Three Approaches to Solve Combinatorial Optimization Problems using Simulated Kalman Filter

Zulkifli Md Yusof, Ismail Ibrahim and Zuwairie Ibrahim
Faculty of Electrical and Electronics Engineering
Universiti Malaysia Pahang
26600 Pekan, Malaysia
zuwairie@ump.edu.my

Khairul Hamimah Abas, Shahdan Sudin
Faculty of Electrical Engineering
Universiti Teknologi Malaysia
81310 Skudai, Johor, Malaysia

Nor Azlina Ab Aziz, Nor Hidayati Abd Aziz
Faculty of Engineering and Technology
Multimedia University
75450 Melaka, Malaysia

Mohd Saberi Mohamad
Faculty of Computing
Universiti Teknologi Malaysia
81310 Johor, Malaysia

Abstract— Inspired by the estimation capability of Kalman filter, we have recently introduced 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 combinatorial optimization problems, three extended versions of SKF algorithm, which is termed as Angle Modulated SKF (AMSKF), Distance Evaluated SKF (DESKF), and Binary SKF (BSKF), are proposed. A set of traveling salesman problems is used to evaluate the performance of the proposed algorithms.

Keywords—simulated kalman filter; traveling salesman problem; combinatorial optimization

1. INTRODUCTION

In solving discrete optimization problems, algorithms such genetic algorithm (GA) [1] have 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 [2]. In order to solve discrete optimization problems with SKF, modification or enhancement is needed. In this paper, three extended versions of SKF algorithm, which is termed as Angle Modulated SKF (AMSKF) [3], Distance Evaluated SKF (DESKF) [4], and Binary SKF (BSKF) [5], have been proposed.

2. SIMULATED KALMAN FILTER

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 such simulated. Agents communicate among them to update and improve the solution during the search process. The simulated Kalman filter (SKF) algorithm is illustrated in Fig. 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,

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

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.

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

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

where $\boldsymbol{X}_{\boldsymbol{i}}(t)$ and $\boldsymbol{X}_{\boldsymbol{i}}(t|t)$ are the current state and current transition/predicted state, respectively, and $P(t)$ and $P(t|t)$ are 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.

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, $\boldsymbol{X}_{\text{true}}$. Measurement, $\boldsymbol{Z}_{\boldsymbol{i}}(t)$, of each individual agent is simulated based on the following equation:

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

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:

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

Then, the estimation of next state, $\boldsymbol{X}_{\boldsymbol{i}}(t+1)$, and the updated error covariant are computed based on Eqn. (7) and Eqn. (8), respectively.

$$\boldsymbol{X}_i(t+1) = \boldsymbol{X}_i(t|t) + \Delta_i \tag{7}$$

$$P(t+1) = \big(1 - K(t)\big) \times P(t|t) \tag{8}$$

where $\Delta_i = K(t) \times (\boldsymbol{Z}_i(t) - \boldsymbol{X}_i(t|t))$. Finally, the next iteration is executed until the maximum number of iterations, t_{max}, is reached.

