Lectures in Applied Mathematics
Volume 00, 1996

Overview of Information-Based Complexity

Columbia University Computer Science Department Report CUCS-024-96

Henryk Woźniakowski

Abstract. We present a brief introduction to information-based complexity. An example of zero finding is chosen to illustrate the basic notions such as nonadaptive and adaptive information, as well as the ε-complexity in the worst case, average case and randomized settings. In the second part, we survey recent results on complexity of multivariate problems. We concentrate on tractability and strong tractability issues as well as on path integration.

1. Introduction

Computational complexity studies the intrinsic difficulty of solving mathematically posed problems. Discrete computational complexity studies discrete problems and often uses the Turing machine model of computation. Continuous computational complexity studies continuous problems and tends to use the real number model.

The real number model is an abstraction of floating point arithmetic, which is universally used in scientific computation. In the real number model, as well as in floating point arithmetic, the cost of operations does *not* depend on the size of the numbers. In the real number model, we assume that operations are performed exactly and at finite cost. In floating point arithmetic we have rounding errors, and numerical stability becomes an important issue. For many problems, we can find algorithms which enjoy optimality properties in the real number model and which are numerically stable in floating point arithmetic. However, there are some counterexamples as well, see [**6**]. For the precise definition of the real number model the reader is referred to [**2, 12**].

Continuous computational complexity may be split into two branches. The first branch deals with problems for which the information is *complete*. Informally,

1991 *Mathematics Subject Classification.* 68Q25.

Key words and phrases. complexity, zero finding, multivariate problems, tractability, strong tractability, path integration.

This research was supported in part by the National Science Foundation and the Air Force Office of Scientific Research.

information may be complete for problems which are specified by a finite number of inputs. Examples include matrix multiplication, and the solution of linear algebraic systems or systems of polynomial equations. Problems with complete information may be NP-complete over the reals. The first such problem was established in [**2**] and this is the problem of deciding whether a system of n real polynomials of degree 4 has a real root.

The other branch of continuous computational complexity is *information-based complexity*, denoted for brevity as IBC. Typically, IBC studies infinite-dimensional problems for which the input is an element of an infinite-dimensional space. Examples of such problems include multivariate integration or approximation, solution of ordinary or partial differential equations, integral equations, optimization, and solving non-polynomial equations. The input of such problems is usually a multivariate function on the reals. Since digital computers can handle only finite sets of numbers, infinite-dimensional objects such as functions must be replaced by finite sets of numbers. These finite sets of numbers may be given as function values at finitely many points. Hence, complete information is not available about such objects. Only *partial* information is available when solving an infinite-dimensional problem on a digital computer. Typically, this partial information is *contaminated* with errors such as round-off errors, measurement errors, or human errors. Thus, the available information is partial and/or contaminated. Therefore, the original problem can be solved only *approximately.* The goal of IBC is to compute such an approximation at minimal cost. The error and the cost of approximation can be defined in different settings including worst case, average case, probabilistic, randomized and mixed settings. The cost is the sum of the information and combinatory costs. The information cost is the cost of computing information, and the combinatory cost is the cost of combining the information to compute an approximation. The ε-complexity is then defined as the minimal cost of computing an approximation with error at most ε. For many problems, sharp complexity bounds are obtained by studying only the information cost.

IBC is a branch of computational complexity and is formulated as an abstract theory with applications in many areas. Since continuous computational problems are of interest in many fields, IBC has common interests and has greatly benefited from these fields. Questions, concepts, and results from approximation theory, numerical analysis, applied mathematics, statistics, complexity theory, algorithmic analysis, number theory, analysis and measure theory have all been influential. The reader who wants to find more about IBC is referred to the books and recent surveys [**3, 9, 11, 18, 24, 26**].

In Section 2, we illustrate the basic concepts of IBC by an example of zero finding for scalar functions. These concepts include nonadaptive and adaptive information, as well as the concepts of error and cost in various settings. The complexity is defined as the minimal cost of computing an approximation with error at most ε. In particular, we recall sharp complexity bounds in the worst case and average case settings for zero finding.

In the second part of this paper we discuss multivariate problems. Many problems in science, engineering, economics and finance are modeled by multivariate problems involving functions of d variables with large or even huge d. For path integration, which is important in the foundation of physics, we even have $d = +\infty$; the approximation of path integrals yields multivariate integration with huge d.

We are interested in the complexity of multivariate problems in various settings. In particular, the complexity depends on the error parameter ε, and on the number d of variables.

In the worst case setting, it is known that many problems are intractable. More specifically, for many problems the complexity is an exponential function of d. This means that for large d the complexity is so huge that it is impossible to solve the problem. This is sometimes called the *curse of dimension*.

