

DA-naklonjen Monte Carlo algoritem je probabilistični algoritem za odločitveni problem (tj. DA/NE-problem), pri katerem je “DA” odgovor (vedno) pravilen, “NE” odgovor pa je lahko nepravilen.

Verjetnost napake za **DA-naklonjen Monte Carlo** algoritem je ε , če za vsak odgovor “DA” algoritem odgovori z “NE” z verjetnostjo kvečjemu ε .

Solovay-Strassen algoritem

1. Izberi naključno število $a \in \mathbb{Z}_n$, $x := \left(\frac{a}{n}\right)$.
2. **if** $x = 0$ **then return**(" n je sestavljeno število").
3. $y := a^{(n-1)/2} \bmod n$,
if $x \equiv y \pmod{n}$
then return(" n je praštevilo")
else return(" n je sestavljeno število").

Verjetnost napake pri Solovay-Strassen algoritmu je kvečjemu $1/2$ (glej nalogo 4.14 v Stinsonu).

Monte Carlo verjetnostni algoritem za odločitveni problem, ali je število sestavljeno: test ponovimo m -krat z naključnimi vrednostmi a . Verjetnost, da bo odgovor napačen m -krat zapored napačen je ε^m , vendar pa iz tega še ne moremo zaključiti, da je verjetnost, da je n praštevilo, $1 - \varepsilon^m$.

Dogodek A :

“naključen lih n določene velikosti je sestavljen”

in dogodek B :

“algoritem odgovori ‘ n je praštevilo’ m -krat zapored.”

Potem očitno velja $P(B/A) \leq \varepsilon^m$, vendar pa nas v resnici zanima $P(A/B)$, kar pa ni nujno isto.

Naj bo $N \leq n \leq 2N$ in uporabimo izrek o gostoti praštevil

$$\frac{2N}{\log 2N} - \frac{N}{\log N} \approx \frac{N}{\log N} \approx \frac{n}{\log n}.$$

Sledi $P(A) \approx 1 - 2/\log n$. Bayesovo pravilo pravi:

$$P(A/B) = \frac{P(B/A)P(A)}{P(B)}.$$

Imenovalec je enak $P(B/A)P(A) + P(B/\bar{A})P(\bar{A})$.

Upoštevajmo še $P(B/\bar{A}) = 1$ in dobimo

$$\begin{aligned} P(A/B) &= \frac{P(B/A)(\log n - 2)}{P(B/A)(\log n - 2) + 2} \leq \\ &\leq \frac{2^{-m}(\log n - 2)}{2^{-m}(\log n - 2) + 2} = \frac{(\log n - 2)}{\log n - 2 + 2^{m+1}}, \end{aligned}$$

kar pomeni, da gre iskana verjetnost eksponentno proti 0.

Monte Carlo verjetnostni algoritem za odločitveni problem ali je število sestavljeno:

Test ponovimo k -krat z različnimi vrednostmi a . Verjetnost, da bo odgovor k -krat zapored napačen, je za nas ocenjena z ε^k .

DN: Iz naslednjega izreka izpeljite, da za izračun Jacobijevega simbola ne potrebujemo praštevilske faktorizacije števila n .

Gaussov izrek

Izrek o kvadratni recipročnosti (1796)

Če sta p in q različni lihi praštevíli, potem velja

$$\left(\frac{p}{q}\right) \left(\frac{q}{p}\right) = (-1)^{\frac{p-1}{2} \cdot \frac{q-1}{2}}$$

ter za praštevílo 2

$$\left(\frac{2}{p}\right) = (-1)^{\frac{p^2-1}{8}}.$$

Zakaj je ta izrek tako pomemben?

Pomaga nam, da odgovorimo, kdaj imajo kvadratne kongruence rešitev, saj velja multiplikativno pravilo

$$\left(\frac{ab}{p}\right) = \left(\frac{a}{p}\right) \left(\frac{b}{p}\right).$$

Predstavlja pa tudi nepričakovano zvezo med pari praštevil (pravilo, ki ureja praštevila).