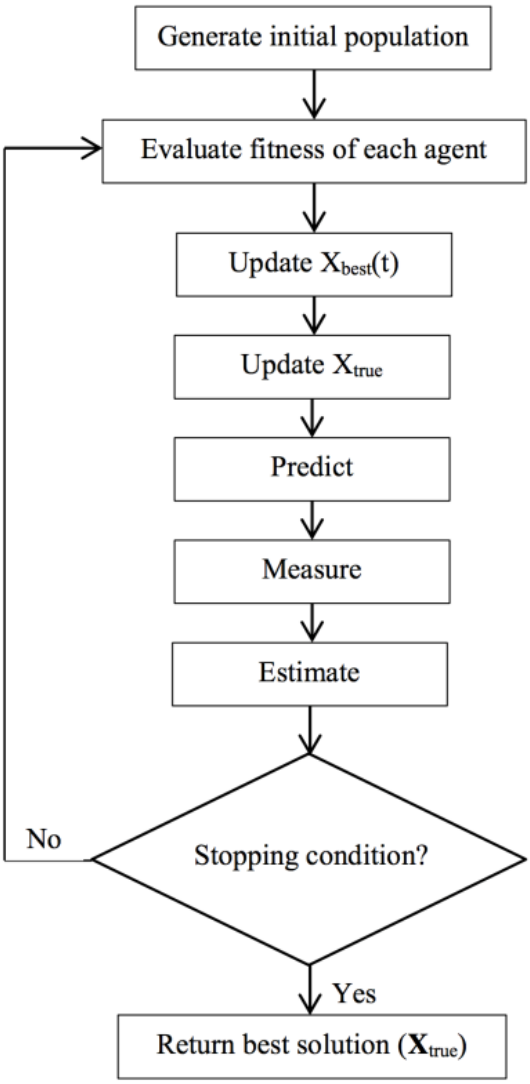

Figure 1: The original simulated Kalman filter (SKF) algorithm.

3. DISTANCE EVALUATED SIMULATED KALMAN FILTER ALGORITHM

In population-based search algorithm, generally, agents are randomly positioned in the search space. Then, the agents move in the search space to find global minimum or maximum. During the beginning of the search, exploration is preferred to make sure the search covers almost all regions in the search space. In this stage of search process, the position between agents is normally far with each other. As the search process continues, during the end of the search, exploration is no longer preferred because fine-tuning or exploitation is more preferred. During exploitation, agents becomes closer to each other and hence, the distance among them decreases.

The position of agents in a search space during a typical search process is illustrated in Fig. 2, Fig. 3, and Fig. 4. Normally, as the iteration continues, the distance between agents and the best-so-far solution decreases. This distance plays an important role in the proposed distance evaluated simulated Kalman filter algorithm (DESKF). In DESKF, the distance is mapped into a probabilistic value [0,1] and then the probabilistic value will be compared with a random number [0,1] to update a bit string or solution to a combinatorial optimization problem. In detail, most of the calculations in the proposed DESKF are similar to the original SKF. Modifications are needed only during initialization and generation of solution to combinatorial optimization problem.

Figure-2. Position of agents at the beginning of the search.

Figure-3. Position of agents during the middle of the search.

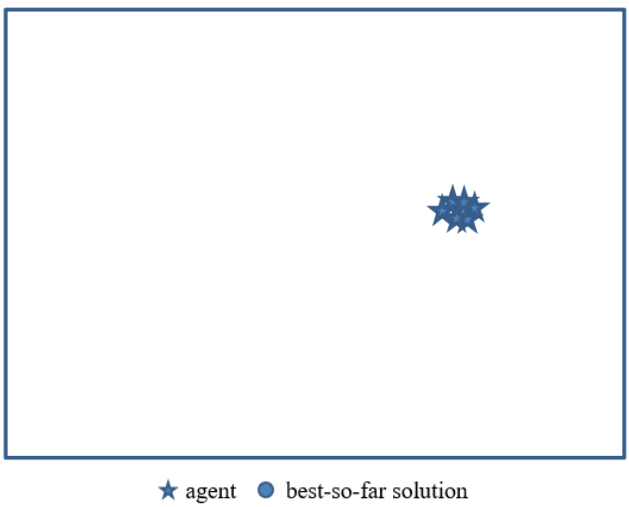

Figure-4. Position of agents at the end of the search.

During the initialization of agents, in SKF, the states of each agent are given randomly. An additional initialization is introduced in DESKF. Every agent is associated with a random bit string as well. The length of the bit string is problem dependent and subjected to the size of the problem. Thus, 2 types of variables are associated with an agent in SKF: continuous variable, *x*, which is produced as estimated value of SKF (which is also similar to the position of agents in a search space), and a bit string, Σ, which is usually used to represent solution to a combinatorial optimization problem.

In DESKF, for a particular *d*th dimension, the distance between an *i*th agent to the best-so-far solution at iteration *t* can be calculated as follows:

$$D_i^d(t) = x_i^d(t) - x_{best-so-far}^d(t) \tag{9}$$

In binary gravitational search algorithm (BGSA) [6], a function, as shown in Fig. 5, is used to map a velocity value into a probabilistic value within interval [0,1]. Similar function is used in DESKF. This distance value, $D_i^d(t)$, is mapped to a probabilistic value within interval [0,1] using a probability function, $S(D_i^d(t))$, as follows:

$$S\left(D_i^d(t)\right) = \left|\tanh\left(D_i^d(t)\right)\right| \tag{10}$$

After the $S\left(D_i^d(t)\right)$ is calculated, a random number, *rand*, is generated and a binary value at dimension *d* of an *i*th agent, Σ_i^d, is updated according to the following rule:

$$\begin{aligned} &\text{if } rand < S\left(D_i^d(t)\right) \\ &\text{then } \Sigma_i^d(t+1) = \text{complement}\Sigma_i^d(t+1) \\ &\text{else } \Sigma_i^d(t+1) = \Sigma_i^d(t+1) \end{aligned} \tag{11}$$

