Functional Optimization of Fluidic Devices with Differentiable Stokes Flow

TAO DU, MIT CSAIL
KUI WU, MIT CSAIL
ANDREW SPIELBERG, MIT CSAIL
WOJCIECH MATUSIK, MIT CSAIL
BO ZHU, Dartmouth College
EFTYCHIOS SIFAKIS, University of Wisconsin-Madison

ACM Reference Format:
Tao Du, Kui Wu, Andrew Spielberg, Wojciech Matusik, Bo Zhu, and Eftychios Sifakis. 2020. Functional Optimization of Fluidic Devices with Differentiable Stokes Flow. *ACM Trans. Graph.* 39, 6, Article 197 (December 2020), 2 pages. https://doi.org/10.1145/3414685.3417795

1 SIGNED DISTANCE FUNCTIONS OF BÉZIER CURVES

1.1 Basic Case

Consider a Bézier curve $\boldsymbol{s}(t) : \mathcal{R} \rightarrow \mathcal{R}^2$ defined as follows:

$$\boldsymbol{s}(t) = \begin{bmatrix}\boldsymbol{c}_0 & \boldsymbol{c}_1 & \boldsymbol{c}_2 & \boldsymbol{c}_3\end{bmatrix} \begin{bmatrix}1 & -3 & 3 & -1\\ 0 & 3 & -6 & 3\\ 0 & 0 & 3 & -3\\ 0 & 0 & 0 & 1\end{bmatrix} \begin{bmatrix}1\\ t\\ t^2\\ t^3\end{bmatrix} \tag{1}$$

where each $\boldsymbol{c}_i \in \mathcal{R}^2$ represents a control point and $t \in [0, 1]$. Given any point $\boldsymbol{p} \in \mathcal{R}^2$, the distance from $\boldsymbol{p}$ to $\boldsymbol{s}(t)$ is defined as $\|\boldsymbol{p} - \boldsymbol{s}(t^*)\|_2$ where $t^* = \arg\min_{0\le t\le 1} \|\boldsymbol{p} - \boldsymbol{s}(t)\|_2$. By setting the gradients of $\|\boldsymbol{p} - \boldsymbol{s}(t)\|_2^2$ with respect to t to 0, we obtain the follow 5-th order polynomial:

$$(\boldsymbol{p} - \boldsymbol{s}(t))^T \boldsymbol{s}'(t) = 0 \tag{2}$$

where $\boldsymbol{s}'(t)$ is

$$\boldsymbol{s}'(t) = \begin{bmatrix}\boldsymbol{c}_0 & \boldsymbol{c}_1 & \boldsymbol{c}_2 & \boldsymbol{c}_3\end{bmatrix} \begin{bmatrix}1 & -3 & 3 & -1\\ 0 & 3 & -6 & 3\\ 0 & 0 & 3 & -3\\ 0 & 0 & 0 & 1\end{bmatrix} \begin{bmatrix}0 & 0 & 0\\ 1 & 0 & 0\\ 0 & 2 & 0\\ 0 & 0 & 3\end{bmatrix} \begin{bmatrix}1\\ t\\ t^2\end{bmatrix}. \tag{3}$$

Therefore, t^* is either at the bounds (0 or 1) or one of the roots of the polynomial. We can find t^* by evaluating $\|\boldsymbol{p} - \boldsymbol{s}(t)\|_2$ with all the candidate t and selecting the minimum.

To determine the sign of the distance, we assume that varying t from 0 to 1 circles the solid region counterclockwise. Once we obtain t^*, the sign is determined by checking the sign of the determinant of the matrix $[\boldsymbol{s}(t^*) - \boldsymbol{p} \quad \boldsymbol{s}'(t^*)]$. We have now described how the signed distance function of a Bézier curve is computed. We evaluate the signed distance at every grid node in our algorithm.

To explain how we compute the gradients of the signed distance at each grid node with respected to the shape parameters, we again consider a point $\boldsymbol{p} \in \mathcal{R}^2$ and its distance $d(\boldsymbol{C}) = \|\boldsymbol{p} - \boldsymbol{s}(t^*(\boldsymbol{C}))\|_2$ where $\boldsymbol{C} = [\boldsymbol{c}_0 \quad \boldsymbol{c}_1 \quad \boldsymbol{c}_2 \quad \boldsymbol{c}_3]$ represents all the control points. We are interested in computing the gradients of d with respect to $\boldsymbol{C}$:

$$\frac{\partial d}{\partial \boldsymbol{C}_{ij}} = -\frac{(\boldsymbol{p} - \boldsymbol{s}(t^*(\boldsymbol{C})))^T}{\|\boldsymbol{p} - \boldsymbol{s}(t^*(\boldsymbol{C}))\|_2} \frac{\partial \boldsymbol{s}(t^*(\boldsymbol{C}), \boldsymbol{C})}{\partial \boldsymbol{C}_{ij}}. \tag{4}$$

