Electronic Colloquium on Computational Complexity, Report No. 142 (2012)

Noncommutativity makes determinants hard

Markus Bläser
Saarland University
mblaeser@cs.uni-saarland.de

We consider the complexity of computing the determinant over arbitrary finite-dimensional algebras. We first consider the case that A is fixed. We obtain the following dichotomy: If $A/\operatorname{rad} A$ is noncommutative, then computing the determinant over A is hard. "Hard" here means #P-hard over fields of characteristic 0 and $\mathsf{Mod}_p\mathsf{P}$-hard over fields of characteristic $p > 0$. If $A/\operatorname{rad} A$ is commutative and the underlying field is perfect, then we can compute the determinant over A in polynomial time.

We also consider the case when A is part of the input. Here the hardness is closely related to the nilpotency index of the commutator ideal of A. The commutator ideal $\operatorname{com}(A)$ of A is the ideal generated by all elements of the form $xy - yx$ with $x, y \in A$. We prove that if the nilpotency index of $\operatorname{com}(A)$ is linear in n, where $n \times n$ is the format of the given matrix, then computing the determinant is hard. On the other hand, we show the following upper bound: Assume that there is an algebra $B \subseteq A$ with $B = A/\operatorname{rad}(A)$. (If the underlying field is perfect, then this is always true.) The center $Z(A)$ of A is the set of all elements that commute with all other elements. It is a commutative subalgebra. We call an ideal J a complete ideal of noncommuting elements if $B + Z(A) + J = A$. If there is such a J with nilpotency index $o(n/\log n)$, then we can compute the determinant in subexponential time. Therefore, the determinant cannot be hard in this case, assuming the counting version of the exponential time hypothesis.

Our results answer several open questions posed by Chien et al. [4].

1 Introduction

The determinant of a matrix $M = (m_{i,j}) \in k^{n\times n}$ is given by the well-known formula

$$\det M = \sum_{\sigma\in S_n} \operatorname{sgn}(\sigma) m_{1,\sigma(1)} \cdots m_{n,\sigma(n)}.$$

The determinant plays a central role in linear algebra. It can be efficiently computed, for instance, by Gaussian elimination. In fact, there are even efficient algorithms when the matrix M has entries from some commutative algebra, see [12] and the references given therein.

A related polynomial is the permanent of M, given by

$$\operatorname{per} M = \sum_{\sigma\in S_n} m_{1,\sigma(1)} \cdots m_{n,\sigma(n)}.$$

ISSN 1433-8092

If M is $\{0,1\}$-valued, then $\operatorname{per} M$ is the number of perfect matchings of the bipartite graph defined by M. While the determinant is easy over commutative algebras, the permanent is hard already over the rationals. Valiant [15] showed that evaluating the $\{0,1\}$-permanent over the rationals is at least as hard as counting the number of satisfying assignments of a formula in 3-CNF.

Since the determinant and the permanent have similar formulas, it is tempting to try to modify algorithms for the determinant and use them to compute the permanent. Godsil and Gutman [9] used the determinant to approximate the permanent. They designed a matrix-valued random variable. In expectation, the square of the determinant of this random variable is the permanent. However, the variance is huge. Karmarkar et al. [11] showed how to lower the variance by extending the underlying field to the complex numbers. Chien et al. [6], building upon the work by Barvinok [2], showed that if one could compute the determinant of an $n \times n$-matrix the entries of which are themselves matrices of size $cn \times cn$ for some constant c, then there is a fully polynomial time randomized approximation scheme for the permanent of $\{0,1\}$-matrices. See [13] for further results in this direction. (Of course, there is a fully polynomial randomized approximation scheme based on Markov chains, see [10]. However, if we could evaluate noncommutative determinants as fast as commutative ones, then we would get much faster approximation schemes.)

Therefore, it is important to understand the complexity of the determinant over arbitrary finite-dimensional algebras, especially over noncommutative ones, and not only over fields or commutative algebras. The first to study this problem was Nisan [14]. He proved an exponential lower bound for the size of an algebraic branching program for computing the determinant over the free noncommutative algebra $k\langle X_{i,j}\rangle$. While the lower bound is strong, the setting is limited, because it only applies to a restricted circuit model and only to a very "powerful" algebra. Chien and Sinclair [5] extended these bounds to a wide range of "concrete" algebras by analysing their polynomial identities, for instance to matrix algebras and the Hamiltonian quaternions, albeit only in the algebraic branching program model.

Recently Arvind and Srinivasan [1] showed that the noncommutative determinant cannot have small circuits unless the permanent has small circuits. Finally, Chien et al. [4] made further progress by proving the #P-hardness and $\mathsf{Mod}_p\mathsf{P}$-hardness of the determinant for odd p for large classes of algebras.

The fundamental question behind these results is: Which properties of the algebra makes the determinant hard? In this work, we prove that this is exactly noncommutativity.

1.1 A crash course on the structure of algebras

An associative algebra A over some field k is a k-vector space together with a bilinear mapping $\cdot : A \times A \to A$, the multiplication in A. Multiplication is associative and distributes over addition. If $\lambda \in k$, then $\lambda(x \cdot y) = (\lambda x) \cdot y = x \cdot (\lambda y)$ for all $x, y \in A$. We will always assume that A is finite-dimensional (as a vector space) and contains a unit element, which we denote by 1.