4. ANGLE MODULATED EVALUATED SIMULATED KALMAN FILTER ALGORITHM

The angle modulated SKF (AMSKF) algorithm is shown in Fig. 5. 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 Eqn. (12).

$$g(x) = \sin(2\pi(x-a) \times b \times \cos(A)) + d \tag{12}$$

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

Fig. 6 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.

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.

3. BINARY SIMULATED KALMAN FILTER

In order to solve a combinatorial optimization problem using SKF, the Δ_i term in Eqn. (7) is mapped into a probabilistic value [0,1]. Then, the probabilistic value is compared to a random number [0,1] to update a bit string. In BSKF, most calculations are similar to the original SKF. Modifications are needed only during initialization and generation of solution to combinatorial optimization problem.

During the initialization of agents, a random bit string, Σ_i , is generated for each agent. Each bit in the bit string is associated to a dimension. The length of the bit string is problem dependent and subjected to the size of the problem. In this study, the term Δ_i is mapped to a probabilistic value within interval [0,1] using a mapping function, $S(\Delta i(t))$, as follows:

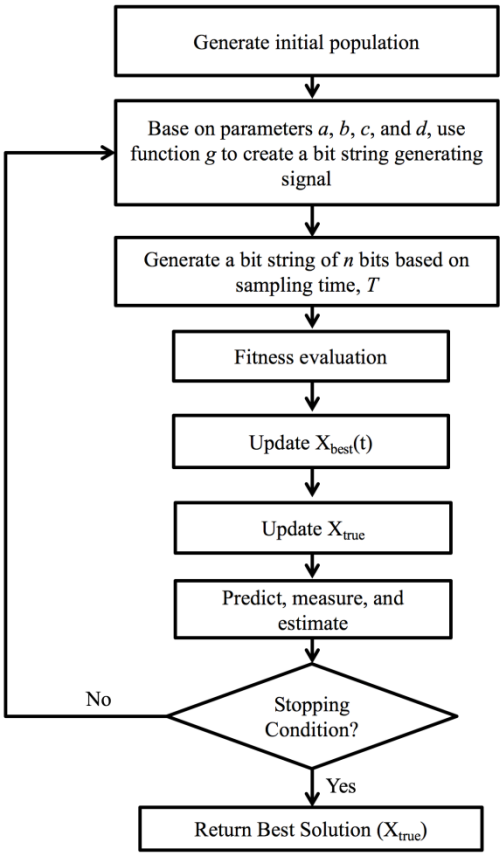

Figure-5. The angle modulated SKF (AMSKF) algorithm.

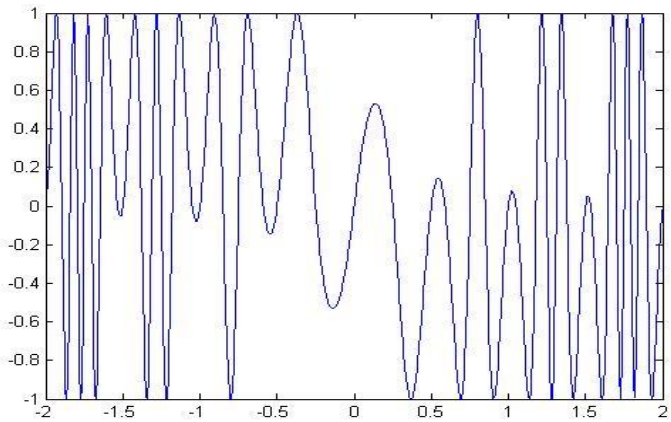

Figure-6. An example of $g(x)$ plot.

