Int J Adv Manuf Technol (2013) 68:1155–1166
DOI 10.1007/s00170-013-4908-y

ORIGINAL ARTICLE

An energy-based surface flattening method for flat pattern development of sheet metal components

Qiaosheng Liu · Juntong Xi · Zhuoqi Wu

Received: 27 September 2012 / Accepted: 8 March 2013 / Published online: 2 April 2013

Abstract Sheet metal parts are widely used in various industry branches. The flat pattern development of sheet metal components is a prerequisite for sheet metal fabrication. The conventional graphic solution or the analytical method carried out manually is laborious, time-consuming, and lacks accuracy. The feature-based flattening could help to achieve the objective of automatic pattern generation; however, it is only suitable for bend formable sheet metal components composed of flat and simple curved surface. Inspired by its wide applications in cloth and shoemaking industries, the energy-based flattening for mesh surface is then introduced into the flat pattern derivation of sheet metal parts in this paper. A simplified energy model is proposed as our physical model. An innovative energy relaxation process based on variable step size is introduced to promote efficiency. Some key issues like seed triangle selection and overlap correction are also concerned. For those complex surfaces, a hierarchy flattening methodology is proposed to release the flattened distortions. A comparison between the energy method and the classic as-rigid-as-possible mesh parameterization is made to show the effectiveness of our method. The numerical experimental results show that the energy-based flattening method is a practical, intelligent, and automatic approach for flat pattern generation of sheet metal components.

Keywords Sheet metal components · Energy-based · Surface flattening · Flat pattern development

Q. Liu · J. Xi · Z. Wu
State Key Laboratory of Mechanical System and Vibration, Shanghai Jiaotong University, Shanghai 200240, China

J. Xi (✉)
Major Laboratory of Network Manufacturing and Information Enterprise of Shanghai, Shanghai 200030, China
e-mail: jtxi@sjtu.edu.cn

1 Introduction

Sheet metal parts are widely used in various industry branches like the car manufacturing, shipbuilding, and so on. Flat pattern development of sheet metal components is a prerequisite for sheet metal fabrication and it plays an important role in the whole manufacturing processes. The conventional flat pattern development methods mainly consist of the graphic solution and the analytic flattening. The graphic solution largely relies on the experience of skillful engineers. As it is carried out manually, it is laborious, time-consuming, and lacks accuracy. Compared with the graphic one, the analytical method is more efficient and accurate with the aid of computer. However, it is only suited for analytical surface while it is helpless for freeform complex surface. Aiming at these above shortages, many researches on automatic flat pattern generation have been done for decades. Among them, a surface model was usually extracted from its original 3D solid model [1]. Various feature recognition methodologies were then adopted to extract the target features [1–7]. Shunmugam and Kannan [8] extracted planar and curved features for sheet metal components automatically from the orthographic projections. Joshi and Chang [9] developed a graph approach named as attributed adjacency graph for feature recognition. The attributed adjacency graph [2, 3, 8] or the face-edge graph [1] represented the connection relations of the adjacent features. Bend allowance considering the manufacturing process was also incorporated into the development to ensuring the flattening accuracy [10–12]. After some translation or rotation operations for some specific planes, the entire flat pattern was then obtained automatically. Taking geometry shape into consideration, Prasad and Selvaraj [13, 14] had made comprehensive researches on the loft generation of sheet metal components specific to the aircraft industry. The various geometry features were subdivided into principal flat surface, non-principal flat

surface, flanges, holes, bend relief, cutouts, etc. [15]. Various practical methods were then developed according to the geometry characteristics of the sheet metal components. Process requirements such as bend allowances were also fully considered to obtain a computer-aided flat pattern development system. Although all the above-mentioned methods have been employed in actual production with success, they are only suitable for bend formable sheet metal components composed of flat and simple curved surface. A universal method on how to efficiently convert a specified 3D sheet metal part with relatively complex surfaces into the requisite 2D patterns is still worth studying.

With the rapid development of surface modeling and 3D scanning techniques, 3D mesh model is gradually becoming a popular surface representation especially for the complex freeform surface. If we substitute the surface model depicted by lines and arcs with the mesh model, the flat pattern development will become a process of flattening every elemental mesh or grid of the original 3D model onto the 2D plane. If every mesh could be flattened onto the plane with little shape distortion and stitched together to form an entire patch, it is obvious that we can also achieve the objective of automatic flat pattern generation from another point of view. Actually, such a surface flattening methodology has already been introduced into the textile and shoemaking industries. A new flat pattern development method named energy-based flattening has been widely adopted. In order to lower the distortion of the mapped image, Maillot et al. [16] viewed the whole polygonized surface as a spring net and firstly introduced a simple length energy which was initially lying on the surface. An energy-minimization process was then taken to reduce the distortions of the mapped image. Aiming at deriving pattern from the intended final garment shape, McCartney et al. [17] incorporated a similar energy model in terms of the strain energy required to deform the edges of the triangular mesh. They made a comprehensive and pioneering study on energy-based surface flattening by sequentially placing the nodes onto the plane using unconstrained or constrained type. An orthogonal trial movement for nodes was conducted to relax energy in partial flattening. As at every step the nodes moved at an incremental distance in four orthogonal directions with a trial, the energy relaxation process was then very time-consuming. Whereafter, they successively subdivided the total energy into the shear energy and the strain energy by considering the property of material so that it could better fit the cloth-making industry [18, 19]. Wang et al. [20] formally provided a planar triangular spring–mass system to simulate a dynamic energy-releasing process. Two criteria of evaluating the accuracy of a resulting developed surface were proposed and the Lagrange equation was applied in the energy model to release energy. The energy relaxation algorithm then became a process of solving the Lagrange equations of motion, which made it very complex and really took time. Other issues like finding cutting paths were also considered in [21, 22]. Li et al. [23] made minor revisions to the traditional mass–spring model by adding crossed springs. They used triangle strips to enhance flattening efficiency while overlap correction problem was also addressed. Other works utilizing the energy-based methodology can be found in [24–26].

Inspired by its wide applications in clothing and shoemaking industries, we decide to introduce the energy-based method into flat pattern development of sheet metal parts. The methodology we take is to flatten the 3D meshed surface onto the plane with the goal of keeping the shape of every triangular mesh and the surface area of the entire developed pattern as less distorted as possible. Considering the inefficiency of the energy relaxation process mentioned before, we propose a simplified energy model. An innovative energy relaxation algorithm based on variable step size is then developed. It is simple, effective, and easy to implement compared with the traditional ones. When dealing with the complex surface with a relatively high Gaussian curvature, seams or cuts are usually introduced into the flattening process to reduce the stretch or distortion in cloth pattern generation. Considering the forming process of some certain sheet metal part, such a relaxation method will be unsuitable. In this paper, we present a hierarchy flattening method to ease the distortion of some complex flattened surfaces. As we introduce the concept of energy, we call the energy method the physical approach. From another perspective, the triangulated surface flattening can be viewed as a mapping between each point on the domain and its corresponding point on the mesh. It is called mesh parameterization and has a wide application in computer graphics. As for the details of the mesh parameterization, readers can refer to references [27, 28]. Compared with the former energy method, the mesh parameterization was conducted completely from a geometrical point. We then name it geometrical approach. In this paper, we will make a comparison between both two methods and figure out their advantages and disadvantages.

