Determinant versus Permanent: salvation via
generalization?
The algebraic complexity of the Fermionant and the
Immanant

N. de Rugy-Altherre

Abstract The fermionant Ferm$_k^{n}(\bar{x}) = \sum_{\sigma \in S_n} (-k)^{c(\pi)} \prod_{i=1}^{n} x_{i,j}$ can be seen as a generalization of both the permanent (for $k = -1$) and the determinant (for $k = 1$). We demonstrate that it is VNP-complete for any rational $k \neq 1$. Furthermore it is $\#P$-complete for the same values of $k$. The immanant is also a generalization of the permanent (for a Young diagram with a single line) and of the determinant (when the Young diagram is a column). We demonstrate that the immanant of any family of Young diagrams with bounded width and at least $n^\epsilon$ boxes at the right of the first column is VNP-complete.

Keywords Algebraic complexity, Immanant, Fermionant, Counting complexity

1 Introduction

In algebraic complexity (more specifically Valiant’s model\cite{Valiant}) one of the main question is to know whether $VP = VNP$ or not. Answering this is considered to be a very good step towards the resolution of $P = NP$. This question is very close to the problem $\text{per vs. det}$, where we ask if the permanent can be computed in polynomial time (in the size of the input - a matrix), as is the determinant.

The main idea of this paper is to find a generalization of both the permanent and the determinant in order to study exactly where the difference between them lies. A generalization is here understood as a parameter, let us say $t$, and a function $f(t, \bar{x})$ such that for example $f(0, \bar{x}) = \det(\bar{x})$ and $f(1, \bar{x}) = \text{per}(\bar{x})$. If we have a complete classification of the complexity of $f(t, \bar{x})$ for any $t$ (with $t$ fixed), we should be able to see where we step from
VP to VNP and maybe understand a little bit more why the permanent is hard and not the determinant.

Here we will study two different generalizations. First the fermionant, secondly the immanant. The fermionant was introduced by Chandrasekharan and Wiese [3] in 2011 in a context of quantum physics. It is defined with a real parameter $k$ such that for $k = 1$ it is the determinant and for $k = -1$ it is the permanent. Mertens and Moore [7] have demonstrated its hardness for $k \geq 3$, in the framework of counting complexity.

Likewise, but in a different framework and with a complete different proof, we demonstrate the hardness of the fermionant seen as a polynomial for any rational $k \neq 1$ (and of course for $k \neq 0$). This give a interesting point of view on where the hardness of the permanent lies. We also get a bonus: we use a technique developed by Valiant to demonstrate the hardness of the fermionant in the counting complexity framework for $k \neq 1$. We thus extend the results of Mertens and Moore [7]; in particular to the case $k = 2$, which is, from what I understand, the most interesting case for physicist.

The second generalization is more classical and comes from the field of group representation. It is the immanant, introduced by Littlewood [6] in 1940. Immanants are families of polynomials indexed by Young diagrams. If the Young diagrams are a single column with $n$ boxes, the immanant is the determinant. At the opposite end, if it is a single line of $n$ boxes, the immanant is the permanent. The main question is: for which Young diagrams do we step from VP to VNP?

We know that if there are only a finite number of boxes on the right of the first column, the immanant is still in VP (cf [2]). On the other hand, a few hardness results have been found, fundamentally for Young diagrams in which the permanent is hidden. For example, the hook (a line of $n$ boxes and a column of any number of boxes) and the rectangle (any number of lines each with $n$ boxes) are hard (cf [2]), or more generally if the maximal difference between the size of two consecutive lines is as big as a power of $n$ (cf [1]).

Here we will demonstrate that for Young diagrams with only two columns, each with $n$ boxes, the immanant is hard, which was an open question (cf [2] Problem 7.1). As each line of these Young diagrams has length no more than two, the permanent is not hidden in there. More generally for any family of Young diagrams with a bounded number of columns and with at least $n^\epsilon$ boxes at the right of the first column, the immanant is hard.

For a complete classification of the immanant in algebraic complexity, one 'just' has to determine the complexity of the ziggurat: the Young diagrams where the first line has $n$ boxes, the second $n - 1$, the third $n - 2$ etc. and the last 1 box. This immanant is most probably also hard. The complexity of the immanant with a logarithmic number of boxes at the right of the first column is also unknown.
2 Definitions

We work within Valiant’s algebraic framework. Here is a brief introduction to this complexity theory. For a more complete overview, see [2].

An arithmetic circuit over \( \mathbb{Q} \) is a labeled directed acyclic connected graph with vertices of indegree 0 or 2 and only one sink. The vertices with indegree 0 are called input gates and are labeled with variables or constants from \( \mathbb{Q} \). The vertices with indegree 2 are called computation gates and are labeled with \( \times \) or +. The sink of the circuit is called the output gate.

The polynomial computed by a gate of an arithmetic circuit is defined by induction: an input gate computes its label; a computation gate computes the product or the sum of its children’s values. The polynomial computed by an arithmetic circuit is the polynomial computed by the sink of the circuit.

A p-family is a sequence \((f_n)\) of polynomials such that the number of variables as well as the degree of \( f_n \) is polynomially bounded in \( n \). The complexity \( L(f_n) \) of a polynomial \( f_n \in \mathbb{Q}[x_1, \ldots, x_n] \) is the minimal number of computational gates of an arithmetic circuit computing \( f_n \) from variables \( x_1, \ldots, x_n \) and constants in \( \mathbb{Q} \).

Two of the main classes in this theory are: the analog of \( P \), \( VP \), which contains of every p-family \((f_n)\) such that \( L(f_n) \) is a function polynomially bounded in \( n \); and the analog of \( NP \), \( VNP \). A p-family \((f_n)\) is in \( VNP \) iff there exists a VP family \((g_n)\) such that for all \( n \),

\[
f_n(x_1, \ldots, x_n) = \sum_{\epsilon \in \{0,1\}^n} g_n(x_1, \ldots, x_n, \epsilon_1, \ldots, \epsilon_n)
\]

In this theory the most common notion of reduction is the projection: \( f \) is a projection of \( g \) is one can compute \( f \) just by evaluating \( g \) on certain variables and constants. But here, we will need an other notion of reduction, the c-reduction: the oracle complexity \( L^g(f) \) of a polynomial \( f \) with an oracle access to \( g \) is the minimum number of computation gates and evaluations of \( g \) over previously computed values that are sufficient to compute \( f \) from the variables \( x_1, \ldots, x_n \) and constants from \( \mathbb{Q} \). A p-family \((f_n)\) c-reduces to \((g_n)\) if there exists a polynomially bounded function \( p \) such that \( L^{g^p} (f_n) \) is a polynomially bounded function.

\( VNP \) is closed under c-reductions (See [8] for an idea of the proof). However this reduction does not distinguish the lower classes. For example, 0 is VP-complete for c-reductions. In this paper we will demonstrate hardness results, a smallest notion of reduction (as projection) is thus not needed.

The permanent is \( VNP \)-complete for c-reductions (2). The determinant is in \( VP \) (but a priori not \( VP \)-complete, finding a \( VP \)-complete polynomial is still a open question).
3 The fermionant

3.1 In algebraic complexity

Let $A$ be an $n \times n$ matrix. The fermionant of $A$, with parameter $k$ is defined as

$$\text{Ferm}_k^A = \sum_{\pi \in S_n} (-k)^{c(\pi)} \prod_{i=1}^{n} A_{i, \pi(i)}$$

where $S_n$ denotes the symmetric group of $n$ objects and, for any permutation $\pi \in S_n$, $c(\pi)$ denotes the number of cycles of $\pi$. To study the complexity of such a function, we work within the algebraic complexity framework. The algebraic equivalent of the fermionant is the polynomial obtain when we compute the fermionant on the matrix $(x_{i,j})_{1 \leq i, j \leq n}$. If we write $\text{Ferm}_k$ the p-family $(\text{Ferm}_k^n)_{n \in \mathbb{N}}$, we have a complete classification of the algebraic complexity of those polynomials.

