A Data-Driven Model for the Energy-Efficient No-Wait Flexible Flow Shop Scheduling Problem with Learning and Deteriorating Effects

Journal of Scientia Iranica

Danial Hatami, Department of Industrial Management, Qazvin Branch, Islamic Azad University, Qazvin, Iran; Email: d_hatami@qiau.ac.ir

Alireza Irajpour*, Department of Industrial Management, Qazvin Branch, Islamic Azad University, Qazvin, Iran; Email: Alireza.irajpour@iau.ac.ir

Reza Ehtesham Rasi, Department of Industrial Management, Qazvin Branch, Islamic Azad University, Qazvin, Iran; Email: rezaehteshamrasi@gmail.com

*Corresponding Author

S1. Literature reviewed summary

The summary of the literature review is presented in Table S.1.

S2. The main steps of the SARIMA algorithm

The main steps of the SARIMA algorithm are as (Alizadeh, Jalilzadeh, et al., 2023; Dubey et al., 2021):

- Identifying the initial ARIMA model: In this step, by analyzing the time series and checking the existing patterns, the initial ARIMA model is identified. The ARIMA model consists of three parts: the autoregressive part, which examines the effect of the previous series on the time series trend, the moving average part, which examines the effect of previous errors on the time series trend, and the integrated part which is applied to remove unpredictable summation from time series.
- Parameter estimation: In this step, ARIMA model parameters are estimated through estimation methods such as Least Squares method or Maximum Likelihood Estimation (MLE) method. These parameters include independent, dependent and aggregation parameters in the ARIMA model.
- Monsoon nature pattern detection: If there is a monsoon nature pattern in the time series, this step checks whether the monsoon nature pattern is reliably identified or not.
- Modeling of monsoon nature: If the pattern of monsoon nature is identified, the SAR (Seasonal Autoregressive) model is added to describe and predict this pattern. The SAR model is also similar to the ARIMA model, but it examines the effect of the monsoon pattern in time series.
- Estimation of parameters of monsoon nature: SAR model parameters are estimated by estimation methods such as least squares method or maximum likelihood method.
- Forecasting and evaluation: Using ARIMA and SARIMA models, it is possible to predict the values of time series for future times. Model performance is evaluated based on criteria such as mean absolute error (MAE) or mean square error (MSE).

The SARIMA algorithm, due to its consideration of seasonal and temporal variations in data, estimates the deterioration rate with appropriate accuracy. The use of this algorithm is justified by its high accuracy and its capability to account for various temporal conditions, enabling a highly precise estimation of the deterioration rate based on documented past data. This algorithm estimates the deterioration rate in a fuzzy manner, making it well-suited for use in methods that address uncertainty through fuzzy logic.

S3. The main steps of the SVR algorithm

The main steps of the SVR algorithm are as follows (Aghelpour et al., 2019; Nayeri et al., 2023): The data preprocessing step involves preparing and cleaning the data, including scaling, outlier removal, and transforming invalid data. Next, the SVR model is defined to predict continuous values by using a kernel function to find a hyperplane that separates the closest data points. Key model parameters, such as kernel and cost function parameters, are then selected to optimize performance. The model is trained using training data, optimizing the hyperplane for better predictions. After training, the model's accuracy is evaluated using metrics like MAE and MSE. Finally, the trained model is used to predict new data.

Estimating processing time is of high importance in scheduling problems. Therefore, the SVR algorithm has been used to estimate the processing time, taking into account various parameters such as season, operator gender, work shift, operator age, etc. This allows for a more accurate estimation of processing time to be used in the model.

S4. The flowcharts of the algorithms

The flowchart of metaheuristic algorithms is shown in Figures S.1 and S.2.

S5. Parameters setting

Setting the parameters is a crucial aspect of implementing metaheuristic algorithms. In this study, the Taguchi method is utilized to effectively tune the parameters of the metaheuristic algorithms. The Taguchi method provides a systematic approach for parameter optimization, enabling the identification of optimal parameter settings that result in improved algorithm performance. By employing the Taguchi method, the research aims to enhance the efficiency and effectiveness of the metaheuristic algorithms by selecting the most appropriate parameter values. In this regard, Table S.2 shows the selected values for the parameters of the metaheuristic algorithms. It should be noted that these values have been selected according to the research literature. After implementing the Taguchi method using Minitab software, the obtained signal-to-noise (S/N) charts are depicted in Figure S.3.

