Frontiers of Information Technology & Electronic Engineering
www.zju.edu.cn/jzus; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Multi-objective layout optimization of a satellite module using the Wang-Landau sampling method with local search*

Jing-fa LIU[1,2], Liang HAO[†‡1,2], Gang LI[3], Yu XUE[1,2], Zhao-xia LIU[4], Juan HUANG[1,2]

([1]*Jiangsu Engineering Center of Network Monitoring, Nanjing University of Information Science & Technology, Nanjing 210044, China*)

([2]*School of Computer & Software, Nanjing University of Information Science & Technology, Nanjing 210044, China*)

([3]*School of Mathematics and Statistics, Nanjing University of Information Science & Technology, Nanjing 210044, China*)

([4]*Office of Informationization Construction and Management, Nanjing University of Information Science & Technology, Nanjing 210044, China*)

[†]E-mail: haoliang90518@163.com

Received Sept. 6, 2015; Revision accepted Feb. 28, 2016; Crosschecked May 10, 2016

Abstract: The layout design of satellite modules is considered to be NP-hard. It is not only a complex coupled system design problem but also a special multi-objective optimization problem. The greatest challenge in solving this problem is that the function to be optimized is characterized by a multitude of local minima separated by high-energy barriers. The Wang-Landau (WL) sampling method, which is an improved Monte Carlo method, has been successfully applied to solve many optimization problems. In this paper we use the WL sampling method to optimize the layout of a satellite module. To accelerate the search for a global optimal layout, local search (LS) based on the gradient method is executed once the Monte-Carlo sweep produces a new layout. By combining the WL sampling algorithm, the LS method, and heuristic layout update strategies, a hybrid method called WL-LS is proposed to obtain a final layout scheme. Furthermore, to improve significantly the efficiency of the algorithm, we propose an accurate and fast computational method for the overlapping depth between two objects (such as two rectangular objects, two circular objects, or a rectangular object and a circular object) embedding each other. The rectangular objects are placed orthogonally. We test two instances using first 51 and then 53 objects. For both instances, the proposed WL-LS algorithm outperforms methods in the literature. Numerical results show that the WL-LS algorithm is an effective method for layout optimization of satellite modules.

Key words: Packing, Layout design, Satellite module, Wang-Landau algorithm
http://dx.doi.org/10.1631/FITEE.1500292 **CLC number:** TP391; V474

1 Introduction

The layout design of a satellite module involves placing a certain number of objects, including various instruments and devices, in a particular satellite module, while satisfying various constraints with specific objectives. Stability and service life are essential requirements of a successful layout design, and other performances of the whole satellite module system should also be considered.

Due to engineering and mathematics complexities, satellite module layout design is known as an NP-hard problem. It stems from the bin packing problem (Crainic *et al*., 2011; Gonçalves and Resende, 2011; Khanafer *et al*., 2012), which can mainly be classified into two categories according to the study subjects: (1) 2D bin packing problem; (2) 3D bin packing problem. For the 2D bin packing problem, Gonçalves and Resende (2011) proposed a parallel multi-population genetic algorithm that hybridizes a novel placement procedure with a genetic algorithm based on random keys. Khanafer *et al*. (2012)

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (Nos. 61373016 and 61403206), the Six Talent Peaks Project of Jiangsu Province, China (No. DZXX-041), the Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions, and the Natural Science Foundation of Jiangsu Province, China (No. BK20141005)

ORCID: Liang HAO, http://orcid.org/0000-0003-0824-2656

proposed a tree decomposition method based heuristic strategy to decompose the original problem into several sub-problems. Li *et al*. (2012) proposed a heuristic particle swarm optimization approach with a quasi-human strategy. Zhang *et al*. (2005) proposed a hybrid heuristic algorithm based on divide-and-conquer and greedy strategies for the 2D rectangular packing problem. Liu *et al*. (2009) proposed an improved energy landscape paving (ELP) method by incorporating a new configuration update mechanism into the ELP method to solve the circular packing problem. For the 3D bin packing problem, Martello *et al*. (2000) proved that the asymptotical worst-case performance of the continuous lower bound is 1/8, and presented an exact branch-and-bound algorithm. Moon and Nguyen (2014) presented a mixed integer programming model which features upper and lower bounds for the 3D bin packing problem. To deal with the 3D parallelepiped bin packing problem, Allen *et al*. (2011) proposed a hybrid placement strategy, and the test results showed that it outperforms other methods from the literature.

In this paper, we study the layout design of a satellite module with performance constraints that can be considered a special case of the bin packing problem mentioned above. Various methods have been proposed to deal with the layout design of satellite modules. Sun and Teng (2003) proposed a two-stage layout method, the centripetal balancing method, for global layout design in the first stage, and an ant colony optimization algorithm for detailed layout design of a satellite module in the second stage. Zhang *et al*. (2008) presented a hybrid method by combining soft computing techniques, including the Hopfield neural network, genetic algorithm/particle swarm optimization, and quasi-principal component analysis. Teng *et al*. (2010) developed a dual-system framework based on the cooperative coevolutionary genetic algorithm with the merits of great population diversity and a decrease in premature convergence. He *et al*. (2013) proposed a quasi-physical algorithm based on coarse and fine adjustment. Tang and Teng (1999) presented a decimal coded adaptive genetic algorithm, which decreases combinatorial explosion and premature convergence. Wang and Teng (2009) proposed a knowledge fusion approach, which harnesses the potential of both humans and computers through evolutionary computation. Liu and Teng (2008) presented a human algorithm knowledge layout design (HAKD) method, which fuses human intelligence, computer intelligence (evolution algorithm), and prior knowledge (relevant layout diagrams) at the gene level of the evolution algorithm. Huo and Teng (2009) optimized a layout design using four steps. First, the whole layout problem was decomposed into several sub-layout problems. Second, a relaxation model was adopted to distribute all objects among subspaces. Third, a coevolutionary genetic algorithm was adopted to solve sub-layout problems. Finally, a heuristic combination-rotation method was adopted to adjust the constraints to obtain the final layout. Lei and Qiu (2006) presented a novel adaptive particle swarm optimizer based on multi-modified strategies, which can not only escape from the attraction of local optima, but also maintain the characteristic of a fast speed search in the early convergence phase. Zhou *et al*. (2005) presented a constraint handling strategy suitable for particle swarm optimization, and used direct search to intensify its local search ability. Jin and Teng (2007) proposed a case retrieval algorithm for reusing previously stored design solutions and optimizing layout design using prior knowledge and an evolutionary approach. Chen *et al*. (2008) presented an improved differential evolution algorithm, which makes random micro-perturbations to the location of components to obtain a better layout scheme. Zhang *et al*. (2013) proposed a parallel dual-system cooperative coevolutionary differential evolution algorithm with human-computer cooperation. Xu and Xiao (2008) presented an ant colony optimization algorithm based on a step-by-step positioning technique. Liu *et al*. (2011) proposed a heuristic algorithm based on tabu search, where tabu search was used to jump out of the local minima. Liu and Li (2010) proposed the basin filling algorithm by combining the improved energy landscape paving method, the gradient method, and the configuration update strategy. Liu *et al*. (2010) proposed a heuristic simulated annealing algorithm by incorporating the heuristic neighborhood search mechanism and the adaptive gradient method into the simulated annealing procedure.

Although the approaches mentioned above enable effective layout design of the satellite module, their efficiency still needs to be improved. In fact, a reasonable combination of the stochastic algorithm

with global search ability, the local accurate search method, and some heuristic strategies may be an effective way to construct a high-performance algorithm for layout optimization of satellite modules. The Wang-Landau (WL) sampling algorithm (Wang and Landau, 2001; Landau *et al*., 2004) is an improved Monte Carlo algorithm. Unlike conventional Monte Carlo simulations that generate a probability distribution at a given temperature, the WL sampling method can estimate the density of states accurately via a random walk, which produces a flat histogram in the energy space. There have been many improvements on and applications of the WL sampling algorithm. For example, Zhou and Bhatt (2005) proved the convergence of the WL sampling algorithm, and found that the histogram increases uniformly with small fluctuations after a stage of initial accumulation. Seaton *et al*. (2010) used it to describe the thermodynamic behavior of a continuous homopolymer. Schulz *et al*. (2003) proposed a simple modification of the WL sampling algorithm. This modification removes the systematic error that occurs at the boundary of the range of energy over which the random walk takes place. The greatest challenge of the layout optimization of the satellite module is that the function to be optimized is characterized by a multitude of local minima separated by high-energy barriers. The WL sampling method can visit all the accessible states of the system, which means that it can jump out of these high-energy barriers. Therefore, the WL sampling method is an ideal global search algorithm for layout optimization of satellite modules. In this paper, we put forward a hybrid method called WL-LS by incorporating the WL sampling algorithm, the local search (LS) method, and the heuristic layout update strategies to deal with the layout optimization of a satellite module with performance constraints. Numerical results show that the proposed WL-LS algorithm is an effective method in designing the layout of satellite modules.

