Čínská věta o zbytcích
RSA

Matematické algoritmy (11MAG)

Jan Přikryl

Ústav aplikované matematiky
ČVUT v Praze, Fakulta dopravní

5. přednáška 11MAG
pondělí 10. listopadu 2014

verze: 2014-11-10 11:20

Obsah přednášky

Čínská věta o zbytcích
(Chinese Reminder Theorem, CRT)

Více vzájemně ekvivalentních tvrzení z algebry a teorie čísel.
Nejstarší zmínka z Číny ve 3. století našeho letopočtu.

Problém

Jak najít x, jenž je řešením více kongruencí najednou, například

$$x \equiv 2 \pmod 3,$$
$$x \equiv 3 \pmod 5,$$
$$x \equiv 2 \pmod 7?$$

Čínská věta o zbytcích

Postup řešení (1/3)

Zbytkové třídy jsou $[2]_3$, $[3]_5$ a $[2]_7$, výsledné řešení musí spadat do všech tří z nich.

$$
\begin{aligned}
2\kappa_1 + 3\kappa_2 + 2\kappa_3 &\equiv 2 \pmod 3,\\
2\kappa_1 + 3\kappa_2 + 2\kappa_3 &\equiv 3 \pmod 5,\\
2\kappa_1 + 3\kappa_2 + 2\kappa_3 &\equiv 2 \pmod 7.
\end{aligned}
$$

Čínská věta o zbytcích

Postup řešení (2/3)

V prvním kroku hledáme nulové a jednotkové zbytkové třídy pro kombinace původních modulů. V našem případě platí

$$\begin{aligned}
\kappa_1 &= 70 \equiv 0 \pmod{5 \cdot 7} &\wedge\quad \kappa_1 &= 70 \equiv 1 \pmod{3},\\
\kappa_2 &= 21 \equiv 0 \pmod{3 \cdot 7} &\wedge\quad \kappa_2 &= 21 \equiv 1 \pmod{5},\\
\kappa_3 &= 15 \equiv 0 \pmod{3 \cdot 5} &\wedge\quad \kappa_3 &= 15 \equiv 1 \pmod{7}.
\end{aligned}$$

Čínská věta o zbytcích

Postup řešení (3/3)

Řešením dané soustavy kongruencí je v takovém případě číslo

$$\hat{x} = 2 \cdot 70 + 3 \cdot 21 + 2 \cdot 15 = 233.$$

Minimální hodnota x je dána třídou kongruence modulo $3 \cdot 5 \cdot 7 = 105$, tedy

$$x = 233 \bmod 105 = 23.$$

Čínská věta o zbytcích

Vlastní tvrzení

Nechť $n_1, n_2, \ldots, n_k$ jsou navzájem nesoudělná přirozená čísla, $n_i \geq 2$ pro všechna $i = 1, \ldots, k$. Potom řešení soustavy rovnic

$$
\begin{aligned}
x &\equiv a_1 \pmod{n_1} \\
x &\equiv a_2 \pmod{n_2} \\
&\vdots \\
x &\equiv a_k \pmod{n_k}
\end{aligned}
$$

existuje a je určeno jednoznačně v modulo $n = n_1 \cdot n_2 \cdot \ldots \cdot n_k$.

Jak si Sun Tzu ušetří práci

Lehký náznak důkazu

Díky nesoudělnosti existuje ve třídě operací modulo n_i ke každému $N_i = n/n_i$ jeho multiplikativní inverze M_i, tedy

$$M_i \cdot N_i \equiv 1 \pmod{n_i}$$

a platí

$$x = \sum_{i=1}^{k} a_i M_i N_i .$$

Ve výše uvedeném případě se zbytkovými třídami $[2]_3$, $[3]_5$ a $[2]_7$ je

$$x = 2 \cdot 2 \cdot 35 + 3 \cdot 1 \cdot 21 + 2 \cdot 1 \cdot 15 = 233.$$

Praktický význam věty

Výpočty modulo velké M lze převést na výpočty modulo menší součinitelé čísla M – zrychlení výpočtu.

