POINTWISE BESOV SPACE SMOOTHING OF IMAGES

GREGERY T. BUZZARD, ANTONIN CHAMBOLLE, JONATHAN D. COHEN, STACEY E. LEVINE, AND BRADLEY J. LUCIER

1. Pointwise norms for Besov spaces. We first define the seminorm in the Besov space $B^1_\infty(L_1(I))$, where $I = [0,1]^2$ is the domain of our images. We do this in a way that allows us to define an "equivalent" discrete Besov norm.

The rth order difference at the point $x \in I$ in the direction $h \in \mathbb{R}^2$ of a function $f\colon I \to \mathbb{R}$ is defined by

$$\Delta^0 f(x,h) = f(x),\ \ \Delta^{r+1} f(x,h) = \Delta^r f(x+h,h) - \Delta^r f(x,h). \tag{1.1}$$

For $0 < p \leq \infty$, the rth order modulus of smoothness in $L_p(I)$ is defined by

$$\omega_r(f,t)_p = \sup_{|h|\leq t} \|\Delta^r f(\cdot,h)\|_{L_p(I_{rh})}, \tag{1.2}$$

where $I_{rh} = \{x \in I \mid x + rh \in I\}$ is the set of all $x \in I$ for which $\Delta^r f(x,h)$ is defined.

With these definitions, one of the equivalent seminorms for $B^1_\infty(L_1(I))$ is

$$|f|_{B^1_\infty(L_1(I))} = \sup_{t>0} \frac{1}{t}\omega_2(f,t)_1. \tag{1.3}$$

Combining these definitions, we have

$$|f|_{B^1_\infty(L_1(I))} = \sup_{|h|>0} \frac{1}{|h|}\|\Delta^2 f(\cdot,h)\|_{L_1(I_{2h})} \tag{1.4}$$

and

$$\|f\|_{B^1_\infty(L_1(I))} = \|f\|_{L_1(I)} + |f|_{B^1_\infty(L_1(I))}.$$

We get an equivalent norm if we replace $\Delta^2 f(\cdot,h)$ by $\Delta^r f(\cdot,h)$ for any $r > 1$; see [**?**].

Note that we are not dividing by $|h|^2$; if we were, the functions in the space would consist of functions whose second derivatives are measures. More to the point, because

$$\omega_2(f,nt)_1 \leq n^2 \omega_2(f,t)_1,$$

(see Chapter 2 of [**?**]) we have

$$\frac{1}{(nt)^2}\omega_2(f,nt)_1 \leq \frac{1}{t^2}\omega_2(f,t)_1,$$

so

$$\sup_{t>0} \frac{1}{t^2}\omega_2(f,t)_1 = \lim_{t\to 0} \frac{1}{t^2}\omega_2(f,t)_1.$$

The supremum in (**??**) is generally not taken in the limit as $t \to 0$, however; for example, if f has two continuous derivatives in $L_1(I)$, then as $\delta \to 0$, $|\delta h|^{-2}\Delta^2 f(x,\delta h) \to D^2_h f(x)$, the second directional derivative of f in the direction h at x, and

$$\frac{1}{t}\omega_2(f,t)_1 = t\Big(\frac{1}{t^2}\omega_2(f,t)_1\Big) \to 0$$

as $t \to 0$, as the quantity in parentheses is finite. So computations with the $B^1_\infty(L_1(I))$ seminorm must take into account all the values of $|h| > 0$, not just the limit as $|h| \to 0$. On the other hand, we have that $\omega_r(f,t)_p \le 2^r \|f\|_{L_p(I)}$ for $1 \le p \le \infty$, so $t^{-1}\omega_2(f,t)_1 \to 0$ as $t \to \infty$. So the supremum in (**??**) is found on some finite interval $[0, t^*]$.

Note also that $\mathrm{BV}(I)$, the space of functions of bounded variation on I is contained in $B^1_\infty(L_1(I))$, because an equivalent seminorm for $\mathrm{BV}(I)$ is

$$|f|_{\mathrm{BV}(I)} = \sup_{t>0} \frac{1}{t}\omega_1(f,t)_1 = \lim_{t\to 0} \frac{1}{t}\omega_1(f,t)_1,$$

and $\omega_2(f,t)_1 \le 2\omega_1(f,t)_1$ by (**??**)—functions in $B^1_\infty(L_1(I))$ can be "more singular" than functions in $\mathrm{BV}(I)$.