The last term is defined as follows:

$$\frac{\partial \boldsymbol{s}(t^*(\boldsymbol{C}), \boldsymbol{C})}{\partial \boldsymbol{C}_{ij}} = \boldsymbol{I}_{ij} \begin{bmatrix}1 & -3 & 3 & -1\\ 0 & 3 & -6 & 3\\ 0 & 0 & 3 & -3\\ 0 & 0 & 0 & 1\end{bmatrix} \begin{bmatrix}1\\ t^*(\boldsymbol{C})\\ (t^*(\boldsymbol{C}))^2\\ (t^*(\boldsymbol{C}))^3\end{bmatrix} \tag{5}$$

where $\boldsymbol{I}_{ij}$ is a 2×4 matrix whose entry at the i-th row and the j-th is 1 and 0 elsewhere. Note that the contribution of $\frac{\partial t^*}{\partial \boldsymbol{C}_{ij}}$ in the gradients above is ignored because of the envelope theorem.

1.2 2D and 3D Variants

To support continuous closed loops in 2D, we consider concatenating multiple Bézier curves with shared first and last control points. Specifically, we pick k Bézier curves $\boldsymbol{s}^0(t), \boldsymbol{s}^1(t), \cdots, \boldsymbol{s}^{k-1}(t)$ and let $\boldsymbol{c}_0^i, \boldsymbol{c}_1^i, \boldsymbol{c}_2^i, \boldsymbol{c}_3^i$ be the control points of the i-th Bézier curve. By enforcing $\boldsymbol{c}_3^i = \boldsymbol{c}_0^{\mathrm{mod}(i+1,k)}$ and $\boldsymbol{c}_2^i, \boldsymbol{c}_3^i, \boldsymbol{c}_1^{\mathrm{mod}(i+1,k)}$ are colinear for every i, we obtain a continuous closed loop parameterized by the union of all control points. To avoid self loops, we parameterize the control points in a polar coordinate system whose origin is at the center of the loop, and we place each control point counterclockwise. For any point $\boldsymbol{p} \in \mathcal{R}^2$, its signed distance to this parameterized loop and its gradients can be easily inferred by first determining the nearest Bézier curve followed by applying the computation derived in the previous section.

We extend our discussion to 3D surface parameterization by considering the NURBS surfaces formed by the product of a closed loop in the xy-plane and a Bézier curve along the z direction. Regarding the signed distance function and its gradients, we approximate the distance from any point $\boldsymbol{p} \in \mathcal{R}^3$ to such a NURBS surface with the distance from $\boldsymbol{p}$ to the slice of the NURBS surface at the same z-coordinate as $\boldsymbol{p}$, which allows us to reuse the results from 2D Bézier curves.

Authors' addresses: Tao Du, MIT CSAIL, taodu@csail.mit.edu; Kui Wu, MIT CSAIL, kuiwu@csail.mit.edu; Andrew Spielberg, MIT CSAIL; Wojciech Matusik, MIT CSAIL, wojciech@csail.mit.edu; Bo Zhu, Dartmouth College, bo.zhu@dartmouth.edu; Eftychios Sifakis, University of Wisconsin-Madison, sifakis@cs.wisc.edu.

0730-0301/2020/12-ART197
https://doi.org/10.1145/3414685.3417795

2 FITTING BOUNDARIES IN MIXED CELLS

We now describe in detail how the boundaries are determined in a mixed cell as well as how the gradients are computed. We explain the fitting procedure in 3D, and the 2D case can be derived similarly. Without loss of generality, we consider a mixed cell of unit size whose lower left corner is at the origin $(0, 0, 0)$. Given the signed distance $d_{ijk} \in \mathcal{R}, i, j, k \in \{0, 1\}$ at its 8 corners (i, j, k), our goal is to find a plane whose signed distances evaluated at these corners well approximate d_{ijk}. Since the signed distance function of a plane is linear, we assume $d_{ijk} \approx ai + bj + ck + d$ and find (a, b, c, d) by solving the following linear least-square problem:

$$\min_{(a,b,c,d)} \sum_{ijk} (ai + bj + ck + d - d_{ijk})^2 \tag{6}$$

or in the equivalent matrix form:

$$(a, b, c, d) = \arg\min_{\boldsymbol{x} \in \mathcal{R}^4} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{d}\|_2^2 = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{d} \tag{7}$$

where $\boldsymbol{A}$ and $\boldsymbol{d}$ are defined as follows:

$$\boldsymbol{A} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}^T$$
$$\boldsymbol{d} = \begin{bmatrix} d_{000} & d_{001} & d_{010} & d_{011} & d_{100} & d_{101} & d_{110} & d_{111} \end{bmatrix}^T. \tag{8}$$

To derive the gradients of (a, b, c, d) with respect to the shape parameters, we notice that $(\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T$ is a constant matrix. By applying the chain rule, the gradients of (a, b, c, d) are $(\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T$ times the gradients of d_{ijk} with respect to the shape parameters, which can be found in the previous section.