2 Problem statement

A simplified model of the international commercial communication satellite module (INTELSAT-III) (Teng *et al*., 2010) can be described as follows. The satellite module consists of several parts including a cylindrical container, a standing column, two bearing plates, and N objects which need to be packed. The column is located at the center of the module, perpendicular to the base surface of the satellite module (Fig. 1a). The objects can be fixed on either the top or bottom surfaces (P_1–P_4) of both plates (Fig. 1b). So, there are four surfaces available for locating these N objects, A_i (i=1, 2, …, N). In this study, objects are considered to be homogeneous rigid bodies, and are simplified as cuboids and cylinders. The two circular bearing plates have the same thickness, mass, and radius R_0. The objective of the layout optimization of the satellite module is to make the radius of the satellite module and the inertia moments of the whole system as small as possible. The final layout needs to meet the following constraints: (1) All the objects

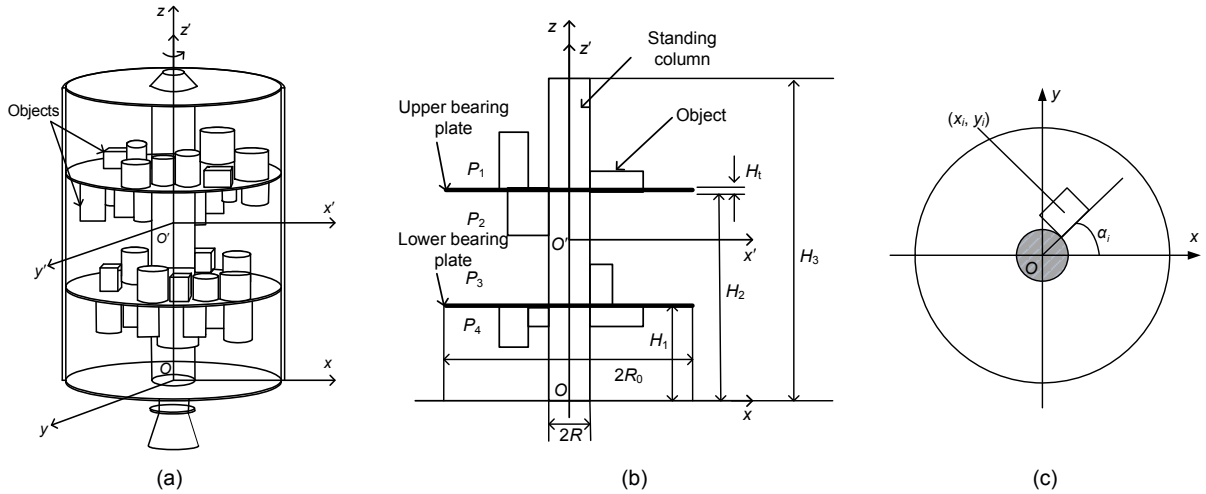

Fig. 1 Diagrammatic sketch of the simplified satellite layout: (a) 3D diagrammatic sketch of the layout; (b) 2D diagrammatic sketch of the layout; (c) layout of one object on a bearing plate

should be placed within the layout space and not overlap each other; (2) The error between the centroid of the whole system and the expected centroid should not exceed an allowable value, and be as small as possible; (3) The equilibrium degree error of the whole system should not exceed an allowable value, and be as small as possible.

In this paper, to describe clearly the mathematical model of the problem, we adopt four coordinate systems introduced by Sun and Teng (2003): $Oxyz$, $O'x'y'z'$, $O''x''y''z''$, and $O'''x'''y'''z'''$ (Fig. 2). In coordinate system $Oxyz$, the xOy plane coincides with the base surface of the satellite module, and the origin O is its geometric center. The upward direction of the z axis is from the satellite's base to top. The $x'O'y'$ plane and the xOy plane are parallel, and the origin O' of coordinate system $O'x'y'z'$ is the centroid of the whole satellite module. The z' axis is parallel or coincides with the z axis, and they have the same direction. Coordinate system $O''x''y''z''$ is used to calculate the inertia moments of objects in relation to their own axes. The origin O'' is the centroid of the object, and the x'', y'', and z'' axes are the geometric symmetry axes of the object. The coordinate system $O'''x'''y'''z'''$ is used to calculate the inertia angles of the whole system. The origin O''' coincides with O', and the x''', y''', and z''' axes have angles ($\theta_{x'}$, $\theta_{y'}$, $\theta_{z'}$) with the x', y', and z' axes. The counter-clockwise direction is positive.

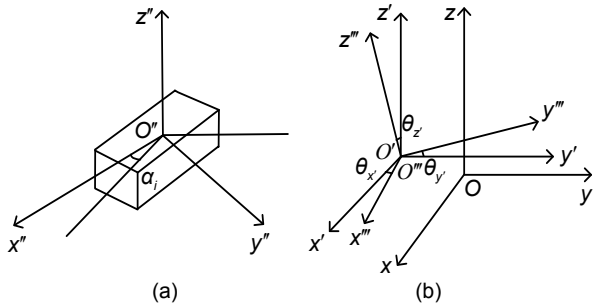

Fig. 2 Sketch map of coordinate systems: (a) coordinate system $O''x''y''z''$; (b) coordinate systems $Oxyz$, $O'x'y'z'$, and $O'''x'''y'''z'''$

The mathematical model of the layout optimization problem discussed above can be described as follows:

$$\min f_1(X), \tag{1}$$

$$\min f_2(X)=J_{x'}(X)+J_{y'}(X)+J_{z'}(X) \tag{2}$$

subject to

1. non-overlapping constraints:

$$g_1(X)=\sum_{i=0}^{N-1}\sum_{l=i+1}^{N}\mathrm{int}(A_i)\cap\mathrm{int}(A_l)=\varnothing, \tag{3}$$

2. centroid error constraints:

$$g_2(X)=|x_c-x_e|\le 3.0, \tag{4}$$

$$g_3(X)=|y_c-y_e|\le 3.0, \tag{5}$$

$$g_4(X)=|z_c-z_e|\le 3.0, \tag{6}$$

3. equilibrium degree error constraints:

$$g_5(X)=|\theta_{x'}(X)|\le 0.03, \tag{7}$$

$$g_6(X)=|\theta_{y'}(X)|\le 0.03, \tag{8}$$

$$g_7(X)=|\theta_{z'}(X)|\le 0.03. \tag{9}$$

Here $X=(x_1, y_1, z_1, \alpha_1, \ldots, x_i, y_i, z_i, \alpha_i, \ldots, x_N, y_N, z_N, \alpha_N)$; α_i are the orientation angles for the rectangles shown in Fig. 1c (for the circles, α_i do not exist); N is the number of layout objects; (x_i, y_i, z_i) are the 3D coordinates of object A_i; $f_1(X)$ is the space utilization function, which is measured by radius R_0 of the satellite module; $f_2(X)$ is the inertia moment of the whole system; $J_{x'}(X)$, $J_{y'}(X)$, and $J_{z'}(X)$ are inertia moments of the whole system with respect to the coordinate system $O'x'y'z'$; $\mathrm{int}(A_i)$ denotes the internal part of object A_i; (x_e, y_e, z_e) is the expected centroid of the whole system; (x_c, y_c, z_c) is the real centroid position of the whole system; $\theta_{x'}(X)$, $\theta_{y'}(X)$, and $\theta_{z'}(X)$ (Fig. 2) are the angles between the principal inertia axes of the module and the coordinate system $O'x'y'z'$ axes.

$J_{x'}(X)$, $J_{y'}(X)$, and $J_{z'}(X)$ are calculated as follows:

$$J_{x'}(X)=\sum_{i=0}^{N}(J_{x''i}\cos^2\alpha_i+J_{y''i}\sin^2\alpha_i)+\sum_{i=0}^{N}m_i(y_i^2+z_i^2)-(y_c^2+z_c^2)\sum_{i=0}^{N}m_i, \tag{10}$$

$$J_{y'}(X)=\sum_{i=0}^{N}(J_{y''i}\cos^2\alpha_i+J_{x''i}\sin^2\alpha_i)+\sum_{i=0}^{N}m_i(x_i^2+z_i^2)-(x_c^2+z_c^2)\sum_{i=0}^{N}m_i, \tag{11}$$

$$J_{z'}(X)=\sum_{i=0}^{N}J_{z''i}+\sum_{i=0}^{N}m_i(x_i^2+y_i^2)-(x_c^2+y_c^2)\sum_{i=0}^{N}m_i, \tag{12}$$

where $J_{x''i}$, $J_{y''i}$, and $J_{z''i}$ are inertia moments of the ith

object with respect to coordinate system $O''x''y''z''$, and m_i is the mass of the ith object.