The supremum in (**??**) is taken over all directions h, and we cannot compute over all directions. For effective computations, we note as a consequence of a result of Ditzian and Ivanov [**?**] that an equivalent seminorm $|f|_{B^1_\infty(L_1(I))}$ arises if we replace the set of all directions h in (**??**) by multiples of any three of the directions $(1,0)$, $(0,1)$, $(1,1)$, and $(1,-1)$. For the sake of symmetry, we use all four directions.

In our computations we work (conceptually) on the torus $\Omega = [0,2] \times [0,2]$ with periodic boundary conditions; this simulates Neumann boundary conditions on I. The definitions of forward differences, moduli of smoothness, and Besov spaces are the same on Ω as on I (only now we can calculate in $L_p(\Omega)$ instead of $L_p(\Omega_{rh})$). Given a function f defined on I, we extend it to all of Ω by setting

$$\begin{aligned} f(x_1,x_2) &= f(2-x_1,x_2), \quad 0 \le x_2 < 1,\ 1 \le x_1 \le 2, \text{ and then} \\ f(x_1,x_2) &= f(x_1,2-x_2), \quad 1 \le x_2 < 2,\ 0 \le x_1 \le 2. \end{aligned}$$

If a function f is in the Besov space $B^\alpha_q(L_p(I))$ with $0 < \alpha < 1$, $0 < p \le \infty$, and $0 < q \le \infty$, whose seminorm is defined by

$$|f|_{B^\alpha_q(L_p(I))} = \left(\int_0^\infty [t^{-\alpha}\omega_1(f,t)_p]^q \frac{dt}{t} \right)^{1/q}$$

(with the usual modification when $q = \infty$), then, by the Whitney extension theorem (see [**?**]), the extended f is in the space $B^\alpha_q(L_p(\Omega))$, i.e., it has the same smoothness on Ω as the original function had on I.

The reader may be more familiar with the special case $q = \infty$, in which case the Besov space $B^\alpha_\infty(L_p(\Omega))$, $0 < \alpha < 1$, is simply the Lipschitz space $\mathrm{Lip}(\alpha, L_p(I))$, with seminorm

$$|f|_{\mathrm{Lip}(\alpha,L_p(I))} = \sup_{t>0} t^{-\alpha}\omega_1(f,t)_p.$$

We consider f to represent the intensity field of an image, and typically images are discontinuous across one-dimensional curves, in which case $\alpha < \frac{1}{p} \le 1$ when $p \ge 1$. Thus, this extension preserves the smoothness of the functions to which we intend to apply it. Obviously, for any values of α, p, and q, if $f \in B^\alpha_q(L_p(\Omega))$ then the restriction of f to I is in $B^\alpha_q(L_p(I))$.

2. Some variational image smoothing methods. We consider a number of variational approaches to noise removal and image smoothing in general.

The first model uses the $L_2(\Omega)$ norm as a measure of the difference between the original and smooth function: Given a function f defined on I and a parameter

$\lambda > 0$, extend f to $L_2(\Omega)$ and find the minimizer over all g with $f - g \in L_2(\Omega)$ and $g \in B^1_\infty(L_1(\Omega))$ of

$$E(g) = \frac{1}{2\lambda}\|f-g\|^2_{L_2(\Omega)} + |g|_{B^1_\infty(L_1(\Omega))}. \tag{2.1}$$

In image processing applications we would take $g|_I$ as our smoothed image.

One could replace the $L_2(\Omega)$ norm in the previous expression with an $L_1(\Omega)$ norm, and so come up with the second model: Given a function f defined on I and a parameter $\lambda > 0$, extend f to $L_1(\Omega)$ and find the minimizer over all g with $f - g \in L_1(\Omega)$ and $g \in B^1_\infty(L_1(\Omega))$ of

$$E(g) = \frac{1}{\lambda}\|f-g\|_{L_1(\Omega)} + |g|_{B^1_\infty(L_1(\Omega))}. \tag{2.2}$$

