Assembly Operation Optimization Based on Social Radiation Algorithm for Autobody

Yanfeng Xing and Yansong Wang

Automobile Engineering College, Shanghai University of Engineering Science, 333 Longteng Road, Shanghai 201620, China

Correspondence should be addressed to Yanfeng Xing; xyf2001721@163.com

Received 4 December 2013; Accepted 10 January 2014; Published 17 April 2014

Academic Editor: Hongyuan Jiang

Copyright © 2014 Y. Xing and Y. Wang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Assembly dimensional quality affects wind noise and driving steady and whole external appearance. The quality can be improved by reducing part tolerance and fixture tolerance and optimizing key control characteristics (KCCs). However, reducing tolerance should largely increase manufacturing costs, and then the paper assembly tolerance is decreased by selecting optimal KCCs. In this work, a fitness function is presented to evaluate assembly operations based on the linear assembly variation analysis model. Afterwards, a new social radiation algorithm (SRA) is proposed to optimize KCCs, and some test functions are used to evaluate optimum performance between the genetic algorithm (GA) and SRA, and the results show that the performance of SRA is better than that of GA. Finally two cases are used to illustrate process of assembly operation optimization by SRA, and the results show that the SRA has higher precision and efficiency than that of GA.

1. Introduction

The quality of autobody dimension is related to the whole external appearance, wind noise, and even the steady of driving. The dimension quality of autobody is mainly influenced by automobile parts design, assembly process, and manufacturing variations [1]. As the current level of manufacturing precision control is close to the limit, deviation reduced in manufacturing technology demands a qualitative breakthrough. Dimension quality can be improved through selecting assembly sequence, designing fixture layouts, and optimizing key control characteristics (KCCs). According to the analysis of assembly operations, assembly tolerance changes due to different assembly operations, which change with different assembly sequences. Therefore the assembly operation optimization is an important means for improving the assembly dimensional quality [2–4].

Currently, the assembly operation was optimized through the genetic algorithm (GA) and the improved genetic algorithm (IGA) [5, 6]. Carlson [7] used the secondary sensitivity equation to optimize fixture scheme, and Vishnupriyan et al. applied GA to select the best fixture locator positions [8]. For overcoming the disadvantages of GA, Xing and Wang [9] proposed the hybrid algorithm of particle swarm optimization and genetic algorithm to improve optimum efficiency. Since the current optimization algorithms are not able to converge to the global optimal solutions, this paper proposes a new social radiation algorithm (SRA) to improve optimization precision and efficiency for increasing the probability of searching global solutions. This method is proposed based on human competition in society. SRA is a swarm intelligence optimization algorithm according to the principle of survival of the fittest. The innovation points of SRA are that the population is divided into different subgroups and resorted at every generation.

2. Objective Function of Optimizing Assembly Operations

Many scholars had been studying assembly variation propagation since 1900s. Cai et al. [10] proposed a method of linear assembly variation analysis model to calculate assembly variation due to fixture deviation. Qin et al. [11] put forward a method of assembly variation analysis based on rigid model. Through calculating assembly variation of 2D (2013), the paper makes use of linear assembly variation analysis model [10] to evaluate assembly operation. The translational
and orientational variations at center of workpiece can be expressed as a vector:

$$\delta \mathbf{q}_i = -J^{-1}_f \Phi_R \delta \mathbf{R}_f,$$

(1)

where the Jacobian is expressed as

$$J_i = \begin{bmatrix} -n_{ix} & -n_{iy} & n_{iy} x_i & -n_{ix} y_i \end{bmatrix}, \quad i = 1, 2, 3,$$

$$\Phi_R = \begin{bmatrix} n_{1x} & n_{1y} & 0 & 0 & 0 & 0 \\ 0 & 0 & n_{2x} & n_{2y} & 0 & 0 \\ 0 & 0 & 0 & n_{3x} & n_{3y} \end{bmatrix}.$$

(2)

Here, $\delta \mathbf{R} \equiv [\delta x_1, \delta y_1, \delta x_2, \delta y_2, \delta x_3, \delta y_3]^T$ represents the source variation vector, which can be calculated by multiplying deviation and direction at locating point. $n_i \equiv [n_{ix}, n_{iy}]^T, i = 1, 2, 3$ stands for the surface normal vector at locating points.