S6. The LSD chart for comparing the algorithms based on the RPD criterion

for the statistical validation, the Least Significant Deviation (LSD) chart is depicted for algorithms in terms of the solutions' quality (RPD criterion) in Figure S.4. Based on this figure, the developed L-GA-SA has significantly better performance in comparison with the other algorithms.

Table S.1. Literature reviewed summary

Author	Aim	Learning effect	deteriorating effects	Energy consumption	Data-driven	Uncertainty approach	Method
(Fu et al., 2018)	Scheduling of stochastic workshop flow considering the effect of deterioration and learning in Industry 4.0	*	*			---	fireworks algorithm
(Fu et al., 2019)	Bi-objective workshop flow scheduling considering deterioration rate	*	*			---	Bi-Population Evolutionary Algorithm
(Li et al., 2019)	Scheduling of workshop flow considering the effect of job learning	*				---	Hybrid Artificial Bee Colony Algorithm
(Tirkolaee et al., 2020)	Fuzzy mathematical programming to solve the workshop scheduling problem	*		*		Fuzzy approach	Self-Adaptive Artificial Fish Swarm
(Y. Wang & Xie, 2021)	Flexible workshop flow scheduling	*		*		Gray Approach (G-FFSP)	artificial bee colony
(Qin et al., 2022)	Workshop flow scheduling considering energy constraints			*	*	---	improved iterated greedy algorithm
(Zhao et al., 2022)	Blocked workshop flow scheduling			*		---	water wave algorithm
(Seyedhamzeh et al., 2022)	Scheduling the workshop flow of assembly lines in uncertain times	*		*		Robust optimization	Genetic algorithm
(Castaneda et al., 2022)	Workshop flow scheduling considering time and learning uncertainty	*				Fuzzy and stochastic approach	Simulation
(Zhang et al., 2023)	real-time in multi-objective evolutionary algorithms for flowshop problems		*		*	---	MOEAs – Decision Tree
(Mraihi et al, 2023)	Solving the flow shop scheduling problem by considering workers' flexibility and learning	*				---	Simulation

Author	Aim	Learning effect	deteriorating effects	Energy consumption	Data-driven	Uncertainty approach	Method
(Y.-J. Wang et al., 2023)	Workshop flow scheduling considering energy consumption uncertainty			*		Fuzzy approach	ENSGA-II algorithm
(Yu et al, 2024)	The development of a new hyperintelligence algorithm to solve the flow shop scheduling problem			*		---	MILP - KBIG
(Zhao et al, 2024)	Development of a model based on graph neural network to solve the combined flow shop problem	*			*	---	HGNN
This study	Scheduling of workshop flow considering the deterioration of machines and the effect of learning and energy consumption	*	*	*		Robust data-driven optimization	Hybrid SA-GA-PSO