A left (right, twosided) *ideal* of an algebra is a vector space that is closed under multiplication with arbitrary elements of A from the left (right, both sides). If S is a subset of A, then the left (right, twosided) ideal of A generated by S is the intersection of all left (right, twosided) ideals that contain S. Alternatively, it can be defined as the linear span generated by all elements xs (sy, xsy) with $x, y \in A$ and $s \in S$.

A left (right, twosided) ideal I is called *nilpotent*, if $I^s = \{0\}$ for some positive integer s.

The nilpotency index of I is the smallest s such that $I^s = \{0\}$. If there is no such s, then the index is infinite.

The sum of all nilpotent left ideals of A is a nilpotent twosided ideal, which contains every nilpotent right ideal of A. This twosided ideal is called the *radical* of A and is denoted by $\operatorname{rad} A$. The quotient algebra $A/\operatorname{rad} A$ contains no nilpotent ideals other than the zero ideal. Since A is finite dimensional, we can alternatively define the radical of A as the intersection of all maximal twosided ideals. An ideal is maximal if it is not contained in any other ideal and is not equal to A.

We call an algebra A *semisimple*, if $\operatorname{rad} A = \{0\}$. By the above fact, $A/\operatorname{rad} A$ is semisimple. An algebra A is called *simple*, if there are no twosided ideals in A except the zero ideal and A itself. An algebra D is called a *division algebra*, if $D^\times = D \setminus \{0\}$. Here $D^\times$ is the set of all invertible elements in D. An algebra A is called *local*, if $A/\operatorname{rad} A$ is a division algebra.

The following fundamental theorem describes the structure of semisimple algebras.

Theorem 1 (Wedderburn) *Every finite dimensional semisimple algebra is isomorphic to a finite direct product of simple algebras. Every finite dimensional simple k-algebra A is isomorphic to an algebra $D^{n\times n}$ for an integer $n \geq 1$ and a k-division algebra D. The integer n and the algebra D are uniquely determined by A (the latter one up to isomorphism).*

For an introduction to associative algebras, we recommend [8].

1.2 Our results

First we will consider the problem when the underlying algebra A is fixed: We are given a matrix $M \in A^{n\times n}$ as an input and our task is to compute $\det M$. We prove that the determinant over A is hard if $A/\operatorname{rad} A$ is noncommutative. If $A/\operatorname{rad} A$ is commutative, then the problem is polynomial time computable. That means, we get a complete dichotomy (Theorem 4). More precisely, we show that

- computing the determinant over A is #P-hard if $A/\operatorname{rad} A$ is noncommutative and the characteristic of k is 0.
- computing the determinant over A is $\mathsf{Mod}_p\mathsf{P}$-hard if $A/\operatorname{rad} A$ is noncommutative and the characteristic p of k is positive.

Chien et al. show that if $A/\operatorname{rad} A$ is commutative and the field k is perfect, then the determinant can be computed in polynomial time. A field is perfect if every irreducible polynomial over k has distinct roots. Any "reasonable" field ist perfect, for instance, fields of characteristic zero are prefect, finite fields are perfect as well as algebraically closed fields.[1]

Our dichotomy extends the results of Chien et al. in two ways: First it works for arbitrary algebras A such that $A/\operatorname{rad} A$ is noncommutative. Chien et al. proved this only for algebras whose semisimple part $A/\operatorname{rad} A$ contained at least one matrix algebra. For instance, it did not apply to local algebras and in particular, division algebras like Hamiltonian quaternions. Second, we get $\mathsf{Mod}_2\mathsf{P}$-hardness, that is, $\oplus\mathsf{P}$-hardness, over fields of characteristic 2. The proof by Chien et al. did not work in this case.

[1] What is actually needed by Chien et al. is that there is a subalgebra B of A such that $A = B \oplus \operatorname{rad} A$ (as vector spaces). This is true if the algebra A is separable. Over perfect fields, every algebra is separable. Any of these implications is often called the Wedderburn-Malcev Theorem. The existence of the algebra B is only needed for upper bound not for the hardness result.

Then we turn to the case when the algebra is given as a part of the input. Beside the matrix M, we also get a basis and the multiplication table of the algebra A from which the entries of M are taken. It seems to be natural that the dimension of A should be polynomial in the size of M. The setting above subsumes the case where we have a familiy of algebras A_n and our task is to compute the $n \times n$-determinant over A_n, for instance, computing the determinant of $n \times n$-matrices with upper triangular $n \times n$-matrices as entries. This setting is of interest because there could be a sequence of algebras each of which is noncommutative but still the determinant is easy to compute. This of course is only possible if $A_n/\operatorname{rad} A_n$ is commutative, by our first result.

We give evidence that the quantity that determines the hardness is the *nilpotency index* of the *commutator ideal* of A. The commutator ideal $\operatorname{com}(A)$ of an algebra A is the ideal generated by all elements of the form $xy - yx$ with $x, y \in A$. If the commutator ideal $\operatorname{com}(A) = \{0\}$, then A is commutative. If its nilpotency index is finite, then $A/\operatorname{rad} A$ is commutative. We prove that if the nilpotency index of the commutator ideal of A is linear in n, then computing the determinant of $n \times n$-matrices is as hard as counting the number of solutions of a formula in 3-CNF modulo the characteristic of k (Theorem 5).