3. Discrete norms, spaces, and variational problems. For computations, we want to extend the idea of spaces, seminorms, extensions, etc., from the continuous domain to the discrete domain. So for discrete functions f, g defined for $I^h = \{i = (i_1, i_2) \mid 0 \le i_1, i_2 < N\}$, $h = 1/N$, we define a discrete inner product,

$$\langle f, g\rangle = \sum_{i \in I^h} f_i g_i \, h^2,$$

and the discrete $L_p(I^h)$ norm for $1 \le p < \infty$

$$\|g\|^p_{L_p(I^h)} = \sum_{i\in I^h} |g_i|^p \, h^2.$$

We extend discrete functions f defined on I^h to functions defined on

$$\Omega^h = \{i = (i_1, i_2) \mid 0 \le i_1, i_2 < 2N\}$$

by

$$f_{(i_1,i_2)} = f_{(2N-1-i_1,i_2)} \text{ for } N \le i_1 < 2N \text{ and } 0 \le i_2 < N$$

and then

$$f_{(i_1,i_2)} = f_{(i_1,2N-1-i_2)} \text{ for } 0 \le i_1 < 2N \text{ and } N \le i_2 < 2N.$$

Functions f are then extended periodically with period $2N$ in each direction.

We need a discrete $B^1_\infty(L_1(I))$ seminorm. We start by defining a discrete "second gradient"

$$\nabla^2 g = \{\nabla^2_j g \mid j \in \mathbb{N}\} = \{\nabla^2_j g_i \in \mathbb{R}^4 \mid i \in I^h,\ j \in \mathbb{N}\}$$

centered at location $i \in I^h$ with differences on a scale of $j \in \mathbb{N} = \{1, 2, \dots\}$ pixels as

$$\nabla^2_j g_i = \begin{pmatrix} \dfrac{g_{i+(j,0)} - 2g_i + g_{i-(j,0)}}{jh} \\ \dfrac{g_{i+(0,j)} - 2g_i + g_{i-(0,j)}}{jh} \\ \dfrac{g_{i+(j,j)} - 2g_i + g_{i-(j,j)}}{\sqrt{2}jh} \\ \dfrac{g_{i+(j,-j)} - 2g_i + g_{i-(j,-j)}}{\sqrt{2}jh} \end{pmatrix}. \tag{3.1}$$

If an index in this calculation falls outside I^h, we use the corresponding value of the extended periodic function.

Our discrete $B^1_\infty(L_1(I^h))$ seminorm is then

$$|g|_{B^1_\infty(L_1(I^h))} = \sup_{j>0} \sum_{i\in I^h} |\nabla_j^2 g_i|\, h^2 = \|\{\||\nabla_j^2 g_i|_{\mathbb{R}^4}\|_{L_1(I^h)}\}\|_{\ell_\infty(\mathbb{N})}.$$

Because $\nabla_j^2 g_i \to 0$ as $j\to\infty$, the sequence $\big\{\||\nabla_j^2 g_i|_{\mathbb{R}^4}\|_{L_1(I^h)}\big\}$ lies in $c_0(\mathbb{N})$, the subspace of $\ell_\infty(\mathbb{N})$ of sequences that tend to 0 as $j\to\infty$, and the dual of $c_0(\mathbb{N})$ is $\ell_1(\mathbb{N})$. Thus, if we introduce sequences of vector fields on I^h

$$p = \{p^j \mid j\in\mathbb{N}\} = \{p_i^j \in \mathbb{R}^4 \mid j\in\mathbb{N},\ i\in I^h\},$$

with the inner product

$$\langle p, q\rangle = \sum_{j\ge 1}\sum_{i\in I^h} p_i^j\cdot q_i^j\, h^2,$$

we can consider the operator adjoint to $\nabla^2 g$ in this inner product, which we denote by $\nabla^2\cdot p$. If we denote the components of the vector p_i^j by $p_i^j(1)$, etc., then

$$\begin{aligned}\nabla_j^2\cdot p_i^j = {}& \frac{p^j_{i+(j,0)}(1) - 2p_i^j(1) + p^j_{i-(j,0)}(1)}{jh} + \frac{p^j_{i+(0,j)}(2) - 2p_i^j(2) + p^j_{i-(0,j)}(2)}{jh}\\ &+ \frac{p^j_{i+(j,j)}(3) - 2p_i^j(3) + p^j_{i-(j,j)}(3)}{\sqrt{2}jh} + \frac{p^j_{i+(j,-j)}(4) - 2p_i^j(4) + p^j_{i-(j,-j)}(4)}{\sqrt{2}jh}\end{aligned}$$

