Solving Assembly Sequence Planning using Angle Modulated Simulated Kalman Filter

Ainizar Mustapa, Zulkifli Md. Yusof, Asrul Adam, Badaruddin Muhammad and Zuwairie Ibrahim
Universiti Malaysia Pahang, 26600 Pekan, Pahang Malaysia
zmdyusof@ump.edu.my

Abstract. This paper presents an implementation of Simulated Kalman Filter (SKF) algorithm for optimizing an Assembly Sequence Planning (ASP) problem. The SKF search strategy contains three simple steps; predict-measure-estimate. The main objective of the ASP is to determine the sequence of component installation to shorten assembly time or save assembly costs. Initially, permutation sequence is generated to represent each agent. Each agent is then subjected to a precedence matrix constraint to produce feasible assembly sequence. Next, the Angle Modulated SKF (AMSKF) is proposed for solving ASP problem. 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 performance of the proposed AMSKF is compared against previous works in solving ASP by applying BGSA, BPSO, and MSPSO. Using a case study of ASP, the results show that AMSKF outperformed all the algorithms in obtaining the best solution.

1. Introduction
Assembly optimisation in the production planning stage deals with determination of optimum assembly sequence and determination of optimum location of each resource. Solving the Assembly Sequence Planning (ASP) problem is crucial because it will determine many assembly aspects including tool changes, fixture design and assembly freedom. Assembly sequence also influences overall productivity because it determines how fast and accurate the product is assembled.

Assembly sequence planning consists of assembly, operations, existing assembly techniques, and some details of relations between parts. Some researchers have dedicated their work on some important issues related to concurrent engineering analyses on assembly sequence planning. These issues are the representation of a product to be assembled, the generation of assembly sequence plans and the determination of precedence constraints, the representation of resulting assembly sequence plans, and the selection of the optimum assembly sequence planning. In this paper, assumptions for ASP are as follows: (1) setup time and the actual assembly time for each part and component are given, (2) transfer time between workstations is included in set up time, and (3) downtime of machines and workstations is omitted.

The total assembly time is the combination of setup time and actual assembly time. It is assumed that regardless of the assembly sequence, the actual assembly time is constant. A proper tool and setup for each component to be assembled is required. These two items depend on the geometry of the component itself and the components assembled to that point. The setup time for a component can be predicted using the following function in Eq. (1):
\( \sum_{i=1}^{c} p_{i} + \sum_{b=1}^{e} p_{ab} q_{ab} \)  
\[ (1) \]

where \( a \) is the number of component to be assembled, \( p_{a0} \) is the setup time for product \( (a) \) being the first component, \( p_{ab} \) is the contribution to the setup time due to the presence of part \( (b) \) when entering part \( a \), and \( q_{ab} = 1 \) if component \( (b) \) has already been assembled. Otherwise, \( q_{ab} = 0 \). The total assembly time is the summation of setup time and actual assembly time. Hence, the objective function for minimizing the assembly time is shown in the Eq. (2).

\[ \text{Min } T_{\text{Assembly}} = \sum_{b=1}^{c} \left( T_{\text{Setup}} (a) + A_{a} \right) \]
\[ (2) \]

where \( A_{a} \) is the assembly time for component \( a \).

2. Angle Modulated Simulated Kalman Filter (AMSKF) Algorithm

The simulated Kalman filter (SKF) algorithm [1-2] 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, \( t_{\text{max}} \), 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 \( \{ X_{1}(0), X_{2}(0), X_{3}(0), \ldots, X_{n-2}(0), X_{n-1}(0), X_{n}(0) \} \). The fitness values are compared and the agent having the best fitness value at every iteration, \( t \), is registered as \( X_{\text{best}}(t) \).

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

\[ X_{i}(t+1) = X_{i}(t) \]
\[ P(t+1) = P(t) + Q \]
\[ (3) \]

where \( X_{i}(t) \) and \( X_{i}(t+1) \) are the current state and current transition/predicted state, respectively, and \( P(t) \) and \( P(t+1) \) 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, \( X_{i}(t|t) \), to the true
value, \( X_{\text{true}} \). Measurement, \( Z_i(t) \), of each individual agent is simulated based on the following equation:

\[
Z_i(t) = X_i(t|t) + \sin(rand \times 2\pi) \times |X_i(t|t) - X_{\text{true}}|
\]  

(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 estimate. During this step, Kalman gain, \( K(t) \), is computed as follows:

\[
K(t) = \frac{P(t|t)}{P(t|t)+R}
\]  

(6)

Then, the estimation of next state, \( X_i(t+1) \), and the updated error covariant, \( P(t+1) \), are computed based on Eq. (7) and Eq. (8), respectively.

\[
X_i(t+1) = X_i(t|t) + \Delta_i
\]  

(7)

\[
P(t+1) = (1-K(t)) \times P(t|t)
\]  

(8)

where \( \Delta_i = K(t) \times (Z_i(t) - X_i(t|t)) \). Finally, the next iteration is executed until the maximum number of iterations, \( i_{\text{max}} \), is reached.

