An assessment of numerical and geometrical quality of bases on surface fitting on Powell-Sabin triangulations

M. A. Fortes[a], M. Raydan[b], M. L. Rodríguez[*,a], A. M. Sajo-Castelli[c]

[a] *Departamento de Matemática Aplicada, Universidad de Granada, Campus de Fuentenueva s/n, 18071 Granada, Spain*

[b] *Universidade Nova de Lisboa, Center for Mathematics and Applications (NovaMath), Campus da Caparica, 2829-516, Caparica, Portugal*

[c] *Departamento de Cómputo Científico y Estadística, Universidad Simón Bolívar, Edificio de Ciencias Básicas I, Caracas,Venezuela*

Abstract

It is well known that the problem of fitting a dataset by using a spline surface minimizing an energy functional can be carried out by solving a linear system. Such a linear system strongly depends on the underlying functional space and, particularly, on the basis considered. Some papers in the literature study the numerical behavior and processing of the above-mentioned linear systems in specific cases. The bases that have local support and constitute a partition of unity have been shown to be interesting in the frame of geometric problems. In this work, we investigate the numerical effects of considering these bases in the quadratic Powell-Sabin spline space. Specifically, we present a direct approach to explore different preconditioning strategies and assess whether the already known 'good' bases also possess favorable numerical properties. Additionally, we introduce an inverse optimization approach based on a nonlinear optimization model to identify new bases that exhibit both good geometric and numerical properties.

Keywords: Minimal energy, Preconditioning, Basis quality, Powell-Sabin, Surface fitting.

[*]Corresponding author

Email address:

mafortes@ugr.es,m.raydan@fct.unl.pt,miguelrg@ugr.es,asajo@usb.ve (M. A. Fortes[a], M. Raydan[b], M. L. Rodríguez[*,a], A. M. Sajo-Castelli[c])

1. Introduction

The problem of obtaining fitting surfaces to cloud of points constitutes an active research field due to its application to several problems arisen in different disciplines such as earth sciences, computer vision in robotics, structural simulations, and computer aided geometric design, among others; see e.g., [4, 12].

In most approaches for solving this problem, the fitting surface is a spline obtained by minimizing an *energy functional* defined in a suitable functional vector space and, frequently, a linear system must be solved in order to obtain it; see e.g., [1, 8, 15, 20]. For a review on this topic we recommend [14].

Most papers on the field of fitting surfaces rely on geometric aspects of the problem (for example approximation order of the fitting surface to the data points, smoothing properties, and geometric constraints to be fulfilled), but the literature on the numerical issues concerning the solution of the linear systems behind the fitting surface problem is still poorly explored. We think that this is an interesting matter insofar as obtaining proper, –faithful to the data points– fitting surfaces necessarily leads to large-scale linear systems whose conditioning, moreover, is strongly linked to the basis chosen in the spline functional. In this frame, in [20] an iterative proposal, based on the preconditioned conjugate gradient method, is presented in order to handle the linear system arisen when considering the usual Hermite basis, while in [12] a numerical inverse-free recursive method is developed to handle a fitting surface problem considered in a multi-resolution context. In [13] the authors show that preconditioning the coefficient matrix of a linear fitting system is in fact equivalent to considering the linear fitting system when working with another basis, i.e., they determine that a basis can be associated to each preconditioner and vice versa. In this frame, they consider the problem of determining whether good (or effective) preconditioners lead to good basis and vice versa, in the understanding that *good* bases are the ones constituting a partition of unity. Nevertheless, no numerical studies have been carried out in order to handle the linear systems arisen when using bases that have local support and constitute a partition of unity (see e.g. [8]). It is worth mentioning that these bases are considered to be somehow 'good' –and have been profusely used in the literature of fitting surfaces– as they present some attractive properties when handling geometric problems.

In this work we study the numerical aspects and we develop preconditioning strategies when considering these bases in the quadratic Powell-Sabin spline space. More precisely, we determine whether the preconditioning strategies for the ill-conditioned coefficient matrix of the linear system involved either

cluster the eigenvalues or manage to shift them into a very few discrete clusters.

Regarding Powell-Sabin, other literature has explored various aspects of this topic. It is worthy to mention [21] (where polar forms of the Powell-Sabin B-spline representation of quadratic polynomials or splines are introduced); [10] (where quasi-interpolation in a space of sextic splines defined over Powell-Sabin triangulations is studied); [16] (devoted to the construction of a normalized basis for a quadratic condensed Powell-Sabin-12 macro-element space) and [23] (handling $\mathcal{C}^1$-cubic splines over a triangulation with Powell-Sabin refinement).

The organization of the document is as follows. In Section 2 we recall the necessary basic concepts for this work, most of them related to the concept of partition of unity basis and to the underlying general fitting surface problem. In Section 3 we present a direct approach, based on the use of preconditioning strategies, to evaluate the numerical quality of known bases that have good geometric properties. In Section 4 we introduce an inverse approach, based on a nonlinear optimization model, for the discovery of potential new bases that exhibit both good geometric and numerical properties. Finally, in Section 5, we provide concluding remarks primarily focusing on the numerical findings presented in Sections 3 and 4.

2. Preliminaries

2.1. Powell-Sabin sub-triangulation

Let $D \subset \mathbb{R}^2$ be a polygonal domain, let $\mathcal{T}$ be a Δ^1- triangulation of $\overline{D}$ (see e.g. [6]) and $\mathcal{T}_6$ let be its associated Powell-Sabin triangulation $\mathcal{T}$ (a detailed description can be found in [13]). Let $\{t_k = (x_k, y_k)\}_{k=1}^{\ell}$ be the knots of $\mathcal{T}$. It is well known ([17]) that:

Theorem 1. *Given any set of triplets $\{\mathcal{F}_k = (f_k, f_{xk}, f_{yk})\}_{k=1}^{\ell}$ there exists a unique S in $\mathcal{S}_2^1(D, \mathcal{T}_6) = \left\{S \in \mathcal{C}^1(D) : S|_{T'} \in \mathbb{P}_2(T') \quad \forall T' \in \mathcal{T}_6\right\}$ such that*

$$S(t_k) = f_k, \quad \frac{\partial S}{\partial x}(t_k) = f_{xk}, \quad \frac{\partial S}{\partial y}(t_k) = f_{yk}, \qquad \text{for all } k = 1, \ldots, \ell. \quad (1)$$

Theorem 1 shows that $\mathcal{S}_2^1(D, \mathcal{T}_6)$ has dimension $N = 3\ell$ and provides an interpolation scheme for constructing a basis of $\mathcal{S}_2^1(D, \mathcal{T}_6)$. Later on in this work we will recall some well-known facts about how to construct 'proper' bases based on (1). In fact, we will study some numerical aspects related to

the choice of these bases. Before, we describe the fitting problem posed on the space $\mathcal{S}_2^1(D,\mathcal{T}_6)$ to be handled in the paper: let

$$\mathcal{P} = \{p_1, \ldots, p_q\} \tag{2}$$

be set of data points in D; $\mathcal{Z} = \{z_1, \ldots, z_q\} \in \mathbb{R}^q$, and $\rho : \mathcal{S}_2^1(D,\mathcal{T}_6) \longrightarrow \mathbb{R}^q$ defined by $\rho(v) = (v(p_1), \ldots, v(p_q))$. Let $|\cdot|_j$ represent the usual seminorm

$$|u|_j = \left(\sum_{|\beta|=j} \int_D \partial^\beta u(x)^2 dx \right)^{\frac{1}{2}}, \quad j = 1, 2;$$

for which $(\cdot,\cdot)_j$ represents the corresponding inner semi-product

$$(u, v)_j = \sum_{|\beta|=j} \int_D \partial^\beta u(x) \partial^\beta v(x) dx; \quad j = 1, 2;$$

and let us consider the functional $\mathcal{J} : \mathcal{S}_2^1(D,\mathcal{T}_6) \longrightarrow \mathbb{R}$ defined by

$$\mathcal{J}(v) = < \rho(v) - \mathcal{Z} >_q^2 + \tau_1 |v|_1^2 + \tau_2 |v|_2^2, \tag{3}$$

where $\tau_1 \geq 0$, $\tau_2 > 0$, and $< \cdot >_q$ stands for the usual Euclidean norm in $\mathbb{R}^q$. $\mathcal{J}$ provides a weighted measure of the fitting objective and of the 'minimal energy condition'. The functional $\mathcal{J}$ allows to dispose of certain fairness control of the fitting surface. In [1] it is shown that:

Theorem 2. *There exists a unique $\sigma \in \mathcal{S}_2^1(D,\mathcal{T}_6)$ minimizing $\mathcal{J}$.*

Given a basis $\mathcal{B} = \{v_1, \ldots, v_N\}$ of $\mathcal{S}_2^1(D,\mathcal{T}_6)$, the vector $(\xi_1, \ldots, \xi_N)$ leading to the unique $\sigma = \sum_{i=1}^N \xi_i v_i$ in Theorem 2 is the solution of the linear system

$$M_N x = b_N, \tag{4}$$

where $M_N = P_N\, P_N^T + Q_N$,

$$P_N = \begin{pmatrix} v_1(p_1) & \ldots & v_1(p_q) \\ \vdots & & \vdots \\ v_N(p_1) & \ldots & v_N(p_q) \end{pmatrix}; \quad Q_N = (\tau_1 (v_i, v_j)_1 + \tau_2 (v_i, v_j)_2)_{i,j=1}^N;$$

and $b_N = \left((< \rho(v_i), \mathcal{Z} >_q)_{i=1}^N \right)^T$. M_N is symmetric and, moreover, in [1] it is shown that it is also positive definite for all bases $\mathcal{B}$. An important issue to consider when solving (4) is how to choose a proper basis on the

interpolation scheme (1) insofar as the conditioning of this linear system strongly depends on $\mathcal{B}$. A very well-known one is the *Hermite basis*, defined as the set $\{v_1, \ldots, v_N\}$ verifying

$$\left\{\begin{array}{l} v_{3(i-1)+1}(t_k) = \delta_{ik}, \\ \frac{\partial v_{3(i-1)+1}}{\partial x}(t_k) = 0, \\ \frac{\partial v_{3(i-1)+1}}{\partial y}(t_k) = 0, \end{array}\right. \quad \left\{\begin{array}{l} v_{3(i-1)+2}(t_k) = 0, \\ \frac{\partial v_{3(i-1)+2}}{\partial x}(t_k) = \delta_{ik}, \\ \frac{\partial v_{3(i-1)+2}}{\partial y}(t_k) = 0, \end{array}\right. \quad \left\{\begin{array}{l} v_{3(i-1)+3}(t_k) = 0, \\ \frac{\partial v_{3(i-1)+3}}{\partial x}(t_k) = 0, \\ \frac{\partial v_{3(i-1)+3}}{\partial y}(t_k) = \delta_{ik}, \end{array}\right. \tag{5}$$

for $i, k = 1, \ldots, \ell$.
In [8] the authors considered, for each t_k, the B-splines B_k^j, for $j = 1, 2, 3$, solutions of the interpolation problem (1) with data

$$\begin{array}{l} \mathcal{F}_i^j = (0,0,0) \text{ for } i \neq k \text{ and } j = 1,2,3, \\ \mathcal{F}_k^j = (\alpha_k^j, \beta_k^j, \gamma_k^j) \neq (0,0,0) \text{ for } j = 1,2,3, \end{array} \tag{6}$$

where $(\alpha_k^1, \beta_k^1, \gamma_k^1)$, $(\alpha_k^2, \beta_k^2, \gamma_k^2)$ and $(\alpha_k^3, \beta_k^3, \gamma_k^3)$ are three linearly independent vectors for each k. It holds then that $\{B_i^j\}_{1 \leq i \leq \ell, 1 \leq j \leq 3}$ is a basis of $\mathcal{S}_2^1(D, \mathcal{T}_6)$ where each basis function B_k^j has local support since it vanishes outside the ***molecule*** M_k of vertex t_k, defined as the union of all triangles of $\mathcal{T}$ having t_k as vertex.
It is well known (see e.g. [4]) that disposing of B-spline basis $\{B_i^j\}_{1 \leq i \leq \ell, 1 \leq j \leq 3}$ which form a partition of unity, i.e., verifying

$$\sum_{i=1}^{\ell} \sum_{j=1}^{3} B_i^j(x,y) = 1 \qquad \text{for all } (x,y) \in \overline{D} \qquad \text{and} \tag{7}$$

$$B_k^j(x,y) \geq 0 \qquad \text{for all } (x,y) \in \overline{D}, \tag{8}$$

has several geometric advantages. It easy to show that imposing (7) to the basis $\{B_i^j\}_{1 \leq i \leq \ell, 1 \leq j \leq 3}$ is equivalent to ask the equations

$$\alpha_k^1 + \alpha_k^2 + \alpha_k^3 = 1; \qquad \beta_k^1 + \beta_k^2 + \beta_k^3 = 0; \qquad \gamma_k^1 + \gamma_k^2 + \gamma_k^3 = 0 \tag{9}$$

hold for all $k = 1, \ldots, \ell$. On the other hand, for all knot t_k let T_k be the associated so-called *control triangle*, defined as the one having vertices

$$\begin{array}{l} Q_{k1} = (X_{k1}, Y_{k1}) = \left(x_k + \frac{\alpha_k^2\gamma_k^1 + \gamma_k^2(1-\alpha_k^1)}{d_k}, y_k + \frac{-\alpha_k^2\beta_k^1 - \beta_k^2(1-\alpha_k^1)}{d_k}\right), \\ Q_{k2} = (X_{k2}, Y_{k2}) = \left(x_k + \frac{-\alpha_k^1\gamma_k^2 - \gamma_k^1(1-\alpha_k^2)}{d_k}, y_k + \frac{\alpha_k^1\beta_k^2 + \beta_k^1(1-\alpha_k^2)}{d_k}\right), \\ Q_{k3} = (X_{k3}, Y_{k3}) = \left(x_k + \frac{-\alpha_k^1\gamma_k^2 + \gamma_k^1\alpha_k^2}{d_k}, y_k + \frac{\alpha_k^1\beta_k^2 - \beta_k^1\alpha_k^2}{d_k}\right), \end{array} \tag{10}$$

where $d_k = \begin{vmatrix} \alpha_k^1 & \alpha_k^2 & \alpha_k^3 \\ \beta_k^1 & \beta_k^2 & \beta_k^3 \\ \gamma_k^1 & \gamma_k^2 & \gamma_k^3 \end{vmatrix} = \beta_k^1\gamma_k^2 - \gamma_k^1\beta_k^2 \neq 0$. In [8] it is shown that in order that the functions B_k^j satisfy (8) we just have to consider values $\{\alpha_k^j, \beta_k^j, \gamma_k^j\}$ in (6) in such a way that for all $k = 1, \ldots, \ell$

$$\begin{array}{l} T_k \text{ contains } t_k \text{ and all the points having barycentric coordinates} \\ (1,1,0) \text{ and } (1,0,1) \text{ with respect to all the triangles } \triangle(t_k, P, Q) \\ \text{of } \mathcal{T}_6 \text{ having } t_k \text{ as a common vertex.} \end{array} \tag{11}$$

