An Evaluation of Tetrahedral Mesh Generation for Non-Rigid Registration of Brain MRI

Panagiotis A. Foteinos, Yixun Liu, Andrey N. Chernikov, and Nikos P. Chrisochoides

Abstract In this paper, we assess the impact of mesh generation on Non-Rigid Registration of brain MR images. The solution accuracy and the speed of finite element solvers depend on how well the underlying mesh approximates the surface of the biological object (fidelity) and how well the elements of this mesh are shaped (quality). Fidelity and quality, however, are two contradicting requirements, as increased fidelity usually implies poor quality and vice versa.

In this paper, we evaluate three public mesh generators and examine how this quality-fidelity trade-off affects the accuracy and the speed of non-rigid registration solvers for brain images.

1 Introduction

In Computer Aided Surgery (CAS) and specifically in image guided neurosurgery, Magnetic Resonance Images (MRI) obtained before the procedure (pre-operative) provide extensive information which can help surgeons to plan a resection path.

Panagiotis A. Foteinos
Computer Science Department, College of William and Mary, VA, 23187, USA,
Computer Science Department, Old Dominion University, Norfolk, VA, 23529, USA,
e-mail: pafotein@gmail.com

Yixun Liu
Computer Science Department, College of William and Mary, VA, 23187, USA,
Computer Science Department, Old Dominion University, Norfolk, VA, 23529, USA,
e-mail: yxliuwm@gmail.com

Andrey N. Chernikov
Computer Science Department, Old Dominion University, Norfolk, VA, 23529, USA,
e-mail: andrey.n.chernikov@gmail.com

Nikos P. Chrisochoides
Computer Science Department, Old Dominion University, Norfolk, VA, 23529, USA,
e-mail: npchris@gmail.com

Careful planning is important to achieve the maximal removal of malignant tissue from a patient's brain, while incurring the minimal damage to healthy structures and regions of the brain. However, current practices of neurosurgical resection involve the opening of the scull and the dura. This results in a deformation of the brain (known as the brain shift problem) which creates discrepancies between the pre-operative imaging data and the reality during the operation. A correction is possible using non-rigid registration (NRR) of intra-operative MRI with pre-operative data.

In this paper, we target Finite Element (FE) based approaches for the non-rigid registration [6]. These methods use real-time landmark tracking across the entire image volume which makes the non-rigid registration more accurate but computationally expensive, as compared to similar methods that use surface tracking [8]. The non-rigid registration problem should be solved fast enough, so that it can be usable in clinical studies [2, 3].

Real-time Image-to-Mesh (I2M) conversion is a critical component of FE-based non-rigid registration of brain images. Moreover, its solution in N dimensions (with $N \geq 4$) is important for handling geometric uncertainties caused by respiratory motion which complicates planning and treatment.

A mesh is characterized by its *fidelity* and *quality*. Fidelity measures how well the mesh boundary resembles the surface of the biological object. Quality assesses the shape of mesh elements; the higher the minimum dihedral angle of the mesh elements is, the higher the quality.

It is well known that the quality of the mesh affects both the accuracy and the speed of the solver [14], because the angles of the elements influence the condition number of the stiffness matrix. In the literature, a good deal of effort has been put towards high-quality mesh generation [5, 9, 10, 16].

It is not clear, however, what the impact of fidelity on the accuracy and speed of the solver is. The reason is because there is a complicated trade-off between quality and fidelity. The need for a better surface approximation always implies a deterioration of mesh quality, simply because well-shaped elements cannot fill the space formed by sharp surface creases or by surface parts of high curvature. Also, higher fidelity usually results in an increase of the number of mesh elements which in turn affects both the mesher's and the solver's speed.

In this paper, we evaluated the impact of three public mesh generators [9, 11, 15] on the accuracy and speed of NRR. The meshers were chosen carefully to cover a wide range of mesh generation approaches. The Delaunay mesh algorithm in [9] offers simultaneous meshing of the surface and the volume of the object. The algorithm in [15] is Delaunay but requires the surface of the object as input. Finally, the algorithm in [11] is an optimization-based technique which compresses an initial body-centered cubic lattice (BCC) to the surface. (See Section 3 for more details.) For each mesher, we conducted an extensive series of experiments controlling the fidelity of the output mesh used for the subsequent NRR [6].

