A local-adjustment based two-dimensional Delaunay triangular mesh generation method on a bounded domain with moving boundary

Tiancheng Gao
School of Mathematics
andSystems Science
Beihang University
Beijing, China 100091
Email: gtczz@sina.com

Homer Simpson
Twentieth Century Fox
Springfield, USA
Email: homer@thesimpsons.com

James Kirk
and Montgomery Scott
Starfleet Academy
San Francisco, California 96678–2391
Telephone: (800) 555–1212
Fax: (888) 555–1212

***Abstract*—In this talk, we present a two-dimensional triangular Delaunay mesh generation method based on local mesh adjustment on a bounded domain with moving boundary. By employing local mesh adjustment rather global re-generation, the developed method obtains good efficiency, while the Delaunay property of the generated mesh guarantees that the mesh has good quality. Furthermore, high dimensional embedding technology is combined with the proposed mesh generation method to generate the anisotropic mesh for a bounded domain with moving boundary. Some typical numerical examples demonstrate the effectiveness, efficiency and robust of the proposed method.**

I. Introduction

Mesh generation is generally the first procedure in scientific computation, as well as the most costly step.According to the literature, the time needed for grid generation usually accounts for about 60% of the total time of a computing task[1]. Especially for unsteady problem, like geometric deformation or relative motion of multiple bodies, the mesh necessitate update at each time step of the solving procedure, which sharply increase the cost in generating mesh. Therefore, it is in demand to develop efficient and effective methods for dynamic mesh generation along with boundary change, with the mesh quality preserved.

By moving the boundary of the bodies, it is able to describe the deformation or motion of that. So the dynamic mesh can be thought as a method for describing and moving the boundary. Generally speaking, there are two categories of method to achieve this goal: topology-preserved and topology-changed.

Topology-preserved method, as the name shown, keeps the topology property after the change, which means that the points can only change its position but can not emerge or disappear. Meanwhile, to avoid the inverted grids, the motion of a single point must be never too big to cross an edge. Although having these disadvantages, topology-preserved method guarantees the conservation in flow field computation, which plays an important role in the continuity of the final solution. Specifically, spring stretch method and background graph mapping method are belong to this group.

There are surely cases that the mesh can not preserve its topological property. For instant a body splitting into two parts, new edges and new points must be generated between the two parts. Hence the topology-changed methods are developed to deal with these essential topology-changed cases as well as the situation that mesh quality would become terrible if keeping the topology, like the relative motion.

In this paper, we provide a simple but effective method to generate dynamic Delaunay based mesh. Because of the demand of Delaunay property, the topology property can not be preserved. The advantage of this method is the flexibility for large scale motion as well as the robustness provided by Delaunay generation algorithm. And to improve the efficiency, locally change is called instead of global regeneration. More, the high dimensional embedding method, which supports generating anisotropic mesh as well as adaptively refining the mesh, is combined with for better mesh quality.

II. Delaunay triangulation and high dimensional embedding

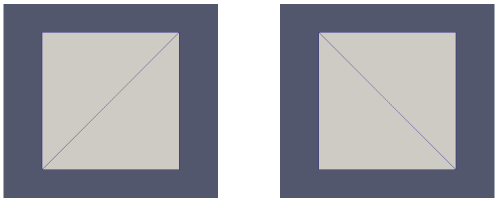
In this section, a brief introduction of Delaunay triangulation along with high dimensional embedding method is involved in order to develop the foundation for further discussion. Specifically, we mainly focus on the Lawson flip algorithm and the refinement algorithm, which are the fundamental algorithm in our dynamic mesh generation strategy.

A. Delaunay triangulation

To begin with, we put the definition of triangulation in 2-D here[3]:

Definition 1: A triangulation of a planar point set S is a subdivision of the plane determined by a maximal set of noncrossing edges whose vertex set is S.

Generally, the triangulations of a certain point set are various. However, it is able to transform from one triangulation to another via a sequence of edge flips[4], which means removing the common edge of to adjoint triangles and adding the other diagonal.For instance the two figures shown in II-A, the transform from 1(a) to 1(b) is an edge flip.



(a) (b)

Fig. 1. sample of an edge flip

Among all the triangulations of a certain point set, there are many circumstances in which certain triangulations are valued more highly than others. The Delaunay triangulation is an extremely important one. It has several equivalent definition from different points of sight. An accessible one is as follow:

Definition 2: For all triangles in a triangulation, if the circumcircle of that triangle has no points other than the three vertices inside, the triangulation is a Delaunay triangulation.