and

$$\big(\nabla^2\cdot p\big)_i = \sum_j \nabla_j^2\cdot p_i^j.$$

Then we can write

$$|g|_{B^1_\infty(L_1(I^h))} = \sup_{\|\{\||p_i^j|_{\mathbb{R}^4}\|_{L_\infty(I^h)}\}\|_{\ell_1(\mathbb{N})}\le 1} \big\langle \nabla^2 g, p\big\rangle = \sup_{\|\{\||p_i^j|_{\mathbb{R}^4}\|_{L_\infty(I^h)}\}\|_{\ell_1(\mathbb{N})}\le 1} \big\langle g, \nabla^2\cdot p\big\rangle.$$

Our discrete version of (**??**) is then: Given a discrete function f and a positive parameter λ, find the minimizer over all g of the functional

$$E^h(g) = \frac{1}{2\lambda}\|f-g\|^2_{L_2(I^h)} + |g|_{B^1_\infty(L_1(I^h))}. \tag{3.2}$$

Similarly, our discrete version of (**??**) is: Given a discrete function f and a positive parameter λ, find the minimizer over all g of the functional

$$E^h(g) = \frac{1}{\lambda}\|f-g\|_{L_1(I^h)} + |g|_{B^1_\infty(L_1(I^h))}. \tag{3.3}$$

4. Discrete algorithms. We can formulate our discrete variational problems (**??**) and (**??**) as saddle-point problems which can be readily solved by algorithms from Chambolle and Pock in [**?**]. In this setting, we consider two Hilbert spaces X

and Y and the bounded linear operator $K\colon X \to Y$ with the usual norm $\|K\| = \sup_{\|g\|\leq 1} \|Kg\|$. We consider the general saddle-point problem

$$\min_{g\in X} \max_{p\in Y} (\langle Kg, p\rangle + G(g) - F^*(p)), \tag{4.1}$$

where G and F^* are proper, convex, lower-semicontinuous (l.s.c.) functions from X and Y (respectively) to $[0, +\infty]$; F^* is itself the convex conjugate of a convex l.s.c. function F. This saddle-point problem is a primal-dual formulation of the nonlinear primal problem

$$\min_{g\in X} (F(Kg) + G(p))$$

and of the corresponding dual problem

$$\max_{p\in Y} -(G^*(-K^*p) + F^*(p)),$$

where $K^*\colon Y \to X$ is the adjoint of K. An overview of these concepts from convex analysis can be found in [**?**]. We use the notation ∂F for the (multi-valued) subgradient of a l.s.c. function F

A number of algorithms are given in [**?**] to compute approximate solutions to (**??**); whether a particular algorithm is appropriate depends on properties of F^* and G. That paper also includes comparisons between the algorithms introduced there and competing algorithms. Our focus here is not on that comparison, but just to indicate that relatively efficient algorithms exist to solve the discrete minimization problems in the following sections.

Algorithm 1 in [**?**] is appropriate for completely general F and G; a special case can be stated as follows.

1. Let $L = \|K\|$. Choose positive τ and σ with $\tau\sigma L^2 < 1$, $(g^0, p^0) \in X \times Y$ and set $\bar g^0 = g^0$.
2. For $n \geq 0$, update g^n, p^n, and $\bar g^n$ as follows:

$$\begin{cases} p^{n+1} = (I + \sigma\partial F^*)^{-1}(p^n + \sigma K \bar g^n), \\ g^{n+1} = (I + \tau\partial G)^{-1}(g^n - \tau K^* p^{n+1}), \\ \bar g^{n+1} = 2g^{n+1} - g^n. \end{cases}$$

Then there exists a saddle point (g^*, p^*) such that $g^n \to g^*$ and $p^n \to p^*$. This algorithm exhibits $O(1/N)$ convergence after N steps.

Algorithm 2 in [**?**] is appropriate when G (or F^*) is uniformly convex, which means that there exists $\gamma > 0$ such that for any g in the domain of ∂G, $w \in \partial G(g)$, and $g' \in X$,

$$G(g') \geq G(g) + \langle w, g' - g\rangle + \frac{\gamma}{2}\|g - g'\|^2.$$

Algorithm 2 can then be written as follows.

