Original Research Paper

Solving Multi-Objective Master Production Scheduling Model of Kalak Refinery System Using Hybrid Evolutionary Imperialist Competitive Algorithm

[1]Shereen Saleem Sadiq, [2]Adnan Mohsin Abdulazeez and [3]Habibollah Haron

[1]*Department of Information Technology, College of Informatics, Duhok Polytechnic University, Duhok, KRG-Iraq*
[2]*Presidency of Duhok Polytechnic University, Duhok, KRG-Iraq*
[3]*Department of Computer Science, Applied Industrial Analytics Research Group, Universiti Teknologi Malaysia, Malaysia*

Abstract: The improvement of operational planning in the field of oil refinery management is becoming increasingly essential and valid. The influential primary factor, among others, is the ever-changing economic climate. The industry must continually assess the potential impacts of variations in the final product demand, price fluctuations, crude oil compositions and even seek out immediate opportunities within the market. The Master Production Schedule (MPS) is a planned process within the Production Management System that provides a mechanism for active collaboration between the marketing and manufacturing processes. However, the problem of MPS is a predictable non-deterministic, polynomial-time and NP-hard combination optimisation issue. The global search for the best solution to the MPS problem involves determination and funds that many industries are reluctant to provide. Hence, the alternative approach using meta-heuristics could provide desirable and workable answers in a realistic computing period. In this paper, a unique hybrid Multi-Objective Evolutionary Imperialist Competitive Algorithm (MOEICA) is proposed. The algorithm combines the advantages of an Imperialist Competitive Algorithm (ICA) and a Genetic Algorithm (GA) to optimise a Multi-Objective Master Production Schedule (MOMPS). The primary objective is to integrate the ICA with GA operators. The paper will also apply the optimised MOMPS to the Kalak Refinery System (KRS) operations using the proposed algorithm. The application involves determining the available capacity of each production line by estimating the parametric values for all failures. In addition, the gross requirements using demand forecasting and neural networks are defined. The proposed algorithm proved efficient in resolving the issues of the MOMPS model within KRS compared to the NSGAII and MOPSO algorithms. The results reflect that the novel MOEICA algorithm outperformed NSGAII and MOPSO in almost all measurements.

Keywords: Master Production Scheduling, Genetic Algorithm, Imperialist Competitive Algorithm, Hybrid Algorithm, Multi-Objective Optimization

Introduction

Oil refinery production planning is a vital but complex, intricate system. Production plan optimisation requires a mathematical model to represent the production system that can be utilised by the optimisation model.

An oil refinery is part of an industry that significantly contributes to and impacts on the economy of any nation.

Refinery management is, therefore, diligently committed to the ongoing improvement of operational planning. With the aprimary factor, among others, being the ever-changing economic climate. The industry must continually assess the potential impacts of variations in the final product demand, price fluctuations and crude oil compositions or even seek out immediate opportunities within the market (Joly *et al.*, 2002).

Crude oil comprises a vast number of hydrocarbons, which are injected into the base of the crude distillation unit. Under the extremely high temperatures within this unit, the hydrocarbons, except the heaviest ones, are subsequently transformed into gas. At the higher levels of the distillation unit typically characterised by lower temperatures, the hydrocarbons are converted back to the liquid stage in the different areas of the distillation. Therefore, the distillation process separates crude oil into typically five streams whose components are of varied weight (Göthe-Lundgren *et al.*, 2002).

For a manufacturing company to remain competitive in a global market, the customer's requirements must be consistently and promptly met with a high standard of products and services. This is challenging for industry decision-makers, particularly in current market conditions. This scenario is characterised by short product life-cycles, varied customised products, irregular and unpredictable demand patterns and short customer lead times. However, the capacity to attain a competitive lead has become more challenging. The difficulty is ascribed to the ambiguity of the current market settings. The constant, hitch-free communication and efficient collaboration of the marketing, sales and manufacturing departments is critical to the management and sustenance of the customer's demand using the company's resources. The Master Production Schedule (MPS) is a planned process within the Production Management System that provides a mechanism for active collaboration between the marketing and manufacturing processes (Supriyanto and Noche, 2011).

In most industries around the world, the formation of an MPS takes into consideration a range of objectives. Typically, these include maximisation of service levels, efficient and cost-effective use of resources and inventory level minimisation. Unfortunately, the level of difficulty and intense effort that entails the creation of a master plan significantly increases as the production scenario rises. Furthermore, it occurs when resources are restricted, which is the case for most industries nowadays. It is due to such complexities that most enterprises will usually resort to using basic heuristics implemented in spreadsheets. This approach aims to provide a speedy plan but may compromise efficiency and expenditure (Soares and Vieira, 2008).

In general, the challenges of MPS are associated with NP-hard problems. This basically means the existence of an algorithm that generates an ideal answer is improbable in polynomial time. Essentially, this indicates the handling time necessary to resolve such hitches will soar rapidly with the increasing enormity of the problem. Consequently, the search for an ideal answer is challenging. Therefore, evolutionary algorithms otherwise termed meta-heuristic algorithms are regularly used to attain optimal solutions. Examples include; genetic algorithm, simulated annealing and Tabu Search among others (Supriyanto and Noche, 2011). The principal benefits of an evolutionary algorithm is that the objective function does not need to be differentiable or continuous. Hence, the evaluation of gradients is not necessary and as such, can deviate from the local minima (Tsoulos, 2009).

The integration of two or more diverse algorithms into a lone hybrid algorithm has been enthused by the prospects of attaining higher performance compared to the individual component algorithms. The outcome is a novel class of algorithms based on the canopy of hybrid algorithms. Typically, the hybrid algorithm integrates the fortes of the different component algorithms. Hence, the resultant algorithm provides collective benefits such as generating enhanced answers within a quicker time span. Moreover, it efficiently addresses any difficulties with enormous sized inputs, particularly regarding the problems of NP (Mitras and Sultan, 2013).

In this paper, a hybrid Multi-Objective Evolutionary Algorithm (MOEICA) has been proposed to solve a multi-objective MPS problem. This algorithm applies the concepts of the Imperialist Competitive Algorithm (ICA) and the Genetic Algorithm (GA). The colonies within each empire in the ICA represent a small population and communicate with each other using genetic operators and imperialist operators. The optimised MOMPS is applied to the Kalak Refinery System (KRS) which is located near Erbil City. The KRS is considered one of the newest and most modern refineries in Kurdistan-Iraq, which is operated by the KAR Group. The construction of the Kalak Refinery in Erbil began in 2005, with the refinery capacity reaching 100,000 Barrels Per Day (BPD) in 2011. The application involves determining the available capacity of each production line and establishes the gross requirements using demand forecasting and neural networks.