Summarizing, the basis $\{B_i^j\}_{1\leq i\leq \ell, 1\leq j\leq 3}$ will constitute a partition of unity whenever the values $\{\alpha_k^j, \beta_k^j, \gamma_k^j\}$ in (6) verify (9) and (11).
In order to construct partition of unity basis we can also consider the inverse problem, i.e., to define, for $k = 1, \ldots, \ell$, triangles

$$\widetilde{T}_k = \{(\widetilde{X}_{k1}, \widetilde{Y}_{k1}), (\widetilde{X}_{k2}, \widetilde{Y}_{k2}), (\widetilde{X}_{k3}, \widetilde{Y}_{k3})\} \tag{12}$$

containing all points described in (11) and then compute the values $\{\alpha_k^j, \beta_k^j, \gamma_k^j\}$ by means of ([8])

$$\begin{pmatrix} \alpha_k^1 & \alpha_k^2 & \alpha_k^3 \\ \beta_k^1 & \beta_k^2 & \beta_k^3 \\ \gamma_k^1 & \gamma_k^2 & \gamma_k^3 \end{pmatrix} \begin{pmatrix} \widetilde{X}_{k1} & \widetilde{Y}_{k1} & 1 \\ \widetilde{X}_{k2} & \widetilde{Y}_{k2} & 1 \\ \widetilde{X}_{k3} & \widetilde{Y}_{k3} & 1 \end{pmatrix} = \begin{pmatrix} x_k & y_k & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}. \tag{13}$$

Next we describe some bases satisfying (9) and (11) previously introduced in the literature. In all cases, it holds that $\mathcal{F}_k$ in (6) are the same for all knots k. A more detailed introduction of these basis is provided in [13].

- *BShi*: They considered ([22]), for all $k = 1, \ldots, \ell$,

$$\mathcal{F}_k^1 = \left(\frac{1}{4}, 0, \varepsilon\right);\ \mathcal{F}_k^2 = \left(\frac{1}{4}, \varepsilon, 0\right);\ \mathcal{F}_k^3 = \left(\frac{1}{2}, -\varepsilon, -\varepsilon\right),$$

  where $\varepsilon \in \left[\frac{1}{4h}, \frac{1}{2h}\right]$ and h is the length of the longest edge of $\mathcal{T}$.

- *BLP* (bases based on linear programming): We consider

$$\mathcal{F}_k^1 = \left(\alpha_k^1, 0, \varepsilon\right);\ \mathcal{F}_k^2 = \left(\alpha_k^2, \varepsilon, 0\right);\ \mathcal{F}_k^3 = \left(1 - \alpha_k^1 - \alpha_k^2, -\varepsilon, -\varepsilon\right).$$

  In the numerical sections of this work we will consider $\mathrm{t}D = (0,1) \times (0,1)$ and the Δ^1-triangulations $\mathcal{T}^n$, defined as the one associated to the uniform partition of $[0,1]$ into n subintervals. It can be checked by doing some calculations that for $\mathcal{T}^n$ condition (11) is equivalent to request $\varepsilon \in [-n, n], \varepsilon \neq 0$, and $\alpha_1 = \alpha_2 = \frac{|\varepsilon|}{3n}$.

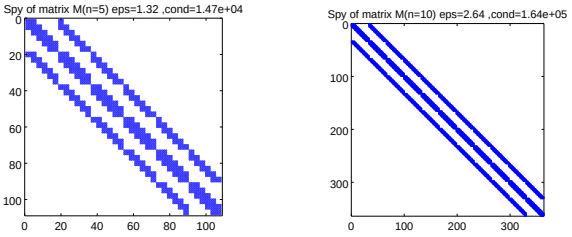

Figure 1: Spy plot of M_n for $n = 5$ and $n = 10$.

- *BCT*: For triangulations $\mathcal{T}^n$ we have considered, for all $k = 1, \ldots, \ell = (n+1)^2$, the triangle $\widetilde{T}_k$ introduced in (12) to be

$$\begin{cases} (\widetilde{X}_{k1}, \widetilde{Y}_{k1}) = t_k + 2r_k(\cos(\varphi), \sin(\varphi)), \\ (\widetilde{X}_{k2}, \widetilde{Y}_{k2}) = t_k + 2r_k \left(\cos\left(\varphi - \frac{2\pi}{3}\right), \sin\left(\varphi - \frac{2\pi}{3}\right)\right), \\ (\widetilde{X}_{k3}, \widetilde{Y}_{k3}) = t_k + 2r_k \left(\cos\left(\varphi + \frac{2\pi}{3}\right), \sin\left(\varphi + \frac{2\pi}{3}\right)\right), \end{cases} \tag{14}$$

  where $r_1 = r_{(n+1)^2} = \frac{1}{4n}, r_i = \frac{\sqrt{5}}{6n}$ for $i \neq 1, (n+1)^2$ and $\varphi \in \left[0, \frac{2\pi}{3}\right]$.

  The sets $\mathcal{F}_k^j$ are then obtained by using (13).

3. Numerical assessment of some well-known partition of unity bases

All the experiments in this section and in the next one have been carried out (using Octave with double precision) over the domain $D = (0,1) \times (0,1)$, using the triangulation $\mathcal{T}^n$, with $q = 2000$ data points in (2), and with smoothing parameters values $\tau_1 = 10^{-5}$ and $\tau_2 = 10^{-9}$ in (3). Let us denote the coefficient matrix M_N in (4) to be M_n when handling the triangulation $\mathcal{T}^n$. We are interested in solving the sparse large-scale linear system $M_n x = b$ using the well-known Preconditioned Conjugate Gradient method (PCG). In this context, M_n is a sparse block tridiagonal matrix. Spy plots in Figure 1 show the sparsity structure. The Conjugate Gradient (CG) iterative scheme only requires one matrix-vector product per iteration and hence it is computationally efficient. Moreover, at each iteration j, x_j is the only point in the subspace explored so far that minimizes the distance to the solution x^* in the M_n-norm (i.e., it minimizes $\|x_j - x^*\|_{M_n}^2 = (x_j - x^*)^T M_n (x_j - x^*)$). Therefore, the CG method ends up with the unique solution of the system $M_n x = b$ in at most p iterations, where p is the number of distinct eigenvalues of M_n (see [5], [18]). In order to apply PCG we seek to evaluate the performance of incomplete factorization preconditioners (accelerators).

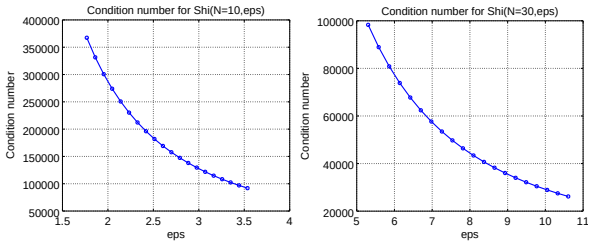

Figure 2: Condition number $\kappa(M_n^{Shi}(\varepsilon))$ vs. parameter ε ($n = 10, 30$).