We stress that the exponential dependence on d is a complexity result and it is impossible to get around it by designing efficient algorithms. The only way to break the curse of dimension is to weaken the notion of error and/or cost. This can sometimes be done by switching from the worst case setting to another setting. Hence, we wish to examine how complexity depends on ε and d in other settings. If the dependence is polynomial in d and ε^{-1} then the curse of dimension is broken.

In fact, we propose a more general approach. Assume for simplicity that the cost of one function evaluation (or one linear functional evaluation) is taken as unity. For a given setting, we wish to check which multivariate problems are *tractable* or even *strongly tractable*. That is, we wish to determine for which multivariate problems the complexity does depend polynomially on d and ε^{-1} (tractability), or the complexity is independent of d and depends polynomially on ε^{-1} (strong tractability), see [**35**].

Tractability results have been obtained for two classes of information. The first class consists of function values and the second one consists of all linear functionals. It turns out that, under mild assumptions, tractability and strong tractability in these two classes are equivalent, although the proof is, in general, not constructive. For multivariate tensor products, we know a constructive proof. Based on Smolyak's algorithm, see [**23**], we derived explicit cost bounds for tensor products, see [**29**]. We specialize these bounds for multivariate integration and approximation.

The final section deals with path integration, see [**30, 31**]. Usually Monte Carlo algorithms are used to approximate path integrals. We study deterministic algorithms in the worst case setting. Then path integration is tractable (i.e., its complexity is polynomial in ε^{-1}) if the class of integrands consists of entire functions. Finite smoothness of integrands is not enough if the measure of the path integration problem is supported on an infinite dimensional subspace. In this case, the classical Monte Carlo algorithm is almost optimal in the randomized setting. We conclude with a remark on Feynman-Kac path integrals.

2. Zero Finding

We illustrate the basic concepts of IBC by using an example of zero finding for scalar functions. More can be found in [**14, 15, 21**].

Let F be a class of scalar functions $f : [a, b] \to \mathbb{R}$ that have a zero. Consider the nonlinear equation

$$f(x) = 0, \qquad f \in F.$$

We want to compute a number z which differs from a zero x of f no more than a given ε,

$$|z - x| \le \varepsilon.$$

We will call such a z an *ε-approximation*.

What is the complexity (minimal cost) of computing an ε-approximation? To answer this question we must specify how an ε-approximation is computed and exactly what we mean by the complexity.

First, observe that the class F plays an important role in our problem. Assume first that $F = F_n$ is the class of univariate polynomials of degree at most n,

$$F_n = \left\{ f : f(x) = \sum_{i=1}^{n} a_i x^i, \ \forall\, x \in \mathbb{R} \quad \text{where } (a_0, a_1, \ldots, a_n) \in \mathrm{DATA}_n \right\}.$$

Here DATA_n denotes the set of permissible coefficients. An example of such a set is provided by assuming that $|a_i| \le 1$ for $i = 0, 1, \ldots, n-1$ and $a_n = 1$.

For the class F_n, we can input $n, a_0, a_1, \ldots, a_n$. Hence, the information about f is provided by its degree and coefficients. This information is *complete* since the input uniquely defines the polynomial f. An ε-approximation can be then computed by performing arithmetic and branching (comparison) operations on the input data. Then the (worst case) complexity is the minimal number of such operations needed to compute an ε-approximation for all f from F_n. The bounds on the complexity may be found in [**16**].

Assume now that the class F is given as

$$F = \{ f : [a, b] \to \mathbb{R} : \ f(a) f(b) < 0, \ f \text{ is continuous} \}. \tag{2.1}$$

In this case, the function f cannot be identified by a finite number of coefficients. The information about f is now obtained by using an oracle (subroutine) which provides the function value $f(t)$. Here t is an input to the subroutine and may be any number from the domain $[a, b]$.

Usually, the output y of the subroutine is perturbed by some error, $y = f(t) + \eta$. The error η can be random or deterministic. It may be random if $f(t)$ is observed and it is deterministic if $f(t)$ is computed (in floating point arithmetic we will definitely have some rounding errors). This means that our information is *noisy*. A thorough study of noisy information may be found in the forthcoming book [**18**]. To simplify our exposition, we assume that $\eta = 0$, that is, the information is exact and we can compute (or observe) $y = f(t)$.

The oracle may be used a finite number of times to obtain function values at different points. Let

$$N(f) = [f(t_1), f(t_2), \ldots, f(t_n)] \tag{2.2}$$

denote the computed information.

Knowing that $f \in F$ and $N(f) = y$, we *cannot* identify the function f. In fact, there are infinitely many functions g from F which share the same information as f, that is, for which $N(g) = y$. This is why we say that information is *partial*. It is clear that partial information does not, in general, supply enough knowledge to find the exact solution. However, if the sample points t_i are well chosen and the number n of sample points is sufficiently large, we may hope to compute an ε-approximation.