We concluded that meshes with very bad fidelity do not affect the accuracy drastically. On the contrary, meshes with very good fidelity hurt the speed of the mesher due to the poor quality they exhibit. We also observed that the speed of the solver is very sensitive to mesh quality rather than to fidelity. For these reasons, we think that
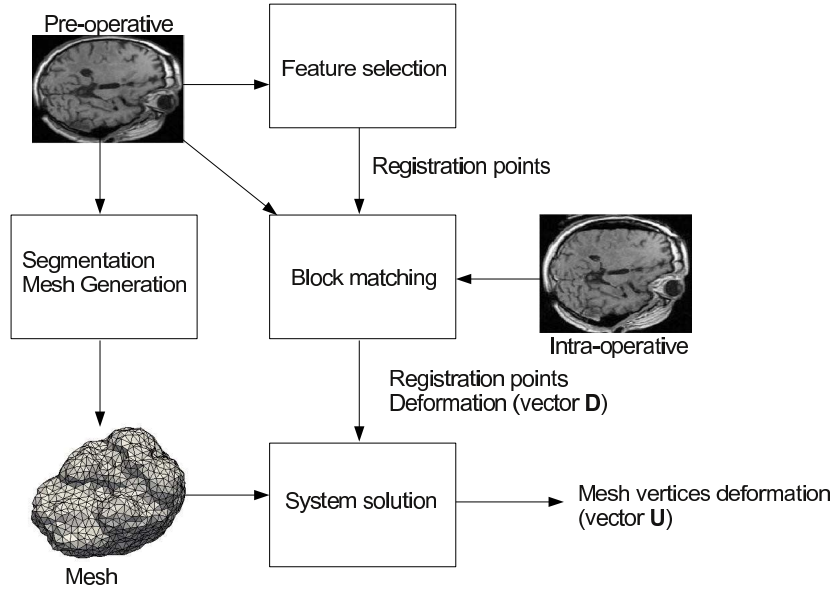
Fig. 1 The non-rigid registration procedure.

mesh generation should first try to produce high quality meshes, possibly sacrificing fidelity.

2 Registration

As our target application, we used the non-rigid registration method described by Clatz *et al.* [6] which is shown to be robust enough to be usable to clinical studies. Below, we outline the main aspects of this NRR method.

The method consists of three steps, namely, *feature points selection*, *block matching*, and *system solution*. See Figure 1 for an illustration. During feature points selection, a sparse set of points is chosen from the pre-operative image. These points are called *registration* points. Then, the correspondence of these points into the intra-operative image is found via a block matching scheme. Specifically, for a given registration point r, a small window around it in the intra-operative image is searched; the corresponding point r' reported is the one that maximizes the correlation coefficient between r' and r.

Having computed the deformation vector **D** on the registration points (as a result of the block matching step), the deformation vector on the mesh vertices **U** (the unknowns) is calculated so that the following energy is minimized:

$$W = \underbrace{(\mathbf{HU} - \mathbf{D})^\top (\mathbf{HU} - \mathbf{D})}_{\text{Error energy}} + \underbrace{\mathbf{U}^\top \mathbf{KU}}_{\text{Mechanical energy}} \tag{1}$$

In the above equation, $\mathbf{K}$ is the $|U| \times |U|$ mechanical stiffness matrix. $\mathbf{H}$ is the linear interpolating matrix of size $|D| \times |U|$; this matrix contains the measurements of the linear shape functions on every registration point. The contributing shape functions for each registration point r_i are those defined over the mesh nodes whose forming mesh element includes r_i.