We prove an upper bound that is a little weaker in two ways: First we need that the nilpotency index of a somewhat larger ideal is bounded and second the upper bound does not fully match the lower bound from the hardness result. Assume that there is an algebra $B \subseteq A$ with $B \cong A/\operatorname{rad}(A)$. (If the underlying field is perfect, then this is always true.) The *center* $Z(A)$ of A is the set of all elements that commute with all other elements. It is a commutative subalgebra. $B + Z(A)$ is a commutative subalgebra, too. We call an ideal J a *complete ideal of noncommuting elements* if $B + Z(A) + J = A$. If there is such a J with nilpotency index r, then we can compute the determinants of $n \times n$-matrices over A in time $n^{O(r)}$ (Theorem 7).

Over fields of characteristic 0 this result is almost tight assuming the counting version of the exponential time hypothesis #ETH as formulated by Dell et al. [7]. From Theorems 7, it follows that if $r = o(n/\log n)$, then computing the determinant over A cannot be #P-hard under #ETH.

2 Determinants, permanents, and cycle covers

Given an $n \times n$-matrix $M = (m_{i,j})$ the entries of which belong to an algebra A, the *(Cayley) determinant* of M is defined by

$$\det M = \sum_{\sigma \in S_n} \operatorname{sgn}(\sigma) m_{1,\sigma(1)} \cdots m_{n,\sigma(n)}. \tag{1}$$

(Since A might be noncommutative, the order of multiplication makes a difference. When the order is by rows, then we get the Cayley determinant.) Similarly, the permanent of M is defined by

$$\operatorname{per} M = \sum_{\sigma \in S_n} m_{1,\sigma(1)} \cdots m_{n,\sigma(n)}. \tag{2}$$

We can interpret the matrix M as an edge-weighted digraph on the vertex set $V = \{1, \ldots, n\}$. There is an edge from i to j if $m_{i,j} \neq 0$ and the weight of this edge is $m_{i,j}$. We denote this graph by $G(M)$. A cycle cover C of a digraph is a subset of the edges such that every node has indegree and outdegree one in C. C encodes a unique permutation, which maps every node i to the node j where (i, j) is the unique edge in C leaving i. We set $C(i) := j$. In this way,

we can interpret C as a permutation. It is easy to see that $\operatorname{sgn}(C) = (-1)^{n+c}$ where c is the number of cycles in C. The weight of a cycle cover is the product of the weights of the edges in C, that is, $m_{1,C(1)} \cdots m_{n,C(n)}$. Again the order is important, since the weights might not commute. For a digraph G, let $\mathrm{CC}(G)$ be the set of its cycle covers. Now we can rewrite (1) and (2) as

$$\det M = \sum_{C \in \mathrm{CC}(G(M))} \operatorname{sgn}(C) m_{1,C(1)} \cdots m_{n,C(n)} \tag{3}$$

and

$$\operatorname{per} M = \sum_{C \in \mathrm{CC}(G(M))} m_{1,C(1)} \cdots m_{n,C(n)}. \tag{4}$$

If G is an edge-weighted digraph, we will often write $\det G$ and $\operatorname{per} G$ for the determinant and permanent of its weighted adjacency matrix.

3 Hardness proofs for the permanent

#3-SAT is the following problem: Given a Boolean formula ϕ in 3-CNF with n variables and m clauses, count the number of satisfying assignment. #3-SAT is #P-complete. It even stays #P-complete if we assume that every variable appears as often unnegated as negated. We can achieve this by adding trivial clauses of the form $\bar{x} \vee x \vee x$ or $\bar{x} \vee \bar{x} \vee x$ for every variable x, if neccessary. This reduction increases the size of ϕ only by a constant factor. Note that thereafter, every assignment sets as many literals to true as to false.

We first briefly review the reduction by Dell et al. [7] of #3-SAT to the permanent, which is similar to the original construction by Valiant [15], but simpler and nicer. (It should go into any modern textboox.) The reduction by Dell et al. is itself derived from the reduction in [3]. Chien et al. [4] used the same approach; however, our gadgets can handle arbitrary noncommutative algebras and not only matrix algebras.

A given formula ϕ is mapped to a graph G_ϕ. This graph will have $O(m)$ edges. For every variable x, there is a selector gadget, see Figure 1 (left-hand side). There are two ways to cover this gadget by a cycle cover, taking the left-hand edge will correspond to setting x to zero and taking the right-hand edge will correspond to setting x to one.

For every clause, there is a clause gadget as depicted in Figure 1 (right-hand side). Each of the three outer edges corresponds to one literal of the clause. Taking one of the three outer edges corresponds to setting the literal to zero. For every subset of the outer edges, except for the one consisting of all three outer edges, there is exactly one cycle cover, see Figure 2. Call the graph constructed so far G'_ϕ. A cycle cover of G'_ϕ is called *consistent*, if the chosen edges in the selector gadgets and the clause gadgets are consistent, that is, whenever we chose the left-hand edge in the selector gadget for x (i.e, $x = 0$), then we choose all corresponding edges in the clause gadgets in which x appears positively and vice versa.

Fact 2 *Satisfying assignments of ϕ and consistent cycle covers of G'_ϕ stand in one-to-one correspondence.*

The last step is to get rid of inconsistent cycle covers. This is done by connecting the edge of a literal ℓ in a clause gadget by the edge in the selector gadget corresponding to setting $\ell = 0$ using an equality gadget, see Figure 3. The edge of the selector gate and the edge of the clause gadget are subdivided, let x and z be the newly introduced vertices. These two vertices

Figure 1: Left-hand side: The selector gadget. In all Figures, edges with out explicitely stated weights have weight 1. Right-hand side: The clause gadget. In the gadget as it is, there is a double edge between the two nodes at the bottom. The lower edge is however subdivided when we introduce the equality gadgets.