We distinguish two types of information. If the sample points t_i, as well as their number n in (2.2) are given a priori and independently of the function f, then the information N is called *nonadaptive*. For example, the points

$$t_i = a + (i-1)(b-a)/(n-1)$$

yield nonadaptive information with equispaced sample points over the interval $[a, b]$. For nonadaptive information, each function value $f(t_i)$ can be computed independently. This is a very desirable property for parallel computation. If we have n processors, then the nonadaptive information N can be computed in the time needed for one function evaluation.

If, however, the sample points t_i in (2.2) depend on the previously computed values $f(t_1), f(t_2), \ldots, f(t_{i-1})$, for $i = 2, 3, \ldots, n$, or if the number $n = n(f)$ of sample points depends on the $f(t_i)$'s, then the information N is called *adaptive*.

Adaptive information is widely used in computation. Examples of adaptive information are provided by information used by bisection or by any iteration method. Adaptive information is inherently sequential since we have to wait until the computation of $f(t_1), f(t_2), \ldots, f(t_{i-1})$ has been completed to compute t_i. Hence, adaptive information is ill-suited for parallel computation.

The power of adaptive information is one of the most interesting issues of IBC. For some problems adaptive information is exponentially more powerful than nonadaptive information. This is the case for zero finding as we shall see later. On the other hand, there are problems for which adaption does not help or helps only a little. The interested reader is referred to a survey of the current state of our knowledge in [**13**].

For given (nonadaptive or adaptive) information N, we approximate a zero of f by an algorithm $\phi : \mathbb{R}^n \to [0, 1]$. That is,

$$z \;=\; \phi\,(N(f))$$

is to be an ε-approximation to a zero of f.

We now comment on the algorithm ϕ. Formally, ϕ is given as a finite composition of operations which are allowed in the model of computation, i.e., arithmetic and branching operations. In IBC, the proofs of lower bounds on the complexity only use the fact that information is partial, and therefore they are true even if an arbitrary ϕ is allowed. This makes the lower bounds stronger since they also hold if the model of computation is extended by allowing more operations.

We are ready to define the *local* error of $z = \phi(N(f))$ as

$$e(f, \phi, N) \;=\; \inf\{\, |z - x| : \; x \in [0, 1] \text{ such that } f(x) = 0 \;\}.$$

The *global* error depends on the setting. In the *worst case* setting, we have

$$e(\phi, N) \;=\; \sup_{f \in F} \, e(f, \phi, N).$$

In the *average case* setting, we assume that the class F is equipped with a probability measure μ, and

$$e(\phi, N) \;=\; \int_F e(f, \phi, N)\, \mu(df).$$

In the *randomized* setting, we assume that $\phi = \phi_t$ and $N = N_t$ depend on a random element t from a set T whose distribution is ρ, and

$$e(\phi, N) \;=\; \sup_{f \in F} \int_T e(f, \phi_t, N_t)\, \rho(dt).$$

One may define the global error in other settings but we stop here.

We now discuss the *cost* of computation. To compute $z = \phi(N(f))$ we need $n = n(f)$ function values. Let $k = k(f)$ be the total number of arithmetic and branching operations needed to compute the sample points t_i and their number $n(f)$ as well as $\phi(y)$ given $y = N(f)$. Assume that the cost of one function value is $\mathbf{c}$, whereas the cost of one arithmetic or branching operation is taken as unity. We stress that usually $\mathbf{c} \gg 1$.

The *local* cost of $z = \phi(N(f))$ is defined as

$$\mathrm{cost}(f, \phi, N) \;=\; \mathbf{c}\, n(f) \;+\; k(f).$$

The *global* cost, just as the error, depends on the setting. In the *worst case* setting, we have

$$\mathrm{cost}(\phi, N) \;=\; \sup_{f \in F} \mathrm{cost}(f, \phi, N).$$

In the *average case* setting, we have

$$\mathrm{cost}(\phi, N) \;=\; \int_F \mathrm{cost}(f, \phi, N)\, \mu(df).$$

In the *randomized* setting, we have

$$\mathrm{cost}(\phi, N) \;=\; \sup_{f \in F} \int_T \mathrm{cost}(f, \phi_t, N_t)\, \rho(dt).$$

We are ready to formalize the concept of complexity. The ε-complexity is defined as

$$\mathrm{comp}(\varepsilon) \;=\; \inf \left\{ \mathrm{cost}(\phi, N) : \; (\phi, N) \text{ such that } e(\phi, N) \le \varepsilon \right\}.$$

This includes the worst case, average case or randomized complexity depending on which definitions of error and cost are used.

