# ADAPTIVE TOPOLOGY-INDEPENDENT SKINNING WEIGHTS FOR STYLIZED 3D CHARACTERS FROM A GEODESIC DISTANCE FIELD ON THE MESH GRAPH

***Xabibullayeva (O'tkirbekova) Madinabonu Ravshanbek qizi***
*Tashkent University of Information Technologies*
*named after Muhammad al-Khwarizmi*
*Tashkent, Uzbekistan.*
*E-mail: otkirbekovamadinabonu@gmail.com*
*Scientific supervisor:* ***Kayumova Gulshan Asrorovna***
*PhD (Ped.), Associate Professor*

**Abstract.** Skinning weights computed from Euclidean distance to bones leak between body parts that are close in space but far apart on the surface, such as an arm resting against the torso. Stylized characters with thick limbs and narrow gaps suffer most. We propose an adaptive geodesic weight algorithm (AGW) that works directly on the mesh graph, needs no template skeleton, voxelization or training data, and consists of three steps: a geodesic distance field from each bone computed by Dijkstra's algorithm on mesh edges; a radius-normalized kernel with pruning; and an intrinsic heat diffusion whose length is set adaptively by the fold-free condition $\ell = 3 \cdot \tan(\theta_max/4) \cdot r$ derived in our previous work. On a torso–arm model with gaps of 0.25–1.0 arm radii and 90° abduction, AGW reduces weight leakage into the torso from 1.5–7.3% of vertices to 0.0–0.4%, lowers the RMS error of the torso by 7.8–14.7 times, and removes all folded edges for both linear and dual quaternion skinning. Diffusion alone does not remove leakage, and geodesic distance alone does not remove folds: both parts of the algorithm are necessary.

**Annotatsiya.** Suyaklargacha bo'lgan Evklid masofasidan hisoblangan vaznlar fazoda yaqin, ammo sirt bo'ylab uzoq qismlar orasida "oqib o'tadi", masalan tanaga tegib turgan qo'l. Maqolada shablon, voksellash va o'quv to'plami talab qilmaydigan, bevosita to'r grafida ishlaydigan adaptiv geodezik vaznlar algoritmi (AGW) taklif etiladi. Algoritm uch

bosqichdan iborat: to’r qirralarida Deykstra algoritmi bilan geodezik masofa maydoni; radius bo’yicha normallangan yadro; qatlanishsizlik sharti $\ell = 3 \cdot \tan(\theta_{max}/4) \cdot r$ asosida uzunligi moslashtiriladigan ichki issiqlik diffuziyasi. Tana–qo’l modelida AGW tanaga vazn oqishini 1,5–7,3% dan 0,0–0,4% gacha kamaytiradi, tana o’rtacha kvadratik xatoligini 7,8–14,7 barobar pasaytiradi va qatlanishlarni to’liq bartaraf etadi.

**Аннотация.** Веса скиннинга, вычисленные по евклидову расстоянию до костей, «перетекают» между частями тела, близкими в пространстве, но далёкими по поверхности. Предложен адаптивный алгоритм геодезических весов (AGW), работающий непосредственно на графе сетки без шаблона, вокселизации и обучающих данных: геодезическое поле расстояний по алгоритму Дейкстры, нормированное по радиусу ядро и внутренняя диффузия с адаптивной длиной $\ell = 3 \cdot \tan(\theta_{max}/4) \cdot r$. На модели «торс–рука» AGW снижает долю вершин торса с перетеканием веса с 1,5–7,3% до 0,0–0,4%, уменьшает среднеквадратичную погрешность торса в 7,8–14,7 раза и полностью устраняет перегибы.

**Keywords:** skinning weights, geodesic distance, mesh graph, Dijkstra algorithm, heat diffusion, Laplace–Beltrami operator, automatic rigging, stylized characters, adaptive framework.

## 1. Introduction

After a skeleton has been placed inside a character, each vertex must receive weights that say how strongly each bone moves it. Early automatic methods used the Euclidean distance from the vertex to the bone, often with a visibility test [8]. Baran and Popović computed weights by solving a heat equilibrium on the surface, where each bone heats only the vertices it can see in a straight line [1]. Jacobson et al. introduced bounded biharmonic weights, which are smooth and local but require a volumetric tetrahedral mesh [3]. Dionne and de Lasa proposed geodesic voxel binding, which measures distance through a voxelized interior and handles non-manifold production meshes [4]. Learning-based systems such as RigNet predict weights from large rigged datasets [12].

Each of these approaches has a cost. Euclidean and visibility methods fail when two

body parts are close: the arm is visible from the torso across a narrow gap. Volumetric and voxel methods need a second discretization of the interior whose resolution limits the thinnest part that can be separated. Learning methods need training data that is dominated by realistic humanoids and generalize poorly to cartoon proportions. The local school of engineering geometry gives the mathematical tools — mesh representation, distances and transformations on surfaces [13] — but has not applied them to weight computation; applied studies of character animation have concentrated on facial modelling [14] and on rigging workflows [17, 18].

