Chapter 10
Diffeomorphic Matching

10.1 Linearized Deformations

A standard way to ensure the existence of a smooth solution of a matching problem is to add a penalty term in the matching functional. This term would complete (9.1) to form

$$E_{I,I'}(\varphi) = \rho(\varphi) + D(\varphi \cdot I, I'). \tag{10.1}$$

A large variety of such methods have been designed, in approximation theory, statistics and signal processing for solving ill-posed problems. The simplest (and typical) form of penalty function is

$$\rho(\varphi) = \|\varphi - \mathrm{id}\|_H^2$$

for some Hilbert (or Banach) space of functions. Some more complex functions of $\varphi - \mathrm{id}$ may also be designed, related to energies of non-linear elasticity (see, among others [13, 27, 28, 89, 123, 144, 237]). Such methods may be called "small deformation" methods because they work on the deviation of $u = \varphi - \mathrm{id}$, and controlling the size or smoothness of u alone is most of the time not enough to guarantee that φ is a diffeomorphism (unless u is small, as we have seen in Sect. 7.1). There is, in general, no way of proving the existence of a solution of the minimization problem within some group of diffeomorphisms G, unless some restrictive assumptions are made on the objects to be matched.

Our focus here is on diffeomorphic matching. Because of this, we shall not detail many of these methods. However, it is interesting to note that these functionals also have a Eulerian gradient within an RKHS of vector fields with a smooth enough kernel, and can therefore be minimized using (9.7). We illustrate this with the following example, in which we skip the proper justification of the existence of derivatives.

L. Younes, *Shapes and Diffeomorphisms*, Applied Mathematical Sciences 171,
https://doi.org/10.1007/978-3-662-58496-5_10

Consider the function $\rho(\varphi) = \int_{\mathbb{R}^d} |d\varphi(x) - \mathrm{Id}|^2 \, dx$, where the matrix norm is

$$|A|^2 = \mathrm{trace}(A^T A) = \sum_{i,j} a_{ij}^2$$

(Hilbert–Schmidt norm). Letting $u = \varphi - \mathrm{id}$, we have

$$(d\rho(\varphi) \mid h) = 2 \int_{\mathbb{R}^d} \mathrm{trace}(du^T dh) dx = -2 \int_{\mathbb{R}^d} \Delta u^T h dx,$$

where Δu is the vector formed by the Laplacian of the coordinates of u (recall that we assume that $u = 0$ at infinity). This implies that (given that $\Delta u = \Delta \varphi$)

$$\left(\bar{\partial}\rho(\varphi) \mid h\right) = -2 \int_{\Omega} \Delta \varphi^T h \circ \varphi dx$$

and

$$\overline{\nabla}_{\varphi}^{V} \rho(\cdot) = -2 \int_{\Omega} K(\cdot, \varphi(x)) \Delta \varphi(x) dx. \tag{10.2}$$

This provides a regularized greedy image-matching algorithm, which includes a regularization term (a similar algorithm may easily be written for point matching).

Algorithm 2 The following procedure is an Eulerian gradient descent, on V, for the energy

$$E_{I,I'}(\varphi) = \int_{\mathbb{R}^d} |d\varphi(x) - \mathrm{id}|^2 \, dx + \frac{1}{\sigma^2} \int_{\mathbb{R}^d} \left| I \circ \varphi^{-1}(x) - I'(x) \right| dx.$$

Start with an initial $\varphi_0 = \mathrm{id}$ and solve the differential equation

$$\begin{aligned} \partial_t \varphi(t, y) &= -2 \int_{\Omega} K(\varphi(t, y), \varphi(t, x)) \Delta \varphi(t, x) dx & (10.3)\\ &+ \frac{2}{\sigma^2} \int_{\Omega} (J(t, x) - I'(x)) K(\varphi(t, y), x) \nabla J(t, x) dx & (10.4) \end{aligned}$$

with $J(t, \cdot) = I \circ \varphi(t)^{-1}(\cdot)$.

This algorithm, which, like the previous greedy procedures, has the fundamental feature of providing a smooth flow of diffeomorphisms to minimize the matching functional, suffers from the same limitations as its predecessors concerning its limit behavior, which are essentially due to the fact that the variational problem itself is not well-posed; minimizers may not exist, and when they exist they are not necessarily diffeomorphisms. In order to ensure the existence of, at least, homeomorphic solutions, the energy must include terms that must not only prevent $d\varphi$ from being too large, but also from being too small (or its inverse from being too large). In [90], the following regularization is proved to ensure the existence of homeomorphic solutions:

$$\delta(\varphi) = \int_\Omega (a\|d\varphi\|^p + b\|\mathrm{Adj}(d\varphi)\|^q + c(\det d\varphi)^r + d(\det d\varphi)^{-s})dx \tag{10.5}$$

under some assumptions on p, q, r and s, namely $p, q > 3$, $r > 1$ and $s > 2q/(q-3)$.

10.2 The Monge–Kantorovitch Problem

We briefly discuss in this section the mass transfer problem, which is, under some assumptions, a diffeomorphic method for matching probability densities, i.e., positive functions on $\mathbb{R}^d$ with integral equal to 1. Consider such a density, ζ, and a diffeomorphism φ on $\mathbb{R}^d$. If an object has density ζ, the mass included in an infinitesimal volume dx around x is $\zeta(x)dx$. Now, if each point x in the object is transported to the location $y = \varphi(x)$, the mass of a volume dy around y is the same as the mass of the volume $\varphi^{-1}(dy)$ around $x = \varphi^{-1}(y)$, which is $\zeta \circ \varphi^{-1}(y)|\det(d(\varphi^{-1}))(y)|dy$ (this provides a physical interpretation of Proposition 9.5).

Given two densities ζ and ζ', the optimal mass transfer problem consists in finding a diffeomorphism φ with minimal cost such that $\zeta' = \zeta \circ \varphi^{-1}|\det(d(\varphi^{-1}))|$. The cost associated to φ in this context is related to the distance along which the transfer is made, measured by a function $\rho(x, \varphi(x))$. The total cost comes after summing over the transferred mass, yielding

$$E(\varphi) = \int_\Omega \rho(x, \varphi(x))\zeta(x)dx.$$

The mass transfer problem now is to minimize E over all φ's such that $\zeta' = \zeta \circ \varphi^{-1}|\det(d(\varphi^{-1}))|$. The problem is slightly different from the matching formulations that we discuss in the other sections of this chapter, because the minimization is associated to exact matching.

It is very interesting that this apparently very complex and highly nonlinear problem can be reduced to linear programming, albeit infinite-dimensional. Let us first consider a more general formulation. Instead of looking for a one-to-one correspondence $x \mapsto \varphi(x)$, one can decide that the mass in a small neighborhood of x is dispatched over all Ω with weights $y \mapsto q(x, y)$, where $q(x, y) \geq 0$ and $\int_\Omega q(x, y)dy = 1$. We still have the constraint that the mass density arriving at y is $\tilde{\zeta}(y)$, which gives

$$\int_\Omega \zeta(x)q(x, y)dx = \tilde{\zeta}(y).$$

The cost now has the simple expression (linear in q)

$$E = \int_{\Omega^2} \rho(x, y)\zeta(x)q(x, y)dxdy.$$

The original formulation can be retrieved by letting $q(x, y)dy \to \delta_{\varphi(x)}(y)$ (i.e., pass to the limit $\sigma = 0$ with $q(x, y) = \exp(-|y - \varphi(x)|^2/2\sigma^2)/(2\pi\sigma^2)^{d/2}$).

If we write $g(x, y) = \zeta(x)q(x, y)$, this relaxed problem is clearly equivalent to minimizing

$$E(g) = \int_{\Omega^2} \rho(x, y)g(x, y)dxdy$$

subject to the constraints $g(x, y) \geq 0$, $\int g(x, y)dy = \zeta(x)$ and $\int g(x, y)dx = \tilde{\zeta}(y)$. In fact, the natural formulation of this problem uses measures instead of densities: given two probability measures μ and $\tilde{\mu}$ on Ω, minimize

$$E(\nu) = \int_{\Omega^2} \rho(x, y)\nu(dx, dy)$$

subject to the constraints that the marginals of ν are μ and $\tilde{\mu}$. This provides the Wasserstein distance between μ and $\tilde{\mu}$, associated to the transportation cost ρ. Note that this formulation generalizes the computation of the Wasserstein distance (9.24) between discrete measures.

This problem is much nicer than the original one, since it is a linear programming problem. The theory of convex optimization (that we only apply formally in this infinite-dimensional context; see [44] for rigorous proofs) implies that it has an equivalent dual formulation which is: maximize

$$F(h) = \int_{\Omega} h d\mu + \int_{\Omega} \tilde{h}\tilde{\mu}$$

subject to the constraint that, for all $x, y \in \Omega$, $h(x) + \tilde{h}(y) \leq \rho(x, y)$.

The duality equivalence means that the maximum of F coincides with the minimum of E. The solutions are, moreover, related by duality conditions (the KKT conditions) that imply that ν must be supported by the set

$$A = \left\{(x, y) : h(x) + \tilde{h}(y) = \rho(x, y)\right\}. \tag{10.6}$$

For the dual problem, one is obviously interested in making h and $\tilde{h}$ as large as possible. Given h, one should therefore choose $\tilde{h}$ as

$$\tilde{h}(y) = \sup_x (\rho(x, y) - h(x)),$$

so that the set in (10.6) is exactly the set of (y^*, y) where y^* is a point that achieves the maximum of $\rho(x, y) - h(x)$.

The situation is particularly interesting when $\rho(x, y) = |x - y|^2/2$. In this situation,

$$\tilde{h}(y) = \frac{y^2}{2} + \sup_x \left(x^T y + \frac{x^2}{2} - h(x) \right).$$

From this equation, it is natural to introduce the auxiliary functions $s(x) = h(x) - x^2/2$ and $\tilde{s}(y) = \tilde{h}(y) - y^2/2$. Using these functions, the set A in (10.6) becomes

$$A = \left\{ (x, y) : s(x) + \tilde{s}(y) = x^T y \right\},$$

with $\tilde{s}(y) = \sup_x (x^T y - s(x))$. Because the latter is a supremum of linear functions, we obtain the fact that $\tilde{s}$ is convex, and so is s by symmetry; $\tilde{s}$ is in fact what is called the convex conjugate of s, denoted $\tilde{s} = s^*$. Convex functions are almost everywhere differentiable, and, in order that $(x, y) \in A$, x must maximize $u \mapsto u^T y - s(u)$, which implies that $y = \nabla s(x)$. So, the conclusion is that, whenever s is the solution of the dual problem, the solution of the primal problem is provided by $y = \nabla s(x)$. This shows that the relaxed mass transport problem has the same solution as the initial one, with $\varphi = \nabla s$, s being a convex function. That φ is invertible is obvious by symmetry: $\varphi^{-1} = \nabla \tilde{s}$.

This result is fundamental, since it is the basis for the construction of a numerical procedure for the solution of the mass transport problem in this case. Introduce a time-dependent vector field $v(t, \cdot)$ and the corresponding flow of diffeomorphisms φ_{0t}^v. Let $h(t, \cdot) = \det(d\varphi_{t0}^v)\, \zeta \circ \varphi_{t0}^v$. Then

$$\det(d\varphi_{0t}^v)\, h(t) \circ \varphi_{0t}^v = \zeta.$$

The time derivative of this equation yields

$$\partial_t h + \operatorname{div}(hv) = 0. \tag{10.7}$$

We have the following theorem [34].

Theorem 10.1 *Consider the following energy:*

$$G(v) = \int_0^1 \int_\Omega h(t, x) |v(t, x)|^2 dx dt$$

and the variational problem: minimize G subject to the constraints $h(0) = \zeta$, $h(1) = \tilde{\zeta}$ and (10.7). If v is the solution of the above problem, then φ_{01}^v solves the optimal mass transport problem.

Proof Indeed, in G, we can make the change of variables $x = \varphi_{0t}(y)$, which yields

$$G(v) = \int_0^1 \int_\Omega \zeta(y) \left| v(t, \varphi_{0t}^v(y)) \right|^2 dy dt$$

$$\begin{aligned}
&= \int_{\Omega} \zeta(y) \int_0^1 \left|\partial_t \varphi_{0t}^v\right|^2 dt \\
&\geq \int_{\Omega} \zeta(y) \left|\varphi_{01}^v(y) - y\right|^2 dy.
\end{aligned}$$

So the minimum of G is always larger than the minimum of E. If φ solves the mass transport problem, then one can take $v(t, x)$ such that $\varphi_{0t}^v(x) = (1-t)x + t\varphi(x)$, which is a diffeomorphism [190] and achieves the minimum of G. □

We refer to [34] for a numerical algorithm that computes the optimal φ. Note that $\rho(x, y) = |x - y|^2$ is not the only transportation cost that can be used in this context, but that others (like $|x - y|$, which is not strictly convex in the distance) may fail to provide diffeomorphic solutions. Important developments on this subject can be found in [49, 119, 296].

We now discuss methods that are both diffeomorphic and metric (i.e., they relate to a distance). They also rely on the representation of diffeomorphisms using flows of ordinary differential equations.

10.3 Optimizing Over Flows

We return in this section to the representation of diffeomorphisms with flows of ordinary differential equations (ODEs) and describe how this representation can be used for diffeomorphic registration. Instead of using a norm to evaluate the difference between φ and the identity mapping, we now consider, as a regularizing term, the distance d_V that was defined in Sect. 7.2.6. More precisely, we set

$$\rho(\varphi) = \frac{1}{2} d_V(\mathrm{id}, \varphi)^2$$

and henceforth restrict the matching to diffeomorphisms belonging to Diff_V.

In this context, we have the following important theorem:

Theorem 10.2 *Let V be a Hilbert space embedded in $C_0^{p+1}(\Omega, \mathbb{R}^d)$ so that* $\mathrm{Diff}_V \subset \mathrm{Diff}_0^{p,\infty}$. *Assume that the functional $U : \mathrm{Diff}_0^{p,\infty} \mapsto \mathbb{R}$ is bounded from below and continuous for the (p, ∞)-compact topology. Then, there exists a minimizer of*

$$E(\varphi) = \frac{1}{2} d_V(\mathrm{id}, \varphi)^2 + U(\varphi) \tag{10.8}$$

over Diff_V.

(The (p, ∞)-compact topology is defined just after Theorem 7.13.)

Proof E has an infimum E_{min} over Diff_V, since it is bounded from below. We need to show that this infimum is also a minimum, i.e., that it is achieved at some $\varphi \in \mathrm{Diff}_V$.

We first use the following lemma (recall that we have denoted by $\mathcal{X}_V^1$ (resp. $\mathcal{X}_V^2$) the set of time-dependent vector fields on Ω with integrable (resp. square integrable) V-norm over [0, 1]):

Lemma 10.3 *Minimizing $E(\varphi) = d(\mathrm{id}, \varphi)^2/2 + U(\varphi)$ over Diff_V is equivalent to minimizing the function*

$$\tilde{E}(v) = \frac{1}{2}\int_0^1 \|v(t)\|_V^2\, dt + U(\varphi_{01}^v) \tag{10.9}$$

over $\mathcal{X}_V^2$.

Let us prove this lemma. For $v \in \mathcal{X}_V^2$, we have, by definition of the distance

$$d_V(\mathrm{id}, \varphi_{01}^v)^2 \leq \int_0^1 \|v(t)\|_V^2\, dt,$$

which implies $E(\varphi_{01}^v) \leq \tilde{E}(v)$. This obviously implies that $\inf_{\mathrm{Diff}_V} E(\varphi) \leq \tilde{E}(v)$, and since this is true for all $v \in \mathcal{X}_V^2$, we have $\inf E \leq \inf \tilde{E}$. Now, assume that φ is such that $E(\varphi) \leq \inf E + \varepsilon/2$. Then, by definition of the distance, there exists a v such that $\varphi = \varphi_{01}^v$ and

$$\int_0^1 \|v(t)\|_V^2\, dt \leq d_V(\mathrm{id}, \varphi)^2 + \varepsilon,$$

which implies that

$$\tilde{E}(v) \leq E(\varphi) + \varepsilon/2 \leq \inf E + \varepsilon,$$

so that $\inf E \leq \inf \tilde{E}$.

We therefore have $\inf E = \inf \tilde{E}$. Moreover, if there exists a v such that $\tilde{E}(v) = \min \tilde{E} = \inf E$, then, since we know that $E(\varphi_{01}^v) \leq \tilde{E}$, we must have $E(\varphi_{01}^v) = \min E$. Conversely, if $E(\varphi) = \min E$, by Theorem 7.22, $E(\varphi) = E(\varphi_{01}^v)$ for some v and this v must achieve the infimum of $\tilde{E}$, which proves the lemma.

This lemma shows that it suffices to study the minimizers of $\tilde{E}$. Now, as done in the proof of Theorem 7.22, one can find, by taking a subsequence of a minimizing sequence, a sequence v^n in $\mathcal{X}_V^2$ which converges weakly to some $v \in \mathcal{X}_V^2$ and $\tilde{E}(v^n)$ tends to E_{min}. Because

$$\liminf \int_0^1 \left\|v^n(t)\right\|_V^2 dt \geq \int_0^1 \|v(t)\|_V^2\, dt$$

and because weak convergence in $\mathcal{X}_2^1$ implies convergence of the flow in the (p, ∞)-compact topology (Theorem 7.13) we also have $U(\varphi_{01}^{v^n}) \to U(\varphi_{01}^v)$, so that $\tilde{E}(v) = E_{min}$ and v is a minimizer. □

The general problem of minimizing functionals such as (10.9) has been called "large deformation diffeomorphic metric mapping", or LDDMM. The first algorithms were introduced for this purpose in the case of landmark matching [159] and image matching [32] (these papers were preceded by theoretical developments in [93, 278, 283]). The following sections describe these algorithms, and other that were recently proposed.

10.4 Euler–Lagrange Equations and Gradient

10.4.1 Gradient: Direct Computation

We now detail the computation of the gradient for energies such as (10.8). As remarked in the proof of Theorem 10.2, the variational problem which has to be solved is conveniently expressed as a problem over $\mathcal{X}^2_V$. The function which is minimized over this space takes the form

$$E(v) = \frac{1}{2}\int_0^1 \|v(t)\|^2_V \, dt + U(\varphi^v_{01}).$$

Assume that V is embedded in $C_0^{p+1}(\Omega, \mathbb{R}^d)$ and that U is differentiable on $\mathrm{Diff}_0^{p,\infty}$. Then Theorem 7.12 and the chain rule implies that E is differentiable on $\mathcal{X}^2_V$ with

$$(dE(v) \mid h) = \big\langle v\,,\, h\big\rangle_{\mathcal{X}^2_V(\Omega)} + \big(dU(\varphi^v_{01}) \,\big|\, \partial_v \varphi^v_{01}\, h\big),$$

where $\partial_v \varphi^v_{01}\, h$ is given in Theorem 7.12.

We now identify the gradient of E for the Hilbert structure of $\mathcal{X}^2_V$. This gradient is a function, denoted $\nabla^V E : v \mapsto \nabla^V E(v) \in \mathcal{X}^2_V$, that satisfies

$$(dE(v) \mid h) = \big\langle \nabla^V E(v)\,,\, h\big\rangle_{\mathcal{X}^2_V} = \int_0^1 \big\langle \nabla^V E(v)(t)\,,\, h(t)\big\rangle_V dt$$

for all v, h in $\mathcal{X}^2_V$.

Since the set V is fixed in this section, we will drop the exponent from the notation, and simply refer to the gradient $\nabla E(v)$. Note that this is different from the Eulerian gradient we have dealt with before; ∇E now represents the usual gradient of a function defined over a Hilbert space. One important thing to keep in mind is that the gradient we define here is an element of $\mathcal{X}^2_V$, henceforth a time-dependent vector field, whereas the Eulerian gradient was an element of V (a vector field on Ω). Theorem 10.5 relates the two (and allows us to reuse the computations that were made in Chap. 9). For this, we need to introduce the following operation of diffeomorphisms acting on vector fields.