**Theorem 1** Let $k$ be a rational.

- $\text{Ferm}_0^A = 0$.
- $\text{Ferm}_1^A$ is in VP
- for other values of $k$ $\text{Ferm}_k^A$ is VNP-complete for c-reductions.

Similarly to the permanent we can see the fermionant as a computation on a graph $G$ with $n$ vertices and where the edge between the vertices $i$ and $j$ is labeled with the variable $x_{i,j}$. A permutation $\pi \in S_n$ can be seen as a cycle cover on this graph. A cycle cover of $G$ is a subset of its edges that covers all vertices of $G$ and that form cycles. We write $\text{CC}(G)$ is the set of all cycle covers of $G$. The weight of such a cycle cover $\pi$ is $\omega(\pi) = \prod_{e \in \pi} x_e$ and if we write $c(\pi)$ its number of cycles then

$$\text{Ferm}_k^A(\bar{x}) = \sum_{\pi \in \text{CC}(G)} (-k)^{c(\pi)} \prod_{e \in \pi} x_e$$

To demonstrate the completeness of the permanent, Burgisser [2] has introduced two gadgets: the iff gadget and the Rosette. He’s iff-gadget is sketched on fig. 1. If you place it between the edges $e_1$ and $e_2$, it cancel in the calculation of the permanent every cycle cover that pass through $e_1$ but not $e_2$ (or vis versa).

**Fig. 1** Burgisser’s iff-gadget
Fig. 1 is Burgisser’s iff gadget. It adds three vertices $p_1, p_2, p_3$ and connect these vertices according to the following matrix:

$$\begin{pmatrix}
-1 & 1 \\ 1 & 1 \\ 1 & -1
\end{pmatrix}$$

We wish to get a similar result for the fermionant; however this gadget change the number of cycles of the cycle covers and thus the value of the fermionant. We have to introduce a new iff-gadget. The idea is to compensate the new cycles by some weight $-\frac{1}{k}$.

Fig. 2 Fermionant’s iff-gadget

$p_1, p_2, p_3$ but we connect them with the following matrix:

$$\begin{pmatrix}
\frac{1}{k} & 1 & \frac{1}{2} \\ 1 & -\frac{1}{k} & -\frac{1}{2} \\ 1 & 1 & \frac{1}{2k}
\end{pmatrix}$$

Let us begin by studying our iff-gadget. This will be done in three steps: first we add one iff-gadget to the graph (lemma [1]). Secondly we add several
gadgets such that no two gadgets are placed on a common edge (lemma 2). The gadget still works if we add several on them, even on a same edge. We will not demonstrate that fact in general, just in the special case that interest us (lemma 3).

**Lemma 1** Let $G$ be a graph, $\pi$ a cycle cover of this graph. We define its weight as $\omega(\pi) = \prod_{e \in \pi} x_e$; and we write $c(\pi)$ for the number of cycles in $\pi$. Let $e, e'$ be two edges of $G$. Let $G'$ be the same graph where we place an iff-gadget between these two edges. Let $\Pi(\pi)$ be the set of every cycle cover $\pi'$ of $G'$ which is equal to $\pi$ on $E(G)$. Then

- if in $G$, $e$ and $e'$ are in $\pi$ then
  \[\sum_{\pi' \in \Pi(\pi)} (-k)^{c(\pi')} \omega(\pi') = (-k)^{c(\pi)} \omega(\pi)\]

- If $e$ is in $\pi$ and not $e'$, or vis versa,
  \[\sum_{\pi' \in \Pi(\pi)} (-k)^{c(\pi')} \omega(\pi') = 0\]

- Finally if neither $e$ nor $e'$ are in $\pi$ then
  \[\sum_{\pi' \in \Pi(\pi)} (-k)^{c(\pi')} \omega(\pi') = \left(\frac{1}{2}(1 - k)\right) (-k)^{c(\pi)} \omega(\pi)\]

**Proof** Let $e = (u, v)$ and $e' = (u', v')$.

- if $e$ and $e'$ are in $\pi$, there is only one cycle covers that cover the gadget with the vertices $c_-, c_+, e, c_+$ and then
  \[\sum_{\pi' \in \Pi(\pi)} (-k)^{c(\pi')} \omega(\pi') = (-k)^{c(\pi)} \omega(\pi)\]
- if $e$ is in $\pi$ but not $e'$, the two possible cycle covers are in blue and red. The red one has as many cycle as $\pi$ and for weight $\frac{1}{2}\omega(\pi)$. The blue one has one more cycle than $\pi$ and for weight $\frac{1}{2k}\omega(\pi)$. Thus

$$
\sum_{\pi' \in \Pi(\pi)} (-k)^{c(\pi')}\omega(\pi') = \frac{1}{2}(-k)^{c(\pi)}\omega(\pi) + \frac{1}{2k}(-k)^{c(\pi)+1}\omega(\pi) = 0
$$

**Case $e \in \pi$ and $e' \notin \pi$**

- if $e'$ is in $\pi$ but not $e$, the two possible cycle covers are in blue and red. The red one has one more cycle than $\pi$ and for weight $\frac{1}{2k}\omega(\pi)$. The blue one has two more cycles than $\pi$ and for weight $-\frac{1}{2k}\omega(\pi)$. Thus

$$
\sum_{\pi' \in \Pi(\pi)} (-k)^{c(\pi')}\omega(\pi') = \frac{1}{-k}(-k)^{c(\pi)+1}\omega(\pi) - \frac{1}{k^2}(-k)^{c(\pi)+2}\omega(\pi) = 0
$$

**Case $e' \in \pi$ and $e \notin \pi$**
– if neither $e$ nor $e'$ are in $\pi$, then the six possible cycle covers are listed below. Thus,

$$\sum_{\pi' \in \Pi(\pi)} (-k)^{c(\pi')} \omega(\pi') = \left( -\frac{1}{2k^2}(-k)^2 - \frac{1}{2k}(-k)^2 - \frac{1}{2k^3}(-k)^3 \right)$$

$$= \left( \frac{1}{2} - \frac{1}{2}k - \frac{1}{2}k + \frac{1}{2}k + \frac{1}{2}k + \frac{1}{2}k \right) (-k)^{c(\pi)} \omega(\pi)$$

$$= \frac{1}{2} (1 - k) (-k)^{c(\pi)} \omega(\pi)$$

Case $e$ and $e'$ not in $\pi$

This end the demonstration. Notices that if $k = -1$ we have Burgisser’s iff gadget. If both $e$ and $e'$ are loops (i.e. $u' = v'$ and $u = v$), the gadget is represented in the following scheme and I let the reader check that it works the same way.

Loop case
We add now several iff-gadget simultaneously in a graph.

**Lemma 2** Let $G$ be a graph with $n$ vertices and $(e'_1, e'_2)_{1 \leq i \leq 1}$ be a set of pairs of edges of $G$ such that no two edges in this set are equal. Let $G'$ be the same graph but where we place an iff-gadget between every pair $(e'_1, e'_2)$. Let $\pi$ be a cycle cover of $G$, $\Pi(\pi)$ be the set of cycle covers of $G'$ that match $\pi$ on $E(G)$.

- If there is a pair $(e'_1, e'_2)$ of edges such that $e'_1 \in \pi$ and $e'_2 \notin \pi$, or vice versa, then
  \[ \sum_{\pi' \in \Pi(\pi)} (-k)^{c(\pi')} \omega(\pi') = 0 \]

- Else, let $d(\pi)$ be the number of pair $(e'_1, e'_2)$ of edges such that $e'_1 \notin \pi$ and $e'_2 \notin \pi$. Then
  \[ \sum_{\pi' \in \Pi(\pi)} (-k)^{c(\pi')} \omega(\pi') = \left( \frac{1}{2} (1 - k) \right)^d(\pi) (-k)^{c(\pi)} \omega(\pi) \]

**Proof** It is simply an induction on $l$, the number of gadgets we add. If $l = 1$, i.e., we add only one iff-gadget, then this case has been dealt with in Lemma 1. Let us suppose the lemma true for $l - 1$ iff-gadgets. Let $\pi$ be a cycle cover of $G$.

