Complexity of Counting

NP theory captures the difficulties of finding certificates.

In some applications we are interested in counting certificates.

Leslie Valiant studied counting complexity in late 70's.

1. The Complexity of Enumeration and Reliability Problems. SIAM J. Computing 8:410-421, 1979.
2. The Complexity of Computing the Permanent. Theoretical Computer Science, 8:189-201, 1979.

Synopsis

1. Counting Problem
2. $\sharp\mathbf{P}$
3. Valiant Theorem
4. Universal Hash Function
5. Valiant-Vazirani Theorem
6. Toda Theorem

Counting Problem

♯CYCLE

♯CYCLE is the problem of computing the number of simple cycle in a digraph G.

Finding a simple cycle can be done in linear time.

$\sharp$SAT

The counting version of SAT:

- $\sharp$SAT is the problem of computing, given a boolean formula ϕ, the number of satisfying assignments of ϕ.

A problem equivalent to $\sharp$SAT is the following:

- Given a boolean formula with n variables, what is the fraction of the satisfying assignments with $x_1 = 1$?

Network Reliability

Given a digraph on n nodes, where each node/edge can fail with probability $1/2$. Compute the probability that node 1 can reach n.

The problem boils down to computing the number of node/edge induced subgraphs in which there is a path from 1 to n.

A counting problem can be difficult even if the corresponding decision problem is easy.

Counting can be Harder than Decision

Theorem If $\sharp\texttt{CYCLE}$ has a polynomial algorithm, then $\mathbf{P} = \mathbf{NP}$.

Given a digraph G with n-nodes, we create a digraph G' by replacing every edge of G from s to t by a digraph such that there are 2^m paths from s to t, where $m = n\log n$.

- If G has a Hamiltonian cycle, G' has at least $2^{mn} = n^{n^2}$ cycles.
- If G has no Hamiltonian cycle, G' has fewer than $n^{n-1}2^{m(n-1)} = \frac{1}{2}\cdot 2^{n^2}$ cycles.

We have reduced an NP-complete problem to $\sharp\texttt{CYCLE}$.

$\sharp$P

Complexity Class $\sharp\mathbf{P}$

A function $f\colon \{0,1\}^* \to \mathbf{N}$ is in $\sharp\mathbf{P}$ if there exists a polynomial $p : \mathbf{N} \to \mathbf{N}$ and a P-time TM $\mathbb{M}$ such that for every $x \in \{0,1\}^*$ the following holds:

$$f(x) = \left|\left\{y \in \{0,1\}^{p(|x|)} \mid \mathbb{M}(x,y) = 1\right\}\right|.$$

- $f(x)$ has polynomial bits.
- $\sharp\mathbf{P}$ can also be defined in terms of P-time NDTM.

Complexity Class **FP**

Let **FP** be the set of functions : $\{0,1\}^* \to \mathbf{N}$ computable by P-time Turing Machines.

$\mathbf{FP} \subseteq \sharp\mathbf{P}$.

Proof.

Suppose $f \in \mathbf{FP}$. Then "*if* $y < \llcorner f(x) \lrcorner$ *then* 1 *else* 0" witnesses $f \in \sharp\mathbf{P}$. □

Complexity Class **FP**

If $\sharp\mathbf{P} = \mathbf{FP}$ then $\mathbf{NP} = \mathbf{P}$.

If $\mathbf{PSPACE} = \mathbf{P}$ then $\sharp\mathbf{P} = \mathbf{FP}$.

PP as a Decision Version of $\sharp$**P**

Recall the definition of **PP** introduced in Randomized Computation.

A language L is in **PP** if there exists a polynomial $p : \mathbf{N} \rightarrow \mathbf{N}$ and a P-time TM $\mathbb{M}$ such that for every $x \in \{0,1\}^*$ the following holds:

$$x \in L \text{ iff } \left|\left\{y \in \{0,1\}^{p(|x|)} \mid \mathbb{M}(x,y) = 1\right\}\right| \geq \frac{1}{2} \cdot 2^{p(|x|)}.$$

PP looks at the most significant bit of counting value.

