How To Test If a Polynomial Is Identically Zero?

- $\det(A^G)$ is a polynomial in n^2 variables.
- There are exponentially many terms in $\det(A^G)$.
- Expanding the determinant polynomial is not feasible.
  - Too many terms.
- If $\det(A^G) \equiv 0$, then it remains zero if we substitute *arbitrary* integers for the variables $x_{11}, \ldots, x_{nn}$.
- But what is the likelihood of obtaining a zero when $\det(A^G) \not\equiv 0$?

Number of Roots of a Polynomial

Lemma 59 (Schwartz (1980)) *Let* $p(x_1, x_2, \ldots, x_m) \not\equiv 0$ *be a polynomial in* m *variables each of degree at most* d*. Let* $M \in \mathbb{Z}^+$*. Then the number of* m*-tuples*

$$(x_1, x_2, \ldots, x_m) \in \{0, 1, \ldots, M-1\}^m$$

such that $p(x_1, x_2, \ldots, x_m) = 0$ *is*

$$\leq mdM^{m-1}.$$

- By induction on m (consult the textbook).

Density Attack

- The density of roots in the domain is at most

$$\frac{mdM^{m-1}}{M^m} = \frac{md}{M}. \tag{8}$$

- So suppose $p(x_1, x_2, \ldots, x_m) \not\equiv 0$.
- Then a random

$$(x_1, x_2, \ldots, x_m) \in \{ 0, 1, \ldots, M-1 \}^m$$

  has a probability of $\leq md/M$ of being a root of p.
- Note that M is under our control!
  - One can raise M to lower the error probability, e.g.

Density Attack (concluded)

Here is a sampling algorithm to test if $p(x_1, x_2, \ldots, x_m) \not\equiv 0$.

```
Choose $i_1, \ldots, i_m$ from $\{0, 1, \ldots, M-1\}$ randomly;
if $p(i_1, i_2, \ldots, i_m) \neq 0$ then
    return "$p$ is not identically zero";
else
    return "$p$ is (probably) identically zero";
end if
```

A Randomized Bipartite Perfect Matching Algorithm[a]

We now return to the original problem of bipartite perfect matching.

```
Choose $n^2$ integers $i_{11}, \ldots, i_{nn}$ from $\{0, 1, \ldots, 2n^2 - 1\}$ randomly; {So $M = 2n^2$.}
Calculate $\det(A^G(i_{11}, \ldots, i_{nn}))$ by Gaussian elimination;
if $\det(A^G(i_{11}, \ldots, i_{nn})) \neq 0$ then
    return "$G$ has a perfect matching";
else
    return "$G$ has no perfect matchings";
end if
```

---

[a] Lovász (1979). According to Paul Erdős, Lovász wrote his first significant paper "at the ripe old age of 17."

Analysis

- If G has no perfect matchings, the algorithm will always be correct as $\det(A^G(i_{11}, \ldots, i_{nn})) = 0$.
- Suppose G has a perfect matching.
  - The algorithm will answer incorrectly with probability at most $md/M = 0.5$ with $m = n^2$, $d = 1$ and $M = 2n^2$ in Eq. (8) on p. 473.
  - Run the algorithm *independently* k times.
  - Output "G has no perfect matchings" if and only if *all* say no.
  - The error probability is now reduced to at most 2^{-k}.

Lószló Lovász (1948–)

Remarks[a]

- Note that we are calculating

$$\text{prob}[\,\text{algorithm answers “no”}\,|\,G \text{ has no perfect matchings}\,],$$

$$\text{prob}[\,\text{algorithm answers “yes”}\,|\,G \text{ has a perfect matching}\,].$$

- We are *not* calculating[b]

$$\text{prob}[\,G \text{ has no perfect matchings}\,|\,\text{algorithm answers “no”}\,],$$

$$\text{prob}[\,G \text{ has a perfect matching}\,|\,\text{algorithm answers “yes”}\,].$$

---

[a] Thanks to a lively class discussion on May 1, 2008.

[b] *Numerical Recipes in C* (1988), “[As] we already remarked, statistics is *not* a branch of mathematics!”

But How Large Can det$(A^G(i_{11}, \ldots, i_{nn}))$ Be?