The paper is structured as follows: Section II presents a brief revision of the foundation of the MPS and MOMPS model of KRS. The proposed algorithm aims to resolve the problem, as reflected in Section III. In Section IV, the MOMPS is created and proposed for KRS. The MOEICA implementation and results of the MPS-KRS model are outlined in Section V. The comparisons between MOEICA, NSGAII and MOPSO are shown in Section VI. The conclusions of this study are presented in Section VII.

Foundation of MPS

The MPS is a strategic process that combines production planning and control with the aim of interpreting the critical objectives of a business plan into

a predicted report. Hence, it is the starting point from which all other schedules at the lower levels originate (Higgins and Browne, 1992). Likewise, the MPS is a critical crossing point between manufacturing and marketing. It is a straight link to customer service and ensures efficient usage of production resource (Vollmann *et al.*, 1997). The mishandling of MPS is termed an essential operational problem, possibly resulting in poor customer satisfaction. Therefore, Slack *et al.* (2010) described MPS as a vital schedule for planning and controlling the operations of any business.

According to the American Production and Inventory Control Society (APICS) (Proud, 1999), the MPS declares what the company expects to manufacture. Hence, it is a decision system that emphasizes the Material Requirements Planning (MRP).

It symbolises the company's production plan based on the particular configuration, quantities and dates. Although the MSP cannot be termed a sales forecast, it is regarded as one of the most vital informational constituents of the system. Other significant concerns must be accounted for, such as incomplete orders, materials availability, current capacity and administrative policies and goals, among others. It also presents factors such as projected inventory balance, Available To Promise (ATP) and capacity used.

Wu *et al.* (2002) developed a mathematical model and genetic algorithm for MPS that unites many methods that fulfil the limitations for making an optimised MPS for assembly, processing and production lines. Vieira and Ribas (2004) adapted simulated annealing to resolve the problem of production planning or precisely an MPS problem. The study exposed some weaknesses of simulated annealing such as overcoming the local optimum (Soares and Vieira, 2008) developed a novel genetic algorithm structure for resolving MPS problems. The study formulated a fitness function aimed at reducing inventory levels, maximising service level (unfulfilled reduced condition), lessen overtime and inventory levels beneath the safety stock. Finally, Vieira *et al.* (2004) associated and applied genetic algorithms and simulated annealing to MPS difficulties.

Mathematical Model of Multi Objective MPS for KRS

This paper is based on the mathematical model proposed by (Sadiq *et al.*, 2018). The block diagram and simplified model, extracted from the article is shown in Fig. 1. A detailed explanation of the mathematical model is presented in the paper:

$$Min\ EI = \frac{\sum_{k=1}^{7}\sum_{p=1}^{P}EI_{kp}}{TH} \tag{1}$$

$$Min\ RNM = \frac{\sum_{k=1}^{7}\sum_{p=1}^{P}RNM_{kp}}{TH} \tag{2}$$

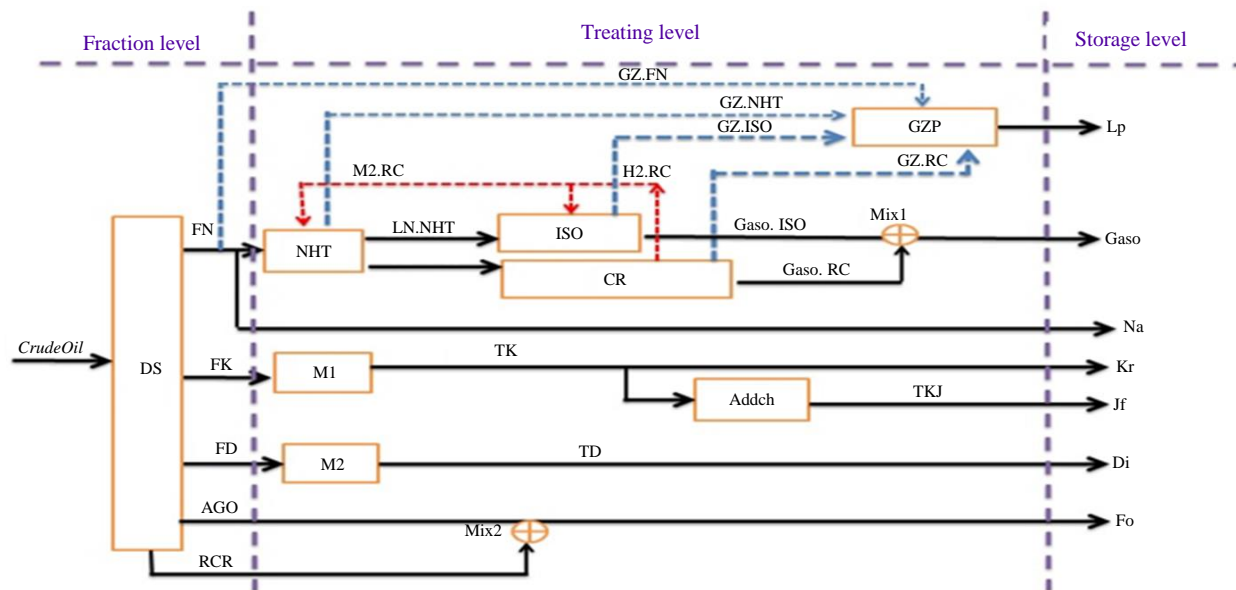$$Min\ BSS = \frac{\sum_{k=1}^{7}\sum_{p=1}^{P}BSS_{kp}}{TH} \tag{3}$$