This definition implies that the Delaunay triangulation provides the more regular triangles than other triangulations, which exactly the mesh necessitates for its quality. Therefore, Delaunay based mesh generation methods are various, such as divide and conquer algorithm, incremental algorithm, get the dual graph of voronoi, and etc. In our strategy, the incremental algorithm based on lawson flip[4](the algorithm to transform a random triangulation into Delaunay via a selected sequence of edge flip) is used due to the flexibility as well as the robustness provided, even though the computational complexity is not the best.

Here the 1-3 flip and 2-4 flip refer to the point inserting strategy. If the point is in an triangle, then play a 1-3 flip transforming the triangle into 3 triangles. If the point locates on an edge of 2 triangles, then call a 2-4 flip transforming the 2 triangles into 4 triangles. Show in II-A.

B. Constrained Delaunay triangulation and Delaunay refinement

Delaunay triangulation provides an effective method generating quality mesh. Further more, the Delaunay triangulation of a certain point set is unique as long as the points are in general position,which means that no group of four points of are cocircular and no two points are in coincidence. However, in specific case, there may be some of the edges are necessary to form the boundary of the body while they are not in the Delaunay triangulation.Therefore, we have to constrain these edge to the final triangulation, which is called constrained Delaunay triangulation.

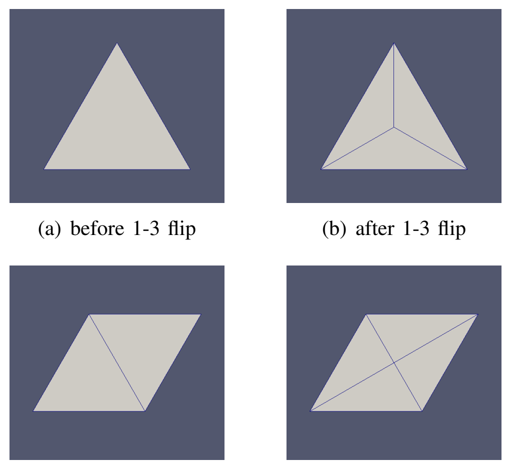
Since the constrained Delaunay triangulation is global Delaunay with locally non-Delaunay, it is an effective way to generate a constrained Delaunay triangulation from the Delaunay triangulation via a series of flip to recover the constrained edge[5].

As for Delaunay refinement,it is a method to enhance the mesh quality via adding points outer from the original point set. Since the original point set may be non-uniform on the plane, even Delaunay triangulation may provide mesh with quite poor quality.For the time being, this method can be called to improve the quality via inserting extra points spliting big cell and causing flips to avoid small angle.Specificly, this procedure checks every triangles and insert points at the circlecenter of the too big ones,either, insert points at the constrained edges of the ones with too small angle[7].By iterating the above procedure, the original Delaunay triangulation is refined and more uniformed.

Algorithm 1: Delaunay triangulation generation algorithm

Input:
A point set S;
Output:
A Delaunay triangulation T;

Randomly select 3 points to generate the first triangle.Remove the 3 points from S;
while *S is not empty* **do**
  Take one point,denoting p, and locate the point on current triangulation;
  if *p is in the convex hull of current triangulation* **then**
    Call a 1-3 flip or 2-4 flip to insert the new point into the triangulation;
  else
    Generate a new triangle with the proper edge on the convex hull and the new point;
  end
  Call the lawson flip to transform the new triangulation to Delaunay;
end



(a) before 1-3 flip (b) after 1-3 flip

(c) before 2-4 flip (d) after 2-4 flip

Fig. 2. Insert strategy

C. High dimensional embedding

In [2], Dassi, Si and Perotto provide a novel method for generating anisotropic mesh via high dimensional embedding.The main idea of this method is lift all the points into a high dimensional space,call embedding space, via an embedding function defined before. Then generate a quasi-Delaunay triangulation in this embedding space via a sequence of splitting , contracting and flipping of edges based on the embedding distance.Finally, project the quasi-Delaunay triangulation into the original plane.It is emphasized that the anisotropic property is inheriting from the embedding function.In other word, it is able to change the embedding function to view this method as an adaptive refinement algorithm.

III. Dynamic mesh generation

Our dynamic mesh generation strategy is based on the constrained Delaunay triangulation introduced in the last section.A series of consecutive constrained edge to represent the boundary of the body.The main quest is to move these edge to simulate the motion or deformation of the body.

A. Overview of the algorithm

Frankly speaking, the main idea of our grid-moving strategy is 'digging a hole' around the moving bodies to enable the motion of boundary without any intersection with other edges or points.After the movement of the boundary, lawson flip and delaunay refinement are called locally to refine the Delaunay property as well as the grid quality.

This algorithm follows the commonsense of clear the barrier of moving objects. So that the objects can be moved directly without any collision. Since the algorithm employs removing and inserting of points, it is obvious an topology-changed dynamic mesh generation algorithm, which enables deformation,relative motion,and large displacement.Never the less, this generation strategy greatly decreases the time cost because no iteration is called in each step.

