Energy Function for the
Representation of Immersed
Elastic Boundaries

C. Peskin
Fall 2007
Consider an elastic material, the configuration of which is given by

\[ x = X(s) \]

where \( s = (s_1, \ldots, s_m) \) are material coordinates.

Let the elastic energy of the configuration \( X(s) \) be denoted

\[ E[X] \]

Now consider a perturbation \( X(s) \rightarrow X(s) + \delta X(s) \).

Then, to first order, the corresponding \( \delta E \) is a linear functional of \( \delta X \). As such, it may be written

\[ \delta E = \int \nabla \cdot \delta X(s) \, ds \]

where \( ds = ds_1 \ldots ds_m \). By definition, the coefficient \( \frac{\delta E}{\delta X(s)} \) is the "variational derivative" \( \delta E \), and is denoted \( \frac{\delta E}{\delta X(s)} \).
Example: A fiber continuum

\[ E = \int \mathcal{E} \left( \left| \frac{\partial X}{\partial \xi_1} \right| \right) \, d\xi \]

Note that

\[ \left| \frac{\partial X}{\partial \xi_1} \right| = \left( \frac{\partial X}{\partial \xi_1} \cdot \frac{\partial X}{\partial \xi_1} \right)^{1/2} \]

So

\[ \int \left| \frac{\partial X}{\partial \xi_1} \right| = \frac{1}{2} \left( \frac{\partial X}{\partial \xi_1} \cdot \frac{\partial X}{\partial \xi_1} \right)^{-1/2} \cdot \frac{\partial X}{\partial \xi_1} \cdot \frac{2}{\partial \xi_1} \left( \frac{\partial X}{\partial \xi_1} \right) \]

and therefore

\[ \delta E = \int \epsilon' \left( \left| \frac{\partial X}{\partial \xi_1} \right| \right) \left( \frac{\partial X}{\partial \xi_1} \right) \left( \frac{\partial X}{\partial \xi_1} \right) \, d\xi \]

\[ = \int \frac{2}{\partial \xi_1} \left[ \epsilon' \left( \left| \frac{\partial X}{\partial \xi_1} \right| \right) \left( \frac{\partial X}{\partial \xi_1} \right) \right] \, d\xi \]
Let
\[ T = \varepsilon' \left( \frac{\partial X}{\partial \xi_2} \right) = \text{tensin} - \]

\[ T = \frac{\partial X}{\partial \xi_2} \quad \text{unit tangent} \]

and we see that

\[- \frac{\delta E}{\delta X} = \frac{\partial}{\partial \xi_1} (T) \]

In general, the principle of virtual work tells us that the elastic force density \( F \) is given by

\[ F = - \frac{\delta E}{\delta X} \]

so, in the case of a fiber continuum

\[ F = \frac{2}{\xi_1} (T) \]
Let $E$ be any elastic energy function that is invariant under translation.

A translation may be generated by choosing any fixed $V$ independent of $\zeta$, and with

$$\frac{dX}{dt}(\zeta, t) = V$$

Then

$$0 = \frac{d}{dt} E\left[ X(\zeta, t) \right]$$

$$= \int \frac{dE}{dX} (\zeta, t) \cdot \frac{dX}{dt} (\zeta, t) \, d\zeta$$

$$= -\left( \int F(\zeta, t) \, d\zeta \right) \cdot V$$

Since $V$ is arbitrary,

the force $\int F(\zeta, t) \, d\zeta = 0$
Similarly, let $E$ be any elastic energy function that is invariant under rotation.