- It is at most
$$n!\left(2n^2\right)^n.$$
- Stirling's formula says $n! \sim \sqrt{2\pi n}\,(n/e)^n$.
- Hence
$$\log_2 \det(A^G(i_{11}, \ldots, i_{nn})) = O(n \log_2 n)$$
bits are sufficient for representing the determinant.
- We skip the details about how to make sure that all *intermediate* results are of polynomial sizes.

An Intriguing Question[a]

- Is there an $(i_{11}, \ldots, i_{nn})$ that will always give correct answers for the algorithm on p. 475?
- A theorem on p. 571 shows that such an $(i_{11}, \ldots, i_{nn})$ exists!
  - Whether it can be found efficiently is another matter.
- Once $(i_{11}, \ldots, i_{nn})$ is available, the algorithm can be made deterministic.

---

[a] Thanks to a lively class discussion on November 24, 2004.

Randomization vs. Nondeterminism[a]

- What are the differences between randomized algorithms and nondeterministic algorithms?
- One can think of a randomized algorithm as a nondeterministic algorithm but with a probability associated with every guess/branch.
- So each computation path of a randomized algorithm has a probability associated with it.

---

[a] Contributed by Mr. Olivier Valery (`D01922033`) and Mr. Hasan Alhasan (`D01922034`) on November 27, 2012.

Monte Carlo Algorithms[a]

- The randomized bipartite perfect matching algorithm is called a **Monte Carlo algorithm** in the sense that
  - If the algorithm finds that a matching exists, it is always correct (no **false positives**).
  - If the algorithm answers in the negative, then it may make an error (**false negatives**).

---

[a] Metropolis and Ulam (1949).

Monte Carlo Algorithms (concluded)

- The algorithm makes a false negative with probability ≤ 0.5.[a]
  - Note this probability refers to[b]

$$\text{prob}[\,\text{algorithm answers "no"} \,|\, G \text{ has a perfect matching}\,]$$

not

$$\text{prob}[\,G \text{ has a perfect matching} \,|\, \text{algorithm answers "no"}\,].$$

- This probability is *not* over the space of all graphs or determinants, but *over* the algorithm's own coin flips.
  - It holds for *any* bipartite graph.

---

[a] Equivalently, among the coin flip sequences, at most half of them lead to the wrong answer.

[b] In general, prob[algorithm answers "no" | input is a "yes" instance].

The Markov Inequality[a]

Lemma 60 *Let x be a random variable taking nonnegative integer values. Then for any $k > 0$,*

$$\text{prob}[\, x \geq kE[\, x\,]\,] \leq 1/k.$$

- Let p_i denote the probability that $x = i$.

$$\begin{aligned} E[\, x\,] &= \sum_i i p_i \\ &= \sum_{i < kE[\, x\,]} i p_i + \sum_{i \geq kE[\, x\,]} i p_i \\ &\geq kE[\, x\,] \times \text{prob}[x \geq kE[\, x\,]]. \end{aligned}$$

---

[a] Andrei Andreyevich Markov (1856–1922).

Andrei Andreyevich Markov (1856–1922)

An Application of Markov's Inequality

- Suppose algorithm C runs in expected time $T(n)$ and always gives the right answer.
- Consider an algorithm that runs C for time $kT(n)$ and rejects the input if C does not stop within the time bound.
- By Markov's inequality, this new algorithm runs in time $kT(n)$ and gives the wrong answer with probability $\leq 1/k$.

An Application of Markov's Inequality (concluded)

- By running this algorithm m times (the total running time is $mkT(n)$), we reduce the error probability to $\leq k^{-m}$.[a]
- Suppose, instead, we run the algorithm for the same running time $mkT(n)$ once and rejects the input if it does not stop within the time bound.
- By Markov's inequality, this new algorithm gives the wrong answer with probability $\leq 1/(mk)$.
- This is much worse than the previous algorithm's error probability of $\leq k^{-m}$ for the same amount of time.

---

[a] With the same input. Thanks to a question on December 7, 2010.

FSAT for k-SAT Formulas (p. 453)

- Let $\phi(x_1, x_2, \ldots, x_n)$ be a k-SAT formula.
- If ϕ is satisfiable, then return a satisfying truth assignment.
- Otherwise, return "no."
- We next propose a randomized algorithm for this problem.