We briefly report what is known about the complexity of zero finding in the worst case and average case settings for the class F defined by (2.1). We begin with the worst case setting.

Assume first that nonadaptive information with n function values is used. It is easy to show, see [**25**], that the minimal error of algorithms using such nonadaptive information is $0.5(b-a)/(n+1)$. Hence, we can compute an ε-approximation using roughly $0.5(b-a)/\varepsilon$ nonadaptive function values.

If adaptive information with n function values is used then the minimal error is exponentially smaller and is equal to $(b-a)/2^{n+1}$, see [**5**]. Hence, an ε-approximation can be computed using roughly $\log_2 (b-a)/\varepsilon$ function values. This can be done by using *bisection* for which the number of arithmetic and branching operations is also roughly $\log_2 (b-a)/\varepsilon$. This proves that the worst case complexity is given by

$$\mathrm{comp}(\varepsilon) \;\approx\; \mathbf{c} \log_2 (b-a)/\varepsilon.$$

The same complexity, as well as optimality of bisection, holds if we permit more general information and/or if we shrink the class F. That is, as shown in [**21, 22**], we may allow the computation of arbitrary linear functionals on f, and we may take as F the class of polynomials (with unbounded degrees) for which $f(a)f(b) < 0$.

We now turn to the average case setting, see [**15, 19**]. We need to equip the class F of (2.1) with a probability measure μ. We choose μ to be a folded Wiener measure which is concentrated on functions with continuous rth derivatives and with fixed boundary conditions at the end points a and b. Here $r \geq 2$.

As before, first consider nonadaptive information. In fact, we may even consider adaptive information with fixed $n(f) = n$. That is, the choice of the sample points t_i can now be adaptive; however, the total number of t_i is always the same. It is proven in [**19**] that in this case, the minimal average case error is of the same order as the minimal worst case error, i.e., it is of order $(b-a)/2^{n+1}$. So we may compute an ε-approximation with the average case cost of order $\log_2(b-a)/\varepsilon$ which also is the worst case complexity.

It turns out that adaptive choice of $n(f)$ is crucial. That is, as shown in [**15**], there exists a hybrid secant-bisection method that computes an ε-approximation with cost proportional to $\log_2 \log_2(b-a)/\varepsilon$, which is exponentially smaller than the previous bound. Furthermore, this method is optimal since the average case complexity is of the same order. We remark that the order of the average case complexity does not depend on the smoothness r of functions as long as $r \geq 2$. The case $r < 2$ is open.

3. Multivariate Problems

We now discuss complexity of multivariate problems. By a multivariate problem we mean an approximation of an operator defined on functions f of d variables. More precisely, let F_d be a class of functions $f : [0,1]^d \to \mathbb{R}$, and let

$$S_d : F_d \to G_d,$$

where G_d is a normed linear space.

We wish to approximate $S_d(f)$ for $f \in F_d$. Two primary examples are *multivariate integration*,

$$S_d(f) = \int_{[0,1]^d} f(t)\, dt \qquad \text{with } G_d = \mathbb{R},$$

and *multivariate approximation*,

$$S_d(f) = f \qquad \text{with } G_d = L_2([0,1]^d),$$

with F_d being a class of functions that are continuously r times differentiable.

If S_d is a linear operator (or can be extended to a linear operator) and F_d is a unit ball in a normed linear space, then the multivariate problem is called *linear*. Note that our two examples are linear problems.

We consider two classes of (partial) information. The first class Λ^{std} is given by function evaluations, that is, $L \in \Lambda^{\text{std}}$ iff for some $x \in [0,1]^d$, we have $L(f) = f(x)$, $\forall\, f \in F_d$. The second class Λ^{all} is more general and consists of all linear functionals $L : F_d \to \mathbb{R}$. As in Section 2, the cost of one function evaluation (or one evaluation of $L(f)$) is denoted by $\mathbf{c}$. To stress the dependence on the number d of variables we write $\mathbf{c} = \mathbf{c}(d)$.

As always, we wish to compute an ε-approximation with minimal cost. We are particularly interested in large d and/or in large ε^{-1}. The ε-complexity of the multivariate problem is defined as in the previous section with obvious changes. For instance, an approximation $z = \phi(N(f)) \in G_d$, and the local error is now given by

$$e(f, \phi, N) = \|S_d(f) - \phi(N(f)\|.$$

The local cost is defined similarly, however, we now permit operations in the space G_d. That is, we can multiply $\alpha\, g$ and add g_1+g_2 for $\alpha \in \mathbb{R}$ and $g, g_1, g_2 \in G_d$. The cost of such operations is taken as unity.

