Smooth Interpolation of Data by Efficient Algorithms
by C. Fefferman[1]

In 1934, H. Whitney [16,17,18] posed several basic questions on smooth extension of functions. Those questions have been answered in the last few years, thanks to work of E. Bierstone, Y. Brudnyi, C. Fefferman, P. Milman, W. Pawłuski, P. Shvartsman and others. (See [2,3,4,6,7,8,12].) The solution of Whitney's problems has led to a new algorithm for interpolation of data, due to C. Fefferman and B. Klartag [9,10]. The new algorithm is theoretically best possible, but far from practical. We hope it can be modified to apply to practical problems.

In this expository paper, we briefly review Whitney's problems, then formulate carefully the problem of interpolation of data. Next, we state the main results of Fefferman-Klartag [9,10] on efficient interpolation. Finally, we present some of the ideas in the proofs.

Let us set up notation. We fix positive integers m, n throughout the paper. We work in $C^m(\mathbb{R}^n)$, the space of m times continuously differentiable functions $F : \mathbb{R}^n \to \mathbb{R}$ for which the norm

$$\|F\| = \sup_{x\in\mathbb{R}^n} \max_{|\alpha|\leq m} |\partial^\alpha F(x)|$$

is finite. (We would like to be able to treat Sobolev spaces also. Work on Sobolev interpolation of data is just beginning; see A. Israel, G. Luli, C. Fefferman and P. Shvartsman, to appear.)

Let $F \in C^m(\mathbb{R}^n)$ and $x \in \mathbb{R}^n$ be given. We write $J_x(F)$ (the "jet"of F at x) to denote the m^{th} order Taylor polynomial of F at x:

$$[J_x(F)](y) = \sum_{|\alpha|\leq m} \frac{1}{\alpha!}(\partial^\alpha F(x)) \cdot (y-x)^\alpha.$$

Thus, $J_x(F)$ belongs to $\mathcal{P}$, the vector space of all real-valued polynomials of degree at most m on $\mathbb{R}^n$. The jet $J_x(F)$ encodes the values at x of F and its derivatives through order m.

[1]Supported by NSF Grant No. DMS-09-01-040 and ONR Grant No. N00014-08-1-0678

Whitney's classic problems are as follows. Suppose we are given a subset $E \subset \mathbb{R}^n$ and a function $f : E \to \mathbb{R}$. We make no assumptions on the set E.

Question 1: How can we tell whether f extends to a C^m function on $\mathbb{R}^n$? That is, how can we tell whether there exists $F \in C^m(\mathbb{R}^n)$ such that $F = f$ on E?

If such an F exists, then we ask:

Question 2: How small can we take $\|F\|$?

Question 3: What can we say about $J_x(F)$ for a given point x lying in or near E?

Question 4: Can we take F to depend linearly on f?

Whitney himself settled these questions in the one-dimensional case ($n = 1$). He also proved the Whitney extension theorem, answering an easier version of Questions $1 \cdots 4$. Here is

Theorem 1. ***Whitney's Extension Theorem [16]:***
Let $E \subset \mathbb{R}^n$ be a closed set, and let $(P^x)_{x\in E}$ be a family of polynomials $P^x \in \mathcal{P}$, indexed by the points of E.

Then the following are equivalent.

(A) There exists $F \in C^m(\mathbb{R}^n)$ such that $J_x(F) = P^x$ for each $x \in E$.

(B) There exists a real number $M > 0$ such that:

- $|\partial^\alpha P^x(x)| \le M$ *for* $|\alpha| \le m, \quad x \in E$;
- $|\partial^\alpha (P^x - P^y)(y)| \le M|x-y|^{m-|\alpha|}$ *for* $|\alpha| \le m-1, \quad x, y \in E$; *and*
- $|\partial^\alpha (P^x - P^y)(y)|/|x-y|^{m-|\alpha|}$ *tends to zero as* $|x-y| \to 0, \quad |\alpha| \le m, \quad x, y \in E$.

Moreover, the least possible M in (B) is comparable to the least possible $(\inf)\|F\|$ *in (A). That is, $cM \le \inf \|F\| \le CM$, for constants c, C depending only on m, n.*

Given a family of jets $(P^x)_{x\in E}$ satisfying *(B)*, Whitney's proof exhibits an $F \in C^m(\mathbb{R}^n)$ as in (A) such that $\|F\| \le CM$. This F is given by an explicit formula. In particular, it depends linearly on the family $(P^x)_{x\in E}$.

The proof of Whitney's extension theorem is a milestone in our understanding of analysis. Along with the work of Marcinkiewicz [14], it is an early appearance of the idea of a Calderón-Zygmund decomposition.

In Questions $1 \cdots 4$, we are given merely a function value $f(x)$ at each point $x \in E$, rather than a jet P^x as in the Whitney Extension Theorem. It is our responsibility to compute (or guess) the derivatives of F up to order m at the points of E.

Progress on Whitney's problems occurred over several decades. In 1958, G. Glaeser [12] answered Questions 1 and 3 in the case of $C^1(\mathbb{R}^n)$ ($m = 1$). Glaeser's solution was based on a geometric construction, which he called the "iterated paratangent space."