- If we are in the first case, then let $(e'_1, e'_2)$ be a pair of edges such that $e'_1 \in \pi$ and $e'_2 \notin \pi$. Let $G'$ be the graph with all the same iff-gadgets than $G''$ but the one between $e'_1$ and $e'_2$. By induction the sum of every cycle covers of $G''$ that match $\pi$ on $E(G')$ is either 0 or $\left( \frac{1}{2} (1 - k) \right)^{d(\pi)} (-k)^{c(\pi)} \omega(\pi)$. Let $\pi'$ one of those cycle covers. Remarque that $e'_1 \in \pi'$ and $e'_2 \notin \pi'$.
  We add an iff-gadget between $e'_1$ and $e'_2$ in $G'$ and obtain $G''$. We can consider $G'$ as a graph with only one iff-gadget and apply lemma 1 with this graph and $\pi'$. Then the sum of every cycle covers that match $\pi'$ on $E(G'')$ is 0. Thus the result.
- If we are in the second case, i.e., every edges of the same pair are simultaneously in or out of $\pi$. Let $G_l$ be the same graph as $G'$ but with no iff-gadget between $e_1^l$ and $e_2^l$. By induction, if we write $H_l(\pi)$ the set of cycle cover of $G_l$ that match $\pi$ on $E(G)$ and $d_l(\pi)$ the number of pair that are not in $\pi$, but $(e_1^l, e_2^l)$,

$$\sum_{\pi' \in H_l(\pi)} (-k)^{c(\pi')} \omega(\pi') = \left(\frac{1}{2}(1-k)\right)^{d(\pi)} (-k)^c(\pi) \omega(\pi)$$

We now see $G'$ as a graph with only one iff-gadget, the one between $e_1^l$ and $e_2^l$. Let $\pi' \in H_l(\pi)$ and $A$ be the set of cycle covers of $G'$ that match $\pi'$ on $E(G')$ minus every edges of this iff-gadget. If $e_1^l$ and $e_2^l$ are in $\pi$, then they are in $\pi'$, $d(\pi) = d_l(\pi)$ and

$$\sum_{\lambda \in A(\pi')} (-k)^{c(\lambda)} \omega(\lambda) = (-k)^c(\pi') \omega(\pi')$$

Thus,

$$\sum_{\pi' \in H(\pi)} (-k)^{c(\pi')} \omega(\pi') = \sum_{\pi' \in H_l(\pi)} \sum_{\lambda \in A(\pi')} (-k)^{c(\lambda)} \omega(\lambda) = \sum_{\pi' \in H_l(\pi)} (-k)^{c(\pi')} \omega(\pi') = \left(\frac{1}{2}(1-k)\right)^{d(\pi)} (-k)^c(\pi) \omega(\pi)$$

If $e_1^l$ and $e_2^l$ are not in $\pi$, then they are not in $\pi'$ and

$$\sum_{\lambda \in A(\pi')} (-k)^{c(\lambda)} \omega(\lambda) = \left(\frac{1}{2}(1-k)\right) (-k)^c(\pi') \omega(\pi')$$

Thus,

$$\sum_{\pi' \in H(\pi)} (-k)^{c(\pi')} \omega(\pi') = \sum_{\pi' \in H_l(\pi)} \sum_{\lambda \in A(\pi')} (-k)^{c(\lambda)} \omega(\lambda) = \sum_{\pi' \in H_l(\pi)} \left(\frac{1}{2}(1-k)\right) (-k)^c(\pi') \omega(\pi') = \left(\frac{1}{2}(1-k)\right)^{d(\pi)+1} (-k)^c(\pi) \omega(\pi) = \left(\frac{1}{2}(1-k)\right)^{d(\pi)} (-k)^c(\pi') \omega(\pi)$$

Finally we add several gadgets, possibly more than one on the same edge. As we said before, we will not demonstrate it in the general case, just in the case we need.
Lemma 3 Let $G$ be a graph with $n$ vertices. We make $l$ copies of $G$ and name them $G_1, \ldots, G_l$. Let $\tilde{F}^l$ be the disjoint union of those copies in which we label the edges of $G_1$ with the same weight as those of $G$ and the edges of $G_i$ for $i \geq 2$ with 1. If $e$ is an edge of $G$, we call $e_i$ the corresponding edge in $G_i$. We name $F^l$ the graph $\tilde{F}^l$ where for any edge $e \in E(G)$ and any $1 \leq i \leq l$, we have placed an iff-gadget between $e_i$ and $e_{i+1}$. Let $\pi$ be a cycle cover of $G$ and $\Pi(\pi)$ be the set of cycle covers of $F^l$ that match $\pi$ on $E(G_1)$. Then

$$\sum_{\pi' \in \Pi(\pi)} (-k)^{c(\pi')} \omega(\pi') \equiv \left(\frac{1}{2} (1 - k)\right)^{|E(G)| - n} (-k)^{l \times c(\pi)} \omega(\pi)$$

Proof Here is an example of $F^3$, the iff-gadgets are represented by a red edge. In black we have $\tilde{F}^3$.

![Diagram](image-url)

The idea is, with the help of the iff-gadget, to copy a cycle cover from $G_1$ to every other copies of $G$, without changing the weight of this cycle cover, just multiplying the number of cycles. The demonstration is by induction on $l$.

If $l = 2$, then we simultaneously add $|E(G)|$ iff-gadgets, but only one on each edge. By design, a cycle cover $\pi$ on $G_1$ is repeated on $G_2$ (i.e., if $e_1$ is in $\pi$ then $e_2$ is also in $\pi$ as there is a iff-gadget between $e_1$ and $e_2$, see Lemma [2]). The edges of $G_2$ are labeled with 1 and therefore do not contribute to the weight of the cycle cover. The number of cycles of $\pi' \in \Pi(\pi)$ is twice the number of cycles of $\pi$. There is $|E(G)|$ iff-gadgets in $F^2$. A cycle cover of $G$ passes through $n$ edges and therefore activates exactly $n$ iff-gadgets. The $|E(G)| - n$ other iff-gadgets are not activated and thus each of them gave a contribution of $\frac{1}{2} (1 - k)$ to the sum.

Suppose the lemma true for $l - 1$ copies. Let $F^{l-1}$ be the disjoint union of $l - 1$ copies of $G$ with iff-gadgets. We add a new copy $G_l$ of $G$ linked to $F^{l-1}$ with iff-gadgets to obtain $F^l$. Let $\pi$ be a cycle cover of $G$, $\Pi(\pi)$ the set of every cycle covers of $F^l$ that match $\pi$ on $E(G_1)$ and $\Pi^{l-1}(\pi)$ the same but of $F^{l-1}$. By induction,

$$\sum_{\pi' \in \Pi^{l-1}(\pi)} (-k)^{c(\pi')} \omega(\pi') \equiv \left(\frac{1}{2} (1 - k)\right)^{|E(G)| - n} (-k)^{(l-1) \times c(\pi)} \omega(\pi)$$

Let $\tilde{F}^l$ be the disjoint union of $F^{l-1}$ and $G_l$. To obtain $F^l$ from this graph, one has just to add a iff-gadget between every edge $e_{l-1}$ and $e_l$. We can apply
then Lemma 2 to this graph. If \( \pi'' \) is a cycle cover of \( \hat{F}^l \) that match \( \pi \) on \( G_1 \), let \( A(\pi'') \) be the set of cycle covers of \( F^l \) that match \( \pi'' \) on \( E(\hat{F}^l) \). Then, if we call \( d(\pi'') \) the number of pairs \( (e_{l-1}, e_l) \) that are not in \( \pi'' \),

\[
\sum_{\lambda \in A(\pi'')} (-k)^{c(\lambda)} \omega(\lambda) = \left( \frac{1}{2} (1 - k) \right)^{d(\pi'')} (-k)^{c(\pi'')} \omega(\pi'')
\]

Let us study a little bit more \( \pi'' \). It is a cycle cover of two disjoint graphs, \( F^{l-1} \) and \( G_l \). Therefore it is composed of two sub cycle covers: \( \sigma \) a cycle cover of \( F^{l-1} \) which by induction is in a \( H^{l-1}(\pi) \) and a cycle cover \( \lambda \) of \( G_l \). However, as every edge of \( G_l \) is linked with an iff-gadget to its image in \( G_{l-1} \) in \( F^l \), the cycle cover \( \pi'' \) will contribute to the last sum if and only if it contain both \( e_{l-1} \) and \( e_l \), or neither \( e_{l-1} \) and \( e_l \). Thus, \( \lambda \) must be the copy of \( \pi \) in \( G_l \), which we write \( \lambda_x \) and \( c(\pi'') = c(\sigma) + c(\lambda_x) = c(\sigma) + c(\pi) \).