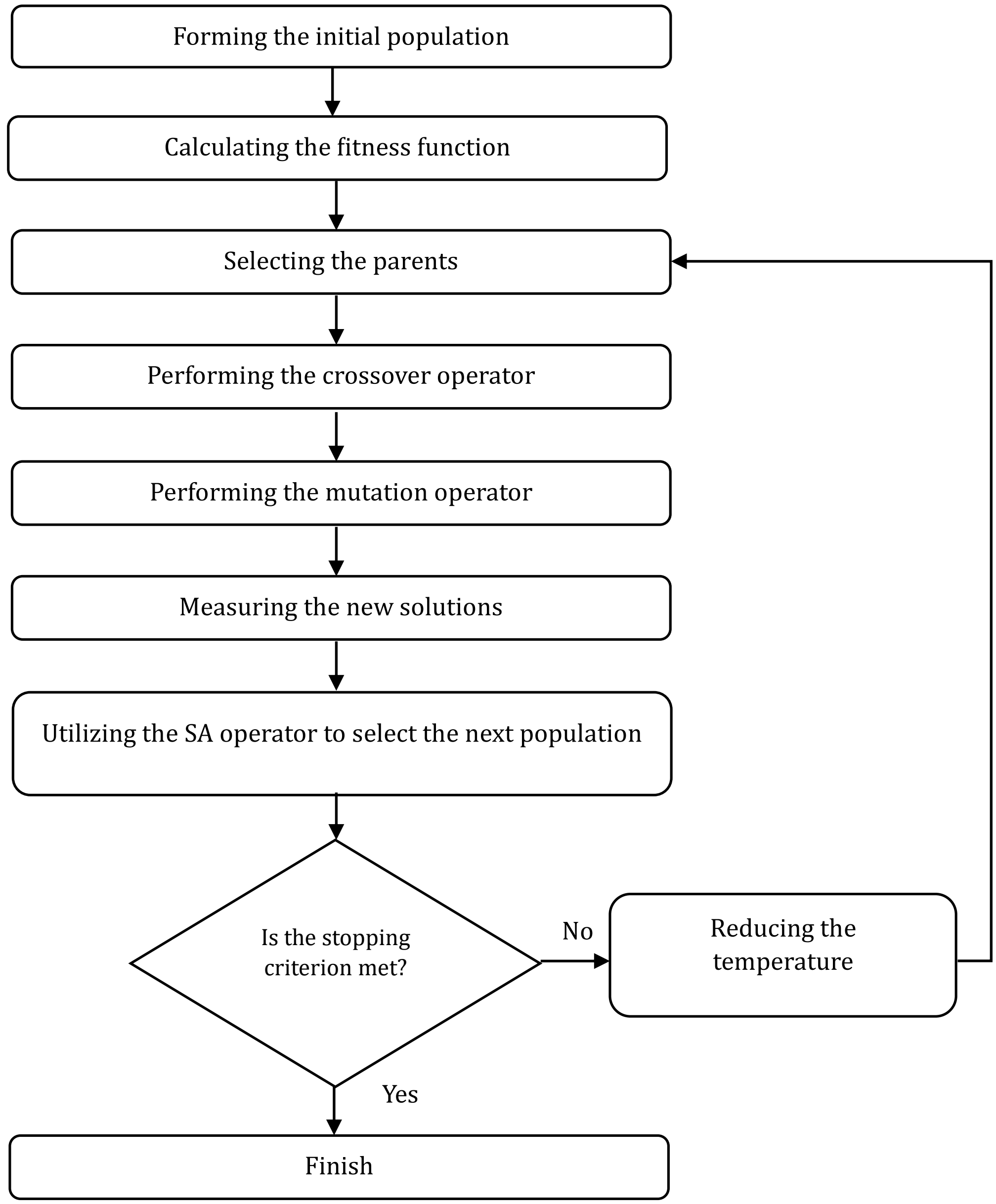


Figure S.1. The flowchart of GA-SA

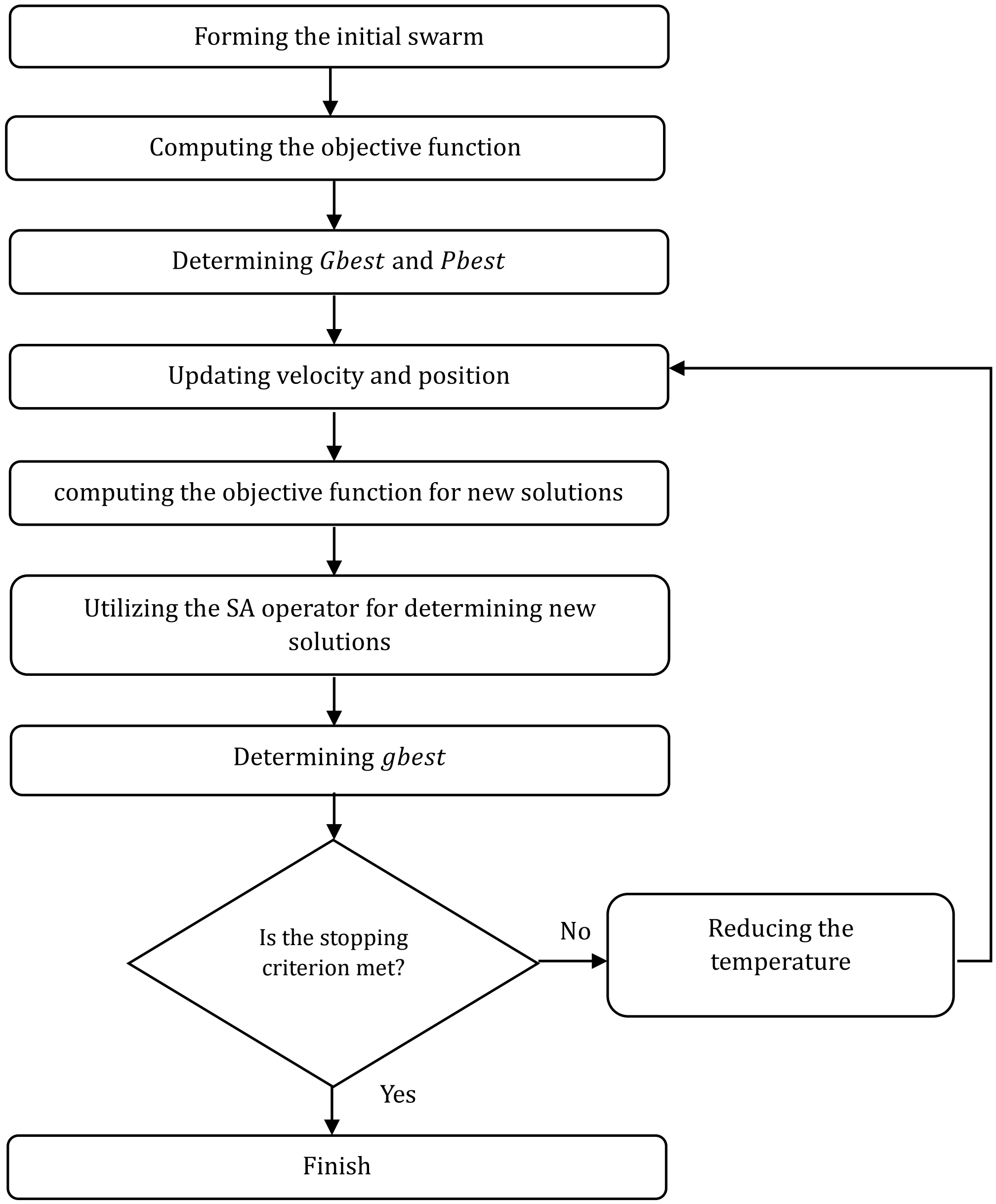
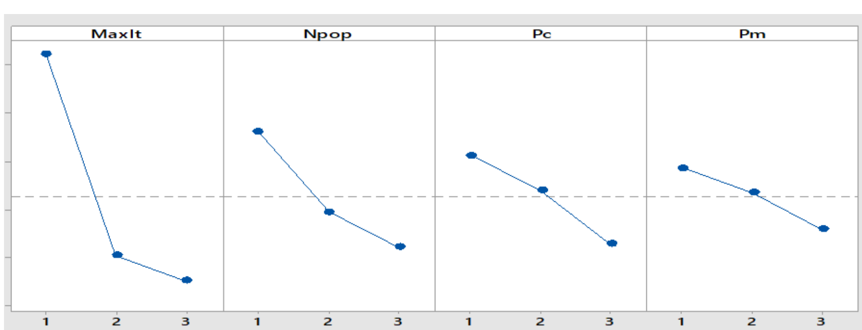


Figure S.2. The flowchart of PSO-SA

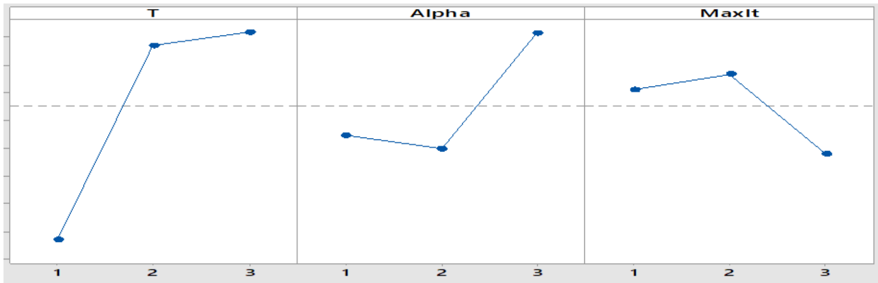
Table S.2. The values of the parameters of the metaheuristic algorithms at different levels

Algorithm	**Parameter**	**Level**		
		1	2	3
GA	$MaxIt$	300	350	400
	$Npop$	55	65	75
	Pc	0.6	0.7	0.8