In the following sections, we will present a detailed implementation of a modified energy-based surface flattening and its applications in flat pattern development of sheet metal components. The classical energy model adopted by Wang et al. [20], the simplified energy model we proposed and the flattening process are fully illustrated in Section 2. An innovative energy relaxation and the hierarchy flattening are also presented in this section. In Section 3, we conduct some numerical experiments on some classic sheet metal parts to verify the energy-based method. A comparison between this physical approach and the geometrical approach is also made. Finally, we will make some discussions in Section 3 and come to a conclusion in Section 4.

2 A modified energy-based surface flattening

In this section, a comprehensive implementation of the energy-based surface flattening will be conducted. A traditional energy model presented in the existing studies is first introduced as our theory foundation. The main drawback of adopting this model is also illustrated after a detailed description. To overcome its defects, a simplified energy model is proposed so that it can be utilized in the following energy relaxation algorithm. A classical flattening process is then presented and the related key issues are fully addressed. As the method is based on optimal local positioning of projected nodes by a process of sequential addition of the nodes, one of its significant shortcomings is that the optimization process will be relatively time-consuming. Therefore, a good optimization algorithm is very essential. In this section, an innovative energy relaxation algorithm based on our proposed simplified energy model is then developed to promote efficiency. Meanwhile, we bring out the methodology of hierarchy flattening to reduce the stretch of the flattened pattern.

2.1 Energy model

2.1.1 The traditional energy model

In the existing studies, the traditional energy model is represented with a planar triangular spring–mass system to simulate and control the shape of triangular mesh. The whole model consists of virtual masses and each of them is connected with its neighbors by springs. The extensional or compressive deformation of the springs produces the internal force which resists in tension or compression. The underlying elastic deformation energy is only associated with the deformation of the springs. Most of the physical quantities such as mass and force are derived from their corresponding geometric quantities as described below [20]. Accordingly, in the 3D mesh surface model, the nodes are abstracted as masses with certain quantities while the edges as springs with a unified elastic modulus. As shown in Fig. 1, the node V_0 is linked with its adjacent node V_i by the edge (V_0, V_i). The movement of its position (V_0) is restricted by the elastic force f (V_0, V_i) generated by the connected springs. The distortions of the edge length will then introduce the elastic deformation energy E (V_0, V_i). For each node V_i alone, the total elastic force $\overrightarrow{f(v_i)}$ and the associated total elastic energy E (V_i) are calculated as in Eqs. 1 and 2.

$$\overrightarrow{f}(v_i) = \sum_{j=1}^{n} c\left(\left|V_iV_j\right| - d_j\right)\overrightarrow{n}v_iv_j \tag{1}$$

$$E(V_i) = \sum_{j=1}^{n} \frac{1}{2} c\left(\left|V_iV_j\right| - d_j\right)^2 \tag{2}$$

where c is the spring constant, $\left|V_iV_j\right|$ is the distance between flattened nodes P_i and P_j on the planar surface, d_j is the distance of their corresponding original nodes in 3D surface, $\overrightarrow{n}v_iv_j$ is unit vector pointing from P_i to P_j, and n is the number of nodes connected to P_i. In order to get a global least-distorted flattening result, we have to minimize the total energy function E_{total} as defined in Eq. 3, where m is the total number of nodes.

$$E_{\text{total}} = \sum_{i=1}^{m} E(V_i) \tag{3}$$

Assume that the surface area density is distributed evenly and the surface mass is concentrated on its neighbored nodes. Thus, the mass m_i of node V_i can be written as

$$m_i = \frac{1}{3}\rho \sum A_j \tag{4}$$

where ρ is the area density of the surface, and A_j is the area of the triangle j which contains the node V_i.

Under the action of the resultant force $\overrightarrow{f(v_i)}$, the displacement of the particle V_i with the mass m_i occurs. The node moves at the directions of the resultant force so that the springs will return. Based on Newton's law, the relation between the acceleration $\ddot{x}_i$ of the node V_i and its acting force can be expressed as

$$\ddot{x}_i(t) = \frac{f_i(t)}{m_i}. \tag{5}$$

As it is a second-order differential equation, the Euler method [20] or the Verlet method [23] can be applied to solve it. Given the position $x_i(t_n)$, velocity $\dot{x}_i(t_n)$, and acceleration $\ddot{x}_i(t_n)$ at a time step n, Eq. 5 can be further written as

$$\dot{x}_i(t_{n+1}) = \dot{x}_i(t_n) + \Delta t\,\ddot{x}_i(t_n) \tag{6}$$

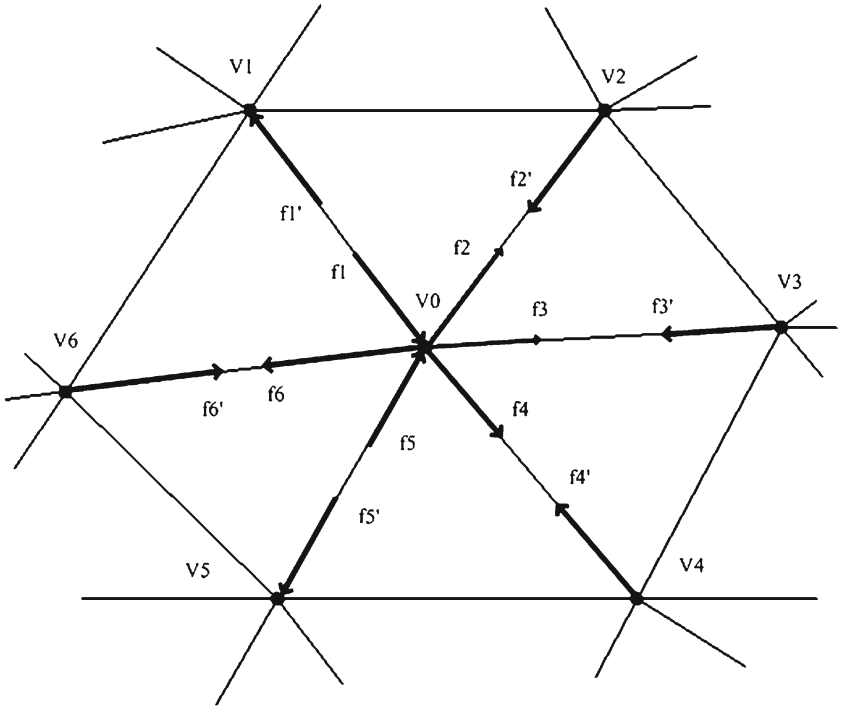
$$x_i(t_{n+1}) = x_i(t_n) + \Delta t\,\dot{x}_i(t_n) + \frac{\Delta t^2}{2}\ddot{x}_i(t_n) \tag{7}$$

At the end of each step, the resultant force is renewed by Eq. 1. The new position at the next step is then obtained by Eqs. 5, 6, and 7. The iterative process continues until the node reaches a relative equilibrium position, which means that the total elastic energy has been reduced to an acceptable level.