In the 1970's, Y. Brudnyi and P. Shvartsman ([3], [4]) discovered the key idea of "finiteness principles." The simplest finiteness principle is as follows:

Theorem 2. *Let $E \subset \mathbb{R}^2$ be finite, and let $f : E \to \mathbb{R}$. For every subset $S \subset E$ with at most 6 points, suppose there exists $F^S \in C^2(\mathbb{R}^2)$ with norm at most 1, such that $F^S = f$ on S. Then there exists $F \in C^2(\mathbb{R}^2)$ with norm less than a universal constant, such that $F = f$ on E.*

Brudnyi and Shvartsman also proved the analogous result for $C^2(\mathbb{R}^n)$; the number of points in the subset S is at most $k^\# \equiv 3 \cdot 2^{n-1}$ ($= 6$ in two dimensions); Brudnyi–Shvartsman showed that this $k^\#$ is best possible. For several related results and conjectures, we refer the reader to [3,4].

In 2002, E. Bierstone, P. Milman and W. Pawłuski [2] found an analogue of Glaeser's iterated paratangent space relevant to $C^m(\mathbb{R}^n)$. They answered a close relative of Questions 1 and 3 for the case in which $E \subset \mathbb{R}^n$ is a subanalytic set.

Finally, the papers [6,7,8] proved a finiteness principle for $C^m(\mathbb{R}^n)$, modified slightly the iterated paratangent space of Bierstone–Milman–Pawłuski, and gave complete answers to Questions $1 \cdots 4$. For further details, we refer the reader to the expository paper [11].

Our purpose here is to discuss the problem of interpolation of data. Again, we fix $m, n \geq 1$, and we work in $C^m(\mathbb{R}^n)$. This time, we are given $f : E \to \mathbb{R}$ with E finite; say, $\#(E) = N$, where $\#(E)$ denotes the number of points of E. We may not choose to fit f perfectly. Therefore, we suppose we are given a "tolerance" $\sigma : E \to [0, \infty)$.

We want to compute a function $F \in C^m(\mathbb{R}^n)$ and a real number $M \geq 0$ such that:

(1) $\|F\| \leq M$, and $|F(x) - f(x)| \leq M\sigma(x)$ for all $x \in E$.

We would like to take M as small as possible. In the special case $\sigma \equiv 0$, we demand that $F = f$ on E. [2]

Let us define $\|f\|_{E,\sigma}$ to be the infimum of M over all pairs (F, M) that satisfy (1). Elementary examples show that this infimum needn't be attained. Therefore, for a constant $C > 1$, we define a "C–optimal interpolant" for (E, f, σ) to be a function $F \in C^m(\mathbb{R}^n)$ such that, for some $M \geq 0$, we have:

(2) $\|F\| \leq M$; $|F(x) - f(x)| \leq M\sigma(x)$ for all $x \in E$; and $M \leq C\|f\|_{(E,\sigma)}$.

We can now give a crude statement of the problem of interpolating data. Given m, n, E, f, σ, we pose:

[2] It is perhaps natural to use two different positive numbers M_1, M_2 in the two inequalities in (1). However, since we are free to multiply $\sigma(x)$ by our favorite positive constant, we lose no generality in taking a single M.

Problem 1: Compute a C–optimal interpolant with C not too big.

Problem 2: Compute the order of magnitude of $\|f\|_{(E,\sigma)}$.

Our solution will consist of an algorithm, to be run on an (idealized) computer.

To make the above problems precise, we owe the reader several clarifications. We must explain

- What it means to say that C is "not too big."
- What we mean by the "order of magnitude."
- What's an (idealized) computer?
- What it means to "compute a function."
- Efficient vs. wasteful computation.

The explanations are as follows:

A "not-too-big"constant is simply a constant depending only on m and n. (Recall, we are working in $C^m(\mathbb{R}^n)$.) We denote such such constants by c, C, C', etc. These symbols may denote different constants in different occurrences.

To "compute the order of magnitude"of a real number $X \geq 0$ is to compute some real number Y for which we guarantee that

$cX \leq Y \leq CX$.

Our idealized "computer" has standard von Neumann architecture. Unlike a real computer, our idealized machine is assumed here to deal with exact real numbers, without roundoff error. Thus, we assume that an arbitrary real number may be stored at a single memory address, and that the registers perform arithmetic operations (including powers and logarithms) to perfect accuracy.[3]

Our idealized computer can deal with only finitely many real numbers. What does it mean to "compute a function" $F \in C^m(\mathbb{R}^n)$?

We have in mind the following dialogue with the computer.

First, we enter the data (m, n, E, f, σ). The computer performs one-time-work, then responds to our queries.

A query consists of a point $\underline{x} \in \mathbb{R}^n$. When we enter the coordinates of a query point $\underline{x}$, the computer responds by producing the list of numbers $(\partial^\alpha F(\underline{x}))_{|\alpha|\leq m}$.

[3]This unrealistic model of computation is subject to serious criticism [15]. In Fefferman–Klartag [9,10], we make a rigorous analysis of the roundoff error. That analysis is omitted in this expository article for the sake of simplicity.

Since $F \in C^m(\mathbb{R}^n)$, this is the most we can expect.

The above notion of "computing a function" is clearly too restrictive for many purposes. It demands in particular that $F(\underline{x})$ can be computed exactly from $\underline{x}$ by performing finitely many arithmetic operations. Thus, Bessel functions cannot be computed according to our stringent definition. Since our main theorem will assert that the desired F can be computed, we are undisturbed by the overly strict definition.

Finally, we owe the reader a discussion of "efficient" vs. "wasteful" computation. The resources used to compute $F \in C^m(\mathbb{R}^n)$ (as explained above) are as follows.