There are \( n \) edges in the last image \( G_l \) that are passed through by \( \pi'' \). Therefore, there are \( (|E(G)| - n) \) iff-gadgets that are not activated by \( \pi'' \) (i.e., \( d(\pi'') = |E(G)| - n \)). Thus,

\[
\sum_{\pi' \in H(\pi)} (-k)^{c(\pi')} \omega(\pi') = \sum_{\pi'' \in H'(\pi)} \sum_{\lambda \in A(\pi'')} (-k)^{c(\lambda)} \omega(\lambda)
\]

\[
= \sum_{\pi'' \in H'(\pi)} \left( \frac{1}{2} (1 - k) \right)^{|E(G)| - n} (-k)^{c(\pi'')} \omega(\pi'')
\]

\[
= \sum_{\sigma \in H^{l-1}(\pi)} \sum_{\lambda \in \mathsf{CC}(G_l)} \left( \frac{1}{2} (1 - k) \right)^{|E(G)| - n} (-k)^{c(\sigma) + c(\lambda)} \omega(\lambda) \omega(\sigma)
\]

\[
= \left( \frac{1}{2} (1 - k) \right)^{|E(G)| - n} (-k)^{c(\lambda_x)} \sum_{\sigma \in H^{l-1}(\pi)} (-k)^{c(\sigma)} \omega(\sigma)
\]

\[
= \left( \frac{1}{2} (1 - k) \right)^{|E(G)| - n \times (l - 1)} (-k)^{c(\pi)} \omega(\pi)
\]

Where \( H(\pi) \) is the set of cycle covers of \( F^l \) that match \( \pi \) on \( E(G) \); \( H'(\pi) \) the set of cycle covers of \( F^l \) that match \( \sigma \) on \( E(G) \) and for \( \pi'' \in H'(\pi) \), \( A(\pi'') \) the set of cycle covers that match \( \pi'' \) on \( E(\hat{F}^l) \). We have \( H(\pi) = \bigcup_{\pi'' \in H'(\pi)} A(\pi'') \) which completes our demonstration.

Now we have all the tools necessary to demonstrate our theorem. I recall its statement:

**Theorem 2** (Theorem 1) Let \( k \) be a rational.

- \( V_{\mathsf{FP}}^0 = 0 \).
- \( V_{\mathsf{FP}}^1 \) is in \( \mathsf{VP} \).
- for other values of \( k \) \( V_{\mathsf{FP}}^k \) is \( \mathsf{VNP} \)-complete for \( c \)-reductions.
Proof (Proof of theorem 7) The case where $k = 0$ is trivial. When $k = 1$ $Ferm^k_n(\bar{x}) = det_n(\bar{x})$ and it is a well known result that the determinant can be computed with an arithmetic circuit of polynomial size. Now, let $k$ be a rational different than 0 and 1.

Let us write $P_l(G)$ the graph obtained in the previous lemma, when we duplicate $l$ times $G$ and add iff-gadgets to repeat every cycle cover $l$ times. We have seen that

$$Ferm^k_n(P_l(G))(\bar{x}) = \sum_{\pi \in CC(G)} (-k)^{l \times c(\pi)} \prod_{e \in \pi} \omega(e) \times \left( \frac{1}{2} (1 - k) \right)^{(|\pi| - n)}$$

Let us write $c_m = \sum_{\pi \in CC(G) : c(\pi) = m} \prod_{e \in \pi} \omega(e)$, $\alpha = (\frac{1}{2} (1 - k))^{|E(G)| - n}$, $f_l = Ferm^k_n(P_l(G))$ and $\omega_l = (-k)^l$, then

$$f_l = \alpha^{l-1} \sum_{k=1}^{n} c_k \omega^m_l$$

Ergo

$$\begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{pmatrix} = \begin{pmatrix} \alpha & 0 & \ldots & 0 & \omega_1 & \omega^2_1 & \ldots & \omega^n_1 \\ 0 & \alpha^2 & \ldots & 0 & \omega_2 & \omega^2_2 & \ldots & \omega^n_2 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \alpha^n & \omega_n & \omega^2_n & \ldots & \omega^n_n \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix}$$

This system of equation is a Vandermonde system and therefore is invertible (if $k \neq 1$ and $k \neq -1$, because in these cases, some $\omega_i$ are equal and the matrix is not invertible): there exists some rationals $\omega^*_l,m$, such that for any $m$,

$$c_m = \sum_{l=1}^{n} \omega^*_l,m f_l(\bar{x})$$

Therefore, for any $m$, we have a $c$-reduction from $c_m$ to the fermionant, $(c_m) \leq_c Ferm^k$. But, $c_1 := \sum_{\pi \in S_n: c(\pi) = 1} \prod_{i=1}^{n} x_i, \pi(i) = Ham_n(\bar{x})$, where $Ham_n$ is the Hamiltonian, which is known to be $\VNP$-complete ([2], Corollary 3.19).

3.2 In counting complexity

We have seen that the fermionant can be expressed as a linear combination of polynomial size of the Hamiltonian. From that we have concluded that the fermionant is $\VNP$-complete. However, the Hamiltonian is also $\#P$-complete, when considered as a counting problem (when we see the $x_i$ as number and not as variables). This gives us a Turing reduction from the Hamiltonian to the fermionant and thus it is also $\#P$-complete.

However, our Turing reductions requires rationals ($\frac{1}{2}, -\frac{1}{2}$, etc) and thus it is $\#P$-complete only on rational inputs. We can adapt the proof of Valiant for
the $\#P$-completeness of the permanent \cite{9} to replace those rationals by some gadgets only using 0 and 1 and obtain a more general completeness, over 0, 1 matrices.

Mertens and Moore \cite{7} have an even better completeness: for 0, 1 and planar matrices (A priori our construction of $P_l G$ in lemma 3 does not preserve planarity). We just extend their result to $k = 2$.

**Theorem 3** For every $k \neq 1$ and $k \neq 0$, $\text{Ferm}_k^k$ is $\#P$-complete for matrices over $\{0, 1\}$.

**Proof** The proof here is similar as Valiant’s \cite{9}, but with every gadgets adapt to work with the fermionant, i.e., each gadget is design to compensate the number of cycles it add, or to add a constant number of cycles.

Let $k$ be an integer neither null nor equal to 1. We have demonstrated in Theorem 1 that there exists rationals $w_i^*$ such that for every rational matrix $A$:

$$\text{Ham}(A) = \sum_{i=1}^{n} w_i^* \text{Ferm}_k^k(P_iA)$$

$P_i$ is the transformation that duplicate the underlying graph $i$ time and add iff-gadgets. The only non-integer rationals it use are $\frac{1}{k}$, $\frac{1}{2}$, $\frac{1}{2k}$ and their opposites, if there are no non-integer rationals in $A$. To get rid of those, we just have to multiply every edge weight of $P_iA$ by $2^k$. Let us write $P_i' A$ the new weight matrix. Then, if we call $n_i$ the size (i.e., the number of rows) of $P_iA$ (and of $P_i' A$):

$$\text{Ham}(A) = \sum_{i=1}^{n} w_i^* \left( \frac{1}{2k} \right)^{n_i} \text{Ferm}_k^k(P_i' A)$$

From this new equation, we can say that the fermionant is $\#P$-complete for integer matrices. For the rest of the demonstration, we will suppose that $A$ is a $\{0, 1\}$-matrix. The idea of Valiant to go from integers to natural numbers, i.e., to get ride of the $-1$, is to compute everything modulo a number $\Lambda$ greater than anything else. And switch every $-1$ into $\Lambda + 1$.