Algorithm 2: Dynamic mesh generation algorithm

Input:
A point set S
A series of vectors describing the motion of boundary points of each step $\{v_s^t\}$
Max step $Maxstep$
Output:
A series of meshes $\{M^t\}$

Generate the triangulation of the first step;
$step = 0$;
Output the triangulation as M^{step};
while $step < Maxstep$ **do**
    Recover the segments from the refined triangulation by removing the free segment points;
    Remove the collision points and edges;
    Move the boundary edges via moving the constrained segment points $S_i^{step+1} = S_i^{step} + v_{S_i}^{step}$;
    Call locally lawson flip and locally Delaunay refinement;
    Output the triangulation as M^{step};
end

B. Algorithm detail discussion

In this subsection, the details of the above algorithm is discussed.

Since the discription of boundary motion is restricted on the motion vectors at the constrained segment points, the free segment points must be removed to ensure the uniform of boundary edges and reduce the amount of triangles, which induces to the decrease of calculation.

To remove the collision points, the collision points must be found out from the list of points first. According to the motion of each boundary edge, the points in the region it sweeping over are all collide to the edge. Therefore, it is able to remove the combine of collision points of all boundary edges. However, this strategy is computational inefficient since the search of collision points need to loop the full points list for each boundary edge.Thus, it is able to loose the criterion of collision points removing more points while decrease the search computation. In our case, a rectangle box,parelleling to coordinate, is employed covering the moving object and the place where it should move to. Moreover, to avoid too bad triangle after moving,the box is expand a little on each direction.Then it is able to loop only once within the points list to find out all the collision points according to this criterion. However,merely removing the collision points is not abundant beacuse collision edges may remain,shown in fig **??**.So to deal with the collision edges, it is available to flip all the collision edge from closer to further, to clear a way for moving the constrained segment points directly.

After all the constrained segment points moved, the lawson flip and Delaunay refinement algorithm is called locally,to reduce the computational cost.Detailly, only the triangles adjacent to the constrained segment points are pushed into the flip queue as well as the refinement array.

C. Dynamic high dimensional embedding

To employ the high dimensional embedding algorithm into the dynamic mesh generation strategy, it is a trivial idea to modify the embedding function step by step, adjusting to the moving objects. Following this idea, the dynamic mesh generation strategy can be modified as locally lawson flip and Delaunay refinement replaced with high dimensional embedding function with dynamic embedding funciton.

This algorithem enable the generation of adaptive refined mesh along with dynamic generation, for instance, anisotropic grids around deformation boundary to simulate the boundary layer.

IV. Results

A. Deformation cases

Subsection text here.

Algorithm 3: Dynamic mesh generation algorithm

Input:
A point set S
A series of vectors describing the motion of boundary points of each step $\{v_s^t\}$
Max step $Maxstep$
Dynamic embedding funcition
$F(x, y, t), \quad t = 1, 2, \cdots, Maxstep$
Output:
A series of meshes $\{M^t\}$

Generate the triangulation of the first step;
$step = 0$;
Output the triangulation as M^{step};
while $step < Maxstep$ **do**
    Recover the segments from the refined triangulation by removing the free segment points;
    Remove the collision points and edges;
    Move the boundary edges via moving the constrained segment points $S_i^{step+1} = S_i^{step} + v_{S_i}^{step}$;
    Dynamic high dimensional embedding;
    Output the triangulation as M^{step};
end

B. Relative motion cases

1) : Subsubsection text here.

V. Conclusion

The conclusion goes here.

Acknowledgment

The authors would like to thank...

References

[1] Zhe Jiang. Research of the Generation of Dynamic Unstructured Meshes[D]. Nanjing University of Science & Technology, 2006.
[2] Dassi F, Si H, Perotto S, et al. Anisotropic Finite Element Mesh Adaptation via Higher Dimensional Embedding[J]. Procedia Engineering, 2015, 124:265-277.
[3] Devadoss S L, O'Rourke J. Discrete and computational geometry[M]. Springer, 2001.
[4] Charles Lawson. Transforming triangulations. Discrete Mathematics, Volume 3, pages 365C372, 1972.
[5] Chew L P. Constrained Delaunay triangulations[J]. Algorithmica, 1989, 4(1-4):97-108.
[6] Shewchuk, Jonathan R. (2008). "General-Dimensional Constrained Delaunay and Constrained Regular Triangulations, I: Combinatorial Properties". 39 (1-3): 580C637.
[7] Ruppert, Jim (1995). "A Delaunay refinement algorithm for quality 2-dimensional mesh generation". Journal of Algorithms. 18 (3): 548C585. doi:10.1006/jagm.1995.1021