

## ORIGINAL ARTICLE

## Title of the Paper (Use Title Style Arial 16pt)

J. Doe<sup>1</sup>, W. Mellon<sup>1\*</sup>, H. Potter<sup>2</sup>, T. Stark<sup>2</sup>, J. Clouseau<sup>3</sup> and G. Focker<sup>3</sup>

<sup>1</sup>Faculty of Manufacturing Engineering, Universiti Malaysia Pahang, 26600 Pahang, Malaysia.

<sup>2</sup>Faculty of Electrical and Electronics Engineering, Universiti Malaysia Pahang, 26600 Pahang, Malaysia.

<sup>3</sup>Faculty of Engineering and Technology, Multimedia University, 75450 Melaka, Malaysia.

**ABSTRACT** – English - The abstract should be written using the Normal style with Arial Narrow font, size 10. 100-200 words. The abstract should state briefly the purpose of the research; the methods applied, the major findings and conclusions, and must be able to stand alone. There should not be any in-text citation in the abstract. Manuscripts written in Malay must include an abstract written in English.

**ABSTRAK** Bahasa Melayu – Artikel BM perlu disertakan dengan terjemahan abstrak dalam Bahasa Inggeris. The abstract should be written using the Normal style with Arial Narrow font, size 10. 100-200 patah perkataan, merangkumi objektif kajian, metodologi, dapatan utama dan kesimpulan, tanpa sebarang sitasi. tempus.

**KEYWORDS** Sertakan 3-5 perkataan kata kunci menggambarkan kajian anda.

### ARTICLE HISTORY

Received: xxxx

Revised: xxxx

Accepted: xxxx

### KEYWORDS

Keyword 1

Keyword 2

Keyword 3

Keyword 4

Keyword 5

### PENDAHULUAN / INTRODUCTION (ARIAL, 11PT)

The introduction should state the objectives of the work, and an adequate background that includes a review of relevant literature. Review of literature may be divided into subsections with appropriate headings. (Times New Roman, 10pt)

The main goal of an optimization problem is to obtain the best combination of variables of a fitness function such that the value of the fitness is maximum or minimum. This can be done effectively by using a population-based optimization algorithm. A new population-based optimization algorithm termed as simulated Kalman filter (SKF) is inspired by the estimation capability of Kalman filter [1]. Designed from the procedure of Kalman filtering, which incorporates prediction, measurement, and estimation, the global minimum or maximum can be estimated. Measurement process, which is needed in Kalman filtering, is mathematically modelled and simulated. Agents interact with each other to update and optimize the solution during the search process.

The concept of opposition-based learning (OBL) can be used to improve the performance of population-based optimization algorithm [2]. The important idea behind the OBL is the concurrent consideration of an estimate and its corresponding opposite estimate which is closer to the global optimum. OBL was initially implemented to improve learning and back propagation in neural networks [3], and until now, it has been employed in various optimization algorithms, such as differential evolution, particle swarm optimization and ant colony optimization [4].

In this research, inspired by the concept of current optimum opposition-based learning (COOBL), we propose a modified SKF which is called as current optimum opposition-based simulated Kalman filter (COOBSKF) to enhance the performance of SKF. From the SKF perspective, this is the first attempt to improve its performance through COOBL strategy. The COOBSKF compares the fitness of an individual to its opposite and maintain the fitter one in the population. Experimental results show that the proposed algorithm can achieve better solution quality.

The remainder of this paper is organized as follows: Section 2 briefly presents an overview of optimization algorithms and opposition-based learning application. Section 3 explains the standard simulated Kalman filter algorithm, the concept of opposition-based learning and the proposed enhance version of SKF. Section 4 provides the experimental settings and discusses the experimental results. Section 5 concludes the paper.

### METODOLOGI KAJIAN / MATERIAL AND METHODS

The materials used and methods applied in the research should be sufficiently and explicitly explained

This part provides a brief overview of optimization algorithms followed by the application of OBL in optimization algorithms. Some of optimization algorithms are based on population-based where the search process is performed with multiple agents. One example of population-based optimization algorithm is particle swarm optimization (PSO). In PSO, a swarm of agent searches for the global optimum solution by velocity and position updates, which are depending on current position of agent, personal best, and global best of the swarm. They move towards those particles which have better fitness values and finally attain the best solution.

Another population-based optimization algorithm is gravitational search algorithm (GSA). GSA was designed according to the Newtonian gravity law and mass interactions. In the algorithm, agents and their performance is evaluated by their masses which rely on fitness function values. The location of each agent in the search space indicates a problem solution. The heaviest mass is the optimum solution in the search space and by lapse of time, masses are attracted by the heaviest mass and converged to the better solution.