To evaluate the final flattening result, we take the overall length accuracy and area accuracy as the two measurement criteria. The length accuracy ε_l and the area accuracy ε_A are calculated in Eqs. 8 and 9.

$$\varepsilon_l = \frac{\sum \left|l - l^0\right|}{\sum l^0} \tag{8}$$

Fig. 1 The energy model



$$\varepsilon_A = \frac{\sum |A - A^0|}{\sum A^0} \tag{9}$$

where l^0 is the original length of the edge on the 3D surface, l is the corresponding edge length on the flattened surface, A^0 is the original area of the triangular patch on the 3D surface, and A is the area of its corresponding flattened triangular patch.

As has been argued, all the above-mentioned physical quantities such as spring constant and area density do not have their real physical meanings. In the existing researches, each of them is set to an appropriate value. They serve as scale factors so that the energy relaxation process can run smoothly. Therefore, there is no real physical meaning to calculate the exact value of the residue total elastic energy. As the change of the total elastic energy only depends on the deformations of the edge lengths, we prefer to calculate the length and area accuracy rather than the value of the total elastic energy in real work. Both of them are further chosen as termination criteria for the flattening procedure. The permissible area accuracy ϕ_A and the permissible length accuracy ϕ_l are predefined. The flattening procedure goes on until the following constraints are fulfilled:

$$\varepsilon_A \leq \phi_A \tag{10}$$

$$\varepsilon_l \leq \phi_l \tag{11}$$

2.1.2 A simplified energy model

As stated above, the traditional energy model based on the typical spring–mass system has been fully presented. The biggest characteristic of this model is that all the masses (nodes) and the springs (edges) strictly obey Newton's laws. The position of the node can only be determined after solving its corresponding equations of motion. Among the equations, all the related physical parameters need to be assigned with a suitable value to obtain an effective pattern. Taking the spring constant c in Eq. 1 for example, the accuracy of local surface regions will be affected when the constant changes [20]. Moreover, the parameters in the motion equations will also impact on the flattening procedure. Taking the time step Δt in Eq. 6 for example, only a suitable value can lead to a stable result with an acceptable speed. To sum up, tedious calculations are required to obtain a reliable flattening result based on this traditional energy model.

In order to reduce the tedious work such as choosing an appropriate value for the spring constant, we decide to propose a simplified energy model which only concerns on the resultant force direction and the maximum edge length distortion. As for calculating the resultant force direction, we simply set the spring constant value to 1 and then Eq. 1 can be rewritten as

$$\overrightarrow{f}(v_i) = \sum_{j=1}^{n} \left(|V_i V_j| - d_j\right) \overrightarrow{n} v_i v_j \tag{12}$$

The unit vector of the resultant force direction $\overrightarrow{n}(v_i)$ then can be easily obtained as

$$\overrightarrow{n}(v_i) = \frac{\overrightarrow{f}(v_i)}{\left|\overrightarrow{f}(v_i)\right|} \tag{13}$$

Based on this simplified energy model, we will propose an innovative energy relaxation algorithm in Section 2.3.2, which will largely improve the convergence speed. The details can be found below.

Thus, energy model has been established, which provides theoretical basis for the following flattening process, and the flattening process will be illustrated in the next paragraph.

2.2 Flattening process

The procedure of the flattening algorithm is illustrated in Fig. 2. In one word, the flattening process can be viewed as a process of sequential addition of the nodes. A certain triangular mesh, which is usually located closest to an average position within the given surface, is selected as the seed triangle at the initial stage. It is firstly flattened onto the plane without any distortions. After it has been flattened, its adjacent triangles are then chosen and flattened one by one. The whole flattening process terminates until the whole 3D mesh has been unfolded.

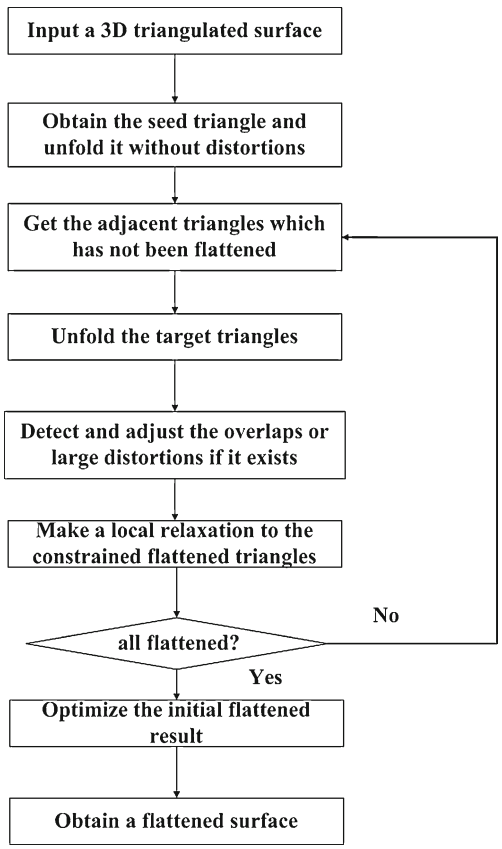


Fig. 2 Flow chart of flattening process

During the flattening process, there are two kinds of flattening: unconstrained triangle flattening and constrained triangle flattening [17] (see Fig. 3.). The first situation happens when trying to flatten the triangle F in Fig. 3b. One edge ($\mathbf{v_1}$, $\mathbf{v_2}$) of the triangle F has already been flattened while the third node $\mathbf{v_0}$ is going to be located onto the plane. In this situation, the position $\mathbf{v_0}$ on 2D flattening pattern could be uniquely determined by the intersection of two circles. The two circles have origins at $\mathbf{v_1}$ and $\mathbf{v_2}$ with radii r1 and r2, respectively, which are the lengths of the corresponding edges on the original 3D triangulated surface as shown in Fig. 3a. Since the lengths of the other two edges (edge ($\mathbf{v_0}$, $\mathbf{v_1}$) and edge ($\mathbf{v_0}$, $\mathbf{v_2}$)) on the flat pattern are kept without any distortions, the strain energy is equal to 0. Therefore, no subsequent relaxing of the strain energy needs to be executed. Thus, we name the first situation as unconstrained triangle flattening.

Another situation also happens when trying to flatten the triangle F. At this time, the third node of the triangle F has been previously flattened as shown in Fig. 3c. In implementing an unconstrained flattening on node $\mathbf{v_0}$, a new position $\mathbf{v_0'}$ is first acquired. In most cases, this position will conflict with the already obtained one of $\mathbf{v_0}$. In order to solve this problem, a medium position between them is then taken as the final position $\mathbf{v_0''}$ and we call this situation as constrained flattening. Resulting from this averaging process, the elastic energy lying on the stretched edges then arises and an energy relaxation stage is needed to relieve the strain energy in the stretched edges, which we will specify in the following section.

After all the adjacent triangles are added into the flattened part, a global energy relaxation will be done to minimize the strain energy of the whole mesh. Overlapping issue will also be considered by providing a local correction method, which will be described further in Section 2.3.

2.3 Key issues to consider

In this section, we will handle the key issues like selecting the seed triangle, relaxing the overall strain energy, and adjusting the overlapping to accomplish the whole flattening process. The technical details will be presented as follows.