1. Let $L = \|K\|$. Choose positive τ_0 and σ_0 with $\tau_0\sigma_0 L^2 \leq 1$, $(g^0, p^0) \in X \times Y$ and set $\bar g^0 = g^0$.
2. For $n \geq 0$, update g^n, p^n, $\bar g^n$ and σ_n, τ_n, θ_n as follows:

$$\begin{cases} p^{n+1} = (I + \sigma_n\partial F^*)^{-1}(p^n + \sigma_n K \bar g^n), \\ g^{n+1} = (I + \tau_n\partial G)^{-1}(g^n - \tau_n K^* p^{n+1}), \\ \theta_n = 1/\sqrt{1 + 2\gamma\tau_n},\ \tau_{n+1} = \theta_n\tau_n,\ \sigma_{n+1} = \sigma_n/\theta_n, \\ \bar g^{n+1} = g^{n+1} + \theta_n(g^{n+1} - g^n). \end{cases}$$

Then there exists a unique saddle point (g^*, p^*) such that $g^n \to g^*$ and $p^n \to p^*$. This algorithm exhibits $O(1/N^2)$ convergence after N steps.

These algorithms can directly be applied to solve problems (**??**) and (**??**). For example, we can rewrite (**??**) as

$$\begin{aligned}
\inf_g E^h(g) &= \inf_g \Big(|g|_{B^1_\infty(L_1(I^h))} + \frac{1}{2\lambda}\|f-g\|^2_{L_2(I^h)} \Big) \\
&= \inf_g \Big(\|\{\| |\nabla^2_j g_i|_{\mathbb{R}^4} \|_{L_1(I^h)}\}\|_{\ell_\infty(\mathbb{N})} + \frac{1}{2\lambda}\|f-g\|^2_{L_2(I^h)} \Big) \\
&= \inf_g \sup_{\|\{\| |p^j_i|_{\mathbb{R}^4} \|_{L_\infty(I^h)}\}\|_{\ell_1(\mathbb{N})} \le 1} \Big(\big\langle \nabla^2 g, p \big\rangle + \frac{1}{2\lambda}\|f-g\|^2_{L_2(I^h)} \Big) \\
&= \inf_g \sup_p \Big(\big\langle \nabla^2 g, p \big\rangle + \frac{1}{2\lambda}\|f-g\|^2_{L_2(I^h)} - \delta(p) \Big),
\end{aligned}$$

where

$$\delta(p) = \begin{cases} 0, & \text{if } \|\{\| |p^j_i|_{\mathbb{R}^4} \|_{L_\infty(I^h)}\}\|_{\ell_1(\mathbb{N})} \le 1, \\ \infty, & \text{otherwise.} \end{cases}$$

This matches formula (**??**) with

$$Kg = \nabla^2 g,\ K^*p = \nabla^2 \cdot p,\ G(g) = \frac{1}{2\lambda}\|f-g\|^2_{L_2(I^h)},\ \text{and } F^*(p) = \delta(p).$$

We know that G is uniformly convex with $\gamma = \lambda^{-1}$, so Algorithm 2 applies. In fact, we have that the iterates g^n and K^*p^n in Algorithm 2 converge to a pair u and $\nabla^2 \cdot p$ that satisfy

$$u = f - \lambda \, \nabla^2 \cdot p.$$

(While K^*p^n converges, p^n may not itself converge, as Y, which contains the range of K, is not compact.)

For (**??**), we can rewrite

$$\begin{aligned}
\inf_g E^h(g) &= \inf_g \Big(|g|_{B^1_\infty(L_1(I^h))} + \frac{1}{\lambda}\|f-g\|_{L_1(I^h)} \Big) \\
&= \inf_g \Big(\|\{\| |\nabla^2_j g_i|_{\mathbb{R}^4} \|_{L_1(I^h)}\}\|_{\ell_\infty(\mathbb{N})} + \frac{1}{\lambda}\|f-g\|_{L_1(I^h)} \Big) \\
&= \inf_g \sup_{\|\{\| |p^j_i|_{\mathbb{R}^4} \|_{L_\infty(I^h)}\}\|_{\ell_1(\mathbb{N})} \le 1} \Big(\big\langle \nabla^2 g, p \big\rangle + \frac{1}{\lambda}\|f-g\|_{L_1(I^h)} \Big) \\
&= \inf_g \sup_p \Big(\big\langle \nabla^2 g, p \big\rangle + \frac{1}{\lambda}\|f-g\|_{L_1(I^h)} - \delta(p) \Big).
\end{aligned}$$