Theorem. **PP** $=$ **P** if and only if $\sharp$**P** $=$ **FP**.

Suppose $f \in \sharp\mathbf{P}$. Let $\mathbb{M}$ be a P-time TM and p be a polynomial such that for all x,

$$f(x) = \left|\left\{y \in \{0,1\}^{p(|x|)} \mid \mathbb{M}(x,y) = 1\right\}\right|.$$

Let $\ell \in \{0,1\}^{p(|x|)}$. Define a TM $\mathbb{L}$ as follows:

$$\mathbb{L}(x, by) = \textit{if } b = 1 \textit{ then } \mathbb{M}(x,y) \textit{ else if } y < \ell \textit{ then } 1 \textit{ else } 0.$$

If **PP** $=$ **P**, we can decide in P-time if $f(x) + \ell \geq 2^{p(|x|)}$. A binary search produces the ℓ' rendering true the equality $f(x) + \ell' = 2^{p(|x|)}$.

♯**P**-Completeness

A function $f\colon \{0,1\}^* \rightarrow \mathbf{N}$ gives rise to an oracle

$$O_f = \{\langle x, i, d\rangle \mid f(x)_i = d \wedge (d = 0 \vee d = 1)\}.$$

We write $\mathbf{FP}^f$ for the set of functions computable by P-time TM's with oracle O_f.

---

f is ♯**P**-complete if it is in ♯**P** and every ♯**P**-problem is in $\mathbf{FP}^f$.

Theorem. $\sharp$SAT is $\sharp$**P**-complete.

Suppose $\mathbb{M}$ is a TM witnessing $f \in \sharp\mathbf{P}$.

- The Cook-Levin reduction gives rise to a P-time algorithm that calculates f using $\sharp$SAT as an oracle.

We are done using the parsimonious property.

The counting version of many NP-complete problems are known to be $\sharp$**P**-complete.

Valiant Theorem

Leslie Valiant provided convincing argument that computing permanent is far more difficult than calculating determinant.

---

1. Leslie Valiant. The Complexity of Computing the Permanent. Theoretical Computer Science, 8:189-201, 1979.

Permanent and Determinant

The permanent of an $n \times n$ matrix A is the "sum-of-product"

$$\mathtt{perm}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} A_{i,\sigma(i)},$$

where S_n is the set of all permutations of $\{1, \ldots, n\}$.

The determinant of an $n \times n$ matrix A is

$$\mathtt{det}(A) = \sum_{\sigma \in S_n} (-1)^{sgn(\sigma)} \prod_{i=1}^{n} A_{i,\sigma(i)},$$

where $sgn(\sigma) = 1$ if $\sharp\{(j,k) \mid j < k \wedge \sigma(j) > \sigma(k)\}$ is odd, and $sgn(\sigma) = 0$ if otherwise.

Using Gauss elimination determinant is computable in $O(n^3)$ time.

Combinatorial Interpretation of Permanent

Combinatorial interpretation of matrix:

- The adjacency matrix of a weighted bipartite graph.
- The adjacency matrix of a weighted complete digraph admitting self loops.

For a 0-1 matrix the permanent is the number of perfect matching in the former interpretation and the number of cycle cover in the latter interpretation.

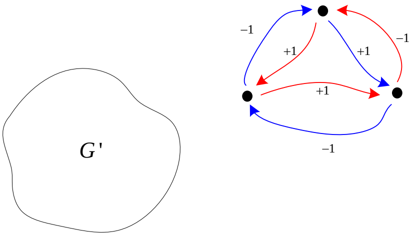
Theorem (Valiant, 1979). `Perm` for 0-1 matrix is $\sharp$**P**-complete.

The proof consists of two reductions:

- A reduction from $\sharp$`SAT` to the permanent problem of matrix.
- A reduction from the latter to the permanent problem of 0-1 matrix.

Valiant's First Reduction

A basic technique in Valiant's reduction can be explained using the following digraph.