The angle modulated simulated Kalman filter (AMSKF) algorithm [3] is an extension of simulated Kalman filter (SKF) algorithm. It has been applied as feature selection for EEG signal peak classification [4]. The main idea of the angle modulated approach in solving combinatorial optimization problem is to use a function, \( g(x) = \sin(2\pi(x-a) \times b \times \cos(A)) + d \), to create a continuous signal, where \( A = 2\pi(x-a) \times c \). 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. Angle Modulated Simulated Kalman Filter (AMSKF) for the ASP

To find an optimal solution, each agent representing feasible assembly sequence must be evaluated to obtain its fitness value. This evaluation of the fitness value is done after initial population is generated and Precedence Matrix (PM), coefficient table and actual assembly time are loaded. These three processes are illustrated along others in the proposed approach based on AMSKF, as depicted in Figure 2. The PM can be referred to Table 1. The coefficient table can be referred in [5] for details. The PM shows relationship between 19 components as illustrated in Figure 3. It is worth pointing out that the components of free to be assembled are the components that can be placed regardless of any part of a sequence.

As a result, each agent produces a feasible assembly sequence. The optimum one is then selected from the feasible assembly sequences by evaluating fitness of each agent. The best of the population which is a more optimal until the stopping condition is met. After the stopping condition is met, the performance of the AMSKF can be investigated. To conclude the finding, the performance of AMSKF in solving the ASP problem is compared against some related metaheuristic methods such as Binary Gravitational Search Algorithm (BGSA) [6], Binary Particle Swarm Optimization (BPSO) [7], and Multi-State Particle Swarm Optimization (MSPSO) [8].
Figure 2. The AMSKF algorithm for the ASP.

![Algorithm Diagram]

Components free to be assembled

Figure 3. The assembly precedence diagram for the case study.

![Assembly Precedence Diagram]

Table 1. Precedence matrix (PM) for the case study

| Comp. ($a$) | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
|-------------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|
| (a)         |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |
| 1           | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| 2           | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 3           | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 4           | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 5           | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 6           | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 7           | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 8           | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 9           | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 10          | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 11          | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 12          | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 13          | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 14          | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 15          | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 16          | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 17          | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 18          | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| 19          | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
Table 2. The parameters and its value used for AMSKF, BGSA, BPSO, and MSPSO experiments

| Parameter                                | AMSKF (proposed work) | BGSA   | BPSO   | MSPSO |
|------------------------------------------|------------------------|--------|--------|-------|
| Iteration                                | 1000                   | 500    | 500    | 500   |
| Component No.                            | 19                     | 19     | 19     | 19    |
| No. of agent                             | 10                     | 50     | 50     | 25    |
| Initial error covariance estimate, P(0)  | 100                    | -      | -      | -     |
| Process noise, Q                         | 0.5                    | -      | -      | -     |
| Measurement noise, R                     | 0.5                    | -      | -      | -     |
| Initial temperature                      | -                      | 100°C  | -      | -     |
| Cooling rate                             | -                      | 0.95   | -      | -     |
| Inertia weight, ω                        | -                      | -      | 0.6    | -     |
| Learning factor                          | -                      | -      | 1.42   | -     |
| $\omega$ Initial                         | -                      | -      | -      | 0.9   |
| $\omega$ Final                           | -                      | -      | -      | 0.4   |
| Coefficient factor, $c_1$ and $c_2$      | -                      | -      | -      | 2     |

Table 3. The results of AMSKF against BGSA, BPSO, and MSPSO

| Algorithm | Min  | Mean | Max  | SD   | Assembly sequence                  |
|-----------|------|------|------|------|-----------------------------------|
| AMSKF     | 503.91 | 520.19 | 528.70 | 4.94 | 1-2-12-4-3-9-13-15-11-5-16-6-18-7-8-14-10-17-19 |
| BGSA      | 508.20 | 523.79 | 531.60 | 4.53 | 2-1-4-9-3-12-5-13-15-18-16-6-11-7-8-10-14-17-19 |
| MSPSO     | 514.00 | 530.00 | 538.50 | 4.50 | 2-4-3-1-9-12-5-13-15-18-16-11-6-7-8-10-14-17-19 |
| BPSO      | 515.80 | 520.80 | 523.60 | 3.10 | 13-2-3-5-12-15-16-4-1-11-9-18-6-7-8-10-14-17-19 |

4. Experiments, Results, and Discussion

In this study, the assembly of product consisting 19 components is considered. In experiments, the parameters and its value used for AMSKF, BGSA, BPSO, and MSPSO are presented in Table 2. The quality of results of AMSKF is then measured based on the fitness values of the best solutions in minimizing the total assembly time. For instance, if the number of independent trials on the case study is 50, the quality of results is determined based on the fitness values of 50 solutions.

