Vorlesung Kryptographie

WiSe '23/'24, hhu
K. Halupczok

K11: DL-Problem am Quantencomputer

Stichworte: diskreter Logarithmus, DL in $\mathbb{Z}_m^\times$ und Vergleich mit DL-Problem auf elliptischen Kurven, Shors Algorithmus zu DL in $\mathbb{Z}_p^\times$

11.1. Neben dem Faktorisierungsproblem gibt es vor allem ein weiteres, schwer zu lösendes mathematisches Problem, auf dem in der Praxis benutzte Kryptoverfahren beruhen (z.B. Diffie-Hellman, el Gamal-Signatur und -Verschlüsselung, ...):

Das Problem des diskreten Logarithmus (DL-Problem):

Geg. sei eine abelsche Gruppe, wir beschreiben das Problem multiplikativ und additiv:

In $(G, \cdot, 1)$:	In $(G, +, 0)$:
Sei $x \in G$, $n = \mathrm{ord}(x)$, $y \in \langle x \rangle = \{x^l ; l \in \mathbb{Z}\}$.	Sei $x \in G$, $n = \mathrm{ord}(x)$, $y \in \langle x \rangle = \{l \cdot x ; l \in \mathbb{Z}\}$.
Bestimme $k \bmod n$ mit $y = x^k$. ("diskreter Logarithmus")	Bestimme $k \bmod n$ mit $y = k \cdot x$. ("diskreter Logarithmus")

11.2. Bem.: Ist eine Gruppe G gegeben, in der das DL-Problem schwer lösbar ist, kann dies für ein Kryptoverfahren genutzt werden.

- Im Fall $G = (\mathbb{Z}_m^\times, \cdot, 1)$ ist das DL-Problem vergleichbar schwer wie das Faktorisierungsproblem.
- Die Gruppe $(\mathbb{Z}_m^\times, \cdot, 1)$ ist zyklisch genau für $m \in \{1, 2, 4\}$ oder $m = p^k$ oder $m = 2p^k$ für ein bel. $k \in \mathbb{N}$ und eine bel. Primzahl $p > 2$, nach dem Satz von Euler über die Existenz von Primitivwurzeln (= Erzeuger von $\mathbb{Z}_m^\times$). Man kann auch einfach ein $x \in G$ (mit großer Ordnung in G) wählen und $\tilde{G} = \langle x \rangle$ anstelle G betrachten.

11.3. Im Fall, dass $G = (\tilde{E}(\mathbb{R}), +, \mathcal{O})$ die Gruppe einer (kryptographisch) geeigneten elliptischen Kurve ist, ist das DL-Verfahren quasi unlösbar. Die besten bekannten klassischen Algorithmen (d.h. auf einem klassischen Computer) sind langsamer als die für das DL-Problem für $\mathbb{Z}_m^\times$:

vgl. auch die Details dazu in Kapitel K25

* Für $m=p$ prim etwa gibt es das Index-Calculus-Verfahren für das DL-Problem in $\mathbb{Z}_p^\times$, das eine Laufzeit von $O(\exp(c\sqrt{\log(p)\log\log(p)}))$ besitzt [Yan, §2.4.3]. Weiter den Silver-Pohlig-Hellman-Algorithmus für das DL-Problem in $\mathbb{Z}_{p^s}^\times$ mit einer Laufzeit von $O(\sqrt{p})$ [Yan, §2.4.2], den Baby-Step-Giant-Step-Algorithmus von Shanks für das DL-Problem in $\mathbb{Z}_m^\times$ mit $O(\sqrt{m})$ [Yan, §2.4.1].
* Für das DL-Problem auf einer elliptischen Kurve gibt es den Index-Calculus-Algo, welcher neue Probleme einbringt (das Liften der Punkte einer elliptischen Kurve über $\mathbb{F}_p$ auf eine über $\mathbb{Q}$), die kaum machbar sind; subexponentielle Verfahren ex. dafür nicht, es sei denn, die ell. Kurve ist kryptographisch ungeeignet, vgl. Kapitel K25. (Auch andere Verfahren, z.B. das "Xedni"-Calculus-Verfahren, haben dieses Problem.) [Yan, §2.4.4].

Darauf fendet die als höher angesehene Sicherheit bei der Kryptographie mit elliptischen Kurven. → höhere Schlüssellängen bei $\mathbb{Z}_m^\times$ erforderlich (bei gleicher Sicherheit)

Auf Quantencomputern ist das DL-Problem allerdings schnell lösbar:

11.4. Shors Algorithmus zur polynomiell schnellen Lösung des DL-Problems in $\mathbb{Z}_p^\times$:
Geg. sei eine Primitivwurzel g von $\mathbb{Z}_p^\times$. Problem: Zu $x \in \mathbb{Z}_p^\times$ finde $0 \le r \le p-1$ mit $g^r \equiv x \,(p)$.

