Acta Numerica (2016), pp. 321–382
doi:10.1017/S0962492916000027
© Cambridge University Press, 2016
Printed in the United Kingdom

Probabilistic analyses of condition numbers

Felipe Cucker*
Department of Mathematics,
City University of Hong Kong
E-mail: macucker@cityu.edu.hk

In recent decades, condition numbers have joined forces with probabilistic analysis to give rise to a form of condition-based analysis of algorithms. In this paper we survey how this analysis is done via a number of examples. We precede this catalogue of examples with short primers on both condition numbers and probabilistic analyses.

CONTENTS

1 A brief introduction to condition numbers 322
2 A brief introduction to probabilistic analysis 326
3 A repertoire of examples 330
References 380

Condition numbers were independently introduced by Alan Turing (1948) and by John von Neumann and Herman Goldstine (1947) as a tool in their efforts to understand the accuracy (or equivalently the loss of precision) in the solution of linear systems in the presence of finite-precision arithmetic. Very shortly thereafter, Magnus Hestenes and Eduard Stiefel (1952) showed that these condition numbers played a central role in the determination of the speed of convergence of the conjugate gradient method, thus adding time to accuracy in the list of features that appeared to be controlled, at least partially, by the condition of the input data.

During the more than sixty years that followed the publication of these fundamental articles, condition numbers became pervasive characters in the literature of numerical algorithms. Unfortunately, this ubiquity of condition numbers is balanced by some difficulty in nailing them down: it appears that computing the condition number of an input, for a given problem, is no easier than solving the problem for this particular input. A programme

* This work was partially supported by GRF grant CityU 100813.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

that emerged from this situation – going back to Goldstine and von Neumann (1951) and strongly advocated by James Demmel (1988) and Steve Smale (1997) among others – was to focus on the behaviour of an algorithm for a *typical input*, rather than for a specific one. Such analysis would allow one to ascertain how well the algorithm behaves in general. To implement this approach, one would select an algorithmic feature to be analysed (say time or loss of precision), obtain bounds for this feature in terms of a relevant condition number, and then, having endowed the space of inputs with a probability measure, derive results for the behaviour above in the form of either expectations or probability tails for the selected algorithmic feature. A recent twist in this programme, known as *smoothed analysis* and championed by Daniel Spielman and Shang-Hua Teng (2002), shifted the focus from a typical input to a *typical small perturbation of an arbitrary input*.

The following pages offer an overview of the development of this programme. Like Julius Caesar's Gaul (book I, ch. 1), this paper is divided into three parts. In Section 1 we give a summary of condition numbers that focuses on how are they defined in various frameworks. In Section 2 we do the same for the two forms of probabilistic analysis we mentioned above. Finally, in Section 3, we provide examples of problems and algorithms for which this condition-based analysis has been implemented.

The subject of this overview has been given a detailed treatment in the recent book by Bürgisser and Cucker (2013), and most of the concepts, methods and results herein can be further explored there. A notable exception is the material in Sections 3.8(g) and 3.9.

1. A brief introduction to condition numbers

1.1. Condition and accuracy

Assume we want to compute a function φ via a computer algorithm $\mathcal{A}^\varphi$. In the simplest possible setting,

$$\varphi : \mathbb{R}^m \to \mathbb{R}^q. \tag{1.1}$$

If computations are going to be done with finite precision only, the point $\widetilde{\varphi(a)}$ returned by $\mathcal{A}^\varphi$ will be just an approximation of $\varphi(a)$, for every $a \in \mathbb{R}^m$. The immediate relevant question is, how good is this approximation?

Leaving aside, for the moment, the fact that to answer this question we need a way of measuring the quality of an approximation, our starting point is the fact that, historically, this problem has been split into two simpler problems. On the one hand, we seek to determine how well the algorithm $\mathcal{A}^\varphi$ computes φ. That is, how badly does $\mathcal{A}^\varphi$ accumulate errors? On the other hand, we attempt to estimate the sensitivity of the data a with respect to φ. That is, how much does $\varphi(a)$ change when a is slightly perturbed? Condition numbers are numerical answers to this last question.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

A fairly general definition of the (absolute) condition number $\mathsf{acond}^{\varphi}(a)$ of a with respect to φ would be

$$\mathsf{acond}^{\varphi}(a) := \lim_{\delta\to 0} \sup_{\mathsf{Error}(\tilde{a})\leq\delta} \frac{\mathsf{Error}(\varphi(\tilde{a}))}{\mathsf{Error}(\tilde{a})}, \tag{1.2}$$

where the numerator and denominator in the fraction measure the size of the error in the output $\varphi(\tilde{a})$ and the input perturbation $\tilde{a}$, respectively. These measures can be componentwise or normwise, and in the latter case, still vary depending on the selected norm. In addition, relative errors can be considered instead of absolute errors, in which case the (relative) condition number $\mathsf{cond}^{\varphi}(a)$ of a with respect to φ takes the form

$$\mathsf{cond}^{\varphi}(a) := \lim_{\delta\to 0} \sup_{\mathsf{RelError}(\tilde{a})\leq\delta} \frac{\mathsf{RelError}(\varphi(\tilde{a}))}{\mathsf{RelError}(\tilde{a})}. \tag{1.3}$$

The ultimate choice for all these possibilities will have to depend on the situation at hand (and the examples in Section 3 below will make this clear). A common feature for all those choices, however, is the presence of *ill-posed* data. These are the elements $a \in \mathbb{R}^m$ for which the limit in (1.2) or (1.3) is infinite, the set of which is usually denoted by Σ.

Some of these choices lead to well-known objects. Thus, in the case when φ is differentiable and we measure the size of both the input perturbation and the error in the output using (Euclidean) norms, the absolute condition number $\mathsf{acond}^{\varphi}(a)$ becomes

$$\mathsf{acond}^{\varphi}(a) = \lim_{\delta\to 0} \sup_{\|a-\tilde{a}\|\leq\delta} \frac{\|\varphi(a)-\varphi(\tilde{a})\|}{\|a-\tilde{a}\|} = \|D\varphi(a)\|,$$

and the corresponding relative condition number is therefore given by

$$\mathsf{cond}^{\varphi}(a) = \|D\varphi(a)\|\frac{\|a\|}{\|\varphi(a)\|}.$$

Condition numbers, however, come in more guises than the equalities above suggest. The simple reason is that not all computational problems consist in computing a differentiable function. We next briefly describe some alternative situations.

(a) Multiple-valued functions. We use this to refer to situations where given input data may have many possible (and equally valid) outputs. For instance, given a complex univariate polynomial P, compute a zero ζ of P. If P has degree d then it has at most d zeros (and, generically, it has exactly d zeros) and any of them is a possible output of a zero-finding algorithm. If $\zeta_1, \ldots, \zeta_d$ are these zeros, then definition (1.2) (or (1.3)) yields a measure of condition $\mathsf{acond}(P, \zeta_i)$ for each one of them. But how does one define the

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

condition of P? There is no canonical way to do so. But three possibilities arise by taking the best, worst, or average condition for P over its zeros.

More generally, for an input a, let $\mathsf{Sol}(a)$ be its associated set of solutions (*i.e.* all the possible outputs for a). If for each $y \in \mathsf{Sol}(a)$ we have a number $\xi(a, y)$ quantifying the condition of a with respect to y, we may define the condition $\xi(a)$ of a by taking any of the following options:

$$\xi(a) := \inf_{y \in \mathsf{Sol}(a)} \xi(a, y), \quad \xi(a) := \mathop{\mathbb{E}}_{y \in \mathsf{Sol}(a)} \xi(a, y), \quad \xi(a) := \sup_{y \in \mathsf{Sol}(a)} \xi(a, y),$$

where the expectation in the middle expression is for some distribution on $\mathsf{Sol}(A)$. Which of the three we choose will depend on the problem at hand, and sometimes on the algorithmic approach we use to solve it.

(b) Condition on manifolds. Another way in which the setting (1.1) may be too restrictive has to do with the spaces where data and outputs live. As well as the obvious fact that φ may be only partially defined, for a number of problems these spaces are manifolds more general than Euclidean spaces. For instance, if (λ, v) is an eigenpair of a matrix $A \in \mathbb{C}^{n \times n}$, the eigenvector v naturally lives in the projective space $\mathbb{P}(\mathbb{C}^n)$, which motivates the following general framework for condition numbers.

Let X be a manifold of inputs, Y a manifold of outputs, and let $V \subseteq X \times Y$ be a submanifold of 'solutions' to some computational problem. We assume that X and V have the same dimension n to guarantee a local uniqueness of solutions. Consider the projection $\pi_1 \colon V \to X, (x, y) \mapsto x$, and its derivative $D\pi_1(x, y) \colon T_{(x,y)}V \to T_x X$, which is the restriction of the projection

$$T_x X \times T_y Y \to T_x X, \quad (\dot{x}, \dot{y}) \mapsto \dot{x},$$

to the subspace $T_{(x,y)}V$. We define the *set of ill-posed solutions* as

$$\Sigma' := \big\{(x, y) \in V \mid \mathsf{rank}\, D\pi_1(x, y) < \dim X\big\}. \tag{1.4}$$

If $(x_0, y_0) \in V \setminus \Sigma'$, then $D\pi_1(x_0, y_0) \colon T_{(x_0,y_0)}V \to T_{x_0}X$ is a linear isomorphism. The implicit function theorem shows that $\pi_1 \colon V \to X$ can be locally inverted around (x_0, y_0). Its inverse $x \mapsto (x, G(x))$ is given by the *solution map* $G \colon X \supseteq U \to Y$ defined on an open neighbourhood U of x_0. So we have $(x, G(x)) \in V$ for all $x \in U$. The derivative

$$DG(x_0) \colon T_{x_0}X \to T_{y_0}Y$$

will be called the *condition map*. The inverse of $D\pi_1(x_0, y_0)$ is given by

$$T_{x_0}X \to T_{(x_0,y_0)}V, \quad \dot{x} \mapsto \big(\dot{x}, DG(x_0)(\dot{x})\big). \tag{1.5}$$

A common situation is that in which V is given as the zero set of a smooth map $F \colon X \times Y \to \mathbb{R}^n$. In this case we have the following characterization of Σ', in terms of the function F itself,

$$\Sigma' = \big\{(x, y) \in V \mid \mathsf{rank}\, \partial F/\partial y(x, y) < n\big\}. \tag{1.6}$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Here the partial derivative $\partial F/\partial y(x,y)\colon T_yY \to \mathbb{R}^n$ is defined as the restriction of $DF(x,y)$ to T_yY.

Now suppose that X and Y are Riemannian manifolds. That is, we have an inner product on each tangent space T_xX and T_yY that varies smoothly with x and y, respectively. In particular, T_xX and T_yY are normed vector spaces. In this case, we may define the (absolute) normwise condition number

$$\mathsf{acond}^G(x_0) := \|DG(x_0)\| = \max_{\|\dot{x}\|=1} \|DG(x_0)(\dot{x})\|$$

as the operator norm of $DG(x_0)$. We note that in this general framework, it does not make sense to define relative condition numbers. However, implicitly, we can model relative notions of condition numbers by choosing the manifolds appropriately. For instance, working with projective spaces means studying ratios, which accounts for a relative notion.

This framework relates to our previous discussion. For an element x_0 in the manifold of inputs X we have finitely many points $(x_0,y_1),\dots,(x_0,y_s)$ in the fibre $\pi_1^{-1}(x_0)$. Each of them has a corresponding solution map G_j, $j=1,\dots,s$, and a condition number $\mathsf{acond}^{G_j}(x_0)$. The condition of x_0 will depend on the computational problem we are considering associated with the geometric situation $V \subseteq X \times Y$. As described in (a) above, the three typical choices are

$$\begin{aligned}
\mathsf{acond}(x_0) &:= \inf_{j\le s} \mathsf{acond}^{G_j}(x_0),\\
\mathsf{acond}(x_0) &:= \mathop{\mathbb{E}}_{j\le s} \mathsf{acond}^{G_j}(x_0),\\
\mathsf{acond}(x_0) &:= \sup_{j\le s} \mathsf{acond}^{G_j}(x_0).
\end{aligned}$$

Note that the last two choices force us to define the *set of ill-posed inputs* to be the set $\Sigma := \pi_1(\Sigma')$, whereas the first choice requires

$$\Sigma := \big\{x \in X \mid (x,y) \in \Sigma' \text{ for all } (x,y) \in \pi^{-1}(x)\big\}.$$

(c) Finite-valued functions. Here we consider a function $\varphi : \mathbb{R}^m \to Y$ where Y is a finite set (say $Y = \{y_1,\dots,y_k\}$). In this situation the quantity $\mathsf{acond}^\varphi(a)$ in (1.2) is of little use. It is immediate to check that $\mathsf{acond}^\varphi(a) = \infty$ when a is in the boundary of some $S_j := \{a \in \mathbb{R}^m \mid \varphi(a) = y_j\}$, for $j=1,\dots,k$, and that $\mathsf{acond}^\varphi(a) = 0$ otherwise.

The union Σ of the boundaries between the sets S_j is composed of data a for which there is no hope that a finite-precision computation with input a will yield a reliable output.

An idea championed by Jim Renegar (1994*b*, 1995*a*, 1995*b*) is to define the condition for finite-valued problems to be the inverse of the distance to

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

ill-posedness, that is,

$$\mathsf{cond}^{\varphi}(a) := \frac{1}{\mathsf{dist}(a, \Sigma)},$$

or a relative inverse distance

$$\mathsf{cond}^{\varphi}(a) := \frac{\|a\|}{\mathsf{dist}(a, \Sigma)} \tag{1.7}$$

if the scaling properties of the problem suggest that this is better, which is often the case. Other approaches to defining conditions for these problems have been used as well.

Notable within the class of finite-valued problems is the subclass of *decision problems*. These consist of a question whose answer is either `Yes` or `No` (*e.g.* Is a symmetric matrix A positive semidefinite? Does a given linear programme have feasible points?) Although less relevant in the context of numerical algorithms (as opposed to their relevance to discrete computations, where they form the backbone of complexity theory) they nonetheless form a class deserving analysis.

1.2. Condition and complexity

The first problem for which a condition number was considered was matrix inversion (von Neumann and Goldstine 1947, Turing 1948). For an $n \times n$ matrix A, the relative condition number, usually denoted by $\kappa(A)$, was characterized as $\kappa(A) = \|A\| \, \|A^{-1}\|$. Shortly thereafter, this quantity occurred as a parameter in the complexity analysis of the conjugate gradient method. This is an iterative algorithm to calculate the solution x^* of the linear system $Ax = b$, where A is a symmetric $n \times n$ positive definite matrix and $b \in \mathbb{R}^n$, given an initial approximation x_0. Let x_k denote the approximation to x^* after k iterations of the algorithm and let $\varepsilon_k := \|x_k - x^*\|_A$ be the error of this approximation (with respect to a certain A-norm: see (3.13)). Then (and we will return to this in more detail in Section 3.4), the number of arithmetic operations – the *cost* of the execution with input matrix A – to ensure that $\varepsilon_k \leq \delta \varepsilon_0$ for a given $\delta > 0$ can be shown to be $O(n^2 \sqrt{\kappa(A)} \ln(1/\delta))$.

This situation – the fact that the cost of an iterative algorithm, even assuming infinite precision, depends on the condition number of the input – turned out to be pervasive.

2. A brief introduction to probabilistic analysis

A general feeling among researchers is that computing a condition number is no easier than computing the corresponding function. An attempt to make this feeling precise was given by Renegar (1994*a*). A different approach (Cheung and Cucker 2005) showed that this is the case for all condition

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

numbers given by (1.7) – which we will call *conic*. The value of condition numbers thus appears to be questionable: they provide useful information on the difficulty of computing a function but they are at least as difficult to compute as the function itself.

A first way out of this quandary was proposed almost as early as condition numbers themselves. Goldstine and von Neumann (1951), in a sequel to their founding paper (von Neumann and Goldstine 1947), endowed the set of $n \times n$ real matrices with a probability measure and estimated the probability that 'an approximate inverse' is found; they concluded that this 'will usually' be the case provided n is not too large. The particular bounds for n (as a function of the desired precision) are of no consequence here; these bounds have been much improved since then. The overall idea, in contrast, is of paramount importance. Instead of focusing on the difficulty of particular instances of a problem (*e.g.* specific $n \times n$ matrices), Goldstine and von Neumann proposed that one should attempt to understand how difficult the problem is in general (*e.g.* how difficult it is to invert general $n \times n$ matrices). In other words, instead of computing condition numbers for a specific instance, one aims at probabilistic estimates for the condition number of random instances. These estimates usually take the form of either an expectation or a probability tail, and a sense of the difficulty of the problem is derived from their order of growth with respect to the size or dimension of the problem (the integer n in the example above). That the latter was the goal was clear in the paper of Goldstine and von Neumann. They write that 'the coefficients ... are probably not optimal, but result from the crudeness of our probabilistic techniques. However, our principal aim was not to produce "best possible" estimates, but was merely to obtain a feeling for the sizes of the errors likely to be encountered' (Goldstine and von Neumann 1951, p. 201).

The approach outlined above received a strong push in the work of Jim Demmel in the late 1980s. First (see Demmel 1987) he showed that for many problems, the condition number defined in (1.3) can be either characterized by the expression in (1.7) or bounded by a small multiple of this expression. The import of this paper was thus that many problems admit conic condition numbers. Then (see Demmel 1988) he gave bounds for the expectation of conic condition numbers. These bounds were general and expressed in terms of the dimension of the ambient space and the degree and codimension of the set Σ. We will return to this subject, and give more details, in Section 3.10.

Demmel's work was motivated by issues relating to finite precision. But the idea of understanding the 'typical' behaviour of an algorithm through a probabilistic analysis of its condition number naturally applies to complexity as well. Steve Smale (1997) proposed a general approach to the complexity of numerical problems. It consisted of first providing bounds for an algorithm's cost in terms of both the size and condition of the input,

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

and then removing the condition from these bounds by replacing them with probabilistic estimates.

Whether for accuracy or complexity analyses, probabilistic estimates will rely on the underlying probability measure assumed. In addition, they depend on what exactly the randomization of the input is meant to capture.

The current literature exhibits two different ideas for the latter. In the first one, called *average analysis*, data are assumed to be drawn from 'evenly spread' distributions. The analysis performed for such a randomization is meant to describe the behaviour of the analysed quantity for 'typical' data.

The second idea for randomization, known as *smoothed analysis*, replaces this average data with a small random perturbation of worst-case data. That is, it considers an arbitrary element $\overline{a}$ in $\mathbb{R}^m$ (and thus, in particular, the instance at hand) and assumes that $\overline{a}$ is affected by random noise. The distribution for this perturbed input is usually taken to be centred and isotropic around $\overline{a}$, and assumed to have a small variance.

(a) Average analysis. Most of the condition numbers occurring in the literature are *scale-invariant*, that is, they satisfy $\mathsf{cond}(a) = \mathsf{cond}(\lambda a)$ for all $a \in \mathbb{R}^m$ and $\lambda \in \mathbb{R} \setminus \{0\}$. We will therefore confine our discussion to scale-invariant random variables.

We have already mentioned that the underlying distribution in average analysis is 'evenly spread' and that expected values for such distributions are meant to capture the behaviour of the random variable on the 'average Joe' inhabitant of the data space. In the context we are interested in, the probabilistic analysis of condition numbers, this data space is usually a Euclidean space $\mathbb{R}^m$. So the question that poses itself is with which distribution should we endow $\mathbb{R}^m$ for the analysis. Assume we have a 'natural' system of coordinates in $\mathbb{R}^m$. A first step towards an answer consists in noting that such a distribution should be rotationally invariant. The undistinguished character of the 'average Joe' cannot favour any particular direction. Furthermore, scale invariance allows us to take a second step. Indeed, for such a function of $\mathbb{R}^m$ the value of the density along a half-line with origin at 0 is not relevant. It follows that we can take *any* rotationally invariant distribution on $\mathbb{R}^m$, and the collection of remarkable features of the standard Gaussian makes this distribution the obvious choice.

Scale invariance also suggests the choice of an underlying distribution that is evenly spread by definition, namely, the uniform distribution on the unit sphere $\mathbb{S}^{m-1}$ (this requires us to consider data on the sphere only).

All the above, however, is subject to a 'natural' system of coordinates in $\mathbb{R}^m$. And while some situations may suggest such a system, its choice remains a bone of contention for average-case analysis. The most common objection to average-case analysis is that its underlying probability measures may not accurately reflect the distributions occurring in practice, in

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

particular, that they may be 'optimistic' in the sense that they may put more probability mass on the instances for which the values of the function ψ under consideration are small. Such optimism would produce an expectation $\mathbb{E}\psi$ smaller than the true one.

(b) Smoothed analysis. Smoothed analysis was introduced mainly to overcome this objection. Its underlying idea, we recall, was to look at the behaviour of a function for small perturbations of arbitrary data. In the case of a condition number, this amounts to understanding the condition of slight perturbations of ill-posed data. When compared with the average analysis, it replaces the goal of showing that

> for a random a, it is unlikely that $\mathsf{cond}(a)$ will be large

by the following one:

> for all $\overline{a}$, it is unlikely that a slight random perturbation $\overline{a} + \Delta a$ of $\overline{a}$ will have $\mathsf{cond}(\overline{a} + \Delta a)$ large.

To perform a smoothed analysis, a family of distributions (parametrized by a parameter r controlling the size of the perturbation) is considered with the following characteristics:

(a) the density at a point a depends only on the distance $\|a - \overline{a}\|$,

(b) the value of r is closely related to the variance of $\|a - \overline{a}\|$.

A first possible choice for this family of distributions is the set of Gaussians $N(\overline{a}, \sigma^2 \operatorname{Id}_m)$. The role of r is in this case played by $\sigma > 0$. Because of scale invariance, one usually assumes that the base point $\overline{a}$ lives in $\mathbb{S}^{m-1}$. In this way, the value of the parameter σ controlling the size of the perturbations is directly comparable with the size of $\overline{a}$.

A different choice of distributions consists in taking $\overline{a} \in \mathbb{S}^{m-1}$ and, for each $r \in (0, \pi]$, the uniform measure on the spherical cap,

$$\mathsf{cap}(\overline{a}, r) := \{x \in \mathbb{S}^{m-1} \mid \langle x, \overline{a} \rangle \geq \cos r\}.$$

Table 2.1 shows a schematic comparison of the quantities computed in worst-case, average-case, and smoothed analyses for a scale-invariant function $\psi : \mathbb{R}^m \to \mathbb{R}$ in both the uniform and Gaussian cases. Usually, the quantities estimated in the first two columns are functions of m. For the estimate in a smoothed analysis there is, in addition, a dependence on r or σ. This dependence appears to interpolate between worst-case and average-case. Indeed, in the uniform case, when r approaches 0, the value of

$$\sup_{\overline{a} \in \mathbb{S}^{m-1}} \mathop{\mathbb{E}}_{a \sim U(\mathsf{cap}(\overline{a}, r))} \psi(a)$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Table 2.1.