A Random Walk Algorithm for ϕ in CNF Form

```
Start with an *arbitrary* truth assignment $T$;
for $i = 1, 2, \ldots, r$ do
    if $T \models \phi$ then
        return "$\phi$ is satisfiable with $T$";
    else
        Let $c$ be an unsatisfied clause in $\phi$ under $T$; {All of
        its literals are false under $T$.}
        Pick any $x$ of these literals *at random*;
        Modify $T$ to make $x$ true;
    end if
end for
return "$\phi$ is unsatisfiable";
```

3SAT vs. 2SAT Again

- Note that if ϕ is unsatisfiable, the algorithm will not refute it.
- The random walk algorithm needs expected exponential time for 3SAT.
  - In fact, it runs in expected $O((1.333\cdots+\epsilon)^n)$ time with $r = 3n$,[a] much better than $O(2^n)$.[b]
- We will show immediately that it works well for 2SAT.
- The state of the art as of 2006 is expected $O(1.322^n)$ time for 3SAT and expected $O(1.474^n)$ time for 4SAT.[c]

---

[a] Use this setting per run of the algorithm.

[b] Schöning (1999).

[c] Kwama and Tamaki (2004); Rolf (2006).

Random Walk Works for 2SAT[a]

Theorem 61 *Suppose the random walk algorithm with* $r = 2n^2$ *is applied to any satisfiable* 2SAT *problem with* n *variables. Then a satisfying truth assignment will be discovered with probability at least 0.5.*

- Let $\hat{T}$ be a truth assignment such that $\hat{T} \models \phi$.
- Assume our starting T differs from $\hat{T}$ in i values.
  - Their Hamming distance is i.
  - Recall T is arbitrary.

---

[a] Papadimitriou (1991).

The Proof

- Let $t(i)$ denote the expected number of repetitions of the flipping step[a] until a satisfying truth assignment is found.
- It can be shown that $t(i)$ is finite.
- $t(0) = 0$ because it means that $T = \hat{T}$ and hence $T \models \phi$.
- If $T \neq \hat{T}$ or any other satisfying truth assignment, then we need to flip the coin at least once.
- We flip a coin to pick among the 2 literals of a clause not satisfied by the present T.
- At least one of the 2 literals is true under $\hat{T}$ because $\hat{T}$ satisfies all clauses.

---

[a] That is, Statement 7.

The Proof (continued)

- So we have at least 0.5 chance of moving closer to $\hat{T}$.
- Thus
$$t(i) \leq \frac{t(i-1)+t(i+1)}{2}+1$$
for $0 < i < n$.
  - Inequality is used because, for example, T may differ from $\hat{T}$ in both literals.
- It must also hold that
$$t(n) \leq t(n-1)+1$$
because at $i = n$, we can only decrease i.

The Proof (continued)

- Now, put the necessary relations together:

$$\begin{aligned} t(0) &= 0, && (9) \\ t(i) &\leq \frac{t(i-1)+t(i+1)}{2}+1, \quad 0<i<n, && (10) \\ t(n) &\leq t(n-1)+1. && (11) \end{aligned}$$

- Technically, this is a one-dimensional random walk with an absorbing barrier at $i = 0$ and a reflecting barrier at $i = n$ (if we replace "$\leq$" with "$=$").[a]

---

[a] The proof in the textbook does exactly that. But a student pointed out difficulties with this proof technique on December 8, 2004. So our proof here uses the original inequalities.

The Proof (continued)

- Add up the relations for $2t(1), 2t(2), 2t(3), \ldots, 2t(n-1), t(n)$ to obtain[a]

$$\begin{aligned} & 2t(1) + 2t(2) + \cdots + 2t(n-1) + t(n) \\ \leq\; & t(0) + t(1) + 2t(2) + \cdots + 2t(n-2) + 2t(n-1) + t(n) \\ & +2(n-1) + 1. \end{aligned}$$

- Simplify it to yield

$$t(1) \leq 2n - 1. \tag{12}$$

---

[a] Adding up the relations for $t(1), t(2), t(3), \ldots, t(n-1)$ will also work, thanks to Mr. Yen-Wu Ti (D91922010).

The Proof (continued)

- Add up the relations for $2t(2), 2t(3), \ldots, 2t(n-1), t(n)$ to obtain

$$\begin{aligned} & 2t(2) + \cdots + 2t(n-1) + t(n) \\ \leq\ & t(1) + t(2) + 2t(3) + \cdots + 2t(n-2) + 2t(n-1) + t(n) \\ & +2(n-2) + 1. \end{aligned}$$

- Simplify it to yield

$$t(2) \leq t(1) + 2n - 3 \leq 2n - 1 + 2n - 3 = 4n - 4$$

by Eq. (12) on p. 495.

