Haisheng LI[1], Yuhang ZENG[1], Qiang CAI[1], Yuewu LIU[2], Junping DU[3]

College of Computer and Information Engineering, Beijing Technology and Business University, Beijing 100048, China (1),
Institute of Mechanics, Chinese Academy of Science, Beijing 100190, China (2),
School of Computer Science, Beijing University of Posts and Telecommunications, Beijing 100876, China (3)

Constraint Voronoi Grid Generation in 2D Arbitrary Domain by Refinement Algorithm

***Abstract**. In this paper, based on studying the traditional constraint Voronoi diagram generation techniques, an optimized trapezium examining strip refinement algorithm for constraint Voronoi grid generation is presented .First, the initial isosceles trapezoid examining strip sets are settled according to the constraint condition, then by introducing several control factors to subdivide the examining strip to realize the speedy generation of constraint Voronoi grids. Experimental results show the proposed algorithm can get satisfied results even in the complex domain including internal boundary constraints, pencil of lines constraints and irregular areas.*

Streszczenie:** W opracowaniu, w celu wytworzenia siatek Voronoi z ograniczeniami, na podstawie badań tradycyjnej techniki wytwarzania diagramem Voronoi, przedstawiono algorytm rafinacyjny tworzenia trapezowej wstęgi badań. Wstępnie ustalono równomierne trapezowe wstęgi badań zgodnie z warunkami ograniczeń. Następnie, aby przyspieszyć tworzenie siatek, podzielono badane wstęgi przez wprowadzenie kilku współczynników kontroli. Wyniki badań pokazują, że proponowany algorytm daje satysfakcjonujące wyniki w złożonych obszarach, włącznie z ograniczeniami wewnętrznymi granicami i wiązkami linii oraz w przypadku nieregularnych pól.* ***Tworzenie siatek Voronoi ograniczeniami w przestrzeni dwuwymiarowej przy pomocy algorytmu rafinacyjnego

Keywords: Constraint Voronoi Diagram, Delaunay Triangulation, Trapezium Examining Strip, Refinement Algorithm, Control Factor.
Słowa kluczowe: Diagram Voronoi z ograniczeniami, Triangulacja Delaunaj'a, Trapezowa wstęga badań, Algorytm rafinacyjny, Współczynnik kontroli

1. Introduction

Constraint Voronoi diagram has the characteristics of the perpendicular bisector; it can simulate the irregular boundary and solve convergence problem the triangular mesh is used in some numerical simulation calculation. In addition, the mesh refinement and coarsening nature is better. Therefore constraint Voronoi diagram have good application prospects in the field of engineering technology, especially in numerical reservoir simulation, groundwater exploration, robot path planning, fluid mechanics and other fields.

One popular way to efficiently construct Voronoi diagrams consists in exploiting its duality property with the Delaunay triangulation [1]. At present; researches on constraint Voronoi grid generation methods can never meet the practical requirements of the situation [2].Okabe proposed constraint Voronoi diagram generation methods, and these algorithms are simple and easy to implement [3]. But conforming conditions of these algorithms are relatively simple and only contain outer boundary and well point and they do not contain inner boundary. KAPPA also proposed a generation method of constraint Voronoi diagram [4]. The method assumes that inner boundary (such as fault) and outer boundary is straight-line segment, and considers mutual interference of nodes near the boundary, corner points and nodes near the vertical well. The algorithm is universal relatively, as it needs deal with about 10 interference circumstances, the algorithm is relatively complex. Literature [5, 6] proposed control circle algorithms for generating constraint Voronoi diagram. These algorithms are simple, easy to implement and efficient, which can solve the generation problem of constraint Voronoi diagram in ordinary conforming conditions. However its robustness is not very good and it is not convergent in some certain complex conforming conditions. Literature [7, 8] proposed Voronoi refinement algorithms to generate two-dimensional constraint Voronoi diagram. The algorithm adaptive is not good under the complex constraint conditions. In the case of smaller angle between two limit lines, rectangular detection with two boundaries is very close to the limit line, operation of refinement trapezium examining strip will produce a large number of such deformities constraint Voronoi diagram grid cell. In addition, the latter part of the algorithm needs to be done a lot of size and quality control, especially in the constraint conditions near the handle more complicated.

