Algorithms
Professor John Reif

ALG 4.3

Hashing Polynomials and Algebraic Expressions:

(a) Identity Testing of Polynomials
(b) Applications of Polynomial Hashing
(c) Hashing Classes of Algebraic Expressions

Reading Selection:

Handout: Ibarra & Moran, "Probabilistic Algorithms for Deciding Equavalence of Straight-Line Programs", JACM, Vol. 30, No. 1, pp. 217-228, Jan. 1983.

Main Goal of Lecture:

Develop techniques for testing equality of Expressions

test $\varepsilon_1 = \varepsilon_2$?

by using test

$$\text{hash}\left(\varepsilon_1\right) = \text{hash}\left(\varepsilon_2\right)?$$

Goals:

(1) provable bounds on error probability
(2) applicable to largest possible class of expressions

Definitions:

polynomial expression:
1 or any variable, or integer, or
$\alpha+\beta$, $\alpha-\beta$, $\alpha\cdot\beta$, or $\alpha\uparrow\kappa$, where

α,β are polynomial expressions, and
κ is a positive integer.

***Straight Line* Pr*ogram* Π: *Input* $x_1,\ldots,x_n$**

sequence assignments--

$$\textit{length } (\theta)\ \textit{assignments}\ \begin{cases} x_{n+1} \leftarrow x_{i_1}\ \theta_1\ x_{j_1} \\ x_{n+2} \leftarrow x_{i_2}\ \theta_2\ x_{j_2} \\ \qquad\vdots \end{cases}$$

output x_L where $L = \text{length}(\Pi)$.

allow operations $\theta_\kappa \in \{+,-,\cdot,\uparrow\}$

$\Pi(x_1,\ldots,x_n)$ denotes output value.

Notes:

(1) **Given a polynomial expression α, can construct a straight-line program of size linear in input polynomial α.**

(2) **A straight-line program**

$$\Pi(x_1,\ldots,x_n)$$

will yield a polynomial expression α_Π with integer coefficients where

$$\textbf{degree}\,(\alpha_\Pi) \le 2^{\text{length}(\Pi)}$$

If $\Pi(x_1,\ldots,x_n)$ is a program over Q,

$$|\Pi(x_1,\ldots,x_n)| \le 2^{2\,\text{length}(\Pi)}$$

can be proved by induction on length (Π).

basis: true for case length $\left(\prod\ \right) = 0$

induction step: if true for length $\left(\prod\ \right) \le k-1$ and

$$\prod(x_1,\ldots,x_k) =$$
$$\prod{}_1(x_1\ldots x_k)\theta_k\prod{}_2(x_1\ldots x_k),$$

then $\left|\prod(x_1\ldots x_k)\right| \le 2^{2\,length\left(\prod\ \right)}$.

Q.E.D.

Let Q be an infinite field.

Let $P(x_1,\ldots,x_n)$ be nonzero polynomial degree d.

<u>Lemma</u>: If $A \le Q$ size $\kappa = |A| > d$, then $\exists$ at least $(\kappa - d)^n$ elements $\bar{a} \in A^n$ s.t. $P(\bar{a}) \neq 0.$

<u>Proof</u>: By induction on *n*

<u>Basis</u>: If n=1, then P has $\le$ d roots in Q.

<u>Induction</u>: Suppose lemma holds for polynomials with less than n variables. Since P nonzero, $\exists (a_1,\ldots,a_{n-1},c)$ s.t. $P(a_1,\ldots,a_{n-1},c) \neq 0.$

So by induction hypothesis $\exists$ at least $(\kappa - d)^{n-1}$ such $(a_1,\ldots,a_{n-1}) \in A^{n-1}$ s.t. $P(a_1,\ldots,a_{n-1},c) \neq 0.$ But the $\check{P}(x_n) = P(a_1,\ldots,a_{n-1},x_n)$ is nonzero polynomial with at least $\kappa - d$ elements in A s.t. $\check{P}(x_n) \neq 0.$ Lemma follows: Q.E.D.

This is the <u>key Lemma</u> used to justify hashing polynomials!