The Proof (continued)

- Continuing the process, we shall obtain
$$t(i) \leq 2in - i^2.$$
- The worst upper bound happens when $i = n$, in which case
$$t(n) \leq n^2.$$
- We conclude that
$$t(i) \leq t(n) \leq n^2$$
for $0 \leq i \leq n$.

The Proof (concluded)

- So the expected number of steps is at most n^2.
- The algorithm picks $r = 2n^2$.
  - This amounts to invoking the Markov inequality (p. 484) with $k = 2$, resulting in a probability of 0.5.[a]
- The proof does *not* yield a polynomial bound for 3SAT.[b]

---

[a] Recall p. 486.

[b] Contributed by Mr. Cheng-Yu Lee (`R95922035`) on November 8, 2006.

Christos Papadimitriou (1949–)

Boosting the Performance

- We can pick $r = 2mn^2$ to have an error probability of
$$\leq \frac{1}{2m}$$
by Markov's inequality.
- Alternatively, with the same running time, we can run the "$r = 2n^2$" algorithm m times.
- The error probability is now reduced to
$$\leq 2^{-m}.$$

Primality Tests

- PRIMES asks if a number N is a prime.
- The classic algorithm tests if $k \mid N$ for $k = 2, 3, \ldots, \sqrt{N}$.
- But it runs in $\Omega(2^{(\log_2 N)/2})$ steps.

Primality Tests (concluded)

- Suppose $N = PQ$ is a product of 2 distinct primes.
- The probability of success of the density attack (p. 434) is
$$\approx \frac{2}{\sqrt{N}}$$
when $P \approx Q$.
- This probability is exponentially small in terms of the input length $\log_2 N$.

The Fermat Test for Primality

Fermat's "little" theorem (p. 437) suggests the following primality test for any given number N:

```
Pick a number $a$ randomly from $\{1, 2, \ldots, N-1\}$;
if $a^{N-1} \neq 1 \bmod N$ then
    return "$N$ is composite";
else
    return "$N$ is a prime";
end if
```

The Fermat Test for Primality (concluded)

- **Carmichael numbers** are composite numbers that will pass the Fermat test for *all* $a \in \{1, 2, \ldots, N-1\}$.[a]
  - The Fermat test will return "N is a prime" for all Carmichael numbers N.
- Unfortunately, there are infinitely many Carmichael numbers.[b]
- In fact, the number of Carmichael numbers less than N exceeds $N^{2/7}$ for N large enough.
- So the Fermat test is an incorrect algorithm for PRIMES.

---

[a] Carmichael (1910).

[b] Alford, Granville, and Pomerance (1992).

Square Roots Modulo a Prime

- Equation $x^2 = a \bmod p$ has at most two (distinct) roots by Lemma 57 (p. 442).
  - The roots are called **square roots**.
  - Numbers a with square roots *and* $\gcd(a, p) = 1$ are called **quadratic residues**.
    - They are

$$1^2 \bmod p, 2^2 \bmod p, \ldots, (p-1)^2 \bmod p.$$

- We shall show that a number either has two roots or has none, and testing which is the case is trivial.[a]

---

[a] But no efficient *deterministic* general-purpose square-root-extracting algorithms are known yet.

Euler's Test

Lemma 62 (Euler) *Let p be an odd prime and $a \neq 0 \bmod p$.*

1. *If*
$$a^{(p-1)/2} = 1 \bmod p,$$
then $x^2 = a \bmod p$ has two roots.

2. *If*
$$a^{(p-1)/2} \neq 1 \bmod p,$$
then
$$a^{(p-1)/2} = -1 \bmod p$$
and $x^2 = a \bmod p$ has no roots.