Lze generalizovat pro soudělná čísla.

Význam hlavně v šifrovacích systémech.

Problém nůše s vejci

Ilustrace použití CRT

V nůši je v vajec. Pokud z ní odebíráme vejce po dvou, třech a pěti najednou, v nůši nakonec zůstane 1, 2, respektive 4 vejce. Pokud odebíráme vejce po sedmi kusech, v nůši nakonec nezůstane vejce žádné.

Jaká je nejmenší hodnota v pro niž může uvedená situace nastat?

Problém nůše s vejci

Ilustrace použití CRT (2)

Zbytkové třídy jsou $[1]_2$, $[2]_3$, $[4]_5$ a $[0]_7$.

Hledáme řešení soustavy

$$\begin{aligned} v &\equiv 1 \pmod 2 \\ v &\equiv 2 \pmod 3 \\ v &\equiv 4 \pmod 5 \\ v &\equiv 0 \pmod 7 \end{aligned}$$

Výsledek bude nějaká třída kongruence modulo 210.

Problém nůše s vejci

Ilustrace použití CRT (3)

Pro jednotlivé ekvivalence máme

i	n_i	N_i	M_i	a_i
1	2	105	1	1
2	3	70	1	2
3	5	42	3	4
4	7	30	4	0

$$\begin{aligned} v &= (1 \cdot 1 \cdot 105 + 2 \cdot 1 \cdot 70 + 4 \cdot 3 \cdot 42 + 0 \cdot 4 \cdot 30) \bmod 210 \\ &= (105 + 140 + 504 + 0) \bmod 210 = 749 \bmod 210 = 119 \end{aligned}$$

Obsah přednášky

Modulární mocnění

Výpočet $c \equiv b^r \pmod{n}$

Neefektivně lze opakovaným násobením a redukcí:

$$c = \underbrace{b\,[\,b\,[\ldots[\,b \cdot b}_{r-\text{krát}} \mod n\,]\ldots]\mod n\,]\mod n$$

Jde to ale i lépe.

Modulární mocnění

Rychlejší výpočet $c \equiv b^r \pmod{n}$

Opakovaný kvadrát

Efektivní algoritmus pro $b \in \mathbb{Z}$, $r \in \mathbb{N}$ je následující

Require: $b \in \mathbb{Z},\ r, n \in \mathbb{N}$
Ensure: $c \equiv b^r \pmod{n}$

    Nechť $r = \sum_{j=0}^{k} a_j \cdot 2^j,\ a_j \in \{0, 1\}$
    $c \leftarrow 1 + a_0 \cdot (b - 1);\ b_0 \leftarrow b$
    for $j = 1$ **to** k **do**
        $b_j \leftarrow b_{j-1}^2 \bmod n$
        if $a_j > 0$ **then**
            $c \leftarrow c \cdot b_j \bmod n$
        end if
    end for
    return $c \equiv b^r \pmod{n}$

Modulární mocnění

Příklad

Příklad (Spočtěte $c = 3^{17} \bmod 7$)

Nejprve rozložíme $r = 17 = 10001_{\mathrm{b}}$. Je $a_0 = 1$ a proto prvotní hodnota $c = b = 3$ a $b_0 = 3$. Potom

$$
\begin{aligned}
b_1 &= 3^2 \bmod 7 = 9 \bmod 7 = 2, a_1 = 0,\\
b_2 &= 2^2 \bmod 7 = 4 \bmod 7 = 4, a_2 = 0,\\
b_3 &= 4^2 \bmod 7 = 16 \bmod 7 = 2, a_3 = 0,\\
b_4 &= 2^2 \bmod 7 = 4 \bmod 7 = 4, a_4 = 1.
\end{aligned}
$$

Nyní přepočteme $c = 3 \cdot 4 \bmod 7 = 12 \bmod 7 = 5$. Další binární cifry už v r nejsou, výsledkem je proto $c = 5$.
Kontrola: $3^{17} = 129140163 \bmod 7 = 5$.