Worst-case	Average-case	Smoothed
$\sup_{a\in\mathbb{S}^{m-1}} \psi(a)$	$\mathop{\mathbb{E}}_{a\sim U(\mathbb{S}^{m-1})} \psi(a)$	$\sup_{\overline{a}\in\mathbb{S}^{m-1}} \mathop{\mathbb{E}}_{a\sim U(\mathsf{cap}(\overline{a},r))} \psi(a)$
$\sup_{a\in\mathbb{S}^{m-1}} \psi(a)$	$\mathop{\mathbb{E}}_{a\sim N(0,\mathrm{Id}_m)} \psi(a)$	$\sup_{\overline{a}\in\mathbb{S}^{m-1}} \mathop{\mathbb{E}}_{a\sim N(\overline{a},\sigma^2\mathrm{Id}_m)} \psi(a)$

approaches $\sup_{a\in\mathbb{R}^m} \psi(a)$, while when $r = \pi$ it equals $\mathbb{E}_{a\sim U(\mathbb{S}^{m-1})}\psi(a)$ (since $\mathsf{cap}(\overline{a},\pi) = \mathbb{S}^{m-1}$ for all $\overline{a}$). A similar situation holds in the Gaussian framework.

The local nature of randomization in smoothed analysis, coupled with its worst-case dependence on the input data, removes from smoothed analysis the objection to average-case analysis mentioned above. A satisfying result in this context (usually a low-degree polynomial bound in the input size m and in the inverse of the dispersion parameter r) is consequently considered a more reliable indication that one may expect low values for the function ψ in practice. In addition, there is an emerging impression that smoothed analysis is robust in the sense that its dependence on the chosen family of measures is low.

Notation. In all that follows, we will omit the subindex m denoting dimension in Id_m. These dimensions are always clear from the context.

3. A repertoire of examples

In this last section – the longest in this overview – we summarize for a number of examples the role played by a condition number in the evaluation of some measure of algorithmic performance, and the way a probabilistic analysis (average or smoothed) allows for a direct assessment of this evaluation. The goal is not to prove all, or even most, of the results described (due to obvious space limitations), let alone to prove the sharpest possible results (for most of them such a proof would be too long, and interesting to only a few specialists). Instead we want to offer a catalogue as broad as possible of instances of the general framework sketched in the preceding sections. To do so, we have chosen to give full proofs for a few results we believe foundational, to sketch the proofs of some recent breakthroughs, and to simply state, omitting proofs, some state-of-the-art technical results. The division between these three categories is, needless to say, a personal choice.

In all cases we indicate references in the literature where notions have been developed, algorithms designed, and theorems established. As well

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

as these references, a recurring source is the recent book by Bürgisser and Cucker (2013), where most of the themes we deal with in this overview are given full-length discussion.

The problems of inverting matrices or solving linear systems engendered condition numbers; a lively narrated story of the events that led Turing to his pioneer paper (Turing 1948) is given by Wilkinson (1971). The same reasons (I will not spell them out here) also explain why the condition number for these problems is ubiquitous in the literature – and why we will devote some time to it.

3.1. Linear systems

(a) Loss of precision. Assume we are solving the system $Ax = b$ (or inverting the matrix A) with a real-number algorithm $\mathcal{A}$ executed using floating-point arithmetic, and let ϵ_{mach} be the epsilon-machine (or round-off unit) at hand (for a primer on finite-precision computations see Higham 1996, Chapters 2 and 3). This means that all floating-point numbers occurring during the execution of $\mathcal{A}$ are rounded to $|\log \epsilon_{\mathsf{mach}}|$ significant digits (here log refers to logarithm in base 10). In particular, this is the *precision* of the entries in the data (A, b). On the other hand, the number of correct digits in the ith component of the computed solution $\tilde{x}$ is approximately $-\log \mathsf{RelError}(\tilde{x}_i)$, where

$$\mathsf{RelError}(\tilde{x}_i) := \frac{\mathsf{Error}(\tilde{x}_i)}{|x_i|} = \frac{|x_i - \tilde{x}_i|}{|x_i|}.$$

Therefore, the value

$$\mathsf{LoP}(x_i) := \log \frac{\mathsf{RelError}(\tilde{x}_i)}{\epsilon_{\mathsf{mach}}} = |\log \epsilon_{\mathsf{mach}}| - |\log \mathsf{RelError}(\tilde{x}_i)|$$

quantifies the *loss of precision* in the computation of x_i. If we take the precision of $\tilde{x}$ to be

$$-\log \mathsf{RelError}(\tilde{x}) := -\log \frac{\|x - \tilde{x}\|}{\|x\|},$$

then the loss of precision in the computation of x is

$$\mathsf{LoP}(x) := \log \frac{\mathsf{RelError}(\tilde{x})}{\epsilon_{\mathsf{mach}}} = |\log \epsilon_{\mathsf{mach}}| + \log \mathsf{RelError}(\tilde{x}). \tag{3.1}$$

Next assume that algorithm $\mathcal{A}$ corresponds to computing x via the *Householder QR factorization method*. The following result is a well-known backward error analysis for this algorithm (see Higham 1996, Theorem 19.3).

In all that follows, for any square matrix M, $\|M\|_F$ and $\|M\|$ denote the Frobenius and spectral norms of M, respectively.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Theorem 3.1. Let $A \in \mathbb{R}^{n\times n}$ be invertible and $b \in \mathbb{R}^n$. If the system $Ax = b$ is solved using the Householder QR factorization method, then the computed solution $\tilde{x}$ satisfies

$$\tilde{A}\tilde{x} = \tilde{b},$$

where $\tilde{A}$ and $\tilde{b}$ satisfy the relative error bounds

$$\|\tilde{A} - A\|_F \leq n\gamma_{sn}\|A\|_F \quad \text{and} \quad \|\tilde{b} - b\| \leq n\gamma_{sn}\|b\|$$

for a small constant s and with $\gamma_k := k\epsilon_{\mathsf{mach}}/(1 - k\epsilon_{\mathsf{mach}})$. □

We can tie this result to the general discussion on condition numbers in Section 1.1. To do so, we define the *set of ill-posed matrices*

$$\Sigma := \{A \in \mathbb{R}^{n\times n} \mid \det(A) = 0\},$$

write $\mathcal{D} := \mathbb{R}^{n\times n} \setminus \Sigma$ and let $\varphi : \mathcal{D} \times \mathbb{R}^n \to \mathbb{R}^n$ be given by $\varphi(A, b) = A^{-1}b$. Taking

$$\mathsf{RelError}(\tilde{A}, \tilde{b}) := \max\left\{\frac{\|\tilde{A} - A\|}{\|A\|}, \frac{\|\tilde{b} - b\|}{\|b\|}\right\}, \tag{3.2}$$

Theorem 3.1 (and the bound $\|\cdot\|_F \leq \sqrt{n}\|\cdot\|$) shows that $\tilde{x} = \varphi(\tilde{A}, \tilde{b})$ with

$$\mathsf{RelError}(\tilde{A}, \tilde{b}) \leq n^{3/2}\gamma_{sn}.$$

We now fix a normwise measure for the relative error of $\tilde{x}$ (*i.e.* $\mathsf{RelError}(\tilde{x}) := \|\tilde{x} - x\|/\|x\|$), consider the relative condition number $\mathsf{cond}^{\varphi}(A, b)$ with respect to the relative errors in both the input (A, b) and the solution x, and set $\epsilon_{\mathsf{mach}} \to 0$. Then, $\gamma_k = k\epsilon_{\mathsf{mach}} + o(\epsilon_{\mathsf{mach}})$ and (recall definition (1.3))

$$\mathsf{RelError}(\tilde{x}) \leq \mathsf{RelError}(\tilde{A}, \tilde{b})\,\mathsf{cond}^{\varphi}(A, b) + o(\epsilon_{\mathsf{mach}}).$$

We conclude that this relative error satisfies

$$\frac{\|\tilde{x} - x\|}{\|x\|} \leq sn^{5/2}\epsilon_{\mathsf{mach}}\mathsf{cond}^{\varphi}(A, b) + o(\epsilon_{\mathsf{mach}}), \tag{3.3}$$

and consequently, that the loss of precision is bounded by

$$\mathsf{LoP}(A^{-1}b) \leq \frac{5}{2}\log n + \log \mathsf{cond}(A, b) + \log s + o(1). \tag{3.4}$$

(b) Characterizations of $\mathsf{cond}^{\varphi}(A, b)$. We begin by proving an assertion we claimed early in Section 1.2. Define the map $\kappa : \mathcal{D} \to \mathbb{R}$ by

$$\kappa(A) := \|A\|\|A^{-1}\|. \tag{3.5}$$

Then $\kappa(A) \geq 1$, since $1 = \|\mathrm{Id}\| \leq \|A\|\|A^{-1}\| = \kappa(A)$.

Theorem 3.2. Let $\mathsf{cond}^{\varphi}(A, b)$ be as above. Then,

$$\mathsf{cond}^{\varphi}(A, b) = \kappa(A) + \frac{\|A^{-1}\|\|b\|}{\|A^{-1}b\|}.$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

In particular, we have

$$\kappa(A) \leq \mathsf{cond}^{\varphi}(A,b) \leq 2\kappa(A).$$

Proof. Let $\tilde{A} = A - E$ and $\tilde{b} = b + f$. By definition, $\|E\| \leq \mathsf{R}\|A\|$ and $\|f\| \leq \mathsf{R}\|b\|$, where for simplicity, $\mathsf{R} = \mathsf{RelError}(A,b)$. We have, for $\mathsf{R} \to 0$,

$$\begin{aligned}(A-E)^{-1} &= A^{-1}(\mathrm{Id} - EA^{-1})^{-1} = A^{-1}\big(\mathrm{Id} + EA^{-1} + o(\mathsf{R})\big)\\ &= A^{-1} + A^{-1}EA^{-1} + o(\mathsf{R}).\end{aligned}$$

This implies, writing $x := A^{-1}b$ and $\tilde{x} := \tilde{A}^{-1}\tilde{b}$,

$$\tilde{x} - x = (A-E)^{-1}(b+f) - x = A^{-1}Ex + A^{-1}f + o(\mathsf{R}). \tag{3.6}$$

Taking norms we conclude that

$$\begin{aligned}\|\tilde{x} - x\| &\leq \|A^{-1}E\|\|x\| + \|A^{-1}\|\|f\| + o(\mathsf{R})\\ &\leq \|A^{-1}\|\|E\|\|x\| + \|A^{-1}\|\|f\| + o(\mathsf{R})\\ &\leq \|A^{-1}\|\|A\|\|x\|\mathsf{R} + \|A^{-1}\|\|b\|\mathsf{R} + o(\mathsf{R}),\end{aligned}$$

and hence

$$\frac{\|\tilde{x} - x\|}{\mathsf{R}\|x\|} \leq \kappa(A) + \frac{\|A^{-1}\|\|b\|}{\|x\|},$$

which shows the upper bound in the claimed equality.

For the corresponding lower bound we choose $y \in \mathbb{R}^n$ such that $\|y\| = 1$ and $\|A^{-1}y\| = \|A^{-1}\|$. Further, we choose $v = x/\|x\| \in \mathbb{R}^n$ so that $\|v\| = 1$ and $v^{\mathrm{T}}x = \|x\|$. Now we put

$$E := \mathsf{R}\|A\|\, yv^{\mathrm{T}}, \quad f := \pm\mathsf{R}\|b\|\, y. \tag{3.7}$$

We note that

$$\|E\| = \mathsf{R}\|A\|, \quad \|f\| = \mathsf{R}\|b\|,$$

the first equality holding since $\|yv^{\mathrm{T}}\| = \|y\|\|v\| = 1$. We have

$$A^{-1}Ex = \mathsf{R}\|A\|\, A^{-1}y\, v^{\mathrm{T}}x = \mathsf{R}\|A\|\|x\|A^{-1}y$$

and hence $\|A^{-1}Ex\| = \kappa(A)\|x\|\mathsf{R}$. Similarly, $A^{-1}f = \pm R\|b\|A^{-1}y$ and $\|A^{-1}f\| = \|A^{-1}\|\|b\|\mathsf{R}$. Since $A^{-1}Ex$ and $A^{-1}f$ are both proportional to $A^{-1}y$, we obtain from (3.6)

$$\|\tilde{x} - x\| = \kappa(A)\|x\|\mathsf{R} + \|A^{-1}\|\|b\|\mathsf{R},$$

if we choose the sign for f in (3.7) appropriately. This proves the claimed lower bound. □

The next result shows that κ actually coincides with the condition number for the problem of matrix inversion.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Theorem 3.3. Let $\psi\colon \mathcal{D} \to \mathbb{R}^{n\times n}$ be given by $\psi(A) = A^{-1}$. We measure the relative error on the data space and solution space with respect to spectral and Euclidean norms, respectively. Then we have

$$\mathsf{cond}^{\psi}(A) = \kappa(A).$$

Proof. Let $E \in \mathbb{R}^{n\times n}$ be such that $\tilde{A} = A - E$. Then $\mathsf{RelError}(A) = \|E\|/\|A\|$. As in the proof of Theorem 3.2, we have for $\|E\| \to 0$,

$$\|\tilde{A}^{-1} - A^{-1}\| = \|A^{-1}EA^{-1}\| + o\big(\|E\|\big). \tag{3.8}$$

Hence, $\|A^{-1}EA^{-1}\| \le \|A^{-1}\|\|E\|A^{-1}\|$. Consequently, we obtain

$$\mathsf{RelError}(A^{-1}) = \frac{\|\tilde{A}^{-1} - A^{-1}\|}{\|A^{-1}\|} \le \|A^{-1}\|\|E\| + o\big(\|E\|\big)\big).$$

We conclude that

$$\frac{\mathsf{RelError}(A^{-1})}{\mathsf{RelError}(A)} \le \|A\|\|A^{-1}\| + o(1)$$

and hence $\mathsf{cond}^{\psi}(A) \le \kappa(A)$.

To prove the reverse inequality it is enough to find arbitrarily small matrices E such that $\|A^{-1}EA^{-1}\| = \|A^{-1}\|^2\|E\|$, since then we can proceed from (3.8) as we did in Theorem 3.2 from (3.6).

To do so, let $y \in \mathbb{R}^n$ be such that $\|y\| = 1$ and $\|A^{-1}y\| = \|A^{-1}\|$. Define $x := \|A^{-1}\|^{-1}A^{-1}y$, so that $A^{-1}y = \|A^{-1}\|x$ and $\|x\| = \|y\| = 1$. For any $B \in \mathbb{R}^{n\times n}$ we have

$$\|A^{-1}BA^{-1}\| \ge \|A^{-1}BA^{-1}y\| = \|A^{-1}\|\cdot\|A^{-1}Bx\|.$$

Since $\|x\| = \|y\| = 1$ there exists an orthogonal matrix $B \in \mathbb{R}^{n\times n}$ such that $Bx = y$. Therefore,

$$\|A^{-1}BA^{-1}\| \ge \|A^{-1}\|\cdot\|A^{-1}y\| = \|A^{-1}\|^2.$$

Taking $E = \delta B$ with arbitrarily small δ finishes the proof. □

(c) The condition number theorem. We mentioned in Section 2 that Demmel (1987) showed that many problems admit conic condition numbers. The first result of this kind, which preceded the expression 'condition number', was proved by Eckart and Young (1936). It states that for $A \in \mathbb{R}^{n\times n} \setminus \Sigma$, we have

$$\|A^{-1}\| = \frac{1}{\mathsf{dist}(A, \Sigma)} \tag{3.9}$$

where dist is the distance induced by either the spectral or the Frobenius norm. It follows that

$$\kappa(A) = \frac{\|A\|}{\mathsf{dist}(A, \Sigma)}.$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

(d) Probabilistic analyses. Now we prove an average analysis for $\log \kappa(A)$. The bounds we obtain are not the sharpest possible (see Theorem 3.12) but we can use them to derive sufficiently good bounds for the average loss of precision.

For an $n \times n$ matrix A, let $\hat{A}_i$ refer to A with the ith column removed and, for an $n \times (n-1)$ matrix A and $b \in \mathbb{R}^n$, let $(A, b)_i$ refer to the $n \times n$ matrix obtained from A by inserting b as the ith column. Also, let us denote by A_i the ith column of A.

The next lemma relates the norm of the inverse of an $n \times n$ matrix to a geometric condition on the columns of the matrix.

Lemma 3.4. Let A be an invertible $n \times n$ matrix. If $\|A^{-1}\| = x$, then there is an $i \in \{1, 2, \ldots, n\}$ and a vector $B_i \in \mathbb{R}^n$ with $\|B_i\| \leq \sqrt{n}/x$ and such that $A_i + B_i$ is in the space spanned by $A_j, j \neq i, j \in \{1, \ldots, n\}$.

Proof. That $\|A^{-1}\| = x$ means that there is a unit vector $b \in \mathbb{R}^n$ such that $v = A^{-1}b$ and $\|v\| = x$. Choose i such that $|v_i| \geq x/\sqrt{n}$. By Cramer's rule

$$v_i = \frac{\det(\hat{A}_i, b)_i}{\det A}.$$

Hence $\det(\hat{A}_i, {v_i}^{-1}b)_i = \det A$ and $\det(\hat{A}_i, A_i - {v_i}^{-1}b)_i = 0$. As the columns of A are independent, this implies that $A_i - {v_i}^{-1}b$ is spanned by the other columns. Also, $\|{v_i}^{-1}b\| \leq \sqrt{n}/x$ so the statement holds for $B_i := -{v_i}^{-1}b$. □

We shall let $\mathcal{M}_{(n,m)}$ denote the space of real $n \times m$ matrices, and if $m = n$ we shall denote this space simply by $\mathcal{M}_n$. Let $\mathbb{S}^{n^2-1} \subset \mathcal{M}_n$ be the unit sphere with respect to the Frobenius norm. Also, let $\mathbb{B}^{n^2-n}$ be the unit ball in the space $\mathcal{M}_{(n,n-1)}$ and for $i = 1, \ldots, n$ the map

$$\begin{array}{rccc} \pi_i : & \mathcal{M}_n & \to & \mathcal{M}_{(n,n-1)} \\ & A & & \hat{A}_i. \end{array}$$

For any $\varepsilon > 0$ we consider $N_\varepsilon \subset \mathbb{S}^{n^2-1}$ the set of $A \in \mathbb{S}^{n^2-1}$ such that A is not invertible or $\|A^{-1}\| > 1/\varepsilon$. Equation (3.9) provides a geometric characterization of N_ε, namely

$$N_\varepsilon = \{A \in \mathbb{S}^{n^2-1} \mid \mathsf{dist}_F(A, \Sigma) < \varepsilon\}$$

where dist_F is the distance in $\mathcal{M}_n$ induced by the Frobenius norm.

Our goal is to estimate the volume of N_ε. To do so, consider for $\varepsilon > 0$ and $i = 1, \ldots, n$ the set

$$W_{\varepsilon,i} = \{A \in \mathbb{S}^{n^2-1} \mid A_i \text{ is within } \sqrt{n}\,\varepsilon \text{ of the space spanned by the columns of } \hat{A}_i\}.$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Lemma 3.5. For all $\varepsilon > 0$,

$$\mathsf{vol}(N_\varepsilon) \leq \sum_{i=1}^{n} \mathsf{vol}(W_{\varepsilon,i}).$$

Proof. By Lemma 3.4 we have that $N_\varepsilon \subset \bigcup_{i=1}^{n} W_{\varepsilon,i}$. □

Now we turn to $W_{\varepsilon,i}$.

Proposition 3.6. For all $\varepsilon > 0$, $B \in \mathcal{M}_{(n,n-1)}$, and $i = 1, \ldots, n$, we have

$$\mathsf{vol}(W_{\varepsilon,i} \cap \pi_i^{-1} B) \leq 2\sqrt{n}\,\varepsilon\, \mathsf{vol}\Big(\mathbb{S}^{n-2}_{(1-\|B\|_F^2)^{1/2}}\Big),$$

where $\mathbb{S}^{m-1}_r$ is the sphere of radius r in $\mathbb{R}^m$.

Proof. If $A \in W_{\varepsilon,i} \cap \pi_i^{-1} B$ then $\|A_i\| = (1 - \|B\|_F^2)^{1/2}$, that is,

$$A_i \in \mathbb{S}^{n-1}_{(1-\|B\|_F^2)^{1/2}},$$

and A_i is within $\sqrt{n}\varepsilon$ of the space spanned by the columns of B.