The biggest integer in $P_i' A$ is $2k$. Then, if we call $n_i$ the size of $P_i(A)$,

$$\text{Ferm}_k^k(P_i' A) = \sum_{\pi \in S_{n_i}} (-k)^{c(\pi)} \prod_{j=1}^{n_i} (P_i' A)_{j, \pi(j)}$$

$$\leq \sum_{\pi \in S_{n_i}} k^{n_i} \prod_{j=1}^{2k} 2k$$

$$\leq n_i! k^{n_i} (2k)^{n_i}$$

To make sure that $\Lambda$ is big enough, we take $A > \sum_{i=1}^{n} |w_i^*| n_i! (2k)^{2n_i}$. Let us switch in $P_i' A$ every negative integer $-m$ by $(\Lambda + 1) \times m$ and let us call $P''_i A$ the result. It is a matrix of natural numbers and
\[ \text{Ham}(A) \equiv \sum_{i=1}^{n} w^*_i \left( \frac{1}{2k} \right)^{n_i} \text{Ferm}^k(P''_i A) [A] \]

This equation has of course no meaning, as \( w^*_i \left( \frac{1}{2k} \right)^{n_i} \) is a rational and cannot be computed modulo \( \Lambda \). Let \( w^*_i = \frac{p_i}{q_i} \), where \( p_i \) and \( q_i \) are integers and \( \omega = \prod_{i=1}^{n} q_i \). Thus \( \tilde{w}^*_i := \omega \times w^*_i \) is an integer. Let also \( m_i = n_n - n_i \), where \( n_i \) is the size of \( P''_i A \). The right equation is then

\[ \text{Ham}(A) \times (2k)^n \times \omega \equiv \sum_{i=1}^{n} \tilde{w}^*_i (2k)^{m_i} \text{Ferm}^k(P''_i A) [A] \]

The next step is to replace every edge labeled by a natural number \( a \) by a gadget with only edges labeled by 1. The idea of Valiant is, for example if \( a = 20 \) to write it in base 2, \( 20 = 2^2 + 2^4 \) and to replace the edge \((u, v)\) of weight 20 by two double paths, one of length 2 and the other of length 4. (cf fig. 4)

\[ \text{Valiant's gadget} \]

\[ \text{Fermionant's gadget} \]

As usual we modify this gadget so it can work for the fermionant. In Fig. 5 we give an example of if for an edge \( e \) between \( u \) and \( v \) labeled with \( a = 20 = 2^2 + 2^4 \). A cycle cover \( \pi \) that pass thought \( e \) in the graph now can go in 16 different ways thought \( p^1_1, p^4_2 \) and \( p^1_3 \) and in 4 different ways trough \( p^2_1 \). We want that the number of cycles this gadget add to be the same, whatever path \( \pi \) took. The loop-gadgets (centered in \( q'_i \)) are here for that.

\[ \text{Fermionant's gadget} \]
Now instead of an edge that counts in a cycle cover for a weight $a$ we have $a$ different cycle covers that pass through this edge. However, this gadget needs \{0, 1, 2\} matrices to represent the double edges. We have to replace in this gadget the double edges by a diamond-gadget:

**Diamond gadget**

The idea of this diamond gadget is obvious. Here also the loop-gadgets are place to keep constant the number of cycles we add.

Let $e$ be an edge between two vertices $u$ and $v$ labeled by a non negative integer $a$ in a graph $G$. In base 2, $a$ is written

$$a = \sum_{i=1}^{\lfloor a \rfloor} a_i 2^i$$

We delete the edge $e$. For every $i \leq \lfloor a \rfloor$ such that $a_i \neq 0$, we add $i - 1$ new vertices $p_1^i, \ldots, p_{i-1}^i$ and we connect $p_j^i$ to $p_{j+1}^i$ with a diamond gadget. The first of this new vertices is connected similarly with $u$ and the last with $v$. Finally we add on every new vertex $p_j^i$ a new looped vertex $q_j^i$ linked by a cycle to $p_j^i$ (i.e., we add on every new vertix a loop-gadget).
Let $G'$ the graph where $e$ has been replaced by the gadget $M(a)$. If $\pi$ is a cycle cover of $G$, let us write $\tilde{\omega}(\pi)$ the sum of every cycle cover $\pi'$ in $G'$ that match $\pi$ in $G' - \{e\}$.

If $e \in \pi$ then $\omega(\pi)(-k)^{|a|+2(|a|+1)} = \tilde{\omega}(\pi)$, where $|a|$ is the number of bits of $a$. Indeed, the weight is the same but we add with our gadget $|a| + 1$ diamond gadgets and $|a|$ loop gadgets. Therefore, we add $2(|a| + 1) + |a|$ new cycles. Furthermore, because of the loop-gadget we add, even if $e \notin \pi$ we still have $\omega(\pi)(-k)^{|a|+2(|a|+1)} = \tilde{\omega}(\pi)$. If both cases the final result is multiplied by $(-k)^{|a|+2(|a|+1)}$.

If we switch every labeled edge in $P''_i A$ by a similar gadget, we multiply the result by a power of $(-k)$. However, this power does not depend on $\pi$, only on the number $n$ of edges of $G$ (the number of 1 of $A$).

We write $P'''_i A$ the $\{0,1\}$-matrix obtained from $P''_i A$ by replacing every edge labeled with $a \neq 0,1$ by the gadget $M(a)$. The $M(a)$'s gadgets are only add on iff-gadgets (as $A$ is a $\{0,1\}$ matrix). A motivated reader could count the number of $M(a)$'s gadgets we put and know the number $(-k)^{\gamma_l}$ we add to the computation.

$$\text{Ham}(A) \times (2k)^n \times \frac{1}{\omega} \equiv \sum_{i=1}^n \tilde{w}_i (2k)^{m_i} (\frac{1}{-k})^{\gamma_i} \text{Ferm}_k (P'''_i A) [A]$$

As before, this equation has no meaning, as we try to compute rationals modulo $\Lambda$. But, if we write $\gamma = \max \gamma_i$,

$$\text{Ham}(A) \times (2k)^n \times (-k)^{\gamma} \times \frac{1}{\omega} \equiv \sum_{i=1}^n \tilde{w}_i (2k)^{m_i} (-k)^{\gamma-\gamma_i} \text{Ferm}_k (P'''_i A) [A]$$

And therefore the $\text{Ferm}_k$ is $\#P$-complete for $\{0,1\}$ matrices. Note that $P'''_i A$ has a size polynomially bounded in the size of $A$.

4 Immanant with constant length

Immanants are defined with characters of representations of $S_n$. Such characters can be indexed by Young diagrams of $n$ boxes (i.e. collections of boxes
arranged in left-adjusted rows with a decreasing row length). As all the work of representation theory has already be done (Lemma 4), I will not define more those characters. We will only work on Young diagrams.

The immanant associated with a Young diagram $Y$ (and its associate character $\chi_Y$) is

$$\text{im}_Y(\bar{x}) = \sum_{\pi \in S_n} \chi_Y(\pi) \prod_{i=1}^{n} x_{\pi(i)}$$

For example, if the Young diagram is a single row of $n$ boxes, then for any $\sigma \in S_n$, $\chi_Y(\sigma) = 1$ and thus $\text{im}_Y = \text{per}$. At the opposite end, if $Y$ is a single column with $n$ boxes, $\chi_Y(\sigma) = \text{sg}(\sigma)$ and $\text{im}_Y = \text{det}$. For more details (and for a nice demonstration of the Murnaghan-Nakayama rule, one of the main part of our proof), see [5].