3 GRADIENTS OF ELASTIC ENERGY IN MIXED CELLS

We consider a 3D mixed cell of unit size whose solid-fluid boundary is given by the plane $ax + by + cz + d = 0$. The elastic energy is the weighted sum of energy densities evaluated at fixed quadrature points, and only the weights are affected by the choice of shape parameters. To compute these weights, we rewrite Proposition 6.1 with $d = 3$:

$$w = |[0, 1]^3 \cap H| = \sum_{\boldsymbol{q} \in F^0 \cap H} \frac{(-1)^{|\boldsymbol{q}_0|}((a, b, c) \cdot \boldsymbol{q} + d)^3}{6abc} \tag{9}$$

where H is the half-space $ax + by + cz + d \geq 0$. Since the weight is computed for each subcell every quadrature belongs to, we need to scale the coordinates before applying Proposition 6.1 so that the subcell has unit size.

Computing the gradients of the stiffness matrix $\boldsymbol{K}$ requires computing the gradients of each weight with respect to the shape parameters. By applying the chain rule, this can be further reduced to the partial derivatives of $|[0, 1]^3 \cap H|$ with respect to (a, b, c, d):

$$\frac{\partial w}{\partial a} = \sum_{\boldsymbol{q} \in F^0 \cap H} \frac{(-1)^{|\boldsymbol{q}_0|}((a, b, c) \cdot \boldsymbol{q} + d)^2((2a, -b, -c) \cdot \boldsymbol{q} - d)}{6a^2bc}$$
$$\frac{\partial w}{\partial d} = \sum_{\boldsymbol{q} \in F^0 \cap H} \frac{(-1)^{|\boldsymbol{q}_0|}((a, b, c) \cdot \boldsymbol{q} + d)^2}{2abc} \tag{10}$$

Note that the partial derivatives with respect to b and c are omitted as a, b, and c are symmetric.

4 GRADIENTS OF CONSTRAINTS IN MIXED CELLS

As discussed in the paper, computing the constraints requires computing an integral over the solid-fluid boundary inside a mixed cell, which is discretized by partitioning the boundary into regions inside each subcell and projecting the quadrature onto the boundary plane. We will focus on explaining the contribution of one quadrature point to the integral as well as its gradients with respect to the shape parameters.

Consider a quadrature $\boldsymbol{p}$ and its subcell and assume the boundary plane in the mixed cell is $ax+by+cz+d = 0$. The projected quadrature $\boldsymbol{p}^\perp$ onto the boundary plane is given by the following equation:

$$\boldsymbol{p}^\perp = \boldsymbol{p} - \frac{(a, b, c) \cdot \boldsymbol{p} + d}{(a, b, c)^2}(a, b, c) \tag{11}$$

and the gradients of $\boldsymbol{p}^\perp$ with respect to (a, b, c) can be derived analytically. To compute the area of the boundary plane inside $\boldsymbol{p}$'s subcell, we reuse $\frac{\partial w}{\partial d}$ from the previous section: the derivative of w with respect to d should equal the area of the partition scaled by the length of the normal (a, b, c). Computing its gradients with respect to the shape parameters requires the second-order derivatives of w:

$$\frac{\partial^2 w}{\partial d \partial a} = \sum_{\boldsymbol{q} \in F^0 \cap H} \frac{(-1)^{|\boldsymbol{q}_0|}((a, b, c) \cdot \boldsymbol{q} + d)((2a, -b, -c) \cdot \boldsymbol{q} - d)}{3a^2bc}$$
$$- \sum_{\boldsymbol{q} \in F^0 \cap H} \frac{(-1)^{|\boldsymbol{q}_0|}((a, b, c) \cdot \boldsymbol{q} + d)^2}{6a^2bc}$$
$$\frac{\partial^2 w}{\partial d^2} = \sum_{\boldsymbol{q} \in F^0 \cap H} \frac{(-1)^{|\boldsymbol{q}_0|}((a, b, c) \cdot \boldsymbol{q} + d)}{abc} \tag{12}$$

Remark. It is possible that our projected quadrature may be outside of the boundary in the corresponding subcell. A more rigorous projection can be enforced by solving the following QP problem with bound constraints:

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \|\boldsymbol{p} - \boldsymbol{x}\|_2 \\ s.t. \quad & (a, b, c) \cdot \boldsymbol{x} + d = 0 \\ & \boldsymbol{x}_l \leq \boldsymbol{x} \leq \boldsymbol{x}_u \end{aligned} \tag{13}$$

where $\boldsymbol{x}_l, \boldsymbol{x}_u \in \mathcal{R}^3$ are the lower and upper bounds of the subcell. Indeed, $\boldsymbol{p}^\perp$ is the solution to this QP problem without the bound constraints. While it is possible to solve $\boldsymbol{p}^\perp$ from the more rigorous QP problem and derive its gradients by running sensitivity analysis on the KKT system, we didn't notice any significant differences when implementing both methods in 2D. Therefore, we stick to our current discretization paradigm due to its simplicity and smoothness.