# Foundations of Interactive Proofs
# Midterm Homework

We have seen in class that the sumcheck protocol provides an extremely powerful tool for performing *high end* interactive proofs: it can handle any statement from PSPACE. In this homework, we will see that sumchecks, when used correctly, also provide an amazing tool in the *low end* regime: namely, for problems in P that can be solved in some polynomial time $T(n)$, it is sometimes possible to design a sumcheck protocol where a weak client running in time $\ll T(n)$ *delegates* to a more powerful server the task of solving the problem (in time $T(n)$). The goal is for the server to *prove* that it correctly solved the problem using little communication with the client, such that computing the proof should not be much more expensive than solving the problem itself.

**Rules.** You have two weeks to complete the homework. **Due date: Wednesday, January 28, 12:45pm**. It is a long homework: you don't necessarily have to answer every question and can get a good grade without finishing it. Everything is allowed, but I strongly suggest that you try to the best of your ability to solve the questions by yourself before looking for resources that provide hints or solutions: you will benefit much more from thinking hard about it, and it's a good training for the exam.

**Notations.** $\mathbb{F}$ always denotes a finite field of prime order. Given a field $\mathbb{F}$, $\mathbb{F}[X_1, \cdots, X_n]$ denotes the ring of all $n$-variate polynomials with coefficients over $\mathbb{F}$, $\mathbb{F}^{\leq d}[X_1, \cdots, X_n]$ denotes the subset of $n$-variate polynomials of individual degree at most $d$, and $\mathbb{F}^{(\leq d)}[X_1, \cdots, X_n]$ denotes the subset of $n$-variate polynomials of total degree at most $d$ (a monomial $\prod_{i=1}^{n} X_i^{d_i}$ has individual degree $d$ if $d_i \leq d$ for all $i$, and total degree $d$ if $\sum_i d_i = d$). We use bold font $(\mathbf{u}, \mathbf{v}, \mathbf{x})$ to denote vectors over $\mathbb{F}$, but standard font $(u, v, x)$ to denote elements of $\mathbb{F}$ or bitstrings. We typically identify bitstrings $x \in \{0,1\}^n$ with the vector $(x_1, \cdots, x_n)$ of its bits. We write $[n]$ for the set $\{1, 2, \cdots, n\}$. For notational convenience, we always assume below that $n$ is a power of two, and write $m = \log_2(n)$. Given a finite set $S$, we write $x \leftarrow S$ to denote that $x$ is sampled uniformly at random from $S$. We typically identify a polynomial with the function it represents and view $P \in \mathbb{F}[X_1, \cdots, X_n]$ as a function $P : \mathbb{F}^n \to \mathbb{F}$.

## 1 Three Lemmas

**Lemma 1.** *Let $f : \{0,1\}^n \to \mathbb{F}$. There exists a unique polynomial $Q_f \in \mathbb{F}^{\leq 1}[X_1, \cdots, X_n]$ such that for all $\mathbf{x} = (x_1, \cdots, x_n) \in \{0,1\}^n$, $f(\mathbf{x}) = Q_f(\mathbf{x})$.*

**Question 1.** Prove the uniqueness part of Lemma 1.

**Lemma 2.** *Let $S \subseteq \mathbb{F}$ be a subset of $\mathbb{F}$. Let $Q \in \mathbb{F}^{\leq d}[X]$ be a nonzero univariate polynomial of degree at most $d$. Then*

$$\Pr_{x \leftarrow S}[Q(x) = 0] \leq \frac{d}{|S|}.$$

**Question 2.** Prove Lemma 2 (you can use any standard result about polynomials).

**Lemma 3.** *Let $Q \in \mathbb{F}^{(\leq d)}[X_1, \cdots, X_n]$ be a nonzero $n$-variate polynomial of total degree at most $d$. Then*

$$\Pr_{(x_1, \cdots, x_n) \leftarrow \mathbb{F}^n}[Q(x_1, \cdots, x_n) = 0] \leq \frac{d}{|\mathbb{F}|}.$$