- The number of computer operations used to perform the one-time work
- The number of operations used to respond to a query
- The number of memory cells in the RAM. (Recall that each memory cell can hold a single real number.)

We refer to these, respectively, as the "one-time work," the "query work," and the "storage." For an "efficient" algorithm, all the resources used are as small as possible.

We note a few trivial lower bounds for the resources needed to compute a C–optimal interpolant for (m, n, E, f, σ). First of all, any interpolation algorithm must at least read the data and reproduce $f(\underline{x})$ perfectly for $\underline{x} \in E$ if we take $\sigma \equiv 0$.

Since E consists of N points, it follows that any interpolation algorithm entails at least N operations of one-time work, and at least N memory cells of storage.

Similarly, to respond to a query $\underline{x}$, any interpolation algorithm must at least read the query and print out a response. Thus, the query work for any interpolation algorithm is at least 1.

We point out also that any algorithm that computes the order of magnitude of $\|f\|_{(E,\sigma)}$ requires at least N operations, since at least it looks at the data.

We are now ready to state our main results. We are given positive integers m, n; a finite set $E \subset \mathbb{R}^n$ consisting of N points; and functions $f : E \to \mathbb{R},\ \sigma : E \to [0, \infty)$.

Theorem 3. *The algorithm given in [10] computes a C_1–optimal interpolant, using one-time work at most $C_2 N \log N$, storage at most $C_3 N$, and query work at most $C_4 \log N$.*

Theorem 4. *The algorithm given in [9] computes the order of magnitude of $\|f\|_{(E,\sigma)}$ using work at most $C_5 N \log N$ and storage at most $C_6 N$.*

Recall that $C_1 \cdots C_6$ denote constants depending only on m and n.

The computer resources indicated by Theorems 3 and 4 differ only by a factor $\log N$ from the trivial lower bounds we pointed out above. Very likely, Theorems 3 and 4 are best possible. Nevertheless, the algorithms in [9,10] are not of practical use, because they compute a C_1–optimal interpolant for a very large constant C_1. (The constants $C_2, \cdots, C_6$ are not nearly

so bad.) We hope that an improved version of our algorithm may (someday) yield practical results.

Note that our algorithms apply to arbitrary interpolation problems (m, n, E, f, σ). We have made no assumptions on the geometry of the set E. Such assumptions greatly simplify the task of interpolating data.

The rest of this article gives some ideas from the proofs of Theorems 3 and 4. We will define certain basic convex sets $\Gamma(x, M)$, compute their approximate size and shape, and use them to compute a C–optimal F. We begin with the definition. Let m, n, E, f, σ be given. For each $x \in \mathbb{R}^n$ and $M > 0$, we define $\Gamma(x, M)$ to consist of the Taylor polynomials $J_x(F)$ of all functions $F \in C^m(\mathbb{R}^n)$ that satisfy

(3) $\|F\| \leq M$ and $|F - f| \leq M\sigma$ on E.

Immediately from the definition, we see that $\Gamma(x, M)$ is a (possibly empty) convex subset of the vector space $\mathcal{P}$ of m^{th} degree polynomials. (In particular, $\Gamma(x, M)$ is empty if we take M too small, since there are then no functions F satisfying (3).)

The convex sets $\Gamma(x, M)$ are a key tool in computing a C–optimal interpolant. Moreover, we hope to convince the reader that they are interesting in their own right. To see this, consider a trivial one-dimensional example: Suppose I take a car trip. My position is a function of time. Say $y = F(t)$. I don't know the function F, but I know that its second derivative satisfies $|F''(t)| \leq M$ for some explicit M, because my car can't accelerate very fast. At particular times $t_1, t_2, \cdots, t_N$, I look out the window and observe my approximate position. This tells me that

$$|F(t) - f(t)| \leq M\sigma(t) \quad \text{for} \quad t \in E \equiv \{t_1,\ t_2,\ \cdots,\ t_N\},$$

where f and σ arise from my observations. Given the above, what can we say about my position, velocity and acceleration at some given time t_0? This question amounts to asking us to compute $\Gamma(t_0, M)$ in a one-dimensional case.

We now explain how to compute the approximate size and shape of $\Gamma(x, M)$. Our goal is to compute convex sets

$$\Gamma_*(x, M) \subset \mathcal{P} \quad \text{(possibly empty)}, \quad \text{such that}$$

$$\Gamma_*(x, cM) \subset \Gamma(x, M) \subset \Gamma_*(x, M).$$

We explain how to exhibit such $\Gamma_*(x, M)$. In this expository article, we restrict attention to $x \in E$, even though $\Gamma_*(x, M)$ can be computed for general $x \in \mathbb{R}^n$.

By induction on $\ell \geq 0$, we will define (possibly empty) convex sets $\Gamma_\ell(x, M)$ (all $x \in E$). These Γ_ℓ will satisfy

$$\Gamma_\ell(x, M) \supset \Gamma(x, M), \text{ and}$$

$$\Gamma_\ell(x, M) \supset \Gamma_{\ell+1}(x, M), \quad \text{for each } x \in E \quad \text{and} \quad \ell \geq 0.$$

We will then set $\mathit{\Gamma}_*(x, M) \equiv \mathit{\Gamma}_{\ell_*}(x, M)$, for a large enough integer constant ℓ_* depending only on m and n.

The inductive definition of the $\mathit{\Gamma}_\ell(x, M)$ proceeds as follows.