1. Schritt: Sei $q = 2^L$, $p < q < 2p$. Initialisiere 3 Quantenregister mit

$$\frac{1}{p-1} \sum_{a,b=0}^{p-2} |a, b, g^a x^{-b}(p)\rangle.$$

2. Schritt: Führe eine QFT durch, erhalte so Zustand

$$\frac{1}{(p-1)q} \sum_{a,b=0}^{p-2} \sum_{c,d=0}^{q-1} e\left(\frac{ac+bd}{q}\right) |c, d, g^a x^{-b}(p)\rangle.$$

3. Schritt: Führe eine Messung durch, die W., $|c, d, y\rangle$ zu messen mit $y = g^q(p)$, d.h. $g^a x^b \equiv g^q(p) \;(=) \; a - br \equiv k \;(p-1)$, beträgt $|P|^2$ mit

$$P := \frac{1}{(p-1)q} \sum_{\substack{a,b=0 \\ a \equiv br+k\,(p-1)}}^{p-2} e\left(\frac{ac+bd}{q}\right).$$ Einsetzen von $a = br + k - (p-1)\left\lfloor \frac{br+k}{p-1} \right\rfloor$ ergibt:

$$P = \frac{1}{(p-1)q} \sum_{b=0}^{p-2} e\left(\frac{1}{q}\left(brc + kc - c(p-1)\left\lfloor \frac{br+k}{p-1} \right\rfloor + bd\right)\right).$$

Da $|P|^2$ betrachtet wird, spielt $e\left(\frac{kc}{q}\right)$ keine Rolle, da von b unabh., dieser Term kc kann in P o.E. weggelassen werden. Erhalte: $P = \frac{1}{(p-1)q} \sum_{b=0}^{p-2} e\left(\frac{bT}{q}\right) e\left(\frac{V}{q}\right)$, (P)

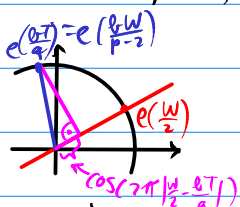
wo $T := rc + d - \frac{r}{p-1} \cdot R$, $V := \left(\frac{br}{p-1} - \left\lfloor \frac{br+k}{p-1} \right\rfloor\right) \cdot R$,

wobei $R \equiv c(p-1) \;(q)$ sei mit $-q/2 < R \leq q/2$.

Ist nun j die zu $\frac{T}{q}$ nächstgelegene ganze Zahl, nenne gemessenen Zustand $|c,d,y\rangle$ gut, falls (*) $|T - jq| \leq \frac{1}{2}$ und $|R| \leq \frac{q}{12}$ gilt. Dann ist auch $|V| \leq \frac{q}{12}$.

- Sei $|c,d,y\rangle$ gut, dann durchläuft $e\left(\frac{bT}{q}\right)$ mit $b = 0, \dots, p-2$ die Phasen $0, \dots, 2\pi i W$, wo $W = \frac{p-2}{q}(T - jq) \rightsquigarrow |W| \leq \frac{1}{2}$. (Phase = Argument φ von $\exp(\varphi)$)

Die Länge der Projektion von $e\left(\frac{bT}{q}\right)$ in Richtung $e\left(\frac{W}{2}\right)$ ist $= \cos\left(2\pi \left| \frac{W}{2} - W\frac{b}{p-2} \right|\right)$ (so: richtiges Vorzeichen).



Da $|V| \leq \frac{q}{12}$, kann in (P) mit $e\left(\frac{V}{q}\right)$ die Phase um höchstens $\frac{\pi i}{6}$ variieren.

Die Projektionslänge eines Summanden in (P) ist $\geq \cos\left(2\pi \left| \frac{W}{2} - W\frac{b}{p-2} \right| + \frac{\pi}{6}\right)$.

Der Absolutbetrag von (P) ist also $\geq \frac{1}{(p-1)q} \sum_{b=0}^{p-2} \cos\left(2\pi \left| \frac{W}{2} - W\frac{b}{p-2} \right| + \frac{\pi}{6}\right)$.

$|W| \leq \frac{1}{2} \rightarrow |\cdot| \leq \frac{1}{2} \rightsquigarrow 2\pi|\cdot| \in [0, \frac{\pi}{2}]$

- Ersetze die Σ durch ein $\int$, der Term ist $= \frac{2}{q} \int_0^{1/2} \cos\left(\frac{\pi}{6} + 2\pi|W|n\right) dn + O\left(\frac{|W|}{pq}\right)$.

Da $|W| \leq 1/2$, ist der Fehler $O\left(\frac{1}{pq}\right)$.

