CS 860 Algebraic Complexity Theory May 06, 2024

Determinant is VBP-complete

Instructor: Rafael Oliveira *Scribe: Omkar Baraskar*

Remark These notes have not been subjected to the scrutiny of a journal publication, and it may contain errors. If you find one, please contact the instructor.

1 Introduction

In this lecture we introduce a new algebraic computational class VBP which consists of polynomials which can be computed by "small" ABP. The size of an ABP is defined as the number of edges present in the ABP[1]. Formally the class VBP is defined as follows:

Definition 1 (VBP)**.** *A p-bounded family $\{f_n\}$ belongs to* VBP *iff there exists polynomial function $t : \mathbb{N} \to \mathbb{N}$ such that f_n can be computed by a* ABP*of size $\leq t(n)$ for all $n \in \mathbb{N}$*

As in boolean complexity we look for complete problems (i.e languages) of a complexity class under some reduction (most common being karp-reduction). Similarly for algebraic computational classes we are interested in complete polynomial with p-projection as the notion of reduction. And it turn out for the class VBP determinant is a complete polynomial!

In section 2 we show that the family $\{\mathsf{Det}_n\}$ is hard for the class VBP which means that every polynomial family $\{f_n\}$ in class VBP is a p-projection of $\{\mathsf{Det}_n\}$. In section 3 we show that the family $\{\mathsf{Det}_n\}$ is in VBP. Hence this shows that $\{\mathsf{Det}_n\}$ is VBP-complete.

2 Determinant is VBP-hard Polynomial

To show that $\{\mathsf{Det}_n\}$ is VBP-hard we will first look at a graph-theoretic interpretation of determinants.

2.1 Determinant and Cycle covers

Given a graph G with vertex set $[n]$ and edge set E, let A be the adjacency matrix of graph of G (In case of weighted graph it is a weighted adjacency matrix).

Definition 2 (Cycle Cover)**.** *We say $C = C_1 \cup \cdots \cup C_m$ is a cycle cover of G if*

- *Each C_i is a cycle of G*
- *$V(C_1) \cup \cdots \cup V(C_m) = [n]$*
- *For every $i \neq j$ we have $V(C_1) \cap V(C_2) = \phi$*

where $V(C_i)$ is the vertex set of C_i for every $i \in [m]$.

Definition 3 (Weight Of Cycle Cover)**.** *If the graph G is weighted and has weight function is given by $w : [n] \to \mathbb{F}$, then we define weight and sign of a cycle cover $C = C_1 \cup \cdots \cup C_m$ as*

$$w(C) = \prod_{i=1}^{m} w(C_i) \qquad \mathsf{sgn}(C) = \prod_{i=1}^{n} \mathsf{sgn}(C_i)$$

where $w(C_i) = \prod_{e \in C_i} w(e)$ and $\mathsf{sgn}(C_i) = (-1)^{|C_i|+1}$

[1] We can also use number of vertices as a complexity measure for ABP as $e \leq \mathcal{O}(v^2)$ where e, v are number of edges and number of vertices respectively

The following lemma will show cycle covers of a graph and determinant polynomial are related.

Lemma 4. *Let X be a $n \times n$ symbolic matrix i.e the (i,j)-th entry is labelled by the variable x_{ij}. Moreover let G be a complete graph on $[n]$ vertices with edge (i,j) labelled by x_{ij} for $i,j \in [n]$*[2]*. Then we have*

$$\mathsf{Det}(X) = \sum_{C \in \mathcal{C}(G)} \mathsf{sgn}(C) w(C)$$

where $\mathcal{C}(G)$ is the set of all cycles covers in G.

Proof. We show this by showing one-to-one correspondence between cycle covers of G and permutations on $[n]$ such that it preserves the sign.

- Given a permutation π on $[n]$ we know that it can written as a composition of cycles $C = C_1 \circ \cdots \circ C_r$. Note since G is a complete graph these C_i are cycles in it. And it is easy to see that $(C_1, \ldots, C_r)$ is a cycle cover. Also note that the sign of $\mathsf{sgn}(\pi) = \mathsf{sgn}(C)$.
- Given a cycle cover $C = (C_1, \ldots, C_r)$ on G note that one can also view these C_i as injective function from C_i to itself (Say an edge (a,b) is present in C_i then our function maps $a \mapsto b$). It is easy to check that $\pi = C_1 \circ \cdots \circ C_r$ is a permuatation on $[n]$ because of properties of cycle cover. And it is also easy to see that $\mathsf{sgn}(\pi) = \mathsf{sgn}(C)$.