Eisensteinova lema. $p > 2$ praštevilo, $p \nmid q \in \mathbb{N}$.

Naj bo $A := \{2, 4, 6, \dots, p-1\}$ in $r_a := qa \pmod p$ za $a \in A$. Potem je

$$\left(\frac{q}{p}\right) = (-1)^{\sum_{a \in A} r_a}.$$

Dokaz: Za $a, a' \in A$, $a \neq a'$, ne more veljati

$r_a(-1)^{r_a} = r_{a'}(-1)^{r_{a'}}$ oziroma $qa \equiv \pm qa' \pmod p$,

saj bi od tod sledilo $a = \pm a'$, kar pa ni mogoče.

Opozorimo še, da so vsa števila $r_a(-1)^{r_a} \pmod p$ soda, torej pretečejo ravno vse elemente množice A .

Od tod dobimo

$$\prod a \equiv (-1)^{\sum r} \prod r \pmod{p},$$

očitno pa neposredno iz definicije sledi tudi

$$q^{\frac{p-1}{2}} \prod a \equiv \prod r \pmod{p}.$$

Torej velja $q^{\frac{p-1}{2}} \equiv (-1)^{\sum r} \pmod{p}$ in po Eulerjevem kriteriju še

$$\left(\frac{q}{p}\right) \equiv q^{\frac{p-1}{2}} \pmod{p}. \quad \blacksquare$$

Oglejmo si Eisensteinov *dokaz Gaussovega izreka o kvadratni recipročnosti*. Očitno velja

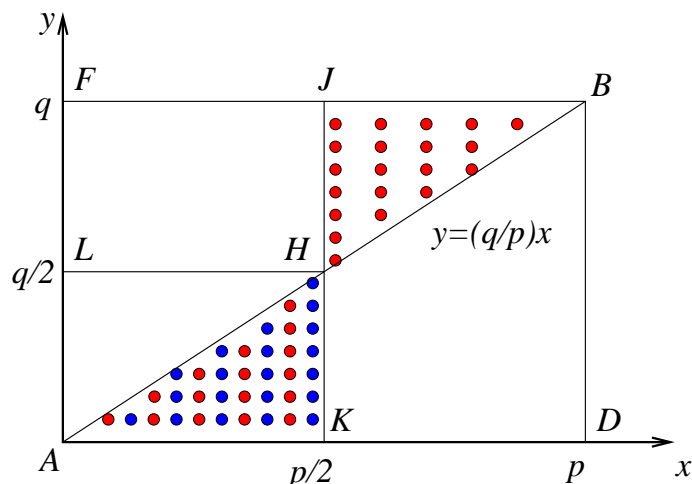
$$\sum qa = p \sum \left[\frac{qa}{p} \right] + \sum r .$$

Ker so elementi a vsi sodi in je p lih, velja

$$\sum r \equiv \sum \left[\frac{qa}{p} \right] \pmod{2}$$

in zato iz Eisensteinove leme sledi

$$\left(\frac{q}{p} \right) = (-1)^{\sum \left[\frac{qa}{p} \right]} .$$



Vsota $\sum \left\lfloor \frac{qa}{p} \right\rfloor$ je enaka številu celoštevilčnih točk sodo x -koordinato, ki ležijo v notranjosti trikotnika ABD . Sedaj pa si oglejmo točke z x -koordinato večjo od $p/2$. Ker pa je $q - 1$ sod, je parnost števila $\left\lfloor \frac{qa}{p} \right\rfloor$ točk z isto x -koordinato pod diagonalo AB enako številu točk z isto sodo x -koordinato nad diagonalo AB .

To pa je po drugi strani enako številu točk pod diagonalo AB z liho x -koordinato $p - a$ (bijektivna korespondenca med točkami s sodo x -koordinato v BHJ in liho x -koordinato v AHK). Od tod sledi, da ima vsota $\sum \lfloor \frac{qa}{p} \rfloor$ enako parnost kot številu μ celoštevilčnih točk v notranjosti trikotnika AHK , tj.

$$\binom{q}{p} = (-1)^\mu.$$

Če zamenjamo p in q , dobimo še število ν celoštevilčnih točk v notranjosti trikotnika AHL , kar nam da

$$\binom{p}{q} = (-1)^\nu$$

in skupaj s prejšnjo relacijo Gaussov izrek. ■

Še en Monte Carlo algoritem za testiranje sestavljenosti števil.