According to (1), the coefficient of variation propagation is related to assembly operations. The smaller the coefficient is, the better the assembly operations are. Based on optimization of KCCs [12], the Euclidean norm of inverse Jacobian matrix is used as the objective function to evaluate assembly operation, which is shown in (3);

$$F(\mathbf{X}) = \|J^{-1}(\mathbf{X})\|,$$

(3)

where $\mathbf{X}$ represents selected KCCs and $\|J^{-1}(\mathbf{X})\|$ represents the Euclidean norm of inverse Jacobian matrix.

3. Social Radiation Algorithm

SRA is proposed in this section, which is a swarm intelligence algorithm based on the development of human society. The algorithm shares similar characteristics to GA; however, the manner in which the two algorithms traverse the search space is fundamentally different. The algorithm randomly generates different capabilities and radiations for a population of individuals at first. Each individual can change its capability and radiation by the last population. In SRA, the population is divided into different subgroups through objective function. The individual can change its subgroups by updating its capability and radiation. The optimization of assembly operations can be divided into the following steps: (1) codifying KCCs to generate initial population, (2) generating new individuals by updating capabilities and radiations, and (3) optimizing assembly operations based on the SRA.

3.1. Codifying KCCs. According to [12], the positions of KCCs can be expressed by a continuous function for simple workpiece. However, the KCCs can be codified by setting consecutive number of discrete characteristics for complex assemblies. For example, an array (20-25-32-38-43-50) stands for a combination of KCCs. The numbers of the array must not overlap. Although the numbers of the KCCs are not overlapped, these KCCs maybe not satisfy the principle of six-point location, which means that the rank of Jacobian (see (1)) is less than six. At this time, a big value can be given for the set of KCCs. Then (3) can be written as follows:

$$F(\mathbf{X}) = \begin{cases} \|J^{-1}(\mathbf{X})\|, & \text{Rank } (J) = 6, \\
10000, & \text{Rank } (J) < 6. \end{cases}$$

(4)

3.2. Updating Individual Capability and Radiation. The section will present how to generate a new individual population. The new population can be generated according to the following steps.

Step 1. This includes calculating individual radiation $R_{d_{i}^{g+1}}$, which is related to the radiation of current iteration, the global best value, and the subgroup best values. The equation can be written as

$$R_{d_{i}^{g+1}} = \omega R_{d_{i}^{g}} + (sBPA_{j} - PA_{i}^{g}) r_{1} + (gBPA - PA_{i}^{g}) r_{2}.$$

(5)

Here, $\omega$, $(F_{\text{best}} - F_{\text{worst}}) / F_{\text{worst}}$, represents the weight of current radiation, which can be calculated by the best value $F_{\text{best}}$ and the worst value $F_{\text{worst}}$. $PA_{i}^{g}$ means the individual capability, $sBPA_{j}$ means the best individual capability of subgroup (j means the different subgroup), and $gBPA$ stands for the global best individual capability. $r_{1}$ and $r_{2}$ represent random influence in [0, 1]. $g$ means the evolution iteration.

Step 2. This includes updating temp individual capability by adding current capability and new radiation, which is shown

$$PA_{\text{temp},i}^{g+1} = PA_{i}^{g} + R_{d_{i}^{g+1}}.$$

(6)
Table I: Comparing results of three functions.

| Function | Theory | GAOT | SRA |
|----------|--------|------|-----|
| De Jong F1 | | | |
| Iterations | 100 | 99 |
| Variables | 0.0, 0.0, 0.0 | 2.6e−03, −1.4e−03, 5.9e−03 | 1.119e−23, 4.378e−25, −9.017e−24 |
| The best value | 0.0 | 4.26e−5 | 2.066e−46 |
| De Jong F2 | | | |
| Iterations | 97 | 99 |
| Variables | 1.0, 1.0 | 1.207, 1.458 | 1.0, 1.0 |
| The best value | 0.0 | 0.043 | 1.11e−10 |
| De Jong F5 | | | |
| Iterations | 32 | 20 |
| Variables | −32.0, −32.0 | −31.98, −31.98 | −31.98, −31.98 |
| The best value | 0.998 | 0.998 | 0.998 |
| Shaffer F6 | | | |
| Iterations | 56 | 98 |
| Variables | 0.0, 0.0 | 2.76, −1.495 | −1.6e−03, −2.0e−05 |
| The best value | 0.0 | 0.972 | 2.42e−04 |