The block matching deformation d_i of a registration point r_i affects the deformation of a mesh node v_j, only if v_j is incident upon a mesh element e that contains r_j. In fact, if the minimization of the error energy (also known as matching energy) in Equation (1) was perfect (i.e., if it vanished), then the linear interpolation (of the solution of the mesh nodes of e) on r_i would give the value d_i. As Clatz shows in [6] (and as we can see from Equation (1)), this method tries to minimize this exact error energy E:

$$E = \sqrt{(\mathbf{HU} - \mathbf{D})^{\top} (\mathbf{HU} - \mathbf{D})} = ||\mathbf{HU} - \mathbf{D}|| \qquad (2)$$

which is the interpolation error on the registration points $r_1, r_2, \ldots, r_{|D|}$.

The mechanical energy in Equation (1) is used to model the deformation of the brain as a physical body based on FEM. This, in turn, is used to discover and discard the outlier registration points, i.e., points whose deformation estimation from block matching contradicts the physical properties of the brain. For information about the construction of the mechanical stiffness matrix $\mathbf{K}$, see Delingette and Ayache [7].

The deformation vector $\mathbf{U}$, over which energy W is minimized, is computed through the following iterative equations:

$$\mathbf{F}_0 = 0 \,,$$
$$\left(\mathbf{K} + \mathbf{H}^{\top}\mathbf{H}\right)\mathbf{U}_i = \mathbf{H}^{\top}\mathbf{D} + \mathbf{F}_{i-1}, i = 1, 2, \ldots \,,$$
$$\mathbf{F}_i = \mathbf{K}\mathbf{U}_i, i = 1, 2, \ldots$$

In [6], it is proved that the system above converges. Also, observe that $\mathbf{K} + \mathbf{H}^{\top}\mathbf{H}$ is the matrix responsible for the robustness of NRR; its condition number affects both the accuracy and the speed of the solution.

3 Mesh Generation

In this paper, we tested the influence of three meshers on NRR, namely, *High Quality Delaunay* mesher (HQD) [9], *Tetgen* [15], and *Point Based Matching* mesher (PBM) [11]. Below, we briefly describe each of them.

HQD meshes both the surface and the volume of the object at the same time without an initial dense sampling of the object surface, as is the case in other Delaunay volume techniques [12, 13]. As a result, the number of elements of the output mesh is small.

Tetgen is a Delaunay mesh generator as well. However, it assumes that the surface of the object is already meshed and represented as a polyhedron. This polyhe-

dron is also known as a *Piecewise Linear Complex* (PLC). Tetgen requires a PLC of the object surface as its input. We used the algorithm in [4] for the PLC generation, implemented in the *Computational Geometry Algorithms Library* (CGAL) [1].

PBM is an optimization-based approach. It starts with a triangulation of a regular grid, i.e., a body-centered cubic lattice (BCC), and then it compresses the outer nodes closer to the object surface as a result of energy minimization. In fact, the smaller the energy achieved, the better the fidelity of the output mesh. This method is able to recover the surface of multi-tissue objects. In this paper, only the single-tissue version of PBM is considered.

4 Evaluation

4.1 Methodology

As mentioned in Section 2, registration computes the deformation on the mesh nodes, so that the error energy $E = ||\mathbf{HU} - \mathbf{D}||$ is minimized. Mesh generation affects how accurately the error energy is minimized. Therefore, we assess the accuracy of registration by keeping track of this error E. For every run, we let the system iterate for 10 times.

Observe, however, that the outcome of the registration depends on the accuracy of the block matching step (vector $\mathbf{D}$). Also, notice that the mesh does not affect the result of block matching (see Figure 1). Since we are interested in evaluating the impact of mesh generation on registration, we wanted to make registration independent of block matching. For this reason, we synthetically deformed the pre-operative image according to the bio-mechanical properties of the brain. More specifically, we initially ran the registration procedure to register the pre-operative with the intra-operative image as shown in Figure 1, but that time we did not focus on the behavior of the mesh. We just wanted the solution on the mesh nodes. Then, by (linearly) interpolating the solution of the mesh nodes on any point of the image, we obtained a synthetically deformed (intra-operative) image. After this initial registration, all the other registrations (aiming at evaluating mesh generation) are performed between the pre-operative and the synthetically deformed image; that is, the real intra-operative image is replaced by the deformed one. In this way, we achieve two things:

- we know the "true" deformation on any point, and therefore we know the "true" block matching result on any set of registration points, and
- we do not simulate an arbitrary deformation, but rather a realistic one, because the deformed image was obtained taking into account the elasticity properties of the brain through the stiffness matrix $\mathbf{K}$ of Equation 1.