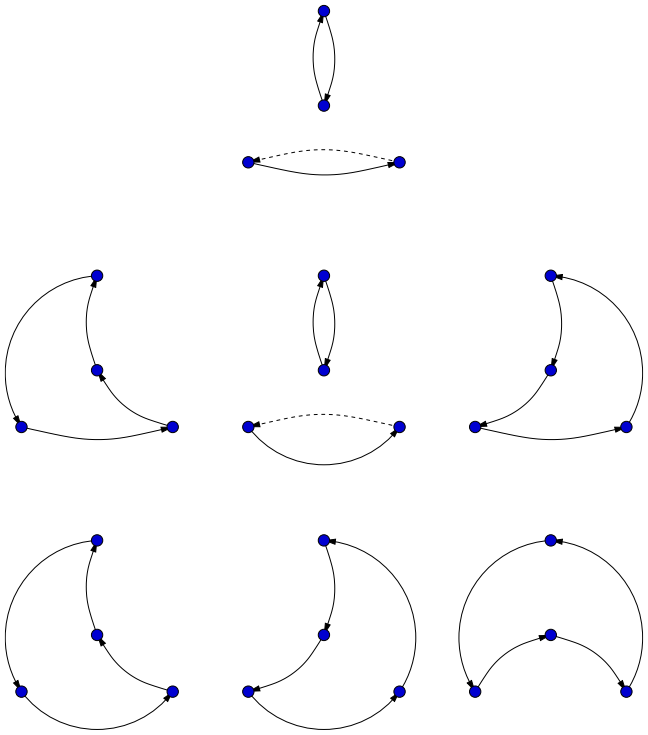

Figure 2: Every proper subset of the outer edges can be extended to a unique cycle cover of the clause gadget. The dashed edge only appears in the covers with an even number of cycle. It will get weight -1 later, when we consider the determinant, to compensate the sign flip.

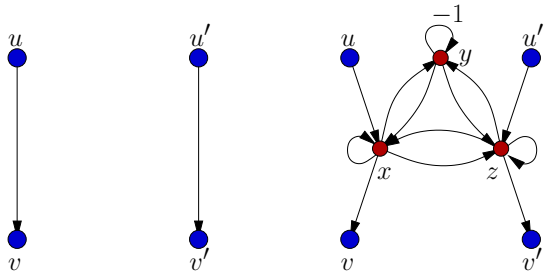

Figure 3: The equality gadget. The pair of edges (u, v) and (u', v') of the left-hand side, one of them is an edge of the selector gadget and the other is the corresponding outer edge of a clause gadget, is connected as shown on the right-hand side.

are connected as depicted in Figure 3. Since a literal appears in several clauses, the edge of the selector gadget is subdivived as many times.

Every consistent cycle cover of G'_ϕ can be extended to several cycle covers of G_ϕ. If the two edges connected by a equality gadget are both taken, then we take both path $u - x - v$ and $u' - y - v'$ in G_ϕ. The interior vertex y is covered by the self-loop, yielding a weight of -1. If both edges are not taken, then we take none of the corresponding paths. There are six possibility to cover the interior nodes x, y, and z; four of them have weight 1, two of them have weight -1. This sums up to 2. (The six different covers, albeit with different weights, are shown in Figure 5.) Therefore, every consistent cycle cover is mapped to several cycle covers with a total weight of $(-1)^p 2^q$ where p is the number of literals set to zero and q is the number of literals set to one. Since we normalized ϕ, $p = q = 3m/2$.

There are also cycle covers that do not cover equality gadget consistently. This can either mean that the path $u - x - v$ is taken but not $u' - y - v'$ or that we enter the gadget via u but leave it via v'. One can prove that all cycle covers in which at least one equality gadget is not covered consistently sum up to zero. Altogether, we get that $\operatorname{per} G_\phi = (-2)^{3m/2} \cdot \#\text{3-SAT}(\phi)$, where $\#\text{3-SAT}(\phi)$ denotes the number of satisfying assignments of ϕ.

4 Hardness of the noncommutative determinant

We adapt the construction of the previous section to the determinant over noncommutative algebras. Note that now every cycle cover C is weighted by $\operatorname{sgn}(C)$ and the order in which the edge weights are multiplied is important. The selector gadgets stay the same. The clause gadgets stay almost the same, the only difference is that one edge gets weight -1 as is done by Chien et al., too, see Figure 2. As before, for every proper subset of the outer edges, there is one cycle cover covering the clause gadget. The new -1-weight compensates the fact that some covers contain an odd number of cycles and some an even number. Let again G'_ϕ denote the resulting graph. Consistent cycle covers of G'_ϕ with sign stand in one-to-one correspondance with satisfying assignments of ϕ.

Note that since we are now working over some noncommutative algebra, the order of the vertices can be important: Up to now, we used only edge weights 1 or -1. Therefore, the order of the vertices does not matter so far.

The structure of the equality gadgets also stays the same, but we use different weights. To

construct the weights, we use the following lemma.

