.82	1.99	1.96	1.28	0.66	3.17	2.24	1.97	3.93	1.55
<i>D2</i>	1.93	–	1.82	1.99	1.96	1.28	0.66	3.17	2.24	1.97	3.93	1.55
<i>D3</i>	2.14	5.10	2.02	2.21	2.17	1.42	0.73	3.52	2.49	2.18	4.36	1.72
<i>D4</i>	1.93	–	1.82	1.99	1.96	1.28	0.66	3.17	2.24	1.97	3.93	1.55

Dumping rate: (1) *D1*, *D2*, and *D4* equals 10 tonnes/min (Truck/Trailer). (2) *D3* equals 9 tonnes/min (Tractor)

**Table 5** Input parameters; set up time of vehicle ( $s_{i,j}$ )

Set up time ( $s_{i,j}$ )	$j1$	$j2$	$j3$	$j4$	$j5$	$j6$	$j7$	$j8$	$j9$	$j10$	$j11$	$j12$	
<i>D1</i>		1	–	1	1	1	1	1	4	1	1	4	1
<i>D2</i>		1	–	1	1	1	1	1	4	1	1	4	1
<i>D3</i>		1	1.5	1	1	1	1	1	4	1	1	4	1
<i>D4</i>		1	–	1	1	1	1	1	4	1	1	4	1

Set up time of truck is equal to 1 min/truck

Set up time of trailer (Head and tail) is equal to 4 min/truck

Set up time of tractor (modified) is equal to 1.5 min/truck

where *RI* is the relative improvement (%) between the current practice and the proposed algorithms,  $Sol_{alg}$  is the solution obtained from the proposed heuristics and DE, and  $Sol_{cur}$  is the solution obtained from the current practice.

### 5.3 Computational results

For each problem, the computational results are represented in terms of solution quality and computational time as tabulated in Tables 8 and 9. The solutions were obtained from the mathematical model, the heuristics, and 10 runs of the DE for each problem. Parameter test was performed to find suitable parameter values for DE as shown in Table 7. From Tables 8 and 9, it is evident that the mathematical model is suitable for solving small-scale problems. For large-scale problems, the mathematical model needs excessive computational time. Hence, the heuristics and DE were developed to determine a near optimal solution for large-scale problems, and the solutions of these methods are shown in Table 9. From this table, the heuristic performance (*HP*) for H3, H5, H4, and H2 algorithms are respectively, in better *HP* order, 96.55%, 95.65%, 94.78%, and 89.60%. For the case of DE, the *HP* for DE9, DE8, DE7, and DE6 are respectively and in better *HP* order, 99.70%, 99.68%, 99.56%, and 98.51%.

The performance of the proposed heuristics and the DE in terms of the percentage relative improvement (*RI*) is shown in Table 10 in which all the proposed heuristics and the DE (H3–H5 and DE6–DE9) were compared with on the current practice

**Table 6** Design experiment of test problems

Problem test	Number of machines ( $M$ )	Number of jobs ( $N$ )	Number of periods ( $t$ )	Group size number ( $G$ )	Type of job	Number of constraints	Number of decision variables
1	4	4	2	4	3	189	125
2, 3	4	8	4	4, 8	3	761, 761	441, 441
4, 5, 6	4	12	6	4, 6, 12	3	1717, 1717, 1717	949, 949, 949
7, 8, 9, 10	4	24	12	4, 6, 12, 24	3	6889, n/a, n/a, n/a	3625, n/a, n/a, n/a
11, 12	6, 8	24	12	24	3	n/a	n/a
13, 14, 15, 16, 17, 18	4	240	120	12, 24, 48, 60, 120, 240	3	n/a	n/a
19, 20	6, 8	240	120	240	3	n/a	n/a



**Table 7** The setting of DE control parameters

Parameter setting	Parameter value
Number of population, $NP$	300
Stop time	120 s and 300 s (for small and large-scale problems respectively)
Scaling factor, $F$	2
Crossover rate, $CR$	0.8