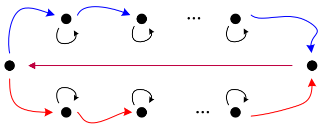


Valiant's First Reduction

Given a 3CNF φ with n variables and m clauses, we construct a digraph by piecing together variable digraphs and clause digraphs via exclusive-or digraphs.

Valiant's First Reduction

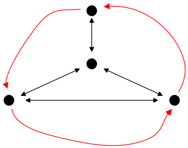
For each variable there is a variable digraph containing a true cycle (of true edges) and a false cycle (of false edges) that shares an additional common edge.



- The true cycle and the false cycle are exclusive.
- Both contribute 1 to the overall weight.

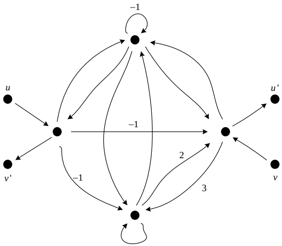
Valiant's First Reduction

The following is a clause digraph:



- A cycle cover may not contain all three literal edges.
- There is only one cycle cover that has none, or one specific, or two specific literal edges; each contributes 1 to the overall weight.

Valiant's First Reduction



The above diagram is the exclusive-or of $u \to u'$ and $v \to v'$.

---

1. Precisely one of $u \to u'$, $v \to v'$ appears in a cycle cover.
   - A cycle cover that passes through the four nodes contribute to weight 4.
2. Neither $u \to v'$ nor $v \to u'$ need be considered.
   - The total weight a cycle cover over the top and the bottom node [+ the left node] [+ the right node] cancels out.

Valiant's First Reduction

A literal edge of x ($\neg x$) in a clause diagraph is connected to a true (false) edge of the variable digraph of x via an exclusive-or digraph.

Lemma. The permanent of the digraph is $4^{3m}\sharp\varphi$, where $\sharp\varphi$ is the number of the assignments that validate φ.

Proof.

The cycle covers of the variable digraphs correspond to the true assignments. Each edge of a clause digraph contributes to a factor of 4. □

Valiant's Second Reduction

1. Transforming a matrix to a $\{-1, 0, 1\}$-matrix:
   - An edge with weight $2^{a_k} + 2^{a_{k-1}} + \ldots + 2^{a_1}$ is replaced by k parallel edges with weights $2^{a_k}, 2^{a_{k-1}}, \ldots, 2^{a_1}$ respectively.
   - An edge with weight 2^a is replaced by a edges of weight 2.
   - An edge with weight 2 is decomposed into a $\diamond$-shape diagraph.

   Introduce self-loops to all the new nodes.

2. Turning an $n \times n$ $\{-1, 0, 1\}$-matrix to a 0-1-matrix:
   - The absolute value of such a permanent is $\leq n! < 2^{n^2} + 1$. So we replace an edge with weight -1 by an edge with weight 2^{n^2}.
   - Repeat the previous transformation.

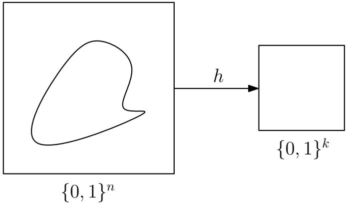
The permanent of the end matrix is calculated modular $2^{n^2} + 1$.

Universal Hash Function

Independent hash functions are costly.

Using k-wise independence one may reduce the amount of randomness.

Uniformity.



Efficiency.

---

1. Carter and Wegman. Universal Classes of Hash Functions. JCSS, 1979.

Universal Hash Function

Suppose $\mathcal{H} \subseteq B^A$, where B^A is the set of functions from A to B.

$\mathcal{H}$ is a universal Hash function family if

$$\Pr_{h \in_R \mathcal{H}}[h(x) = h(x')] \leq \frac{1}{|B|}$$

for all $x, x' \in A$ such that $x \neq x'$.

Example: m-wise Independent Hash Function Family

m-wise Independent Hash Function Family

Suppose $\mathcal{H}_{n,k}$ is a set of functions from $\{0,1\}^n$ to $\{0,1\}^k$.