A classical theorem states that the irreducible characters of the symmetric group form a basis for the class functions on $S_n$. Class functions are real functions defined on $S_n$ and stable under conjugation (i.e. $\forall \pi, \sigma \in S_n, f(\pi \sigma \pi^{-1}) = f(\sigma)$). The function $\pi \mapsto (-k)^{\text{sg}(\pi)}$ is such a class function and thus is a linear combination of characters. Mertens and Moore [7] have computed these characters, and applied to the immanant we get:

**Lemma 4** For any integers $k$ and $n$, if we write $\Lambda^n_k$ for the set of every Young diagram with $n$ boxes and at most $k$ columns, then there exists some constants $d^n_k$ such that for any matrix $A$:

$$\text{Ferm}^k_n(A) = \sum_{Y \in \Lambda^n_k} d^n_k \text{im}_Y(A)$$

Intuitively this suggests that the family of every immanants of bounded width is VNP-complete. In algebraic complexity this is not that interesting, as this family is very large. But if we prove that with a certain family of immanant we can compute every immanants of width less than a certain $k$, then this family will be VNP-complete. It is exactly what we are going to do for the demonstration of the following proposition.

**Proposition 1** The $(\text{im}^n_{Y_n})_{n \in \mathbb{N}}$ is VNP-complete for $c$-reductions when $Y_n$ is the square Young diagram with two columns, each with $n$ rows.

**Proof** One of the main tool of our demonstration is the Murnaghan-Nakayama rule, which is a way to compute characters. A skew hook in a Young diagram is a connected collection of boxes in the border of the diagram such that if you remove this hook it is still a Young diagram (i.e. the row sizes are still decreasing).
**Proposition 2 (Murnaghan-Nakayama rule)** Let $Y$ be a Young diagram on $n$ boxes, let $l \leq n$ and $\sigma$ be a cycle of size $l$. Then for any $\pi \in S_n$,

$$\chi_Y(\pi.\sigma) = \sum_{\mu} (-1)^{r(\mu)} \chi_{Y-\mu}(\pi)$$

Where the sum is over every skew hook $\mu$ of $Y$ of size $l$ and $r(\mu)$ the number of vertical step of the skew hook.

**Young diagrams**

$$[4,4]$$

$$[4,2]$$

Now, let $[l_1, l_2]$ be the two columns Young diagram with $l_1$ boxes in the first column and $l_2$ in the second. We want to demonstrate that the family $(\im_{[n,n]})$ is VNP-complete.

The family of Young diagrams of width at most 2 and of $n$ boxes is $(l, [l, n - l]) \in [n/2, n]$. Each of them can be obtained from the square diagram $[l, l]$ by removing a skew hook of size $\delta = (l - (n - l)) = 2l - n$. Furthermore, if you remove a skew hook of size $\delta$ to $[l, l]$, you can obtain only $[l, n - l]$ and $[l - 1, n - l + 1]$. The Murnaghan-Nakayama rule implies that:

$$\im_{[l,l]}(\bar{x}, 2l - m) = (-1)^{l-1} \im_{[l,l]}(\bar{x}) + (-1)^{l} \im_{[l-1,n-l+1]}(\bar{x})$$

Where $2l - n$ is an encoding of a cycle of length $2l - n$. We know that, from Lemma 4:

$$\text{Ferm}_n^2(\bar{x}) = \sum_{Y \in A_n^2} d_Y^2 \im_Y(\bar{x}) = \sum_{l=n/2}^{n} d_{[l,n-l]}^2 \im_{[l,n-l]}(\bar{x})$$

From those two facts, we can compute the fermionant from the square immanant. We just have to take new constants: let $\alpha_{[n-1,1]} = d_{[n-1,1]}(-1)^n$ and for any $2 \leq l \leq n/2$, $\alpha_{[l-1,l]} = (-1)^{l}(d_{[l,n-l]} - \alpha_{[l+1,n-l-1]}(-a)^{l+1})$. For
simplicity, we write \( \alpha_l = \alpha_{[l,n-l]} \). If \( n \) is even

\[
\sum_{l=n/2}^{n-1} \alpha_l \text{im}_{[l,l]}(\bar{x},2l-n) = \sum_{l=n/2}^{n-1} \alpha_l \left( (-1)^{l-1} \text{im}_{[l,n-l]}(\bar{x}) + (-1)^l \text{im}_{[l-1,n-l+1]}(\bar{x}) \right)
\]

\[
= \sum_{l=n/2}^{n-1} \alpha_l(-1)^{l-1}\text{im}_{[l,n-l]}(\bar{x}) + \sum_{l=n/2}^{n-1} \alpha_l(-1)^l\text{im}_{[l-1,n-l+1]}(\bar{x})
\]

\[
= d_{[n-1,1]}\text{im}_{[n-1,1]}(\bar{x}) + \sum_{l=n/2}^{n-2} d_l \text{im}_{[l,n-l]}(\bar{x})
\]

\[
- \sum_{l=n/2}^{n-1} \alpha_{l+1}(-1)^{l+1}\text{im}_{[l,n-l]}(\bar{x}) + \sum_{l=n/2}^{n-1} \alpha_l(-1)^l\text{im}_{[l-1,n-l+1]}(\bar{x})
\]

\[
= \sum_{l=n/2}^{n-1} d_l \text{im}_{[l,n-l]}(\bar{x}) - \sum_{l=n/2+1}^{n-1} \alpha_l(-1)^l\text{im}_{[l-1,n-l+1]}(\bar{x})
\]

\[
+ \sum_{l=n/2}^{n-1} \alpha_l(-1)^l\text{im}_{[l-1,n-l+1]}(\bar{x})
\]

\[
= \sum_{l=n/2}^{n-1} d_l \text{im}_{[l,n-l]}(\bar{x}) + \alpha_{n/2+1}(-1)^{n/2}\text{im}_{[n/2,n/2]}(\bar{x})
\]

Furthermore, \( \text{im}_{[n,0]}(\bar{x}) = \det_n(\bar{x}) \) and then can be computed with only a polynomial number of arithmetic operations. Thus,

\[
\sum_{l=n/2}^{n-1} \alpha_l \text{im}_{[l,l]}(\bar{x},2l-n) = \sum_{l=n/2}^{n} \alpha_l (-1)^{l-\frac{n}{2}}\text{im}_{[l,l]}(\bar{x},2l-n) = \text{Ferm}^2_n(\bar{x})
\]

We obtain an arithmetic circuit of polynomial size that compute \( \text{Ferm}^2_n \) with \( n/2 \) oracles that can compute \( \text{im}_{[l,l]} \) for \( l \in [n/2,n] \). To obtain a \( c \)-reduction from the fermionant to the immant, we just have to notice that \( \text{im}_{[l,l]} \leq_p \text{im}_{[l',l']} \) as soon as \( l' \geq l \). Indeed, we just have to erase the first \( l' - l \)-th rows, which can be done by Corollary 3.2 of [1].

Now we have to demonstrate the same result for \( n \) odd. It is the same demonstration, but with technical modifications. Let \( n = 2m-1 \). Then
\[\sum_{l=m}^{n-1} \alpha_l \text{i}m_{[l,l]}(A, 2l - n) = \sum_{l=m}^{n-1} (-1)^{l-1} \alpha_l (\text{i}m_{[l,n-l]}(A) + (-1)^l \text{i}m_{[l-1,n-l+1]}(A)) = \sum_{l=m}^{n-1} \text{d} \text{i}m_{[l,n-l]}(A) - \sum_{l=m+1}^{n-1} \alpha_l (-1)^l \text{i}m_{[l-1,n-l+1]}(A) + \sum_{l=m}^{n-1} \alpha_l (-1)^l \text{i}m_{-[l-1,n-l+1]}(A) = \sum_{l=m}^{n-1} \text{d} \text{i}m_{[l,n-l]}(A) + \alpha_{m+1} (-1)^{m+1} \text{i}m_{[m,n-m]}(A)\]

The remaining immanant \(\text{i}m_{[m,n-m]} = \text{i}m_{[m,m-1]}\) is not a square immanant. But if you remove a skew hook of size 1 from \([m, m]\), you can only obtain \([m, m-1]\). Thus, \(\text{i}m_{[m,m-1]}(A) = \text{i}m_{m,m}(A, 1)\). Therefore, we also have a c-reduction from \(\text{Ferm}^2_n\) to \(\text{i}m_{[n,n]}\).