$\theta_{x'}(X)$, $\theta_{y'}(X)$, and $\theta_{z'}(X)$ are calculated as follows:

$$\theta_{x'}(X)=\frac{1}{2}\arctan\left(\frac{2J_{x'y'}(X)}{J_{x'}(X)-J_{y'}(X)}\right), \tag{13}$$

$$\theta_{y'}(X)=\frac{1}{2}\arctan\left(\frac{2J_{x'z'}(X)}{J_{z'}(X)-J_{x'}(X)}\right), \tag{14}$$

$$\theta_{z'}(X)=\frac{1}{2}\arctan\left(\frac{2J_{y'z'}(X)}{J_{z'}(X)-J_{y'}(X)}\right), \tag{15}$$

where $J_{x'y'}$, $J_{x'z'}$, and $J_{y'z'}$ are the products of inertia of the whole system with respect to coordinate system $O'x'y'z'$. Their calculations are as follows:

$$J_{x'y'}(X)=\sum_{i=0}^{N}\left[m_i x_i y_i+\frac{J_{x''i}+m_i(y_i^2+z_i^2)}{2}\sin(2\alpha_i)\right]-\sum_{i=0}^{N}\left[\frac{J_{y''i}+m_i(x_i^2+z_i^2)}{2}\sin(2\alpha_i)\right]-x_c y_c\sum_{i=0}^{N}m_i, \tag{16}$$

$$J_{x'z'}(X)=\sum_{i=0}^{N}m_i x_i z_i-x_c z_c\sum_{i=0}^{N}m_i, \tag{17}$$

$$J_{y'z'}(X)=\sum_{i=0}^{N}m_i y_i z_i-y_c z_c\sum_{i=0}^{N}m_i. \tag{18}$$

3 Framework for WL-LS

In this work, all tested objects were distributed onto the four bearing plate surfaces in advance according to the centripetal balancing method and human-computer cooperative method, as presented by Sun and Teng (2003) and Liu and Teng (2008). Therefore, in the process of layout optimization, every object always stays on the same surface; that is to say, the surface to which an object belongs is fixed. Suppose the layout $X=(X_1, X_2, X_3, X_4)=(x_{11}, y_{11}, z_{11}, \alpha_{11}, \ldots, x_{1j}, y_{1j}, z_{1j}, \alpha_{1j}; x_{21}, y_{21}, z_{21}, \alpha_{21}, \ldots, x_{2k}, y_{2k}, z_{2k}, \alpha_{2k}; x_{31}, y_{31}, z_{31}, \alpha_{31}, \ldots, x_{3m}, y_{3m}, z_{3m}, \alpha_{3m}; x_{41}, y_{41}, z_{41}, \alpha_{41}, \ldots, x_{4n}, y_{4n}, z_{4n}, \alpha_{4n})$, where X_1, X_2, X_3, and X_4 are the layouts of the four surfaces P_1, P_2, P_3, and P_4, respectively, and j, k, m, and n are the numbers of the objects distributed onto the four surfaces, respectively, $j+k+m+n=N$. Our goal is to obtain an optimal layout which minimizes the two objectives $f_1(X)$ and $f_2(X)$ and meanwhile satisfies the constraints given by Eqs. (3)–(9).

Suppose that all N objects, the satellite module, and its fixed components are smooth elastic solids. Since no object transfers between any two different surfaces are allowed, once one object is distributed onto a certain surface, it will never overlap another object on a different surface. So, we need to calculate simply the overlapping depth between each object and the fixed vessel, and the overlapping depth between any two different objects which are distributed onto the same surface. Using the quasi-physical strategy (Liu *et al*., 2010; 2011), the extrusive elastic potential energy of the whole system is as follows:

$$\begin{aligned}f_3(X)&=\sum_{i=0}^{N-1}\sum_{l=i+1}^{N}ud_{il}^2=f_3(X_1)+f_3(X_2)+f_3(X_3)+f_3(X_4)\\&=\sum_{i=0}^{j-1}\sum_{l=i+1}^{j}ud_{il}^2+\sum_{i=0}^{k-1}\sum_{l=i+1}^{k}ud_{il}^2+\sum_{i=0}^{m-1}\sum_{l=i+1}^{m}ud_{il}^2+\sum_{i=0}^{n-1}\sum_{l=i+1}^{n}ud_{il}^2,\end{aligned} \tag{19}$$

where u is a physical coefficient. We set u=1 in this study. $f_3(X_1)$, $f_3(X_2)$, $f_3(X_3)$, and $f_3(X_4)$ denote the extrusive elastic potential energy which all objects allocated on four surfaces P_1–P_4 undergo, respectively. The calculation of d_{il} is described in Section 4.

For a given radius R_0 of the satellite module, we can convert the constrained optimization problem (2)–(9) into an unconstrained optimization problem by using the quasi-physical strategy and the penalty function method. The objective function is as follows:

$$\begin{aligned}\min E(X)=&\omega_1 f_2(X)+\omega_2 f_3(X)\\&+\omega_3[g_2(X)+g_3(X)+g_4(X)]\\&+\omega_4[g_5(X)+g_6(X)+g_7(X)],\end{aligned} \tag{20}$$

where ω_1, ω_2, ω_3, and ω_4 are the penalty coefficients.

If a layout X satisfies constraints (3)–(9), then X is a feasible layout of the layout optimization problem for a satellite module. Therefore, if we find an effective algorithm which could solve the unconstrained optimization problem min($E(X)$), we can obtain the optimal layout or approximate optimal layout of the original problem (1)–(9) by using effective search strategies, such as a dichotomous search, to obtain the smallest radii of the four surfaces. Suppose that the

upper bound of the interval of the dichotomous search for a given surface is $\overline{R_i}$ ($i\in\{1, 2, 3, 4\}$), which is initialized to the original radius R_0 of the satellite module, and that the lower bound of the search interval is $\underline{R_i}$ ($i\in\{1, 2, 3, 4\}$), which is initialized to the radius R of the standing column. The largest of the four radii obtained by the dichotomous search method is considered to be the smallest radius of the satellite module. The detailed procedure of the dichotomous search for a given surface P_i ($i\in\{1, 2, 3, 4\}$) is as follows:

(1) Set $\overline{R_i}=R_0$, $\underline{R_i}=R$.

(2) Set $r_i=(\overline{R_i}+\underline{R_i})/2$.

(3) Run the WL-LS algorithm. If a feasible layout is obtained, set $\overline{R_i}=r_i$; otherwise, set $\underline{R_i}=r_i$.

(4) If $|\overline{R_i}-\underline{R_i}|>10^{-4}$, go to (2); otherwise, output $\overline{R_i}$ as the smallest radius for this surface and exit.

4 Calculation of overlapping depth

To calculate the overlapping depth between two objects embedding each other, we can transfer the 3D model of the problem to a 2D one. Cylinders and cuboids are simplified as circles and rectangles, respectively. For a given bearing plate surface, a 2D Cartesian coordinate system Oxy is set up, which coincides with this surface, and its origin is located at the center of this surface. R_0 is the radius of this surface. The set of rectangles to be located is RECT= {Rect_1, Rect_2, …, Rect_r}, where r is the number of rectangles distributed onto this surface; the set of circles to be located is CIR={Cir_1, Cir_2, …, Cir_s}, where s is the number of circles distributed onto this surface. Cir indicates the fixed column. The ith rectangle is denoted by $\text{Rect}_i(p_i, \alpha_i, a_i, b_i)$, where $p_i=(x_i, y_i)$ is the position of its centroid, $\alpha_i\in[0, \pi]$ is the orientation angle between the longer edge of the ith rectangle and the x axis in coordinate system Oxy, a_i is the length of its longer edge, and b_i is the length of its shorter edge. In this study, rectangles are placed orthogonally, i.e., $\alpha_i=0$ or $\pi/2$. The jth circle is denoted by $\text{Cir}_j(p_j, r_j)$, where $p_j=(x_j, y_j)$ is the position of its centroid and r_j is its radius.

Various approaches have been proposed to compute the overlapping depth or the interference between objects, for example, the no-fit polygon (Bennell *et al*., 2001), the octree method (Wu *et al*., 1997), and the projection-separation approach (Li, 2010). In this study, we do not adhere to a single approach. Instead, according to the graphical characteristics of the two objects involved, we adopt different approaches that are most efficient. For example, if the two objects are rectangles, we adopt the projection approach; if the two objects are circles, we compare the distance between their centroids with the sum of the two radii; if the two objects are a rectangle and a circle, we adopt an approach based on a no-fit polygon. In the related previous work, discussions of approaches are quite general. Here, we give the specific calculation of overlapping depth which is suitable for orthogonally placed rectangles and circles.

4.1 Overlapping depth between a rectangle and a circle

Given a circle $\text{Cir}_j(p_j, r_j)$ and a rectangle $\text{Rect}_i(p_i, \alpha_i, a_i, b_i)$, where $\alpha_i=0$ or $\pi/2$, and d_{ij} is the overlapping depth between Rect_i and Cir_j, for every point p_t on the circumference of Cir_j, imagine there is a rectangle $\text{Rect}_t(p_t, \alpha_i, a_i, b_i)$, whose centroid is p_t. Thus, there is a set of rectangles T={Rect_t: t=1, 2, …}. Imagine every rectangle in T moves along the direction of vector $\boldsymbol{V}=(x_t-x_j, y_t-y_j)$ until it just moves out of Cir_j. After all rectangles in T move translationally in the plane, the new centroids of the rectangles in T form a continuous closed curve C (Fig. 3). C can be used to detect whether Rect_i overlaps Cir_j. If the centroid of Rect_i falls inside the region enclosed by C, then Rect_i overlaps Cir_j, and vice versa.

For simplicity, we assume $p_j=(0, 0)$, and $\alpha_i=0$. The equation for TL (top line in Fig. 3) is given by

$$y=r_j+b_i/2,\ \ x\in[-a_i/2, a_i/2]. \tag{21}$$

The equation for TRA (top-right arc in Fig. 3) is given by

$$y=\sqrt{r_j-(x-a_i/2)}+r_j,\ \ x\in(a_i/2, r_j+a_i/2). \tag{22}$$

The equation for RL (right line in Fig. 3) is given by

$$x=r_j+a_i/2,\quad y\in[-b_i/2, b_i/2]. \tag{23}$$

Since C is symmetrical about the x and y axes, the equations for the other parts of C can be deduced easily. To calculate d_{ij}, the region enclosed by C is divided into nine sub-regions (Fig. 4). If p_i falls into S_2, S_4, S_6, or S_8, d_{ij} equals the distance from p_i to TL, LL, RL, or BL, respectively. For example, in Fig. 4, if p_i falls into S_6, the overlapping depth d_{ij} equals the distance from p_i to RL. If p_i falls into S_1, S_3, S_7, or S_9, d_{ij} equals r_j minus the distance between p_i and point a, b, c, or d, respectively. For example, in Fig. 4, if p_i falls into S_9, $d_{ij} = r_j - d_{dp_i}$, where d_{dp_i} is the distance between d and p_i. If p_i falls into S_5, d_{ij} is the shortest distance between p_i and TL, LL, RL, and BL.