This matches formula (**??**) with

$$Kg = \nabla^2 g,\ K^*p = \nabla^2 \cdot p,\ G(g) = \frac{1}{\lambda}\|f-g\|_{L_1(I^h)},\ \text{and } F^*(p) = \delta(p).$$

Neither F^* nor G are uniformly convex, so we're limited to using Algorithm 1.

The function F^* is the same for both problems, and for any $\sigma > 0$ we have $(I + \sigma\partial F^*)^{-1}(p) = P_{\mathcal{K}}(p)$, the projection of p onto the set

$$\mathcal{K} = \{p \mid \|\{\| |p^j_i|_{\mathbb{R}^4} \|_{L_\infty(I^h)}\}\|_{\ell_1(\mathbb{N})} \le 1\}. \tag{4.2}$$

Although this projection is nontrivial, it is reasonable to compute; we give an algorithm for this operation in the next section.

For both (**??**) and (**??**) computing $(I+\tau\partial G)^{-1}$ involves simple, pointwise operations. For the first case we have

$$(I+\tau\partial G)^{-1}(g)_i = \frac{g_i + (\tau/\lambda) f_i}{1+\tau/\lambda},$$

while for (**??**) we have

$$(I+\tau\partial G)^{-1}(g)_i = \begin{cases} g_i - \tau/\lambda, & g_i \geq f_i + \tau/\lambda, \\ g_i + \tau/\lambda, & g_i \leq f_i - \tau/\lambda, \\ f_i, & \text{otherwise.} \end{cases}$$

To apply Algorithms 1 and 2 we need a bound on $L = \|K\|$. We have (using $(a+b+c)^2 \leq 3(a^2+b^2+c^2)$)

$$\begin{aligned} |\nabla_j^2 g_i|^2 \leq \frac{3}{(jh)^2}\Big(& g_{i+(j,0)}^2 + 4g_i^2 + g_{i-(j,0)}^2 + g_{i+(0,j)}^2 + 4g_i^2 + g_{i-(0,j)}^2 \\ & + \frac{1}{2}\big(g_{i+(j,j)}^2 + 4g_i^2 + g_{i-(j,j)}^2 + g_{i+(j,-j)}^2 + 4g_i^2 + g_{i-(j,-j)}^2\big)\Big). \end{aligned}$$

So

$$\sum_{j>0,\ i\in I^h} |\nabla_j^2 g_i|^2\, h^2 \leq \sum_{j>0} \frac{54}{j^2h^2} \sum_{i\in I^h} g_i^2\, h^2 = \frac{9\pi^2}{h^2}\|g\|_{L_2(I^h)}^2.$$

So $L = \|K\| \leq 3\pi/h \approx 9.4248/h$.

Theorem 2 of [**?**] gives an a priori error bound for the iterates of Algorithm 2 of the following form: Choose $\tau_0 > 0$, $\sigma_0 = 1/(\tau_0 L^2)$. Then for any $\epsilon > 0$ there is a N_0 (depending on ϵ and $\gamma\tau_0$) such that for any $N > N_0$

$$\|\hat{g} - g^N\|^2 \leq \frac{1+\epsilon}{N^2}\Big(\frac{\|\hat{g}-g^0\|^2}{\gamma^2\tau_0^2} + \frac{L^2}{\gamma^2}\|\hat{p}-p^0\|^2\Big).$$

Computational results are given in [**?**] that indicate that N_0 is quite moderate (≈ 100) in many situations.

In our case, we just want $\|\hat{g} - g^N\| \lessapprox \varepsilon$ greyscales. So we take

$$\frac{\|\hat{g}-g^0\|^2}{\gamma^2\tau_0^2} \leq \varepsilon^2$$

or (because $\|\hat{g}-g^0\| \leq 256$ greyscales, as pixel values lie between 0 and 256)

$$\frac{256}{\gamma\varepsilon} \leq \tau_0.$$

Then we take

$$\frac{1}{N^2}\frac{L^2}{\gamma^2}\|\hat{p}-p^0\|^2 \leq \varepsilon^2,$$

but $p^0 = 0$ and $\|\hat{p}\| \leq 1$, so

$$L/(\gamma\varepsilon) \leq N$$

We choose the least τ_0 and N that satisfy these two inequalities.