□

How can this interpretation of determinant help us? Notice that ABP is a graph hence looking at it's cycle covers might help us find a matrix whose determinant is the polynomial computed by the ABP. It is not clear how we can find such a matrix, but we before that we have a even bigger problem which is that an ABP is a acyclic graph, hence it has no cycles! But it turns out by a simple yet clever modification of the ABP we will solve both of these problems.

2.2 Modifying the ABP

We will modify the ABP Φ to form a DAG Φ' by adding the following two types of edges:-

1. Add a edge directed from t to s of weight $(-1)^b$ where we will set b later[3]
2. Add self-loops of weight 1 on every vertex of the ABP

Let's demonstrate this by an example, the following figure shows an ABP computing $x^2 - y^2$ and the DAG we obtain after modifying it.

Since Φ was acyclic hence the cycle covers of Φ' are well-structured as shown in the following lemma.

Lemma 5. *The following statements are equivalent*

- *$C = C_1 \cup \cdots \cup C_m$ is a cycle cover*
- *For some $i \in [m]$ we have $C_i = p \to t \to s$ where p is a $s \rightsquigarrow t$ path. And the rest of the C_j's are self-loops for $j \neq i$ on vertices not present in p and $V(C_1) \cup \cdots \cup V(C_m) = [n]$*

Proof. Let $C = C_1 \cup \cdots \cup C_m$ be a cycle cover of Φ', and without loss of generality assume $s \in V(C_1)$. Notice that for s to be in a cycle the edge from $t \to s$ should always be present, because else it would contradict the fact that Φ is a DAG, thus $t \in V(C_1)$. Removing the edge $t \to s$ gives us p a $s \rightsquigarrow t$ path. Hence $C_1 = p \to t \to s$. Notice that once we remove C_1 we get a graph hence where every cycle is a self-loops as Φ is a DAG. So the only way to cover rest of the vertices is through self-loops. Also since it is a cycle cover $V(C_1) \cup \cdots \cup V(C_m) = [n]$.

Let p be a $s \rightsquigarrow t$ path then let $C_1 = p \to t \to s$ as we have a directed edge from $t \to s$. And let $C_2, \ldots, C_m$ be self-loops on vertices not present in p. Hence it is easy to see that $C_1 \cup \cdots \cup C_m$ is a cycle cover. □

[2] We allow graph G to have self-loops

[3] In our reduction we are able to do that because as you will see that the value of b only depends on the structure of the ABP

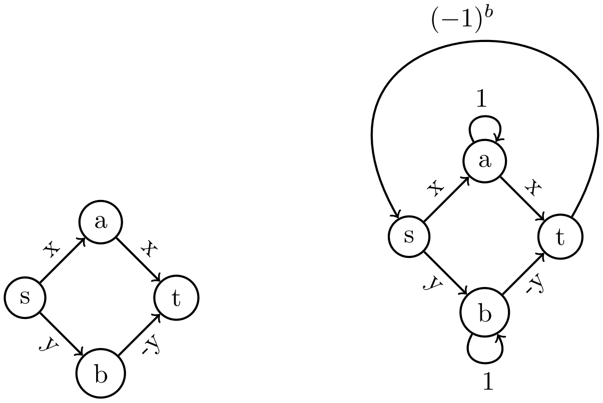

Figure 1: Modification of Φ to Φ'

We can easily verify lemma 5 on example in figure 1, as the two cover cycles present it it are $(s \to a \to t \to s) \cup (b \to b)$ and $(s \to b \to t \to s) \cup (a \to a)$.

Hence combining lemma 5 and lemma 4 we have get our desired result.