Miller-Rabinov test: *testiramo liho število n .*

1. $n - 1 = 2^k m$, kjer je m liho število,
2. izberemo naključno naravno število $a < n$,
3. izračunamo $b \equiv a^m \pmod{n}$,
4. **if** $b \equiv 1 \pmod{n}$ **then** n je praštevilo; **exit**;
5. **for** $i = 0$ **to** $k - 1$ **do**
 - if** $b \equiv -1 \pmod{n}$
then n je praštevilo;
 - exit**;
 - else** $b \equiv b^2 \pmod{n}$,
7. število n je sestavljeno.

Izrek: *Miller-Rabinov algoritem za problem sestavljenih števil je DA-naklonjen Monte Carlo algoritem.*

Dokaz: Predpostavimo, da algoritem odgovori “ n je sestavljeno število” za neko praštevilo p .

Potem je $a^m \not\equiv 1 \pmod{n}$.

Sledi $a^{2^i m} \not\equiv -1 \pmod{n}$ za $i \in \{0, 1, \dots, k-1\}$.

Ker je $n = 2^k m + 1$ praštevilo, iz Fermatovega izreka sledi

$$a^{2^k m} \equiv 1 \pmod{n}$$

in je $a^{2^{k-1} m}$ koren od 1 po modulu n .

Iz $x^2 \equiv 1 \pmod{n}$ oziroma $n \mid x^2 - 1 = (x - 1)(x + 1)$ sledi

$$x \equiv 1 \pmod{n} \quad \text{ali} \quad x \equiv -1 \pmod{n}$$

oziroma v našem primeru $a^{2^{k-1}m} \equiv 1 \pmod{n}$. Na isti način pridemo do

$$a^m \equiv 1 \pmod{n},$$

kar je protislovje, saj bi algoritem v tem primeru odgovoril “ n je praštevilo”. ■

Za konec omenimo brez dokaza še, da je verjetnost napake Miller-Rabinovega algoritma kvečjemu $1/4$.

Napadi na RSA

Odličen pregledni članek “Twenty Years of Attacks on the RSA kriptosystem”, je objavil Dan Boneh v *Notices of AMS*, Feb. 1999, pp. 203-212.

Mi bomo omenili le nekaj osnovnih napadov.

Če poznamo $\varphi(n)$ in n , dobimo p , q iz naslednjega sistema dveh enačb

$$n = pq \quad \text{in} \quad \varphi(n) = (p - 1)(q - 1).$$

Odšifrirni eksponent kriptosistema RSA

Trditev: Vsak algoritem A , ki najde odšifrirni eksponent d , lahko uporabimo kot podprogram v probabilističnem algoritmu, ki najde faktorje števila n .

Od tod sledi, da iskanje odšifrirnega eksponenta ni nič lažje kot problem faktorizacije.

Opozorilo: če “izgubimo” d , moramo poleg šifrirnega eksponenta zamenjati tudi modul n .

Naj bo $\varepsilon \in [0, 1)$. **Las Vegas algoritem** je probabilističen algoritem, ki za dani primer problema, lahko *ne da odgovora* z verjetnostjo ε (se pravi, da konča s sporočilom “ni odgovora”). Če pa algoritem odgovori, potem je *odgovor gotovo pravilen*.

DN: Pokaži, da je povprečno pričakovano število ponovitev algoritma vse dokler ne dobimo odgovora, enako $1/(1 - \varepsilon)$ (glej nalogo 4.15).

Če Las Vegas algoritem faktorizira število n z verjetnostjo vsaj ε in ga ponovimo m -krat, potem bo število n faktorizirano z verjetnostjo vsaj $1 - \varepsilon^m$.