Obsah přednášky

Eulerova funkce $\phi(n)$

Rozšíření Malé Fermatovy věty

Definice (Eulerova věta)

Malou Fermatovu větu lze zobecnit na tvar

$$a^{\phi(n)} \equiv 1 \pmod{n},$$

kde $\phi(n)$ je tak zvaná **Eulerova funkce**, která udává počet přirozených čísel $1 \leq x \leq n$, jež jsou s n nesoudělná.

Někdy $\phi(n)$ označuje názvem totient.

Eulerova funkce $\phi(n)$

Rozšíření Malé Fermatovy věty

Pro prvočísla je

$$\phi(p) = p - 1,$$

pro nesoudělná x a y platí

$$\phi(x \cdot y) = \phi(x) \cdot \phi(y)$$

a proto pro prvočísla p a q také

$$\phi(p)\phi(q) = (p-1)(q-1).$$

Pro libovolné přirozené n platí také

$$\phi(n) = n \prod_{p|n} \left(1 - \frac{1}{p}\right).$$

Obsah přednášky

Šifrování

Symetrické a asymetrické šifry

Existují dvě základní skupiny šifrovacích algoritmů:

- **Symetrické šifry** u nichž se ten samý klíč používá jak k šifrování, tak i k dešifrování zprávy. Odesílatel i příjemce musí mít k dispozici identické klíče. Příkladem je DES, 3DES, AES.
- **Asymetrické šifry** u nichž se šifruje jiným klíčem, než je klíč určený k dešifrování. Odesílatel po zašifrování již nemá možnost zprávu dešifrovat. Příkladem je RSA (PGP), GnuPG, ElGamal.

Symetrické šifry jsou při stejné délce šifrovacího klíče výrazně bezpečnější, než šifry asymetrické ...

Diffieho-Hellmanova výměna klíčů

Jak se dohodnout na klíči přes nezabezpečený kanál

... ale symetrické šifrování má základní problém: distribuci klíčů.

Diffie a Hellman, 1976

Alice a Bob se na klíči mohou dohodnout přes nezabezpečený komunikační kanál. Je pouze třeba zajistit, aby operace, jež Alice a Bob provádějí, *nebyly výpočetně snadno invertovatelné*.

Diffieho-Hellmanova výměna klíčů

Veřejně známé prvočíslo p a $\alpha \in \{2, \dots, p-2\}$. Oba jako klíč použijí $\alpha^{xy} \bmod p$ – Alice si vymyslí veliké $x \in \mathbb{N}$ a Bobovi pošle $\alpha^x \bmod p$, Bob pošle Alici $\alpha^y \bmod p$. Alice pak provede $(\alpha^y)^x \bmod p$, Bob obdobně.

Diffieho-Hellmanova výměna klíčů

Vysvětlení

Vzhledem k tomu, že $[a]_p \cdot [b]_p = [ab]_p$ platí pro Alicí přijaté Bobovo α^y mod p následující:

$$\alpha^y \bmod p \equiv [\underbrace{\alpha \cdot \alpha \cdots \alpha}_{y-\text{krát}}]_p,$$

a po umocnění na x-tou:

$$\left([\underbrace{\alpha \cdot \alpha \cdots \alpha}_{y-\text{krát}}]_p \right)^x = \underbrace{[\underbrace{\alpha \cdot \alpha \cdots \alpha}_{y-\text{krát}}]_p \cdot [\underbrace{\alpha \cdot \alpha \cdots \alpha}_{y-\text{krát}}]_p \cdots [\underbrace{\alpha \cdot \alpha \cdots \alpha}_{y-\text{krát}}]_p}_{x-\text{krát}} \equiv$$

$$\equiv [\underbrace{\alpha \cdot \alpha \cdots \alpha}_{xy-\text{krát}}]_p.$$

Recipročně to platí i pro Bobem přijaté Aličino α^x mod p.

