

ORIGINAL ARTICLE

Single-agent Finite Impulse Response Optimizer vs Simulated Kalman Filter Optimizer

Tasiransurini Ab Rahman¹, Nor Azlina Ab. Aziz² and Nor Hidayati Abdul Aziz²

¹Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia, Johor, Malaysia.

²Faculty of Engineering and Technology, Multimedia University, Melaka, Malaysia.

ABSTRACT – Single-agent Finite Impulse Response Optimizer (SAFIRO) is a new estimation-based optimization algorithm which mimics the work procedure of the ultimate unbiased finite impulse response (UFIR) filter. In a real UFIR filter, the horizon length, N , plays an important role to obtain the optimal estimation. In SAFIRO, N represents the repetition number of estimation part that needs to be done in finding an optimal solution. On the other hand, Simulated Kalman Filter (SKF) is also an estimation-based optimization algorithm inspired by the estimation capability of Kalman filtering. In literature, substantial amount of works has been devoted to SKF, both in applied research and fundamental enhancements. Thus, in this paper, a performance comparison of both SAFIRO and SKF is presented. It is found that the SAFIRO outperforms the SKF significantly.

ARTICLE HISTORY

Received: 24 April 2019

Accepted: 4 July 2019

KEYWORDS

Optimization

SAFIRO

SKF

Introduction

Single-agent Finite Impulse Response Optimizer (SAFIRO) is a new estimation-based optimization algorithm which mimics the work procedure of the ultimate unbiased finite impulse response (UFIR) filter [1]. Introduced in 2018, the SAFIRO is the latest edition of estimation-based algorithms.

On the other hand, a metaheuristic algorithm called simulated Kalman filter (SKF), has been proposed in 2015 for numerical optimization problems [2-4]. It was introduced as population-based metaheuristics, where the search for optimal solution is conducted by a group of agents. The agents of SKF work like Kalman filters [5], where they go through prediction, measurement, and estimation process in every iteration. The measurement in SKF is a simulated measurement which is obtained using mathematical equation.

Many studies on SKF can be found in literature. For example, the SKF has been studied fundamentally [6-7]. The SKF also has been extended for binary optimization problems [8] and combinatorial optimization problems [9-11]. Hybridization of SKF with particle swarm optimization (PSO), gravitational search algorithm (GSA), and opposition-based learning [12-18] have also been proposed for better performance. Other variants called parameter-less SKF and randomized SKF algorithms were proposed in [19-20]. The SKF has also been applied for real world problems like the adaptive beamforming in

wireless cellular communication [21-24], airport gate allocation problem [25-26], feature selection of EEG signal [27-28], system identification [29-30], image processing [31-32], controller tuning [33], and PCB drill path optimization [34-35].

Given the popularity of SKF algorithm and the amount of studies reported on the fundamental improvements and application of SKF, the performance of the SAFIRO is compared with that of SKF algorithm based on CEC2014 benchmark dataset. In the subsequent section, the SKF algorithm is firstly introduced. Then, the SAFIRO is explained. Next, experimental procedure is presented, and the result of the experiment is shown. Finally, the result of a statistical analysis is shown to conclude the findings of this study.

The Simulated Kalman Filter

The SKF algorithm follows the algorithm shown in Fig. 1. One iteration consists of fitness evaluation, update the best solution, predict, measure, and estimate.

Using n agents, a set of solution can be denoted as $X(t) = \{X_1(t), X_2(t), \dots, X_n(t)\}$, where t is the iteration number. The SKF starts with random initialization of solutions. In each iteration, the fitness of the agents' are evaluated. Then, the agent with the best fitness value is identified as the best solution of the current population, $X_{\text{best}}(t)$. Next, the best $X_{\text{best}}(t)$ from the first iteration is selected as X_{true} .