(H2). It can be seen that the relative improvement (RI) value of H3, H4, H5, DE6, DE7, DE8 and DE9 algorithms, are, respectively and in increasing order, 8.41%, 9.44%, 11.33%, 12.85%, 13.98%, 13.03%, and 14.03%. These improvements may not appear large but they are considerable when converted to monetary values. The highest efficiency of RI is given by the H5 algorithm. This algorithm attempts to reduce the waiting time by matching jobs with the equal processing time. Therefore, any two vehicles with the lowest difference in processing time will be selected. This technique reduced the waiting time between the two vehicles, hence helped reduce the makespan. Although the metaheuristic of DE used longer computational time than the proposed constructive heuristics, as shown in Table 11, the DE metaheuristic solution quality was better than the proposed constructive heuristics. Likewise, when the metaheuristics of DE was compared with traditional DE (DE6), It is seen that the RI increased by 1.33%, 0.21%, and 1.38% for DE7, DE8, and DE9, respectively. Moreover, convergence behavior of the traditional DE and the modified DE methods in finding the solutions for the test problem 11 is presented in Fig. 13. The results illustrate not only that the quality of solutions of the proposed modified DE was better than traditional DE, but also better solutions and faster convergence. This means the modified DE converged to a better solution faster than the traditional DE.

Based on the aforementioned discussion, it can be inferred that all the proposed methods can be applied to minimize the makespan in the scheduling and sequencing of the sugarcane vehicles for the dump tippers in the sugar mill. This not only helps the growers by reducing the waiting time of sugarcane vehicles in the system but also helps the sugar mill to supply raw material in a smooth flow, leading to high efficiency of production. The reduction of the makespan can increase the amount of sugarcane handled by the mill. For the sugar mill in this study, if the proposed heuristics and the DE are adopted for scheduling and sequencing the sugarcane vehicles, the mill's capacity can be increase by 7.08–10.41% (i.e. 85,000–125,000 tonnes of sugarcane per crop year), resulting to the lower production cost per ton of sugarcane for the mill and also the higher utilization of the growers' vehicles.

## 6 Conclusions and future developments

This research addresses the scheduling and sequencing of sugarcane vehicles for the dump tippers at a sugar mill in Thailand. The problem can be formulated as the parallel capacitated machines with machine restrictions, job grouping, and sequencing

**Table 8** Computational results of the makespan (minutes)

Problem	Number of machines ( <i>M</i> )	Number of jobs ( <i>N</i> )	Group size number	M0	H2	H3	H4	H5	DE6 Best	DE7 Best	DE8 Best	DE9 Best
1	4	4	4	9.52	9.52	9.52	9.52	9.59	9.52	9.52	9.52	9.52
2	4	8	4	18.97	19.65	19.65	19.65	19.03	18.97	18.97	18.97	18.97
3	4	8	8	15.36	19.65	15.54	15.36	15.42	15.36	15.36	15.36	15.36
4	4	12	4	29.86	31.25	31.25	30.82	30.20	29.86	29.86	29.86	29.86
5	4	12	6	26.09	31.25	31.14	30.71	26.29	26.09	26.09	26.09	26.09
6	4	12	12	25.56	31.25	26.33	26.32	26.30	25.92	25.83	25.68	25.71
7	4	24	4	54.38	59.31	59.31	57.84	56.88	55.33	55.04	55.27	55.05
8	4	24	6	n/a	59.31	59.20	57.73	52.52	50.97	50.92	51.22	50.92
9	4	24	12	n/a	59.31	54.38	53.35	49.53	49.09	48.66	49.28	48.24
10	4	24	24	n/a	59.31	48.12	47.72	48.82	48.85	47.48	47.93	47.72
11	6	24	24	n/a	48.40	47.57	47.52	50.43	18.43	18.20	18.55	18.36
12	8	24	24	n/a	48.65	48.12	47.72	48.82	18.41	17.42	18.51	18.21
13	4	240	12	n/a	556.29	534.20	527.48	495.25	476.75	461.98	472.77	470.47
14	4	240	24	n/a	556.29	513.62	506.24	492.08	476.04	459.49	476.53	463.07
15	4	240	48	n/a	556.29	486.77	477.77	492.05	475.56	468.05	475.69	458.46
16	4	240	60	n/a	556.29	484.35	478.44	484.51	472.89	463.55	472.43	466.08
17	4	240	120	n/a	556.29	464.51	461.00	487.30	475.25	461.53	476.84	456.48
18	4	240	240	n/a	556.29	463.70	466.41	483.15	478.95	462.13	474.47	461.96
19	6	240	240	n/a	531.91	459.12	459.61	453.16	135.04	137.55	135.33	136.73
20	8	240	240	n/a	514.72	463.75	466.39	472.39	135.96	137.47	137.27	135.90

