

SOME OBSERVATIONS ON LOCAL LEAST SQUARES

R. H. BARTELS¹, G. H. GOLUB², and F. F. SAMAVATI³ *

¹*School of Computer Science, University of Waterloo, Waterloo, Ontario
N2L 3G1, Canada. email: rhbartel@uwaterloo.ca*

²*Department of Computer Science, Stanford University, Stanford, California
94305, U.S.A. email: golub@stanford.edu*

³*Department of Computer Science, University of Calgary, Calgary, Alberta
T2N 1N4, Canada. email: samavati@cpsc.ucalgary.ca*

Abstract.

In previous work we introduced a construction to produce biorthogonal multiresolutions from given subdivisions. The approach involved estimating the solution to a least squares problem by means of a number of smaller least squares approximations on local portions of the data. In this work we use a result by Dahlquist, et. al. on the *method of averages* to make observational comparisons between this local least squares estimation and full least squares approximation. We have explored examples in two problem domains: data reduction and data approximation. We observe that, particularly for design matrices with a repetitive pattern of column entries, the least squares solution is often well estimated by local least squares, that the estimation rapidly improves with the size of the local least squares problems, and that the quality of the estimate is largely independent of the size of the full problem.

AMS subject classification (2000): 93E24.

Key words: subdivision, refinement, interpolation, approximation, least squares, projection, subspace angles, left inverse, B-splines.

Dedication: To the memory of Germund Dahlquist.

1 Introduction

The material developed in [2, 9] proposed a local, least-squares construction for processing geometric data into a multiresolution form. A number of linear algebraic issues arise in that construction, some of which have a wider implication.

The construction, briefly, begins with an overdetermined ($m \times n$) system of equations

$$(1.1) \quad \mathbf{Pz} \approx \mathbf{f}$$

and has the goal of producing associated matrices \mathbf{A} , \mathbf{B} , and \mathbf{C} such that

$$(1.2) \quad \mathbf{Af} \rightarrow \mathbf{g}$$

represents a solution comparable to the ordinary least squares solution \mathbf{z} of (1.1), and such that

$$(1.3) \quad \mathbf{f} \equiv \mathbf{Pg} + \mathbf{Cd},$$

*This work was carried out under funding by the Natural Sciences and Engineering Research Council of Canada and by the National Science Foundation of the U. S.

where

$$(1.4) \quad \mathbf{B}\mathbf{f} \rightarrow \mathbf{d}.$$

In this regard $\begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix}$ and $[\mathbf{P} \ \mathbf{C}]$ are inverses.

The construction was designed for banded matrices \mathbf{P} of full column rank. Our experience has been gained up to this point with matrices defining regular subdivision; the refinement of B-splines by inserting knots uniformly into uniform knot sequences offers one example. In this context the matrices are “ σ -shifted,” meaning that each column has the same entries as its predecessor (perhaps save for a few initial and final “boundary columns”), but these entries are displaced at least ($\sigma > 1$) rows downward from the predecessor. We expand on that experience in this paper by including B-spline approximation problems. As well as including approximation on regular meshes, we also consider approximation and refinement with varying amounts of irregularity.

The construction consists of two parts; the first part builds the matrix \mathbf{A} , and the second part builds the matrices \mathbf{B} and \mathbf{C} together. The approximate solution vector \mathbf{g} , and the vector \mathbf{d} by which the residuals are represented, are simple byproducts of matrix multiplication. It is the construction of \mathbf{A} that will concern us here.

In Section 2 we give more details on the formation of \mathbf{A} . In Section 3 we recall the material due to Dahlquist, et. al., on subspace angles, which was used to study the method of averages ([1] and [3]). We establish that this material applies to \mathbf{A} , which gives us a means to compare the resulting vector \mathbf{g} of (1.2) with the ordinary least squares solution vector \mathbf{z} of (1.1). The power of the Dahlquist approach is that it provides an exact statement of the worst possible ratio $\frac{\|\mathbf{f}-\mathbf{P}\mathbf{z}\|}{\|\mathbf{f}-\mathbf{P}\mathbf{g}\|}$ for all possible data vectors \mathbf{f} .

Section 4 will consider some examples of \mathbf{A} for some choices of \mathbf{P} . Section 5 will deliver some observations, and indicate what quality of solutions \mathbf{g} might be expected in what class of situations. Section 6 will conclude with a mention of how the construction of \mathbf{A} can be realized in the context of a 2D surface mesh and provide a concrete example of local least squares used successfully on a large data set.

2 Formation of \mathbf{A}

Briefly, in the construction each row of \mathbf{A} is produced by selecting the next column in sequence from \mathbf{P} , sectioning out a block of rows containing some or all of the nonzero entries of that column so that an overdetermined submatrix of all the nontrivial subcolumns of \mathbf{P} contained in that block of rows is formed, and determining the nonzero row elements of \mathbf{A} from a least squares problem based on that submatrix. The motivation is simple: the j^{th} column of \mathbf{P} represents the j^{th} unknown’s contribution to the right hand side,

$$\mathbf{f} = \cdots + z_j \mathbf{p}_j + \cdots.$$

If \mathbf{P} is banded, this contribution is restricted to a portion of \mathbf{f}

$$\begin{bmatrix} \vdots \\ f_{i_a} \\ \vdots \\ f_{i_b} \\ \vdots \end{bmatrix} = \cdots + z_j \begin{bmatrix} \mathbf{0} \\ p_{i_a,j} \\ \vdots \\ p_{i_b,j} \\ \mathbf{0} \end{bmatrix} + \cdots.$$

We would hope to get a good estimate of z_j from a least squares subproblem localized upon the nonzero portion of column \mathbf{p}_j . And, if the matrix \mathbf{P} is σ -shifted, any selection of the columns

having nonzeros in the rows containing the nonzeros of \mathbf{p}_j will tend to yield overdetermined submatrices. As such, the construction of \mathbf{A} is designed to form an approximate solution of (1.1) by generating each component of the approximation through a local least squares estimate.

We view the solution of (1.1) as requiring the estimation of each z_j in \mathbf{z} individually. Consider rows indexed i_a through $i_b > i_a$ of column j of \mathbf{P} (banded), chosen so that there is a nontrivial intersection with the nonzero elements of the j^{th} column, which are contained beginning with row $i_a + \lambda$ and running through row $i_b - \mu$ for integers λ and μ .

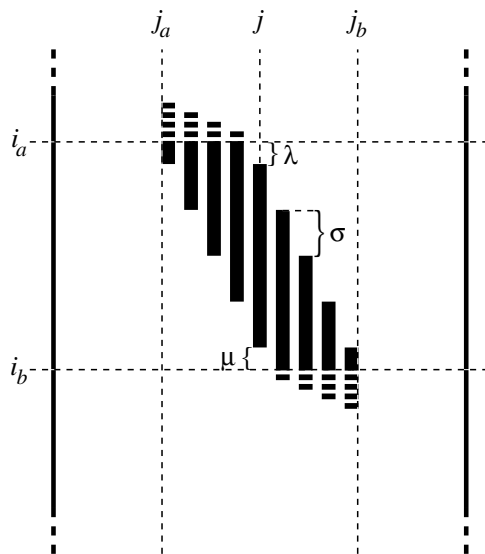


Figure 2.1: Local submatrix \mathbf{P}_j for estimating z_j

Clearly, the values of λ and μ are fixed by the location of the nonzeros in the j^{th} column and the choice of i_a and i_b . If λ and μ are both nonnegative, then all nonzero elements of the j^{th} column have been included in the selection of rows i_a through i_b . If either or both are negative, then some number of nonzero entries from column j will have been ignored in the construction.