The volume of $W_{\varepsilon,i} \cap \pi_i^{-1} B$ is thus at most the volume of the cylinder of base $\mathbb{S}^{n-2}_{(1-\|B\|_F^2)^{1/2}}$ and height $2\sqrt{n}\,\varepsilon$. □

Lemma 3.7. Let $\mathbb{B}^{m-\ell} \subset \mathbb{R}^{m-\ell}$ be the unit ball and $k \geq 0$. Then

$$\frac{1}{\mathsf{vol}(\mathbb{S}^{m-1})} \int_{x \in \mathbb{B}^{m-\ell}} \frac{\mathsf{vol}\big(\mathbb{S}^k_{(1-\|x\|^2)^{1/2}}\big)}{(1-\|x\|^2)^{1/2}}\, \mathrm{d}x_1 \ldots \mathrm{d}x_{m-\ell} = \frac{\pi^{\frac{k-\ell+1}{2}} \Gamma(\frac{m}{2})}{\Gamma(\frac{m+k-\ell+1}{2})}.$$

Proof. The left-hand side in the statement equals

$$\frac{\mathsf{vol}(\mathbb{S}^k)}{\mathsf{vol}(\mathbb{S}^{m-1})} \int_{x \in \mathbb{B}^{m-\ell}} (1-\|x\|^2)^{\frac{k-1}{2}}\, \mathrm{d}x_1 \ldots \mathrm{d}x_{m-\ell},$$

which using spherical coordinates equals

$$\frac{\mathsf{vol}(\mathbb{S}^k)\, \mathsf{vol}(\mathbb{S}^{m-\ell-1})}{\mathsf{vol}(\mathbb{S}^{m-1})} \int_0^1 (1-r^2)^{\frac{k-1}{2}} \cdot r^{m-\ell-1}\, \mathrm{d}r.$$

Finally, using that for $p, q \in \mathbb{R}$,

$$\int_0^1 t^p (1-t^2)^q\, \mathrm{d}t = \frac{1}{2} \frac{\Gamma(\frac{p+1}{2}) \Gamma(q+1)}{\Gamma(\frac{p+1}{2} + q + 1)},$$

the fact that

$$\mathsf{vol}(\mathbb{S}^{p-1}) = \frac{2\pi^{p/2}}{\Gamma(\frac{p}{2})},$$

and some elementary computations, we obtain the right-hand side. □

We can now estimate the volume of N_ε.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Proposition 3.8. For any $\varepsilon > 0$ and any $n \geq 2$,

$$\frac{\mathsf{vol}(N_\varepsilon)}{\mathsf{vol}(\mathbb{S}^{n^2-1})} \leq \varepsilon n^{5/2}.$$

Proof. By Lemma 3.5 it suffices to show that

$$\frac{\mathsf{vol}(W_{\varepsilon,i})}{\mathsf{vol}(\mathbb{S}^{n^2-1})} \leq \varepsilon n^{3/2}.$$

Let $\chi_{W_{\varepsilon,i}}$ be the characteristic function of $W_{\varepsilon,i}$ and let $\pi\colon \mathbb{S}^{n^2-1} \to \mathbb{B}^{n^2-n}$ be the restriction of the orthogonal projection. Then

$$\begin{aligned}
&\frac{\mathsf{vol}(W_{\varepsilon,i})}{\mathsf{vol}(\mathbb{S}^{n^2-1})} \\
&\quad = \frac{1}{\mathsf{vol}(\mathbb{S}^{n^2-1})} \int_{A\in\mathbb{S}^{n^2-1}} \chi_{W_{\varepsilon,i}}(A)\,\mathrm{d}A \\
&\quad = \frac{1}{\mathsf{vol}(\mathbb{S}^{n^2-1})} \int_{B\in\mathbb{B}^{n^2-n}} \int_{Q\in\pi^{-1}(B)} \chi_{W_{\varepsilon,i}}(Q)(1-\|B\|_F^2)^{-1/2}\,\mathrm{d}Q\,\mathrm{d}B \\
&\quad = \frac{1}{\mathsf{vol}(\mathbb{S}^{n^2-1})} \int_{B\in\mathbb{B}^{n^2-n}} \mathsf{vol}(W_{\varepsilon,i}\cap\pi_i^{-1}(B))(1-\|B\|_F^2)^{-1/2}\,\mathrm{d}B \\
&\quad \leq \frac{2\sqrt{n}\varepsilon}{\mathsf{vol}(\mathbb{S}^{n^2-1})} \int_{B\in\mathbb{B}^{n^2-n}} \mathsf{vol}\Big(\mathbb{S}^{n-2}_{(1-\|B\|_F^2)^{1/2}}\Big)(1-\|B\|_F^2)^{-1/2}\,\mathrm{d}B,
\end{aligned}$$

the second equality by the transformation formula (using that the normal Jacobian of π at A is $(1-\|\pi(A)\|_F^2)^{-1/2}$), and the inequality by Proposition 3.6. Using Lemma 3.7 (with $m = n^2$, $k = n-2$ and $\ell = n$) we deduce that the last line is at most

$$\frac{2\sqrt{n}\varepsilon}{\pi^{1/2}} \frac{\Gamma(\frac{n^2}{2})}{\Gamma(\frac{n^2-1}{2})} \leq \frac{2^{1/2}}{\pi^{1/2}}\varepsilon n^{3/2} \leq \varepsilon n^{3/2},$$

the first inequality since

$$\frac{\Gamma(x+1/2)}{\sqrt{x}\Gamma(x)} < 1, \quad \text{for all } x > 0. \qquad \square$$

Theorem 3.9. The probability that $\kappa(A) > 1/\varepsilon$ for the standard Gaussian probability distribution $N(0,\mathrm{Id})$ on $\mathcal{M}_n$ is less than or equal to $n^{5/2}\varepsilon$. Also,

$$\mathop{\mathbb{E}}_{A\sim N(0,\mathrm{Id})} \log\kappa(A) \leq \frac{5}{2}\log n + O(1).$$

Proof. For the first statement in the theorem, define a second condition number κ_F of $n\times n$ matrices by

$$\kappa_F(A) = \|A^{-1}\|\,\|A\|_F.$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Clearly, $\kappa(A) \leq \kappa_F(A)$. Hence, it suffices to prove the statement for κ_F. Also, since the latter is homogeneous of degree 0, it suffices to prove the theorem for the uniform distribution on the sphere. And finally, since N_ε is the set of $A \in \mathbb{S}^{n^2-1}$ where $\mu(A) \geq 1/\varepsilon$, Proposition 3.8 gives the desired bound.

The bound for the expectation follows from integrating the bound for the probability tail. □

The same ideas allow one to derive bounds for the tail and expectation of the condition number of complex matrices. The set $N_\varepsilon^{\mathbb{C}}$ is defined as in the real case, and one can prove that

$$\frac{\mathsf{vol}(N_\varepsilon^{\mathbb{C}})}{\mathsf{vol}(\mathbb{S}^{2n^2-1})} \leq n^4\varepsilon^2.$$

The following result for the set $\mathcal{M}_n^{\mathbb{C}}$ of complex $n \times n$ matrices is an easy corollary of this bound.

Theorem 3.10. The probability that $\mu(A) > 1/\varepsilon$ or that $\kappa(A) > 1/\varepsilon$ for either the standard Gaussian probability distribution on $\mathcal{M}_n^{\mathbb{C}}$ or the uniform distribution on $\mathbb{S}^{2n^2-1}$ (the unit sphere in $\mathcal{M}_n^{\mathbb{C}}$) is less than or equal to $n^4\varepsilon^2$. Also,

$$\mathop{\mathbb{E}}_{A \sim N_{\mathbb{C}}(0,\mathrm{Id})} \log \kappa(A) \leq 2 \log n + O(1). \qquad \square$$

(e) Average loss of precision in linear equation solving. The bound in (3.4) and Theorem 3.9 immediately yield the following result on the expected loss of precision in linear equation solving.

Theorem 3.11. For all $b \in \mathbb{R}^n$,

$$\mathbb{E}_{A \sim N(0,\mathrm{Id})} \mathsf{LoP}(A, b) \leq 5 \log n + O(1). \qquad \square$$

(f) Additional results. The condition number κ is doubtless better studied than any other condition number. Because of this, the content of the previous pages is just a small fraction of what is known about them. Without intending to fill this gap entirely, we next state two results that either sharpen or add to this content.

The first, by Alan Edelman (1988), shows a sharp bound for $\mathbb{E} \log \kappa(A)$.

Theorem 3.12. For both real and complex standard Gaussian $n \times n$ matrices A we have

$$\mathbb{E} \log \kappa(A) = \log n + C + O(1),$$

where $C = 1.537$ in the real case and $C = 0.982$ in the complex case. □

The second one is the best smoothed analysis, as of today, for $\kappa(A)$. It is due to Mario Wschebor (2004).

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Theorem 3.13. There exists a universal constant $K \geq 1$ such that for all $\overline{A} \in \mathbb{R}^{n\times n}$ with $\|\overline{A}\| \leq 1$, all $0 < \sigma \leq 1$, and all $t \geq 1$, we have

$$\operatorname*{Prob}_{A\sim N(\overline{A},\sigma^2\mathrm{Id})}\{\kappa(A) \geq t\} \leq Kn\frac{1}{\sigma t}.$$

Consequently, for all $\sigma \in (0,1]$ and all $\overline{A} \in \mathbb{R}^{n\times n}$ with $\|\overline{A}\| \leq 1$, we have

$$\mathbb{E}_{A\sim N(\overline{A},\sigma^2\mathrm{Id})}(\log\kappa(A)) \leq \log n + \log\frac{1}{\sigma} + 2 + \log K. \qquad \square$$

3.2. Triangular linear systems

Theorem 3.11 shows that the expected loss of precision in the solution of linear systems is logarithmic in the dimension of the system. This is a reassuring result which appears to tally with computational practice.

In contrast, the solution of triangular systems of linear equations has puzzled practitioners for decades. They have long observed that these systems are generally solved to high accuracy in spite of being, in general, ill-conditioned. Thus, for instance, J. H. Wilkinson (1963, p. 105): 'In practice one almost invariably finds that if L is ill-conditioned, so that $\|L\|\|L^{-1}\| \gg 1$, then the computed solution of $Lx = b$ (or the computed inverse) is far more accurate than [forward stability analysis] would suggest.'

The statement that in general, triangular linear systems are poorly conditioned, was given a precise meaning by Viswanath and Trefethen (1998). They showed that, if L_n denotes a random lower triangular $n \times n$ matrix (with the entries in the diagonal and below independently distributed and drawn from $N(0,1)$), then

$$\sqrt[n]{\kappa(L_n)} \to 2 \quad \text{almost surely}$$

as $n \to \infty$. It follows from this result that $\mathbb{E}(\log\kappa(L)) = \Omega(n)$, a result that suggests an expected loss of precision linear in n. Were computational practice to conform to this bound, we would not be able to accurately solve triangular linear systems. But in practice, on almost every occasion, the accuracy of the computed solution is very high. It seems there is a gap between computing practice and theoretical predictions (at least so long as random matrices are modelled with a Gaussian distribution).

Nicholas Higham (1989) pointed out that the backward error analysis given by Wilkinson (1963, Ch. 3, § 19) for forward substitution yields (small) *componentwise* bounds on the perturbed matrix (see also Bürgisser and Cucker 2013, Proposition 3.5).

Proposition 3.14. Let $L = (l_{ij}) \in \mathbb{R}^{n\times n}$ be a non-singular triangular matrix, $b \in \mathbb{R}^n$, and assume ϵ_{mach} is sufficiently small (of the order of

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

$(\log n)^{-1}$). Then, the solution $\hat{x}$ of the system $Lx = b$ computed using forward substitution satisfies

$$(L + E)\hat{x} = b,$$

for some matrix $E = (e_{ij}) \in \mathbb{R}^{n\times n}$ with

$$\frac{|e_{ij}|}{|l_{ij}|} \leq (2\log_2 n)\epsilon_{\mathsf{mach}}. \qquad \square$$

An argument similar to that used to derive (3.3), with Proposition 3.14 replacing Theorem 3.1, allows us to prove

$$\max_{i\leq n} \frac{|\tilde{x}_i - x_i|}{|x_i|} \leq 2\log_2(n)\,\epsilon_{\mathsf{mach}}\mathsf{Cw}^{\varphi}(L, b) + o(\epsilon_{\mathsf{mach}}), \tag{3.10}$$

where $\mathsf{Cw}^{\varphi}(L, b)$ is the *componentwise condition number* given by (1.3) with

$$\mathsf{RelError}(\tilde{L}, \tilde{b}) := \max_{\substack{1\leq i\leq j\leq n\\ 1\leq \ell\leq n}} \left\{ \frac{|\tilde{l}_{ij} - l_{ij}|}{|l_{ij}|}, \frac{|\tilde{b}_\ell - b_\ell|}{|b_\ell|} \right\}.$$

Consequently, the loss of precision in solving triangular systems is bounded by

$$\mathsf{LoP}(L^{-1}b) \leq \log(\log_2 n) + \log \mathsf{Cw}^{\varphi}(L, b) + \log 2 + o(1). \tag{3.11}$$

As in Section 3.1, we want to trade the occurrence of $\log \mathsf{Cw}^{\varphi}(L, b)$ in exchange for some probabilistic uncertainty. The next result (Theorem 1.1 in Cheung and Cucker 2015) does this. For a matrix A we define the max norm

$$\|A\|_{\max} := \max_{ij} |a_{ij}|.$$

Theorem 3.15. Let $\mathcal{T}$ denote the set of $n \times n$ lower triangular matrices. Let $\bar{L} \in \mathcal{T}$ and $\bar{b} \in \mathbb{R}^n$ be such that $\|\bar{L}\|_{\max} \leq 1$ and $\|\bar{b}\|_\infty \leq 1$. For $L \in \mathcal{T}$ and $b \in \mathbb{R}^n$ let $\mathsf{Cw}(L, b)$ denote the componentwise condition number, for the problem of linear equation solving, of the pair (L, b). Then, for any real number $t > n(n + 1)$ we have

$$\operatorname*{Prob}_{(L,b)\sim N_{\mathcal{T}}((\bar{L},\bar{b}),\sigma^2\mathrm{Id})} \{\mathsf{Cw}(L, b) > t\} \leq \left(\frac{1+\sigma}{\sigma}\right)\left(\frac{n^3(n+1)^2}{t - n(n+1)}\right)\sqrt{\frac{2}{\pi}}$$

and

$$\mathbb{E}_{(L,b)\sim N_{\mathcal{T}}((\bar{L},\bar{b}),\sigma^2\mathrm{Id})}(\log(\mathsf{Cw}(L, b))) \leq \log\left(\frac{1+\sigma}{\sigma}\right) + 5\log(n) + 1.16.$$

The subindex $\mathcal{T}$ in $N_{\mathcal{T}}((\bar{L}, \bar{b}), \sigma^2\mathrm{Id})$ is meant to denote that L is triangular. That is, the only entries of L which are drawn from the Gaussian $N((\bar{L}, \bar{b}), \sigma^2\mathrm{Id})$ are those in its lower part. $\square$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Note that if we replace (L, b) with (cL, cb) we obtain the same Cw. That is, $\mathsf{Cw}(L, b) = \mathsf{Cw}(cL, cb)$. Indeed, the relative errors will be the same for both pairs. The bounds on the norms of $\bar{L}$ and $\bar{b}$ are, as usual, to give some feel for the magnitude of σ.

Theorem 3.15 together with (3.11) provides a smoothed analysis for the loss of precision for the solution of linear triangular systems via forward substitution.

Corollary 3.16. Assume we solve systems $Lx = b$ using forward substitution. Then, for all $\bar{L} \in \mathcal{T}$ and $\bar{b} \in \mathbb{R}^n$ with $\|\bar{L}\|_{\max} \leq 1$ and $\|\bar{b}\|_\infty \leq 1$ we have

$$\mathbb{E}(\mathsf{LoP}(x)) \leq \log\left(\frac{1+\sigma}{\sigma}\right) + 5\log n + \log(\log_2 n) + 1.452 + o(1).$$

Here $x = L^{-1}b$ with $(L, b) \sim N_{\mathcal{T}}((\bar{L}, \bar{b}), \sigma^2 \mathrm{Id})$ and $o(1)$ is a quantity that tends to zero with ϵ_{mach}. □

3.3. Linear least-squares

A way of dealing with overdetermined linear systems going back to Gauss is to compute for them a point minimizing the residual. More precisely, for a rectangular matrix $R \in \mathbb{R}^{q\times n}$ with $q > n$ and a point $c \in \mathbb{R}^q$, one computes

$$x := \arg\min_{v\in\mathbb{R}^n} \|Rv - c\|.$$

This point x is given by $R^\dagger c$ where $R^\dagger$ is the *Moore–Penrose inverse* of R which, when R is full-rank, satisfies $R^\dagger = (R^{\mathrm{T}}R)^{-1}R^{\mathrm{T}}$. To actually compute x, QR factorization again turns out to be of help. Indeed, Theorem 3.1 is a special case of a more general result (Theorem 19.3 of Higham 1996), which we now quote in full.

Theorem 3.17. Let $R \in \mathbb{R}^{q\times n}$ have full rank, $q \geq n$, $c \in \mathbb{R}^q$, and suppose that the least-squares problem $\arg\min_v \|c - Rv\|$ is solved using the Householder QR factorization method. The computed solution $\tilde{x}$ is the exact solution to

$$\arg\min_{v\in\mathbb{R}^n} \|\tilde{c} - \tilde{R}v\|,$$

where $\tilde{R}$ and $\tilde{c}$ satisfy the relative error bounds

$$\|\tilde{R} - R\|_F \leq n\gamma_{cq}\|R\|_F \quad \text{and} \quad \|\tilde{c} - c\| \leq n\gamma_{sq}\|c\|$$

for a small constant s. □

Reasoning as in Section 3.1, we see that the loss of precision in the computation of x is bounded by

$$\mathsf{LoP}(R^\dagger c) \leq \log(n^{3/2}q) + \log \mathsf{cond}^\dagger(R, c) + \log s + o(1), \tag{3.12}$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

where $\mathsf{cond}(R, c)$ is the normwise condition number for linear least-squares (with respect to the spectral norm). That is,

$$\mathsf{cond}^{\dagger}(R, c) := \lim_{\delta \to 0} \sup_{\max\{\mathsf{RelError}(\tilde{R}), \mathsf{RelError}(\tilde{c})\} \leq \delta} \frac{\mathsf{RelError}(\tilde{x})}{\delta},$$

where

$$\mathsf{RelError}(\tilde{x}) := \frac{\|x - \tilde{x}\|}{\|x\|}$$

and similarly for $\mathsf{RelError}(\tilde{R})$ and $\mathsf{RelError}(\tilde{c})$. If, following (3.5), we define

$$\kappa_{\dagger}(R) := \|R\| \|R^{\dagger}\|,$$

we can prove that $\mathsf{cond}^{\dagger}(R, c)$, even though not tightly approximated by $\kappa_{\dagger}(R)$, is bounded by a constant times $\kappa_{\dagger}^2(R)$ (see Wedin 1973). Consequently, to obtain expected bounds (or a smoothed analysis) for the loss of precision $\mathsf{LoP}(R^{\dagger}c)$ from equation (3.12), it is enough to perform the corresponding analysis for $\log \kappa_{\dagger}(R)$. The following result (Theorem 4.2 in Bürgisser and Cucker 2013) exhibits a smoothed analysis for $\kappa_{\dagger}(R)$.

Theorem 3.18. For all $\lambda_0 \in (0, 1)$ there exists q_0 such that for all $1 \leq n \leq q$ with $\lambda = (n-1)/q \leq \lambda_0$ and $q \geq q_0$, we have for all σ with $1/\sqrt{n} \leq \sigma \leq 1$ and all $\overline{R} \in \mathbb{R}^{q \times n}$ with $\|\overline{R}\| \leq 1$, that

$$\mathbb{E}_{R \sim N(\overline{R}, \sigma^2 \mathrm{Id})}(\kappa_{\dagger}(R)) \leq \frac{20.1}{1 - \lambda}. \qquad \square$$

Jensen's inequality immediately yields the following consequence.

Corollary 3.19. Under the hypothesis of Theorem 3.18,

$$\sup_{\|\overline{R}\| \leq 1} \mathbb{E}_{R \sim N(\overline{R}, \sigma^2 \mathrm{Id})}(\log \kappa_{\dagger}(R)) \leq \log\left(\frac{20.1}{1 - \lambda}\right). \qquad \square$$

It is worth noting that the bounds in Theorem 3.18 and Corollary 3.19 are independent of q and n, as long as q is sufficiently large and the elongation λ is sufficiently small. But they depend only on λ. Indeed, perhaps surprisingly, they are even independent of σ. In fact, Corollary 3.19 indicates that for large, reasonably elongated matrices, one may expect the loss of precision in the solution of least-squares problems to derive mostly from the backward error bounds of the algorithm used (*e.g.* QR factorization).

3.4. The conjugate gradient method

In the preceding paragraph we saw that the solution $x \in \mathbb{R}^n$ of the problem

$$x := \arg\min_{v \in \mathbb{R}^n} \|Rv - c\|,$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

when $R \in \mathbb{R}^{q\times n}$ is full-rank, is given by

$$x = R^{\dagger}c = (R^{\mathrm{T}}R)^{-1}R^{\mathrm{T}}c.$$

It follows that we can find x as the solution of the system $Ax = b$ with $A := R^{\mathrm{T}}R$, $A \in \mathbb{R}^{n\times n}$, and $b := R^{\mathrm{T}}c$.

Now, by construction, A is symmetric and positive definite. One may therefore consider algorithms exploiting symmetry and positive definiteness.

Notable among the algorithms doing so is the conjugate gradient method. In what follows we briefly describe how this algorithm works, how its convergence rate is controlled by the condition number $\kappa(A)$, and how probabilistic analyses of the latter yield probabilistic estimates on the complexity of the conjugate gradient. The role played by $\kappa(A)$ in this development regards the complexity of an algorithm rather than its numerical stability. To better focus on this role, we will disregard all issues concerning finite precision and assume, instead, infinite precision in all computations.

(a) The algorithm. For $x \in \mathbb{R}^n$ we define its A-norm by

$$\|x\|_A := (x^{\mathrm{T}}Ax)^{1/2} \tag{3.13}$$

coming from the scalar product

$$\langle x, y\rangle_A := x^{\mathrm{T}}Ay.$$

We say that x, y are *A-orthogonal* (or *conjugated*) if $\langle x, y\rangle_A = 0$. Let $(d_0, \dots, d_{n-1})$ be an A-orthogonal basis of $\mathbb{R}^n$, that is, $\langle d_i, d_j\rangle_A = 0$ for $i \neq j$. Moreover, let $\bar{x}$ be the solution of $Ax = b$ and $x_0 \in \mathbb{R}^n$ be an initial point. Expand

$$\bar{x} - x_0 = \sum_{i=0}^{n-1} \alpha_i d_i,$$

with $\alpha_i \in \mathbb{R}$. Then, by A-orthogonality,

$$\alpha_k = \frac{\langle \bar{x} - x_0, d_k\rangle_A}{\|d_k\|_A^2}.$$

Define

$$x_k := \sum_{i=0}^{k-1} \alpha_i d_i + x_0.$$

Then we have $\langle x_k - x_0, d_\ell\rangle_A = 0$, for all $\ell = k, \dots, n-1$. In particular, taking $\ell = k$,

$$\langle \bar{x} - x_0, d_k\rangle_A = \langle \bar{x} - x_k, d_k\rangle_A = d_k^{\mathrm{T}}A(\bar{x} - x_k) = d_k^{\mathrm{T}}r_k,$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

with the *residue* $r_k := b - Ax_k = A\bar{x} - Ax_k$. We obtain that

$$\alpha_k = \frac{\langle d_k, r_k \rangle}{\|d_k\|_A^2}.$$

Note that α_k depends only on d_k and r_k.

Proposition 3.20.

(a) For $k \geq 1$, x_k minimizes the function $x \mapsto \|x - \bar{x}\|_A$ on the line $x_{k-1} + \mathbb{R}d_{k-1}$ as well as on the affine space $x_0 + \mathsf{span}\{d_0, \ldots, d_{k-1}\}$.

(b) We have $\langle r_k, d_i \rangle = 0$ for $i < k$. In particular, $r_n = 0$ and $x_n = \bar{x}$.

Proof. For part (a) note that the point $x_k - x_0$ is the A-orthogonal projection of $\bar{x} - x_0$ onto $\mathsf{span}\{d_0, \ldots, d_{k-1}\}$. Therefore, $x_k - x_0$ minimizes the A-distance to $\bar{x}$,

$$x \;\mapsto\; \|x - \bar{x}\|_A = \|(x - x_0) - (\bar{x} - x_0)\|_A,$$

on $x_0 + \mathsf{span}\{d_0, \ldots, d_{k-1}\}$.