Definition 10.4 Let φ be a diffeomorphism of Ω and v a vector field on Ω. We denote by $\mathrm{Ad}_\varphi v$ the vector field on Ω defined by

$$\mathrm{Ad}_\varphi v(x) = (d\varphi\, v) \circ \varphi^{-1}(x). \tag{10.10}$$

Ad_φ is called the adjoint representation of φ.

If $\varphi \in \mathrm{Diff}^{p+1,\infty}(\Omega)$, then an application of Lemma 7.3 and the Leibnitz formula implies that $\mathrm{Ad}_\varphi v \in C_0^p(\Omega, \mathbb{R}^d)$ as soon as $v \in C_0^p(\Omega, \mathbb{R}^d)$ and more precisely that Ad_φ is a bounded linear operator from $C_0^p(\Omega, \mathbb{R}^d)$ to itself. We can therefore define its conjugate on $C_0^p(\Omega, \mathbb{R}^d)^*$, with $\mathrm{Ad}_\varphi^* \rho$ given by

$$\big(\mathrm{Ad}_\varphi^* \rho \,\big|\, v\big) = \big(\rho \,\big|\, \mathrm{Ad}_\varphi v\big) \tag{10.11}$$

for $\rho \in C_0^p(\Omega, \mathbb{R}^d)^*$, $v \in C_0^p(\Omega, \mathbb{R}^d)$. Note that $\mathrm{Ad}_\varphi^* \rho$ is, a fortiori, in V^*, because V is continuously embedded in $C_0^{p+1}(\Omega, \mathbb{R}^d)$.

Let $\mathbb{L} : V \to V^*$ denote the duality operator on V and $V^{(r)}$ denote the set of vector fields $v \in V$ such that $\mathbb{L}v \in C_0^r(\Omega, \mathbb{R}^d)^*$ (for $r \leq p+1$). Then, for $v \in V^{(p)}$, we can define, with $\mathbb{K} = \mathbb{L}^{-1}$,

$$\mathrm{Ad}_\varphi^T v = \mathbb{K}(\mathrm{Ad}_\varphi^* \mathbb{L} v). \tag{10.12}$$

This is well-defined, because, by construction, $\mathrm{Ad}_\varphi^* \mathbb{L} v \in C_0^p(\Omega, \mathbb{R}^d)^* \subset V^*$. We have in particular, for $v \in V^{(p)}$ and $w \in V$,

$$\big\langle \mathrm{Ad}_\varphi^T v\,,\ w\big\rangle_V = \big(\mathrm{Ad}_\varphi^* \mathbb{L} v \,\big|\, w\big) = \big(\mathbb{L} v \,\big|\, \mathrm{Ad}_\varphi w\big).$$

Recall that the Eulerian derivative of U is defined by

$$\big(\bar{\partial} U(\varphi) \,\big|\, w\big) = (dU(\varphi) \mid w \circ \varphi).$$

Using Theorem 7.12, we have

$$\partial_v \varphi_{01}^v\, h = \int_0^1 (d\varphi_{u1}^v h(u)) \circ \varphi_{0u}^v\, du = \int_0^1 (\mathrm{Ad}_{\varphi_{u1}^v} h(u)) \circ \varphi_{01}^v\, du$$

so that

$$\begin{aligned}\big(dU(\varphi_{01}^v) \,\big|\, \partial_v \varphi_{01}^v\, h\big) &= \Big(\bar{\partial} U(\varphi_{01}^v) \Big| \int_0^1 (\mathrm{Ad}_{\varphi_{u1}^v} h(u))\, du\Big) \\ &= \int_0^1 \big(\bar{\partial} U(\varphi_{01}^v) \,\big|\, \mathrm{Ad}_{\varphi_{u1}^v} h(u)\big)\big)\, du.\end{aligned}$$

With this notation, we have the following theorem.

Theorem 10.5 *Assume that V is continuously embedded in $C_0^{p+1}(\Omega, \mathbb{R}^d)$ and that U is continuously differentiable on* $\mathrm{Diff}_0^{p,\infty}$. *Then, the $\mathcal{X}_V^2$ gradient of $\tilde{U} : v \mapsto U(\varphi_{01}^v)$ is given by the formula*

$$\nabla \tilde{U}(v)(t) = \mathbb{K}\mathrm{Ad}^*_{\varphi_{t1}^v} \bar{\partial} U(\varphi_{01}^v) = \mathrm{Ad}^T_{\varphi_{t1}^v} \overline{\nabla} U(\varphi_{01}^v). \tag{10.13}$$

This important result has the following simple consequences.

Proposition 10.6 *Let U satisfy the assumptions of Theorem 10.5. If $v \in \mathcal{X}_V^2$ is a minimizer of*

$$\tilde{E}(v) = \frac{1}{2} \int_0^1 \|v(t)\|_V^2 dt + U(\varphi_{01}^v), \tag{10.14}$$

then, for all t

$$v(t) = \mathrm{Ad}^T_{\varphi_{t1}^v} v(1), \tag{10.15}$$

with $v(1) = -\overline{\nabla}^V U(\varphi_{01}^v)(x)$. In particular, v is a continuous function of t and $v(t) \in V^{(p)}$ for all t.

Corollary 10.7 *Under the same conditions on U, if $v \in \mathcal{X}_V^2$ is a minimizer of*

$$\tilde{E}(v) = \frac{1}{2} \int_0^1 \|v_t\|_V^2 dt + U(\varphi_{01}^v)$$

then, for all t,

$$v_t = \mathrm{Ad}^T_{\varphi_{t0}^v} v_0, \tag{10.16}$$

with $v_0 \in V^{(p)}$.

Proposition 10.6 is a direct consequence of Theorem 10.5. For the corollary, we need to use the fact that $\mathrm{Ad}_\varphi \mathrm{Ad}_\psi = \mathrm{Ad}_{\varphi \circ \psi}$, which can be checked by direct computation, and write

$$v_t = \mathrm{Ad}^T_{\varphi_{t1}^v} v_1 = \mathrm{Ad}^T_{\varphi_{t1}^v} \mathrm{Ad}^T_{\varphi_{10}^v} v_0 = (\mathrm{Ad}_{\varphi_{10}^v} \mathrm{Ad}_{\varphi_{t1}^v})^T v_0 = \mathrm{Ad}^T_{\varphi_{t0}^v} v_0.$$

Equations $v_t = \mathrm{Ad}^T_{\varphi_{t0}^v} v_0$ and $v_1 = -\overline{\nabla}^V U(\varphi_{01}^v)(x)$ together are equivalent to the Euler–Lagrange equations for $\tilde{E}$ and will lead to interesting numerical procedures. Equation (10.16) is a cornerstone of the theory. It describes a general mechanical property called the conservation of momentum, to which we will return later.

10.4.2 Derivative Using Optimal Control

We can also apply the Pontryagin maximum principle (see Appendix D) to obtain an alternative expression of the optimality conditions and gradient. Indeed, we can repeat the construction made in Sect. 7.2.2 with a slightly different notation, letting $f(\omega, v) = v \circ (\mathrm{id} + \omega)$, defined over $C_0^p(\Omega, \mathbb{R}^d) \times V$. With $g(\omega, v) = \|v\|_V^2$, we are in the framework described in Sect. D.3.1, leading to Theorem D.7, where ω represents the state and v is the control. Introducing a co-state μ, define the Hamiltonian

$$H_v(\mu, \omega) = \left(\mu \mid v \circ (\mathrm{id} + \omega)\right) - \|v\|_V^2/2.$$

Letting $\xi_\varphi : v \mapsto v \circ \varphi$ from V to $C_0^p(\Omega, \mathbb{R}^d)$, we obtain the fact that an optimal solution must satisfy (with $\varphi = \mathrm{id} + \omega$), for some $\mu : [0, 1] \to C_0^p(\Omega, \mathbb{R}^d)^*$

$$\begin{cases} \partial_t \varphi(t) = v(t) \circ \varphi(t) \\ \left(\partial_t \mu(t) \mid h\right) = -\left(\mu(t) \mid dv(t) \circ \varphi(t)\, h\right), \quad \forall h \in C_0^p(\Omega, \mathbb{R}^d) \\ \mathbb{L}v(t) = \xi_{\varphi(t)}^* \mu(t) \end{cases} \tag{10.17}$$

with $\varphi(0) = \mathrm{id}$ and $\mu(1) = -dU(\varphi(1))$. One can check that the second equation is equivalent to

$$\left(\mu(t) \mid h\right) = \left(\mu(0) \,\middle|\, (d\varphi(t))^{-1} h\right),$$

which is Corollary 10.7 expressed in terms of the co-state μ. Applying Eq. (D.12), we obtain

$$d\tilde{E}(v)(t) = -\xi_{\varphi(t)}^* \mu(t) + 2\mathbb{L}v(t), \tag{10.18}$$

where φ and μ satisfy the first two equations of (10.17).

10.4.3 An Alternative Form Using the RKHS Structure

The conjugate of the adjoint can be put into a form explicitly involving the reproducing kernel of V. Before detailing this, we introduce a notation that will be used throughout this chapter. If ρ is a linear form on function spaces, we have been denoting by $\left(\rho \mid v\right)$ the result of ρ applied to v. In the formulas that will come, we will need to emphasize the variable on which v depends, and we will use the alternative notation $\left(\rho \mid v(x)\right)_x$ to denote the same quantity. Thus,

$$\rho(v) = \left(\rho \mid v\right) = \left(\rho \mid v(x)\right)_x.$$

In particular, when v depends on two variables, the notation $\left(\rho \mid v(x, y)\right)_x$ will represent ρ applied to the function $x \mapsto v(x, y)$ with y considered as constant.

We still assume that V is continuously embedded in $C_0^{p+1}(\Omega, \mathbb{R}^d)$. Then, the following theorem holds.

Theorem 10.8 *Assume that $\varphi \in C_0^{q+1}(\Omega, \mathbb{R}^d)$ and $\rho \in C_0^r(\Omega, \mathbb{R}^d)^*$, with $r = \min(p + 1, q)$. Let $v = \mathbb{K}\rho$ and $(e_1, \dots, e_d)$ be an orthonormal basis of $\mathbb{R}^d$. Then, for $y \in \Omega$, we have*

$$\mathrm{Ad}_\varphi^T v(y) = \sum_{i=1}^d \big(\rho \;\big|\; \mathrm{Ad}_\varphi(K(x,y)e_i)\big)_x e_i, \tag{10.19}$$

where K is the reproducing kernel of V.

Proof For $b \in \mathbb{R}^d$, we have

$$\begin{aligned} b^T \mathrm{Ad}_\varphi^T v(y) &= \big\langle \mathrm{Ad}_\varphi^T v\,,\; K(\cdot,y)b\big\rangle_V \\ &= \big\langle v\,,\; \mathrm{Ad}_\varphi(K(\cdot,y)b)\big\rangle_V \\ &= \big(\rho \;\big|\; \mathrm{Ad}_\varphi(K(x,y)b)\big)_x. \end{aligned}$$

Theorem 10.8 is now a consequence of the decomposition

$$\mathrm{Ad}_\varphi^T v(y) = \sum_{i=1}^d e_i^T \mathrm{Ad}_\varphi^T v(y) e_i.$$

□

Recall that $K(\cdot,\cdot)$ is a matrix, so that $K(\cdot,y)e_i$ is the ith column of $K(\cdot,y)$, which we can denote by K^i. Equation (10.19) states that the ith coordinate of $\mathrm{Ad}_\varphi^T v$ is $\big(\rho \;\big|\; \mathrm{Ad}_\varphi K^i(x,y)\big)_x$.

Using Proposition 10.6 and Theorem 10.8, we obtain another expression of the V-gradient of E:

Corollary 10.9 *Under the hypotheses of Proposition 10.6, the V-gradient of*

$$\tilde{E}(v) = \frac{1}{2}\int_0^1 \|v(t)\|_V^2 dt + U(\varphi_{01}^v)$$

is equal to

$$\nabla^V \tilde{E}(v)(y) = v(t,y) + \sum_{i=1}^d \big(\rho(1) \;\big|\; d\varphi_{t1}(\varphi_{1t}^v(x))K^i(\varphi_{1t}^v(x),y)\big)_x e_i \tag{10.20}$$

with $\rho(1) = \bar{\partial}U(\varphi_{01}^v)(x)$.

10.5 Conservation of Momentum

10.5.1 Interpretation

Equation (10.16) can be interpreted as a momentum conservation equation. The justification of the term momentum comes from the analogy of $E_{\mathrm{kin}} := (1/2)\|v(t)\|_V^2$

with the total kinetic energy at time t of a dynamical system. In fluid mechanics, this energy is usually defined as (introducing a mass density, z)

$$E_{\text{kin}} = \frac{1}{2}\int z(t, y)|v(t, y)|^2 dy,$$

the momentum here being $\rho(t) = z(t, y)v(t, y)dy$ with $E_{\text{kin}} = (1/2)(\rho \mid v)$. In our case, taking $\rho(t) = \mathbb{L}v(t)$, we still have $E_{\text{kin}} = (1/2)(\rho \mid v)$, so that $\rho(t)$ is also interpreted as a momentum.

To interpret (10.16) as a conservation equation, we need to understand how a change of coordinate system affects the momentum. Indeed, interpret $v(t, y)$ as the velocity of a particle located at coordinates y, so $v = dy/dt$. Now assume that we want to use a new coordinate system, and replace y by $x = \varphi(y)$. In the new coordinates, the same particle moves with velocity

$$\partial_t x = d\varphi(y)\partial_t y = d\varphi(y)\, v(t, y) = (d\varphi\, v(t)) \circ \varphi^{-1}(x)$$

so that the translation from the old to the new expression of the velocity is precisely given by the adjoint operator: $v(y) \to \tilde{v}(x) = \mathrm{Ad}_\varphi v(x)$ if $x = \varphi(y)$. To obtain the correct transformation of the momentum, it suffices to notice that the energy of the system must remain the same if we just change the coordinates, so that, if ρ and $\tilde{\rho}$ are the momenta before and after the change of coordinates, we must have

$$(\tilde{\rho} \mid \tilde{v}) = (\rho \mid v)$$

which yields $\mathrm{Ad}^*_\varphi \tilde{\rho} = \rho$ or $\tilde{\rho} = \mathrm{Ad}^*_{\varphi^{-1}}\rho$.

Now, we return to Eq. (10.16). Here, $v(t, y)$ is the velocity at time t of the particle that was at $x = \varphi^v_{t0}(y)$ at time 0. So it is the expression of the velocity in a coordinate system that evolves with the flow, and $\mathbb{L}v(t)$ is the momentum in the same system. By the previous argument, the expression of the momentum in the fixed coordinate system, taken at time $t = 0$, is $\mathrm{Ad}^*_{\varphi^v_{0t}} \mathbb{L}v(t)$. Equation (10.16) simply states that this expression remains constant over time, i.e., the momentum is conserved when measured in a fixed coordinate system.

The conservation of momentum equation, described in Corollary 10.7, is a fundamental equation in *Geometric Mechanics* [149, 187], which appears in a wide variety of contexts. It has been described in abstract form by Arnold [18, 19] in his analysis of invariant Riemannian metrics on Lie groups. This equation also derives from an application of the Euler–Poincaré principle, as described in [149, 150, 188]. Combined with a volume-preservation constraint, this equation is equivalent to the Euler equation for incompressible fluids, in the case when $\|v(t)\|_V = \|v(t)\|_2$, the L^2 norm. Another type of norm on V (called the H^1_α norm) relates to models of waves in shallow water, and provides the Camassa–Holm equation [50, 116, 149]. A discussion of (10.16) in the particular case of template matching is provided in [205], and a parallel with the solitons emerging from the Camassa–Holm equation is discussed in [151].

10.5.2 Properties of the Momentum Conservation Equation

Combining Eq. (10.19) and the fact that $\partial_t \varphi^v_{0t} = v(t, \varphi^v_{0t})$, we get, for the optimal v (letting $v_0 = \mathbb{K}\rho_0$)

$$\partial_t \varphi(t, y) = \sum_{i=1}^{d} \big(\rho_0 \bigm| (d\varphi(t, x))^{-1} K^i(\varphi(t, x), \varphi(t, y))\big)_x e_i .$$

Letting $\varphi = \mathrm{id} + \omega$, we consider the equation

$$\partial_t \omega(t, y) = \sum_{i=1}^{d} \big(\rho_0 \bigm| (\mathrm{Id} + d\omega(t, x))^{-1} K^i(x + \omega(t, x), y + \omega(t, y))\big)_x e_i . \tag{10.21}$$

We now consider this equation as an ODE over $C^p_0(\Omega, \mathbb{R}^d)$ and discuss conditions on ρ_0 ensuring the existence and uniqueness of solutions. We will make the following assumptions.

(I) V is continuously embedded in $C^{p+1}_0(\Omega, \mathbb{R}^d)$ and its kernel, K, is such that all derivatives $\partial^k_1 \partial^k_2 K(y, y)$ are bounded over Ω for $k \leq p + 1$.

(II) $\rho_0 \in C^r(\Omega, \mathbb{R}^d)^*$ for some $r \leq p - 1$.

(III) ρ_0 is compactly supported: there exists a compact subset $Q' \subset \mathbb{R}^d$ such that $(\rho_0 \mid f) = 0$ for all $f \in C^r_0(\Omega, \mathbb{R}^d)$ such that $f(x) = 0$ for all $x \in Q'$.

Assumption (I) is true in particular when $\Omega = \mathbb{R}^d$ and K is translation-invariant.

Taking Q slightly larger than Q' in assumption (III), and choosing a C^∞ function ε such that $\varepsilon = 1$ on Q' and $\varepsilon = 0$ on Q^c, we have $(\rho_0 \mid f) = (\rho_0 \mid \varepsilon f)$ for all $f \in C^r_0(\Omega, \mathbb{R}^d)$, from which we can deduce that, for some constant C

$$(\rho_0 \mid f) \leq C \| f \|_{r,Q},$$

where

$$\| f \|_{r,Q} = \max_{x \in Q} \max_{|J| \leq r} |\partial_J f(x)|.$$

The following lemma provides the required properties for the well-posedness of (10.21).

Let $\mathcal{O} = \mathrm{Diff}^p_0 - \mathrm{id}$, an open subset of $C^p_0(\Omega, \mathbb{R}^d)$.

Lemma 10.10 *Let*

$$\mathscr{V}(\omega)(y) = \sum_{i=1}^{d} \big(\rho_0 \bigm| (\mathrm{Id} + d\omega(t, x))^{-1} K^i(x + \omega(t, x), y + \omega(t, y))\big)_x e_i . \tag{10.22}$$

Under assumptions (I), (II), (III) above, $\mathscr{V}$ is a differentiable mapping from $\mathcal{O}$ into $C^p_0(\Omega, \mathbb{R}^d)$ and, letting $\varphi = \mathrm{id} + \omega$,

$$\|d\mathscr{V}\|_{op(C_0^p(\Omega,\mathbb{R}^d))} \leq C(\|d\varphi(\cdot)^{-1}\|_\infty, \|\varphi\|_{p,\infty}) \tag{10.23}$$

for some continuous function C.

Proof Step 1. We first check that the right-hand side of (10.21) is well defined. Since we assume that V is embedded in $C_0^{p+1}(\Omega,\mathbb{R}^d)$, we know that, for all $0 \leq r, s, \leq p+1$, $\partial_1^r\partial_2^s K^i$ is in $C_0(\Omega,\mathbb{R}^d)$ with respect to each of its variables. In particular, $x \mapsto (\mathrm{Id} + d\omega(t,x))^{-1}K^i(x+\omega(t,x), y+\omega(t,y))$ is in $C_0^{p-1}(\Omega,\mathbb{R}^d)$ as soon as $\omega \in C_0^p(\Omega,\mathbb{R}^d)$, so that ρ_0 can be evaluated on it.