Lemma 3 *Let A be an associative algebra. $A/\operatorname{rad} A$ is noncommutative if and only if there are invertible $i, j \in A$ such that $1 - iji^{-1}j^{-1}$ is not nilpotent.*

Proof. Assume that $A/\operatorname{rad} A$ is noncommutative and let $A/\operatorname{rad} A = A_1 \times \cdots \times A_t$ be its decomposition into simple algebras as given by Wedderburn's Theorem. One of these factors, say A_1, is either a matrix algebra of the form $B^{s\times s}$ with B being a division algebra and $s \geq 2$ or a noncommutative division algebra D. In the first case $A_1 = B^{s\times s}$, set

$$i' = \begin{pmatrix} 1 & 1 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix} \quad \text{and} \quad j' = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix}$$

It is easy to check that

$$i'j' - j'i' = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & -1 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix}.$$

$(i'j' - j'i')^2$ is idempotent in A_1 and therefore $i'j' - j'i'$ cannot be nilpotent. In the second case $A_1 = D$, we choose i' and j' to be noncommuting elements in D. $i'j' - j'i'$ is nonzero and therefore invertible in A_1, as D is a division algebra. The elements $i = (i', 1, \dots, 1)$ and $j = (j', 1, \dots, 1)$ are invertible in $A/\operatorname{rad} A$ and can be lifted to invertible elements of A. $ij - ji = (i'j' - j'i', 0, \dots, 0)$ is not nilpotent. We have

$$1 - iji^{-1}j^{-1} = -(ij - ji) \cdot i^{-1}j^{-1},$$

which is not nilpotent, either.[2]

For the converse direction, note that $1 - iji^{-1}j^{-1} \notin \operatorname{rad} A$, since $1 - iji^{-1}j^{-1}$ is not nilpotent. Therefore the image of $1 - iji^{-1}j^{-1}$ in $A/\operatorname{rad} A$ under the canonical projection is nonzero and thus, $A/\operatorname{rad} A$ is not commutative. ∎

Let A be an algebra such that $A/\operatorname{rad} A$ is noncommutative. Choose i and j as constructed above. Let $k = ij$. The edges of the equality gadget get weights as depicted in Figure 4. The three new vertices x, y, and z of each gadget appear consecutively in the order x, y, z in the ordering of all the vertices. Let G_ϕ denote the resulting graph.

Now we analyse what happens with a consistent cycle cover C of G'_ϕ when moving over to G_ϕ, see Figure 5. If both paths in the equality gadget are taken, then we cover y by the self-loop. This adds one cycle to the cycle cover, which toggles the sign. If both paths are not taken, then there are six cycles covers. Two of them, have one cycle and signed weights[3]

[2] To not fall into the same trap as a STOC'12 referee, please note that this is not true in general. Here this holds because of the choice of i' and j'. Either A_1 is a noncommutative division algebra or A_1 is a matrix algebra. In the first case, being nonzero already means invertible. In the second case, note that $-(ij - ji) \cdot i^{-1}j^{-1}$ is a matrix with an invertible 2×2-matrix in the upper left corner and zeros elsewhere.

[3] The term signed weight also includes the change of sign induced by the parity change of the cycles.

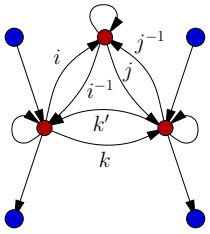

Figure 4: The modified equality gadget. i and j are the elements constructed in the proof of Lemma 3, $k = ij$, and $k' = i^{-1}j^{-1}$. The edges between x and y have weight i and i^{-1}, between y and z weights j and j^{-1}, and between z and x weights k' and k.

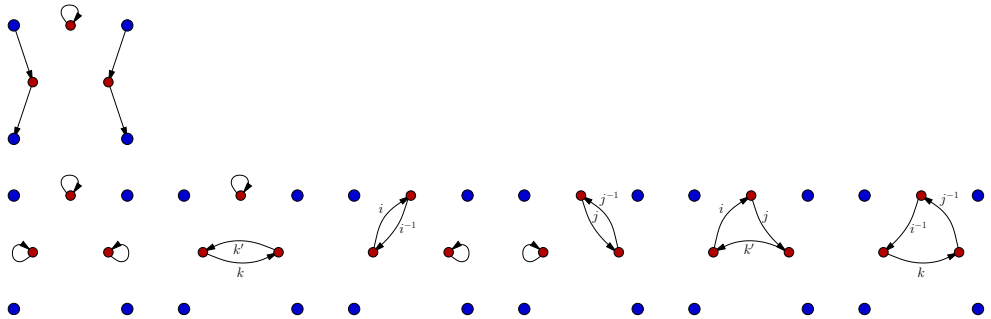

Figure 5: First row: The one possible configuration if both edges are taken. Second row: The six possible configurations if none of the edges is taken.

$-ijk' = -iji^{-1}j^{-1}$ and $-ki^{-1}j^{-1} = -iji^{-1}j^{-1}$. Three of them have two cycles and signed weights $ii^{-1} = 1$, $jj^{-1} = 1$, and $kk' = iji^{-1}j^{-1}$. Finally, there is one cycle cover with three cycles and signed weight -1. The total signed weight contribution is $1 - iji^{-1}j^{-1}$. Doing this for all equality gadgets, we get that every consistent cycle cover of G'_ϕ can be extended to consistent cycle covers of G_ϕ with total signed weight

$$(-1)^{3m/2}(1 - iji^{-1}j^{-1})^{3m/2}.$$

Recall that we normalized ϕ such that every assignment sets $3m/2$ literals to true and $3m/2$ literals to false. Since $1 - iji^{-1}j^{-1}$ is not nilpotent, this weight is nonzero.