5. Computing $P_{\mathcal{K}}$. It remains to give the algorithm for $P_{\mathcal{K}}$, where $\mathcal{K}$ is given by (**??**). The problem can be formulated as: Given the sequence of vector fields $p = \{p_i^j\}$ find $q = \{q_i^j\}$ that minimizes

$$\frac{1}{2}\sum_{i,j}|p_i^j - q_i^j|^2 \text{ such that } \sum_j \max_i |q_i^j| \leq 1. \tag{5.1}$$

In practice we only compute p_i^j, $i \in I^h$, for a finite number of scales $j = 1, 2, 3, \ldots j_{max}$ where j_{max} is defined in section **??**. Thus given $p \in \mathbb{R}^{4\times N^2 \times j_{max}}$ we are searching for vector fields $q \in K \subseteq \mathbb{R}^{4\times N^2\times j_{max}}$ satisfying (**??**). Instead of directly solving (**??**), we can search for $q = \{q_i^j\}$ that minimizes

$$\frac{1}{2}\sum_{i,j}|p_i^j - q_i^j|^2 \text{ such that } |q_i^j| \leq \mu^j \text{ where } \sum_j \mu^j \leq 1, \text{ and } \mu^j \geq 0. \tag{5.2}$$

If μ is known, we know that at a minimizer, q satisfies

$$q_i^j = \begin{cases} \mu^j \dfrac{p_i^j}{|p_i^j|} & \mu^j < |p_i^j|, \\ p_i^j, & \mu^j \geq |p_i^j| \end{cases}. \tag{5.3}$$

Therefore, we aim to find appropriate $\mu \in \mathbb{R}^{j_{max}}$, a problem which can be simplified even further. To this end, denote the characteristic function on the set $\{i \in I^h \mid \mu^j < |p_i^j|\}$ as

$$[\mu^j < |p_i^j|] := \begin{cases} 1 \text{ if } \mu^j < |p_i^j| \\ 0 \text{ otherwise} \end{cases},$$

so at a minimum we have

$$\sum_{i,j}|p_i^j - q_i^j|^2 = \sum_{i,j}\left|p_i^j - p_i^j\frac{\mu^j}{|p_i^j|}\right|^2[\mu^j < |p_i^j|] = \sum_{i,j}|p_i^j|^2\left(1 - \frac{\mu^j}{|p_i^j|}\right)^2[\mu^j < |p_i^j|].$$

The Karush–Kuhn–Tucker theorem states that there are nonnegative Lagrange multipliers η^j ($\eta^j > 0 \Rightarrow \mu^j = 0$) and ν ($\nu > 0 \Rightarrow \sum_j \mu^j = 1$) such that the projector $P_{\mathcal{K}}p$ can be computed from the μ that minimizes

$$\frac{1}{2}\sum_{i,j}|p_i^j|^2\left(1 - \frac{\mu^j}{|p_i^j|}\right)^2[\mu^j < |p_i^j|] - \sum_j \mu^j\eta^j - \left(1 - \sum_j \mu^j\right)\nu.$$

Differentiating with respect to μ^j gives

$$\sum_i |p_i^j|^2\left(1 - \frac{\mu^j}{|p_i^j|}\right)\frac{-1}{|p_i^j|}[\mu^j < |p_i^j|] - \eta^j + \nu = 0,$$

or

$$\sum_i (|p_i^j| - \mu^j)[\mu^j < |p_i^j|] + \eta^j - \nu = 0. \tag{5.4}$$

If $\nu = 0$, then (**??**) implies that $\eta^j = 0$ for all j and $\mu^j \geq |p_i^j|$ for all i, j, so (**??**) implies $q = p$.

