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

A new energy landscape paving heuristic for satellite module layouts*

Jing-fa LIU[1,2], Juan HUANG[†‡1,2], Gang LI[3], Wen-jie LIU[1,2], Ting-zhao GUAN[4], Liang HAO[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: huangjuan4455@163.com

Received Sept. 17, 2015; Revision accepted Feb. 18, 2016; Crosschecked Sept. 20, 2016

Abstract: This article describes a study of the satellite module layout problem (SMLP), which is a three-dimensional (3D) layout optimization problem with performance constraints that has proved to be non-deterministic polynomial-time hard (NP-hard). To deal with this problem, we convert it into an unconstrained optimization problem using a quasi-physical strategy and the penalty function method. The energy landscape paving (ELP) method is a class of Monte-Carlo-based global optimization algorithm that has been successfully applied to solve many optimization problems. ELP can search for low-energy layouts via a random walk in complex energy landscapes. However, when ELP falls into the narrow and deep valleys of an energy landscape, it is difficult to escape. By putting forward a new update mechanism of the histogram function in ELP, we obtain an improved ELP method which can overcome this drawback. By incorporating the gradient method with local search into the improved ELP method, a new global search optimization method, nELP, is proposed for SMLP. Two representative instances from the literature are tested. Computational results show that the proposed nELP algorithm is an effective method for solving SMLP with performance constraints.

Key words: Three-dimensional packing, Energy landscape paving, Layout optimization, Performance constraints
http://dx.doi.org/10.1631/FITEE.1500302 **CLC number:** TP391; V474

1 Introduction

Given *N* objects, each with given shape and size, and a bounded space, layout problems (Martello and Vigo, 2000; Lodi *et al.*, 2002; Zhang and Deng, 2005) involve determining how to pack these objects best into the bounded space without overlap. This is relevant to many areas, such as aerospace, very-large-scale integration design, automobile design, transportation, and other industries. In general, layout problems are divided into one-dimensional (1D), two-dimensional (2D), and three-dimensional (3D) problems. In recent years, studies have focused on 2D and 3D layout optimization problems. For the 2D problems, del Valle *et al.* (2012) presented a greedy randomized adaptive search procedure based heuristic for knapsack and cutting stock problems with items of irregular shape. He *et al.* (2013) proposed a quasi-physical algorithm based on coarse and fine adjustment. Galiev and Lisafina (2013) proposed a heuristic algorithm based on linear models. Thomas and Chaudhari (2014) used a genetic algorithm based on a hyper heuristic approach to obtain the optimal or sub-optimal solution. For 3D strip packing problems, Bansal *et al.* (2013) obtained a harmonic algorithm

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (No. 61373016), 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: Juan HUANG, http://orcid.org/0000-0003-2830-7699

which establishes a connection between bin packing and strip packing solutions. Silveira *et al.* (2013) proposed a new ant colony optimization algorithm for the 3D bin packing problem in the steel industry. Tsai *et al.* (2014) devised an efficient global optimization method that transforms a 3D open dimension rectangular packing problem into a mixed-integer linear program. Jansen and Prädel (2014) presented a new asymptotic approximation algorithm for 3D strip packing.

In practical production, the satellite module layout problem (SMLP) (Cagan *et al.*, 2002; Sun and Teng, 2003; Sun *et al.*, 2003; Li *et al.*, 2011) has been proposed, and has become a major issue to be resolved. SMLP involves putting a set of specified devices in a known container, and these devices have to meet some practical constraint conditions, such as non-interference, compact packing, equilibrium, and inertia angles. SMLP is classed as an NP-hard problem in mathematics and a complex engineering system problem in engineering. The difficulties involved in solving this problem are the complex problem description, the establishment of mathematical models and the solving process.

In recent years, some layout design algorithms for solving SMLP have been proposed. Sun and Teng (2003) proposed a two-stage layout method, which includes a 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 and Teng (2005) proposed particle swarm optimization based on a pyramid model. Huo *et al.* (2006) proposed a human-guided genetic algorithm. Zhang *et al.* (2008) presented a hybrid method which firstly uses a Hopfield neural network to allocate the given apparatuses and equipment to the bearing plate surfaces in the satellite module, and then integrates genetic algorithm/particle swarm optimization and quasi-principal component analysis to deal with the detailed layout optimization. Liu and Teng (2008) extended their work by introducing a human algorithm knowledge based layout design method, and Wang and Teng (2009) put forward a human-computer cooperative knowledge fusion approach. Teng *et al.* (2010) proposed a dual-system framework based on a variable-grain cooperative coevolutionary algorithm. Chen and Teng (2010) proposed a cooperative coevolutionary genetic algorithm with a coarse-to-fine grain strategy. Wang *et al.* (2011) firstly built a hierarchical tree of the product structure to describe the satellite module, and then proposed a solution approach based on a multi-agent system to solve the optimization problem. Cuco *et al.* (2014) presented a multi-objective methodology which combines CAD and optimization tools in an integrated environment and automatically finds solutions for a 3D layout for equipment in a spacecraft.

This article focuses on SMLP, and proposes a new energy landscape paving (nELP) method (Hansmann and Wille, 2002; Schug *et al.*, 2005) to deal with it. The ELP method is an improved Monte-Carlo (MC) method. There have been many improvements to the ELP method, and it has been widely applied (Zhan *et al.*, 2006; Hamacher, 2007; Liu *et al.*, 2009; Shanker and Bandyopadhyay, 2011; Rakshit and Bandyopadhyay, 2013). Based on the combination of ELP and basin hopping, which has the ability to surpass high-energy barriers and find lower-energy regions, Zhan *et al.* (2006) proposed a new MC optimization method, basin paving. Hamacher (2007) applied detrended fluctuation analysis (DFA) to the time series of energy values observed in minimization runs of the ELP method. The results showed that ELP is optimal under DFA. Liu *et al.* (2009) proposed a new configuration update mechanism in an ELP method for the problem of packing circles into a larger containing circle. Shanker and Bandyopadhyay (2011) put forward an MC-based temperature basin paving algorithm to determine low-energy structures of water clusters $(H_2O)_{20}$ and $(H_2O)_{25}$. Rakshit and Bandyopadhyay (2013) used the temperature basin paving MC method with effective fragment potential to find low-energy minima of $(H_2O)_{25}$ and $(H_2O)_{30}$. In this study, based on the original ELP (Hansmann and Wille, 2002) and its alternative version (Liu *et al.*, 2009), we put forward a new update mechanism of the histogram function in the ELP method and obtain a new improved ELP method. Then, by incorporating the gradient method (GM) into the improved ELP method, a new global search optimization method for SMLP, nELP, is proposed. Computational results show that the proposed algorithm is an effective method for solving SMLP.

2 Problem statement

To solve SMLP, we take a simplified international commercial communication satellite module, INTELSAT-III, as an example. The statement of the problem is as follows.

We need to put N objects in a cylindrical satellite module, which is a rotating cylindrical cavity within a shell (Fig. 1a) (Zhang *et al.*, 2008; Liu *et al.*, 2016). The upper and lower surfaces (P_1–P_4) (Fig. 1b) (Liu *et al.*, 2016) of two bearing plates attached to a standing column in the module are used to hold the N objects A_i (i=1, 2, …, N), which in this study are simplified as cuboids or cylinders. These objects are regarded as rigid bodies with the same mass allocation. The sizes, thicknesses, and masses of the two circular bearing plates are the same, and their radii are R_0.

The basic problem here is to optimize the moment of inertia of the whole module when the following technological conditions exist:

1. N objects must be placed inside the container, and any two objects must have no interference. Each object must not overlap with the module shell.

2. The centroid position error of the whole system should be within a given value range.

3. The inertia angle error of the whole system should not exceed a permissible value.

According to the characteristics of the satellite module and for convenience of description, four coordinate systems are defined as follows (Zhang *et al.*, 2008):

1. The reference coordinate system, $Oxyz$, which is used to calculate the centroid of the whole satellite and describe the positions of N objects. The surface Oxy is the satellite module's bottom surface.