In the base case $\ell = 0$, we simply define

$$\mathit{\Gamma}_0(x, M) = \{P \in \mathcal{P} : |\partial^\alpha P(x)| \le M \quad \text{for } |\alpha| \le m, \quad \text{and} \quad |P(x) - f(x)| \le M\sigma(x)\}$$

for each $x \in E$. Note that $\mathit{\Gamma}_0(x, M)$ is a (possibly empty) convex subset of $\mathcal{P}$, and that $\mathit{\Gamma}_0(x, M) \supset \mathit{\Gamma}(x, M)$.

The set $\mathit{\Gamma}_0(x, M)$ is defined trivially, ignoring the information available at points of E other than x.

For the induction step, we fix $\ell \ge 0$, and we suppose that $\mathit{\Gamma}_\ell(x, M)$ has already been defined for all $x \in E$. We suppose that each $\mathit{\Gamma}_\ell(x, M)$ is a (possibly empty) convex subset of $\mathcal{P}$, and that $\mathit{\Gamma}_\ell(x, M) \supset \mathit{\Gamma}(x, M)$ for each $x \in E$.

Our task is to define a (possibly empty) convex set $\mathit{\Gamma}_{\ell+1}(x, M)$ for each $x \in E$, and check that

$$\mathit{\Gamma}_\ell(x, M) \supset \mathit{\Gamma}_{\ell+1}(x, M) \supset \mathit{\Gamma}(x, M).$$

This will complete our induction on ℓ.

To define $\mathit{\Gamma}_{\ell+1}(x, M)$, we just use Taylor's theorem, which can be stated in the following form.

Let $F \in C^m(\mathbb{R}^n)$ with $\|F\| \le M$, and suppose $x, y \in \mathbb{R}^n$.

Set $P = J_x(F)$ and $P' = J_y(F)$. Then

$$|\partial^\alpha (P - P')(x)| \le M|x - y|^{m-|\alpha|} \quad \text{for all } |\alpha| \le m.$$

As a corollary, we obtain a basic property of the sets $\mathit{\Gamma}(x, M)$.

Proposition. *Let $x, y \in E$. Given $P \in \mathit{\Gamma}(x, M)$, there exists $P' \in \mathit{\Gamma}(y, M)$ such that*

$$|\partial^\alpha (P - P')(x)| \le M|x - y|^{m-|\alpha|} \quad \textit{for } |\alpha| \le m.$$

The above Proposition motivates our definition of $\mathit{\Gamma}_{\ell+1}(x, M)$. For each $x \in E$, we take $\mathit{\Gamma}_{\ell+1}(x, M)$ to consist of all $P \in \mathit{\Gamma}_\ell(x, M)$ such that, for each $y \in E$, there exists $P' \in \mathit{\Gamma}_\ell(y, M)$ for which we have

$$|\partial^\alpha (P - P')(x)| \le M|x - y|^{m-|\alpha|} \quad (\text{all } |\alpha| \le m).$$

Note that $\mathit{\Gamma}_{\ell+1}(x, M)$ is a (possibly empty) convex subset of $\mathcal{P}$, and that $\mathit{\Gamma}_\ell(x, M) \supset \mathit{\Gamma}_{\ell+1}(x, M)$.

Moreover, $\mathit{\Gamma}_{\ell+1}(x, M) \supset \mathit{\Gamma}(x, M)$, thanks to the above Proposition.

This completes our induction on ℓ. We have succeeded in defining the $\mathit{\Gamma}_\ell(x, M)$ for all $x \in E, \quad \ell \geq 0$.

Note that our definition of $\mathit{\Gamma}_{\ell+1}(x, M)$ for a given $x \in E$ involves $\mathit{\Gamma}_\ell(y, M)$ for all $y \in E$. We will return to this point soon.

We know that each $\mathit{\Gamma}_\ell(x, M)$ is a (possibly empty) convex subset of P, and that

$\mathit{\Gamma}_\ell(x, M) \supset \mathit{\Gamma}(x, M)$ for each ℓ; and

$\mathit{\Gamma}_\ell(x, M) \supset \mathit{\Gamma}_{\ell+1}(x, M)$.

The basic mathematical result on the above $\mathit{\Gamma}_\ell$ is as follows

Theorem 5. *For a large enough integer constant ℓ_* (depending only on m and n), we have*

$\mathit{\Gamma}_{\ell_*}(x, cM) \subset \mathit{\Gamma}(x, M) \subset \mathit{\Gamma}_{\ell_*}(x, M)$ *for all* $x \in E$ *and* $M > 0$.

Thus, we have succeeded in computing the approximate size and shape of the $\mathit{\Gamma}(x, M)$. Unfortunately, the above computation is too expensive. Recall that each $\mathit{\Gamma}_{\ell+1}(x, M)$ is defined using all the $\mathit{\Gamma}_\ell(y, M)(y \in E)$. Since each y talks to each x, the work required to compute all the $\mathit{\Gamma}_{\ell+1}(x, M)(x \in E)$ from all the $\mathit{\Gamma}_\ell(y, M)(y \in E)$ contains a factor N^2. In Theorems 3 and 4, we promised algorithms that do only $N \log N$ work.

Thus, we cannot use the above $\mathit{\Gamma}_\ell$'s to compute the approximate size and shape of the $\mathit{\Gamma}(x, M)$.

The remedy is to change the definition of the $\mathit{\Gamma}_\ell$ without losing their usefulness. To do so, we bring in an idea from computer science, namely the Well-Separated Pairs Decomposition (WSPD), due to Callahan-Kosaraju [5]; see also Har-Peled and Mendel [13]. (We also use the closely related Balanced Box Decomposition Tree of Arya-Mount-Netanyahu-Silverman-Wu [1], but we suppress that discussion here.)