For the initial registration procedure employed to synthetically deform the pre-operative image, we set the parameters to the same values as described in [6] with parameter λ assigned to 1.0. In this way, we obtained a set of 4,000 registration points.

Since we want to measure the influence of mesh generation, only the mesh changes in every experiment. That is, for all the various meshes, the pre-operative image and the set of registration points (together with their deformation **D** of course) remain fixed. Note that for these subsequent registration procedures, we do not reject registration points as potential outliers, simply because the synthetic deformation implies that there are no outliers (this is not the case for the initial registration). For the same reason, we do not weight vector **D** according to the confidence of the registration points deformation. For all the experiments, parameter λ was set to 1.0.

4.2 Measuring and Varying Fidelity

As mentioned above, we wish to have control over the fidelity of the output mesh produced by the different meshers. In this paper, we use the *two-sided Hausdorff distance H* to measure fidelity.

In our case, metric H is defined upon two finite sets A, B as follows:

$$H(A,B) = \max\{h(A,B), h(B,A)\}, \text{ where}$$
$$h(A,B) = \max_{a \in A} \min_{b \in B} ||a - b||$$

The lower the value of $H(A,B)$, the more similar sets A, B are. In fact, $H(A,B)$ is equal to 0 if and only if sets A, B are identical.

Fidelity of a mesh is measured as the 2-sided Hausdorff distance H of the following sets:

- set A: a densely sampled point set on the surface of the biological object, and
- set B: a densely sampled point set on the boundary facets of the mesh.

Notice that the mesh boundary point set B does not consist of only boundary mesh vertices. The reason is because otherwise, at least one side of the Hausdorff distance of the meshes produced by HQD would always be 0 (or very close to 0), since this method guarantees that the boundary mesh vertices lie precisely on the object surface.

Having defined fidelity, we proceed by explaining how we control fidelity for each mesher.

For HQD, this is possible through the parameter δ (see [9] for a more detailed explanation). Low values of δ increase the sampling on the object surface which yields better fidelity. High values of δ produce meshes whose boundary crudely approximates the real surface.

For Tetgen, we had to change the fidelity of the PLC given by CGAL. We, therefore, had to adjust two parameters responsible for the PLC's fidelity. The first imposes an upper bound on the circumradius of the *Delaunay balls* and the second forces an upper bound on the distance between the circumcenter of the boundary facets and the corresponding center of their Delaunay balls. More information can be found in [4].

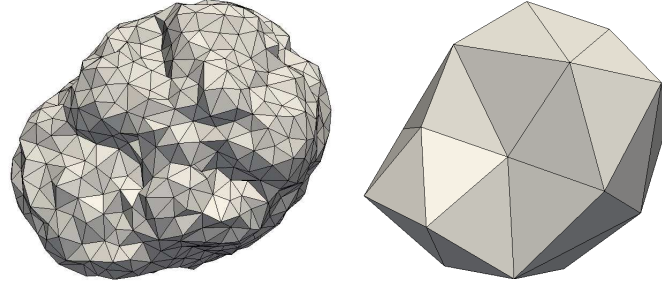**Table 1** Meshes with varying fidelity obtained by HQD.