The concept of opposition-based learning is applicable to a wide range of optimization algorithms. Even though the proposed approach is originally embedded in differential evolution (DE), it is universal enough to be employed in other optimization algorithms. In [5], the OBL has been used to accelerate the convergence rate of DE. The proposed opposition-based DE (ODE) implements the OBL at population initialization and also for generation jumping. Besides that, a comprehensive investigation was conducted by using 58 benchmark functions with a purpose to analyze the effectiveness of ODE. Various sets of experiments are performed separately to examine the influence of opposite points, dimensionality, population size and jumping rates on the ODE algorithm.

Opposition-based differential evolution using the current optimum (COODE) was introduced for function optimization. In the COODE, the optimum agent in the current population is dynamically functioned as the symmetry point between an estimate and its respective opposite estimate. The distance between opposite numbers and the global optimum is short enough to maintain a significant rate of applying OBL throughout the search process.

Opposition-based particle swarm optimization (OPSO) is proposed by employing OBL to population initialization, generation jumping and the swarm's best particle. Initially, swarms are initialized with random velocities and positions. The opposite swarm is determined by calculating the opposite of velocity and position, and then the fittest of swarm and opposite swarm is chosen as the next population. The similar approach is used in current generations by applying jumping rate and dynamic constriction factor, which is used to improve the convergence rate.

In other report, the OBL technique has been used to enhance the quality of solutions and convergence rate of an ant colony system (ACS). Five versions of implementing opposition idea have been proposed to extend the solution construction phase of ACS, known as, free opposition, free quasi-opposition, synchronous opposition, opposite pheromone per node (OPN) and opposite pheromone per edge (OPE). Results of these algorithms on TSP problems indicate that only OPN technique shows significant improvement.

## DAPATAN KAJIAN / RESEARCH FINDING

### Sub topik

The simulated Kalman filter (SKF) algorithm is shown in Figure 1. The algorithm started with initialization of  $n$  agents, in which the positions of each agent are initialized randomly in the search space. The maximum number of iterations,  $t_{max}$ , is defined as the stopping condition for the algorithm. The initial value of error covariance estimate,  $P(0)$ , the process noise value,  $Q$ , and the measurement noise value,  $R$ , which are needed in Kalman filtering, are also determined during initialization stage. After that, each agent is subjected to fitness evaluation to generate initial solutions. The fitness values are checked and the agent having the best fitness value at every iteration,  $t$ , is recorded as  $X_{best}(t)$ . For function minimization problem,

$$X_{best}(t) = fit_i(X(t)) \tag{1}$$

and for function maximization problem,

$$X_{best}(t) = fit_i(X(t)) \tag{2}$$

The best so far solution in SKF is named as  $X_{true}$ . The  $X_{true}$  is updated only if the  $X_{best}(t)$  is better ( $X_{best}(t) < X_{true}$  for minimization problem, or  $X_{best}(t) > X_{true}$  for maximization problem) than the  $X_{true}$ . The subsequent computations are basically identical to the prediction, measurement and estimation procedures in Kalman filter. In the prediction stage, the following time-update equations are calculated:

$$X_i(t|t) = X_i(t) \tag{3}$$

$$P(t|t) = P(t) + Q \tag{4}$$

where  $X_i(t)$  and  $X_i(t|t)$  are the previous state and predicted state, respectively, and  $P(t)$  and  $P(t|t)$  are previous error covariant estimate and predicted error covariant estimate, respectively. Note that the error covariant estimate is influenced by the process noise,  $Q$ .

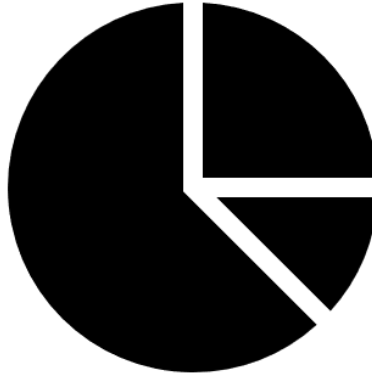


Figure 1. Figure 1 Caption.

**Sub Topik**

The concept of Opposition-based learning (OBL) is to concurrently assess the current solutions and its opposite solutions in order to obtain a better approximation of the current candidate solutions. Figure 2 illustrate the opposite point which is determined in domain  $[a,b]$ . Let  $x \in [a, b]$  be a minimum and maximum values of variable in current population. The opposite number  $ox$  is determined as:

$$ox = a + b - x \tag{5}$$

**Sub Topik**

In the original OBL concept, the agents and their opposite agents are asymmetric on the midpoint within the range of variables' current interval. This opposite agents might possibly flee from the global optimum, which leads to decrease the contribution of opposite points. Therefore, opposition-based learning using the current optimum (COOBL) was proposed in [9] to address this drawback. So this approach is used to enhance the effectiveness of the SKF. The proposed algorithm is known as current optimum opposition-based simulated Kalman filter (COOBSKF).