Fig. 1: KRS block diagram

Subject to:

$$BI_{kp} = \begin{cases} OH_k & if\ (p=1) \\ EI_{k(p-1)} & if\ (p>1) \end{cases} \tag{4}$$

$$EI_{kp} = \max\left[0,\left(\left(MPST_{kp}+BI_{kp}\right)-GR_{kp}\right)\right] \tag{5}$$

$$RNM_{kP} = \max\left[0,\left(GR_{kp}-\left(MPST_{kp}+BI_{kp}\right)\right)\right] \tag{6}$$

$$BSS_{kp} = \max\left[0,\left(SS_{kp}-EI_{kp}\right)\right] \tag{7}$$

$$\begin{aligned} MPST_{1p} &= Na_p, MPST_{2p}=Kr_p, MPST_{3p}=Di_p, MPST_{4p} \\ &= Fo_p, MPST_{5p}=JF_p, MPST_{6p}Gaso_p, MPST_{7p}=LP_p \end{aligned} \tag{8}$$

where:

$$MPST_{kp} \geq 0; k=1\ldots7; p=1\ldots P$$

Distillation Unit Constraints

$$\begin{aligned} \frac{1}{0.98985}*\left(FN+FK+FD+AGO+RCR\right) &\leq CrudeOil \\ &\leq \frac{1}{0.97004}*\left(FN+FK+FD+AGO+RCR\right) \end{aligned} \tag{9}$$

$$CrudeOil \leq n*4769.619 \tag{10}$$

NHT Unit Constraints

$$\begin{aligned} \frac{1}{0.94002}\left(HN.NHT+LN.NHT\right) \\ \leq FN \geq \frac{1}{0.95996}\left(HN.NHT+LN.NHT\right) \end{aligned} \tag{11}$$

$$FN \leq n*1373.173 \tag{12}$$

$$0.01788*FN \leq GZ.FN \leq 0.01858*FN \tag{13}$$

$$0.0199*\left(FN-Na\right) \leq GZ.NHT \leq 0.0233*\left(FN-Na\right) \tag{14}$$

$$Na+GZ.FN+GZ.NHT+LN.NHT+HN.NHT \leq FN \tag{15}$$

$$I_{Na} \leq 5830.8 \tag{16}$$

ISO Unit Constraints

$$\frac{1}{0.76783}*ISO \leq LN.NHT \leq \frac{1}{0.76781}*ISO \tag{17}$$

$$GZ.ISO+Gaso.ISO \leq LN.NHT+H2.RC \tag{18}$$

$$0.0125*LN.NHT \leq GZ.ISO \leq 0.085*LN.NHT \tag{19}$$

CR Unit Constraints

$$\frac{1}{0.86088}*CR \leq HN.NHT \leq \frac{1}{0.86086}*CR \tag{20}$$

$$H2.RC+Gaso.RC \leq HN.NHT \tag{21}$$

$$0.0188*HN.NHT \leq GZ.CR \leq 0.0199*HN.NHT \tag{22}$$

$$0.0455*HN.NHT \leq H2.CR \leq 0.046*HN.NHT \tag{23}$$

Merox1 Constraints (M1)

$$\frac{1}{0.999}\left(TK+TKJ\right) \leq FK \leq \frac{1}{0.998}\left(TK+TKJ\right) \tag{24}$$

$$FK \leq n*202.947 \tag{25}$$

$$TK+TKJ \leq FK \tag{26}$$

$$I_{Kr} \leq 6268.368 \tag{27}$$

$$I_{JF} \leq 2274.184 \tag{28}$$

Merox2 Constraints (M2)

$$\frac{1}{0.9317}\left(TD\right) \leq FD \leq \frac{1}{0.945}\left(TD\right) \tag{29}$$

$$FD \leq n*750.738 \tag{30}$$

$$I_{Di} \leq 13572.52 \tag{31}$$

Mix 1 Constraints

$$Gaso.ISO+Gaso.CR \geq Gaso \tag{32}$$

$$I_{Gaso} \leq 11673.64 \tag{33}$$

Mix 2 constraints

$$0.96572*\left(AGO+CRC\right) \geq Fo \tag{34}$$

$$AGO \leq n*166.937 \tag{35}$$

$$RCR \leq n*2227.412 \tag{36}$$

$$I_{Fo} \leq 34623.6 \tag{37}$$

GZP Constraints

$$LP \leq GZ.NHT+GZ.RC+GZ.ISO+GZ.FN \tag{38}$$

$$I_{LP} \leq 1548 \tag{39}$$

In this formulation, each character expresses the value as follows. p: Total number of planning periods, TH: Total planning horizon, EI_{kp}: Ending inventory level generated for product k at period p, RNM_{kp}: Requirements not met for product k at period p, BSS_{kp}: Quantity below safety inventory level for product k at period p, BI_{kp}: Initial inventory level of the product k at period p, OH_k: Initial available inventory (on-hand), at the first scheduling period, GR_{kp}: Gross requirement for product k at period p, SS_{kp}: Safety inventory level for product k at period p, $MPST_{kp}$: Total quantity to be manufactured of the product k at period p, n: The number of days in each planning period.

Mathematical Model of Multi Objective MPS for Kalak Refinery System

In this section, the Master Production Schedule (MPS) of KRS was created based on the following steps described.

A. Determine MPS Parameters

Table 1 shows the details of the MPS parameters required to create the MPS problem for the refinery plant. The production rates in Table 2 are primary, before calculating availability.

B. Determine Gross Requirements

The gross forecast for demand is considered the most critical input and plays a vital role in the final MPS formation process. The neural network model for BP consists of a robust performance for fault-tolerant and non-linear map capacity along with learning and self-adaption. It can similarly be adapted to resolve difficulties, for example, the non-determinacy conclusion of a complex causal relation, classification, recognition, judgment among others. It is thus intensely suggested that a BP network algorithm is used for the gross requirements of MPS in the forecast model.

For this paper, we selected the weekly demand data for particular products. The sample data covered the period from 1 January 2013 to 31 December 2017 and comprised a sum of 260 observations. The data were divided into training, validation and testing sets to ascertain the best neural network structure for the experiment. In general, a three-layer feed-forward neural network model was used in this study, as shown in Fig. 2 and Table 3.