We can generalize this result to almost every family of bounded width.

**Proposition 3** Let \((Y_n)\) be a family of Young diagrams with two columns and at least \(\Omega(n^\epsilon)\) boxes in the last one, for \(\epsilon > 0\). Then \((\text{i}m_{Y_n})\) is \(\text{VNP-complete for c-reductions}\).

**Proof** For \(m \in \mathbb{N}\) let \(k^m_1\) be the size of the first column of \(Y_m\) and \(k^m_2\) the size of the second column. We will in this proof consider the case where the difference between these two sizes grows like \(n\) and the case where it is constant.

- If \((k^m_1 - k^m_2) > k^m_2\) (i.e. \(k^m_1 > 2k^m_2\)), then there is only one skew hook of size \(\delta := k^m_1 - k^m_2\) that can be removed from \([k^m_1, k^m_2]\) (in red in the picture).
  
  If we apply the Murnaghan-Nakayama Rule,
  \[\text{i}m_{[k^m_1, k^m_2]}(\bar{x}, \sigma_\delta) = (-1)^{\delta-1} \text{i}m_{[k^m_1, k^m_2]}(\bar{x})\]

  Where \(\sigma_\delta\) is a cycle of size \(\delta\). Thus, \((\text{i}m_{[k^m_1, k^m_2]}) \leq_c (\text{i}m_{[k^m_1, k^m_2]})\) and it is \(\text{VNP-complete by proposition}\)

\[\text{[5, 2]}\]

\[
\begin{array}{c|c|c}
| & & \\
| & & \\
| & & \\
| & & \\
| & & \\
\end{array}
\]

\[k^m_1 \quad k^m_2\]

\[\delta\]
If \( k_m^2 \leq k_l^m \leq 2k_m^2 \) and \( \delta_m := (k_m^1 - k_m^2) = \Omega(n^\epsilon) \) for an \( \epsilon > 0 \). First, by using Corollary 3.2 of [1] (the projection they referring to is a type of c-reduction), we remove the first \( k_m^2 - \delta_m \) lines from \([k_m^1, k_m^2]\) (as \( \delta_m \leq k_m^2 \)).

We obtain the Young diagram \([2\delta_m, \delta_m]\). Then we apply the Murnaghan-Nakayama Rule, with a skew hook of size \( \delta_m \).

\[
im_{2\delta_m, \delta_m}(\bar{x}, \sigma_{\delta_m}) = (-1)^{\delta_m-1} \left( \nim_{[\delta_m, \delta_m]}(\bar{x}) + \nim_{[2\delta_m, 0]}(\bar{x}) \right)
\]

But, \( \nim_{[2\delta_m, 0]}(\bar{x}) = \det(\bar{x}) \) and it can be computed with a number of operation polynomial in \( \delta \). Therefore, with only a polynomial number of additions, multiplications and evaluations of \( \nim_{[k_m^1, k_m^2]} \) we can compute \( \nim_{[\delta_m, \delta_m]} \), which is VNP-complete by the proposition [1].

---

Else, i.e., if \( k_m^m \leq k_l^m \leq 2k_m^m \) and \( \delta_m := (k_m^1 - k_m^2) \) is bounded by a constant. We fixe \( m \in \mathbb{N} \) and we write \( \delta \) for \( \delta_m \), \( k_1 \) for \( k_m^1 \) and \( k_2 \) for \( k_m^2 \).

We want to generate every \([l, n - l]\) for \( \frac{n}{2} \leq l \leq n \) only with the Young diagram \([k_1, k_2]\) so that we can compute \( \text{Ferm}^n \). For a such \( l \), we first remove the first \( k_2 - 1 - l \) rows so we have a \([l + 1 + \delta, l + 1]\) diagram. Then we remove the skew hooks of length \( 2l - n + \delta + 2 \). There is two such skew tool in \([l + 1 + \delta, l + 1]\), which give, if we remove them, \([l, n - l]\) and \([l + \delta + 1, n - l - \delta - 1]\).
We apply the Murnaghan-Nakayama Rule with a skew hook of size $2l - n + \delta + 2$ and we sum all of them. The coefficients $a_{i,j}$ will be defined later.

\[
\sum_{l=\frac{n}{2} + \delta + 1}^{n-\delta} a_{[l,n-l]} \text{im}_{[l+1+\delta,n-l]}(\vec{x}, \sigma_{2l-n+\delta+2})
\]

\[
= \sum_{l=\frac{n}{2} + \delta + 1}^{n-\delta-1} a_{[l,n-l]} (-1)^{2l-n+\delta+1} \left( \text{im}_{[l,n-l]} + \text{im}_{[l+\delta+1,n-l-\delta-1]} \right)
\]

We put together the same immanants.

\[
=(-1)^{\delta-n+1} \left( \sum_{l=\frac{n}{2} + \delta + 1}^{n-\delta-1} a_{[l,n-l]} \text{im}_{[l,n-l]} + \sum_{p=\frac{n}{2} + 2\delta + 1}^{n} a_{[p-\delta-1,n-p+\delta+1]} \text{im}_{[p,n-p]} \right)
\]

\[
=(-1)^{\delta-n+1} \left( \sum_{l=\frac{n}{2} + 2\delta + 2}^{n-\delta-1} \text{im}_{[l,n-l]} \left( a_{[l,n-l]} + a_{[l-\delta-1,n-l+\delta+1]} \right) + \sum_{l=\frac{n}{2} + \delta + 1}^{\frac{n}{2} + 2\delta + 1} a_{[l,n-l]} \text{im}_{[l,n-l]} \right)
\]

\[
+ \sum_{p=n-\delta}^{n} a_{[p-\delta-1,n-p+\delta+1]} \text{im}_{[p,n-p]} \right)
\]
We take \( a_{[i,j]} \) such that \( (a_{[i,n-\ell]} + a_{[i-\delta-1,n-\ell+\delta+1]}) = d_{[i,n-\ell]} \). Then,

\[
= (-1)^{\delta-n+1} \left( \sum_{l=\frac{n}{2}}^{n-\delta-1} \text{im}_{[i,n-\ell]} d_{[i,n-\ell]} + \sum_{l=0}^{\delta} q_{[i+\delta+1,l,i-\delta-1-\ell]} \text{im}_{[i+\delta+1,l,i-\delta-1-\ell]} + \sum_{l=0}^{\delta} q_{[n-2\delta-1+l,2\delta+1-\ell]} \text{im}_{[n-2\delta-1+l,2\delta+1-\ell]} \right)
\]

And by Lemma 4, we have

\[
= (-1)^{\delta-n+1} \left( \text{Ferm}_{n}^{2} \sum_{l=0}^{\frac{n}{2}} d_{[i+l,n-\ell]} \text{im}_{[i+l,n-\ell]} - \sum_{l=n-\delta}^{n} d_{[i,n-\ell]} \text{im}_{[i,n-\ell]} \right)
+ (-1)^{\delta-n+1} \left( \sum_{l=0}^{\delta} q_{[i+\delta+1+l,i-\delta-1-\ell]} \text{im}_{[i+\delta+1+l,i-\delta-1-\ell]} \right)
+ (-1)^{\delta-n+1} \left( \sum_{l=0}^{\delta} q_{[n-2\delta-1+l,2\delta+1-\ell]} \text{im}_{[n-2\delta-1+l,2\delta+1-\ell]} \right)
\]

\[
= (-1)^{\delta-n+1} \left( \text{Ferm}_{n}^{2} \sum_{l=0}^{\delta} d_{[i+l,n-\ell]} \text{im}_{[i+l,n-\ell]} - \sum_{l=0}^{\delta} d_{[i-n-\delta+\ell,\delta-1-\ell]} \text{im}_{[i-n-\delta+\ell,\delta-1-\ell]} \right)
+ (-1)^{\delta-n+1} \left( \sum_{l=0}^{\delta} q_{[i+\delta+1+l,i-\delta-1-\ell]} \text{im}_{[i+\delta+1+l,i-\delta-1-\ell]} \right)
+ (-1)^{\delta-n+1} \left( \sum_{l=0}^{\delta} q_{[n-2\delta-1+l,2\delta+1-\ell]} \text{im}_{[n-2\delta-1+l,2\delta+1-\ell]} \right)
\]