Problems too large to be solved by Lingo/CPLEX are represented as n/a

**Table 9** Compare solution quality of the proposed heuristics with mathematical model by using HP

Problem test	Number of machines ( <i>M</i> )	Number of jobs ( <i>N</i> )	Group size number ( <i>G</i> )	H2	H3	H4	H5	DE6	DE7	DE8	DE9
1	4	4	4	100.00	100.00	100.00	99.34	100.00	100.00	100.00	100.00
2	4	8	4	96.55	96.55	96.55	99.67	100.00	100.00	100.00	100.00
3	4	8	8	78.18	98.82	100.00	99.62	100.00	100.00	100.00	100.00
4	4	12	4	95.55	95.55	96.90	98.87	100.00	100.00	100.00	100.00
5	4	12	6	83.47	83.76	84.95	99.23	100.00	100.00	100.00	100.00
6	4	12	12	81.80	97.10	97.13	97.21	98.63	98.99	99.53	99.42
7	4	24	4	91.69	91.69	94.01	95.61	98.29	98.80	98.39	98.78
<i>Average</i>				89.60	96.55	94.78	95.65	98.51	99.56	99.68	99.70

*Average*: the average of % HP of a proposed algorithms (heuristics and DE)

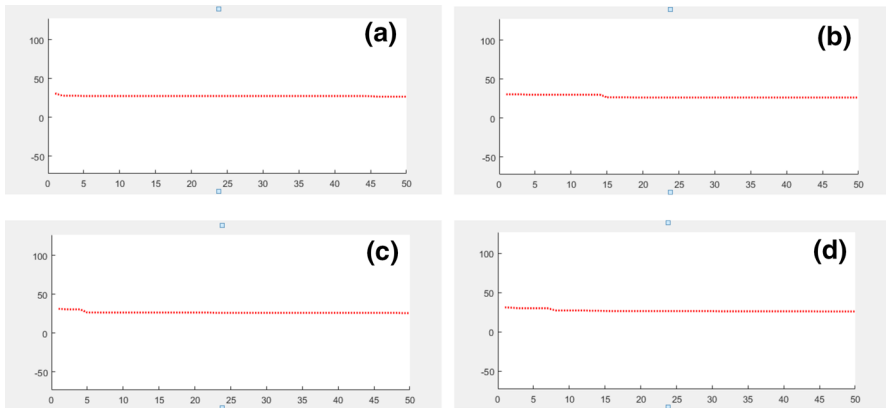
**Table 10** Performance computational results of the makespan (minutes) by using RI

Problem test	Num-ber of machines ( <i>M</i> )	Number of jobs ( <i>N</i> )	Group size number ( <i>G</i> )	H2 versus H3	H2 versus H4	H2 versus H5	H2 versus DE6	H2 versus DE7	H2 versus DE8	H2 versus DE9	DE6 versus DE7	DE6 versus DE8	DE6 versus DE9
1	4	4	4	0.00	0.00	- 0.66	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	4	8	4	0.00	0.00	3.13	3.45	3.45	3.45	3.45	0.00	0.00	0.00
3	4	8	8	20.89	21.82	21.52	21.82	21.82	21.82	21.82	0.00	0.00	0.00
4	4	12	4	0.00	1.40	3.36	4.45	4.45	4.45	4.45	0.00	0.00	0.00
5	4	12	6	0.35	1.75	15.89	16.53	16.53	16.53	16.53	0.00	0.00	0.00
6	4	12	12	15.76	15.79	15.85	17.06	17.37	17.82	17.73	0.36	0.91	0.80
7	4	24	4	0.00	2.47	4.10	6.71	7.20	6.81	7.18	0.52	0.10	0.50
8	4	24	6	0.18	2.65	11.44	14.07	14.15	13.64	14.15	0.09	- 0.49	0.09
9	4	24	12	8.30	10.06	16.48	17.22	17.95	16.90	18.67	0.87	- 0.39	1.74
10	4	24	24	18.86	19.54	17.69	17.63	19.95	19.19	19.55	2.82	1.90	2.33
11	6	24	24	17.25	18.67	18.75	13.77	68.49	68.88	68.29	1.23	- 0.66	0.38
12	8	24	24	17.97	18.86	19.54	17.69	68.95	70.62	68.79	5.38	- 0.53	1.10
13	4	240	12	3.97	5.18	10.97	14.30	16.95	15.01	15.43	3.10	0.84	1.32
14	4	240	24	7.67	9.00	11.54	14.43	17.40	14.34	16.76	3.48	- 0.10	2.72
15	4	240	48	12.50	14.11	11.55	14.51	15.86	14.49	17.59	1.58	- 0.03	3.60
16	4	240	60	12.93	13.99	12.90	14.99	16.67	15.07	16.22	1.98	0.10	1.44
17	4	240	120	16.50	17.13	12.40	14.57	17.03	14.28	17.94	2.89	- 0.33	3.95
18	4	240	240	16.64	16.16	13.15	13.90	16.93	14.71	16.96	3.51	0.94	3.55
19	6	240	240	2.28	15.65	15.56	16.75	75.19	74.73	75.14	- 1.86	- 0.22	- 1.25
20	8	240	240	7.47	16.63	16.16	15.08	75.56	75.29	75.32	- 1.11	- 0.97	0.04
Average				8.41	9.44	11.33	12.85	13.98	13.03	14.03	1.33	0.21	1.38

