VOL. 11, NO. 7, APRIL 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

ANGLE MODULATED SIMULATED KALMAN FILTER ALGORITHM FOR COMBINATORIAL OPTIMIZATION PROBLEMS

Zulkifli Md Yusof[1], Zuwairie Ibrahim[1], Ismail Ibrahim[1], Kamil Zakwan Mohd Azmi[1], Nor Azlina Ab Aziz[2], Nor Hidayati Abd Aziz[1, 2] and Mohd Saberi Mohamad[3]
[1]Faculty of Electrical and Electronics Engineering, Universiti Malaysia Pahang, Pekan, Malaysia
[2]Faculty of Engineering and Technology, Multimedia University, Melaka, Malaysia
[3]Faculty of Computing, Universiti Teknologi Malaysia, Johor, Malaysia

ABSTRACT

Inspired by the estimation capability of Kalman filter, we have recently introduced a novel estimation-based optimization algorithm called simulated Kalman filter (SKF). Every agent in SKF is regarded as a Kalman filter. Based on the mechanism of Kalman filtering and measurement process, every agent estimates the global minimum/maximum. Measurement, which is required in Kalman filtering, is mathematically modelled and simulated. Agents communicate among them to update and improve the solution during the search process. However, the SKF is only capable to solve continuous numerical optimization problem. In order to solve discrete optimization problems, the SKF algorithm is combined with an angle modulated approach. The performance of the proposed angle modulated SKF (AMSKF) is compared against two other discrete population-based optimization algorithms, namely, binary particle swarm optimization (BPSO) and binary gravitational search algorithm (BGSA). A set of traveling salesman problems are used to evaluate the performance of the proposed AMSKF. Based on the analysis of experimental results, we found that the proposed AMSKF is as competitive as BGSA but the BPSO is superior to the both AMSKF and BGSA.

Keywords: simulated kalman filter, angle modulated, combinatorial, traveling salesman problems.

INTRODUCTION

There are a lot of discrete optimization problems in literature and real-world applications. Examples of discrete optimization problems are assembly sequence planning [1-2], DNA sequence design [3-4], VLSI routing [5-6], robotics drill route problem [7], and airport gate allocation problem [8].

In solving discrete optimization problems, algorithms such genetic algorithm (GA) [9] has been originally developed to operate in binary search space. However, not all optimization algorithms are originally developed to operate in binary search space. An example of these algorithms is simulated Kalman filter (SKF), which has been recently introduced by Ibrahim *et al.* in 2015 [10]. In order to solve discrete optimization problems with SKF, modification or enhancement is needed. For example, sigmoid function has been employed as a mapping function to let particle swarm optimization (PSO) to operate in binary search space [11].

The objective of this research is to modify SKF algorithm for solving discrete optimization problem. However, mapping function cannot be integrated in SKF because there is no specific variable in SKF can be used as the input to mapping function. Thus, an angle modulated approach [12] is employed in this research. Angle modulated approach is universal, which means that it can be integrated to any optimization algorithm.

This paper is organized as follows. At first, SKF will be briefly reviewed followed by a detail description of the proposed angle modulated SKF (AMSKF) algorithm. Experimental set up will be explained, results will be shown and discussed. Lastly, a conclusion will be provided at the end of this paper.

SIMULATED KALMAN FILTER ALGORITHM

The simulated Kalman filter (SKF) algorithm is illustrated in Figure-1. Consider n number of agents, SKF algorithm begins with initialization of n agents, in which the states of each agent are given randomly. The maximum number of iterations, $t_{\max}$, is defined. The initial value of error covariance estimate, $P(0)$, the process noise value, Q, and the measurement noise value, R, which are required in Kalman filtering, are also defined during initialization stage.