Diffieho-Hellmanova výměna klíčů

Příklad

Příklad výměny pro $p = 17$ a $\alpha = 5$

Alice si zvolí $x = 1039$.
Bob si zvolí $x = 1271$.

Po nezašifrovaném spojení pošle Alice Bobovi $5^{1039} \bmod 17 = 7$ a Bob pošle Alici $5^{1271} \bmod 17 = 10$.

Bob si spočte svůj klíč jako $7^{1271} \bmod 17 = 12$, Alice jako $10^{1039} \bmod 17 = 12$.

Ve skutečnosti budou p, α, x, y mnohem větší čísla (proč)?

Šifrování veřejným klíčem

Myšlenka RSA

RSA vychází z předpokladu, že faktorizace součinu prvočísel p a q je časově náročná – všichni proto mohou znát šifrovací klíč e a šifrovací modul $n = p \cdot q$, ale nepomůže jim to ke zjištění dešifrovacího klíče d, založeného na p a q.

V praxi je šifrovací modul $n = p \cdot q \in \{0,1\}^{1024}$ až $\{0,1\}^{4096}$.

Poslední faktorizovaný RSA klíč je RSA-768 ($n = \{0,1\}^{768}$, 232 dekadických číslic) za necelé 3 roky na až 618 pracovních stanicích v roce 2009.

Ale pozor: Už v květnu 2007 padlo $M_{1039} = 2^{1039} - 1$ za 11 měsíců v laboratořích EPFL, Uni Bonn a NTT.

Algoritmy šifrování veřejným klíčem

Prerekvizity

Algoritmus RSA staví na několika již objasněných algebraických postupech:

- Modulární mocnění
- Modulární inverze $a^{-1} \cdot a \equiv 1 \pmod{n}$
- Čínská věta o zbytcích
- Eulerova funkce

Algoritmus RSA

Generování veřejného a soukromého klíče

Přípravná fáze:

1. Zvolíme nepříliš si blízká prvočísla p a q.
2. Spočteme **modul** šifrovací a dešifrovací transformace, $n = p \cdot q$.
3. Vypočteme Eulerovu funkci pro n, $\phi(n) = (p-1)(q-1)$.
4. Zvolíme **šifrovací exponent** e takový, že $1 < e < \phi(n)$ a $\gcd(e, \phi(n)) = 1$.
5. Dopočteme **dešifrovací exponent** d tak, aby d bylo multiplikativní inverzí k e modulo $\phi(n)$, $d \cdot e \equiv 1 \pmod{\phi(n)}$.

Veřejný klíč pro zašifrování zprávy je (n, e), **soukromý klíč** pro dešifrování je (n, d).

Algoritmus RSA

Jak to funguje

Princip přenosu zprávy X je primitivní:

Šifrování

Po lince přenášíme šifrovaný text c, jenž vznikne jako

$$c = X^e \bmod n.$$

Dešifrování

Příjemce si z přijatého šifrovaného textu spočítá původní zprávu jako

$$X = c^d \bmod n.$$

Trik celého postupu spočívá v tom, že z pouhé znalosti (n, e) nelze v rozumném čase určit d.

Algoritmus RSA

Důkaz (1/3)

Obdržíme dešifrováním opravdu původní text? Při dešifrování $c \equiv X^e \pmod{n}$ máme

$$c^d \equiv (X^e)^d \equiv X^{ed} \pmod{n} \equiv X^{ed} \pmod{pq}.$$

Prozkoumáme vlastnosti $c^d \equiv X^{ed} \pmod{p}$ a $c^d \equiv X^{ed} \pmod{q}$ a zobecníme je na operace modulo n.

