

On the Hardness of Approximate Multivariate Integration

Ioannis Koutis

Computer Science Department, Carnegie Mellon University
Pittsburgh, PA 15213 USA
`ioannis.koutis@cs.cmu.edu`

Abstract. We show that it is NP-hard to 2^{n^k} -approximate the integral of a positive, smooth, polynomial-time computable n -variate function, for any fixed integer k .

1 Introduction

Suppose $F(\cdot)$ is a real positive function defined on a cube C in Euclidean n -dimensional space \mathbf{R}^n . We consider the problem of approximating the integral $I(F)$ of F over C , with relative error ϵ , under the additional assumption that F satisfies a smoothness condition.

The exact integration of multivariate functions is hard, under the widely conjectured hardness of #P, given the result in [3], which implies that the exact calculation of the volume of an n -dimensional polytope is #P-complete. In view of this, we would like to address the question whether there is an algorithm that returns a value \hat{V} such that $1/(1 + \epsilon) \leq I(F)/\hat{V} \leq (1 + \epsilon)$, in other words an algorithm that ϵ -approximates $I(F)$.

The first somewhat surprising answer to this question came with the major result of Dyer, Frieze and Kannan ([5]), who showed that there is a fully polynomial randomized approximation scheme (FPRAS) for the volume of an n -dimensional convex body. More precisely, they showed that the volume of an n -dimensional convex body \mathcal{K} given by a weak membership oracle \mathcal{M} , can be ϵ -approximated with failure probability ξ , with $\text{poly}(n, \epsilon^{-1}, \log \xi^{-1})$ calls to \mathcal{M} . Here, \mathcal{M} can be thought of as a black-box algorithm that decides whether a given point is in \mathcal{K} . This directly implies that there is a FPRAS for the integration of n -variate concave functions that can be evaluated in time $\text{poly}(n)$ at any point in the cube C .

Subsequently, Applegate and Kannan ([2]), extended this result to positive, smooth and nearly log-concave functions. Define

$$f(X) = \ln F(X)$$

and let c be the edge length of C , $t(n)$ be an upper bound on the time needed to evaluate F at any point in C , and α, β satisfy

$$|f(X) - f(Y)| \leq \alpha \left(\max_{i \in [1, n]} |x_i - y_i| \right) \quad (1)$$

$$f(\lambda X + (1 - \lambda)Y) \geq \lambda f(X) + (1 - \lambda)f(Y) - \beta \quad (2)$$

for all $x, y \in C$ and $\lambda \in [0, 1]$. Their algorithm has running time

$$O(t(n) \frac{n^7}{\epsilon^2} c^2 \alpha^2 e^{2\beta} \log \frac{n}{\xi'} \log \frac{d\alpha n}{\xi}).$$

It can be seen that α measures the smoothness of F . This gives rise to the following definition of smoothness.

Definition 1. A function $F(\cdot)$ is called k -smooth if it satisfies $\alpha \leq n^k$. We denote by \mathcal{S}_k the set of k -smooth functions, and by $\mathcal{S} = \bigcup_k \mathcal{S}_k$ the set of smooth functions .

If $\beta = 0$, the function is log-concave (i.e. its logarithm is concave), so β can be viewed as a measure of the distance of F from log-concavity. The natural question is whether the dependence on β can be removed or somewhat alleviated. The contribution of this paper is to show that for any fixed integer k , it is NP-hard to 2^{n^k} -approximate the integral of positive smooth functions that are computable in polynomial time. In fact, we show that considerably small improvements on the dependence on β would imply unexpected (and rather indirect) algorithmic improvements for well studied NP-complete problems. Formally, we show the following.

Theorem 1. For any fixed integer $k \geq 3$, if there is a (randomized) 2^{n^k} -approximation algorithm with time complexity $O(\text{poly}(\alpha)2^{g(\beta)})$ for the problem of integration of functions from \mathcal{S}_{k+3} , then there is a $O(\text{poly}(\alpha)n^{(g(n))^{k+3}})$ (randomized) algorithm for the Hamilton Path problem on graphs with n vertices.

Corollary 1. For any fixed integer k , it is NP-hard to 2^{n^k} -approximate the integral of polynomial-time computable functions from \mathcal{S} .