**Table 11** Average computational time (seconds)

Problem	Number of machines ( <i>M</i> )	Number of jobs ( <i>N</i> )	Group size number ( <i>G</i> )	M1	H2	H3	H4	H5	DE6	DE7	DE8	DE9
1	4	4	4	1.00	1.97	1.98	1.91	1.95	120	120	120	120
2	4	8	4	1.00	2.24	2.26	2.19	2.19	120	120	120	120
3	4	8	8	4.00	2.70	2.16	2.12	2.29	120	120	120	120
4	4	12	4	6.00	2.35	2.53	2.42	2.13	120	120	120	120
5	4	12	6	22.00	2.20	2.23	2.25	2.39	120	120	120	120
6	4	12	12	9470.00	2.35	2.34	2.50	2.45	120	120	120	120
7	4	24	4	27,762.00	2.52	2.48	2.51	3.08	120	120	120	120
8	4	24	6	n/a (> 85 h)	3.76	3.86	3.91	3.89	120	120	120	120
9	4	24	12	n/a	3.79	3.88	3.90	3.97	120	120	120	120
10	4	24	24	n/a	3.96	4.33	4.16	4.28	120	120	120	120
11	6	24	24	n/a	3.43	3.73	3.46	3.50	120	120	120	120
12	8	24	24	n/a	3.41	3.55	3.55	3.59	120	120	120	120
13	4	240	12	n/a	4.12	4.12	4.61	3.25	300	300	300	300
14	4	240	24	n/a	3.01	3.14	3.26	3.99	300	300	300	300
15	4	240	48	n/a	3.24	3.37	3.39	6.36	300	300	300	300
16	4	240	60	n/a	1.97	1.98	1.91	1.95	300	300	300	300
17	4	240	120	n/a	2.24	2.26	2.19	2.19	300	300	300	300
18	4	240	240	n/a	2.70	2.16	2.12	2.29	300	300	300	300
19	6	240	240	n/a	6.22	6.44	6.62	8.99	300	300	300	300
20	8	240	240	n/a	5.43	5.09	6.47	8.66	300	300	300	300

Problems too large to be solved by Lingo/CPLEX are represented by n/a



**Fig. 13** The best objective function of the generation for test problem 11 **a** traditional DE6, **b** traditional DE7, **c** modified DE8, and **d** modified DE9

independent setup time (PCMGS,  $R_m|p_{i,j}, s_{i,j}, M_{ijt}, Grouping|C_{max}$ ). The objective was to find the optimal scheduling for operational sequences in order to minimize the makespan. To solve the problem, a mathematical model was developed to solve small-scale problems. For the large-scale problems, constructive heuristics and traditional and modified DE were developed. The traditional DE improved the efficiency by adjusting job grouping adaptive parameters. It made a balance between exploration and exploitation of the search space for improvement of the evolution process. The different problem scales were brought to test against the efficiency of the approaches. The experimental results illustrated the effectiveness of the proposed algorithms for solving all problem sizes. Even though the related computational time showed that they used different CPU times under different test problem situations, they were able to provide the best solutions within an acceptable time frame of the sugar mill.