The complexity depends, of course, on all parameters of the multivariate problem. To stress the dependence on the error parameter ε and on the number of variables d we denote the complexity by $\mathrm{comp}(\varepsilon, d)$.

Many multivariate problems are *intractable* and their complexity grows exponentially with the number d of variables. This is sometimes called the *curse of dimension*. Typically,

$$\mathrm{comp}(\varepsilon, d) \,=\, \Theta(\mathbf{c}(d)\varepsilon^{-d/r}), \quad \text{as } \varepsilon \to 0,$$

where r stands for the smoothness of the functions in the class F_d.

Problems which suffer the curse of dimension in the worst case setting include integration, approximation, global optimization, integral and partial differential equations for classes of functions whose rth derivatives are uniformly bounded in L_∞, see [**1, 4, 8, 9, 17, 24, 32**].

In the average case and randomized settings, the curse of dimension is present for approximation over the class of functions with r continuous derivatives which is equipped with the folded isotropic Wiener measure, see [**20, 28**] for the average case and [**7, 10, 27**] for the randomized setting.

For some problems we can break the curse of dimension by switching to a different setting. For example, in the randomized setting, it is well known that the classical Monte Carlo algorithm breaks the curse of dimension for multivariate integration. In the average case setting, the curse of dimension is broken for multivariate integration no matter what probability measure is given on the class of functions. However, in general, the proof is not constructive. For the Wiener sheet measure, the proof is constructive and we know almost optimal algorithms, see [**29, 33**]. For multivariate approximation, the curse of dimension is broken only for some probability measures. For instance, it is broken for the Wiener sheet measure, see [**29, 34**], however, as already mentioned, it is not broken for the isotropic Wiener measure, see [**20, 28**].

It seems natural to characterize which multivariate problems are *tractable* or *strongly tractable* in various settings. More precisely, we say that the multivariate problem is *tractable* if there exist nonnegative numbers K, p and q such that

$$\mathrm{comp}(\varepsilon, d) \,\le\, K\, \mathbf{c}(d)\, d^q\, \varepsilon^{-p}, \quad \forall\, d, \forall\, \varepsilon \le 1. \tag{3.1}$$

If $q = 0$ then we say that the multivariate problem is *strongly tractable*. For strongly tractable problems, the only dependence of the complexity on d is through the cost $\mathbf{c}(d)$.

Tractability and strong tractability of linear multivariate problems have been studied in [**35**] for the information classes Λ^{std} and Λ^{all}. In the worst case and randomized settings we assume that the domain F_d and the range of S_d are Hilbert spaces. In the average case and probabilistic settings we assume that F_d is a Banach space equipped with a Gaussian measure μ_d and that the range of S_d is a Hilbert space.

For the class Λ^{all}, necessary and sufficient conditions for tractability and strong tractability can be obtained by using known IBC results on complexity of linear problems. They are expressed in terms of singular values of S_d or in terms of eigenvalues of the covariance operator of the measure $\mu_d S_d^{-1}$. Roughly speaking, tractability and strong tractability hold if the singular values tend to zero sufficiently fast.

Tractability and strong tractability in the randomized setting and the worst case setting are equivalent, and the corresponding complexities differ only by constants. This follows easily from [**10**]. Similarly, tractability and strong tractability in the probabilistic setting and the average case setting are equivalent due to relations between these two settings for linear problems, see [**24**].

We stress that for the class $\Lambda^{\rm all}$ the construction of an ε-approximation with minimal cost is easy since we know the optimal choice of linear functionals, and that linear algorithms are optimal.

We now turn to the class $\Lambda^{\rm std}$. Under mild assumptions, we prove in [**35**] that tractability and strong tractability in the classes $\Lambda^{\rm std}$ and $\Lambda^{\rm all}$ are equivalent. In particular, we prove that the exponents in ε^{-1} may differ by at most two. The proof of this equivalence is, however, *not* constructive.

One may suspect that only trivial problems are strongly tractable. However, even in the worst case setting, this is not true. More precisely, if F_d is a unit ball of a reproducing kernel Hilbert space and the linear problem is suitably normalized, then there exists a constant K such that

$$\mathrm{comp}(\varepsilon, d) \,\le\, K\, \mathbf{c}(d)\, \varepsilon^{-p},$$

where $p = 2$ for the class $\Lambda^{\rm all}$, and $p = 4$ for the class $\Lambda^{\rm std}$, see [**35**]. It is also known that $p = 2$ for the class $\Lambda^{\rm all}$ is sharp, whereas it is open whether $p = 4$ for the class $\Lambda^{\rm std}$ can be improved.

As before, the proof for the class $\Lambda^{\rm std}$ is not constructive. A construction is known for linear multivariate problems that are defined by tensor products. This is discussed in the next section.