Then, every agent is subjected to fitness evaluation to produce initial solutions $\{\boldsymbol{X}_1(0), \boldsymbol{X}_2(0), \boldsymbol{X}_3(0), \ldots, \boldsymbol{X}_{n-2}(0), \boldsymbol{X}_{n-1}(0), \boldsymbol{X}_n(0)\}$. The fitness values are compared and the agent having the best fitness value at every iteration, t, is registered as $\boldsymbol{X}_{\mathbf{best}}(t)$. For function minimization problem,

$$\boldsymbol{X}_{\mathbf{best}}(t) = \min_{i \in 1,\ldots,n} fit_i(\boldsymbol{X}(t)) \tag{1}$$

whereas, for function maximization problem,

$$\boldsymbol{X}_{\mathbf{best}}(t) = \max_{i \in 1,\ldots,n} fit_i(\boldsymbol{X}(t)) \tag{2}$$

The-best-so-far solution in SKF is named as $\boldsymbol{X}_{\mathbf{true}}$. The $\boldsymbol{X}_{\mathbf{true}}$ is updated only if the $\boldsymbol{X}_{\mathbf{best}}(t)$ is better (($\boldsymbol{X}_{\mathbf{best}}(t) < \boldsymbol{X}_{\mathbf{true}}$ for minimization problem, or $\boldsymbol{X}_{\mathbf{best}}(t) > \boldsymbol{X}_{\mathbf{true}}$ for maximization problem) than the $\boldsymbol{X}_{\mathbf{true}}$.

The subsequent calculations are largely similar to the predict-measure-estimate steps in Kalman filter. In the prediction step, the following time-update equations are computed.

$$\boldsymbol{X}_i(t|t) = \boldsymbol{X}_i(t) \tag{3}$$

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

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

The next step is measurement, which is a feedback to estimation process. Measurement is modelled such that its output may take any value from the predicted state estimate, $\boldsymbol{X}_i(t|t)$, to the true value, $\boldsymbol{X}_{\text{true}}$.

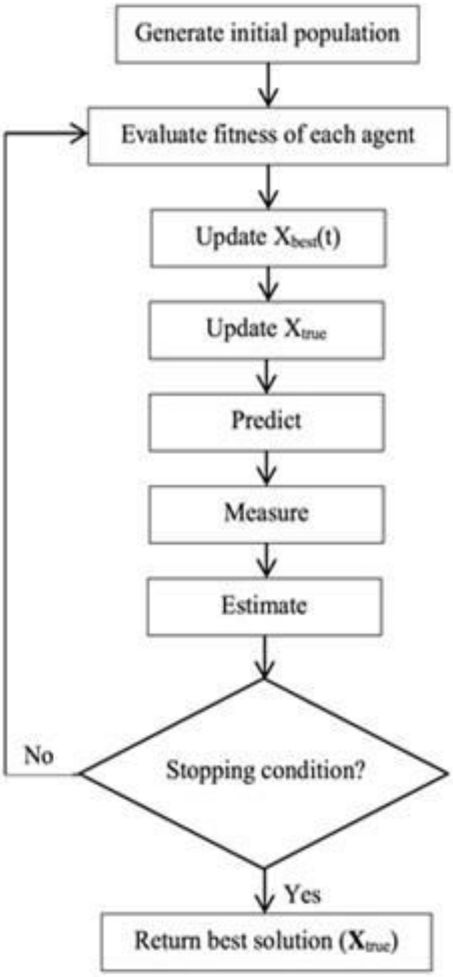


Figure-1. The simulated Kalman filter (SKF) algorithm.

Measurement, $Z_i(t)$, of each individual agent is simulated based on the following equation:

$$\boldsymbol{Z}_i(t) = \boldsymbol{X}_i(t|t) + \sin(rand \times 2\pi) \times |\boldsymbol{X}_i(t|t) - \boldsymbol{X}_{true}| \tag{5}$$