O: the origin of the coordinate, which is at the center of the load-carrying shaft.

z: the longitudinal symmetric axis of the satellite. We assume the upward direction is positive when the satellite is launched.

x: the vertical to the longitudinal surface.

y: the composition of the right-hand rectangular coordinate system with x, z axes.

2. The planet coordinate system $O'x'y'z'$, which is used to calculate the moment of inertia of the whole system.

O': the origin of the coordinate, which is located at the centroid of the satellite.

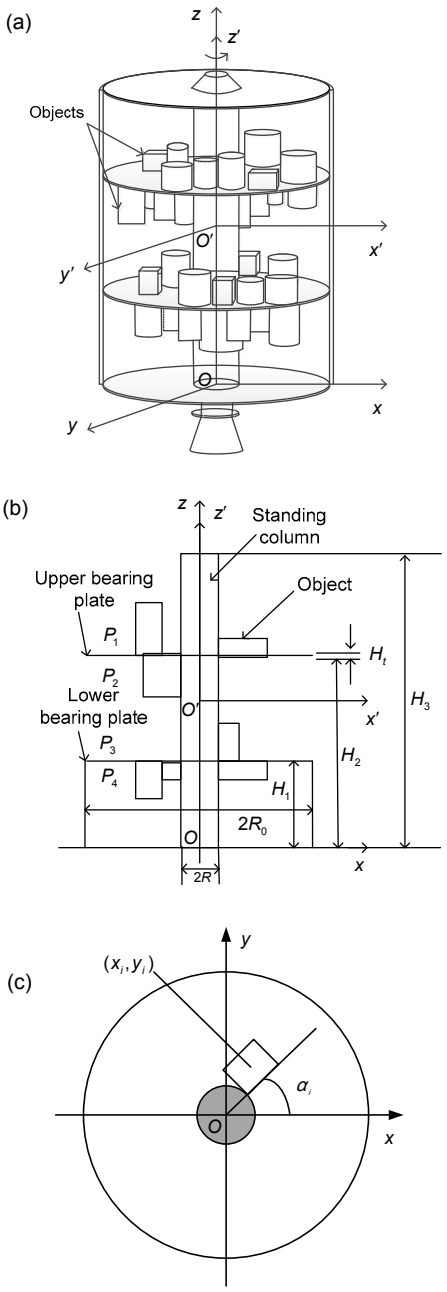

Fig. 1 Diagrammatic sketch of the simplified satellite module layout: (a) 3D layout; (b) 2D layout; (c) layout of one object on a bearing plate

z': the longitudinal symmetric axis of the satellite, which parallels (or coincides) with the z axis.

x', y' are parallel with the x and y axes, respectively.

3. The local coordinate system of object $O''x''y''z''$, which is used to calculate the moment of inertia of each object.

O'': the origin of coordinate, which is located at the centroid of each object.

x'', y'', and z'': the geometric symmetry axes of the object. Usually, z'' parallels with the z axis and x'' parallels with the long side of the rectangle. x'' and y'' have angles α_i ($0\leq\alpha_i\leq\pi/2$) (Figs. 1c and 2) (Zhang *et al.*, 2008; Liu *et al.*, 2016) with x and y axes, respectively. The counter-clockwise direction is positive. For the circle, α_i=0.

4. The inertia angle coordinate system $O'''x'''y'''z'''$, which is used to calculate the inertia angles of the whole system.

O''': the origin of coordinate, which coincides with O'. x''', y''', and z''' have angles ($\theta_{x'}$, $\theta_{y'}$, $\theta_{z'}$) with the x', y', and z' axes. The counter-clockwise direction is positive.

The above coordinate systems and their interrelationships are shown in Fig. 2.

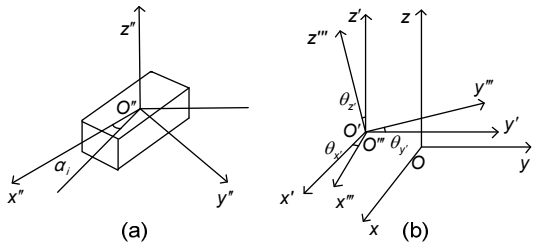

Fig. 2 Sketch map of coordinate systems: (a) coordinate system $O''x''y''z''$; (b) coordinate systems $Oxyz$, $O'x'y'z'$, and $O'''x'''y'''z'''$

According to the above analysis, SMLP for a simplified INTELSAT-III is described as follows:

We want to find a layout $\boldsymbol{X}$ which minimizes

$$f_1(\boldsymbol{X})=J_{x'}(\boldsymbol{X})+J_{y'}(\boldsymbol{X})+J_{z'}(\boldsymbol{X}), \tag{1}$$

and satisfies the non-overlap constraint

$$g_1(\boldsymbol{X})=\sum_{i=0}^{N-1}\sum_{l=i+1}^{N}\mathrm{int}(A_i)\cap\mathrm{int}(A_l)=\varnothing, \tag{2}$$

centroid position constraints

$$g_2(\boldsymbol{X})=|x_c-x_e|\leq 3.0, \tag{3}$$
$$g_3(\boldsymbol{X})=|y_c-y_e|\leq 3.0, \tag{4}$$
$$g_4(\boldsymbol{X})=|z_c-z_e|\leq 3.0, \tag{5}$$

and inertia angle constraints

$$g_5(\boldsymbol{X})=|\theta_{x'}(\boldsymbol{X})|\leq 0.03, \tag{6}$$
$$g_6(\boldsymbol{X})=|\theta_{y'}(\boldsymbol{X})|\leq 0.03, \tag{7}$$
$$g_7(\boldsymbol{X})=|\theta_{z'}(\boldsymbol{X})|\leq 0.03, \tag{8}$$

where $\boldsymbol{X}=(x_1, y_1, z_1, \alpha_1, x_2, y_2, z_2, \alpha_2, \ldots, x_N, y_N, z_N, \alpha_N)$. N is the number of layout objects. (x_i, y_i, z_i) are the 3D coordinates of object A_i. α_i is the orientation angle shown in Fig. 1c. $(J_{x'}(\boldsymbol{X}), J_{y'}(\boldsymbol{X}), J_{z'}(\boldsymbol{X}))$ is the moment of inertia of layout $\boldsymbol{X}$ of the satellite module with regard to $O'x'y'z'$. $\mathrm{int}(A_i)$ denotes the internal part of object A_i, and $\mathrm{int}(A_0)$ denotes the outside part of the shell of the satellite module. (x_c, y_c, z_c) and (x_e, y_e, z_e) are the actual and expected centroids, respectively, of the satellite module. We set x_e=y_e=0. $(\theta_{x'}(\boldsymbol{X}), \theta_{y'}(\boldsymbol{X}), \theta_{z'}(\boldsymbol{X}))$ are the inertia angles between the principal axis of inertia of the module and the satellite coordinate axes x', y', z' along each direction. The calculations of the moments of inertia, centroid position, and inertia angles were given by Zhang *et al.* (2008).

