

Algorithm 8.2 *General registration algorithm.*

Initialize $k = 0$, $\vec{X}^{(k)}$, and $\vec{U}^{(k)} = 0$.
 For $k = 0, 1, 2, \dots$
 compute force $\vec{F}^{(k)} = f(\vec{X}, \vec{U}^{(k)})$;
 solve partial differential equation, $A\vec{U}^{(k+1)} = \vec{F}^{(k)}$;
 if converged, stop, end;
 end.

has either to embed the domain in a cube or to weight the coordinates reciprocal to the resolution. A detailed description is technical and omitted here.

We assume that a finite difference approximation of $A[u]$ at any grid point x_j can be obtained using a convolution filter S^A ,

$$A[g](x_j) \approx \sum_{\kappa \in \mathcal{N}(j) \cup j} S^A_{\kappa} g(x_{\kappa}) =: (S^A * g)(x_j)$$

where $\mathcal{N}(j)$ denotes a neighborhood of the grid point j and S^A denotes a filter connected with the partial differential operator A via its finite difference approximation. Of course, the particular boundary conditions attached to Problem 8.1 have to be incorporated into the discrete formulation. Practically, this can be done easily by a zero padding (Dirichlet), mirroring (Neumann), or adequate copying (periodic).

With the lexicographical ordering (cf., Definition 3.2), we can also deduce a matrix representation of the convolution with S^A ,

$$A \cdot \vec{g} := (S^A * g)(\vec{X}).$$

Using for example the fixed-point type-iteration (8.6), we end up with an overall algorithm as summarized in Algorithm 8.2.

ELASTIC REGISTRATION

Broit (1981) was the first to study what he called *elastic registration*, a method based on a quantitative measure for the deformation, a quantitative measure for similarity between deformed images, and a procedure that uses the measures for obtaining an optimal mapping.

Broit's approach is based on

$$\mathcal{D}[R, T; u] = \mathcal{D}^{\text{SSD}}[R, T; u] = \frac{1}{2} \|T_u - R\|_{L_2(\Omega)} \quad \text{and} \quad \mathcal{S}[u] := \mathcal{P}[u]$$

where $\mathcal{P}$ denotes the *linearized elastic potential* of the displacement u ,

$$\mathcal{P}[u] = \int_{\Omega} \frac{\mu}{4} \sum_{j,k=1}^d (\partial_{x_j} u_k + \partial_{x_k} u_j)^2 + \frac{\lambda}{2} (\text{div } u)^2 dx. \quad (9.1)$$

Here, λ and μ denote the so-called Lamé constants. For this particular choice of regularizer, the Euler-Lagrange equations are nothing more than the Navier-Lamé equations.

Navier-Lamé equations:

$$f = \mu \Delta u + (\lambda + \mu) \nabla \text{div } u. \quad (9.2)$$

In the next section we give a physical motivation for this particular regularizer. The images are viewed as two different observations of an elastic body, one before and one after a *deformation*. The *displacement* of the elastic body is derived using a linear elasticity model. This development follows Broit (1981). In order to introduce different boundary conditions and for later reference, the eigenfunctions and eigenvalues of the Navier-Lamé operator are computed in Section 9.2. In Section 9.3 we show how the Navier-Lamé equations can be derived from the elastic potential using the general framework introduced in Chapter 8. This also provides a weak form of the Navier-Lamé equations.

Based on the general remarks on discretization (see Section 8.4) we derive a finite difference approximation of eqn (9.2). For ease of presentation, we focus on dimensions $d = 2, 3$. In Section 9.5 it is shown that a general class of matrices, including the discrete Navier-Lamé operator, can be diagonalized using fast Fourier transformation (FFT) techniques. The Moore-Penrose pseudo-inverse

of the discrete Navier-Lamé operator is explicitly given for dimension $d = 2$ and traced back to the solution of small 3-by-3 linear systems for dimension $d = 3$. Finally, registration results for academic as well as real-life problems are presented.

9.1 Physical motivation

In this section we provide insight into the physical properties of an elastic body. The essential difference between a rigid and an elastic body is that the relation of particles is no longer fixed but can vary with respect to the elastic properties of the body; see Fig. 9.1.

Applying an external force to an elastic body results in a *deformation* or *strain* of the body. The strain is related to *tension* or *stress* of the body, and the shape of the body results from an equilibrium of outer forces and inner *stress*.

9.1.1 Deformation of a body

The deformation of a body or tissue is described by the time-dependent *transformation* $\varphi : \mathbb{R}^d \times \mathbb{R} \rightarrow \mathbb{R}^d$. Here, $\varphi(P, t)$ denotes the spatial position of a particle P at time t . For any fixed particle P , $x = \varphi(P, 0)$ gives its original position, $\tilde{x} = \varphi(P, t_E)$ gives its position after deformation, and $\varphi(P, t)$, $0 \leq t \leq t_E$, gives the path of the particle P during deformation. For simplicity of notation, we also write $\varphi(x, t)$ for the position of the particle P at time t , where x is given by $x := \varphi(P, 0)$. Note that we use the Lagrange coordinates to describe the physical behavior.

To begin with and for comparison reasons, we describe the transformation properties of a rigid body. Here, the distances and angles between particles remain fixed during transformation. Thus, the transformation is characterized by

$$\varphi(x, t_E) = Q \cdot \varphi(x, 0) + b \quad \text{for all } x \in \mathbb{R}^d,$$



FIG. 9.1 LEFT: Image of a hand, MIDDLE: affine linear deformation of the hand, and RIGHT: elastic deformation of the hand.

where $b \in \mathbb{R}^d$ denotes a translation and $Q \in \mathbb{R}^{d \times d}$ denotes a rotation, i.e., $Q^T Q = I_d$ and $\det Q = 1$. The distance $r := y - x$ of two particles x and y is mapped to $\tilde{r} := \tilde{y} - \tilde{x} = Q \cdot r$.

9.1.2 The strain tensor

For an elastic body, however, the transformation does not need to be rigid. For example, the distance between two particles can be expanded; see Fig. 9.2. Since we are interested in a linear elasticity model, only first order terms in $r = x - y$ will be considered. Thus, to a first order approximation, we have

$$\tilde{r} = \tilde{y} - \tilde{x} = \varphi(y, t_E) - \varphi(x, t_E) \approx \nabla \varphi(x, t_E)(y - x) = \nabla \varphi(x, t_E)r,$$

where

$$\nabla \varphi(x, t) := \begin{pmatrix} \partial_{x_1} \varphi_1 & \cdots & \partial_{x_d} \varphi_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} \varphi_d & \cdots & \partial_{x_d} \varphi_d \end{pmatrix} \in \mathbb{R}^{d \times d}.$$

Introducing the *displacement* $u : \mathbb{R}^d \times \mathbb{R} \rightarrow \mathbb{R}^d$,

$$u(x, t) := \varphi(x, t) - x,$$

we find that $\nabla \varphi(x, t) = I_d + \nabla u(x, t)$.

Remark 9.1 Typically one is interested in a diffeomorphic transformation, i.e., the transformation is not allowed to introduce cracks and/or foldings of the tissue. Thus, it is worthwhile noticing that if $|\partial_{x_j} u_k| \ll 1$, we also have

$$\det(\nabla \varphi(x, t)) \approx 1 + \operatorname{div} u \neq 0$$

and hence φ is diffeomorphic.

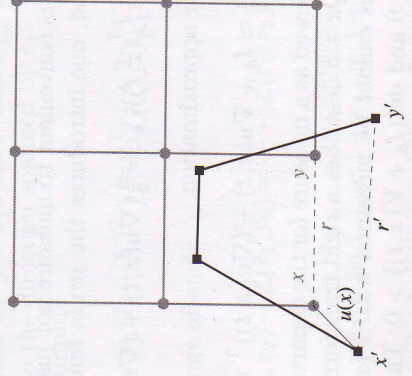


FIG. 9.2 Elastic body, particles before (dots) and after (squares) deformation.

Remark 9.2 For later purposes, we comment here on the material derivative. To this end, let P again be a fixed particle, $\varphi(P, t)$ the path followed by the particle P , and $x = \varphi(P, 0)$ its initial position. Hence, $\varphi(P, t) = x + u(P, t)$ by definition of the displacement u . The velocity v of the particle is defined by

$$v(P, t) := \partial_t \varphi(P, t) = \frac{d}{dt} u(P, t),$$

where ∂_t and d/dt denote the partial and total derivative with respect to time, respectively. Note that P does not depend on the time.

In contrast to the particle focussing Lagrange coordinates we consider the position following Euler coordinates; see also Section 3.3.2. The displacement u and the velocity v can also be related to the spatial position,

$$\tilde{u}(\tilde{x}, t) := u(P, t), \quad \tilde{v}(\tilde{x}, t) := v(P, t),$$

where the particle and the position are connected by $\tilde{x} = \varphi(P, t)$.

From this point of view, we have

$$\tilde{v}(\tilde{x}, t) = v(P, t) = \frac{d}{dt} u(P, t) = \frac{d}{dt} \tilde{u}(\tilde{x}, t) = \frac{d}{dt} \tilde{u}(\varphi(P, t), t).$$

The material derivative is given by

$$\begin{aligned} \frac{d}{dt} \tilde{u}(\tilde{x}, t) &= \nabla_x \tilde{u}(\varphi(P, t), t) \partial_t \varphi(P, t) + \partial_t \tilde{u}(\varphi(P, t), t) \\ &= \nabla_x \tilde{u}(\tilde{x}, t) \tilde{v}(\tilde{x}, t) + \partial_t \tilde{u}(\tilde{x}, t), \end{aligned}$$

where $\nabla_x = (\partial_{x_1}, \dots, \partial_{x_d})^\top$, and the chain rule has been applied. Thus, the partial time derivative of $\tilde{u}$ can also be expressed by

$$\partial_t \tilde{u}(\tilde{x}, t) = \tilde{v}(\tilde{x}, t) - \nabla_x \tilde{u}(\tilde{x}, t) \tilde{v}(\tilde{x}, t).$$

In elasticity theory it is convenient to measure not just the displacement but its alterations. To this end, one introduces the *strain tensor*

$$V(x, t) := (\varepsilon_{j,k}(x, t))_{j,k} := \frac{1}{2} (\nabla u(x, t) + (\nabla u(x, t))^\top). \quad (9.3)$$

Note that, to a first order approximation,

$$[\nabla \varphi(x, t)]^\top \nabla \varphi(x, t) \approx I_d + \nabla u(x, t) + (\nabla u(x, t))^\top \approx (I_d + V(x, t))^2.$$

Thus, $V(x, t)$ might be viewed as a measure for the departure of rigidity of $\varphi(x, t)$. If in particular $\varphi(x) = Qx + b$ describes a rigid transformation, $V(x, t) = 0$. On the other hand, rotations cannot be measured with $V(x, t)$. From Remark 9.1, we have $\det(\nabla \varphi(x, t)) > 0$ and $\det(I_d + V(x, t)) > 0$. Thus,

$$Q(x, t) := (I_d + V(x, t))(\nabla \varphi(x, t))^{-1}$$

is a pure rotation. The transformation may be decomposed into rigid and non-rigid parts, where the non-rigid parts are given by

$$Q(x, t) \tilde{r} - r = V(x, t) \cdot r. \quad (9.4)$$

From eqn (9.4) we obtain that $V(x, t)$ describes the relative changes of lengths and angles introduced to the elastic body by the transformation φ . For dimension one this is obvious since $Q(x, t) = 1$ and $V(x, t) = \varepsilon_{1,1} = (\tilde{r} - r)/r$. Our next example illustrates this relation for dimension three.

Example 9.1 (Elastic body in $\mathbb{R}^3$) For the points

$$P_0 = \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix}, \quad P_1 = \begin{pmatrix} x_0 + x \\ y_0 \\ z_0 \end{pmatrix}, \quad \text{and} \quad P_2 = \begin{pmatrix} x_0 \\ y_0 + y \\ z_0 \end{pmatrix},$$

we define $r_1 := P_0 \vec{P}_1 = x e_1$ and $r_2 := P_0 \vec{P}_2 = y e_2$. Considering only first order terms for the length of $\tilde{r}_1 = \varphi(P_1, t_E) - \varphi(P_0, t_E)$, we obtain

$$\begin{aligned} \|\tilde{r}_1\|_{\mathbb{R}^3}^2 &\approx \|\nabla \varphi \cdot r_1\|_{\mathbb{R}^3}^2 \\ &\approx r_1^\top (I_d + V)^\top r_1 \\ &\approx r_1^\top r_1 + 2r_1^\top V r_1 = x^2(1 + 2\varepsilon_{1,1}). \end{aligned}$$

Hence, utilizing a Taylor expansion, we get

$$\frac{\|\tilde{r}_1\|_{\mathbb{R}^3} - \|r_1\|_{\mathbb{R}^3}}{\|r_1\|_{\mathbb{R}^3}} \approx \sqrt{1 + 2\varepsilon_{1,1}} - 1 \approx \varepsilon_{1,1}.$$

Analogous considerations for $j = 2, 3$ show that $\varepsilon_{j,j}$ are the relative changes of lengths with respect to the direction spanned by e_j .

Computing for example

$$(\tilde{r}_1)^\top \tilde{r}_2 = r_1 (I_d + V)^\top r_2 \approx r_1^\top r_2 + 2r_1^\top V r_2 = 2xy\varepsilon_{1,2},$$

we obtain for the change of angle

$$\begin{aligned} \delta &= \angle(r_1, r_2) - \angle(\tilde{r}_1, \tilde{r}_2) = \frac{\pi}{2} - \angle(\tilde{r}_1, \tilde{r}_2) \\ &\approx \sin \delta = \frac{(\tilde{r}_1)^\top \tilde{r}_2}{\|\tilde{r}_1\| \|\tilde{r}_2\|} \approx \frac{2xy\varepsilon_{12}}{x(1 + \varepsilon_{1,1})y(1 + \varepsilon_{2,2})} \approx 2\varepsilon_{12}, \end{aligned}$$

showing that $\varepsilon_{j,k}$ measures the relative change of angles of the non-rigid body.

In other words, we can express the boundary integral on the left hand side as an integral over the body. The stresses $\sigma_{j,k}$ are collected in the so-called *stress tensor* Σ ,

$$\Sigma(x, t) := (\sigma_{j,k})_{j,k=1,\dots,d} \in \mathbb{R}^{d \times d}. \quad (9.7)$$

Physical considerations show that the stress tensor is symmetric, i.e., $\Sigma = \Sigma^T$; cf., Example 9.2.