On the basis of analysing the necessary and sufficient conditions on the existence of conforming conditions in the constraint Voronoi diagram ,this paper proposed a refinement algorithm of trapezium examining strip for constraint Voronoi diagram by introducing several control factors. It can ensure the algorithm for convergence under complex constraint Voronoi diagram generated mesh has good quality; meet the needs of the numerical calculation, and could be widely used in various applications.

This paper is organized as follows. In section 2, the basic concepts of constraint Voronoi diagram are given. In section 3, a detailed description of refinement algorithm of trapezium examining strip for constraint Voronoi diagram, including algorithmic ideas, the steps of the algorithm and the algorithm time efficient analysis. The experimental results of the algorithm proposed are given in this paper in section 4. Finally, our work of this paper is summarized in the last section.

2. Basic concepts of constraint Voronoi diagram

2.1 Two-dimensional constraint conditions

Voronoi diagram is a fundamental concept in computational geometry [9].Voronoi diagram is underlying data structure concerning space refinement and every Voronoi vertex is the circum-centre of corresponding Delaunay triangle [2].

Two dimensional constraint Voronoi grid generation refers to the given conditions (also called constraints) PSLG (planar straight line graph) **A**. Finding a point set **SV**, arbitrary line of **A** can be expressed as some edge union of constraint Voronoi grid, and the isolated point of **A** called constraint Voronoi diagram of the polygon vertex.

In the two-dimensional case, qualification may be point set or line segment set (may include linear, curve) or polygon area set.

1) **Constraint point**: The given points must be grid nodes.

2) **Constraint line**: the generated grids must distribute along meander line, and there must not be grids across constraint lines.

3) **Constraint surface**: a surface of grid block must be on given constraint surfaces, and there must not be grids across constraint surfaces.

4) **Constraint domain**: there must not be grids inside inner constraint domain and outside outer constraint domain.

2.2 Qualification constraint conditions

The qualification of two-dimensional in the Voronoi diagram is the collection of any constraint point and constraint line in space. In order to facilitate, qualification set recorded as: **CSS** (Constraint Condition Set), expressed as:

(1) $$\mathbf{CSS} = \{\mathbf{C}_i\}, i = 1, 2, \ldots, \mathrm{n}.$$

Where C_i represents a constraint condition of **CSS**, $\mathbf{C_i}$ is a constraint point, also can be a constraint line. The collection of constraint points in **CSS** is called the constraint point set, denoted as: **CPS**(Constraint Point Set) , expressed as:

(2) $$\mathbf{CPS} = \{\mathbf{CP}_i\}, i = 1, 2, \ldots, \mathrm{m}_1.$$

Call the collection of segments in **CSS** the constraint segment set, denoted as: **CSS** (Constraint Segment Set), expressed as:

(3) $$\mathbf{CSS} = \{\mathbf{CS}_i\}, i = 1, 2, \ldots, \mathrm{m}_2.$$

Qualification set is a union of constraint points and constraint line: $\mathbf{CSS} = \mathbf{CPS} \cup \mathbf{CSS}$.

This paper aims to constraint subdivision of any qualification, when there appear repeat points, composite line and at the intersection of line cause unnecessary trouble to the refinement algorithm. Therefore, it should be pre-processed, and make qualification in no coincidence point, no composite line, and no at the intersection of line. The vertices of line are all in constraint set of point. In this way, qualification set has normalization, and this process is called the standardization process.

3. Trapezium examining strip refinement algorithm

In order to eliminate the growing point approximate of the deformed one edge of the Voronoi polygon Voronoi grid unit. Reflect the qualification spatial distribution of regular isosceles trapezium examining strip. As shown in figure 1 shows, in the case of a small angle between the constraint lines, protection circle radius at both ends of the line is different. Compared with the rectangle examining strip, generation point of the trapezium examining strip with boundary refinement operation is far away from the constraint line. This makes it easy to eliminate some of the deformity of grid units. In addition, due to the trapezoidal geometric characteristics, examining strip with two boundary stretching to the constraint line on both sides of increasing examining strip with the radius of circumcircle. In the process of generating grid, trapezium examining strip can reduce the number of area of growing point of without any qualification, to speed up the convergence rate.