If $P\left(x_1 \ldots x_n\right)$ degr*ee d in Q,*

<u>*Theorem*</u>: *If* $\kappa = \left|A\right| \geq 2dn$, *and* $\bar{a}$ *is a random element of* A^n, *then*

$$\text{Prob}\left(P(\bar{a}) \neq 0\right) \geq \tfrac{1}{2}$$

<u>Pr*oof*</u>:

$$\text{Prob}\left(P(\bar{a}) \neq 0\right) = \frac{\left|\left\{\bar{a} : \bar{a} \in A^n, P(\bar{a}) \neq 0\right\}\right|}{\left|A^n\right|}$$

$$= \frac{(\kappa - d)^n}{\kappa^n} \quad \text{by Lemma}$$

$$= \left(1 - \tfrac{d}{\kappa}\right)^n$$

$$\geq \left(1 - \tfrac{1}{2n}\right)^n \text{ since } \kappa \geq 2dn$$

$$\geq \left[\left(1 - \tfrac{1}{2n}\right)^{2n}\right]^{\frac{1}{2}}$$

$$\geq e^{-\frac{1}{2}} \quad \text{since } \left(1 - \tfrac{1}{2n}\right)^{2n} \geq e^{-1}$$

$$\geq \tfrac{1}{2} \quad \text{since } 2 \geq e^{\frac{1}{2}}$$

Q.E.D.

<u>*Lemma 2*</u>*:*

If κ *is an integer s.t.* $1 \leq \kappa \leq 2^{2n2^n}$*,*
and m *is* <u>*randomly chosen*</u> *from* $\left\{1, \ldots, 2^{2^n}\right\}$,
then $\text{Prob}(\kappa \neq 0 \bmod m) \geq \frac{1}{4n}$ *for* $n >> 0$.

<u>*Proof*</u>*:*
By the prime number theorem, the number of primes less than 2^{2n} *is at least* $2^{2n}/2n$ *for large* n.

But κ *has at most* $2n2^n$ *prime divisors.*

Hence, $\text{Prob}(\kappa \neq 0 \bmod m)$

$$\frac{\left(\#\ \text{primes} \leq 2^{2n}\right) \text{ which don@t divide } \kappa}{2^{2n}}$$

$$\geq \frac{2^{2n}/2n - 2n2^n}{2^{2n}} \geq \frac{1}{4n} \quad \textit{Q.E.D.}$$

Algorithm: **Randomized Zero Testing**

Input: *program* $\pi(x_1,\ldots,x_t)$ *length* r

```
begin
    $n = r + t$
    $A = \{1, 2, \ldots, 2t2^r\}$
    for $i = 1, \ldots, 8n$, do
        begin
            choose random $\bar{a} \in A^t$
            choose random $m \in \{1, \ldots, 2^{2n}\}$
            if $\pi(\bar{a}) \neq 0 \bmod m$,
            then return "$\pi \neq 0$"
        end
    return "$\pi = 0$"
end
```

Theorem: $\mathrm{Pr}ob(correct\ output) \geq \frac{1}{2}$

Proof: *If* $\pi \equiv 0$, *then algorithm always correct*.

Suppose $\pi \neq 0$. *By Lemma* 1,

$\mathrm{Pr}ob(\pi(\bar{a}) \neq 0) \geq \frac{1}{2}$. *Also, if* $\pi(\bar{a}) \neq 0$, *then*

$\mathrm{Pr}ob(\pi(\bar{a}) \neq 0 \bmod m) \geq \frac{1}{4n}$, *so*

$\mathrm{Pr}ob(\pi(\bar{a}) \neq 0 \bmod m) \geq \frac{1}{2} \cdot \left(\frac{1}{4n}\right) = \frac{1}{8n}$. *Hence*,

$$\begin{aligned} \mathrm{Pr}ob(correct\ output) &\geq 1 - \left(1 - \tfrac{1}{8n}\right)^{8n} \\ &\geq 1 - e^{-1} \\ &\geq \tfrac{1}{2} \quad Q.E.D. \end{aligned}$$

Applications of Polynomial Zero Testing

(1) Given n × n matrices A, B, C
problem: $A \cdot B = C$?

(2) Given n degree Polynomials
$P_1(x)$, $P_2(x)$, $P_3(x)$
problem: $P_1(x) \cdot P_2(x) = P_3(x)$?

(3) Given n bit integers x_1, x_2, x_3
problem: $x_1 \cdot x_2 = x_3$?

(4) Given n × n Matrix A, integer r
problem: rank (A) = r?

(5) Given graph G of n vertices
problem: does G have perfect matching?

(6) Authentication systems

(7) Testing equality of sets with element addition and deletion operations.

Given:

non integer matrices A, B, C

Theorem:

Can test $A \cdot B = C$?
in time $O(n^2 \log n)$
with success probability $\geq 1 - \frac{1}{n^c}$,
for a constant c.