Starting with the equations of equilibrium and varying over all possible sub-bodies, we rediscover Newton's first law, the *balance of forces*,

$$f = -\operatorname{div} \Sigma,$$

and Newton's second law, the *balance of momentum*,

$$x \times f = -x \times \operatorname{div} \Sigma.$$

The next example motivates that for dimension three, the *stress tensor* is symmetric. To this end, we consider an arbitrary tetrahedron. Note that a general body might be covered by a set of tetrahedra.

Example 9.2 As a reference volume we consider a tetrahedron in $\mathbb{R}^3$ with edges

$$P_0 = \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix}, \quad P_1 = \begin{pmatrix} x_0 + x \\ y_0 \\ z_0 \end{pmatrix}, \quad P_2 = \begin{pmatrix} x_0 \\ y_0 + y \\ z_0 \end{pmatrix}, \quad \text{and} \quad P_3 = \begin{pmatrix} x_0 \\ y_0 \\ z_0 + z \end{pmatrix};$$

cf., Fig. 9.4. The boundary surfaces are denoted by F_1, F_2, F_3 , and F_4 , where e_j is normal to F_j , $j = 1, 2, 3$, and $\vec{n} = (n_1, n_2, n_3)^T$ is normal to F_4 . Using

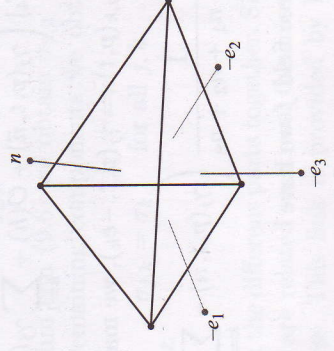


FIG. 9.4 Tetrahedron with outer normals $-e_1, -e_2, -e_3$, and $\vec{n}$.

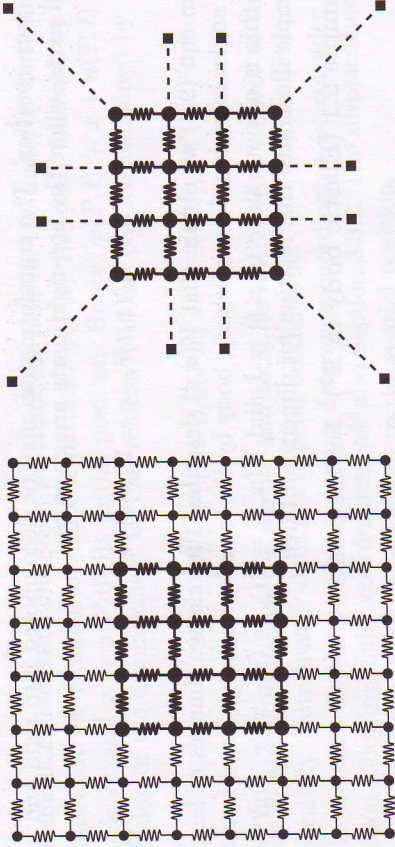


FIG. 9.3 LEFT: Two-dimensional elastic body with springs and part of the body (particles marked with dots); RIGHT: isolated part with strain at the boundary.

9.1.3 The stress tensor

The connectivity of an elastic body can be described in terms of *stress*. In Fig. 9.3, the particles of an elastic body idealized as regular grid points are shown. These particles are connected by small springs. The energy needed to maintain the shape of a particular part of the elastic body when isolated from the rest is expressed by the stress γ , where $\gamma : \mathbb{R}^d \times \mathbb{R} \times \mathbb{R}^d \rightarrow \mathbb{R}^d$. The stress might be viewed as a force field acting on the boundary $\partial\Omega$ of any part Ω of the body. The equations of equilibrium for the elastic body Ω are (cf., e.g., Gurtin (1983))

$$\int_{\Omega} f dx + \int_{\partial\Omega} \gamma dx = 0, \quad (9.5)$$

$$\int_{\Omega} x \times f dx + \int_{\partial\Omega} x \times \gamma dx = 0, \quad (9.6)$$

where f denotes an external force field and $\times$ denotes the vector cross-product.

For the elastic body under consideration we assume that there exist functions $\sigma_{j,k} : \mathbb{R}^d \times \mathbb{R} \rightarrow \mathbb{R}^d$, $j, k = 1, \dots, d$, such that

$$\int_{\partial\Omega} \gamma_j dx = \int_{\partial\Omega} \sum_{k=1}^d \sigma_{j,k} n_k dx =: \int_{\partial\Omega} \langle \sigma_j, \vec{n} \rangle dx,$$

where $\vec{n} = (n_1, \dots, n_d)$ denotes the outer normal on $\partial\Omega$. Using the Gaussian integral or divergence theorem (cf., e.g., Gurtin (1981, §2(5))), we obtain

$$\int_{\partial\Omega} \gamma_j dx = \int_{\Omega} \operatorname{div} \sigma_j, dx, \quad j = 1, \dots, d.$$

elementary calculus for the cross-product $\times$, we obtain for the areas $|F_j|$,

$$\begin{aligned} 2|F_4|\vec{n} &= P_1\vec{P}_2 \times P_1\vec{P}_3 \\ &= (P_0\vec{P}_2 - P_0\vec{P}_1) \times (P_0\vec{P}_3 - P_0\vec{P}_1) \\ &= P_0\vec{P}_2 \times P_0\vec{P}_3 - P_0\vec{P}_1 \times P_0\vec{P}_3 - P_0\vec{P}_2 \times P_0\vec{P}_1 \\ &= 2|F_1|e_1 + 2|F_2|e_2 + 2|F_3|e_3 = 2(|F_1|, |F_2|, |F_3|)^\top. \end{aligned}$$

For fixed numbers $a, b > 0$ and arbitrary $h > 0$ we obtain a family of congruent tetrahedra with $x = h$, $y = ah$, $z = bh$ having the same outer normals. For an arbitrarily chosen point $Q \in F_4$, we have $0\vec{Q} = 0\vec{P}_0 + P_0\vec{Q} = p + q$, where $p = 0\vec{P}_0$ and $q = P_0\vec{Q}$. Since $Q \in F_4$, $\|q\|_{\mathbb{R}^3} \leq h^2(1 + a^2 + b^2)$, and with a $\theta \in [0, 1]$ we get for the j^{th} component of the stress γ_j ,

$$\begin{aligned} \gamma_j(Q, t, \vec{n}) &= \gamma_j(p + q, t, \vec{n}) \\ &= \gamma_j(p, t, \vec{n}) + \nabla \gamma_j(p + \theta q, t, \vec{n})^\top q \\ &= \gamma_j(p, t, \vec{n}) + \mathcal{O}(h). \end{aligned}$$

For the tetrahedron T with boundary ∂T , we thus find

$$\begin{aligned} \int_{\partial T} \gamma_j dx &= \int_{\partial T} \gamma_j(x, t, \vec{n}(x)) dx = \sum_{k=1}^4 \int_{F_k} \gamma_j(x, t, \vec{n}(x)) dx \\ &= \int_{F_4} \gamma_j(x, t, \vec{n}) dx + \sum_{k=1}^3 \int_{F_k} \gamma_j(x, t, -e_k) dx \\ &\approx \gamma_j(p + q, t, \vec{n})|F_4| + \sum_{k=1}^3 \gamma_j(p, t, -e_k)|F_k| \\ &\approx |F_4| \left(\gamma_j(p, t, \vec{n}) + \mathcal{O}(h) + \sum_{k=1}^3 \gamma_j(p, t, -e_k)n_k \right). \end{aligned}$$

With the abbreviation $\sigma_{j,k}(p, t) := \gamma_j(p, t, -e_k)$, we may also write

$$\lim_{h \rightarrow 0} \frac{\int_{\partial T} \gamma_j dx}{\int_{\partial T} dx} = \frac{1}{1 + n_1 + n_2 + n_3} \left(\gamma_j(p, t, n) - \sum_{k=1}^3 \sigma_{j,k}(p, t)n_k \right). \quad (9.8)$$

Now, for a continuous force field f we have

$$\int_T f_j dx \leq C_1 \text{Vol}(T) = C_1 \frac{1}{3} |F_3| \|P_0\vec{P}_3\|_{\mathbb{R}^d} = \frac{C_1}{3} |F_3| bh$$

and hence

$$\frac{\int_{\partial T} f_j dx}{\int_{\partial T} dx} = \frac{C_1}{3} \frac{|F_3|bh}{|F_1| + |F_2| + |F_3| + |F_4|} \leq C_2 bh \xrightarrow{h \rightarrow 0} 0.$$

From eqn (9.8) we finally obtain

$$\gamma_j(p, t, \vec{n}) = \sum_{k=1}^3 \sigma_{j,k}(p, t)n_k = \sigma_j^\top \vec{n}.$$

This derivation gives a linearized formulation of stress with respect to the normal vector $\vec{n}$. Moreover, starting with the equilibrium equation (9.5) and running over all possible bodies K we also find

$$-f_j = \text{div } \sigma_j; \quad = \partial_{x_1} \sigma_{j,1} + \partial_{x_2} \sigma_{j,2} + \partial_{x_3} \sigma_{j,3},$$

which follows from eqn (9.5). Hence,

$$\begin{aligned} (x \times f)_1 &= x_2 f_3 - x_3 f_2 \\ &= -x_2 (\partial_{x_1} \sigma_{3,1} + \partial_{x_2} \sigma_{3,2} + \partial_{x_3} \sigma_{3,3}) \\ &\quad + x_3 (\partial_{x_1} \sigma_{2,1} + \partial_{x_2} \sigma_{2,2} + \partial_{x_3} \sigma_{2,3}) \\ &= -\partial_{x_1} [x_2 \sigma_{3,1}] - \partial_{x_2} [x_2 \sigma_{3,2}] - \partial_{x_3} [x_2 \sigma_{3,3}] + \sigma_{3,2} \\ &\quad + \partial_{x_1} [x_3 \sigma_{2,1}] + \partial_{x_2} [x_3 \sigma_{2,2}] + \partial_{x_3} [x_3 \sigma_{2,3}] - \sigma_{2,3} \\ &= \text{div} (x_3 \sigma_{2,;} - x_2 \sigma_{3,;}) + \sigma_{3,2} - \sigma_{2,3}, \end{aligned}$$

and thus

$$\begin{aligned} \int_K (x \times f)_1 - \sigma_{3,2} + \sigma_{2,3} dx &= \int_K \text{div} (x_3 \sigma_{2,;} - x_2 \sigma_{3,;}) dx \\ &= \int_{\partial K} x_3 \sigma_{2,;}^T n - x_2 \sigma_{3,;}^T n dx = \int_{\partial K} x_3 \gamma_2 - x_2 \gamma_3 dx = - \int_{\partial K} (x \times \gamma)_1 dx. \end{aligned}$$

Exploiting the equilibrium eqn (9.6) we obtain $\sigma_{3,2} - \sigma_{2,3} = 0$. Using similar considerations for the remaining components we obtain the symmetry of the stress tensor, i.e.,

$$\sigma_{j,k} = \sigma_{k,j} \quad \text{for all } j, k.$$

9.1.4 Hooke's law

We are now ready to link the different parts together. Since deformation is related to strain and the forces are related to stress, the missing link is the connection between strain and stress. This link is provided by Hooke's law. Hooke's law "*ut tensio sic vis*" claims that strain is a reaction of stress. Here we use a linear model and take into account the fact that without strain the body undergoes

no stress. Moreover, we exploit the symmetry of the strain and stress tensors $V = (\varepsilon_{j,k})$ and $\Sigma = (\sigma_{j,k})$, respectively. For example, for dimension $d = 3$, these assumptions result in the linear model

$$\begin{pmatrix} \varepsilon_{1,1} \\ \varepsilon_{2,2} \\ \varepsilon_{3,3} \\ \varepsilon_{1,2} \\ \varepsilon_{1,3} \\ \varepsilon_{2,3} \end{pmatrix} = \begin{pmatrix} a_{1,1} & \cdots & a_{1,6} \\ \vdots & \ddots & \vdots \\ a_{6,1} & \cdots & a_{6,6} \end{pmatrix} \begin{pmatrix} \sigma_{1,1} \\ \sigma_{2,2} \\ \sigma_{3,3} \\ \sigma_{1,2} \\ \sigma_{1,3} \\ \sigma_{2,3} \end{pmatrix},$$

with 36 degrees of freedom. A considerable simplification occurs for isotropic elastic bodies. Here, the principal axes of strain and stress coincide. Thus, the strain and the stress tensors can be diagonalized simultaneously. Using an eigen-vector system as a new reference coordinate system, and denoting the eigenvalues of V and Σ by ε_j and σ_j , respectively, where $j = 1, 2, \dots, d$, the strain and stress tensors with respect to this coordinate system are given by

$$\hat{V} = \begin{pmatrix} \varepsilon_1 & & \\ & \ddots & \\ & & \varepsilon_d \end{pmatrix} \quad \text{and} \quad \hat{\Sigma} = \begin{pmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_d \end{pmatrix},$$

respectively. The linear model becomes

$$\begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_d \end{pmatrix} = \begin{pmatrix} b_{1,1} & \cdots & b_{1,d} \\ \vdots & \ddots & \vdots \\ b_{d,1} & \cdots & b_{d,d} \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \vdots \\ \sigma_d \end{pmatrix}. \quad (9.9)$$

Moreover, since the reaction of the isotropic body is independent of the direction, the matrix has to be circular, i.e., $b_{j,k} = b_{(k-j \bmod d)+1}$, and the non-diagonal elements have to be equal, i.e., $b_2 = \dots = b_d$. Two parameters are thus sufficient to describe the properties of an elastic, isotropic body in any spatial dimension d . Introducing Young's modulus of elasticity E and Poisson's contraction ratio ν , we have $b_{j,j} = b_1 = E^{-1}$ and $b_{j,k} = b_k = -\nu E^{-1}$, for $j \neq k$.