In particular, we are interested in observing if these preconditioners either cluster the eigenvalues of the preconditioned matrix around unity and thus $\kappa(C^{-1}\, M_n) \ll \kappa(M_n)$ or manage to shift the eigenvalues into a very few discrete clusters (see e.g. [5, 18]). We present four classes of M_n matrices: the ones associated to bases *Shi*, *BLP* and *BCT*, and the one associated to Hermite basis (5), in order to compare the results for partition of unity basis with another one not having this property. We report on the quality of the preconditioners C from the incomplete factorizations family, in particular Cholesky (ichol(0), $C = LL^T \approx M_n$) and LU (ilu(0), $C = LU \approx M_n$). These preconditioner have no fill-in. The matrices associated to *Shi*, *BLP* and *BCT* depend on a parameter ε. In all cases, we first give an insight of how the matrix condition number depends on ε. Then we construct the preconditioner ichol(0) or ilu(0) and we show in the left column (in blue) the eigenvalues and the associated histogram for $M_n(\varepsilon^*)$, where ε^* has been chosen as the one leading to a lower condition number. In the right column (in red) are reported the eigenvalues for the preconditioned matrix $C^{-1}M_n(\varepsilon^*)$. We also include a table reporting, for $n = 5, \ldots, 40$, the $\kappa\big(M_n(\varepsilon^*)\big)$, $\lambda_{\min}/\lambda_{\max}$ of $C^{-1}M_n(\varepsilon^*)$ and its spectral radius (in these tables ‡ indicates that $\kappa(M_n)$ is a coarse ℓ_1 estimation).

3.1. Class Shi

Matrix $M_n^{\mathrm{Shi}}(\varepsilon)$ is parameterized via $\varepsilon \in [n/(4\sqrt{2}), n/(2\sqrt{2})]$. Figure 2 shows the matrix condition number with respect to ε for different values of n.

3.2. Class BLP

Matrix $M_n^{\mathrm{BLP}}(\varepsilon)$ is parameterized via $\varepsilon \in [-n, n]\backslash\{0\}$. Figure 4 shows the matrix condition number with respect to ε for different values of n. Very poor conditioning can be observed in the neighborhood of zero.

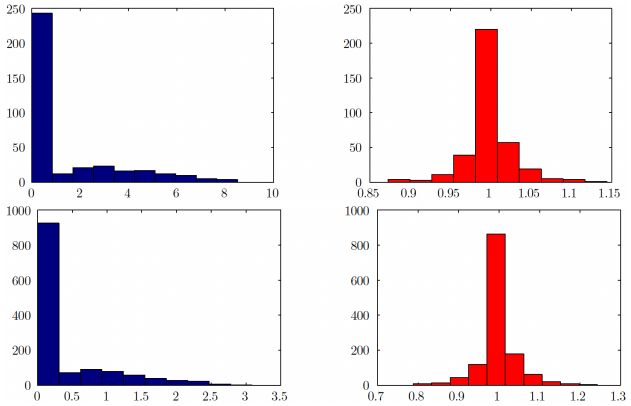

Figure 3: Histograms of eigenvalues for $C^{-1}M_n^{Shi}(\varepsilon^*)$ $(n = 10, 20)$.

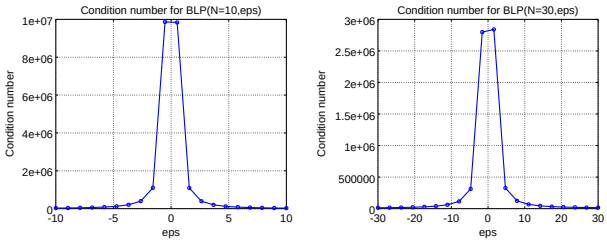

Figure 4: Condition number $\kappa(M_n^{BLP}(\varepsilon))$ vs. parameter ε $(n = 10, 30)$.

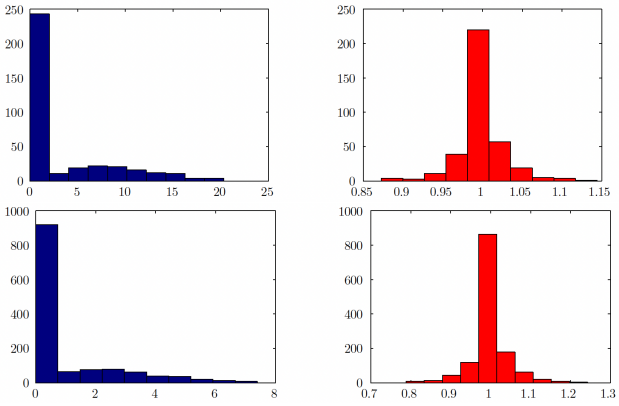

Figure 5: Eigenvalues of $C^{-1}M_n^{BLP}(\varepsilon^*)$ (left) and its corresponding Precond matrix (right), for $N = 10, 20$.

n	$\kappa(M_n)$	C	$\lambda_{\max}/\lambda_{\min}$	$\lambda(C^{-1}M_n)$
5	8.127×10^4	ichol	1.18	[0.923, 1.093]
10	9.19×10^4	ichol	1.40	[0.8723, 1.224]
20	5.24×10^4	ichol	1.59	[0.7883, 1.256]
30	‡ 1.20×10^5	ilu	1.84	[0.711, 1.306]
40	‡ 9.86×10^4	ichol	1.89	[0.693, 1.308]

Table 1: Conditioning of M_n^{Shi}.

n	$\kappa(M_n)$	C	$\lambda_{\max}/\lambda_{\min}$	$\lambda(C^{-1}M_n)$
5	3.21×10^3	ichol	1.05	[9.22e-1, 9.75e-1]
10	2.76×10^4	ichol	1.17	[8.72e-1, 1.024]
20	1.99×10^4	ichol	1.40	[7.88e-1, 1.107]
30	‡ 1.38×10^4	ilu	2.10	[7.11e-1, 1.493]
40	‡ 9.07×10^7	ichol	2.19	[6.92e-1, 1.523]

Table 2: Conditioning of M_n^{BLP}.

3.3. Class BCT

Matrix $M_n^{\mathrm{BCT}}(\varphi)$ is parameterized via $\varphi \in [0, 2\pi/3]$. Figure 6 shows the matrix condition number with respect to ε for $n = 5$. For this case φ does not influence the matrix condition number.

3.4. Class Hermite

For the Hermite basis, M_n does not depend on any parameter.
In Table 4,"ichol+d I" indicates that ichol(0) is applied to $M_n + d\ I_n$ for $d > 0$, and I_n being the identity matrix of size n. We note that the Hermite basis is not competitive, from the numerical point of view, as compared with the partition of unity bases.

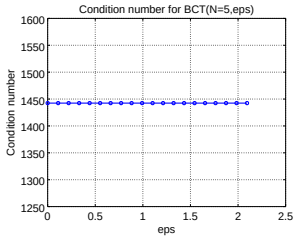

Figure 6: Condition number $\kappa(M_n^{BCT}(\varepsilon))$ vs. parameter ε ($n = 5$).

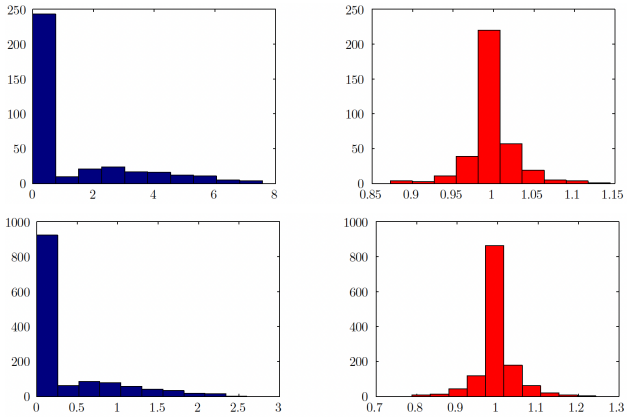

Figure 7: Histogram of eigenvalues of $C^{-1}M_n^{BCT}(\varepsilon^*)$ (left) and its corresponding Precond matrix (right), for $n = 10, 20$ and a suitable and arbitrary φ.