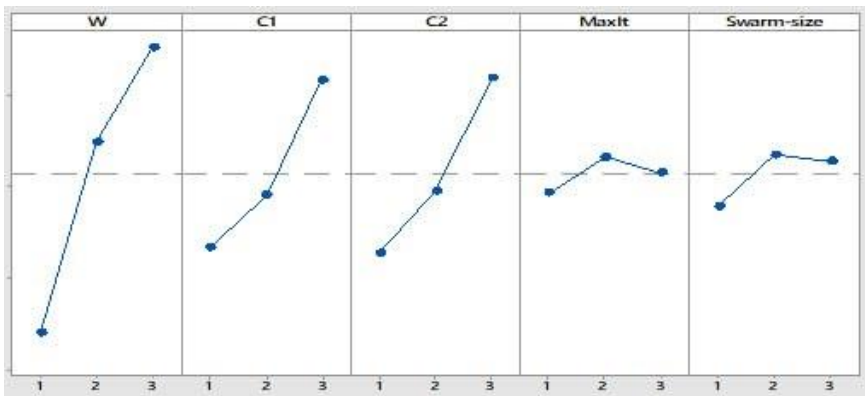
Algorithm	Parameter	Level		
		1	2	3
	Pm	0.1	0.2	0.3
PSO	$MaxIt$	300	350	400
	$Swam-Size$	55	65	75
	$c1$	2	2.1	2.2
	$c2$	2	2.1	2.2
SA	$MaxIt$	300	350	400
	T	35	40	45
	$Alpha$	0.9	0.95	0.99
GA-SA	$MaxIt$	300	350	400
	$Npop$	55	65	75
	Pc	0.6	0.7	0.8
	Pm	0.1	0.2	0.3
	T	35	40	45
	$Alpha$	0.9	0.95	0.99
PSO-SA	$MaxIt$	300	350	400
	$Swam-Size$	55	65	75
	$c1$	2	2.1	2.2
	$c2$	2	2.1	2.2
	T	35	40	45
	$Alpha$	0.9	0.95	0.99