We have to compute from \( \text{im}_{[k_{1}, k_{2}]} \) the residual terms. \( \sum_{l=0}^{\delta} q_{[n-2\delta-1+l,2\delta+1-\ell]} \text{im}_{[n-2\delta-1+l,2\delta+1-\ell]} \)
and \( \sum_{l=0}^{\delta} d_{[i-n-\delta+\ell,\delta-1-\ell]} \text{im}_{[i-n-\delta+\ell,\delta-1-\ell]} \) have both less than \( 2\delta + 2 \) boxes in the
last column. With the algorithm found in [2], we can compute those
immanants in a number of operations polynomially bounded in \( n \) but
exponential in \( \delta \). However, \( \delta \) is here a constant. Therefore, those residual terms
can be counted in a polynomial number of operations.
Now, let us look at

\[
\sum_{l=0}^{\delta} q_{[i+\delta+1+l,i-\delta-1-\ell]} \text{im}_{[i+\delta+1+l,i-\delta-1-\ell]}\]
We use a technique similar to before.
\[
\sum_{l=0}^{\delta} a[\frac{n}{2} + \delta + 1 + l, \frac{n}{2} - \delta + 1 - l] \text{im}[\frac{n}{2} + \delta + 1 + l, \frac{n}{2} + 1 + l](\bar{x}, \sigma_{\delta + 2l + 2})
\]
\[
= \sum_{l=0}^{\delta} a[\frac{n}{2} + \delta + 1 + l, \frac{n}{2} - \delta - 1 - l] \left( (-1)^{\delta + 2l + 1} \text{im}[\frac{n}{2} + \delta + 1 + l, \frac{n}{2} - 1 - \delta - l] + (-1)^{\delta + 2l + 2} \text{im}[\frac{n}{2} + l, \frac{n}{2} - l] \right)
\]
Indeed, there are two skew hooks of size $\delta + 2l + 2$ that we can remove from $[\frac{n}{2} + \delta + 1 + l, \frac{n}{2} + 1 + l]$; the one in the second column and the one represented in red in the graphic.

Therefore, from $\text{im}[\frac{n}{2} + 2\delta + 1, \frac{n}{2} + 1 + \delta]$ we can compute in only a polynomially number of operations
\[
\sum_{l=0}^{\delta} a[\frac{n}{2} + \delta + 1 + l, \frac{n}{2} - \delta - 1 - l] \text{im}[\frac{n}{2} + \delta + 1 + l, \frac{n}{2} - \delta - 1 - l] + \sum_{l=0}^{\delta} a[\frac{n}{2} + \delta + 1 + l, \frac{n}{2} - \delta - 1 - l] \text{im}[\frac{n}{2} + l, \frac{n}{2} - l]
\]
The first term is the one we were trying to compute, the second one can be added to the last residual term. Finally, we have to compute from $\text{im}[\bar{x}, \delta_{k_2}]
\[
\sum_{l=0}^{\delta} (d[\frac{n}{2} + l, \frac{n}{2} - l] + a[\frac{n}{2} + \delta + 1 + l, \frac{n}{2} - \delta - 1 - l]) \text{im}[\frac{n}{2} + l, \frac{n}{2} - l]
\]

We use the same technique.
\[
\sum_{l=0}^{\delta} b[\frac{n}{2} + l, \frac{n}{2} - l] \text{im}[\frac{n}{2} - l + \delta, \frac{n}{2} - l](\bar{x}, \sigma_{\delta - 2l})
\]
\[
= \sum_{l=0}^{\delta} b[\frac{n}{2} + l, \frac{n}{2} - l] (-1)^{\delta + 2l + 1} \left( \text{im}[\frac{n}{2} + l, \frac{n}{2} - l] + \text{im}[\frac{n}{2} - l + \delta, \frac{n}{2} - l + \delta] \right)
\]
\[
= (-1)^{\delta + 1} \left( \sum_{l=0}^{\delta} b[\frac{n}{2} + l, \frac{n}{2} - l] \text{im}[\frac{n}{2} + l, \frac{n}{2} - l] + \sum_{p=0}^{\delta} b[\frac{n}{2} + \delta - p, \frac{n}{2} + p - \delta] \text{im}[\frac{n}{2} + p, \frac{n}{2} - p] \right)
\]
\[
= (-1)^{\delta + 1} \sum_{l=0}^{\delta} \text{im}[\frac{n}{2} + l, \frac{n}{2} - l] \left( b[\frac{n}{2} + l, \frac{n}{2} - l] + b[\frac{n}{2} + \delta - p, \frac{n}{2} + p - \delta] \right)
\]
\[
= (-1)^{\delta + 1} \sum_{l=0}^{\delta} (d[\frac{n}{2} + l, \frac{n}{2} - l] + a[\frac{n}{2} + \delta + 1 + l, \frac{n}{2} - \delta - 1 - l]) \text{im}[\frac{n}{2} + l, \frac{n}{2} - l]
\]
Theorem 4 Let $(Y_n)$ be a family of Young diagrams of length bounded by $k \geq 2$ such that its number of boxes is $\Omega(n)$. Then

- if the number of boxes in the right of the first column if bounded by a constant $c$, then $(\text{im} Y_n)$ is in VP.
- otherwise, if there is an $\epsilon > 0$ and at least $n^\epsilon$ boxes at the right of the first column, $(\text{im} Y_n)$ is VNP-complete for $c$-reductions.

Proof The first case is a result of the algoritheorem found in [2]. For the second case, we can suppose that every column counted, i.e., that the number of boxes in the last row is of $\Omega(n^\epsilon)$ for an $\epsilon > 0$. Indeed, if it is not, let $l < k$ be such that the number of boxes in the last $l$ columns is bounded by a constant $c$ in every $Y_n$ but the number of boxes in the $l+1$ last column is not. Then if we remove from every $Y_n$ the first $c$ rows then for every $n$, $Y_n$ has no boxes in the last $l$ columns and $\Omega(n^\epsilon)$ in the $l+1$ last column.

Now that every column has at a nonconstant number of boxes, especially the last two, let us remove the first $k-2$ columns. The Young diagrams we obtain, $\mu_n$, have only two columns but at least $\Omega(n^\epsilon)$ boxes for an $\epsilon > 0$. And the second column has a non-bounded number of boxes. Therefore, we are in the case of $k = 2$ and $(\text{im} \mu_n)$ is VNP-complete. Furthermore, we have, by Corollary 3.2 of [1], $(\text{im} \mu_n) \leq_p (\text{im} Y_n)$. Thus $(\text{im} Y_n)$ is VNP-complete.

5 Conclusion and Perspectives

The generalization via the fermionant tell us that the determinant is really special: the coefficients $1$ and $-1$ allows us, in a simplify way, to cancel some monomials and not to have to compute everything. The $k$ in the fermionant, even thinly different than $1$, separates these monomials and prevents the cancelations.

As for the immanant, the interpretation of the result is harder. Especially as our theorem does not completely classify immanants of constant width; what about the immanant of $[n, \log n]$? Burgisser’s algoritheorem gives a subexponential upper bound, but does not put it in VP. However, under the extended Valiant hypothesis (end of chapter 2 in [2]), it can not be VNP-complete. Is it a good candidate to be neither VP nor VNP-complete? Or even VP-complete? Or is it as hard as the determinant? This is unknown.

Other generalizations also can be imagined. For example generating functions of a graph property are polynomials that generalize the permanent and some of them can be computed as fast as the determinant. This framework allows us to use our knowledge on graph theory to understand where we step from VP to VNP. There is no classification of these generation functions, but some results have been found [2,4].

I thank to both of my doctoral advisors, A. Durand and G. Malod as well to M. Casula and E. Boulat who try to explain to me the physic behind the fermionant.
References