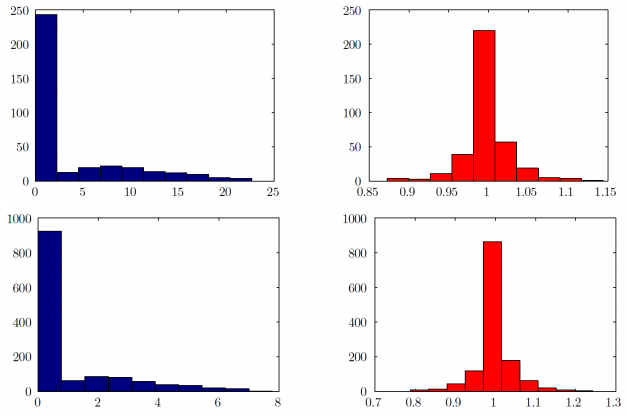

Figure 8: Histogram of eigenvalues of $C^{-1}M_n^{Hermite}(\varepsilon^*)$ (left) and its corresponding Precond matrix (right), for $n = 10, 20$.

n	$\kappa(M_n)$	C	$\lambda_{\max}/\lambda_{\min}$	$\lambda(C^{-1}M_n)$
5	1.44×10^3	ichol	1.123	[9.22e-1, 1.036]
10	3.51×10^4	ichol	1.345	[8.723e-1, 1.174]
20	7.34×10^4	ichol	1.734	[7.883e-1, 1.367]
30	‡ 1.73×10^5	ilu	1.991	[7.110e-1, 1.415]
40	‡ 2.20×10^5	ichol	2.111	[6.925e-1, 1.462]

Table 3: Conditioning of M_n^{BCT}.

n	$\kappa(M_n)$	C	$\lambda_{\max}/\lambda_{\min}$	$\lambda(C^{-1}M_n)$
5	1.19×10^5	ichol	1.524	[9.22e-1, 1.405]
10	3.14×10^6	ichol	1.75	[8.72e-1, 1.153]
20	6.42×10^6	ichol	1.9	[7.88e-1, 1.5]
30	‡ 1.43×10^7	ichol	2.107	[7.11e-1, 1.498]
40	‡ 1.76×10^7	ichol+0.1 I	7.003	[1.64e-1, 1.153]

Table 4: Conditioning of $M_n^{Hermite}$.

4. Inverse approach to evaluate the relationship between numerical and geometric quality of bases

In Section 3, we have established that the 3 known bases (*BShi*, *BLP* and *BCT*), which have excellent geometric properties, also have very good numerical properties for solving the associated linear systems. In this section, we plan to study whether in fact any basis that satisfies the constraints (9) also produces good numerical results. Furthermore, we would like to explore the possibility that without forcing the geometric conditions given by (9), we can still generate good bases from a numerical point of view. To accomplish these objectives, we will present a suitabrepeatle inverse model, and we will propose a low-cost effective numerical scheme for iteratively solving the constrained optimization problems that naturally arise from it.

Let us start by recollecting a few facts that will be needed throughout this section: M_N is symmetric and positive definite where $N = 3(n+1)^2$. The matrices M_N have a band structure, with a bandwidth much smaller than the dimension (especially when the value of n increases), and within the band there is a structure of zeros and non-zeros, that is, the band of M_N is in turn sparse. Moreover, the matrices M_N are ill-conditioned, i.e., $\kappa(M_N) = \lambda_{\max}(M_N)/\lambda_{\min}(M_N)$ is a large number, and that ill-conditioning increases when the value of n increases.

As in *BShi*, *BLP* and *BCT*, we consider $\mathcal{F}_k$ in (6) to be the same for all

knots, in such a way that M_N and b_N depend only on the 9 real parameters $\{(\alpha_i, \beta_i, \gamma_i),\ i = 1, 2, 3\}$, which in the direct problem (already understood and solved in Section 3) are given, and in the inverse case they are the unknowns whose values must be obtained.
Our plan is to explore an inverse approach, that is, assign values to the parameters $\alpha_i, \beta_i, \gamma_i$ in such a way that the matrix M_N has attractive properties when solving the linear system, always in the PCG context. Whether they are attractive or good will become clear later and will be modeled with an appropriate optimization problem based on a smooth objective function subject to linear equality constraints.

Optimization model and a low-cost numerical method

Consider $P = \{p_i = (\alpha_i, \beta_i, \gamma_i) \in \mathbb{R}^3, i = 1, 2, 3)\}$, and let Ω_{M_N} be the set of all square matrices with the same dimension and sparse pattern of M_N. We propose to solve the following optimization problem:

$$\begin{aligned} \min_{p \in P} \quad & f(p) = \tfrac{1}{2}\left\|C_N\, M_N(p) - I_N\right\|_F^2 \\ \text{subject to} \quad & \textstyle\sum_{i=1}^{3} \alpha_i = 1,\ \ \sum_{i=1}^{3} \beta_i = 0,\ \ \sum_{i=1}^{3} \gamma_i = 0. \end{aligned} \tag{15}$$

We note that the symmetric matrices I_N (Identity matrix), C_N, and $M_N(p)$ are all $N \times N$, and that the objective function $f : \mathbb{R}^9 \to \mathbb{R}$. From now on, for simplicity, $M_N(p)$ will also be denoted as M_N. We also note that the 3 linear equality constraints, that define the feasible region of (15), have been justified in Section 2 and can also be written as $p_1 + p_4 + p_7 = 1$, $p_2 + p_5 + p_8 = 0$, and $p_3 + p_6 + p_9 = 0$.
The motivation for the optimization problem (15) is the following. This problem consists of finding a convenient matrix M_N (choosing the 9 variables) so that later when applying the direct approach (already studied) a preconditioning matrix C_N of M_N is obtained such that when multiplying both, that product $C_N\, M_N$ is as close as possible to the identity matrix I_N in Frobenius norm, which for a given matrix A is defined as

$$\|A\|_F = \left(\sum_{i=1}^{N}\sum_{j=1}^{N} |a_{ij}|^2\right)^{1/2}.$$

This effectively indicates that C_N approximates M_N^{-1} without having to compute the inverse explicitly. Then, of course, that matrix C_n will become

the preconditioner to be used when solving the system $M_N x = b_N$. Observe that for any choice of the variables, the matrix M_N will indeed have the band sparse pattern already described above, and then, by solving the direct problem on M_N, a C_N is generated with "Incomplete LU or Choleski" ideas that generate a matrix $C_N \in \Omega_{M_N}$ with no additional effort. Hence, we do not need to impose $C_N \in \Omega_{M_N}$.
The methods to be proposed for solving the non-linear problem (15) are, obviously, iterative. We can offer a general algorithmic scheme without describing the appropriate method in detail. Starting at a feasible vector p_0, set $k = 0$ and do the following:

- **Step 1:** If the stopping criteria of the chosen numerical method are satisfied, STOP (returning p_k).
- **Step 2:** If not, build $M_N(p_k)$ (that has the band structure described above).
- **Step 3:** Apply to $M_N(p_k)$ the direct approach (already developed) to build $C_N^k \in \Omega_{M_N}$.
- **Step 4:** Using C_N^k as a fixed matrix, apply the chosen numerical scheme to find a feasible vector p_{k+1} (the next iterate), set $k = k + 1$, and repeat steps 1–4.