The matrices on which we carry out our construction are to be banded. We expect them to have only zero entries in rows i_a through i_b for all columns sufficiently far to the left and right of the j^{th} , say columns with indices ℓ satisfying $|\ell - j| > K$ for a positive integer K . And we expect that the nonzeros of each successive column are located further down the matrix. Our experience has been with matrices \mathbf{P} where any column's nonzero content is shifted down σ rows from the content of its predecessor. Figure 2.1 gives an impression when λ and μ are positive.

Arriving at the situation in Figure 2.1, may require a little adjustment. Choose rows \tilde{i}_a through \tilde{i}_b as a beginning, representing the desired rows that intersect with column j of \mathbf{P} , and look at the submatrix consisting of all columns of \mathbf{P} trimmed of rows before \tilde{i}_a and after \tilde{i}_b . This submatrix will have to be narrowed by casting out columns on the left or right that have only zeros. This result may then have to be expanded by adding in successive rows before \tilde{i}_a and/or after \tilde{i}_b , and correspondingly the columns whose nonzero entries intersect with these added rows, if the result is initially underdetermined or of less than full rank. Clearly, all of \mathbf{P} constitutes a full-rank, overdetermined submatrix of \mathbf{P} containing rows \tilde{i}_a through \tilde{i}_b , and our intent is to acquire a full-rank, square or overdetermined submatrix of \mathbf{P} with the smallest number of rows containing rows \tilde{i}_a through \tilde{i}_b . After these adjustments, we are left with a (possibly) expanded range of rows i_a

through i_b and correspondingly with columns j_a through j_b . We denote the resulting submatrix by \mathbf{P}_j , and we use \mathbf{P}_j to estimate z_j by least squares.

An estimate of z_j can be extracted from

$$(2.1) \quad \left(\mathbf{P}_j^T \mathbf{P}_j \right)^{-1} \mathbf{P}_j^T \mathbf{f}_j,$$

where \mathbf{f}_j is the vector comprising f_{i_a}, \dots, f_{i_b} . (Numerically, of course, (2.1) would be obtained via an equivalent *QR* or *Choleski* factorization; the inverse is simply a notational convenience here.) This estimates $z_{j_a}, \dots, z_j, \dots, z_{j_b}$. However, all except z_j are discarded, so that we retain only

$$(2.2) \quad z_j \approx \mathbf{e}_j^T \left(\mathbf{P}_j^T \mathbf{P}_j \right)^{-1} \mathbf{P}_j^T \mathbf{f}_j,$$

where \mathbf{e}_j is a vector of the appropriate length with 1 in position j and zeros elsewhere. This is repeated for each component of the vector \mathbf{z} .

The row vector $\mathbf{a}_j = \mathbf{e}_j^T \left(\mathbf{P}_j^T \mathbf{P}_j \right)^{-1} \mathbf{P}_j^T$ will have $i_b - i_a + 1$ elements, and it can be expanded into a vector $[\mathbf{0} \ \mathbf{a}_j \ \mathbf{0}]$ of length n by prefixing zeros in positions 1 through $i_a - 1$ and suffixing zeros in positions $i_b + 1$ through n . The matrix \mathbf{A} is constructed by assembling all such expanded rows j , $j = 1, \dots, n$. $\mathbf{g} = \mathbf{A}\mathbf{f}$ is to be taken as an estimate of \mathbf{z} .

When we speak about the “width of locality” hereafter, we will mean the width of the nonzero segment \mathbf{a}_j ; that is, $i_b - i_a + 1$.

THEOREM 2.1. \mathbf{A} is a left inverse of \mathbf{P}

PROOF. The j^{th} row of \mathbf{A} has zeros initially and finally, by construction, so that it could only form nonzero products by left multiplication with the elements in rows i_a through i_b of \mathbf{P} . Within these rows, the only nontrivial elements of \mathbf{P} are to be found in columns j_a through j_b . Thus, the j^{th} row of \mathbf{A} has the same product with \mathbf{P} that the vector \mathbf{a}_j has with the submatrix \mathbf{P}_j ; namely, 1 for column j and zero for all other columns. The construction produces a row \mathbf{a}_j for each column j of \mathbf{P} . The product $\mathbf{A}\mathbf{P}$ is, as a consequence, the $n \times n$ identity. \square

3 Analysis of \mathbf{A}

Dahlquist [3] has compared the “method of averages” to the ordinary method of least squares for overdetermined linear systems. Briefly, the method of least squares solves (1.1) as

$$(3.1) \quad \left(\mathbf{P}^T \mathbf{P} \right)^{-1} \mathbf{P}^T \mathbf{f} \rightarrow \mathbf{z},$$

while the method of averages solves (1.1) as

$$(3.2) \quad \left(\mathbf{G}^T \mathbf{P} \right)^{-1} \mathbf{G}^T \mathbf{f} \rightarrow \mathbf{h}.$$

The $n \times m$ matrix \mathbf{G} is a *summation matrix*, meaning that any row of \mathbf{G}^T has the form $[0, \dots, 0, 1, \dots, 1, 0, \dots, 0]$ for some number of consecutive unit entries located at some position in the row. Only those summation matrices for which $\mathbf{G}^T \mathbf{P}$ is nonsingular are to be considered, of course. Anderson [1] has extended the class of matrices \mathbf{G} useful for problems where \mathbf{P} arises from approximating data by a Chebyshev system of functions to *rectangular smoothing matrices*. These are defined by requiring \mathbf{G} to have full rank and requiring every $n \times n$ minor of \mathbf{G} formed by deleting rows to be nonnegative.

Both papers compare \mathbf{z} , the solution of (1.1) via (3.1), and \mathbf{h} , the solution of (1.1) via (3.2), by looking at how much their corresponding residuals could differ. More precisely, they look at

$$(3.3) \quad \eta = \inf_{\mathbf{f}} \left(\frac{\|\mathbf{f} - \mathbf{P}\mathbf{z}\|^2}{\|\mathbf{f} - \mathbf{P}\mathbf{h}\|^2} \right).$$

A simple geometric argument, proceeding from the observation that $\mathbf{f} - \mathbf{Pz}$ is orthogonal to \mathcal{P} , the column space of \mathbf{P} , and that $\mathbf{f} - \mathbf{Ph}$ is orthogonal to \mathcal{G} , the column space of \mathbf{G} , establishes that

$$(3.4) \quad \frac{\|\mathbf{f} - \mathbf{Pz}\|^2}{\|\mathbf{f} - \mathbf{Ph}\|^2} = \cos^2 [\mathcal{P}, \mathcal{G}] ,$$

where $[\mathcal{P}, \mathcal{G}]$ is the final principal angle between the column spaces as ordered in the definition of [7] (Chapter 12, Section 12.4.3).