Trditev sledi iz algoritma, ki uporablja naslednje:
za $n = pq$, kjer sta p, q lihi praštevili,

$$x^2 \equiv 1 \pmod{n}, \quad \text{tj. } pq \mid (x-1)(x+1),$$

dobimo štiri rešitve; dve (trivialni) rešitvi iz enačb

$$x \equiv 1 \pmod{n} \quad \text{in} \quad x \equiv -1 \pmod{n}$$

in s pomočjo kitajskega izreka o ostankih iz

$$x \equiv 1 \pmod{p}, \quad x \equiv -1 \pmod{q}$$

in

$$x \equiv -1 \pmod{p}, \quad x \equiv 1 \pmod{q}$$

še dve (netrivialni) rešitvi.

Algoritem za faktorizacijo z danim šifr. eksp. d

1. Izberi naključno naravno število $w < n$,
2. izračunaj $x = D(w, n)$,
3. **if** $1 < x < n$ **then exit**(uspeh $x = p$ ali $x = q$)
4. izračunaj $d = A(e, n)$ in zapiši $de - 1 = 2^s r$, r lih,
5. izračunaj $v = w^r \pmod n$,
6. **if** $v \equiv 1 \pmod n$ **then exit**(neuspeh)
7. **while** $v \not\equiv 1 \pmod n$ **do** $v_0 = v$, $v = v^2 \pmod n$
8. **if** $v_0 \equiv -1 \pmod n$ **then exit**(neuspeh)
 else izračunaj $x = D(v_0 + 1, n)$
 (uspeh: $x = p$ ali $x = q$) .

Naključne napake

(Boneh, DeMillo in Lipton, 1997)

Če uporabimo CRT in pride pri samo enem izmed C_p in C_q do napake, npr. C_p je pravilen, \hat{C}_q pa ni, potem je $\hat{C} = t_p C_p + t_q \hat{C}_q$ očitno nepravilen podpis, saj je $\hat{C}^e \neq M \pmod{N}$. Vendar pa je

$$\hat{C}^e = M \pmod{p}, \quad \text{medtem, ko je } \hat{C}^e \neq M \pmod{q}$$

in nam $D(n, \hat{C}^e - M)$ odkrije netrivialni faktor števila n .

Rabinov kriptosistem

Temelji na tem, da je težko najti faktorizacijo produkta dveh velikih praštevil p in q .

$$n = pq, \quad p \neq q, \quad p, q \equiv 3 \pmod{4}, \quad \mathcal{P} = \mathcal{C} = \mathbb{Z}_n$$

$$\mathcal{K} = \{(n, p, q, B); 0 \leq B \leq n - 1\}$$

Za izbrani ključ $K = (n, p, q, B)$ naj bo:

$$e_K(x) = x(x + B) \pmod{n},$$

$$d_K(y) = \sqrt{y + B^2/4} - B/2.$$

Javni ključ je (n, B) , zasebni ključ pa (p, q) .

Trditev: Naj bo $\omega^2 \equiv 1 \pmod{n}$ netrivialen koren (kongruenca ima 4 rešitve: 1, -1 in še dve netrivialni), in $x \in \mathbb{Z}_n$, potem velja:

$$e_K(\omega(x + B/2) - B/2) = e_K(x).$$

Imamo 4 čistopise, ki ustrezajo tajnopisu $e_K(x)$:

$$x, \quad -x - B, \quad \omega(x + \frac{B}{2}) \quad \text{in} \quad -\omega(x + \frac{B}{2}).$$

V splošnem se ne da ugotoviti, kateri je pravi.

Odšifriranje

Imamo tajnopis y in iščemo x , ki zadošča naslednji enačbi:

$$x^2 + Bx \equiv y \pmod{n}.$$

Poenostavimo: $x = x_1 - B/2$,

$$x_1^2 \equiv y + B^2/4 \pmod{n}, \quad C = y + B^2/4.$$

Iščemo kvadratne korene enačbe $x_1^2 \equiv C \pmod{n}$.