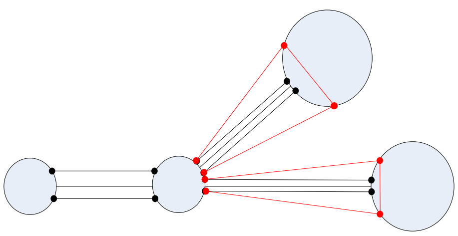

Fig.1. Rectangle examining strip and trapezium examining strip with comparison

3.1 Algorithm ideas

As shown in figure 2. To limit the line **AB**, for example, we respectively construct the protection round in the endpoint with certain radius, and initial growing point $\mathbf{P_1P_2}$ and $\mathbf{P_3P_4}$. An isosceles trapezium examining strip is constituted by $\mathbf{P_1P_2P_4P_3}$. **P'P"** is a median line of a trapezium. The point O is a trapezium circumcircle canter of a circle, radius for **|OP₁|**, circles **O** doesn't contain other growing point inside, **AO** and **OB** must be the constraint Voronoi diagram of two Voronoi edge. If the circle **O** inside contains other growing point, add **P'**, **P"** to the growing point of collection in the **SV**, then the line AB logically can be divided into three sub-segments $\mathbf{AO_1}, \mathbf{O_1O_2}, \mathbf{O_2B}$. $\mathbf{O_1}$ is the circumcircle canter of the isosceles trapezium $\mathbf{P_1P'P''P_3}$, $\mathbf{O_2}$ is the circumcircle canter of the isosceles trapezium $\mathbf{P'P_2P_4P''}$. Repeat this similar empty round detection and when necessary the corresponding growing point operation increasing, until all detected circle is empty (i.e., internal circular does not contain any growing point)

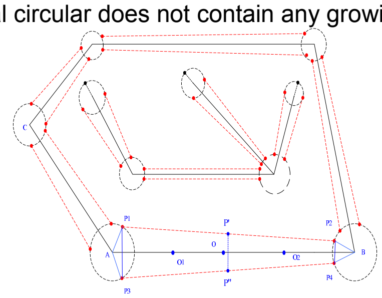

Figure 2 Refinement trapezium examining strip

3.2Description and analysis algorithm

This paper studies how to generate two-dimensional constraint Voronoi diagram. That is, given PLSG (planar straight line graph) **A** is said to be planar areas [10], study how to generate its constraint Voronoi diagram. The PSLG **A**, after 2.1 normalization algorithm, further expressed a union of a point set **PS** (Point Set) and a segment Set **SS** (Segment Set), to be $\mathbf{A} = \mathbf{PS} \cup \mathbf{SS}$. Segment set of **SS** further expressed as external line collection of **OSS** (the outer boundary line set) and the internal segment (within the boundary line and an independent set of line segments) **ISS**, to with $\mathbf{SS} = \mathbf{OSS} \cup \mathbf{ISS}$.

Definition 1: Let **S** and **T** are closed set of point .**S** and **T**, the minimum distance

(4) $$\mathbf{dis}(\mathbf{S}, \mathbf{T}) = \min\{\mathrm{dis}(\mathbf{p}, \mathbf{q}) \mid \mathbf{p} \in \mathbf{S}, \in \mathbf{T}\}$$

Definition 2: Given a PSLG **A**, the local feature size **lfs (p)** of a point **p** with respect to **A** is the minimum distance between **p** and on the elements of **A** that do not contain **p**. Obviously, the **lfs (p)** > 0.As shown in figure 3, the size of the circle in the figure represent different points of triangulation domain local feature size.

Definition3: Given a PSLG A, s∈SS

(5) $$\mathbf{d}_{\min}(\mathbf{s}) = \min\{\mathrm{dis}(\mathbf{s}, \mathbf{t}) \mid \mathbf{t} \in \mathbf{A}\}$$

t and **s** no public point. Obviously, the $\mathbf{d_{min}(s)}$ > 0.