For part (b), consider the function

$$f(x) := \frac{1}{2} x^{\mathrm{T}} A x - b^{\mathrm{T}} x.$$

The gradient of f at x is $\mathsf{grad}\, f(x) = Ax - b$, and it is easy to see (Bürgisser and Cucker 2013, § 5.2) that

$$f(x) = \frac{1}{2}\|x - \bar{x}\|_A^2 + f(\bar{x}).$$

Part (a) then implies that x_k minimizes f on $x_0 + \mathsf{span}\{d_0, \ldots, d_{k-1}\}$. Hence, $\mathsf{grad}\, f(x_k)$ is orthogonal to $\mathsf{span}\{d_0, \ldots, d_{k-1}\}$. But $\mathsf{grad}\, f(x_k) = -r_k$, which proves the first statement in (b). The second statement follows from the fact that r_n is then orthogonal to $d_0, \ldots, d_{n-1}$ (and hence it must be 0). This, in turn, implies $x_n = \bar{x}$. □

Thus far, we have assumed an A-orthogonal basis $\{d_0, \ldots, d_{n-1}\}$ at hand. The conjugate gradient method actually computes one, along with the computation of the x_k, by sequentially choosing the directions d_k as conjugates of the gradients $-r_k$. Algorithm 1 shows how this is done (a proof of the A-orthogonality of the resulting set can be found in Bürgisser and Cucker 2013, Theorem 5.12).

(b) First complexity considerations. We begin by looking at the cost (*i.e.* number of arithmetic operations performed) of each iteration of the algorithm. The cost of computing an inner product, such as $\langle r_k, d_k \rangle$, is $2n - 1$. Consequently, the cost of a matrix–vector multiplication, such as Ax_k, is $2n^2 - n$. It follows that computing an A-inner product costs $2n^2 + n - 1$. At each iteration of Conj_Grad the computation of each of x_{k+1} and

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Algorithm 1. Conj_Grad

Input: $A \in \mathbb{R}^{n\times n}$, $b \in \mathbb{R}^n$, $x_0 \in \mathbb{R}^n$

Preconditions: A is positive definite

$r_0 := d_0 := b - Ax_0$
$k := 0$
`while` $d_k \neq 0$ `do`
$\alpha_k := \frac{\langle r_k, d_k\rangle}{\|d_k\|_A^2}$
$x_{k+1} := x_k + \alpha_k d_k$
$r_{k+1} := b - Ax_{k+1}$
$\beta_k := -\frac{\langle r_{k+1}, d_k\rangle_A}{\|d_k\|_A^2}$
$d_{k+1} := r_{k+1} + \beta_k d_k$
$k := k + 1$
`end while`
`return` x_k

Output: $x \in \mathbb{R}^n$

Postconditions: $Ax = b$

d_{k+1} thus takes $O(n)$ arithmetic operations, and those of α_k, r_{k+1}, and β_k take $2n^2 + O(n)$ each (note that for the latter we use the already computed $\|d_k\|_A^2$). That is, the cost of an iteration of Conj_Grad is $6n^2 + O(n)$.

It follows from this bound that the cost of computing the solution $\bar{x}$ of $Ax = b$ is $6n^3 + O(n^2)$. This bound is larger than the $\frac{2}{3}n^3$ bound for the cost of Gaussian elimination, making the conjugate gradient a poor choice for the computation of $\bar{x}$.

We note at this juncture, however, that unlike Gaussian elimination, the conjugate gradient produces a sequence x_k of approximations of $\bar{x}$. If we are only interested in a good approximation, as opposed to the exact value of $\bar{x}$, then early termination of the conjugate gradient might outperform Gaussian elimination. To understand this issue we need to estimate the speed of convergence of the former.

(c) Convergence analysis. Let $\epsilon_k := x_k - \bar{x}$ be the kth *error*.

Theorem 3.21. The error at the kth step of the conjugate gradient satisfies

$$\|\epsilon_k\|_A \leq 2\left(\frac{\sqrt{\kappa(A)} - 1}{\sqrt{\kappa(A)} + 1}\right)^k \|\epsilon_0\|_A.$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Corollary 3.22. For all $\delta > 0$ we have $\|\epsilon_k\|_A \leq \delta\|\epsilon_0\|_A$ whenever

$$k \geq \ln\left(\frac{\sqrt{\kappa(A)}+1}{\sqrt{\kappa(A)}-1}\right)^{-1} \ln\left(\frac{2}{\delta}\right) \approx \frac{1}{2}\sqrt{\kappa(A)}\ln\left(\frac{1}{\delta}\right).$$

With the proof of Theorem 3.21 in mind, we introduce some notation. We $\mathcal{P}_k$ denote the linear space of all real polynomials in one variable X with degree at most k. We also write $\mathcal{Q}_k$ for the subset of $\mathcal{P}_k$ of polynomials with constant coefficient 1.

Theorem 3.23. The error ϵ_k at the kth step of the conjugate gradient method satisfies

$$\|\epsilon_k\|_A^2 \leq \min_{q\in\mathcal{Q}_k} \max_{j\leq n} q(\lambda_j)^2 \cdot \|\epsilon_0\|_A^2,$$

where $\lambda_1, \ldots, \lambda_n$ are the eigenvalues of A.

Proof. By Proposition 3.20(a) we know that x_k minimizes the A-distance of $\bar{x}$ to the affine space

$$x_0 + S_{k-1} = x_0 + \mathsf{span}\{d_0, \ldots, d_{k-1}\} = x_0 + \mathsf{span}\{r_0, Ar_0, \ldots, A^{k-1}r_0\}.$$

An element x of $x_0 + S_{k-1}$ can therefore be written as

$$x = x_0 - p(A)r_0,$$

with $p \in \mathcal{P}_{k-1}$, and conversely, for any such polynomial we obtain $x \in x_0 + Sk - 1$. Using $r_0 = b - Ax_0 = A(\bar{x} - x_0)$ we get

$$\begin{aligned} x - \bar{x} &= x_0 - \bar{x} - p(A)A(\bar{x} - x_0) = x_0 - \bar{x} + Ap(A)(x_0 - \bar{x}) \\ &= (\mathrm{Id} + Ap(A))(x_0 - \bar{x}). \end{aligned}$$

It follows that the error $\epsilon_k = x_k - \bar{x}$ at the kth step of the conjugate gradient method satisfies

$$\|\epsilon_k\|_A = \min_{x\in x_0+S_{k-1}} \|x - \bar{x}\|_A = \min_{p\in\mathcal{P}_{k-1}} \|(\mathrm{Id} + Ap(A))\epsilon_0\|_A. \tag{3.14}$$

Suppose that $v_1, \ldots, v_n$ is an orthonormal basis of eigenvectors of A corresponding to the eigenvalues $\lambda_1 \geq \cdots \geq \lambda_n > 0$. Write

$$\epsilon_0 = x_0 - \bar{x} = \sum_{j=1}^n \xi_j v_j$$

for some $\xi_1, \ldots, \xi_n \in \mathbb{R}$. Then

$$\|\epsilon_0\|_A^2 = \sum_{j=1}^n \sum_{k=1}^n \xi_j\xi_k\, v_j^{\mathrm{T}} A v_k = \sum_{j=1}^n \lambda_j \xi_j^2.$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Moreover, for any polynomial $p \in \mathcal{P}_{k-1}$,

$$(\mathrm{Id} + Ap(A))\epsilon_0 = \sum_{j=1}^{n}(1 + \lambda_j p(\lambda_j))\xi_j v_j.$$

Therefore

$$\|(\mathrm{Id} + Ap(A))\epsilon_0\|_A^2 = \sum_{j=1}^{n}(1 + \lambda_j p(\lambda_j))^2\lambda_j\xi_j^2 \leq \max_{j\leq n}(1 + \lambda_j p(\lambda_j))^2 \sum_{j=1}^{n}\lambda_j\xi_j^2,$$

and using (3.14),

$$\|\epsilon_k\|_A^2 \leq \min_{p\in\mathcal{P}_{k-1}} \max_{j\leq n}(1 + \lambda_j p(\lambda_j))^2 \sum_{j=1}^{n}\lambda_j\xi_j^2.$$

The result now follows by observing that $\mathcal{Q}_k = 1 + X\mathcal{P}_{k-1}$. □

Theorem 3.23 is hard to apply in concrete situations. It depends on all the eigenvalues of A and it optimizes a function of them over the space $\mathcal{Q}_k$. It is nevertheless a building block in the proof of Theorem 3.21. We proceed to see why.

Suppose that $\lambda_n = a$ and $\lambda_1 = b$, for $0 < a < b$. What are the optimal polynomials $q \in \mathcal{Q}_k$ if nothing is known about the location of the eigenvalues λ except that $\lambda \in [a, b]$? In this case we have to minimize the quantity

$$\max_{a\leq\lambda\leq b} q(\lambda)^2$$

over all real polynomials $q \in \mathcal{Q}_k$. This minimization problem can also be considered for arbitrary $a, b \in \mathbb{R}$ with $a < b$. In the particular case $a = -1$, $b = 1$ it turns out that its solution is given by the *Chebyshev polynomials* T_k (defined by the equation $\cos(k\phi) = T_k(\cos(\phi))$).

Proposition 3.24. For all $0 < a < b$ we have

$$\min_{\substack{\deg q\leq k\\ q(0)=1}} \max_{a\leq\lambda\leq b} q(\lambda)^2 = \frac{1}{T_k\left(\frac{b+a}{b-a}\right)^2}.$$

Proof. The polynomial

$$p(\lambda) := \frac{T_k\left(\frac{b+a-2\lambda}{b-a}\right)}{T_k\left(\frac{b+a}{b-a}\right)}$$

has degree k, satisfies $p(0) = 1$, and

$$\max_{a\leq\lambda\leq b} p(\lambda)^2 = \frac{1}{T_k\left(\frac{b+a}{b-a}\right)^2} =: c^2.$$

(Note that $\lambda \mapsto (b + a - 2\lambda)/(b - a)$ maps $[a, b]$ onto $[-1, 1]$ and T_k has maximal value 1 on this interval.)

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

In order to show the optimality of p, suppose that for some polynomial $q \in \mathcal{Q}_k$,

$$\max_{a \le \lambda \le b} |q(\lambda)| < \max_{a \le \lambda \le b} |p(\lambda)|.$$

Since p alternately takes the values $c, -c, c, -c, \ldots$ exactly $k+1$ times in the interval $[a, b]$, we conclude that the polynomial $p - q$ has at least k roots in this interval (intermediate value theorem). Moreover, $p(0) - q(0) = 1 - 1 = 0$. Hence $p - q$ has degree at most k and $k + 1$ distinct roots. It follows that $p = q$. □

Proof of Theorem 3.21. To estimate the optimal value in Proposition 3.24, note that for $z \in \mathbb{C}$, $z = \cos(\phi) + i \sin(\phi) = x + i\sqrt{1 - x^2}$, we have

$$T_k(x) = \mathsf{Re}(z^k) = \frac{1}{2}(z^k + \overline{z}^k) = \frac{1}{2}\Big(\big(x + \sqrt{x^2 - 1}\big)^k + \big(x - \sqrt{x^2 - 1}\big)^k\Big).$$

Again take $a = \lambda_n$, $b = \lambda_1$, the smallest and largest eigenvalues of A. Then $\kappa = \kappa(A) = b/a$. For

$$x = \frac{b + a}{b - a} = \frac{\kappa + 1}{\kappa - 1}$$

we get

$$x^2 - 1 = \frac{(\kappa + 1)^2 - (\kappa - 1)^2}{(\kappa - 1)^2} = \frac{4\kappa}{(\kappa - 1)^2}$$

and therefore

$$x + \sqrt{x^2 - 1} = \frac{\kappa + 1 + 2\sqrt{\kappa}}{\kappa - 1} = \frac{(\sqrt{\kappa} + 1)^2}{\kappa - 1} = \frac{\sqrt{\kappa} + 1}{\sqrt{\kappa} - 1}.$$

Hence

$$T_k(x) \ge \frac{1}{2}\left(\frac{\sqrt{\kappa} + 1}{\sqrt{\kappa} - 1}\right)^k.$$ □

(d) Conjugate gradient on random data. Theorem 3.21 gives bounds for the rate of convergence of the conjugate gradient in terms of the condition $\kappa(A)$ of the matrix A. An average (or smoothed) analysis of $\kappa(A)$ would therefore result in an average (resp. smoothed) analysis for this rate of convergence. The question that emerges is which probability distribution we draw A from.

In the case of unstructured or triangular matrices A, we have endowed the corresponding class of matrices with a Gaussian distribution. In our present case, with A positive definite, this choice is no longer available. A look at our original computational problem may, however, shed some light. Matrix A is obtained as $A = R^{\mathrm{T}}R$. It then makes sense to consider R as our primary random data – and for R we can assume Gaussianity – and endow A with the distribution inherited from that of R. In the particular case

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

that R is standard Gaussian, this inherited distribution of A is known as the *Wishart distribution with q degrees of freedom*. It extends the chi-square distribution, since it coincides with a χ^2_q when $n = 1$.

By the Singular Value Decomposition, there exist orthogonal matrices U and V and positive reals $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > 0$ such that $R = U\,\mathsf{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)V^{\mathrm{T}}$. Then

$$A = R^{\mathrm{T}}R = V\,\mathsf{diag}(\sigma_1^2, \sigma_2^2, \ldots, \sigma_n^2)V^{\mathrm{T}}$$

and, consequently, $\kappa(A) = \kappa_\dagger^2(R)$.

It follows from the above that the analysis of $\kappa(A)$ for a Wishart distribution reduces to the analysis of $\kappa_\dagger(R)$ when R is standard Gaussian (and similarly for 'non-centred Wishart distributions').

Corollary 3.22 shows that the number k of iterations needed to decrease $\|x_0 - \bar{x}\|_A$ by a factor of δ is proportional to $\sqrt{\kappa(A)}$, that is, proportional to $\kappa_\dagger(R)$. Theorem 3.18 shows that for q large enough and $\lambda = (n-1)/q$ small enough we have, for all $0 < \sigma \leq 1$,

$$\sup_{\|\overline{R}\| \leq 1} \mathbb{E}_{R \sim N(\overline{R}, \sigma^2 \mathrm{Id})}(\kappa_\dagger(R)) \leq \frac{20.1}{1-\lambda}.$$

Thus, if A is obtained as $R^{\mathrm{T}}R$ for a large elongated rectangular matrix R, then we should expect to effect the decrease mentioned above with about

$$\frac{1}{2} \cdot \frac{20.1}{1-\lambda} \ln\left(\frac{1}{\delta}\right)$$

iterations (where $\lambda = q/n$ is the elongation). Since each iteration takes $6n^2 + O(n)$ arithmetic operations, the expected cost of the computation is

$$3n^2 \frac{20.1}{1-\lambda} \ln\frac{1}{\delta} + O(n) = \frac{60.3n^2}{1-\lambda} \ln\frac{1}{\delta} + O(n).$$

The leading term in this expression is smaller than the $\frac{2}{3}n^3$ operations performed by Gaussian elimination as long as

$$\delta \geq e^{-\frac{n(1-\lambda)}{91}}.$$

For large n (and λ not too close to 1) this bound produces very small values of δ, and therefore Conj_Grad yields, on average (both for a Gaussian distribution of data and for Gaussian perturbations of arbitrary data), remarkably good approximations of $\bar{x}$.

We conclude that for reasonable $\delta > 0$ and large n, computing an approximation of the solution $\bar{x}$ of $Ax = b$ using Algorithm Conj_Grad will typically be faster than computing $\bar{x}$ with Gaussian elimination.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

3.5. Feasibility of polyhedral conic systems

(a) The feasibility problem and its condition. A homogenization of the linear programming feasibility programme can be stated as follows. Given $A \in \mathbb{R}^{m\times n}$, $n > m$, consider the *primal feasibility problem*

$$\exists x \in \mathbb{R}^n \setminus \{0\}, \quad Ax = 0,\ x \geq 0, \tag{PF}$$

and the *dual feasibility problem*

$$\exists y \in \mathbb{R}^m \setminus \{0\}, \quad A^{\mathrm{T}} y \leq 0. \tag{DF}$$

We say that A is *primal feasible* or *dual feasible* when (PF) or (DF), respectively, is satisfied. In both cases we talk about *strict* feasibility when the satisfied inequality is strict.

Let $\mathcal{F}_P$ and $\mathcal{F}_D$ denote the sets of matrices A where (PF) and (DF) are satisfied, respectively, and let $\mathcal{F}_P^\circ$ and $\mathcal{F}_D^\circ$ be the sets of strictly primal and strictly dual feasible matrices. Finally, let

$$\Sigma := \mathcal{F}_P \cap \mathcal{F}_D.$$

It is well known that, modulo a set of lower dimension, primal (dual) feasibility is equivalent to dual (resp. primal) strict infeasibility. The following result (see Bürgisser and Cucker 2013, Theorem 6.14) is more precise. To state it, we let $\mathsf{int}(M)$, $\overline{M}$, and $\partial M = \overline{M} \setminus \mathsf{int}(M)$ denote the interior, closure, and boundary of a subset M of Euclidean space.

Theorem 3.25. Both $\mathcal{F}_P$ and $\mathcal{F}_D$ are closed subsets of $\mathbb{R}^{m\times n}$. In addition, this space is partitioned as

$$\mathbb{R}^{m\times n} = \mathsf{int}(\mathcal{F}_P^\circ) \cup \mathsf{int}(\mathcal{F}_D^\circ) \cup \Sigma,$$

and we have

$$\Sigma = \partial\mathcal{F}_P = \partial\mathcal{F}_D.$$

Furthermore,

$$\begin{aligned} \mathcal{F}_P^\circ &\supseteq \mathsf{int}(\mathcal{F}_P), \\ \mathcal{F}_P^\circ \setminus \mathsf{int}(\mathcal{F}_P) &= \mathcal{F}_P^\circ \cap \Sigma = \{A \in \Sigma \mid \mathsf{rank}\, A < m\}, \\ \text{and} \quad \mathcal{F}_D^\circ &= \mathsf{int}(\mathcal{F}_D). \end{aligned}$$

□

We see that for matrices in Σ, arbitrarily small perturbations can lead to a change with respect to feasibility. In contrast, in the set $\mathcal{D} = \mathbb{R}^{m\times n} \setminus \Sigma$ the following problem is well-defined:

$$\text{Given } A \in \mathcal{D}, \text{ decide whether } A \in \mathcal{F}_P^\circ \text{ or } A \in \mathcal{F}_D^\circ.$$

We call this the *polyhedral cone feasibility problem*. For all $A \in \Sigma$ the problem is *ill-posed*.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

The polyhedral cone feasibility problem is a decision problem (as described in Section 1.1(c)). We will define for it a measure of condition in terms of a relative distance to ill-posedness. We have just seen that the set of ill-posed instances is the boundary between feasible and infeasible instances. This motivates the following definition.

Definition 3.26. Let $A \in \mathbb{R}^{m\times n}$ be given with non-zero columns a_i. Suppose $A \notin \Sigma$. Then, $A \in \mathcal{F}_S^\circ$ for $S \in \{P, D\}$. We define

$$\Delta(A) := \sup\left\{\delta > 0 \;\middle|\; \forall A' \in \mathbb{R}^{m\times n} \left(\max_{i\le n} \frac{\|a_i' - a_i\|}{\|a_i\|} < \delta \;\Rightarrow\; A' \in \mathcal{F}_S^\circ\right)\right\},$$

where a_i' stands for the ith column of A'. The *GCC condition number* of A is defined as

$$\mathscr{C}(A) := \frac{1}{\Delta(A)}.$$

If $A \in \Sigma$, we set $\Delta(A) = 0$ and $\mathscr{C}(A) = \infty$.

We have written the definition in such a way that it becomes clear that we measure the *relative* size of the perturbation for each row a_i, where the relativization is with respect to the norm of a_i. Also, it is clear from the definition that $\Delta(A)$ is scale-invariant in the sense that

$$\Delta([\lambda_1 a_1, \ldots, \lambda_n a_n]) = \Delta([a_1, \ldots, a_n]) \quad \text{for } \lambda_i > 0.$$

For the analysis of Δ we may therefore assume, without loss of generality, that $\|a_i\| = 1$ for all i. Hence we can see the matrix A with columns $a_1, \ldots, a_n$ as an element in the product $(\mathbb{S}^{m-1})^n$ of spheres. The scale invariance of $\mathscr{C}(A)$, together with the characterization

$$\|A\|_{12} = \max_{i\le n} \|a_i\|$$

immediately yields the following result.

Proposition 3.27.

$$\mathscr{C}(A) = \frac{\|A\|_{12}}{d_{12}(A, \Sigma)}. \qquad \square$$

A number of different characterizations of $\mathscr{C}(A)$ are shown in Bürgisser and Cucker (2013, Chapter 6).

(b) Algorithms and complexity. During the late 1970s and early 1980s two algorithms proved that linear programming problems (with integer coefficients) can be solved in polynomial time: the ellipsoid algorithm and interior-point methods. These algorithms are naturally described as real number algorithms. But when considered over real (as opposed to integer) data their halting time is no longer bounded by a function of the input size only. Instead, the condition of the data at hand appears to play a role.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Of course, one may argue that the condition of the data is now replacing the bit-size of the largest component in this data. But the fact remains that this bit-size is directly related to the size of the input whereas the condition of the input is not (and it is not even known *a priori*).

Of these two methods, the most efficient (both in theory and in practice) is the interior-point method. There is a vast literature on this method (the book by Renegar 2000 being a very good introduction to them for general conic programming). For our purposes, it should suffice to state a result with the complexity of an interior-point algorithm solving the polyhedral cone feasibility problem. The proof of this result (and a self-contained introduction to interior-point methods for linear programming) can be found in Chapter 9 of Bürgisser and Cucker (2013).

Theorem 3.28. We exhibit an interior-point algorithm which, on input $A \notin \Sigma$, returns either a strictly feasible primal solution, *i.e.*, a point $x \in \mathbb{R}^n$ such that $Ax = 0$, $x > 0$, or a strictly feasible dual solution, *i.e.*, a point $y \in \mathbb{R}^m$ such that $A^{\mathrm{T}}y < 0$. In both cases, the algorithm halts after at most

$$O(\sqrt{n}(\log n + \log \mathscr{C}(A)))$$

iterations. The total number of arithmetic operations is bounded by

$$O\big(n^{3.5}(\log n + \log \mathscr{C}(A))\big). \qquad \square$$

A version of this result which takes into consideration the effects of finite precision, and shows complexity bounds of the same order, can be found in Cucker and Peña (2002).

(c) Condition of random data. Theorem 3.28 is an instance of a common phenomenon in the analysis of iterative algorithms (which we already met regarding the conjugate gradient method). The cost of each iteration is bounded by a function of the size or dimension of the input (in our case, it is $O(n^3)$ where, we recall, $A \in \mathbb{R}^{m\times n}$ with $n > m$) but the number of iterations performed by the algorithm depends on the condition of the input at hand. To get an idea of how efficient the algorithm is 'in general', a probabilistic analysis is a must.

There is a sequence of papers on probabilistic analyses of $\mathscr{C}(A)$. The sharpest bound for the expectation of $\log \mathscr{C}(A)$ appears in Bürgisser, Cucker and Lotz (2010). It shows that

$$\mathbb{E}_{A\sim N(0,\mathrm{Id})} \ln \mathscr{C}(A) \leq 2\ln(m+1) + 3.31.$$

As for a smoothed analysis, Amelunxen and Bürgisser (2012) showed that

$$\sup_{\overline{A}\in(\mathbb{S}^{m-1})^n} \mathbb{E}_{A\sim N(\overline{A},\sigma^2\mathrm{Id})} \log \mathscr{C}(A) = O\left(\log \frac{n}{\sigma}\right).$$

These two bounds carry a very clear message: the cost of solving a typical

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

polyhedral feasibility problem (or a typical perturbation of an arbitrary one) is $O(n^{3.5} \log n)$. The contribution of the condition of the input to this cost is very minor.

