mekatronika – Journal of Intelligent Manufacturing & Mechatronics
VOL. 01, ISSUE 02, 93 – 97
DOI: https://doi.org/10.15282/mekatronika.v1i2.4990

ORIGINAL ARTICLE

Design of an Helical Spring using Single-solution Simulated Kalman Filter Optimizer

[1]M. Abdullah Azzam, [2]Uzma Batool and [3]Hilman Fauzi

[1]Telco Solution Chaosmatic Co. Ltd. Bandung, Indonesia.
[2]Department of Computer Science, University of Wah, Wah Cantt, Pakistan.
[3]Faculty of Electrical Engineering, Telkom University, Bandung, Indonesia.

ABSTRACT –Optimization is one of the important process in solving engineering problems. Regrettably, there are numerous problems in practical optimization that cannot be solved flawlessly within reasonable computational effort. Thus, metaheuristic approach is often useful to get near-optimal solution when the best solution is not achievable. This paper demonstrates the usefullness of a metaheuristic algorithm called single-solution simulated Kalman filter (ssSKF) in helical spring design, which is an example of structural engineering design problem. The ssSKF is a single agent-based optimization algorithm based on the Kalman filtering. The solution obtained by the ssSKF is compared againsts the genetic algorithm, co-evolutionary particle swarm optimization, co-evolutionary differential evolution, bat algorithm, and artificial bee colony.

ARTICLE HISTORY
Received: 2 February 2019
Accepted: 10 Jun 2019

KEYWORDS
Helical spring
Simulated Kalman filter
Design
Optimization

Introduction

Spring is an elastic device used to store mechanical potential energy. The most common design used in everyday life is the coil springs. Spring is used to absorb or controlling energy due to shock and vibration, storing energy, measuring forces and control motion. In automotive industry, helical spring compression with good flexibility can absorb a lot of energy is often used. The efficiency of the vehicle depends on the total weight. Minimizing weight is usually the main goal in designing any automotive components.

Design optimization can be defined as the process of finding the optimal parameters, which yield minimum or maximum value of an objective function and at the same time satisfy a particular set of constraints. This can be solved by an exact method or approximation method. The exact method ensures an optimal solution but the cost of this method can be very huge as the intricacy to compute increases. The approximate method is more efficient in terms of the utilization of time and memory [1]. An approximation method ensures a bounded solution that determines how close the solution acquired from the ideal optimal, however the optimum solution is not always guaranteed, especially for complicated problems. Optimization algorithms are among the well-known approximation methods.

The simulated Kalman filter (SKF) [2] is one of the optimization algorithms that has been developed based on Kalman filtering. The SKF can operates using many agents or one agent only. If many agents are used, the SKF is called population-based SKF [3], whereas if one agent is used, the SKF is called single-solution SKF (ssSKF) [4]. To date, the population-based SKF has been applied in solving many practical problems [5-17], however, applications of ssSKF is still lacking [18]. In this paper, the usefullness of ssSKF is demonstrated by solving a helical spring design problem.

Design of Spring

A spring is an important mechanical component competent of giving massive flexible misshaping. A spring is fundamentally characterized as an elastic body whose work is to misshape when it given load and to recuperate its initial shape structure when the load is detached. It is a mechanical element that exert force when deformed [19].

***CORRESPONDING AUTHOR** | M. Abdullah Azzam | ✉ azzam.cyber@gmail.com

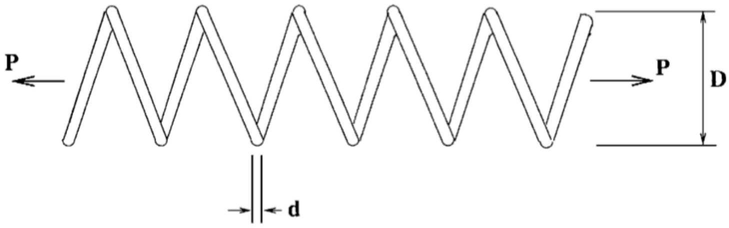

Figure 1. Helical spring.