An orthonormal basis for \mathcal{P} is given by the columns of the “skinny” QR factorization of \mathbf{P} : $\mathbf{P} = \mathbf{Q}_P \mathbf{R}_P$ ([7], Chapter 5, Section 5.2.6), and likewise for \mathcal{G} . The singular values of the product matrix $\mathbf{Q}_P^T \mathbf{Q}_G$ provide the cosines of the principal angles between \mathcal{P} and \mathcal{G} ([7] Chapter 12, Section 12.4.3). The cosine in (3.4) is given by the smallest singular value.

In Theorem 1 of [3] it is established that η is given by $\cos^2 [\mathcal{P}, \mathcal{A}]$. It is straightforward to verify the restatement of that theorem for our own notation as:

THEOREM 3.1.

$$(3.5) \quad \gamma = \inf_{\mathbf{f}} \left(\frac{\|\mathbf{f} - \mathbf{Pz}\|^2}{\|\mathbf{f} - \mathbf{Pg}\|^2} \right) = \cos^2 [\mathcal{P}, \mathcal{A}] ,$$

where \mathbf{g} is the solution to (1.1) by $\mathbf{A}\mathbf{f} \rightarrow \mathbf{g}$ and \mathcal{A} is the column space of \mathbf{A}^T .

This theorem provides the worst possible result (in terms of residuals) in using \mathbf{g} as a substitute for \mathbf{z} . It does so by locating the data vector \mathbf{f} that yields the smallest possible ratio shown in (3.5). It is useful to ask what any data vector might provide as a ratio.

COROLLARY 3.2. For any \mathbf{f} there is a vector \mathbf{d} of unit Euclidian norm giving

$$(3.6) \quad \frac{\|\mathbf{f} - \mathbf{Pz}\|^2}{\|\mathbf{f} - \mathbf{Pg}\|^2} = \sum_i \sigma_i^2 \delta_i^2$$

where the δ_i are the components of \mathbf{d} , the σ_i are the singular values of $\mathbf{Q}_{P^\perp}^T \mathbf{Q}_{A^\perp}$, and the columns of \mathbf{Q}_{P^\perp} and \mathbf{Q}_{A^\perp} form orthonormal bases for the orthogonal complements of the column spaces of \mathbf{P} and \mathbf{A}^T respectively. (The full QR factorization of an $m \times n$ matrix provides such a basis directly as the final $n - m$ columns of the orthogonal factor.)

The corollary can be argued briefly as follows, with justification being found in intermediate results developed by Dahlquist, et. al. to prove the main theorem in their paper [3]. The ratio

$$(3.7) \quad \frac{\|\mathbf{f} - \mathbf{Pz}\|^2}{\|\mathbf{f} - \mathbf{Pg}\|^2}$$

over all vectors \mathbf{f} can also be expressed as the minimum of

$$\frac{\|\mathbf{Q}_{P^\perp} \mathbf{Q}_{A^\perp}^T \mathbf{c}\|^2}{\|\mathbf{Q}_{A^\perp} \mathbf{c}\|^2}$$

over all vectors \mathbf{c} . The orthogonality of the \mathbf{Q} matrices allows this to be re-expressed as

$$\frac{\|\mathbf{Q}_{P^\perp}^T \mathbf{Q}_{A^\perp} \mathbf{c}\|^2}{\|\mathbf{c}\|^2} = \frac{\|\mathbf{USV}^T \mathbf{c}\|^2}{\|\mathbf{c}\|^2} = \frac{\|\mathbf{USV}^T \mathbf{c}\|^2}{\|\mathbf{V}^T \mathbf{c}\|^2} = \left\| \mathbf{S} \frac{\mathbf{d}}{\|\mathbf{d}\|} \right\|^2$$

where the second ratio shows the singular value decomposition of $\mathbf{Q}_{P^\perp}^T \mathbf{Q}_{A^\perp}$, the third ratio makes use of the orthogonality of \mathbf{V} , and letting $\mathbf{d} = \mathbf{V}^T \mathbf{c}$ provides the last ratio. The singular values give the cosines of the subspace angles between \mathcal{P}^\perp and \mathcal{A}^\perp . The minimum ratio results by letting \mathbf{d} pick out the smallest such cosine from \mathbf{S} , but anything other than this special \mathbf{d} will produce a weighted average of the cosines with nonnegative weights. If the cosines are rich in values close

to 1; that is, if \mathcal{P}^\perp and \mathcal{A}^\perp are reasonably well aligned (equivalently, if \mathcal{P} and \mathcal{A} are reasonably well aligned), then the chances are that this combination of singular values will be higher in value for a randomly-encountered data vector \mathbf{f} than the minimum singular value. In Section 5.8 we shall return to this briefly with supporting evidence.

4 Examples

In this section we make observations comparing the effectiveness of using local least squares; that is, using \mathbf{A} in place of $(\mathbf{P}^T \mathbf{P})^{-1} \mathbf{P}^T$, on example problem classes whose structures conform in varying degrees to the assumptions underlying our construction. The classes of problems used as a basis for the observations were:

- I. B-spline based approximation,
- II. B-spline based refinement,
- III. Butterfly subdivision (a refinement process not designed from B-splines [5]),
- IV. Matrices \mathbf{P} with randomly-generated nonzero column entries.

For each problem class we explored the following issues:

- a. For “fixed locality”; that is, for fixed and consistent choice of \tilde{i}_a and \tilde{i}_b with respect to each column j , how do the subspace angles and residuals compare between the local and the ordinary least squares methods of solution as a function of the size of \mathbf{P} ?
- b. For fixed size of \mathbf{P} , how do the subspace angles and residuals compare between the local and the ordinary methods of solution as the size of locality is increased; that is, as the number of rows in the selection from \tilde{i}_a to \tilde{i}_b (the size of the submatrix \mathbf{P}_j) is increased?
- c. For each of a. and b. above, how do the subspace angles and residuals behave as varying amounts of randomness are introduced into the underlying problem that generates \mathbf{P} ?

Our underlying problems were chosen so that we would obtain examples of \mathbf{P} :

- i. that were cyclic,
- ii. that were not cyclic.

Varying amounts of randomness were obtained by:

- A. randomly adjusting the insertion or data-approximation points within the knot interval into which they fell, which provided a limited amount of randomness,
- B. placing the insertion or approximation points entirely at random within the spline domain, which placed no restriction on the randomness,
- C. creating nonzero elements in each column of \mathbf{P} entirely from random numbers.

All of i., ii., A, and B were obtained by inserting knots regularly into regular meshes for uniform B-splines (periodic or non-periodic), or by asking for data-value approximation at regular mesh points by uniform B-splines (periodic or non-periodic). The additional randomness that might be provided by using non-uniform B-splines with random knot meshes was not considered. It should also be understood that the examples of Butterfly subdivision (problems of class III.) could not be made random due to the nature of the definition of Butterfly subdivision.

In all example problems, the rows \tilde{i}_a and \tilde{i}_b have been chosen so that $\lambda = \mu \geq 0$, and the necessary adjustments that were made to arrive at i_a and i_b maintained these conditions; that is, the intent was to site the nonzeros of column j centrally with respect to the submatrix \mathbf{P}_j and to include all nonzeros of column j within \mathbf{P}_j . (This is to be understood cyclically for periodic problems, and it has to be understood as applying with respect only to sufficiently interior rows for non-periodic problems.)