2.3.1 Selection of the central triangle

As the flattening process starts with an arbitrary mapping of the seed triangle with no distortion, the flattening orders of other triangles are dependent on the selected seed mesh. The central mesh or meshes are usually taken as the seed triangle; thus, the iterations required to choose the adjacent meshes can be dramatically reduced. Here, we adopt a

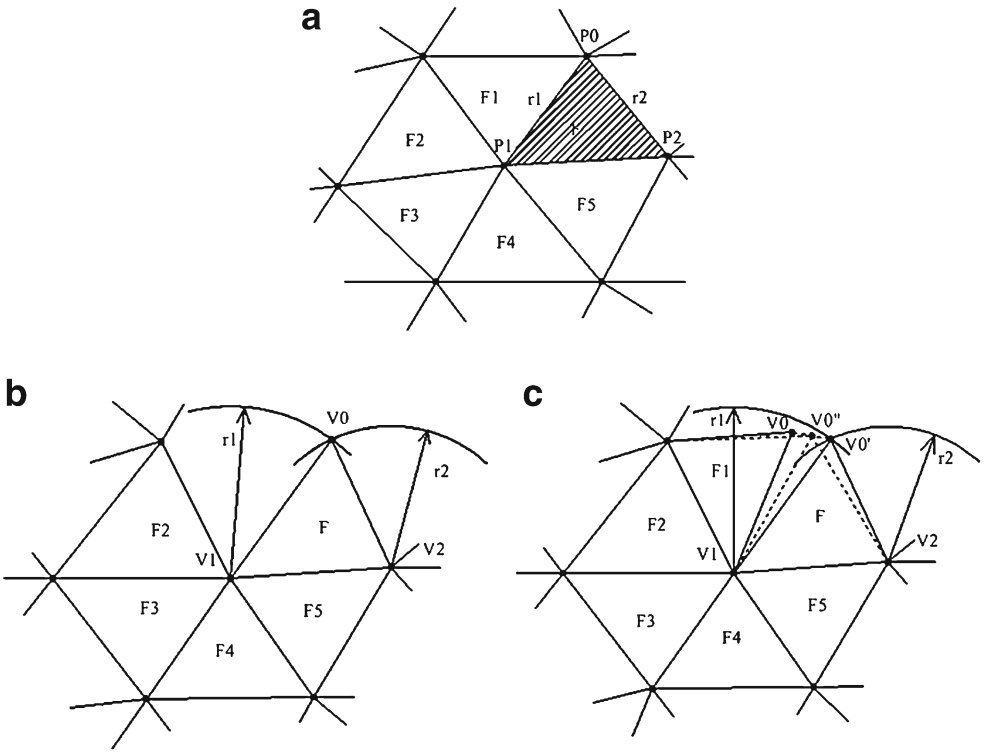


Fig. 3 Two kinds of flattening. **a** 3D original surface. **b** Unconstrained flattening. **c** Constrained flattening

similar method proposed by Li et al. [23] to figure out the central triangle(s). This selecting algorithm is described as follows:

1. Traverse the whole surface to find all the boundary meshes T_i; assign each of them with an index W_i and set W_i=1.
2. Search all the triangles T_j adjacent to those with the maximum index value; set their indices as W_j=W_i+1.
3. Repeat step 2 until all the triangles are labeled.
4. Select the triangles labeled with the maximum index value as the seed mesh or arbitrarily choose one among them if more eligible choices exist.

An example of the above algorithm is shown in Fig. 4. The boundary meshes are first found out and labeled one.

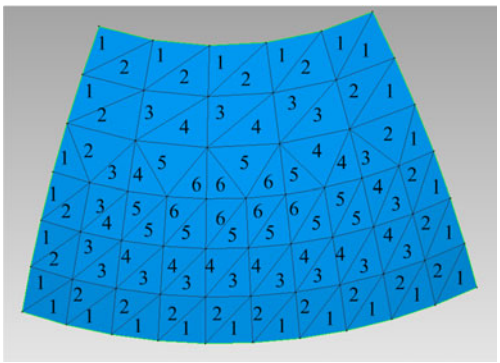


Fig. 4 An example of seed triangle selection

The searching process goes on and the maximum index value increases. Finally, several meshes labeled six have been fixed as the candidate triangles. The seed triangle then can be arbitrarily selected among them and the flattening process starts.

2.3.2 Energy relaxation algorithm

The energy relaxation algorithm plays an essential role in the whole procedure. It determines both the flattening accuracy and efficiency. The relaxation process needs to be taken both in local energy dispersion and overall flattening optimization. After the initial flattened result has been obtained, it has to undergo several relaxation iterations until it meets the accuracy requirement mentioned above. In previous studies, most of the exiting relaxation methods are focusing on how to refine the energy function representation or how to choose a rapid energy-releasing direction. McCartney et al. [17] moved the nodes in four orthogonal directions at an increment to find out the best moving orthogonal direction, which was very time-consuming, and the optional relaxation directions were largely restricted. Wang et al. [20] had to solve the Lagrange equations of motion to figure out a suitable position. Li et al. [23] built up a set of kinetic equations to simulate the node movements. Liang and Bin [26] described the energy-relaxing process as a minimum problem based on the constraints and used penalty function method to solve it. All the above-mentioned optimization methods were very complex to be conducted. Aiming to

handle versatile situations without taking such a complex implementation and without considering material characteristics, we tackle this problem by seeking a convenient and efficient direction to steeply reduce the residue strain energy. It is based on the simplified energy model which is proposed in Section 2.1.2. At each step, the node moves to an optimal position in a desired direction decided by the algorithm defined as follows:

An innovative energy relaxation algorithm based on a variable step size Before a detailed elaboration to the algorithm, the symbols appearing in the following formulas are first defined as follows: the node to be optimized is $\mathbf{v}_i$, its finally optimized position is $v_i^{'}$, the number of optimization iterations is N_{ite}, iteration termination follows Eqs. 10 and 11, $L_{\max}^{i}$ is the maximum edge length distortion among the edges connected to the node $\mathbf{v}_i$ during the ith iteration, the position of the node $\mathbf{v}_i$ during the ith iteration is v_i^i, and the step it takes is $\overrightarrow{\text{step}_\text{i}}$, and then the algorithm begins as follows:

1. Preset the maximum iteration number N_{ite} and the threshold value of area and length accuracy.
2. Calculate the resultant force $\overrightarrow{f}\left(v_i^i\right)$ on the node $\mathbf{v}_i$ and get its unit vector $\overrightarrow{n}\left(v_i^i\right)$ by Eqs. 12 and 13; calculate the maximum edge length distortion $L_{\max}^{i}$ by Eq. 14.
3. Obtain the step size $\overrightarrow{\text{step}_\text{i}}$ of the iteration I by Eq. 15, where μ is a positive constant less than 1 which controls the step size.
4. Update the coordinate v_i^i of the ith iteration by Eq. 16.
5. Calculate Eq. 10 or Eq. 11. If it satisfies, the iterations are terminated and the final optimum position $v_i^{'}$ is then obtained; otherwise, go to step 2. If the iteration steps exceed the maximum value N_{ite}, iterations are also terminated.

$$L_{\max}^{i} = \max\left\{\left|\left|v_i^i v_j\right| - d_j\right|\right\} \tag{14}$$

$$\overrightarrow{\text{step}_\text{i}} = \mu \times L_{\max}^{i} \times \overrightarrow{n}\left(v_i^i\right) \quad (0<\mu<1) \tag{15}$$

$$v_i^i = v_i^{i-1} + \overrightarrow{step_i} \tag{16}$$

The sketch map of Fig. 5 shows the procedure of how a node $\mathbf{v_0}$ gradually achieves its optimum position $v_0^{'}$ under a variable step size iteration. Figure 6 shows a result of flattening a half-sphere triangulated model using this optimization algorithm.