3 Basis for solving the problem

In this work, all objects of the tested instances had been distributed to the four bearing plate surfaces in advance according to the Hopfield neural network (Zhang *et al.*, 2008), so the proposed nELP algorithm needs only to determine their accurate location. We suppose the layout $\boldsymbol{X}=(\boldsymbol{X}_1, \boldsymbol{X}_2, \boldsymbol{X}_3, \boldsymbol{X}_4)=(x_{11}, y_{11}, z_{11}, \alpha_{11}, \ldots, x_{1n_1}, y_{1n_1}, z_{1n_1}, \alpha_{1n_1}; x_{21}, y_{21}, z_{21}, \alpha_{21}, \ldots, x_{2n_2}, y_{2n_2}, z_{2n_2}, \alpha_{2n_2}; x_{31}, y_{31}, z_{31}, \alpha_{31}, \ldots, x_{3n_3}, y_{3n_3}, z_{3n_3}, \alpha_{3n_3}; x_{41}, y_{41}, z_{41}, \alpha_{41}, \ldots, x_{4n_4}, y_{4n_4}, z_{4n_4}, \alpha_{4n_4})$, where $\boldsymbol{X}_1$, $\boldsymbol{X}_2$, $\boldsymbol{X}_3$, and $\boldsymbol{X}_4$ represent the layouts of objects allocated on the four surfaces P_1, P_2, P_3, and P_4, respectively, and n_1, n_2, n_3, and n_4 are the numbers of the objects allocated to each surface, $n_1+n_2+n_3+n_4=N$, the total number of objects. Our goal was to find a layout $\boldsymbol{X}$ satisfying all of the constraints given by Eqs. (2)–(8) to make the design object $f_1(\boldsymbol{X})$ as small as possible.

To make layouts $\boldsymbol{X}_1$–$\boldsymbol{X}_4$ of P_1–P_4 satisfy the non-overlap conditions, we introduce the quasi-physical strategy (Liu *et al.*, 2011). Imagine that all N objects and the shell of the satellite module are smooth elastic solids. Due to the segmentation effect of the bearing plate, once all objects are assigned to the plate surfaces P_1–P_4, any two objects on a different plate surface have no interference. So, we need only to calculate the overlapping depth between each

object and the fixed vessel, and the overlapping depth between any two different objects on the same plate surface. Obviously, the extrusive elastic potential energy of the whole satellite module is

$$\begin{aligned} f_2(\boldsymbol{X}) &= \sum_{i=0}^{N-1}\sum_{j=i+1}^{N} ud_{ij}^2 \\ &= f_2(\boldsymbol{X}_1)+f_2(\boldsymbol{X}_2)+f_2(\boldsymbol{X}_3)+f_2(\boldsymbol{X}_4) \\ &= \sum_{i=0}^{n_1-1}\sum_{j=i+1}^{n_1} ud_{ij}^2 + \sum_{i=0}^{n_2-1}\sum_{j=i+1}^{n_2} ud_{ij}^2 + \sum_{i=0}^{n_3-1}\sum_{j=i+1}^{n_3} ud_{ij}^2 + \sum_{i=0}^{n_4-1}\sum_{j=i+1}^{n_4} ud_{ij}^2, \end{aligned} \tag{9}$$

where u is a physical coefficient. We set u=1 in this study. $f_2(\boldsymbol{X}_1)$, $f_2(\boldsymbol{X}_2)$, $f_2(\boldsymbol{X}_3)$, and $f_2(\boldsymbol{X}_4)$ denote the extrusive elastic potential energy between all objects allocated on the corresponding plate surfaces P_1–P_4, respectively. Considering that all objects are cuboids or cylinders and that the module shell is cylindrical, we can transfer the computation of the overlapping depth in a 3D space to that in a 2D plane. d_{ij} (i, j=0, 1, …, N, $i\neq j$) is computed in the following five cases:

1. For two circular objects A_i and A_j, we define

$$d_{ij} = d_{ji} = \begin{cases} r_i + r_j - [(x_i - x_j)^2 + (y_i - y_j)^2]^{1/2}, \\ \quad \text{if } r_i + r_j > [(x_i - x_j)^2 + (y_i - y_j)^2]^{1/2}, \\ 0, \quad \text{else}, \end{cases} \tag{10}$$

where r_i is the radius of circular object A_i, and (x_i, y_i) are the coordinates of the center of circular object A_i.

2. For circular object A_i and the fixed vessel A_0, we define

$$d_{i0} = d_{0i} = \begin{cases} r_i + (x_i^2 + y_i^2)^{1/2} - R_0, \\ \quad \text{if } r_i + (x_i^2 + y_i^2)^{1/2} > R_0, \\ 0, \quad \text{else}. \end{cases} \tag{11}$$

3. If two rectangular objects A_l and A_m embed each other, d_{lm} is computed according to the following two cases: (1) If object A_m locates in object A_l (Fig. 3a), or object A_m crosses through object A_l (Fig. 3b), we define d_{ml}=d_{lm}=min{$|a_lc_m|$, $|b_ld_m|$, $|c_la_m|$, $|d_lb_m|$}. Here, $|a_lc_m|$ is the distance between vertices a_l and c_m, and so on. (2) If object A_l intersects object A_m (Figs. 3c–3d), d_{lm} is defined as the diagonal length of the intersecting rectangle formed by objects A_l and A_m. For example, in Fig. 3c, we define d_{lm}=d_{ml}=$|a_mc_l|$, and in Fig. 3d, d_{ml}=d_{lm}=$|a_mp_m|$, where p_m is a vertex of the intersecting rectangle.

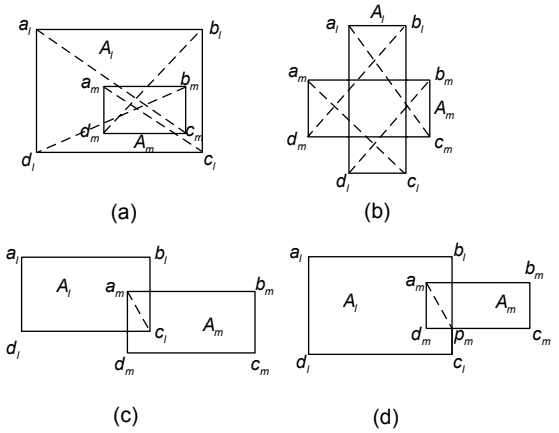

Fig. 3 The overlapping depth between two rectangular objects: (a) A_m within A_l; (b) A_l crossing through A_m; (c) A_m intersecting A_l from an angle; (d) A_m intersecting A_l from an edge

4. For rectangular object A_m and the fixed vessel A_0, we define

$$\begin{aligned} d_{m0} &= d_{0m} \\ &= \begin{cases} \left[\left(|x_m| + 0.5l_m\right)^2 + \left(|y_m| + 0.5w_m\right)^2\right]^{1/2} - R_0, \\ \quad \text{if } \left[\left(|x_m| + 0.5l_m\right)^2 + \left(|y_m| + 0.5w_m\right)^2\right]^{1/2} > R_0, \\ 0, \quad \text{else}, \end{cases} \end{aligned} \tag{12}$$

where l_m and w_m are the length and width, respectively, of rectangular object A_m, and (x_m, y_m) are the coordinates of the center of rectangular object A_m.

5. For circular object A_i and rectangular object A_m, an example of an overlapping boundary between A_i and A_m is shown in Fig. 4. o_i is the center point of circular object A_i. a_m, b_m, c_m, and d_m are the four vertices of the rectangular object A_m. (x_i, y_i) are the coordinates of the center of circular object A_i with radius r_i. (x_m, y_m) are the coordinates of the center of rectangular object A_m. l_m and w_m are the length and width, respectively, of rectangular object A_m. Obviously, the distances between o_i and a_m, b_m, c_m, and d_m are $|o_ia_m|$=$[(x_m-0.5l_m-x_i)^2+(y_m+0.5w_m-y_i)^2]^{1/2}$, $|o_ib_m|$=$[(x_m+0.5l_m-$