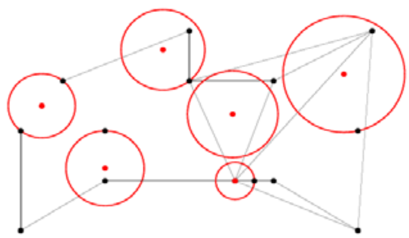

Fig.3. Local feature sizes

Algorithm 1: The main process of the algorithm.
Input: PSLG **A** ($\mathbf{A} = \mathbf{PS} \cup \mathbf{SS}, \mathbf{SS} = \mathbf{OSS} \cup \mathbf{ISS}$) and the uniform distribution the number of ***un***.
Output: Constraint Voronoi diagram which fulfills PLSG **A**.
Algorithm steps:
Step1：Calculations of uniform mesh size **us**, outside the boundary area for **obs** .To with $\mathbf{us} = \mathbf{obs} / \mathbf{un}$.
Step2：With **A** and the **us** as the parameters, call the algorithm 2 for calculating the initial growing point, get an initialization good trapezium examining strip with a collection of **XSS** and an initialization good growth of a collection of **SV**.
Step3: With **XSS** and the **SV** as the parameters, call the algorithm 3 for carrying examining strip with refinement.
Step4：In the sphere of influence within the region of the borehole, radial distributions and add good points to the growth of a collection of **SV**.
Step5：Call the Delaunay triangulation algorithm set by the growing point of the **SV** to construct **D (SV)**, by using **D (SV)** get **CVD (SV)**.
Algorithm 2: Calculating the initial growing point (Sub procedure).
Input: PSLG **A** ($\mathbf{A} = \mathbf{PS} \cup \mathbf{SS}, \mathbf{SS} = \mathbf{OSS} \cup \mathbf{ISS}$) and the uniform mesh size **us**.
Output: Initialize a good examining strip with a collection of **XSS** and an initialized good growth collection of **SV**.
Algorithm steps:
Step1: s∈SS, Calculation dmin(s).
Step2: s∈SS, Let AB be the endpoint. To point A, calculation the lfs (A) and radius rA of point A protection. if s∈OSS, then

$$\mathbf{r_A} = \min\{\boldsymbol{\theta} * \mathbf{lfs}(\mathrm{A}), \boldsymbol{\alpha} * \mathrm{us}\} \quad (6)$$

(0 ＜ θ ≤1/3,0 ＜ α , typically θ =0.3, α =1.0); If s∈OSS, then

$$\mathbf{r_A} = \min\{\boldsymbol{\theta} * \mathbf{lfs}(\mathrm{A}), \boldsymbol{\beta} * \mathrm{us}\} \quad (7)$$

(0 ＜ θ ≤1/3,0 ＜ β , typically θ =0.3, β =0.7), α is external constraint line endpoint protection radius control factors, β is internal constraint line endpoint protection radius control factors. Similarly computing the lfs (B) and rB.
Step3: s∈SS, Let AB be the endpoint. Find the SS except s to A is the endpoint of the line segment, recorded as RS (A). If RS (A) is empty, reverse extension cord of s and the protection of the intersection point A round for p, add p to SV. Otherwise, calculate between the s and RS (A) the minimum angle αAmin. Similarly calculate αBmin.
Step4: Calculate

$$\mathbf{r_1} = 2 * \mathbf{r_A}(\mathbf{s}) = 2 * \min\{\mathbf{r_A} * \sin(\alpha\mathrm{Amin}/3), \mathrm{dmin}(\mathrm{s})/3\} \quad (8)$$

$$\mathbf{r_2} = 2 * \mathbf{r_B}(\mathbf{s}) = 2 * \min\{\mathbf{r_B} * \sin(\alpha\mathrm{Bmin}/3), \mathrm{dmin}(\mathrm{s})/3\} \quad (9)$$