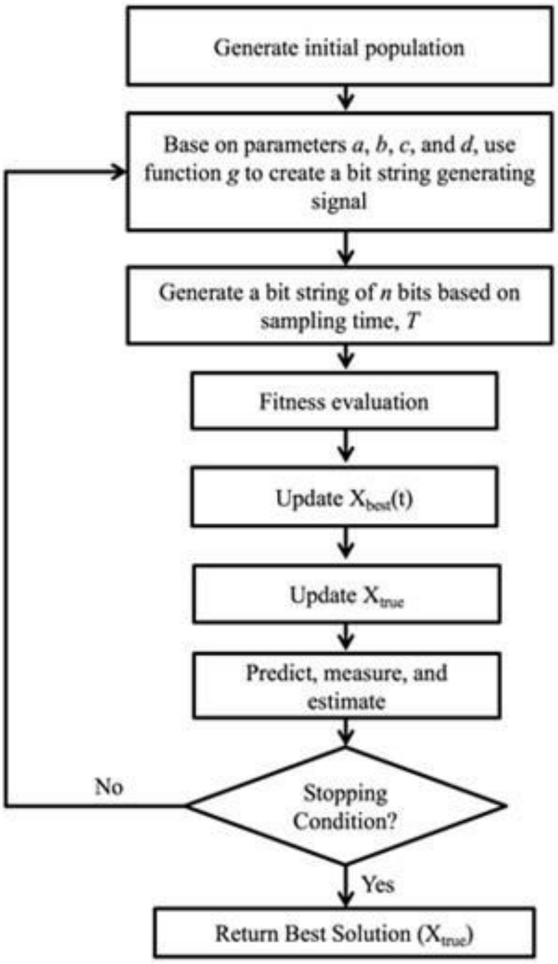
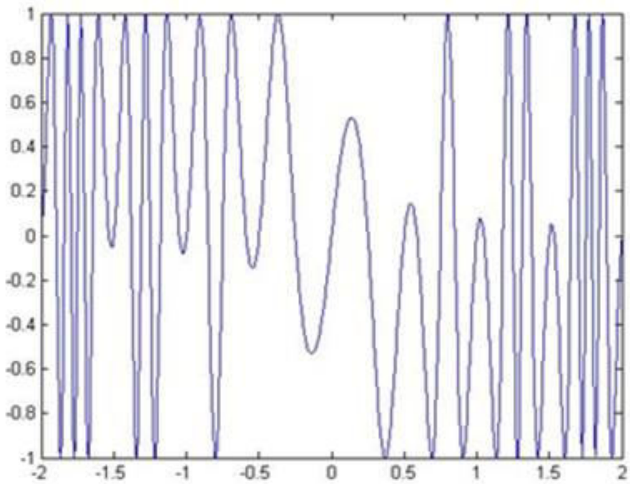


Figure-2. The angle modulated SKF (AMSKF) algorithm.



An example of $g(x)$ plot.

The $\sin(rand \times 2\pi)$ term provides the stochastic aspect of SKF algorithm and $rand$ is a uniformly distributed random number in the range of [0,1].

The final step is the estimation. During this step, Kalman gain, $K(t)$, is computed as follows:

ARPN Journal of Engineering and Applied Sciences

$$K(t) = \frac{P(t|t)}{P(t|t)+R} \quad (6)$$

Then, the estimation of next state, $X_i(t+1)$, is computed based on Eqn. (7).

$$X_i(t+1) = X_i(t|t) + K(t) \times (Z_i(t) - X_i(t|t)) \quad (7)$$

Finally, the next iteration is executed until the maximum number of iterations, t_{max}, is reached.

ANGLE MODULATED SIMULATED KALMAN FILTER ALGORITHM

The angle modulated SKF (AMSKF) algorithm is shown in Figure-2. The main idea of the angle modulated approach in solving combinatorial optimization problem is to use a function, $g(x)$, to create a continuous signal. The shape of signal $g(x)$ is determined by 4 variables, namely, a, b, c, and d, as shown in Equation (9).

where A = $2\pi(x-a) \times c$.

Figure-3 shows an example of $g(x)$ plot for the case of $a = 0$, $b = 1$, $c = 1$, and $d = 0$. The region $g(x) > 0$ is called binary 1 region and region $g(x) < 0$ is called binary 0 region. After that sampling based on sampling time, T, is executed to generate a bit string of length n. The required length of the bit string is problem dependent and determined by the size of a combinatorial optimization problem. For example, in traveling salesman problem (TSP), if the size of the problem is 10 cities, 4 × 10 bit string is required to represent a solution of TSP. At a specific x value, if $g(x) > 0$, bit 1 is assigned, otherwise, bit 0 is assigned, as shown in Figure-4.