The mathematical and heuristic models developed in this paper are easily adaptable and should prove to be beneficial to other sugar industries including other similar agro-food sectors in Thailand and around the world for sustainable production by increasing productivity for both growers and mills. However, there is still much opportunity to extend our work in various aspects. Since there is uncertainty in road traffic of sugarcane vehicles and also different vehicle capacities, future research should be in optimization-simulation modeling for processes from sugarcane fields to the mill-yard. Another valuable avenue for future research is to consider failures of dump tippler machines. We believe that this can add to the ability of our work to model real world problems and will be a valuable extension. In addition, in our work, DE was applied. Even though this method is widely used, new metaheuristic methods for scheduling dump tippler machines to determine better and more efficient solutions should be carried out in future work.

**Acknowledgments** This study was supported by the Thailand Research Fund (TRF) through the Research and Researchers for Industries-RRi (Grant No. PHD58I0041) in collaboration with Rajburi Sugar Company Limited, Department of Business Administration, Faculty of Business, Economics and Statistics,

University of Vienna, Vienna, Austria, and Research Unit on System Modeling for Industry (SMI), Department of Industrial Engineering, Faculty of Engineering, Khon Kaen University, Khon Kaen, Thailand. Thanks are also due to the staff of the sugar mill for their assistance and facilities given during the time of this research. The authors would also like to thank Prof. Somnuk Theerakulpisut and Mr. Ian Thomas for English language review of the manuscript.

## References

- Arjona E, Bueno G, Salazar L (2001) An activity simulation model for the analysis of the harvesting and transportation systems of a sugarcane plantation. *Comput Elect Agric* 32(3):247–264
- Astika IW, Cahyoutomo R, Lilik Mulyantara FX (2001) Development computer software for optimum scheduling of farm machinery operations in a sugarcane plantation. *IFAC Proc Vol* 34(11):217–220. [https://doi.org/10.1016/S1474-6670\(17\)34135-6](https://doi.org/10.1016/S1474-6670(17)34135-6)
- Baikow VE (2013) Manufacture and refining of raw cane sugar. *Manuf Refin Raw Cane Sugar*. <https://doi.org/10.1016/b978-1-4832-3212-6.50016-5>
- Bocanegra-Herrera CC, Vidal-Holguín CJ (2016) Development of a simulation model as a decision support system for sugarcane supply. *DYNA* 83(198):180
- Chakravarthy SR, Karatzas HD (2013) Two-server parallel system with pure space sharing and Markovian arrivals. *Comput Operat Res* 40(1):510–519
- Chiang C-W, Lee W-P, Heh J-S (2010) A 2-Opt based differential evolution for global optimization. *Applied Soft Computing* 10(4):1200–1207
- Centeno G, Armacost RL (1997) Parallel machine scheduling with release time and machine eligibility restrictions. *Comput Ind Eng* 33(1–2):273–276. [https://doi.org/10.1016/S0360-8352\(97\)00091-0](https://doi.org/10.1016/S0360-8352(97)00091-0)
- Dechompai D et al (2017) A differential evolution algorithm for the capacitated VRP with flexibility of mixing pickup and delivery services and the maximum duration of a route in poultry industry. *J Intell Manuf* 28(6):1357–1376. <https://doi.org/10.1007/s10845-015-1055-3>
- Dolgui A et al (2009) Multi-product lot-sizing and scheduling on unrelated parallel machines to minimize makespan. In: *IFAC proceedings volumes (IFAC-PapersOnline)*. IFAC. <https://doi.org/10.3182/20090603-3-ru-2001.0553>
- Fox K, Korupolu M (2013) Weighted flowtime on capacitated machines. In: *Proceedings of the 2013 annual ACM-SIAM symposium on discrete algorithms*, pp 129–143. Retrieved from <https://dblp.uni-trier.de/db/conf/soda/soda2013.html#FoxK13>. <https://doi.org/10.1137/1.9781611973105.10>
- Garey MR, Johnson DS (1978) “Strong” NP-completeness results: motivation, examples, and implications. *J ACM* 25(3):499–508. <https://doi.org/10.1145/322077.322090>
- Gokhale R, Mathirajan M (2012) Scheduling identical parallel machines with machine eligibility restrictions to minimize total weighted flowtime in automobile gear manufacturing. *Int J Adv Manuf Technol* 60(9–12):1099–1110. <https://doi.org/10.1007/s00170-011-3653-3>
- Higgins A et al (1998) Optimising harvest date in sugar production: a case study for the Mossman mill region in Australia: I. Development of operations research model and solution. *Field Crops Res* 57(2):153–162. [https://doi.org/10.1016/S0378-4290\(97\)00116-0](https://doi.org/10.1016/S0378-4290(97)00116-0)
- Higgins A, Davies I (2005) A simulation model for capacity planning in sugarcane transport. *Comput Electron Agric* 47(2):85–102
- Hu X, Bao JS, Jin Y (2010) Minimising makespan on parallel machines with precedence constraints and machine eligibility restrictions. *Int J Prod Res* 48(6):1639–1651. <https://doi.org/10.1080/00207540802620779>
- Iannoni AP, Morabito R (2006) A discrete simulation analysis of a logistics supply system. *Transport Res Part E: Logist and Transport Rev* 42(3):191–210
- Jenkins GH (2013) *Introduction to cane sugar technology*. Elsevier, Amsterdam
- Kachitvichyanukul V (2012) Comparison of three evolutionary algorithms: GA, PSO, and DE. *Ind Eng Manag Syst* 11(3):215–223. <https://doi.org/10.7232/iems.2012.11.3.215>
- Lee K, Leung JYT, Pinedo ML (2011) Scheduling jobs with equal processing times subject to machine eligibility constraints. *J Sched* 14(1):27–38. <https://doi.org/10.1007/s10951-010-0190-0>
- Leung JY, Li C (2008) Scheduling with processing set restrictions: a survey. *Int J Prod Econ* 116(2):251–262