We note here that, in general, only a few negative results concerning the approximability of counting problems are known. As observed in [6], the hardness of counting problems in most cases follows either from the NP-completeness of the corresponding decision problem, or from applying some “boosting” reduction which exploits an embedded NP-complete problem (see [10,6]). There appears to be a paucity of results that prove the hardness of approximate counting problems for some other more “interesting” reason. One such case is [4], which proves that there is no FPRAS for counting the number of independent sets in graphs of maximum degree $\Delta \geq 25$, unless NP=RP. As noted in [7], in view of the lack of “satisfactory” results that prove inapproximability under reasonable complexity-theoretical assumptions, research efforts have often been directed towards proving that certain restricted algorithmic approaches fail (see section 4 of [7] and the references therein).

The rest of the paper is organized as follows. In section 2 we give an overview of the proof technique, in section 3 we give the details of the proof and finally in section 4 we make some concluding comments.

2 Overview

We derive the result through a reduction from Hamilton Path (HP for short). Recall that HP is one of the first problems shown to be NP-complete (see [9]). Given a graph G (in some usual representation), HP asks whether there exists a simple path of length n , i.e. a path that goes through every vertex of G exactly once.

With every graph G , we associate a function F_G . If G has n vertices, F_G is a function of n^2 variables. The function F_G has the following useful characteristics. It can be computed at any point x in a cube C of interest, in time polynomial in n . The parameters α, β of F_G (defined in inequalities 1,2), are polynomial in n . Also, the value of the integral of F_G depends on whether G contains a Hamilton Path or not. Specifically, if there is a HP, the integral of F_G over a cube C of constant edge size c , is lower bounded by an explicitly known quantity I_H . If not, it is upper bounded by I_{NH} , with $I_H/I_{NH} \geq 2^{n^k}$, for any fixed constant k . It follows that the integral is not 2^{n^k} -approximable.

Also, since $\beta = O(n^d)$ for some constant d (the smallest value of d we are able to exhibit in this paper is 6), an improvement of the running time of the integration algorithm to $poly(n, \epsilon^{-1}, \alpha, 2^{b^{(1/d)} - \epsilon'})$, for any $\epsilon' > 0$, would give a $2^{o(n)}$ randomized algorithm for Hamilton Path (the best currently known upper bound is $O(2^n)$, see [1]), and through the Sparsification Lemma of [8] a $2^{o(n)}$ randomized algorithm for 3-SAT, where now n is the number of variables.

3 The Proof

3.1 Definition and Properties of the Function F_G

Let G be a graph with n vertices and \mathcal{P} be the set of length- n paths of G . The function $F_G(X)$ is a function of n^2 variables, $X = \{x_{11}, \dots, x_{nn}\}$. Each path $p \in \mathcal{P}$ is associated with a term $f_p(X)$, and $F_G(X) = \sum_{p \in \mathcal{P}} f_p(X)$.

We now describe the term $f_p(X)$ for a path p . Assume an arbitrary numbering of the graph vertices with numbers in $[n]$. We consider p as an ordered set of vertices v_1, \dots, v_n , where $v_i \in [n]$. We let $m = n^k$, where k is an integer constant to be discussed later. We define

$$f_p(X) = \prod_{i=1}^n g_i(X)$$

with

$$g_i(X) = \frac{\prod_{j=1}^{i-1} x_{v_i j}^m}{x_{v_i i}^m}$$

We will integrate F_G over the cube $C = [1, c]^{n^2}$, so we study its properties in this cube. Each term $f_p(X)$ is increasing in the variables appearing in the

numerator and decreasing in the variables appearing in the denominator. By setting the former to c and the latter to 1, we get that the maximum value of $f_p(X)$ is $O(c^{n^2m})$. Since there are at most $n!$ paths, it follows that for any $X \in C$, $F_G(X)$ can be expressed with $O(mn^3 \log n)$ bits.

As noted in [2], the smoothness parameter α , can be upper bounded by

$$\alpha \leq n^2 \max_{X \in C, x_i \in X} \left| \frac{\partial}{\partial x_i} \ln F(X) \right| = n^2 \max_{X \in C, x_i \in X} \left| \frac{\sum_{p \in \mathcal{P}} \partial f_p(X) / \partial x_i}{\sum_{p \in \mathcal{P}} f_p(X)} \right| \quad (3)$$

Let $x_i \in X$ be any variable. Since the exponent of x_i is at most nm , for all points X in C , we have

$$\frac{\partial f_p(X)}{\partial x_i} \leq nm f_p(X)$$

which combined with inequality 3, gives $\alpha \leq n^3m$.