4. Multivariate Tensor Products

Assume that the linear multivariate problem $S_d : F_d \to G_d$ is given by tensor products of a one dimensional linear problem $S_1 : F_1 \to G_1$, see [**29, 36**]. For such multivariate problems tractability and strong tractability are equivalent under mild assumptions, and this holds for both classes $\Lambda^{\rm all}$ and $\Lambda^{\rm std}$.

In particular, in the worst case setting strong tractability holds iff the norm of S_1 is less than one and the nth singular value of S_1 goes to zero at least as fast as some positive power of $1/n$.

For tractable tensor product problems and for the class $\Lambda^{\rm std}$, we construct polynomial-time algorithms, see [**29**]. This construction is based on Smolyak's algorithm, see [**23**]. More precisely, in the worst case and average case settings, we present linear algorithms that compute an ε-approximation for the multivariate tensor product problem with cost

$$\mathrm{cost}(\varepsilon, d) \,\le\, (\mathbf{c}(d) + 2)\; \beta_1 \left(\beta_2 \,+\, \beta_3\, \frac{\ln 1/\varepsilon}{d-1}\right)^{\beta_4 (d-1)} \left(\frac{1}{\varepsilon}\right)^{\beta_5}.$$

The coefficients β_i's do not depend on d; they are determined by the properties of the problem for $d = 1$.

Note the intriguing dependence of the cost bound on d. The leading term $\varepsilon^{-\beta_5}$ does not depend on d, whereas $\ln 1/\varepsilon$ is divided by a multiple of $d-1$ and then raised to a multiple of $d-1$. If the tensor product problem is tractable then the cost bound does not exceed $\mathbf{c}(d)\, K\, \varepsilon^{-p}$ for some numbers K and p, both independent of d.

We illustrate the results for multivariate approximation and integration. For the class $\Lambda^{\rm all}$ consider the approximation problem, $S_1 f = f$, with the Sobolev class $F_1 = W^r[a,b]$ of periodic functions $f : [a,b] \to \mathbb{R}$ such that $f(a) = 0$ and with absolutely continuous $(r-1)$st derivatives and r-th derivatives bonded by one in $L_2([a,b])$. Then, see [**36**], strong tractability holds in the worst case setting iff $b - a < 2\,\pi$, and then the exponent of strong tractability is $p = p^*/r$, where p^* is determined by the equation $\sum_{n=1}^{\infty} \left((b-a)/(2\,\pi\, n)\right)^{p^*} = 1/2$, and

$$\mathrm{comp}(\varepsilon, d) \,\le\, (\mathbf{c}(d) + 2)\, \frac{b-a}{\pi}\, \varepsilon^{-p^*/r}.$$

In particular, for $b - a = 1$ we have $p^* \simeq 1.23$, and for $b - a = 2\sqrt{3}$ we have $p^* = 2$.

We now turn to the class $\Lambda^{\rm std}$ of function values and consider the approximation problem $S_d f = f$ in the average case setting for the class of continuous functions $f : [0,1]^d \to \mathbb{R}$ equipped with the Wiener sheet measure. Then we know a linear algorithm, see [**29**], that computes an ε-approximation with cost

$$\mathrm{cost}(\varepsilon, d) \,\le\, \mathbf{c}(d)\, 0.8489 \left(2.9974 \,+\, 4.3869\, \frac{-0.9189 \,+\, \ln 1/\varepsilon}{d-1}\right)^{2(d-1)} \left(\frac{1}{\varepsilon}\right)^2.$$

This algorithm has optimal powers of ε^{-1} and $\ln 1/\varepsilon$ since

$$\mathrm{comp}(\varepsilon, d) \,=\, \Theta\left(\varepsilon^{-2} \left(\ln 1/\varepsilon\right)^{2(d-1)}\right),$$

see [**35**]. This approximation problem is strongly tractable since

$$\mathrm{cost}(\varepsilon, d) \,\le\, \mathbf{c}(d)\, 2.37632\, \varepsilon^{-5.672}.$$

The exponent 5.672 seems to be too high; however, no smaller exponent has been found so far.

We would like again to add that the choice of the Wiener sheet measure is essential. It is known, see [**28**], that if we replace the Wiener sheet measure by the isotropic Wiener measure then the approximation problem is intractable since $\mathrm{comp}(\varepsilon, d) = \Theta\left(\mathbf{c}(d)\, \varepsilon^{-2d}\right)$.