r1, r2 for two bottoms in the shape of a trapezium an isosceles of long, set the two bottoms A and B respectively the protection in P1P2 and P3P4 into round, the four points to join SV.
Algorithm3: refinement trapezium examining strip (Sub procedure).
Input: The trapezium examining strip with a collection of XSS, growth element collection SV and uniform mesh size us.
Output: After refinement the trapezium examining strips with a collection of XSS and growth element collection SV.
Algorithm steps:
Step1: Definition n, Let n = 0.
Step2: xs ∈XSS, xs corresponds to the constraint line is s, xs circumcircle C. If xs is the initial trapezium examining strip of s, two endpoint protection circles is said to be C1 and C2 respectively. Otherwise, let circumcircle be C1 and C2, in the endpoint of xs .Let the distance from the canter of C to the centre of C1 and C2 be d1 and d2.when s∈OSS, if min{ d1, d2} > γ □us (0 < γ , typically γ =1.6) ,to perform a Step4. Otherwise, check C whether it contains any point in the SV. If contains, to perform a Step4.When s∈ISS ,if min{ d1, d2} > δ □us (0 < δ , typically δ =1.0),to perform Step4. Otherwise, check C whether it contains any point in the SV.If contains, to perform Step4.
Step3: If n > 0, return to Step1 start execution; otherwise, end the algorithm.
Step4: With trapezium median line segments divide xs into xs1, xs2 and add to XSS, the intersection of the median line and xs is P'P". Add P'P" to SV, $\mathbf{n} = \mathbf{n} + \mathbf{1}$, return and continue the implementation of Step2.
Algorithm2 when calculating **lfs** (P) and $\mathbf{d_{min}(s)}$, for each qualification, use variable to record the minimum distance. So the time complexity is $\mathbf{O}\left(\mathbf{N_1}^2\right)/\mathbf{2}$ ($\mathbf{N_1}$ is the number of PSLG **A** qualification). **Algorithm3** every time check examining strips with the circumcircle to use walking algorithm to fast positioning [11]. Of the distribution operation of step4 time complexity and set the radius of influence of parts, the number of radiation and other related ,little effect on the efficiency of algorithm time. Set the number of final growing point for **N**.Step5 you can use the empty algorithm to generate the Delaunay triangulation mesh, its time complexity is $\mathbf{O}\left(\mathbf{NlogN}\right)$ [11]. The generated constraint Voronoi diagram, each generate a constraint Voronoi diagram grid cell call first seen walking algorithm for fast positioning of the growing parts, then generated directly constraint Voronoi diagram grid cell. The number of the input PSLG $\mathbf{N_1}$ is far less than quantity of growing points in the generating constraint Voronoi diagram, therefore, the algorithm computation lies in the algorithm Step5, to be $\mathbf{O}\left(\mathbf{NlogN}\right)$.
Based on the above analysis, the time complexity of the entire algorithm is $\mathbf{O}\left(\mathbf{NlogN}\right)$.

4. Experimental results

The experiment was performed using Visual C++ and CGAL. Hardware Environment mainly consists of CPU Core2 Duo T5670 1.8GHz、Memory2G、Video Memory 256M,and Operating System is Windows 7 Ultimate.

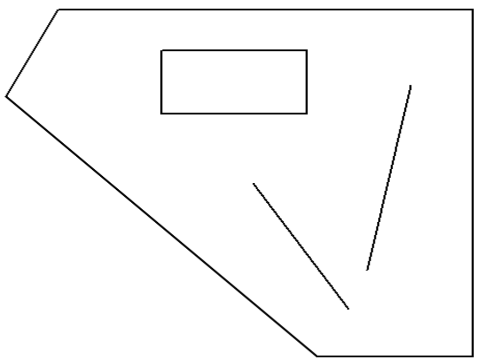

Fig.4. PSLG A

Constraint conditions include constraint point and constraint line (outer boundary and inner fault) and constraint domain.
Take PSLG **A** showed in Figure 4 as input and generate conforming Voronoi diagram showed in Figure 5 using the algorithm proposed in the paper, which takes 1017milliseconds to generate the 2D constraint Voronoi

diagram and produce 730 grid cells. The PSLG **B** shown in Figure 6 takes 3946 milliseconds to generate the conforming Voronoi diagram and produce 1694 grid cells. Experiments indicate that the algorithm proposed in the paper can effectively deal with inner boundary, linear conforming conditions and irregular area to generate constraint Voronoi diagram with good quality.