- Li YL, Zhang J (2011) A new differential evolution algorithm with dynamic population partition and local restart. In: Proceedings of the 13th annual conference on genetic and evolutionary computation, pp 1085–1092. <https://doi.org/10.1145/2001576.2001723>
- Martinez S, Dauzère-Pérès S, Guéret C, Mati Y, Sauer N (2006) Complexity of flowshop scheduling problems with a new blocking constraint. *Euro J Operat Res* 169(3):855–864
- Masoud M, Kozan E, Kent G (2015) Hybrid metaheuristic techniques for optimising sugarcane rail operations. *Int J Prod Res* 53(9):2569–2589. <https://doi.org/10.1080/00207543.2014.957870>
- Moonsri K, Sethanan K, Sangsawang C (2015) Metaheuristics for scheduling unrelated parallel machines with sequence-dependent setup time and machine eligibility. *Chiang Mai Univ J Nat Sci* 14(4):431–446. <https://doi.org/10.12982/cmujns.2015.0097>
- Muchow R et al (1998) Optimising harvest date in sugar production: a case study for the Mossman mill region in Australia. *Field Crops Res* 57(3):243–251. [https://doi.org/10.1016/S0378-4290\(97\)00135-4](https://doi.org/10.1016/S0378-4290(97)00135-4)
- Nait Tahar D, Yalaoui F, Chu C, Amodeo L (2006) A linear programming approach for identical parallel machine scheduling with job splitting and sequence-dependent setup times. *Int J Prod Eco* 99(1–2):63–73
- Nearchou AC (2006) Meta-heuristics from nature for the loop layout design problem. *Int J Product Eco* 101(2):312–328
- Office of Global Analysis (2015) Sugar: world markets and trade global sugar consumption outpaces production, foreign agricultural service. <http://usda.mannlib.cornell.edu/usda/fas/sugar/2010s/2015/sugar-11-19-2015.pdf>. Accessed 2 Sept 2016
- Office of the Cane and Sugar Board (2017) The annual report of sugarcane plantation and sugar production in 2016/2017. <http://www.ocsb.go.th/upload/journal/fileupload/923-9999.pdf>. Accessed 2 Sept 2016
- Pinedo ML (1995) Scheduling: Theory, algorithms, and systems. Theory, Algorithms, and Systems, Scheduling. <https://doi.org/10.1007/978-0-387-78935-4>
- Pitakaso R, Sethanan K (2016) Modified differential evolution algorithm for simple assembly line balancing with a limit on the number of machine types. *Eng Optim* 48(2):253–271. <https://doi.org/10.1080/0305215X.2015.1005082>
- Potts CN, Kovalyov MY (2000) Scheduling with batching: a review. *Eur J Oper Res* 120(2):228–249. [https://doi.org/10.1016/S0377-2217\(99\)00153-8](https://doi.org/10.1016/S0377-2217(99)00153-8)
- Rabadi G, Moraga R, Al-Salem A (2006) Heuristics for the unrelated parallel machine scheduling problem with setup times. *J Intell Manuf* 17:85–97. <https://doi.org/10.1007/s10845-005-5514-0>
- Sartori MMP et al (2001) Determination of the optimal quantity of crop residues for energy in sugarcane crop management using linear programming in variety selection and planting strategy. *Energy* 26(11):1031–1040. [https://doi.org/10.1016/S0360-5442\(01\)00052-4](https://doi.org/10.1016/S0360-5442(01)00052-4)