Step 2. We now prove that the right-hand side of (10.21) is in $C_0^p(\Omega,\mathbb{R}^d)$, which ensures that (10.21) forms an ODE in this space. Let

$$v^\varphi(y) = \sum_{i=1}^d \big(\rho_0 \;\big|\; d\varphi(x)^{-1}K^i(\varphi(x), y)\big)_x e_i$$

so that (10.21) can be written as $\partial_t\omega = v^{\mathrm{id}+\omega} \circ (\mathrm{id}+\omega)$. We want to show that $v^\varphi \in C_0^p(\Omega,\mathbb{R}^d)$ when $\varphi = \mathrm{id}+\omega$ and $\omega \in C_0^p(\Omega,\mathbb{R}^d)$. It is obviously sufficient to prove that each coordinate

$$v_i^\varphi(y) = \big(\rho_0 \;\big|\; d\varphi(x)^{-1}K^i(\varphi(x), y)\big)_x$$

belongs to $C_0^p(\Omega,\mathbb{R})$. We first justify the fact that v_i^φ is p-times differentiable, with

$$d^r v_i^\varphi(y) = \big(\rho_0 \;\big|\; d\varphi(x)^{-1}\partial_2^r K^i(\varphi(x), y)\big)_x$$

for $r \leq p$. Using a Taylor expansion, we can write (letting $h^{(k)}$ denote the k-tuple $(h,\ldots,h)$)

$$\begin{aligned}K^i(\varphi(x), y+h) &= \sum_{k=0}^{p+1}\frac{1}{k!}\partial_2^k K^i(\varphi(x),y)h^{(k)}\\ &\quad+\frac{1}{p!}\int_0^1(\partial_2^{p+1}K^i(\varphi(x),y+th) - \partial_2^{p+1}K^i(\varphi(x),y))h^{(p+1)}(1-t)^p\,dt\end{aligned}$$

so that

$$\begin{aligned}v_i^\varphi(y+h) &= \sum_{k=0}^{p+1}\frac{1}{k!}\Big(\rho_0 \;\Big|\; d\varphi(x)^{-1}\partial_2^k K^i(\varphi(x),y)h^{(k)}\Big)_x\\ &+\frac{1}{p!}\Big(\rho_0\;\Big|\; d\varphi(x)^{-1}\int_0^1(\partial_2^{p+1}K^i(\varphi(x),y+th) - \partial_2^{p+1}K^i(\varphi(x),y))h^{(p+1)}(1-t)^p\,dt\Big)_x\end{aligned}$$

and it suffices to prove that the remainder is an $o(|h|^{p+1})$. This will be true provided

$$\lim_{y'\to y} \|\partial_2^{p+1} K^i(\cdot, y') - \partial_2^{p+1} K^i(\cdot, y)\|_{r,\infty} = 0.$$

For $k_1 \le r$, we have, using Eq. (8.8),

$$\begin{aligned}
&\|\partial_1^{k_1}\partial_2^{p+1} K^i(\cdot, y') - \partial_1^{k_1}\partial_2^{p+1} K^i(\cdot, y)\|_{r,\infty} \\
&\quad \le C \max_{h:|h|=1} \|\partial_2^{p+1} K^i(\cdot, y')(h^{(p+1)}) - \partial_2^{p+1} K^i(\cdot, y)(h^{(p+1)})\|_V \\
&\quad = C \left|\partial_1^{p+1}\partial_2^{p+1} K^{ii}(y', y') + \partial_1^{p+1}\partial_2^{p+1} K^{ii}(y, y) - 2\partial_1^{p+1}\partial_2^{p+1} K^{ii}(y, y')\right|^{1/2}
\end{aligned}$$

for some constant C, where K^{ij} denotes the i, j entry of K. This proves the desired result, since $\partial_1^{p+1}\partial_2^{p+1} K$ is continuous. A similar argument can be made to prove the continuity of $y \mapsto d^p v(y)$.

To prove that $v^\varphi \in C_0^p(\Omega, \mathbb{R}^d)$, it suffices to show that, for all $k \le p+1$, $\|\partial_2^k K^i(\cdot, y)\|_{r,Q}$ goes to 0 when y goes to infinity. (This is where we use the fact that ρ_0 has compact support.)

To reach a contradiction, assume that there exists sequences (x_n), (y_n) with $x_n \in Q$ and y_n tending to infinity or $\partial\Omega$ such that $|\partial_1^{k_1}\partial_2^{k_2} K^i(x_n, y_n)| > \varepsilon$, for some fixed $\varepsilon > 0$ and $k_1 \le r, k_2 \le p+1$. Replacing (x_n) by a subsequence if needed, we can assume that x_n converges to some $x \in Q$. Note that $\partial_1^{k_1}\partial_2^{k_2} K^{ij}(x, y_n) = \partial_1^{k_2}\partial_2^{k_1} K^{ji}(y_n, x)$. Since $\partial_2^{k_1} K^j(\cdot, x) \in V$, we can conclude that $\partial_1^{k_2}\partial_2^{k_1} K^j(y_n, x) \to 0$ for all j, implying that $\partial_1^{k_1}\partial_2^{k_2} K^i(x, y_n) \to 0$ for all i, too.

Similarly, $\partial_1^{k_1}\partial_2^{k_2} K^{ij}(x_n, y_n) - \partial_1^{k_1}\partial_2^{k_2} K^{ij}(x, y_n)$ is the ith entry of $\partial_1^{k_2}\partial_2^{k_1} K^j(y_n, x_n) - \partial_2^{k_1}\partial_1^{k_2} K^j(y_n, x)$ and

$$\begin{aligned}
&\sup_y |\partial_1^{k_2}\partial_2^{k_1} K^j(y, x_n) - \partial_1^{k_2}\partial_2^{k_1} K^j(y, x)| \\
&\quad \le C \max_{h:|h|=1} \|\partial_2^{k_1} K^j(\cdot, x_n)(h^{(k_1)}) - \partial_2^{k_1} K^j(\cdot, x)(h^{(k_1)})\|_V \\
&\quad \le C \left|\partial_1^{k_1}\partial_2^{k_1} K^{jj}(x_n, x_n) - 2\partial_1^{k_1}\partial_2^{k_1} K^{jj}(x_n, x) + \partial_1^{k_1}\partial_2^{k_1} K^{jj}(x, x)\right|^{1/2},
\end{aligned}$$

which goes to 0. This is our contradiction.

Step 3: We now study the differentiability of the mapping $\mathscr{V} : \omega \mapsto v^{\mathrm{id}+\omega} \circ (\mathrm{id} + \omega)$ from $C_0^p(\Omega, \mathbb{R}^d)$ into itself. The candidate for $d\mathscr{V}(\omega)\eta$ is

$$
\begin{aligned}
(\mathscr{W}(\omega)\eta)(y) = &-\sum_{i=1}^{d}\big(\rho_0 \ \big| d\varphi(x)^{-1}d\eta(x)^{-1}d\varphi(x)^{-1}K^i(\varphi(x),\varphi(y))\big)_x e_i \\
&+\sum_{i=1}^{d}\big(\rho_0 \ \big| d\varphi(x)^{-1}\partial_1 K^i(\varphi(x),\varphi(y))\eta(x)\big)_x e_i \\
&+\sum_{i=1}^{d}\big(\rho_0 \ \big| d\varphi(x)^{-1}\partial_2 K^i(\varphi(x),\varphi(y))\eta(y)\big)_x e_i,
\end{aligned}
$$

still with $\varphi = \mathrm{id} + \omega$. We can decompose $\mathcal{V}(\omega+\eta)(y) - \mathcal{V}(\omega)(y) - (\mathscr{W}(\omega)\eta)(y)$ as the sum of five terms

$$
\sum_{k=1}^{5}\sum_{i=1}^{d}\big(\rho_0 \ \big| A^i(x,y)\big)_x
$$

(described below), which we will study separately. For each term, we need to prove that, for $k_1 \le r$, $k_2 \le p$, one has

$$
\sup_{x,y} |\partial_1^{k_1}\partial_2^{k_2} A_k^i(x,y)| = o(\|\eta\|_{p,\infty}).
$$

The important point in the following discussion is that none of the estimates will require more than p derivatives in φ and η, and no more than $p+1$ in K.

(i) We let

$$
\begin{aligned}
A_1^i(x,y) = ((d\varphi(x) + d\eta(x))^{-1} - d\varphi(x)^{-1} + d\varphi(x)^{-1}d\eta(x)d\varphi(x)^{-1})& \\
K^i(\varphi(x)+\eta(x), \varphi(y)+\eta(y)).&
\end{aligned}
$$

We first note that $Inv : M \mapsto M^{-1}$ is infinitely differentiable on $\mathrm{GL}_d(\mathbb{R})$ with

$$
d^q Inv(M)(H_1,\ldots,H_q) = (-1)^q \sum_{\sigma\in\mathfrak{S}_q} M^{-1}H_{\sigma(1)}M^{-1}\cdots M^{-1}H_{\sigma(q)}M^{-1},
$$

where $\mathfrak{S}_q$ is the set of permutations of $\{1,\ldots,q\}$. In particular, $\|d^q Inv(M)\| = O(\|M^{-1}\|^{q+1})$. Writing

$$
\begin{aligned}
(d\varphi(x) + d\eta(x))^{-1} - d\varphi(x)^{-1} + d\varphi(x)^{-1}d\eta(x)d\varphi(x)^{-1} =& \\
\int_0^1 d^2 Inv(d\varphi(x) + td\eta(x))(d\eta(x), d\eta(x))(1-t)\,dt,&
\end{aligned}
$$

we see that

$$
\big\|d^{k_1}((d\varphi(x) + d\eta(x))^{-1} - d\varphi(x)^{-1} + d\varphi(x)^{-1}d\eta(x)d\varphi(x)^{-1})\big\|_\infty
$$

will be less than $C(\varphi)\|d\varphi^{-1}\|_\infty^{k_1+3}\|\eta\|_{k_1+1,\infty}^2$. Using the bound

$$\|\partial_2^{k_2}K^i(\cdot, y)\|_{p+1,\infty}^2 \leq C\partial_1^{k_2}\partial_2^{k_2}K^{ii}(y, y),$$

applying Lemma 7.3 and the product formula, we see that the desired conclusion holds for A_1^i.

(ii) Let

$$A_2^i(x, y) = d\varphi(x)^{-1}(K^i(\varphi(x) + \eta(x), \varphi(y) + \eta(y)) - K^i(\varphi(x), \varphi(y) + \eta(y)) \\ - \partial_1 K^i(\varphi(x), \varphi(y) + \eta(y))\eta(x)).$$

Writing the right-hand side in the form

$$d\varphi(x)^{-1}\int_0^1 \partial_1^2 K^i(\varphi(x) + t\eta(x), \varphi(y) + \eta(y))(\eta(x), \eta(x))(1 - t)\, dt,$$

the same estimate on the derivative of K can be used, based on the fact that $k_1 + 2 \leq p + 1$.

(iii) The third term is

$$A_3^i(x, y) = d\varphi(x)^{-1}(K^i(\varphi(x), \varphi(y) + \eta(y)) - K^i(\varphi(x), \varphi(y)) \\ - \partial_2 K^i(\varphi(x), \varphi(y))\eta(y)).$$

It can be handled similarly, requiring $k_2 + 1 \leq p + 1$ derivatives of K^i in the second variable.

(iv) These were the three main terms in the decomposition and the remaining two are just bridging gaps. The first one is

$$A_3^i(x, y) = d\varphi(x)^{-1}d\eta(x)d\varphi(x)^{-1} \\ (K^i(\varphi(x) + \eta(x), \varphi(y) + \eta(y)) - K^i(\varphi(x), \varphi(y))).$$

Here, we note that, for some constants C and $\tilde{C}$,

$$\begin{aligned}&\sup_x |\partial_1^{k_1}\partial_2^{k_2}K^i(x, y') - \partial_1^{k_1}\partial_2^{k_2}K^i(x, y)|^2 \\ &\leq C\left|\partial_1^{k_2}\partial_2^{k_2}(K(y', y') - 2K(y', y) + K(y, y))\right| \\ &\leq \tilde{C}(\partial_1^{k_2+1}\partial_2^{k_2+1}K(y, y) + \partial_1^{k_2+1}\partial_2^{k_2+1}K(y', y'))|y - y'|\end{aligned}$$

(with a similar inequality when the roles of x and y are reversed) and these estimates can be used to check that

$$\partial_1^{k_1}\partial_2^{k_2}(K^i(\varphi(x) + \eta(x), \varphi(y) + \eta(y)) - K^i(\varphi(x), \varphi(y)))$$

tends to 0 uniformly in x and y.
(v) The last term is

$$A_5^i(x, y) = d\varphi(x)^{-1}(\partial_1 K^i(\varphi(x), \varphi(y) + \eta(y)) - \partial_1 K^i(\varphi(x), \varphi(y)))\eta(x)$$

and can be handled similarly.

Step 4. It remains to check that $\mathscr{W}(\omega)$ maps $C_0^p(\Omega, \mathbb{R}^d)$ to itself. This can be done in the same way we proved that $\mathscr{V}(\omega) \in C_0^p(\Omega, \mathbb{R}^d)$, using Taylor expansions and the fact that $d^k(\mathscr{W}(\omega)\eta)(y)$ will involve no more than k derivatives of ω and η, and $k+1$ of K. This shows that $\mathscr{W} = d\mathscr{V}$. The bound (10.23) can also be shown using the same techniques. We leave the final details to the reader. □

Lemma 10.10 implies that (10.21) has unique local solutions (unique solutions over small enough time intervals). If we can prove that $\|(d\varphi)^{-1}\|_\infty$ and $\|\varphi\|_{p,\infty}$ remains bounded over solutions of the equation, inequality (10.23) will be enough to ensure that solutions exist over arbitrary times intervals. This fact will be obtained at the end of the next section.

10.5.3 Time Variation of the Eulerian Momentum

Assume that φ satisfies $\partial_t \varphi(t) = v(t) \circ \varphi(t)$ with $v \in \mathcal{X}^{p+1,1}(\Omega)$. If $\rho_0 \in C^{p-1}(\Omega, \mathbb{R}^d)^*$, we can apply the chain rule to the equation

$$(\rho(t) \mid w) = \left(\rho_0 \,\middle|\, \mathrm{Ad}_{\varphi(t)^{-1}} w\right) = \left(\rho_0 \,\middle|\, d\varphi(t)^{-1} w \circ \varphi(t)\right),$$

in which we assume that $w \in C_0^p(\Omega, \mathbb{R}^d)$. We have (with $\partial_t d\varphi = dv \circ \varphi \, d\varphi$)

$$\begin{aligned}\partial_t \mathrm{Ad}_{\varphi(t)^{-1}} w &= -d\varphi(t)^{-1} dv(t) \circ \varphi(t)\, w \circ \varphi(t) + d\varphi(t)^{-1} dw \circ \varphi(t)\, v(t) \circ \varphi(t) \\ &= -\mathrm{Ad}_{\varphi(t)^{-1}}(dv(t)\, w - dw\, v(t)).\end{aligned}$$

The term in the right-hand side involves the adjoint representation of $v(t)$, as expressed in the following definition.

Definition 10.11 If v is a differentiable vector field on Ω, we denote by ad_v the mapping that transform a differentiable vector field w into

$$\mathrm{ad}_v w = dv\, w - dw\, v. \tag{10.24}$$

Observe that $dv\, w - dw\, v = -[v, w]$, where the latter is the Lie bracket between right-invariant vector fields over the group of diffeomorphisms. Note that ad_v continuously maps $C_0^p(\Omega, \mathbb{R}^d)$ to $C_0^{p-1}(\Omega, \mathbb{R}^d)$. With this notation, we therefore have, for $w \in C_0^p(\Omega, \mathbb{R}^d)$:

$$\partial_t \mathrm{Ad}_{\varphi(t)^{-1}} w = -\mathrm{Ad}_{\varphi(t)^{-1}} \mathrm{ad}_{v(t)} w$$

so that

$$\partial_t(\rho(t) \mid w) = -\big(\rho(t) \,\big|\, \mathrm{ad}_{v(t)} w\big).$$

This yields the equation, called EPDiff, in which we let $\tilde\rho(t)$ denote the restriction of $\rho(t)$ to $C_0^p(\Omega, \mathbb{R}^d)$,

$$\partial_t \tilde\rho(t) + \mathrm{ad}^*_{v(t)}\rho(t) = 0. \tag{10.25}$$

Equation (10.25) can be used to prove the following proposition.

Proposition 10.12 *Let $\varphi(t) = \mathrm{id} + \omega(t)$, where ω is a solution of (10.21). Let $v_0 = K\rho_0$ and $v(t) = \mathrm{Ad}^T_{\varphi(t)^{-1}} v_0$. Then $\|v(t)\|_V$ is independent of time.*

Proof Indeed, we have, for $\varepsilon > 0$,

$$\frac{1}{\varepsilon}(\|v(t+\varepsilon)\|^2 - \|v(t)\|_V^2) = \frac{2}{\varepsilon}(\rho(t+\varepsilon) - \rho(t) \mid v(t)) + \frac{1}{\varepsilon}\|v(t+\varepsilon). - v(t)\|_V^2$$

Since $v(t) \in V \subset C_0^p(\Omega, \mathbb{R}^d)$, (10.25) implies that the first term on the right-hand side converges to

$$-2\big(\rho(t) \,\big|\, \mathrm{ad}_{v(t)} v(t)\big) = 0.$$

For the second term, we have

$$\begin{aligned}\|v(t+\varepsilon) - v(t)\|_V &= \sup_{\|w\|_V \le 1} (\rho(t+\varepsilon) - \rho(t) \mid w) \\ &= \sup_{\|w\|_V \le 1} \int_0^\varepsilon \big(\rho(t+s) \,\big|\, \mathrm{ad}_{v(t+s)} w\big) ds,\end{aligned}$$

which tends to 0 with ε. □

We can now prove that (10.21) has a unique solution over arbitrary time intervals.

Theorem 10.13 *Under the hypotheses of Lemma 10.10, Eq. (10.21) has solutions over all times, uniquely specified by its initial conditions.*

Proof As already mentioned, Lemma 10.10 implies that solutions exist over small time intervals. Inequality (10.23) implies that these solutions can be extended as long as $\|d\varphi(t)^{-1}\|_\infty$ and $\|\varphi(t)\|_{p,\infty}$ remain finite. However, both these quantities are controlled by $\int_0^t \|v(t)\|_V \, dt$. For the latter, this is a consequence of (C.6). For $d\varphi(t)^{-1}$, we can note that

$$\partial_t(d\varphi(t)^{-1}) = -d\varphi(t)^{-1} dv(t) \circ \varphi(t)$$

and use Gronwall's lemma to ensure that

$$\|d\varphi(t)^{-1}\|_\infty \le \exp\left(C \int_0^1 \|v(s)\|_V \, ds\right)$$

for some constant C. □

10.5.4 Explicit Expression

The assumption that $\rho_0 \in C_0^{p-1}(\Omega, \mathbb{R}^d)^*$ "essentially" expresses the fact that the evaluation of $(\rho_0 \mid w)$ will involve no more than $p-1$ derivatives of w. This implies that the evaluation of the right-hand side of (10.21) will involve derivatives up to order p in $\varphi = \mathrm{id} + \omega$. In numerical implementations, it is often preferable to track the evolution of these derivatives over time, rather than approximate them using, e.g., finite differences. It often happens, for example, that the evaluation of ρ_0 only requires the evaluation of φ and its derivatives over a submanifold of lower dimension, and tracking their values on a dense grid becomes counter-productive.