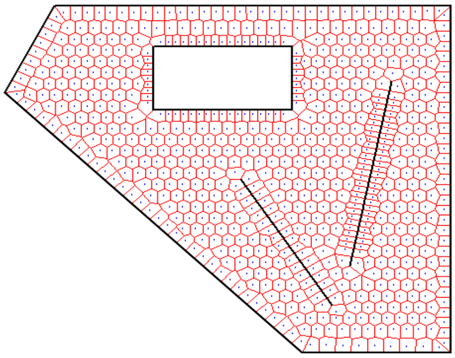

Fig.5. Generated Voronoi Diagram

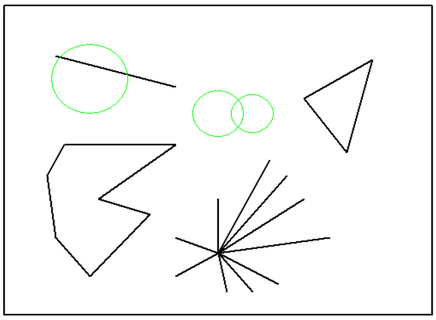

Fig.6. PSLG B

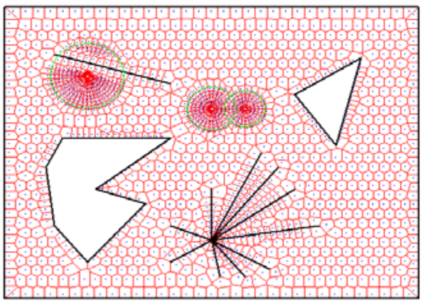

Fig.7. Generated Voronoi Diagram

5. Conclusions and future work

This paper proposed a refinement algorithm of trapezium examining strip for constraint Voronoi diagram. In the case of a small angle between the constraint lines, the trapezium examining strip can eliminate a part of the growing point and approach the edge of the deformed grid cell in the Voronoi .Reduce the number of the growing point constraint segments outside the region, and improve the efficiency of the implementation of the algorithm. At the same time, introduce several control factors to simplify the processing of the near qualification and ensure the algorithm convergence under complex constraints. So that the generation constraint Voronoi diagram grid cell has a better quality to meet the needs of Coalbed Methane numerical simulation calculation .The next work, on the basis of the proposed algorithm , study the three-dimensional the constraint Voronoi diagram generation algorithm, and generate three-dimensional constraint Voronoi diagram, control the quality and size.

Acknowledgements

This work was supported in part by National Basic Research Program of China (973 Program) 2012CB821200 (2012CB821206) and Beijing Natural Science Foundation4112016 and Funding Project for Academic Human Resources Development in Institutions of Higher Learning Under the Jurisdiction of Beijing Municipality PHR201008239.

REFERENCES

[1] Jane Tournois, Pierre Alliez and Olivier Devillers.2D Centroidal Voronoi Tessellations with Constraints. doi:10.4208/nmtma.2010.32s.6.Vol.3, No. 2, pp. 212-222
[2] Qi Li, Haisheng Li, Qiang Cai, Yuewu Liu. Generation of 2D Conforming Voronoi Diagram in Complex Domain[C]. 2011 International Conference on Applied Informatics and Communication. Xi'an, China. Springer Verlag, PP32-39
[3] Okabe A, Boots B, Sugihara K. et al. Spatial tessellations: concepts and applications of Voronoi diagram (2nd edition) [M]. UK: John Wiley & Sons, Ltd. 2000. 242~245.
[4] KAPPA Engineering E06–Voronoi numerical module [EB/OL].Technical reference, http://www.egr.uh.edu/courses/Engi6314/TechnicalReferences/ TR6Advanced Interpretation. PDF.
[5] Qiang Cai, Qin Yang, Xianhai Meng. Research on 2-D constraint PEBI grid generation technique [J] Journal of Engineering Graphics, 2005 26(2):69-72.
[6] Quanyi Yang, Qiang Cai, Jigang Li. Research on 2D constraint PEBI grid generation technique[J].Computer Engineering and Applications, 2004, 22(7):80-83.
[7] Jigang Li.Constraint Voronoi triangulation [D] The doctoral dissertation of Beijing University of Aeronautics and Astronautics.
[8] Verma S. A flexible gridding scheme for reservoir simulation [A]. Presented at SPE Intl, Student Paper Contest[C]. 1995, 657~672.
[9] Toshihiro Tonuma, Hirishi Imai, Sonoko Moriyama. Revisiting Hyperbolic Voronoi Diagrams from Theoretical, Applied and Generalized Viewpoints. International Symposium on Voronoi Diagrams in Science and Engineering 2010.
[10] Jiaye Wang, Wenping Wang, Changhe Tu. Computational geometry and applications [M]. Beijing: Science Press, 2011
[11] A Delaunay Refinement Algorithm for Quality 2-Dimensional Mesh Generation [J]. Journal of Algorithms, 1995, 18 (3): 548 – 558.

***Authors**: Professor Haisheng Li, College of Computer and Information Engineering, Beijing Technology and Business University, No. 33, Fucheng Road, Haidian District, Beijing, P.R. China, 100048, E-mail:lihsh@th.btbu.edu.cn; Yuhang Zeng, E-mail: yuhangyuan1986@163.com; Qiang Cai, E-mail:caiq@th.btbu.edu.cn; Yuewu Liu, E-mail: lywu@imech.ac.cn; Junping Du, E-mail: junpingdu@126.com.*