3.6. Linear programming

(a) The problem. The *linear programming problem in standard form* is to find $x \in \mathbb{R}^n$ solving

$$\min c^{\mathrm{T}} x \quad \text{subject to } Ax = b,\ x \geq 0, \tag{OP}$$

where $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$ are given. The well-known Duality Theorem states that if a minimizer x^* exists then a maximizer y^* exists as well for the *dual problem*

$$\max b^{\mathrm{T}} y \quad \text{subject to } A^{\mathrm{T}} y \leq c, \tag{OD}$$

and we have $c^{\mathrm{T}} x^* = b^{\mathrm{T}} y^*$. It additionally ensures that this is the case if and only if both the primal and dual are feasible. That is, there exists a pair $(x, y) \in \mathbb{R}^{n+m}$ such that $Ax = b$, $x \geq 0$ and $A^{\mathrm{T}} y \leq c$. In this case, we say that the triple $d = (A, b, c)$ is *feasible*.

For any subset B of $\{1, 2, \ldots, n\}$, we let A_B denote the submatrix of A obtained by removing from A all the columns with index not in B. If $x \in \mathbb{R}^n$, x_B is defined analogously. Also, let $N := \{1, 2, \ldots, n\} \setminus B$. Then A_N and x_N are defined similarly. We say that B is a *basis* for $d = (A, b, c)$ when $|B| = m$ and A_B is invertible.

In this case, we may solve $A_B x = b$. Consider the point $x^* \in \mathbb{R}^n$ defined by $x^*_N = 0$ and $x^*_B = x$. Clearly, $Ax^* = b$. We say that x^* is the *primal basic solution* of (OP) associated with B. If, in addition, $x^* \geq 0$, which is equivalent to $x^*_B \geq 0$, then we say that x^* is a *primal basic feasible solution*.

Similarly, we may solve $A_B^{\mathrm{T}} y = c_B$. The point y^* thus obtained is said to be the *dual basic solution* of (OD) associated with B. If, in addition, $A^{\mathrm{T}} y^* \leq c$, then y^* is said to be a *dual basic feasible solution*.

Definition 3.29. A basis B for a data triple d is called an *optimal basis* if both the primal and dual basic solutions associated with B are feasible, in which case the latter are called the *basic optimal solutions* of d associated with B.

There is a rationale for the word 'optimal' in this definition. If B is an optimal basis and x^*, y^* are its associated basic optimal solutions, then x^* and y^* are optimizers for the problems (OP) and (OD), respectively. Furthermore, optimal bases exist (Bürgisser and Cucker 2013, Theorem 8.13).

Theorem 3.30. Let $d = (A, b, c)$ be a data triple satisfying $\mathsf{rank}\, A = m$.

(a) There is an optimal basis for d if and only if both (OP) and (OD) are feasible.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

(b) Let B be a basis for d. Then B is optimal if and only if both the primal and the dual basic solutions associated with B are optimizers for (OP) and (OD) respectively.

(c) If there is more than one optimal basis, say $B_1, \ldots, B_s$, $s \geq 2$, then the set of optimizers for (OP) contains the convex hull of $x_1^*, \ldots, x_s^*$, where $x_i^* \in \mathbb{R}^n$ is the primal basic solution associated with B_i, $i = 1, \ldots, s$. Likewise for the set of optimizers for (OD). □

Theorem 3.30 suggests a strategy to compute optimizers for feasible data with full-rank matrices, namely, to compute an optimal basis B and its associated basic solutions x^* and y^*. These points are vertices in the polyhedra of feasible points for (OP) and (OD), respectively. The simplex algorithm searches for such a solution by examining a cleverly constructed sequence in this set of vertices. Interior-point methods instead produce a sequence of points in the interior of these polyhedra that leads to the desired vertex.

(b) Ill-posedness and condition. The problems of computing optimizers and optimal bases are, strictly speaking, different. But both their sets of ill-posed data and their condition numbers are, as one can expect, very closely related.

Even though we will only be interested in feasible data (in this example we compute optimizers for data of this type) we still need to mention infeasible data at the boundary of the set of feasible data. A more detailed discussion of ill-posedness, with connections to the notion of degeneracy, can be found in Bürgisser and Cucker (2013, § 8.4).

We observe that infeasible triples should be considered as ill-posed if and only if arbitrarily small perturbations can make these triples feasible (and, consequently, create optimizers for them). We thus define

$$\Sigma_{\mathcal{I}} := \{d \mid d \text{ is infeasible and } \forall \varepsilon > 0 \ \exists d' \text{ feasible with } \|d - d'\| \leq \varepsilon\}$$

and call this the set of *infeasible ill-posed triples*.

For the problem of computing optimizers one can show that the property of having more than one pair (x^*, y^*) of optimizers can be destroyed by arbitrarily small perturbations of the triple $d = (A, b, c)$. Also, for problems having a unique pair of optimizers, small perturbations of the data will preserve this character and the resulting pair of optimizers will be close to the original one. Consequently, we define the set of ill-posed triples for this problem to be

$$\Sigma_{\mathsf{opt}} := \{d \mid d \text{ has at least two optimizers } (x, y)\} \cup \Sigma_{\mathcal{I}}.$$

Let us focus next on the problem of computing an optimal basis. In contrast with the preceding problem, this one is discrete-valued: the output for any given datum d is an m-element subset in $[n]$. Therefore, we can take the set Σ_{OB} of ill-posed triples as the boundary between these possible

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

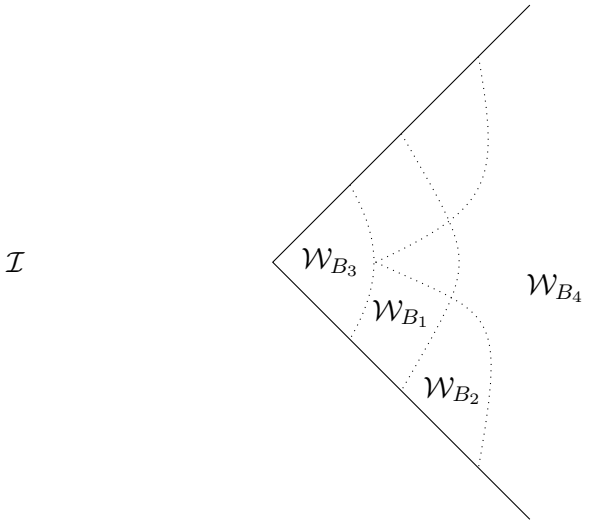


Figure 3.1. The situation in the space of data.

outputs. In other words, if $\mathsf{OB}(d)$ denotes the set of optimal bases for a triple d, we define

$$\Sigma_{\mathsf{OB}} := \{d \mid \forall \varepsilon > 0 \ \exists d' \text{ such that } \mathsf{OB}(d) \neq \mathsf{OB}(d') \text{ and } \|d - d'\| \leq \varepsilon\}.$$

Note that $\Sigma_{\mathcal{I}}$ corresponds to triples d in Σ_{OB} with $\mathsf{OB}(d) = \emptyset$.

Proposition 3.31. Let $d = (A, b, c)$ be feasible. Then we have $d \in \Sigma_{\mathsf{opt}} \iff d \in \Sigma_{\mathsf{OB}}$. Furthermore, if $\mathsf{rank}\, A < m$ then $d \in \Sigma_{\mathsf{OB}}$. □

Following the above we say that d is *feasible well-posed* when there exists a unique optimal solution (x^*, y^*) for d. In this case, we write $d \in \mathcal{W}$. If more than one optimal solution exists, we say that d is *feasible ill-posed*.

Let $\mathcal{B} := \{B \subseteq [n] \mid |B| = m\}$. Theorem 3.30(c) implies that if $d \in \mathcal{W}$ is feasible well-posed with optimal solution (x^*, y^*), then there is a unique $B \in \mathcal{B}$ such that (x^*, y^*) is the basic optimal solution associated with B. We may then define, for any $B \in \mathcal{B}$, the set

$$\mathcal{W}_B := \{d \in \mathcal{W} \mid B \text{ is the optimal basis for } d\}.$$

The set $\mathcal{W}$ is thus partitioned by the family $\{\mathcal{W}_B \mid B \in \mathcal{B}\}$. Figure 3.1 schematically summarizes the situation in the space of data.

The left of the diagram is not relevant to us: it represents the set $\mathcal{I}$ of infeasible triples. At the right of the diagram we have the set $\mathcal{W}$ of well-posed feasible triples, which is partitioned into the subsets $\mathcal{W}_B$. The dotted lines separating these subsets correspond to data with more than one optimal basis. The two continuous segments are the boundary between feasible and infeasible triples. The union of these lines – the dotted lines

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

and the continuous segments – forms the set $\Sigma_{\mathsf{opt}} = \Sigma_{\mathsf{OB}}$ of ill-posed triples for both the optimal solution and the optimal basis problems.

The picture we just described allows for a very natural definition of condition number. We only need a norm in the space of data triples, and for this we take

$$\|d\| = \left\| \begin{pmatrix} A & b \\ c^{\mathrm{T}} & 0 \end{pmatrix} \right\|.$$

Definition 3.32. Let d be feasible well-posed. We define the *distance to ill-posedness* to be

$$\varrho(d) = \inf\{\|\Delta d\| \mid d + \Delta d \in \Sigma_{\mathsf{opt}}\}.$$

Moreover, we define the *RCC condition number* $\mathscr{K}(d)$ of d by

$$\mathscr{K}(d) = \frac{\|d\|}{\varrho(d)}.$$

If d is feasible but not well-posed, we let $\varrho(d) = 0$ and $\mathscr{K}(d) = \infty$.

(c) Algorithms and complexity. There is no shortage of algorithms for the linear programming optimization problem. A distinctive feature of the one described in Cheung and Cucker (2006) is that it works with finite precision. Another feature is that both complexity and accuracy are analysed in terms of $\mathscr{K}(d)$ (which, somewhat conveniently, was introduced and studied in Cheung and Cucker 2004).

Theorem 3.33. There exists a finite-precision algorithm which, given a feasible well-posed triple $d = (A, b, c)$, finds an optimal basis B for d. The machine precision varies during the execution of the algorithm (allowing for rereading the input d with higher accuracy) and the finest required precision satisfies

$$\epsilon^*_{\mathsf{mach}} = \frac{1}{O((m+n+1)^{26}\mathscr{K}(d)^{16})}.$$

The total number of arithmetic operations is bounded by

$$O\big(n^{3.5}(\log n + \log \mathscr{K}(d)) \log\log(\mathscr{K}(d)+1)\big). \qquad \square$$

The feasibility of the triple d needs to be ascertained before the execution of the algorithm in Theorem 3.33, but it is shown in Cheung and Cucker (2006) that this can be done within the accuracy and complexity bounds in the theorem.

The bound for $\epsilon^*_{\mathsf{mach}}$ shows that the largest number of digits needed for the floating-point numbers occurring during the computation is bounded as

$$O(\log(n) + \log \mathscr{K}(d)). \tag{3.15}$$

In Bürgisser and Cucker (2013, Chapter 11) there is a version of Theorem 3.33 for infinite precision. Needless to say, the corresponding proof is

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

much simplified. There is also a description of a different algorithm with a slightly smaller complexity bound (without the $\log\log \mathscr{K}(d)$ factor) but requiring an initial pair (x_0, y_0) satisfying some properties.

One final remark is essential. Once we have computed the optimal basis B for the triple d we still want to compute the optimizers. For this, recall that we need to compute

$$x_B^* = A_B^{-1}b \quad \text{and} \quad y^* = A_B^{-\mathrm{T}}c_B.$$

Whereas the complexity of this computation is dominated by that of computing B, its accuracy analysis (such as the one in Section 3.1) depends on $\kappa(A_B) = \kappa(A_B^{\mathrm{T}})$. However, the inequality

$$\kappa(A_B) \leq \mathscr{K}(d)$$

(see Bürgisser and Cucker 2013, Proposition 11.22) reduces all our needs to one: understanding how large $\mathscr{K}(d)$ is in general.

(d) Probabilistic analyses. An average analysis of $\mathscr{K}(d)$ was obtained in Cheung and Cucker (2013).

Theorem 3.34. We have

$$\mathbb{E}_{d\sim N(0,\mathrm{Id})}\big(\log \mathscr{K}(d) \mid d \in \mathcal{W}\big) \leq \frac{5}{2}\log(n+1) + \log(m+1) + \log 24e,$$

as well as

$$\mathbb{E}_{d\sim N(0,\mathrm{Id})}\big(\log\log \mathscr{K}(d) \mid d \in \mathcal{W}\big) = \log\log n + O(1)$$

and

$$\mathbb{E}_{d\sim N(0,\mathrm{Id})}\big(\log \mathscr{K}(d)\log\log \mathscr{K}(d) \mid d \in \mathcal{W}\big) = \log n \log\log n + O(1). \quad \square$$

Theorems 3.33 and 3.34 combine to yield the following.

Corollary 3.35. There exists a finite-precision algorithm that, given a triple $d = (A, b, c)$, finds an optimal basis B for d. The average cost of the algorithm, on Gaussian data d, conditioned to d being feasible well-posed, is bounded by

$$O\big(n^{3.5}\log n \log\log n\big).$$

The machine precision varies during the execution of the algorithm. The largest number of digits needed for the floating-point numbers occurring in the computation is bounded on average as

$$\mathbb{E}_{d\sim N(0,\mathrm{Id})} \log \epsilon^*_{\mathsf{mach}}(d) = O(\log n). \quad \square$$

Remarkably, as of today, the problem of showing a smoothed analysis version of Theorem 3.34 remains open.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

3.7. Homotopy methods

In all the previous examples, the behaviour of an algorithm (either with respect to numerical stability or with respect to complexity) is determined by, in addition to size, the condition of the input at hand. The examples in the next two paragraphs showcase a more intricate situation in which the complexity of an execution depends on the condition of an infinite collection of objects, namely a curve in the product space $X \times Y$, where X is a manifold of inputs and Y a manifold of solutions. We are therefore in the framework described in Section 1.1(b), which the reader might want to refresh before continuing.

In our context, it will be convenient to look at the elements in the manifold of inputs as functions defined on Y with values on some linear space. The computational problem is, given $f \in X$, find $y \in Y$ such that $f(y) = 0$. Homotopy (or continuation) methods are a family of algorithms that do just this. They require a pair (g, ζ) at hand with $g \in X$ and $\zeta \in Y$ a zero of g.

The general idea of the method is to consider a path

$$\gamma : [0,1] \to X, \quad t \mapsto q_t,$$

such that $q_0 = g$ and $q_1 = f$. This path needs to be computable (in the sense that q_t may be computed from f, g, and t).

Under certain conditions the path γ can be lifted to a path Γ in the solution manifold $V \subset X \times Y$ such that $\Gamma(0) = (q_0, \zeta)$. If this is the case, by projecting onto Y we obtain a path $\{\zeta_t\}_{t\in[0,1]}$ on Y such that $\zeta_0 = \zeta$ and $f(\zeta_1) = 0$. The goal of homotopy methods is to 'follow' (or 'continue') the path on V to eventually obtain an approximation of ζ_1. The algorithmic scheme, broadly described, is shown in Algorithm 2. Figure 3.2 attempts to give an idea of this process.

To turn this broad description into a working algorithm, a number of issues need to be clarified: What is 'an approximation' z of a zero ζ_t of q_t? How are the approximations z_i computed? How are the points t_i computed?

It turns out that Newton's method and Smale's theory of approximate zeros give an elegant answer to these questions.

When we talk about Newton's method we are referring to a general idea that is adapted to different contexts. The general idea is that to a function $f : Y \to \mathbb{C}^n$ we associate its Newton's operator $N_f : Y \to Y$, which is defined almost everywhere on Y. The key property of this operator is that if we start from a point $z_0 \in Y$ sufficiently close to a simple zero ζ of f and we iteratively compute its image under N_f, then the resulting sequence converges immediately and quadratically fast to ζ. Newton's method does not actually find a zero of f, but starting with a modest approximation of such a zero, it refines its accuracy in a very fast manner.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Algorithm 2. Adaptive_Homotopy

Input: $f, g \in X$, $\zeta \in Y$

Preconditions: $g(\zeta) = 0$

set $i := 0$, $z_i := \zeta$, and $t_i := 0$
while $t_i < 1$ do
    compute t_{i+1} from t_i, z_i, and q_{t_i}
    compute an approximation z_{i+1} of $\zeta_{t_{i+1}}$
        from z_i and $q_{t_{i+1}}$
end while
return z_k

Output: $z \in Y$

Postconditions: z is an approximate zero of f

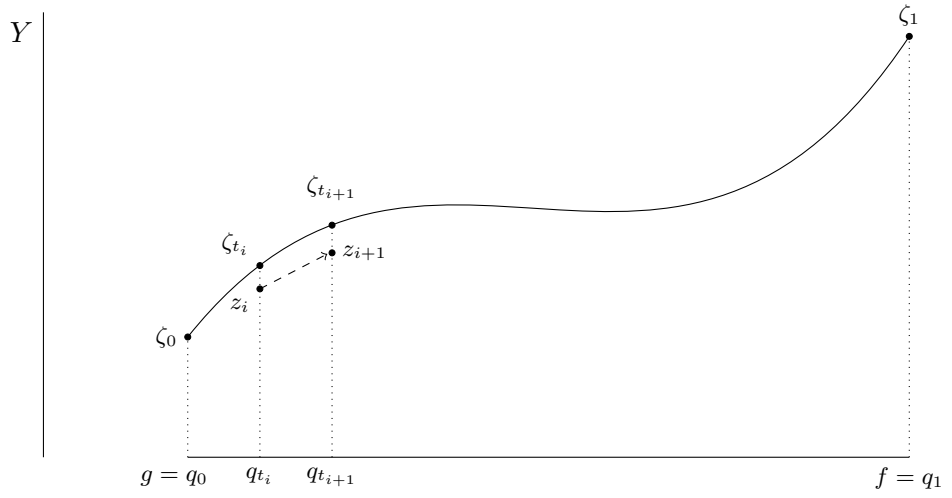


Figure 3.2. The continuation of the solution path.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

To make this notion into a definition we assume a distance dist on Y.

Definition 3.36. We say that $z \in Y$ is an *approximate zero* of f if the sequence given by $z_0 = z$ and $z_{i+1} = N_f(z_i)$ is defined for all natural numbers i, and there is a ζ with $f(\zeta) = 0$ such that, for all $i \in \mathbb{N}$,

$$\mathsf{dist}(z_i - \zeta) \leq \left(\frac{1}{2}\right)^{2^i - 1} \mathsf{dist}(z - \zeta).$$

We say that ζ is the *associated zero* of z.

This definition gives an answer to one of our questions above, What is an approximation of a zero? Newton's method itself gives an answer to another of these questions, How are the z_i computed? Indeed, we make the second computation in the `while` loop in Algorithm 2 precise by taking

$$\texttt{compute } z_{i+1} := N_{q_{t_{i+1}}}(z_i).$$

It remains to give an answer to the third question: How are the points t_i computed? That is, how do we compute t_{i+1} from t_i, z_i, and q_{t_i}?

The short answer is: *we choose t_{i+1} so that z_i is an approximate zero of $q_{t_{i+1}}$.*

Procedures for a choice of t_{i+1} ensuring this property turn out to be easy to implement and difficult to justify. In all cases, they rely on an estimate of the size of the *basin of quadratic attraction* of $f \in X$ around a zero ζ. That is, how large is the set

$$\{z \in Y \mid z \text{ is an approximate zero of } f \text{ with associated zero } \zeta\}.$$

Smale's *Gamma theory* gives an answer to this question in terms of a quantity $\gamma(f, \zeta)$ that relates to Hadamard's formula for the radius of convergence of an analytic function around a given point. As it happens, $\gamma(f, \zeta)$ is closely related to condition as well.

The combination of z_i being an approximate zero of $q_{t_{i+1}}$ and z_{i+1} being defined as $N_{q_{t_{i+1}}}(z_i)$ ensures that, for all i, the point z_i is very close to the zero ζ_i. Figure 3.3 depicts the situation. The curve in the centre is the lifted path $\Gamma \subset V$, that is, the set $\{(q_t, \zeta_t)\}_{t \in [0,1]}$. The lightly shaded area is the set of points (q_t, z) such that z is an approximate zero of q_t with associated zero ζ_t. The dark shaded area is a contraction of the lightly shaded one centred at Γ with coefficient $1/2$. The sequence of approximations z_i lie on this dark shaded region. In a way, they are 'strong approximate' zeros.

Both the role played by condition numbers in the choice of the homotopy's step-size and the way they affect the speed of the algorithm will be main characters in the next two examples.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

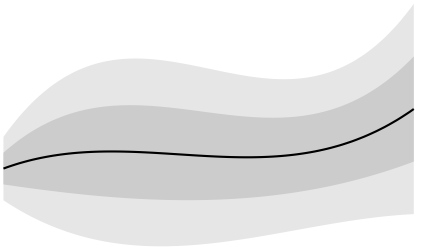


Figure 3.3. Approximate and 'strong approximate' zeros.

3.8. Smale's 17th problem

(a) The problem. Let $\mathbf{d} = (d_1, \ldots, d_n)$ with $d_i \in \mathbb{N}$, $d_i \geq 2$. We let $\mathcal{H}_{\mathbf{d}}$ denote the linear space of homogeneous polynomial systems $f = (f_1, \ldots, f_n)$, where $f_i \in \mathbb{C}[X_0, \ldots, X_n]$ is homogeneous of degree d_i.

Steve Smale (2000) proposed a list of 18 problems for the mathematicians of the twenty-first century. The 17th problem asks whether there exists an algorithm that, given $f \in \mathcal{H}_{\mathbf{d}}$, returns an approximate zero of f in time polynomial (on average) in the size of f. Here 'approximate zero' is in the sense of Definition 3.36 (with respect to a projective Newton's method tailored to this problem by Mike Shub 1993) and 'average' is with respect to the standard normal distribution for the coefficients of the f_i when these polynomials are written in the Weyl basis (a simple modification to the usual basis that makes the associated Hermitian product unitarily invariant: for details see Bürgisser and Cucker 2013, Section 16.1). The 'size' of f is the total number of these coefficients or, equivalently, the dimension N of $\mathcal{H}_{\mathbf{d}}$ over $\mathbb{C}$. This yields

$$N = \sum_{i=1}^{n} \binom{n + d_i}{d_i}.$$

Attempts to give an affirmative answer to Smale's 17th problem have revolved around homotopy algorithms. The works by Beltrán and Pardo (2009, 2011) provided a partial answer in the form of a randomized algorithm whose expected running time (with respect to both the randomizations and the input f) is polynomial in N. A further development was given in Bürgisser and Cucker (2011), where a deterministic algorithm working in nearly polynomial time was described. All these works heavily relied on the analysis of a condition number. In what follows we define this condition number, specify the choice of step-length in the homotopy algorithm, and show (sketching the proofs) how a smoothed analysis leads to the results mentioned above.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

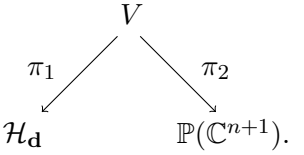
(b) Condition numbers. To define the condition for Smale's 17th problem we use the setting described in Section 1.1(b). Here the manifold of inputs is $\mathcal{H}_{\mathbf{d}}$, the manifold of outputs is $\mathbb{P}^n := \mathbb{P}(\mathbb{C}^{n+1})$, and the submanifold $V \subset \mathcal{H}_{\mathbf{d}} \times \mathbb{P}(\mathbb{C}^{n+1})$ of solutions is the set of pairs (f, ζ) with $f(\zeta) = 0$.