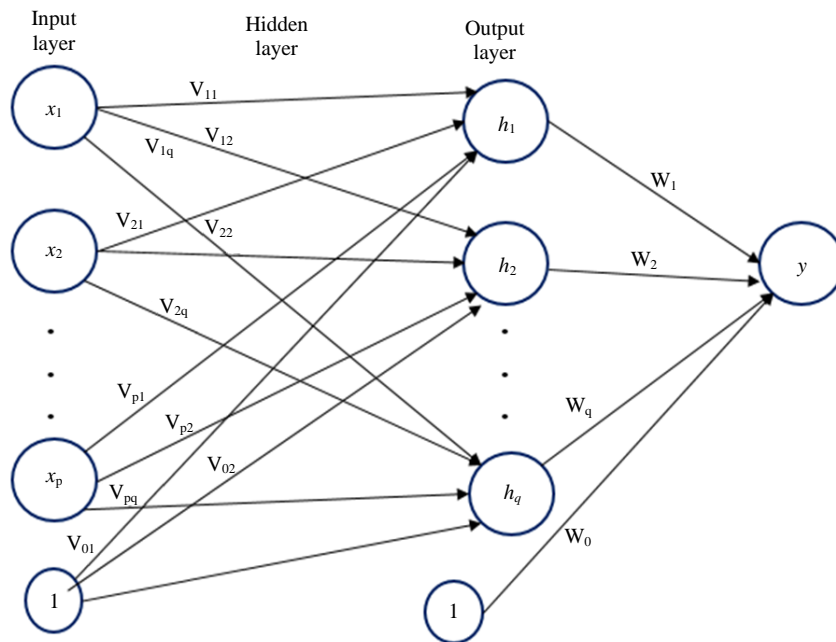**Fig. 2:** The architecture of the artificial neural network model (p, q, 1)

Table 1: The input parameters of MPS

Parameters	Value
K	7(Na, Kr, Di, FO, JF, Gaso, LPG)
P	8 weeks
UR_k	Table 2.
OH_k	(196.576, 90.016, 554.032, 1831.616, 92.24, 731.47, 34.416)
SS_{kp}	(1017, 1093.32, 2367.3, 6039, 396.66, 2344.05, 270)

Table 2: The production rates (M3/Hour)

Product	Quantity
Na	55
Kr /Jf	9
Di	31
FO	87
Gaso	55.6
LPG	2

Table 3: The best ANN structures and gross requirements

Prod.	Str.	Weeks							
		1st	2nd	3rd	4th	5th	6th	7th	8th
Na	4,4,1	1606.83	2131.23	5137.1700	2103.36	2131.23	1497.80	2131.23	1834.29
Kr	2,6,1	823.24	641.39	641.4000	641.37	641.37	641.37	641.37	641.37
Di	2,6,1	4515.18	3951.38	4514.3500	4215.19	4512.42	4447.62	4507.16	4499.41
Fo	2,4,1	14224.66	10899.63	15734.4900	16231.74	14506.88	15160.65	14339.80	14521.74
JF	2,4,1	759.76	647.53	797.1039	796.63	790.00	790.39	790.83	790.79
Gaso	4,4,1	5386.78	5974.22	5876.3900	5889.54	5923.58	6008.18	5991.96	5997.70
LPG	3,4,1	250.14	248.47	248.4700	248.47	248.47	248.47	248.47	248.47

Table 4: The parameter values of failure control

Fail.	Res.	Probabilistic Distribution	Availability	MTTF		MTBF	
				μ	σ	μ	σ
STO_1	Ds	Normal	90.36%	21.68	7.2	2.3	0.9
STO_2	GZP	Normal	91.11%	21.86	7.2	2.1	0.9
STO_3	Mix1	Normal	88.39%	21.20	7.0	2.8	1.1
STO_4	M1	Normal	94.94%	22.78	7.5	1.2	0.5
STO_5	M2	Normal	85.97%	20.62	6.8	3.4	1.4
STO_6	Mix2	Normal	92.00%	22.07	7.3	1.9	0.8

Table 5: The available production rates (M3/Hour)

Product	Quantity
Na	49.70
Kr /Jf	8.20
Di	27.40
FO	82.60
Gaso	47.80
LPG	1.84

C. Determine Production Rates

To determine production rates, the random machine failure resulting in a specific probabilistic distribution was simulated. The ideal scenario for this simulation is to initiate failure in active work units. Six failure controls STO_1, STO_2, STO_3, STO_4, STO_5 and STO_6 are set to trigger the random failure in work units DS, GZP, Mix1, M1, M2 and Mix2, respectively. The parameter values of all failure controls are presented in Table 4. Table 5 represents the available productions rates dependent on the availability ratios.

D. Create MPS Model

After substituting parameters values from Table 1, Table 2 into the following equations, the three objectives for the MPS problem and production units an obtained as follows:

$$Min\ EI = \frac{\sum_{k=1}^{7}\sum_{p=1}^{8} EI_{kp}}{8} \tag{40}$$

$$Min\ RNM = \frac{\sum_{k=1}^{7}\sum_{p=1}^{8} RNM_{kp}}{8} \tag{41}$$

$$Min\ BSS = \frac{\sum_{k=1}^{7}\sum_{p=1}^{8} BSS_{kp}}{8} \tag{42}$$

S.T:

$$BI_{kp} = \begin{cases} OH_k & if\ (p=1) \\ EI_{k(p-1)} & if\ (p>1) \end{cases} \tag{43}$$

$$EI_{kp} = \max\left[0,\left(\left(MPST_{kp} + BI_{kp}\right) - GR_{kp}\right)\right] \tag{44}$$

$$RNM_{kP} = \max\left[0,\left(GR_{kp} - \left(MPST_{kp} + BI_{kp}\right)\right)\right] \tag{45}$$

$$BSS_{kp} = \max\left[0,\left(SS_{kp} - EI_{kp}\right)\right] \tag{46}$$

$$MPST_{1p} = Na_p, MPST_{2p} = Kr_p, MPST_{3p} = Di_p, MPST_{4p} = Fo_p, MPST_{5p} = JF_p, MPST_{6p} Gaso_p, MPST_{7p} = LP_p \tag{47}$$

where:

$$MPST_{kp} \geq 0; k = 1\ldots7; p = 1\ldots8$$

Distillation unit Constraints

$$\frac{1}{0.98985} * (FN + FK + FD + AGO + RCR) \leq CrudeOil$$
$$\leq \frac{1}{0.97004} * (FN + FK + FD + AGO + RCR) \tag{48}$$

$$CrudeOil \leq 4769.619 \tag{49}$$

NHT unit Constraints

$$\frac{1}{0.94002}(HN.NHT + LN.NHT) \leq FN$$
$$\geq \frac{1}{0.95996}(HN.NHT + LN.NHT) \tag{50}$$