To je ekvivalentno sistemu:

$$\left. \begin{array}{l} x_1^2 \equiv C \pmod{p} \\ x_1^2 \equiv C \pmod{q} \end{array} \right\}$$

Eulerjev izrek:
 $C^{(p-1)/2} \equiv 1 \pmod{p}$

↓

predpostavka: $p \equiv 3 \pmod{4}$
 $\Rightarrow (\pm C^{(p+1)/4})^2 \equiv C \pmod{p}$

$$\left. \begin{array}{l} x_1 \equiv x_{1,2} \pmod{p} \\ x_1 \equiv x_{3,4} \pmod{q} \end{array} \right\}$$

\Rightarrow korena prve enačbe sta:
 $x_{1,2} = \pm C^{(p+1)/4}$

↓ KIO
 x_1, x_2, x_3, x_4

korena druge enačbe pa:
 $x_{3,4} = \pm C^{(q+1)/4}$

Primer: $n = 77 = 7 \cdot 11$, $B = 9$

$$e_K(x) = x^2 + 9x \pmod{77}$$

$$d_K(y) = \sqrt{y + B^2/4} - B/2 = \sqrt{1 + y} - 43 \pmod{77}$$

Tajnopis: $y = 22$. Poiskati moramo rešitve:

$$\begin{array}{l|l} x^2 \equiv 23 \pmod{7} & (x \equiv \pm 4 \pmod{7}) \\ x^2 \equiv 23 \pmod{11} & (x \equiv \pm 1 \pmod{11}) \end{array}$$

Dobimo štiri sisteme dveh enačb z dvema neznankama, npr.:

$$x \equiv 4 \pmod{7}, \quad x \equiv 1 \pmod{11}$$

Po kitajskem izreku o ostankih velja:

$$x = 4 \cdot 11 \cdot (11^{-1} \bmod 7) + 1 \cdot 7 \cdot (7^{-1} \bmod 11).$$

Vse rešitve so:

$$\begin{aligned} x_1 &= 67 \pmod{77}, & x_2 &= 10 \pmod{77}, \\ x_3 &= 32 \pmod{77}, & x_4 &= -32 \pmod{77}. \end{aligned}$$

Odšifrirani tekst je:

$$\begin{aligned} d_K(y) &= 67 - 43 \pmod{77} = 24 \\ &10 - 43 \pmod{77} = 44 \\ &32 - 43 \pmod{77} = 66 \\ &45 - 43 \pmod{77} = 2, \end{aligned}$$

vse štiri rešitve pa se zašifrirajo v 22.

Varnost Rabinovega kriptosistema

Hipotetični algoritem A za dekripcijo Rabinovega kriptosistema lahko uporabimo kot podprogram v algoritmu tipa Las Vegas za faktorizacijo števila n z verjetnostjo vsaj $1/2$.

1. Izberemo r , $1 \leq r \leq n - 1$,
2. $y := r^2 - B^2/4 \pmod{n}$ ($y = e_K(r - B/2)$),
3. $x := A(y)$,
4. $x_1 := x + B/2$ ($x_1^2 \equiv r^2 \pmod{n}$),
5. če velja $x_1 \equiv \pm r \pmod{n}$, potem ni odgovora, sicer ($x_1 \equiv \pm \omega \cdot r \pmod{n}$, kjer je $\omega \equiv 1 \pmod{n}$ netrivialni koren) $D(x_1 + r_1, n) = p$ (ali q).

V zadnjem primeru $n \mid (x_1 - r)(x_1 + r)$, vendar $n \nmid (x_1 - r)$ in $n \nmid (x_1 + r) \Rightarrow D(x_1 + r, n) \neq 1$.

Verjetnost, da uspemo v enem koraku:

Def: $r_1 \sim r_2 \Leftrightarrow r_1^2 \equiv r_2^2 \pmod{n}$ ($r_1, r_2 \neq 0$).

To je ekvivalenčna relacija, ekvivalenčni razredi v $Z_n \setminus \{0\}$ imajo moč 4: $[r] = \{\pm r, \pm \omega r\}$.