Step 3. This includes updating individual capability. The new individual can be defined by the following equation according to objective function:

\[
\text{Min} : PA_{g+1}^i = \begin{cases} 
PA_{g}^{(temp)_i}, & \text{if } f(\text{PA}_{g}^{(temp)_i}) < f(\text{PA}_{g}^i), \\
PA_{g}^i, & \text{if } f(\text{PA}_{g}^{(temp)_i}) \geq f(\text{PA}_{g}^i). 
\end{cases}
\] (7)

3.3. Flowchart of SRA. SRA can solve real problems by swarm intelligence. In each iteration, the individual capability is updated by the radiation, the subgroup best solution, and the global best solution. The initial population is divided into different subgroups according to their fitness value, and then the final optimal results are generated by continuous updating individual capabilities and radiations. The flowchart of SRA is shown as Figure 1.

It is assumed that the number of individuals is \( N \) in \( D \) dimensional space. The steps of SRA are shown as follows.

Step 1 (initializing parameters). This includes (1) setting the maximum iterations, size of population, and the number of subgroups; (2) initializing individual capability and radiation \( PA_{g}^i \) and \( Rd_{g}^i \), \( i = 1, 2, \ldots, N \); (3) setting counter \( g = 0 \).

Step 2 (evaluating individuals). This includes (1) judging the feasibility of individual through rank of Jacobian matrix; (2) calculating fitness values by (4); the smaller the fitness value is, the better the assembly operation is.

Step 3 (judging whether to stop). If the step condition is met, the optimization will be eliminated; otherwise the flowchart will go to Step 4.

Step 4 (generating new populations). This includes (1) sorting all individuals and dividing into different subgroups and (2) updating individual capability and radiation from (5) to (7).

Step 5. This includes setting \( g = g + 1 \) and repeating from Step 2 to Step 5.

3.4. Optimum Performance of SRA. The GA proposed by Holland as a global optimization algorithm had been widely used in different fields from engineering to social science since 1970s. However, users need to write GA program for specific problems and then want to have a general GA program. The GAOT package in MATLAB can satisfy the requirements [13]. The main program of GAOT provides the external interface and the function, which can be written as follows:

\[
[P, \text{endPop}, \text{bPop}, \text{traceInfo}] = \text{ga} (\text{bounds, evalFN, evalOps, startPop, opts, termFN, termOps, selectFN, selectOps, xOverFNs, xOverOps, mutFNs, mutOps}).
\]

\( P \) stands for the best solution including chromosome and fitness value, \( \text{endPop} \) means the last population, \( \text{bPop} \) represents trace of best chromosomes, and \( \text{traceInfo} \) is the best and average fitness of each generation.

It is very difficult to evaluate performance of optimization algorithms because real problems are very complex. Some scholars had proposed some test functions to analyze the optimization performance. In this paper, De Jong \( F_1, F_2, \) and \( F_5 \) and Shaffer \( F_6 \) are used to compare results of GA and SRA.

De Jong \( F_1 \)

\[
f_1(x_1, x_2, x_3) = \sum_{i=1}^{3} x_i^2, \quad -512 \leq x_i \leq 512, \quad i = 1, 2, 3.
\]
This is a simple sum of squares function and it has a minimum point only $f_1(0, 0, 0) = 0$.

**De Jong F2**

$$f_2(x_1, x_2) = 100(x_1^2 - x_2)^2 + (1 - x_1)^2,$$

$$-2.048 \leq x_i \leq 2.048, \ i = 1, 2.$$  \hspace{1cm} (9)

This is a two-dimensional function and it has a global minimum point $f_2(1, 1) = 0$. The function is a single peak function, but a morbid function. Thus the global minimum value is difficult to get.

**De Jong F5**

$$f_5(x_1, x_2) = \frac{1}{0.002 + \sum_{j=1}^{25} \left( \frac{1}{j + \sum_{i=1}^{2} (x_i - a_{ij})^5} \right)},$$

$$-65.536 \leq x_i \leq 65.635, \ i = 1, 2,$$

$$[a_{ij}] = \begin{bmatrix} -32 & -16 & 0 & 16 & 32 & -32 & -16 & \cdots & -16 & 0 & 16 & 32 \\ -32 & -32 & -32 & -32 & -32 & -16 & -16 & \cdots & 32 & 32 & 32 & 32 \end{bmatrix}.$$  \hspace{1cm} (10)
This is a multipeak function that has 25 local minimum points. There is a global minimum value $f_5(-32, -32) = 0.998$.