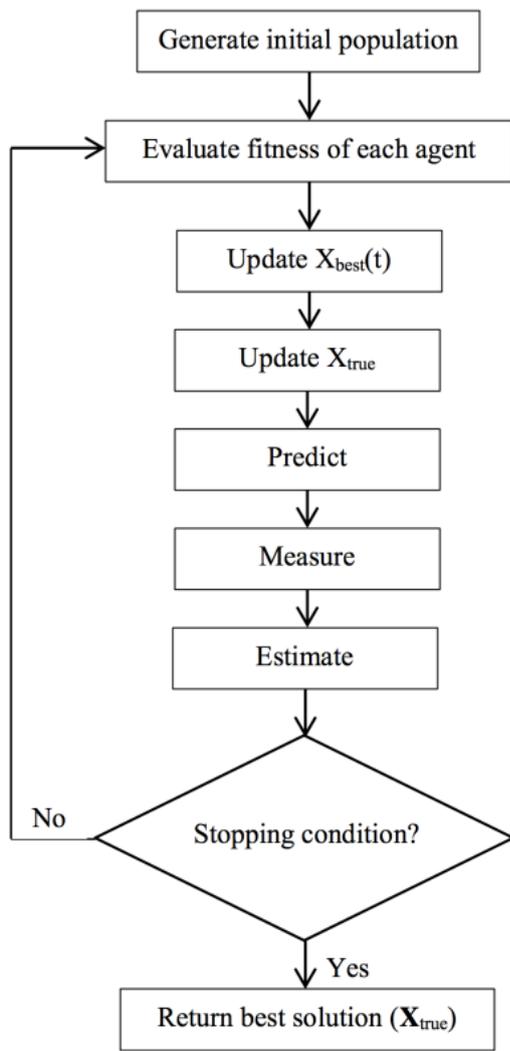


Figure 1. The flowchart of the SKF.

During the prediction phase, the current predicted state, $X_i(t|t+1)$, is assumed to be the estimated value:

$$X_i(t|t+1) = X_i(t) \tag{1}$$

The error covariant is also updated as follows:

$$P(t) = P(t) + Q \tag{2}$$

where $P(t)$ and $P(t|t+1)$ denote the current error covariant estimate and current transition error covariant estimate, respectively. Note that the error covariant estimate is influenced by the process noise, Q .

In SKF, measurements are simulated using an agent’s prediction and X_{true} . The dimensional wise calculation of measured value for each dimension of agent i th is calculated as follows:

$$Z_i(t) = X_i(t) + \sin(2\pi r_i(t)) \times |X_i(t) - X_{true}| \tag{3}$$

where $r_i(t)$ is a random value within the range of $[0,1]$. The estimation phase follows the measurement phase and the estimated next value is updated using (4):

$$X_i(t+1) = X_i(t|t+1) + K(t) \times (Z_i(t) - X_i(t|t+1)) \tag{4}$$

where $K(t)$ is the Kalman gain, which is calculated as follows:

$$K(t) = P(t|t+1)/(P(t|t+1)+R) \tag{5}$$

where R is the measurement noise. Then, the current error covariant estimate is updated in estimation phase using (6):

$$P(t+1) = (1-K(t)) \times P(t|t+1) \tag{6}$$

These steps continue until at the end of the iteration or at the end of the fitness evaluation.

The Single-agent Finite Impulse Response Optimizer (SAFIRO)

The SAFIRO is a single-agent metaheuristic algorithm, recently proposed for single-objective numerical optimization problems, which mimics the framework of the estimation process in UFIR filter [1]. Like the UFIR filter, SAFIRO’s agent responsible to perform the measurement and estimation stage to estimate the solution. The solution in SAFIRO represented by the estimation of the agent’s position.

The flowchart of SAFIRO is depicted in Fig. 2. The horizon length, N is the parameter that needs to be used in SAFIRO. As SAFIRO needs N measurements to begin the optimization process, the value of N is defined during the initialization stage. Then, by using a single agent, SAFIRO starts the process with the initialization of N measurements, $Y(t) = rand(U[X_{min}, X_{max}])$. After initialization, these N random initial measurements are evaluated by using the fitness function of the problem to determine the initial $X_best_so_far$. $X_best_so_far$ represents the best-so-far solution. For minimization problem, the initial measurement with the smallest fitness value is recorded as $X_best_so_far$, meanwhile, for the maximization problem, the initial measurement with the largest fitness value is recorded as $X_best_so_far$.