The evolution of the derivatives of φ can easily be computed by differentiating (10.21) with respect to the y variable. This is summarized in the system

$$\begin{cases} \partial_t \varphi(t,y) = \sum_{i=1}^d \big(\rho_0 \mid (d\varphi(t,x))^{-1} K^i(\varphi(t,x), \varphi(t,y))\big)_x e_i \\ \partial_t d\varphi(t,y)a = \sum_{i=1}^d \big(\rho_0 \mid (d\varphi(t,x))^{-1} \partial_2 K^i(\varphi(t,x), \varphi(t,y))(d\varphi(t,y)a)\big)_x e_i \\ \quad \vdots \\ \partial_t d^p \varphi(t,y)(a_1, \ldots, a_p) \\ \qquad = \sum_{i=1}^d \big(\rho_0 \mid (d\varphi(t,x))^{-1} \partial_2^p K^i(\varphi(t,x), \varphi(t,y))(a_1, \ldots, a_p)\big)_x e_i. \end{cases} \tag{10.26}$$

It should be clear from this system that, if the computation of $(\rho_0 \mid w)$ only requires the evaluation of w and its derivatives on some subset of $\mathbb{R}^d$, then φ and its derivatives only need to be tracked for y belonging to the same subset.

10.5.5 The Hamiltonian Form of EPDiff

We now provide an alternative form of (10.26), using the optimal control formulation discussed in Sect. 10.4.2, in which we introduced the co-state

$$(\mu(t) \mid w) = \big(\rho_0 \,\big|\, d\varphi(t)^{-1} w\big) = \big(\rho(t) \,\big|\, w \circ \varphi(t)^{-1}\big). \tag{10.27}$$

Let $M(t) = (d\varphi(t))^{-1}$ so that $\partial_t M = -M\,(\partial_t d\varphi)\,M$. The second equation of (10.26) then becomes

$$\partial_t M(t,y)a = -\sum_{i=1}^{d} \big(\rho_0 \mid M(t,x)\partial_2 K^i(\varphi(t,x),\varphi(t,y))a\big)_x M(t,y)e_i.$$

This implies that, for any $w \in V$

$$\begin{aligned}\Big(\partial_t\mu(t)\Big| w\Big) &= -\Big(\rho_0 \Big| \partial_t M\, w\Big)\\ &= \sum_{i=1}^{d} \big(\rho_0 \mid \big(\rho_0 \mid M(t,x)\partial_2 K^i(\varphi(t,x),\varphi(t,y))w(y)\big)_x M(t,y)e_i\big)_y\\ &= -\sum_{i=1}^{d} \big(\mu(t) \mid \big(\mu(t) \mid \partial_2 K^i(\varphi(t,x),\varphi(t,y))w(y)\big)_x e_i\big)_y.\end{aligned}$$

We therefore have the system

$$\begin{cases} \partial_t\varphi(t,y) = \sum_{i=1}^{d} \big(\mu(t) \mid K^i(\varphi(t,x),\varphi(t,y))\big)_x e_i \\ \Big(\partial_t\mu(t)\Big| w\Big) = -\sum_{i=1}^{d} \big(\mu(t) \mid \big(\mu(t) \mid \partial_2 K^i(\varphi(t,x),\varphi(t,y))w(y)\big)_x e_i\big)_y. \end{cases} \tag{10.28}$$

Note that this system is an alternative expression of the first two equations of (10.17). When $(\rho_0 \mid w)$ does not depend on the derivatives of w (more precisely, $\rho_0 \in C_0^0(\Omega,\mathbb{R}^d)^*$), this provides an ordinary differential equation in the variables (φ,μ) (of the form $(d/dt)(\varphi,\mu) = F(\varphi,\mu)$). The initial conditions are $\varphi_0 = \mathrm{id}$ and $\mu_0 = \rho_0$.

10.5.6 The Case of Measure Momenta

An interesting feature of (10.28) is that it can easily be reduced to a smaller number of dimensions when ρ_0 takes specific forms. As a typical example, we perform the computation in the case

$$\rho_0 = \sum_{k=1}^{N} z_k(0,\cdot)\gamma_k, \tag{10.29}$$

where γ_k is an arbitrary measure on Ω and $z_k(0)$ a vector field. (We recall the notation $(z\gamma \mid w) = \int z(x)^T w(x)\,\gamma(dx)$.) Most of the Eulerian differentials that we have computed in Chap. 9 have been reduced to this form. From the definition of $\mu(t)$, we have $\mu(t) = \sum_{k=1}^{N} z_k(t,.)\gamma_k$ (where $z_k(t,x) = d\varphi_{0t}(x)^{-T} z_k(0,x)$). The first equation in (10.28) is

$$\partial_t\varphi(t,y) = \sum_{i=1}^d\sum_{k=1}^N \int_\Omega z_k(t,x)^T K^i(\varphi(t,x),\varphi(t,y))e_i d\gamma_k(x).$$

For a matrix A with ith column vector A^i, and a vector z, $z^T A^i$ is the ith coordinate of $A^T z$. Applying this to the previous equation yields

$$\partial_t\varphi(t,y) = \sum_{k=1}^N \int_\Omega K(\varphi(t,y),\varphi(t,x))z_k(t,x)d\gamma_k(x), \tag{10.30}$$

where we have used the fact that $K(\varphi(t,x),\varphi(t,y))^T = K(\varphi(t,y),\varphi(t,x))$. The second equation in (10.28) becomes

$$\begin{aligned}
&\Big(\partial_t\mu(t)\,\Big|\, w\Big)\\
&= -\sum_{i=1}^d \Big(\mu(t)\,\Big|\,\Big(\mu(t)\,\Big|\,\partial_2 K^i(\varphi(t,x),\varphi(t,y))w(y)\Big)_x e_i\Big)_y\\
&= -\sum_{k,l=1}^N \int_\Omega\int_\Omega \sum_{i=1}^d z_l^T(t,x)\partial_2 K^i(\varphi(t,x),\varphi(t,y))w(y)\, z_k(t,y)^T e_i d\gamma_l(x)d\gamma_k(y)\\
&= -\sum_{k=1}^N \int_\Omega \left(\int_\Omega \sum_{l=1}^N\sum_{i=1}^d z_k^i(t,y)z_l(t,x)^T\partial_2 K^i(\varphi(t,x),\varphi(t,y))d\gamma_l(x)\right) w(y)d\gamma_k(y),
\end{aligned}$$

where z_k^i is the ith coordinate of z_k. From the expression of $\mu(t)$, we also have

$$\partial_t\mu = \sum_{k=1}^N(\partial_t z_k)\gamma_k.$$

Letting K^{ij} denote the entries of K, we can identify $\partial_t z_k$ as

$$\begin{aligned}
&\partial_t z_k(t,y) =\\
&-\int_\Omega \sum_{l=1}^N\sum_{i,j=1}^d z_k^i(t,y)z_l^j(t,x)\nabla_2 K^{ij}(\varphi(t,x),\varphi(t,y))d\gamma_l(x)\\
&= -\int_\Omega \sum_{l=1}^N\sum_{i,j=1}^d z_l^i(t,y)z_k^j(t,x)\nabla_1 K^{ij}(\varphi(t,y),\varphi(t,x))d\gamma_l(x).
\end{aligned}\tag{10.31}$$

This equation is somewhat simpler when K is a scalar kernel, in which case $K^{ij}(x,y) = \Gamma(x,y)$ if $i = j$ and 0 otherwise, where Γ takes real values. We get, in this case

$$\begin{aligned}\partial_t z_k(t,y) &= -\sum_{l=1}^N \int_\Omega \nabla_2 \Gamma(\varphi(t,x),\varphi(t,y)) z_k(t,y)^T z_l(t,x) d\gamma_l(x)\\ &= -\sum_{l=1}^N \int_\Omega \nabla_1 \Gamma(\varphi(t,y),\varphi(t,x)) z_k(t,y)^T z_l(t,x) d\gamma_l(x).\end{aligned}$$

In all cases, we see that the evolution of μ can be completely described using the evolution of $z_1, \ldots, z_N$. In the particular case when the z_k's are constant vectors (which corresponds to most of the point-matching problems), this provides a finite-dimensional system on the μ part.

10.6 Optimization Strategies for Flow-Based Matching

We have formulated flow-based matching as an optimization problem over time-dependent vector fields. We discuss here other possible optimization strategies that take advantage of the different formulations that we obtained for the EPDiff equation. They will correspond to taking different control variables with respect to which the minimization is performed, and we will in each case provide the expression of the gradient of E with respect to a suitable metric. Optimization can then be performed by gradient descent, conjugate gradient or higher-order optimization algorithms when feasible (see Appendix D or [221]).

After discussing the general formulation of each of these strategies, we will provide the specific expression of the gradients for point-matching problems, in the following form: minimize

$$E(\varphi) = \frac{1}{2} d_V(\mathrm{id}, \varphi)^2 + F(\varphi(x_1), \ldots, \varphi(x_N)) \tag{10.32}$$

with respect to φ, where $x_1, \ldots, x_N$ are fixed points in Ω. These problems are important because, in addition to the labeled and unlabeled point matching problems we have discussed, other problems, such as curve and surface matching, end up being discretized in this form (we will discuss algorithms for image matching in the next section). The following discussion describes (and often extends) several algorithms that have been proposed in the literature, in [32, 159, 203, 204, 289, 309] among other references.

10.6.1 Gradient Descent in $\mathcal{X}_V^2$

The original problem having been expressed in this form, Corollary 10.9 directly provides the expression of the gradient of E considered as a function defined over

$\mathcal{X}_V^2$, with respect to the metric in this space. Using $t \mapsto v(t, \cdot)$ as an optimization variable has some disadvantages, however. The most obvious is that it results in solving a huge dimensional problem (over a $d+1$-dimensional variable) even if the original objects are, say, collections of N landmarks in $\mathbb{R}^d$.

When the matching functional U is only a function of the deformation of a fixed object, i.e.,

$$U(\varphi) = F(\varphi \cdot I),$$

then some simplifications can be made. To go further, we will need to compute derivatives in the object space, and henceforth assume that $\mathcal{I}$ is an open subset of a Banach space $\boldsymbol{I}$. We assume that Diff_0^{p+1} acts on $\mathcal{I}$ and that the mapping Φ_I : $\varphi \mapsto \varphi \cdot I$ is differentiable on Diff_0^{p+1} for all $I \in \mathcal{I}$, so that an infinitesimal action is defined by (see Sect. B.5.3)

$$h \cdot I = d\Phi_I(\mathrm{id})\, h \in \boldsymbol{I}$$

for $h \in C_0^{p+1}(\Omega, \mathbb{R}^d)$. We assume as usual that V is continuously embedded in $C_0^{p+1}(\Omega, \mathbb{R}^d)$ so that $v \cdot I$ is well defined for $v \in V$ and $d\Phi_I(\mathrm{id})$ restricted to V is also bounded with respect to $\| \cdot \|_V$.

Let $v \in \mathcal{X}_V^2$. If $\partial_t \varphi = v \circ \varphi$, let $J(t) = \varphi(t) \cdot I$ be the deforming object. Then $\partial_t J(t) = v(t) \cdot J(t)$. With this in mind, we can write, when $\tilde{E}$ is given by (10.14)

$$\min_{v(t,\cdot)} \tilde{E}(v) = \min_{J(t,\cdot), J(0)=I} \left(\min_{v:\, \partial_t J = v(t) \cdot J(t)} \tilde{E}(v) \right).$$

The iterated minimization first minimizes with respect to v for fixed object trajectories, then optimizes over the object trajectories.

When $J(t, \cdot)$ is given, the inner minimization is

$$\begin{aligned} \min_{v:\, \partial_t J = v(t) \cdot J(t)} \tilde{E}(v) &= \min_{v:\, \partial_t J = v(t) \cdot J(t)} \left(\frac{1}{2} \int_0^1 \|v(t)\|_V^2 \, dt + F(J(1)) \right) \\ &= \frac{1}{2} \int_0^1 \left(\inf_{w:\, \partial_t J = w \cdot J(t)} \|w\|_V^2 \right) dt + F(J(1)) \end{aligned} \tag{10.33}$$

since the constraints apply separately to each $v(t)$. This expression only depends on the trajectory $J(t)$. One can therefore try to compute its gradient with respect to this object trajectory and run a minimization algorithm accordingly. One difficulty with this approach is that, given an object trajectory $J(t)$, there may exist no $w \in V$ such that $\partial_t J = w \cdot J(t)$ (which results in the minimum in the integral being infinite), so that the possibility of expressing the trajectory as evolving according to a flow is a constraint of the problem. This may be intractable in the general case, but always satisfied for point-matching problems as long as the points remain distinct. We will discuss this in the next section.

However, what (10.33) tells us is that, if a time-dependent vector field $\tilde{v}(t, \cdot)$ is given, one always reduces the value of $\tilde{E}(\tilde{v})$ by replacing $\tilde{v}(t, \cdot)$ by

$$v(t, \cdot) = \underset{w:\, w \cdot J(t) = \tilde{v} \cdot J(t)}{\operatorname{argmin}} \|w\|_V^2 \tag{10.34}$$

with $J(t) = \varphi_{0t}^{\tilde{v}} \cdot I$. Introduce the space

$$N_J = \mathit{Null}(d\Phi_J(\mathrm{id})) = \{u \in V : u \cdot J = 0\}$$

and its perpendicular $V_J = N_J^{\perp} = \left\{u \in V : \left\langle u\, ,\, \tilde{u} \right\rangle_V = 0 \text{ for all } \tilde{u} \in N_J\right\}$. Then we have the following lemma.

Lemma 10.14 *Let $I \in \mathcal{I}$ and $\tilde{v} \in V$. Then, the minimizer of $\|w\|_V^2$ over all $V \in V$ such that $w \cdot J = \tilde{v} \cdot J$ is given by $\pi_{V_J}(\tilde{v})$, the orthogonal projection of $\tilde{v}$ on V_J.*

Proof Let $v = \pi_{V_J}(\tilde{v})$. We want to prove that v is a minimizer of $\| \cdot \|_V^2$ over the set of all $w \in V$ such that $w = \tilde{v} + u$ with $u \in N_J$. For such a w, we have

$$\pi_{V_J}(w) = v + \pi_{V_J}(u) = v$$

and $\|w\|_V^2 \geq \|\pi_{V_J}(w)\|_V^2$. Moreover, from the characteristic properties of an orthogonal projection, we have $\tilde{v} - v \in V_J^{\perp} = N_J$, the inequality holding because N_J is closed (because it is the null set of a bounded linear map). □

The numerical computation of this orthogonal projection is not always easy, but when it is, it generally has a form which is more specific than a generic time-dependent vector field, and provides an improved gradient descent algorithm in $\mathcal{X}_V^2$ as follows. Assume that, at time τ in the algorithm, the current vector field v^τ in the minimization of E is such that $v^\tau(t) \in V_{J^\tau(t)}$ at all times t. Then define a vector field at the next step $\tau + \delta\tau$ by

$$\tilde{v}^{\tau+\delta\tau}(t, y) = v^\tau(t, y) - \delta\tau \left(v(t, y) + \sum_{i=1}^{d} \left(\rho(1) \,\middle|\, d\varphi_{t1}(\varphi_{1t}^v(x)) K^i(\varphi_{1t}^v(x), y) \right)_x e_i \right),$$

which corresponds to one step of gradient descent, as specified in (10.20), then compute $J(t) = \varphi_{0t}^{\tilde{v}^{\tau+\delta\tau}} \cdot I$ and define

$$v^{\tau+\delta\tau}(t) = \pi_{V_{J(t)}}(\tilde{v}^{\tau+\delta\tau})$$

at all times t.

Application to Point Matching

Consider the point-matching energy. In this case, letting

$$U(\varphi) = F(\varphi(x_1), \ldots, \varphi(x_N)),$$

we have

$$\rho(1) = \bar{\partial} U(\varphi_{01}^v) = \sum_{i=1}^N \partial_i F(\varphi_{01}^v(x))\delta_{\varphi_{01}^v(x_k)}.$$

We therefore have, by Corollary 10.9, with $\tilde{U}(x) = U(\varphi_{01}^v)$,

$$\begin{aligned}
\nabla^V \tilde{U}(v)(t, y) &= \sum_{i=1}^d \left(\rho(1) \,\middle|\, d\varphi_{t1}^v(\varphi_{1t}^v(x))K(\varphi_{1t}^v(x), y)e_i \right)_x e_i \\
&= \sum_{i=1}^d \sum_{q=1}^N \left(\partial_q F(\varphi_{01}^v(x))^T d\varphi_{t1}^v(\varphi_{0t}^v(x_q))K(\varphi_{0t}^v(x_q), y)e_i\right)e_i \\
&= \sum_{q=1}^N K(y, \varphi_{0t}^v(x_q))d\varphi_{t1}^v(\varphi_{0t}^v(x_q))^T \partial_q F(\varphi_{01}^v(x_q)),
\end{aligned}$$

so that

$$\nabla^V \tilde{E}(v)(t, y) = v(t, y) + \sum_{q=1}^N K(y, \varphi_{0t}^v(x_q))d\varphi_{t1}^v(\varphi_{0t}^v(x_q))^T \partial_q F(\varphi_{01}^v(x_q)). \tag{10.35}$$

So, a basic gradient descent algorithm in $\mathcal{X}_V^2$ would implement the evolution (letting τ denote the algorithm time)

$$\partial_\tau v^\tau(t, y) = -\gamma \left(v^\tau(t, y) + \sum_{q=1}^N K(y, \varphi_{0t}^{v^\tau}(x_q))d\varphi_{t1}^{v^\tau}(\varphi_{0t}^{v^\tau}(x_q))^T \partial_q F(\varphi_{01}^{v^\tau}(x_q)) \right). \tag{10.36}$$

The two-step algorithm defined in the previous section is especially efficient with point sets. When $x = (x_1, \ldots, x_N)$, $v \cdot x = (v(x_1), \ldots, v(x_N))$, the projection on

$$V_x = \{v : v \cdot x = 0\}^\perp = \{v : v(x_1) = \cdots = v(x_N) = 0\}^\perp$$

is given by spline interpolation with the kernel, as described in Theorem 8.8, i.e.,

$$V_x = \left\{ v = \sum_{k=1}^N K(., x_k)a_k, a_1, \ldots, a_N \in \mathbb{R}^d \right\}. \tag{10.37}$$

More precisely, define $x_i^v(t) = \varphi_{0t}^v(x_i)$. We assume that, at time τ, we have a time-dependent vector field v^τ which takes the form

$$v^\tau(t, y) = \sum_{i=1}^N K(y, x_i^{v^\tau}(t))\alpha_i^\tau(t). \tag{10.38}$$

Using (10.36), we define

$$\tilde{v}(t, y) = v^{\tau} - \delta\tau\big(v^{\tau}(t, y) + \sum_{q=1}^{N} K(y, \varphi_{0t}^{v^{\tau}}(x_q))d\varphi_{t1}^{v^{\tau}}(\varphi_{0t}^{v^{\tau}}(x_q))^T \partial_q F(\varphi_{01}^{v^{\tau}}(x_q))\big).$$

The values of $\tilde{v}(t, \cdot)$ are in fact only needed at the points $x_i^{\tilde{v}}(t) = \varphi_{0t}^{\tilde{v}}(x_i)$. These points are obtained by solving the differential equation

$$\partial_t x = v^{\tau}(t, x) - \delta\tau\left(v^{\tau}(t, x) + \sum_{i=1}^{N} K(x_i^{v^{\tau}}(t), x)d\varphi_{t1}^{v^{\tau}}(x_i^{v^{\tau}}(t))^T F_i(x^{v^{\tau}}(1))\right) \tag{10.39}$$

with $x(0) = x_i$. Solving this equation provides both $x_i^{\tilde{v}}(t)$ and $\tilde{v}(x_i^{\tilde{v}}(t))$ for $t \in [0, 1]$.