It turns out that the *set of ill-posed solutions* is then (recall (1.6))

$$\Sigma' = \{(f, \zeta) \in V \mid \mathsf{rank}\, Df(\zeta) < n\} \tag{3.16}$$

and that, for $(f, \zeta) \in V$, $(f, \zeta) \in \Sigma'$ if and only if z is a multiple zero of f.

Recall the projections π_1 and π_2:



We also define the set $\Sigma := \pi_1(\Sigma')$ consisting of those systems $f \in \mathcal{H}_{\mathbf{d}}$ that have some multiple zero ζ.

For a pair $(f_0, \zeta_0) \in V \setminus \Sigma'$ we have defined the absolute condition number $\mathsf{acond}(f_0, \zeta_0)$ to be the operator norm of the solution map $G\colon U \to \mathbb{P}^n, G(f) = \zeta$ that locally inverts $\pi_1\colon V \to \mathcal{H}_{\mathbf{d}}$ around (f_0, ζ_0) on an open neighbourhood U of f_0 in $\mathcal{H}_{\mathbf{d}}$. To make the condition independent of the scaling of f we take, for $(f, \zeta) \in V \setminus \Sigma'$,

$$\mu(f, \zeta) := \|f\| \cdot \|DG(f)\|, \tag{3.17}$$

where the operator norm is with respect to the standard Hermitian product (with respect to the Weyl's basis) on $\mathcal{H}_{\mathbf{d}}$ and the Hermitian product on $T_\zeta \mathbb{P}^n$ coming from the standard one on $\mathbb{C}^{n+1}$.

The condition number $\mu(f, \zeta)$ can also be expressed in terms of the derivative $Df(\zeta)\colon \mathbb{C}^{n+1} \to \mathbb{C}^n$ of $f\colon \mathbb{C}^{n+1} \to \mathbb{C}^n$. Euler's identity implies that ζ lies in the kernel of $Df(\zeta)$. So, if $Df(\zeta)$ is of full rank, its kernel is $\mathbb{C}\zeta$ and T_ζ equals its orthogonal complement. The inverse of the restriction $Df(\zeta)_{|T_\zeta}$ is described by the Moore–Penrose inverse $Df(\zeta)^\dagger$, and we have

$$\mu(f, \zeta) = \|f\| \cdot \|Df(\zeta)^\dagger\, \mathsf{diag}(\|\zeta\|^{d_i - 1})\|.$$

The function μ is a condition number in the classical sense: it bounds the speed of change of the solutions when the data change. Indeed, if $[0, 1] \to V$, $t \mapsto (f_t, \zeta_t) \in V$ is a smooth curve such that $f_t \in \mathbb{S}(\mathcal{H}_{\mathbf{d}})$ for all t, then

$$\|\dot{\zeta}_t\| \leq \mu(f_t, \zeta_t)\|\dot{f}_t\|.$$

For technical reasons, it has proved to be convenient to work with a slight modification to μ. We define the *normalized condition number* $\mu_{\mathrm{norm}}(f, \zeta)$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

for $(f,\zeta) \in V$ by

$$\mu_{\text{norm}}(f,\zeta) := \|f\| \cdot \left\|Df(\zeta)^\dagger \,\mathsf{diag}(\sqrt{d_i}\|\zeta\|^{d_i-1})\right\|. \tag{3.18}$$

Note that the introduction of the $\sqrt{d_i}$ factors is the only change compared with $\mu(f,\zeta)$. In particular, we have

$$\mu(f,\zeta) \le \mu_{\text{norm}}(f,\zeta) \le \sqrt{D}\,\mu(f,\zeta) \tag{3.19}$$

where $D := \max_i d_i$. Furthermore, since our algorithms work with approximations of zeros only, it will be convenient to extend $\mu_{\text{norm}}(f,z)$ to the case that z is not a zero of f.

Definition 3.37. For $f \in \mathcal{H}_{\mathbf{d}}$ and $z \in (\mathbb{C}^{n+1})_*$ we define the *normalized condition number* $\mu_{\text{norm}}(f,z)$ as

$$\mu_{\text{norm}}(f,z) := \|f\| \cdot \left\|Df(z)|_{T_z}^{-1} \,\mathsf{diag}\big(\sqrt{d_i}\,\|z\|^{d_i-1}\big)\right\|$$

if $Df(z)|_{T_z} : T_z \to \mathbb{C}^n$ is invertible. Otherwise, we set $\mu_{\text{norm}}(f,z) := \infty$.

By Euler's identity, if $f(z) = 0$, this is consistent with the original definition (3.18):

$$\mu_{\text{norm}}(f,z) = \|f\| \cdot \left\|Df(z)^\dagger \,\mathsf{diag}\big(\sqrt{d_i}\|z\|^{d_i-1}\big)\right\|.$$

We note, however, that using this formula in the case $f(z) \neq 0$ would lead to a different notion of $\mu_{\text{norm}}(f,z)$ (although the difference is small if z is close to ζ).

The condition number μ_{norm} has several properties. It is invariant under scaling: that is, for $\lambda_1, \lambda_2 \in \mathbb{C}_*$, we have $\mu_{\text{norm}}(\lambda_1 f, \lambda_2 z) = \mu_{\text{norm}}(f,z)$. It is bounded below: $\mu_{\text{norm}}(f,z) \ge \sqrt{n}$ for all $(f,z) \in \mathcal{H}_{\mathbf{d}} \times (\mathbb{C}^{n+1})_*$. And, crucially, it controls the size of the basin of quadratic attraction of f around a zero ζ (see Bürgisser and Cucker 2011, Theorem 2.2). Let $d_{\mathbb{P}}$ denote the Riemannian distance in projective space $\mathbb{P}^n$ (*i.e.* the angle).

Theorem 3.38. Let $(f,\zeta) \in V$ and $x \in \mathbb{P}^n$ such that

$$d_{\mathbb{P}}(x,\zeta) \le \frac{3-\sqrt{7}}{D^{3/2}\mu_{\text{norm}}(f,\zeta)}.$$

Then x is an approximate zero of f with associated zero ζ. □

(c) The homotopy algorithm. It is this last theorem (along with a bound on the Lipschitz constant for μ_{norm}) that allows for a very simple choice of step-size in the homotopy, namely,

$$\Delta\tau := \frac{6.67 \times 10^{-3}}{d_{\mathbb{S}}(f,g) D^{3/2} \mu_{\text{norm}}^2(q_i, z_i)},$$

$$\tau_{i+1} := \min\{\tau_i + \Delta\tau, 1\},$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

where $d_{\mathbb{S}}(f,g)$ denotes the angle between f and g. Algorithm 2 – given a system f and an initial pair (g,ζ) – endowed with this choice of step-size returns an approximate zero of f. The following result (see Bürgisser and Cucker 2011, Theorem 3.1, or Bürgisser and Cucker 2013, Theorem 17.3) shows that this is so, and gives a first estimate of the cost of the homotopy.

Theorem 3.39. Suppose that the segment $[f,g]$ does not intersect the discriminant variety Σ. Then the homotopy stops after at most K iterations with

$$K := K(f,g,\zeta) :=\leq C\, D^{3/2}\, d_{\mathbb{S}}(f,g) \int_0^1 \mu^2_{\mathrm{norm}}(q_\tau,\zeta_\tau)\,\mathrm{d}\tau$$

for some constant C. The returned point z is an approximate zero of f with associated zero ζ_1. □

The number K of iterations performed by the algorithm is closely related to the total cost since it is easy to see (Bürgisser and Cucker 2013, Proposition 16.32) that each iteration is performed with $O(N)$ arithmetic operations.

We note that the ingredients in the integral in Theorem 3.39, the angular distances and the condition number μ_{norm} are scale-invariant. This is natural since the number of iterations of the homotopy algorithm is invariant under scaling of f and/or g.

(d) Random data and/or initial pairs. The bound in Theorem 3.39 depends on an integral which, in turn, depends on f and the pair (g,ζ). In order to obtain more concrete bounds one needs to eliminate this integral, a computation that requires either specific choices for f or (g,ζ) or assumes them drawn from a probability distribution. In the case of f, the latter amounts to working in the average complexity framework of Smale's 17th problem. In the case of (g,ζ), it allows for the use of randomized algorithms. For now, we will consider randomizing both f and the pair (g,ζ).

Note that to do the latter we need two additional ingredients: a probability measure on the solution manifold V and an algorithmic procedure to draw pairs from this measure.

For the first, we adopt the measure ρ_{st} (we call it the *standard distribution* on V) that is informally described by the following procedure:

`draw` g `from` $N(0,\mathrm{Id})$

`draw` ζ `from the uniform distribution on the` $\mathcal{D}$ `zeros of` g

Here $\mathcal{D} := \prod_{i=1}^n d_i$ is the Bézout number. Systems in $\mathcal{H}_{\mathbf{d}}$ have, with probability 1, $\mathcal{D}$ different zeros. While this description has the advantage of clarity, it does not suggest an algorithmic procedure to actually draw pairs from ρ_{st}: for this we should have to have at hand the $\mathcal{D}$ zeros of g!

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Such a procedure (along with the randomization approach for Smale's 17th problem) was devised by Beltrán and Pardo. It is described in Beltrán and Pardo (2011) (a more efficient variant can be found in Bürgisser and Cucker 2013, Section 17.6). It allows for a Las Vegas type algorithm – denoted by LV – that, with input $f \in \mathcal{H}_{\mathbf{d}}$, first draws (g, ζ) from V and then computes an approximate zero of f by passing f and the initial pair (g, ζ) to Algorithm 2.

The central question is: What is the expected complexity of LV? Here the expectation is with respect to both the input f and the pair (g, ζ). The expected number of iterations of the call to Algorithm 2 is then

$$\mathbb{E}_{f\sim N(0,\mathrm{Id})}\, \mathbb{E}_{(g,\zeta)\sim\rho_{\mathsf{st}}} K(f, g, \zeta),$$

a quantity that, using Theorem 3.39 and the description of ρ_{st}, can be bounded by

$$CD^{3/2} d_{\mathbb{S}}(f, g) \mathbb{E}_{f\sim N(0,\mathrm{Id})}\, \mathbb{E}_{g\sim N(0,\mathrm{Id})} \frac{1}{\mathcal{D}} \sum_{j=1}^{\mathcal{D}} \int_0^1 \mu^2_{\mathrm{norm}}(q_\tau, \zeta_\tau^{(j)})\, \mathrm{d}\tau. \tag{3.20}$$

Here $\{(q_\tau, \zeta_\tau^{(j)})\}_{\tau\in[0,1]}$ is the lifting of the segment $[g, f]$ corresponding to the jth zero of g. To further analyse (3.20), we note that neither f nor g explicitly appears in q_τ. They both appear implicitly through a complicated function of τ. This function can be simplified by changing the parametrization of the segment $[g, f]$ from the ratio τ of angles to a ratio t of Euclidean distances. Elementary geometry shows that this change of parametrization is given by

$$t(\tau) = \frac{\|g\|}{\|f\|(\sin\alpha \cot(\tau\alpha) - \cos\alpha) + \|g\|},$$

where $\alpha = d_{\mathbb{S}}(f, g)$; elementary calculus shows that (3.20) is bounded by

$$CD^{3/2} d_{\mathbb{S}}(f, g) \mathbb{E}_{f\sim N(0,\mathrm{Id})}\, \mathbb{E}_{g\sim N(0,\mathrm{Id})} \|f\|\|g\| \frac{1}{\mathcal{D}} \sum_{j=1}^{\mathcal{D}} \int_0^1 \frac{\mu^2_{\mathrm{norm}}(q_t, \zeta_t^{(j)})}{\|q_t\|^2}\, \mathrm{d}t.$$

A further argument (involving Gaussian distributions truncated at $\sqrt{2N}$ and scale invariance: see Bürgisser and Cucker 2011 for details) shows that this last expression is bounded by

$$2CD^{3/2} d_{\mathbb{S}}(f, g) N \mathbb{E}_{f\sim N(0,\mathrm{Id})}\, \mathbb{E}_{g\sim N(0,\mathrm{Id})} \frac{1}{\mathcal{D}} \sum_{j=1}^{\mathcal{D}} \int_0^1 \frac{\mu^2_{\mathrm{norm}}(q_t, \zeta_t^{(j)})}{\|q_t\|^2}\, \mathrm{d}t.$$

Further, Tonelli's theorem allows us to move the integral to the left of the expectations and obtain

$$2CD^{3/2} d_{\mathbb{S}}(f, g) N \int_0^1 \mathbb{E}_{f\sim N(0,\mathrm{Id})}\, \mathbb{E}_{g\sim N(0,\mathrm{Id})} \frac{\mu^2_{\mathsf{av}}(q_t)}{\|q_t\|^2}\, \mathrm{d}t, \tag{3.21}$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

where

$$\mu_{\mathsf{av}}^2(q_t) := \frac{1}{\mathcal{D}} \sum_{j=1}^{\mathcal{D}} \mu_{\mathrm{norm}}^2(q_t, \zeta_t^{(j)})$$

is the *mean square condition number* of q_t. The advantage of (3.21) is that, for a fixed $t \in [0,1]$, the distribution of $q_t = tf + (1-t)g$ is Gaussian with centre 0 and covariance matrix $(t^2+(1-t)^2)\mathrm{Id}$. That is, we can replace (3.21) with the simpler expression

$$2CD^{3/2} d_{\mathbb{S}}(f,g) N \int_0^1 \mathbb{E}_{q_t \sim N(0,(t^2+(1-t)^2)\mathrm{Id})} \frac{\mu_{\mathsf{av}}^2(q_t)}{\|q_t\|^2} \, \mathrm{d}t. \tag{3.22}$$

We are therefore left with a need to bound the expectation in this expression (for a fixed $t \in [0,1]$). Before proceeding with this task, though, it will be useful to discuss the more involved framework of a deterministic choice of the initial system g.

For the time being, we will still assume such a g in the sphere $\mathbb{S}(\mathcal{H}_{\mathbf{d}})$ and a zero ζ of g which is randomly chosen (as is the input f). In this case, the reasoning above, replacing the starting bound (3.20) with

$$CD^{3/2} d_{\mathbb{S}}(f,g) \mathbb{E}_{f \sim N(0,\mathrm{Id})} \frac{1}{\mathcal{D}} \sum_{j=1}^{\mathcal{D}} \int_0^1 \mu_{\mathrm{norm}}^2(q_\tau, \zeta_\tau^{(j)}) \, \mathrm{d}\tau,$$

ends with the bound

$$CD^{3/2} d_{\mathbb{S}}(f,g) \sqrt{2N} \|g\| \int_0^1 \mathbb{E}_{q_t \sim N((1-t)g, t^2\mathrm{Id})} \frac{\mu_{\mathsf{av}}^2(q_t)}{\|q_t\|^2} \, \mathrm{d}t. \tag{3.23}$$

Again, we are left with a need to bound the expectation in this expression, but this time the expectation leads to a smoothed (as opposed to average) analysis.

(e) Smoothed analysis of $\mu_{\mathsf{av}}^2(q)/\|q\|^2$. This is the most technically involved part of the example. The result (see Bürgisser and Cucker 2011 § 8, for details of the proof) is as follows.

Theorem 3.40. For $\overline{q} \in \mathcal{H}_{\mathbf{d}}$ and $\sigma > 0$ we have

$$\mathbb{E}_{q \sim N(\overline{q}, \sigma^2 \mathrm{Id})} \left(\frac{\mu_{\mathsf{av}}^2(q)}{\|q\|^2} \right) \leq \frac{e(n+1)}{2\sigma^2}.$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

Sketch of the proof. For $\zeta \in (\mathbb{C}^{n+1})_*$ we consider the subspace R_ζ of $\mathcal{H}_{\mathbf{d}}$ consisting of all systems h that vanish at ζ to higher order:

$$R_\zeta := \{h \in \mathcal{H}_{\mathbf{d}} \mid h(\zeta) = 0, Dh(\zeta) = 0\}.$$

We further decompose the orthogonal complement $R_\zeta^\perp$ of R_ζ in $\mathcal{H}_{\mathbf{d}}$. Let L_ζ denote the subspace of $R_\zeta^\perp$ consisting of the systems vanishing at ζ, and let C_ζ denote its orthogonal complement in $R_\zeta^\perp$. Then we have an orthogonal decomposition

$$\mathcal{H}_{\mathbf{d}} = C_\zeta \oplus L_\zeta \oplus R_\zeta \tag{3.24}$$

parametrized by $\zeta \in (\mathbb{C}^{n+1})_*$. Note that the spaces C_ζ, L_ζ, and R_ζ depend only on $[\zeta] \in \mathbb{P}^n$.

Let $\varphi_{\mathcal{H}_{\mathbf{d}}}$ denote the density of the Gaussian $N(\overline{q}, \sigma^2\mathrm{Id})$ on $\mathcal{H}_{\mathbf{d}}$, where $\overline{q} \in \mathcal{H}_{\mathbf{d}}$ and $\sigma > 0$. For fixed $\zeta \in \mathbb{S}(\mathbb{C}^{n+1})$ we decompose the mean $\overline{q}$ as

$$\overline{q} = \overline{k}_\zeta + \overline{g}_\zeta + \overline{h}_\zeta \in C_\zeta \oplus L_\zeta \oplus R_\zeta$$

according to (3.24). If we denote by φ_{C_ζ}, φ_{L_ζ}, and φ_{R_ζ} the densities of the Gaussian distributions in the spaces C_ζ, L_ζ, and R_ζ with covariance matrices $\sigma^2\mathrm{Id}$ and means $\overline{k}_\zeta, \overline{M}_\zeta$, and $\overline{h}_\zeta$, respectively, then the density $\varphi_{\mathcal{H}_{\mathbf{d}}}$ factors as

$$\varphi_{\mathcal{H}_{\mathbf{d}}}(k + g + h) = \varphi_{C_\zeta}(k) \cdot \varphi_{L_\zeta}(g) \cdot \varphi_{R_\zeta}(h). \tag{3.25}$$

Let $\mathscr{M} := \mathbb{C}^{n\times(n+1)}$ and consider the manifold

$$\hat{W} := \{(M, \zeta) \in \mathscr{M} \times \mathbb{S}(\mathbb{C}^{n+1}) \mid M\zeta = 0\},$$

whose projection $p_2 \colon \hat{W} \to \mathbb{S}(\mathbb{C}^{n+1}), (M, \zeta) \mapsto \zeta$, has the fibres

$$W_\zeta := \{M \in \mathscr{M} \mid M\zeta = 0\}.$$

One can prove (see Bürgisser and Cucker 2013, Proposition 16.16) that we have isometric linear maps

$$W_\zeta \to L_\zeta, \quad M \mapsto g_{M,\zeta} := \big(\sqrt{d_i}\,\langle X, \zeta\rangle^{d_i - 1} \textstyle\sum_j m_{ij} X_j\big). \tag{3.26}$$

Thus the Gaussian density φ_{L_ζ} on L_ζ induces a Gaussian density φ_{W_ζ} on the fibre W_ζ with covariance matrix $\sigma^2\mathrm{Id}$ via this map, so that $\varphi_{W_\zeta}(M) = \varphi_{L_\zeta}(g_{M,\zeta})$.

Now consider the lifting $\hat{V} \subseteq \mathcal{H}_{\mathbf{d}} \times \mathbb{S}(\mathbb{C}^{n+1})$ of the solution variety $V \subseteq \mathcal{H}_{\mathbf{d}} \times \mathbb{P}^n$ induced by the natural projection $\mathbb{S}(\mathbb{C}^{n+1}) \to \mathbb{P}^n$.

We can derive from the Gaussian distribution $\varphi_{\mathcal{H}_{\mathbf{d}}}$ on $\mathcal{H}_{\mathbf{d}}$ a probability distribution $\rho_{\hat{V}}$ on $\hat{V}$ (naturally extending $\rho_{\mathbf{st}}$) as follows. One draws (q, ζ) at random from $\hat{V}$ by first choosing $q \in \mathcal{H}_{\mathbf{d}}$ from $N(\overline{q}, \sigma^2\mathrm{Id})$, then choosing one of its $\mathcal{D}$ zeros $[\zeta] \in \mathbb{P}^n$ at random from the uniform distribution on

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

$\{1,\dots,\mathcal{D}\}$, and finally choosing a representative ζ in the unit circle $[\zeta]\cap\mathbb{S}^n$ uniformly at random.

One can then prove that

$$\mathbb{E}_{q\sim N(\bar{q},\sigma^2\mathrm{Id})}\left(\frac{\mu^2_{\mathsf{av}}(q)}{\|q\|^2}\right)=\mathbb{E}_{(q,\zeta)\sim\rho_{\hat{V}}}\left(\frac{\mu^2_{\mathrm{norm}}(q,\zeta)}{\|q\|^2}\right). \tag{3.27}$$

Put $\Delta:=\mathsf{diag}(\sqrt{d_i})$. Then the condition number $\mu_{\mathrm{norm}}(q,\zeta)$ can be described in terms of the linearization map

$$\Psi\colon\hat{V}\to\hat{W},\quad (q,\zeta)\mapsto(M,\zeta),\quad \text{where } M:=\Delta^{-1}Dq(\zeta),$$

as follows:

$$\frac{\mu_{\mathrm{norm}}(q,\zeta)}{\|q\|}=\|M^\dagger\|,\quad \text{where } (M,\zeta)=\Psi(q,\zeta).$$

Hence

$$\mathbb{E}_{(q,\zeta)\sim\rho_{\hat{V}}}\left(\frac{\mu^2_{\mathrm{norm}}(q,\zeta)}{\|q\|^2}\right)=\mathbb{E}_{M\sim\rho_{\mathscr{M}}}\big(\|M^\dagger\|^2\big), \tag{3.28}$$

where $\rho_{\mathscr{M}}$ denotes the pushforward density of $\rho_{\hat{V}}$ with respect to the map $p_1\circ\Psi\colon\hat{V}\to\mathscr{M}$.