$$FN \leq 1373.173 \tag{51}$$

$$0.01788 * FN \leq GZ.FN \leq 0.01858 * FN \tag{52}$$

$$0.0199 * (FN - Na) \leq GZ.NHT \leq 0.0233 * (FN - Na) \tag{53}$$

$$Na + GZ.FN + GZ.NHT + LN.NHT + HN.NHT \leq FN \tag{54}$$

$$I_{Na} \leq 5830.8 \tag{55}$$

ISO Unit Constraints

$$\frac{1}{0.76783} * ISO \leq LN.NHT \leq \frac{1}{0.76781} * ISO \tag{56}$$

$$GZ.ISO + Gaso.ISO \leq LN.NHT + H2.RC \tag{57}$$

$$0.0125 * LN.NHT \leq GZ.ISO \leq 0.085 * LN.NHT \tag{58}$$

CR Unit Constraints

$$\frac{1}{0.86088} * CR \leq HN.NHT \leq \frac{1}{0.86086} * CR \tag{59}$$

$$H2.RC + Gaso.RC \leq HN.NHT \tag{60}$$

$$0.0188 * HN.NHT \leq GZ.CR \leq 0.0199 * HN.NHT \tag{61}$$

$$0.0455 * HN.NHT \leq H2.CR \leq 0.046 * HN.NHT \tag{62}$$

Merox1 Constraints (M1)

$$\frac{1}{0.999}(TK + TKJ) \leq FK \leq \frac{1}{0.998}(TK + TKJ) \tag{63}$$

$$FK \leq n * 202.947 \tag{64}$$

$$TK + TKJ \leq FK \tag{65}$$

$$I_{Kr} \leq 6268.368 \tag{66}$$

$$I_{JF} \leq 2274.184 \tag{67}$$

Merox2 Constraints (M2)

$$\frac{1}{0.9317}(TD) \leq FD \leq \frac{1}{0.945}(TD) \tag{68}$$

$$FD \leq n * 750.738 \tag{69}$$

$$I_{Di} \leq 13572.52 \tag{70}$$

Mix 1 Constraints

$$Gaso.ISO + Gaso.CR \geq Gaso \tag{71}$$

$$I_{Gaso} \leq 11673.64 \tag{72}$$

Mix 2 Constraints

$$0.96572 * (AGO + CRC) \geq Fo \tag{73}$$

$$AGO \leq n * 166.937 \tag{74}$$

$$RCR \leq n * 2227.412 \tag{75}$$

$$I_{Fo} \leq 34623.6 \tag{76}$$

GZP Constraints

$$LP \leq GZ.NHT + GZ.RC + GZ.ISO + GZ.FN \tag{77}$$

$$I_{LP} \leq 1548 \tag{78}$$

Proposed Algorithm Description

In this section, the proposed solution procedure is presented in detail. Hence, the standard ICA and GA must be first expounded, followed by an explanation of the hybrid algorithm.

A. Imperialist Competitive Algorithm (ICA)

The imperialist Competitive Algorithm (ICA) is a novel evolutionary algorithm within the evolutionary computation field based on the human socio-political evolution. The algorithm is initiated by creating a series of random candidate answers for optimisation in the search space problem. The randomly generated points are called the initial population (nations of the world). Furthermore, nations are classified into two sets: colonies and imperialists. The imperialist group is more powerful with significantly more colonies. However, the functional cost of the optimisation problem controls the power of individual nations. Based on each nation's prospects, a handful of

some first nations (i.e., nations with the least cost function), grow into imperialists thereby controlling other nations (colonies) to form empires (Atashpaz-Gargari and Lucas, 2007). The three critical operatives of this algorithm are assimilation, competition and revolution. This algorithm adopts the operation of assimilation. According to this policy, the imperialists try to advance their colonies' economies, cultures and politics. Such strategies enhance the colonies' enthusiasm in the direction of the imperialists. Furthermore, assimilation makes each empire's colonies draw closer to the imperialist in terms of socio-political features (optimisation search space). Revolution instigates abrupt or unplanned variations in the position of some nations in the search space. During assimilation and revolution, a colony could advance its position and the opportunity to control an entire empire or replace the imperialist state of the present empire (Atashpaz Gargari *et al.*, 2008).

In rival operations, imperialists often try to amass more colonies that draw closer to the imperialists. All empires attempt to conquer and possess the colonies that belong to other empires. An empire's power depends on the influence of its imperialists and colonies. In the stages of each algorithm, the empires can prospectively control a single or more colony of the frailest empire, based on their power or status. Thus, during the competition, the powerful imperialists are strengthened, whereas the weak imperialists are dissolved. With time, the frailer empires will mislay all their colonies, whereas the imperialists wander to the triumphant empires' colonies. Lastly, all weak empires will collapse, resulting in a single powerful empire. The colonies are unsystematically dispersed between the imperialists. Hence, the more powerful imperialists control more colonies (Hartmanis, 1982). The algorithm remains until the stop condition is fulfilled, such as a single imperialist remain. At this phase, the situation of the imperialist and its colonies will be unchanged (Atashpaz-Gargari and Lucas, 2007; Atashpaz Gargari *et al.*, 2008).

B. Genetic Algorithms (GAs)

Genetic algorithms consist of computational models developed by (Holland, 1992). Typically, the models are based on the concepts of natural or evolutionary biology. For a precise problem, the GA codes the candidate solution as a distinct chromosome. The method is initiated by a preliminary chromosome population that denotes the collective initial search points of the solution space problem. Next, the genetic operators including the selection, crossover and mutation, are imputed to generate new chromosomes. Since the operators are based on the code of "survival of the fittest, extinction of the most unfit", the general quality of the chromosomes is expected to progress with growing generation. The procedure is reiterated up to the conclusion criteria is

fulfilled and the highest chromosome of the final age is conveyed as the concluding answer (Holland and Goldberg, 1989).

C. Hybrid Proposed Imperialist Competitive Algorithm

A Multi-Objective Evolutionary Imperialist Competitive Algorithm (MOEICA) is the proposed algorithm used in this paper. The objective is to solve multi-objective MPS problems explicitly applied to the MPS of KRS. Before presenting the stepwise procedure of the proposed algorithm, the input information must be described. Likewise, the encoding and creation of a solution must be devised. The operators of crossover, mutation, assimilation, revolution and competition will also be demonstrated. To implement the MPS problem of the KRS, using the proposed algorithm, the details of the necessary steps are clarified as follows:

Input Information

For optimisation of KRS using master scheduling, the software considers numerous parameters that exist in the actual industrial settings:

- Numerical products depiction
- Numerical depiction of the duration of each period (diverse durations are acceptable)
- Preliminary (on-hand) inventories (quantity of products at the start of the planning horizon)
- Gross requests, i.e., essential product quantity, per period, expected from predicting the customer's orders
- Level of safety record per product and period
- Rate of Production - the quantity of product or resource manufactured per unit time
- Accessible volume per resource per period

The Objective Function

The master production schedule objective function for KRS has three objectives and these objectives are defined as:
Where, the objective function is intended to minimize the average of Ending Inventory level (EI), the Requirement Not Met (RNM) and the inventory Below Safety Stock (BSS).