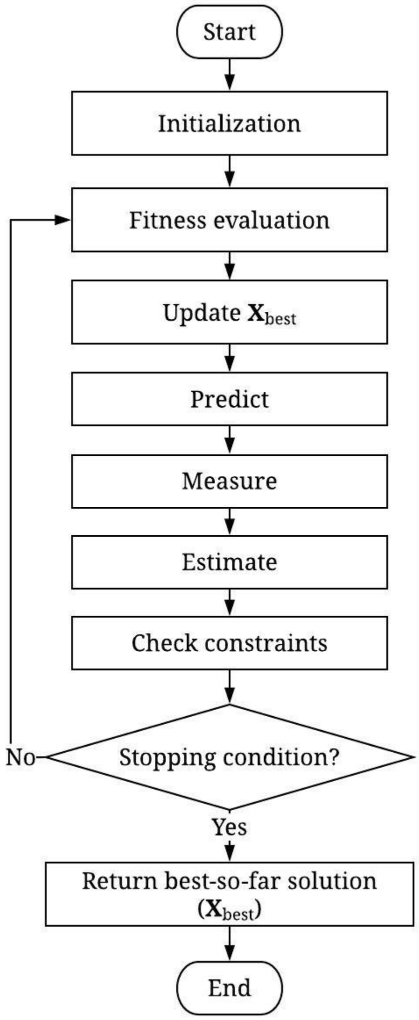

Figure 2. Flowchart of the ssSKF algorithm [4].

Tension/compression string design problem is introduced by Arora [20] and Belegundu [21]. The objective is to minimize the weight of an helical spring shown in Figure 1 [22] subject to a set of constraints, which are minimum deflection, shear stress, surge frequency, limits on outside diameter, and design variables. The mean coil diameter, D, the wire diameter, d, and the number of active coil, P, are the design variables. In this paper, $D = x_2$, $d = x_1$, and $P = x_3$.

To minimize the weight, according to [20-22], an objective function is formulated as follows:

$$\text{Minimize } f(\mathbf{x}) = (x_3 + 2)x_2x_1^2 \tag{1}$$

subject to

$$g_1(x) = 1 - \frac{x_2^3x_3}{71785x_1^4} \leq 0 \tag{2}$$

$$g_2(x) = \frac{4x_2^2 - x_1x_2}{12566(x_2\, x_1^3 - x_1^4)} + \frac{1}{5108x_1^2} - 1 \leq 0 \tag{3}$$

$$g_3(x) = 1 - \frac{140.45x_1}{x_2^2x_3} \leq 0 \tag{4}$$

$$g_4(x) = \frac{x_1 + x_2}{1.5} - 1 \leq 0 \tag{5}$$

The Single-solution Simulated Kalman Filter for the Helical Spring Design

The flowchart of ssSKF algorithm [4] is shown in Figure 2. The algorithm begins with random initial solution, $X(0)$. Initial error covariance, $P(0)$, is set to a normally distributed random number.

After that, fitness according to equation (1) is calculated. After that, the best-so-far solution, $\boldsymbol{X}_{best}$, is updated.

During prediction, the following equations are used to predict the optimum solution:

$$\boldsymbol{X}^d(t|t+1) \sim U\left[\boldsymbol{X}_{best}^d - \delta_t, \boldsymbol{X}_{best}^d + \delta_t\right] \tag{6}$$

$$P^d(t|t+1) = P^d(t) + randn^d \tag{7}$$

$$\delta_t = e^{-\frac{\alpha \times t}{t_{Max}}} \times \delta_0 \tag{8}$$

$$\delta_0 = \max(|\text{lowerlimit}|, |\text{upperlimit}|) \tag{9}$$

where t_{Max} is the maximum number of iterations and $randn^d$ is a normally distributed random number. Initially,