In previous work we defined reproducible deformation criteria and derived a fold-free condition for the width of the blending zone [16]. This paper uses both. Its contributions are:

1. An algorithm that computes weights from geodesic distance on the mesh graph itself, with no template, voxel grid, tetrahedral mesh or training set.

2. An adaptive diffusion step whose length follows from the fold-free condition, so that smoothness is set by geometry instead of by hand.

3. A controlled experiment that separates the effect of the distance metric from the effect of diffusion and measures both with the criteria of [16].

## 2. Method

**2.1. Geodesic distance field.** The mesh M = (V, F) is treated as a weighted graph whose nodes are vertices and whose edge lengths are Euclidean lengths of mesh edges. For each bone j with segment [a_j, b_j] and radius r_j, the seed set consists of the vertices that lie on the bone's own surface:

$$S_j = \{\, v \in V : \operatorname{dist}\big(v, [a_j, b_j]\big) < 1.08\, r_j \,\}$$

The distance d_j(v) from every vertex to S_j is computed by a multi-source Dijkstra search [9]. Because paths follow the surface, a torso vertex facing the arm across a gap receives a distance equal to the path over the shoulder, not the width of the gap.

**2.2. Radius-normalized kernel.** Raw weights are obtained from a kernel scaled by

the bone radius, normalized, pruned and normalized again:

$$\widetilde{w}_j(v) = \left(1 + \frac{d_j(v)}{r_j}\right)^{-4}, \qquad w_j = \frac{\widetilde{w}_j}{\sum_k \widetilde{w}_k}, \qquad w_j \leftarrow 0 \text{ if } w_j < 0.05$$

Scaling by r_j makes the kernel independent of character size. Pruning removes the long tails that would otherwise move distant parts by small amounts.

**2.3. Adaptive intrinsic diffusion.** Geodesic weights change sharply at the junction of two parts, and a sharp transition folds under large rotations (Proposition 2 in [16]). We smooth the weights by one implicit step of the heat equation on the surface, using the cotangent Laplacian L and the mass matrix M [2, 10, 15]:

$$(M + 1/2\,\ell^2 L)w_j^* = M\,w_j, \qquad \ell = 3\tan\frac{\theta_{\max}}{4}\; r$$

Here θ_max is the largest joint angle the rig must support and r is the radius of the thinner of the two parts at the joint. Diffusion runs along the surface, so it widens the transition without reintroducing leakage across gaps. The result is clamped to non-negative values and normalized.

**2.4. Complexity.** Dijkstra's search costs O(|E| log |V|) per bone. The diffusion step needs one sparse Cholesky or LU factorization shared by all bones and one solve per bone.

## 3. Experiment

**Model.** A torso–arm character was defined as a smooth union of capsules: a torso of radius 0.35 and length 1.6, and an arm of radius r_a = 0.12 hanging parallel to the torso with a gap g of 0.25, 0.5 or 1.0 arm radii, joined through a shoulder capsule. The surface was extracted by marching cubes at resolution 150, giving 28,050–28,542 vertices. Narrow gaps of this kind are typical of stylized characters with thick arms and bodies.

**Bones and pose.** Two bones were used: the spine along the torso axis and the arm from the shoulder joint to the hand. The arm was abducted by 45° and 90° about the shoulder joint.

**Etalon and criteria.** In a correct deformation the torso below the shoulder (z < 0.30)

must not move. This gives an exact etalon: the displacement of those vertices is the error. Following [16] we report the RMS error and the Hausdorff (maximum) error divided by r_a, the number of folded edges N_f (dihedral angle above 90°) and of edges with dihedral angle above 30°. Leakage is the share of torso vertices whose arm weight exceeds 0.05.

**Compared variants.** (A) Euclidean distance to the bone with the same kernel; (B) the geodesic field and kernel without diffusion; (C) the full AGW algorithm with $\ell = 1.25 \cdot r_a$, which is the fold-free value $3 \cdot \tan(22.5°) = 1.24$ rounded up for $\theta_{max} = 90°$; (D) Euclidean distance followed by the same diffusion, to test whether diffusion alone is enough. All variants were run with LBS and DQS [6, 11]. Computations used Python with NumPy, SciPy, scikit-image and the robust Laplacian of Sharp and Crane [15], on one Intel Xeon 2.8 GHz core.