Next, the agent undergoes the measurement stage. In the UFIR filter operation, the measurement data can be obtained from the sensor. However, in SAFIRO, the measurement is simulated by random mutation of $X_best_so_far$ and shrinking local neighbourhood method. Each dimension of the problem to be optimized has a random value in the range of 0 to 1. The dimensions with a random value equal to or smaller than 0.5, keep the $X_best_so_far$ value as their measurement value, as in (7):

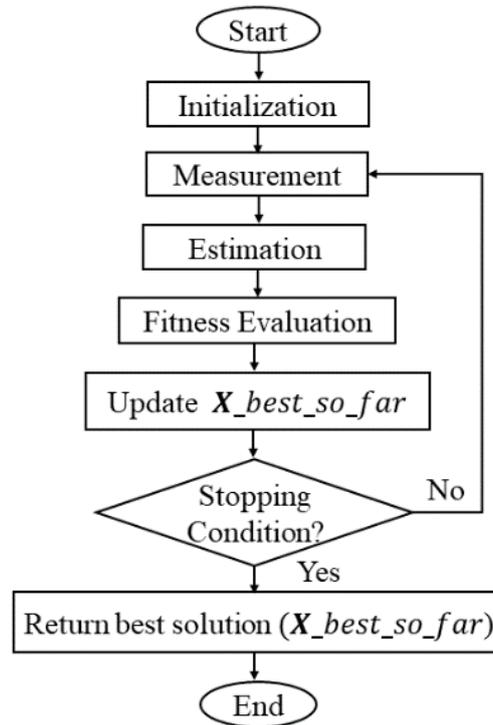


Figure 2. The flowchart of the SAFIRO.

$$Y_d(t) = X_best_so_far_d \quad (7)$$

On the other hand, the dimensions with a random value larger than 0.5, are chosen to be mutated to generate a new candidate solution. This mutation process is conducted in a local neighbourhood of $X_best_so_far$. The measurement value for these dimensions can be calculated as in (8).

$$Y_d(t) = X_best_so_far_d + rand(U[-\delta, \delta]) \quad (8)$$

This measurement process helps to encourage exploration through the mutation process, and at the same time creating a balance between the exploration and exploitation through the shrinking local neighbourhood. In the local search method, the search area is centred around $X_best_so_far$. The radius of the local neighbourhood, δ , can be computed by using as follows;

$$\delta = \exp(-\beta t)(T^{-1}) \times 0.5(X_{\max} - X_{\min}) \quad (9)$$

where t is the number of the current iteration, T is the number of maximum iteration, X_{\max} is the upper limit of search space, X_{\min} is the lower limit of search space,

and β is a coefficient value. This coefficient value controls the reduction speed of the neighbourhood's size.

After the measurement stage, SAFIRO's agent moves to the estimation stage. At this stage, the solution of SAFIRO is updated in a finite length according to the number of N . Each iteration, t , consists of sub-iteration, k . As depicted in Fig. 3, the first two points of the horizon are used for initial estimation, $\bar{X}(k=2)$. Initial estimation is generated randomly between [lower limit, upper limit] of the first and the second point of the horizon. Then, the remaining points are used for iterative estimation update from $\bar{X}(k=3)$ until $\bar{X}(k=N)$. The solution of initial estimation is improved iteratively during this part which is depending on the value of N . The number of repetition for iterative estimation is equal to $N-2$ (as mentioned above, the first two points are used to generate the initial estimation). In this study, the effect of the number of N towards SAFIRO's performance is observed. Iterative estimation can be computed by using (10) and (11):