It remains to analyse what happens with cycle covers of G_ϕ which are not consistent, that is, in which at least one equality gadget is not covered consistently. We will define an involution I without fixed points on the set of all inconsistent cycle covers of G_ϕ such that the weight of C and $I(C)$ cancel each other. From this it follows that the total contribution of the inconsistent cycle covers is zero. To define I, take an inconsistent cycle cover. We order the equality gadgets arbitrarily. Let C be an inconsistent cycle cover and consider the first inconsistent equality gadget. Then either C uses the path $u - x - v$ in this gadget but not $u' - y - v'$ or it enters the gadget via u and leaves it via v'. (The cases where $u' - y - v'$ is used but not $u - x - v$ or the gadget is entered via u' and left via v are symmetric.) Figure 6 shows how I pairs inconsistent cycle covers.

In the first case, C and $I(C)$ only differ in how y and z are covered. On the lefthand side, we use two cycles of weight 1, on the righthand side we use one cycle of weight $jj^{-1} = 1$. So

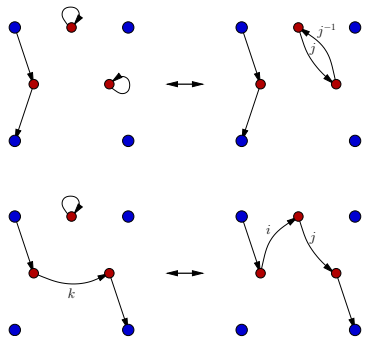

Figure 6: The involution I. I maps the configuration on the left-hand side to the corresponding configuration on the right-hand side and vice versa.

the weights of the cycle covers are the same, but the signs differ, since the cycle cover on the lefthand side has one cycle more. (In the symmetric case, we get two cycles of weight 1 versus one cycle of weight $ii^{-1} = 1$.)

In the second case, we either use one edge of weight k and cover y by a cycle of weight 1 (lefthand side), or we use two edges of weight i and j. Since $k = ij$, the weight of both covers is the same, but again the signs differ, since the second cover has one cycle more. (In the symmetric case, we have one edge with weight $i^{-1}j^{-1}$ and one additional cycle. or two edges with weight $i^{-1}j^{-1}$.)

This finishs the proof that the contribution of the inconsistent cycle covers is 0.

Algother, we get that

$$\det(G) = (-1)^{3m/2}(1 - iji^{-1}j^{-1})^{3m/2}\#\text{3-SAT}(\phi). \tag{5}$$

Note that $(-1)^{3m/2}(1 - iji^{-1}j^{-1})^{3m/2}$ is a fixed nonzero element of A multiplied by the scalar $\#$3-SAT(ϕ).

Theorem 4 *Let k be a field of characteristic p. Let A be an associative algebra over k.*

1. *If $A/\operatorname{rad} A$ is noncommutative and k is perfect, then evaluating the determinant over A is $\#\mathsf{P}$ hard if $p = 0$ and $\mathsf{Mod}_p\mathsf{P}$-hard otherwise.*
2. *If $A/\operatorname{rad} A$ is commutative, then the determinant over A can be evaluated in polynomial time.*

Proof. The first part immediately follows from (5), since $(-1)^{3m/2}(1 - iji^{-1}j^{-1})^{3m/2}$ is a nonzero element of A by the choice of i and j. Note that if $p > 0$, then we get $\#$3-SAT(ϕ), which is a scalar from k, only modulo p.

The second part follows from the fact that there is an algorithm with running time $n^{O(d)}$ for this problem, where d is the dimension of A [4]. Note that A is fixed, so d is a constant. ∎

5 Algebras as part of the input

Theorem 4 resolves the complexity of the noncommutative determinant when the underlying algebra is fixed. Next, we deal with the case when the algebra can grow with the size of

the matrix. This can be modelled in two ways. Either we have an family of algebras A_n and when given an $n \times n$-matrix, we want to compute its determinant over A_n. Even more general is the case when the algebra is part of the input. Here we get the algebra as a basis and a multiplication table. Given such a multiplication table, there are efficient algorithms to compute the nilpotence index of the radical, a Wedderburn-Malcev decomposition, and the commutator of the algebra, see [4] and the references given therein.

If the algebra A is part of the input, the noncommutative determinant might be even hard, if $A/\operatorname{rad} A$ is commutative. For instance, the determinant over the upper triangular matrices of dimension linear in the size is hard as shown by Chien et al. [4]. The goal of this section is to get a characterisation similar to the one in the previous section.

5.1 Hardness result

The set-up is the same as in the previous section. We try to reduce #3-SAT to the computation of the noncommutative determinant. The selector and the clause gadget will stay the same. Figure 7 shows an equality gadget with general weights.

As before, if we take both path $u - x - v$ and $u' - y - v'$, there is only one way to cover the gadget: taking the loop at y. This gives signed weight $-a$. We need that

$$a \neq 0. \tag{6}$$

If we take none of the paths, then there are six ways to cover the gadet, see Figure 5. The total contribution should be nonzero, that means,

$$-cab + ii'b + cjj' + kak' - ijk' - ki'j' \neq 0. \tag{7}$$

If the equality gadget is not covered consistently, then we define the involution I as before, see Figure 6. Since the weight of C and $I(C)$ shall cancel each other, we get the conditions

$$jj' = ab \qquad \text{and} \qquad ii' = ca \tag{8}$$

and

$$ij = ka \qquad \text{and} \qquad i'j' = ak' \tag{9}$$

We set $a = 1$. Then (6) is fullfilled and from (8) and (9) we get

$$b = jj', \quad c = ii', \quad k = ij, \quad k' = i'j'.$$

(7) simplifies to

$$-ii'jj' + ii'jj' + ii'jj' + iji'j' - iji'j' - iji'j' = i(i'j - ji')j' \neq 0.$$