Encoding

This algorithm begins by generating a set of random candidate solutions in the search space of the optimization problem. The generated random points are termed the initial population and consist of countries. The countries in this algorithm are the counterpart chromosomes in GA and particles in PSO, which are an array of candidate solutions. The algorithm initially begins with random countries.

The composition and shape of the countries in MPS problems are represented by a single vector structure. The country represents a set positive number. Each number represents the distribution of quantities of a given product within a specific period. Figure 3 presents the encoding of MPS for the four periods, which show the quantity of naphtha to be manufactured in period 1 in the first position, with the quantity of kerosene to be manufactured in period 1 in the second position and so forth. Thus, the initial solution is devised after the encoding to generate an initial solution.

Create Initial Population

With the right initial solution, the MOEICA algorithm achieves better results in less computer time, since there is less chance of wasting time by seeking solutions that are too far from the optimal. In this algorithm, we create an initial population using uniform random integer numbers.

Firstly, we generate a random value of crude oil quantity (CRp) for each period (p), from uniform distribution U [n*3927, n*4769.619], where n is the number of days in the period and [3927, 4769.619] represents the minimum and maximum daily amount of crude oil that can be extracted. Then randomly calculating the quantity of each product in each period, according to the crude oil quantity and using the ratios of each product in the model.

The algorithm starts with initial random countries of size. In the initial population, we select of the top countries, by means of the non-dominance technique, to determine the Pareto fronts and crowding distance technique (Fig. 4); to calculate the crowding distance for each country in each front in the population to be the imperialists and the remainder, forming the countries, were randomly allocated to different empires equally as colonies.

Searching Through the Solution Space

For each empire, the Genetic and Imperialist Competitive Operators of Crossover, Mutation, Assimilation and Revolution are exerted on colonies to diversely spread the population of imperialists.

Crossover. Crossover. In the implementation, the arithmetic crossover was used. This type of crossover creates a pair of offspring by combining two linear crossovers with two-parent country vectors.

Mutation. The mutation includes two operations, as shown in the below equations: adding and subtracting a random amount between the quantities of naphtha and Treated naphtha (Gasoline and LPG) in the first operation. The second operation represents adding and subtracting a random amount between the quantities of Kr and JF. The purpose of mutation is to maintain the diversity of the population in order to prevent too rapid convergence of the algorithm.

$$Na_p^{new} = Na_p^{old} \pm k, Gaso_p^{new} = Gaso_p^{old}$$
$$\mp 0.85 * k, LPG_p^{new} = LPG_p^{old} \mp 0.04 * k \tag{79}$$

Na$_1$	Kr$_1$	Di$_1$	Fo$_1$	Jf$_1$	Gaso$_1$	Lp$_1$	...	Na$_4$	Kr$_4$	Di$_4$	Fo$_4$	Jf$_4$	Gaso$_4$	Lp$_4$

Fig. 3: The problem representation of the MPS for four periods
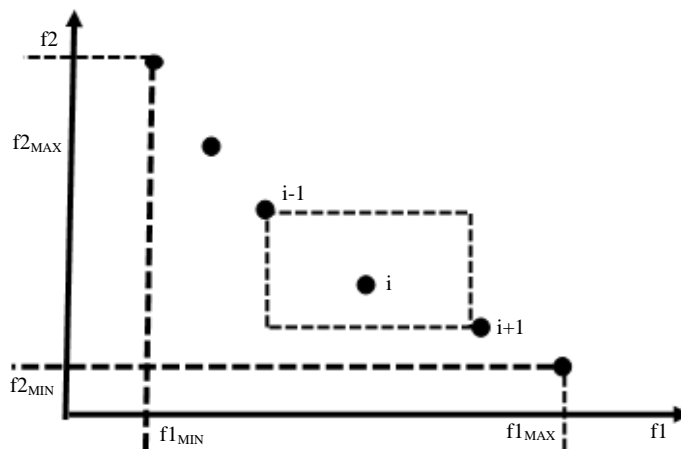

Fig. 4: Crowding distance

Assimilation. the imperialists' countries began to improve their colonies. This step is modelled by moving all the colonies toward the imperialist. Through this movement, some parts of a colony's structure will be similar to the empire's structure. The assimilation operator can be modelled as:

$$\{x\}_{new} = \{x\}_{old} + round\left(\beta * d * rand \otimes \{V_1\} + U(-1,1) * d * \{V_2\}\right) \tag{80}$$

Revolution Operator. In each decade (generation), certain numbers of countries undergo a sudden change, called a revolution. In our implementation, a random period (p) was selected to generate a random value of crude oil quantity (CRp) for the period (p) from uniform distribution U [n*3927, n*4769.619]. Subsequently, this was randomly calculated for the quantity of each product in this period, according to the crude oil quantity and using ratios of each product in the model.

Colonies Updating. In respective empires, the original population of the colonies are obtained by crossover, mutation, assimilating and revolution. These must be combined and the non-dominated based solutions imputed in the records. Based on the non-dominator and distant crowding practices, the top colonies must be designated with size NC (i), whereas the outstanding affiliates must be deleted. When the sum of the non-dominator answers is above the size of the record, Narch, the distant crowding must be computed. In addition, the solutions with more significant distant crowding must be selected by size Narch whereas the remainder is left out.

Archive updating. After updating the colonies, the non-dominated solutions must be imputed in the records. The recorded affiliates are similarly graded based on the non-dominating and distant crowding methods. Likewise, the answers of the initial front are earmarked, while others are left out.

Merge colonies and non-dominated sorting. After Subsequent to the revolution process, the novel population is combined. Next, the top colonies are designated as the primary population using updated archives. In adaptive archives, the selected populace is graded and designated by a non-dominant method. However, the distant crowding and the top population are designated the imperialists, whereas others are regarded as colonies. Evidently, higher-ranked nations are more influential than the toughest in the lesser ranks. More so, the nations with similar positions are likened with the metric of distant crowding.