Compared with the previous existing relaxation algorithms, the main advantages of our method can be summarized as follows:

1. Choosing the resultant force direction to relax residue elastic energy fully utilizes the physical characteristics of the energy model. It converges faster than the tentative movement at four orthogonal directions.
2. Compared with the uniform step size iteration, we take a variable one at each step. At the initial stage, large distortions exist in the edges connecting the target node. Therefore, we take a relatively large step size to rapidly ease the residue elastic energy. At the later stage, the optimized position is quite near the optimum one and then we take a small step size. Thus, it can easily lead to a stable result with acceptable speed.
3. In the former optimization methods, the spring constant assignment and the calculation of the node mass have to be considered practically. Based on the simplified energy model, our method only concerns on choosing a reasonable direction, which can avoid the tedious kinematics and dynamics calculations.

2.3.3 Overlapping correction

When the flattening process goes on, distortions are gradually accumulating as more and more triangles have been flattened. Overlapping will then arise when the distortion reaches a certain amount. Therefore, a local correction operation is needed to erase the overlapping.

As shown in Fig. 7, the overlapping can be summarized as two cases:

Case 1 The overlapped triangle is connected with only one flattened triangle. As shown in Fig. 7a, the triangle F3 ($\mathbf{v_1}$,$\mathbf{v_5}$,$\mathbf{v_4}$) is overlapped and it is only adjacent to one flattened triangle F2 ($\mathbf{v_1}$,$\mathbf{v_3}$,$\mathbf{v_4}$).

Case 2 The overlapped triangle is connected with two flattened triangles. As shown in Fig. 7b, overlapping happens in the triangle F3 ($\mathbf{v_1}$,$\mathbf{v_5}$,$\mathbf{v_4}$) which is adjacent to both the flattened triangle F2 ($\mathbf{v_1}$,$\mathbf{v_3}$,$\mathbf{v_4}$) and F4 ($\mathbf{v_6}$,$\mathbf{v_4}$,$\mathbf{v_5}$).

To tackle case 1 (see Fig. 7a), a local correction method is used as follows. The node $\mathbf{v_5}$, which is opposite the common edge E1 ($\mathbf{v_1}$,$\mathbf{v_4}$) shared by both triangle F3 and F2, is first found out in the overlapped triangle F3. A suitable force f1, which is perpendicularly pointing at the common edge E1 ($\mathbf{v_1}$,$\mathbf{v_4}$), is then imposed on the node $\mathbf{v_5}$. The force f1 continues to act on the node $\mathbf{v_5}$ until it moves to the other side of edge E1. Thus, such a local overlapping correction is finalized.

The case 2 (see Fig. 7b) is treated as follows. The common node $\mathbf{v_4}$, which is shared by triangle F3, F2, and F4, is first found out in the overlapped triangle F3. The two nodes $\mathbf{v_6}$ and $\mathbf{v_1}$ opposite to the node $\mathbf{v_4}$ are also picked out. Two suitable forces f1 and f2 are imposed on node $\mathbf{v_6}$ and $\mathbf{v_1}$, respectively. The direction of f1 is pointing from $\mathbf{v_6}$ to $\mathbf{v_1}$

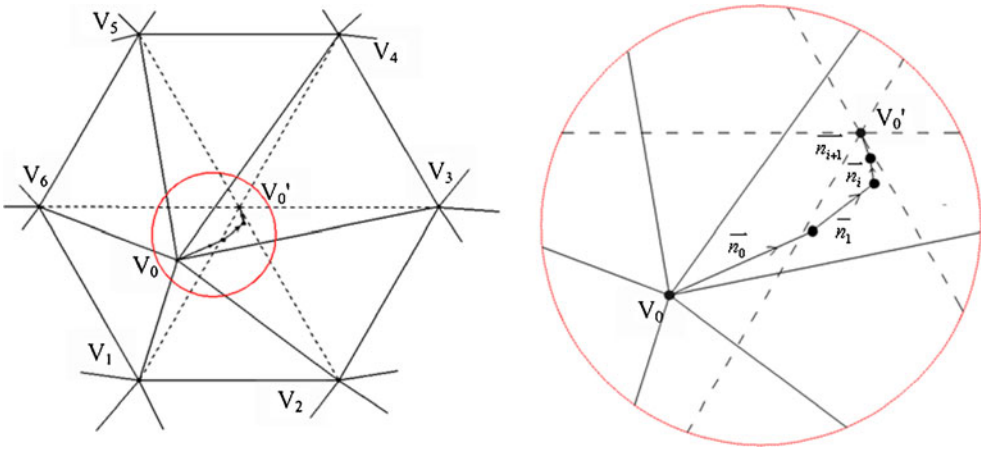


Fig. 5 The sketch map of the energy relaxation procedure

while f2 is the opposite one. The two forces last until the two nodes exchange their place, which means the correction is finalized.

Because a local energy relaxation is always followed after the overlapping correction, it is unnecessary to define the corrected location very precisely. Sometimes we can simply move the target node to the symmetric position in case 1 or exchange the two nodes' positions in case 2 to make a fast correction if possible.

2.4 Hierarchy flattening

As mentioned in Section 2.3, distortions are gradually accumulating as the flattening process continues. The shape of the later flattened triangles will be largely affected by the accumulated distortions caused by previous flattened ones and this situation will become worse when dealing with the complex surface. In order to reduce the bad influence of the accumulated distortions to a minimum, we then propose a hierarchy flattening method by taking a multistep process.

First, we choose a certain part of the target surface as the subsurface, which is usually the central part of the main surface. The subsurface is firstly flattened according to the rule as stated above. The obtained pattern derived from the subsurface serves as a base. The remaining triangles on the same 3D original surface are then developed with reference to the meshes on the developed subsurface. Finally, we can obtain the flat pattern of the whole surface. As it is a process from the subsurface to the main surface, we name it the hierarchy flattening. The classical beetle model is chosen to illustrate the algorithm as shown in Fig. 8. If we directly flatten the whole surface, large distortions are arising near the boundary of the surface caused by the accumulated errors as shown in Fig. 8b. By comparison, a subsurface is first selected and flattened as shown in Fig. 8c, d. The flat pattern of the entire surface is then obtained as shown in Fig. 8e. The comparison of the two obtained flattening results clearly shows the effectiveness of the hierarchy flattening method. The hierarchy flattening presents a new flattening way which is conducted from part to whole. As a general method, it can be adopted by both the existing methods and our proposed one.