**Shaffer F6**

$$f_6(x_1, x_2) = 0.5 + \frac{\sin^2 \sqrt{x_1^2 + x_2^2} - 0.5}{\left[1.0 + 0.001 (x_1^2 + x_2^2)\right]^2}, \quad -100 \leq x_i \leq 100, \quad i = 1, 2.$$  

(11)

This function is a global minimized value $f_6(0, 0) = 0$.

The program of SRA is written in MATLAB. The SRA is used to optimize the three functions and compare with the results of GAOT. The initial parameters are the same, in which the population sizes are both 50 and the maximum iterations are 100. The number of subgroups is 4. The comparative results are shown in Figure 2.

Figure 2 shows the evolution processes of the best chromosomes for the above three test functions. The abscissa is evolution generations and the vertical axis stands for fitness values. The grey diamonds mean the best values of each generation by using GAOT, and the hollow rounds are the best values of each generation through using SRA. Table 1 shows the comparing results of the best value and iterations of the best chromosomes firstly. According to Figure 2 and Table 1, SRA has better performance than that of GAOT for a single peak and multipeak functions. Especially for De Jong F1 and F2 and Shaffer F6, the optimal solutions are almost equal to the best variables of theory. Based on the above test functions, SRA has better optimum performance.

### 4. 2D Rectangular Parts

A case of 2D rectangle workpiece is applied to analyze the optimization problem of KCCs (as shown in Figure 3). The workpiece is located by three points, whose positions and directions are $(x_i, y_i, n_{ix}, n_{iy}), \ i = 1, 2, 3$ respectively. The $[\delta x_1, \delta y_1, \delta x_2, \delta y_2, \delta x_3, \delta y_3]^T$ represents the source variation vector at locating points.

The size of the case is $50 \text{ mm} \times 30 \text{ mm}$. The line LD equation can be written as follows:

$$y = 0.5x, \quad 0 \leq x \leq 44.7.$$  

(12)

The line WL equation is expressed as

$$y = -2x, \quad -13.41 \leq x \leq 0.$$  

(13)

In this section, the Euclidean norm of inverse Jacobian matrix is as an objective function. Based on the rules of locating points, locators $L_1$ and $L_2$ are in the line LD, whose directions are both $(0.447, 0.894)$. Locator $L_3$ is set to in the line WL, whose direction is $(0.894, 0.447)$. SRA is written by MATLAB. We can set the fact that the population size is 50 and the maximum generations are 100 for GAOT and SRA. The number of subgroups is equal to 4. According to the flowcharts of GAOT and SRA, the results can be shown in Figure 4.

From Figure 4, the abscissa is evolution generations and the vertical axis stands for fitness values. The gray diamond means the fitness values by GAOT. The hollow circles represent the best values through SRA. Figure 4 shows that the evolution of the genetic algorithm uses 30 generations to get the optimal value of 5.56309, and SRA uses 5 iterations to generate the best solution. The coordinates and directions of three locating points are $(44.7, 22.35, 0.447, 0.894)$, $(0, 0, 0.447, 0.894)$, and $(0, 0, 0.894, 0.447)$. The best coordinates and directions are the same to the results of GAOT. Thus, SRA is suitable to solve the assembly operation optimization. The optimization efficiency and accuracy can be improved by setting appropriate coefficients.
5. Case of Side Frame Assembly

The side frame is an important subassembly in autobody, which affects assembly quality of the whole body, gap/flush between front door and rear door, and gap/flush between fender and front door. The side frame includes three parts, which is shown in Figure 5.

The first part P1 is the base part, and then part P2 and part P3 are assembled on P1 in turn. The matching zones of P2 and P3 are so complex that the zones are hard to be expressed by equations. Therefore the study creates some locating points to optimize KCCs. The 35 locator points are generated in main plane (normal to y direction) of P2, and in other directions (x and z) we create both 15 points (Figure 6(a)).

According to flowchart of SRA, the objective function of assembly operation optimization uses Euclidean norm of inverse Jacobian matrix. Based on the rules of locating points [12], three points are selected in main plane, whose directions are all (0, 1, 0). Other points are generated from other directions. The population size is 80 and the maximum iterations equal 100. The number of subgroups is set to 4. The results are shown as Figures 6(b) and 6(c).

Based on [12], the objective value of the best assembly operation is 142.1 by the fruit fly optimization algorithm, and
the locator points are (15, 20, 2, 37, 51, 62). In this work, the best value is 132.76 through SRA, and the best locator points are (20, 2, 15, 64, 50, 51). Although the best value is decreased by 9.34, the solution of SRA is the best true solution by enumeration method.