Exchanging positions of the imperialist and a colony. As the colonies transition into imperialists and revolutions happens in selected nations, the prospects that a few colonies will attain elevated positions compared to their corresponding imperialists. In the event of this, the colony and its related imperialist will interchange places. Hence, the algorithms will proceed with the new nation as the imperialist.

Total Power of an Empire

$$T.C_n = T.C(imp_i) + \varepsilon * sum\{T.C(colonies\ of\ empire_n)\} \tag{81}$$

The total power of an empire is affected mainly by the power of the imperialist country. Although, the power of the colonies of an empire affects, albeit negligible, the total power of the empire. Therefore, the equation for the total cost is.

Where the total is the cost of the nth empire and is a positive number which is considered to be less than 1. Represents the total cost of the individual (imperialist or colony) which can be calculated using the following equations:

$$T.C_n = \sum_{k=1}^{K} Cost_{k,n} \tag{82}$$

$$Cost_{k,n} = \frac{\left| f_{k,n} - f_k^{\max} \right|}{\left| f_k^{\max} - f_k^{\min} \right|} \tag{83}$$

where, $f_{k,n}$ the value of objective function k for empire n. is $f_k^{\max}$, $f_k^{\min}$ are the maximum and minimum values of objective function k in each iteration, respectively.

Imperialistic Competition

The most crucial process in ICA is the imperialistic competition, in which all empires attempt to take over the colonies of other empires. The result being that weaker empires gradually lose their colonies to stronger ones. This process is modelled by selecting the weakest colony of the weakest empire and ceding it to the appropriate empire. This case is selected based on the competition between all empires. Figure 5 demonstrates a schematic of this process.

As observed in Fig. 5, empire 1 is denoted as the feeblest empire, whereby its individual colonies are under the process of competition. Hence, the empire from 2 to n is vying to possess its colony. At the onset of the race, the likelihood of possession is first computed by considering the entire influence of the territory. With knowledge of the likelihood of each empire's control, a comparable mechanism to the Roulette Wheel is adopted. This aims to avail the designated colony with an empire based on the probable proportionality.
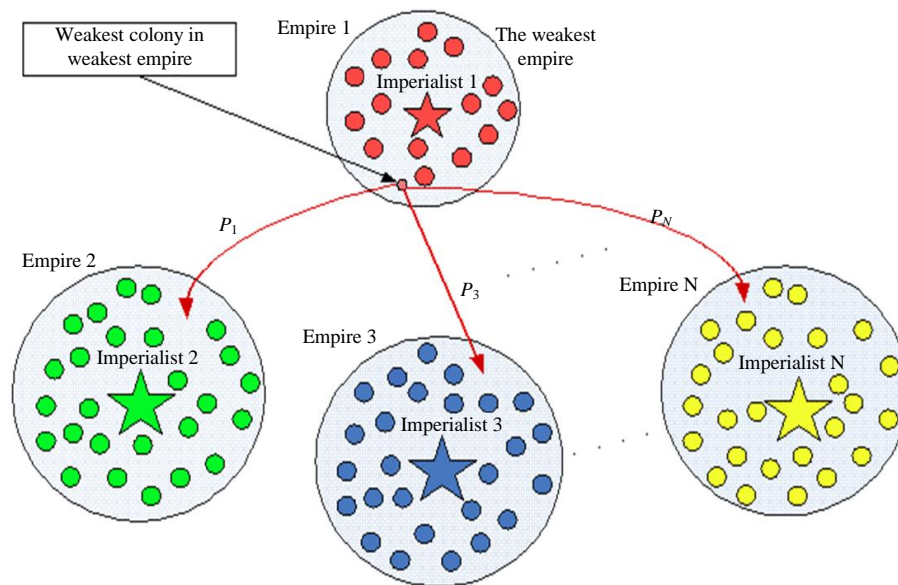
Fig. 5: Imperialistic competition

Eliminating the Powerless Empires

The powerless empires will flop during the competition of the imperialists. Hence, the resulting colonies will be dispersed between the other empires. In simulating the mechanism of collapse, numerous factors are defined to help ascertain the powerless empires. In this study, it is assumed that an empire flops when all its colonies are mislaid.

Stopping Criteria

The competition typically carries on pending when there is only one imperialist remaining in the search space. Conversely, diverse settings are designated as criteria for termination. These include achievement of a maximal sum of repetitions or minuscule progress in the functional objective.

The MOEICA Implementation Result of MPS -KRS Model

The outlined objectives are solved using the MOEICA algorithm. The achievement level of the objectives and crowding distance for some optimal solutions in the Pareto front ranked first are shown in Table 6.

The parameter settings of the MOEICA algorithm are described as follows: Crossover rate is 0.7, mutation rate is 0.4, assimilation coefficients are set to 0.01, revolutionary rates are set as 0.05, Nimp is equal to 10, population size is 100 and the maximum number of iterations is equal to 150 generations without any improvement.

Table 6: Some solutions and their crowding distance from pareto front

	EI	REN	BSS	Crowding distance
1	9512.1924	108.834870	635.30712	0.0008312
2	11013.7490	75.132270	291.57729	0.0008206
3	10559.8470	161.597080	247.40984	0.0002412
4	11523.5030	74.536700	219.23043	0.0002294
5	12804.7350	43.128912	155.81811	0.0001239
6	12287.5840	55.330671	175.27388	0.0001169
7	10062.5540	88.164676	476.19639	0.0001109
8	11829.0420	58.090170	225.39470	0.0002294
9	5392.0798	288.797620	2677.75490	0.0001239
10	10786.8000	62.267747	361.08370	0.0001169

Comparison between MOEICA, NSGAII and MOPSO

The feasibility of the proposed algorithm to solve the multi-objective MPS model of KRS was verified by NSGAII and MOPSO. These were also used to solve the problem using MATLAB 8.1 programming languages executed in the Intel Core 2 Duo 2.20 GHz. The results obtained for solving the MPS problem were compared with other algorithms. The parameter settings of the MOEICA algorithm are as previously described. However, the parametric settings of NSGAII and MOPSO are described in Table 7.

In contrast, the three performance metrics were used to examine the competence of the suggested algorithm. The selected metrics include; Mean Ideal Distance (MID), Quality Metric (QM) and Rate of Achievement to the objectives.