This means that for one equality gadget, we need two noncommuting elements i' and j. We can multiply their commutator from the left and right with i and j', respectively. This means that $i(i'j - ji')j'$ i an element of $\operatorname{com}(A)$. Altogether, we have $3m/2$ equality gadgets. Each of them gets weights i_μ, i'_μ, j_μ, and j'_μ. Every satisfying assignment of ϕ now corresponds to a consistent cycle cover of G_ϕ of weight

$$(-1)^{3m/2} i_1(i'_1 j_1 - j_1 i'_1)j'_1 \cdots i_{3m/2}(i'_{3m/2} j_{3m/2} - j_{3m/2} i'_{3m/2})j'_{3m/2}.$$

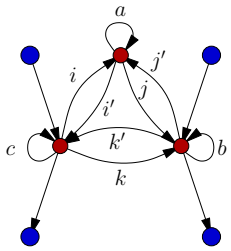

Figure 7: The general equality gadget.

This weight shall be nonzero, that means,

$$i_1(i'_1j_1 - j_1i'_1)j'_1 \cdots i_{3m/2}(i'_{3m/2}j_{3m/2} - j_{3m/2}i'_{3m/2})j'_{3m/2} \neq 0. \tag{10}$$

We can choose such weights if and only if the nilpotency index of the commutator ideal is at least $3m/2$. The elements of the commutator ideal are of the form $a_1(x_1y_1 - y_1x_1)b_1 + \cdots a_s(x_sy_s - y_sx_s)b_s$. However, if we can find $3m/2$ elements of this form such that their product is nonzero, we can also find $3m/2$ elements as in (10) by using distributivity. This show that if the nilpotency index of the commutator of A is large enough, i.e., at least $3m/2$, then computing the determinant over A is hard.[4]

Theorem 5 *Let ϕ be a formula in 3-CNF with m clauses such that every variable appears as many times positively as negatively. Let k be a field of characteristic p and A be an associative k-algebra whose commutator has nilpotence index $\geq 3m/2$. We can construct in polynomial time a matrix M such that*

$$\det M = \#3\text{-SAT}(\phi) \cdot x$$

for some nonzero element $x \in A$.

5.2 Upper bound [5]

The center $Z(A)$ is the set of all elements that commute with all other elements, i.e., $Z(A) = \{x \mid xy - yx = 0 \text{ for all } y \in A\}$. We assume that the underlying field is perfect. By the Wedderburn-Malcev theorem, there is a subalgebra $B \subseteq A$ such that $A = B \oplus \operatorname{rad} A$.[6] Note that B is commutative; otherwise the nilpotency index of the commutator of A would be infinite. $B + Z(A)$ is a commutative subalgebra of A.

Definition 6 *We call an ideal J a complete ideal of noncommuting elements, if $B+Z(A)+J = A$. (Note that $B + Z(A)$ and J might intersect.)*

[4]For the reduction, we also need to be able to compute the weights in polynomial time. This can be done as follows: From the given basis of A, we can compute a basis of $\operatorname{com}(A)$ by just computing all possible elements $a(xy-xy)b$ with a, b, x, y taken from the given basis and choosing a maximum linearly independent subset. Then we successively compute bases of $\operatorname{com}(A)^i$ in the same way. For each newly formed element, we keep track how it is written as a product of i elements from $\operatorname{com}(A)$. Since the nilpotency index of $\operatorname{com}(A)$ is $> 3m/2$, $\operatorname{com}(A)^{3m/2}$ contains a nonzero element and we know how to write it as a product of $3m/2$ elements from $\operatorname{com}(A)$.

[5]I would like to thank the STOC'12 referee from the footnote on page 8 for pointing out a flaw in a previous version of this section.

[6]What we simply need for our proof is the existence of B, which always exists over perfect fields.

On the other hand, if the nilpotency index of any complete ideal of noncommuting elements is small, then there is an efficient algorithm for computing the determinant. Let r be the nilpotency index of *any* such J.

Let $M = (m_{i,j})$ be the given matrix. We decompose $m_{i,j} = b_{i,j} + r_{i,j}$ with $b_{i,j} \in B + Z(A)$ and $r_{i,j} \in J$. Note that this decomposition might not be unique, but this does not matter. Any such decomposition is fine. All we need is that $B + Z(A)$ is commutative and is closed under multiplication and J is an ideal, in particular, closed under multiplication with arbitrary elements.

We write

$$\begin{aligned}
\det M &= \sum_{\sigma \in S_n} \operatorname{sgn}(\sigma)(b_{1,\sigma(1)} + r_{1,\sigma(1)}) \cdots (b_{n,\sigma(n)} + r_{1,\sigma(n)}) \\
&= \sum_{\sigma \in S_n} \operatorname{sgn}(\sigma) \sum_{I \subseteq \{1,\ldots,n\}} t^I_{1,\sigma(1)} \cdots t^I_{n,\sigma(n)} \\
&= \sum_{I \subseteq \{1,\ldots,n\}} \sum_{\sigma \in S_n} \operatorname{sgn}(\sigma) t^I_{1,\sigma(1)} \cdots t^I_{n,\sigma(n)} \\
&= \sum_{I \subseteq \{1,\ldots,n\}} \det(T^I)
\end{aligned}$$