A rotation may be generated by choosing any fixed $\Omega$, independent of $g$, and letting

$$\frac{dX}{dt}(\alpha, t) = \Omega \times X(\alpha, t)$$

Then,

$$0 = \frac{d}{dt} E[X(\alpha, t)] = \int \frac{\partial E}{\partial X}(\alpha, t) \cdot (\Omega \times X(\alpha, t)) \, dg$$

$$= \Omega \cdot \int X(\alpha, t) \times \frac{\partial E}{\partial X}(\alpha, t) \, dg$$

$$= -\Omega \int (X(\alpha, t) \times F(\alpha, t)) \, dg$$

Since $\Omega$ is arbitrary,

$$0 = \text{total torque} = \int X(\alpha, t) \times F(\alpha, t) \, dg$$
It is often convenient to discretize the elastic energy first, and then find the force by differentiation. Thus, we construct

\[ E_N (X_1 \ldots X_N) \sim E [X] \]

Then

\[ \delta E_N = \sum_{k=1}^{N} \frac{\delta E_N}{\delta X_k} \delta X_k \sim \int \frac{\delta E}{\delta X} \delta X \, d\theta \]

where \( \frac{\delta E_N}{\delta X_k} \) denotes the gradient of \( E_N \) with respect to \( X_k \), i.e., the vector with components \( \frac{\partial E_N}{\partial X_{k\alpha}} \), where \( \alpha = 1, 2 \) or \( \alpha = 1, 2, 3 \).

Therefore

\[ \frac{\delta E_N}{\delta X_k} \sim \frac{\delta E}{\delta X} \, d\theta = - F \, d\theta \]

and so we should set

\[ -F_k \Delta \theta = - \frac{\partial E_N}{\partial X_k} \]

(Note that the factor \( \Delta \theta \) is included: \[ - \frac{\partial E_N}{\partial X_k} = (\text{FORCE})_k \])
As in the continuous case, we have the result that if $E(X_1 \ldots X_N)$ is invariant under translations and rotations, the total force and the total torque are zero. The proofs are essentially the same as in the continuous case, but the results involve sums instead of integrals.

Suppose

$$\frac{\partial X_k(t)}{\partial t} = \dot{V}$$

where $\dot{V}$ is independent of $k$. If $E$ is invariant under translations, we have

$$0 = \frac{d}{dt} E(X_1 \ldots X_N) = \left( \sum_{k=1}^{N} \frac{\partial E}{\partial X_k} \right) \cdot \dot{V}$$

Since $\dot{V}$ is arbitrary, it follows that

$$0 = \sum_{k=1}^{N} \frac{\partial E}{\partial X_k}$$

i.e., the total force is zero.
Similarly, if
\[
\frac{\partial x_k}{\partial t} = \Omega \times x_k
\]
where \(\Omega\) is independent of \(k\), and if
\(E\) is invariant under rotations, we have
\[
0 = \frac{d}{dt} E(x_1 \cdots x_N) = \sum_{k=1}^{N} \frac{\partial E}{\partial x_k} \cdot (\Omega \times x_k)
\]
\[
= \Omega \cdot \sum_{k=1}^{N} (x_k \times \frac{\partial E}{\partial x_k})
\]
Since \(\Omega\) is arbitrary
\[
0 = \sum_{k=1}^{N} x_k \times \frac{\partial E}{\partial x_k}
\]
i.e., the total torque is zero.
Remarks:

1) In general, the concept of torque is origin-dependent: the torque may be zero with respect to one origin and non-zero with respect to another. In the case that the elastic energy is invariant with respect to translation and rotation, the above result is true for torque measured with respect to any origin. To check this, note that

\[ \sum_{k=1}^{N} (x_k - z) \times \frac{\partial E}{\partial x_k} \]

\[ = \sum_{k=1}^{N} (x_k \times \frac{\partial E}{\partial x_k}) - z \times \sum_{k=1}^{N} \frac{\partial E}{\partial x_k} \]

The first term is zero by rotational invariance with respect to \( 0 \), and the second term is zero by translational invariance.
2) The function $\phi$ of the immersed boundary method is constructed in such a manner that total force and total torque are preserved during the force-spreading operation:

**Homework:** Prove the following result:

In $d=2$ or $3$ space dimensions:

If

$$f(x) = \sum_{k=1}^{N} F_k \delta_h (x - X_k) \Delta g$$

for all $x \in \mathcal{G}_h$, then

$$\sum_{x \in \mathcal{G}_h} f(x) h^d = \sum_{k=1}^{N} F_k \Delta g$$

$$\sum_{x \in \mathcal{G}_h} x \times f(x) h^d = \sum_{k=1}^{N} (X_k \times F_k) \Delta g$$

where $\times$ denotes the cross product of two vectors, and where $\mathcal{G}_h$ is an infinite Cartesian lattice of mesh width $h$. 
Hint: The properties of $\delta_h^d$ that are needed to prove the above result are

$$\delta_h^d(x) = \frac{1}{h^d} \varphi(\frac{x_1}{h})\varphi(\frac{x_2}{h}) \quad \text{when } d=2$$

$$\delta_h^d(x) = \frac{1}{h^d} \varphi(\frac{x_1}{h})\varphi(\frac{x_2}{h})\varphi(\frac{x_3}{h}) \quad \text{when } d=3$$

where, in both cases

$$\sum_i \varphi(r-i) = 1, \text{ all } r$$

$$\sum_i (r-i)\varphi(r-i) = 0, \text{ all } r$$

As a special case of the above result, we see that when $\int_k \Delta_{q'} = -\partial E/\partial X_k$ and when $E$ is invariant under translations and rotations, then $\sum_{x \in g_h} f(x)h^d = 0$ and $\sum_{x \in g_h} x \cdot x \cdot f(x)h^d = 0$. 
We now consider some examples of elastic energy functions $E(X_1 \ldots X_N)$ that are useful in the representation of immersed elastic boundaries.

Where possible, we write $E(X_1 \ldots X_N)$ as the discretization of some functional. This makes it clear how to refine parameters as the mesh width is refined.

1) Elastic fiber (no bending rigidity)

$$E(X_1 \ldots X_N) = \frac{K_s}{2} \sum_{k=1}^{N-1} \left( \frac{|X_{k+1} - X_k|}{\Delta \xi} - 1 \right) \Delta \xi$$

$$\frac{\partial E}{\partial X_{k\alpha}} = K_s \sum_{k=1}^{N-1} \left( \frac{|X_{k+1} - X_k|}{\Delta \xi} - 1 \right) \frac{1}{2 \Delta \xi} |X_{k+1} - X_k|^{-1}$$

$$2 \left( X_{k+1\alpha} - X_{k\alpha} \right) \left( \delta_{k+1\mu} - \delta_{k\mu} \right) \Delta \xi$$

$$-\frac{\partial E}{\partial X_{k\alpha}} = +K_s \sum_{k=1}^{N-1} \left( \frac{|X_{k+1} - X_k|}{\Delta \xi} - 1 \right) \frac{|X_{k+1} - X_k|}{|X_k - X_k|} (\delta_{k\mu} - \delta_{k+1\mu})$$
Let
\[ T_{k+\frac{1}{2}} = K_k \sum_{k=1}^{N-1} \left( \frac{1}{X_{k+1} - X_k} - 1 \right) \]

\[ \frac{T_k}{X_{k+1} - X_k} = \frac{X_k - X_{k+1}}{X_{k+1} - X_k} \]

Then
\[ \oint \Delta g = \sum_{k=1}^{N-1} T_{k+\frac{1}{2}} \frac{T_k}{X_{k+1} - X_k} (\delta_{k,k+1} - \delta_{k,k+1/2}) \]

We can collapse the sums if we are careful about the ends of the fiber.

\[ F_l = \left( l \neq N \right) T_{l+\frac{1}{2}} \frac{T_l}{X_{l+1} - X_{l+\frac{1}{2}}} - T_{l-\frac{1}{2}} \frac{T_l}{X_{l-\frac{1}{2}} - X_{l}} (l \neq 1) \]

\[ l = 1 \cdots N, \text{ where a condition like } (l \neq N) \]

evaluates to 1 when it is true and 0 when false.
The collapsed form looks like a discretization of
\[ \frac{\partial}{\partial y} (T_1) \]

which indeed it is, but because we have derived the time from a discretized energy function, we also get a specific recipe of what to do at the ends of the fiber.