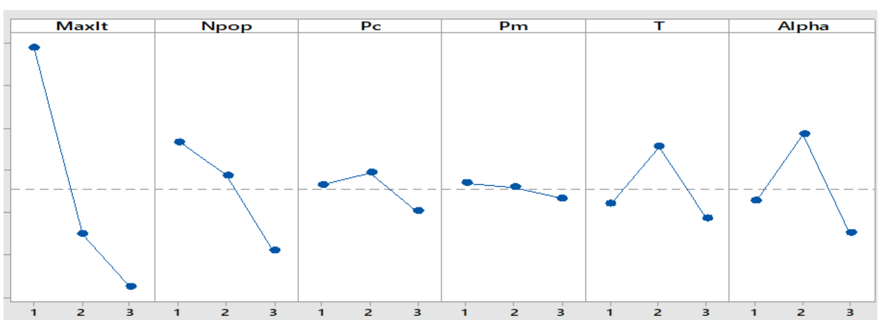
(a) GA



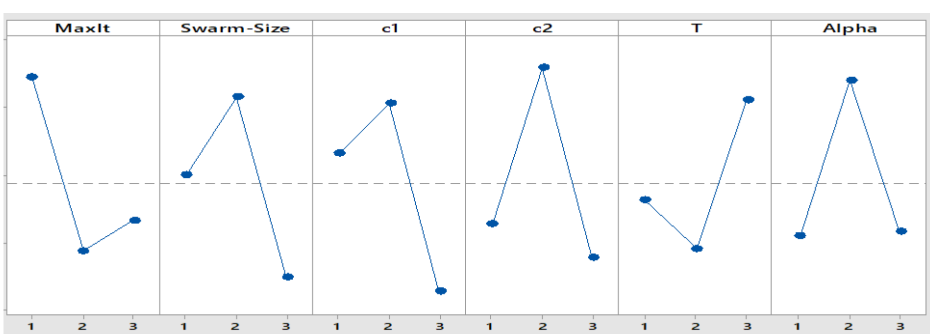
(b) SA



(c) PSO

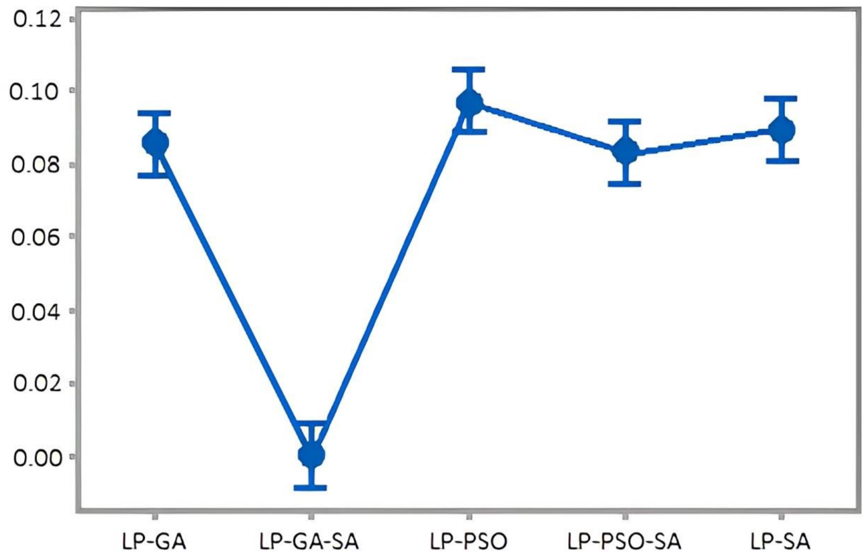


(d) GA-SA

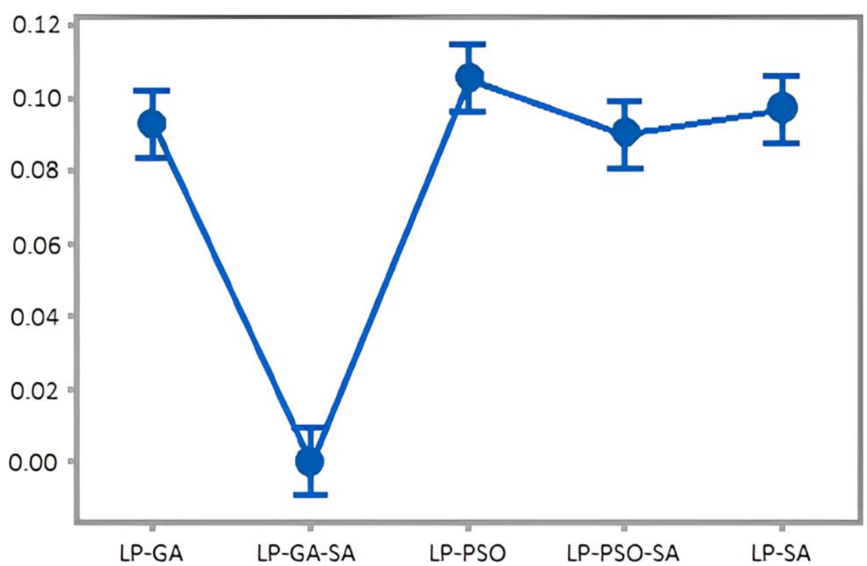


(e) PSO-SA

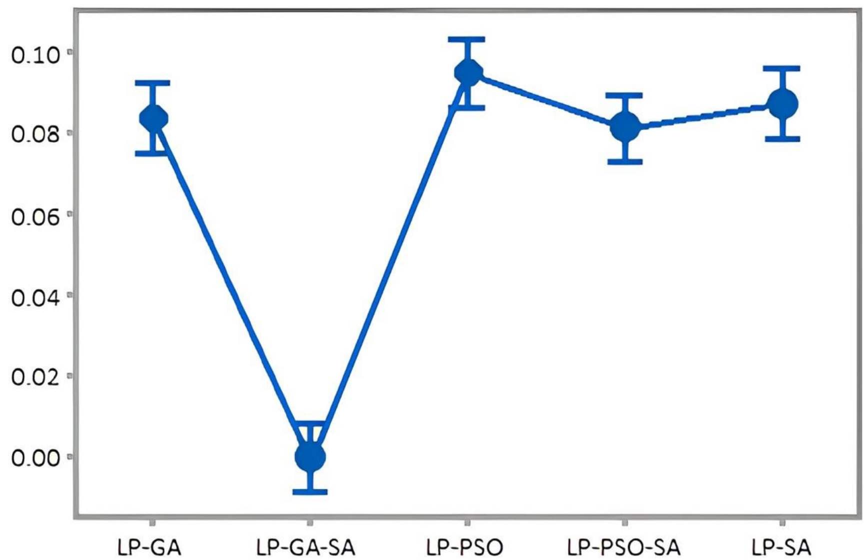
Figure S.3. The S/N charts for the algorithms


0.12
0.10
0.08
0.06
0.04
0.02
0.00
LP-GA
LP-GA-SA
LP-PSO
LP-PSO-SA
LP-SA

(a) RPD 1


0.12
0.10
0.08
0.06
0.04
0.02
0.00
LP-GA
LP-GA-SA
LP-PSO
LP-PSO-SA
LP-SA

(b) RPD 2



(c) RPD 3

Figure S.4. The LSD chart for comparing the algorithms based on the RPD criterion