The Proof (continued)

- Let r be a primitive root of p.
- By Fermat's "little" theorem,

$$r^{(p-1)/2}$$

  is a square root of 1.
- So

$$r^{(p-1)/2} = 1 \text{ or } -1 \bmod p.$$

- But as r is a primitive root, $r^{(p-1)/2} \neq 1 \bmod p$.
- Hence

$$r^{(p-1)/2} = -1 \bmod p.$$

The Proof (continued)

- Let $a = r^k \bmod p$ for some k.
- Then

$$1 = a^{(p-1)/2} = r^{k(p-1)/2} = \left[r^{(p-1)/2} \right]^k = (-1)^k \bmod p.$$

- So k must be even.
- Suppose $a = r^{2j}$ for some $1 \leq j \leq (p-1)/2$.
- Then $a^{(p-1)/2} = r^{j(p-1)} = 1 \bmod p$, and a's two *distinct* roots are $r^j, -r^j (= r^{j+(p-1)/2} \bmod p)$.
  - If $r^j = -r^j \bmod p$, then $2r^j = 0 \bmod p$, which implies $r^j = 0 \bmod p$, a contradiction.

The Proof (continued)

- As $1 \leq j \leq (p-1)/2$, there are $(p-1)/2$ such a's.
- Each such a has 2 distinct square roots.
- The square roots of all the a's are distinct.
  - The square roots of different a's must be different.
- Hence the set of *square roots* is $\{1, 2, \ldots, p-1\}$.
- As a result, $a = r^{2j}$, $1 \leq j \leq (p-1)/2$, exhaust all the quadratic residues.

The Proof (concluded)

- If $a = r^{2j+1}$, then it has no roots because all the square roots have been taken.
- Now,

$$a^{(p-1)/2} = \left[r^{(p-1)/2} \right]^{2j+1} = (-1)^{2j+1} = -1 \bmod p.$$

The Legendre Symbol[a] and Quadratic Residuacity Test

- By Lemma 62 (p. 506) $a^{(p-1)/2} \bmod p = \pm 1$ for $a \neq 0 \bmod p$.
- For odd prime p, define the **Legendre symbol** $(a \,|\, p)$ as

$$(a \,|\, p) = \begin{cases} 0 & \text{if } p \,|\, a, \\ 1 & \text{if } a \text{ is a quadratic residue modulo } p, \\ -1 & \text{if } a \text{ is a } \textbf{quadratic nonresidue} \text{ modulo } p. \end{cases}$$

- Euler's test (p. 506) implies

$$a^{(p-1)/2} = (a \,|\, p) \bmod p$$

for any odd prime p and any integer a.

- Note that $(ab|p) = (a|p)(b|p)$.

---

[a] Andrien-Marie Legendre (1752–1833).

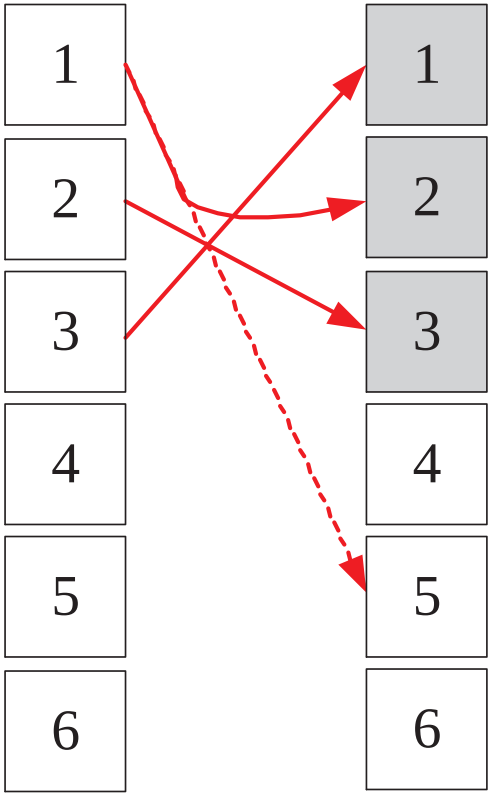
Gauss's Lemma

Lemma 63 (Gauss) *Let p and q be two odd primes. Then $(q|p) = (-1)^m$, where m is the number of residues in $R = \{\, iq \bmod p : 1 \leq i \leq (p-1)/2 \,\}$ that are greater than $(p-1)/2$.*

- All residues in R are distinct.
  - If $iq = jq \bmod p$, then $p|(j-i)\, q$ or $p|q$.
  - But neither is possible.
- No two elements of R add up to p.
  - If $iq + jq = 0 \bmod p$, then $p|(i+j)$ or $p|q$.
  - But neither is possible.