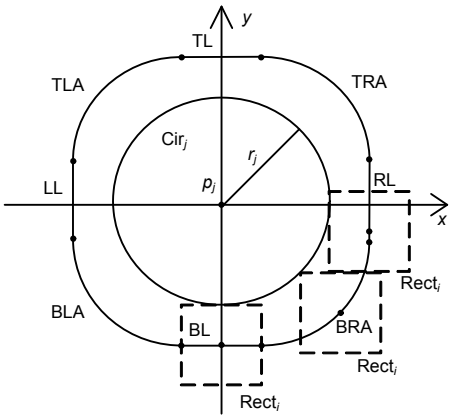

Fig. 3 A curve *C* for examining whether a rectangle overlaps a circle

TL (top line), LL (left line), RL (right line), and BL (bottom line) are lines parallel with the x or y axis. TLA (top-left arc), TRA (top-right arc), BLA (bottom-left arc), and BRA (bottom-right arc) are quadrants of a complete circle Cir$_j$

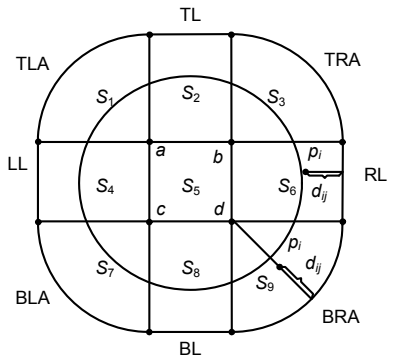

Fig. 4 Sub-regions divided for calculating the overlapping depth between a circle and a rectangle

TL (top line), LL (left line), RL (right line), and BL (bottom line) are lines parallel with the x or y axis. TLA (top-left arc), TRA (top-right arc), BLA (bottom-left arc), and BRA (bottom-right arc) are quadrants of a complete circle Cir$_j$. a, b, c, and d are the four vertices on the boundary of sub-region S_5

4.2 Overlapping depth between two rectangles

Given two rectangles Rect$_i$(p_i, α_i, a_i, b_i) and Rect$_j$(p_j, α_j, a_j, b_j), d_{ij} denotes the overlapping depth between Rect$_i$ and Rect$_j$. l_{ix} and l_{iy} are the lengths of the edges of Rect$_i$ parallel with the x axis and the y axis, respectively, and l_{jx} and l_{jy} are the lengths of the edges of Rect$_j$ parallel with the x axis and y axis, respectively (Fig. 5). To examine whether two rectangles overlap, a straightforward method is to project rectangles Rect$_i$ and Rect$_j$ onto the x and y axes, and then to check whether the projecting lines overlap. The detection formulae are shown as follows:

$$|x_i - x_j| - (l_{ix} + l_{jx})/2 > 0, \tag{24}$$

$$|y_i - y_j| - (l_{iy} + l_{jy})/2 > 0. \tag{25}$$

If any two formulae hold, the two rectangles do not overlap, i.e., d_{ij}=0. If neither of them holds, they overlap and the calculation of the overlapping depth is given by

$$d_{ij} = \left\{[(l_{ix} + l_{jx})/2 - |x_i - x_j|]^2 + [(l_{iy} + l_{jy})/2 - |y_i - y_j|]^2\right\}^{1/2}. \tag{26}$$

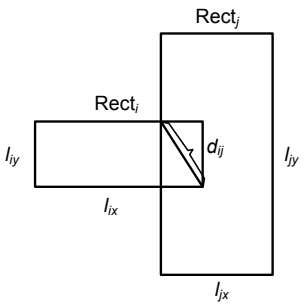

Fig. 5 Two overlapping rectangles

4.3 Overlapping depth between two circles

Given two circles Cir$_i$(p_i, r_i) and Cir$_j$(p_j, r_j), d_{ij} is their overlapping depth (Fig. 6). The detection formula is given by

$$r_i + r_j - \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} > 0. \tag{27}$$

If expression (27) is true, circles Cir$_i$ and Cir$_j$ overlap, and the overlapping depth is calculated by

$$d_{ij} = r_i + r_j - \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}. \tag{28}$$

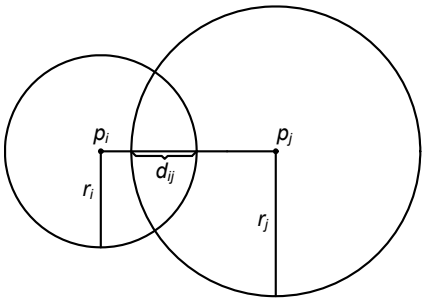

Fig. 6 The overlapping depth of two circles

4.4 Overlapping depth between a circle and a container

Given a circle $\mathrm{Cir}_i(p_i, r_i)$, if Cir_i embeds into the standing column Cir, which is considered as a fixed circle, the calculation of the overlapping depth between Cir_i and Cir is the same as in Section 4.3.

If circle $\mathrm{Cir}_i(p_i, r_i)$ overlaps the shell of the satellite module Cir_0 (Fig. 7), i.e., $\sqrt{x_i^2+y_i^2}+r_i>R_0$, the calculation of overlapping depth d_{oi} between Cir_i and Cir_0 is given by

$$d_{oi}=\sqrt{x_i^2+y_i^2}+r_i-R_0. \quad (29)$$

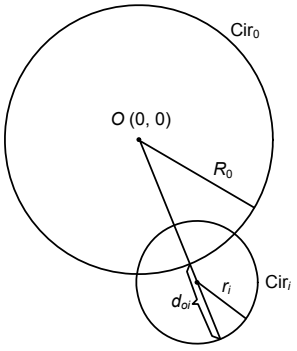

Fig. 7 A circle embedding into the shell of the satellite module

4.5 Overlapping depth between a rectangle and a container

Given a rectangle $\mathrm{Rect}_j(p_j, \alpha_j, a_j, b_j)$, if Rect_j embeds into the standing column Cir, which is considered as a fixed circle, the calculation of the overlapping depth between Rect_j and Cir is the same as in Section 4.1.

Suppose α_j=0. If rectangle $\mathrm{Rect}_j(p_j, \alpha_j, a_j, b_j)$ overlaps the shell of satellite module Cir_0 (Fig. 8), i.e., if $\sqrt{(|x_j|+a_j/2)^2+(|y_j|+b_j/2)^2}>R_0$, the calculation of the overlapping depth d_{oj} between Rect_j and Cir_0 is given by

$$d_{oj}=\sqrt{(|x_j|+a_j/2)^2+(|y_j|+b_j/2)^2}-R_0. \quad (30)$$

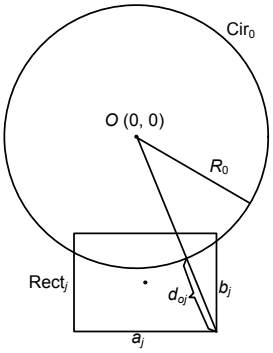

Fig. 8 A rectangle overlapping the satellite module shell

5 Wang-Landau sampling method based on local search

5.1 Wang-Landau sampling method

The Wang-Landau (WL) sampling method is a novel Monte Carlo (MC) method introduced by Wang and Landau (2001). The energy levels of the models treated in the WL sampling method are discrete; however, the idea is very general and can be applied to any parameter. For a layout optimization problem where the energy is continuous, we must first bin the energy. Considering the value of energy in the range as a positive real number, we divide all possible energies in the energy landscape into finite intervals. For example, we divide [0, 5000] into 5000 individual intervals [0, 1), [1, 2), …, [4999, 5000), and numbers which are larger than 5000 are divided into a single energy interval [5000, 5000+). Thus, we obtain 5001 energy intervals. For simplicity, the energy interval $[\lfloor E(X)\rfloor, \lceil E(X)\rceil)$ is denoted by $[E(X)]$, where $\lfloor E(X)\rfloor$ rounds $E(X)$ down to its nearest integer, and $\lceil E(X)\rceil$ rounds $E(X)$ up to its nearest integer. For example, $E(X)$=4.523 falls into energy interval [4, 5), which is denoted by [4.523]. Unlike conventional MC methods that directly generate a canonical distribution at a given temperature, this method is to estimate accurately the density of states $g(E(X))$ for the range of possible energies via a random walk which produces a flat histogram in the energy landscape, where $E(X)$ is the energy of the whole system. The WL sampling

method is based on the observation that if we perform a random walk in the energy landscape with a probability proportional to the reciprocal of the density of states $1/g(E(X))$, then a 'flat' histogram is generated for the energy distribution. By using a carefully controlled modification factor, the estimate for $g(E(X))$ is improved at each step of the random walk, which makes $g(E(X))$ converge to the correct value very quickly.