To understand how the WSPD can overcome the need to do N^2 work, we next discuss a problem much easier than the interpolation problems considered above.

Let $E \subset \mathbb{R}^n$ be a finite set, consisting of N points. Let $f : E \to \mathbb{R}$. We want to compute the Lipschitz constant of f, given by

$$\|f\|_{Lip} = \max_{x', x'' \in E \text{ distinct}} \frac{|f(x') - f(x'')|}{|x' - x''|}.$$

The obvious computation of $\|f\|_{Lip}$ requires work $\sim N^2$. However, a clever method computes $\|f\|_{Lip}$ to within (say) a 1% error by using $O(N \log N)$ operations. The idea is that for certain E', $E'' \subset E$, we can compute the restricted maximum

$$\wedge \equiv \left[\max_{(x',x'') \in E' \times E''} \frac{|f(x') - f(x'')|}{|x' - x''|} \right]$$

much faster than the obvious way.

We will take E' and E'' well-separated, i.e., we suppose that

$$\text{distance}(E', E'') > 10^3 \cdot [\text{diameter } (E') + \text{ diameter } (E'')].$$

Here, of course,

$$\text{distance } (E', E'') = \min\{|x' - x''| : x' \in E',\ x'' \in E''\},$$

$$\text{diameter } (E') = \max\{|x' - y'| : x',\ y' \in E'\}, \text{ and similarly for diameter } (E'').$$

Let us see how to compute $\wedge$ to within a 1% error for such E', E'', faster than the obvious method that takes work $\sim \#(E') \cdot \#(E'')$.

First of all, as (x', x'') varies over $E' \times E''$, the distance $|x' - x''|$ is essentially constant. Therefore, to compute $\wedge$, it's enough to compute

$$\max\{|f(x') - f(x'')| : x' \in E',\ x'' \in E''\}.$$

To achieve the above maximum, either

Case 1: We maximize $f(x')$ over all $x' \in E'$, and
minimize $f(x'')$ over all $x'' \in E''$;

or else

Case 2: We maximize $f(x'')$ over all $x'' \in E''$, and
minimize $f(x')$ over all $x' \in E'$.

It follows easily that $\wedge$ can be computed (up to a 1% error) with work $\sim \#(E') + \#(E'')$.

We have succeeded in beating the trivial algorithm to compute $\wedge$.

The above discussion motivates the following

Theorem 6. *Let $E \subset \mathbb{R}^n$ be a finite set, consisting of N elements. Then $\{(x, y) \in E \times E : x \neq y\}$ can be partitioned into Cartesian products $E'_1 \times E''_1,\ E'_2 \times E''_2,\ \cdots,\ E'_L \times E''_L$, with $L \leq CN$, such that each pair (E'_ℓ, E''_ℓ) is well-separated. Moreover, an efficient algorithm computes the above decomposition.[5]*

We omit a careful discussion of the meaning of the preceding sentence. Returning to the problem of computing the Lipschitz constant, we deliver the coup-de-grâce.

Let $E'_1 \times E''_1,\ E'_2 \times E''_2,\ \cdots,\ E'_L \times E''_L$ be as in Theorem 6. For each $\ell = 1,\ \cdots,\ L$, let $(x'_\ell, x''_\ell) \in E'_\ell \times E''_\ell$ be given. Then the Lipschitz constant of f differs by at most 1% from the quantity

(4)
$$\max\left\{\left|f(x'_\ell) - f(x''_\ell)\right| \ / \ \left|x'_\ell - x''_\ell\right| :\ \ \ell = 1,\ \cdots,\ L\right\}. \qquad \text{(We will prove this in a moment.)}$$

Consequently, once we find the "representatives" $(x'_\ell,\ x''_\ell),\ \ell = 1,\ \cdots,\ L,$ we can then compute the Lipschitz constant of f with work at most $C \cdot L \leq C'N$. To compute the representatives by the algorithm of Callahan-Kosaraju requires work $O(N \log N)$. Thus, computing a Lipschitz constant, a task that seems to require N^2 operations, can actually be done in $O(N \log N)$ operations.

We pause to explain in detail why the Lipschitz constant of f is given by the restricted maximum (4) up to a 1% error. The point is as follows.

Proposition. *Suppose* $|f(x'_\ell) - f(x''_\ell)| \leq |x'_\ell - x''_\ell|$ *for* $\ell = 1,\ \cdots,\ L.$
Then $|f(x') - f(x'')| \leq (1.01)|x' - x''|$ *for all* $x',\ x'' \in E.$

Proof: Suppose not. Let $(x', x'') \in E \times E$ be a counterexample with $|x' - x''|$ as small as possible. Thus,

(5) $\left|f(x') - f(x'')\right| > (1.01)|x' - x''|$ (strict inequality).