The relation between stress and strain can now be rewritten with the material constants E and ν ,

$$E\varepsilon_j = (1 + \nu)\sigma_j - \nu \sum_{j=1}^d \sigma_j, \quad j = 1, \dots, d,$$

and for the strain and stress tensors in the principal axes we have

$$E\hat{V} = (1 + \nu)\hat{\Sigma} - \nu \text{trace}(\hat{\Sigma})I_d.$$

Changing back to the original basis we conclude

$$\begin{aligned} EV &= (1 + \nu)\Sigma - \nu \text{trace}(\Sigma)I_d, \\ E \text{trace}(V) &= (1 + \nu(1 - d)) \text{trace}(\Sigma), \\ (1 + \nu)\Sigma &= EV + \frac{E\nu}{1 + \nu(1 - d)} \text{trace}(V)I_d. \end{aligned}$$

Introducing the Lamé constants

$$\mu := \frac{1}{2} \frac{E}{1 + \nu}, \quad \lambda := \frac{E\nu}{(1 + \nu)(1 - d)} \quad (9.10)$$

we can express stress in terms of strain and deformations with respect to general coordinates,

$$\begin{aligned} \sigma_{j,k} &= 2\mu\varepsilon_{j,k} + \lambda \text{trace}(V)\delta_{j,k} \\ &= \mu(\partial_{x_j}u_k + \partial_{x_k}u_j) + \lambda \text{div } u \delta_{j,k}. \end{aligned}$$

An elementary computation gives

$$\text{div } \sigma_{j,:} = \sum_{k=1}^d \partial_{x_k} \sigma_{j,k} = \mu \Delta u_j + (\lambda + \mu) \partial_{x_j} \text{div } u_j.$$

Summarizing these calculations we end up with the well-known Navier–Lamé equations

$$f = \mu \Delta u + (\lambda + \mu) \nabla \text{div } u;$$

cf., eqn (9.2). Young's modulus and Poisson's ratio can be derived from the Lamé constants λ and μ ,

$$E = \frac{2\mu(2\mu + d\lambda)}{2\mu + (d-1)\lambda}, \quad \nu = \frac{\lambda}{2\mu + (d-1)\lambda}.$$

The constant μ is also called *shear modulus* in the literature. Note that $\lambda = \lambda_d$ depends on the spatial dimension d ; cf., eqn (9.10). In particular for $d = 2$ and $d = 3$, we have

$$\lambda_2 = \frac{E\nu}{1 - \nu(1/4)} \quad \text{and} \quad \lambda_3 = \frac{E\nu}{1 - 2\nu(1/6)}.$$

Our next example illustrates the role of μ and λ .

Example 9.3 We solve a discretized version of the Navier–Lamé equations on a rectangle for various values of the Lamé constants. For $\lambda = 0$, we have $E = 2\mu$ and $\nu = 0$. Thus, the body shows no contraction under deformation. On increasing the value of μ , the body becomes more rigid ($\mu = 10, 25, 50$ in our example, shown in Fig. 9.5).

For fixed μ , $E = E(\lambda)$ and $\nu = \nu(\lambda)$ are monotonically increasing functions with respect to λ : $E(0) = 2\mu$, $E(\infty) = 2\mu d/(d-1)$, and $E'(\lambda) = 4\mu^2/(2\mu + (d-1)\lambda)^2 > 0$, $\nu(0) = 0$, $\nu(\infty) = 1/(d-1)$, and $\nu'(\lambda) = 2\mu/(2\mu + (d-1)\lambda)^2 > 0$.

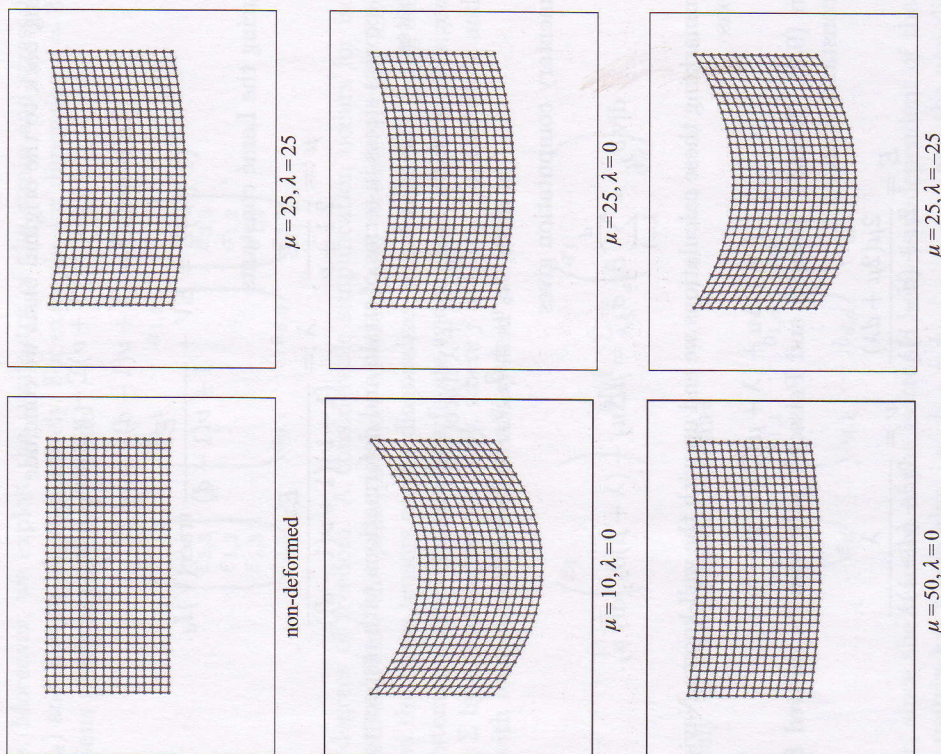


FIG. 9.5 Elastic bodies with various Lamé constants under deformation. Here, $f(x_1, x_2) = (0, -1)$ and the following boundary conditions are chosen: $u(L, x_2) = u(R, x_2) = 0$ on the left and right parts, $u_1(x_1, t) = 0$, and $\partial_{x_1} u_2(x_1, t) = \partial_{x_1} u_2(x_1, l) = 0$ on the upper and lower parts.

On increasing λ , the body becomes more rigid ($\lambda = -25, 0, 25$ in our example). Note that $\lambda \leq 0$ is not physically possible, since E and ν are positive constants.

9.2 Eigenfunctions of the Navier–Lamé operator

For particular boundary conditions, eigenfunctions of the Navier–Lamé operator are known explicitly. This knowledge may serve as a basis for a numerical solution scheme for the Navier–Lamé equations. Expanding the displacement and the

force with respect to the normalized eigenfunctions v_j ,

$$u = \sum_{j=1}^M \alpha_j v_j \quad \text{and} \quad f = \sum_{j=1}^M \beta_j v_j,$$

respectively, we have

$$\mathcal{A}u = \sum_{j=1}^M \alpha_j \mathcal{A}v_j = \sum_{j=1}^M \alpha_j \lambda_j v_j = \sum_{j=1}^M \beta_j v_j = f,$$

showing that the solution u is given by $\alpha_j = 0$ if $\lambda_j = 0$ and $\alpha_j = \beta_j / \lambda_j$ otherwise. Here we assume the system to be consistent. However, this approach has two drawbacks. In a numerical implementation the length M of the expansion has to be finite and one needs to compute the coefficients $\beta_j = \langle f, v_j \rangle$ numerically.

Bro-Nielsen & Gramkow (1996) used a modification of this technique. They worked with basis functions v_j , where $v_j(x) = v_0(x_j - x)$, and computed $\tilde{\beta}_j = \langle v_j, f \rangle = (v_0 * f)(x_j)$. This leads to a considerable improvement with respect to the numerical complexity of the method. However, a statement concerning the quality of this solution is missing.

We now give explicit formulas for the eigenvalues and eigenfunctions of the Navier–Lamé operator $\mathcal{A}$,

$$\mathcal{A}[u] := \mu \Delta u + (\lambda + \mu) \nabla \operatorname{div} u, \quad (9.11)$$

i.e., we are looking for a function $v : \Omega \subset \mathbb{R}^d \rightarrow \mathbb{R}^d$ and a constant κ , satisfying

$$\mathcal{A}[v] = \kappa v.$$

To facilitate the discussion, we assume $d = 3$ and $\Omega = [0, 1]^3$ throughout this section.

From the variational formulation we have implicit boundary conditions; cf., Theorem 9.4. In this section, however, we impose explicit second order boundary conditions on u . These boundary conditions result in an alteration of the partial differential operator.

Three different types of boundary conditions are discussed: *sliding*, *bending*, and *periodic*. With sliding boundary conditions the boundary is mapped onto the boundary but boundary points are allowed to slide on the boundary. Using bending boundary conditions, the corners of the domain are fixed but the sides are allowed to bend in- and outward. Finally, for the periodic boundary conditions, the domain Ω is mapped onto a torus.

Note that the results may also be found in the literature; cf., e.g., Christensen (1994). However, the proofs presented here appear to be new.

9.2.1 Sliding boundary conditions

The sliding boundary conditions considered here are defined by

$$\begin{aligned} u_1(z, x_2, x_3) &= u_2(x_1, z, x_3) = u_3(x_1, x_2, z) = 0, \\ \partial_{x_2} u_1(x_1, z, x_3) &= \partial_{x_3} u_1(x_1, x_2, z) = \partial_{x_1} u_2(z, x_2, x_3) = 0, \\ \partial_{x_3} u_2(x_1, x_2, z) &= \partial_{x_1} u_3(z, x_2, x_3) = \partial_{x_2} u_3(x_1, z, x_3) = 0, \end{aligned}$$

where $z = 0, 1$ and $x_j \in [0, 1]$.

Note that for this particular choice of boundary conditions the transformation φ , with $\varphi(x) = x + u(x)$, maps the boundary $\partial\Omega$ onto $\partial\Omega$. Since a particle on a side of the unit cube may slide along the side it belongs to, these boundary conditions are also called *sliding boundary conditions*.

Theorem 9.1 Let $d = 3$, $\Omega =]0, 1[^3$. For $a, b, c \in \mathbb{N}_0$ with $a^2 + b^2 + c^2 \neq 0$, let $p_1 = (a, b, c)^\top$, $p_2 \perp p_1$, $p_2 \neq 0$, and $p_3 := p_1 \times p_2$. Moreover, let

$$w_{abc}(x_1, x_2, x_3) := \begin{pmatrix} \sin(a\pi x_1) \cos(b\pi x_2) \cos(c\pi x_3) \\ \cos(a\pi x_1) \sin(b\pi x_2) \cos(c\pi x_3) \\ \cos(a\pi x_1) \cos(b\pi x_2) \sin(c\pi x_3) \end{pmatrix}.$$

Then

$$v_{abc;j} := \text{diag}(p_j) w_{abc}$$

is an eigenfunction of the partial differential operator $\mathcal{A}$ (cf., eqn (9.11)) to the eigenvalue

$$\kappa_{abc;j} = -\pi^2(a^2 + b^2 + c^2) \begin{cases} (2\mu + \lambda), & j = 1 \\ \mu, & j = 2, 3, \end{cases}$$

with respect to the sliding boundary conditions.

Proof For fixed a, b , and c , we define $w := w_{abc}$. Hence

$$\begin{aligned} \Delta w &= -\pi^2(a^2 + b^2 + c^2)w, \\ \text{div } w &= \pi(a + b + c)(\cos(a\pi x_1) \cos(b\pi x_2) \cos(c\pi x_3)), \\ \nabla \text{div } w &= -\pi^2(a + b + c) \text{diag}(a, b, c)w. \end{aligned}$$

It is easy to verify that w and hence $v_{abc;j}$ fulfill the boundary conditions. Exploiting the eigenfunction representation

$$\begin{aligned} \mathcal{A}[\text{diag}(\alpha, \beta, \gamma)w] &= \mu \text{diag}(\alpha, \beta, \gamma) \Delta w + (\lambda + \mu) \nabla \text{div}[\text{diag}(\alpha, \beta, \gamma)w] \\ &= -\mu\pi^2(a^2 + b^2 + c^2) \text{diag}(\alpha, \beta, \gamma)w \\ &\quad - \pi^2(\lambda + \mu)(\alpha a + b\beta + c\gamma) \text{diag}(a, b, c)w \end{aligned}$$

we find three solutions, either $\alpha = a$, $\beta = b$, $\gamma = c$, and hence $\kappa = -\pi^2(a^2 + b^2 + c^2)(2\mu + \lambda)$, or $\alpha a + b\beta + c\gamma = 0$ and hence $\kappa = -\pi^2(a^2 + b^2 + c^2)\mu$. Thus, for any triple (a, b, c) we can find three associated eigenfunctions $v_{abc;j} = \text{diag}(p_j)w$. $\square$

9.2.2 Bending boundary conditions

The bending boundary conditions considered here are defined by

$$\begin{aligned} u_1(x_1, z, x_3) &= u_1(x_1, x_2, z) = u_2(z, x_2, x_3) = 0, \\ u_2(x_1, x_2, z) &= u_3(z, x_2, x_3) = u_3(x_1, z, x_3) = 0, \\ \partial_{x_1} u_1(z, x_2, x_3) &= \partial_{x_2} u_2(x_1, z, x_3) = \partial_{x_3} u_3(x_1, x_2, z) = 0, \end{aligned}$$

where $z = 0, 1$ and $x_j \in [0, 1]$. Here, the transformation might move the boundary of the unit cube.

Theorem 9.2 Let $d = 3$, $\Omega =]0, 1[^3$. For $a, b, c \in \mathbb{N}_0$ with $a^2 + b^2 + c^2 \neq 0$, let $p_1 = (a, b, c)^\top$, $p_2 \perp p_1$, $p_2 \neq 0$, and $p_3 := p_1 \times p_2$. Moreover, let

$$w_{abc}(x_1, x_2, x_3) := \begin{pmatrix} \cos(a\pi x_1) \sin(b\pi x_2) \sin(c\pi x_3) \\ \sin(a\pi x_1) \cos(b\pi x_2) \sin(c\pi x_3) \\ \sin(a\pi x_1) \sin(b\pi x_2) \cos(c\pi x_3) \end{pmatrix}.$$

Then

$$v_{abc;j} := \text{diag}(p_j) w_{abc}$$

is an eigenfunction of the partial differential operator $\mathcal{A}$ (cf., eqn (9.11)) to the eigenvalue

$$\kappa_{abc;j} = -\pi^2(a^2 + b^2 + c^2) \begin{cases} (2\mu + \lambda), & j = 1 \\ \mu, & j = 2, 3, \end{cases}$$

with respect to the bending boundary conditions.