3 Numerical experiment and discussion

In this section, several numerical experiments on three typical sheet metal parts are conducted to verify our energy-based method mentioned above. To fully demonstrate the effectiveness of our method, a classical and newly emerging mesh parameterization approach named as-rigid-as-possible (ARAP) method [29] is also presented to make a comparison. The ARAP method is almost an isometric parameterization and showing much more excellent area-preserving capability than other state-of-the-art approaches, which is essential for the initial blank shape design.

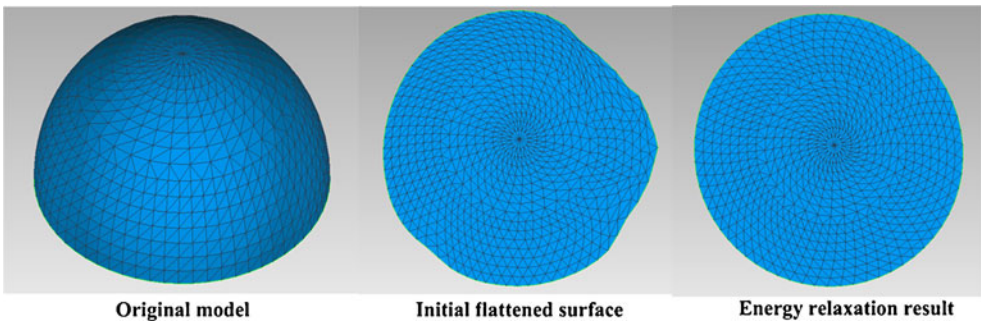


Fig. 6 An example of energy relaxation

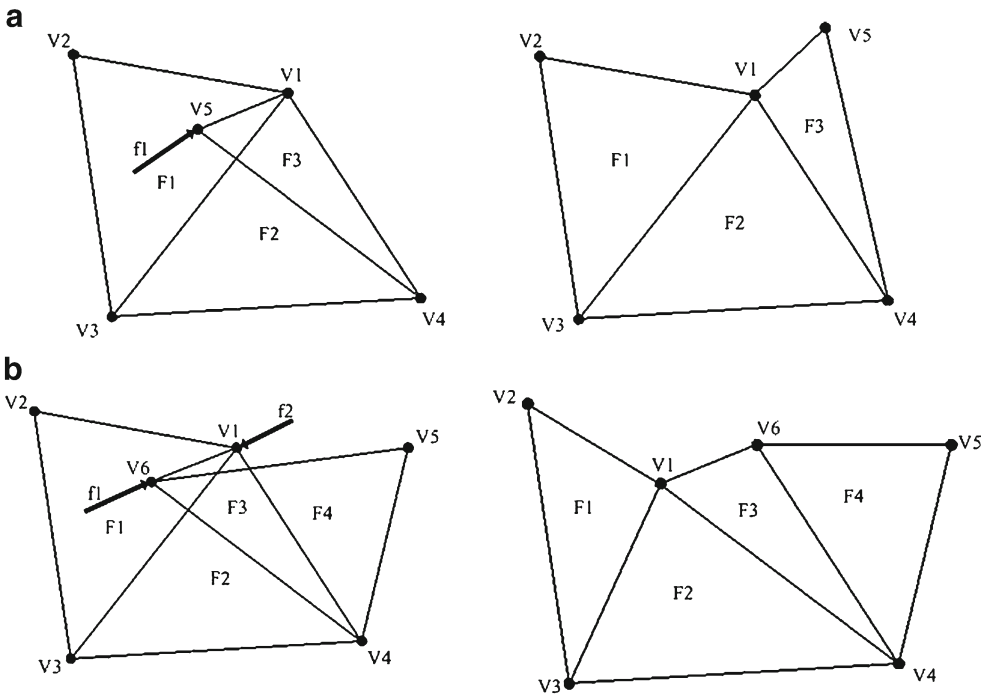


Fig. 7 Overlapping correction. **a** Overlapping of case I. **b** Overlapping of case II

A typical sheet metal part with flanges is first chosen as an example as shown in Fig. 9. A middle surface is abstracted from the solid model and meshed with 1,475 nodes and 2,740 triangular elements. The corresponding flat pattern can be easily obtained by both the energy method and the ARAP approach. The surface area distortions and the computation time can be seen in Table 1 while all the experiments in this section are conducted on a 2.33-GHz Core 2 (2 G RAM) PC. In the table, ε_A is the area error and ε_l is overall length error. As shown in Table 1, both the length error and the area error of both methods are controlled below 1 %, which is satisfactory in 2D pattern generation industry. Compared with the energy-based flattening algorithm, the methods proposed by Prasad and Selvaraj [13, 14] will be very hard to handle this issue because the edge curve connecting the principal flat surface and the flanges is a complex freeform curve.

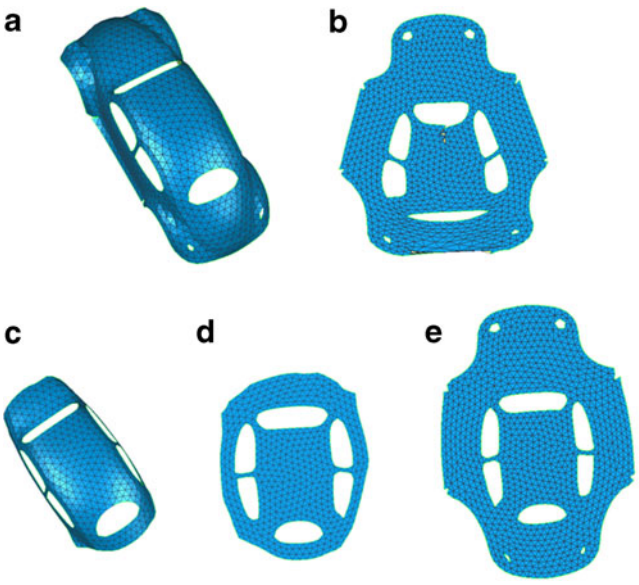


Fig. 8 An example of the hierarchy flattening. **a** Original 3D model. **b** Directly flattened result. **c** Subsurface. **d** Flattened subsurface. **e** Flattened main surface

Another typical sheet metal component with multiple folds is presented in Fig. 10. The abstracted surface is developable and meshed with 1,622 nodes and 2,994 triangular elements. As for such component with multiple levels of folds, Prasad and Selvaraj [13] took a sequential unfold technique to unfold the bends one by one and the whole unfolding process has to take several steps relying on the levers of folds. As for energy-based method, the flat pattern is generated without any shape distortions in a single step as shown in Fig. 10b. However, the result of the ARAP method shown in Fig. 10c is unsatisfactory even if the surface is totally developable. The reason for this is that the ARAP has to predefine at least two vertices onto the plane, which will affect the result if they are improperly chosen. The related surface area distortions and computation time can also be seen in Table 1. As the surface is developed, no shape distortions are happening in the energy-based result while the ARAP method performs worse in this case.