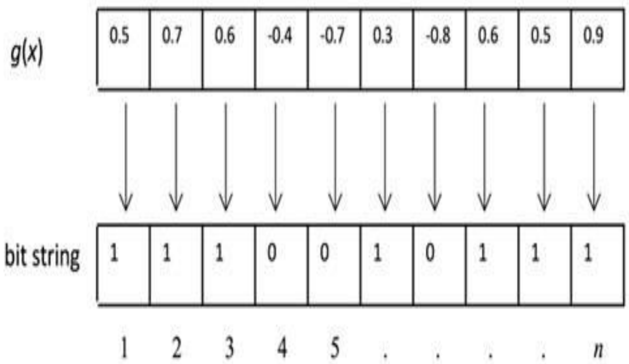


Figure-3. Bit string generation.

The main advantage of angle modulated approach is that complex calculation in producing high dimensional bit string can be avoided. The search process in solving a combinatorial optimization problem can be done by tuning the values of a, b, c, and d only. In this work, the tuning is done by the SKF algorithm.

Table-1. Property of the test problems.

TSP Index	Name	Size
1	Berlin52	52
2	Bier127	127
3	Ch130	130
4	Ch150	150
5	D198	198
6	D493	493
7	D657	657
8	D1291	1291
9	D2103	2103
10	DSJ1000	1000
11	Eil51	51
12	Eil76	76
13	Eil101	101
14	FL1400	1400
15	FL1577	1577
16	GIL262	262
17	KROA100	100
18	KROA150	150
19	KROA200	200
20	KROB100	100
21	KROB200	200
22	KROC100	100
23	KROD100	100
24	KROE100	100
25	LIN105	105
26	LIN318	318
27	P654	654
28	PCB442	442

Table-2. Experimental setting parameters.

Angle modulated parameters	
Parameter	**Value**
x_{min}	0
x_{max}	150
Sampling time, T	0.125
SKF parameters	
Parameter	**Value**
Error covariance, P	1000
Process noise, Q	0.5
Measurement noise, R	0.5
rand	[0,1]

Table-3. Experimental setting parameters BPSO.

Parameter	Value
Inertia weight, ω	0.9-0.4
Cognitive coefficient, c_1	2
Social coefficient, c_2	2

Table-4. Experimental setting parameters BGSA.

Parameter	Value
β	20
Initial gravitational value, Go	100

EXPERIMENTS

The AMSKF is applied to solve a set of TSP. The objective of TSP is to find the shortest distance from a start city to an end city while visiting every city not more than once. In this paper, 28 instances of TSPs are considered, from the size of 51 cities to 2103 cities, as shown in Table-1. These problems were taken from TSPLib [13].

Experimental setting for AMSKF is shown in Table-2. For benchmarking purpose, 2 additional experiments were considered, which are based on the well-established binary particle swarm optimization (BPSO) [11] and binary gravitational search algorithm (BGSA) [14]. Experimental setting for BPSO and BGSA are shown in Table-3 and Table-4, respectively. In all experiments, the number of runs, the number of agents, and the number of iterations are 50, 30, and 1000, respectively.

RESULT AND DISCUSSION

The proposed AMSKF is compared with BGSA and BPSO. The average performances of the three algorithms are presented in Table-5. The numbers written in bold show the best performance. It can be seen that AMSKF is able to find the best result for TSP index 10, 19, and 20. Based on these average performances, Wilcoxon signed rank test is performed. The result of the test is tabulated in Table-6. The level of significant chosen here is $\sigma = 0.05$. It is found that statistically no significant difference is found between AMSKF and BGSA. Both of the algorithms perform as good as each other in solving TSP problems. However, statistically, BPSO is found to perform significantly better than AMSKF in solving the benchmark problems used in this work. Examples of convergence curves are shown in Figure-4, Figure-5, and Figure-6.

Table-5. Average performance.