$$\bar{X}(k) = \bar{X}(k-1) + K(k)(Y(t-N+k) - \bar{X}(k-1)) \quad (10)$$

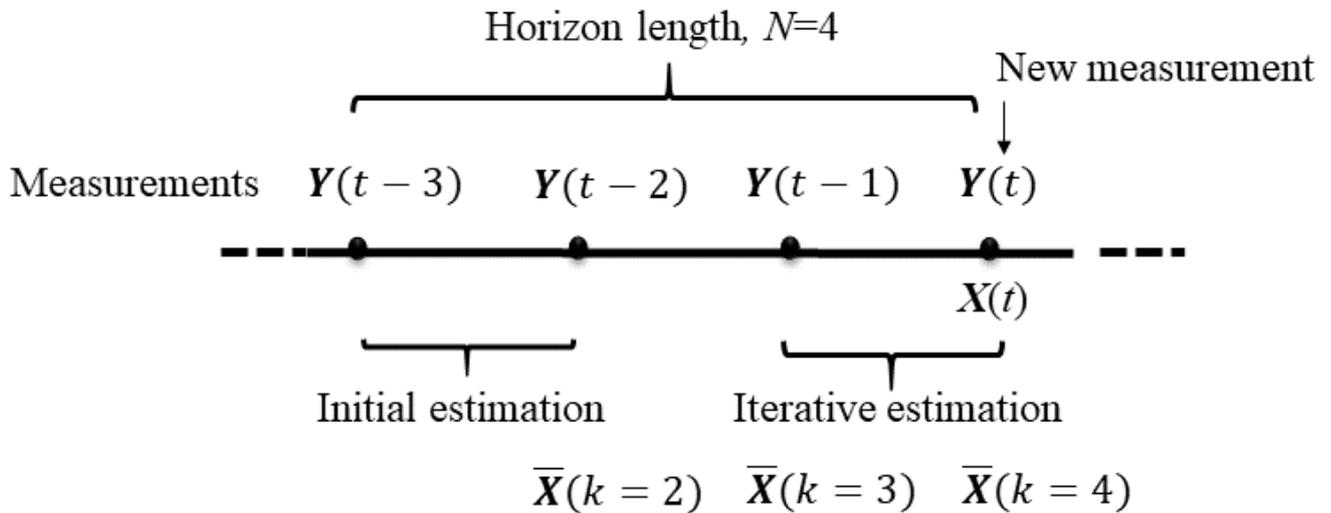


Figure 3. An illustration of an estimation process in SAFIRO ($N=4$).

$$K(k) = (k)^{-1} \quad (11)$$

where $\bar{X}(k)$ is the estimated solution for current point, meanwhile $\bar{X}(k-1)$ is the previous sub-iteration point. The measurement value, $Y(t-N+k)$ and the Kalman-like gain, $K(k)$ influence the improvement of the solution. The value of $K(k)$ can help to improve the estimation as the sub-iteration, k increases. The sub-iteration process end when $k=N$ is reached. The final value of k is then stored as $X(t)=\bar{X}(k)$. The agent’s updated solution for that corresponding iteration is represented by the estimated value of $X(t)$.

After obtaining the solution, $X(t)$, the evaluation stage is done to evaluate its fitness. The fitness of $X(t)$ is then compared to the fitness of $X_{best_so_far}$. The $X_{best_so_far}$ is updated when a better solution is found. For minimization problem, $X_{best_so_far}$ is updated if $fit(X(t)) < fit(X_{best_so_far})$, meanwhile for maximization problem, $X_{best_so_far}$ is updated if $fit(X(t)) > fit(X_{best_so_far})$. Measurement and estimation stages are repeated until the maximum iteration, T is met, and the $X_{best_so_far}$ returns as the solution to the given optimization problem.

Experiment

The CEC 2014 benchmark suite [36] is used for performance comparison. The experimental settings are as listed in Table 1. All parameter values listed in Table 1 follow the parameter setting in the CEC 2014 benchmark suite. The maximum number of function evaluation, $maxFES = 10,000 \text{ iterations} \times \text{dimension}$, D .

Regardless of the number of agents, a fair comparison of algorithms’ performance can be done by setting the same FES. To provide a fair evaluation for each comparison, the complexity of the problem is set as 50 dimensions, while the $maxFES$ is set as 500,000. The stopping condition is set to be the $maxFES$. All experiments are run 51 times on each test function. Hence, the evaluations are based on the mean fitness value over 51 runs time with 500,000 $maxFES$ of 50 problem dimensions. For all functions, the search space in the range of $[-100,100]$ is used for all dimensions.