$$S(\Delta i(t)) = \left|\tanh(\Delta i(t))\right| \tag{13}$$

After the $S(\Delta i(t))$ is calculated, a random number, *rand*, is generated and a binary value at dimension *d* of an *i*th agent, Σ_i^d, is updated according to the following rule:

$$\begin{aligned} &\text{if} \quad rand < S(\Delta i(t)) \\ &\text{then } \Sigma_i^d(t+1) = \text{complement}\Sigma_i^d(t+1) \\ &\text{else } \Sigma_i^d(t+1) = \Sigma_i^d(t+1) \\ &\text{end} \end{aligned} \tag{14}$$

4. EXPERIMENTS, RESULT AND DISCUSSION

The three algorithms are 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, 51 instances of TSPs are considered, from the size of 51 cities to 1400 cities. These problems were taken from TSPLib [6]. Experimental setting parameters for SKF are shown in Table 1. Example of convergence curve are shown in Figure 7.

The averaged results are shown in Table 2 and Table 3. Based on these average performances, Friedman rank test is performed, which shows that the AMSKF is ranked first compared to DESKF and BSKF. However, Friedman Post Hoc analysis in Table 4, which is based on confidence level, $\sigma = 0.05$, shows that the algorithms perform as good as each other in solving TSP problems.

Table 1: Experimental setting parameters.

SKF parameters	
Parameter	**Value**
P	1000
Q	0.5
R	0.5
rand	[0,1]

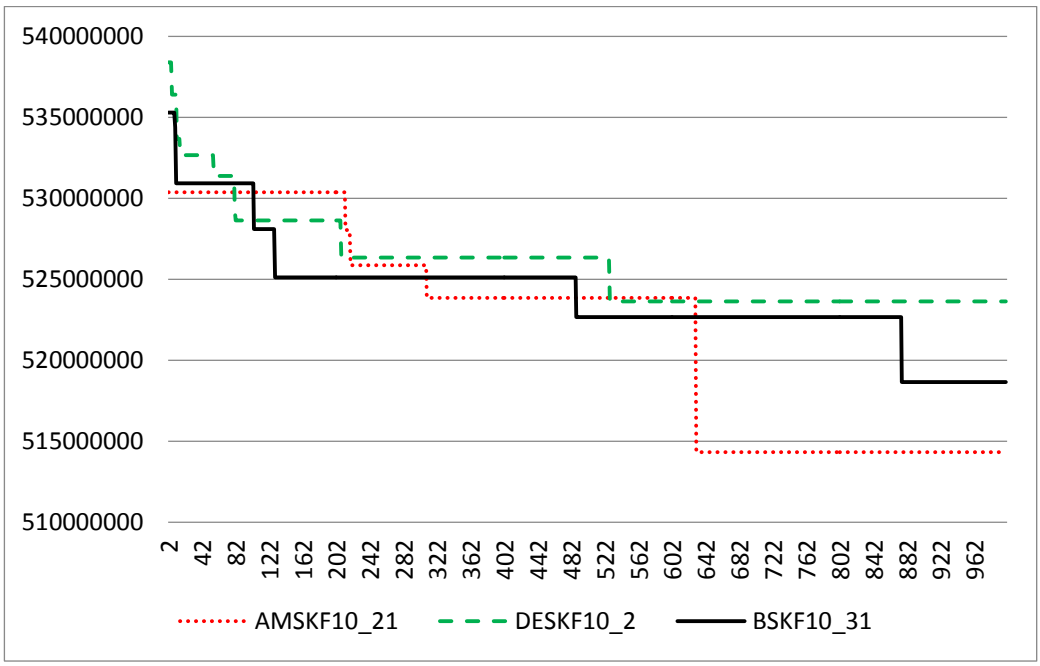

Figure-7. Example of convergence curve for TSP index 10 (DSJ1000)

Table 2: Average performance (1-23)