The Proof (continued)

- Replace each of the m elements $a \in R$ such that $a > (p-1)/2$ by $p-a$.
  - This is equivalent to performing $-a \bmod p$.
- Call the resulting set of residues R'.
- All numbers in R' are at most $(p-1)/2$.
- In fact, $R' = \{1, 2, \ldots, (p-1)/2\}$ (see illustration next page).
  - Otherwise, two elements of R would add up to p, which has been shown to be impossible.



$p = 7$ and $q = 5$.

The Proof (concluded)

- Alternatively, $R' = \{\pm iq \bmod p : 1 \leq i \leq (p-1)/2\}$, where exactly m of the elements have the minus sign.
- Take the product of all elements in the two representations of R'.
- So
$$[(p-1)/2]! = (-1)^m q^{(p-1)/2} [(p-1)/2]! \bmod p.$$
- Because $\gcd([(p-1)/2]!, p) = 1$, the above implies
$$1 = (-1)^m q^{(p-1)/2} \bmod p.$$

Legendre's Law of Quadratic Reciprocity[a]

- Let p and q be two odd primes.
- The next result says their Legendre symbols are distinct if and only if both numbers are 3 mod 4.

Lemma 64 (Legendre (1785), Gauss)

$$(p|q)(q|p) = (-1)^{\frac{p-1}{2} \frac{q-1}{2}}.$$

---

[a] First stated by Euler in 1751. Legendre (1785) did not give a correct proof. Gauss proved the theorem when he was 19. He gave at least 6 different proofs during his life. The 152nd proof appeared in 1963.

The Proof (continued)

- Sum the elements of R' in the previous proof in mod2.
- On one hand, this is just $\sum_{i=1}^{(p-1)/2} i \bmod 2$.
- On the other hand, the sum equals

$$\begin{aligned} & mp + \sum_{i=1}^{(p-1)/2} \left(iq - p \left\lfloor \frac{iq}{p} \right\rfloor \right) \bmod 2 \\ = \quad & mp + \left(q \sum_{i=1}^{(p-1)/2} i - p \sum_{i=1}^{(p-1)/2} \left\lfloor \frac{iq}{p} \right\rfloor \right) \bmod 2. \end{aligned}$$

  - m of the $iq \bmod p$ are replaced by $p - iq \bmod p$.
  - But signs are irrelevant under mod2.
  - m is as in Lemma 63 (p. 512).

The Proof (continued)

- Ignore odd multipliers to make the sum equal

$$m + \left(\sum_{i=1}^{(p-1)/2} i - \sum_{i=1}^{(p-1)/2} \left\lfloor \frac{iq}{p} \right\rfloor \right) \bmod 2.$$

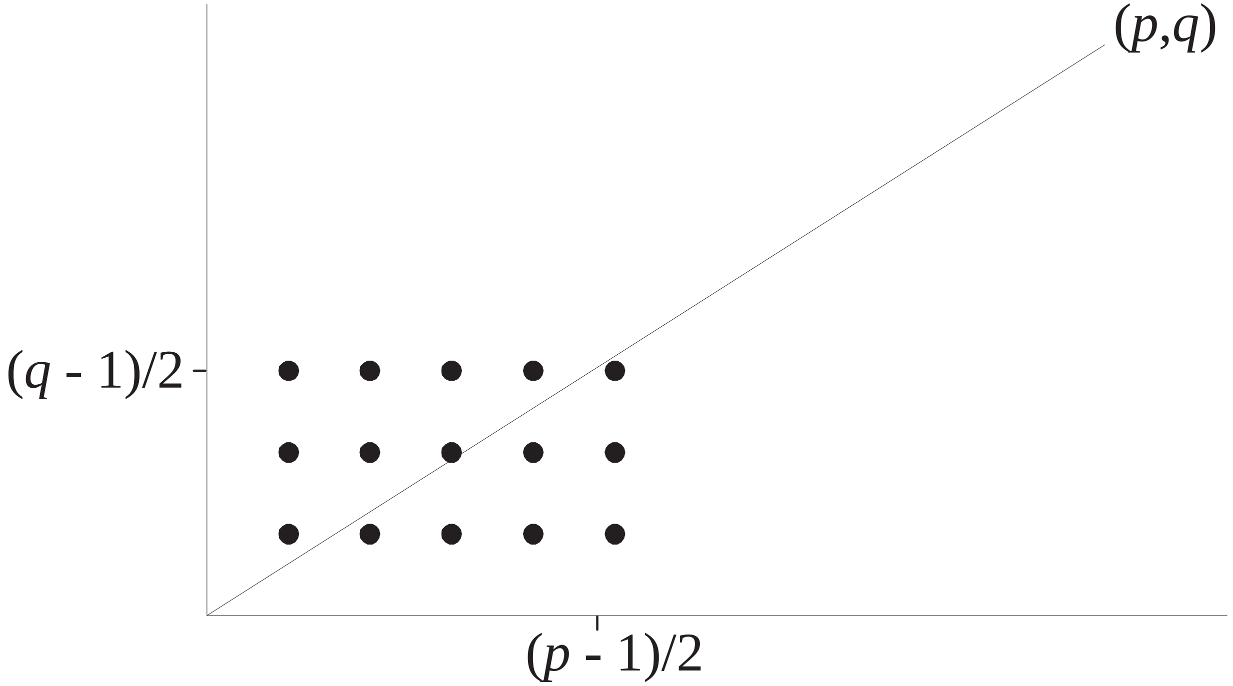
- Equate the above with $\sum_{i=1}^{(p-1)/2} i \bmod 2$ to obtain