4.1 Approximation Examples

Problems of class **I** were taken from data sampling on the model

$$(4.1) \quad \sum_{i=0}^L c_i B_{i,k,\mathcal{Z}}(t_j) \approx f(t_j),$$

where $B_{i,k,\mathcal{Z}}$ are the B-splines of order k with the integers \mathcal{Z} as knots [4], t_j are the evaluation points, f is the function to be approximated, and c_i are the unknowns. The i, j entries of \mathbf{P} are given by the values $B_{i,k,\mathcal{Z}}(t_j)$, which can be calculated using the de Boor recurrence [4].

4.2 Refinement Examples

The context for the problems of classes **II** and **III** is one in which a function $\phi(t)$ is approximated by a linear combination of the basis functions for some space \mathcal{V}^{k+1}

$$\phi(t) \approx \sum_{\ell} f_{\ell} \beta_{\ell}^{k+1}(t),$$

and there is a proper subspace \mathcal{V}^k on which one wishes to find an approximation to $\phi(t)$. Since any basis for \mathcal{V}^k can be represented in terms of the basis for \mathcal{V}^{k+1} ,

$$\beta_j^k = \sum_i p_{i,j} \beta_i^{k+1},$$

we know that

$$\phi(t) \approx \sum_j z_j \beta_j^k(t) = \sum_j z_j \sum_i p_{i,j} \beta_i^{k+1}(t) = \sum_i \left(\sum_j p_{i,j} z_j \right) \beta_i^{k+1}(t),$$

and from this we take

$$\sum_j p_{i,j} z_j \approx f_i$$

as defining the form of the approximation we wish. This is, of course, exactly (1.1).

For class **II** problems, the functions $\beta^k(t)$ are represented by $B_{i,k,\mathcal{Z}}(t)$, and the functions $\beta^{k+1}(t)$ are the B-splines that result from these after the insertion of additional knots. For any matrix \mathbf{P} to be generated in this class of problems, the rows of \mathbf{P} represent the *discrete B-splines* generated by the original knot mesh and the refined knot mesh. The discrete B-splines may be computed using the *Oslo algorithm*, which is a variation on the de Boor recurrence. [6]. For class **III** problems, the matrix \mathbf{P} is given explicitly as the Butterfly recurrence matrix, and the functions $\beta^k(t)$ and $\beta^{k+1}(t)$ are implicitly defined by this recurrence.

Concrete examples of \mathbf{P} and \mathbf{A} for several widths of locality may be found in [2].

4.3 Random Examples

Matrices \mathbf{P} were generated by taking a prescribed *length* of nonzeros to be found in a column and inserting that many random numbers sequentially in each column, with a downward shift from column to column determined from the number of columns and the length as follows: for an $m \times n$ matrix with length ℓ , the ℓ random entries in column j are placed starting in row $1 + \sigma(j-1)$, where $\sigma = (m-\ell)/(n-1)$. This arrangement has the effect of including nonzeros in as many rows as possible for the given size of matrix and given sequence length of nonzeros.

5 Observations

The observations here will be made only for problems of class **II** with some side remarks from the random examples of class **IV**. Concentrating the main observations on class **II** allows us to hold a common thread of presentation and provide comparability from observation to observation. However, the reader should be assured that what is being observed here for problems of class **II** was reflected in the examples for classes **I** and **III**.

For the examples that we ran, the short summary is that, as the problems became less regular, the incentive for considering this approach became weaker. *Regularity* for matrices \mathbf{P} in this sense meant that all columns had the same, or nearly the same, number of nonzero entries. Moreover among such matrices, those with shorter lengths of nonzero entries in each column and for which there tended to be a dominance in the magnitude of the nonzero elements around the central element of the nonzeros showed an advantage. With enough regularity in \mathbf{P} , the local least squares approach can be extremely effective, as we shall show evidence of below.

We shall support this summary by starting with the worst cases and proceeding to the better ones.

5.1 Unrestricted Random Knot Insertions

Figure 5.1 shows the ratios for roughly 4000 test examples generated by randomly inserting knots into a basis of uniform cubic B-splines (non-cyclic problems of class **II** and random situation as described in \mathcal{B}). Each test contributing to this figure had at least 40 rows and 10 columns, with $\tilde{i}_b - \tilde{i}_a + 1 \geq 11$.

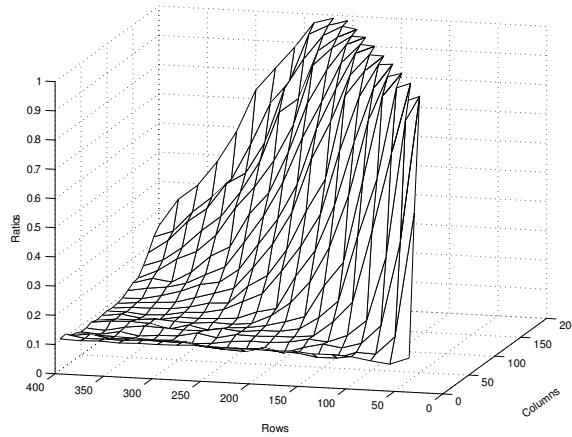


Figure 5.1: Average γ vs. problem size for unrestricted random knot insertion

The matrices for Figure 5.1 had the least structure associated with any of the figures we shall give. The number of nonzero elements could vary greatly from column to column. Figure 5.2 shows the *average* number of nonzeros in the columns in the matrices generated for Figure 5.1; nothing prevented the neighbor of a given column from having a significantly different number of nonzeros. As can be seen from Figure 5.1, the highest values of γ appear only out in the right rear area of the surface plot. This was the area in which

- (a) the matrices were closest to square,

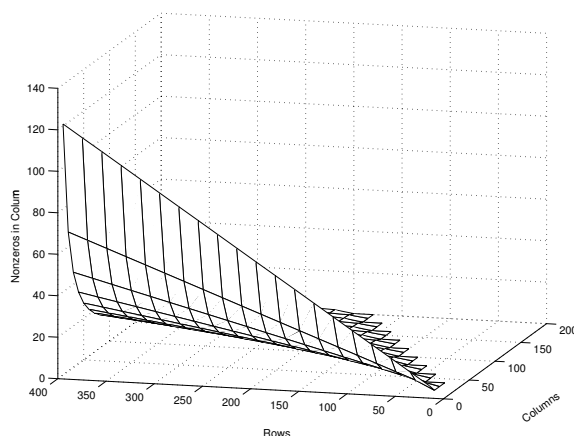


Figure 5.2: Average nonzeros in columns for the matrices of Figure 5.1

- (b) the matrices had the fewest average number of nonzeros per column,
- (c) the numbers of nonzeros in each column tended to be nearly the same.

In following up on each of these, (b) and (c) turned out to be the important properties. Property (a) was simply an artifact of the process of generation that caused columns of short sequences of nonzero entries of nearly equal lengths to appear in nearly square matrices.

5.2 Completely Random Column Entries

In the example of this section and all subsequent sections we generated each matrix \mathbf{P} with essentially all columns having the same number of nonzero entries. The only exception was the occasional need to have a very small number of initial and final columns with fewer nonzero entries to accommodate the column shift σ . Figure 5.3 shows the results of choosing a length ℓ of nonzero column entries, fixing the width of locality $\tilde{i}_b - \tilde{i}_a - 1$ to be ℓ , and using a pseudo-random uniform number generator on $[0, 1]$ to generate all nonzero entries.