$x_i)^2+(y_m+0.5w_m-y_i)^2]^{1/2}$, $|o_ic_m|=[(x_m+0.5l_m-x_i)^2+(y_m-0.5w_m-y_i)^2]^{1/2}$, and $|o_id_m|=[(x_m-0.5l_m-x_i)^2+(y_m-0.5w_m-y_i)^2]^{1/2}$, respectively. The boundary is a rectangular shape with rounded corners (radius is r_i). If the center of circular object A_i resides inside the boundary, it is considered as an overlap with rectangular object A_m. The overlapping depth d_{im} between circular object A_i and rectangular object A_m is

$$d_{im}=d_{mi}=\begin{cases}\min(r_i+0.5l_m-|x_i-x_m|, r_i+0.5w_m-|y_i-y_m|),\\ \qquad\text{if } o_i \text{ is in region II or III (Fig. 4),}\\ r_i-\min(|o_ia_m|, |o_ib_m|, |o_ic_m|, |o_id_m|),\\ \qquad\text{if } o_i \text{ is in region I (Fig. 4),}\\ 0, \qquad\text{else.}\end{cases} \tag{13}$$

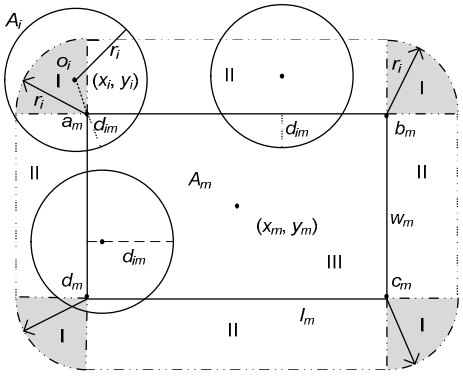

Fig. 4 The overlapping depth between circular object A_i and rectangular object A_m

By introducing the penalty function method, the model described above can be transformed into the unconstrained optimization problem below:

$$\min F(\boldsymbol{X})=\omega_1 f_1(\boldsymbol{X})+\omega_2 f_2(\boldsymbol{X})+\omega_3[g_2(\boldsymbol{X})+g_3(\boldsymbol{X})+g_4(\boldsymbol{X})]+\omega_4[g_5(\boldsymbol{X})+g_6(\boldsymbol{X})+g_7(\boldsymbol{X})], \tag{14}$$

where ω_1, ω_2, ω_3, and ω_4 are the weighting factors.

4 Energy landscape paving method for SMLP

4.1 Improved energy landscape paving method

The ELP method (Hansmann and Wille, 2002; Schug *et al.*, 2005) combines the tabu search algorithm (Glover, 1990a; 1990b) and the idea of energy landscape deformation, by defining a new energy function to avoid the recently visited region. In other words, if layout $\boldsymbol{X}$ is visited, the energy $F(\boldsymbol{X})$ is added by a punishment, given by $\tilde{F}(\boldsymbol{X})=F(\boldsymbol{X})+k*H(F(\boldsymbol{X}),t)$. Here, $H(F(\boldsymbol{X}), t)$ is the histogram function, k is a coefficient, and t is the step number of iterations. If energy $F(\boldsymbol{X})$ falls into a certain bin, the corresponding histogram function $H(F(\boldsymbol{X}), t)$ is increased by a value (set to 1), where a 'bin' refers to an entry of the histogram and all bins in the histogram are of the same size. In this study, the sampling weight is set to $w(\tilde{F}(\boldsymbol{X}))=\exp\{-\tilde{F}(\boldsymbol{X})/(k_BT)\}$ for layout $\boldsymbol{X}$, where k_BT is the thermal energy at the low temperature T, and k_B is the Boltzmann constant.

In the ELP method, when the simulation runs into a local minimum, the penalty term $H(F(\boldsymbol{X}), t)$ will increase with the time during which the system stays in that minimum; consequently, during which the sampling weight w of the local minimum layout will decrease, and the search will jump out of that local minimum to explore higher energy regions. However, it will then fall into a new local minimum. ELP can use the above method to get out of it. This process is repeated until the histogram becomes flat.

However, there may be some defects in the above ELP method. When a new layout $\boldsymbol{X}^2$ is generated from the current layout $\boldsymbol{X}^1$, the algorithm accepts $\boldsymbol{X}^2$ only by satisfying the condition random(0, 1)$<\exp\{|\tilde{F}(\boldsymbol{X}^1)-\tilde{F}(\boldsymbol{X}^2)|/(k_BT)\}$, where random(0, 1) is a random value from 0 to 1. However, according to this mechanism of acceptance, ELP may miss lower-energy layouts near $\boldsymbol{X}^1$. To avoid this, Liu *et al.* (2009) put forward an improved version of ELP. In the improved version, whether $\boldsymbol{X}^2$ is accepted is based on the following two judgments: (1) $F(\boldsymbol{X}^1)>F(\boldsymbol{X}^2)$ and (2) $F(\boldsymbol{X}^1)\le F(\boldsymbol{X}^2)$. For case (1), $\boldsymbol{X}^2$ is accepted undoubtedly. For case (2), if $\boldsymbol{X}^2$ satisfies the condition $\text{random}(0,1)<\exp\{|\tilde{F}(\boldsymbol{X}^1)-\tilde{F}(\boldsymbol{X}^2)|/(k_BT)\}$, $\boldsymbol{X}^2$ is still accepted; otherwise, $\boldsymbol{X}^2$ is given up and $\boldsymbol{X}^1$ is restored as the current layout.

Despite the above improvement, the ELP method still has some drawbacks. Once a new layout $\boldsymbol{X}^2$ is generated, the histogram function is updated no matter whether $\boldsymbol{X}^2$ is accepted. This will cause some newly produced and unaccepted layouts located at surrounding energy barriers of the minima to still not be accepted in subsequent simulations, because the accumulated histogram function $H(F(\boldsymbol{X}), t)$

increasingly modifies their surrounding energy barriers. Thus, it will be hard for the simulation to get away from the minima, especially from those that lay in narrow and deep valleys of the energy landscape. To overcome this, we propose a new update strategy for the histogram function: if $\boldsymbol{X}^2$ is accepted, we modify the histogram function value $H(F(\boldsymbol{X}^2), t)$ of $\boldsymbol{X}^2$, i.e., add 1 to the frequency of the bin into which the energy $F(\boldsymbol{X}^2)$ falls; if $\boldsymbol{X}^2$ is not accepted, to accelerate the process of jumping out of the local minimum $\boldsymbol{X}^1$, we modify similarly the histogram function value $H(F(\boldsymbol{X}^1), t)$ of $\boldsymbol{X}^1$. Thus, as the modified energy $F(\boldsymbol{X}^1)$ of the previous minimum layout $\boldsymbol{X}^1$ has gradually been increased, the simulation may easily jump out of that local minimum and accept the new layout $\boldsymbol{X}^2$.

4.2 Local search mechanism

In ELP, to accelerate the search for the global optimal layout, once a new layout $\boldsymbol{X}^2$ is generated from the current layout $\boldsymbol{X}^1$, we use a much finer local search method, the gradient method (GM), to search for the lower-energy layouts around $\boldsymbol{X}^2$.

GM is also named the steepest descending method. The search direction is the negative gradient direction. In this work, we use an adaptive step size as the search step size. That is to say, if the newly produced energy $F(\boldsymbol{X}')$ is higher than the former energy $F(\boldsymbol{X}^2)$ after one iteration, which means the step size is too large, we shrink h to h*0.8; otherwise, we keep h. In GM, the closer is the iteration to the minimum, the slower the decrease in potential energy becomes. To reduce the run time, an 'early-escape' strategy is introduced. If the newly generated energy $F(\boldsymbol{X}')$ is rather close to the former energy $F(\boldsymbol{X}^2)$, i.e., $|F(\boldsymbol{X}')-F(\boldsymbol{X}^2)|<10^{-4}$, then a local minimum may be reached. At that moment, if energy $F(\boldsymbol{X}')$ is less than 10, we continue to execute GM until a feasible layout or the given step size in GM is reached; otherwise, we finish the GM process in advance.