H	Minimum Dihedral Angle	Average Minimum Dihedral Angle	#Tetrahedra	#Vertices	Condition Number	Error
22.81	8.68	39.92	40	23	35,205.00	0.40
20.22	5.86	35.73	52	27	37,988.00	0.36
19.94	2.73	30.99	96	41	93,179.00	0.47
17.92	4.10	31.81	89	38	62,404.00	0.52
17.52	5.46	33.74	61	30	31,352.00	0.49
16.57	4.10	31.39	77	34	24,984.00	0.32
16.15	6.17	31.83	148	61	99,449.00	0.30
15.28	7.70	37.53	168	71	40,350.00	0.27
13.49	4.08	33.39	297	113	46,260.00	0.40
9.86	2.46	34.05	228	89	26,399.00	0.38
9.23	3.61	36.15	425	157	51,487.00	0.25
9.09	6.01	35.95	578	200	37,427.00	0.26
8.72	2.36	33.63	385	137	53,977.00	0.28
8.47	4.07	36.10	771	261	292,370.00	0.21
7.11	1.27	36.18	1,157	367	319,850.00	0.27
6.24	0.34	35.71	1,681	521	594,820.00	0.24
5.84	0.92	35.93	2,746	814	1,559,500.00	0.25

For PBM, controlling fidelity is accomplished by adjusting the parameter λ. This parameter defines the trade-off between quality and fidelity: high values of λ make the optimization more sensitive to good fidelity, while low values do not change a lot the position of the initial (high-quality) BCC. However, we observed that λ does not offer a very flexible control over flexibility. Therefore, to get meshes of substantially different fidelity, we had to change not only λ but also the density of the initial BCC.

4.3 Results

Table 1 presents the results obtained by various meshes produced by HQD. Each row corresponds to a singe mesh. Column H contains the Hausdorff distance between the mesh and the object surface. The table illustrates meshes ordered in increasing fidelity (i.e., in decreasing Hausdorff distance). It also shows the minimum and the average minimum dihedral angle of the mesh, as well as the total number of tetrahedra and vertices of the mesh. The condition number depicted is of the matrix $\mathbf{K} + \mathbf{H}^\top \mathbf{H}$ which is responsible for the accuracy and speed of the NRR solver (see Section 2). Finally, the last column reports the NRR error —as defined in Equation (2) —obtained after the end of the registration process.

We observe that the error does not fluctuate considerably. All the errors are about less than half the size of a voxel (the size of the voxel is $1 \times 1 \times 1$), even when the H distance is very large. Figure 2 illustrates the meshes obtained by HQD for the best and the worst fidelity.

(a) *H* equal to 5.84 (best fidelity).   (b) *H* equal to 22.81 (worst fidelity).

Fig. 2 Meshes produced by HQD.

Table 2 Meshes with varying fidelity obtained by Tetgen and CGAL.

H	Minimum Dihedral Angle	Average Minimum Dihedral Angle	#Tetrahedra	#Vertices	Condition Number	Error
23.23	5.60	24.09	262	104	292,820.00	0.25
20.03	5.60	24.21	264	105	92,234.00	0.41
18.84	3.61	20.51	371	142	4,175,200.00	0.36
17.33	7.36	24.51	207	82	123,670.00	0.41
16.25	4.92	28.11	179	148	211,360.00	0.40
14.98	6.97	26.84	141	59	17,882.00	0.50
14.36	4.02	22.16	609	224	1,098,800.00	0.33
13.53	4.92	28.49	156	143	298,850.00	0.36
12.43	6.88	26.72	320	185	1,209,500.00	0.39
11.47	5.77	25.83	227	88	58,552.00	0.42
10.22	3.76	21.82	1,052	377	1,715,700.00	0.46
9.74	4.51	21.11	946	337	4,400,400.00	0.37
8.54	2.20	21.58	1,500	531	2,418,900.00	0.41
7.92	2.29	21.54	2,010	710	28,992,000.00	0.45
7.35	1.88	20.77	2,539	878	6,459,100.00	0.43
6.02	1.52	21.17	7,006	2424	1,941,500,000.00	n/a
5.88	1.33	20.65	4,547	1585	205,230,000.00	n/a

Table 2 shows the results for Tetgen. Similarly, fidelity does not seem to affect the error considerably. Also, although the minimum dihedral angles are larger than those in HQD, the average minimum dihedral angles are 10 to 15 degrees less than those in HQD. This results in generally higher error than the error in HQD, but still the differences in accuracy are not very obvious. However, the much larger condition numbers affect the speed of the solver a lot. Actually, for the bottom two runs (corresponding to the meshes with the two best fidelity values and with the two higher condition numbers), the solver could not even converge. Figure 3 illustrates the meshes obtained by Tetgen for the best and the worst fidelity.