The shift σ of the nonzero locations from one column to the next was determined from the total number of rows in \mathbf{P} to ensure that no rows had purely zero entries. Figure 5.3 shows the results of averaging 30 such versions of \mathbf{P} for each choice of ℓ . It presents the purely random setting of class **IV** with the only column structure being that of a common length of nonzero entries. Figure 5.3 is typical of what results from this structure; the values of γ are low, yet they tend to be insensitive to the number of nonzeros in a column. However, one must be aware that for each increase in ℓ there is a corresponding increase in the width of locality, so the constancy of γ indicated by Figure 5.3 represents a combined effect. It is not possible to increase ℓ without usually being forced to increase the width of locality, too. However, with fixed ℓ one can investigate how γ improves as the width of locality is increased, and this will be shown in Section 5.7.

5.3 Random Column Entries with Imposed Structure

The results become better if the random numbers are subjected to some bias so that those larger in magnitude cluster in a common location for each column. Figure 5.4 shows exactly the same situation as Figure 5.3 except that the random numbers entered into each column were multiplied by a bell-shaped weight function designed to give the numbers in the center of each sequence of

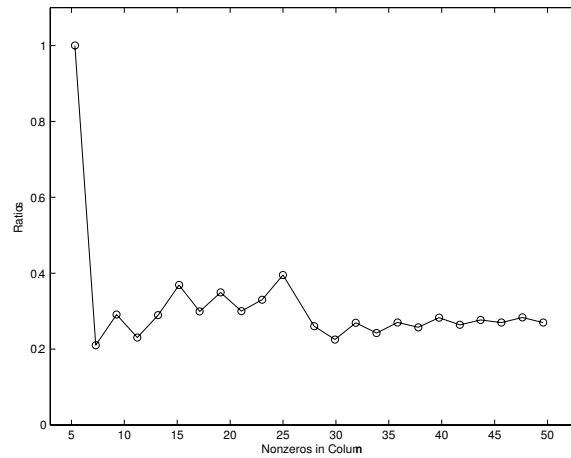


Figure 5.3: Average γ vs. column density for uniform random column entries in $320 \times 60 \mathbf{P}$

nonzero column entries a probability of having a larger magnitude than those at the beginning and end of the sequence.

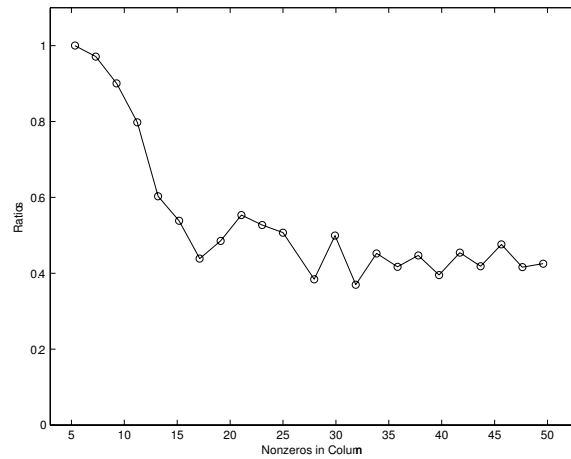


Figure 5.4: Average γ vs. column density for weighted random column entries in $320 \times 60 \mathbf{P}$

5.4 Restricted Random Knot Insertions in Non-Periodic B-Splines

The examples of Sections 5.3 and 5.2 combine matrices \mathbf{P} with various choices of ℓ , the number of nonzero entries in a column. The examples in this and all subsequent sections fix the value of ℓ and consider either the size of \mathbf{P} (this section and Sections 5.5 and 5.6) or the width of locality (Section 5.7).

Figure 5.5 shows a situation that is closer to one arising in practice and one that does not have a structure of such artificiality. Matrices \mathbf{P} of increasing sizes were generated corresponding to bases of uniform cubic B-splines into which a single knot was inserted somewhere randomly within each knot interval. This produces examples of class $\mathbf{II}/\mathbf{ii}/\mathcal{A}$. The analog for class \mathbf{I} would be the setting in which uniform B-splines are used to approximate data sampled at one argument value somewhere in each knot interval, and in that case the results look very similar.

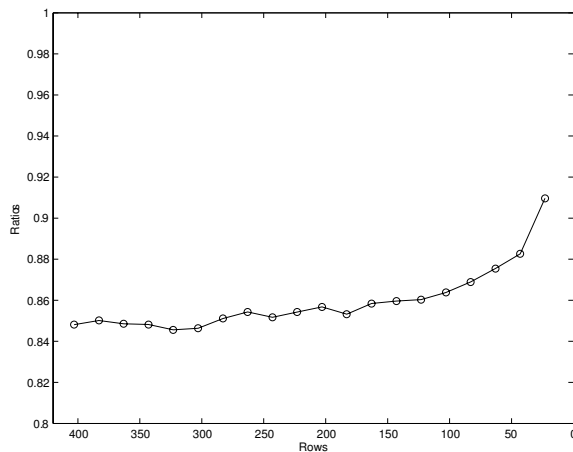


Figure 5.5: Ratios vs. size of \mathbf{P} , one knot randomly inserted per interval, locality width 11

It should be noted that these results are presented for a single width of locality (11) and that the number of nonzeros in any column was 5. The structure of these problems was such that $\tilde{i}_a = i_a$ and $\tilde{i}_b = i_b$; that is, no adjustment of locality was ever needed. Only the number of rows is shown in the figure; the nature of the situation requires the number of rows to be twice the number of columns (single knot insertions have the effect of doubling the dimension of the spline space). The results are much better than those of Section 5.2 and Section 5.3, achieving a ratio γ above 0.8. It should also be noticed that these results are largely independent of the size of the problem, a property that becomes even more pronounced with the examples to follow.

5.5 Midpoint Knot Insertions in Non-Periodic B-Splines

Figure 5.6 was generated in the same way as Figure 5.5 with the difference that the single knots inserted were exactly at the midpoint of each knot interval. This would correspond for problems of class \mathbf{I} to the situation in which data samples are taken on a regular grid to be approximated by uniform splines. In Figure 5.6 the quality of the local approximation varies only slightly for small problems and rapidly becomes constant no matter how large the matrix \mathbf{P} becomes past a certain size.

In Figure 5.6 the slight variation in γ observed for the smaller \mathbf{P} is due to the few initial and final columns that are different from the interior columns, all of which are simply shifted versions of a generic column. As the ratio of generic columns to special columns increases, the relative effect of the special columns becomes negligible.