For SKF, the value of the initial error covariance, $P(0)$, the measurement noise, R , and the process noise, Q are set as random $[0,1]$. For FIROs, the coefficient, β value is assigned as $\beta=10$ to allow moderate transition from the exploration phase to the exploitation phase. The coefficient, β , is a control parameter employed in the shrinking local neighborhood method of FIROs, which is computed during the measurement step. This parameter is used in the exponential decay equation to control the reduction of step-size.

Result and Discussion

The experimental results are tabulated in Table 2, Table 3, Table 4, and Table 5. Even though the SAFIRO outperforms the SKF in majority case studies of hybrid and composition functions, mix results are observed for unimodal and simple multimodal functions. To obtain a concrete conclusion, Wilcoxon signed rank test [37] was used.

Table 1. Experimental setting.

Parameter	Value
Search space	[-100,100]
No. of runs	51
No. of dimensions	50
No. of function evaluations	10,000 iterations×50 dimensions = 500,000 maxFES

Table 2. The mean fitness values of SAFIRO and SKF for unimodal functions.

Function	Ideal Fitness	SAFIRO	SKF
1	100	7.98E+05	1.18E+07
2	200	7695.60	2.35E+08
3	300	300.0000	1.90E+04

Table 3. The mean fitness values of SAFIRO and SKF for simple multimodal functions.

Function	Ideal Fitness	SAFIRO	SKF
4	400	488.72	569.32
5	500	520.00000	520.03
6	600	619.03	636.77
7	700	700.0100	702.87
8	800	994.29	818.99
9	900	1095.90	1086.43
10	1000	5785.20	1635.11
11	1100	6462.40	6810.65
12	1200	1200.10	1200.32
13	1300	1300.60	1300.58
14	1400	1400.500	1400.30
15	1500	1511.20	1627.16
16	1600	1620.60	1619.60

Table 4. The mean fitness values of SAFIRO and SKF for hybrid functions.

Function	Ideal Fitness	SAFIRO	SKF
17	1700	4.60E+04	1.44E+06
18	1800	3980.80	6.33E+07
19	1900	1920.60	1960.45
20	2000	2476.10	3.84E+04
21	2100	6.11E+04	2.30E+06
22	2200	2920.90	3447.67

Table 5. The mean fitness values of SAFIRO and SKF for composition functions.

Function	Ideal Fitness	SAFIRO	SKF
23	2300	2645.00	2656.15
24	2400	2678.70	2668.91
25	2500	2712.50	2733.69
26	2600	2778.70	2792.64
27	2700	3519.20	4003.78
28	2800	5252.50	7476.19
29	2900	2.89E+04	1.20E+05
30	3000	3.87E+04	2.91E+04

The Wilcoxon test usually is used when the population cannot be assumed to be normally distributed or it can be used to compare two related samples, matched samples, or repeated measurements on a single sample to assess whether their population mean ranks differ. Particularly, the null hypothesis for the test assumes that there is no significant difference between the mean error values of test algorithms while the alternative hypothesis tries to determine if there is a significant difference between test algorithms using 5% ($\alpha = 0.05$) significance level. Since the number of samples is 30, the critical value for the test is equal to 152. The sum of ranks where the SKF algorithm outperforms the SAFIRO is denoted as R^- while the sum of ranks where the BH algorithm is outperformed by the SAFIRO is denoted as R^+ . Hence, the SAFIRO is better than the SKF algorithm if $R^+ > R^-$ and the SAFIRO is significantly better than the SKF algorithm if R^- value is less than the critical value. Based on the mean accuracy of SAFIRO and SKF, R^+ and R^- obtained are 382 and 183, respectively. Hence, Wilcoxon test result shows that the SAFIRO algorithm is significantly better than the SKF algorithm.

Conclusions

The SKF is the most popular estimation-based optimization algorithm. On the other hand, the SAFIRO is the latest emerging estimation-based optimization algorithm. Since the original work of SAFIRO did not include the SKF in benchmarking, this paper brings these two algorithms to the same platform for a performance evaluation. It is found that for global optimization, the SAFIRO outperforms the SKF significantly.