$$m = \sum_{i=1}^{(p-1)/2} \left\lfloor \frac{iq}{p} \right\rfloor \bmod 2.$$

The Proof (concluded)

- $\sum_{i=1}^{(p-1)/2} \lfloor \frac{iq}{p} \rfloor$ is the number of integral points *below* the line
$$y = (q/p)\, x$$
for $1 \leq x \leq (p-1)/2$.
- Gauss's lemma (p. 512) says $(q|p) = (-1)^m$.
- Repeat the proof with p and q reversed.
- Then $(p|q) = (-1)^{m'}$, where m' is the number of integral points *above* the line $y = (q/p)\, x$ for $1 \leq y \leq (q-1)/2$.
- As a result, $(p|q)(q|p) = (-1)^{m+m'}$.
- But $m + m'$ is the total number of integral points in the $[1, \frac{p-1}{2}] \times [1, \frac{q-1}{2}]$ rectangle, which is $\frac{p-1}{2} \frac{q-1}{2}$.

Eisenstein's Rectangle



Above, $p = 11$ and $q = 7$.

The Jacobi Symbol[a]

- The Legendre symbol only works for odd *prime* moduli.
- The **Jacobi symbol** $(a \mid m)$ extends it to cases where m is not prime.
- Let $m = p_1 p_2 \cdots p_k$ be the prime factorization of m.
- When $m > 1$ is odd and $\gcd(a, m) = 1$, then
$$(a|m) = \prod_{i=1}^{k} (a \mid p_i).$$
  - Note that the Jacobi symbol equals ± 1.
  - It reduces to the Legendre symbol when m is a prime.
- Define $(a \mid 1) = 1$.

---

[a] Carl Jacobi (1804–1851).

Properties of the Jacobi Symbol

The Jacobi symbol has the following properties, for arguments for which it is defined.

1. $(ab \,|\, m) = (a \,|\, m)(b \,|\, m)$.
2. $(a \,|\, m_1 m_2) = (a \,|\, m_1)(a \,|\, m_2)$.
3. If $a = b \bmod m$, then $(a \,|\, m) = (b \,|\, m)$.
4. $(-1 \,|\, m) = (-1)^{(m-1)/2}$ (by Lemma 63 on p. 512).
5. $(2 \,|\, m) = (-1)^{(m^2-1)/8}$.[a]
6. If a and m are both odd, then
   $(a \,|\, m)(m \,|\, a) = (-1)^{(a-1)(m-1)/4}$.

---

[a] By Lemma 63 (p. 512) and some parity arguments.

Properties of the Jacobi Symbol (concluded)

- These properties allow us to calculate the Jacobi symbol *without* factorization.
- This situation is similar to the Euclidean algorithm.
- Note also that $(a \mid m) = 1/(a \mid m)$ because $(a \mid m) = \pm 1$.[a]

---

[a] Contributed by Mr. Huang, Kuan-Lin (`B96902079`, `R00922018`) on December 6, 2011.