Theorem 6. $\mathsf{Det}(A) = f(\Phi)$ *where* A *is the adjacency matrix of* Φ' *and* $f(\Phi)$ *is polynomial computed by the* ABP Φ

Proof. By lemma 4 we have

$$\mathsf{Det}(A) = \sum_{\mathcal{C}(\Phi')} \mathsf{sgn}(C) \cdot w(C)$$

where $\mathcal{C}(\Phi')$ is the set of all cycle cover in Φ'. For every cycle cover $C = C_1 \cup \cdots \cup C_m$ by lemma 5 we can assume without loss of generality $C_1 = p \to t \to s$ where p is a $s \rightsquigarrow t$ path, and C_j's are self-loops. Hence $w(C) = w(C_1) \cdots w(C_m) = w(C_1) = w(p) \cdot (-1)^b \cdot (-1)^{|p|}$. Since all $s \rightsquigarrow t$ path are of same have the same length equal to length l of the ABP. Hence $w(C) = (-1)^{b+l} \cdot w(p)$. Thus we have the following:

$$\begin{aligned}\mathsf{Det}(A) &= \sum_{\mathcal{C}(\Phi')} \mathsf{sgn}(C) \cdot w(C)\\ &= (-1)^{b+l} \cdot \sum_{C_1 = p \to t \to s} w(p)\end{aligned}$$

By lemma 5 we also know that for every $s \rightsquigarrow t$ path p we know that there is a cycle cover with $p \to t \to s$ is one of the cycles, hence

$$\begin{aligned}\mathsf{Det}(A) &= (-1)^{b+l} \cdot \sum_{C_1 = p \to t \to s} w(p)\\ &= (-1)^{b+l} \cdot \sum_{p \in s \rightsquigarrow t} w(p)\\ &= (-1)^{b+l} f(\Phi)\end{aligned}$$

Hence if we select $b = l$ then we have $\mathsf{Det}(A) = f(\Phi)$. This completes the proof

□

Let's verify this theorem by checking it on example in figure 1, the adjacency matrix A of Φ' is

$$\begin{array}{cc} \begin{matrix} s & a & b & t \end{matrix} & \\ \begin{pmatrix} 0 & x & y & 1 \\ 0 & 1 & 0 & x \\ 0 & 0 & 1 & -y \\ 1 & 0 & 0 & 0 \end{pmatrix} & \begin{matrix} s \\ a \\ b \\ t \end{matrix} \end{array}$$

We can check that $\mathsf{Det}(A) = x^2 + y^2$. So we are done, right? Unfortunately not completely, there is one more bit of nuisance that needs to be addressed. Note that in p-projection we can substitute the variables by other variables or field elements, but the entries of A might have affine forms. How do we fix this? Well it turns out for every ABP Φ there exists another "small" ABP Ψ such that the edges of Ψ are either labelled by field elements or variables and it computes the same polynomial as Φ. This is given by the following lemma:-

Lemma 7. *Given a ABP Φ of size s then there exists an ABP Ψ of size poly(s) computing the same polynomial such that the edges in Ψ are labelled by a variable or a field element*

Proof. Let Ψ have the same vertex set as Φ but with no edges between s and first layer $\{e_1, \dots, e_m\}$. The idea is to add $n+1$ nodes namely $v_0, \dots, v_n$ between s and $\{e_1, \dots, e_m\}$ in Ψ. We assign the edge weights as follows:-

- The edge weight of $s \to v_0$ is 1
- The edge weight of $s \to v_i$ is x_i
- Say $s \to e_i$ is labelled by $l_{i0} + l_{i1}x_1 + \cdots + l_{in}x_n$ then we add edge $v_j \to e_i$ by $l_{ij} \cdot x_j$ for $j \in [n]$, and $v_0 \to e_i$ labelled by l_0

Why does this work? Notice that this modified ABP Ψ indeed computes

$$\sum_{e \in \{e_1, \dots, e_M\}} l_{s \to e} \cdot f(\Phi_{e \to s}) = f(\Phi)$$

where $l_{s\to e}$ is the affine form labelling the edge $s \to e$ in Φ, and $f(\Phi_{e \to t})$ is the ABP computed when we consider e to be source and t to be sink. Doing this for each layer gives us the desired result! □

Figure 2 demonstrates the process described in lemma 7.
This shows that $\{\mathsf{Det}_n\}$ is VBP-hard over any field $\mathbb{F}$.

Remark The reduction also goes through for $\{\mathsf{Perm}_n\}$ hence $\{\mathsf{Perm}_n\}$ is VBP-hard.