Notice that, as it usually happens, solving an inverse problem requires solving a sequence of direct problems, as indicated in **Step 3** of the general iterative algorithm. Effective low-cost numerical methods to solve (15) will need the gradient of the function $f(p)$ to obtain p_{k+1} at the **Step 4** of the algorithm. For the sake of simplicity, let us denote

$$F(p) = C_N\, M_N(p) - I_N.$$

Notice that $F : \mathbb{R}^9 \to \mathbb{R}^{N\times N}$, and also that $f(p)$ can be written as

$$f(p) = \frac{1}{2}\langle F(p), F(p)\rangle_F,$$

where for any given square matrix A, $\langle A, A\rangle_F = \text{trace}(A^T A) = \|A\|_F^2$. Consider the auxiliary function $g : \mathbb{R} \to \mathbb{R}$, given by $g(t) = f(p + tZ)$, for any arbitrary vector Z. From basic calculus we know that $g'(0) = \langle \nabla f(p), Z\rangle_F$. After simple manipulations, using trace properties, we obtain

$$g'(0) = \langle F(p), F'(p)Z\rangle_F = \langle F'(p)^T F(p), Z\rangle_F.$$

Consequently, the gradient of $f(p)$, $\nabla f(p) : \mathbb{R}^9 \to \mathbb{R}^9$, is given by

$$\nabla f(p) = F'(p)^T F(p),$$

where $F'(p)$ (the first derivative or Fréchet derivative, also called "Jacobian" of F at p) is given by $F'(p) = C_N\, M_N'(p)$, where in turn $M_N'(p)$ is the first derivative of $M_N(p)$. Hence, recalling that C_N is symmetric, we obtain

$$\nabla f(p) = M_N'(p)^T (C_N F(p)). \tag{16}$$

As expected, $\nabla f(p)$ in (16) is a vector (one column) in $\mathbb{R}^9$. The $N \times N$ matrix $C_N F(p)$ can be conveniently transformed (stacking up its columns) into a one vector in $\mathbb{R}^{N^2}$. Similarly, let us visualize $M_N(p)$ as a long vector of N^2 functions, each one of them depending on the 9 parameters. That way, $M_N'(p)$ is a genuine Jacobian matrix with N^2 rows and 9 columns. Therefore, the inner product in (16) returns a vector in $\mathbb{R}^9$. Concerning the low-cost numerical methods to obtain p_{k+1} at **Step 4**, a convenient gradient-type scheme is the (inexact) Spectral Projected Gradient (SPG) method [2]. In the SPG method, the iterates are given by

$$p_{k+1} = p_k + \alpha_k (P_\Omega(p_k - \lambda_k \nabla f(p_k)) - p_k),$$

where $P_\Omega(z)$ denotes the projection of z onto the intersection of the 3 linear constraints, the step length $\lambda_k > 0$ is obtained with two inner products involving the gradient vector, and $0 < \alpha_k \leq 1$ is chosen using a non-monotone backtracking globalization strategy. At each SPG iteration we need only one projection on the feasible convex set, and that can be obtained using the classical alternating projection scheme on the intersection of the 3 linear varieties; see, e.g., [11, Chapter 3]. If we want to ignore the 3 constraints (i.e., if we do not want to force the geometric conditions given by (9)), this method is also convenient, and we only need to forget the projection operator and set $\alpha_k = 1$ for all k. In that case, the iterates are given by

$$p_{k+1} = p_k - \lambda_k \nabla f(p_k),$$

and the backtracking, if needed, is performed directly on λ_k. In the case of avoiding the 3 linear constraints, another practical option is the Gauss-Newton method $p_{k+1} = p_k + \lambda_k d_k^{GN}$, where $\lambda_k = 1$ unless a backtracking is required and if so, we reduce λ_k properly. The search direction d_k^{GN} is obtained by solving the linear system $(F'(p_k)^T F'(p_k)) d_k^{GN} = -\nabla f(p_k)$; see, e.g., [7] for details. Notice that, using the stacking strategy described above,

this linear system involves 9 equations with 9 unknowns. Hence, the Gauss-Newton method requires additional computational cost per iteration, but it approximates the Newton direction locally and thus it might require fewer iterations than the SPG scheme. A clear advantage of the SPG method is that it is easy to code, requires low storage, and avoids the need for matrix factorizations (no Hessian matrix is used). More details can be found in [3] and references therein.

Numerical results

We conducted a large number of numerical experiments using the SPG method within the general iterative algorithm, described above, starting from random initial points $p_0 \in \mathbb{R}^9$. Our goal was to identify bases with good numerical properties. We ran these experiments by either forcing the solution vector p to be within the feasible region of (15) or by ignoring the 3 linear constraints. To achieve this goal, we set $n = 5$, resulting in $N = 108$. It is important to note that Steps 2 and 3 of the general iterative algorithm are computationally very expensive, and this cost increases significantly with larger values of N. We are convinced that to conduct thorough exploratory work, by running a large number of tests, selecting $N = 108$ is adequate.
This initial set of experiments reveals that, with few exceptions, the solutions obtained from random initial points are numerically poor. Most of the time, both the condition number of the matrix M_N and the matrix $C_N\, M_N$, where C_N is obtained after applying preconditioning techniques, are very high. The typical obtained condition numbers are $\kappa(M_N) \approx 10^9$ and $\kappa(C_N\, M_N) \approx 10^6$. Although the condition number of $C_N\, M_N$ is lower than that of M_N, it is still very high. In other words, using those bases would require a large number of PCG iterations to solve the preconditioned linear system, regardless of whether p is forced to be in the feasible region or not. Although there is a slight tendency to find good bases when the three linear constraints are imposed, it is almost imperceptible. Nevertheless, using this random exploration, numerically good bases have been discovered (very seldom). Table 5 presents two of those numerically good bases: one within the feasible region and the other outside of it. For both bases, we are reporting the entries of the p vector, the evaluation of the 3 linear constraints, the value of the objective function $f(p)$, $\kappa(M_N)$, and $\kappa(C_N\, M_N)$.
For our second set of experiments, we aimed to evaluate the quality of the three known bases ($BShi(\varepsilon)$, $BLP(\varepsilon)$ and $BCT(\varphi)$) using the inverse approach. To do this, we reapply the SPG method within the framework of the general iterative algorithm, but now choosing as initial iterations small

p (feasible)	p (not feasible)
0.76494	0.07239
1.79997	1.95058
1.03767	-0.66282
-0.28752	-0.25973
-0.52458	1.08007
-2.88065	1.12298
0.52258	0.44957
-1.27539	0.95001
1.84298	0.80622
$p(1)+p(4)+p(7)=1$	$p(1)+p(4)+p(7)=0.26223$
$p(2)+p(5)+p(8)=0$	$p(2)+p(5)+p(8)=3.98066$
$p(3)+p(6)+p(9)=0$	$p(3)+p(6)+p(9)=1.26639$
$f(p)=35.82$	$f(p)=26.4056$
$\kappa(M_N)=3.6\times10^4$	$\kappa(M_N)=1.1\times10^4$
$\kappa(C_N\,M_N)=125$	$\kappa(C_N\,M_N)=80.07$

Table 5: Two numerically good bases obtained using the SPG method from random initial points.

random perturbations of the three bases, and making several valid choices of the parameters (ε or φ) in each case (see Section 3). The results obtained show that the three bases are solutions to the optimization problem (15) for any valid choice of parameters. Furthermore, as a by-product, we note that the objective function in (15) alone can be used as a merit function to evaluate the numerical quality of any potential basis. Similarly, it is also worth noting that $\kappa(C_N\,M_N)$ (when compared with $\kappa(M_N)$) resulting from Steps 2 and 3 of the general iterative algorithm is also a reliable measure of the numerical quality of potential bases. Associated with the three known bases, and taking advantage on those merit functions that we have detected thanks to the inverse approach, Table 6 shows the 3 best versions with their respective optimal parameters. Note that the best versions of each basis match the results reported in Section 3 (Figures 2, 4, and 6).