Consider now the integration problem $S_d f = \int_{[0,1]^d} f(x) dx$ in the average case setting for the class of continuous functions $f : [0,1]^d \to \mathbb{R}$ equipped with the Wiener sheet measure. Then we know a linear algorithm, see [**29**], which computes an ε-approximation with $\mathrm{cost}(\varepsilon, d)$ bounded by

$$\mathrm{cost}(\varepsilon, d) \,\le\, \mathbf{c}(d)\, 3.304 \left(1.77959 \,+\, 2.714\, \frac{-1.12167 \,+\, \ln 1/\varepsilon}{d-1}\right)^{1.5(d-1)} \frac{1}{\varepsilon}.$$

The power of ε^{-1} is optimal and the power of $\ln 1/\varepsilon$ is too large since

$$\mathrm{comp}(\varepsilon, d) \,=\, \Theta\left(\varepsilon^{-1} \left(\ln 1/\varepsilon\right)^{(d-1)/2}\right),$$

see [**33**]. This integration problem is strongly tractable since

$$\mathrm{cost}(\varepsilon, d) \,\le\, \mathbf{c}(d)\, 7.26\, \varepsilon^{-2.454}.$$

The exponent 2.454 is too high. It is well known that there exist algorithms with an exponent at most two. The proof of this latter fact is, however, *not* constructive.

This integration problem is related to discrepancy in the L_2-norm, see [**33**]. Using this relation we obtain an upper bound, which is independent of d, for the number $n(\varepsilon, d)$ of points for which discrepancy (with unequal weights) is at most ε,

$$n(\varepsilon, d) \le 7.26\, \varepsilon^{-2.454}, \quad \forall\, d, \forall\, \varepsilon \le 1.$$

5. Path Integration

Path integrals occur in many applied fields including quantum physics and chemistry, differential equations, and financial mathematics, as well as average case complexity. The path integration problem is defined as the approximation of

$$S(f) = \int_X f(x)\, \mu(dx), \quad \forall\, f \in F.$$

Here, X is a separable infinite dimensional Banach space and μ is a zero mean Gaussian measure on X. The class F is a class of (Borel) measurable real functions defined on X.

A typical approach is to approximate the path integral by high dimensional integrals and apply a Monte Carlo (randomized) algorithm. Do we really need to use randomized algorithms for path integrals? Perhaps we can find an effective *deterministic* algorithm that approximates path integrals with small error. To answer this question, we study the *worst case complexity* of path integration in the class Λ^{std}. Path integration is considered with respect to different Gaussian measures μ and different classes F of integrands.

Tractability of path integration means that the complexity depends polynomially on ε^{-1}. For the class F of integrands that are r times Frechet differentiable, tractability of path integration holds iff the covariance operator of the Gaussian measure μ has finite rank. Hence, if the Gaussian measure μ is supported on an infinite dimensional space then path integration is intractable. In this case, there exists no effective deterministic algorithm, and the use of randomized algorithms is reasonable. In fact, for this class of integrands, the classical Monte Carlo algorithm is optimal and the complexity in the randomized setting is proportional to ε^{-2}, see [**30**].

On the other hand, for a particular class F of entire integrands, the worst case complexity of path integration is at most of order ε^{-p} with p depending on the Gaussian measure μ. Hence, path integration is now tractable. Furthermore, for any Gaussian measure μ, the exponent p is less than or equal to 2. For the Wiener measure we have $p = 2/3$. For this class of entire integrands, we provide effective deterministic algorithms that solve the path integration problem with (worst case) cost that is usually much less than the (randomized) cost of the classical Monte Carlo algorithm, see [**30**].

In [**31**] we consider a class of functions related to the Feynman-Kac formula. More precisely, this is the class of potential and initial conditions functions that define the heat equation. Although these functions do not need to be very smooth, we prove tractability of path integration, and in many cases, the worst case complexity is substantially smaller than ε^{-2}.

Acknowledgments

I thank J. F. Traub, G. W. Wasilkowski and A. G. Werschulz for valuable comments.