Z definice součinu ed v algoritmu RSA plyne

$$ed \equiv 1 \pmod{\phi(n)} \Rightarrow \exists g \in \mathbb{Z} : ed = 1 + g(p-1)(q-1),$$

což můžeme dále upravit na

$$ed = 1 + f(p-1)(q-1) = 1 + g(q-1) = 1 + h(p-1)$$

a tedy

$$ed \equiv 1 \pmod{\phi(n)} \equiv 1 \pmod{\phi(p)} \equiv 1 \pmod{\phi(q)}.$$

Algoritmus RSA
Důkaz (2/3)

Dokazujeme nadále p a q odděleně:

Pro $p \nmid X$ je podle Malé Fermatovy věty $X^{p-1} \equiv 1 \pmod p$ a tedy

$$X^{ed} = X^{1+h(p-1)} = X \cdot X^{h(p-1)} = X \cdot \left(X^{(p-1)}\right)^h \equiv X \cdot 1^h \equiv X \pmod p.$$

Pro $p \mid X$ je

$$X^{ed} \equiv 0^{ed} \pmod p \equiv X \pmod p$$

To samé platí pro q a tedy

$$X^{ed} \equiv X \pmod p$$
$$X^{ed} \equiv X \pmod q$$

Algoritmus RSA

Důkaz (3/3)

Jedním z důsledků CRT je pro nesoudělná x a y ekvivalence

$$\begin{array}{rcl} a & \equiv & b \pmod{x} \\ a & \equiv & b \pmod{y} \end{array} \quad \Leftrightarrow \quad a \equiv b \pmod{xy}.$$

Proto také z

$$X^{ed} \equiv X \pmod{p}$$
$$X^{ed} \equiv X \pmod{q}$$

plyne

$$X^{ed} \equiv X \pmod{pq}.$$

RSA pomocí CRT

Urychlení dešifrování (1)

Jak modul n, tak i dešifrovací exponent d jsou hodně velká čísla, a proces dešifrování

$$X \equiv c^d \pmod{n}$$

trvá dlouho.

Pro $n = pq$ použijme již jednou provedený trik

$$X \equiv X_p \pmod{p} \equiv c^{d_p} \pmod{p},$$
$$X \equiv X_q \pmod{q} \equiv c^{d_q} \pmod{q},$$

a tedy

$$c^d \equiv c^{d_p + j(p-1)} \pmod{p} \equiv c^{d_p} 1^j \pmod{p} \equiv c^{d_p} \pmod{p},$$
$$c^d \equiv c^{d_q + k(q-1)} \pmod{q} \equiv c^{d_q} 1^k \pmod{q} \equiv c^{d_q} \pmod{q}.$$

RSA pomocí CRT

Urychlení dešifrování (2)

Zpráva X je tedy řešením soustavy dvou kongruencí sestavených pro c:

$$X \equiv c^{d_p} \pmod{p},$$
$$X \equiv c^{d_q} \pmod{q}.$$

Řešením je

$$X = \left[c^{d_p} M_p q + c^{d_q} M_q p\right] \bmod pq,$$

kde $M_p = q^{-1} \bmod p$ a $M_q = p^{-1} \bmod q$.

Algoritmus RSA

Prolomení při nevhodné volbě p a q

Pokud zvolíme p a q nevhodně (blízko sebe, příliš malá, atd.), útočník využije znalosti (n, e):

1. Faktorizuje n na p a q.
2. Vypočte Eulerovu funkci pro n, $\phi(n) = (p-1)(q-1)$.
3. Dopočte **dešifrovací exponent** d tak, aby $d \cdot e \equiv 1 \pmod{\phi(n)}$.

Náš **soukromý klíč** pro dešifrování (n, d) v ten okamžik zná i útočník a může moje zprávy dešifrovat.

Obsah přednášky

A co nás čeká příště?

Grafy a grafové algoritmy.