Based on the latest results and the availability of three reliable numerical merit functions, we are motivated to study the bases that can be found within the triangle in $\mathbb{R}^9$ that is defined by the three optimal bases mentioned in Table 6, which are used as the vertices of the triangle. The values of p leading to the three optimal bases, or vertices of the triangle, will be denoted as $BShi^*(\varepsilon)$, $BLP^*(\varepsilon)$ and $BCT^*(\varphi)$, respectively. The associated optimal bases will be denoted as $BShi^*$, BLP^*, and BCT^*.

$p = BShi^*(\varepsilon),\ \varepsilon = n/(2\sqrt{2})$	$p = BLP^*(\varepsilon),\ \varepsilon = n$	$p = BCT^*(\varphi),\ \varphi = 1$
1/4	1/3	1/3
0	0	2.4163
1.76776	5.0	3.7632
1/4	1/3	1/3
1.76776	5.0	2.05085
0	0	-3.97416
1/2	1/3	1/3
-1.76776	-5.0	-4.46715
-1.76776	-5.0	0.211
$f(p) = 26.82$	$f(p) = 7.71$	$f(p) = 10.1$
$\kappa(M_N) = 8270.3$	$\kappa(M_N) = 930.3$	$\kappa(M_N) = 1442.4$
$\kappa(C_N\, M_N) = 32.2$	$\kappa(C_N\, M_N) = 6.37$	$\kappa(C_N\, M_N) = 8.73$

Table 6: The 3 feasible bases *BShi**, *BLP**, and *BCT** for the best possible parameters (ε or φ), and the value of the numerical merit functions $f(p)$, $\kappa(M_N)$, and $\kappa(C_N\, M_N)$.

Notice now that if we consider convex combinations of the form

$$p = \gamma_1\, BShi^*(\varepsilon) + \gamma_2\, BLP^*(\varepsilon) + \gamma_3\, BCT^*(\varphi), \tag{17}$$

where $\gamma_1 + \gamma_2 + \gamma_3 = 1$ and $\gamma_i \geq 0$ for $1 \leq i \leq 3$, and since the 3 bases satisfy the 3 linear constraints, then automatically p is a point of that triangle that satisfies the linear constraints, and therefore p is in the feasible region. In fact any convex combination of the bases *BShi**, *BLP**, and *BCT** will be a partition of unity basis. It is worth noting that this triangle, which is a proper subset of the feasible region, offers the possibility of obtaining infinitely many bases with good numerical quality. Table 7 displays a discrete selection of these bases, and their corresponding numerical properties, generated by systematically exploring the interior of the triangle.
The first three rows in Table 7 correspond to the bases at the center of each edge of the triangle. The fourth row corresponds to the midpoint of the triangle, and the last 3 rows are bases also inside but each one closer to one of the corners of the triangle. Obviously, there are many more good quality bases inside the triangle, which can be obtained by changing γ_i $(1 \leq i \leq 3)$ according to (17). We note from this discrete set of bases that convex linear combinations of the three corners of the triangle have very good numerical quality, better the closer they are to the *BLP*– BCT** edge.
To complete the numerical study developed in this section, we find it interesting to include a measure of how good are the bases presented in Tables 1, 2 and 3 from the geometrical point of view. To this end, we have computed

p	$f(p)$	$\kappa(M_N)(p)$	$\kappa(C_N\, M_N)(p)$
$\gamma_1 = 0$, $\gamma_2 = 0.5$, $\gamma_3 = 0.5$	9.735	1093.9	6.739
$\gamma_1 = 0.5$, $\gamma_2 = 0$, $\gamma_3 = 0.5$	16.344	2727.7	15.633
$\gamma_1 = 0.5$, $\gamma_2 = 0.5$, $\gamma_3 = 0$	12.79	2079.2	10.658
$\gamma_1 = 1/3$, $\gamma_2 = 1/3$, $\gamma_3 = 1/3$	12.84	1666.0	9.179
$\gamma_1 = 0.2$, $\gamma_2 = 0.2$, $\gamma_3 = 0.6$	12.0	1613.7	8.987
$\gamma_1 = 0.2$, $\gamma_2 = 0.6$, $\gamma_3 = 0.2$	10.38	1317.9	7.814
$\gamma_1 = 0.6$, $\gamma_2 = 0.2$, $\gamma_3 = 0.2$	16.8	2823.7	14.173

Table 7: A discrete selection of bases p in the triangle defined by *BShi**, *BLP** and *BCT**, obtained following (17), and the associated values $f(p)$, $\kappa(M_N)(p)$, and $\kappa(C_N\, M_N)(p)$.

the area $\mathcal{A}$ of the control triangles (10) associated to each one of the values of p reported in the aforementioned tables (recall that vector $p \in \mathbb{R}^9$ in this section is $(\alpha_1, \beta_1, \gamma_1, \alpha_2, \beta_2, \gamma_2, \alpha_3, \beta_3, \gamma_3)$ in (10)). It is well-known ([8] or [9]) that the smaller the area of the control triangles of a partition of unity basis, the better the geometrical properties of the bases. As desired, the results provided in Table 8 show that $\mathcal{A}$ decreases as $\kappa(C_N\, M_N)$ does.

p	$\dfrac{\kappa(C_N\, M_N)(p)}{\kappa(M_N)(p)}$	$\mathcal{A}$
$\gamma_1 = 1$, $\gamma_2 = 0$, $\gamma_3 = 0$ (*BShi**(ε) in Table 6)	3.89×10^{-3}	0.013
$\gamma_1 = 0.6$, $\gamma_2 = 0.2$, $\gamma_3 = 0.2$ in Table 7	5.02×10^{-3}	0.020
$\gamma_1 = 0.5$, $\gamma_2 = 0.5$, $\gamma_3 = 0$ in Table 7	5.12×10^{-3}	0.024
$\gamma_1 = 1/3$, $\gamma_2 = 1/3$, $\gamma_3 = 1/3$ in Table 7	5.51×10^{-3}	0.028
$\gamma_1 = 0.2$, $\gamma_2 = 0.2$, $\gamma_3 = 0.6$ in Table 7	5.57×10^{-3}	0.030
$\gamma_1 = 0.5$, $\gamma_2 = 0$, $\gamma_3 = 0.5$ in Table 7	5.73×10^{-3}	0.040
$\gamma_1 = 0.2$, $\gamma_2 = 0.6$, $\gamma_3 = 0.2$ in Table 7	5.93×10^{-3}	0.044
$\gamma_1 = 0$, $\gamma_2 = 0$, $\gamma_3 = 1$ (*BCT**(φ) in Table 6)	6.05×10^{-3}	0.044
$\gamma_1 = 0$, $\gamma_2 = 0.5$, $\gamma_3 = 0.5$ in Table 7	6.16×10^{-3}	0.065
$\gamma_1 = 0$, $\gamma_2 = 1$, $\gamma_3 = 0$ (*BLP**(ε) in Table 6)	6.85×10^{-3}	0.079

Table 8: Area of the control triangles vs. $\frac{\kappa(C_N\, M_N)(p)}{\kappa(M_N)(p)}$ for partition of unity bases in Tables 6 and 7.