We can program the above without special cases at the ends. The expression we need to evaluate is of the form

\[ B_j = \sum_{k=1}^{N-1} A_{k+\frac{1}{2}} \left( \delta_{k,j} - \delta_{k+1,j} \right) \]

for \( j = 1 \ldots N \). Note that \( B \) contains \( N \) elements, whereas \( A \) contains only \( N-1 \). The result can be obtained as follows:

\[ B = [A, 0] - [0, A] \]

where

\[ A = \begin{bmatrix} A_{\frac{1}{2}} & \cdots & A_{N-\frac{1}{2}} \end{bmatrix} \]
a) Elastic fiber with bending rigidity

\[ E(X_1 \ldots X_N) = \frac{K_b}{2} \sum_{k=2}^{N-1} |C_k|^2 \Delta \theta \]

where

\[ C_k = \frac{X_{k+1} - 2X_k + X_{k-1}}{(\Delta \theta)^2}, \quad k = 2, \ldots, N-1 \]

Then

\[ F_k \Delta \theta = -\frac{\partial E}{\partial X_k} = -K_b \sum_{k=2}^{N-1} C_{k\alpha} \frac{\Delta \theta}{(\Delta \theta)^2} \]

\[ F_1 = -K_b \left( \sum_{l=3}^{N} C_{l-1} - (2 \leq l \leq N-1) 2C_l + (l \leq N-2) C_{l+1} \right) \frac{\Delta \theta}{(\Delta \theta)^2} \]

which is a discretization of \( F = -K_b \frac{\partial^4 X}{\partial \theta^4} \), except

that a specific recipe is included on what to do at the ends of the fiber. For each component of \( C, F \), the above is neatly programmed as

\[ C = (X(3:N) - 2*X(1:N-1)) + X(1:N-2)) \text{/} \Delta \theta 12 \]

\[ F = ([0,0,C] - 2*[0,C,0] + [C,0,0]) \text{/} \Delta \theta 12 \]
Remark: The elastic energy we have just defined is zero whenever the points
\[ X_1, \ldots, X_N \]
are equally spaced along a straight line, but there is nothing in that energy
term to control the spacing, i.e., to resist stretch or compression. For this
reason it is often combined with the
energy term of Example 1, above, to
produce a fiber which resists stretch,
compression and bending:

\[
E(X_1, \ldots, X_N) = \frac{K_s}{2} \sum_{k=1}^{N-1} \left( \frac{|X_{k+1} - X_k|}{\Delta g} - 1 \right)^2 \Delta g
\]

\[
+ \frac{K_b}{2} \sum_{k=2}^{N-1} \left( \frac{|X_{k+1} + X_{k-1} - 2X_k|}{(\Delta g)^2} \right)^2 \Delta g
\]

The stretching term keeps \(|X_{k+1} - X_k|\) close
to \(\Delta g\), while the bending term resists curvature.
3) Finite element representation of a general 3D elastic medium

\[ X_\alpha - X_\alpha^0 = \sum_{\beta=1}^{3} A_{\alpha\beta} (Z_\beta - Z_\beta^0) \]

where the \( A_{\alpha\beta} \) are determined by

\[ X_\alpha^1 - X_\alpha^0 = \sum_{\beta=1}^{3} A_{\alpha\beta} (Z_\beta^1 - Z_\beta^0) \]

\[ X_\alpha^2 - X_\alpha^0 = \sum_{\beta=1}^{3} A_{\alpha\beta} (Z_\beta^2 - Z_\beta^0) \]

\[ X_\alpha^3 - X_\alpha^0 = \sum_{\beta=1}^{3} A_{\alpha\beta} (Z_\beta^3 - Z_\beta^0) \]
For each \( \alpha = 1, 2, \alpha 3 \), we have 3 equations in the 3 unknowns \( A_\alpha 1, A_\alpha 2, A_\alpha 3 \).