Proof:

Let $K = c\log n.$
Choose k *random vectors* $\vec{x}_1, \ldots, \vec{x}_k$
each of size n*, from elements in* $\{-1,1\}$

If $\exists i \in \{1,\ldots,k\}$ *s.t.* $A(B\vec{x}_i) \neq (C\vec{x}_i)$
then *output* "$A \cdot B \neq C$"
else *output* "$A \cdot B = C$"

Note*: if* $A \cdot B = C$*, then no errors ever!*
Else*: if* $A \cdot B \neq C$*,* $\forall i \in \{1,\ldots,k\}$
$Prob(A \cdot (B \cdot \vec{x}) \neq C\vec{x})$
$= Prob(D\vec{x}_i \neq 0)$ *where* $D = A \cdot B - C \neq 0$
$\geq \frac{1}{2}$ *since at most* 2^{n-1} *out of* 2^n
vectors $\vec{x}$ *have* $D \cdot \vec{x} = 0$ *if* $D \neq 0.$

So, $Prob(A \cdot (B \cdot \vec{x}_i) \neq C\vec{x}_i \text{ for } i \in \{1,\ldots,k\})$
$\geq 1 - 2^{-k} = 1 - n^{-c}.$

Given Polynomials**:** $P_1(x) \cdot P_2(x), P_3(x)$ **degree** n**.**

Theorem**: Can test** $P_1(x) \cdot P_2(x) = P_3(x)$**? in expected** $0(n)$ **arithmetic steps.**

Pr*oof*: Fix error prob. $\varepsilon \in \left(0, \frac{1}{2}\right)$**.**

Let

$$k = \frac{\lceil 1 \rceil}{\varepsilon},$$

$$w = 2^{\lceil \log(kn) \rceil}$$

Choose random $x_0 \in \{-w+1, -w+2, \ldots, 0, \ldots, w-1, w\}$

if $P_1(x_0) \cdot P_2(x_0) - P_3(x_0) \neq 0$

then return "$P_1(x) \cdot P_2(x) \neq P_3(x)$"

else "$P_1(x) \cdot P_2(x) = P_3(x)$"

Note**:** **If** $P_1 \cdot P_2 = P_3$**, then never any error!**
If $P_1 \cdot P_2 \neq P_3$**, then, since the polynomial**
$Q \equiv P_1 \cdot P_2 - P_3$ **has** ***degree*** $\leq 2n$**,**

$$\Rightarrow \textit{error probability} \leq \frac{2n}{2w} = \frac{n}{w} \leq \varepsilon \qquad Q.E.D.$$

Application to Perfect Matching

Let G = (V, E) be an undirected graph with vertex set V = {1,...,n}.

A perfect matching of G is a set of n edges on E with no common endpoints.

Define n x m matrix M

$$\text{such } M_{ij} = \begin{cases} x_{ij} & \text{if } (i,j) \in E \\ 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

Let $x_{ij} = -x_{ji}$ be indeterminate variables.

Lemma (Edmonds): G has perfect matching iff determinate $(M) \neq 0$.

⇒Randomized Algorithm for matching test:

[1] Choose each x_{ij} to be a random integer in $\{1,\ldots,n^c\}$

[2] If determinate $(M) = 0$

then return, "no perfect matching",

else, return, "a perfect matching exists".

Can set $c > \alpha 3$ to get error $< \frac{1}{n^\alpha}$.

Strongly Universal Hash Functions
(Wegman and Carter)

Let H be a set of hash fns $A \to B$

def: H is strongly universal$_n$ if

$$\forall a_1 \ldots a_n \in A \quad \forall b_1 \ldots b_n \in B$$

then $\frac{|H|}{|B|^n}$ functions in H take $a_i \to b_i$ for $i = 1, \ldots, n$.

Example: Let A, B be sets in some finite field

Let H= class of polynomials degree n of one variable.

Claim: H is strongly universal$_n$.

Proof: Given $a_1, \ldots, a_n,\ b_1, \ldots, b_n$
$\exists$ exactly one polynomial degree n
that interpolates
through distinguished pairs
$a_i \to b_i$ for all $i = 1, \ldots, n$.