If $\nu > 0$ then $\sum_j \mu^j = 1$. If $\eta^j > 0$ for some j, then $\mu^j = 0$ and so $q_i^j = 0$ for all i by (**??**). If $\eta^j = 0$ then

$$\sum_i (|p_i^j| - \mu^j)[\mu^j < |p_i^j|] = \nu \tag{5.5}$$

which gives a one-to-one relationship between μ^j and ν that can also be written as

$$\mu^j(\nu) = -\frac{1}{|[\mu^j < |p_i^j|]|}\nu + \sum_{i \in [\mu^j < |p_i^j|]} |p_i^j|. \tag{5.6}$$

Note that as μ^j decreases, ν increases. Below we outline an algorithm for computing $\mu^j(\nu)$ for any $\nu > 0$. Once this is established, the projection $(P_{\mathcal{K}}p)^j$ can be computed as follows:

If $\{\| |p_i^j|_{\mathbb{R}^4} \|_{L_\infty(I^h)}\}\|_{\ell_1(\mathbb{N})} \leq 1$, then $P_{\mathcal{K}}p = p$. Otherwise, do the following:

I. Find a root $\bar{\nu}$ of

$$F(\nu) = 1 - \sum_{j=1}^{j_{\max}} \mu^j(\nu)$$

which guarantees the contraint $\sum_j \mu^j = 1$ holds. This can be done in a number of ways. We chose to use Ridders' algorithm [**?**], and this part of the algorithm takes an immeasurably small amount of time.

II. With this value of $\bar{\nu}$ we can calculate $\mu^j = \mu^j(\bar{\nu})$ for $j = 1, \ldots, j_{\max}$ and subsequently calculate $q^j = (P_{\mathcal{K}}p)^j$ using (**??**).

Below is the algorithm for computing $\mu^j(\nu)$ for any $\nu \geq 0$, which is based on the relationship (**??**).

For any fixed scale j:

1. Sort the N^2 numbers $\{|p_i^j| \mid 0 \le i_1, i_2 < N\}$ into non-increasing order and add a single element 0 at the end; call this sequence

$$\{a_\ell^j\},\ \ell = 0, \dots, N^2.$$

These are natural candidates for μ^j.

2. Calculate the sequence b_ℓ^j by

$$b_\ell^j = \sum_{k<\ell} a_k^j - \ell \times a_\ell^j, \quad \ell = 0, \dots, N^2.$$

Note that the pairs (b_ℓ^j, a_ℓ^j) lie on the graph of the one-to-one relation (**??**) between μ^j and ν.

3. Given any $\nu \ge 0$, we have the following mechanism for calculating $\mu^j(\nu)$:

   (a) If $\nu \ge b_{N^2}^j$, set $\mu^j(\nu) = 0$.

   (b) Otherwise, use binary search to calculate an index ℓ such that $b_\ell^j \le \nu < b_{\ell+1}^j$. Since (**??**) is linear between any two consecutive pairs (b_ℓ^j, a_ℓ^j) and $(b_{\ell+1}^j, a_{\ell+1}^j)$, with the relationship explicitly given by

$$\mu^j(\nu) = -\frac{1}{\ell+1}\nu + \sum_{i\in[a_\ell^j<|p_i^j|]} |p_i^j|,$$

   we can interpolate linearly between (b_ℓ^j, a_ℓ^j) and $(b_{\ell+1}^j, a_{\ell+1}^j)$ to find the point $(\nu, \mu^j(\nu))$.

$*$ The sorting, which takes time proportional to $j_{\max} N^2 \log N$, dominates the time of this part of the algorithm.

6. The main iteration. The iterations of Algorithm 1 or 2 must estimate $j_{\max}$ as n increases. Here we have no theoretical justification for our argument, and presumably in pathological cases we could get it wrong.

1. We pick an initial value of $j_{\max}$ (typically 4) and set p^j to be zero vector fields for $j = 1, \dots, j_{\max}$.
2. We iterate Algorithm 1 or 2 a fixed number of times.
3. We have a parameter $z_{\min}$ (we've used 4), which is the minimum number of vector fields $p^{j_{\max}}$, $p^{j_{\max}-1}$, etc., that we want to be zero vector fields. Heuristically, if we have $z_{\min}$ trailing zero vector fields p^j, then we think it unlikely that at a minimizer of (**??**) or (**??**) any vector fields p^j with $j > j_{\max}$ will be nonzero. So we adjust $j_{\max}$ so that precisely $z_{\min}$ trailing vector fields are zero vector fields.
4. We then decide whether to stop the iteration or go back to step 2.