$\mathcal{H}_{n,k}$ is *m-independent* if for all pairwise distinct $x_1, \ldots, x_m \in \{0,1\}^n$ and any $y_1, \ldots, y_m \in \{0,1\}^k$, the following equality is valid

$$\Pr_{h \in_{\mathrm{R}} \mathcal{H}_{n,k}} \left[\bigwedge_{i=1}^{m} h(x_i) = y_i \right] = \frac{1}{2^{mk}}.$$

If $\mathcal{H}_{n,k}$ is m-independent, then $\mathcal{H}_{n,k}$ is m'-independent for every $m' \in [m-1]$.

Pairwise Independent Hash Function Family:

- $\Pr_{h \in_{\mathrm{R}} \mathcal{H}_{n,k}}[h(x) = y] = \frac{1}{2^k}$.
- $\Pr_{h \in_{\mathrm{R}} \mathcal{H}_{n,k}}[h(x)=y \wedge h(x')=y'] = \frac{1}{2^{2k}}$.

Efficient m-wise Independent Hash Function Family

Suppose $a_0, \ldots, a_{m-1} \in \mathbf{F}_{2^n}$, the function $h_{a_0,\ldots,a_{m-1}} : \mathbf{F}_{2^n} \to \mathbf{F}_{2^n}$ is defined as follows:

$$h_{a_0,\ldots,a_{m-1}}(x) = \sum_{j\in\{0,\ldots,m-1\}} a_j x^j.$$

For distinct $x_1, \ldots, x_m \in \mathbf{F}_{2^n}$ and any $y_1, \ldots, y_m \in \mathbf{F}_{2^n}$, equalities $h_{a_0,\ldots,a_{m-1}}(x_1) = y_1$, ..., $h_{a_0,\ldots,a_{m-1}}(x_m) = y_m$ give rise to the following equation system

$$\begin{aligned} a_0 + a_1 x_1 + \ldots + a_{m-2} x_1^{m-2} + a_{m-1} x_1^{m-1} &= y_1, \\ &\vdots \\ a_0 + a_1 x_m + \ldots + a_{m-2} x_m^{m-2} + a_{m-1} x_m^{m-1} &= y_m. \end{aligned} \tag{1}$$

The coefficient matrix is a Vandermonde matrix, hence a unique solution.

Efficient *m*-wise Independent Hash Function Family

If $n > k$, $\mathcal{H}_{n,k}$ is obtained from $\mathcal{H}_{n,n}$ by composing with projection.

If $n < k$, $\mathcal{H}_{n,k}$ is obtained from $\mathcal{H}_{k,k}$ by composing with embedding function.

1. Sipser used these functions to prove **BPP** $\subseteq \Sigma_4^p \cap \Pi_4^p$.
2. Stockmeyer applied them to set lower bound for the first time.
3. Babai exploited them in the study of Arthur-Merlin protocol.

---

1. Sipser. A Complexity Theoretic Approach to Randomness. STOC 1983.
2. Stockmeyer. The Complexity of Approximate Counting. STOC 1984.
3. Babai. Trading Group Theory for Randomness. STOC 1985.

Valiant-Vazirani Theorem

Valiant and Vazirani gave a surprising randomized P-time reduction from SAT to USAT.


1. L. Valiant and V. Vazirani. NP is as Easy as Detecting Unique Solutions. Theoretical Computer Science, 47:85-93, 1986.

UP

UP is the class of unambiguous P-time decision problems.

- $L \in \mathbf{UP}$ iff L is accepted by a P-time NDTM $\mathbb{N}$ such that, for every x, $\mathbb{N}(x)$ has at most one accepting computation path.
- Alternatively we can define **UP** in terms of deterministic TM.

Clearly $\mathbf{P} \subseteq \mathbf{UP} \subseteq \mathbf{NP}$. The class was introduced by Valiant.