Let \( A_\alpha \) be the vector with these unknowns as its components. The three equations may then be written

\[
X_\alpha ^1 - X_\alpha ^0 = A_\alpha \cdot (z^1 - z^0)
\]

\[
X_\alpha ^2 - X_\alpha ^0 = A_\alpha \cdot (z^2 - z^0)
\]

\[
X_\alpha ^3 - X_\alpha ^0 = A_\alpha \cdot (z^3 - z^0)
\]

Look for a solution in the form

\[
A_\alpha = \lambda _1 (z^2 - z^0) \times (z^3 - z^0) + \lambda _2 (z^3 - z^0) \times (z^1 - z^0) + \lambda _3 (z^1 - z^0) \times (z^2 - z^0)
\]
Then
\[ X_\alpha^1 - X_\alpha^0 = \lambda_1 6V_z \]
\[ X_\alpha^2 - X_\alpha^0 = \lambda_2 6V_z \]
\[ X_\alpha^3 - X_\alpha^0 = \lambda_3 6V_z \]
where
\[ V_z = \frac{1}{6} \left( (\bar{z}^1 - \bar{z}^0) \times (\bar{z}^2 - \bar{z}^0) \right) \cdot (\bar{z}^3 - \bar{z}^0) \]

= Volume of tetrahedron in its reference configuration.

Thus
\[ A_\alpha = \frac{1}{6V_z} \left[ (X_\alpha^1 - X_\alpha^0)(\bar{z}^2 - \bar{z}^0) \times (\bar{z}^3 - \bar{z}^0) \right. \]
\[ + (X_\alpha^2 - X_\alpha^0)(\bar{z}^3 - \bar{z}^0) \times (\bar{z}^1 - \bar{z}^0) \]
\[ + (X_\alpha^3 - X_\alpha^0)(\bar{z}^1 - \bar{z}^0) \times (\bar{z}^2 - \bar{z}^0) \]
Let the elastic energy density of the material be any function of the $A_{\alpha \beta}$, where "density" means per unit volume in the reference configuration.

Since the $A_{\alpha \beta}$ are constant on the tetrahedron, the total energy of the tetrahedron is simply

$$E = V \left( \cdots A_{\alpha \beta}, \cdots \right).$$

For any particular $\alpha$, $X_0^\alpha$ appears only in

$$A_{\alpha 1}, A_{\alpha 2}, A_{\alpha 3}.$$

Therefore

$$\frac{\partial E}{\partial X_0^\alpha} = V \sum_{\beta=1}^{3} \frac{\partial E}{\partial A_{\alpha \beta}} \frac{\partial A_{\alpha \beta}}{\partial X_0^\alpha}$$

$$= V \frac{\partial E}{\partial A_\alpha} \cdot \frac{\partial A_\alpha}{\partial X_0^\alpha}$$

where $\partial E/\partial A_\alpha$ is the vector with components

$$\frac{\partial E}{\partial A_{\alpha 1}}, \frac{\partial E}{\partial A_{\alpha 2}}, \frac{\partial E}{\partial A_{\alpha 3}}.$$
From our previous formula for $\frac{\partial A_\alpha}{\partial x_\alpha}$, it is easy to evaluate

$$\frac{\partial A_\alpha}{\partial x_\alpha} = -\frac{1}{6V_2} \left[ \left( \frac{Z^2-Z^0}{2} \right) \times \left( Z^3-Z^0 \right) \right. \\
+ \left( \frac{Z^3-Z^0}{2} \right) \times \left( Z^1-Z^0 \right) \\
\left. + \left( \frac{Z^1-Z^0}{2} \right) \times \left( Z^2-Z^0 \right) \right]$$

$$= -\frac{1}{6V_2} \left[ \left( \frac{Z^1 \times Z^2}{2} \right) + \left( \frac{Z^2 \times Z^3}{2} \right) + \left( \frac{Z^3 \times Z^1}{2} \right) \right]$$