Once this is done, define $v^{\tau+\delta\tau}(t, \cdot)$ to be the solution of the approximation problem $\inf_w \|w\|_V$ with $w(x_i^{\tilde{v}}(t)) = v(x_i^{\tilde{v}}(t))$, which will therefore take the form

$$v^{\tau+\delta\tau}(t, y) = \sum_{i=1}^{N} K(y, x_i^{v^{\tau+\delta\tau}}(t))\alpha_i^{\tau+\delta\tau}(t).$$

Solving (10.39) requires evaluating the expression of v^{τ}, which can be done exactly using (10.38). It also requires computing the expression of $d\varphi_{t1}^{v^{\tau}}(x_i^{v^{\tau}}(t))$, which can be obtained from the expression

$$\partial_t d\varphi_{t1}^{v} \circ \varphi_{1t} = \partial_t (d\varphi_{1t})^{-1} = -(d\varphi_{1t})^{-1}(\partial_t(d\varphi_{1t}))(d\varphi_{1t})^{-1},$$

which yields:

$$\partial_t d\varphi_{t1}^{v}(x_i^{v}(t)) = -d\varphi_{t1}^{v}(x_i^{v}(t))dv(t, x_i^{v}(t)).$$

Thus, $d\varphi_{t1}^{v}(x_i^{v}(t))$ is a solution of $\partial_t M = -M dv(x_i^{v}(t))$ with initial condition $M = \mathrm{Id}$. The matrix $dv(t, x_i^{v}(t))$ can be computed explicitly as a function of the point trajectories $x_j^{v}(t)$, $j = 1, \ldots, N$, using the explicit expression (10.38). This algorithm was introduced in [31].

10.6.2 Gradient in the Hamiltonian Form

As we have seen, one can use the optimal control formalism with the Pontryagin principle to compute the gradient of $\tilde{E}$ in v. Given $v \in \mathcal{X}_V^2$, this gradient can be computed by solving (10.28) with boundary conditions $\varphi(0) = \mathrm{id}$ and $\mu(1) = -dU(\varphi(1))$ (which can be achieved by solving the first equation in (10.28) from $t = 0$ to $t = 1$, then the second one backward in time, from $t = 1$ to $t = 0$) and, using (10.18), letting

$$\nabla\tilde{E}(v)(t) = \mathbb{K}(d\tilde{E}(v)(t)) = -\mathbb{K}\xi_{\varphi(t)}^{*}\mu(t) + v(t).$$

This equation (or the maximum principle) implies that the optimal v must be such that $v(t) = \mathbb{K}\xi^*_{\varphi(t)}\mu(t)$ for some $\mu \in C_0^{p-1}(\Omega, \mathbb{R}^d)^*$ and there is therefore no loss of generality in restricting the optimization problem to v's taking this form. With this constraint, we have

$$\|v(t)\|_V^2 = \left\langle \mathbb{K}\xi^*_{\varphi(t)}\mu(t)\,,\ \mathbb{K}\xi^*_{\varphi(t)}\mu(t)\right\rangle_V = \left(\mu(t)\ \middle|\ \xi_{\varphi(t)}\mathbb{K}\xi^*_{\varphi(t)}\mu(t)\right).$$

Let $\mathbb{K}_\varphi = \xi_\varphi \mathbb{K}\xi^*_\varphi$ so that $\|v(t)\|_V^2 = \left(\mu(t)\ \middle|\ \mathbb{K}_{\varphi(t)}\mu(t)\right)$. One has

$$\begin{aligned} a^T(\mathbb{K}_\varphi\mu)(y) &= a^T(\mathbb{K}\xi^*_\varphi\mu)(\varphi(y)) \\ &= \left\langle K(\cdot,\varphi(y))a\,,\ \mathbb{K}\xi^*_\varphi\mu\right\rangle_V \\ &= (\mu \mid K(\varphi(x),\varphi(y))a\,)_x \end{aligned}$$

so that

$$(\mathbb{K}_\varphi\mu)(y) = \sum_{i=1}^d (\mu \mid K(\varphi(x),\varphi(y))e_i\,)_x e_i.$$

With this notation, the state equation $\partial_t\varphi = v \circ \varphi$ becomes $\partial_t\varphi = \mathbb{K}_\varphi\mu$ and the original optimal control problem is reformulated as minimizing

$$E(\varphi,\mu) = \frac{1}{2}\int_0^1 \left(\mu(t)\ \middle|\ \mathbb{K}_{\varphi(t)}\mu(t)\right)dt + U(\varphi_{01})$$

subject to $\partial_t\varphi = \mathbb{K}_\varphi\mu$.

Expressing the problem in this form slightly changes the expression of the differential. The computation of the gradient (and its justification) based on a co-state α and the Hamiltonian

$$H_\mu(\alpha,\varphi) = \left(\alpha\ \middle|\ \mathbb{K}_\varphi\mu\right) - \frac{1}{2}\left(\mu\ \middle|\ \mathbb{K}_\varphi\mu\right)$$

are obtained using the same methods as in Sect. 10.4.2, so we skip the details. Let φ^μ be the solution of $\partial_t\varphi = \mathbb{K}_\varphi\mu$ with $\varphi(0) = \mathrm{id}$. Then, with $\tilde{E}(\mu) = E(\varphi^\mu,\mu)$, we have

$$d\tilde{E}(\mu) = \mathbb{K}_\varphi\alpha - \mathbb{K}_\varphi\mu,$$

where φ and α are given by the system

$$\begin{cases} \partial_t \varphi = \mathbb{K}_\varphi \mu \\ \left(\partial_t \alpha(t) \,\middle|\, w\right) = -\sum_{i=1}^{d} \left(\alpha(t) \mid \left(\mu(t) \mid \partial_2 K^i(\varphi(t,x), \varphi(t,y)) w(y)\right)_x e_i\right)_y \\ \qquad - \sum_{i=1}^{d} \left(\mu(t) \mid \left(\alpha(t) \mid \partial_2 K^i(\varphi(t,x), \varphi(t,y)) w(y)\right)_x e_i\right)_y \\ \qquad + 2\sum_{i=1}^{d} \left(\mu(t) \mid \left(\mu(t) \mid \partial_2 K^i(\varphi(t,x), \varphi(t,y)) w(y)\right)_x e_i\right)_y \end{cases} \tag{10.40}$$

with $\varphi(0) = \mathrm{id}$ and $\alpha(1) = -dU(\varphi(1))$. Unsurprisingly, this system boils down to (10.28) when $d\tilde{E}(\mu) = 0$, i.e., when $\alpha = \mu$.

The gradient of $\tilde{E}$ expressed with respect to the inner product

$$\langle \mu\,,\, \tilde{\mu} \rangle_\varphi = \left(\mu \mid K_\varphi \mu\right).$$

(this choice will be justified in Sect. 11.4 as the dual Riemannian metric on the diffeomorphism group) is

$$\nabla \tilde{E}(\mu) = \alpha - \mu,$$

a remarkably simple expression.

Consider now the case in which $U(\varphi) = F(\varphi \cdot I)$ where I is a fixed object. With the notation and assumptions made in Sect. 10.6.1, we found in Lemma 10.14 that there was no loss of generality in restricting the minimization to $v(t) \in V_{J(t)}$ at all times. This often entails additional constraints on the momentum $\rho(t) = \mathbb{L}v(t)$, or on $\mu(t) = \xi^*_{\varphi(t)^{-1}} \rho(t)$, that can be leveraged to reduce the dimension of the control variable. For example, we have seen that for point sets (in which we let $J = x$) V_x was given by (10.37), so that $\rho(t)$ must take the form

$$\rho(t) = \sum_{k=1}^{N} z_k(t) \delta_{x_k(t)}$$

for some $z_1(t), \ldots, z_N(t) \in \mathbb{R}^d$, from which we can deduce (using $x_k(t) = \varphi(t, x_k(0))$) that $\mu(t)$ must take the form

$$\mu(t) = \sum_{k=1}^{N} z_k(t) \delta_{x_k(0)}.$$

One can then use $z_1, \ldots, z_N$ as a new control, as described below.

Another interesting special case is when $\mu(t)$ can be expressed as a vector measure, because, as discussed in Sect. 10.5.6, one can then assume that

$$\mu(t) = \sum_{k=1}^{N} z_k(t, \cdot)\gamma_k,$$

where $\gamma_1, \ldots, \gamma_N$ are fixed measures. One can then use the vector fields $z_1, \ldots, z_N$ to parametrize the problem. This leads to a simplification when the measures have a sparse support. They are, for example, Dirac measures for point matching. We now review this case in more detail.

Application to Point Matching

When $U(\varphi) = F(\varphi(x_1), \ldots, \varphi(x_N))$, we have

$$(dU(\varphi) \mid h) = \sum_{k=1}^{N} \partial_k F(\varphi(x_1), \ldots, \varphi(x_N))^T h(x_k)$$

so that

$$\mu(1) = -\sum_{k=1}^{N} \partial_k F(\varphi(x_1), \ldots, \varphi(x_N))\delta_{x_k}$$

is a vector measure. We can therefore look for a solution in the form

$$\mu(t) = \sum_{k=1}^{N} z_k(t)\delta_{x_k}$$

at all times, for some coefficients $z_1, \ldots, z_N$.

In order to obtain α in (10.40) given a current μ, it suffices to solve the first equation only for the values of $y_k(t) = \varphi(t, x_k)$, $k = 1, \ldots, N$, which requires us to solve the system

$$\partial_t y_k = \sum_{l=1}^{N} K(y_k, y_l)\xi_l.$$

One then sets

$$\alpha(1) = -\sum_{k=1}^{N} \partial_k F(y_1(1), \ldots, y_N(1))\delta_{x_k}$$

and solves the second equation backward in time, knowing that the solution will take the form

$$\alpha(t) = \sum_{k=1}^{N} \eta_k(t)\delta_{x_k}$$

with $\eta_k(1) = -\partial_k F(y_1(1), \ldots, y_n(1))$ and

$$\begin{aligned}\partial_t \eta_k =& \sum_{l=1}^{N}\sum_{i,j=1}^{d} \eta_l^i z_k^j \nabla_1 K^{ij}(y_k, y_l) + \sum_{l=1}^{N}\sum_{i,j=1}^{d} z_l^i \eta_k^j \nabla_1 K^{ij}(y_k, y_l)\\ &- 2\sum_{l=1}^{N}\sum_{i,j=1}^{d} z_l^i z_k^j \nabla_1 K^{ij}(y_k, y_l).\end{aligned}$$

Given this, we have

$$\nabla \tilde{E}(\mu) = \sum_{k=1}^{N}(\eta_k - z_k)\delta_{x_k}.$$

10.6.3 Gradient in the Initial Momentum

We now use the fact that Eq. (10.16) implies that the optimal $v(t)$ is uniquely constrained by its value at $t = 0$ for formulating the variations of the objective function in terms of these initial conditions. We therefore optimize with respect to v_0, or equivalently with respect to $\mu_0 = \rho_0$. This requires finding ρ_0 such that

$$\frac{1}{2}\int_0^1 \|v(t)\|_V^2 dt + U(\varphi(1))$$

is minimal under the constraints $\partial_t\varphi(t) = v(t) \circ \varphi(t)$, with

$$v(t) = \sum_{i=1}^{d} \left(\rho_0 \mid d\varphi(t) K^{(i)}(x, y)\right)_x e_i.$$

Proposition 10.12 helps us to simplify this expression, since it implies that $\int_0^1 \|v(t)\|^2 dt = (\rho_0 \mid \mathbb{K}\rho_0)$ and the minimization problem therefore is to find ρ_0 such that

$$E(\rho_0) = \frac{1}{2}(\rho_0 \mid \mathbb{K}\rho_0) + U(\varphi(1))$$

is minimal, where (φ, μ) is a solution of system (10.28) with initial conditions $\varphi(0) = \mathrm{id}$ and $\mu(0) = \rho_0$. Writing (10.28) as

$$\partial_t \begin{pmatrix}\varphi\\ \mu\end{pmatrix} = \begin{pmatrix}\mathscr{V}_1(\varphi, \mu)\\ \mathscr{V}_2(\varphi, \mu)\end{pmatrix} = \mathscr{V}(\varphi, \mu)$$

and applying Proposition D.12, we have

$$dE(\rho_0) = \mathbb{K}\rho_0 - p_\mu(0),$$

where the pair $\binom{p_\varphi(t)}{p_\mu(t)}$ satisfies $p_\varphi(1) = -dU(\varphi(1))$, $p_\mu(0) = 0$ and

$$\begin{cases} \partial_t p_\varphi = -\partial_1 \mathscr{V}_1^* \, p_\varphi - \partial_1 \mathscr{V}_2^* \, p_\mu, \\ \partial_t p_\mu = -\partial_2 \mathscr{V}_1^* \, p_\varphi - \partial_2 \mathscr{V}_2^* \, p_\mu. \end{cases} \tag{10.41}$$

The gradient of E with respect to the metric on V^* is then given by $\nabla E(\rho_0) = \rho_0 - \mathbb{K}^{-1} p_\mu$.

The practical application of these formulas requires us to make explicit the expressions of $\partial_i \mathscr{V}_j^*$ for $i, j = 1, 2$. Returning to (10.28), we have

$$\begin{aligned} \big(\partial_1 \mathscr{V}_1^* \, p_\varphi \,\big|\, h\big) = & \sum_{i=1}^d \big(p_\varphi \,\big|\, \big(\mu \,\big|\, \partial_1 K^i(\varphi(x), \varphi(y)) h(x)\big)_x e_i\big)_y \\ & + \sum_{i=1}^d \big(p_\varphi \,\big|\, \big(\mu \,\big|\, \partial_2 K^i(\varphi(x), \varphi(y)) h(y)\big)_x e_i\big)_y, \end{aligned}$$

$$\big(\partial_2 \mathscr{V}_1^* \, p_\varphi \,\big|\, \eta\big) = \sum_{i=1}^d \big(p_\varphi \,\big|\, \big(\eta \,\big|\, K^i(\varphi(x), \varphi(y))\big)_x e_i\big)_y,$$

$$\begin{aligned} \big(\partial_1 \mathscr{V}_2^* \, p_\mu \,\big|\, h\big) = & - \sum_{i=1}^d \big(\mu \,\big|\, \big(\mu \,\big|\, \partial_1 \partial_2 K^i(\varphi(x), \varphi(y))(h(x), p_\mu(y))\big)_x e_i\big)_y \\ & - \sum_{i=1}^d \big(\mu \,\big|\, \big(\mu \,\big|\, \partial_2^2 K^i(\varphi(x), \varphi(y))(h(y), p_\mu(y))\big)_x e_i\big)_y, \text{ and} \end{aligned}$$

$$\begin{aligned} \big(\partial_2 \mathscr{V}_2^* \, p_\mu \,\big|\, \eta\big) = & - \sum_{i=1}^d \big(\eta \,\big|\, \big(\mu \,\big|\, \partial_2 K^i(\varphi(x), \varphi(y)) p_\mu(y)\big)_x e_i\big)_y \\ & - \sum_{i=1}^d \big(\mu \,\big|\, \big(\eta \,\big|\, \partial_2 K^i(\varphi(x), \varphi(y)) p_\mu(y)\big)_x e_i\big)_y. \end{aligned}$$

Forming explicit expressions of $\partial_i \mathscr{V}_j^*$ requires isolating h or η from the right-hand sides. To do this, we will need to change the order in which linear forms are applied to the x and y coordinates. This issue is addressed in the following lemma.

Lemma 10.15 *Assume that $\mu \in C^r(\Omega, \mathbb{R}^d)^*$ and $\nu \in C^{r'}(\Omega, \mathbb{R}^d)^*$. Let $g : \Omega \times \Omega \to \mathbb{R}$ be a function such that $\partial_1^k \partial_2^{k'} g \in C_0(\Omega \times \Omega, \mathbb{R})$ for all $k \le r$ and $k' \le r$. Then, for all $a, b \in \mathbb{R}^d$, $(\mu \mid g(x, \cdot)a)_x \in C^{r'}(\Omega, \mathbb{R}^d)$ and $(\nu \mid g(\cdot, y)b)_y \in C^r(\Omega, \mathbb{R})$, with*

$$\big(\mu \,\big|\, (\nu \mid g(x, y)b)_y a\big)_x = \big(\nu \,\big|\, (\mu \mid g(x, y)a)_x b\big)_y. \tag{10.42}$$

Proof Let $f(y) = (\mu \mid g(x, y)a)_x$. Using Taylor's formula, we can write

$$g(x, y+h) = \sum_{k=0}^{r'} \frac{1}{k!} \partial_2^k g(x, y) h^{(k)} + \\ \frac{1}{(r'-1)!} \int_0^1 (\partial_2^{r'} g(x, y+th) - \partial_2^{r'} g(x, y)) h^{(r')} (1-t)^{r'-1}\, dt$$

so that

$$f(y+h) = \sum_{k=0}^{r'} \frac{1}{k!} \big(\mu \mid \partial_2^k g(x, y) h^{(k)} a\big) \\ + \frac{1}{(r'-1)!} \Big(\mu \Big| \int_0^1 (\partial_2^{r'} g(x, y+th) - \partial_2^{r'} g(x, y)) h^{(r')} (1-t)^{r'-1} a\, dt\Big).$$

The last term (call it R) is such that

$$|R| \le \frac{\|\mu\|_{r,\infty,*} |h|^{r'} |a|}{r'!} \max_{t\in[0,1]} \|\partial_2^{r'} g(\cdot, y+th) - \partial_2^{r'} g(\cdot, y)\|_{r,\infty}.$$

The uniform continuity of $\partial_1^k \partial_2^{r'} g$ for $k \le r$ implies that $R = o(|h|^{r'})$ so that $f \in C^{r'}(\Omega, \mathbb{R})$. Similarly, letting $f'(x) = (\mu \mid g(x, y)b)_y$, one has $f' \in C^r(\Omega, \mathbb{R})$.

