Volume xx (200y), Number z, pp. 1–3

ARAPLBS: Robust and efficient elasticity-based optimization of Weights and Skeleton Joints for Linear Blend Skinning with Parameterized Bones
– Additional Material –
Spines for linear blend skinning transformations

J.-M. Thiery[1,2†] E. Eisemann[1‡]

[1]Deflt University of Technology, The Netherlands [2]LTCI, Telecom-ParisTech, Université Paris-Saclay, Paris, France

Abstract

In this document, we present the formulas for the transformation of 3d points via spines, *a new type of deformer, in the context of linear-blend-skinning transformations. A spine is initially defined as a straight bone, which is virtually subdivided into an infinity of sub-bones, which are then all transformed in the same way. The name of this deformer hints at its use as a spine for 3D models. Compared with traditional rigid bones for LBS, whose parameters are a rotation and translation, a spine deformer has three additional control parameters: a roll axis, a roll amount, and a stretch value.*

1. Important notes

For convenience, this document also contains the related parts of the main article [TE17]. The derivation, which was left out of the paper, is described in this document in Sec. 4.

If you are using the following work, please cite the main article: [TE17].

2. Linear Blend Skinning (LBS) with parameterized bones

The location of a vertex i under LBS [MTLT*88, LCF00], using *parameterized bones*, is given by

$$f : v_i \mapsto \sum_{j \in B(i)} w_{ij} (R_j(v_i) \cdot v_i + T_j(v_i)), \quad (1)$$

where $R_j(v_i)$ is the rotation and $T_j(v_i)$ the translation applied by bone j on vertex i (these are constant for rigid bones – $(R_j(v_i), T_j(v_i)) = (R_j, T_j)\ \forall i$, but depend on v_i otherwise), $B(i)$ is the set of bones influencing i and w_{ij} the *weight* of bone j over i. We will refer to $\{B(i)\}$ as the *bone influence maps* and to $\{w_{ij}\}$ as the *weight maps*.

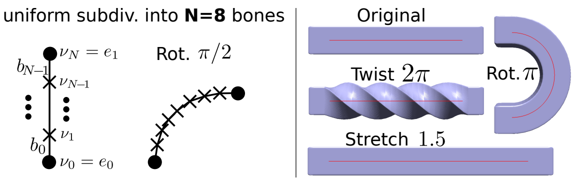


Figure 1: ***Left:*** *Approximation.* ***Right:*** *Continuous setting.*

Typically, skinning weights should verify:

1. **Affinity**: $\sum_j w_{ij} = 1$; reproduces rigid transformations.
2. **Positivity**: $w_{ij} \geq 0$; prevents unnatural behavior.
3. **Sparsity**: only *"few"* $w_{ij} > 0$; leads to *"simpler"* controls and a faster rendering process *(ill-defined)*.
4. **Locality**: bones should have *"small"* influence zones; improves control over editing operations *(ill-defined)*.

3. Spines

An example of parameterized bones are TSBs [JS11]. Our approach is compatible with these, as well as a generaliza-

† jean-marc.thiery@telecom-paristech.fr
‡ e.eisemann@tudelft.nl

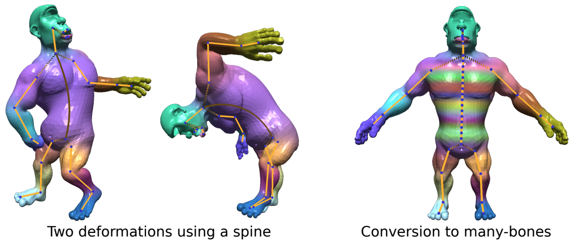


Figure 2: *Conversion of spines to many-bones.* ***Left:*** *deformations using a spine for the torso.* ***Right:*** *Converting the spine to 16 bones and their weight maps.*