References

- [1] Ab Rahman, T., Ibrahim, Z., Ab Aziz, N.A., Zhao, S., Abdul Aziz, N.H. (2018). Single-agent finite impulse response optimizer for numerical optimization problems. *IEEE Access*, vol. 6, pp. 9358-9374.
- [2] Ibrahim, Z., Abdul Aziz, N.H., Ab. Aziz, N.A., Razali, R., and Mohamad, M.S. (2016). Simulated Kalman filter: a novel estimation-based metaheuristic optimization algorithm. *Advanced Science Letters*, vol. 22, pp. 2941-2946.
- [3] Abd Aziz, N.H., Ibrahim, Z., Razali, S., and Ab. Aziz, N.A. (2016) Estimation-based metaheuristics: a new branch of computational intelligence. *The National Conference for Postgraduate Research*, pp. 469-476.
- [4] Ibrahim, Z., Abdul Aziz, N.H., Ab Aziz, N.A., Razali, S., Shapiai, M.I., Nawawi, S.W., and Mohamad, M.S. (2015) A Kalman filter approach for solving unimodal optimization problems. *ICIC Express Letters*, vol. 9, pp. 3415-3422.
- [5] Kalman, R.E. (1960) A new approach to linear filtering and prediction problems. *ASME Journal of Basic Engineering*, vol. 82, pp. 35-45.
- [6] Abd Aziz, N.H., Ibrahim, Z., Razali, S., Bakare, T.A., and Ab. Aziz, N.A. (2016). How important the error covariance in simulated Kalman filter?. *The National Conference for Postgraduate Research*, pp. 315-320.
- [7] Abd Aziz, N.H., Ab. Aziz, N.A., Mat Jusof, M.F., Razali, S., Ibrahim, Z., Adam, A., and Shapiai, M.I. (2018). An analysis on the number of agents towards the performance of the simulated Kalman filter optimizer. *8th International Conference on Intelligent Systems, Modelling and Simulation*, pp. 16-21.
- [8] Md Yusof, Z., Ibrahim, I., Satiman, S.N., Ibrahim, Z., Abd Aziz, N.H., and Ab. Aziz, N.A. (2015). BSKF: binary simulated Kalman filter. *Third International Conference on Artificial Intelligence, Modelling and Simulation*, pp. 77-81.
- [9] Md Yusof, Z., Ibrahim, I., Ibrahim, Z., Abas, K.H., Ab. Aziz, N.A., Abd Aziz, N.H., and Mohamad, M.S. (2016). Local optimum distance evaluated simulated Kalman filter for combinatorial optimization problems. *The National Conference for Postgraduate Research*, pp. 892-901.
- [10] Md Yusof, Z., Ibrahim, I., Ibrahim, Z., Mohd Azmi, K.Z., Ab. Aziz, N.A., Abd Aziz, N.H., and Mohamad, M.S. (2016). Distance evaluated simulated Kalman filter for combinatorial optimization problems. *ARNP Journal of Engineering and Applied Sciences*, vol. 11, pp. 4904-4910.
- [11] Md Yusof, Z., Ibrahim, I., Ibrahim, Z., Mohd Azmi, K.Z., Ab. Aziz, N.A., Abd Aziz, N.H., and Mohamad, M.S. (2016). Angle modulated simulated Kalman filter algorithm for combinatorial optimization problems. *ARNP Journal of Engineering and Applied Sciences*, vol. 11, pp. 4854-4859.
- [12] Muhammad, B., Ibrahim, Z., Mat Jusof, M.F., Ab. Aziz, N.A., Abd Aziz, N.H., and Mokhtar, N. (2017). A hybrid