Therefore

$$-\frac{\partial E}{\partial x_\alpha} = \frac{1}{6} \frac{\partial E}{\partial A_\alpha} \left[ \left( \frac{Z^1 \times Z^2}{2} \right) + \left( \frac{Z^2 \times Z^3}{2} \right) + \left( \frac{Z^3 \times Z^1}{2} \right) \right]$$

With multiple tetrahedra, the force on any node is the sum of the contributions from all tetrahedra that touch that node.
Example: Linear Elasticity

Strain: $\varepsilon_{\mu\nu} = \frac{1}{2} (A_{\mu\nu} + A_{\nu\mu}) - \delta_{\mu\nu}$

Elasticity tensor: $C_{\chi\lambda\mu\nu}$

$\mathcal{E} = \frac{1}{2} C_{\chi\lambda\mu\nu} \varepsilon_{\chi\lambda} \varepsilon_{\mu\nu}$ (summed over $\chi, \lambda$)

$$\frac{\partial \mathcal{E}}{\partial A_{\alpha\beta}} = \frac{1}{2} C_{\chi\lambda\mu\nu} \left( 2 \frac{\partial \varepsilon_{\chi\lambda}}{\partial A_{\alpha\beta}} \varepsilon_{\mu\nu} + \frac{1}{2} \left( \delta_{\chi\alpha} \delta_{\lambda\beta} + \delta_{\chi\beta} \delta_{\lambda\alpha} \right) \varepsilon_{\mu\nu} \right)$$

$$= \frac{1}{2} \left( C_{\chi\beta\mu\nu} + C_{\beta\chi\mu\nu} \right) \varepsilon_{\mu\nu}$$

$$= C_{\alpha\beta\mu\nu} \varepsilon_{\mu\nu}$$
1. Arbitrary network of possibly nonlinear springs

\[ k_1(l), k_2(l) = \text{indices of the points joined by link } l \]
\[ (l_0) \_l = \text{reference length (usually rest length) of link } l \]

\[ E = \frac{1}{N_{\text{links}}} \sum_{l=1}^{N_{\text{links}}} E \left( \frac{1}{k_1(l)} - \frac{1}{k_2(l)} \right) \]

\[ = \text{elastic energy of network} \]

**Note:** The factor \( N_{\text{links}}^{-1} \) is included so that \( E \) will converge to some continuum limit as network is refined.
\[
\frac{F_k}{N_{\text{points}}} = - \frac{\partial E}{\partial x_k} = \\
- \frac{1}{N_{\text{links}}} \sum_{l=1}^{N_{\text{links}}} \varepsilon_l \left( \frac{X(k_2(l) - k_1(l))}{(L_0)^l} \right) \frac{X(k_2(l)) - X_{k_1(l)}}{|X(k_2(l)) - X_{k_1(l)}|} \delta_{k_2(l), k_1(l)} - \delta_{k_2(l), k_1(1)} \\
\]

\[
F_k = \frac{N_{\text{points}}}{N_{\text{links}}} \left( \sum_{l: k_1(l) = k} - \sum_{l: k_2(l) = k} \right) \frac{T_l}{(L_0)^l} \\
\]

where

\[
T_l = \varepsilon_l \left( \frac{X(k_2(l)) - X_{k_1(l)}}{(L_0)^l} \right) \\
\]

\[
\tilde{T}_l = \frac{X(k_2(l)) - X_{k_1(l)}}{|X(k_2(l)) - X_{k_1(l)}|} \\
\]

where \( N_{\text{points}} = \# \text{ of nodes of the network} \).
Matlab code for force density of an arbitrary network of springs

\[ D = X(k2, :) - X(k1, :) \]

\[ D^2 = \sqrt{D(1, 1)^2 + D(2, 2)^2 + D(3, 3)^2} \]

\[ G = (Npoints / Nlinks) \times (T(DD ./ LO) ./ LO) \]

\[ F = zeros(Npoints, 3) \]

for \( l = 1 : Nlinks \)

\[ G = (G(1, :) / DD(l)) \times D(l, :) \]

\[ F(k1(l), :) = F(k1(l), :) + G \]

\[ F(k2(l), :) = F(k2(l), :) - G \]

end
Important Remark

One might think that the loop "for l = 1: Nlinks" could be replaced by the following:

\[ G = (\frac{GG}{DD} \times [1, 1, 1]) \times D \]

\[ F(k1,:) = F(k1,:) + G \]

\[ F(k2,:) = F(k2,:) - G \]