## 4. Results

| Gap g / r_a | Variant | Leakage, % | RMS error / r_a (LBS; DQS) | Hausdorff / r_a (LBS; DQS) | Folded edges N_f (LBS; DQS) | Edges above 30° (LBS; DQS) |
|---|---|---|---|---|---|---|
| 0.25 | A Euclidean | 7.3 | 0.373; 0.393 | 3.27; 3.49 | 16; 6 | 145; 174 |
| 0.25 | B Geodesic | 0.0 | 0.003; 0.003 | 0.19; 0.19 | 22; 78 | 318; 330 |
| 0.25 | C AGW | 0.4 | 0.025; 0.026 | 0.38; 0.39 | 0; 0 | 1; 3 |
| 0.5 | A Euclidean | 5.5 | 0.222; 0.229 | 1.78; 1.86 | 16; 3 | 42; 90 |
| 0.5 | B Geodesic | 0.0 | 0.000; 0.000 | 0.00; 0.00 | 22; 78 | 320; 322 |

| Gap g / r_a | Variant | Leakage, % | RMS error / r_a (LBS; DQS) | Hausdorff / r_a (LBS; DQS) | Folded edges N_f (LBS; DQS) | Edges above 30° (LBS; DQS) |
|---|---|---|---|---|---|---|
| 0.5 | C AGW | 0.2 | 0.018; 0.019 | 0.27; 0.28 | 0; 0 | 0; 1 |
| 1.0 | A Euclidean | 1.5 | 0.093; 0.094 | 0.65; 0.66 | 14; 3 | 31; 39 |
| 1.0 | B Geodesic | 0.0 | 0.000; 0.000 | 0.00; 0.00 | 24; 86 | 291; 314 |
| 1.0 | C AGW | 0.0 | 0.012; 0.012 | 0.18; 0.18 | 0; 0 | 0; 0 |

*Table 1. Arm abduction 90°.*

Compared with Euclidean weights, AGW lowers the torso RMS error by 14.7, 12.1 and 7.8 times for the three gaps (LBS) and removes every folded edge. The advantage grows as the gap narrows, which is the situation of stylized characters. At 45° abduction the pattern is the same: RMS error 0.202, 0.120 and 0.050 r_a for Euclidean weights against 0.014, 0.010 and 0.006 r_a for AGW, with no folds.

**Both steps are necessary.** Geodesic distance alone (B) gives zero leakage but 22–86 folded edges at the shoulder, because the weight transition is too sharp. Diffusion alone (D) removes folds but not leakage: at $\ell = 1.25 \cdot r_a$ Euclidean weights still leak into 9.8%, 6.7% and 0.6% of the torso, with RMS error 0.290, 0.182 and 0.084 r_a (LBS, 90°). Only the combination removes both defects.

**The adaptive length.** Table 2 shows the effect of the diffusion length for the gap $g = 0.5 \cdot r_a$ at 90°.

| ℓ / r_a | 0 | 0.5 | 1.0 | 1.25 | 1.5 | 2.0 |
|---|---|---|---|---|---|---|
| Folded edges, LBS | 22 | 7 | 0 | 0 | 0 | 0 |

| ℓ / r_a | 0 | 0.5 | 1.0 | 1.25 | 1.5 | 2.0 |
|---|---|---|---|---|---|---|
| Folded edges, DQS | 78 | 0 | 0 | 0 | 0 | 0 |
| RMS error / r_a, LBS | 0.000 | 0.003 | 0.012 | 0.018 | 0.026 | 0.042 |
| Leakage, % | 0.0 | 0.0 | 0.0 | 0.2 | 0.4 | 0.8 |

*Table 2. Trade-off between folds and accuracy.*

Folds disappear at $\ell \geq 1.0 \cdot r_a$, while the error grows roughly linearly with $\ell$. The best value is therefore the smallest length that guarantees no folds for the required joint range. The condition from [16] gives this value without trial runs. Here it is slightly conservative, because a Gaussian diffusion profile is smoother than the smoothstep profile for which the condition was derived.

**Mesh-graph versus heat-method distance.** We also computed the geodesic field with the heat method of Crane et al. [5]. On the marching-cubes mesh it produced 1,133 edges above 30° and 79 folds (LBS, gap 0.25, 90°, no pruning), against 163 and 18 for Dijkstra on the mesh graph. The heat method is sensitive to poorly shaped triangles, while graph distance is exact on the graph and monotone. For production meshes of mixed quality the graph distance is the more robust choice.

**Cost and volume.** For 28,198 vertices the geodesic field took 0.48 s and the diffusion 0.47 s, both computed once per character. Global volume change at 90° was 0.35–0.50% for AGW against 2.1–6.5% for Euclidean weights under LBS.

## 5. Discussion

The result supports the working hypothesis of the research: weights derived from the intrinsic geometry of the mesh preserve the shape near the joint better than Euclidean weights. The improvement comes from two geometric facts. Surface distance separates parts that touch in space, and the fold-free condition fixes the smallest smoothing that the joint range allows.