References

1. Bakhvalov, N. S., *On approximate calculation of integrals (in Russian)*, Vestnik MGU, Ser. Mat. Mekh. Astron. Fiz. Khim. **4** (1959), 3–18.
2. Blum, L., Shub, M. and Smale S., *On a Theory of Computation and Complexity over the Real Numbers: NP-Completeness, Recursive Functions and Universal Machines*, Bull. Amer. Math. Soc. **21** (1989), 1–46.
3. Heinrich, S., *Random Approximation in Numerical Analysis*, in Proc. of the Functional Analysis Conference, Essen 1991, Lecture Notes in Pure and Applied Mathematics (K. D. Bierstedt et al., eds.), vol. 150, Marcel Dekker, New York, 1993, pp. 123–171.
4. ———, *Complexity of integral equations and relations to s-numbers*, J. Complexity **9** (1993), 141–153.
5. Kung, H. T., *The complexity of obtaining starting points for solving operator equations by Newton's method*, Analytical Computational Complexity (J. F. Traub, eds.), Academic Press, New York, 1976, pp. 35-75.
6. Miller, W., *Computational complexity and numerical stability*, SIAM J. Comput. **4** (1975), 97–107.
7. Mathé, P., *Random approximation of Sobolev embedding*, J. Complexity **7** (1993), 261–281.
8. Nemirovsky, A. S. and Yudin, D. B., *Problem Complexity and Method Efficiency in Optimization*, Wiley-Interscience, New York, 1983.
9. Novak, E., *Deterministic and Stochastic Error Bounds in Numerical Analysis*, vol. 1349, Lectures Notes in Math., Springer Verlag, Berlin, 1988.
10. ———, *Optimal linear randomization methods for linear operators in Hilbert spaces*, J. Complexity **8** (1992), 22–36.
11. ———, *Algorithms and complexity for continuous problems*, Geometry, Analysis, and Mechanics (J. M. Rassias, eds.), World Scientific, Singapore, 1994, pp. 96–128.
12. ———, *The real number model in numerical analysis*, J. Complexity **11** (1995), 57-73.
13. ———, *On the power of adaption* (1995) (to appear).
14. Novak, E., and Ritter, K., *Some complexity results for zero finding for univariate functions*, J. Complexity **9** (1993), 15–40.
15. Novak, E., Ritter, K. and Woźniakowski, *Average case optimality of a hybrid secant-bisection method*, Math. Comput. **64** (1995), 1517–1539.
16. Pan, V. V., *Optimal (up to polylog factors) sequential and parallel algorithms for approximating complex polynomial zeros*, Proc. 27th Ann. ACM Symp. on Theory of Computing (1995), ACM Press, New York.
17. Pereverzev, S. V., *On the complexity of the problem of finding solutions of Fredholm equations of the second kind with differentiable kernels (in Russian)*, Ukrain. Mat. Sh. **41** (1989), 1422–1425.
18. Plaskota, L., *Noisy Information and Computational Complexity*, Cambridge University Press, Cambridge, 1996.
19. Ritter, K., *Average errors for zero finding: lower bounds for smooth or monotone functions*, Aequationes Math. **48** (1994), 194–219.
20. Ritter, K. and Wasilkowski, G. W., *Integration and L_2-approximation: average case setting with isotropic Wiener measure for smooth functions*, Rocky Mount. J. Math. (1995) (to appear).
21. K. Sikorski, *Optimal solution of nonlinear equations*, J. Complexity **1** (1985), 197-209.
22. ———, *Study of linear information of polynomial equations*, Aequationes Math. **37** (1989), 1–14.
23. Smolyak, S. A., *Quadrature and interpolation formulas for tensor products of certain classes of functions*, Dokl. Akad. Nauk SSSR **4** (1963), 240–243.

24. Traub, J. F., Wasilkowski, G. W., and Woźniakowski, H., *Information-Based Complexity*, Academic Press, New York, 1988.
25. Traub, J. F. and Woźniakowski, H., *A general theory of optimal algorithms*, Academic Press, New York, 1980.
26. ______, *Recent Progress in Information-Based Complexity*, Bulletin of EATCS **51** (1993), 141-154.
27. Wasilkowski, G. W., *Randomization for Continuous Problems*, J. Complexity **5** (1989), 195–218.
28. ______, *Integration and approximation of multivariate functions: average case complexity with isotropic Wiener measure*, Bull. Amer. Math. Soc. (N.S) **28** (1993), 308–314.
29. Wasilkowski, G. W. and Woźniakowski, H., *Explicit cost bounds of algorithms for multivariate tensor product problems*, J. Complexity **11** (1995), 1–56.
30. ______, *On tractability of path integration* (1995) (to appear).
31. ______, *Worst case complexity of Feynman-Kac path integration* (1995) (to appear).
32. Werschulz, A. G., *The Computational Complexity of Differential and Integral Equations: An Information-Based Approach*, Oxford University Press, Oxford, 1991.
33. Woźniakowski, H., *Average Case Complexity of Multivariate Integration*, Bull. Amer. Math. Soc. (N.S) **24** (1991), 185–194.
34. ______, *Average case complexity of linear multivariate problems: Part I. Theory, Part II. Applications*, J. Complexity **8** (1992), 337–372, 373–392.
35. ______, *Tractability and strong tractability of linear multivariate problems*, J. Complexity **10** (1994), 96–128.
36. ______, *Tractability and strong tractability of multivariate tensor product problems*, J. Computing and Information **4** (1994), 1–19.

Department of Computer Science, Columbia University, New York, NY 10027, and Institute of Applied Mathematics, University of Warsaw, ul. Banacha 2, 02-097 Warsaw, Poland.

E-mail address: henryk@cs.columbia.edu