The computation also shows that, for some constant C,

$$\max\big(\big(\mu \mid (\nu \mid g(x, y)b)_y a\big)_x, \big(\nu \mid (\mu \mid g(x, y)a)_x b\big)_y\big) \\ \le C\|\mu\|_{r,\infty,*} \|\nu\|_{r',\infty,*} \|g\|_{r,r',\infty}$$

with

$$\|g\|_{r,r',\infty} = \max_{k\le r, k'\le r'} \|\partial_1^k \partial_2^{k'} g\|_\infty,$$

so that both sides of (10.42) are continuous in g with respect to this norm. To conclude, it suffices to notice that (10.42) is true when g takes the form

$$g(x, y) = \sum_{k=1}^n c_k f_k(x) f_k'(y)$$

and that these functions form a dense set for $\|g\|_{r,r',\infty}$, so that the identity extends by continuity. □

Let us use this lemma to identify the first term in $\big(\partial_1 \mathscr{V}_1^* p_\varphi \mid h\big)$ as a linear form acting on h. Write, letting $\partial_{i,k}$ denote the derivative with respect to the kth coordinate of the ith variable,

$$\begin{aligned}
&\sum_{i=1}^{d}\big(p_\varphi\;\big|\;\big(\mu\;\big|\;\partial_1 K^i(\varphi(x),\varphi(y))h(x)\big)_x e_i\big)_y\\
&=\sum_{i,j,k=1}^{d}\big(p_\varphi\;\big|\;\big(\mu\;\big|\;\partial_{1,k}K^{ij}(\varphi(x),\varphi(y))h^k(x)e_j\big)_x e_i\big)_y\\
&=\sum_{i,j,k=1}^{d}\Big(\mu\;\Big|\;\big(p_\varphi\;\big|\;\partial_{1,k}K^{ij}(\varphi(x),\varphi(y))h^k(x)e_i\big)_y e_j\Big)_x\\
&=\sum_{i,j,k=1}^{d}\Big(\mu\;\Big|\;\big(p_\varphi\;\big|\;\partial_{2,k}K^{ji}(\varphi(y),\varphi(x))e_i\big)_y h^k(x)e_j\Big)_x\\
&=\sum_{j,k=1}^{d}\Big(\mu\;\Big|\;\big(p_\varphi\;\big|\;\partial_{2,k}K^{j}(\varphi(y),\varphi(x))\big)_y h^k(x)e_j\Big)_x\\
&=(\mu\mid\mathscr{U}_1 h),
\end{aligned}$$

where $\mathscr{U}_1(x)$ is the matrix with coefficients

$$\mathscr{U}_1^{j,q}(x)=\big(p_\varphi\;\big|\;\partial_{2,q}K^j(\varphi(y),\varphi(x))\big)_y.$$

Write $\big(\mathscr{U}_1^T\mu\;\big|\;h\big)=(\mu\mid\mathscr{U}_1 h)$, a notation generalizing the one introduced for vector measures. After a similar computation for the second term of $\big(\partial_1\mathscr{V}_1^*\,p_\varphi\;\big|\;h\big)$ (which does not require Lemma 10.15), we get

$$\partial_1\mathscr{V}_1^*\,p_\varphi=\mathscr{U}_1^T\mu+\mathscr{U}_2^T p_\varphi$$

with

$$\mathscr{U}_2^{j,q}(x)=\big(\mu\;\big|\;\partial_{2,q}K^j(\varphi(y),\varphi(x))\big)_y.$$

Consider now $\partial_2\mathscr{V}_1^*\,p_\varphi$, writing

$$\begin{aligned}
\big(\partial_2\mathscr{V}_1^*\,p_\varphi\;\big|\;\eta\big)&=\sum_{i,j=1}^{d}\Big(\eta\;\Big|\;\big(p_\varphi\;\big|\;K^{ij}(\varphi(x),\varphi(y))e_i\big)_y e_j\Big)_x\\
&=\sum_{j=1}^{d}\Big(\eta\;\Big|\;\big(p_\varphi\;\big|\;K^{j}(\varphi(y),\varphi(x))\big)_y e_j\Big)_x,
\end{aligned}$$

so that

$$\partial_2\mathscr{V}_1^*\,p_\varphi(x)=\sum_{j=1}^{d}\big(p_\varphi\;\big|\;K^j(\varphi(y),\varphi(x))\big)_y e_j.$$

With similar computations for $\mathscr{V}_2$, and skipping the details, we find

$$\partial_1 \mathscr{V}_2^* \, p_\mu = -\mathscr{U}_3^T \mu - \mathscr{U}_4^T \mu$$

where

$$\mathscr{U}_3^{jq}(x) = \left(\mu \,\middle|\, \partial_{2,q}(\partial_1 K^j(\varphi(y), \varphi(x)) p_\mu(y))\right)_y, \text{ and}$$
$$\mathscr{U}_4^{jq}(x) = \left(\mu \,\middle|\, \partial_{2,q}(\partial_2 K^j(\varphi(y), \varphi(x)) p_\mu(x))\right)_y.$$

Finally,

$$\partial_2 \mathscr{V}_2^* \, p_\mu(x) = -\sum_{i=1}^{d} \left(\mu \,\middle|\, \partial_2 K^i(\varphi(y), \varphi(x)) p_\mu(x)\right)_y e_i$$
$$- \sum_{i=1}^{d} \left(\mu \,\middle|\, \partial_1 K^i(\varphi(y), \varphi(x)) p_\mu(y)\right)_y e_i.$$

Let us take the special case of vector measures, assuming that $\mu(t) = \sum_{k=1}^N z_k(t, \cdot)\gamma_k$. We will look for p_φ in the form

$$p_\varphi(t) = \sum_{k=1}^{N} \alpha_k(t, \cdot)\gamma_k,$$

p_μ being a function defined over the support of μ.

With these assumptions, we have

- $\partial_1 \mathscr{V}_1^* \, p_\varphi = \sum_{k=1}^{N} \zeta_k^{1,1} \gamma_k$ with

$$\zeta_k^{1,1}(x) = \sum_{l=1}^{N} \sum_{i,j=1}^{d} \left(\gamma_l \,\middle|\, \alpha_l^i(y) z_k^j(x) \nabla_1 K^{ij}(\varphi(x), \varphi(y))\right)_y$$
$$+ \sum_{l=1}^{N} \sum_{i,j=1}^{d} \left(\gamma_l \,\middle|\, z_l^i(y) \alpha_k^j(x) \nabla_1 K^{ij}(\varphi(x), \varphi(y))\right)_y.$$

- $\partial_2 \mathscr{V}_1^* \, p_\varphi(x) = \sum_{k=1}^{N} (\gamma_k \mid K(\varphi(x), \varphi(y)) \alpha_k(y))_y.$
- $\partial_1 \mathscr{V}_2^* \, p_\mu = \sum_{k=1}^{N} \zeta_k^{2,1} \gamma_k$ with

$$\zeta_k^{2,1}(x) = -\sum_{i,j=1}^{d}\sum_{l=1}^{N}\Big(\gamma_l \ \Big|\, z_l^i(y) z_k^j(x)\partial_1\partial_2 K^{ij}(\varphi(x),\varphi(y))p_\mu(y)\Big)_y$$
$$-\sum_{i,j=1}^{d}\sum_{l=1}^{N}\Big(\gamma_l \ \Big|\, z_l^i(y) z_k^j(x)\partial_1^2 K^{ij}(\varphi(x),\varphi(y))p_\mu(x)\Big)_y .$$

- $\partial_2 \mathscr{V}_2^* \, p_\mu(x) = -\sum_{i=1}^{d}\sum_{k=1}^{N}\big(\gamma_k \mid z_k^i(y)\partial_1 K^i(\varphi(x),\varphi(y))p_\mu(x)\big)_y$
$$-\sum_{i=1}^{d}\sum_{k=1}^{N}\big(\gamma_k \mid z_k^i(y)\partial_2 K^i(\varphi(x),\varphi(y))p_\mu(y)\big)_y .$$

System (10.41) can now be simplified as

$$\begin{cases} \partial_t \alpha_k = \zeta_k^{1,1} + \zeta_k^{2,1} \\ \partial_t p_\mu = \sum_{k=1}^{N} (\gamma_k \mid K(\varphi(x),\varphi(y))\alpha_k(y))_y \\ \qquad -\sum_{i=1}^{d}\sum_{k=1}^{N}\big(\gamma_k \mid z_k^i(y)\partial_1 K^i(\varphi(x),\varphi(y))p_\mu(x)\big)_y \\ \qquad -\sum_{i=1}^{d}\sum_{k=1}^{N}\big(\gamma_k \mid z_k^i(y)\partial_2 K^i(\varphi(x),\varphi(y))p_\mu(y)\big)_y . \end{cases} \tag{10.43}$$

Application to Point Matching

We now apply this approach to point-matching problems. Since ρ_0 takes the form

$$\rho_0 = \sum_{k=1}^{N} a_{0,k}\delta_{x_{0,k}}$$

we are in the vector measure case with $\gamma_k = \delta_{x_{0,k}}$. The densities z_k and α_k for μ and p_φ can therefore be considered as vectors in $\mathbb{R}^d$, and p_μ being defined on the support of μ is also a collection of vectors $p_{\mu,k} = p_\mu(x_k)$. Given this, we can therefore immediately rewrite

- $\partial_1 \mathscr{V}_1^* \, p_\varphi = \sum_{k=1}^{N} \zeta_k^{1,1}\delta_{x_{0,k}}$ with

$$\zeta_k^{1,1} = \sum_{l=1}^{N}\sum_{i,j=1}^{d}\Big(\alpha_l^i \nabla_1 K^{ij}(x_k,x_l) z_k^j + z_l^i \nabla_1 K^{ij}(x_k,x_l)\alpha_k^j\Big).$$

- $\partial_2 \mathscr{V}_1^* \, p_\varphi(x_{0,k}) = \sum_{l=1}^N K(x_k, x_l)\alpha_l$.
- $\partial_1 \mathscr{V}_2^* \, p_\mu = \sum_{k=1}^N \zeta_k^{2,1} \delta_{x_{0,k}}$ with

$$\zeta_k^{2,1} = -\sum_{i,j=1}^d \sum_{l=1}^N z_l^i z_k^j (\partial_1 \partial_2 K^{ij}(x_k, x_l)\, p_{\mu,l} + \partial_1^2 K^{ij}(x_k, x_l)\, p_{\mu,k}).$$

- $\partial_2 \mathscr{V}_2^* \, p_\mu(x_k) = -\sum_{i=1}^d \sum_{l=1}^N z_l^i (\partial_1 K^i(x_k, x_l) p_{\mu,k} + \partial_2 K^i(x_k, x_l) p_{\mu,l})$.

This algorithm is illustrated in Fig. 10.1. In the same figure, we also provide (for comparison purposes) the results provided by spline interpolation, which computes $\varphi(x) = x + v(x)$, where v is computed (using Theorem 8.9) in order to minimize

$$\|v\|_V^2 + C \sum_{i=1}^N |v(x_i) - (y_i - x_i)|^2 .$$

Although this is a widely spread registration method [42], Fig. 10.1 shows that it is far from being diffeomorphic for large deformations.

10.6.4 Shooting

The optimality conditions for our problem are $\mu(1) = -dU(\varphi(1))$ with $\mu(t)$ given by (10.28). The shooting approach in optimal control consists in finding an initial momentum $\rho_0 = \mu(0)$ such that these conditions are satisfied. Root finding methods, such as Newton's algorithm, can be used for this purpose. At a given step of Newton's algorithm, one modifies the current value of ρ_0, by letting $\rho_0 \to \rho_0 + \eta$ such that, letting $F(\rho_0) := \mu(1) + dU(\varphi(1))$, one has

$$F(\rho_0) + dF(\rho_0)\eta = 0.$$

One therefore needs to solve this linear equation in order to update the current ρ_0. One has

$$dF(\rho_0) = W_{\mu\mu}(1) + W_{\varphi\mu}(1)^* d^2U(\varphi(1)),$$

where

$$W = \begin{pmatrix} W_{\varphi\varphi} & W_{\varphi\mu} \\ W_{\mu\varphi} & W_{\mu\mu} \end{pmatrix}$$

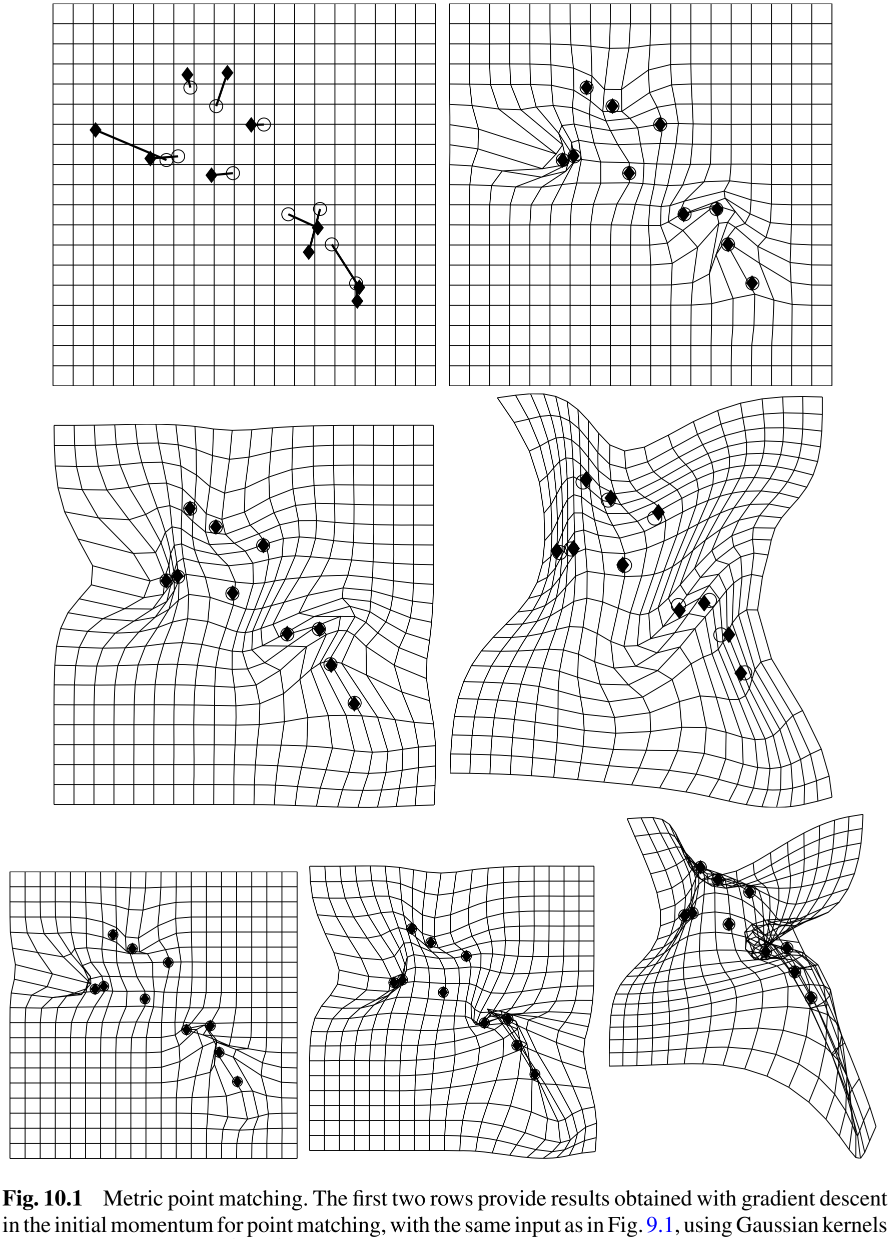

Fig. 10.1 Metric point matching. The first two rows provide results obtained with gradient descent in the initial momentum for point matching, with the same input as in Fig. 9.1, using Gaussian kernels $K(x, y) = \exp(-|x - y|^2/2\sigma^2)$ with $\sigma = 1, 2, 4$ in grid units. The impact of the diffeomorphic regularization on the quality of the result is particularly obvious in the last experiment. The last row provides the output of Gaussian spline registration with the same kernels, exhibiting singularities and ambiguities in the registration

is (using the notation of the previous section) the differential of the solution of the equation

$$\partial_t \begin{pmatrix} \varphi \\ \mu \end{pmatrix} = \mathscr{V}(\varphi, \mu),$$

with respect to its initial condition, i.e., the solution of

$$\partial_t W = d\mathscr{V}(\varphi, \mu) W$$

with initial condition $W(0) = \mathrm{Id}$.

Because one needs to compute the solution of this differential equation at every step of the algorithm, then solve for a linear system, the shooting method is feasible only for problems that can be discretized into a relatively small number of dimensions. One can use it, for example, in point matching problems with no more than a few hundred landmarks (see [290] for an application to labeled point matching), in which case the algorithm can be very efficient. Another issue is that root finding algorithms are not guaranteed to converge. Usually, a good initial solution must be found, using, for example, a few preliminary steps of gradient descent.

10.6.5 Gradient in the Deformable Object

Finally, we consider the option of using the time derivative of the deformable object as a control variable, using the fact that, by (10.33), the objective function can be reduced to

$$E(J) = \int_0^1 L(\partial_t J(t), J(t))dt + F(J(1))$$

with $L(\eta, J) = \min_{w:\, \eta = w\cdot J(t)} \|w\|_V^2$. This formulation is limited, in that $L(\eta, J)$ is not always defined for all (η, J), resulting in constraints in the minimization that are not always easy to handle. Even if well-defined, the computation of L may be numerically demanding. To illustrate this, consider the image-matching case, in which $v \cdot J = -\nabla J^T v$. An obvious constraint is that, in order for

$$\nabla J^T w = -\eta$$

to have at least one solution, the variation η must be supported by the set $\nabla J \neq 0$. To compute this solution when it exists, one can write, for $x \in \Omega$,

$$\nabla J(x)^T w(x) = \big\langle K(\cdot, x)\nabla J(x)\ ,\ w \big\rangle_V,$$

and it is possible to look for a solution in the form

$$w(y) = \int_{\Omega} \lambda(x) K(y, x) \nabla J(x) dx,$$

where $\lambda(x)$ can be interpreted as a continuous family of Lagrange multipliers. This results in a linear equation in λ, namely

$$\int_{\Omega} \lambda(x) K(y, x) \nabla J(y)^T \nabla J(x)) dx = -\eta(y),$$

which is numerically challenging.

For point sets, however, the approach is feasible [159] because L can be made explicit. Given a point-set trajectory $x(t) = (x^{(1)}(t), \ldots, x^{(N)}(t))$, let $S(x(t))$ denote the block matrix with (i, j) block given by $K(x^{(i)}(t), x^{(j)}(t))$. The constraints are $x_t = S(x(t))\xi(t)$ so that $\xi(t) = S(x(t))^{-1}\dot{x}_t$ and the minimization reduces to

$$E(x) = \frac{1}{2} \int_0^1 \dot{x}(t)^T S(x(t))^{-1} \dot{x}(t) dt + U(x(1)).$$

Minimizing this function with respect to x by gradient descent is possible, and has been described in [158, 159] for labeled landmark matching. The basic computation is as follows: if $s_{pq,r} = \partial_r s_{pq}$, we can write (using the fact that $\partial_r(S^{-1}) = -S^{-1}(\partial_r S)S^{-1}$)

$$\begin{aligned}(dE(x) \mid h) = & \int_0^1 \dot{x}(t)^T S(x(t))^{-1} \dot{h}(t) dt \\ & - \int_0^1 \sum_{p,q,r} \xi^{(p)}(t) \xi^{q)}(t) s_{pq,r}(x(t)) h^{(r)}(t) dt + \nabla U(x(1))^T h(1).\end{aligned}$$

After an integration by parts in the first integral, we obtain

$$dE(x) = -\partial_t \left(S(x(t))^{-1}\dot{x}\right) - z(t) + \left(S(x(1))^{-1}\dot{x}(1) + \nabla U(x(1))\right) \delta_1(t),$$

where $z_r(t) = \sum_{p,q} \xi_p(t) \xi_q(t) s_{pq,r}(x(t))$ and δ_1 is the Dirac measure at $t = 1$.

This singular part can be dealt with by computing the gradient in a Hilbert space in which the evaluation function $x(\cdot) \mapsto x(1)$ is continuous. This method has been suggested, in particular, in [129, 161]. Let H be the space of all trajectories $x : t \mapsto x(t) = (x^{(1)}(t), \ldots, x^{(N)}(t))$, with fixed starting point $x(0)$, free end-point $x(1)$ and square integrable time derivative. This is a space of the form $x(0) + H$ where H is the Hilbert space of time-dependent functions $t \mapsto h(t)$, considered as column vectors of size Nk, with $h(0) = 0$ and

$$\left\langle h \,,\, \tilde{h} \right\rangle_H = \int_0^1 \dot{h}^T \dot{\tilde{h}} dt + h(1)^T \tilde{h}(1).$$

To compute the gradient for this inner product, we need to write $(dE(x) \mid h)$ in the form $\left\langle \nabla^H E(x)\, ,\, h\right\rangle_H$. We will make the assumption that

$$\int_0^1 \left|S(x(t))^{-1}\dot{x}(t)\right|^2 dt < \infty,$$

which implies that

$$\int_0^1 \dot{x}(t)^T S(x(t))^{-1}\dot{h}(t)dt \leq \sqrt{\int_0^1 \left|S(x(t))^{-1}\dot{x}(t)\right|^2 dt \int_0^1 \left|\dot{h}(t)\right|^2 dt}$$

is continuous in h. Similarly, the linear form $\xi \mapsto \nabla U(x(1))^T h(1)$ is continuous since

$$\nabla U(x(1))^T h(1) \leq |\nabla U(x(1))|\, |h(1)|\, .$$

Finally, $h \mapsto \int_0^1 z(t)^T \dot{h} dt$ is continuous provided that we assume that

$$\eta(t) = \int_0^t z(s)dt$$

is square integrable over [0, 1], since this yields

$$\int_0^1 z(t)^T h(t)dt = \eta(1)h(1) - \int_0^1 \eta(t)\dot{h}(t)dt,$$

which is continuous in h with respect to the H norm.