1. L. Valiant. Relative Complexity of Checking and Evaluating. Information Processing Letters, 5:20-23, 1976.

Let USAT be the set of CNFs that have unique satisfying assignment. Then USAT $\in$ **UP**.

Formally USAT must be understood as a promise problem.

A promise problem is a generalization of a decision problem where an input is promised to belong to a subset, called the promise, of the set of all possible inputs.

- There is no requirement on the inputs outside the promise set.

Promise problems are introduced in [1]. Many natural problems are actually promise problems.

- Given a Hamiltonian graph, has it got a cycle of even length?
- Factorization referred to in cryptography is a promise problem.

---

1. Even, Selman, Yacobi. The Complexity of Promise Problems with Applications to Publica Key Cryptography. Information and Control, 1984.
2. Goldreich. On Promise Problems. Electronic Colloquium on Computational Complexity, 2005.

Randomized Reduction from **NP** to USAT

Theorem (Valiant and Vazirani, 1986).

There is a P-time PTM $\mathbb{A}$ such that for every n variable formula φ,

$$\begin{aligned} \varphi \in \mathtt{SAT} &\Rightarrow \Pr[\mathbb{A}(\varphi) \in \mathtt{USAT}] \geq 1/8n, \\ \varphi \notin \mathtt{SAT} &\Rightarrow \Pr[\mathbb{A}(\varphi) \in \mathtt{SAT}] = 0. \end{aligned}$$

Corollary. If $\mathtt{USAT} \in$ **RP** then **NP** = **RP**.

To prove Valiant-Vazirani Theorem, we need to construct a P-time PTM $\mathbb{A}$ that reduces an instance in $\mathtt{SAT}$ to an instance in $\mathtt{USAT}$ in a random manner.

Here is the intuition:

- If $\mathtt{SAT} \in \mathbf{P}$ then given $\varphi \in \mathtt{SAT}$ we could construct in P-time a true assignment $x_1 = c_1, \ldots, x_n = c_n$ to φ and obtain $\varphi \wedge (x_1 = c_1 \wedge \ldots \wedge x_n = c_n) \in \mathtt{USAT}$.
- Since we do not know if $\mathtt{SAT} \in \mathbf{P}$, the best we could do is to generate randomly an assignment and conjoin our guess to φ. This is done using hash functions.

Lemma (Valiant and Vazirani, 1986).

Let $\mathcal{H}_{n,k}$ be a collection of pairwise independent hash function from $\{0,1\}^n$ to $\{0,1\}^k$. Let $S \subseteq \{0,1\}^n$ be such that $2^{k-2} \leq |S| < 2^{k-1}$. Then

$$\mathrm{Pr}_{h \in_{\mathrm{R}} \mathcal{H}_{n,k}, y \in_{\mathrm{R}} \{0,1\}^k}[\exists! x {\in} S. h(x) = y] > \frac{1}{8}.$$

- h when restricted to S looks injective if $|S| \ll 2^k$.
- $y \in \{0,1\}^k$ is likely to be covered by $h(S)$ if $|S|$ is comparable to 2^k.

Proof of Valiant-Vazirani Lemma

Fix $y \in \{0,1\}^k$. By assumption $p = \Pr_{h \in_R \mathcal{H}_{n,k}}[h(x) = y] = 2^{-k}$, and for $x \neq x'$,

$$\Pr_{h \in_R \mathcal{H}_{n,k}}[h(x) = y \wedge h(x') = y] = 2^{-2k} = p^2.$$

By inclusion-exclusion principle,

$$\Pr[\exists x \in S. h(x) = y] \quad \geq \quad \sum_{x \in S} \Pr[h(x) = y] - \sum_{x < x'} \Pr\left[\begin{array}{l} h(x) = y, \\ h(x') = y \end{array} \right] \quad = \quad |S|p - \binom{|S|}{2} p^2,$$

and by union bound,

$$\Pr\left[\exists x, x' \in S. \left(\begin{array}{l} x \neq x', \\ h(x) = y, \\ h(x') = y \end{array} \right)\right] \leq \sum_{x < x'} \Pr\left[\begin{array}{l} h(x) = y, \\ h(x') = y \end{array} \right].$$