Proof The proof is along the same lines as the one for Theorem 9.1. $\square$

9.2.3 Periodic boundary conditions

The periodic conditions considered here are defined by

$$\begin{aligned} u_1(0, x_2, x_3) &= u_1(1, x_2, x_3), \\ u_2(x_1, 0, x_3) &= u_2(x_1, 1, x_3), \\ u_3(x_1, x_2, 0) &= u_3(x_1, x_2, 1). \end{aligned}$$

Theorem 9.3 Let $d = 3$, $\Omega =]0, 1[^3$. For $a, b, c \in \mathbb{N}_0$ with $a^2 + b^2 + c^2 \neq 0$, let $p_1 = (a, b, c)^T$, $p_2 \perp p_1$, $p_2 \neq 0$, and $p_3 := p_1 \times p_2$. Moreover, for $r, s, t = 0, 1$, let

$$w_{abc}^{rst}(x) := \begin{pmatrix} -\sin(a\pi x_1 + r\pi/2) \sin(b\pi x_2 + s\pi/2) \sin(c\pi x_3 + t\pi/2) \\ \cos(a\pi x_1 + r\pi/2) \cos(b\pi x_2 + s\pi/2) \sin(c\pi x_3 + t\pi/2) \\ \cos(a\pi x_1 + r\pi/2) \sin(b\pi x_2 + s\pi/2) \cos(c\pi x_3 + t\pi/2) \end{pmatrix}.$$

Then

$$v_{abc;j}^{rst} := \text{diag}(p_j) w_{abc}^{rst}$$

is an eigenfunction of the partial differential operator $\mathcal{A}$ (cf., eqn (9.11)) to the eigenvalue

$$\kappa_{abc;j} = -\pi^2(a^2 + b^2 + c^2) \begin{cases} (2\mu + \lambda), & j = 1 \\ \mu, & j = 2, 3, \end{cases}$$

with respect to the periodic boundary conditions.

Proof By construction w_{abc}^{rst} and thus $v_{abc;j}^{rst}$ are periodic. We only prove the case $r = s = t = 0$. The proofs of the other cases are similar. Putting

$$w(x) = \begin{pmatrix} -\sin(a\pi x_1) \sin(b\pi x_2) \sin(c\pi x_3) \\ \cos(a\pi x_1) \cos(b\pi x_2) \sin(c\pi x_3) \\ \cos(a\pi x_1) \sin(b\pi x_2) \cos(c\pi x_3) \end{pmatrix}$$

we compute

$$\begin{aligned} \Delta w &= -\pi^2(a^2 + b^2 + c^2)w \\ \text{div } w &= -\pi(a + b + c) \cos(a\pi x_1) \sin(b\pi x_2) \sin(c\pi x_3) \\ \nabla \text{div } w &= -\pi^2(a + b + c) \text{diag}(a, b, c)w. \end{aligned}$$

The rest of the proof is identical to that of Theorem 9.1. $\square$

9.3 Variational formulation of the elasticity problem

We are now ready to relate the framework introduced in the previous chapter to elastic registration. We provide a variational formulation of the elasticity problem. The required displacement is characterized as a minimizer of a certain functional $\mathcal{J}$. The functional consists of two terms. The first term $a[u, u]$ is a bi-linear form and serves as a regularizer, modeling the material properties and the inner forces, and the second term $b[u]$ reflects the outer forces.

Let $\Omega =]0, 1[^d \subset \mathbb{R}^d$ and $V[u] = (\varepsilon_{j,k}[u])_{j,k} \in \mathbb{R}^{d \times d}$ be the strain tensor, with $\varepsilon_{j,k}[u] = \frac{1}{2}(\partial_{x_j} u_k + \partial_{x_k} u_j)$; cf., eqn (9.3). We make use of the positive

semi-definite bi-linear form

$$\begin{aligned} a[u, v] &:= \int_{\Omega} 2\mu \text{trace}(V[u]^T V[v]) \\ &\quad + \lambda \text{trace}(V[u]) \text{trace}(V[v]) \, dx \\ &= \int_{\Omega} 2\mu \sum_{j,k=1}^d \varepsilon_{j,k}[u] \varepsilon_{j,k}[v] + \lambda \text{div } u \cdot \text{div } v \, dx \\ &= \int_{\Omega} \frac{\mu}{2} \sum_{j,k=1}^d (\partial_{x_j} u_k + \partial_{x_k} u_j)(\partial_{x_j} v_k + \partial_{x_k} v_j) \\ &\quad + \lambda \text{div } u \cdot \text{div } v \, dx \\ &= \int_{\Omega} \mu \sum_{k=1}^d \langle \nabla u_k + \partial_{x_k} u, \nabla v_k \rangle_{\mathbb{R}^d} + \lambda \text{div } u \text{div } v \, dx \end{aligned} \quad (9.12)$$

and the linear form

$$b[v] := \int_{\Omega} \langle f, v \rangle_{\mathbb{R}^d} \, dx. \quad (9.13)$$

A minimizer of the functional $\mathcal{J}$,

$$\mathcal{J}[u] := \frac{1}{2}a[u, u] + b[u], \quad (9.14)$$

with Dirichlet, Neumann, or periodic boundary conditions for u , is characterized by $d\mathcal{J}[u; v] = 0$ for all appropriate v .

The following theorem may be phrased for less smooth functions u and v as well. However, as we intend to solve the resulting partial differential equations by means of a finite difference scheme, the considered smoothness conditions are what we want.

Theorem 9.4 Let $\mathcal{J}$ be defined by eqn (9.14), a be defined by eqn (9.12), and b be defined by eqn (9.13), respectively. Moreover, let $u \in (C^2(\mathbb{R}^d))^d$. For the perturbation $v \in (C^2(\mathbb{R}^d))^d$, the Gâteaux derivative of $\mathcal{J}$ is given by

$$d\mathcal{J}[u, v] = \int_{\Omega} \langle f - \mu \Delta u - (\mu + \lambda) \nabla \text{div } u, v \rangle_{\mathbb{R}^d} \, dx.$$

Proof A computation gives

$$\begin{aligned}
d\mathcal{J}[u; v] &= \lim_{h \rightarrow 0} \frac{1}{h} (\mathcal{J}[u + hv] - \mathcal{J}[u]) = a[u, v] + b[v] \\
&= \int_{\Omega} \sum_{k=1}^d \mu \langle \nabla u_k + \partial_{x_k} u, \nabla v_k \rangle_{\mathbb{R}^d} + \lambda \operatorname{div} u \operatorname{div} v + \langle f, v \rangle_{\mathbb{R}^d} dx \\
&= \int_{\partial\Omega} \sum_{k=1}^d v_k \langle \nabla u_k + \partial_{x_k} u, \vec{n} \rangle_{\mathbb{R}^d} + \lambda \operatorname{div} u \langle v, \vec{n} \rangle_{\mathbb{R}^d} dx \\
&\quad + \int_{\Omega} \langle f - \mu \Delta u - (\mu + \lambda) \nabla \operatorname{div} u, v \rangle_{\mathbb{R}^d} dx,
\end{aligned}$$

where $\vec{n}$ denotes the outer normal vector on $\partial\Omega$. Exploiting the implicit boundary conditions $\operatorname{div} u = \langle \nabla u_k + \partial_{x_k} u, \vec{n} \rangle_{\mathbb{R}^d} = 0$ on $\partial\Omega$, the boundary integral vanishes, which completes the proof. $\square$

Note that Theorem 9.4 provides a weak formulation of the Navier-Lamé equations (9.2). In image registration, however, where one is typically interested in very smooth deformations, the weak formulation of the problem provides no particular advantage in comparison to a strong formulation.

With this particular bi-linear form, we can express the elastic potential of a displacement u for any dimension d .

Definition 9.1 Let $d \in \mathbb{N}$, $u : \mathbb{R}^d \rightarrow \mathbb{R}^d$, $u_j \in C^2(\mathbb{R}^d)$, $j = 1, \dots, d$. The elastic potential $\mathcal{P}$ of the displacement u is defined by

$$\begin{aligned}
\mathcal{P}[u] &:= \frac{1}{2} a[u, u] \\
&= \int_{\Omega} \mu \operatorname{trace}(V[u]^T V[u]) + \frac{\lambda}{2} \operatorname{trace}(V[u])^2 dx \\
&= \int_{\Omega} \frac{\mu}{4} \sum_{j,k=1}^d (\partial_{x_j} u_k + \partial_{x_k} u_j)^2 + \frac{\lambda}{2} (\operatorname{div} u)^2 dx.
\end{aligned}$$

We note that the elastic potential is invariant under rigid transformations. If considered with additional explicit homogeneous Dirichlet boundary conditions, the bi-linear form a is positive definite.

For the important cases, dimension $d = 2$ and $d = 3$, we give explicit formulas for the elastic potential $\mathcal{P}_d$,

$$\begin{aligned}
\mathcal{P}_2[u] &= \int_{\mathbb{R}^2} \frac{\lambda}{2} (\partial_{x_1} u_1 + \partial_{x_2} u_2)^2 \\
&\quad + \mu \left\{ (\partial_{x_1} u_1)^2 + (\partial_{x_2} u_2)^2 + \frac{1}{2} (\partial_{x_1} u_2 + \partial_{x_2} u_1)^2 \right\} dx, \\
\mathcal{P}_3[u] &= \int_{\mathbb{R}^3} \frac{\lambda}{2} (\partial_{x_1} u_1 + \partial_{x_2} u_2 + \partial_{x_3} u_3)^2 \\
&\quad + \mu \left\{ (\partial_{x_1} u_1)^2 + (\partial_{x_2} u_2)^2 + (\partial_{x_3} u_3)^2 \right\} + \frac{\mu}{2} \left\{ (\partial_{x_1} u_2 + \partial_{x_2} u_1)^2 \right. \\
&\quad \left. + (\partial_{x_1} u_3 + \partial_{x_3} u_1)^2 + (\partial_{x_2} u_3 + \partial_{x_3} u_2)^2 \right\} dx.
\end{aligned}$$

9.4 Finite difference approximation of the Navier-Lamé equations

We investigate numerical schemes for the minimization of $\mathcal{J}$,

$$\mathcal{J}[u] = \mathcal{D}[R, T; u] + \mathcal{P}[u],$$

where $\mathcal{D}$ is given by eqn (8.10) and $\mathcal{P}$ is the elastic potential; cf., Definition 9.1.

Our numerical treatment is based on the Euler-Lagrange equations, which coincide with the Navier-Lamé equations (9.2) for this particular regularizer. In particular, we discuss a scheme based on the fixed-point iteration eqn (8.6). Moreover, the scheme is based on a finite difference approximation for semi-linear partial differential equations. This finite difference approximation with respect to periodic boundary conditions (cf., Section 9.2.3) is explicitly given for dimension $d = 2, 3$.

The main point is that due to the choices of the particular boundary conditions and discretization, we finally end up with a high-dimensional linear system of equations, where the matrix is highly structured. This structure enables us to apply fast Fourier transformation techniques to explicitly invert the matrix in a complexity $\mathcal{O}(N \log N)$, where N is the number of image voxels.

9.4.1 The two-dimensional case

For the Navier-Lamé equations (9.2) and dimension $d = 2$ we have

$$\begin{aligned}
\mathcal{A}[u] &= \mu \Delta u + (\lambda + \mu) \nabla \operatorname{div} u \\
&= \begin{pmatrix} (\lambda + 2\mu) \partial_{x_1 x_1} u_1 + \mu \partial_{x_2 x_2} u_1 + (\lambda + \mu) \partial_{x_1 x_2} u_2 \\ (\lambda + \mu) \partial_{x_1 x_2} u_1 + \mu \partial_{x_1 x_1} u_2 + (\lambda + 2\mu) \partial_{x_2 x_2} u_2 \end{pmatrix}.
\end{aligned}$$

Hence, with

$$\begin{aligned}\mathcal{A}^{1,1}[u_1] &:= (\lambda + 2\mu)\partial_{x_1x_1}u_1 + \mu\partial_{x_2x_2}u_1, \\ \mathcal{A}^{1,2}[u_2] &:= (\lambda + \mu)\partial_{x_1x_2}u_2, \\ \mathcal{A}^{2,1}[u_1] &:= (\lambda + \mu)\partial_{x_1x_2}u_1, \\ \mathcal{A}^{2,2}[u_2] &:= \mu\partial_{x_1x_1}u_2 + (\lambda + 2\mu)\partial_{x_2x_2}u_2,\end{aligned}$$

and, using eqns (8.12) and (8.13),

$$\begin{aligned}S^{1,1} &:= (S^{2,2})^\top = \begin{pmatrix} 0 & (\lambda + 2\mu) & 0 \\ \mu & -2(\lambda + 3\mu) & \mu \\ 0 & (\lambda + 2\mu) & 0 \end{pmatrix}, \\ S^{1,2} &:= (S^{2,1})^\top := \frac{\lambda + \mu}{4} \begin{pmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{pmatrix},\end{aligned}\quad (9.15)$$

we have

$$\begin{aligned}\mathcal{A}[u](X) &= \begin{pmatrix} \mathcal{A}^{1,1}[u_1](X) + \mathcal{A}^{1,2}[u_2](X) \\ \mathcal{A}^{2,1}[u_1](X) + \mathcal{A}^{2,2}[u_2](X) \end{pmatrix} \\ &\approx \begin{pmatrix} S^{1,1} * u_1(X) + S^{1,2} * u_2(X) \\ S^{2,1} * u_1(X) + S^{2,2} * u_2(X) \end{pmatrix}\end{aligned}$$

or

$$\mathcal{A}[u](\vec{X}) \approx \begin{pmatrix} \mathcal{A}^{1,1}u_1(\vec{X}) + \mathcal{A}^{1,2}u_2(\vec{X}) \\ \mathcal{A}^{2,1}u_1(\vec{X}) + \mathcal{A}^{2,2}u_2(\vec{X}) \end{pmatrix} =: A\vec{u}.$$

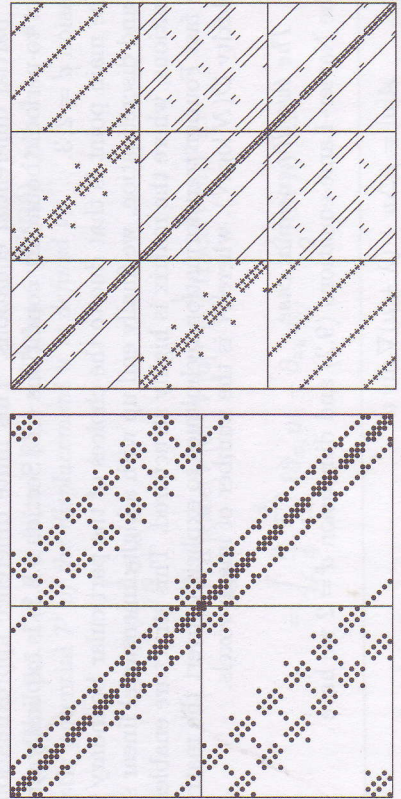


FIG. 9.6 Non-zero pattern of the matrix A for $d = 2$ (LEFT) and $d = 3$ (RIGHT), $n_1 = 5$, $n_2 = 7$, $n_3 = 4$.