The detailed procedure of GM($\boldsymbol{X}^2$) is as follows:

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

2. Under layout $\boldsymbol{X}^2$, compute the gradient vector of energy $F(\boldsymbol{X}^2)$ for each object in x and y directions:

$$\nabla F(\boldsymbol{X}^2)=\left(\frac{\partial F(\boldsymbol{X})}{\partial x_1},\frac{\partial F(\boldsymbol{X})}{\partial y_1},\frac{\partial F(\boldsymbol{X})}{\partial x_2},\frac{\partial F(\boldsymbol{X})}{\partial y_2},\ldots,\frac{\partial F(\boldsymbol{X})}{\partial x_n},\frac{\partial F(\boldsymbol{X})}{\partial y_n}\right)\Bigg|_{\boldsymbol{X}=\boldsymbol{X}^2}. \tag{15}$$

Then, a new layout $\boldsymbol{X}'=\boldsymbol{X}^2-h*\nabla F(\boldsymbol{X}^2)$ is generated.

3. If $F(\boldsymbol{X}')\geq 10$, then

3.1 If $F(\boldsymbol{X}')>F(\boldsymbol{X}^2)$, then $h=h$*0.8;

3.2 Let $\boldsymbol{X}^2=\boldsymbol{X}'$, $\boldsymbol{X}'=\boldsymbol{X}^2-h*\nabla F(\boldsymbol{X}^2)$;

3.3 If $|F(\boldsymbol{X}')-F(\boldsymbol{X}^2)|\geq 10^{-4}$, go to 3.1.

4. If $F(\boldsymbol{X}')<10$, then

4.1 If $F(\boldsymbol{X}')>F(\boldsymbol{X}^2)$, then $h=h$*0.8;

4.2 Set $\boldsymbol{X}^2=\boldsymbol{X}'$, $\boldsymbol{X}'=\boldsymbol{X}^2-h*\nabla F(\boldsymbol{X}^2)$;

4.3 If $f_2(\boldsymbol{X})\geq 10^{-20}$, $h>h_{\min}$, then go to 4.1.

5. Output $\boldsymbol{X}'$.

4.3 Heuristic layout updating strategies

In the ELP method, each iterative step must update the current layout. For each layout area of the bearing plate surfaces, we use the following heuristic layout updating strategies:

Strategy 1 Choose the object with the maximum overlapping depth as the target object. Without loss of generality, suppose the chosen object is A_i. Put randomly the center of A_i at a point in the vacant region which all packed objects (except object A_i) have not occupied.

This strategy could generally reduce overlap between A_i and other placed objects.

Strategy 2 If object A_i is put randomly three times in the vacant region and every time the updated layout is not accepted, object A_i is marked. In the subsequent iterative step, we choose another unmarked object with the maximum overlapping depth to update the current layout.

4.4 Procedure of the layout method

By incorporating GM and heuristic layout updating strategies into the improved ELP method, a new global search algorithm nELP is proposed. In this study, we first use nELP to optimize the layout of objects on every bearing plate surface. Then, we combine the layout of objects on each bearing plate to obtain the layout solution for the whole satellite system. Here, we cite the P_1 bearing plate as the optimized layout area. The procedure of calculating nELP is outlined as follows:

1. In the layout area of the bearing plate P_1 with radius R_0, generate randomly j points as the centers of j objects, and gain an original layout $\boldsymbol{X}_1^1=(x_{11}, y_{11}, z_{11}, \alpha_{11}, \ldots, x_{1j}, y_{1j}, z_{1j}, \alpha_{1j})$. Notice that z_{1l} $(1\leq l\leq j)$ is determined after the objects have been allocated. So, we

need only to optimize the coordinates (x_{1l}, y_{1l}) and orientation angle α_{1l} of objects A_l, l=1, 2, …, j. Set t=1, T=5 K, and k=20. Compute $F(\boldsymbol{X}_1^1)$, and initialize the histogram function. Let $\tilde{F}(\boldsymbol{X}_1^1)=F(\boldsymbol{X}_1^1)+k*H(F(\boldsymbol{X}_1^1),t)$, and $\boldsymbol{X}_1^{\text{opt}}=\boldsymbol{X}_1^1$.

2. Choose the unmarked object with the maximum overlapping depth from layout $\boldsymbol{X}_1^1$, for example, object A_i, and set s=1.

3. Put randomly the center of A_i into the vacant region of P_1, and obtain a new layout $\boldsymbol{X}_1^2$.

4. Implement GM, and let $\boldsymbol{X}_1^2$=GM($\boldsymbol{X}_1^2$).

5. Compute $F(\boldsymbol{X}_1^2)$. If $F(\boldsymbol{X}_1^2)<F(\boldsymbol{X}_1^{\text{opt}})$, then let $\boldsymbol{X}_1^{\text{opt}}=\boldsymbol{X}_1^2$ and $F(\boldsymbol{X}_1^2)=F(\boldsymbol{X}_1^{\text{opt}})$.

6. If $F(\boldsymbol{X}_1^1)>F(\boldsymbol{X}_1^2)$, then accept layout $\boldsymbol{X}_1^2$, and let $\boldsymbol{X}_1^1=\boldsymbol{X}_1^2$, $F(\boldsymbol{X}_1^1)=F(\boldsymbol{X}_1^2)$, and update $H(F(\boldsymbol{X}_1^2), t)$, let $\tilde{F}(\boldsymbol{X}_1^2)=F(\boldsymbol{X}_1^2)+k*H(F(\boldsymbol{X}_1^2),t)$, go to step 9; otherwise, go to step 7.

7. If random(0,1)<exp{$|\tilde{F}(\boldsymbol{X}_1^1)-\tilde{F}(\boldsymbol{X}_1^2)|/(k_{\text{B}}T)$}, then accept layout $\boldsymbol{X}_1^2$, let $\boldsymbol{X}_1^1=\boldsymbol{X}_1^2$, $F(\boldsymbol{X}_1^1)=F(\boldsymbol{X}_1^2)$, and update $H(F(\boldsymbol{X}_1^2), t)$, let $\tilde{F}(\boldsymbol{X}_1^2)=F(\boldsymbol{X}_1^2)+k*H(F(\boldsymbol{X}_1^2),t)$, and go to step 9; otherwise, recover $\boldsymbol{X}_1^1$, and update $H(F(\boldsymbol{X}_1^1), t)$, let $\tilde{F}(\boldsymbol{X}_1^1)=F(\boldsymbol{X}_1^1)+k*H(F(\boldsymbol{X}_1^1),t)$, s=s+1, go to step 8.

8. If s>3, then mark object A_i, and go to step 2; otherwise, go to step 3.

9. If t<10^6, then t=t+1, and go to step 2; otherwise, output $\boldsymbol{X}_1^{\text{opt}}$ and $F(\boldsymbol{X}_1^{\text{opt}})$.

For the layout areas of bearing plates P_2, P_3, and P_4, we execute the nELP algorithm above, and obtain the optimal layouts $\boldsymbol{X}_2^{\text{opt}}$, $\boldsymbol{X}_3^{\text{opt}}$, and $\boldsymbol{X}_4^{\text{opt}}$, respectively. If layout $\boldsymbol{X}$=($\boldsymbol{X}_1^{\text{opt}}$, $\boldsymbol{X}_2^{\text{opt}}$, $\boldsymbol{X}_3^{\text{opt}}$, $\boldsymbol{X}_4^{\text{opt}}$) satisfies constraints (2)–(8), then we obtain a feasible layout scheme $\boldsymbol{X}$; otherwise, we start over again to run the nELP algorithm.

5 Experimental results and analysis

We tested two instances from the literature (Zhang *et al.*, 2008). We implemented the nELP algorithm in Java language and ran it on a PC with a 1.5 GHz processor and 2.0 GB RAM. For each instance, the nELP algorithm was run 50 times independently to optimize the layout of objects on four bearing plate surfaces.