Thus, under these assumptions, $h \mapsto (dE(x) \mid h)$ is continuous over H, and the Riesz representation theorem implies that $\nabla^H E(x)$ exists as an element of H. We now proceed to its computation. Letting

$$\mu(t) = \int_0^t S(x(s))^{-1}\dot{h}(s)ds$$

and $a = \nabla U(x(1))$, the problem is to find $\zeta \in H$ such that, for all $h \in H$,

$$\left\langle \zeta\, ,\, h\right\rangle_H = \int_0^1 \dot{\mu}^T \dot{h} dt + \int_0^1 z(t)^T h(t)dt + a^T \xi(1).$$

This expression can also be written

$$\int_0^1 \left(\dot{\zeta} + \zeta(1)\right)^T \dot{h} dt = \int_0^1 (\dot{\mu} + \eta(1) - \eta(t) + a)^T \dot{h} dt.$$

This suggests selecting ζ such that $\zeta(0) = 0$ and

$$\dot{\zeta} + \zeta(1) = \dot{\mu} + \eta(1) - \eta(t) + a,$$

which implies

$$\zeta(t) + t\zeta(1) = \mu(t) - \int_0^t \eta(s)ds + t(\eta(1) + a).$$

At $t = 1$, this yields

$$2\zeta(1) = \mu(1) - \int_0^1 \eta(s)ds + \eta(1) + a$$

and we finally obtain

$$\zeta(t) = \mu(t) - \int_0^t \eta(s)ds + \frac{t}{2}\left(\int_0^1 \eta(s)ds - \mu(1) + \eta(1) + a\right).$$

We summarize this in an algorithm, in which τ is again the computation time.

Algorithm 3 (*Gradient descent algorithm for landmark matching*) Start with initial landmark trajectories $x(t, \tau) = (x^{(1)}(t, \tau), \ldots, x^{(N)}(t, \tau))$.

Solve

$$\begin{aligned}\partial_\tau x(t, \tau) = -\gamma\Big(\mu(t, \tau) - \int_0^t \eta(s, \tau)ds \\ + \frac{t}{2}\Big(\int_0^1 \eta(s, \tau)ds - \mu(1, \tau) + \eta(1, \tau) + a(\tau)\Big)\Big)\end{aligned}$$

with $a(\tau) = \nabla U(x(1, \tau))$, $\mu(t, \tau) = \int_0^t \xi(s, \tau)ds$, $\eta(t, \tau) = \int_0^t z(s, \tau)dt$ and

$$\begin{aligned}\xi(t, \tau) &= S(x(t, \tau))^{-1}\dot{x}(t, \tau) \\ z^{(q)}(t, \tau) &= \sum_{p,r} \xi^{(p)}(t, \tau)\xi^{(r)}(t, \tau)s_{pq,r}(x(t, \tau)).\end{aligned}$$

10.6.6 Image Matching

We now take an infinite-dimensional example to illustrate some of the previously discussed methods and focus on the image-matching problem. We therefore consider

$$U(\varphi) = \frac{\lambda}{2}\int_\Omega (I \circ \varphi^{-1} - \tilde{I})^2 dx,$$

where I, $\tilde{I}$ are functions $\Omega \to \mathbb{R}$, I being differentiable. The Eulerian differential of U is given by (9.21):

$$\bar{\partial} U(\varphi) = -\lambda (I \circ \varphi^{-1} - I') \nabla (I \circ \varphi^{-1}) dx.$$

So, according to (10.19), and letting $\tilde{U}(v) = U(\varphi_{01}^v)$,

$$\begin{aligned}
\nabla \tilde{U}(v)(t, y) &= \sum_{i=1}^{d} \left(\bar{\partial} U(\varphi_{01}^v) \;\middle|\; d\varphi_{t1}^v(\varphi_{1t}^v(.)) K(y, \varphi_{1t}^v(.)) e_i \right) e_i \\
&= -\lambda \sum_{i=1}^{d} e_i \int_{\Omega} (I \circ \varphi_{10}^v(x) - \tilde{I}(x)) \\
&\quad \nabla (I \circ \varphi_{10}^v)(x)^T d\varphi_{t1}^v(\varphi_{1t}^v(x)) K(y, \varphi_{1t}^v(x)) e_i dx \\
&= -\lambda \sum_{i=1}^{d} e_i \int_{\Omega} (I \circ \varphi_{10}^v(x) - \tilde{I}(x)) \\
&\quad \nabla I \circ \varphi_{10}^v(x)^T d\varphi_{10}^v(x) d\varphi_{t1}^v(\varphi_{1t}^v(x)) K(y, \varphi_{1t}^v(x)) e_i dx \\
&= -\lambda \sum_{i=1}^{d} e_i \int_{\Omega} (I \circ \varphi_{10}^v(x) - \tilde{I}(x)) \\
&\quad \nabla I \circ \varphi_{10}^v(x)^T d\varphi_{t0}^v(\varphi_{1t}^v(x)) K(y, \varphi_{1t}^v(x)) e_i dx \\
&= -\lambda \sum_{i=1}^{d} e_i \int_{\Omega} (I \circ \varphi_{10}^v(x) - \tilde{I}(x)) \nabla (I \circ \varphi_{t0}^v)(\varphi_{1t}^v(x))^T K(y, \varphi_{1t}^v(x)) e_i dx \\
&= -\lambda \sum_{i=1}^{d} e_i \int_{\Omega} (I \circ \varphi_{10}^v(x) - \tilde{I}(x)) e_i^T K(\varphi_{1t}^v(x), y) \nabla (I \circ \varphi_{t0}^v)(\varphi_{1t}^v(x)) dx \\
&= -\lambda \int_{\Omega} (I \circ \varphi_{10}^v(x) - \tilde{I}(x)) K(\varphi_{1t}^v(x), y) \nabla (I \circ \varphi_{t0}^v)(\varphi_{1t}^v(x)) dx.
\end{aligned}$$

This provides the expression of the V-gradient of $\tilde{E}$ for image matching, namely

$$\begin{aligned}
(\nabla^V E(v))(t, y) &= v(t, y) \\
&- \lambda \int_{\Omega} (I \circ \varphi_{10}^v(x) - \tilde{I}(x)) K(\varphi_{1t}^v(x), y) \nabla (I \circ \varphi_{t0}^v)(\varphi_{1t}^v(x)) dx.
\end{aligned} \tag{10.44}$$

Using a change of variable in the integral, the gradient may also be written as

$$\begin{aligned}
(\nabla^V E)(t, y) &= v(t, y) \\
- \lambda \int_{\Omega} (I \circ \varphi_{t0}^v(x) - \tilde{I} \circ \varphi_{t1}^v(x)) & K(x, y) \nabla (I \circ \varphi_{t0}^v)(x) \det(d\varphi_{t1}^v(x)) dx,
\end{aligned} \tag{10.45}$$

the associated gradient descent algorithm having been proposed in [32].

Let us now consider an optimization with respect to the initial ρ_0. First notice that, by (9.20), $\mu(1) = \lambda \det(d\varphi(1))(I - I' \circ \varphi(1))d\varphi(1)^{-T}\nabla I\, dx$ is a vector measure. Also, we have

$$\begin{aligned}(\rho_0 \mid w) &= (\mu(1) \mid d\varphi(1)w)\\ &= \lambda\big(dx \;\big|\det(d\varphi(1))(I - I' \circ \varphi(1))\nabla I^T w\big),\end{aligned}$$

which shows that one can assume that $\rho_0 = z_0 dx$ for some vector-valued function z_0 (with $z_0 = \det(d\varphi(1))(I - I' \circ \varphi(1))\nabla I$ for an optimal control).

We now make explicit the computation of the differential of the energy with respect to ρ_0. We have $\mu(t) = z(t, \cdot)dx$, with $z(0) = z_0$ and

$$\begin{cases}\partial_t\varphi(t, y) = \displaystyle\int_{\mathbb{R}^d} K(\varphi(t, y), \varphi(t, x))z(t, x)dx\\ \partial_t z(t, y) = -\displaystyle\int_{\mathbb{R}^d} z^i(t, y)z^j(t, x)\nabla_1 K^{ij}(\varphi(t, y), \varphi(t, x))dx.\end{cases} \tag{10.46}$$

The differential $dE(\rho_0) = K\rho_0 - p_\mu(0)$ is computed by solving, using $\alpha(1) = \lambda \det(d\varphi(1))(I - I' \circ \varphi(1))d\varphi(1)^{-T}\nabla I$ and $p_\mu(1) = 0$,

$$\begin{cases}\partial_t\alpha = \zeta^{1,1} + \zeta^{2,1}\\ \partial_t p_\mu = \displaystyle\int_{\mathbb{R}^d} K(\varphi(x), \varphi(y))\alpha(y)dy\\ \qquad - \displaystyle\sum_{i=1}^{d}\int_{\mathbb{R}^d} z^i(y)\partial_1 K^i(\varphi(x), \varphi(y))p_\mu(x)dy\\ \qquad - \displaystyle\sum_{i=1}^{d}\int_{\mathbb{R}^d} z^i(y)\partial_2 K^i(\varphi(x), \varphi(y))p_\mu(y)dy,\end{cases} \tag{10.47}$$

in which

$$\zeta^{1,1}(x) = \sum_{i,j=1}^{d}\int_{\mathbb{R}^d}(\alpha^i(y)z^j(x) + z^i(y)\alpha^j(x))\nabla_1 K^{ij}(\varphi(x), \varphi(y))dy$$

and

$$\begin{aligned}\zeta^{2,1}(x) = &-\sum_{i,j=1}^{d}\int_{\mathbb{R}^d} z^i(y)z^j(x)\partial_1\partial_2 K^{ij}(\varphi(x), \varphi(y))p_\mu(y)dy\\ &-\sum_{i,j=1}^{d}\int_{\mathbb{R}^d} z^i_l(y)z^j_k(x)\partial_1^2 K^{ij}(\varphi(x), \varphi(y))p_\mu(x)dy.\end{aligned}$$

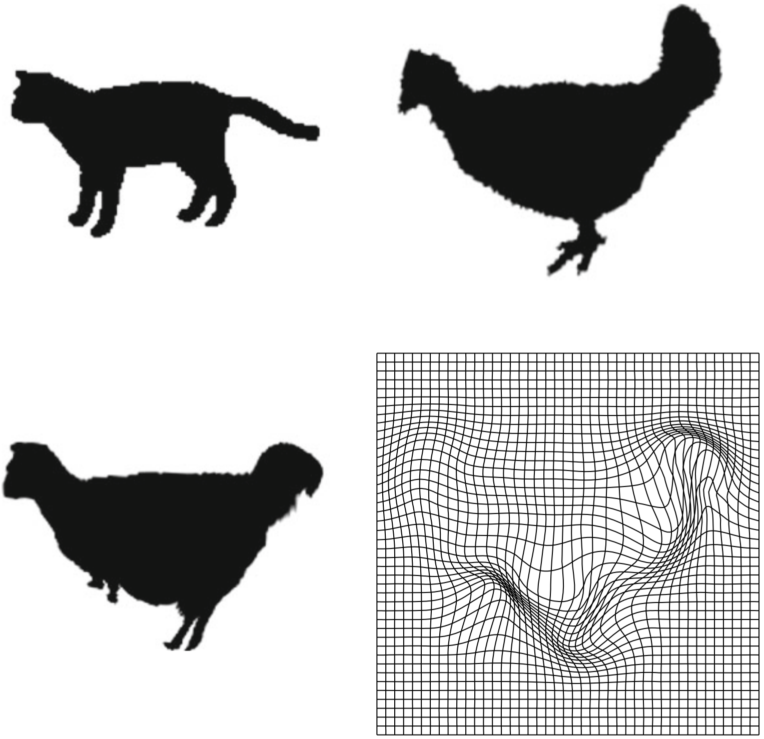

Fig. 10.2 Metric image matching. Output of Algorithm 4 when estimating a deformation of the first image to match the second one (compare to Fig. 9.2). The third image is the obtained deformation of the first one and the last provides the deformation applied to a grid

We summarize the computation of the gradient of the image-matching functional with respect to z_0 such that $\rho_0 = z_0 dx$:

Algorithm 4

1. Solve (10.46) with initial conditions $\varphi(0) = \mathrm{id}$ and $z(0) = z_0$ and compute $dU(\varphi(1)) = -\lambda(I - I' \circ \varphi(1)) \det(d\varphi(1)) d\varphi(1)^{-T} \nabla I$.
2. Solve, backwards in time, until time $t = 0$ the system (10.47) with boundary conditions $\alpha(1) = -dU(\varphi(1))$ and $p_\mu(1) = 0$.
3. Set $\nabla E(z_0) = 2z_0 - \mathbb{K}^{-1} p_\mu(0)$.

The gradient is computed with the metric $\left\langle z\,,\, z' \right\rangle = \int_{\mathbb{R}^d} z(y)^T \mathbb{K} z(y) dy$. Results obtained with this algorithm are presented in Fig. 10.2.

One can also use the fact that $z_0 = f_0 \nabla I$ for a scalar-valued f_0. Since we have

$$\left(dE(z_0) \,\middle|\, h_0 \right) = \int_\Omega (\mathbb{K} z_0 - p_\mu(0))^T h_0 dy,$$

we can write, with $\tilde{E}(f_0) = E(f_0 \nabla I)$:

$$\left(d\tilde{E}(f_0) \,\middle|\, u_0 \right) = \int_\Omega (\mathbb{K}(f_0 \nabla I) - p_\mu(0))^T \nabla I u_0 dy,$$

which leads to replacing the last step in Algorithm 4 by

$$\nabla E(f_0) = -\nabla I^T (\mathbb{K}(f_0 \nabla I) - p_\mu(0)),$$

which corresponds to using the L^2 metric in f_0 for gradient descent. However, a more natural metric, in this case, is the one induced by the kernel, i.e.,

$$\langle f\,,\, f' \rangle_I = \int_\Omega \int_\Omega \mathbb{K}(f \nabla I)(y)(f'(y) \nabla I(y) dy = \int_\Omega K_I(x, y) f(x) f'(y) dx dy$$

with $K_I(x, y) = \nabla I(x)^T K(x, y) \nabla I(y)$. With this metric, z_0 is updated with

$$\nabla E(f_0) = z_0 - \mathbb{K}_I^{-1} \nabla I^T p_\mu(0).$$

Although this metric is more satisfactory from a theoretical viewpoint, the inversion of $\mathbb{K}_I$ might be difficult, numerically.

10.6.7 Pros and Cons of the Optimization Strategies

In the previous sections we have reviewed several possible choices of control variables with respect to which the optimization of the matching energy can be performed. For all but the shooting method, this results in specific expressions of the gradient that can then be used in optimization procedures such as those discussed in Appendix D.

All these procedures have been implemented in the literature to solve a diffeomorphic-matching problem in at least one specific context, but no extensive study has ever been made to compare them. Even if the outcome of such a study is likely to be that the best method depends on the specific application, one can still provide a few general facts that can help a user decide which one to use.

When feasible (that is, when the linear system it involves at each step can be efficiently computed and solved), the shooting method is probably the most efficient. If the initialization is not too far from the solution, convergence can be achieved in a very small number of iterations. One cannot guarantee, however, that the method will converge starting from any initial point, and shooting needs to be combined with some gradient-based procedure in order to find a good starting position.

Since they optimize with respect to the same variable, the most natural procedure to combine with shooting is optimization with respect to the initial momentum. Even when shooting is not feasible (e.g., for large-scale problems), this specific choice of control variable is important, because it makes sure that the final solution satisfies the

EPDiff equation, which guarantees the consistency of the momentum representation, which will be discussed in Sect. 11.5.2. The limitation is that, with large and complex deformations, the sensitivity of the solution to small changes in the control variable can be large, which may result in an unstable optimization procedure.

The other methods, which optimize with respect to time-dependent quantities, are generally more able to compute very large deformations. Beside the obvious additional burden in computer memory that they require, one must be aware that the discrete solution can sometimes be far from satisfying the EPDiff equation unless the time discretization is fine enough (which may be impossible to achieve within a feasible implementation for large-scale problems). Therefore, these methods do not constitute the best choice if obtaining a reliable momentum representation is important. Among the three time-dependent control variables that we have studied (velocity, momentum and deformable object), one may have a slight preference for the representation using the time-dependent momenta, even if the computation it involves is slightly more complex than the others. There are at least two reasons for this. First, the momenta are generally more parsimonious in the space variables, because they incorporate normality constraints to transformations that leave the deformable objects invariant. Second, because the forward and backward equations solved at each iteration immediately provide a gradient with respect to the correct metric, so that the implementation does not have to include the solution of a possibly large-dimensional linear system which is required by other representations.

10.7 Numerical Aspects

10.7.1 Discretization

The implementation of the diffeomorphic matching algorithms that were just discussed requires a proper discretization of the different variables that are involved. The discretization in time of optimal control problems is discussed in Sect. D.4. This discussion directly applies here and we refer the reader to the relevant pages in the chapter for more details. If the deformed objects are already discrete (e.g., points sets), this suffices in order to design a numerical implementation.

When the deformed objects are continuous, some discrete approximation must obviously be made. One interesting feature of the problems that we have discussed is that they all derive from the general formulation (10.8), but can be reduced, using Sect. 10.6.2, to a situation in which the state and controls are finite dimensional after discretization. Typically, starting from (10.8), the discretization implies that only the end-point cost function is modified, replacing $U(\varphi) = F(\varphi \cdot I_0)$ by an approximation taking the form $U^{(n)}(\varphi) = F^{(n)}(\varphi, I_0^{(n)})$. For example, when matching curves, one may replace the objective function $F(\varphi \cdot I_0) = \|\mu_{\varphi \cdot I_0} - \mu_{I'}\|^2_{W^*}$ in (9.40) by the discrete approximation in (9.46), in which the curves I_0 and I' are approximated by point sets. Similar approximations can be made for the other types of cost functions discussed for curves and surfaces. In such cases, the following proposition can be applied to compare solutions of the original problem with their discrete approximations.