Vsak element iz $[r]$ nam da isto vrednost y .

Podprogram A nam vrne x , $[x] = \{\pm x, \pm \omega x\}$,

$r = \pm x : 4$ ni odgovora $r = \pm \omega x :$ dobimo odgovor.

Ker izberemo r slučajno, je vsaka od teh možnosti enako verjetna \Rightarrow verjetnost, da uspemo, je $1/2$.

Algoritmi za faktorizacijo števil

Poskušanje

Število n delimo z vsemi lihimi števili do \sqrt{n} :

$i := 3,$

until $i \leq \sqrt{n}$ **repeat**

if $i \mid n$, potem smo našli faktor,

else $i := i + 2.$

Algoritem je uporaben za manjše n (npr. $n \leq 10^{12}$).
Časovna zahtevnost za k bitov je $2^{k/2-1}$ deljenj.

Metoda $p - 1$ (Pollard, 1974)

Podatki: n (lih, želimo faktorizirati) in B (meja)

Algoritem temelji na naslednjem preprostem dejstvu:

če je p praštevilo, ki deli n , in za vsako praštevilsko potenco q , ki deli $p - 1$, velja $q \leq B$, potem $(p - 1) | B!$

Primer: $B = 9$, $p = 37$, $p - 1 = 36 = 2^2 \cdot 3^2$

$2^2 \leq B, 3^2 \leq B \Rightarrow 2^2 \cdot 3^2 | 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 \cdot 7 \cdot 8 \cdot 9$

Algoritem

Podatki: n, B

1. $a := 2$

2. $j = 2, \dots, B$ $(a \equiv 2^{B!} \pmod{n})$

$a := a^j \pmod{n}$ $(\Rightarrow a \equiv 2^{B!} \pmod{p})$

3. $d = D(a - 1, n)$ $(\text{Fermat: } 2^{p-1} \equiv 1 \pmod{p})$

4. Če velja $1 < d < n$, je d faktor števila n (saj $p|d$)
sicer ni uspeha (to se zgodi, kadar je $d=1$).

Če $B \geq \sqrt{n}$, vedno uspemo, vendar algoritem ni učinkovit.

Časovna zahtevnost

- $B - 1$ potenciranje po modulu n ,
za vsako rabimo $2 \log_2 B$ množenj po modulu n ,
- največji skupni delitelj z Evklid. alg.: $\mathcal{O}((\log n)^3)$.

Skupaj $\mathcal{O}(B \log B (\log n)^2 + (\log n)^3)$, kar pomeni, da je za $B \approx (\log n)^i$ algoritem polinomski.

Primer: $n = 143$, $B = 4$, $a \equiv 2^{2 \cdot 3 \cdot 4} \equiv 131 \pmod{143}$.

Torej je $a - 1 = 130$ in od tod $D(130, 143) = 13$.

Za varen RSA izberemo $p = 2p_1 + 1$ in $q = 2q_1 + 1$,
kjer sta p_1 in q_1 praštevili.

Dixonov algoritem in kvadratno rešeto

$$(x \not\equiv \pm y \pmod{n}, x^2 \equiv y^2 \pmod{n}) \implies D(x-y, n) \neq 1$$

Sestavimo bazo faktorjev $\mathcal{B} = \{p_1, \dots, p_B\}$, kjer so p_i “majhna” praštevila. Naj bo C malo večji kot B (npr. $C = B + 10$). Najdemo C kongruenc:

$$x_j^2 \equiv p_1^{\alpha_{1,j}} \times p_2^{\alpha_{2,j}} \times \dots \times p_B^{\alpha_{B,j}} \pmod{n}, \quad 1 \leq j \leq C$$

Označimo $a_j := (\alpha_{1,j} \bmod 2, \dots, \alpha_{B,j} \bmod 2)$.

Če najdemo podmnožico $\{a_1, \dots, a_C\}$, v kateri se vektorji seštejejo v $(0, 0, \dots, 0) \bmod 2$, potem bo produkt x_j uporabil vsak faktor iz \mathcal{B} sodo mnogokrat.