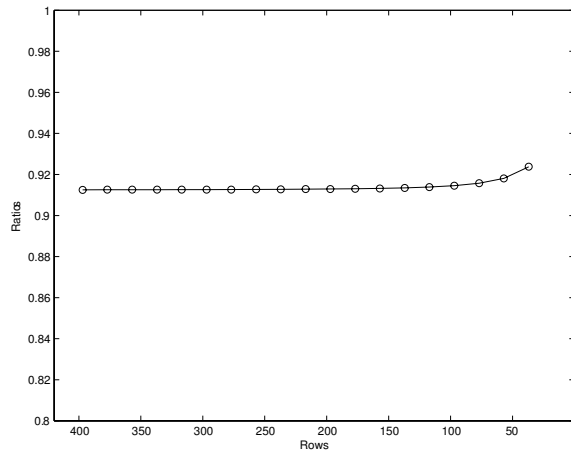


Figure 5.6: Ratios vs. size of \mathbf{P} , midpoint knot insertion, noncyclic case, width 11

5.6 Midpoint Knot Insertions in Periodic B-Splines

If we move to cyclic matrices; that is, trade problems of class **ii** for those of **i** in which all columns are shifted versions of a generic column, then there is no variation of γ with problem size! This is the best of all situations, and it is illustrated by Figure 5.7. It is simply the situation of Figure 5.6 except that the B-spline basis has now been chosen to be periodic, resulting in a matrix \mathbf{P} that is cyclic: all columns are identical save for the shift of σ . While this figure was generated by a problem of class **II**, it would correspond to a class **I** problem of approximating data on a regular grid in a periodic setting. It also corresponds to all Butterfly subdivision examples (type **III**).

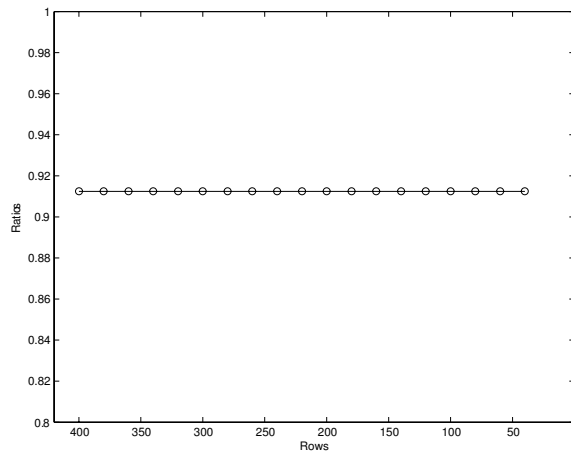


Figure 5.7: Ratios vs. size of \mathbf{P} , midpoint knot insertion, cyclic case, width 11

5.7 *Improvement with Wider Locality*

In Sections 5.4 through 5.6 the width of locality was fixed at 11 and the the number of nonzeros in any column was 5 (or less in the few necessarily exceptional columns at the extreme right and left of \mathbf{P}). In Sections 5.2 and 5.3 the width of locality varied, as did the number of nonzeros in each column. Figure 5.8 displays matrices from Sections 5.2 through 5.5, made comparable by taking a common matrix size and freezing the number of nonzeros at exactly 5 per column in the cases of the random matrices of Sections 5.2 and 5.3. Widths shown in Figure 5.8 varied from 0.02 to 0.1 times the number of rows in \mathbf{P} . This figure shows the rapid improvement in γ as the width of locality is increased and is typical of this behavior. The results for the cyclic matrices \mathbf{P} of Section 5.6 were visually indistinguishable from those of Section 5.5 and are not shown.

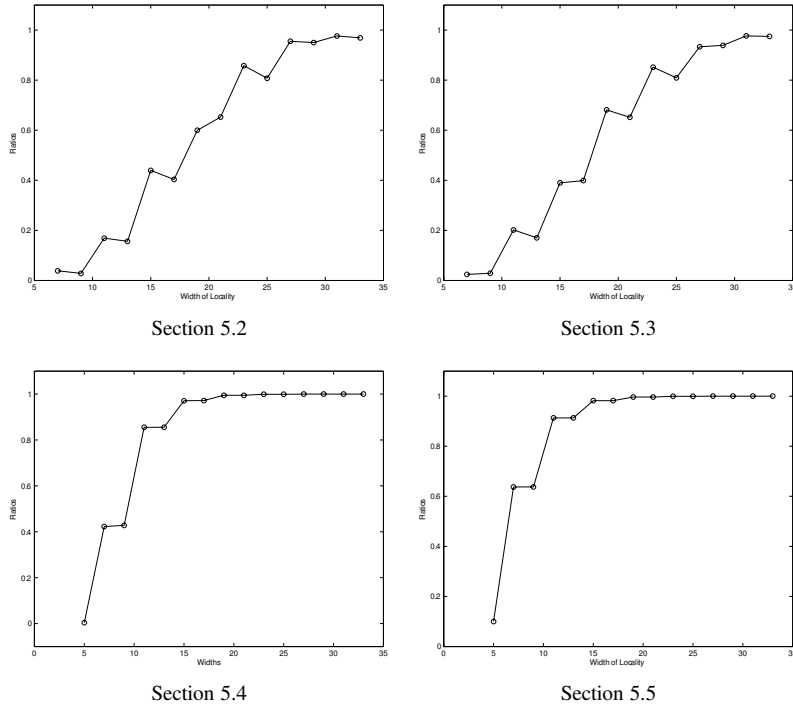


Figure 5.8: γ vs. locality for types of matrices discussed in previous sections

5.8 *Better Than Worst Case Results*

The ratio with which the theory has provided us is the worst possible ratio. Corollary 3.2 suggested that, for actual matrices \mathbf{P} and \mathbf{A} and actual right hand sides \mathbf{f} the ratio can be better. Figure 5.9 shows some measured ratios comparing actual approximate solutions \mathbf{z} (least squares) and \mathbf{g} (local least squares) in the case that \mathbf{P} is the matrix arising from inserting midpoint knots in cyclic cubic splines and \mathbf{A} is the local least squares left inverse having 7 nonzero elements per row. The ratios correspond to generated vectors \mathbf{f} with components taken from the uniform-[0,1] random-number generator provided in Matlab. The minimum ratio given by the theorem is 0.63687. Note that all the ratios in Figure 5.9 are above 0.85. Figure 5.10 shows the subspace

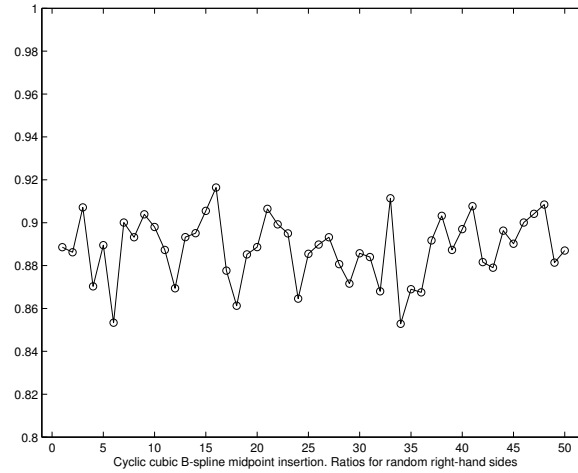


Figure 5.9: Ratios for random right-hand sides, midpoint knot insertion, cyclic case, width 7

angle cosines for the matrices \mathcal{P} and \mathcal{A} used to provide the ratios in Figure 5.9.