A general identity in probability theory tells us that

$$\mathbb{E}_{M\sim\rho_{\mathscr{M}}}\big(\|M^\dagger\|^2\big)=\mathbb{E}_{\zeta\sim\rho_{\mathbb{S}(\mathbb{C}^{n+1})}}\big(\mathbb{E}_{M\sim\tilde{\rho}_{W_\zeta}}\big(\|M^\dagger\|^2\big)\big), \tag{3.29}$$

where $\rho_{\mathbb{S}(\mathbb{C}^{n+1})}$ is the pushforward density of $\rho_{\hat{V}}$ with respect to $p_2\circ\Psi\colon\hat{V}\to\mathbb{S}(\mathbb{C}^{n+1})$ and $\tilde{\rho}_{W_\zeta}$ denotes the conditional density on the fibre W_ζ. This conditional density turns out to be (and this is intricate) of the form

$$\tilde{\rho}_{W_\zeta}(M)=c_\zeta^{-1}\cdot\det(MM^*)\,\varphi_{W_\zeta}(M), \tag{3.30}$$

with c_ζ denoting a normalization factor. At this stage one can use invariance under the unitary group acting on $\mathbb{C}^{n+1}$ to reduce the study of the inner expectation in (3.29) to the case $\zeta=(1,0,\dots,0)$. In this case we can identify W_ζ with $\mathbb{C}^{n\times n}$, and $\tilde{\rho}_{W_\zeta}$ takes the form

$$\tilde{\rho}(A)=c^{-1}|\det A|^2\varphi(A),$$

where $c:=\mathbb{E}_{A\sim N(0,\mathrm{Id})}|\det A|^2$ and φ is the standard Gaussian density in $\mathbb{C}^{n\times n}$. The density $\tilde{\rho}$ is thus closely related to a standard Gaussian, and one can prove that

$$\mathbb{E}_{M\sim\tilde{\rho}_{W_\zeta}}\big(\|M^\dagger\|^2\big)=\mathbb{E}_{A\sim\tilde{\rho}}\big(\|A^{-1}\|^2\big)\le\frac{e(n+1)}{2\sigma^2}. \tag{3.31}$$

This implies by (3.29) that

$$\mathbb{E}_{M\sim\rho_{\mathscr{M}}}\big(\|M^\dagger\|^2\big)\le\frac{e(n+1)}{2\sigma^2},$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

and this bound, substituted in (3.28) and hence in (3.27), completes this sketch of the proof. □

(f) The complexity results. Theorem 3.40 can be easily used to obtain complexity bounds for LV.

Theorem 3.41. Algorithm LV returns an approximate zero of its input f almost surely. Its expected cost (number of arithmetic operations performed) is bounded by $O(D^{3/2}nN^2)$.

Proof. By Theorem 3.40 (with $\overline{q} = 0$) we have

$$\mathbb{E}_{q_t \sim N(0,(t^2+(1-t)^2)\mathrm{Id})} \frac{\mu_{\mathsf{av}}^2(q_t)}{\|q_t\|^2} \leq \frac{e(n+1)}{t^2+(1-t)^2}$$

and therefore

$$\int_0^1 \mathbb{E}_{q_t \sim N(0,(t^2+(1-t)^2)\mathrm{Id})} \frac{\mu_{\mathsf{av}}^2(q_t)}{\|q_t\|^2}\,\mathrm{d}t \leq \int_0^1 \frac{e(n+1)}{t^2+(1-t)^2}\,\mathrm{d}t = \frac{\pi e(n+1)}{2}.$$

Replacing this expression in the bound (3.22) for the expected number of iterations performed by the continuation shows that this is $O(D^{3/2}nN)$. And multiplying this bound by the $O(N)$ cost of each iteration finishes the proof. □

The analysis of deterministic algorithms is more lengthy. It requires the following result.

Theorem 3.42. For $g \in \mathbb{S}(\mathcal{H}_{\mathbf{d}}) \setminus \Sigma$ we have

$$CD^{3/2}d_{\mathbb{S}}(f,g)\sqrt{2N}\mathbb{E}_{f\sim N(0,\mathrm{Id})} \int_0^1 \frac{\mu_{\mathsf{av}}^2(q_t)}{\|q_t\|^2}\,\mathrm{d}t = O(D^3Nn\mu_{\max}^2(g)),$$

where $\mu_{\max}(g) := \max_{\zeta|g(\zeta)=0} \mu_{\mathrm{norm}}(g,\zeta)$.

Sketch of the proof. The starting point is the bound (3.23) for the left-hand side in the statement. Within this bound, the expression to be estimated is

$$\int_0^1 \mathbb{E}_{q_t \sim N((1-t)g,t^2\mathrm{Id})} \frac{\mu_{\mathsf{av}}^2(q_t)}{\|q_t\|^2}\,\mathrm{d}t.$$

The idea is to split the range $[0,1]$ of this integral into two subintervals $[0,t_0]$ and $[t_0,1]$ with a small t_0. Indeed, one takes $t_0 = t(\tau_0)$ with τ_0 proportional to

$$D^{3/2}\sqrt{N}\mu_{\max}^2(g).$$

In the interval $[0,t_0]$ the system q_t is close to g, and one proves that $\mu_{\mathsf{av}}^2(q_t)/\|q_t\|^2$ can be bounded by a small multiple of $\mu_{\max}^2(g)$. This implies the desired bound on the integral over $[0,t_0]$.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

In the interval $[t_0, 1]$ we use Theorem 3.40 to deduce

$$\begin{aligned}\int_{t_0}^{1} \mathbb{E}_{q_t \sim N((1-t)g, t^2\mathrm{Id})} \frac{\mu_{\mathsf{av}}^2(q_t)}{\|q_t\|^2}\,\mathrm{d}t &= \int_{t_0}^{1} \frac{e(n+1)}{t^2}\,\mathrm{d}t \\ &= e(n+1)\frac{1}{t}\Big|_1^{t_0} \\ &= O\big(D^{3/2} n\sqrt{N}\mu_{\max}^2(g)\big).\end{aligned}$$

Replacing this bound in (3.23) finishes the proof. □

Theorem 3.43. There is a deterministic algorithm that on input $f \in \mathcal{H}_{\mathbf{d}}$ computes an approximate zero of f in average time $N^{O(\log\log N)}$. Moreover, if we restrict data to polynomials satisfying

$$D \leq n^{\frac{1}{1+\varepsilon}} \quad \text{or} \quad D \geq n^{1+\varepsilon},$$

for some fixed $\varepsilon > 0$, then the average time of the algorithm is polynomial in the input size N.

Sketch of the proof. We proceed differently according to whether $D \leq n$. If $D \leq n$ the algorithm amounts to executing Algorithm 2 with initial pair $(\bar{U}, \mathbf{z}_1)$, where $\bar{U} \in \mathcal{H}_{\mathbf{d}}$ is given by

$$\bar{U}_1 = \frac{1}{\sqrt{2n}}(X_0^{d_1} - X_1^{d_1}), \ldots, \bar{U}_n = \frac{1}{\sqrt{2n}}(X_0^{d_n} - X_n^{d_n}) \tag{3.32}$$

(the scaling factor guaranteeing that $\|\bar{U}\| = 1$) and $\mathbf{z}_1 = (1, 1, \ldots, 1)$.

The reason for this choice is the strong presence of symmetries. These symmetries guarantee (at least) two features. Firstly,

$$\mu_{\mathrm{norm}}(\bar{U}, \mathbf{z}_j) = \mu_{\mathrm{norm}}(\bar{U}, \mathbf{z}_1), \quad \text{for all } j = 1, \ldots, \mathcal{D}.$$

Here $\mathbf{z}_1, \ldots, \mathbf{z}_{\mathcal{D}}$ are all the zeros of $\bar{U}$. Secondly (and more importantly),

$$\mathbb{E}_{f\sim N(0,\mathrm{Id})} K(f, \bar{U}, \mathbf{z}_1) = \mathbb{E}_{f\sim N(0,\mathrm{Id})} \frac{1}{\mathcal{D}} \sum_{j=1}^{\mathcal{D}} K(f, \bar{U}, \mathbf{z}_j).$$

That is, the average (with respect to f) number of iterations of Algorithm 2 with initial system $\bar{U}$ is the same no matter whether the zero of $\bar{U}$ is taken at random or set to be $\mathbf{z}_1$. This property, together with the bound

$$\mu_{\max}^2(\bar{U}) \leq 2\,(n+1)^D,$$

allows us to apply (3.23) and Theorem 3.42 to deduce that the average number of iterations is bounded by

$$O(D^3 N n^{D+1}).$$

For any fixed $\varepsilon > 0$ this bound is polynomial in N in the range $D \leq n^{1/(1+\varepsilon)}$. In the range $n^{1/(1+\varepsilon)} \leq D \leq n$ it is bounded by $N^{O(\log\log N)}$.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

If, instead, $D > n$, one can use classical algorithms for zero isolation (such as the one in Renegar 1987). These algorithms (adapted to return approximate zeros in the sense of Definition 3.36) have average complexity polynomial in N and $\mathcal{D}$. Again, for any fixed $\varepsilon > 0$ this is polynomial in N in the range $D \geq n^{1+\varepsilon}$, and in the range $n < D < n^{1+\varepsilon}$ it is bounded by $N^{O(\log\log N)}$. □

(g) Recent developments. The advances regarding Smale's 17th problem described above encouraged the same approach for the eigenpair problem, which will occupy us in the next example. Deterministic average polynomial time was not an issue in this case: it was cleanly achieved. But an additional goal presented itself, namely to minimize the degree of the (polynomial) complexity bound. Some of the new ideas that helped us towards this goal have recently been applied in the context of polynomial systems (Armentano *et al.* 2015*a*).

The basis of the new developments is a modification to the computation of the step-length in the homotopy that improves Theorem 3.39 as follows.

Theorem 3.44. Suppose that the segment $[f, g]$ does not intersect the discriminant variety Σ. Then the (modified) homotopy stops after at most K^* iterations with

$$K^* := K^*(f, g, \zeta) := C\, D^{3/2}\, d_{\mathbb{S}}(f, g) \int_0^1 \mu_{\mathrm{norm}}(q_\tau, \zeta_\tau) \|(\dot{q}_\tau, \dot{\zeta}_\tau)\|\, \mathrm{d}\tau$$

for some constant C. Here $(\dot{q}_\tau, \dot{\zeta}_\tau)$ is the tangent vector to the curve $\{(q_\tau, \zeta_\tau)\}_{\tau\in[0,1]} \subset V$ at the point (q_τ, ζ_τ). The returned point z is an approximate zero of f with associated zero ζ_1. □

Theorem 3.39 may be seen as a consequence of Theorem 3.44 since the very nature of the condition number $\mu_{\mathrm{norm}}(q_\tau, \zeta_\tau)$ is the fact that $\|\dot{\zeta}_\tau\|$ is bounded by (a small multiple of) $\mu_{\mathrm{norm}}(q_\tau, \zeta_\tau)\|\dot{q}_\tau\|$. This nature is commonly expressed in the sentence 'condition numbers measure the worst-case magnification in the output of a perturbation in the input'. The new idea is to bound the *expected* magnification in $\|\dot{\zeta}_\tau\|$ instead of the worst-case one. This leads to the notion of *stochastic condition number*, which goes back to the work of Fletcher (1985) and was further pursued by Weiss, Wasilkowski, Woźniakowski and Shub (1986), Stewart (1990) and Armentano (2010).

For $f \in \mathbb{S}(\mathcal{H}_{\mathbf{d}})$, let $T_f\mathbb{S}(\mathcal{H}_{\mathbf{d}})$ denote the tangent space at f on $\mathbb{S}(\mathcal{H}_{\mathbf{d}})$ and let us denote by $\mathbb{S}(T_f)$ the unit sphere in this tangent space. In the case that both f and g are drawn from the uniform measure in $\mathbb{S}(\mathcal{H}_{\mathbf{d}})$ (which is equivalent, in our context, to drawing them both from $N(0, \mathrm{Id})$) one can

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

prove that

$$d_{\mathbb{S}}(f,g)\mathbb{E}_{f,g\in\mathbb{S}(\mathcal{H}_{\mathbf{d}})}\int_0^1 \frac{1}{\mathcal{D}}\sum_{\zeta_\tau|q_\tau(\zeta_\tau)=0}\mu_{\mathrm{norm}}(q_\tau,\zeta_\tau)\|(\dot{q}_\tau,\dot{\zeta}_\tau)\|\,\mathrm{d}\tau$$
$$=\frac{\pi}{2}\,\mathbb{E}_{f\in\mathbb{S}(\mathcal{H}_{\mathbf{d}})}\,\mathbb{E}_{\dot{f}\in\mathbb{S}(T_f)}\,\frac{1}{\mathcal{D}}\sum_{\zeta|f(\zeta)=0}\mu_{\mathrm{norm}}(f,\zeta)\|(\dot{f},\dot{\zeta})\|,$$

where both spheres are endowed with the uniform distribution. It is this uniformity on $\mathbb{S}(T_f)$ which allows one, for a fixed $f\in\mathcal{H}_{\mathbf{d}}$ and $\zeta\in\mathbb{P}^n$ with $f(\zeta)=0$, to look at $\mathbb{E}_{\dot{f}\in\mathbb{S}(T_f)}\|(\dot{q}_\tau,\dot{\zeta}_\tau)\|$ as the stochastic condition number of the pair (f,ζ) and use the techniques in Armentano (2010) to estimate this expectation. This estimate, together with Theorem 3.40 (with $\overline{q}=0$ and $\sigma=1$) are finally combined to prove the following improvement to Theorem 3.41.

Theorem 3.45. Algorithm LV (with the modified homotopy routine) returns an approximate zero of its input f almost surely. Its expected cost is bounded by $O(D^{3/2}nN^{3/2})$. □

3.9. The eigenvalue problem

The second example we want to mention – of a homotopy method for which complexity estimates are obtained via a probabilistic analysis of condition numbers – has to do with the computation of eigenpairs (*i.e.* eigenvalue–eigenvector pairs) for complex matrices.

The state of the art for the problem was, until very recently, lacunary in the sense that the existing algorithms were either well understood regarding complexity but numerically unstable in practice or they exhibited stable behaviour but defied complexity analysis. The best-known example of the former is that of computing the characteristic polynomial χ_A of the $n\times n$ input matrix A and then approximating the zeros $\lambda_1,\ldots,\lambda_n$ of χ_A. Both ingredients are known to have low complexity but the whole procedure often fails due to numerical instability. A perfect example of the latter (and today's algorithm of choice in practice) is the QR iteration with implicit shifts, about which Jim Demmel (1997, p. 139) wrote:

> It is interesting that after more than 30 years of dependable service, convergence failures of this algorithm have quite recently been observed, analysed, and patched ... But there is still no global convergence proof, even though the current algorithm is considered quite reliable. So the problem of devising an algorithm that is numerically stable and globally (and quickly!) convergent remains open.

An answer to this demand was very recently given in Armentano *et al.* (2015*b*), where a homotopy method for computing eigenpairs is analysed and shown to have polynomial-time complexity on average. Homotopy methods

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

had already been considered for eigenvalue problems (see *e.g.* Li and Sauer 1987) but these previous works did not provide any form of complexity analysis.

We give one final remark before proceeding with the technical core of this example. Since the eigenpair problem has an algebraic nature, one may wonder whether the algorithms for zero finding in the previous example can be applied. Indeed, if we let a_i denote the rows of matrix A, the equality $Av = \lambda v$ can be written as n quadratic equations $a_i v - \lambda v_i = 0$, $i = 1, \ldots, n$, and we may add the extra equation $v_1^2 + \ldots + v_n^2 = 1$ ensuring that $v \neq 0$. We thus have a system of $n+1$ equations in $n+1$ variables which, once homogenized, could be considered as an input for algorithm LV, for example. Unfortunately, though, with high probability, the execution will loop forever. The reason is that, generically, a system of $n+1$ quadratic equations has 2^{n+1} solutions. In our case, n of these are the eigenpairs we are looking for and the remaining $2^{n-1} - n$ are ill-posed zeros at infinity (*i.e.* with $\lambda = \infty$). Following a path corresponding to one of the latter will take infinite time.

(a) The setting. Given a matrix $A \in \mathbb{C}^{n\times n}$, we want to compute an *eigenpair* of A, that is, a pair $(\lambda, v) \in \mathbb{C} \times (\mathbb{C}^n \setminus \{0\})$ such that $Av = \lambda v$. Clearly, if (λ, v) is such a pair then, for all $c \in \mathbb{C} \setminus \{0\}$, the pair (λ, cv) is also an eigenpair of A. Therefore we consider eigenpairs in the product $\mathbb{C} \times \mathbb{P}(\mathbb{C}^n)$. A matrix $A \in \mathbb{C}^{n\times n}$ has, generically, n different eigenpairs in this product space.

The space $\mathbb{C}^{n\times n} \times \mathbb{C} \times \mathbb{P}(\mathbb{C}^n)$ is endowed with the Riemannian product structure. The resulting distance equals

$$\mathsf{dist}((A, \lambda, v), (A', \lambda', v'))^2 := \|A - A'\|_F^2 + |\lambda - \lambda'|^2 + d_{\mathbb{P}}(v, v')^2.$$

We will only use this distance on $\mathbb{S} \times \mathbb{C} \times \mathbb{P}(\mathbb{C}^n)$. Note that for $A, A' \in \mathbb{S}$, the distance $\mathsf{dist}((A, \lambda, v), (A', \lambda', v'))$ is smaller than or equal to the natural geodesic (product) distance in $\mathbb{S} \times \mathbb{C} \times \mathbb{P}(\mathbb{C}^n)$. For any non-zero matrix $A \in \mathbb{C}^{n\times n}$ (not necessarily of unit norm) we will write

$$\mathsf{dist}_A((\lambda, v), (\lambda', v'))^2 := \frac{|\lambda - \lambda'|^2}{\|A\|_F^2} + d_{\mathbb{P}}(v, v')^2.$$

Note that for any non-zero $A \in \mathbb{C}^{n\times n}$, dist_A is a distance function in $\mathbb{C} \times \mathbb{P}(\mathbb{C}^n)$ and if $A \in \mathbb{S}$, then

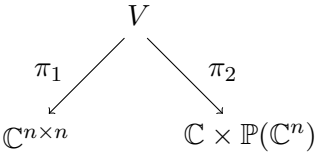
$$\mathsf{dist}_A((\lambda, v), (\lambda', v')) = \mathsf{dist}((A, \lambda, v), (A, \lambda', v')).$$

A number of notions are defined as in the previous example. Thus we consider the *solution manifold*

$$V = V_n := \big\{(A, \lambda, v) \in \mathbb{C}^{n\times n} \times \mathbb{C} \times \mathbb{P}(\mathbb{C}^n) \mid (A - \lambda \,\mathrm{Id})v = 0\big\},$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

with its natural projections



as well as the *set* Σ' *of ill-posed triples* $(A, \lambda, v) \in V$ for which $D\pi_1(A, \lambda, v)$ is not an isomorphism, and the set $\Sigma := \pi_1(\Sigma')$ of *ill-posed matrices*.

For $v \in \mathbb{P}(\mathbb{C}^n)$ we let $v^\perp = \{x \in \mathbb{C}^n \mid \langle x, v\rangle = 0\}$ denote the tangent space to $\mathbb{P}(\mathbb{C}^n)$ at v. Let $P_{v^\perp} \colon \mathbb{C}^n \to v^\perp$ be the orthogonal projection. Given $(A, \lambda, v) \in \mathbb{C}^{n\times n} \times \mathbb{C} \times \mathbb{P}(\mathbb{C}^n)$, we let $A_{\lambda,v} \colon v^\perp \to v^\perp$ be the linear operator given by

$$A_{\lambda,v} := P_{v^\perp} \circ (A - \lambda \operatorname{Id})|_{v^\perp}. \tag{3.33}$$

One can then prove (Armentano 2014, Lemma 2.7) that a triple $(A, \lambda, v) \in V$ is well-posed if and only if $A_{\lambda,v}$ is invertible, which occurs if and only if λ is a single eigenvalue of A.

One can also devise, for a non-zero matrix $A \in \mathbb{C}^{n\times n}$, a *Newton map* $N_A : \mathbb{C} \times \mathbb{P}(\mathbb{C}^n) \to \mathbb{C} \times \mathbb{P}(\mathbb{C}^n)$ associated with A by

$$N_A\begin{pmatrix}\lambda\\ v\end{pmatrix} = \begin{pmatrix}\lambda\\ v\end{pmatrix} - \begin{pmatrix}\dot\lambda\\ \dot v\end{pmatrix}, \quad \text{where } \begin{pmatrix}\dot\lambda\\ \dot v\end{pmatrix} = \big(DF_A(\lambda, v)\,|_{\mathbb{C}\times v^\perp}\big)^{-1} F_A\begin{pmatrix}\lambda\\ v\end{pmatrix}$$

and $F_A(\lambda, v) = (A - \lambda \operatorname{Id})v$ is the evaluation map. This map was introduced in Armentano (2014) as a Newton-like operator associated with the evaluation map F_A, and the following formulas were obtained for $\dot v$ and $\dot\lambda$ (recall the definition of $A_{\lambda,v}$ from (3.33)):

$$\dot v = {A_{\lambda,v}}^{-1} P_{v^\perp} A v, \quad \dot\lambda = \frac{\langle (\lambda \operatorname{Id} - A)(v - \dot v), v\rangle}{\langle v, v\rangle}. \tag{3.34}$$

It is immediate to see that $N_A(\lambda, v)$ does not depend on the representative in $\mathbb{C}^n \setminus \{0\}$ for $v \in \mathbb{P}(\mathbb{C}^n)$. We also note that $N_A(\zeta, w)$ can be computed from the matrix A and the pair (ζ, w) in $O(n^3)$ operations, since the cost of this computation is dominated by that of inverting a matrix (or simply solving a linear system).

With Newton's algorithm at our disposal the following definition is natural.

Definition 3.46. Given $(A, \lambda, v) \in V \setminus \Sigma'$ we say that $(\zeta, w) \in \mathbb{C} \times \mathbb{P}(\mathbb{C}^n)$ is an *approximate eigenpair* of A with associated eigenpair (λ, v) when, for all $k \geq 1$, the kth iterate $N_A^k(\zeta, w)$ of the Newton map at (ζ, w) is well

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

defined and satisfies

$$\mathsf{dist}_A\big((N_A^k(\zeta,w)),(\lambda,v)\big) \le \left(\frac{1}{2}\right)^{2^k-1} \mathsf{dist}_A\big((\zeta,w),(\lambda,v)\big).$$

The following result (for a proof see Armentano 2014, Theorem 5) is a version of Theorem 3.38.