Primer: $n = 15770708441$, $\mathcal{B} = \{2, 3, 5, 7, 11, 13\}$

$$8340934156^2 \equiv 3 \times 7 \pmod{n} \quad a_1 = (0, 1, 0, 1, 0, 0)$$

$$12044942944^2 \equiv 2 \times 7 \times 13 \pmod{n}, \quad a_2 = (1, 0, 0, 1, 0, 1)$$

$$2773700011^2 \equiv 2 \times 3 \times 13 \pmod{n}, \quad a_3 = (1, 1, 0, 0, 0, 1)$$

Iz $a_1 + a_2 + a_3 = (0, 0, 0, 0, 0, 0) \pmod{2}$ sledi

$$(8340934156 \times 12044942944 \times 2773700011)^2 \equiv$$

$$\equiv (2 \times 3 \times 7 \times 13)^2 \pmod{n}$$

$$\text{ozioroma } 9503435785^2 \equiv 546^2 \pmod{n}$$

$$\text{in } D(9503435785 - 546, 15770708441) = 115759.$$

- Linearno odvisnost med vektorji $\{a_1, a_2, \dots, a_C\}$ poiščemo npr. z Gaussovo eliminacijo.
- $C > B + 1$: vendar imamo raje več različnih odvisnosti, da bo vsaj ena dala faktorizacijo.
- Števila x_j , za katere se da $x_j^2 \pmod{n}$ faktorizirati v \mathcal{B} , iščemo v množici $\{x_j = j + \lfloor \sqrt{n} \rfloor \mid j = 1, 2, \dots\}$ z metodo **kvadratnega rešeta** (Pomerance).
- Če je \mathcal{B} velik, je večja možnost, da se da neko število faktorizirati v \mathcal{B} , a potrebujemo več kongruenc, da najdemo linearno odvisnost. ($|\mathcal{B}| \approx \sqrt{e^{\sqrt{\ln n \ln \ln n}}}$).

Algoritmi za faktorizacijo v praksi

Kvadratno rešeto $O(e^{(1+o(1))\sqrt{\ln n \ln \ln n}})$

Eliptične krivulje $O(e^{(1+o(1))\sqrt{\ln p \ln \ln p}})$

Številsko rešeto $O(e^{(1.92+o(1))(\ln n)^{1/3}(\ln \ln n)^{2/3}})$

$o(1) \rightarrow 0$, ko $n \rightarrow \infty$

p - najmanjši praštevilski faktor n

V najslabšem primeru, ko je $p \approx \sqrt{n}$, imata kvadratno rešeto in eliptične krivulje približno enako časovno zahtevnost, sicer pa je boljše kvadratno rešeto.

Faktorizacije velikih števil s kvadratnim rešetom:

$$(n = p \cdot q, p \approx q)$$

69 cifer	1983	Davis, Holdredge, Simmons (sestavljen faktor od $2^{251} - 1$)
< 106 cifer	1989	Lenstra, Manasse + pomočniki
129 cifer	1994	Atkins, Graff, Lenstra, Leyland +600 pomočnikov

Fermatova števila:

$2^{2^{11}} - 1$ eliptične krivulje: 1988 (Brent)

$2^{2^9} - 1$ številsko rešeto:
1990 (Lenstra, Lenstra, Manasse, Pollard)

Prof. Vidav je leta 1997 zastavil naslednje vprašanje (morda tudi zato, da preveri trenutne moči namiznih računalnikov): poišči prafaktorje števila

$$10^{64} + 1$$

in namignil, da so vsi prafaktorji, če jih je kaj, oblike $128k + 1$.

Večina osebnih računalnikov z Mathematica/Maple hitro najde en faktor:

1265011073

55-mestni ostanek pa povzroči težave.

V Waterlooju sem končno našel hiter računalnik (cacr: Alpha ???) ter hitro programsko opremo (glej <http://www.informatik.th-darmstadt.de/TI/LiDIA/>), ki je v manj kot 10-ih minutah našla še preostala prafaktorja

15343168188889137818369

515217525265213267447869906815873.