In view of the strict inequality, x' and x'' are distinct. Consequently, we have $(x', x'') \in E'_\ell \times E''_\ell$ for some ℓ. Fix that ℓ. Since E'_ℓ and E''_ℓ are well-separated, and since also $(x'_\ell, x''_\ell) \in E'_\ell \times E''_\ell$, it follows that

$$|x' - x'_\ell| + |x'' - x''_\ell| \leq \text{ diam } (E'_\ell) + \text{diam } (E''_\ell) \leq 10^{-3} \text{ dist } (E', E'')$$

and therefore

(6) $|x' - x'_\ell| + |x'' - x''_\ell| \leq 10^{-3}|x'_\ell - x''_\ell|.$

Thanks to our choice of x' and x'' to minimize $|x' - x''|$, it follows from (6) that

$$|f(x') - f(x'_\ell)| \leq (1.01)|x' - x'_\ell| \ \text{ and } \ |f(x'') - f(x''_\ell)| \leq (1.01)|x'' - x''_\ell|.$$

Consequently,

$$\begin{aligned} |f(x') - f(x'')| &\leq |f(x') - f(x'_\ell)| + |f(x'_\ell) - f(x''_\ell)| + |f(x''_\ell) - f(x'')| \\ &\leq (1.01)|x' - x'_\ell| + |x'_\ell - x''_\ell| + (1.01)|x'' - x''_\ell| \\ &\leq (1.01) \cdot 10^{-3}|x'_\ell - x''_\ell| + |x'_\ell - x''_\ell| + (1.01) \cdot 10^{-3}|x'_\ell - x''_\ell| \\ &\leq (1.003) \cdot |x'_\ell - x''_\ell| \leq (1.01)|x' - x''|, \end{aligned}$$

where the last inequality follows from (6).

Thus, $|f(x')-f(x'')| \le (1.01)|x'-x''|$, contradicting (5). ■

This concludes our discussion of the Lipschitz constant. Returning to the interpolation problem, we now change the definition of the convex sets $\Gamma_\ell(x, M)$ by using the well-separated pairs decomposition. Our new Γ_ℓ's can be computed with $N \log N$ work, and they have the following key properties in common with the old, expensive Γ_ℓ's.

Property 0: $\Gamma_0(x, M) = \{P \in \mathcal{P} : |\partial^\alpha P(x)| \le M \quad (\text{all } \alpha)$
$\text{and} \quad |P(x) - f(x)| \le M\}$.

Property 1: $\Gamma_\ell(x, M)$ is a (possibly empty) convex subset of $\mathcal{P}$.

Property 2: $\Gamma_\ell(x, M) \supset \Gamma(x, M)$ for each $x \in E$, $\ell \ge 0$.

Property 3: $\Gamma_\ell(x, M) \supset \Gamma_{\ell+1}(x, M)$ for each $x \in E$, $\ell \ge 0$.

Property 4: Let $x, y \in E$, and let $P \in \Gamma_{\ell+1}(x, M)$.
Then there exists $P' \in \Gamma_\ell(y, M)$ such that
$|\partial^\alpha (P - P')(x)| \le M|x - y|^{m-|\alpha|}$ (all $|\alpha| \le m$).

In our earlier discussion, we essentially took Property 4 to be the definition of $\Gamma_{\ell+1}(x, M)$.

There are additional key properties enjoyed by the new $\Gamma_\ell(x, M)$, but we omit them here. Indeed, we omit the definition of our new Γ_ℓ, which requires additional structure of the WSPD not discussed in this article.

The proof of Theorem 5 uses only the key properties on the above list, not the precise definition of the Γ_ℓ. Therefore, thanks to Theorem 5 (in its generalized form), our new, cheaper Γ_ℓ allow us to compute the approximate size and shape of the convex sets $\Gamma(x, M)$.[4] The computation, running over all $x \in E$, takes at most $O(N \log N)$ operations, as promised in Theorems 3 and 4.

This concludes our discussion of the computation of the approximate size and shape of the $\Gamma(x, M)$.

At last it is time to explain how to prove Theorems 3 and 4, once we have computed the approximate size and shape of the $\Gamma(x, M)$ (by computing our cheap $\Gamma_\ell(x, M)$). These results reduce quickly to Theorem 5. In fact, by definition, the set $\Gamma(x, M)$ is non-empty if and only if $M \ge \|f\|_{(E,\sigma)}$.

By computing the approximate size and shape of the $\Gamma(x, M)$, we have in particular computed the order of magnitude of $\|f\|_{(E,\sigma)}$. Thus, Theorem 4 reduces easily to Theorem 5. Moreover, the proof of Theorem 5 is constructive. Given $P \in \Gamma_{\ell_*}(x, cM)$, we prove that

[4]For the rest of this article, whenever we refer to Theorem 5, we mean the generalized version in which the Γ_ℓ are not specified, but merely assumed to satisfy a list of key properties.

$P \in \Gamma(x, M)$ by constructing an explicit interpolant F, satisfying $\|F\| \leq M$, $|F - f| \leq M\sigma$ on E, and $J_x(F) = P$. If we take M as small as possible with $\Gamma_{\ell *}(x, cM)$ nonempty, then the interpolant F is C–optimal. The proof of Theorem 3 amounts to an efficient implementation of the proof of Theorem 5.

Thus, everything comes down to Theorem 5, which (after a trivial localization using a partition of unity) in turn amounts to solving the following LOCAL INTERPOLATION PROBLEM.

LIP (Q, x_0, P_0): Suppose we are given a cube Q, a point $x_0 \in E \cap Q$, and a polynomial $P_0 \in \Gamma_\ell(x_0, M)$ for a suitable ℓ. Produce a function $F_Q \in C^m(Q)$ such that:

(*1) The m^{th} derivatives of F are bounded by CM on Q;

(*2) $|F_Q(x) - f(x)| \leq CM\sigma(x)$ for all $x \in E \cap Q$; and

(*3) $J_{x_0}(F_Q) = P_0$.