Although this computation of \( G \) is perfectly fine, the lines here that attempt to update \( F \) will give INCORRECT RESULTS, since each of the arrays \( k1 \) and \( k2 \) will, in general, contain many repeat values. For any particular \( k \), there may be several \( l \) such that \( k1(l) = k \), and several \( l \) such that \( k2(l) = k \). This causes no difficulty on the right-hand side of the assignment statements, but on the left-hand side it amounts to an instruction that several different values be stored in the same memory location. In fact, the last value to be written there will prevail, but what we actually want is the sum of the different values to be recorded.
As a simple test to illustrate what happens in MATLAB, note that

\[ A([1, 1]) = [5, 7] \] yields \( A(1) = 7 \)

\[ A([1, 1]) = [7, 5] \] yields \( A(1) = 5 \)

And what we actually want is neither of the above, but instead \( A(1) = 12 \).

This is not just a MATLAB quirk; it is a fundamental issue in parallel computing. Special hardware or software precautions are needed when several processes that are supposed to execute in parallel need to update the same variable.
Example: Nearly rigid body in fluid

By choosing enough links of sufficient stiffness, we can make an essentially rigid network of springs to simulate immersed rigid bodies.

Example: Bodies that swim by undergoing prescribed deformation. (L. Fauci)

Let a deformation be prescribed as follows:

\[ \mathbf{3} \rightarrow \mathbf{N}(\mathbf{3}, t) \]

Choose sample points \( \mathbf{\frac{3}{k}}, k = 1, \ldots, N \) points.

Define \( (x_0)(t) = \mathbf{N}(\mathbf{\frac{3}{k_2}(t)}, t) - \mathbf{N}(\mathbf{\frac{3}{k_1}(t)}, t) \)

Now the energy function \( E \) defined above will tend to make the points \( Y_1 \ldots Y_N \) points follow the same deformation. Note however, that all information about absolute displacement or absolute orientation that may have been encoded in \( \mathbf{N}(\mathbf{3}, t) \) is discarded, as it should be.
5) Target Points

Sometimes, we want to tie a structure down to fixed points in space, e.g., to study flow past a body, as in traditional fluid mechanics. This can be done in an extremely simple way:

$$E = \frac{K}{2} \sum_{k=1}^{N} |X_k - Z_k|^2 (\Delta Z)_k$$

where the $Z_k$ are the fixed, given locations that we call target points, and where the $X_k$ are the usual immersed boundary points that interact with the fluid. The numbers $(\Delta Z)_k$ represent the measures that we associate with the target points. They are only provided so that the above expression approximates an integral, which may be a volume, area, or line integral, depending on its application. As usual, we derive

$$F_k(\Delta Z)_k = -\frac{\partial E}{\partial X_k} = -K(X_k - Z_k) \Delta Z_k$$

$$\mathbf{F}_k = -K(\mathbf{X}_k - \mathbf{Z}_k)$$
The force described by $F_k$ provides a feedback mechanism that keeps $X_k$ close to $Z_k$.

Note that the above energy function $E(X_1 \cdots X_N)$ is not invariant under translations and rotations, since the $Z_k$ are fixed. Therefore, we typically have

$$\sum_{k=1}^{N} F_k(\Delta Z_k) \neq 0$$

Recall that these quantities are preserved during structure-fluid interaction by the immersed boundary method. In flow past a body, they are important output quantities, since they are equal to minus the total force (which may be further resolved into lift and drag) and minus the total torque that are applied by the fluid to the body. Note that these quantities are being evaluated without any consideration of the fluid stress tensor at the surface of the body.
In most applications target points, the constant $K$ has no physical meaning and we are interested in the limit $K \to \infty$. In this limit $X_k \to Z_k$, and \( \overline{F}_k \to \) the force that is needed to hold the immersed body or boundary in place. This \( K \) plays the role of a penalty parameter.