5.1 Test instances

For the first instance, there were 60 objects located on the four bearing plate surfaces. The first 24 objects were cuboids and the remaining 36 were cylinders. The dimensions, masses, and allocation scheme of all objects were obtained from the second example given by Zhang *et al.* (2008). The 24 cuboids were placed orthogonally. So, the orientation angle α_i was 0 or π/2. The parameters of the satellite module were as follows: the radius of each bearing plate was R_0=500 mm, and the radius of the standing column was R=100 mm. The heights from the satellite base to the bottoms of the upper and lower bearing plates were H_2=830 mm and H_1=300 mm, respectively. The height of the standing column was H_3=1150 mm. The thickness of each bearing plate was H_{t}=20 mm. The mass of the satellite (including the module shell and the fixed components) was m=576.53 kg. In the reference coordinate system $Oxyz$, the mass center of the satellite module was C_{e}=(0, 0, 553.56) mm. The inertia of the satellite with no objects is given by

$$\boldsymbol{J}_0=\begin{pmatrix} J_x & J_{xy} & J_{xz} \\ J_{yx} & J_y & J_{yz} \\ J_{zx} & J_{zy} & J_z \end{pmatrix} =\begin{pmatrix} 352.2 & 0 & 0 \\ 0 & 352.2 & 0 \\ 0 & 0 & 106.8 \end{pmatrix} \text{kg}\cdot\text{m}^2, \tag{16}$$

where J_x, J_y, and J_z, the diagonal elements of the inertia matrix $\boldsymbol{J}_0$, are the moments of inertia of the empty module along the x, y, and z directions, respectively. The other non-diagonal elements of the inertia matrix $\boldsymbol{J}_0$ are called the products of inertia. Here, J_{xy}=J_{yx}=0, J_{xz}=J_{zx}=0, and J_{yz}=J_{zy}=0. The objective function is given in Eq. (14), where ω_1=10^{-2}, ω_2=10^3, ω_3=1, and ω_4=10^3.

For the second instance, there were 32 cylinders located on the four bearing plate surfaces. The dimensions, masses, and allocation scheme of all objects were obtained from the first example given by Zhang *et al.* (2008). The parameters of the satellite module were as follows: the radius of each bearing plate was R_0=505 mm, and the radius of the standing column was R=85 mm. The heights from the satellite base to the bottoms of the upper and lower bearing

plates were H_2=740 mm, H_1=240 mm, respectively. The height of the standing column was H_3=1000 mm. The thickness of each bearing plate was H_t=20 mm. The mass of the satellite (including the module shell and the fixed components) was m=776.5 kg. In the reference coordinate system $Oxyz$, the mass center of the satellite module was C_e=(0, 0, 500.0) mm. The inertia of the satellite with no objects was the same as that in instance 1. The weighting factors of each sub-objective function in Eq. (14), ω_1, ω_2, ω_3, and ω_4, were the same as those in instance 1.

5.2 Experimental results and comparisons

For the first instance, the numerical results obtained by nELP are listed in Table 1, in comparison with those obtained by using particle swarm optimization based on a pyramid model (PPSO) (Zhang and Teng, 2005), genetic algorithm/particle swarm optimization and quasi-principal component analysis (QPGP) (Zhang *et al.*, 2008), the cooperative coevolutionary genetic algorithm with a coarse-to-fine grain strategy (CCGA-CFG) (Chen and Teng, 2010), and the dual-system variable-grain cooperative coevolutionary genetic algorithm (DVGCCGA) (Teng *et al.*, 2010). From Table 1, for the moment of inertia, the average and best results obtained by nELP were superior to those obtained by the other four algorithms. For the overlap and the centroid position error, the average and best results obtained by nELP were equal or superior to those obtained by the other four algorithms. For the inertia angle error, the average value obtained by nELP was worse than that obtained by QPGP, but within the allowable error range given in Section 2, while the best result obtained by nELP was equal or superior to those obtained by the other four algorithms. Detailed data for the best layout obtained by nELP are given in Table 2. A 2D schematic diagram of the best layout obtained by nELP is shown in Fig. 5.

For the second instance, the numerical results from nELP are listed in Table 3. For the moment of the inertia, the overlap, the centroid position error, and the inertia angle error, the average and best results from nELP were equal or superior to those obtained by the Powell algorithm, genetic algorithm (GA), particle swarm optimization (PSO), genetic algorithm/particle swarm optimization (GAPSO), and QPGP in the literature (Zhang *et al.*, 2008).

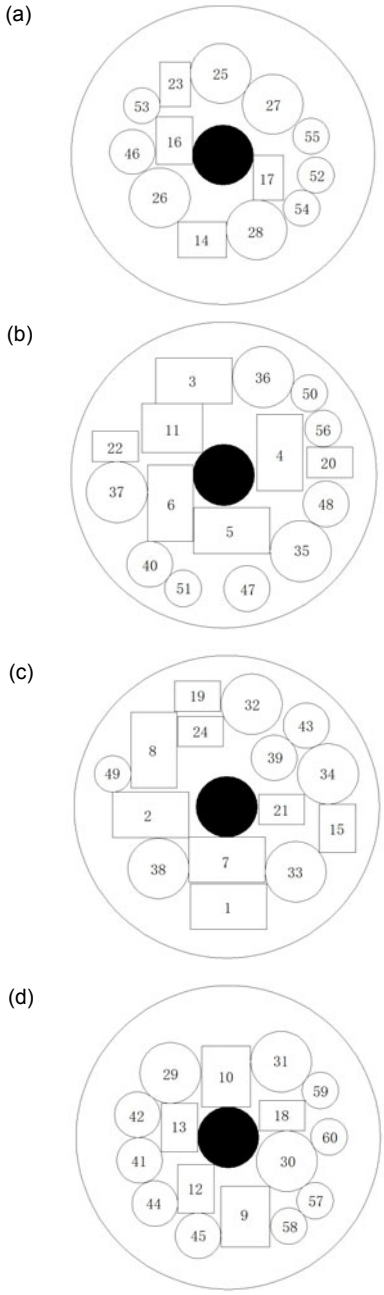

Fig. 5 A 2D schematic diagram of the best layout obtained by nELP for instance 1: (a) layout of objects on P_1; (b) layout of objects on P_2; (c) layout of objects on P_3; (d) layout of objects on P_4

Note that, compared with the average and best results reported in the literature, nELP reduces the moment of inertia by (844.1−795.06)/844.1×100%= 5.81% and (843.88−795.02)/843.88×100%=5.79%, respectively. For the average performance indexes of the overlap, the centroid position error, and the inertia

Table 1 Comparison of the average and best performance indexes obtained by PPSO, QPGP, CCGA-CFG, DVGCCGA, and nELP for instance 1

Algorithm	Moment of inertia (kg·mm^2)		Overlap (mm^2)		Centroid position error (mm)		Inertia angles error (rad)	
	Average	Best	Average	Best	Average	Best	Average	Best
PPSO	762.81	−	−	−	0.00	−	0.002	−
QPGP	773.58	758.33	18	0	0.13	0.000	0.000	0.000
CCGA-CFG	−	714.96	−	0	−	2.040	−	0.010
DVGCCGA	718.93	712.99	−	0	−	0.048	−	0.002
nELP	718.23	712.37	0	0	0.00	0.000	0.002	0.000

PPSO: particle swarm optimization based on a pyramid model; QPGP: hybrid method based on soft computing techniques, integrating genetic algorithm/particle swarm optimization (GA/PSO) and quasi-principal component analysis (QPCA); CCGA-CFG: cooperative coevolutionary genetic algorithm with a coarse-to-fine grain strategy; DVGCCGA: dual-system variable-grain cooperative coevolutionary genetic algorithm; nELP: new energy landscape paving method. The moment of inertia is the sum of the moments of inertia in the x', y', and z' axes, i.e., $\boldsymbol{J}_{x'}(\boldsymbol{X})+\boldsymbol{J}_{y'}(\boldsymbol{X})+\boldsymbol{J}_{z'}(\boldsymbol{X})$. The centroid position error is the sum of the centroid errors in the x, y axes, i.e., $|x_c-x_e|+|y_c-y_e|$. The inertia angle error is the sum of the inertia angles in the x', y', and z' axes, i.e., $|\theta_{x'}(\boldsymbol{X})|+|\theta_{y'}(\boldsymbol{X})|+|\theta_{z'}(\boldsymbol{X})|$