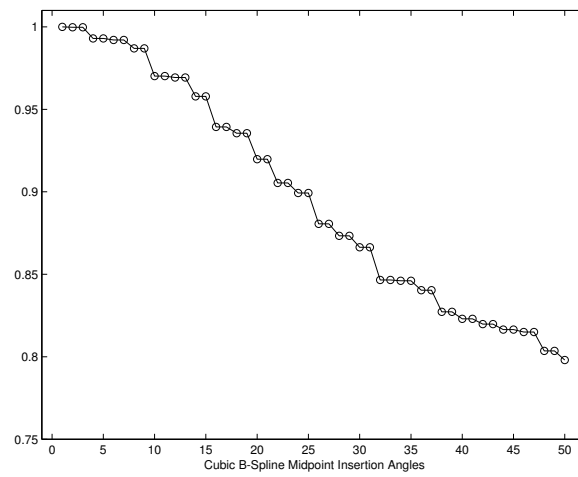


Figure 5.10: Cosines for \mathcal{P} and \mathcal{A} of Figure 5.9

In section 6 we use versions of \mathcal{P} and \mathcal{A} derived from a mesh-based subdivision due to Litke *et al* [8]. Figure 5.11 shows an example of the cosine distribution, which is particularly rich in values close to 1.0, indicating that we should expect approximation residuals for \mathbf{g} to be reasonably close to those for \mathbf{z} .

5.9 Some Conclusions

To comment on the best results we have seen, note that the least squares solution to approximation/refinement problems on regular grids, which are very common in practice, can be reproduced

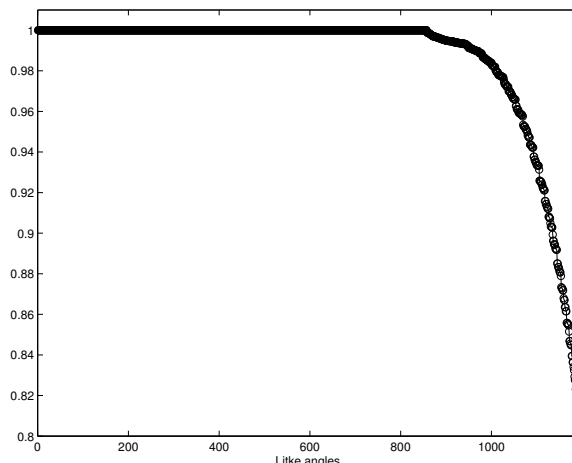


Figure 5.11: Cosines for \mathcal{P} and \mathcal{A} for Litke subdivision

with exceptional fidelity at the cost of only a small number of small local least squares problems to set the row elements of \mathbf{A} . The size of the given problem does not influence the size required of the local problems. As a concrete example, using submatrices \mathbf{P}_j with only 19 rows one can achieve essentially the correct least squares solution *no matter what the size of \mathbf{P}* in the problems of Sections 5.4 through 5.6. In the case of the cyclic problems of Section 5.6, solving only a *single problem* with a single, generic \mathbf{P}_j achieves essentially the correct least squares solution, and in the case of the problems of Section 5.5, only on the order of 3 local problems need to be solved, one for a generic \mathbf{P}_j representing interior columns and 2 more versions of \mathbf{P}_j to handle the special columns on the extreme right and left of the matrix.

6 Example: Refinement of Geographic Data on a Hexagonal Grid

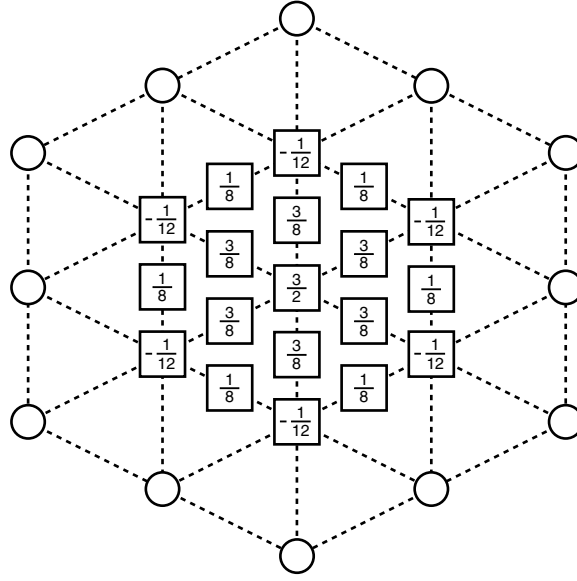
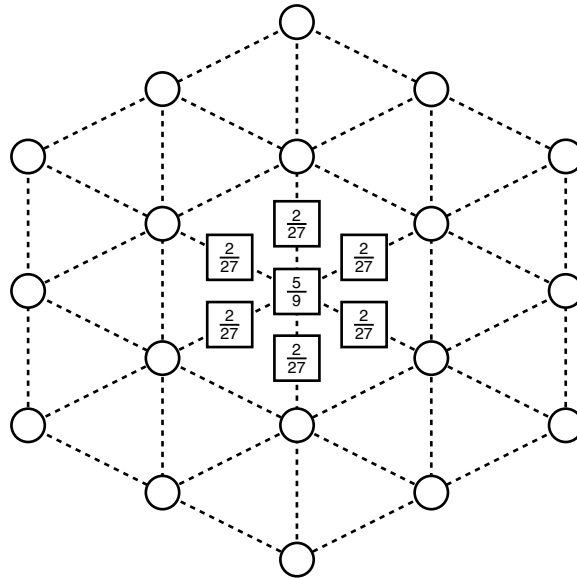
The local least squares approach has been extended in the case of the problems described in Section 4.2 to the subdivision of data on hexagonal grids; details may be found in [9].

The columns of the matrix \mathbf{P} may be represented graphically, and for a subdivision due to Litke *et. al* [8] they are shown in Figure 6.1. This figure shows the relevant nonzero elements of a single column of \mathbf{P} written on the nodes of the mesh representing \mathbf{f} to which these elements contribute in the product $\mathbf{P}\mathbf{z} \approx \mathbf{f}$. The nodes of the mesh associated with \mathbf{z} are shown in the background.

The rows of the matrix \mathbf{A} have a similar representation, and the most local \mathbf{A} that can be constructed from the \mathbf{P} of Figure 6.1 is shown in Figure 6.2. This figure shows the relevant nonzero elements of a single row of \mathbf{A} from which the result $\mathbf{z} \approx \mathbf{g} = \mathbf{A}\mathbf{f}$ is formed. For this version of \mathbf{A} , the ratio γ of Theorem 3.1 is ≈ 0.669

Figure 6.3 shows U. S. Geological Survey data of a portion of the Grand Canyon displayed on a hexagonal grid. This portion contains 131072 points. Since the points on Figure 6.3 are so densely packed, we display only a portion of the mesh near the lower left, as shown in Figure 6.4.

The subdivision due to Litke, *et. al.*, is geometry preserving; that is, the rows of \mathbf{P} represent affine transformations. The local least squares construction has been shown to produce matrices \mathbf{A} that are also geometry preserving when \mathbf{P} is so [2]. Thus, it makes sense to display the elements of $\mathbf{g} = \mathbf{A}\mathbf{f}$ on a (coarser) hexagonal grid. The subdivision is such that the grid of \mathbf{g} has $1/3$ as many points as the grid of \mathbf{f} , and the portion corresponding to Figure 6.4 is shown in Figure 6.5.

Figure 6.1: Litke *et. al* subdivision stencil (a column of \mathbf{P})Figure 6.2: A minimal-width local inverse (a row of \mathbf{A})

A second application of \mathbf{A} yields Figure 6.6, just to show that this data reduction can be carried out more than once, and that the general geometry, though simplified each time by losing finer details, is still preserved. Finally, applying \mathbf{P} to the points of $\mathbf{g} = \mathbf{A}\mathbf{f}$ yields \mathbf{f} again with residuals.