tion, *spines*, described here. A *spine* $\mathbf{s}$ is a skeletal segment $[e_0, e_1]$ subdivided into infinitely-small bones undergoing the same transformation (Fig. 1). The name reflects its suitability to represent spines in models. Fixing e_0, its transformation combines a stretch $\sigma_\mathbf{s}$, affecting its length, and a rotation around an axis $a_\mathbf{s}$, such that the accumulated rotation from e_0 to e_1 amounts to $\theta_\mathbf{s}$. Similar to other bones, a spine $\mathbf{s}$ also has a rigid transformation applied to e_0, the *spine's base transformation* $(R_\mathbf{s}, t_\mathbf{s})$. In consequence, spines have 5 parameters: $(R_\mathbf{s}, t_\mathbf{s}, \sigma_\mathbf{s}, \theta_\mathbf{s}, a_\mathbf{s})$, and TSBs are twist-restricted spines (i. e., $a_\mathbf{s} = \overrightarrow{e_0e_1}/||\overrightarrow{e_0e_1}||$ in their case).

A point p with parameter $u_p \in [0,1]$ (describing "which small bone" p is attached to) is transformed by a spine as

$$p \mapsto R_\mathbf{s} R_{\text{loc}}(u_p)\cdot p \quad + \quad R_\mathbf{s}\cdot t_{\text{loc}}(u_p) + t_\mathbf{s} \quad \text{, with} \tag{2}$$

$$R_{\text{loc}}(u) := Rot(a_\mathbf{s}, u\theta_\mathbf{s}) \text{, and} \tag{3}$$

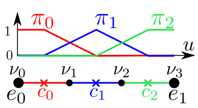
$$\begin{aligned} t_{\text{loc}}(u) := {} & (\text{Id} - R_{\text{loc}}(u))\cdot e_0 + \\ & \sigma_\mathbf{s}.(\sin(u\theta_\mathbf{s})/\theta - u\cos(u\theta_\mathbf{s}))(\text{Id} - a_\mathbf{s}\cdot {a_\mathbf{s}}^{\text{T}})\cdot \overrightarrow{e_0e_1} + \\ & \sigma_\mathbf{s}.(u\sin(u\theta_\mathbf{s}) + (\cos(u\theta_\mathbf{s}) - 1)/\theta_\mathbf{s}) a_\mathbf{s}\times \overrightarrow{e_0e_1} + \\ & u.(\sigma_\mathbf{s} - 1) R_{\text{loc}}(u)\cdot \overrightarrow{e_0e_1}. \end{aligned} \tag{4}$$

3.1. Mesh parameterization

Given a spine $\mathbf{s}$ with vertices e_0, e_1, one needs to define the above parametrization $u_\mathbf{s} : \mathcal{V} \mapsto [0,1]$ on the mesh vertices $\mathcal{V}$. It is possible to use complex (or artistically-driven) definitions [JS11], but by default, we use a linear parameterization: $u_\mathbf{s}(v_i) = \max(0, \min(1, \overrightarrow{e_0v_i}^{\text{T}}\cdot \overrightarrow{e_0e_1}/||\overrightarrow{e_0e_1}||^2))$.

3.2. Conversion of spines to many-bones

If a system only supports rigid bones, $\mathbf{s}$ can be converted into a set of them: $\mathbf{s}$ is cut into n bones with joints $\nu_0\cdots\nu_n$ and for each ν_j its corresponding u-parameter u_j is determined. We define a unity partition as piecewise-linear functions $\{\pi_j : [0,1] \mapsto [0,1]\}$, where $\pi_j(c_k) = \delta_j^k$, with $c_j = (u_j + u_{j+1})/2$, and impose $\pi_0(0) = \pi_n(1) = 1$ (the inset shows the case for 3 bones). The weight of a mesh vertex v_i (with weight $w_{i\mathbf{s}}$ w.r.t. the spine) w.r.t. bone k is defined as $w_{i\mathbf{s}}\pi_k(u_\mathbf{s}(v_i))$.