Proposition 10.16 *Assume that V is continuously embedded in $C_0^{p+1}(\mathbb{R}^d, \mathbb{R}^d)$. Consider a family of optimal control problems minimizing*

$$E^{(n)}(v) = \frac{1}{2}\int_0^1 \|v\|_V^2\, dt + U^{(n)}(\varphi_{01}^v), \tag{10.48}$$

with $U^{(n)}$ continuous for the (p, ∞)-compact topology. Let U be continuous with respect to the same topology and assume that, for some $p > 0$, the following uniform convergence is true: for all $A > 0$ and $\varepsilon > 0$, there exists an n_0 such that, for all $n \geq n_0$, for all $\varphi \in \mathrm{Diff}_0^{p,\infty}$ such that $\max(\|\varphi\|_{p,\infty}, \|\varphi^{-1}\|_{p,\infty}) < A$, *one has $|U^{(n)}(\varphi) - U(\varphi)| < \varepsilon$.*

Then, given a sequence $v^{(n)}$ of minimizers of (10.48), one can extract a subsequence $v^{(n_k)}$ that weakly converges to v in $\mathcal{X}_V^2$, with v minimizing

$$E(w) = \frac{1}{2}\int_0^1 \|w\|_V^2\, dt + U(\varphi_{01}^w). \tag{10.49}$$

Proof Let w be a minimizer of (10.49). Our assumptions implying that $U^{(n)}(\varphi_{01}^w)$ converges to $U(\varphi_{01}^w)$ (so that their difference is bounded), we see that $E^{(n)}(w) \leq E(w) + C$ for some constant C, so that, letting $v^{(n)}$ be a minimizer of $E^{(n)}$, we have $\|v^{(n)}\|_{\mathcal{X}_V^2}^2 \leq 2E^{(n)}(v^{(n)}) \leq 2E(w) + 2C$. From this we find that $v^{(n)}$ is a bounded sequence in $\mathcal{X}_V^2$, so that, replacing it with a subsequence if needed, we can assume that it weakly converges to some $v \in \mathcal{X}_V^2$. Applying Theorem 7.13, we find that $\varphi_{01}^{v^{(n)}}$ converges to φ^v in the (p, ∞)-compact topology. Moreover, Theorem 7.10 implies that the sequences $(\|\varphi_{01}^{v^{(n)}}\|_{p,\infty}, \|\varphi_{10}^{v^{(n)}}\|_{p,\infty})$ are bounded. Applying the uniform convergence of $U^{(n)}$ to U on bounded sets and the continuity of U, we see that $U^{(n)}(\varphi_{01}^{v^{(n)}})$ converges to $U(\varphi_{01}^v)$ as n tends to infinity. Since, in addition

$$\|v\|_{\mathcal{X}_V^2} \leq \liminf \|v^{(n)}\|_{\mathcal{X}_V^2}$$

we obtain the fact that $E(v) \leq \liminf E^{(n)}(v^{(n)})$. We also have

$$E^{(n)}(v^{(n)}) \leq E^{(n)}(w) = E(w) + U(\varphi_{01}^w) - U^{(n)}(\varphi_{01}^w) \to E(w),$$

so that $E(v) = E(w)$ and v is also a minimizer of (10.49). □

Curves and Surfaces. We can apply this theorem to curve and surface matching according to the following discussion, in which we focus on surface matching using currents, but which can, with very little modification, be applied to curves, and to measure or varifold matching terms. Let Σ and $\tilde{\Sigma}$ be regular surfaces and $S^{(n)}$, $\tilde{S}^{(n)}$ be sequences of triangulated surfaces that converge to them as defined before Theorem 4.3. Let (fixing an RKHS W with kernel ξ)

$$U(\varphi) = \|\nu_{\varphi\cdot\Sigma} - \nu_{\tilde{\Sigma}}\|^2_{W^*},$$

using the vector measures defined in Eq. (9.49), and

$$U^{(n)}(\varphi) = \|\nu_{\varphi\cdot S^{(n)}} - \nu_{\tilde{S}^{(n)}}\|^2_{W^*},$$

using the discrete version as in (9.56). Then, Theorem 4.3, slightly modified to account for double integrals, can be used to check that the assumptions of Proposition 10.16 are satisfied.

Images. The image matching problem can be discretized using finite grids, assuming that the considered images are supported by the interval $[0, 1]^d$. Consider the cost function

$$U(\varphi) = \|I \circ \varphi^{-1} - \tilde{I}\|_2^2,$$

in which we assume, to simplify, that I and $\tilde{I}$ are compactly supported (say, on $\mathcal{K} = [-M, M]^d$) and bounded. We first start with a discretization that can be applied to general L^2 functions. Let $\mathcal{G}_n = \{-M + 2^{-n+1}kM, k = 0, \ldots, 2^n\}^d$ provide a discrete grid on $\mathcal{K}$ and associate to each point $z \in \mathcal{G}_n$ its Voronoï cell, $\Gamma_n(z)$, provided by the set of points in $\mathcal{K}$ that are closer to x than to any other point in the grid (i.e., $\Gamma_n(z)$ is the intersection of $\mathcal{K}$ and the cube of size 2^{-n} centered at x). Define

$$I^{(n)}(x) = \sum_{z\in\mathcal{G}_n} \bar{I}^{(n)}(z)\mathbf{1}_{\Gamma_n(z)}(x),$$

where

$$\bar{I}^{(n)}(z) = \frac{1}{|\Gamma_n(z)|}\int_{\Gamma_n(z)} I(x)\, dx$$

is the average value of I over $\Gamma_n(z)$.

Define $\tilde{I}^{(n)}$ similarly and consider the approximation of U given by $U^{(n)}(\varphi) = \|I^{(n)} \circ \varphi^{-1} - \tilde{I}^{(n)}\|_2^2$. Then $U^{(n)}$ and U satisfy the hypotheses of Proposition 10.16.

Indeed, assume that $\max(\|\varphi\|_{1,\infty}, \|\varphi^{-1}\|_{1,\infty}) < A$. We have

$$\begin{aligned}|U^{(n)}(\varphi) - U(\varphi)| &\leq 2\|I \circ \varphi^{-1} - I^{(n)} \circ \varphi^{-1}\|_2^2 + 2\|\tilde{I} - \tilde{I}^{(n)}\|_2^2\\ &\leq 2C(A)\|I - I^{(n)}\|_2^2 + 2\|\tilde{I} - \tilde{I}^{(n)}\|_2^2,\end{aligned}$$

where the second inequality is obtained after a change of variable in the first L^2 norm and $C(A)$ is an upper bound for the Jacobian determinant of φ depending only on A. As a consequence, Proposition 10.12 will be true as soon as one shows that $I^{(n)}$ and $\tilde{I}^{(n)}$ converge in L^2 to I and $\tilde{I}$ respectively (and will also be true for any sequence of approximations of I and $\tilde{I}$ that satisfies this property). The L^2 convergence is true in our case because $I^{(n)}$ is the orthogonal projection of I on the space W_n of L^2 functions that are constant on each set $\underline{\Gamma_n(z),\ z \in \mathcal{G}}_n$. This implies that $I^{(n)}$ converges in L^2 to the projection of I on $W_\infty = \overline{\bigcup_{n\geq 1} W_n}$ (see Proposition A.11), but one has $W_\infty = L^2$, because any function J orthogonal to this space would have its integral vanish on any dyadic cube, which is only possible for $J = 0$.

Note that, with this approximation, one can write

$$\begin{aligned}\|I^{(n)} \circ \varphi^{-1} - \tilde I\|_2^2 = \sum_{z\in\mathcal{G}_n} I(z)^2 |\varphi(\Gamma_n(z))| + \sum_{z\in\mathcal{G}_n} \tilde I(z)^2 |\Gamma_n(z)| \\ - 2 \sum_{z,z'\in\mathcal{G}_n} I(z)\tilde I(z')|\varphi(\Gamma_n(z)) \cap \Gamma_n(z')|,\end{aligned}$$

where $|A|$ denotes the volume of $A \subset \mathbb{R}^d$. To make this expression computable, one needs to approximate the sets $\varphi(\Gamma_n(z))$, where the simplest approximation is to take the polyhedron formed by the image of the vertices of $\Gamma_n(z)$ by φ (which will retain the same topology as the original cube is n is large enough). The verification that this approximation is valid (in the sense of Proposition 10.16) is left to the reader.

However, even with this approximation, the numerical problem is still highly computational, since it becomes a point set problem over $\mathcal{G}_n$, which is typically a huge set. Most current implementations use a simpler scheme, in which $I^{(n)}$ is interpolated between the values $(I(z), z \in \mathcal{G}_n)$, who are therefore assumed to be well defined, and the cost function is simply approximated by

$$U^{(n)}(\varphi) = \sum_{z\in\mathcal{G}_n} (I(\varphi^{-1}(z)) - \tilde I(z))^2 |\Gamma_n(z)|.$$

Here again, we leave to the reader to check that this provides a valid approximation in the sense of Proposition 10.16 as soon as, say, I and $\tilde I$ are continuous and one uses a linear interpolation scheme, as described below.

Using this approximation (for a fixed n that we will remove from the notation), we now work the implementation in more detail, starting with the computation of the gradient in (10.45). Assume that time is discretized at $t_k = kh$ for $h = 1/Q$ and that $v_k(\cdot) = v(t_k, \cdot)$ is discretized over a regular grid $\mathcal{G}$.

It will be convenient to introduce the momentum and express v_k in the form

$$v_k(y) = \sum_{z\in\mathcal{G}} K(y, z)\rho_k(z). \tag{10.50}$$

We can consider $(\rho_k(z), z \in \mathcal{G})$ as new control variables, noting that (10.45) directly provides the gradient of the energy in V^*, namely

$$(\nabla^{V^*} E)(t, y) = 2\rho(t) - 2 \det(d\varphi_{t1}^v)(I \circ \varphi_{t0}^v - \tilde I \circ \varphi_{t1}^v)\nabla(I \circ \varphi_{t0}^v)dx.$$

From this expression, we see that we can interpret the family $(\rho_k(z), z \in \mathcal{G})$ as discretizing a measure, namely

$$\rho_k = \sum_{z\in\mathcal{G}} \rho_k(z)\delta_z.$$

Given this, the gradient in V^* can be discretized as

$$\xi_k(z) = 2\rho_k(z) - 2\det(d\varphi^v_{t_k 1}(z))(I \circ \varphi^v_{t_k 0}(z) - \tilde{I} \circ \varphi^v_{t_k 1}(z))\nabla(I \circ \varphi^v_{t_k 0}(z))\delta_z,$$

which can be used to update $\rho_k(z)$.

The last requirement in order to obtain a fully discrete procedure is to select interpolation schemes for the computation of the diffeomorphisms φ^v and for the compositions of I and I' with them. Interpolation algorithms (linear, or cubic, for example) are standard procedures that are included in many software packages [234]. In mathematical representation, they are linear operators that take a discrete signal f on a grid $\mathcal{G}$ (i.e., $f \in \mathbb{R}^{\mathcal{G}}$) and return a function, that we will denote by $\mathcal{R}f$, defined everywhere. By linearity, we must have

$$(\mathcal{R}f)(z) = \sum_{z\in\mathcal{G}} r_z(x) f(z)$$

for some "interpolants" $r_z(\cdot)$, $z \in \mathcal{G}$. In the approximation of the data attachment term, one can then replace I by $\mathcal{R}(I_{|\mathcal{G}})$, the interpolation of the restriction of I to $\mathcal{G}$.

Linear interpolation, for example, corresponds, in one dimension, to $r_z(x) = 1 - 2^n|z - x|$ if $|z - x| < 2^{-n}$ and 0 otherwise. In dimension d, one takes

$$r_z(x) = \prod_{i=1}^{d}(1 - 2^n|z_i - x_i|)$$

if $\max_i(|z_i - x_i|) < 2^{-n}$ and 0 otherwise (where $z = (z_1, \dots, z_d)$ and $x = (x_1, \dots, x_d)$).

Given an interpolation operator $\mathcal{R}$, one can replace, say, $I \circ \varphi_{t_k 0}(z)$ in the expression of the gradient by

$$(\mathcal{R}I)(\varphi_{t_k 0}(z)) = \sum_{z'\in\mathcal{G}} r_{z'}(\varphi_{t_k 0}(z)) I(z').$$

For computational purposes, it is also convenient to replace the definition of v_k in (10.50) by an interpolated form

$$v_k(x) = \sum_{z\in\mathcal{G}} r_z(x) \sum_{i\in\mathcal{G}} K(z, z')\rho_k(z') \qquad (10.51)$$

because the inner sum can be computed very efficiently using Fourier transforms (see the next section).

To complete the discretization, introduce

$$\psi_{lk} = (\mathrm{id} - hv_l) \circ \cdots \circ (\mathrm{id} - hv_{k-1}),$$

where an empty product of compositions is equal to the identity, so that ψ_{lk} is an approximation of $\varphi_{t_k t_l}$. Define the cost function, which is explicitly computable as a function of $\rho_0, \ldots, \rho_{Q-1}$:

$$E(\rho) = \sum_{k=0}^{Q-1} \sum_{z,z' \in \mathcal{G}} K(z,z')\rho_k(z)^T \rho_k(z') + \sum_{z \in \mathcal{G}} ((\mathcal{R}I)(\psi_{0Q}(z)) - \tilde{I}(z))^2.$$

If we make a variation $\rho \mapsto \rho + \varepsilon\delta\rho$, then $v \mapsto v + \varepsilon\delta v$ with (using the interpolated expression of v)

$$\delta v_k(y) = \sum_{z \in \mathcal{G}} r_z(y) \sum_{z' \in \mathcal{G}} K(z,z')\delta\rho_k(z')$$

and letting $\delta\psi_{lk} = \partial_\varepsilon \psi_{lk}$, we have, by direct computation

$$\delta\psi_{lk} = -h \sum_{q=l}^{k-1} d\psi_{lq} \circ \psi_{qk}\, \delta v_q \circ \psi_{q+1k}.$$

Using this, we can compute the variation of the E, yielding

$$\begin{aligned}(\partial_\varepsilon E \mid \delta\rho) = 2 &\sum_{k=0}^{Q-1} \sum_{z,z',\in \mathcal{G}} K(z,z')\, \rho_k(z')^T \delta\rho_k(z) \\ &- 2h \sum_{k=0}^{Q-1} \sum_{z,z',y \in \mathcal{G}} K(z,z')\, r_z(\psi_{k+1\,Q}(y))\, ((\mathcal{R}I)(\psi_{0Q}(y)) - \tilde{I}(y)) \\ &\qquad \nabla(\mathcal{R}I)(\psi_{0Q}(y))^T (d\psi_{0k} \circ \psi_{k\,Q}(y)\, \delta\rho_k(z'))\end{aligned}$$

This provides the expression of the gradient of the discretized E in V^*, namely

$$\begin{aligned}&(\nabla^{V^*} E(\rho))_k(z) = 2\rho_k(z) \\ &- 2h \sum_{z' \in \mathcal{G}} r_z(\psi_{k+1\,Q}(z'))((\mathcal{R}I)(\psi_{0Q}(z')) - \tilde{I}(z'))\nabla(\mathcal{R}I \circ \psi_{0k})(\psi_{kQ}(z')).\end{aligned}$$

10.7.2 Kernel-Related Numerics

Most of the previously discussed methods included repeated computations of linear combination of the kernel. A basic such step is to compute, given points $y_1, \ldots, y_M$, $x_1, \ldots, x_N$ and vectors (or scalars) $\alpha_1, \ldots, \alpha_N$, the sums

$$\sum_{k=1}^{N} K(y_j, x_k)\alpha_k, \quad j = 1, \ldots, M.$$

Such sums are involved when deriving velocities from momenta, for example, or when evaluating dual RKHS norms in curve or surface matching.

Computing these sums explicitly requires NM evaluations of the kernel (and this probably several times per iteration of an optimization algorithm). When N or M are reasonably small (say, less than 1,000), such a direct evaluation is not a problem. But for large-scale methods, such as triangulated surface matching, where the surface may have tens of thousands of nodes, or image matching, where a three-dimensional grid typically has millions of nodes, this becomes unfeasible (the feasibility limit has however been pushed further by recent efficient implementations on GPUs [59, 157, 247]).

If $x = y$ is supported by a regular grid $\mathcal{G}$, and K is translation invariant, i.e., $K(x, y) = \Gamma(x - y)$, then, letting $x_k = hk$ where k is a multi-index ($k = (k_1, \ldots, k_d)$) and h the discretization step, we see that

$$\sum_{k \in \mathcal{G}} \Gamma(h(k - l))\alpha_l$$

is a convolution that can be implemented with $O(N \log N)$ operations, using fast Fourier transforms (with $N = |\mathcal{G}|$). The same conclusion holds if K takes the form $K(x, y) = A(x)^T \Gamma(x - y) A(y)$ for some matrix A (which can be used to censor the kernel at the boundary of a domain), since the resulting operation is

$$A(x_k)^T \left(\sum_{k \in \mathcal{G}} \Gamma(h(k - l))(A(x_l)\alpha_l) \right),$$

which can still be implemented in $O(N \log N)$ operations.

The situation is less favorable when x and y are not regularly spaced. In such cases, feasibility must come with some approximation.

Still assuming a translation-invariant kernel $K(x, y) = \Gamma(x - y)$, we can associate to a grid $\mathcal{G}$ in $\mathbb{R}^d$ the interpolated kernel

$$K_{\mathcal{G}}(x, y) = \sum_{j, j' \in \mathcal{G}} r_z(x) \Gamma(h(z - z')) r_{z'}(y),$$

where the r_z's are interpolants adapted to the grid. This approximation provides a non-negative kernel, with null space equal to the space of functions with vanishing interpolation on $\mathcal{G}$. With such a kernel, we have

$$\sum_{k=1}^{N} K(y_j, x_k)\alpha_k = \sum_{z\in\mathcal{G}} r_z(y_j) \sum_{z'\in\mathcal{G}} \Gamma(h(z-z')) \sum_{k=1}^{N} r_{z'}(x_k)\alpha_k.$$

The computation of this expression therefore requires using the following sequence of operations:

1. Compute, for all $z' \in \mathcal{G}$, the quantity

$$a_{z'} = \sum_{k=1}^{N} r_{z'}(x_k)\alpha_k.$$

   Because, for each x_k, only a fixed number of $r_{z'}(x_k)$ are non-vanishing, this requires an $O(N)$ number of operations.

2. Compute, for all $z \in \mathcal{G}$,

$$b_z = \sum_{z'\in\mathcal{G}} \Gamma(h(z-z'))a_{z'},$$

   which is a convolution requiring $O(|\mathcal{G}| \log |\mathcal{G}|)$ operations.

3. Compute, for all $j = 1, \ldots, M$, the interpolation

$$\sum_{z\in\mathcal{G}} r_z(y_j)b_z,$$

   which requires $O(M)$ operations.

So the resulting cost is $O(M + N + |\mathcal{G}| \log |\mathcal{G}|)$, which must be compared to the original $O(MN)$, the comparison being favorable essentially when MN is larger than the number of nodes in the grid, $|\mathcal{G}|$. This formulation (which has been proposed in [156]) has the advantage that the resulting algorithm is quite simple, and that the resulting $K_{\mathcal{G}}$ remains a non-negative kernel, which is important.

Another class of methods, called "fast multipole", computes sums such as

$$\sum_{k=1}^{N} K(y, x_k)\alpha_k$$

by taking advantage of the fact that $K(y, x)$ varies slowly as x varies in a region which is far away from y. By grouping the x_k's in clusters, assigning centers to these clusters and approximating the kernel using asymptotic expansions valid at a large enough distance from the clusters, fast multipole methods can organize the computation of the sums with a resulting cost of order $M + N$ when M sums over

N terms are computed. Even if it is smaller than a constant times $(M + N)$, the total number of operations increases (via the size of the constant) with the required accuracy. The interested reader may refer to [30, 140] for more details.

Another important operation involving the kernel is the inversion of the system of equations (say, with a scalar kernel)

$$\sum_{k=1}^{N} K(x_k, x_l)\alpha_l = u_k, \quad k = 1, \ldots, N. \tag{10.52}$$

This is the spline interpolation problem, but it is also part of several of the algorithms that we have discussed, including for example the projection steps that have been introduced to obtain a gradient in the correct metric.

Such a problem is governed by an uncertainty principle [258] between accuracy of the approximation, which is given by the distance between a smooth function $x \mapsto u(x)$ and its interpolation

$$x \mapsto \sum_{k=1}^{N} K(x, x_l)\alpha_l,$$

where $\alpha_1, \ldots, \alpha_N$ are given by (10.52) with $u_k = u(x_k)$, and the stability of the system (10.52) measured by the condition number (the ratio of the largest to the smallest eigenvalue) of the matrix $S(x) = (K(x_i, x_j), i, j = 1, \ldots, N)$, evaluated as a function of the smallest distance between two distinct x_k's ($S(x)$ is singular if two x_k's coincide).

When $K(x, y) = \Gamma(x - y)$, the trade-off is measured by how fast $\xi \mapsto \hat{\Gamma}(\xi)$ (the Fourier transform of Γ) decreases at infinity. One extreme is given by the Gaussian kernel, for which $\hat{\Gamma}$ decreases like $e^{-c|\xi|^2}$, which is highly accurate and highly unstable. On the other side of the range are Laplacian kernels, which decrease polynomially in the Fourier domain. In this dilemma, one possible rule is to prefer accuracy for small values of N, therefore using a kernel like the Gaussian, and go for stability for large-scale problems (using a Laplacian kernel with high enough degree).

For the numerical inversion of system (10.52), iterative methods, such as conjugate gradient, should be used (especially for large N). Methods using preconditioned conjugate gradient have been introduced, for example, in [105, 141] and the interested reader may refer to these references for more details.