Figure 6.3: Elevation data on a hexagonal grid from a section of the Grand Canyon

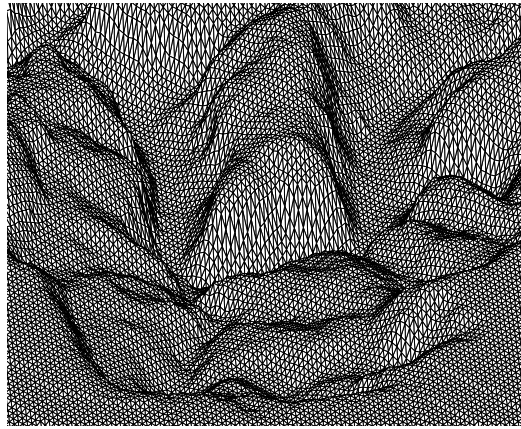
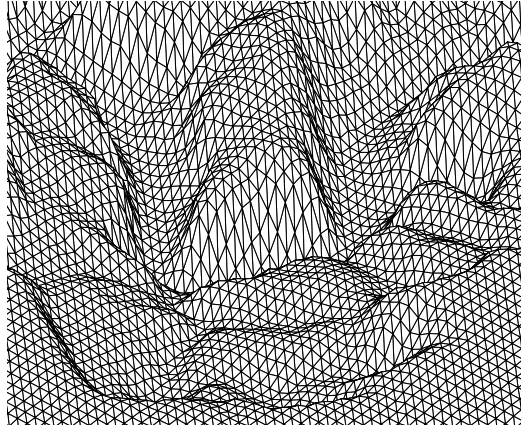
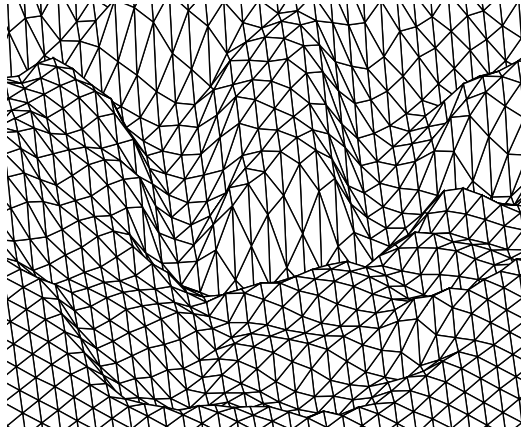


Figure 6.4: A portion of the Grand Canyon data shown in Figure 6.3

Figure 6.7 shows $\mathbf{Pg} \approx \mathbf{f}$, and this figure can be compared with Figure 6.4.

It should be remarked that the numbers in the data set lie between 0 and 12.5, that the residual Euclidian norm is $\|\mathbf{f} - \mathbf{Pg}\| \approx 9.0$, and the maximum component error in \mathbf{Pg} is ≈ 0.52 , which compares favorably with the least squares residual ≈ 6.2 and a maximum component error of ≈ 0.43 . More to the point, producing \mathbf{g} requires 6.12×10^5 flops. To produce the least squares approximation \mathbf{z} requires dealing with a matrix \mathbf{P} having 131072 rows, 43691 columns, and 830129 nonzero entries; that is, a highly sparse nonzero density of 1.4496×10^{-04} , in which the nonzeros are not naturally arranged in a banded way. In directly solving the least squares problem using orthogonal transformations, fill-in is inevitable. Each element to be eliminated in a column will share its row with 3 other elements (in the case of 2/3 of the elements of a column) or with 6 other elements (in the case of 1/3 of the elements of a column). There will be 19 elements in each column, and each of the shared elements of those from the 19 that must be eliminated will typically introduce a nonzero in the “pivotal row” of the elimination. In the procedure we used for checking the least squares solution, the original 830129 entries expanded to over 1.07×10^9

Figure 6.5: Grand Canyon data after one application of \mathbf{A} Figure 6.6: Grand Canyon data after a second application of \mathbf{A}

nonzero entries in the 131072×43691 orthogonal factor. Since each of these entries needed to be numerically “touched” at least once, the least squares solution cost $\mathcal{O}(10^9)$ flops vs. the $\mathcal{O}(10^5)$ needed for local least squares. Admittedly, the procedure did not make an effort to band reduce the $m \times n$ matrix \mathbf{P} , so this may overestimate what would be optimal for least squares, but the local least squares we have chosen to use can be expected to be cheaper in any event, since it amounts to no more than n inner products involving short, 7-element subvectors from \mathbf{f} .

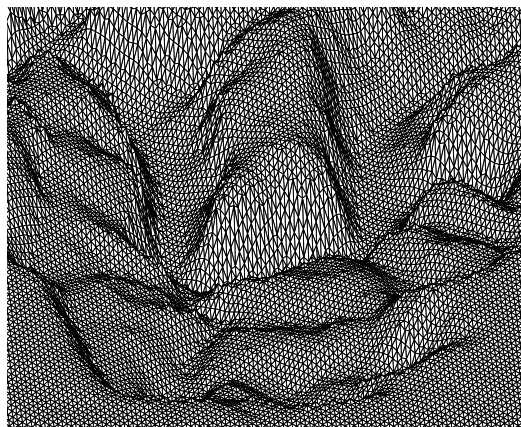


Figure 6.7: $\mathbf{P}\mathbf{g} \approx \mathbf{f}$ (compare with Figure 6.4)

REFERENCES

1. N. ANDERSON, *A generalization of the method of averages for overdetermined linear systems*, *Math. Comp.*, 29 (1975), pp. 607–614.
2. R. H. BARTELS AND F. F. SAMAVATI, *Reversing subdivision rules: Local linear conditions and observations on inner products*, *J. Comput. Appl. Math.*, 119 (2000), pp. 29–67.
3. G. DAHLQUIST, B. SJÖBERG, AND P. SVENSSON, *Comparison of the method of averages with the method of least squares*, *Math. Comp.*, 22 (1968), pp. 833–845.
4. C. DE BOOR, *A Practical Guide to Splines*, Applied Mathematical Sciences 27, Springer-Verlag, 1978.
5. N. DYN, D. LEVIN, AND J. GREGORY, *A 4-point interpolatory subdivision scheme for curve design*, *Comput. Aided Geom. Design*, 4 (1988), pp. 257–268.
6. R. N. GOLDMAN AND T. LYCHE, eds., *Knot Insertion and Deletion Algorithms for B-Spline Curves and Surfaces*, SIAM, 1993.
7. G. H. GOLUB AND C. F. V. LOAN, *Matrix Computations*, The Johns Hopkins University Press, second ed., 1989.
8. N. LITKE, J. LEVIN, AND P. SCHRÖDER, *Trimming for subdivision surfaces*, *Comput. Aided Geom. Design*, 18 (2001), pp. 463–481.
9. F. F. SAMAVATI AND R. H. BARTELS, *Reversing subdivision using local linear conditions: Generating multiresolutions on regular triangular meshes*, tech. rep., University of Calgary, Computer Science, TR 2002-711-14, December 2002.