Theorem 3.47. There is a constant $c_0 > 1/5$ with the following property. Let $(A,\lambda,v) \in V \setminus \Sigma'$ with $\|A\|_F = 1$ and let $(\zeta,w) \in \mathbb{C}\times\mathbb{P}(\mathbb{C}^n)$. If

$$\mathsf{dist}_A\big((\lambda,v),(\zeta,w)\big) \le \frac{c_0}{\mu(A,\lambda,v)},$$

then (ζ,w) is an approximate eigenpair of A with associated eigenpair (λ,v). □

(b) Condition numbers. The framework we described in Section 1.1(b) along with the projections

$$\begin{array}{ccc} & \mathbb{C}\times\mathbb{P}(\mathbb{C}^n) & \\ {\scriptstyle\pi_\lambda}\swarrow & & \searrow{\scriptstyle\pi_v} \\ \mathbb{C} & & \mathbb{P}(\mathbb{C}^n) \end{array}$$

allow us to define two condition numbers for eigenvalue and eigenvector, respectively. Formulas for both of them have been known at least since Van Loan (1987). Armentano (2014, § 3) has shown that one can merge the two, which, with minor changes, leads to the definition of the *condition number* of $(A,\lambda,v) \in \mathbb{C}^{n\times n}\times\mathbb{C}\times\mathbb{P}(\mathbb{C}^n)$ in Armentano *et al.* (2015*b*), namely

$$\mu(A,\lambda,v) := \|A\|_F \|A_{\lambda,v}^{-1}\|, \tag{3.35}$$

(or ∞ if $A_{\lambda,v}$ is not invertible) where $\|\cdot\|$ is the operator norm. This coincides with $\mu_v(A,\lambda,v)$ in Armentano (2014). When restricted to the class of normal matrices, the condition number μ admits the following elegant characterization.

Lemma 3.48 (Armentano 2014, Lemma 3.12). Let $A \in \mathbb{C}^{n\times n}\setminus\Sigma$ be normal, and let $(\lambda_1,v_1),\dots,(\lambda_n,v_n)$ be its eigenpairs. Then

$$\mu(A,\lambda_1,v_1) = \frac{\|A\|_F}{\min_{j=2,\dots,n}|\lambda_j-\lambda_1|}.$$ □

As in the case of polynomial systems, we can define the the *mean square condition number* of A by

$$\mu_{\mathsf{av}}^2(A) := \frac{1}{n}\sum_{j=1}^n \mu^2(A,\lambda_j,v_j).$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

The version for the eigenpair problem of Theorem 3.40 is as follows (see Armentano *et al.* 2015*b*, Theorem 2.14).

Theorem 3.49. For $\overline{A} \in \mathbb{C}^{n\times n}$ and $\sigma > 0$ we have

$$\mathbb{E}_{A\sim N(\overline{A},\sigma^2\mathrm{Id})}\left(\frac{\mu^2_{\mathsf{av}}(A)}{\|A\|_F^2}\right) \leq \frac{n}{\sigma^2}. \qquad \square$$

(c) Complexity results: deterministic algorithms. Theorems 3.47 and 3.49 allow us to follow the main reasoning in the previous example to design and analyse homotopy algorithms for the eigenpair problem. Indeed, as in the previous example, one can take the step-length in the homotopy to be proportional to the inverse of $\mu^2(B_i, \zeta_i, w_i)$, where B_i is the matrix dealt with at the ith iteration and (ζ_i, w_i) is an approximate eigenpair of B_i. A result like Theorem 3.39 (without the $D^{3/2}$ factor) likewise follows, and an appropriate choice of initial triples leads to the following two results (for computing one or all of the eigenpairs of a matrix).

Theorem 3.50 (Armentano *et al.* 2015*b*, Theorem 2.24). There exists a homotopy algorithm Single_Eigenpair which returns (almost surely) an approximate eigenpair of its input $A \in \mathbb{C}^{n\times n}$. Its average cost is bounded by $O(n^7)$. $\square$

Theorem 3.51 (Armentano *et al.* 2015*b*, Theorem 2.27). There exists a homotopy algorithm All_Eigenpairs which returns (almost surely) n approximate eigenpairs of its input $A \in \mathbb{C}^{n\times n}$ with pairwise different associated eigenpairs. Its average cost is bounded by $O(n^9)$. $\square$

(d) Complexity results: randomized algorithms. The ideas described in Section 3.8(g) can be used in the context of the eigenpair problem. A modification to the computation of the step-length of the homotopy leads to the following result (Armentano *et al.* 2015*b*, Theorem 2.28).

Proposition 3.52. The expected (with respect to both the input matrix A and the initial triple (A_0, λ_0, v_0)) number of homotopy steps satisfies

$$\mathbb{E}_{A\sim N(0,\mathrm{Id})}\mathbb{E}_{A_0\sim N(0,\mathrm{Id})}\ \frac{1}{n}\sum_{(\lambda_0,v_0)|(A_0,\lambda_0,v_0)\in V} K(A, A_0, \lambda_0, v_0) \leq Cn^2,$$

where C is a universal constant. $\square$

If we had a procedure to efficiently draw triples from the distribution in V informally described by

```
randomly choose A_0 ~ N(0, Id)
randomly choose one eigenpair (λ_0, v_0) of A_0
```

we could design a Las Vegas algorithm like the one in the previous example, computing a single eigenpair of its input matrix in average randomized

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

time $O(n^5)$ (since each iteration of the homotopy has a cost of $O(n^3)$). As it happens, we do not have such a procedure. Nonetheless, a randomized algorithm can be designed whose analysis takes advantage of Proposition 3.52. This analysis, however, has an overhead cost that eventually leads to an average randomized cost of $O(n^7)$ (Armentano *et al.* 2015*b*, Theorem 2.34).

3.10. General bounds for conic condition numbers

The last example in this repertoire is not, strictly speaking, an example but rather a general result from which many examples can be derived.

Recall that at the beginning of Section 2 we defined *conic* condition numbers as those that admit an expression of the form

$$\mathscr{C}(a) := \frac{\|a\|}{\mathsf{dist}(a, \Sigma)}.$$

In what follows we assume that the data $a \in \mathbb{K}^{p+1}$ where $\mathbb{K}$ is $\mathbb{R}$ or $\mathbb{C}$ and that the norm and distance above are induced by an inner (resp. Hermitian) product. We also assume that the set Σ of ill-posed instances is an algebraic cone in $\mathbb{K}^{p+1}$, that is, a Zariski closed subset that is closed under multiplication by elements in $\mathbb{K}$. The main contention of Demmel (1987) is that many (if not most) of the condition numbers we associate with a variety of problems are – or are closely approximated by – a conic condition number as above.

The fact that Σ is a cone implies that for all $a \in \mathbb{K}^{p+1}$ and all $\lambda \in \mathbb{K} \setminus \{0\}$ we have $\mathscr{C}(a) = \mathscr{C}(\lambda a)$. Hence, we may restrict attention to data $a \in \mathbb{P}^p := \mathbb{P}(\mathbb{K}^{p+1})$ in projective space over $\mathbb{K}$ for which the condition number takes the form

$$\mathscr{C}(a) = \frac{1}{d_{\mathsf{sin}}(a, \Sigma)}, \tag{3.36}$$

where, abusing notation, Σ is interpreted now as a subset of $\mathbb{P}^p$ and $d_{\mathsf{sin}} = \sin d_{\mathbb{P}}$ denotes the sine distance in $\mathbb{P}^p$ (see Figure 3.4). Because Σ is a projective variety, it has associated with it a number of geometric invariants, notably a *dimension* $m = \dim_{\mathbb{K}} \Sigma$ and a *degree* $d = \mathsf{deg}\Sigma$.

In a sequel to his 1987 paper, Jim Demmel (1988) endowed projective space $\mathbb{P}^p$ with the uniform distribution and exhibited bounds for the probability tail of $\mathscr{C}$ depending only on the parameters p, m and d. Estimates for the expectation of $\mathscr{C}$ or of $\log \mathscr{C}$ (also depending on these parameters) follow from these bounds using well-known arguments. This average analysis was extended to a smoothed analysis in Bürgisser, Cucker and Lotz (2006, 2008) but, unlike the previous examples, not for Gaussian perturbations but for perturbations uniformly chosen on balls in projective space.

Since the total volume of $\mathbb{P}^p$ for its Riemannian measure is finite, it makes sense to talk about the uniform probability distribution on the closed ball

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

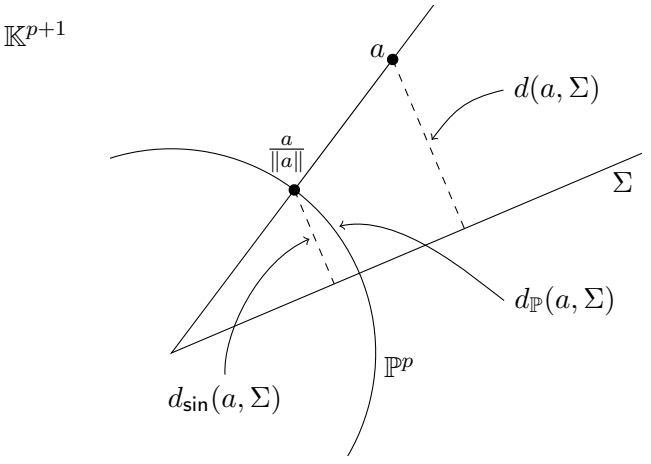


Figure 3.4. Three distances.

$B(\overline{a},\sigma)$ of radius σ around $\overline{a} \in \mathbb{P}^p$ with respect to $d_{\sin}$. We thus consider a smoothed analysis of $\mathscr{C}$ whereby a random perturbation a of $\overline{a} \in \mathbb{P}^p$ is modelled by choosing $a \in B(\overline{a},\sigma)$ from the uniform distribution on $B(\overline{a},\sigma)$.

The case $\mathbb{K} = \mathbb{C}$ allows for a very general result, the only restriction being that the set of ill-posed inputs Σ is purely dimensional, that is, all of its irreducible components have the same dimension. This is almost invariably the case for common computational problems. The version of this result in Bürgisser and Cucker (2013, Theorem 20.1) reads as follows.

Theorem 3.53. Let $\mathscr{C}$ be a conic condition number with a set of ill-posed inputs $\Sigma \subseteq \mathbb{P}(\mathbb{C}^{p+1})$, of pure dimension m, $0 < m < p$. Let

$$K(p,m) := 2\frac{p^{3p}}{m^{3m}(p-m)^{3(p-m)}}.$$

Then, for all $\overline{a} \in \mathbb{P}(\mathbb{C}^{p+1})$, all $\sigma \in (0,1]$, and all

$$t \geq \frac{p\sqrt{2m}}{p-m},$$

we have

$$\operatorname*{Prob}_{a\in B(\overline{a},\sigma)}\{\mathscr{C}(a) \geq t\} \leq K(p,m)\,\mathsf{deg}\Sigma\left(\frac{1}{t\sigma}\right)^{2(p-m)}\left(1+\frac{p}{p-m}\frac{1}{t\sigma}\right)^{2m},$$

as well as

$$\mathbb{E}_{a\in B(\overline{a},\sigma)}(\mathscr{C}(a)) \leq 2eK(p,m)^{\frac{1}{2(p-m)}}\left(\mathsf{deg}\Sigma\right)^{\frac{1}{2(p-m)}}\frac{pm}{p-m}\frac{1}{\sigma}$$

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

and

$$\mathbb{E}_{a\in B(\bar{a},\sigma)}(\log \mathscr{C}(a)) \leq \frac{1}{2(p-m)}\big(\log K(p,m) + \log \mathsf{deg}\Sigma + 3\log e\big) + \log\frac{pm}{p-m} + \log\frac{1}{\sigma}.$$

Taking $\sigma = 1$, we obtain an average-case analysis. □

In most applications, the set of ill-posed inputs Σ is a hypersurface. That is, Σ is the zero set $Z_{\mathbb{P}}(f)$ of a non-zero homogeneous polynomial f, and thus $\mathsf{deg}\Sigma$ is at most the degree of f. In this case, we have the following corollary.

Corollary 3.54. Let $\mathscr{C}$ be a conic condition number with set of ill-posed inputs $\Sigma \subseteq \mathbb{P}(\mathbb{C}^{p+1})$. Assume $\Sigma \subseteq Z_{\mathbb{P}}(f)$ with $f \in \mathbb{C}[X_0,\dots,X_p]$ homogeneous of degree d. Then, for all $\bar{a} \in \mathbb{P}(\mathbb{C}^{p+1})$, all $\sigma \in (0,1]$, and all $t \geq \sqrt{2}\,p^{3/2}$,

$$\operatorname*{Prob}_{a\in B(\bar{a},\sigma)}\{\mathscr{C}(a) \geq t\} \leq 2e^3p^3d\left(\frac{1}{t\sigma}\right)^2\left(1+p\frac{1}{t\sigma}\right)^{2(p-1)}$$

as well as

$$\mathbb{E}_{a\in B(\bar{a},\sigma)}(\mathscr{C}(a)) \leq 4(2e^5)^{1/2}p^{3/2}\,d^{1/2}\,\frac{1}{\sigma}$$

and

$$\mathbb{E}_{a\in B(\bar{a},\sigma)}(\log \mathscr{C}(a)) \leq \frac{3}{2}\log p + \frac{1}{2}\log d + \log\frac{1}{\sigma} + 3\log e + \frac{3}{2}.$$

Taking $\sigma = 1$, one obtains an average-case analysis. □

The case $\mathbb{K} = \mathbb{R}$ turns out to be more complicated. Demmel (1988) states a result which he attributed to Adrian Ocneanu, estimating the volume of ε-tubes around real algebraic varieties, and makes a conjecture on the asymptotic behaviour of these volumes when ε tends to zero. Ocneanu's result, however, is cited as an unpublished report and, as of today, it remains so. Very recently, Martin Lotz (2015) gave a proof of this result and, in passing, proved Demmel's conjecture. But he did not derive a probabilistic analysis for conic condition numbers. Moreover, as well as the condition on pure dimensionality assumed in the complex case, these results impose a more stringent requirement, namely that the real algebraic variety is a complete intersection.

The general bound in Bürgisser, Cucker and Lotz (2008) (which we cite again from the more recent version in Bürgisser and Cucker 2013) is weaker than its complex counterpart.

Theorem 3.55. Let $\mathscr{C}$ be a conic condition number with a set of ill-posed inputs Σ. Assume that Σ is contained in a real algebraic hypersurface, given

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

as the zero set of a homogeneous polynomial of degree d. Then, for all $0 < \sigma \leq 1$ and all $t \geq (2d+1)(p/\sigma)$, we have

$$\sup_{\overline{a} \in \mathbb{S}^p} \operatorname*{Prob}_{a \in B_{\sin}(\overline{a},\sigma)} \{\mathscr{C}(a) \geq t\} \leq 4e\, \mathrm{d}p\, \frac{1}{\sigma t}$$

and

$$\sup_{\overline{a} \in \mathbb{S}^p} \mathbb{E}_{a \in B_{\sin}(\overline{a},\sigma)} \log \mathscr{C}(a) \leq \log p + \log d + \log \frac{1}{\sigma} + \log(4e^2).$$

In particular (take $\sigma = 1$), for all $t \geq (2d+1)p$,

$$\operatorname*{Prob}_{a \in \mathbb{S}^p} \{\mathscr{C}(a) \geq t\} \leq 4e\, \mathrm{d}p\, \frac{1}{t}$$

and

$$\mathbb{E}_{a \in \mathbb{S}^p} (\log \mathscr{C}(a)) \leq \log p + \log d + \log(4e^2). \qquad \square$$

Very likely, one can extend Theorem 3.55 to obtain probability tails with a rate of decay of the order of t^{p-m} in the case when Σ is a complete intersection, by adapting the results in Lotz (2015).

REFERENCES[1]

D. Amelunxen and P. Bürgisser (2012), 'Robust smoothed analysis of a condition number for linear programming', *Math. Program.* **131**, 221–251.

D. Armentano (2010), 'Stochastic perturbations and smooth condition numbers', *J. Complexity* **26**, 161–171.

D. Armentano (2014), 'Complexity of path-following methods for the eigenvalue problem', *Found. Comput. Math.* **14**, 185–236.

D. Armentano, C. Beltrán, P. Bürgisser, F. Cucker and M. Shub (2015*a*), 'Condition length and complexity for the solution of polynomial systems', accepted by *J. Found. Comput. Math.*

D. Armentano, C. Beltrán, P. Bürgisser, F. Cucker and M. Shub (2015*b*), A stable, polynomial-time algorithm for the eigenpair problem. Preprint. arXiv:1505.03290

C. Beltrán and L. Pardo (2009), 'Smale's 17th problem: average polynomial time to compute affine and projective solutions', *J. Amer. Math. Soc.* **22**, 363–385.

C. Beltrán and L. Pardo (2011), 'Fast linear homotopy to find approximate zeros of polynomial systems', *Found. Comput. Math.* **11**, 95–129.

P. Bürgisser and F. Cucker (2011), 'On a problem posed by Steve Smale', *Ann. of Math.* **174**, 1785–1836.

P. Bürgisser and F. Cucker (2013), *Condition*, Vol. 349 of *Grundlehren der Mathematischen Wissenschaften*, Springer.

[1] The URLs cited in this work were correct at the time of going to press, but the publisher and the authors make no undertaking that the citations remain live or are accurate or appropriate.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

P. Bürgisser, F. Cucker and M. Lotz (2006), 'Smoothed analysis of complex conic condition numbers', *J. Math. Pures Appl.* **86**, 293–309.
P. Bürgisser, F. Cucker and M. Lotz (2008), 'The probability that a slightly perturbed numerical analysis problem is difficult', *Math. Comp.* **77**, 1559–1583.
P. Bürgisser, F. Cucker and M. Lotz (2010), 'Coverage processes on spheres and condition numbers for linear programming', *Ann. Probab.* **38**, 570–604.
D. Cheung and F. Cucker (2004), 'Solving linear programs with finite precision I: Condition numbers and random programs', *Math. Program.* **99**, 175–196.
D. Cheung and F. Cucker (2005), 'A note on level-2 condition numbers', *J. Complexity* **21**, 314–319.
D. Cheung and F. Cucker (2006), 'Solving linear programs with finite precision II: Algorithms', *J. Complexity* **22**, 305–335.
D. Cheung and F. Cucker (2013), 'On the average condition of random linear programs', *SIAM J. Optim.* **23**, 799–810.
D. Cheung and F. Cucker (2015), 'Smoothed analysis of componentwise condition numbers for sparse matrices', *IMA J. Numer. Anal.* **35**, 74–88.
F. Cucker and J. Peña (2002), 'A primal–dual algorithm for solving polyhedral conic systems with a finite-precision machine', *SIAM J. Optim.* **12**, 522–554.
J. Demmel (1987), 'On condition numbers and the distance to the nearest ill-posed problem', *Numer. Math.* **51**, 251–289.
J. Demmel (1988), 'The probability that a numerical analysis problem is difficult', *Math. Comp.* **50**, 449–480.
J. Demmel (1997), *Appl. Numer. Linear Algebra*, SIAM.
C. Eckart and G. Young (1936), 'The approximation of one matrix by another of lower rank', *Psychometrika* **1**, 211–218.
A. Edelman (1988), 'Eigenvalues and condition numbers of random matrices', *SIAM J. Matrix Anal. Appl.* **9**, 543–556.
R. Fletcher (1985), 'Expected conditioning', *IMA J. Numer. Anal.* **5**, 247–273.
H. Goldstine and J. von Neumann (1951), 'Numerical inverting matrices of high order II', *Proc. Amer. Math. Soc.* **2**, 188–202.
M. Hestenes and E. Stiefel (1952), 'Methods of conjugate gradients for solving linear systems', *J. Res. Nat. Bur. Standards* **49**, 409–436.
N. Higham (1989), 'The accuracy of solutions to triangular systems', *SIAM J. Numer. Anal.* **26**, 1252–1265.
N. Higham (1996), *Accuracy and Stability of Numerical Algorithms*, SIAM.
Julius Caesar (ed. 1917), *De Bello Gallico* (with an English translation by H. J. Edwards), Harvard University Press.
T.-Y. Li and T. Sauer (1987), 'Homotopy method for generalized eigenvalue problems $Ax = \lambda Bx$', *Linear Algebra Appl.* **91**, 65–74.
M. Lotz (2015), 'On the volume of tubular neighborhoods of real algebraic varieties', *Proc. Amer. Math. Soc.* **143**, 1875–1889.
J. von Neumann and H. Goldstine (1947), 'Numerical inverting matrices of high order', *Bull. Amer. Math. Soc.* **53**, 1021–1099.
J. Renegar (1987), 'On the worst case arithmetic complexity of approximating zeros of polynomials', *J. Complexity* **3**, 90–113.
J. Renegar (1994*a*), 'Is it possible to know a problem instance is ill-posed?', *J. Complexity* **10**, 1–56.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press

J. Renegar (1994*b*), 'Some perturbation theory for linear programming', *Math. Program.* **65**, 73–91.
J. Renegar (1995*a*), 'Incorporating condition measures into the complexity theory of linear programming', *SIAM J. Optim.* **5**, 506–524.
J. Renegar (1995*b*), 'Linear programming, complexity theory and elementary functional analysis', *Math. Program.* **70**, 279–351.
J. Renegar (2000), *A Mathematical View of Interior-Point Methods in Convex Optimization*, SIAM.
M. Shub (1993), Some remarks on Bézout's theorem and complexity theory. In *From Topology to Computation: Proceedings of the Smalefest* (M. Hirsch, J. Marsden and M. Shub, eds), Springer, pp. 443–455.
S. Smale (1997), Complexity theory and numerical analysis. In *Acta Numerica*, Vol. 6 (A. Iserles, ed.), Cambridge University Press, pp. 523–551.
S. Smale (2000), Mathematical problems for the next century. In *Mathematics: Frontiers and Perspectives* (V. Arnold, M. Atiyah, P. Lax and B. Mazur, eds), AMS, pp. 271–294.
D. Spielman and S.-H. Teng (2002), Smoothed analysis of algorithms. In *Proceedings of the International Congress of Mathematicians*, Vol. I, pp. 597–606.
G. Stewart (1990), 'Stochastic perturbation theory', *SIAM Rev.* **32**, 579–610.
A. Turing (1948), 'Rounding-off errors in matrix processes', *Quart. J. Mech. Appl. Math.* **1**, 287–308.
G. Van Loan (1987), 'On estimating the condition of eigenvalues and eigenvectors', *Linear Algebra Appl.* **88–89**, 715–732.
D. Viswanath and L. Trefethen (1998), 'Condition numbers of random triangular matrices', *SIAM J. Matrix Anal. Appl.* **19**, 564–581.
P.-Å. Wedin (1973), 'Perturbation theory for pseudo-inverses', *Nordisk Tidskr. Informationsbehandling (BIT)* **13**, 217–232.
N. Weiss, G. Wasilkowski, H. Woźniakowski and M. Shub (1986), 'Average condition number for solving linear equations', *Linear Algebra Appl.* **83**, 79–102.
J. Wilkinson (1963), *Rounding Errors in Algebraic Processes*, Prentice Hall.
J. Wilkinson (1971), 'Some comments from a numerical analyst', *J. Assoc. Comput. Mach.* **18**, 137–147.
M. Wschebor (2004), 'Smoothed analysis of $\kappa(a)$', *J. Complexity* **20**, 97–107.

https://doi.org/10.1017/S0962492916000027 Published online by Cambridge University Press