At the beginning of the WL sampling algorithm, all possible energies and the density of states function $g(E(X))$ are unknown. The density of states is set to be self-adaptive. If the random walk finds a new energy each time, we mark it as visited and set its density of states and the corresponding histogram to 1. In the simulations of the layout optimization for the satellite module, we begin the random walk in the energy landscape by heuristic layout update strategies (see Section 5.2), but the energy associated with each layout is accepted only with a probability proportional to the reciprocal of the density of states. Therefore, the acceptance probability from layout X^1 to X^2 is as follows: $P(X^1 \to X^2)=\min\{\exp(g(E(X^1))-g(E(X^2))), 1\}$. If X^2 is accepted, then $g(E(X^2))$ will be multiplied by a modification factor λ_i, and its histogram $H([E(X^2)])$ will be increased by one; that is to say, $g(E(X^2))=\lambda_i * g(E(X^2))$, $H([E(X^2)])=H([E(X^2)])+1$ (i is initialized to 0). If X^2 is not accepted, then $g(E(X^1))=\lambda_i * g(E(X^1))$, $H([E(X^1)])=H([E(X^1)])+1$. If λ_0 is too small, it will take a long time to find all the possible energies. On the contrary, if λ_0 is too large, there will be statistical errors. In this study, we set $\lambda_0=1$. The convergence of the WL sampling method is controlled by the flatness of the histogram. However, it is very difficult to obtain an absolutely flat histogram in practice. The so-called 'flat histogram' in the Wang-Landau sampling method means that all the entries of $H([E(X)])$ are not less than the histogram's average $\langle H([E(X)])\rangle$ multiplied by k ($0<k<1$), where k is decided by the complexity of the system and the expected precision of $g(E(X))$. In this study, we set $k=0.8$, and check whether the histogram is flat every 10^3 MC sweep. When the histogram is flat, all the possible energies have been roughly visited an equal number of times, and the density of states converges to the true value with accuracy proportional to the modification factor $\ln\lambda_i$. Then we reduce the modification factor λ_i to a finer one using a monotone decreasing function such as $\lambda_{i+1}=0.5\lambda_i$, reset $H([E(X)])$ to 0 for all visited energy intervals $[E(X)]$, and begin the next random walk. In this study, the modification factor and the corresponding decreasing function are selected by both experience and trial-and-error. From previous experience we set up candidate factors and their decreasing functions that might be suitable for the packing problem. Then from these candidates we select the best factor and its decreasing function through trial-and-error. When the modification factor λ_i is less than a threshold λ_{final}, the algorithm is terminated, and $g(E(X))$ converges to its real value with precision. λ_{final} is the control parameter of $g(E(X))$ and determines the number of MC iterations in the whole simulation process. If λ_{final} is too small, the simulation will take a long time. On the contrary, if λ_{final} is too large, $g(E(X))$ will not converge to its real value. In this study, we set $\lambda_{\text{final}}=0.00001$.

Our goal is to find the layout with the lowest energy, so in the simulations we also keep the lowest energy $E_{\min}$ and the corresponding layout $X_{\min}$ each time we find a new lower-energy layout.

5.2 Heuristic layout update strategies

An efficient layout update strategy is also impactful in the WL sampling simulations. According to the characteristics of the layout optimization of the satellite module, we propose the following heuristic layout update strategies:

Strategy 1 In each surface of current layout X, we pick out an object A_j that has the largest relative extrusive elastic potential energy E_j/S_j to relocate, where $E_j=\sum_{l=0,l\neq j}^{N} d_{lj}^2$ is the extrusive elastic potential energy of the jth object exerted by other objects, and S_j is the area of the jth object.

Strategy 2 Randomly generate 100 vacant points in each surface, where vacant points are the points that are inside this surface but not inside any object. The procedures are as follows:

(1) Generate a random point inside the surface but outside the central column.

(2) Judge whether this point falls into a certain object through computing the distance between this point and the centroid of the object. If this point does not fall into any object, save it as a vacant point and go to (3); otherwise, go to (1).

(3) If 100 vacant points have been saved, exit; otherwise, go to (1).

Temporarily place the centroid of the chosen object at every vacant point. Here, if the chosen object is a rectangle, it is placed in two ways: one is with its

long edge parallel with the x axis, and the other is with its long edge perpendicular to the x axis. Then, compute the extrusive elastic potential energy E_j that associates to every vacant point and placing pattern. Finally, formally place the centroid of the chosen object A_j at the vacant point, where the extrusive elastic potential energy of A_j is the lowest. With the positions of other objects unchanged, we gain a new layout X'.

5.3 Local search

When a new layout X' is obtained by heuristic layout update strategies, it may be very close to the global optimal layout, so any random layout update in the WL sampling algorithm may make search far from (and even farther and farther from) this global optimal layout. To avoid this, we adopt the local search (LS) method based on the gradient method (GM), which is a quasi-physical algorithm (Huang and Kang, 2004; He *et al.*, 2013), to search for an optimal layout near X'.

GM is also known as the steepest descent method. The search direction is the negative gradient direction. We adopt an adaptive step size in this study. If the energy increases after one iteration, which indicates that the step size for this step is too large, we decrease h to $0.8h$. If the energy decreases, we keep h. The procedures of the GM with an adaptive step size are described as follows:

(1) Set h=1, $h_{\min}=10^{-4}$, $\varepsilon=10^{-20}$.

(2) Under the current layout X', compute the gradient vector $\nabla(E(X'))$ of $E(X')$ for each object A_i (i=1, 2, …, N) in the x and y directions, and let $X^2=X'-h*\nabla(F(X'))$.

(3) If $E(X^2)>E(X')$, set $h=h*0.8$.

(4) Let $X'=X^2$, $X^2=X'-h*\nabla(E(X'))$.

(5) If $f_3(X^2)<\varepsilon$ or $h<h_{\min}$, return X^2; otherwise, go to (3).

5.4 Description of WL-LS

By combining the WL sampling method with heuristic layout update strategies and the LS procedure, we propose a hybrid WL-LS algorithm for the layout optimization of the satellite module. The calculation procedure of the WL-LS algorithm is outlined as follows:

(1) Randomly produce an initial layout X^1 based on four bearing plate surfaces. Set $X_{\min}=X^1$, $E_{\min}=E(X^1)$. Let the set of intervals containing visited energies be $S=\{[E(X^1)]\}$. Set the density of states function as $g(E(X^1))=1$, and the histogram function as $H([E(X^1)])=1$. Set i=0, l=0, λ_0=1, k=0.8.

(2) In each surface P_i (i=1, 2, 3, 4) of the current layout X^1, pick object A_j (j=1, 2, 3, 4) with the largest E_j/S_j, where E_j is the extrusive elastic potential energy of the jth object. Copy the current layout X^1.

(3) Relocate each picked object A_j by using heuristic layout update strategy 2 in surface P_i (i=1, 2, 3, 4), and gain a new layout X'.

(4) Call the GM procedure. The outcome layout of GM is denoted by X^2. Compute $E(X^2)$. Set $g(E(X^2))=1$, $H([E(X^2)])=1$, $l=l+1$.

(5) If $[E(X^2)]\notin S$, let $S=S\cup\{[E(X^2)]\}$.

(6) If random(0, 1)<min{exp[$g(E(X^1))-g(E(X^2))$], 1}, then accept X^2 (i.e., let $X^1=X^2$, $E(X^1)=E(X^2)$); otherwise, do not accept X^2.

(7) Update the density of states $g(E(X))$ and the histogram $H([E(X)])$. That is, if X^2 is accepted, let $g(E(X^2))=g(E(X^2))*\lambda_i$, $H([E(X^2)])=H([E(X^2)])+1$, and go to (8); otherwise, let $g(E(X^1))=g(E(X^1))*\lambda_i$, $H([E(X^1)])=H([E(X^1)])+1$, and go to (9).

(8) If $E(X^2)<E_{\min}$, let $X_{\min}=X^2$, and $E_{\min}=E(X^2)$.

(9) If l%1000=0, go to (10); otherwise, go to (2).

(10) If $H([E(X)])\geq k<H([E(X)])>$ for all visited energy intervals $[E(X)]\in S$, then go to (11); otherwise, go to (2).

(11) Set $\lambda_{i+1}=\lambda_i*0.5$, $i=i+1$.

(12) If $\lambda_i<0.00001$, then output $E_{\min}$ and $X_{\min}$, and terminate the iteration; otherwise, reset $H([E(X)])=0$ and keep $g(E(X))$ for all visited energies $E(X)$, and go to (2).

6 Experimental results and discussion

To test the computational performance of the WL-LS algorithm, we applied it in two instances. Both instances are based on the international commercial communication satellite module INTELSAT-III with different technological parameters, one with 51 objects and the other with 53 objects. We implemented the WL-LS algorithm in the Java programming language and ran it on a PC with 1.5 GHz CPU and 2.0 GB RAM. For each instance, the WL-LS algorithm was run 50 times independently to optimize the layout of objects on the four bearing plate surfaces. According to these 50 results, we can gain a pareto

optimal set for each instance. In this study, we specify the solution with the smallest enveloping radius as the optimal solution for preference.