3 Determinant is in VBP

We will now show determinant is in VBP more precisely we will show Det_n can be computed by a ABP of size *poly*(n). From previous section we might say consider the symbolic matrix X_n as a adjacency matrix for a graph on n vertices, and since determinant is the signed sum over cycle covers hence we can try to encode every cycle cover on different paths but this will lead to exponential sized ABP (since every monomial in Det_n represents a different cover and there are exponential number of monomials), thus this naive approach does not work! Second one might also want to try to use the self-reducibility[4] but one will realise that this does not work as well, because the resulting ABP is of exponential width!. A width of an ABP is usually

[4] This refers to the fact that $\mathsf{Det}_n = \sum_{j=1}^{n} (-1)^{j+1}\, x_{1j} \cdot \mathsf{Det}(M_{1j})$ where M_{1j} is the $(1, j)$ minor of X_n

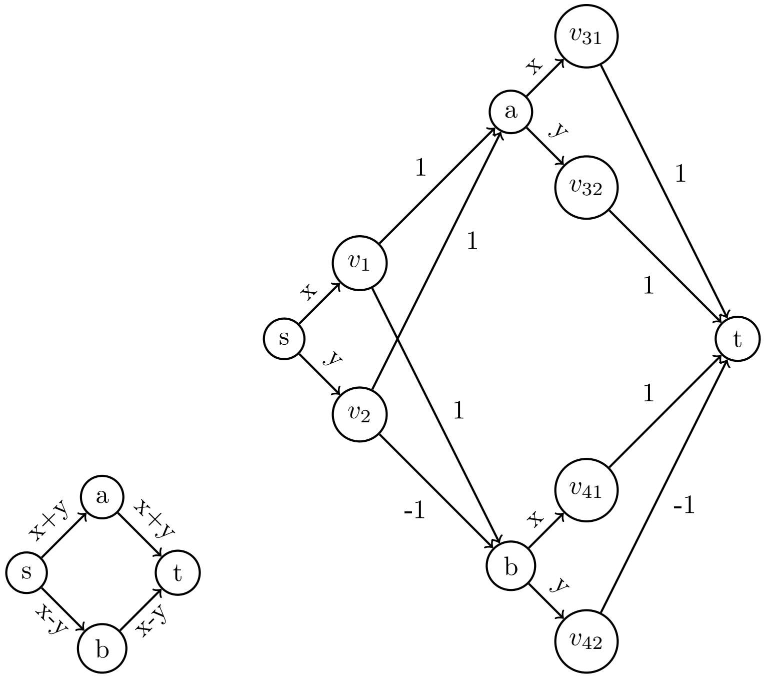

Figure 2: Modification of Φ to Ψ

symbolises the memory used in the computation, so the above approaches suggest that encoding cycle-covers takes a lot of memory. According to [Sap21] the key-insight of Mahajan and Vinay [MV97] was to "weaken" the notion of cycle covers to *clow-sequences* which can be constructed with less-memory. We present thier proof of construction below.

3.1 Clow-Sequences

We will define the following for graph G on n-vertices and edge set E. We will call A be the adjacency matrix of graph of G (In case of weighted graph it is a weighted adjacency matrix).

Definition 8 (Clow)**.** *A closed walk or clow on G is a sequence $W = (v_1, \dots, v_l)$ such that $v_1 < v_i$ for all $2 \leq i \leq l$. Also,*

- $\mathsf{head}(W) = v_1$
- $\mathsf{length}(W) = l$
- $\mathsf{sgn}(W) = (-1)^{l-1}$
- $\mathsf{wt}(W) = \prod_{i=1}^{l} w(v_i \to v_{i+1})$

where $v_{l+1} = v_1$ and $w : E \to \mathbb{F}$ is the weight function of G.

Definition 9 (Clow-Sequences)**.** *A clow-sequence is a sequence of clows $(W_1, \dots, W_r)$ satisfying $\mathsf{head}(W_1) < \mathsf{head}(W_2) < \cdots < \mathsf{head}(W_{r-1}) < \mathsf{head}(W_r)$. Also as in the case of clow we have,*

- $\mathsf{length}(W) = \sum_{i=1}^{r} \mathsf{length}(W_i)$
- $\mathsf{sgn}(W) = \prod_{i=1}^{r} \mathsf{sgn}(W_i) = (-1)^{\mathsf{length}(W)-r}$

Now the next lemma tells us why these clow-sequences are useful for us,

Lemma 10. *Let X be a $n \times n$ symbolic matrix i.e the (i, j)-th entry is labelled by the variable x_{ij}. Moreover let G be a complete graph on $[n]$ vertices with edge (i, j) labelled by x_{ij} for $i, j \in [n]$*[5]*. Then we have*

$$\mathsf{Det}(X) = \sum_{W \in \mathcal{CC}(G,n)} \mathsf{sgn}(W) \cdot \mathsf{wt}(W)$$

where $\mathcal{CC}(G, n)$ is the set of all clow-sequences of length n.