The significant difference is the formation of opposite population in COOBSKF is depends on the best agent so far which is identified by fitness calculation on particular objective function. The opposite population is generated using Equation 10.

$$ox_i = 2x_{co} - x_i \tag{6}$$

where  $x_{co}$  is the best agent so far or current optimum agent.



Figure 2. Opposite point defined in domain [a,b].

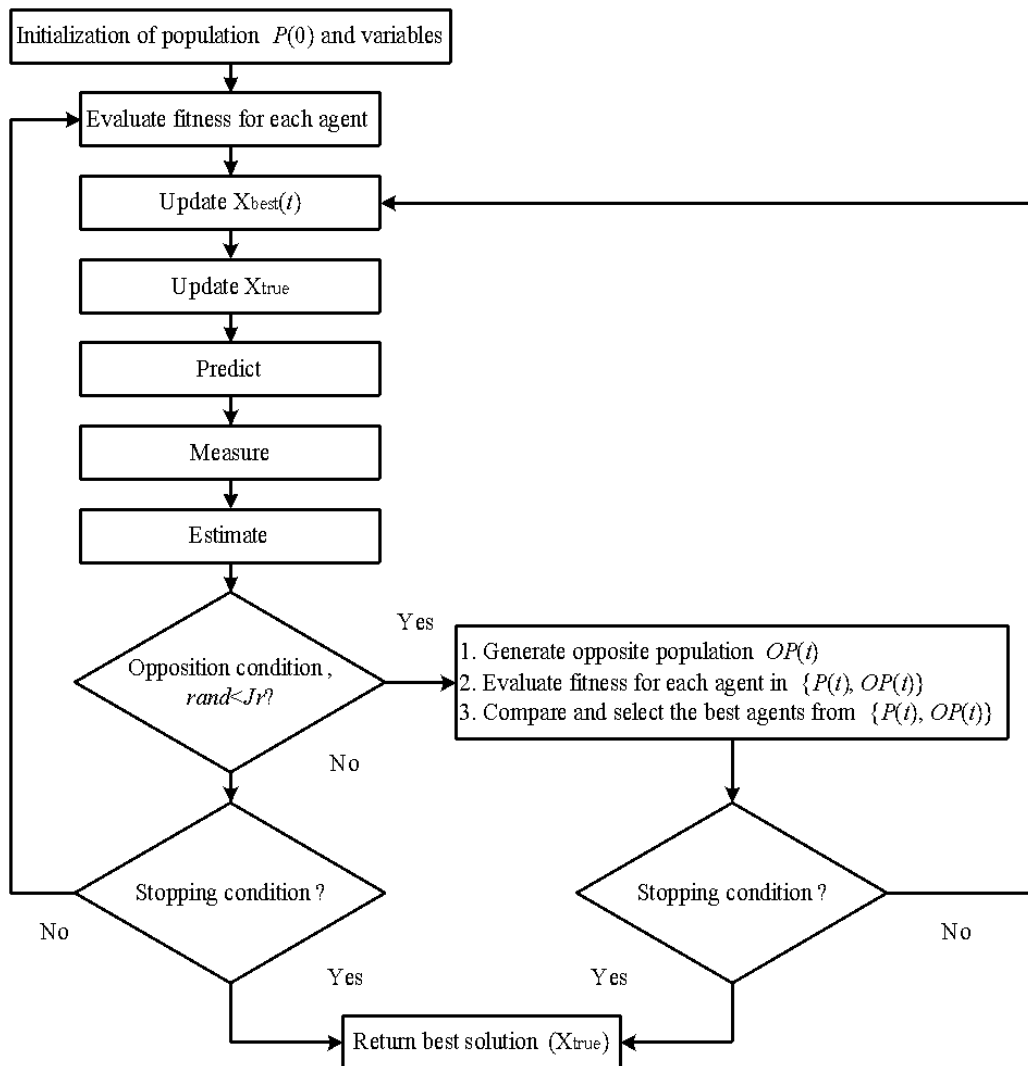


Figure 3. Flowchart of COOSKF algorithm.

### Enhancing SKF using current optimum opposition-based learning

The original SKF is selected as a parent algorithm and the COOBL strategies are embedded in SKF to boost its performance. COOBL is employed at one stage of SKF which is after estimation process of SKF. This implementation generated opposite population which is potentially fitter compared to the current ones. Figure 3 shows the flowchart of the proposed algorithm.