For the first instance, we suppose that 51 objects have been allocated on the four bearing surfaces in the satellite module. The objects' dimensions and masses have been given by Liu and Teng (2008). The first 31 objects are cylinders, and the remaining 20 objects are cuboids. The parameters of the satellite module are as follows: the radius of the bearing plate is R_0=500 mm; the radius of the column is R=100 mm; the heights from the base of the satellite to the bottoms of the lower plate and the upper plate are H_1=500 mm and H_2=1050 mm, respectively; the height of the column is H_3=1400 mm; the thickness of each plate is H_t= 20 mm; the mass of the satellite module (including the shell, two bearing plates, and the column) is 349.557 kg, and its centroid is (0, 0, 859) mm in the coordinate system $Oxyz$, with an inertia matrix

$$\boldsymbol{J}_0=\begin{pmatrix} J_x & J_{xy} & J_{xz} \\ J_{yx} & J_y & J_{yz} \\ J_{zx} & J_{zy} & J_z \end{pmatrix} =\begin{pmatrix} 347.2921 & 0 & 0 \\ 0 & 485.5460 & 0 \\ 0 & 0 & 210.5343 \end{pmatrix} \mathrm{kg\cdot m^2}. \tag{31}$$

The technological requirements for the final layout of the whole system are given by Eqs. (3)–(9), with parameters x_e=0, y_e=0, z_e=780 mm. The penalty coefficients are set as follows: $\omega_1=10^{-1}$, $\omega_2=10^6$, $\omega_3=10^4$, and $\omega_4=10^4$.

For this instance, the pareto optimal set by WL-LS is $P(Q, M)$={(460.61, 690.22), (459.28, 691.70), (458.95, 692.65), (457.76, 693.07)}, where Q denotes the enveloping radius and M the inertia moment of the system. We choose the solution with Q=457.76, M=693.07 as the optimal solution for preference. The best and average computational results by WL-LS for instance 1 are shown in Tables 1 and 2, respectively, in comparison with those by the HAKD method (Liu and Teng, 2008). From Tables 1 and 2, we can see that the proposed WL-LS algorithm outperforms the HAKD method in every aspect. Compared with the best results by the HAKD method, the inertia moment of the best layout obtained by WL-LS decreases by (711.55−693.07)/711.55×100% =2.60%; the enveloping radius decreases by 0.35%; the centroid position error decreases by 29.12%; the inertia angle error reduces by 76.5%. Table 3 shows the data of the optimal layout obtained by WL-LS. Fig. 9 shows the diagram of the optimal layout obtained by WL-LS.

To further understand the effects of each component in WL-LS, we ran WL alone, WL with local

Table 1 Comparison of the best results by HAKD, WL, WL+GM, WL+HS, and WL-LS for instance 1

Algorithm	Overlapping area (mm^2)	Centroid position error (mm)	Inertia angle error (rad)	Inertia moment (kg·mm^2)	Enveloping radius (mm)
HAKD	0	5.94e-2	2.00e-2	711.55	459.37
WL	0	1.03e-0	1.20e-1	710.96	465.05
WL+GM	0	6.09e-2	5.41e-3	695.34	459.17
WL+HS	0	9.77e-1	1.21e-1	708.31	462.48
WL-LS	0	4.21e-2	4.70e-3	693.07	457.76

Centroid position error is the sum of centroid errors in the x and y axes, i.e., $|x_c-x_e|+|y_c-y_e|$. Inertia angle error is the sum of inertia angles in the x', y', and z' axes, i.e., $|\theta_{x'}(X)|+|\theta_{y'}(X)|+|\theta_{z'}(X)|$. Inertia moment is the sum of inertia moments in the x', y', and z' axes, i.e., $J_{x'}(X)+J_{y'}(X)+J_{z'}(X)$

Table 2 Comparison of the average results by HAKD, WL, WL+GM, WL+HS, and WL-LS for instance 1

Algorithm	Overlapping area (mm^2)	Norm of centroid position error $\sqrt{\delta x_c^2+\delta y_c^2+\delta z_c^2}$ (mm)	Norm of inertia angle error $\sqrt{\theta_{x'}^2+\theta_{y'}^2+\theta_{z'}^2}$ (rad)	Norm of inertia moment $\sqrt{J_{x'}(X)^2+J_{y'}(X)^2+J_{z'}(X)^2}$ (kg·mm^2)	Enveloping radius (mm)
HAKD	0	0.255	1.65e-2	423.18	467.16
WL	306.13	2.47	6.98e-1	424.67	471.86
WL+GM	20.80	0.20	7.19e-3	418.84	463.17
WL+HS	8.10	2.45	7.05e-1	421.35	469.67
WL-LS	0	0.187	6.44e-3	417.35	461.48

Table 3 Data for optimal layout by WL-LS for instance 1

No.	x (mm)	y (mm)	α (rad)*	Surface	No.	x (mm)	y (mm)	α (rad)*	Surface
1	–217.84	–0.42	–	1	27	350.28	–46.72	–	4
2	147.91	161.17	–	4	28	277.92	28.06	–	3
3	194.26	–51.03	–	3	29	145.07	185.07	–	2
4	205.98	–31.61	–	4	30	146.73	178.25	–	1
5	–108.97	168.58	–	1	31	–245.06	187.89	–	3
6	44.78	351.66	–	4	32	–85.81	–178.42	0	1
7	279.44	143.57	–	3	33	–186.38	–185.01	0	2
8	–97.17	–325.27	–	3	34	143.89	–207.56	0	4
9	141.47	–182.72	–	1	35	–46.66	230.91	π/2	3
10	–182.75	–298.32	–	4	36	–225.69	–1.42	0	4
11	–286.61	–177.14	–	4	37	103.78	227.56	π/2	3
12	5.61	328.83	–	2	38	–200.14	–4.87	π/2	2
13	230.61	–3.46	–	1	39	39.40	–200.39	0	2
14	84.30	–239.61	–	3	40	–203.61	174.31	0	4
15	258.43	–257.10	–	3	41	–127.32	171.04	π/2	2
16	343.95	–166.80	–	4	42	180.83	9.15	π/2	2
17	–315.39	–181.75	–	3	43	–27.95	176.56	π/2	4
18	338.21	162.85	–	4	44	330.91	87.74	0	2
19	–350.70	–74.49	–	2	45	–110.69	–143.71	0	4
20	68.45	–351.64	–	2	46	210.83	323.66	π/2	4
21	–373.91	63.65	–	3	47	309.81	–12.33	0	2
22	–336.78	–29.37	–	3	48	239.80	–158.79	0	2
23	–1.93	150.47	–	2	49	–192.63	–12.29	π/2	3
24	315.31	–140.44	–	3	50	148.78	–333.69	0	4
25	15.10	270.25	–	1	51	–115.17	–174.10	0	3
26	–291.25	226.75	–	2					

* The orientation angle is defined only for the cuboids, and it does not exist for the cylinders

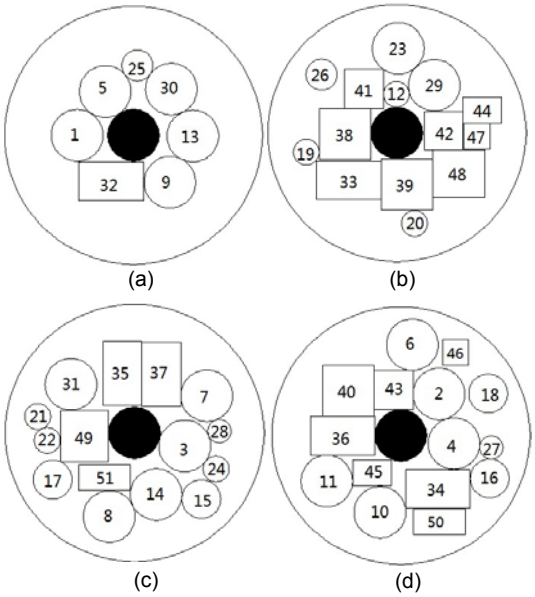

Fig. 9 The 2D diagram of the optimal layout obtained by WL-LS for instance 1: (a) surface 1; (b) surface 2; (c) surface 3; (d) surface 4

search GM and without heuristic layout update strategies (denoted as WL+GM), and WL with heuristic layout update strategies (HS) and without local search GM (denoted as WL+HS) 50 times. The best and average results for 50 independent runs by WL, WL+GM, and WL+HS are listed in Tables 1 and 2, respectively, for comparison. The pareto optimal set by WL is $P(Q, M)$={(468.58, 696.78), (467.24, 697.40), (466.06, 699.08), (465.05, 710.96)}, and we choose the solution with Q=465.05, M=710.96 as the optimal solution for preference. The pareto optimal set by WL+GM is $P(Q, M)$={(462.51, 693.88), (462.03, 694.79), (460.63, 695.16), (459.17, 695.34)}, and we choose the solution with Q=459.17, M=695.34 as the optimal solution for preference. The pareto optimal set by WL+HS is $P(Q, M)$={(465.09, 705.86), (464.00, 706.16), (463.01, 707.47), (462.48, 708.31)}, and we choose the solution with Q=462.48, M=708.31 as the optimal solution for preference. Note that in each run of WL and WL+GM, we

randomly update the layout. In WL-LS, the WL method is used mainly to execute a global search. The heuristic layout update strategies are used to generate new layouts, and the local search procedure based on the gradient method is used to search for lower-energy layouts near newly generated layouts. Through the comparison of computational results, one can see that WL+GM and WL+HS improve the performance of WL alone in different aspects. However, by combining the merits of GM and HS, WL-LS significantly outperforms the other three algorithms WL, WL+GM, and WL+HS in both the best and average results.