Note that different boundary conditions result in different convolution schemes; see also Section 8.4. A typical non-zero pattern of the matrix A is shown in Fig. 9.6.

9.4.2 The three-dimensional case

For the Navier-Lamé equations (9.2) and dimension $d = 3$ we have

$$\begin{aligned}\mathcal{A}[u] &= \mu\Delta u + (\lambda + \mu)\nabla \operatorname{div} u \\ &= \begin{pmatrix} (\lambda + 2\mu)\partial_{x_1x_1}u_1 + \mu\partial_{x_2x_2}u_1 + \mu\partial_{x_3x_3}u_1 \\ \mu\partial_{x_1x_1}u_2 + (\lambda + 2\mu)\partial_{x_2x_2}u_2 + \mu\partial_{x_3x_3}u_2 \\ \mu\partial_{x_1x_1}u_3 + \mu\partial_{x_2x_2}u_3 + (\lambda + 2\mu)\partial_{x_3x_3}u_3 \end{pmatrix} \\ &\quad + (\lambda + \mu) \begin{pmatrix} \partial_{x_1x_2}u_2 + \partial_{x_1x_3}u_3 \\ \partial_{x_1x_2}u_1 + \partial_{x_2x_3}u_3 \\ \partial_{x_1x_3}u_1 + \partial_{x_2x_3}u_2 \end{pmatrix}.\end{aligned}$$

Hence, with

$$\begin{aligned}\mathcal{A}^{1,1}[u_1] &:= (\lambda + 2\mu)\partial_{x_1x_1}u_1 + \mu\partial_{x_2x_2}u_1 + \mu\partial_{x_3x_3}u_1, \\ \mathcal{A}^{1,2}[u_2] &:= (\lambda + \mu)\partial_{x_1x_2}u_2, \\ \mathcal{A}^{1,3}[u_3] &:= (\lambda + \mu)\partial_{x_1x_3}u_3, \\ \mathcal{A}^{2,1}[u_1] &:= (\lambda + \mu)\partial_{x_1x_2}u_1 \\ \mathcal{A}^{2,2}[u_2] &:= \mu\partial_{x_1x_1}u_2 + (\lambda + 2\mu)\partial_{x_2x_2}u_2 + \mu\partial_{x_3x_3}u_2, \\ \mathcal{A}^{2,3}[u_3] &:= (\lambda + \mu)\partial_{x_2x_3}u_3, \\ \mathcal{A}^{3,1}[u_1] &:= (\lambda + \mu)\partial_{x_1x_3}u_1 \\ \mathcal{A}^{3,2}[u_2] &:= (\lambda + \mu)\partial_{x_2x_3}u_2 \\ \mathcal{A}^{3,3}[u_3] &:= \mu\partial_{x_1x_1}u_3 + \mu\partial_{x_2x_2}u_3 + (\lambda + 2\mu)\partial_{x_3x_3}u_3,\end{aligned}$$

and with the matrix stencils summarized in Table 9.1, where eqns (8.12) and (8.13) are used, we have

$$\begin{aligned}\mathcal{A}[u](X) &= \begin{pmatrix} \mathcal{A}^{1,1}[u_1](X) + \mathcal{A}^{1,2}[u_2](X) + \mathcal{A}^{1,3}[u_3](X) \\ \mathcal{A}^{2,1}[u_1](X) + \mathcal{A}^{2,2}[u_2](X) + \mathcal{A}^{2,3}[u_3](X) \\ \mathcal{A}^{3,1}[u_1](X) + \mathcal{A}^{3,2}[u_2](X) + \mathcal{A}^{3,3}[u_3](X) \end{pmatrix} \\ &\approx \begin{pmatrix} S^{1,1} * u_1(X) + S^{1,2} * u_2(X) + S^{1,3} * u_3(X) \\ S^{2,1} * u_1(X) + S^{2,2} * u_2(X) + S^{2,3} * u_3(X) \\ S^{3,1} * u_1(X) + S^{3,2} * u_2(X) + S^{3,3} * u_3(X) \end{pmatrix} \\ \text{or} \\ \mathcal{A}[u](\vec{X}) &\approx \begin{pmatrix} \mathcal{A}^{1,1}u_1(\vec{X}) + \mathcal{A}^{1,2}u_2(\vec{X}) + \mathcal{A}^{1,3}u_3(\vec{X}) \\ \mathcal{A}^{2,1}u_1(\vec{X}) + \mathcal{A}^{2,2}u_2(\vec{X}) + \mathcal{A}^{2,3}u_3(\vec{X}) \\ \mathcal{A}^{3,1}u_1(\vec{X}) + \mathcal{A}^{3,2}u_2(\vec{X}) + \mathcal{A}^{3,3}u_3(\vec{X}) \end{pmatrix} =: A\vec{u}.\end{aligned}$$

Table 9.1 Elastic registration: matrix stencils for dimension $d = 3$.

$$\begin{aligned}
S_{j,k,\ell}^{1,1} &= \alpha_{j,k,\ell}, & S_{j,k,\ell}^{2,2} &= \alpha_{k,\ell,j}, & S_{j,k,\ell}^{3,3} &= \alpha_{\ell,j,k}, \\
\alpha_{j,k,\ell} &= \begin{cases} -2(\lambda + 4\mu), & j = k = \ell = 2, \\ \lambda + 2\mu, & j = 1, 3, k = \ell = 2, \\ \mu, & j = 2, k = 2, \ell = 1, 3, \\ \mu, & j = 2, \ell = 2, k = 1, 3, \\ 0, & \text{else,} \end{cases} \\
S_{j,k,\ell}^{1,2} &= S_{j,k,\ell}^{2,1} = q(\ell, j, k), \\
S_{j,k,\ell}^{1,3} &= S_{j,k,\ell}^{3,1} = q(k, \ell, j), \\
S_{j,k,\ell}^{2,3} &= S_{j,k,\ell}^{3,2} = q(j, k, \ell), \\
q(a, b, c) &= \frac{\lambda + \mu}{4} \begin{cases} 1, & a = 2 \wedge (b = c = 1 \vee b = c = 3), \\ -1, & a = 2 \wedge (b = 1 \wedge c = 3 \vee b = 3 \wedge c = 1), \\ 0, & \text{else.} \end{cases}
\end{aligned}$$

Note that different boundary conditions result in different convolution schemes; see also Section 8.4. The non-zero pattern of the matrix A is shown in Fig. 9.6 for $n_1 = 5$, $n_2 = 7$, and $n_3 = 4$.

The remaining problem after discretization is to solve linear systems of equations $Au = f$. Note that the dimension of A can be huge. The registration of two n_1 -by- n_2 -by- n_3 images leads to $A \in \mathbb{R}^{(3N) \times (3N)}$, where $N = n_1 n_2 n_3$. For example, if the images are 256^3 , we have to deal with approximately $5 \cdot 10^7$ unknowns. The registration of a part of the human brain presented at the end of this chapter results in linear systems with 52 428 800 unknowns. Thus, the usage and acceptance of this registration is strongly related to fast and efficient schemes for solving the linear system. In the next section we provide a solution scheme which is based on fast Fourier transformation techniques and leads to an $\mathcal{O}(N \log N)$ implementation.

9.5 Factorization of a structured matrix

Here, we identify a class of matrices which may be diagonalized by fast Fourier transformation (FFT) techniques. Our discussion follows that of Fischer & Modersitzki (1999). As it turns out, for dimension $d = 2, 3$, the matrices arising from the finite difference discretization of the Navier–Lamé equations introduced in the last section belong to this class if periodic boundary conditions are used.

Important ingredients for the following decompositions are the basic *circular matrix*

$$C_n := \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & 1 \\ 1 & 0 & \cdots & \cdots & 0 \end{pmatrix} \in \mathbb{C}^{n \times n}, \quad (9.16)$$

the n^{th} root of unity

$$\omega_n := e^{-2\pi i/n}, \quad (9.17)$$

the Fourier matrix $F_n \in \mathbb{C}^{n \times n}$,

$$F_n := \frac{1}{\sqrt{n}} \left(\omega_n^{(j-1)(k-1)} \right)_{j,k=1,\dots,n}, \quad (9.18)$$

and

$$\Omega_n := \text{diag}(\omega_n^0, \dots, \omega_n^{n-1}). \quad (9.19)$$

In this section we make intensive use of the Kronecker calculus; see, e.g., Horn & Johnson (1991, §4) or Brewer (1978).

9.5.1 The three-dimensional case

We start by defining this class of diagonalizable matrices for the three-dimensional case. The two-dimensional case is deduced later. Let $A \in \mathbb{R}^{(3n_1 n_2 n_3) \times (3n_1 n_2 n_3)}$ be a 3-by-3 block matrix with block circular matrices $A^{p,q}$,

$$A = \begin{pmatrix} A^{1,1} & A^{1,2} & A^{1,3} \\ A^{2,1} & A^{2,2} & A^{2,3} \\ A^{3,1} & A^{3,2} & A^{3,3} \end{pmatrix} \in \mathbb{R}^{(3n_1 n_2 n_3) \times (3n_1 n_2 n_3)}, \quad (9.20)$$

where for $p, q = 1, 2, 3$,

$$\begin{aligned}
A^{p,q} &= C_{n_3}^{-1} \otimes A_3^{p,q} + I_{n_3} \otimes A_2^{p,q} + C_{n_3} \otimes A_1^{p,q} \\
&= \begin{pmatrix} A_2^{p,q} & A_1^{p,q} & A_3^{p,q} \\ A_3^{p,q} & \ddots & \ddots \\ A_1^{p,q} & \ddots & A_2^{p,q} \end{pmatrix} \in \mathbb{R}^{(n_1 n_2 n_3) \times (n_1 n_2 n_3)}. \quad (9.21)
\end{aligned}$$

Each block $A_r^{p,q}$, $r = 1, 2, 3$, is block circular,

$$A_r^{p,q} = C_{n_2}^{-1} \otimes A_{r,3}^{p,q} + I_{n_2} \otimes A_{r,2}^{p,q} + C_{n_2} \otimes A_{r,1}^{p,q} \quad (9.22)$$

$$= \begin{pmatrix} A_{r,2}^{p,q} & A_{r,1}^{p,q} & A_{r,3}^{p,q} \\ A_{r,3}^{p,q} & \ddots & \ddots \\ \ddots & A_{r,2}^{p,q} & A_{r,1}^{p,q} \end{pmatrix} \in \mathbb{R}^{(n_1 n_2) \times (n_1 n_2)},$$

with circular blocks $A_{r,s}^{p,q}$, $s = 1, 2, 3$,

$$A_{r,s}^{p,q} = S_{3,s,r}^{p,q} C_{n_1}^{-1} + S_{2,s,r}^{p,q} I_{n_1} + S_{1,s,r}^{p,q} C_{n_1} \quad (9.23)$$

$$= \begin{pmatrix} S_{2,s,r}^{p,q} & S_{1,s,r}^{p,q} & S_{3,s,r}^{p,q} \\ S_{3,s,r}^{p,q} & \ddots & \ddots \\ \ddots & S_{2,s,r}^{p,q} & S_{1,s,r}^{p,q} \end{pmatrix} \in \mathbb{R}^{n_1 \times n_1}.$$

The 3-by-3-by-3 arrays $S_{t,r,s}^{p,q} = (S_{t,s,r}^{p,q})_{t,r,s=1,2,3}$, $p, q = 1, 2, 3$, are given and, in the case of elastic registration, coincide with the matrix stencils of the underlying discretization; cf., Table 9.1.

A main ingredient of our analysis is the fact that any circular matrix may be diagonalized by a Fourier matrix, a proof of which may be found for example in Davis (1979, Theorem 3.2.2).

Lemma 9.5 *Let $n \in \mathbb{N}$, and F_n and Ω_n be defined by eqns (9.18) and (9.19), respectively.*

1. *The Fourier matrix is unitary, $F_n^{-1} = F_n^H$.*
2. *The circular matrix C_n (cf., eqn (9.16)) is diagonalized by F_n , $F_n^H C_n F_n = \Omega_n$.*
3. *Any circular matrix $Z_n = \sum_{j=-n}^n \alpha_j C_n^j$ is diagonalized by F_n ,*

$$F_n^H Z_n F_n = \sum_{j=-n}^n \alpha_j (\Omega_n)^j.$$

Next, we show how to factorize the block matrix A ; cf., eqn (9.20).