Finally, an oil pan, which is a typical deep drawing workpiece, is presented in Fig. 11. Compared with the above-presented bend formable sheet metal parts, it is very complex and large distortions are happening during its forming process. Fewer researches based on the conventional

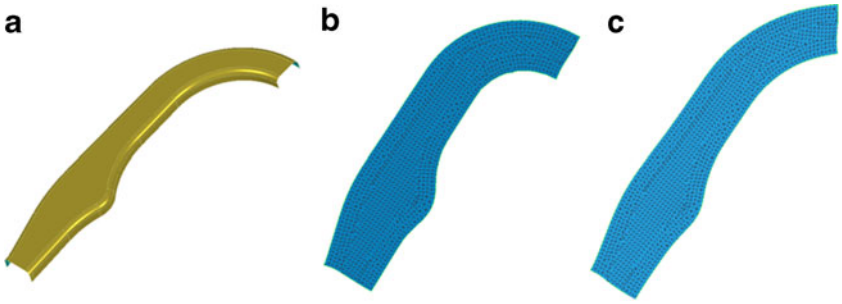


Fig. 9 The flat pattern development of a sheet metal part with flanges. **a** Original 3D model. **b** Flat pattern obtained by energy-based flattening. **c** Flat pattern obtained by ARAP method

graphic solutions have been done on this kind of parts. The target surface is first meshed with 3,239 nodes and 6,317 triangular elements. With the aid of hierarchy flattening, the flatten pattern is easily generated as shown in Fig. 11b while the similar result is obtained by ARAP method shown in Fig. 11c. As the surface is undevelopable, residue shape distortions still exist even after the energy relaxation. Nevertheless, the derived pattern based on the energy-based method is still acceptable as both the length and area errors are within 5 %. However, the boundary of the ARAP result is much smoother than that of the energy-based one because it can better preserve the single mesh shape. From Table 1, we can see that the energy-based method still performs well on length and area preservation while dealing with undevelopable surface. However, it runs much slower than the ARAP method.

As has been stated, a simplified energy model and a new energy relaxation algorithm are proposed in Section 2, which we claim that it will greatly promote the flattening efficiency. In this section, a comparison between our proposed method and the existing methods is also made to verify our statement. In reference [20], Wang et al. tested their traditional energy relaxation algorithm on a PIII 667-MHz PC. From their experimental results, 37 s was needed to flatten a surface containing less than 400 triangles. Meanwhile, the computation time of our method can be seen in Table 1. As the two experiments were conducted on different platforms with different subjects, the quantitative differences of the efficiency between them cannot be specifically calculated. However, it is evident that our efficiency outperforms the traditional one only by a rough calculation. According to our observations, only the algorithm proposed by Li et al. [23] achieved a close efficiency to ours by using triangle strip flattening.

Based on the above-presented experimental results, we believe that the energy method is a practical method to automatically obtain the flat pattern of sheet metal component whether they are bend formable parts or deep drawing workpiece. Its merits and drawbacks can be summarized as follows:

1. It is a heuristic algorithm as the triangles can be flattened automatically after the selection of the seed triangle. Compared with the conventional graphic solutions or the feature-based pattern developing method, it is intelligent and easy to operate without any manual intervention.
2. In lofting or blank generation of sheet metal components, the blank shape is usually determined according to the rule that the blank area equals with its original part. Thus, it is especially suitable for flat pattern development of sheet metal component as it is a length-preserving and area-preserving surface flattening method.
3. Compared with the ARAP parameterization method, it does not need an initial solution, which makes it especially suitable for approximately developable surfaces.
4. The energy relaxation process of the energy method is a sequent position adjustment of nodes, which then makes it extremely slow compared with the ARAP method. However, it is still faster than graphic solutions or the feature-based flattening methods.
5. If the target surface has a very bad developability, it will confront with great difficulties as the energy-based method is obstinate to try to preserve every edge length.

Finally, there is one point we have to stress in particular. As in the energy-based method, the mod of change of length

Table 1 Experimental results of the energy-based and the ARAP method

Figures	No. of nodes	No. of triangles	ε_A (energy-based) (%)	ε_A (ARAP) (%)	ε_l (energy-based) (%)	ε_l (ARAP) (%)	Runtime (s) (energy-based)	Runtime (s) (ARAP)
Figure 9	1,475	2,740	0.29	0.58	0.00	0.27	4	2
Figure 10	1,622	2,994	0.00	3.10	0.00	1.20	3	2
Figure 11	3,239	6,317	4.50	4.20	1.80	2.00	16	5

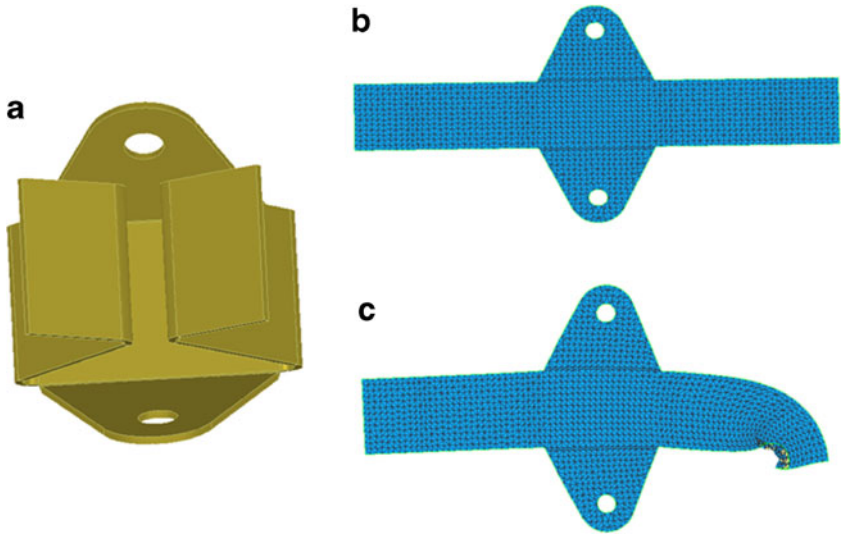


Fig. 10 The flat pattern development of a sheet metal part with multiple folds. **a** Original 3D model. **b** Flat pattern obtained by energy-based flattening. **c** Flat pattern obtained by ARAP method

and mod of change of area are used to evaluate the flattening result. However, it does not mean that both an increase and a decrease in length are treated identically. Actually, in the real sheet metal working, both the material property and the manufacturing process will affect the final obtained result. In another word, the energy-based method only provides an approximate solution. In order to improve the flattening accuracy, two methods can be further employed. As for the bend formable sheet metal parts, bending allowances which consider the bending radius, bending angle, and plate thickness can be taken into account to enhance the result. As for the deep drawing parts, our flattened result can be taken as an initial solution. Based on our proposed initial solution, a more accurate result can be obtained by using the inverse approach which is widely used in the numerical sheet forming simulation [30].