For the second instance, we suppose that 53 objects have also been allocated on the four bearing surfaces in the satellite module. The objects' dimensions and masses have been given by Sun and Teng (2003). The first 24 objects are cuboids, and the remaining 29 objects are cylinders. The parameters of the satellite module are as follows: the radius of the bearing plate is R_0=500 mm; the radius of the column is R=100 mm; the heights from the base of the satellite to the bottoms of the lower plate and the upper plate are H_1=300 mm and H_2=830 mm, respectively; the height of the column is H_3=1150 mm; the thickness of both plates is H_t=20 mm; the mass of the satellite module (including the shell, two bearing plates, and the column) is 776.53 kg, and its centroid is (0, 0, 553.56) mm in coordinate system $Oxyz$, with an inertia matrix

$$\boldsymbol{J}_0=\begin{pmatrix} J_x & J_{xy} & J_{xz} \\ J_{yx} & J_y & J_{yz} \\ J_{zx} & J_{zy} & J_z \end{pmatrix} =\begin{pmatrix} 452.25 & 0 & 0 \\ 0 & 452.25 & 0 \\ 0 & 0 & 146.82 \end{pmatrix} \mathrm{kg\cdot m^2}. \tag{32}$$

The technological requirements for the final layout of the whole system are given by Eqs. (3)–(9) with parameters x_e=0, y_e=0, z_e=523.26 mm. The penalty coefficients are set as follows: $\omega_1=10^{-1}$, $\omega_2=10^6$, $\omega_3=10^4$, and $\omega_4=10^4$.

For this instance, the pareto optimal set by WL-LS is $P(Q, M)$={(479.31, 791.67), (478.14, 792.64), (476.07, 794.33), (475.11, 795.59)}, where Q denotes the enveloping radius and M the inertia moment of the system. We choose the solution with Q=475.11, M=795.59 as the optimal solution for preference. The best computational results by WL-LS for instance 2 are shown in Table 4, in comparison with those by the hybrid knowledge fusion (HKF) method (Wang and Teng, 2009). From Table 4, we can see that the proposed WL-LS algorithm outperforms the HKF method in every aspect. Compared with the best results by the HKF method, the inertia moment of the best layout obtained by WL-LS decreases by (796.15−795.59)/796.15×100%=0.07%; the centroid position error decreases by 99.70%; the inertia angle error decreases by 67.81%; the enveloping radius decreases by 4.98%. A diagram of the optimal layout obtained by WL-LS is shown in Fig. 10. Table 5 shows the data for the optimal layout obtained by WL-LS.

We also compare the best and average results for 50 independent runs by WL, WL+GM, and WL+HS in Tables 4 and 6, respectively. The pareto optimal set by WL is $P(Q, M)$={(491.40, 799.71), (489.43, 800.77), (488.30, 801.34), (487.05, 802.13)}, and we choose the solution with Q=487.05, M=802.13 as the optimal solution for preference. The pareto optimal set by WL+GM is $P(Q, M)$={(480.74, 795.27), (479.69, 796.67), (479.05, 796.96), (478.50, 797.01)}, and we choose the solution with Q=478.50, M=797.01 as the optimal solution for preference. The

Table 4 Comparison of the best results by HKF, WL, WL+GM, WL+HS, and WL-LS for instance 2

Algorithm	Overlapping area (mm^2)	Centroid position error (mm)	Inertia angle error (rad)	Inertia moment ($kg\cdot mm^2$)	Enveloping radius (mm)
HKF	0	1.139e-0	1.46e-2	796.15	500.00
WL	0	3.60e-1	5.58e-1	802.13	487.05
WL+GM	0	3.59e-3	5.45e-3	797.01	478.50
WL+HS	0	3.60e-1	5.55e-1	800.69	483.49
WL-LS	0	3.40e-3	4.70e-3	795.59	475.11

Centroid position error is the sum of centroid errors in the x and y axes, i.e., $|x_c-x_e|+|y_c-y_e|$. Inertia angle error is the sum of inertia angles in the x', y', and z' axes, i.e., $|\theta_{x'}(X)|+|\theta_{y'}(X)|+|\theta_{z'}(X)|$. Inertia moment is the sum of inertia moments in the x', y', and z' axes, i.e., $J_{x'}(X)+J_{y'}(X)+J_{z'}(X)$

pareto optimal set by WL+HS is $P(Q, M)$={(486.41, 797.21), (485.58, 798.54), (484.32, 799.79), (483.49, 800.69)}, and we choose the solution with Q=483.49, M=800.69 as the optimal solution for preference. Through the comparison of computational results, we can draw the same conclusion as that for instance 1.

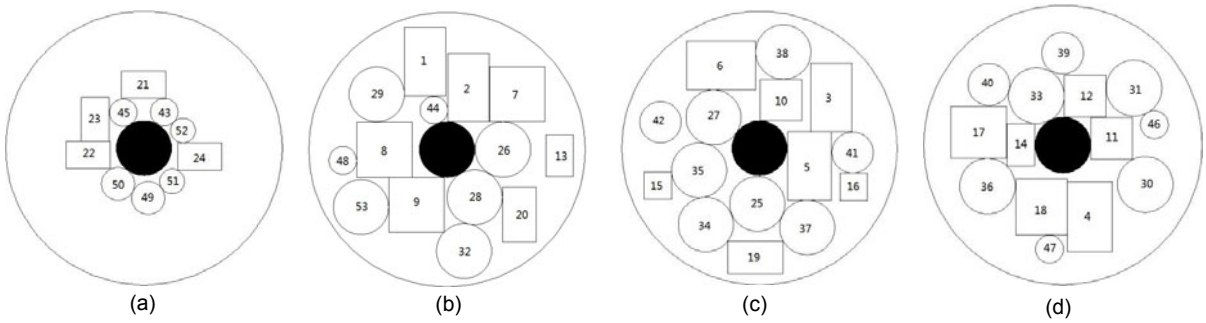

Fig. 10 Optimal layout obtained by WL-LS for instance 2: (a) surface 1; (b) surface 2; (c) surface 3; (d) surface 4

Table 5 Data for optimal layout by WL-LS for instance 2

No.	x (mm)	y (mm)	α (rad)*	Surface	No.	x (mm)	y (mm)	α (rad)*	Surface
1	–79.05	319.48	π/2	2	28	100.17	–173.28	–	2
2	78.05	225.00	π/2	2	29	–254.14	200.41	–	2
3	260.45	185.65	π/2	3	30	295.20	–140.60	–	4
4	95.75	–255.70	π/2	4	31	255.16	203.91	–	4
5	181.14	–65.03	π/2	3	32	61.88	–369.81	–	2
6	–139.51	302.44	0	3	33	–95.35	176.46	–	4
7	254.23	200.50	0	2	34	–195.06	–279.58	–	3
8	–226.27	–0.46	0	2	35	–219.13	–80.85	–	3
9	–110.75	–200.75	π/2	2	36	–271.08	–146.97	–	4
10	78.43	175.44	π/2	3	37	173.10	–290.27	–	3
11	175.30	24.83	π/2	4	38	86.55	351.24	–	3
12	79.88	175.50	0	4	39	–1.02	328.75	–	4
13	407.59	–22.17	π/2	2	40	–265.75	216.82	–	4
14	–151.76	0.84	π/2	4	41	338.10	–16.37	–	3
15	–369.61	–137.32	0	3	42	–359.36	94.89	–	3
16	342.41	–142.06	0	3	43	73.70	132.14	–	1
17	–303.53	46.47	0	4	44	–47.16	142.71	–	2
18	–77.43	–221.91	π/2	4	45	–74.85	130.06	–	1
19	–14.58	–399.76	0	3	46	328.34	72.74	–	4
20	261.60	–236.80	π/2	2	47	–48.19	–372.69	–	4
21	–0.38	232.51	0	1	48	–376.89	–39.99	–	2
22	–201.84	–25.59	0	1	49	15.61	–182.85	–	1
23	–176.88	104.78	π/2	1	50	–93.44	–131.03	–	1
24	202.03	–31.08	0	1	51	101.89	–122.07	–	1
25	–7.95	–204.16	–	3	52	141.24	64.32	–	1
26	202.73	0.15	–	2	53	–311.05	–209.25	–	2
27	–167.13	112.41	–	3					

* The orientation angle is defined only for the cuboids, and it does not exist for the cylinders

Table 6 Comparison of the average results by WL, WL+GM, WL+HS, and WL-LS for instance 2

Algorithm	Overlapping area (mm^2)	Norm of the centroid position error $\sqrt{\delta x_c^2+\delta y_c^2+\delta z_c^2}$ (mm)	Norm of the inertia angle error $\sqrt{\theta_{x'}^2+\theta_{y'}^2+\theta_{z'}^2}$ (rad)	Norm of the inertia moment $\sqrt{J_{x'}(X)^2+J_{y'}(X)^2+J_{z'}(X)^2}$ ($kg\cdot mm^2$)	Enveloping radius (mm)
WL	280.30	3.50e-1	6.62e-1	491.95	488.39
WL+GM	19.64	2.64e-3	7.11e-3	489.81	479.92
WL+HS	6.46	3.48e-1	6.89e-1	491.20	484.43
WL-LS	0	2.40e-3	3.60e-3	489.06	476.65

7 Conclusions and future work

When optimizing the layout design of a satellite module, it is easy for an algorithm to get trapped in local minima separated by high-energy barriers. To address this problem, we use a hybrid WL-LS method, which incorporates the LS procedure based on gradient descent and heuristic layout update strategies into the Wang-Landau sampling method. To improve the efficiency of WL-LS, we adopt an accurate and fast method for computing the overlapping depth between two objects (such as two rectangular objects, two circular objects, or a rectangular object and a circular object) embedding each other. Numerical results show that WL-LS outperforms methods in the literature. There are also several problems that need to be solved in the future: (1) In this study, the rectangles are placed orthogonally, which restricts the flexibility of the layout design. We need to find the methods for computing the overlapping depth with respect to arbitrarily placed rectangles. (2) The distribution of objects onto different surfaces is not discussed in this paper, and we simply adopt the allocation scheme proposed in the literature. In the future, we will focus research on these aspects.