Table 3 presents the results for the PBM mesh. We observe that the quality is very good: the minimum and the average minimum dihedral angles reach perfection. This results in much lower condition numbers and generally lower error than HQD and Tetgen. Again, we observe that fidelity does not play that important role in the
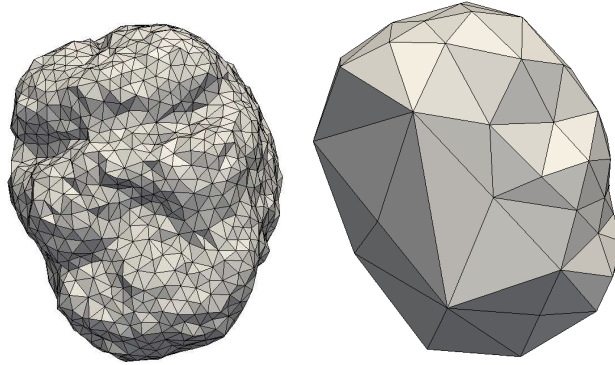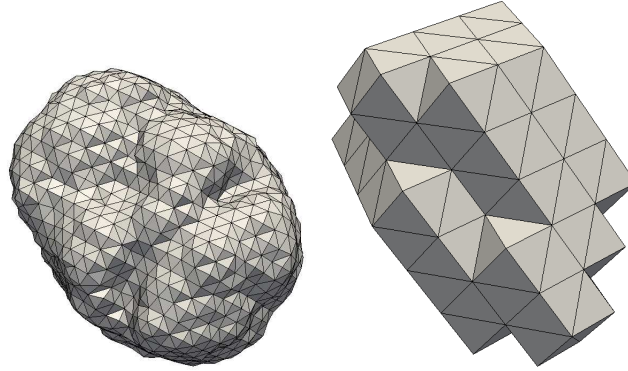
(a) H equal to 5.88 (best fidelity). (b) H equal to 23.23 (worst fidelity).

Fig. 3 Meshes produced by Tetgen and CGAL.

Table 3 Meshes with varying fidelity obtained by PBM.

H	Minimum Dihedral Angle	Average Minimum Dihedral Angle	#Tetrahedra	#Vertices	Condition Number	Error
21.02	60.00	60.00	465	139	122,470.00	0.19
19.56	60.00	60.00	1,144	303	203,230.00	0.53
18.30	60.00	60.00	2,126	519	223,540.00	0.50
15.39	41.21	54.43	465	139	19,711.00	0.17
14.25	35.54	54.48	1,144	303	23,405.00	0.35
13.94	31.68	53.83	1,144	303	21,213.00	0.29
13.58	19.41	53.09	1,144	303	18,980.00	0.27
12.61	39.27	53.81	465	139	16,897.00	0.18
12.02	35.77	53.55	465	139	15,978.00	0.19
10.39	34.13	54.78	2,126	519	29,397.00	0.17
9.88	31.92	54.22	2,126	519	25,926.00	0.17
9.39	30.04	53.63	2,126	519	23,485.00	0.17
7.01	59.99	60.00	18,780	3811	367,690.00	0.06
6.42	14.28	55.15	5,764	1277	21,300.00	0.15
5.25	35.61	56.83	18,780	3811	101,700.00	0.09
4.99	31.92	56.47	18,780	3811	78,449.00	0.09
4.94	27.50	56.13	18,780	3811	76,065.00	0.09

accuracy of the NRR. Even meshes with very bad fidelity yield an error less than half the size of the voxel. Figure 4 illustrates the meshes obtained by PBM for the best and the worst fidelity.

As you can see in the last two rows of Table 2 (where the error is n/a), the low average minimum dihedral angles seem to substantially affect the speed of the solver: in these two specific runs the solver did not even converge. Also, see that in these two rows the condition number is extremely large. We wanted to look into the timings of both the meshers and the solver in more depth, and see what the merit of fidelity to speed is.