The algorithm needs only the mesh and the skeleton. It does not depend on a humanoid template, so the same code applies to animals and stylized characters, and it

needs no GPU cluster or training data. This fits the aim of an adaptive framework: the weight transition at each joint is set from the local radius and the required motion range.

The experiment has limits. The model has two bones and an ideal etalon; real characters have many bones, and their etalon must come from artist-made rigs. The seed rule assumes a roughly tubular part around each bone; for flat or very irregular parts the seed set must be taken from the skeleton extraction stage, for example from mesh contraction [7]. Mesh-graph distance slightly overestimates true geodesic distance on coarse meshes, which is acceptable for weights but should be checked on low-polygon models.

## 6. Conclusion

An adaptive algorithm for skinning weights on the mesh graph has been proposed. It combines a geodesic distance field computed by Dijkstra's algorithm, a radius-normalized kernel and intrinsic diffusion whose length follows from the fold-free condition $\ell = 3 \cdot \tan(\theta_{max}/4) \cdot r$. On a torso–arm model with narrow gaps it reduced torso error by 7.8–14.7 times against Euclidean weights, reduced leakage to at most 0.4% and removed all folds under both LBS and DQS. The next step is to test the algorithm on a set of artist-rigged stylized characters and to connect it with automatic skeleton extraction.

**References**

1. Baran I., Popović J. Automatic rigging and animation of 3D characters // ACM Transactions on Graphics. 2007. Vol. 26, No. 3. Article 72.

2. Pinkall U., Polthier K. Computing discrete minimal surfaces and their conjugates // Experimental Mathematics. 1993. Vol. 2, No. 1. P. 15–36.

3. Jacobson A., Baran I., Popović J., Sorkine O. Bounded biharmonic weights for real-time deformation // ACM Transactions on Graphics. 2011. Vol. 30, No. 4. Article 78.

4. Dionne O., de Lasa M. Geodesic voxel binding for production character meshes // Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation. 2013. P. 173–180.

5. Crane K., Weischedel C., Wardetzky M. Geodesics in heat: a new approach to computing distance based on heat flow // ACM Transactions on Graphics. 2013. Vol. 32, No. 5. Article 152.

6. Kavan L., Collins S., Žára J., O'Sullivan C. Geometric skinning with approximate dual quaternion blending // ACM Transactions on Graphics. 2008. Vol. 27, No. 4. Article 105.

7. Au O.K.-C., Tai C.-L., Chu H.-K., Cohen-Or D., Lee T.-Y. Skeleton extraction by mesh contraction // ACM Transactions on Graphics. 2008. Vol. 27, No. 3. Article 44.

8. Wareham R., Lasenby J. Bone glow: an improved method for the assignment of weights for mesh deformation // Articulated Motion and Deformable Objects (AMDO 2008). LNCS 5098. Berlin: Springer, 2008. P. 63–71.

9. Dijkstra E.W. A note on two problems in connexion with graphs // Numerische Mathematik. 1959. Vol. 1. P. 269–271.

10. Botsch M., Kobbelt L., Pauly M., Alliez P., Lévy B. Polygon Mesh Processing. Natick: A K Peters, 2010. 250 p.

11. Jacobson A., Deng Z., Kavan L., Lewis J.P. Skinning: real-time shape deformation // ACM SIGGRAPH 2014 Courses. 2014.

12. Xu Z., Zhou Y., Kalogerakis E., Landreth C., Singh K. RigNet: neural rigging for articulated characters // ACM Transactions on Graphics. 2020. Vol. 39, No. 4. Article 58.

13. Nazirov Sh.A., Nuraliyev F.M., To'rayev B.Z. Kompyuter grafikasi va dizayn: o'quv qo'llanma. Toshkent: Fan va texnologiya, 2015. 256 b. ISBN 978-9943-990-80-7.

14. Kayumova G., Boymurodov B. Uch o'lchamli personajlarning yuz holatini modellashtirish va animatsiyalashning zamonaviy yondashuvlari // Al-Farg'oniy avlodlari. 2025. Vol. 1, No. 3. P. 151–158. DOI: 10.5281/zenodo.17295640.

15. Sharp N., Crane K. A Laplacian for nonmanifold triangle meshes // Computer Graphics Forum. 2020. Vol. 39, No. 5. P. 69–80.

16. O’tkirbekova M.R. Skeleton construction and weight distribution in 3D animation // Ilm-fan taraqqiyotida raqamli iqtisodiyot va zamonaviy ta’limning o’rni hamda rivojlanish omillari: konferensiya materiallari. 2025.

17. Parent R. Computer Animation: Algorithms and Techniques. 3rd ed. Waltham: Morgan Kaufmann, 2012.