According to the above steps, the KCCs of P3 can be optimized by SRA. The parameters of SRA are the same to the above parameters. The results are shown in Figure 7. The best value is 125.4 by the fruit fly optimization algorithm, and the best fitness value is 122.9 through SRA and enumeration method. SRA shows better efficiency to optimize assembly operations for compliant parts.

6. Conclusions

This paper proposed a new method of social radiation algorithm to optimize assembly operation between parts through the linear assembly variation analysis model. The four test functions are used to compare optimum performance between SRA and GAOT, and the results show that the new algorithm of this paper has higher efficiency and precision. Afterwards, the algorithm is applied to optimize KCCs of 2D rectangle workpiece. Finally, the case of side frame is used to illustrate the process of optimizing assembly operations by SRA. The proposed social radiation optimization algorithm not only can improve the efficiency and accuracy of optimization of the assembly control features but also can be used for some other problems of swarm intelligence.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by Shanghai Natural Science Foundation (11ZR1414700) and the National Natural Science Foundation of China (51105241).

References

[1] S. Dahlstrom and L. Lindkvist, “Variation simulation of sheet metal assemblies using the method of influence coefficients with contact modeling,” Journal of Manufacturing Science and Engineering, vol. 129, no. 3, pp. 615–622, 2007.
[2] J. A. Camellio, S. J. Hu, and S. P. Marin, “Compliant assembly variation analysis using component geometric covariance,” Journal of Manufacturing Science and Engineering, vol. 126, no. 2, pp. 355–360, 2004.
[3] X. Liao and G. G. Wang, “Employing fractals and FEM for detailed variation analysis of non-rigid assemblies,” International Journal of Machine Tools and Manufacture, vol. 45, no. 4-5, pp. 445–454, 2005.
[4] X.-M. Lai, Y.-F. Xing, J. Sun, and G.-L. Chen, “Optimisation of assembly sequences for compliant body assemblies,” International Journal of Production Research, vol. 47, no. 21, pp. 6129–6143, 2009.
[5] Y. Xing, X. Lai, S. Jin, Z. Xu, and G. Chen, “Application of improved genetic algorithm to assembly operations optimization,” Journal of Computer-Aided Design and Computer Graphics, vol. 19, no. 10, pp. 1298–1305, 2007.
[6] Y.-F. Xing, G.-L. Chen, X.-M. Lai, Y.-B. Li, and S. Jin, “Selecting optimal assembly operations using genetic algorithm,” Journal of Shanghai Jiaotong University, vol. 42, no. 2, pp. 223–226, 2008 (Chinese).
[7] J. S. Carlson, “Quadratic sensitivity analysis of fixtures and locating schemes for rigid parts,” Journal of Manufacturing Science and Engineering, vol. 123, no. 3, pp. 462–472, 2001.
[8] S. Vishnupriyan, M. C. Majumder, and K. P. Ramachandran, “Optimization of machining fixture layout for tolerance requirements under the influence of locating errors,” International Journal of Engineering Science and Technology, vol. 2, no. 1, pp. 152–162, 2010.
[9] Y. F. Xing and Y. S. Wang, “Assembly operation optimization based on hybrid particle swarm optimization and genetic algorithm,” Computer Integrated Manufacturing System, vol. 18, no. 4, pp. 747–753, 2012 (Chinese).
[10] C. O. Cai, S. J. Hu, and J. X. Yuan, “A variational method of robust fixture configuration design for 3-D workpieces,” Journal of Manufacturing Science and Engineering, vol. 119, no. 4, pp. 593–601, 1997.
[11] G. Qin, W. Zhang, Z. Wu, and M. Wan, “Systematic modeling of workpiece-fixture geometric default and compliance for the prediction of workpiece machining error,” Journal of Manufacturing Science and Engineering, vol. 129, no. 4, pp. 789–801, 2007.
[12] Y. F. Xing, “Design and optimization of key control characteristics based on improved fruit fly optimization algorithm,” Kybernetes, vol. 42, no. 3, pp. 466–481, 2013.
[13] A. J. Chipperfield and P. J. Fleming, “The MATLAB genetic algorithm toolbox,” in Proceedings of the IEE Colloquium on Applied Control Techniques Using MATLAB, pp. 10.1–10.4, January 1995.