It follows that

$$\Pr_{h \in_R \mathcal{H}_{n,k}}[\exists! x \in S. h(x) = y] \geq |S|p - 2\binom{|S|}{2} p^2 > \frac{1}{8}.$$

Proof of Valiant-Vazirani Theorem

1. $\mathbb{A}$ chooses $k \in_{\mathrm{R}} \{2, \ldots, n+1\}$ and $h \in_{\mathrm{R}} \mathcal{H}_{n,k}$ randomly.
   - Let S be the set of satisfying assignments of φ.
   - Then $2^{k-2} \leq |S| < 2^{k-1}$ holds with probability $1/n$.

   Consider the formula $\varphi(x_1, \ldots, x_n) \wedge (h(x_1, \ldots, x_n) = 0^k)$.
   - If φ is unsatisfiable, then the formula is unsatisfiable.
   - If φ is satisfiable, then with at least probability $1/8$ there is a unique satisfying assignment that validates the equality.

2. $\mathbb{A}$ gets $\tau(x, y)$ by applying Cook-Levin reduction to h with the requirement $\exists! x_1, \ldots, x_n. \varphi(x_1, \ldots, x_n) \wedge h(x_1, \ldots, x_n) = 0^k$. Here y are introduced by Cook-Levin reduction.

3. Let $\mathbb{A}(\varphi) = \varphi(x) \wedge \tau(x, y)$. If $\varphi(x)$ is satisfiable, then $\Pr[\exists! x, y. \mathbb{A}(\varphi)] \geq 1/8n$.

Valiant-Vazirani Theorem Relativizes

We remark that the construction by $\mathbb{A}$ is independent of φ.

- The construction does not even take a look at φ.
  - The formula φ may contain variables other than $x_1, \ldots, x_n$.
  - The set S in the proof of Valiant-Vazirani Theorem can take the set of all true assignments projected at $x_1, \ldots, x_n$.

Can we boost the correctness probability of the Valiant-Vazirani Theorem from $1/8n$ to over $1/2$?

- We don't know how to union a set of boolean formulae such that it has a unique satisfying assignment if and only if at least one of the formulae has a unique satisfying assignment.

The parity **P** now comes into the picture.

Parity **P**, Counting in $\mathbf{F}_2$

A language L is in complexity class $\oplus$**P**, parity **P**, iff there is a P-time NDTM $\mathbb{N}$ such that $x \in L$ if and only if the number of accepting paths of $\mathbb{N}$ on input x is odd.

- Like **PP**, we see $\oplus$**P** as another decision version of $\sharp$**P**.
- $\oplus$**P** looks at the least significant bit of counting value.
- Obviously **UP** $\subseteq \oplus$ **P**.

The complexity class $\oplus$**P** was introduced by Papadimitriou and Zachos.

---

1. Papadimitriou and Zachos. Two Remarks on the Power of Counting. Lecture Notes in Computer Science 145, 260-276, 1983.

$\oplus$SAT

1. $\oplus$ is the quantifier defined as follows: $\oplus_{x_1,\ldots,x_n}\varphi(x_1,\ldots,x_n)$ is true if and only if the number of assignments to $x_1,\ldots,x_n$ validating $\varphi(x_1,\ldots,x_n)$ is odd. Notice that

$$\oplus_{x_1,\ldots,x_n}\varphi(x_1,\ldots,x_n) \Leftrightarrow \oplus_{x_1}\ldots\oplus_{x_n}\varphi(x_1,\ldots,x_n).$$

2. $\oplus$SAT is the set of all true quantified formulas of the form

$$\oplus_{x_1,\ldots,x_n}\varphi(x_1,\ldots,x_n),$$

   where $\varphi(x_1,\ldots,x_n)$ is quantifier free.

3. $\oplus$SAT is $\oplus$**P**-complete by Cook-Levin reduction.