Our task is to solve LIP (Q, x_0, P_0) for a cube Q of sidelength 1. To carry out this task, we will also consider LIP (Q, x_0, P_0) for smaller cubes Q.

In the explanation below, we sacrifice accuracy for ease of understanding.

A local interpolation problem carries a "label" $\mathcal{A}$, to be explained later. For the moment, we just remark that the label $\mathcal{A}$ tells us certain information on the geometry of the convex set $\Gamma_\ell(x_0, M)$ in LIP (Q, x_0, P_0). When we "attach" the label $\mathcal{A}$ to the problem LIP (Q, x_0, P_0), we guarantee in advance that the geometric conditions indicated by $\mathcal{A}$ hold for the convex set $\Gamma_\ell(x_0, M)$. This information may help us in constructing functions F_Q that satisfy (*1) $\cdots$ (*3) above.

Thus LIP (Q, x_0, P_0) may or may not carry a given label $\mathcal{A}$; but if it does, then we have extra information that may help us solve the problem LIP (Q, x_0, P_0).

One particular label plays a special role; it is the empty set $\varnothing$, which provides no information whatever on the geometry of $\Gamma_\ell(x_0, M)$. Every LIP (Q, x_0, P_0) carries the label $\varnothing$.

There is a natural order relation $<$ on the set of all labels. If $\mathcal{A}' < \mathcal{A}$, then an interpolation problem that carries the label $\mathcal{A}'$ is easier than a problem that carries the label $\mathcal{A}$. In particular, the label $\varnothing$ corresponds to the hardest version of LIP (Q, x_0, P_0), in which we are given no additional information to work with.

There are only finitely many possible labels; indeed, the number of labels is a constant depending only on m and n.

We will solve a local interpolation problem with a given label $\mathcal{A}$ by reducing it to local interpolation problems with labels $\mathcal{A}' < \mathcal{A}$. Thus, we proceed by induction on the label. Let us now describe that induction.

In the base case, our local interpolation problem LIP (Q, x_0, P_0) carries the easiest possible label (the label called $\mathcal{M}$ later on). We may then simply take $F_Q = P_0$, and one checks without

difficulty that F_Q satisfies (*1) $\cdots$ (*3). Thus, we can easily solve LIP (Q, x_0, P_0) in the base case.

For the induction step, we fix a label $\mathcal{A}$, and suppose that we can solve any local interpolation problem that carries a label $\mathcal{A}' < \mathcal{A}$. We must solve a LIP (Q, x_0, P_0) that carries the label $\mathcal{A}$. To do so, we partition Q into finitely many subcubes Q_ν, and introduce a "representative point" $x_\nu \in E \cap Q_\nu$ for each ν. The construction of the partition involves the label $\mathcal{A}$ and the geometry of the $\Gamma_{\ell-1}(x, M)$ for all the points $x \in E \cap Q$. For each x_ν, we invoke the key property called Property 4 above, with ℓ in Property 4 replaced by our present $\ell - 1$. Thus, for each ν, we obtain a polynomial

(7) $P_\nu \in \Gamma_{\ell-1}(x_\nu, M)$, such that $|\partial^\alpha (P_\nu - P_0)(x_0)| \leq M|x_\nu - x_0|^{m-|\alpha|}$ for $|\alpha| \leq m$.

We now have a cube Q_ν, a point $x_\nu \in E \cap Q_\nu$, and a polynomial $P_\nu \in \Gamma_{\ell-1}(x_\nu, M)$. Thus, we can pose the local interpolation problem

LIP $(\nu) \equiv$ LIP(Q_ν, x_ν, P_ν) for each ν.

Our partition $\{Q_\nu\}$ was constructed to guarantee that each of the above problems LIP(ν) carries a label $\mathcal{A}'_\nu < \mathcal{A}$. Therefore, by our induction hypothesis, we can solve each LIP(ν), to produce a "local interpolant" $F_\nu \in C^m(Q_\nu)$ that satisfies:

(*1)$_\nu$ The m^{th} derivatives of F_ν are bounded by CM on Q_ν;

(*2)$_\nu$ $|F_\nu(x) - f(x)| \leq CM\sigma(x)$ for all $x \in E \cap Q_\nu$; and

(*3)$_\nu$ $J_{x_\nu}(F_\nu) = P_\nu$.

By using a partition of unity adapted to the partition $\{Q_\nu\}$, we can then patch together the F_ν into a function $F_Q \in C^m(Q)$.

If we are careful, then our F_Q will satisfy (*1) $\cdots$ (*3) and thus solve LIP (Q, x_0, P_0). This will complete our induction on the label $\mathcal{A}$, solve all local interpolation problems, and complete the proof of Theorem 5.

To make the above construction work, we have to take one main precaution. We must pick the polynomials P_ν in (7) to satisfy the consistency condition

(8) $|\partial^\alpha (P_\nu - P_{\nu'})(x_\nu)| \leq CM\,|x_\nu - x_{\nu'}|^{m-|\alpha|}$ (all $|\alpha| \leq m$)

whenever the cubes Q_ν and $Q_{\nu'}$ touch. This is much stronger than the defining condition (7), since Q_ν and $Q_{\nu'}$ may be much smaller than Q, and therefore $|x_\nu - x_{\nu'}|$ may be much smaller than $|x_0 - x_\nu|$. If (8) fails, then we have no chance to control the m^{th} derivatives of F as in (*1).