To simplify the understanding of this work, fitness or objective value and solution is now called total assembly time and feasible assembly sequence, respectively. The average (mean), minimum (min), and maximum (max) of total assembly time of 50 feasible assembly sequences, and the standard deviation (SD) are recorded. Table 3 presents comparison of the result of AMSKF against BGSA, BPSO, and MSPSO. Based on the results given in Table 3, AMSKF outperformed BGSA, BPSO, and MSPSO in minimizing total assembly time and obtaining minimum average time of the ASP problem. The minimum total assembly time obtained by AMSKF is 503.91 unit of time with associated sequence of components suggested by the AMSKF is 1-2-12-4-3-9-13-15-11-5-16-6-18-7-8-14-10-17-19. The average assembly time of AMSKF is 520.19 unit of time and this average value is better than BGSA, BPSO, and MSPSO, which indicate AMSKF’s consistency over 50 runs.
It is true that the finding reported in this paper is much dependent on the parameter values used by other algorithms, which are BGSA, MSPSO, and BPSO. However, in this study, there was no attempt to replicate the experiments of BGSA, MSPSO, and BPSO. All the results were taken from the published paper.

5. Conclusion

The ASP is a combinatorial optimization problem with a large-scale candidate solution. In this study, an approach based on a variant of SKF called AMSKF is proposed to solve ASP problem. To evaluate the performance of the proposed approach, a case study of ASP consisting nineteen components is chosen, and the performance of AMSKF is evaluated against three different approaches that uses BGSA, BPSO, and MSPSO as the search engine. Experimental results obtained showed that the proposed AMSKF outperformed the other three approaches.

In future, the AMSKF could be applied to solve ASP problem with different constraints such as assembly stability, machine and workstation assignment, and work load. Perhaps the experiments reported in this paper can be reimplemented with adjusted parameter values to get better result and more convincing comparison.

Acknowledgement

The researchers are indebted to the Ministry of Higher Education of Malaysia for supporting this research by awarding a Fundamental Research Grant Scheme (FRGS) research grant (RDU170106) to Universiti Malaysia Pahang (www.ump.edu.my). This research is also supported using UMP-Research University grant, (RDU1603109).

References

[1] Z Ibrahim, NH Abdul Aziz, NA Ab Aziz, S Razali, MI Shapiai, SW Nawawi, and MS Mohamad 2015 A Kalman filter approach for solving unimodal optimization problems, ICIC Express Letters, 9(12), 3415-3422.

[2] Z Ibrahim, NH Abdul Aziz, NA Ab Aziz, S Razali, and MS Mohamad 2016, Simulated Kalman Filter: a novel estimation-based metaheuristic optimization algorithm, Advance Science Letters, 22, 2941-2946.

[3] Z Md Yusof, I Ibrahim, Z Ibrahim, KZ Mohd Azmi, NH Abd Aziz, NA Ab Aziz, and MS Mohamad 2016 Angle modulated simulated Kalman filter algorithm for combinatorial optimization problems, ARPN Journal of Engineering and Applied Sciences, 11(7), 4854-4859.

[4] A Adam, Z Ibrahim, N Mokhtar, MI Shapiai, M Mubin, and I Saad 2016 Feature selection using angle modulated simulated Kalman filter for peak classification of EEG signals, SpringerPlus, 5(1580).

[5] YK Choi, DM Lee YB Cho 2008 An approach to multi-criteria assembly sequence planning using genetic algorithms, International Journal of Advanced Manufacturing Technology, 42(1-2), 180-188.

[6] I Ibrahim, Z Ibrahim, H Ahmad, ZM Yusof, MI Shapiai, SW Nawawi, and M Mubin 2014 An assembly sequence planning approach with binary gravitational search algorithm. In: Fujita H., Ali M., Selamat A., Sasaki J., Kurematsu M. (eds) New Trends in Software Methodologies, Tools and Techniques, IOS Press, 179-193.

[7] JAA Mukred, Z Ibrahim, I Ibrahim, A Adam, K Wan, ZM Yusof, and N Mokhtar 2012 A binary particle swarm optimization approach to optimize assembly sequence planning, Advance Science Letters, 13, 732-738.

[8] I Ibrahim, Z Ibrahim, H Ahmad, and ZM Yusof 2016 An assembly sequence planning approach with a multi-state particle swarm optimization. In: Fujita H., Ali M., Selamat A., Sasaki J., Kurematsu M. (eds) Trends in Applied Knowledge-Based Systems and Data Science. IEA/AIE 2016. Lecture Notes in Computer Science, vol 9799. Springer, Cham.