where

$$t^I_{i,j} = \begin{cases} r_{i,j} & \text{if } i \in I, \\ b_{i,j} & \text{if } i \notin I \end{cases}$$

and

$$T^I = (t^I_{i,j}).$$

Since the nilpotency index of J is r, we can use $\det T^I = 0$ if $|I| \geq r$. Therefore,

$$\det M = \sum_{|I| < r} \det(T^I).$$

We can now use the algorithm of Chien et al. to compute each $\det T^I$. The algorithm of Chien et al. uses essentially the same decomposition, but it first decomposes the entries of M as a sum of an element from B and from $\operatorname{rad} A$. But what they only need is that the former is a commutative subalgebra and the latter is a nilpotent ideal. This is also true in our case: $B + Z(A)$ is a commutative subalgebra and J is a nilpotent ideal. (If J is not nilpotent, we can replace it by a strictly smaller ideal that is nilpotent by the definition of complete ideal of noncommuting elements.) Therefore, we can simply use the algorithm by Chien et al. to compute each $\det T^I$ in time $n^{O(r)}$. Since there are $n^{O(r)}$ such determinants, we get the following theorem.

Theorem 7 *Over perfect fields, there is an algorithm that given a finite dimensional algebra A that has a complete ideal of noncommuting elements with nilpotency index $\leq r$ and a matrix $M \in A^{n \times n}$, computes the determinant of M over A in time $n^{O(r)}$, provided that* $\dim A = \operatorname{poly}(n)$.

If the dimension of A is larger, then we have to multiply the running time by the time needed to multiply elements in A.

6 Conclusions

It is an interesting question whether the smallest ideal J can be much larger than $\mathrm{com}(A)$ and how much their nilpotency indices can differ. There seems to be no general answer, mainly because there is no analogue of Wedderburn's theorem for the radical. For the algebra of upper triangular matrices, we have $J = \mathrm{com}(A) = \mathrm{rad}(A)$. For the free noncommutative algebra $k\langle x, y, z\rangle$ modulo the ideal of all monomials of degree d and the relations that make x commute with y and z, we have $\mathrm{rad}(A) \supsetneq J \supsetneq \mathrm{com}(A)$ for any J. More precisely, $\mathrm{rad}\, A$ is generated by x, y, and z, J is generated by y and z, and $\mathrm{com}(A)$ is generated by $yz - zy$. In our upper bound, we can take the minimum over all complete ideals J of noncommuting elements. Is there an easy characterisation of the best J?

Acknowledgement

I would like to thank Prahladh Harsha for drawing my attention to this problem.

References

[1] V. Arvind and S. Srinivasan. On the hardness of the noncommutative determinant. In *Proc. 42nd ACM Symp. on Theory of Comput. (STOC)*, pages 677-686, 2010.

[2] A. Barvinok. Polynomial time algorithms to approximate permanents and mixed discriminants within a simply exponential factor. *Random Struct. Algorithms*, 14(1):29–61, 1999.

[3] M. Bläser and H. Dell. Complexity of the cover polynomial. In *Proc. 34th Int. EATCS Coll. on Automata, Languages, and Programming (ICALP)*, LNCS 4596, 801–812, 2007.

[4] S. Chien, P. Harsha, A. Sinclair, and S. Srinivasan. Almost settling the hardness of noncommutative determinant. In *Proc. 43rd ACM Symp. on Theory of Comput. (STOC)*, pages 499–508, 2011.

[5] S. Chien and A. Sinclair. Algebras with polynomial identities and computing the determinant. *J. Comput. Sys. Sci.* 67(2):263–290, 2003.

[6] S. Chien, L. Rasmussen, and A. Sinclair. Clifford algebras and approximating the permanent. *J. Comput. Sys. Sci.* 67(2):263-290, 2003.

[7] H. Dell, T. Husfeldt, and M. Wáhlen. Exponential time complexity of the permanent and the Tutte polynomial. In *Proc. 37th Int. EATCS Coll. on Automata, languages, and programming (ICALP)*, LNCS 6198, 426–437, 2010.

[8] Y.A. Drozd and V.V. Kirichenko. *Finite dimensional algebras*, Springer, 1994.

[9] C.D. Godsil and I. Gutman. On the matching polynomial of a graph. In: L. Lovász and V.T. Sós, eds., *Algebraic Methods in Graph Theory, Vol. 1*, pages 241–249, North-Holland, 1981.

[10] M, Jerrum, A. Sinclair, and E. Vigoda. A polynomial-time approximation algorithm for the permanent of a matrix with nonnegative entries. *J. ACM* 51(4):671–697, 2004.

[11] N. Kamarkar, R.M. Karp, R.J. Lipton, L. Lovász, and M. Luby. A Monte-Carlo algorithm for estimating the permanent. *SIAM J. Comput.* 22(2):284–293, 1993.

[12] M. Mahajan and V. Vinay. Determinant: Old algorithms and new insights. *SIAM J. Discrete Math.* 12(4):474–490, 1999.

[13] C. Moore and A. Russell. Approximating the permanent via nonabelian determinants, 2009. arXiv:0906.1702.

[14] N. Nisan. Lower bounds for noncommutative computation. In *Proc. 23rd ACM Symp. on Theory of Comput. (STOC)*, pages 410–418, 1991.

[15] L.G. Valiant. The complexity of computing the permanent. *Theoret. Comput. Sci.* 8:189–201, 1979.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de