4 Conclusions

In this paper, we make a comprehensive survey on the flat pattern development methods of sheet metal components and the traditional energy-based flattening methods. The conventional graphic solution or the analytical method carried out manually is laborious, time-consuming, and lacks accuracy. The feature-based flattening could help to achieve the objective of automatically pattern generation; however, most of them are only suitable for bend formable sheet metal components composed of flat and simple curved surface. Inspired by its wide applications in cloth and shoemaking industries, we introduce the energy-based flattening for mesh surface into the flat pattern derivation of sheet metal parts. A simplified energy model is then adapted. An innovative energy relaxation process based on variable step size is introduced to promote efficiency. Some key issues like seed triangle selection and overlap correction are also concerned. For those complex surfaces, a hierarchy flattening methodology is proposed to release the flattened distortions. Due to its intrinsic property of length preservation, we made a comparison between the energy method and the classic ARAP surface parameterization method. Qualified flat patterns of three typical sheet metal parts are then generated based on our method. Comparative experimental results show that the energy-based flattening method is very practical in flat pattern generation of sheet metal components as it is intelligent, automatic, and easy to operate. Its merit and defect compared with other approaches are also fully discussed. As quasi-developable surfaces

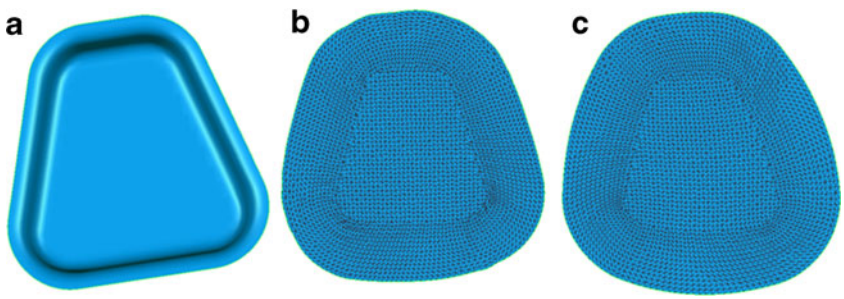


Fig. 11 The flat pattern development of a typical deep drawing workpiece. **a** Original 3D model. **b** Energy-based flattening method. **c** ARAP flattening result

extensively exist in sheet metal components, the energy-based method has the great potential in those areas. In our future work, the process requirements such as bend allowances will be fully considered and introduced into the energy-based method, which will undoubtedly further improve the accuracy of the generated flat pattern.

Acknowledgments This work was supported by the National Natural Science Foundation of China (project no. 50821003) and the National Natural Science Foundation Project for Creative Group of China (no. 51121063).

References

1. Chuang SH, Huang SF (1996) Feature decomposition from solid models for automatic flattening. Comput Aided Des 28(6–7):473–481
2. Ye XG, Fuh JYH, Lee KS (2001) A hybrid method for recognition of undercut features from moulded parts. Comput Aided Des 33(14):1023–1034
3. Gao S, Shah JJ (1998) Automatic recognition of interacting machining features based on minimal condition subgraph. Comput Aided Des 30(9):727–739
4. Tyan LW, Devarajan V (1998) Automatic identification of non-intersecting machining features from 2D CAD input. Comput Aided Des 30(5):357–366
5. Jagirdar R, Jain VK, Batra JL (2001) Characterization and identification of forming features for 3-D sheet metal components. Int J Mach Tools Manuf 41(9):1295–1322
6. Ganesan R, Devarajan V (1998) Intersecting features extraction from 2D orthographic projections. Comput Aided Des 30(11):863–873
7. Meeran S, Taib JM (1999) A generic approach to recognising isolated, nested and interacting features from 2D drawings. Comput Aided Des 31(14):891–910
8. Shunmugam MS, Kannan TR (2002) Automatic flat pattern development of sheet metal components from orthographic projections. Int J Mach Tools Manuf 42(13):1415–1425
9. Joshi S, Chang TC (1988) Graph-based heuristics for recognition of machined features from a 3D solid model. Comput Aided Des 20(2):58–66
10. Choi JC, Kim BM, Kim C (1999) An automated progressive process planning and die design and working system for blanking or piercing and bending of a sheet metal product. Int J Adv Manuf Technol 15(7):485–497
11. Rao YQ, Huang G, Li PG, Shao XY, Yu DY (2007) An integrated manufacturing information system for mass sheet metal cutting. Int J Adv Manuf Technol 33(5–6):436–448
12. Toh KHS, Loh HT, Nee AYC, Lee KS (1995) A feature-based flat pattern development system for sheet metal parts. J Mater Process Technol 48(1–4):89–95
13. Prasad KSRK, Selvaraj P (2004) Practical methods of computer-aided flat pattern development for sheet metal components. Int J Prod Res 42(15):3011–3039
14. Prasad KSRK, Selvaraj P (2005) On the considerations for the design of an automated aircraft sheet metal component production loft generation system. Int J Prod Res 43(14):3045–3067
15. Selvaraj P, Radhakrishnan P, Adithan M (2009) An integrated approach to design for manufacturing and assembly based on reduction of product development time and cost. Int J Adv Manuf Technol 42(1–2):13–29
16. Maillot J, Yahia H, Verroust A (1993) Interactive texture mapping. In: SIGGRAPH 93 Computer Graphics Proceeding, New York, USA, pp. 27–34
17. McCartney J, Hinds BK, Seow BL (1999) The flattening of triangulated surfaces incorporating darts and gussets. Comput Aided Des 31(4):249–260
18. McCartney J, Hinds BK, Seow BL, Gong D (2000) An energy based model for the flattening of woven fabrics. J Mater Process Technol 107(1–3):312–318
19. McCartney J, Hinds BK, Chong KW (2005) Pattern flattening for orthotropic materials. Comput Aided Des 37(6):631–644
20. Wang CCL, Smith SSF, Yuen MMF (2002) Surface flattening based on energy model. Comput Aided Des 34(11):823–833
21. Wang CCL, Wang Y, Tang K, Yuen MMF (2004) Reduce the stretch in surface flattening by finding cutting paths to the surface boundary. Comput Aided Des 36(8):665–677
22. Wang CCL, Tang K, Yuen MMF (2005) Freeform surface flattening based on fitting a woven mesh model. Comput Aided Des 37(8):799–814
23. Li JT, Zhang DL, Lu GD, Peng YY, Wen X, Sakaguti Y (2005) Flattening triangulated surfaces using a mass-spring model. Int J Adv Manuf Technol 25(1–2):108–117
24. Zhong YQ, Xu BG (2006) A physically based method for triangulated surface flattening. Comput Aided Des 38(10):1062–1073
25. Chung WJ, Kim SH, Shin KH (2008) A method for planar development of 3D surfaces in shoe pattern design. J Mech Sci Technol 22(8):1510–1519
26. Liang WW, Bin HZ (2004) Optimal flattening of freeform surfaces based on energy model. Int J Adv Manuf Technol 24(11–12):853–859
27. Floater MS, Hormann K (2005) Surface parameterization: a tutorial and survey. In: Dodgson NA, Floater MS, Sabin MA (eds) Advances in multiresolution for geometric modelling. Springer, Heidelberg, pp 157–186
28. Sheffer A, Praun E, Rose K (2006) Mesh parameterization methods and their applications. Found Trends Comput Graph Vis 2(2):105–171
29. Liu LG, Zhang L, Xu Y, Gotsman C, Gortler SJ (2008) A local/global approach to mesh parameterization. Comput Graph Forum 27(5):1495–1504
30. Batoz JL, Guo YQ, Mercier F (1998) The inverse approach with simple triangular shell elements for large strain predictions of sheet metal forming parts. Eng Comput 15(7):864–892