	AMSKF	**DESKF**	**BSKF**
1 Berlin52	22874.864	22932.196	22847.638
2 Bier127	544059.54	544106.72	542440
3 Ch130	39357.698	39254.37	39267.001
4 Ch150	46168.048	46270.792	46174.032
5 D198	158018.62	157618.45	158476.86
6 D493	411931.21	411998.9	411621.01
7 D657	796174.92	796175.26	796929.4
8 D1291	1646428.1	1645013.4	1648226.7
9 D2103	3123015.2	3123370	3123722.9
10 DSJ1000	523006026	524027900	523661043
11 Eil51	1266.8089	1268.4167	2127.613
12 Eil76	2039.9669	2052.8553	23782.281
13 Eil101	2856.4297	2845.6591	2853.754
14 FL1400	1581634.6	1581880.8	1581580.5
15 FL1577	1295453.1	1294521	1295395
16 GIL262	23851.59	23846.462	23853.898
17 KROA100	136954.87	137042.97	137188.72
18 KROA150	215813.74	216442.08	215796.9
19 KROA200	291098.84	291940.41	291063.76
20 KROB100	134818.21	134923.42	134786.51
21 KROB200	285558.87	285802.7	286095.52
22 KROC100	135858.77	135469.49	135539.31
23 KROD100	131561.24	131622.26	131396.8

ACKNOWLEDGEMENTS

The authors will like to thank Universiti Malaysia Pahang for providing internal financial support through grant GRS1503120. This research is also supported by Fundamental Research Grant Scheme (FRGS) awarded to Universiti Malaysia Pahang (RDU RDU140132).

Table 3: Average performance (24-51)

	AMSKF	**DESKF**	**BSKF**
24 KROE100	137716.4	138503.85	138610.66
25 LIN105	98766.643	99036.194	99045.127
26 LIN318	528817.07	527049.45	529112.66
27 P654	1848103.2	1845492	1849636.9
28 PCB442	707728.27	708486.35	708016.85
29 PCB1173	1335123.6	1333055.1	1335923.2
30 PR76	461175.97	461023.28	461949.66
31 PR107	446571.49	446386.83	449263.34
32 PR124	573148.53	580257.8	579691.23
33 PR136	689959.69	690108.27	689880.39
34 PR144	686191.06	682605.35	682410.76
35 PR152	886368.97	886217.15	886457.25
36 PR226	1477167	1479082.5	1482490.1
37 PR264	954069.8	954199	958776.77
38 PR299	667263.24	664536.98	666494.59
39 PR439	1732577.2	1737005.4	1731522.6
40 PR1002	6078577.3	6085012.6	6079543.2
41 PR2392	14689847	14656690	14683027
42 RAT99	6718.7332	6696.1764	6732.7713
43 RAT195	19441.651	19422.964	19461.392
44 RAT575	104311.85	103909.28	104247.95
45 RAT783	167018.76	166512.95	166982.97
46 RD100	45664.306	46096.301	45944.33
47 RL1304	8908134.4	8917743	8916298.5
48 RL1323	9303447	9303793.8	9302485.9
49 RL1889	14159546	14171974	14157634
50 ST70	2902.0918	2882.9964	2890.8754
51 TS225	1410332.7	1411955.3	1409168.9

Table 4: Friedman rank

	AMSKF	**DESKF**	**BSKF**
Ranking	1.8627	2.0000	2.1373

REFERENCES

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

[2] Z. Ibrahim, N. H. Abdul Aziz, N. A. Ab. Aziz, S. Razali, M. I. Shapiai, S. W. Nawawi, and M. S. Mohamad, A Kalman Filter Approach for Solving Unimodal Optimization Problems, ICIC Express Letters, Vol. 9, Issue 12, pp. 3415-3422, 2015.

[3] Z. Md Yusof, Z. Ibrahim, I. Ibrahim, K. Z. Mohd Azmi, N. A. Ab Aziz, N. H. Abd Aziz, and M. S. Mohamad, Angle Modulated Simulated Kalman Filter Algorithm for Combinatorial Optimization Problems, ARPN Journal of Engineering and Applied Sciences, Vol. 11, No. 7, pp. 4854-4859, 2015.

[4] Z. Md Yusof, Z. Ibrahim, I. Ibrahim, K. Z. Mohd Azmi, N. A. Ab Aziz, N. H. Abd Aziz, and M. S. Mohamad, Distance Evaluated Simulated Kalman Filter for Combinatorial Optimization Problems, ARPN Journal of Engineering and Applied Sciences, Vol. 11, No. 7, pp. 4904-4910, 2015.

[5] Z. Md Yusof, I. Ibrahim, S. N. Satiman, Z. Ibrahim, N. H. Abd Aziz, and N. A. Ab Aziz, BSKF: Binary Simulated Kalman Filter, Third International Conference on Artificial Intelligence, Modelling and Simulation, pp. 77-81, 2015.

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