Initially, COOSKF generates randomly initial population or candidate solutions. The initial value of error covariance estimate,  $P(0)$ , the process noise value,  $Q$ , the measurement noise value,  $R$ , and jumping rate value,  $Jr$ , are also determined during initialization stage. Then, the fitness of agents in the population is calculated based on the

objective function. Next,  $X_{best}(t)$  and  $X_{true}$  are updated based on SKF algorithm steps. The algorithm continues with prediction, measurement and estimation similar to SKF algorithm using Equation 3 to Equation 8.

After that, COOBL is applied to the current solution in order to check a potential solution on opposite side. This action is performed probabilistically influenced by a parameter known as the jumping rate,  $Jr \in [0, 1]$ .  $Jr$  is a control parameter to form or ignore the formation of opposite population at specific iteration. The following jumping condition is considered:

```

if rand < Jr
    then
        apply COOBL
    else
        check stopping condition
else

```

**Table 1.** Table caption

Column number 1	Column number 2	Column number 3
Parameter 1 (N)	12.3	1.5
Parameter 2 (kg)	34.50	12.00
Parameter 3 (mm)	25	9

### PERBINCANGAN / DISCUSSION

The discussion should explore the significance of the findings and state them clearly in relation to the reviewed literature. A combined Findings and Discussion section is also allowed. The discussion should also highlight the contributions achieved by the work

This experiment investigates the performance of COOBSKF in comparison with other optimization algorithms such as particle swarm optimization (PSO), grey wolf optimizer (GWO), genetic algorithm (GA), gravitational search algorithm (GSA) and black hole (BH). The experimental parameters used in this experiment are shown in Table 7. For COOBSKF, the  $Jr$  value used is 0.9. For GSA,  $\alpha$  is set to 20 and initial gravitational constant,  $G_0$  is set to 100. For PSO, cognitive coefficient,  $c_1$ , and social coefficient,  $c_2$ , are set to 2. The inertia factor is linearly decreased from 0.9 to 0.4. For GWO, components of  $a$  are linearly decreased from 2 to 0. Lastly, for GA, the probabilities of selection and mutation are set to 0.5 and 0.2, respectively.

### KESIMPULAN / CONCLUSION

The main conclusions of the study may be presented in a short Conclusion section, which may include the limitations of the study

This paper reports the first attempt to enhance the exploration capability of SKF by applying COOBL technique. In addition, jumping rate is also integrated in the proposed method. Once the jumping rate condition is met, the opposite solution is selected if the solution is better than the current one. The analysis confirmed that the proposed COOBSKF is superior to SKF and better than GA, GWO, PSO and BH. For future research, different OBL techniques shall be considered to enhance further the SKF.

### PENGHARGAAN / ACKNOWLEDGEMENT

Sila nyatakan penghargaan kepada penaja kajian anda. Eg - The authors would like to thank UMP for funding this work under an internal grant RDU123456.

### RUJUKAN / REFERENCES

[1] K. Taylor, A. Post, T. B. Hoshizaki, and M. D. Gilchrist, "The effect of a novel impact management strategy on maximum principal strain for reconstructions of American football concussive events," *Proc. Inst. Mech. Eng. Part P J. Sport. Eng. Technol.*, vol. 233, no. 4, pp. 503–513, 2019, doi: 10.1177/1754337119857434.

[2] H. Tan, L. Qin, Z. Jiang, Y. Wu, and B. Ran, "A hybrid deep learning based traffic flow prediction method and its understanding," *Transp. Res. Part C Emerg. Technol.*, vol. 90, no. January, pp. 166–180, 2018, doi: 10.1016/j.trc.2018.03.001.

- [3] J. B. Caccese *et al.*, “Head and neck size and neck strength predict linear and rotational acceleration during purposeful soccer heading,” *Sport. Biomech.*, vol. 17, no. 4, pp. 462–476, 2018, doi: 10.1080/14763141.2017.1360385.
- [4] S. L. James *et al.*, “Global, regional, and national incidence, prevalence, and years lived with disability for 354 diseases and injuries for 195 countries and territories, 1990–2017: a systematic analysis for the Global Burden of Disease Study 2017,” *Lancet*, vol. 392, no. 10159, pp. 1789–1858, 2018, doi: 10.1016/S0140-6736(18)32279-7.
- [5] X. Huang, J. Sun, and J. Sun, “A car-following model considering asymmetric driving behavior based on long short-term memory neural networks,” *Transp. Res. Part C Emerg. Technol.*, vol. 95, no. February, pp. 346–362, 2018, doi: 10.1016/j.trc.2018.07.022.