TSP Index	AMSKF	BGSA	BPSO
1	22874.86	23086.34	**22645.1**
2	544059.5	**510137.9**	534565.1
3	39357.7	39533.3	**39113.46**
4	46168.05	46660.14	**46159.44**
5	158018.6	**73367.02**	113701.9
6	411931.2	**281229.8**	314509.4
7	796174.9	**360067**	488599.4
8	1646428	462130.4	**371718.8**
9	3123015	1521030	**470854.3**
10	**523006025**	543116116	523506056
11	1266.809	1274.02	**1265.18**
12	2039.967	2059.6	**2036.86**
13	2856.43	**2694.38**	2834.8
14	1581635	**218859.2**	636912.1
15	1295453	**457890.4**	742191.3
16	23851.59	23897.04	**23829.98**
17	136954.9	137849.9	**136400.5**
18	215813.7	217679.2	**214231.5**
19	**291098.8**	293208.2	291490.1
20	**134818.2**	136383.4	134948.9
21	**285558.9**	287752.4	286063.4
22	135858.8	136650.5	**134922.5**
23	131561.2	132814.4	**131014.7**
24	137716.4	139215.2	**137300**
25	98766.64	**74610.86**	92507.64
26	528817.1	**307040.7**	316849.4
27	1848103	**655530.7**	746735.1
28	707728.3	**499374.4**	547372.8

Table-6. Wilcoxon test result.

Comparison	R^+	R^-
AMSKF vs BGSA	144	262
AMSKF vs BPSO	49	357

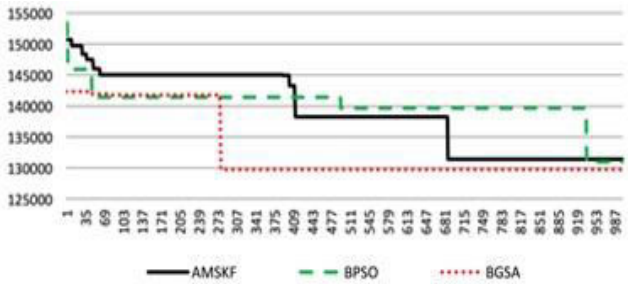


Figure-4. An example of convergence curve for TSP index 10.

ARPN Journal of Engineering and Applied Sciences

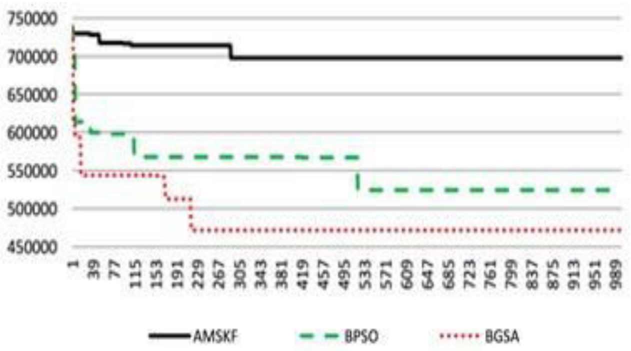


Figure-5. An example of convergence curve for TSP index 24.

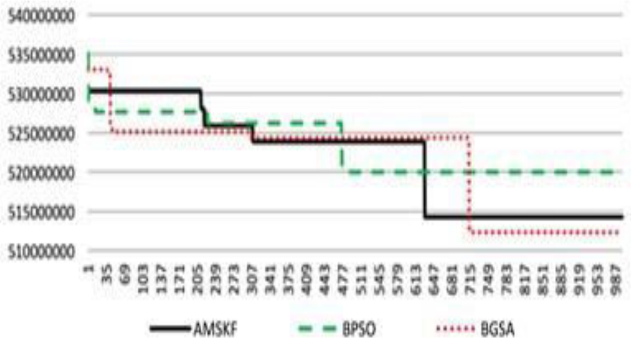


Figure-6. An example of convergence curve for TSP index 28.

CONCLUSIONS

This paper reports the first attempt to use SKF for solving combinatorial optimization problems. The proposed approach employed an angle modulated approach to generate a bit string that is required to represent solutions to a combinatorial optimization problem. Based on the proposed AMSKF, the best 4 variables are searched to produce the best $g(x)$ signal that is able to generate the best bit string representing the best answer to TSP. Experimental result and analysis showed the potential of AMSKF. Even though the performance of BPSO is better than AMSKF, AMSKF performed as good as BGSA. Currently, more experiments are being done. In particular, various TSP instances are considered. After that, computation time and complexity will be analysed in order to obtain a more concrete conclusion.