3.3. Jacobians

The Jacobian of a normal bone is simply its rotation. For a parameterized bone with transformation given by Eq. 2, its Jacobian at point p with parameter $u(p)$ is

$$J_p = R_\mathbf{s}\left(R_{\text{loc}}(u_p) + (R'_{\text{loc}}(u_p)\cdot p + t'_{\text{loc}}(u_p))\cdot \overrightarrow{\nabla} u^{\text{T}}(p)\right) \tag{5}$$

In Eq.5, $\overrightarrow{\nabla} u(p)$ is needed. Our work uses a linear parametrization, which is is not differentiable at $u = 0$ and 1. Moreover, such parameterizations are generally defined on the mesh only (e. g., hand-painted, or resulting from a diffusion on the mesh), and their gradients do not have a closed-form expression. We thus estimate the gradient $\overrightarrow{\nabla} u_i$ at vertex i (while keeping it aligned with the bone) as

$$\overrightarrow{\nabla} u_i := \underset{g \mid g\times\overrightarrow{e_0e_1} = \vec{0}}{\text{argmin}} \sum_{k\in V_1(i)} \lambda_{ik} ||g^{\text{T}}\cdot e_{ik} - (u(v_k) - u(v_i))||^2$$

Noting $a_{\mathbf{s}[\times]}$ the 3×3-matrix such that $a_{\mathbf{s}[\times]}\cdot v = a_\mathbf{s}\times v$, $\forall v \in \mathbb{R}^3$, the derivatives of $R_{\text{loc}}(u)$ and $t_{\text{loc}}(u)$ are given by:

$$\begin{aligned} R'_{\text{loc}}(u) = {} & \theta_\mathbf{s}\cos(u\theta_\mathbf{s}) a_{\mathbf{s}[\times]} + \theta_\mathbf{s}\sin(u\theta_\mathbf{s})(a_\mathbf{s}\cdot {a_\mathbf{s}}^{\text{T}} - \text{Id}) \\ t'_{\text{loc}}(u) = {} & - R'_{\text{loc}}(u)\cdot e_0 + \\ & \sigma_\mathbf{s}.(\cos(u\theta_\mathbf{s}) + u\theta_\mathbf{s}\sin(u\theta_\mathbf{s}))(\text{Id} - a_\mathbf{s}\cdot {a_\mathbf{s}}^{\text{T}})\cdot \overrightarrow{e_0e_1} + \\ & \sigma_\mathbf{s}.(u\theta_\mathbf{s}\cos(u\theta_\mathbf{s}) - \sin(u\theta_\mathbf{s})) a_\mathbf{s}\times \overrightarrow{e_0e_1} + \\ & (\sigma_\mathbf{s} - 1) R_{\text{loc}}(u)\cdot \overrightarrow{e_0e_1} + \\ & u.(\sigma_\mathbf{s} - 1) R'_{\text{loc}}(u)\cdot \overrightarrow{e_0e_1}. \end{aligned}$$

4. Local transformation by a Spine

Here, we first derive the *local* transformation $(R_{\text{loc}}(u), t_{\text{loc}}(u))$ of a spine (where local implies that e_0 is fixed). Initially, we will investigate the case, where the spine is subdivided into N subsegments, which all undergo the same local rotation of axis $\mathbf{a}$, angle θ/N (hereby, the accumulated rotation at e_1 is θ), and a stretch σ in the direction of the spine. In this formulation, bone i links vertices ν_i and ν_{i+1}. Induced by the preceding bones of the spine, it undergoes a global transformation given by the stretch in direction $\overrightarrow{e_0e_1}$, as well as a rotation R_i, and a translation t_i (i. e., $p \rightarrow R_i\cdot p + t_i$). Finally, we will express the rigid transformation along the segment at a parameter $u \in [0,1]$ via a rotation $R_{\text{loc}}(u)$ and a translation $t_{\text{loc}}(u)$ by considering the limit for $N \rightarrow \infty$: $((R_{\text{loc}}(u), t_{\text{loc}}(u)) = \lim_{N\rightarrow\infty}(R_{[uN]}, t_{[uN]}))$.

In a first step, we notice that

$$R_{\text{loc}}(u) = Rot(\mathbf{a}, u\theta)$$

We note $\bar{p}$ the transformation induced by the stretch at point p with parameter u: $\bar{p} = p + u(\sigma - 1)\overrightarrow{e_0e_1}$.