After that in measurement step, the simulated measurement value, $\mathbf{Z}^d(t)$, is computed as follows:

$$\boldsymbol{Z}^d(t) = \boldsymbol{X}^d(t|t+1) + \Delta \tag{10}$$

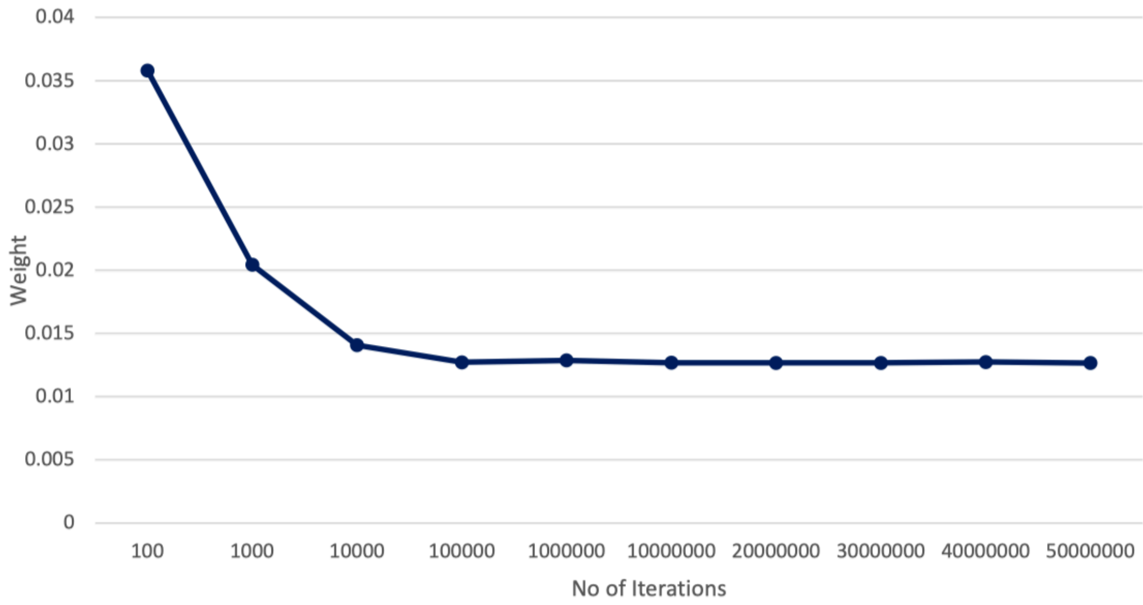

Figure 2. Convergence curve.

Table 1. Experimental setting.

Variable	Value
Maximum iterations	50,000,000
α	10
x_{1min}	0.05
x_{1max}	2
x_{2min}	0.25
x_{2max}	1.3
x_{3min}	2
x_{3max}	15

where

$$\Delta = sin(rand^d \times 2\pi) \times \left|\boldsymbol{X}^d(t|t+1) - \boldsymbol{X}^d_{best}\right| \quad (11)$$

Finally, during the estimation step, the solution and error covariance estimates for the next iteration are calculated as follows.

$$K^d(t) = \frac{P^d(t|t+1)}{P^d(t|t+1) + rand^d} \quad (12)$$

$$\mathbf{X}^d(t+1) = \mathbf{X}^d(t|t+1) + \gamma \quad (13)$$

$$\gamma = K^d(t) \times (\mathbf{Z}^d(t) - \mathbf{X}^d(t|t+1)) \quad (14)$$

$$P^d(t+1) = (1 - K^d(t)) \times P^d(t|t+1) \quad (15)$$

A solution is accepted if all the constraints computed based on equations (2-5) are valid. This process continues until the maximum number of iterations.

Experiment, Result, and Discussion

Table 1 shows the parameter setting of the ssSKF. 50,000,000 iterations is selected. The x_{1min}, x_{1max}, x_{2min}, x_{2max}, x_{3min}, and x_{3max} values follows the design problem introduced in [20-21]. Based on the experimental setting parameters, $f(\boldsymbol{x})$ = 0.0126658 is obtained. The mean coil diameter, D, the wire diameter, d, and the number of active coil, P, are 0.3969, 0.0533, and 9.2981, respectively. A convergence curve is shown in Figure 3.