References

Allen, S.D., Burke, E.K., Kendall, G., 2011. A hybrid placement strategy for the three-dimensional strip packing problem. *Eur. J. Oper. Res.*, **209**(3):219-227. http://dx.doi.org/10.1016/j.ejor.2010.09.023

Bennell, J.A., Dowsland, K.A., Dowsland, W.B., 2001. The irregular cutting-stock problem—a new procedure for deriving the no-fit polygon. *Comput. Oper. Res.*, **28**(3): 271-287. http://dx.doi.org/10.1016/S0305-0548(00)00021-6

Chen, W., Shi, Y.J., Teng, H.F., 2008. An improved differential evolution with local search for constrained layout optimization of satellite module. *LNCS*, **5227**:742-749. http://dx.doi.org/10.1007/978-3-540-85984-0_89

Crainic, T.G., Perboli, G., Rei, W., *et al*., 2011. Efficient lower bounds and heuristics for the variable cost and size bin packing problem. *Comput. Oper. Res.*, **38**(11):1474-1482. http://dx.doi.org/10.1016/j.cor.2011.01.001

Gonçalves, J.F., Resende, M.G., 2011. A parallel multi-population genetic algorithm for a constrained two-dimensional orthogonal packing problem. *J. Comb. Optim.*, **22**(2):180-201. http://dx.doi.org/10.1007/s10878-009-9282-1

He, K., Mo, D.Z., Xu, R.C., *et al*., 2013. A quasi-physical algorithm based on coarse and fine adjustment for solving circles packing problem with constraints of equilibrium. *Chin. J. Comput.*, **36**(6):1224-1234 (in Chinese). http://dx.doi.org/10.3724/SP.J.1016.2013.01224

Huang, W.Q., Kang, Y., 2004. A short note on a simple search heuristic for the diskspacking problem. *Ann. Oper. Res.*, **131**(1-4):101-108. http://dx.doi.org/10.1023/B:ANOR.0000039514.14699.03

Huo, J.Z., Teng, H.F., 2009. Optimal layout design of a satellite module using a coevolutionary method with heuristic rules. *J. Aerosp. Eng.*, **22**(2):101-111. http://dx.doi.org/10.1061/(ASCE)0893-1321(2009)22:2(101)

Jin, B., Teng, H.F., 2007. Case-based evolutionary design approach for satellite module layout. *J. Sci. Ind. Res.*, **66**(12):989-994.

Khanafer, A., Clautiaux, F., Talbi, E.G., 2012. Tree-decomposition based heuristics for the two-dimensional bin packing problem with conflicts. *Comput. Oper. Res.*, **39**(1):54-63. http://dx.doi.org/10.1016/j.cor.2010.07.009

Landau, D.P., Tsai, S.H., Exler, M., 2004. A new approach to Monte Carlo simulations in statistical physics: Wang-Landau sampling. *Am. J. Phys.*, **72**(10):1294-1302. http://dx.doi.org/10.1119/1.1707017

Lei, K.Y., Qiu, Y.H., 2006. A study of constrained layout optimization using adaptive particle swarm optimizer. *J. Comput. Res. Dev.*, **43**:1724-1731 (in Chinese).

Li, Z.Q., 2010. A fast projection-separation approach for collision detection between polytopes. *J. Comput. Aid. Des. Comput. Graph.*, **22**(4):639-646 (in Chinese).

Li, Z.Q., Xie, Y.F., Tian, Z.J., *et al*., 2012. A heuristic particle swarm optimization with quasi-human strategy for weighted circles packing problem. 8th Int. Conf. on Natural Computation, p.723-727. http://dx.doi.org/10.1109/ICNC.2012.6234660

Liu, J.F., Li, G., 2010. Basin filling algorithm for the circular packing problem with equilibrium behavioral constraints. *Sci*. *Chin*. *Inf*. *Sci*., **53**(5):885-895 (in Chinese). http://dx.doi.org/10.1007/s11432-010-0080-2

Liu, J.F., Xue, S.J., Liu, Z.X., *et al*., 2009. An improved energy landscape paving algorithm for the problem of packing circles into a larger containing circle. *Comput*. *Ind*. *Eng*., **57**(3):1144-1149. http://dx.doi.org/10.1016/j.cie.2009.05.010

Liu, J.F., Li, G., Chen, D.B., *et al*., 2010. Two-dimensional equilibrium constraint layout using simulated annealing. *Comput*. *Ind*. *Eng*., **59**(4):530-536. http://dx.doi.org/10.1016/j.cie.2010.06.009

Liu, J.F., Li, G., Geng, H.T., 2011. A new heuristic algorithm for the circular packing problem with equilibrium constraints. *Sci*. *Chin*. *Inf*. *Sci*., **54**(8):1572-1584. http://dx.doi.org/10.1007/s11432-011-4351-3

Liu, Z.W., Teng, H.F., 2008. Human–computer cooperative layout design method and its application. *Comput*. *Ind*. *Eng*., **55**(4):735-757. http://dx.doi.org/10.1016/j.cie.2006.11.007

Martello, S., Pisinger, D., Vigo, D., 2000. The three-dimensional bin packing problem. *Oper*. *Res*., **48**(2):256-267. http://dx.doi.org/10.1287/opre.48.2.256.12386

Moon, I., Nguyen, T.V.L., 2014. Container packing problem with balance constraints. *OR Spectr*., **36**(4):837-878. http://dx.doi.org/10.1007/s00291-013-0356-1

Schulz, B.J., Binder, K., Müller, M., *et al*., 2003. Avoiding boundary effects in Wang-Landau sampling. *Phys*. *Rev*. *E*, **67**(6):067102. http://dx.doi.org/10.1103/PhysRevE.67.067102

Seaton, D.T., Wüst, T., Landau, D.P., 2010. Collapse transitions in a flexible homopolymer chain: application of the Wang-Landau algorithm. *Phys*. *Rev*. *E*, **81**(1):011802. http://dx.doi.org/10.1103/PhysRevE.81.011802

Sun, Z.G., Teng, H.F., 2003. Optimal layout design of a satellite module. *Eng*. *Optim*., **35**(5):513-529. http://dx.doi.org/10.1080/0305215031000160233 5

Tang, F., Teng, H.F., 1999. A modified genetic algorithm and its application to layout optimization. *J*. *Softw*., **10**(10): 1096-1102 (in Chinese).

Teng, H.F., Chen, Y., Zeng, W., *et al*., 2010. A dual-system variable-grain cooperative coevolutionary algorithm: satellite-module layout design. *IEEE Trans*. *Evol*. *Comput*., **14**(3):438-455. http://dx.doi.org/10.1109/TEVC.2009.2033585

Wang, F., Landau, D.P., 2001. Efficient, multiple-range random walk algorithm to calculate the density of states. *Phys*. *Rev*. *Lett*., **86**(10):2050. http://dx.doi.org/10.1103/PhysRevLett.86.2050

Wang, Y.S., Teng, H.F., 2009. Knowledge fusion design method: satellite module layout. *Chin*. *J*. *Aeronaut*., **22**(1): 32-42. http://dx.doi.org/10.1016/S1000-9361(08)60066-7

Wu, M.H., Yu, Y.X., Zhou, J., 1997. An octree algorithm for collision detection using space partition. *Chin*. *J*. *Comput*., **20**(9):849-854 (in Chinese).

Xu, Y.C., Xiao, R.B., 2008. Ant colony algorithm for layout optimization with equilibrium constraints. *Contr*. *Dec*., **23**(1):25-29 (in Chinese).

Zhang, B., Teng, H.F., Shi, Y.J., 2008. Layout optimization of satellite module using soft computing techniques. *Appl*. *Soft Comput*., **8**(1):507-521. http://dx.doi.org/10.1016/j.asoc.2007.03.004

Zhang, D.F., Deng, A.S., Kang, Y., 2005. A hybrid heuristic algorithm for the rectangular packing problem. Int. Conf. on Computational Science, p.783-791. http://dx.doi.org/10.1007/11428831_97

Zhang, Z.H., Wang, Y.S., Teng, H.F., *et al*., 2013. Parallel dual-system cooperative co-evolutionary differential evolution algorithm with human-computer cooperation for multi-cabin satellite layout optimization. *J*. *Converg*. *Inform*. *Technol*., **8**(4):711-720. http://dx.doi.org/10.4156/jcit.vol8.issue4.82

Zhou, C., Bhatt, R.N., 2005. Understanding and improving the Wang-Landau algorithm. *Phys*. *Rev*. *E*, **72**(2):025701. http://dx.doi.org/10.1103/PhysRevE.72.025701

Zhou, C., Gao, L., Gao, H.B., 2005. Particle swarm optimization based algorithm for constrained layout optimization. *Contr*. *Dec*., **20**(1):36-40 (in Chinese).