The final transformation can be obtained by first stretching the space in the spine direction (given by $p \mapsto \bar{p}$), and then applying the transformation of the stretched spine on the stretched space (which we note $\bar{p} \mapsto R_{\text{loc}}(u)\cdot\bar{p} + t_{\text{loc}}^{str(\sigma)}(u)$).

Regarding the latter, one can check by recurrence that: $t_i^{str(\sigma)} = \sum_{k=1}^{i} [R_{\text{loc}}((k-1)/N) - R_{\text{loc}}(k/N)] \cdot \overline{v_k}$, (since bones $k-1$ and k both transform $\overline{v_k}$ similarly, i. e., $R_k \overline{v_k} + t_k^{str(\sigma)} = R_{k-1}\overline{v_k} + t_{k-1}^{str(\sigma)}$), and that $t_0^{str(\sigma)} = 0$, which, when using $\overline{v(x)} = e_0 + x\sigma\overrightarrow{e_0 e_1}$ and taking the limit, gives:

$$t_{\text{loc}}^{str(\sigma)}(u) = \int_{x=0}^{u} \frac{-dR_{\text{loc}}}{dx}(x)dx \cdot e_0 + \sigma \int_{x=0}^{u} -x\frac{dR_{\text{loc}}}{dx}(x)dx \cdot \overrightarrow{e_0 e_1}$$

By decomposing the rotation matrix $R_{\text{loc}}(x) = Rot(\mathbf{a}, x\theta)$ as $R_{\text{loc}}(x) = \cos(x\theta)\text{Id} + \sin(x\theta)\mathbf{a}_{[\times]} + (1-\cos(x\theta))\mathbf{a}\cdot\mathbf{a}^{\text{T}}$ using Rodrigues' formula, we obtain

$$\begin{aligned} t_{\text{loc}}^{str(\sigma)}(u) =& (\text{Id} - R_{\text{loc}}(u)) \cdot e_0 + \\ & \sigma.(\sin(u\theta)/\theta - u\cos(u\theta)) \left(\text{Id} - \mathbf{a}\cdot\mathbf{a}^{\text{T}}\right) \cdot \overrightarrow{e_0 e_1} + \\ & \sigma.(u\sin(u\theta) + (\cos(u\theta)-1)/\theta)\mathbf{a} \times \overrightarrow{e_0 e_1} \end{aligned}$$

Since the final transformation is the composition of the stretch and the transformation by the stretched spine on the stretched space (first stretch, then rotate) (i. e., $p \mapsto R_{\text{loc}}(u)\cdot\bar{p} + t_{\text{loc}}^{str(\sigma)}(u) = R(u)_{\text{loc}} \cdot (p + u(\sigma-1)\overrightarrow{e_0 e_1}) + t_{\text{loc}}^{str(\sigma)}(u)$), we finally obtain

$$t_{\text{loc}}(u) = t_{\text{loc}}^{str(\sigma)}(u) + u.(\sigma-1)R_{\text{loc}}(u)\cdot\overrightarrow{e_0 e_1}$$

Note that, although not strictly defined for $\theta = 0$, the terms involving θ tend to 0 when θ tends to 0.

References

[JS11] JACOBSON A., SORKINE O.: Stretchable and twistable bones for skeletal shape deformation. *ACM Transactions on Graphics 30*, 6 (2011), 165. 1, 2

[LCF00] LEWIS J. P., CORDNER M., FONG N.: Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques* (2000), pp. 165–172. 1

[MTLT*88] MAGNENAT-THALMANN N., LAPERRIRE R., THALMANN D., ET AL.: Joint-dependent local deformations for hand animation and object grasping. In *Proceedings on Graphics interface 88* (1988). 1

[TE17] THIERY J.-M., EISEMANN E.: Araplbs: Robust and efficient elasticity-based optimization of weights and skeleton joints for linear blend skinning with parametrized bones. *Computer Graphics Forum* (2017). URL: http://dx.doi.org/10.1111/cgf.13161, doi:10.1111/cgf.13161. 1