Proof. We first observe that all cycle covers are clow-sequences of length n and the two agree on notions of sign and length. We now show that the rest of the clow sequences cancel each other.
Say $W = (W_1, \ldots, W_r)$ is not a cycle cover, then there exists a s such that $(W_{s+1}, \ldots, W_r)$ is union of disjoint cycles but $(W_s, \ldots, W_r)$ is not. Let j be the first index in W_s such that one of the following happens

- **Case 1:** $v_j = v_{j'}$ for some $j' > j$ and $v_j \notin W_{s+i}$ for all $i > 0$
- **Case 2:** $v_j \in W_{s+i}$ for some $i > 0$

Now we will show bijection between clow-sequences satisfying case 1 and case 2.

- Say a clow sequence W satisfies case 1 then say $j' > j$ be the smallest index such that $v_j = v_{j'}$. Let $W_s = (v_1, \ldots, v_j, \ldots, v_{j'}, \ldots, v_m)$ then we create a new clow sequence W' by removing W_s and adding $W_{s1} = (v_1, \ldots, v_j, v_{j'+1}, \ldots, v_m)$ and $W_{s2} = (v_j, \ldots, v_{j'-1})$. Note we can place W_{s2} appropriately in the clow-sequence. Also it is easy to see that $\mathsf{wt}(W) = \mathsf{wt}(W')$. Since number of clows in W' is one more than W hence $\mathsf{sgn}(W) = -\mathsf{sgn}(W')$. Thus $\mathsf{sgn}(W) \cdot \mathsf{wt}(W) + \mathsf{sgn}(W') \cdot \mathsf{wt}(W') = 0$
- Say a clow sequence W satisfies case 2 then there exists a unique i such that $v_j \in W_s \cap W_{s+i}$. Say $W_s = (v_1, \ldots, v_j, \ldots, v_m)$ and $W_{s+i} = (v_j, u_1, \ldots, u_{m'})$, now we remove W_s and W_{s+i} from W and add $W'_s = (v_1, \ldots, v_j, u_1, \ldots, u_{m'}, v_j, \ldots, v_m)$ to form the clow sequence W'. First note that W'_s is a "valid" clow as $v_1 < v_j < u_i$ for all $i \in [1, m']$. Also it satisfies case 1 (because there exists no other i such that $v_j \in W_s \cap W_{s+i}$). Also it is easy to see $\mathsf{wt}(W) = \mathsf{wt}(W')$. Since number of clows in W' is one more than W hence $\mathsf{sgn}(W) = -\mathsf{sgn}(W')$. Thus $\mathsf{sgn}(W) \cdot \mathsf{wt}(W) + \mathsf{sgn}(W') \cdot \mathsf{wt}(W') = 0$

From the above correspondence it is clear that the clow-sequence which are not cycle-cover cancel each other. Figure 4 shows clow sequences which are not cycle covers. Left side has clow sequences satisfying case 1 and right has clow sequences satisfying case 2. Moreover adjacent clow sequences are ones in one-to-one correspondence w.r.t transformation described above.

□

3.2 Construction using the clow-sequences

In this section we assume G to be a complete graph on n vertices with the symbolic matrix X_n to be the weighted-adjacency matrix for G. By lemma 10 we have determinant can be written as a signed weighted sum of clow-sequences of G whose length is n, hence we are going to construct an ABP such that there is one-to-one correspondence between $s \rightsquigarrow t$ path and clow-sequence of G whose length is n.

We have $n + 1$ layers labelled by $[0, n]$. All layers other than 0 and n have n^2 nodes. The following is the description of the nodes:-

- Layer 0 is the source node labelled by $v_{1,1}^{(0)}$
- Layer $n + 1$ is the sink node labelled by t
- Layer l has n^2 nodes labelled by $v_{i,j}^{(l)}$ for $i, j \in [n]$ and for all $1 \le l \le n - 1$

[5] We allow graph G to have self-loops

Interpretation of layers and nodes: A node $v_{i,j}^{(l)}$ represents a "partial" clow-sequence of length l with i being the head of the current clow and j being the current vertex of the clow.