Table 7: The parameters values of NSGAII and MOPSO

Algorithm	Parameter	Value	Parameter	Value
NSGAII	N_{pop}	100.0	N_{gen}	150.0
	P_c	0.7	P_m	0.4
MOPSO	N_{pop}	100.0	N_{gen}	150.0
	C_1	1.0	C_2	2.0
	W	0.5	W_{damp}	
	$Alpha$	0.1	$Beta$	2.0
	$Gamma$	2.0	P_m	0.1
	N_{Grid}	5.0	N_{Rep}	

Table 8: The comparison results of measurement between MOEICA, NSGAII and MOPSO

Algorithm	QM	MID	RAS
MOEICA	1.00	813466.35	396.66
NSGAII	0.34	1441226.77	962.50
MOPSO	0.40	3185308.06	856.98

The outcomes of the three algorithms are represented in Table 8, according to the metric values computed for individual algorithms in the MOMPS KRS model. The results indicate that the optimal solutions of MOEICA are better compared to the other algorithms. Furthermore, the findings indicate that the superior algorithm based on performance is MOEICA as evident in the quality metric; QM = 1 (the higher the QM, the better the algorithm). The MOEICA algorithm also displayed the highest performance in terms of the mean ideal distance metric; MID = 813466.35 (lower is better). Furthermore, MOEICA exhibited a lower achievement rate compared to the simultaneously objective metric of MOPSO but higher than NSGAII and the RAS = 396.66 (lower is better). Moreover, MOEICA exhibited more relevant results in the other test problems based on the metrics when related to the different algorithms.

Conclusion

This paper proposes a hybrid approach called MOEICA consisting of the ICA and GA for solving multi-objective MPS problems. In addition, the performance of MOEICA was evaluated using the MOMPS model of KRS. The results showed that MOEICA could efficiently solve the multi-objective model of KRS based on the optimal Pareto points deduced in the study.

Future Works of this study may consider objective functions with actual cost rather than quantity units, such as cost of production, cost of inventory and failure to meet demand. In addition Installing a Master Production Scheduling (MPS) of other refineries in Iraq as Aldora refinery and Alqayara refinery taking into consideration the issue of design differences in those refineries.

Author's Contributions

All authors equally contributed in this work.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

Atashpaz-Gargari, E. and C. Lucas, 2007. Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition. Proceedings of the IEEE Congress on Evolutionary Computation, Sept. 25-28, IEEE Xplore Press, Singapore, pp: 4661-67.
DOI: 10.1109/CEC.2007.4425083

Atashpaz Gargari, E., F. Hashemzadeh, R. Rajabioun and C. Lucas, 2008. Colonial competitive algorithm: A novel approach for PID controller design in MIMO distillation column process. Int. J. Intell. Comput. Cybernet., 1: 337-55.
DOI: 10.1108/17563780810893446

Göthe-Lundgren, M., J.T. Lundgren and J.A. Persson, 2002. An optimization model for refinery production scheduling. Int. J. Product. Econom., 78: 255-70. DOI: 10.1016/S0925-5273(00)00162-6

Hartmanis, J., 1982. Computers and intractability: A guide to the theory of NP-completeness (Michael r. Garey and David s. Johnson). Siam Rev., 24: 90-91. DOI: 10.1137/1024022

Higgins, P. and J.I.M. Browne, 1992. Master production scheduling: A concurrent planning approach. Product. Plann. Control, 3: 2-18.
DOI: 10.1080/09537289208919369

Holland, J.H. and D. Goldberg, 1989. Genetic Algorithms in Search, Optimization and Machine Learning. 13th Edn., Addison-Wesley, Massachusetts, ISBN-10: 0201157675, pp: 412.

Holland, J.H., 1992. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence. 1st Edn., MIT Press,
ISBN-10: 0262581116, pp: 232.

Joly, M., L.F.L. Moro and J.M. Pinto, 2002. Planning and scheduling for petroleum refineries using mathematical programming. Brazilian J. Chem. Eng., 19: 207-28.
DOI: 10.1590/S0104-66322002000200008

Mitras, B.A. and J.A. Sultan, 2013. A novel hybrid imperialist competitive algorithm for global optimization. Australian J. Basic Applied Sci., 7: 330-41.

Proud, J.F., 1999. Master Scheduling: A Practical Guide to Competitive Manufacturing. 2rd Edo., John-Wile-Sons Inc., New York,
ISBN-10: 0471243221, pp: 610.

Sadiq, S.S., A.M. Abdulazeeza and H. Haron, 2018. Mathematical model of master production schedule for kalak refinery plant. Proceedings of the 17th International Conference on New Trends in Intelligent Software Methodologies, Tools and Techniques, (MTT' 18), IOS Press, pp: 303-332.

Slack, N., S. Chambers and R. Johnston, 2010. Operations management. Pearson education.

Soares, M.M. and G.E. Vieira, 2008. A new multi-objective optimization method for master production scheduling problems based on genetic algorithm. Int. J. Adv. Manufact. Technol., 41: 549-67. DOI: 10.1007/s00170-008-1481-x

Supriyanto, I. and B. Noche, 2011. Fuzzy multi-objective linear programming and simulation approach to the development of valid and realistic master production schedule. Logist. J., 7: 1-14. DOI: 10.2195/LJ_proc_supriyanto_de_201108_01

Tsoulos, I.G., 2009. Solving constrained optimization problems using a novel genetic algorithm. Applied Math. Comput., 208: 273-83. DOI: 10.1016/j.amc.2008.12.002

Vieira, G.E., F. Favaretto and P.C. Ribas, 2004. Comparing genetic algorithms and simulated annealing in master production scheduling problems. Proceeding of 17th International Conference on Production Research, (CPR' 04).

Vieira, G.E. and P.C. Ribas, 2004. A new multi-objective optimization method for master production scheduling problems using simulated annealing. Int. J. Product., 42: 4609-4622. DOI: 10.1080/00207540410001733869

Vollmann, T.E, W.L. Berry and D.C. Whybark, 1997. Manufacturing planning and control systems. Irwin, Boston.

Wu, Y., M. Liu and C. Wu, 2002. A genetic algorithm for optimizing the MPS of a processing-assembly production line with identical machines. Proceedings of the International Conference on Machine Learning and Cybernetics, Nov. 4-5, IEEE Xplore Press, Beijing, China, pp: 1172-77. DOI: 10.1109/ICMLC.2002.1167385