Note the ease of representing arbitrary geometry with target points. All we have to do is lay out the points so they model the geometry. Since the points all act individually, there is no topological structure at all—just an array of points that may be numbered arbitrarily.

In fact, there is no need for the target points to stay at fixed locations. To model bodies that move in a prescribed manner, just replace \( Z_k \) in the foregoing by \( Z_k(t) \), which are given functions of time.
As a further generalization, we consider target points which carry additional mass (in excess of the mass that is present anyway because there is fluid everywhere in an immersed boundary computation).

Instead of moving in a prescribed manner, these target points move according to Newton’s equation of motion (including the effect of gravity).

Let \( M_k(\Delta Z)_k \) be the additional mass associated with the target point whose index is \( k \), so that \( M_k \) is the density of the additional mass with respect to the measure \( (\Delta Z)_k \).

The equation of motion of the target point is

\[
M_k(\Delta Z)_k \frac{d^2 Z_k}{dt^2} = +K(X_k - Z_k)(\Delta Z)_k - g M_k(\Delta Z)_k \hat{e}_3
\]

where \( \hat{e}_3 \) is a vector pointing “up”, i.e., against gravity.
In this way, we get the penalty immersed boundary (PIB) method for immersed boundaries with mass (Yang, Sam, Kim), the spatially discretized formulation of which is as follows:

\[
\rho \left( \frac{du}{dt} + S(u)u \right) + Dp = mL + f - \rho g e_3
\]

\[
D \cdot u = 0
\]

\[
f(x,t) = \sum_{k=1}^{N} \left( -K(x_k - z_k)(\Delta z)_k - \frac{2E_0(x_1 \ldots x_N)}{\partial x_k} \delta_h(x - x_k(t)) \right)
\]

\[
\frac{dx_k}{dt} = \sum_{x \in \mathcal{O}_h} u(x,t) \delta_h(x - x_k(t)) h
\]

\[
M_k \frac{d^2 z_k}{dt^2} = K(x_k - z_k) - M_k g e_3
\]

where \(E_0(x_1 \ldots x_N)\) describes an elastic interaction unrelated to the penalty spring.
If we formally take the limit $K \to \infty$ in the above formulation of the pT-B method, we get:

$$Z_k \to X_k$$

$$K(X_k - Z_k) \to M_k \left( \frac{d^2 X_k}{dt^2} + g e_3 \right)$$

and hence the limiting system:

$$\rho \left( \frac{du}{dt} + S(g) u \right) + \nabla p = \mu L u + f - \rho g e_3$$

$$\nabla \cdot u = 0$$

$$f(x,t) = \sum_{k=1}^{N} \left( -M_k (\Delta Z_k) \left( \frac{d^2 X_k}{dt^2} + g e_3 \right) - \frac{\partial E_0(X_1 \ldots X_N)}{\partial X_k} \right) \delta_h (x - X_k(t))$$

$$\frac{dX_k}{dt} = \sum_{x \in \delta_h} u(x,t) \delta_h (x - X_k(t)) \frac{d}{dx}$$
In comparison to the above limiting equations, the advantage of the pIB method is that it avoids the explicit computation of the D'Alembert force

\[ -M_k (\Delta z)_k \frac{d^2 X_k}{dt^2} \]

which can lead to stability problems once time has been discretized as well.

As a special case of the pIB method, note that we may completely drop the elastic interaction given by \( E_0 (X_1 \cdots X_N) \), and then we get a method for adding mass to a fluid for the purpose of doing two-fluid, multifluid, or stratified fluid problems.