Lemma 9.6 *Let $n_1, n_2, n_3 \in \mathbb{N}$, $p, q = 1, 2, 3$.*

1. *The matrices $A_{r,s}^{p,q} \in \mathbb{R}^{n_1 \times n_1}$ (cf., eqn (9.23)) are diagonalized by F_{n_1} ,*

$$D_{r,s}^{p,q} := F_{n_1}^H A_{r,s}^{p,q} F_{n_1},$$

$$D_{r,s}^{p,q} = \text{diag}(D_{r,s;j}^{p,q}, j = 1, \dots, n_1),$$

$$\text{where } D_{r,s;j}^{p,q} = S_{2,s,r}^{p,q} + S_{3,s,r}^{p,q} \bar{\omega}_{n_1}^{j-1} + S_{1,s,r}^{p,q} \omega_{n_1}^{j-1}.$$

2. *The matrices $A_r^{p,q} \in \mathbb{R}^{(n_1 n_2) \times (n_1 n_2)}$ (cf., eqn (9.22)) are diagonalized by $F_{n_2} \otimes F_{n_1}$,*

$$D_r^{p,q} := (F_{n_2} \otimes F_{n_1})^H A_r^{p,q} (F_{n_2} \otimes F_{n_1}),$$

$$D_r^{p,q} = \text{diag}\left(D_{r;k,j}^{p,q}, j = 1, \dots, n_1, k = 1, \dots, n_2\right),$$

$$\text{where } D_{r;k,j}^{p,q} = D_{r,2;j}^{p,q} + \bar{\omega}_{n_2}^{k-1} D_{r,3;j}^{p,q} + \omega_{n_2}^{k-1} D_{r,1;j}^{p,q}.$$

3. *The matrices $A^{p,q} \in \mathbb{R}^{(n_1 n_2 n_3) \times (n_1 n_2 n_3)}$ (cf., eqn (9.21)) are diagonalized by $F_{n_3} \otimes F_{n_2} \otimes F_{n_1}$,*

$$D^{p,q} := (F_{n_3} \otimes F_{n_2} \otimes F_{n_1})^H A^{p,q} (F_{n_3} \otimes F_{n_2} \otimes F_{n_1}),$$

$$D^{p,q} = \text{diag}\left(D_{\ell,k,j}^{p,q}, j = 1, \dots, n_1, k = 1, \dots, n_2, \ell = 1, \dots, n_3\right),$$

$$\text{where } D_{\ell,k,j}^{p,q} = D_{2;k,j}^{p,q} + \bar{\omega}_{n_3}^{\ell-1} D_{3;k,j}^{p,q} + \omega_{n_3}^{\ell-1} D_{1;k,j}^{p,q}.$$

4. *For the matrix $A \in \mathbb{R}^{(3n_1 n_2 n_3) \times (3n_1 n_2 n_3)}$ (cf., eqn (9.20)) we have*

$$F^H A F = \begin{pmatrix} D^{1,1} & D^{1,2} & D^{1,3} \\ D^{2,1} & D^{2,2} & D^{2,3} \\ D^{3,1} & D^{3,2} & D^{3,3} \end{pmatrix}$$

$$\text{where } F = I_3 \otimes F_{n_3} \otimes F_{n_2} \otimes F_{n_1}.$$

Proof

1. Follows directly from Lemma 9.5.
2. From

$$\begin{aligned}
B_r^{p,q} &:= (I_{n_2} \otimes F_{n_1})^H A_r^{p,q} (I_{n_2} \otimes F_{n_1}) \\
&= C_{n_2}^{-1} \otimes (F_{n_1}^H A_{r,3}^{p,q} F_{n_1}) \\
&\quad + I_{n_2} \otimes (F_{n_1}^H A_{r,2}^{p,q} F_{n_1}) + C_{n_2} \otimes (F_{n_1}^H A_{r,1}^{p,q} F_{n_1}) \\
&= C_{n_2}^{-1} \otimes D_{r,3}^{p,q} + I_{n_2} \otimes D_{r,2}^{p,q} + C_{n_2} \otimes D_{r,1}^{p,q}
\end{aligned}$$

we deduce that $B_r^{p,q}$ is a block circular matrix with diagonal blocks.

A suitable permutation P yields

$$P^{-1} B_r^{p,q} P = \text{diag} (L_{r,1}^{p,q}, \dots, L_{r,n_1}^{p,q}),$$

where $L_{r,j}^{p,q} = C_{n_2}^{-1} D_{r,3,j}^{p,q} + I_{n_2} D_{r,2,j}^{p,q} + C_{n_2} D_{r,1,j}^{p,q} \in \mathbb{R}^{n_2 \times n_2}$ and the permutation maps $(1, 2, \dots, n_1, n_1 + 1, n_1 + 2, \dots, n_1^2)$ to $(1, n_1 + 1, \dots, n_1(n_1 + 1) + 1, 2, n_1 + 2, \dots, n_1^2)$. Finally, $L_{r,j}^{p,q}$ may be diagonalized by F_{n_2} showing that

$$D_{r;k,j}^{p,q} = D_{r,2,j}^{p,q} + \bar{\omega}_{n_2}^{k-1} D_{r,3,j}^{p,q} + \omega_{n_2}^{k-1} D_{r,1,j}^{p,q}, \quad k = 1, \dots, n_2,$$

are the eigenvalues of $L_{r,j}^{p,q}$ and thus

$$D_r^{p,q} = (F_{n_2} \otimes F_{n_1})^H A_r^{p,q} (F_{n_2} \otimes F_{n_1}).$$

3. From

$$\begin{aligned}
B^{p,q} &:= (I_{n_3} \otimes F_{n_2} \otimes F_{n_1})^H A^{p,q} (I_{n_3} \otimes F_{n_2} \otimes F_{n_1}) \\
&= C_{n_3}^{-1} \otimes D_3^{p,q} + I_{n_3} \otimes D_2^{p,q} + C_{n_3} \otimes D_1^{p,q}
\end{aligned}$$

we deduce that also $B^{p,q}$ is a block circular matrix with diagonal blocks. A suitable permutation P yields

$$P^{-1} B^{p,q} P = \text{diag} (L_1^{p,q}, \dots, L_{n_1 n_2}^{p,q}),$$

where for $j = 1, \dots, n_1$, $k = 1, \dots, n_2$, and $m = (k-1)n_2 + j$,

$$L_m^{p,q} = C_{n_3}^{-1} D_{3,k,j}^{p,q} + I_{n_3} D_{2,k,j}^{p,q} + C_{n_3} D_{1,k,j}^{p,q} \in \mathbb{R}^{n_3 \times n_3}.$$

Finally, $L_m^{p,q}$ may be diagonalized by F_{n_3} .

4. The matrices $A^{p,q}$ are simultaneously diagonalizable. □

9.5.2 The two-dimensional case

For two-dimensional registration one is also interested in a 2-by-2 block matrix $A \in \mathbb{R}^{(2n_1 n_2) \times (2n_1 n_2)}$,

$$A = \begin{pmatrix} A_0^{1,1} & A_0^{1,2} \\ A_0^{2,1} & A_0^{2,2} \end{pmatrix} \in \mathbb{R}^{(2n_1 n_2) \times (2n_1 n_2)}, \quad (9.24)$$

where for $p, q = 1, 2$, the matrices $A_0^{p,q}$ are defined in eqn (9.22) using the 3-by-3 stencils $S^{p,q}$, where, for convenience, we set $S_{t,s}^{p,q} := S_{t,s,0}^{p,q}$, $t, s = 1, 2, 3$.

The eigendecomposition of the matrix for the two-dimensional registration can be deduced from Lemma 9.6 and is summarized in the following corollary.

Corollary 9.7 For the matrix $A \in \mathbb{R}^{(2n_1 n_2) \times (2n_1 n_2)}$ (cf., eqn (9.24)) we have

$$F^H A F = \begin{pmatrix} D_0^{1,1} & D_0^{1,2} \\ D_0^{2,1} & D_0^{2,2} \end{pmatrix}$$

where $F = I_2 \otimes F_{n_2} \otimes F_{n_1}$, $A_0^{p,q}$ is given by eqn (9.22),

$$D_0^{p,q} := (F_{n_2} \otimes F_{n_1})^H A_0^{p,q} (F_{n_2} \otimes F_{n_1}),$$

$$D_0^{p,q} = \text{diag} \left(D_{0,k,j}^{p,q}, j = 1, \dots, n_1, \right. \\ \left. D_{0,k,j}^{p,q}, k = 1, \dots, n_2 \right),$$

$$\text{where } D_{0,k,j}^{p,q} = D_{0,2,j}^{p,q} + \bar{\omega}_{n_2}^{k-1} D_{0,3,j}^{p,q} + \omega_{n_2}^{k-1} D_{0,1,j}^{p,q},$$

$$\text{and } D_{0,s,j}^{p,q} = S_{2,s}^{p,q} + S_{3,s}^{p,q} \bar{\omega}_{n_1}^{j-1} + S_{1,s}^{p,q} \omega_{n_1}^{j-1}.$$

9.5.3 Inversion of the structured matrices

Our goal is to explicitly invert $F^H A F$. The trouble is that the matrix A might be singular (and for the particular choices of S^p we are interested in, it is). Therefore we compute the so-called Moore–Penrose pseudo-inverse; see, e.g., Golub & van Loan (1989). To begin with, we investigate the Moore–Penrose pseudo-inverse $B^\dagger$ of a 2-by-2 matrix B .

Lemma 9.8 For the symmetric matrix $B = \begin{pmatrix} b^1 & b^2 \\ b^2 & b^4 \end{pmatrix} \in \mathbb{R}^{2 \times 2}$ the Moore–Penrose pseudo-inverse $B^\dagger$ is given by

$$B^\dagger = \begin{cases} B^{-1}, & \text{for } \det(B) \neq 0, \\ 1/(b^1 + b^4)^2 \cdot B, & \text{for } \det(B) = 0, \quad b^1 \neq 0 \vee b^4 \neq 0, \\ 0, & \text{otherwise.} \end{cases}$$

Proof

1. If $\det(B) \neq 0$, then $B^\dagger = B^{-1}$.
2. Let $\det(B) = 0$ but $b^1 \neq 0 \vee b^4 \neq 0$. Then B has a singular value decomposition $B = Q \operatorname{diag}(b^1 + b^4, 0) Q^\top$, where

$$Q = \frac{1}{\sqrt{(b^1)^2 + (b^2)^2}} \begin{pmatrix} b^1 & b^2 \\ b^2 & -b^1 \end{pmatrix} \quad \text{for } b^1 \neq 0,$$

$$Q = \frac{1}{\sqrt{(b^4)^2 + (b^2)^2}} \begin{pmatrix} b^2 & b^4 \\ b^4 & -b^2 \end{pmatrix} \quad \text{for } b^1 = 0.$$

Hence, the pseudo-inverse is given by $B^\dagger = 1/(b^1 + b^4)^2 B$.

3. Let $\det(B) = 0$ and $b^1 = b^4 = 0$. Thus, $B = \mathbf{0}$ and $B^\dagger = \mathbf{0}$. $\square$

We are now ready to present our main result. For the computation of the pseudo-inverse of a symmetric 3-by-3 matrix we forgo a lengthy formula and instead make use of numerical schemes.

Theorem 9.9

1. For $d = 2$, let A be defined by eqn (9.24), where $S^{p,q} \in \mathbb{R}^{3 \times 3}$ are such that $D_0^{p,q}$ are real and $D_0^{2,1} = D_0^{1,2}$. Then $A^\dagger = F D^\dagger F^H$, where

$$D^\dagger := \begin{pmatrix} D_0^{1,1,\dagger} & D_0^{1,2,\dagger} \\ D_0^{1,2,\dagger} & D_0^{2,2,\dagger} \end{pmatrix}$$

and the diagonal matrices $D_0^{p,q,\dagger} := \operatorname{diag}(D_{0,k,j}^{p,q,\dagger})$ are defined by

$$\begin{pmatrix} D_{0,k,j}^{1,1,\dagger} & D_{0,k,j}^{1,2,\dagger} \\ D_{0,k,j}^{1,2,\dagger} & D_{0,k,j}^{2,2,\dagger} \end{pmatrix} := \begin{pmatrix} D_{0,k,j}^{1,1} & D_{0,k,j}^{1,2} \\ D_{0,k,j}^{1,2} & D_{0,k,j}^{2,2} \end{pmatrix}^\dagger, \quad \begin{matrix} j = 1, \dots, n_1, \\ k = 1, \dots, n_2. \end{matrix}$$

2. For $d = 3$, let A be defined by eqn (9.20), where $S^{p,q} \in \mathbb{R}^{3 \times 3 \times 3}$ are such that $D_{\ell,k,j}^{p,q}$ are real, and $D^{p,q} = D^{q,p}$ for $p, q = 1, 2, 3$. Then $A^\dagger = F D^\dagger F^H$, where

$$D^\dagger := \begin{pmatrix} D_{\ell,k,j}^{1,1,\dagger} & D_{\ell,k,j}^{1,2,\dagger} & D_{\ell,k,j}^{1,3,\dagger} \\ D_{\ell,k,j}^{2,1,\dagger} & D_{\ell,k,j}^{2,2,\dagger} & D_{\ell,k,j}^{2,3,\dagger} \\ D_{\ell,k,j}^{3,1,\dagger} & D_{\ell,k,j}^{3,2,\dagger} & D_{\ell,k,j}^{3,3,\dagger} \end{pmatrix}$$

and the diagonal matrices $D^{p,q,\dagger} = \operatorname{diag}(D_{\ell,k,j}^{p,q,\dagger})$ are defined by

$$\begin{pmatrix} D_{\ell,k,j}^{1,1,\dagger} & D_{\ell,k,j}^{1,2,\dagger} & D_{\ell,k,j}^{1,3,\dagger} \\ D_{\ell,k,j}^{2,1,\dagger} & D_{\ell,k,j}^{2,2,\dagger} & D_{\ell,k,j}^{2,3,\dagger} \\ D_{\ell,k,j}^{3,1,\dagger} & D_{\ell,k,j}^{3,2,\dagger} & D_{\ell,k,j}^{3,3,\dagger} \end{pmatrix} := \begin{pmatrix} D_{\ell,k,j}^{1,1} & D_{\ell,k,j}^{1,2} & D_{\ell,k,j}^{1,3} \\ D_{\ell,k,j}^{2,1} & D_{\ell,k,j}^{2,2} & D_{\ell,k,j}^{2,3} \\ D_{\ell,k,j}^{3,1} & D_{\ell,k,j}^{3,2} & D_{\ell,k,j}^{3,3} \end{pmatrix}^\dagger, \quad \begin{matrix} j = 1, \dots, n_1, \\ k = 1, \dots, n_2, \\ \ell = 1, \dots, n_3. \end{matrix}$$

Proof We only prove part 1. The proof of part 2 is along the same lines. The starting point is Lemma 9.6 and in particular Corollary 9.7. A suitable permutation matrix P yields

$$P^{-1} \begin{pmatrix} D_0^{1,1} & D_0^{1,2} \\ D_0^{1,2} & D_0^{2,2} \end{pmatrix} P = \operatorname{diag} \left(B_{k,j}, \quad \begin{matrix} j = 1, \dots, n_1, \\ k = 1, \dots, n_2 \end{matrix} \right),$$

where

$$B_{k,j} = \begin{pmatrix} D_{0,k,j}^{1,1} & D_{0,k,j}^{1,2} \\ D_{0,k,j}^{1,2} & D_{0,k,j}^{2,2} \end{pmatrix} \in \mathbb{R}^{2 \times 2}.$$

The pseudo-inverses of $B_{k,j}$ are given by Lemma 9.8. $\square$

9.5.4 Inversion of the discrete Navier–Lamé equations

In view of Theorem 9.9, the solution of the linear system in Algorithm 8.2 may be computed explicitly by applying a number of FFTs and by computing the pseudo-inverse of the block diagonal matrix D . The implementation details for this process are summarized in Algorithm 9.10 for dimension $d = 2$ and in Algorithm 9.11 for dimension $d = 3$.