$Q.E.D.$

Applications of Polynomial Hashing to Authentication System:

Let $M =$ possible message set
$T =$ authentication tags

1. public knows set functions H from $\mathsf{M} \to \mathsf{T}$
2. sender / receiver share secret random $\mathsf{f} \in \mathsf{H}$
3. sender sends message m in M with authentication tag $\mathsf{f}(\mathsf{m})$

case: $\mathsf{H} =$ strongly $\mathsf{univers}_2$ set fns $\mathsf{M} \to \mathsf{T}$
$=$ polynomials degree $< |\mathsf{M}|$

Claim: unbreakable with prob $\geq 1 - \frac{1}{|\mathsf{T}|}$

Proof: If f random fn in H forger must pick correct fn f from $\mathsf{H}' = \{\mathsf{h} \in \mathsf{G} | \mathsf{f}(\mathsf{m}) = \mathsf{h}(\mathsf{m})\}$ and substitute m' for m s.t. $\mathsf{f}(\mathsf{m}') = \mathsf{f}(\mathsf{m})$, but, by definition of strongly $\mathsf{univers}_2$ fns, only $\frac{1}{|\mathsf{T}|}$ of fns in H' map m' to $f(m)$. *Q.E.D.*

Application to Testing Set Equality

Given: set elements $A = \{a_1, \ldots, a_n\}$ and
sets $S_1, \ldots, S_m$ initially empty

Operations:

1. add element a_i to set S_j
2. delete element a_i from set S_j
3. test equality $S_{j_1} = S_{j_2}$?

Implementation:

Use set hash fn H, which is strongly *universal$_n$* for each n.
Each $f \in H$ maps from A to B.
assume: B is group with operation $\oplus$ and inverse

Example: Analyze following implementation
(Use variables $V_1, \ldots, V_m$ initially all fixed $b_0 \in B$.)

Operatio:	Implementati:
$\mathsf{S_j} \leftarrow \mathsf{S_j} \cup \{\mathsf{a_i}\}$	$\mathsf{V_j} \leftarrow \mathsf{V_j} \oplus \mathsf{f}(\mathsf{a_i})$
$\mathsf{S_j} \leftarrow \mathsf{S_j} - \{\mathsf{a_i}\}$	$\mathsf{V_j} \leftarrow \mathsf{V_j} \oplus \mathsf{f}(\mathsf{a_i})^{-1}$
test $\mathsf{S_{j_1}} = \mathsf{S_{j_2}}$?	test $\mathsf{V_{j_1}} = \mathsf{V_{j_2}}$?

Hashing Algebraic Expressions

(Gonnet, "Determining Equilibrium of Expressions in Random Polynomial Time", 1984 STOC)

Generalizations:

(1) complex arithmetic expressions

Partial Results:

(2) expressions with roots & rational components

(3) expressions with exponents

(4) expressions with trigonometric fns

Hashing Complex Expressions

Assume p prime > 2

Lemma: $\exists i$ s.t. $i^2 = -1 \bmod p$, iff $p = 4k + 1$ for some k.

Proof: Since any prime $p > 2$ is odd so $(p-1)/2$ is integer.

Let α be generator of mult. group of Z_p. Then $\alpha^{p-1} \equiv 1 \bmod p$ and $\alpha^{(p-1)/2} \equiv -1 \bmod p$. Thus $i^2 \equiv \alpha^{(p-1)/2} \equiv -1 \bmod p$ if $i = \alpha^k$ where $k = (p-1)/4$. $Q.E.D.$

Example: For $p = 13$, $i^2 = -1, \bmod p$ for $i = 5$.

Then: Can do equivalence testing of complex expressions in random polynomial time.

Hashing Expressions with Constant Exponents in Finite Fields

Expressions:

$\mathsf{E}^{\mathsf{E}'}$ allow E to have +,-, x,÷ operations.
(Compute E mod p.)
requires E' only to have +,- operartions.
(Compute E' mod $\mathsf{p}-1$.)
Since multiplication group in Z_p is a cyclic group with one less element than entire group Z_p.

Hashing Expressions with Square Roots

Proposition:

If $p = 4nj + 1$ is *prime* > 2,
then $\sqrt{j}$ mod p is defined.

Hashing Expressions with Trigonometric Functions

(no provable method)

Extensions: (Morton)

Can extend construction to find
e, π s.t. $e^{i\pi} = -1$ for certain primes p.

Open Problem:

⇒ get a provable method for identity testing of trigonometric functions $\sin(x), \cos(x)$, etc.

Idea: Use equivalences

$$\sin(x) = \left(e^{ix} - e^{-ix}\right)/2i$$

$$\cos(x) = \left(e^{ix} + e^{-ix}\right)/2$$