Table 2 The position data of the best layout obtained by nELP for instance 1

No.	x (mm)	y (mm)	α (rad)	Surface	No.	x (mm)	y (mm)	α (rad)	Surface	No.	x (mm)	y (mm)	Surface
1	6.56	−330.13	0	P_3	21	180.23	−8.54	0	P_3	41	−240.79	−217.84	P_4
2	−249.38	−26.13	0	P_3	22	−355.48	93.79	0	P_2	42	−298.63	74.59	P_4
3	−96.84	308.00	0	P_2	23	−157.28	236.34	π/2	P_1	43	260.79	268.59	P_3
4	183.25	72.89	π/2	P_2	24	−95.27	365.86	0	P_3	44	−291.48	−75.38	P_4
5	26.72	−182.36	0	P_2	25	−7.37	273.15	–	P_1	45	−99.13	−323.90	P_4
6	−175.00	−91.87	π/2	P_2	26	−208.42	−143.09	–	P_1	46	−298.99	10.63	P_1
7	0.92	−175.00	0	P_3	27	164.06	170.15	–	P_1	47	75.73	−371.85	P_2
8	−238.84	187.29	π/2	P_3	28	111.21	−250.65	–	P_1	48	334.85	−94.60	P_2
9	55.87	−260.36	π/2	P_4	29	−190.46	212.17	–	P_4	49	−373.84	108.87	P_3
10	−6.25	200.00	π/2	P_4	30	192.93	−78.23	–	P_4	50	280.09	267.73	P_2
11	−168.78	152.59	0	P_2	31	173.75	248.94	–	P_4	51	−133.34	−371.27	P_2
12	−105.79	−168.90	π/2	P_4	32	81.73	339.67	–	P_3	52	306.15	−67.28	P_1
13	−160.00	31.81	π/2	P_4	33	225.92	−214.81	–	P_3	53	−267.38	164.92	P_1
14	−68.79	−283.36	0	P_1	34	332.01	108.74	–	P_3	54	259.15	−177.69	P_1
15	363.14	−71.26	π/2	P_3	35	251.72	−249.59	–	P_2	55	289.88	63.00	P_1
16	−160.00	48.03	π/2	P_1	36	128.98	320.32	–	P_2	56	325.33	152.04	P_2
17	150.05	−75.65	π/2	P_1	37	−350.00	−56.21	–	P_2	57	285.66	−208.62	P_4
18	177.59	71.77	0	P_4	38	−224.08	−204.08	–	P_3	58	199.47	−292.12	P_4
19	−85.89	248.95	0	P_3	39	155.47	161.78	–	P_3	59	302.89	154.47	P_4
20	347.77	41.10	0	P_2	40	−242.53	−291.87	–	P_2	60	331.57	1.64	P_4

angle error, the nELP algorithm obtained results which were superior to those from the other five algorithms (Zhang *et al.*, 2008). For the best results of the overlap and the centroid position error from different algorithms, nELP, GAPSO, and QPGP obtained the optimal solutions which were missed by Powell, GA, and PSO. The least inertia angle error obtained by nELP was superior to those obtained by the other five algorithms. Detailed data for the best layout obtained by nELP are given in Table 4. A 2D schematic diagram of the best layout obtained by nELP is shown in Fig. 6.

Table 3 Comparison of the average and best performance indexes obtained by different methods in Zhang *et al.* (2008) and nELP for instance 2

Algorithm	Moment of inertia (kg·mm^2)		Overlap (mm^2)		Centroid position error (mm)		Inertia angle error (rad)	
	Average	Best	Average	Best	Average	Best	Average	Best
Powell	843.72	844.17	73 426	47 786	0.44	5.26	0.018	0.048
GA	846.81	845.99	27 578	7846	0.36	1.61	0.045	0.709
PSO	846.40	847.27	49 675	24 391	0.40	1.25	0.081	0.708
GAPSO	844.40	843.88	9254	0	0.07	0.00	0.048	0.021
QPGP	844.10	843.88	3093	0	0.02	0.00	0.049	0.008
nELP	795.06	795.02	0	0	0.00	0.00	0.000	0.000

GA: genetic algorithm; PSO: particle swarm optimization; GAPSO: genetic algorithm/particle swarm optimization; QPGP: hybrid method based on soft computing techniques, integrating GAPSO and quasi-principal component analysis (QPCA); nELP: new energy landscape paving method. The moment of inertia is the sum of the moments of inertia in the x', y', and z' axes, i.e., $\boldsymbol{J}_{x'}(\boldsymbol{X})+\boldsymbol{J}_{y'}(\boldsymbol{X})+\boldsymbol{J}_{z'}(\boldsymbol{X})$. The centroid position error is the sum of the centroid errors in the x, y axes, i.e., $|x_c-x_e|+|y_c-y_e|$. The inertia angle error is the sum of the inertia angles in the x', y', and z' axes, i.e., $|\theta_{x'}(\boldsymbol{X})|+|\theta_{y'}(\boldsymbol{X})|+|\theta_{z'}(\boldsymbol{X})|$

Table 4 The position data for the best layout obtained by nELP for instance 2

No.	x (mm)	y (mm)	Surface	No.	x (mm)	y (mm)	Surface	No.	x (mm)	y (mm)	Surface
1	285.18	306.16	P_1	12	293.61	298.07	P_3	23	−295.22	14.13	P_2
2	305.94	−285.41	P_1	13	−285.74	−305.64	P_4	24	295.23	−14.12	P_2
3	−285.35	−305.99	P_1	14	−305.43	285.95	P_4	25	−2.32	295.79	P_3
4	−305.92	285.32	P_1	15	285.93	305.45	P_4	26	295.70	2.35	P_3
5	−281.53	309.52	P_2	16	305.42	−285.92	P_4	27	2.31	−295.82	P_3
6	309.49	281.56	P_2	17	10.41	−295.72	P_1	28	−295.64	−2.30	P_3
7	281.32	−309.71	P_2	18	−10.39	295.72	P_1	29	−9.89	295.73	P_4
8	−309.49	−281.51	P_2	19	−295.36	−10.42	P_1	30	9.87	−295.73	P_4
9	298.07	−293.62	P_3	20	295.36	10.40	P_1	31	295.38	9.90	P_4
10	−298.28	293.40	P_3	21	14.12	295.56	P_2	32	−295.38	−9.88	P_4
11	−293.70	−297.99	P_3	22	−14.10	−295.56	P_2				

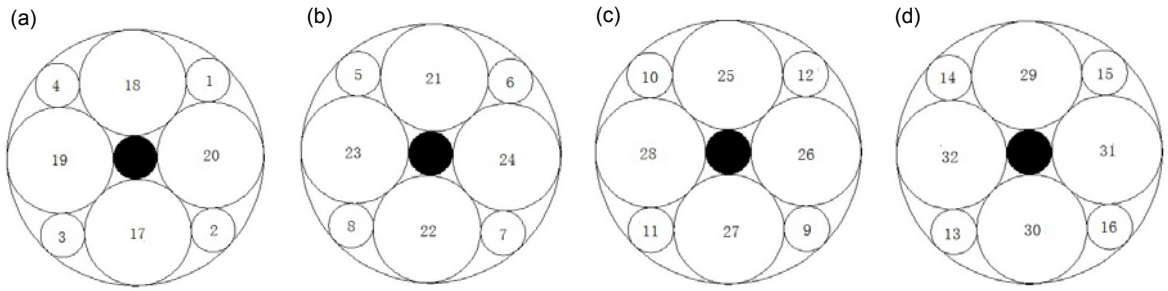

Fig. 6 A 2D schematic diagram of the best layout obtained by nELP for instance 2: (a) layout of objects on P_1; (b) layout of objects on P_2; (c) layout of objects on P_3; (d) layout of objects on P_4