**Question 3.** Prove Lemma 3 using Lemma 2.

## 2 Arithmetization of triangle counting: a naive attempt

Consider a simple undirected graph $G = ([n], E)$ (no self-loops) and let $A \in \{0,1\}^{n \times n}$ denote the adjacency matrix of $G$, *i.e.*, the matrix $A$ such that $A_{i,j} = 1$ if $(i,j) \in E$, and $A_{i,j} = 0$ else. From now on, we are interested in the following task: a weak verifier $\mathsf{V}$ is given a graph $G$ and an integer $t \in [n^3]$. The verifier wants to verify the following claim: the graph $G$ contains exactly $t$ triangles. To that end, $\mathsf{V}$ interacts with a powerful but untrusted prover $\mathsf{P}$. Crucially, the verifier must be *optimally efficient*: they should run in time $O(n^2)$ (note that up to a constant, if $G$ is dense, this is the time it takes to read the graph, so we cannot expect a smaller amount of work in general).

**Remark**

Counting triangles is a ubiquitous problem in computer science and in real-world applications. In graph analysis, it allows computing the clustering coefficient of a graph, which measures the "small-world" effect in a social graph (e.g., people who have a friend in common tend to be friends themselves). It can also be used to test graph properties, detect frauds or security issues (e.g., in a transaction graph, triangles reveal circular payments; in social graphs with bots, spikes in the total triangle count correlate with coordinated bot campains), and the hardness of counting triangles is a core problem in fine-grained complexity (together with its generalization, counting $k$-cliques).

**Question 4.** Write a polynomial $P_A : \mathbb{F}^n \to \mathbb{F}$ such that on input $x \in \{0,1\}^n$, $P_A(x) = 1$ iff (1) the Hamming weight of $x$ is exactly 3, and (2) denoting $(i,j,k)$ the indices of the ones in $x$, $(i,j),(j,k)$, and $(k,i)$ all belong to $E$.

**Question 5.** Observe that proving the statement "$G$ contains $t$ triangles" reduces to proving the statement $\sum_{x \in \{0,1\}^n} P_A(x) = t$. When using the sumcheck protocol to prove this statement,

- What is the computational complexity of the verifier?
- What is the communication complexity of the protocol?
- What is the number of rounds of the protocol?

Is this a useful protocol? Justify your claims.

## 3 Arithmetization of triangle counting: a better attempt

The complexity of the naive solution from the previous section is clearly unsatisfying. Intuitively, this stem from the choice of arithmetizing the problem with a high-degree polynomial, which yields inefficient sumchecks. In this section, we derive a much better arithmetization of the triangle counting problem.

**Question 6.** Prove the following statement: for any $(i,j) \in [n]^2$ and $k \in \mathbb{N}$, $(A^k)_{i,j}$ is the total number of paths from $i$ to $j$ of length exactly $k$ in $G$.

**Question 7.** Using the above characterization, give a concise formula for the numbers of triangles in $G$ using powers of $A$.

**(Bonus) Question 8.** Can you suggest an alternative protocol to the naive attempt, with higher communication but much lower computation, and a *single round of communication* from $\mathsf{P}$ to $\mathsf{V}$ (i.e., an MA protocol)? The protocol uses the characterization from question 7 as well as Lemma 2 to reduce verifying the statement to a few matrix-vector products.

Given $a \in \{0,1\}^n$, let $\delta_a : \{0,1\}^n \to \mathbb{F}$ be defined as $\delta_a : x \mapsto \begin{cases} 1 \text{ if } x = a \\ 0 \text{ else} \end{cases}$.

**Question 9.** Provide the arithmetization $Q_{\delta_a}$ of $\delta_a$.

**Question 10.** Use the polynomials $Q_{\delta_a}$ to prove the existential part of Lemma 1.

Given a matrix $M \in \mathbb{F}^{n \times n}$, we now denote $\tilde{M} : \{0,1\}^{2m} \to \mathbb{F}$ the function defined by

$$\tilde{M}(i_1, \cdots, i_m, j_1, \cdots, j_m) = M_{i,j}.$$

We slightly abuse notations and write $Q_M = Q_{\tilde{M}}$ for the arithmetization of $\tilde{M}$ guaranteed by Lemma 1.