A note about the algorithm of [2] is due here. The algorithm operates on a grid imposed on C . The coordinates of the grid are multiples of $\gamma \leq 1/2\alpha$. From the bound on α it follows that we are interested in evaluating F_G at points which are rationals expressible in polynomial space. From the definition of F_G , its value at any point of the grid is also a rational expressible in polynomial space.

The definition of β trivially implies that any upper bound for $f(X)$ is also an upper bound for β . From the above analysis we get $\beta \leq O(mn^3 \log n)$. For a lower bound on β note that $f(X)$ can be written as $f(X) = \ln P(X) - m \sum_{i,j \in [n]} \ln x_{ij}$, where $P(X)$ is a multivariate polynomial. Since $P(X)$ is not log-concave in general, the value of β can be lower bounded from the value of β for the function $\hat{f}(X) = -m \sum_{i,j \in [n]} \ln x_{ij}$, which can be seen to be $O(mn^2)$. Thus, we get $\beta \geq mn^2$.

We finally note that $F_G(X)$ has some additional interesting properties. First, F_G has derivatives of any order, everywhere in the cube C . Also, its form is relatively simple, as it is a sum of rational multivariate polynomials. In addition, given a graph G we can easily obtain a closed form for the integral of F_G , though of exponential length.

3.2 A Polynomial Time Algorithm for the Evaluation of F_G

We give an algorithm that computes $F_G(X)$ at any point X , in n time steps. We extend the definition of the path terms, to paths of length t . Concretely, we let

$$f_p(X) = \prod_{i=1}^t g_i(X)$$

with

$$g_i(X) = \frac{\prod_{j=1}^{i-1} x_{v_i j}^m}{x_{v_i i}^m}$$

Let $\mathcal{P}_t(v)$ be the set of paths of length t that end in node v . Also, let $Q_1(v) = x_{v1}^{-m}$. Inductively, assume that just before time step t , for every $v \in V$ we have computed

$$Q_{t-1}(v) = \sum_{p \in \mathcal{P}_{t-1}(v)} f_p(X)$$

Let $N(v)$ denote the set of neighbors of node v . At step t , for each node v we compute

$$Q_t(v) = \left(x_{vt}^{-m} \prod_{j=1}^{t-1} x_{vj}^m \right) \sum_{v' \in N(v)} Q_{t-1}(v')$$

After n steps the quantities $Q_n(v)$ have been computed for all vertices $v \in V$. Then,

$$F(X) = \sum_{v \in V} Q_n(v)$$

The computation of $Q_t(V)$ requires a polynomial number of operations. Since there are n steps and n vertices, it follows that F_G can be computed with a polynomial number of operations. The points we are interested in are rationals expressible in polynomial space, and from the observations of the previous subsection, all the intermediate quantities are expressible in polynomial space. It follows that $F_G(X)$ can be evaluated exactly, at any point $X \in C$, in time polynomial in n .

3.3 Bounding the Integrals

We integrate $F(X)$ over a cube $C = [1, c]^{n^2}$. Let $dX = dx_{11} \dots dx_{nn}$ and π be a permutation of the variable names. Since

$$\int_{X \in C} \sum_{p \in \mathcal{P}} f_p(X) dX = \sum_{p \in \mathcal{P}} \int_{X \in C} f_p(X) dX$$

we can consider the integral of each path separately. We will refer to the value of the integral of a term corresponding to a path p as the integral of p . Also, since

$$\int_{X \in C} f_p(x_{11}, \dots, x_{nn}) dX = \int_{X \in C} f_p(\pi(x_{11}), \dots, \pi(x_{nn})) dX$$

we can rename the variables in any term of F . It is then easy to see that the integral of a path depends only on the structure of the path and not on the particular vertices appearing on it.