Calculation of $(2200|999)$

$$
\begin{aligned}
(202|999) &= (2|999)(101|999) \\
&= (-1)^{(999^2-1)/8}(101|999) \\
&= (-1)^{124750}(101|999) = (101|999) \\
&= (-1)^{(100)(998)/4}(999|101) = (-1)^{24950}(999|101) \\
&= (999|101) = (90|101) = (-1)^{(101^2-1)/8}(45|101) \\
&= (-1)^{1275}(45|101) = -(45|101) \\
&= -(-1)^{(44)(100)/4}(101|45) = -(101|45) = -(11|45) \\
&= -(-1)^{(10)(44)/4}(45|11) = -(45|11) \\
&= -(1|11) = -1.
\end{aligned}
$$

A Result Generalizing Proposition 10.3 in the Textbook

Theorem 65 *The group of set* $\Phi(n)$ *under multiplication* mod n *has a primitive root if and only if* n *is either 1, 2, 4,* p^k*, or* $2p^k$ *for some nonnegative integer* k *and and odd prime* p*.*

This result is essential in the proof of the next lemma.

The Jacobi Symbol and Primality Test[a]

Lemma 66 *If* $(M|N) = M^{(N-1)/2} \bmod N$ *for all* $M \in \Phi(N)$*, then* N *is a prime. (Assume* N *is odd.)*

- Assume $N = mp$, where p is an odd prime, $\gcd(m, p) = 1$, and $m > 1$ (not necessarily prime).
- Let $r \in \Phi(p)$ such that $(r \,|\, p) = -1$.
- The Chinese remainder theorem says that there is an $M \in \Phi(N)$ such that

$$\begin{aligned} M &= r \bmod p, \\ M &= 1 \bmod m. \end{aligned}$$

---

[a] Mr. Clement Hsiao (`B4506061`, `R88526067`) pointed out that the textbook's proof for Lemma 11.8 is incorrect in January 1999 while he was a senior.

The Proof (continued)

- By the hypothesis,

$$M^{(N-1)/2} = (M \,|\, N) = (M \,|\, p)(M \,|\, m) = -1 \bmod N.$$

- Hence

$$M^{(N-1)/2} = -1 \bmod m.$$

- But because $M = 1 \bmod m$,

$$M^{(N-1)/2} = 1 \bmod m,$$

  a contradiction.

The Proof (continued)

- Second, assume that $N = p^a$, where p is an odd prime and $a \geq 2$.
- By Theorem 65 (p. 525), there exists a primitive root r modulo p^a.
- From the assumption,

$$M^{N-1} = \left[M^{(N-1)/2} \right]^2 = (M|N)^2 = 1 \bmod N$$

for all $M \in \Phi(N)$.

The Proof (continued)

- As $r \in \Phi(N)$ (prove it), we have
$$r^{N-1} = 1 \bmod N.$$
- As r's exponent modulo $N = p^a$ is $\phi(N) = p^{a-1}(p-1)$,
$$p^{a-1}(p-1) \,|\, (N-1),$$
which implies that $p \,|\, (N-1)$.
- But this is impossible given that $p \,|\, N$.

The Proof (continued)

- Third, assume that $N = mp^a$, where p is an odd prime, $\gcd(m, p) = 1$, $m > 1$ (not necessarily prime), and a is even.
- The proof mimics that of the second case.
- By Theorem 65 (p. 525), there exists a primitive root r modulo p^a.
- From the assumption,

$$M^{N-1} = \left[M^{(N-1)/2} \right]^2 = (M|N)^2 = 1 \bmod N$$

for all $M \in \Phi(N)$.

The Proof (continued)

- In particular,
$$M^{N-1} = 1 \bmod p^a \tag{13}$$
for all $M \in \Phi(N)$.
- The Chinese remainder theorem says that there is an $M \in \Phi(N)$ such that
$$\begin{aligned} M &= r \bmod p^a, \\ M &= 1 \bmod m. \end{aligned}$$
- Because $M = r \bmod p^a$ and Eq. (13),
$$r^{N-1} = 1 \bmod p^a.$$

The Proof (concluded)

- As r's exponent modulo $N = p^a$ is $\phi(N) = p^{a-1}(p-1)$,

  $$p^{a-1}(p-1) \mid (N-1),$$

  which implies that $p \mid (N-1)$.

- But this is impossible given that $p \mid N$.