- simulated Kalman filter - gravitational search algorithm (SKF-GSA). *International Conference on Artificial Life and Robotics*, pp. 707–710.
- [13] Muhammad, B., Ibrahim, I., Mohd Azmi, K.Z., Abas, K.H., Ab. Aziz, N.A., Abd Aziz, N.H., and Mohamad, M.S. (2016). Performance evaluation of hybrid SKF algorithms: hybrid SKF-PSO and hybrid SKF-GSA. *The National Conference for Postgraduate Research*, pp. 865-874.
- [14] Muhammad, B., Ibrahim, Z., Mohd Azmi, K.Z., Abas, K.H., Ab. Aziz, N.A., Abd Aziz, N.H., and Mohamad, M.S. (2016). Four different methods to hybrid simulated Kalman filter (SKF) with particle swarm optimization (PSO). *The National Conference for Postgraduate Research*, pp. 843-853.
- [15] Muhammad, B., Ibrahim, Z., Mohd Azmi, K.Z., Abas, K.H., Ab. Aziz, N.A., Abd Aziz, N.H., and Mohamad, M.S. (2016). Four different methods to hybrid simulated Kalman filter (SKF) with gravitational search algorithm (GSA). *The National Conference for Postgraduate Research*, pp. 854-864.
- [16] Muhammad, B., Ibrahim, Z., Ghazali, K.H., Mohd Azmi, K.Z., Ab. Aziz, N.A., Abd Aziz, N.H., and Mohamad, M.S. (2015). A new hybrid simulated Kalman filter and particle swarm optimization for continuous numerical optimization problems. *ARPN Journal of Engineering and Applied Sciences*, vol. 10, pp. 17171-17176.
- [17] Ibrahim, Z., Mohd Azmi, K.Z., Ab. Aziz, N.A., Abd Aziz, N.H., Muhammad, B., Mat Jusof, M.F., and Shapiai, M.I. (2018). An oppositional learning prediction operator for simulated Kalman filter. *The 3rd International Conference on Computational Intelligence and Applications*, pp. 139-143.
- [18] Mohd Azmi, K.Z., Ibrahim, Z., Pebrianti, D., Mat Jusof, M.F., Abdul Aziz, N.H., Ab. Aziz, N.A. (2019). Enhancing simulated Kalman filter algorithm using current optimum opposition-based learning. *Mekatronika*, vol. 1, pp. 1-13.
- [19] Abd Aziz, N.H., Ibrahim, Z., Ab. Aziz, N.A., and Razali, S. (2017). Parameter-less simulated Kalman filter. *International Journal of Software Engineering and Computer Systems*, vol. 3, pp. 129-137.
- [20] Abd Aziz, N.H., Ab. Aziz, N.A., Ibrahim, Z., Razali, S., Mat Jusof, M.F., Abas, K.H., Mohamad, M.S., and Mokhtar, N. (2017). Simulated Kalman filter with randomized Q and R parameters. *International Conference on Artificial Life and Robotics*, pp. 711-714.
- [21] Lazarus, K., Noordin, N.H., Mat Jusof, M.F., Ibrahim, Z., and Abas, K.H. (2017). Adaptive beamforming algorithm based on a simulated Kalman filter. *International Journal of Simulation: Systems, Science and Technology*, vol. 18, pp. 10.1-10.5.
- [22] Lazarus, K., Noordin, N.H., Mohd Azmi, K.Z., Abd Aziz, K.Z., and Ibrahim, Z. (2016). Adaptive beamforming algorithm based on generalized opposition-based simulated Kalman filter. *The National Conference for Postgraduate Research*, pp. 1-9.
- [23] Lazarus, K., Noordin, N.H., Ibrahim, Z., Mat Jusof, M.F., Mohd Faudzi, M.A., Subari, N., and Mohd Azmi, K.Z. (2017). An opposition-based simulated Kalman filter algorithm for adaptive beamforming. *IEEE International Conference on Applied System Innovation*, pp. 91-94.
- [24] Lazarus, K., Noordin, N.H., Ibrahim, Z., and Abas, K.H. (2016). Adaptive beamforming algorithm based on simulated Kalman filter. *Asia Multi Conference on Modelling and Simulation*, pp. 19-23.
- [25] Md Yusof, Z., Satiman, S.N., Mohd Azmi, K.Z., Muhammad, B., Razali, S., Ibrahim, Z., Aspar, Z., and Ismail, S. (2015). Solving airport gate allocation problem using simulated Kalman filter. *International Conference on Knowledge Transfer*, pp. 121-127.