5. Final remarks

For solving the linear systems related to the fitting surface problem on Powell-Sabin triangulations, the bases *BShi*, *BLP* and *BCT*, are known to

have excellent geometric properties. In this work, by using the preconditioned conjugate gradient (PCG) method for solving those large-scale linear systems, we have found that these three bases also have very good numerical properties. To accomplish this task, we focused on the use of two well-known preconditioning strategies: ichol(0) and ilu(0), noting that both have shown very good performance. Indeed, the reported results (Section 3) show how consistently the eigenvalues are clustered around the identity, which is the most important feature to evaluate the effect of a preconditioning technique. We have also noticed that in some cases the preferred ichol(0) fails for which ilu(0) is used performing in a similar attractive way.

Motivated by these positive results associated to the three known bases, we decided to analyze whether any basis that has good geometric properties (in the sense that it constitutes a partition of unity) also has good numerical properties in the context of using the PCG method for solving the linear systems. Furthermore, we were interested in studying whether geometric properties are a necessary condition for a basis to have good numerical properties. In other words, we wanted to answer the question: Can there be bases with good numerical properties that do not satisfy the traditional geometric requirements?

To explore this line of thought we have introduced an inverse approach (Section 4), which requires to solve a linearly equality constrained optimization problem. To solve the optimization problems, we use a suitable well-known low-cost method. The obtained numerical results provided us with some interesting evidence. First, with very few exceptions, the solutions obtained from a random exploration of all possible options are numerically poor, regardless of the geometric quality of the basis. Second, as expected, the bases $BShi$, BLP and BCT are solutions to the considered optimization problem, showing that they in fact have both good geometric and good numerical quality. As a by-product, we detected a couple of trustworthy merit functions that can be used to evaluate the numerical quality of any potential basis. Based on these merit functions, we discovered that the triangle defined by the three known bases (as vertices of the triangle) offers the possibility of obtaining infinitely many bases with good numerical quality and good geometric properties. By exploring this triangle, we have displayed a discrete selection of these specialized new bases.

In terms of practical implications of this paper, the background in the field of fitting surfaces -and its applications- reveals that using the classical Hermite basis or, more generally, another spline bases not being partition of the unity, leads to very ill-conditioned linear systems. This fact becomes a drawback in several cases, for example when the triangulations to be con-

sidered must be particularly fine because the dataset is very large, or simply because the surface must be very close to the dataset. Also when non-uniform triangulations are recommended as a consequence of the sparsity of the dataset, because handling non-uniform triangulations usually leads to acute triangles showing a very bad behavior from the numerical point of view. We think that it may be appropriate to undertake further experimentations in problems somehow related to fitting surface by using basis that present both good numerical and geometrical properties. More precisely, we find it interesting to consider such a bases in the two specific problems previously described.

Acknowledgements

We would like to thank two anonymous reviewers for their constructive comments and suggestions.

Funding

The second author was financially supported by the Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) under the scope of the projects
UIDB/MAT/00297/2020 (doi.org/10.54499/UIDB/00297/2020) and
UIDP/MAT/00297/2020 (doi.org/10.54499/UIDP/00297/2020) (Center for Mathematics and Applications).

[1] D. Barrera, M.A. Fortes, P. González and M. Pasadas, *Minimal energy surfaces on Powell-Sabin type triangulations*, Appl. Numer. Math., 58(5) (2008), 635–645.

[2] E.G. Birgin, J.M. Martínez, and M. Raydan, *Inexact Spectral Projected Gradient methods on convex sets*, IMA J. Numer. 23 (2003), 539–559.

[3] E.G. Birgin, J.M. Martínez, and M. Raydan, *Spectral Projected Gradient methods: Review and Perspectives*, J. Stat. Softw., 60(3) (2014), 1–21.

[4] W. Böhm, G. Farin and J. Kahmann, *A survey of curve and surface methods in CAGD*, Comput. Aided Geom. Des., 1 (1984), 1–60.

[5] K. Chen, *Matrix Preconditioning Techniques and Applications*, Cambridge University Press (2005).

[6] O. Davydov, G. Nürnberger and F. Zeilfelder, *Approximation order of bivariate spline interpolation for arbitrary smoothness*, J. Comput. Appl. Math., 90 (1998), 117–134.

[7] J. E. Dennis and R. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, SIAM, Philadelphia, (1996).

[8] P. Dierckx, S. Van Leemput and T. Vermeire, *Algorithms for surface fitting using Powell-Sabin splines*, IMA J. Numer., 12 (1992), 271–299.

[9] P. Dierckx, *On calculating normalized Powell-Sabin B-splines*, Comput. Aided Geom. Des., 15 (1997), 61–78.

[10] S. Eddargani, M. J. Ibáñez, A. Lamnii, M. Lamnii and D. Barrera, *Quasi- interpolation in a space of $\mathcal{C}^2$-sextic splines over Powell-Sabin triangulations*, Mathematics, 9 (2021), 2276. https://doi.org/10.3390/math9182276.

[11] R. Escalante and M. Raydan, *Alternating Projection Methods*, SIAM, Philadelphia (2011).

[12] M. A. Fortes, M. Raydan and A. M. Sajo-Castelli, *Inverse-free recursive multiresolution algorithms for a data approximation problem*, Comput. Math. with Appl., 72 (2016), 1177–1187.

[13] M. A. Fortes, M. Raydan and A. M. Sajo-Castelli, *Relationship between quality of basis for surface approximation and the effect of applying preconditioning strategies to their resulting linear systems*, Comput. Appl. Math., vol 40, 41 (2021).

[14] G. Greiner, *Surface construction based on variational principles, in wavelets, images and surface fitting*, P. J. Laurent, A. Le Méhauté and L. L. Schumaker, Eds., Wellesley, (1994), 277–286.

[15] M. Laghchim-Lahlou and P. Sablonnière, *$\mathcal{C}^r$-finite elements of Powell-Sabin type on the three direction mesh*, Adv. in Comput. Math., 6 (1996), 191–206.

[16] A. Lamnii, M. Lamnii and H. Mraoui, *A normalized basis for condensed $\mathcal{C}^1$- Powell-Sabin-12 splines*, Comput. Aided Geom. Design, 34 (2015), 5–20.

[17] M. J. D. Powell and M. A. Sabin, *Piecewise quadratic approximations on triangles*, ACM Trans. Math. Software, 3(4) (1977), 316–325.

[18] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2^{nd} edition, SIAM (2003).

[19] P. Sablonnière, *Error bounds for Hermite interpolation by quadratic splines on an α-triangulation*, IMA J. Numer., 7 (1987), 495–508.

[20] A. M. Sajo-Castelli, M. A. Fortes and M. Raydan, *Preconditioned conjugate gradient method for finding minimal energy surfaces on Powell-Sabin triangulations*, J. Comput. Appl. Math., 268 (2014), 34–55.

[21] D. Sbibih, A. Serghini and A. Tijini, *Polar forms and quadratic spline quasi-interpolants on Powell-Sabin partitions*, Appl. Num. Math., 59(5) (2009), 938–958.

[22] X. Shi, S. Wang, W. Wang and R. H. Wang, *The $\mathcal{C}^1$-quadratic spline space on triangulations*, Report 86004, Department of Mathematics, Jilin University, Changchun (1986).

[23] H. Speleers, *A new B-spline representation for cubic splines over Powell-Sabin triangulations*, Comput. Aided Geom. Design, 37 (2015), 42–56.