We first consider the integral of a HP. Since HP is a simple path, there are no cancellations of variables and its integral is

$$\begin{aligned} I_{HP} &= \int_{X \in C} \left(\prod_{i=1}^n \frac{1}{x_{ii}^m} \right) \cdot \left(\prod_{1 \leq i \leq n, 1 \leq j \leq i-1} x_{ij}^m \right) dX = \\ &= (m+1)^{-n(n-1)/2} (m-1)^n (1-c^{-m+1})^n (c^{m+1}-1)^{n(n-1)/2} (c-1)^{n(n-1)/2} \end{aligned}$$

Let us now consider the integrals of other non-simple paths. Suppose a path p goes through $n - d$ distinct nodes. Then, the corresponding term f_p is of the form

$$f_p(X) = \left(\prod_{i=1}^{n-d} \frac{1}{x_i^m} \right) \cdot \left(\prod_{i=n-d+1}^{n(n+1)/2-t} x_i^{a_i m} \right)$$

where t and a_i are integers that depend on the structure of p . In this case, d monomials in the denominator cancel with variables in the numerator, so that

$$\sum_{i=n-d+1}^{n(n-1)/2-t} a_i = n(n-1)/2 - d$$

By integrating, we get

$$\begin{aligned} \int_{X \in C} f_p(X) &\leq (1 - c^{-m+1})^{n-d} (c - 1)^{n(n+1)/2+t} \prod_{i=n-d+1}^{n(n-1)/2-t} c^{a_i m+1} \\ &\leq c^{n^2} c^{m(\sum_{i=n-d+1}^{n(n-1)/2-t} a_i)} = c^{n^2} c^{mn(n-1)/2} c^{-md} \end{aligned}$$

Now suppose we are given a non-Hamiltonian graph. Since there are at most $n! \leq c^{n^2}$ paths in the graph, the integral of the associated function is

$$I_{NH} \leq c^{2n^2} c^{mn(n-1)/2} c^{-m}$$

On the other hand, if the given graph is Hamiltonian (and even if we consider only the integral of the HP), the integral of the associated function is

$$I_H \geq c^{-O(n^2 \log n)} c^{mn(n-1)/2}$$

which gives a large gap, namely

$$\frac{I_H}{I_{NH}} \geq c^{m-O(n^2 \log n)}$$

Recall that $m = n^k$. By taking any fixed $k \geq 3$ we get Theorem 1.

4 Conclusions

We showed that it is NP-hard to 2^{n^k} -approximate the integral of smooth positive n -variate functions, for any fixed integer k . We also argued that the currently best known integration algorithm cannot be substantially improved, unless there exist faster algorithms for Hamilton Path and 3-SAT.

Note that the 2^{n^k} -inapproximability holds for $(k + 3)$ -smooth functions, with $k \geq 3$. Also, in order to obtain the full range of our inapproximability result, we make use of functions that progressively become less efficiently computable. It

is an interesting question whether similar inapproximability properties can be shown for classes of functions with different trade-offs between their evaluation time complexity and the value of their α, β parameters.

We feel that the most interesting open question is whether a lower bound can be proved on β , for any smooth polynomially computable function F_G which can be constructed using the techniques of this paper.

References

1. Noga Alon, Raphael Yuster, Uri Zwick: Color Coding, *Journal of the ACM*, **42(4)** (1995) 844–856
2. David Applegate, Ravi Kannan: Sampling and Integration of Near Log-concave Functions, *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing* (1991) 156–163
3. Graham Brightwell, Peter Winkler: Counting Linear Extensions is #P-complete, *Proceedings of the 23th Annual ACM Symposium on Theory of Computing*, (1991) 175–181
4. Martin Dyer, Alan Frieze, Mark Jerrum: On Counting Independent Sets in Graphs, *Proceedings of the 40nd Annual Symposium on the Foundations of Computer Science*, (1999) 210–217
5. Martin Dyer, Alan Frieze, Ravi Kannan: A Random Polynomial-time Algorithm for Approximating the Volume of Convex Bodies, *Journal of the ACM*, **38**, (1991) 1–17
6. Martin Dyer, Leslie Ann Goldberg, Catherine Greenhill, Mark Jerrum: On the Relative Complexity of Approximate Counting Problems, *Algorithmica*, to appear
7. Martin Dyer, Catherine Greenhill: Random Walks on Combinatorial Objects, *Surveys in Combinatorics*, **267** (1999) 101–136
8. Rusell Impagliazzo, Ramamohan Paturi: Which Problems Have Strongly Exponential Complexity?, *Proceedings of the 39th Annual Symposium on the Foundations of Computer Science* (1998)
9. Christos Papadimitriou: *Computational Complexity*, Addison-Wesley (1994)
10. Alistair Sinclair: *Algorithms for Random Generation and Counting: A Markov Chain Approach*, *Progress in Theoretical Computer Science*, (1993)