Table 2 shows the results obtained against other results reported in literature based on different algorithms. Those algorithms are co-evolutionary particle swarm optimization (CPSO) [21], genetic algorithm (GA) [11], artificial bee colony (ABC) [11], co-evolutionary differential evolution (CDE) [11], and bat algorithm (BA) [11]. The comparison shows that the design variables obtained by the ssSKF is better than the rest of the algorithm.

Conclusions

This study finds the minimum weight of a helical spring using ssSKF algorithm. The objective to minimize the weight of the spring is achieved. The performance of the ssSKF have been compared against other algorithm reported in literature the result shows the ssSKF able to outperform co-evolutionary particle swarm optimization (CPSO), genetic algorithm (GA), artificial bee colony (ABC), co-evolutionary differential evolution (CDE), and bat algorithm (BA) in solving helical spring design problem.

Table 2. Results comparison.

Design variables	ssSKF	CPSO [22]	GA [23]	ABC [24]	CDE [25]	BA [26]
d	0.0533	0.051728	0.05148	0.051749	0.051609	0.05169
D	0.3969	0.357644	0.351661	0.358179	0.354714	0.35673
P	9.2981	11.244543	11.6322	11.203763	11.410831	11.2885
$g_1(x)$	-0.0020	−0.000845	−0.003	-0.00	-4e-05	0
$g_2(x)$	-4.7747e04	−1.2600e-05	−0.0001	-0.00	-0.0002	0
$g_3(x)$	-4.1134	−4.051300	−4.0263	-4.056663	-4.0486	−4.0538
$g_4(x)$	-0.6999	−0.727090	−0.7312	-0.726713	-0.7291	−0.7277
$f(x)$	0.0126668	0.0126747	0.0127048	0.012667	0.0126702	0.01267

Conclusions

This study finds the minimum weight of a helical spring using ssSKF algorithm. The objective to minimize the weight of the spring is achieved. The performance of the ssSKF have been compared against other algorithm reported in literature the result shows the ssSKF able to outperform co-evolutionary particle swarm optimization (CPSO), genetic algorithm (GA), artificial bee colony (ABC), co-evolutionary differential evolution (CDE), and bat algorithm (BA) in solving helical spring design problem.

References

[1] Talbi, E. (2009). Metaheuristics. Hoboken, NJ: John Wiley & Sons.

[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] 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.

[4] Abdul Aziz, N.H., Ibrahim, Z., Ab. Aziz, N.A, Mohamad, M.S., and Watada, J. (2018). Single-solution simulated Kalman filter algorithm for global optimisation problems. Sadhana, vol. 43, issue 7, article 103.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] Abdul 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.

[18] 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.

[19] Porteiro, J.L. (2010). Spring Design Optimization with Fatigue. Master Thesis. University of South Flourida.

[20] Arora, J.S. (1989). Introduction to Optimum Design. McGraw-Hill, New York.

[21] Belegundu, A.D. (1982). A study of mathematical programming methods for structural optimization. Department of Civil and Environmental Engineering, University of Iowa, Iowa City, Iowa.

[22] He, Q. and Wang, L. (2007). An effective co-evolutionary particle swarm optimization for constrained engineering design problems. Engineering Applications of Artificial Intelligence, vol. 20, pp. 89-99.

[23] Coello, C.A.C. (2000). Use of a self-adaptive penalty approach for engineering optimization problems. Computers in Industry, vol. 41, no. 2, pp. 113-127.

[24] Akay, B. and Karaboga, D. (2012). Artificial bee colony algorithm for large-scale problems and engineering design optimization. Journal of Intelligent Manufacturing, vol. 23, no. 4, pp. 1001-1014.

[25] Huang, F.Z., Wang, L. and He, Q. (2007). An effective co-evolutionary differential evolution for constrained optimization. Applied Mathematics and Computation, vol. 186, no. 1, pp. 340-356.

[26] Gandomi, A.H., Yang, X.S., Alavi, A.H. and Talatahari, S. (2013). Bat algorithm for constrained optimization tasks. Neural Computing and Applications, vol. 22, no. 6, pp. 1239-1255.