We note that for the particular choices of the stencils S (cf., eqn (9.15) and Table 9.1), the computation of $D^\dagger$ simplifies considerably. For dimension $d = 2$, we have

$$\begin{aligned} D_{0,k,j}^{1,1} &= S_{2,2}^{1,1} + 2 \cos(2\pi(j-1)/n_1) S_{1,2}^{1,1} + 2 \cos(2\pi(k-1)/n_2) S_{2,1}^{1,1}, \\ D_{0,k,j}^{2,2} &= S_{2,2}^{1,1} + 2 \cos(2\pi(k-1)/n_2) S_{1,2}^{1,1} + 2 \cos(2\pi(j-1)/n_1) S_{2,1}^{1,1}, \\ D_{0,k,j}^{1,2} &= D_{0,k,j}^{2,1} = -4 S_{1,1}^{1,2} \sin(2\pi(j-1)/n_1) \sin(2\pi(k-1)/n_2), \end{aligned}$$

and for dimension $d = 3$, we have

$$\begin{aligned} D_{\ell,k,j}^{1,1} &= S_{2,2,2}^{1,1} + 2 \cos(2\pi(j-1)/n_1) S_{1,2,2}^{1,1} + 2 \cos(2\pi(k-1)/n_2) S_{1,2,2}^{1,1} \\ &\quad + 2 S_{2,1,2}^{1,1} (\cos(2\pi(k-1)/n_2) + \cos(2\pi(\ell-1)/n_3)), \\ D_{\ell,k,j}^{2,2} &= S_{2,2,2}^{1,1} + 2 \cos(2\pi(k-1)/n_2) S_{1,2,2}^{1,1} \\ &\quad + 2 S_{2,1,2}^{1,1} (\cos(2\pi(j-1)/n_1) + \cos(2\pi(\ell-1)/n_3)), \\ D_{\ell,k,j}^{3,3} &= S_{2,2,2}^{1,1} + 2 \cos(2\pi(\ell-1)/n_3) S_{1,2,2}^{1,1} \\ &\quad + 2 S_{2,1,2}^{1,1} (\cos(2\pi(j-1)/n_1) + \cos(2\pi(k-1)/n_2)), \\ D_{\ell,k,j}^{1,2} &= D_{\ell,k,j}^{2,1} = -4 S_{1,1,2}^{1,2} \sin(2\pi(j-1)/n_1) \sin(2\pi(k-1)/n_2), \\ D_{\ell,k,j}^{1,3} &= D_{\ell,k,j}^{3,1} = -4 S_{1,1,2}^{1,2} \sin(2\pi(j-1)/n_1) \sin(2\pi(\ell-1)/n_3), \\ D_{\ell,k,j}^{2,3} &= D_{\ell,k,j}^{3,2} = -4 S_{1,1,2}^{1,2} \sin(2\pi(k-1)/n_2) \sin(2\pi(\ell-1)/n_3). \end{aligned}$$

Algorithm 9.10 *Algorithm for solving the discrete two-dimensional Navier–Lamé equations using a standard 2d FFT.*

Compute $\tilde{F}^1 := \text{fft2}(f_1(X))$ and $\tilde{F}^2 := \text{fft2}(f_2(X))$.

For $j = 1, \dots, n_1$, for $k = 1, \dots, n_2$,

$$\tilde{U}_{k,j}^1 = D_{k,j}^{1,1,\dagger} \tilde{F}_{k,j}^1 + D_{k,j}^{1,2,\dagger} \tilde{F}_{k,j}^2,$$

$$\tilde{U}_{k,j}^2 = D_{k,j}^{2,1,\dagger} \tilde{F}_{k,j}^1 + D_{k,j}^{2,2,\dagger} \tilde{F}_{k,j}^2,$$

end.

Compute $U_1 := \text{fft2}^{-1}(\tilde{U}^1)$, $U_2 := \text{fft2}^{-1}(\tilde{U}^2)$.

Algorithm 9.11 *Algorithm for solving the discrete three-dimensional Navier–Lamé equations using a standard 3d FFT.*

For $r = 1, 2, 3$,

compute $\tilde{F}^r := \text{fft3}(f_r(X))$,

end.

For $j = 1, \dots, n_1$, for $k = 1, \dots, n_2$, for $\ell = 1, \dots, n_3$,

solve

$$\begin{pmatrix} \tilde{U}_{\ell,k,j}^1 \\ \tilde{U}_{\ell,k,j}^2 \\ \tilde{U}_{\ell,k,j}^3 \end{pmatrix} = \begin{pmatrix} D_{\ell,k,j}^{1,1} & D_{\ell,k,j}^{1,2} & D_{\ell,k,j}^{1,3} \\ D_{\ell,k,j}^{2,1} & D_{\ell,k,j}^{2,2} & D_{\ell,k,j}^{2,3} \\ D_{\ell,k,j}^{3,1} & D_{\ell,k,j}^{3,2} & D_{\ell,k,j}^{3,3} \end{pmatrix}^{\dagger} \begin{pmatrix} \tilde{F}_{\ell,k,j}^1 \\ \tilde{F}_{\ell,k,j}^2 \\ \tilde{F}_{\ell,k,j}^3 \end{pmatrix},$$

end, end, end.

For $r = 1, 2, 3$,

compute $U^r := \text{fft3}^{-1}(\tilde{U}^r)$,

end.

Theorem 9.12 *Let $F_r \in \mathbb{R}^{n_1 \times \dots \times n_d}$, $r = 1, \dots, d$, be given right hand sides and $N = n_1 \dots n_d$. The numerical complexity of the outlined scheme for the solution of the discrete Navier–Lamé equations*

$$A(U_1, \dots, U_d)^{\top} = (F_1, \dots, F_d)^{\top}$$

with periodic boundary conditions is $\mathcal{O}(N \log N)$.

Proof In addition to the application of a d -dimensional FFT and its inverse, N small linear systems of size d -by- d have to be solved. $\square$

9.6 Solving the Navier–Lamé equations

Using the fixed-point iteration (8.6), and the finite difference approximation of the Navier–Lamé operator with periodic boundary conditions as introduced in the previous section, we end up with the overall algorithm summarized in Algorithm 8.2.

For the computation of the force we use a bi- or tri-linear interpolation scheme; cf., Section 3.1.3. The time consuming part is, however, the solution of the linear system, which has to be done in each iteration. Exploiting the direct solver introduced in this chapter gives an $\mathcal{O}(N \log N)$ implementation. It is worthwhile noticing that iterative schemes may provide an attractive alternative to this solution process. However, using iterative solvers one also introduces an additional inner iteration and computes only an approximation to the solution of the linear system.

9.7 Regularizing the updates of the displacements

A very brief summary of elastic registration reads as follows. Given $u^{(k)}$, compute the force $f(\cdot, u^{(k)}(\cdot))$, and obtain the new update $u^{(k+1)}$ from $u^{(k+1)} = \mathcal{A}^{\dagger} f(\cdot, u^{(k)}(\cdot))$. The final deformation is given by $x + u^{(K)}(x)$, where K denotes the number of iterations needed to fulfill a certain stopping criterion.

Though this approach presents a physically meaningful model of an elastic tissue, it appears to be too restrictive for many image registration problems. The remedy is to use a regularized incremental update. Here, the final deformation is given by $x + \sum_{k=1}^K u^{(k)}(x)$, where $u^{(k+1)} = \mathcal{A}^{\dagger} f(\cdot, u^{(k)}(\cdot))$. Using this idea, the overall elastic deformation is decomposed into K small steps, where the elastic memory of the body is erased from step to step. This variant is used for the examples shown in the following section.

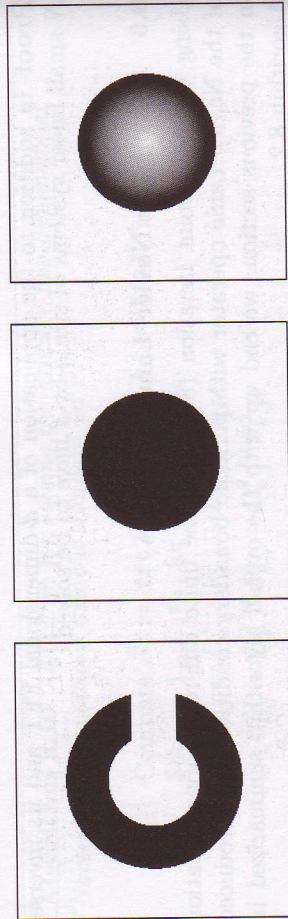
A more general approach which is related to optical flow computation is to introduce a flow $I(x, t)$ and to minimize

$$\mathcal{J}[u] = \int_{\Omega} \int_0^1 \left(\frac{d}{dt} I(x - u(x, t), t) \right)^2 dx + \alpha \mathcal{S}[u] dt$$

where $I(x, 0) = R(x)$, $I(x, 1) = T(x)$, and additional boundary conditions for $I(\partial\Omega, t)$ are to be imposed; cf., Keeling & Ring (2002).

9.8 Elastic registration: • to C

We start with an academic example, given the two images depicted in Fig. 9.7. A MATLAB description of both m -by- m images is given in Table 9.2. A warping

FIG. 9.7 Reference C, template $\bullet$, and shaded template.Table 9.2 MATLAB code for reference C and template $\bullet$.

```
[x,y] = meshgrid(linspace(-1,1,m));
R = zeros(m);
R(x.^2+y.^2>0.33^2 & x.^2+y.^2<0.64^2) = 255;
R(x>0 & abs(y)<0.16) = 0;
T = 0.5^2 - x.^2 - y.^2;
T(T>0) = 255;
```

from $\bullet$ to C is impossible with an elastic deformation, because the images differ by too much.

In Fig. 9.8 we visualize the deformations for the elastic registrations with Dirichlet, Neumann, and periodic boundary conditions. Obviously, the boundary conditions have an impact on the deformations particularly near the boundary but also globally. However, since we assume the images to be embedded completely in a background, these differences are regarded as irrelevant for the overall registration goal.

In Fig. 9.9, we show intermediate results for $k = 0(20)100$. In order to visualize the deformation, the template has been shaded. Note that the computations have been performed on the original images. In this example, the Lamé constants are $\lambda = 0$ and $\mu = 5000$. For this example we have numerical convergence of the scheme after about 100 iterations, cf., Fig. 9.8 for periodic boundary conditions.

9.9 Elastic registration: hands

The next example is slightly more realistic. Figure 9.10 shows two modified X-ray images of human hands. We also show the linearly registered template, the linearly and elastically registered template, and the differences of the original, linearly and elastically deformed template with the reference image. For this example, we chose the Lamé constants $\mu = 500$ and $\lambda = 0$. As is apparent from this figure, the additional elastic registration reduces the image difference considerably. Intermediate registration results are depicted in Fig. 9.11.

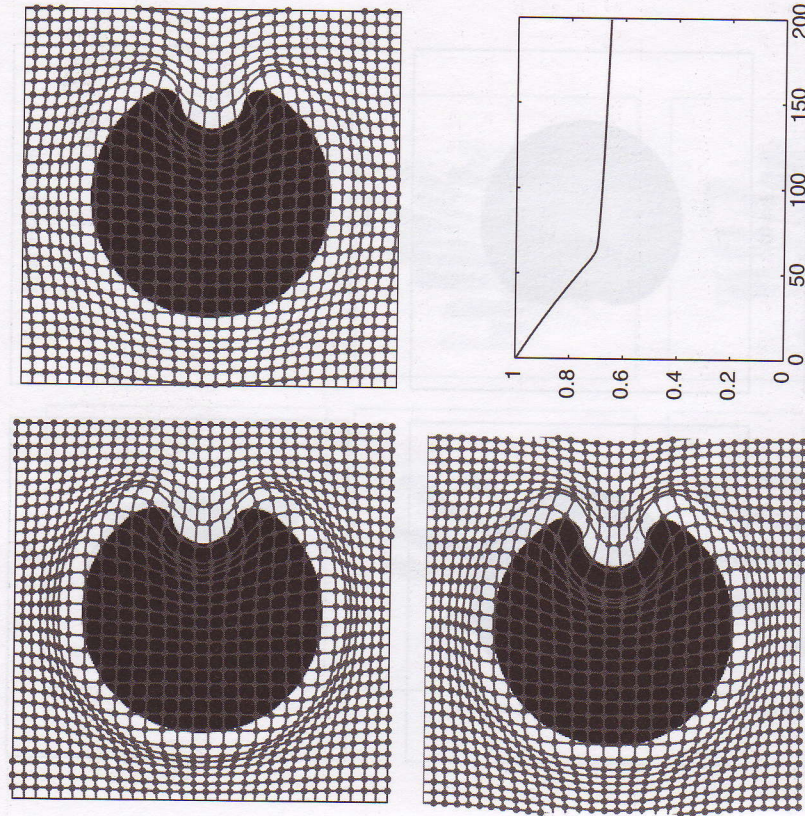


FIG. 9.8 Deformations after elastic registration: (TOP LEFT) with Dirichlet boundary conditions (TOP RIGHT) with Neumann boundary conditions (BOTTOM LEFT) with periodic boundary conditions (BOTTOM RIGHT) convergence history for periodic boundary conditions.

9.10 Elastic registration: brain

An example of a two-dimensional elastic registration has already been presented in Chapter 2, see Fig. 2.4. We now also present a registration of a part of the human brain covering the visual cortex, see Fig. 9.12.