Randomized Reduction from **NP** to $\oplus\mathtt{SAT}$

Corollary.

There is a P-time PTM $\mathbb{A}$ such that for every n variable formula φ,

$$\begin{aligned} \varphi \in \mathtt{SAT} &\Rightarrow \Pr[\mathbb{A}(\varphi) \in \oplus\mathtt{SAT}] \geq 1/8n, \\ \varphi \notin \mathtt{SAT} &\Rightarrow \Pr[\mathbb{A}(\varphi) \in \oplus\mathtt{SAT}] = 0. \end{aligned}$$

The probability $1/8n$ in the corollary can be boosted significantly, which

- leads to a randomized reduction from **PH** to $\oplus\mathtt{SAT}$.

Toda Theorem

Toda proved a remarkable result in his Gödel Award paper (1998) that problems in **PH** can be solved efficiently using a $\sharp$**P** oracle.

1. Toda. PP is as Hard as the Polynomial-Time Hierarchy. SIAM Journal of Computing, 20:865-877, 1991.

Normalizing Formulas Containing $\oplus$

Let $\sharp\varphi$ denote the number of satisfying assignments of φ.

It is easy to define $\varphi \cdot \psi$ and $\varphi + \psi$ such that

$$\begin{aligned}
\sharp(\varphi(x) \cdot \psi(y)) &= \sharp(\varphi(x))\sharp(\psi(y)), \quad \text{where } x \cap y = \emptyset \\
\sharp(\varphi(x) + \psi(y)) &= \sharp(\varphi(x)) + \sharp(\psi(y)), \quad \text{where } x \subseteq y
\end{aligned}$$

and the size of $\varphi \cdot \psi$ and $\varphi + \psi$ is polynomial in the size of φ, ψ.

We write $\varphi(x_1, \ldots, x_n) + 1$ for $z \wedge \varphi(x_1, \ldots, x_n) \vee \overline{z} \wedge \overline{x_1} \wedge \ldots \wedge \overline{x_n}$.

Normalizing Formulas Containing $\oplus$

The following are obvious:

$$
\begin{array}{rcl}
\oplus_x \varphi(x) \wedge \oplus_y \psi(y) & \Leftrightarrow & \oplus_{x,y}(\varphi \cdot \psi)(x, y), \\
\oplus_x \varphi(x) \vee \oplus_y \psi(y) & \Leftrightarrow & \oplus_{x,y,z}((\varphi + 1) \cdot (\psi + 1) + 1)(x, y, z), \\
\neg \oplus_x \varphi(x) & \Leftrightarrow & \oplus_{x,z}(\varphi + 1)(x, z).
\end{array}
$$

Conclusion:

- $\oplus$ can switch position with $\wedge, \vee, \neg$.
- $\forall$ can be replaced in favour of $\exists$.

Normalizing Parity

Lemma. There is a P-time TM $\mathbb{T}$ such that, for every formula α, the formula $\beta = \mathbb{T}(\alpha, 1^\ell)$ satisfies the following:

$$\alpha \in \oplus\mathtt{SAT} \;\Rightarrow\; \sharp\beta = -1 \pmod{2^{\ell+1}},$$
$$\alpha \notin \oplus\mathtt{SAT} \;\Rightarrow\; \sharp\beta = 0 \pmod{2^{\ell+1}}.$$

It is easy to check that

$$\sharp\tau = -1 \pmod{2^{2^i}} \;\Rightarrow\; \sharp(4\tau^3 + 3\tau^4) = -1 \pmod{2^{2^{i+1}}},$$
$$\sharp\tau = 0 \pmod{2^{2^i}} \;\Rightarrow\; \sharp(4\tau^3 + 3\tau^4) = 0 \pmod{2^{2^{i+1}}}.$$

Let $\alpha_0 = \alpha$ and $\alpha_{i+1} = 4\alpha_i^3 + 3\alpha_i^4$. Let $\beta = \alpha_{\log(\ell+1)}$.