6 Conclusions

This paper focused on the satellite module layout problem (SMLP) with performance constraints. First, we converted the problem into an unconstrained optimization problem by the quasi-physical strategy and the penalty function method. To avoid the energy landscape paving (ELP) method becoming trapped in the narrow and deep valleys of the energy landscape, a new update mechanism for the histogram function

in the ELP method was put forward. By incorporating the gradient method (GM) with local search and some heuristic layout updating strategies in the improved ELP method, a new global search method nELP was proposed. Borrowing a layout from the literature, the nELP algorithm was introduced mainly to determine the accurate location of objects on every bearing plate surface. Numerical results illustrate that the nELP algorithm is a promising method for SMLP. Moreover, it is easy to see from the construction process of nELP that combining a stochastic algorithm, a local search method, and some heuristic strategies could be an effective way to design a high-performance algorithm in a certain field.

References

Bansal, N., Han, X., Iwama, K., *et al.*, 2013. A harmonic algorithm for the 3D strip packing problem. *SIAM J. Comput.*, **42**(2):579-592. http://dx.doi.org/10.1137/070691607

Cagan, J., Shimada, K., Yin, S., 2002. A survey of computational approaches to three-dimensional layout problems. *Comput. Aided Des.*, **34**(8):597-611. http://dx.doi.org/10.1016/S0010-4485(01)00109-9

Chen, Y., Teng, H.F., 2010. Coevolutionary algorithm with coarse-to-fine grain strategy and its application to layout design of satellite module. *J. Dalian Univ. Tech.*, **50**(6): 931-936 (in Chinese).

Cuco, A.P.C., de Sousa, F.L., Neto, A.J.S., 2014. A multi-objective methodology for spacecraft equipment layouts. *Optim. Eng.*, **16**(1):165-181. http://dx.doi.org/10.1007/s11081-014-9252-z

del Valle, A.M., de Queiroz, T.A., Miyazawa, F.K., *et al.*, 2012. Heuristics for two-dimensional knapsack and cutting stock problems with items of irregular shape. *Expert Syst. Appl.*, **39**(16):12589-12598. http://dx.doi.org/10.1016/j.eswa.2012.05.025

Galiev, S.I., Lisafina, M.S., 2013. Linear models for the approximate solution of the problem of packing equal circles into a given domain. *Eur. J. Oper. Res.*, **230**(3):505-514. http://dx.doi.org/10.1016/j.ejor.2013.04.050

Glover, F., 1990a. Tabu search—part I. *Informs J. Comput.*, **1**(1): 89-98.

Glover, F., 1990b. Tabu search—part II. *Orsa J. Comput.*, **2**(1): 4-32.

Hamacher, K., 2007. Energy landscape paving as a perfect optimization approach under detrended fluctuation analysis. *Phys. A*, **378**(2):307-314. http://dx.doi.org/10.1016/j.physa.2006.11.071

Hansmann, U.H.E., Wille, L.T., 2002. Global optimization by energy landscape paving. *Phys. Rev. Lett.*, **88**(6):068105.

He, K., Zeng, M.D., 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. http://dx.doi.org/10.3724/SP.J.1013.2013.01224

Huo, J.Z., Shi, Y.J., Teng, H.F., 2006. Layout design of a satellite module using a human-guided genetic algorithm. IEEE Int. Conf. on Computational Intelligence and Security, p.230-235.

Jansen, K., Prädel, L., 2014. A new asymptotic approximation algorithm for 3-dimensional strip packing. *LNCS*, **8327**: 327-338. http://dx.doi.org/10.1007/978-3-319-04298-5_29

Li, Z.Q., Zhang, H.L., Zheng, J.H., *et al.*, 2011. Heuristic evolutionary approach for weighted circles layout. *Commun. Comput. Inform. Sci.*, **86**:324-331. http://dx.doi.org/10.1007/978-3-642-19853-3_47

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., Geng, H.T., 2011. A new heuristic algorithm for the circular packing problem with equilibrium constraints. *Sci. China Inform. Sci.*, **54**(8):1572-1584. http://dx.doi.org/10.1007/s11432-4351-3

Liu, J.F, Hao, L., Li, G., *et al*., 2016. Muti-objective layout optimization of a satellite module using the Wang-Landau sampling method with local search. *Front. Inform. Technol. Electron. Eng.*, **17**(6):527-542. http://dx.doi.org/10.1631/FITEE.1500292

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

Lodi, A., Martello, S., Monaci, M., 2002. Two-dimensional packing problems: a survey. *Eur. J. Oper. Res.*, **141**(2): 241-252. http://dx.doi.org/10.1016/S0377-2217(02)00123-6

Martello, S., Vigo, D., 2000. The three-dimensional bin packing problem. *Oper. Res.*, **48**(2):256-267.

Rakshit, A., Bandyopadhyay, P., 2013. Finding low energy minima of $(H_2O)_{25}$ and $(H_2O)_{30}$ with temperature basin paving Monte Carlo method with effective fragment potential: new 'global minimum' and graph theoretical characterization of low energy structures. *Comput. Theor. Chem.*, **1021**:206-214. http://dx.doi.org/10.1016/j.comptc.2013.07.023

Schug, A., Wenzel, W., Hansmann, U.H.E., 2005. Energy landscape paving simulations of the trp-cage protein. *J. Chem. Phys.*, **122**(19):194711.

Shanker, S., Bandyopadhyay, P., 2011. Monte Carlo temperature basin paving with effective fragment potential: an efficient and fast method for finding low-energy structures of water clusters $(H_2O)_{20}$ and $(H_2O)_{25}$. *Phys. Chem. A*., **115**(42):11866-11875. http://dx.doi.org/10.1021/jp2073864

Silveira, M.E., Vieira, S.M., Sousa, J.M.D.C., 2013. An ACO algorithm for the 3D bin packing problem in the steel industry. *LNCS*, **7906**:535-544. http://dx.doi.org/10.1007/978-3-642-38577-3_55

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

Sun, Z.G., Teng, H.F., Liu, Z., 2003. Several key problems in automatic layout design of spacecraft modules. *Prog. Nat. Sci.*, **13**(11):801-808.

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

Thomas, J., Chaudhari, N.S., 2014. Design of efficient packing system using genetic algorithm based on hyper heuristic approach. *Adv. Eng. Softw.*, **73**(5):45-52. http://dx.doi.org/10.1016/j.advengsoft.2014.03.003

Tsai, J.F., Wang, P.C., Lin, M.H., 2014. A global optimization approach for solving three-dimensional open dimension rectangular packing problems. *Optimization*, **64**(12):1-18. http://dx.doi.org/10.1080/02331934.2013.877906

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

Wang, Y.S., Yue, B.X., Teng, H.F., *et al.*, 2011. Satellite payloads configuration design using multi-agent system based on ant colony optimization. IEEE Int. Conf. on soft Computing and Pattern Recognition, p.336-341.

Zhan, L.X., Jeff, Z., Chen, Y., *et al.*, 2006. Monte Carlo basin paving: an improved global optimization method. *Phys. Rev. E.*, **73**(1pt2):015701.

Zhang, B., Teng, H.F., 2005. Particle swarm optimization based on pyramid model for satellite module layout. *Chin. J. Mech. Eng.*, **18**(4):530-536.

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., 2005. An effective hybrid algorithm for the problem of packing circles into a larger containing circle. *Comput. Oper. Res.*, **32**(8):1941-1951. http://dx.doi.org/10.1016/j.cor.2003.12.006