We selected 5 meshes from each method of approximately the same fidelity respectively and measured the time for meshing and the time for solving the regis-

(a) *H* equal to 4.94 (best fidelity).   (b) *H* equal to 21.02 (worst fidelity).

Fig. 4 Meshes produced by PBM.

Table 4 Timings (in seconds) for various meshes obtained by different methods. Both the mesh and the solver execution times are reported.

H	HQD			Tetgen			PBM		
	Mesher	Solver	Total	Mesher	Solver	Total	Mesher	Solver	Total
15-16.5	6.89	0.04	6.93	0.01	0.06	0.07	132.34	0.05	132.39
14-15.5	6.4	0.05	6.45	0.01	0.17	0.18	165.02	0.06	165.08
13-14.5	10.23	0.06	10.29	0.02	0.16	0.18	164.93	0.06	164.99
8.5-9.5	21.57	0.08	21.65	0.09	4.88	4.97	189.19	0.09	189.28
7-8	17.62	0.46	18.08	0.13	**45**	**45.13**	263.39	0.19	263.58

tration problem. For each case, the solver has been running until the error becomes less than 0.5 (half the size of the voxel). Table 4 summarizes the results.

We observe that the meshing time of PBM is extremely large: more than 2 minutes in all cases. Actually, most of this time is spent for the initial BCC creation. On the other hand, the CGAL+Tetgen scheme is very fast: less than 2 seconds in all cases, even for the bottom mesh which consists of 2,539 elements.

As far as the solver's time is concerned, PBM yields the best meshes. Overall, however, the registration process is much slower than the other methods due to the time consuming mesh generation time. For Tetgen, the solver took much time, when the Hausdorff distance dropped below 8.5 (see bold entries). As Table 2 shows, the minimum dihedral angle for this fidelity is more than 1°, but the very low average minimum dihedral angle (the lowest among all the methods) seems to affect the condition number a lot and consequently the speed of the solver. Although the HQD meshes have elements with very small angles, the average minimum angle is much better than Tetgen (10 to 15 degrees larger). This is why when the solver ran on HQD's meshes, its execution time was less than 2 seconds in all cases, yielding a good overall execution time, even when the *H* distance drops below 8.5.

5 Conclusions

In this section, we summarize our findings. The two Delaunay meshes (i.e., HQD and Tetgen) exhibit low quality when the fidelity increases substantially (when the Hausdorff distance drops below 8 units approximately, in our case studies). As Table 1 and Table 2 show, this quality deterioration yields a very large condition number which affects the execution time of the solver (see Table 4). We also observe that not only the minimum but also the average minimum dihedral angle plays an important role to the solver's speed. To see it, compare the solver's speed of HQD to the solver's speed of Tetgen when the Hausdorff distance of the meshes is between 7 and 8 units. When Tetgen's mesh was used, the solver was 45 times slower. For these values of fidelity, Tetgen meshes have better minimum dihedral angles than HQD meshes, but they also have much lower average minimum dihedral angles (15 degrees smaller), which is likely to be the reason for a much worse condition number and the consequent large execution time of the solver.

The accuracy of the solver on the meshes produced by the two Delaunay meshers does not fluctuate significantly by the different fidelity values (see Table 1 and Table 2). That means that the need for good surface approximation does not seem to affect the accuracy of the solver. Meshes approximating very crudely the object surface (see Figure 2(b) and Figure 3(b) for an illustration) yielded an error less than half the voxel size.

The main characteristic of the optimization-based mesher (i.e., PBM) is the high minimum and average dihedral angles, even in the case of very good fidelity. The reason is because relatively dense initial BCCs can easily capture the object surface without so much compression, thus preserving the good angles of the BCC triangulation. Of course, the number of elements increases significantly, which makes the mesh generation time extremely slow (see Table 4). We also observe that the solver on PBM's meshes exhibit the least error which in fact is achieved when fidelity is very good (less than 5 units approximately). This is reasonable because, as Table 3 suggests, good fidelity does not deteriorate the quality as much as is the case for the two Delaunay meshes. Notice, however, that even when the PBM meshes have very bad fidelity (see Figure 4(b)), the error does not increase significantly.