- Silva C, Magalhaes JM (2006) Heuristic lot size scheduling on unrelated parallel machines with applications in the textile industry. *Comput Indus Eng* 50(1–2):76–89
- Sethanan K, Neungmacha W (2014) Multi-objective particle swarm optimization for mechanical harvester route planning of sugarcane field operations. *Eur J Oper Res* 252(3):969–984. <https://doi.org/10.1016/j.ejor.2016.01.043>
- Sethanan K, Pitakaso R (2016a) Differential evolution algorithms for scheduling raw milk transportation. *Comput Electron Agric* 121:245–259. <https://doi.org/10.1016/j.compag.2015.12.021>
- Sethanan K, Pitakaso R (2016b) Improved differential evolution algorithms for solving generalized assignment problem. *Expert Syst Appl* 45:450–459. <https://doi.org/10.1016/j.eswa.2015.10.009>
- Sharma R et al (2010) An agent based dynamic resource scheduling model with FCFS-job grouping strategy in grid computing. *World Acad Sci Eng Technol* 4(4):823–827
- Storn R, Price K (1997) Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J Glob Optim* 11(4):341–359. <https://doi.org/10.1023/A:1008202821328>
- Tinnongwathana T (2013) Sugar cane harvest and transportation, Office of the Cane and Sugar Board. <http://oldweb.ocsb.go.th/udon/Alltext/I.Article/01-ArticleP9.1.htm>. Accessed 2 Sept 2016
- USDA (2017) Record global production spurs record consumption sugar overview. <https://public.govde.livery.com/accounts/USDAFAS/subscriber/new>. Accessed 2 Sept 2016
- Webster S, Baker K (1995) Scheduling groups of jobs on a single machine. *Oper Res* 43(4):692–703. <https://doi.org/10.1287/opre.43.4.692>
- Wu X, Che A (2018) A memetic differential evolution algorithm for energy-efficient parallel machine scheduling. *Omega (United Kingdom)*. <https://doi.org/10.1016/j.omega.2018.01.001>



- Weaver VM, Johnson M, Kasichayanula K, Ralph J, Luszczek P, Terpstra D, Moore S (2012) Measuring energy and power with PAPI. In: 2012 41st international conference on parallel processing workshops, pp 262–268. IEEE. <https://doi.org/10.1109/ICPPW.2012.39>
- Yuan S et al (2011) A job grouping approach for planning container transfers at automated seaport container terminals. *Adv Eng Inform* 25(3):413–426. <https://doi.org/10.1016/j.aei.2011.01.004>
- Yalaoui F, Chu C (2003) An efficient heuristic approach for parallel machine scheduling with job splitting and sequence-dependent setup times. *IIE Transactions* 35(2):183–190
- Yanyan Z, Tieke L, Bailin W (2011) Scheduling parallel multiple capacitated machines with sequence-dependent constraint. In: 2011 international conference on E-business and E-government, ICEE2011 - Proceedings, pp 2570–2573. <https://doi.org/10.1109/ICEBEG.2011.5882009>

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.