Thus, it is essential to pick the P_ν to satisfy (8) in addition to (7). Achieving this extra consistency is the most delicate point in our proof of Theorem 5. We postpone for a few paragraphs our brief remarks on how to achieve (8).

We owe the reader an explanation of a local interpolation problem LIP (Q, x_0, P_0) with a "label" $\mathcal{A}$. In fact, a label $\mathcal{A}$ is simply a subset of the set $\mathcal{M}$ of all multi-indices $\alpha =$

$(\alpha_1, \alpha_2, \ \cdots, \ \alpha_n) \in \{0, 1, 2, \cdots\}^n$ of order $|\alpha| = \alpha_1 + \cdots + \alpha_n \leq m - 1$. The problem LIP (Q, x_0, P_0) carries the label $\mathcal{A}$ if there exist polynomials $P_\alpha \in P$ indexed by $\alpha \in \mathcal{A}$, such that the following conditions are satisfied, where δ_Q denotes the sidelength of Q and $\delta_{\beta\alpha}$ denotes the Kronecker delta.

- $P_0 + M\delta_Q^{m-|\alpha|} P_\alpha \in \Gamma_\ell(x_0, CM)$ for each $\alpha \in \mathcal{A}$.
- $\partial^\beta P_\alpha(x_0) = \delta_{\beta\alpha}$ for $\beta, \alpha \in \mathcal{A}$.
- $\left|\partial^\beta P_\alpha(x_0)\right| \leq C\delta_Q^{|\alpha|-|\beta|}$ for $\beta \in \mathcal{M}, \ \alpha \in \mathcal{A}$.

These conditions assert that the convex set $\Gamma_\ell(x_0, CM) \subset \mathcal{P}$ is "fat enough "in certain directions corresponding to the $P_\alpha \in P$.

Note that the above conditions hold vacuously when $\mathcal{A}$ is the empty set $\varnothing$. Thus, as promised, the label $\varnothing$ provides no extra information.

However, when $\mathcal{A}$ is non-empty, the above conditions provide us with some room to maneuver–we can change P_0 and stay inside $\Gamma_\ell(x_0, CM)$. In the proof of Theorem 5 sketched above, we exploit this freedom of maneuver for each of the local problems LIP(Q_ν, x_ν, P_ν). Initially, the P_ν satisfy the consistency condition in (7), but not the strong consistency condition (8). However, by exploiting our freedom of maneuver, we can modify slightly each P_ν, to achieve (8) without sacrificing (7).

This completes our sketch of the proof of Theorem 5. We again warn the reader that it isn't completely accurate. See [9,10] for the correct version.

I am grateful to Frances Wroblewski for TeXing this paper.

References

[1] S. Arya, D. Mount, N. Netanyahu, R. Silverman, A. Wu, An optimal algorithm for approximate nearest neighbor searching in fixed dimensions, *Journal of the Association for Computing Machinery* **45** no. 6 (1998), 891-923.

[2] E. Bierstone, P. Milman, W. Pawłucki, Differentiable functions defined on closed sets, A problem of Whitney, *Inventiones Math.* **151** no. 2 (2003), 329-352.

[3] Y. Brudnyi and P. Shvartsman, Generalizations of Whitney's extension theorem, *Int. Math. Research Notices* **3** (1994), 129-139.

[4] Y. Brudnyi and P. Shvartsman, The Whitney problem of existence of a linear extension operator, *J. Geometric Analysis* **7** no. 4 (1997) , 515-574.

[5] P. B. Callahan and S. R. Kosaraju, A decomposition of multi-dimensional point sets with applications to k-nearest neighbors and n-body potential fields, *Journal of the Association for Computing Machinery* **42** no. 1 (1995), 67-90.

[6] C. Fefferman, A sharp form of Whitney's extension theorem, *Annals of Math* **161** (2005), 509-577.

[7] C. Fefferman, Whitney's extension problem for C^m, *Annals of Math* **164** no. 1 (2006), 313-359.

[8] C. Fefferman, C^m extension by linear operators, *Annals of Math* **166** no. 3 (2007), 779-835.

[9] C. Fefferman and B. Klartag, Fitting a C^m-smooth function to data I, *Annals of Math* **169** no. 1 (2009), 315-346.

[10] C. Fefferman and B. Klartag, Fitting a C^m-smooth function to data II, *Revista Mat. Iberoamericana* **25** no. 1 (2009), 49-273.

[11] C. Fefferman, Whitney's extension problems and interpolation of data, *Bulletin A.M.S.* **46** no. 2 (2009), 207-220.

[12] G. Glaeser, Etudes de quelques algebres tayloriennes, *J. d'Analyse* **6** (1958), 1-124.

[13] S. Har-Peled and M. Mendel, Fast construction of nets in low-dimensional metrics and their applications, *SIAM Journal on Comut.* **35** no. 5 (2006), 1148-1184.

[14] J. Marcinkiewicz, Sur les series de Fourier, *Fund. Math.* **27** (1936), 38-69.

[15] A. Schönhage, On the power of random access machines, Proc. 6th Internat. Colloq. Automata Lang. Program, *Lecture Notes Comput. Sci.* **71**, Springer Verlag (1979), 520-529.

[16] H. Whitney, Analytic extensions of differentiable functions defined in closed sets, *Trans. A.M.S.* **36** (1934), 63-89.

[17] H. Whitney, Differentiable functions defined in closed sets I, *Trans. A.M.S.* **36** (1934), 369-389.

[18] H. Whitney, Functions differentiable on the boundaries of regions, *Annals of Math* **35** (1934), 482-485.