Randomized Reduction from **PH** to $\oplus\mathtt{SAT}$

Lemma. Given m, there exists a $\mathrm{poly}(n, m)$ time probabilistic reduction $\mathbb{F}$ from $i\mathtt{QBF}$ to $\oplus\mathtt{SAT}$ such that

$$\begin{aligned}\psi \in i\mathtt{QBF} \quad &\Rightarrow \quad \Pr[\mathbb{F}(\psi) \in \oplus\mathtt{SAT}] \geq 1 - 2^{-m}, \\ \psi \notin i\mathtt{QBF} \quad &\Rightarrow \quad \Pr[\mathbb{F}(\psi) \in \oplus\mathtt{SAT}] \leq 2^{-m}.\end{aligned}$$

Proof

Suppose $\exists x.\varphi(u,x)$ is true with $|\varphi| = n$.

- By induction, φ can be converted to a $\oplus$ formula $\oplus_z\psi(u,x,z)$ with $|z| = \mathrm{poly}(n)$ and error probability $\leq 2^{-m-1}$. [See the remark on Valiant-Vazirani Theorem.]

The P-time PTM $\mathbb{F}$ is defined inductively as follows:

1. run the Valiant-Vazirani reduction on ψ for $8n(m+1)$ times;
2. let ϕ be the $\bigvee$ of all the new $\oplus$-formulas;
3. turn ϕ into a single $\oplus$ formula $\oplus\varphi'$.

The error probability is bounded by $(1 - 1/8n)^{8n(m+1)} \approx 2^{-m-1}$.

Putting things together, one has the following:

- If $\exists x.\varphi$ is true, then $\oplus\varphi'$ is true with error probability 2^{-m}.
- If $\exists x.\varphi$ is false, then $\oplus\varphi'$ is false.

Toda's key observation is that the above randomized algorithm can be derandomized by counting success rate.

Toda Theorem

Theorem (Toda, 1991). $\mathbf{PH} \subseteq \mathbf{P}^{\sharp\mathbf{P}[1]}$.

We prove that $\mathbf{PH} \subseteq \mathbf{P}^{\sharp\mathtt{SAT}[1]}$.

Proof of Toda Theorem

1. Let $\mathbb{F}$ be a randomized reduction from **PH** to $\oplus$SAT and $m = 2$.
   Think of $\mathbb{F}$ as a TM with an additional R-bit string input r.
   Let $\mathbb{T}$ be the reduction in one of the previous lemmas with $\ell = R + 2$.

---

2. Given QBF ψ, consider the following value

$$\sum_{r \in \{0,1\}^R} \left(\sharp(\mathbb{T}(\mathbb{F}(\psi, r), 1^\ell)) \bmod 2^{\ell+1} \right). \tag{2}$$

   If ψ is true, (2) is in $2^R[-1, -\frac{3}{4}]$. If ψ is false, (2) is in $2^R[-\frac{1}{4}, 0]$.

---

3. By Cook-Levin reduction we get a formula Ψ from $\mathbb{T}(\mathbb{F}(\psi, r), 1^\ell)$ with ψ hardwired.
Then (2) can be obtained by querying the oracle $\sharp$SAT for $\sharp\Psi$.

Toda Theorem is often stated as $\mathbf{PH} \subseteq \mathbf{P}^{\mathbf{PP}}$.

Theorem. $\mathbf{P}^{\mathbf{PP}} = \mathbf{P}^{\sharp\mathbf{P}}$.

Proof.

It is sufficient to prove that $\mathbf{P}^{\natural\mathtt{SAT}} = \mathbf{P}^{\sharp\mathtt{SAT}}$. See the proof of **Theorem**. □

$\langle\varphi, i\rangle \in \natural\mathtt{SAT}$ if more than i assignments make φ true. $\natural\mathtt{SAT}$ is **PP**-complete.

Toda Theorem implies that a question like

"Is this the smallest circuit with the given functionality?"

can be effectively turned into a question of the form

"How many satisfying assignments does this formula have?"

Problem classification, proof technique, and thought provoking results.