We will introduce two kinds of edges in the ABP

1. For every $0 \leq l \leq n-1$ we have an edge between $x_{ij}^{(l)}$ and $x_{ik}^{(l+1)}$ for every $k \geq i$ labelled by $-x_{jk}$ for all $i, j \in [n]$
2. For every $0 \leq l \leq n-1$ we have an edge between $x_{ij}^{(l)}$ and $x_{kk}^{(l+1)}$ labelled by $x_{j,i}$ for all $k > i$ for $i, j \in [n]$

Interpretation of edges: The two kinds of edges in the ABP have the following interpretation:

1. The first kind of edge is between $x_{ij}^{(l)}$ and $x_{ik}^{(l+1)}$ for some $i, j, k \in [n]$ and $l \in [0, n-1]$. When we are at $x_{ij}^{(l)}$ node it means that our "partial" clow-sequence has i as its current head and the current vertex is j. The edge represents that we choose to stay in the same clow (with i as the head), and we add the edge (j, k) to the clow hence we label it by $-x_{jk}$. Considering this interpretation the edge is rightfully directed towards $x_{ik}^{(l+1)}$
2. The second kind of edge is between $x_{ij}^{(l)}$ and $x_{kk}^{(l+1)}$ for some $i, j, k \in [n]$ and $l \in [0, n-1]$. When we are at $x_{ij}^{(l)}$ node it means that our "partial" clow-sequence has i as its current head and the current vertex is j. The edge represents that we close the current clow and start a new clow at k. Hence the edge is labelled by x_{ji}. Considering this interpretation the edge is rightfully directed towards $x_{kk}^{(l+1)}$

From the above discussion we have a one-one correspondence between $s \rightsquigarrow t$ paths of the ABP and clow-sequence in G of length n. It is also clear that the absolute value of the polynomial computed by an $s \rightsquigarrow t$ is equal to weight of some clow-sequence of length n, but does the sign match?

Sign of clow-sequence: As we can observe that the only edges which do not have a negative sign are one of the second kind. In the ABP we take this edge when we want to close the clow. Hence for each each clow in the clow-sequence there is exactly one edge with sign 1 all rest of the edges have -1. Thus if a $s \rightsquigarrow t$ path represents a clow-sequence $W = (W_1, \ldots, W_r)$ then the sign of the polynomial computed is $(-1)^{\mathsf{length}(W)-r}$ which is exactly equal to $\mathsf{sgn}(W)$.

Thus by lemma 10 we have the following

Theorem 11. *There exists a* ABP *of size* $\mathcal{O}(n^3)$ *computing* Det_n *over any field* $\mathbb{F}$

3.3 Clow sequences on three vertice graph

Below are all the cycle covers on a complete graphs on 3 vertices

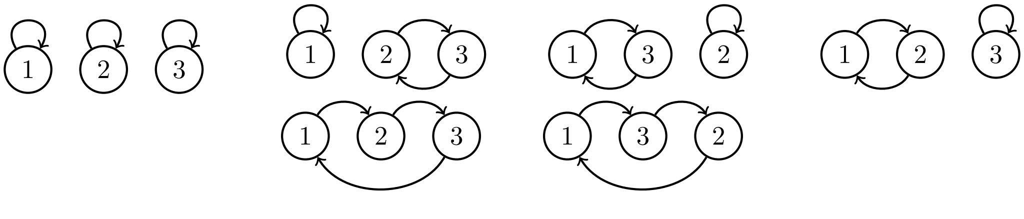

Figure 3: All cycle covers

Here are clow-sequences whose are not cycle covers

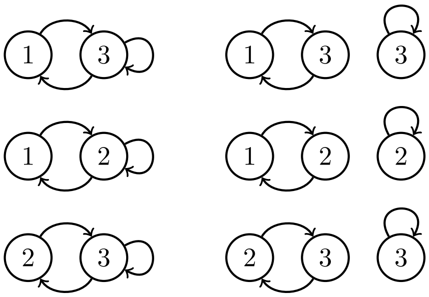

Figure 4: Clow sequences which are not cycle covers

References

[MV97] Meena Mahajan and V. Vinay. A combinatorial algorithm for the determinant. In *Proceedings of the Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '97, page 730–738, USA, 1997. Society for Industrial and Applied Mathematics.

[Sap21] Ramprasad Saptharishi. *A survey of lower bounds in arithmetic circuit complexity.* 2021.