Das $\int$ wird minimal bei $|W| = \frac{1}{2}$, d.h. die W. für einen guten Zustand ist $\geq \left(\frac{1}{q} \frac{2}{\pi} \int_{\pi/6}^{2\pi/3} \cos u \, du\right)^2 > \underbrace{\frac{1}{20q^2}}_{\text{reicht, wie beim anderen Shor-Algo}}$. $\left(\frac{1}{2} + \frac{1}{6} = \frac{2}{3}\right)$

• Zähle nun die guten Zustände: Die #(c,d) mit $|T-jq| \leq \frac{1}{2}$ ist die # der c (d fest, damit festgelegt)
Die # c mit ⊗ ist q/6, sofern nicht $(p-1, q)$ groß. (Selbst dann: #c ist q/12, [ohne Beweis])
→ # Paare ist $\geq (q/12) \cdot \#y \approx \frac{pq}{12}$ viele gute Zustände.
Die W., guten Zustand zu messen, ist insg. $\gtrapprox \frac{pq}{12} \cdot \frac{1}{20q^2} = \frac{p}{240q} \underset{q<2p}{>} \frac{1}{480}$. (konstante reicht)

4. Schritt: Berechne r aus einem guten Zustand $|c,d,y\rangle$:
Erhalte $-\frac{1}{2q} \leq \frac{d}{q} + r \cdot \left(\frac{c(p-1)-R}{(p-1)q}\right) - j \leq \frac{1}{2q}$
Nenner $p-1$, da $q \mid c(p-1)-R$ ⇝ Bruch hat die Form $\frac{c'}{p-1}$
Somit: runde $\frac{d}{q}$ auf nächstes Vielfaches von $\frac{1}{p-1}$ (muss dann $\frac{rc'}{p-1}$ sein),
dann: teile Zähler (mod p-1) durch c'
Aber benötige: $(c', p-1) = 1$.
[Bei guten (c,d) ist dies aber recht wahrscheinlich, ohne Beweis.] ⇝ Shors Algo ist polynomiell schnell

---

11.5 **Bem.:** • Shors Algorithmus zur Berechnung des diskreten log mod p zeigt, dass die Sicherheit der auf DL beruhenden Algorithmen (Diffie-Hellman, ElGamal) hinfällig werden, sobald ein leistungsfähiger Quantencomputer zur Verfügung steht.
• Der Shor-Algo für DL konnte mittlerweile auch für das DL-Problem auf elliptischen Kurven-Gruppen übertragen werden [Proos/Zalka 2004].
Ein solcher Angriff braucht nur etwa halb so lange wie der klassische Shor-Algorithmus zur Lösung des Faktorisierungsproblems bei einem RSA-Verfahren mit vergleichbarer Sicherheit. Auch der Speicheraufwand eines Quantencomputers (Anzahl benötigter Qubits) ist dabei um (grob) den Faktor 1/3 geringer.

[Tabelle aus Proos/Zalka]

Faktorisierung			DL auf ellipt. Kurve			Klassische Zeit
n	#Qubits (=2n)	Zeit ($4n^3$)	n	#Qubits	Zeit ($360n^3$)	(C für n=110)
1024		$4.3 \cdot 10^9$	163	≈1000	$1.6 \cdot 10^9$	$C \cdot 10^8$
2048		$34 \cdot 10^9$	224	1300	$4.0 \cdot 10^9$	$C \cdot 10^{17}$
3072		$120 \cdot 10^9$	256	1500	$6.0 \cdot 10^9$	$C \cdot 10^{22}$
15360		$1.5 \cdot 10^{13}$	512	2800	$50 \cdot 10^9$	$C \cdot 10^{60}$

- Bei Aufkommen von Quantencomputern werden die ECC-Verfahren daher Jahre vor den entsprechenden RSA-Verfahren geknackt sein. Dies ist ein Grund, warum wieder eher zur RSA-Technologie geraten wird (mit entsprechend hoher Schlüssellänge).
- Der Shor-Algo von K10 kann auch direkt zur Lösung von DL in $\mathbb{Z}_p^*$ eingesetzt werden: Ist wie oben g eine PW mod p, weiter x geg. und r mit $g^r \equiv x \ (p)$ gesucht, kann mit dem Shor-Algo von K10 erst $s = ord_p(x)$ bestimmt werden, also s minimal mit $x^s \equiv 1$, dann ist $g^{rs} \equiv x^s \equiv 1$, also $p-1 \mid rs$, etwa $rs = (p-1) \cdot m$. Dann ist $r = \frac{p-1}{s} \cdot m$. Das kleinste solche $m \leq s$ kann durch sukzessives Probieren ermittelt werden, vor allem wenn s eher klein ist. Klappt das nicht, kann man demnach so verfahren wie in 11.4 beschrieben.
- 2019 wurde von Google AI und NASA die "Quantum supremacy" verkündet; die Beh., mit einer 54-Qubit-Maschine eine Berechnung durchgeführt zu haben, die auf klassischen Computer unmöglich gewesen wäre. Inwieweit das richtig ist, wird bis heute kontrovers diskutiert und erforscht.