After an appropriate pre-registration as explained in Part I, each section $S^{(\nu)}$, $\nu = 1, \dots, 100$, has been registered with $\varphi^{(\nu)}$, where the $\varphi^{(\nu)}$'s are such that

$$\sum_{\nu=2}^{100} \|S^{(\nu)} \circ \varphi^{(\nu)} - S^{(\nu-1)} \circ \varphi^{(\nu-1)}\|_{L_2(\Omega)}^2 + \sum_{\nu=1}^{100} S^{\text{elas}}[\varphi^{(\nu)}] \longrightarrow \min.$$

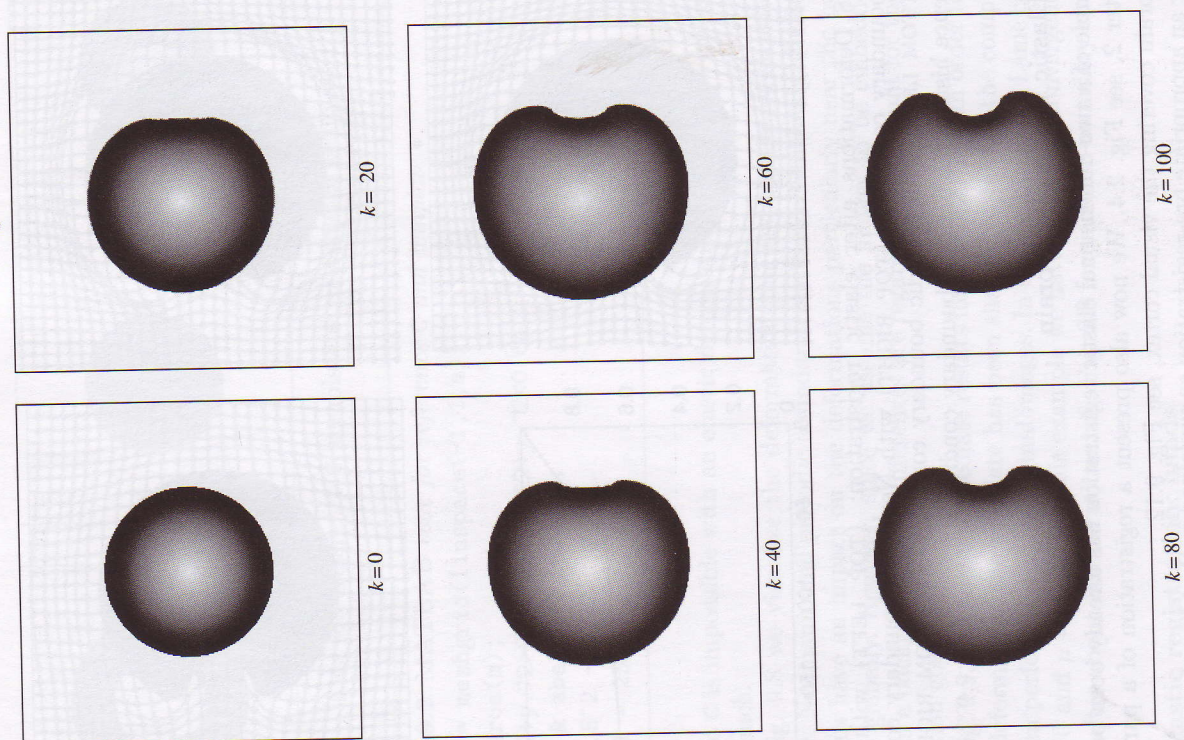


FIG. 9.9 Elastic registration of $\bullet$ to C with periodic boundary conditions, $\mu = 5000$, and $\lambda = 0$; intermediate results for $k = 0(20)100$.

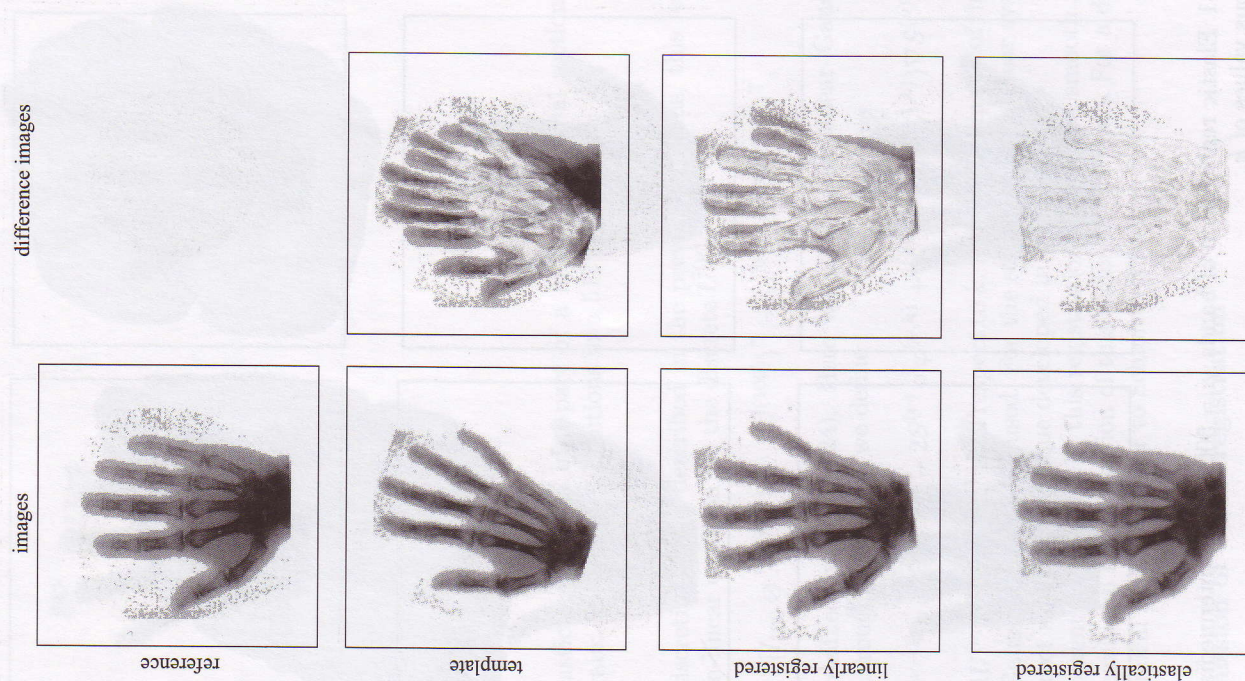


FIG. 9.10 LEFT: Reference, template, linearly registered template, and elastically registered template; RIGHT: difference images.

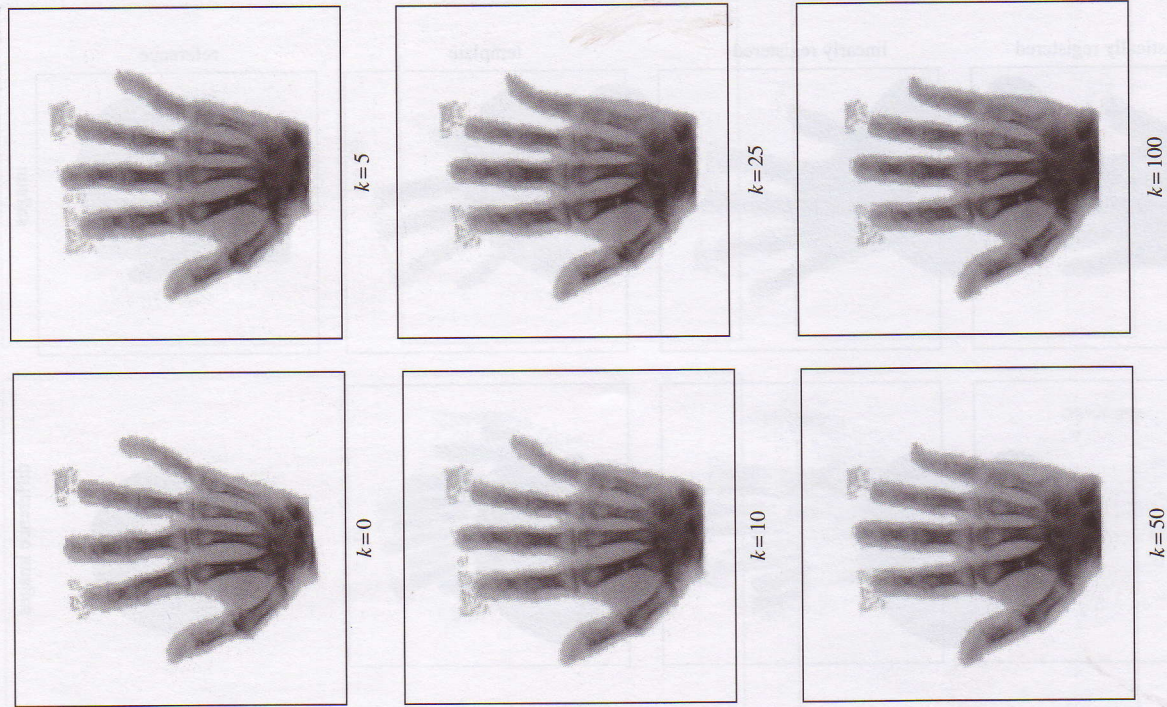


FIG. 9.11 Elastic registration of hands, $\mu = 500$, $\lambda = 0$; intermediate results for various values of k .

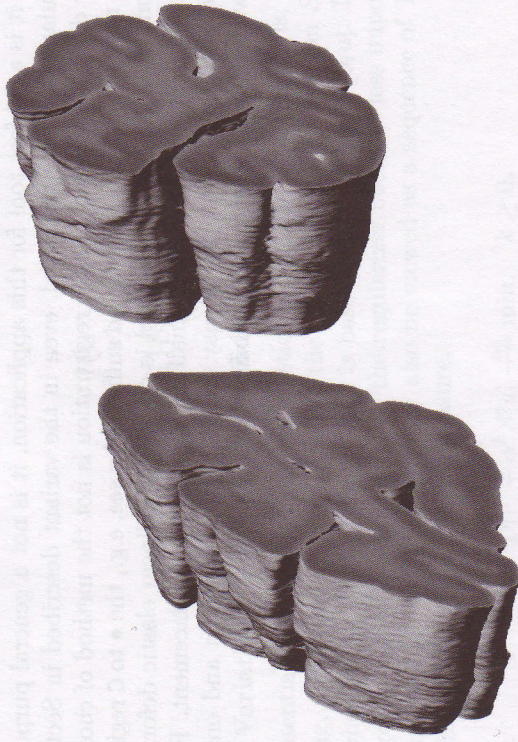


FIG. 9.12 Elastic registration of a part of a histological serial sectioning of a human brain; 100 registered sections in a 512^2 resolution.

After a discretization as described in the previous sections, this leads to a system of non-linear equation for the discrete $\vec{U}^{(\nu)}$,

$$[I_{100} \otimes A](\vec{U}^{(1)}, \dots, \vec{U}^{(100)})^\top = (\vec{F}^{(1)}, \dots, \vec{F}^{(100)})^\top,$$

where A is defined by eqn (9.24). Here we used a non-linear Gauss-Seidel iteration, i.e., roughly speaking we define

$$F^{(\nu,k)} = (S^{(\nu-1)} \circ \varphi^{(\nu-1,k+1)} - 2S^{(\nu)} \circ \varphi^{(\nu,k)} + S^{(\nu+1)} \circ \varphi^{(\nu+1,k)}) \nabla S^{(\nu)} \circ \varphi^{(\nu,k)}$$

and iterate $AU^{(\nu,k+1)} = F^{(\nu,k)}$ with respect to k . A multi-resolution of the images based on a Gaussian pyramid is used. For the solution of the linear systems we apply the direct solution technique developed in this chapter.

As is apparent from Fig. 9.12 this registration results in a smooth and anatomically meaningful reconstruction of this region of the brain. For a discussion of the anatomical issues, we refer to Schmitt (2001).

9.11 Discussion of elastic registration

Elastic registration presents an interesting registration tool. The driving force of the registration is the minimization of a distance measure and the transformation is restricted to the class of elastic deformations.

This technique seems to be appropriate for the registration of images arising from a sectioning of an elastic material, e.g., within the HNSP (cf., Chapter 2).

Though it is a perfect tool for this application, it is not a general purpose tool. The regularizer is very restricted, even in the variant described in Section 9.7. For very dissimilar objects, elastic registration is not the method of choice. The distance measure may not be reduced sufficiently; see, e.g., the $\bullet$ to $\mathbb{C}$ registrations in Section 9.8. Note that the method is designed for linear elastic deformations only and implicitly assumes only small changes of the displacement. Relaxing the Lamé constants by too much may violate this assumption and one might end up with a non-diffeomorphic transformation.

10

FLUID REGISTRATION

10.1 Variational motivation of fluid registration

The essential difference between elastic and fluid registration is, roughly speaking, that for elastic registration the regularization is based on $\mathcal{S}^{\text{elas}}[u] = \mathcal{P}[u]$, whereas in the fluid approach it is based on $\mathcal{S}^{\text{fluid}}[u] = \mathcal{P}[\partial_t u]$. Thus, if u converges to a steady-state solution, the regularizer becomes less important. In this context, the fluid approach may also be viewed as a sequence of Tychonov regularizations for the distance measure,

$$\mathcal{D}[u] + \alpha^k \mathcal{S}[u] \xrightarrow{u} \min, \quad k \geq 0,$$

where α^k goes to zero.

From Theorem 9.4 we obtain

$$f(\cdot, u(\cdot, \cdot)) = \mu \Delta v + (\lambda + \mu) \nabla \operatorname{div} v, \quad v = \partial_t u,$$

as a characterizing equation for fluid registration.

In contrast to the elastic model the strain in a fluid model depends on the rate of change. Elastic models are characterized by a spatial smoothing of the displacement field. In contrast, fluid models are characterized by a spatial smoothing of the velocity field. Thus, in principle, any displacement can be obtained given enough time: the internal stresses disappear eventually.

10.2 Physical motivation of fluid registration

In this section we present a coherent description of the elastic and fluid continuum models of the deformation of a body, see also Christensen (1994) or Bro-Nielsen (1996). Note that the fluid model is also called *visco-elastic*; cf., e.g., Wollny & Kruggel (2000).

The transformation $\varphi : \mathbb{R}^d \times \mathbb{R} \rightarrow \mathbb{R}$ (cf., Section 9.1.1) is now time dependent and smooth. For any time t , $\varphi(\cdot, t)$ has to be bijective and to preserve the orientation of the tissue. With the displacement $u : \mathbb{R}^d \times \mathbb{R} \rightarrow \mathbb{R}^d$ (cf., eqn (8.1)) the location of a particle P can be described by

$$\tilde{x} = \varphi(x, t) = x + u(x, t),$$

where $x := \varphi(P, 0)$ is the initial position of the particle and $\tilde{x}$ its current position at a fixed time t . With $\varphi(\cdot, t)$ being bijective, we can make use of its inverse