ACKNOWLEDGEMENTS

This work is financially supported by the UMP Post Graduate Research Scheme (GRS1503120) and by the Fundamental Research Grant Scheme (FRGS/1/2015/TK04/MMU/03/2) awarded by the Ministry of Higher Education (MOHE) to Multimedia University (MMU).

REFERENCES

[1] Mukred J.A.A, Ibrahim Z., Ibrahim I., Adam A., Wan K., Yusof Z.M., and Mokhtar N. 2012. A Binary Particle Swarm Optimization Approach to Optimize Assembly Sequence Planning, Advance Science Letters, Vol. 13, pp. 732-738.

[2] Ibrahim I., Ibrahim Z., Ahmad H., Mat Jusof M.F, Yusof Z.M., Nawawi S.W., and Mubin M. 2015. An Assembly Sequence Planning Approach with a Rule-based Multi State Gravitational Search Algorithm, The International Journal of Advanced Manufacturing Technology, Vol. 79, pp. 1363-1376.

[3] Yakop F., Ibrahim Z., Zainal Abidin A.F., Yusof Z.M., Mohamad M.S., Wan K., and Watada J. 2012. An Ant Colony System for Solving DNA Sequence Design Problem in DNA Computing, International of Innovative Computing, Information and Control, Vol. 8, No. 10, pp. 7329-7339.

[4] Ibrahim Z., Khalid N.K., Ibrahim I., Lim K.S., Bunyamin S., Yusof Z.M., and Muhammad M.S. 2011. Function Minimization in DNA Sequence Design Based on Binary Particle Swarm Optimization. Journal Teknologi D (UTM), No. 54, pp. 331-342.

[5] Yusof Z.M., Zainal Abidin A.F., Salam M.N.A., Khalil K., Mukred J.A.A., Hani M.K., and Ibrahim Z. 2011. A Binary Particle Swarm Optimization Approach for Buffer Insertion in VLSI Routing. International Journal of Innovative Management, Information and Production, Vol. 2, No. 3, pp. 34-39.

[6] Yusof. Z.M., Zainal Abidin A.F., Adam A., Khalil K., Mukred J.A.A., Mohamad M.S., Hani M.K., and Ibrahim Z. 2012. A Two Step Binary Binary Particle Swarm Optimization Approach for Routing in VLSI. ICIC Express Letters, Vol. 6, No. 3, pp.771-776.

[7] Othman M.H., Zainal Abidin A.F., Adam A., Yusof Z.M., Ibrahim Z., Mustaza S.M., and Lai Y.Y. 2011. A Binary Particle Swarm Optimization Approach for Routing in PCB Holes Drilling Process. International Conference on Robotic Automation System (ICORAS2011), pp. 201-206.

[8] Bouras A., Ghaleb M.A., Suryahatmaja U.S., and Salem A.M. 2014. The Airport Gate Assignment Problem: A Survey. The Scientific World Journal, Vol. 2014, Article ID 923859.

[9] Goldberg D. E. 1989. Genetic algorithm in search, optimization and machine learning. Addison-Wesley Longman Publishing Co., Inc. Boston, USA.

[10] Ibrahim Z., Abdul Aziz N. H., Ab Aziz N. A., Razali S., Shapiai M. I., Nawawi S. W., and Mohamad M. S. A Kalman filter approach for solving unimodal optimization problems. 2015. ICIC Express Letters (accepted).

[11] Kennedy, J. and Eberhart, R. 1997. A discrete binary version of the particle swarm algorithm. IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation, Vol. 5, pp. 4104–4108.

[12] Pampara, G. 2013. Angle modulated population based algorithms to solve binary problems. Master Thesis, University of Pretoria.

[13] comopt.ifi.uni-heidelberg.de/software/TSPLIB95/

[14] Rashedi, E., Nezamabadi-pour, H., and Saryazdi. S. 2010. BGSA: binary gravitational search algorithm. Natural Computing, Vol. 9, Issue 3, pp. 727-745.