References

[1] CGAL, Computational Geometry Algorithms Library. `http://www.cgal.org`

[2] Archip, N., Clatz, O., Fedorov, A., Kot, A., Whalen, S., Kacher, D., Chrisochoides, N., Jolesz, F., Golby, A., Black, P., Warfield, S.K.: Non-rigid alignment of preoperative MRI, fMRI, DT-MRI, with intra-operative MRI for en-

chanced visualization and navigation in image-guided neurosurgery. Neuroimage **35**(2), 609–624 (2007)

[3] Bajaj, C.L., Oden, J.T., Diller, K.R., Browne, J.C., Hazle, J., Babuska, I., Bass, J., Bidaut, L., Demkowicz, L.F., Elliott, A., Feng, Y., Fuentes, D., Kwon, B., Prudhomme, S., Stafford, R.J., Zhang, Y.: Using Cyber-Infrastructure for Dynamic Data Driven Laser Treatment of Cancer. In: International Conference on Computational Science, pp. 972–979 (2007)

[4] Boissonnat, J.D., Oudot, S.: Provably good sampling and meshing of surfaces. Graphical Models **67**(5), 405–451 (2005)

[5] Cheng, S.W., Dey, T.K., Edelsbrunner, H., Facello, M.A., Teng, S.H.: Sliver exudation. Journal of the ACM **47**(5), 883–904 (2000)

[6] Clatz, O., Delingette, H., Talos, I.F., Golby, A.J., Kikinis, R., Jolesz, F., Ayache, N., Warfield, S.: Robust non-rigid registration to capture brain shift from intra-operative MRI. IEEE Transactions on Medical Imaging **24**(11), 1417–1427 (2005)

[7] Delingette, H., Ayache, N.: Soft tissue modeling for surgery simulation. Handbook of Numerical Analysis, Computational Models for the Human Body **XII**, 453–550 (2004)

[8] Ferrant, M.: Physics-based deformable modeling of volumes and surfaces for medical image registration, segmentation and visualization. Ph.D. thesis, Universite Catholique de Louvain (2001). URL `http://spl.harvard.edu:8000/pages/papers/ferrant/thesis/ferrant.pdf`

[9] Foteinos, P., Chernikov, A., Chrisochoides, N.: Guaranteed Quality Tetrahedral Delaunay Meshing for Medical Images. In: Proceedings of the 7th International Symposium on Voronoi Diagrams in Science and Engineering, pp. 215–223. Quebec City, Canada (2010)

[10] Labelle, F., Shewchuk, J.R.: Isosurface stuffing: fast tetrahedral meshes with good dihedral angles. ACM Transactions on Graphics **26**(3), 57 (2007)

[11] Liu, Y., Foteinos, P., Chernikov, A., Chrisochoides, N.: Multi-tissue mesh generation for brain images. In: Proceedings of the 19th International Meshing Roundtable. Chattanooga, Tennessee, USA (2010). To appear

[12] Oudot, S., Rineau, L., Yvinec, M.: Meshing Volumes Bounded by Smooth Surfaces. In: International Meshing Roundtable, pp. 203–220. Springer-Verlag, San Diego, California, USA (2005)

[13] Rineau, L., Yvinec, M.: Meshing 3D Domains Bounded by Piecewise Smooth Surfaces. In: International Meshing Roundtable, pp. 443–460 (2007)

[14] Shewchuk, J.R.: What is a Good Linear Element? - Interpolation, Conditioning, and Quality Measures. In: Proceedings of the 11th International Meshing Roundtable, pp. 115–126. Sandia National Laboratories (2002)

[15] Si, H.: Tetgen version 1.4.3. `http://tetgen.berlios.de/`

[16] Jane Tournois, Rahul Srinivasan, Pierre Alliez: Perturbing Slivers in 3D Delaunay Meshes. In: Proceedings of the 18th International Meshing Roundtable, pp. 157–173. Sandia Labs, Salt Lake City, Utah, USA (2009)