- [26] Mohd Azmi, K.Z., Md Yusof, Z., Satiman, S.N., Muhammad, B., Razali, S., Ibrahim, Z., Ab. Aziz, N.A., and Abd Aziz, N.H. (2016). Solving airport gate allocation problem using angle modulated simulated Kalman filter. *The National Conference for Postgraduate Research*, pp. 875-885.
- [27] Muhammad, B., Mat Jusof, M.F., Shapiai, M.I., Adam, A., Md Yusof, Z., Mohd Azmi, K.Z., Abdul Aziz, N.H., Ibrahim, Z., and Mokhtar, N. (2018). Feature selection using binary simulated Kalman filter for peak classification of EEG signals. *2018 8th International Conference on Intelligent Systems, Modelling and Simulation*, pp. 1-6.
- [28] Adam, A., Ibrahim, Z., Mokhtar, N., Shapiai, M.I., Mubin, M., and Saad, I. (2016). Feature selection using angle modulated simulated Kalman filter for peak classification of EEG signals. *SpringerPlus*, vol. 5, 1580.
- [29] Muhammad, B., Mohd Azmi, K.Z., Ibrahim, Z., Mohd Faudzi, A.A., and Pebrianti, D. (2018). Simultaneous computation of model order and parameter estimation for system identification based on opposition-based simulated Kalman filter. *SICE International Symposium on Control Systems 2018*, pp. 105-112.
- [30] Mohd Azmi, K.Z., Ibrahim, Z., Pebrianti, D., and Mohamad, M.S. (2017). Simultaneous computation of model order and parameter estimation for ARX model based on single and multi swarm simulated Kalman filter. *Journal of Telecommunication, Electronic, and Computer Engineering*, vol. 9, pp. 151-155.
- [31] Ann, N.Q., Pebrianti, D., Bayuaji, L., Daud, M.R., Samad, R., Ibrahim, Z., Hamid, R., and Syafrullah, M. (2018). SKF-based image template matching for distance measurement by using stereo vision. *Intelligent Manufacturing and Mechatronics*, pp. 439-447.
- [32] Ann, N.Q., Pebrianti, D., Ibrahim, Z., Mat Jusof, M.F., Bayuaji, L., and Abdullah, N.R.H. (2018). Illumination-invariant image matching based on simulated Kalman filter (SKF). *Journal of Telecommunication, Electronics and Computer Engineering*, vol. 10, pp. 31-36.
- [33] Muhammad, B., Pebrianti, D., Abdul Ghani, N., Abd Aziz, N.H., Ab. Aziz, N.A., Mohamad, M.S., Shapiai, M.I., and Ibrahim, Z. (2018). An application of simulated Kalman filter optimization algorithm for parameter tuning in proportional-integral-derivative controllers for automatic voltage regulator system. *SICE International Symposium on Control Systems 2018*, pp. 113-120.
- [34] Abd Aziz, N.H., Ab. Aziz, N.A., Ibrahim, Z., Razali, S., Abas, K.H., and Mohamad, M.S. (2016). A Kalman filter approach to PCB drill path optimization problem. *IEEE Conference on Systems, Process and Control*, pp. 33-36.
- [35] Abd Aziz, N.H., Ibrahim, Z., Ab. Aziz, N.A., Md Yusof, Z., and Mohamad, M.S. (2018). Single-solution simulated Kalman filter algorithm for routing in printed circuit board drilling process. *Intelligent Manufacturing and Mechatronics*, pp. 649-655.
- [36] Liang, J.J., Qu, B.Y., Suganthan, P.N. (2013). Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization. *Tech. Rep. 201311. Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Nanyang Technological University, Singapore.*

- [37] García, S., Molina, D., Lozano, M., Herrera, F. (2008). A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the CEC 2005 special session on real parameter optimization. *Journal of Heuristics*, vol. 15, pp. 617-644.