**Question 11.** Let $(A, B, C) \in (\mathbb{F}^{n \times n})^3$ be three matrices. Prove that for all $(\mathbf{u}, \mathbf{v}) \in (\mathbb{F}^m)^2$,

$$Q_{ABC}(\mathbf{u}, \mathbf{v}) = \sum_{\substack{i_1, \cdots i_m \in \{0,1\} \\ j_1, \cdots j_m \in \{0,1\}}} Q_A(\mathbf{u}, i_1, \cdots, i_m) \cdot Q_B(i_1, \cdots, i_m, j_1, \cdots, j_m) \cdot Q_C(j_1, \cdots, j_m, \mathbf{v}).$$

**Question 12.** Let $S \subseteq (\{0,1\}^m)^2$ and $t \in [n^3]$. Assume that $\mathsf{V}$ knows $A, B, C$ and is allowed to run in time at most $O(n^2)$. Describe the sumcheck protocol to prove statements of the form

$$\sum_{(i,j) \in S} Q_{ABC}(i,j) = t.$$

**Question 13.** What is the maximum degree in each variable of the sumcheck polynomial? What is the communication complexity of the protocol? And its round complexity? Conclude with how to apply this approach to the triangle counting problem.

**Question 14.** What is the computational complexity of the prover in this triangle counting protocol? If you cannot characterize it precisely, a reasonable upper bound suffices.

**Remark**

A more involved analysis and optimization of the prover shows that it can actually be implemented with *optimal* complexity (as efficient as running the best-possible algorithm for counting triangles) up to constants, but this is outside of the scope of this homework.

**Question 15.** Consider a variant of the previous protocol in which the goal is to prove a statement of the form "The $(i,j)$-th entry of $A^k$ is equal to $t$" (equivalently, the number of length-$k$ paths from $i$ to $j$ in $G$ is $t$). What is the verifier complexity, communication complexity, and round complexity of this variant?

## 4 An even better attempt

Let $k \in \mathbb{N}$ be an integer and consider the arithmetization $Q_{A^k}$ of $A^k$.

**Question 16.** Using the same reasoning as for Question 11, write $Q_{A^k}(\mathbf{u}, \mathbf{v})$ using a sum of terms that depend only on $Q_{A^{k/2}}$ and use this decomposition to describe (at a high-level) a sumcheck protocol that reduces proving a statement of the form $Q_{A^k}(\mathbf{u}, \mathbf{v}) = t$ to proving two statements of the form $Q_{A^{k/2}}(\mathbf{u}_i, \mathbf{v}_i) = t_i$ for $i = 1, 2$.

**Question 17.** Let $P : \mathbb{F}^n \to \mathbb{F}$ denote a multivariate polynomial. Consider a pair of claims of the form

$$P(\mathbf{u}_1, \mathbf{v}_1) = t_1 \qquad\qquad P(\mathbf{u}_2, \mathbf{v}_2) = t_2.$$

Using the line polynomials $\ell_1, \ell_2 : \mathbb{F} \to \mathbb{F}$ defined as $\ell_i(x) = \mathbf{u}_i + (\mathbf{v}_i - \mathbf{u}_i) \cdot x$, design a one-round sumcheck-like protocol that reduces proving the two claims above to proving a single claim of the form $P(\mathbf{u}, \mathbf{v}) = t$.

**Question 18.** Conclude from Question 16 and Question 17: what is the verifier complexity, communication complexity, and round complexity of the full improved sumcheck protocol for proving a statement of the form $Q_{A^k}(\mathbf{u}, \mathbf{v}) = t$? Compare to the protocol of Question 15.

**Question 19.** Let $s$ be a polynomial, let $M$ denote an $s(n)$-space Turing Machine, and let $x \in \{0,1\}^n$ denote an input. Let $N = 2^{c \cdot s(n)}$, where $c$ is a constant such that the configuration graph $G_{M,x}$ of $M$ on input $x$ has at most $2^{c \cdot s(n)}$ nodes (see tutorial 2). Suppose that the configuration graph has a single accepting configuration. Let 1 denote the index of the starting configuration and $N$ denote the index of the accepting configuration. What is $(A^N)_{1,N}$? Use the result of Question 18 to provide a new direct proof that $\mathsf{IP} = \mathsf{PSPACE}$.