From linear algebra to post quantum cryptography

Steven Galbraith

Cyber Security Foundry & Mathematics Department

University of Auckland, New Zealand

Acknowledgements

- Presidents of NZMS, past and present.
- Conference organisers: Tammy Lynch, Chris Tuffley, Richard Brown, Christine Burr, Luke Fullard, Robert McLachlan, Cami Sawyer, David Simpson, Bruce van Brunt, Nicholas Witte.
- My supervisors (Kevin Broughan, Steve Demko, Bryan Birch, Fred Piper, Peter Wild, Alfred Menezes, Gerhard Frey, Hans-Georg Rück,...), teachers, and colleagues.
- My co-authors and students, especially (for this talk) Shi Bai.
- University of Auckland Faculty Research Fund, Marsden Fund.
- You, for attending a talk on the morning after the conference dinner.

What is Applied Mathematics?

Applied Mathematics is not a definable scientific field but a human attitude . . . (t)he(y) must be willing to make compromises regarding rigorous mathematical completeness; (t)he(y) must supplement theoretical reasoning by numerical work, plausibility considerations and so on.
– *Courant (1965)*

The motivation of the applied mathematician is to understand the world and perhaps to change it . . . techniques are chosen for and judged by their effectiveness (the end is what's important); and the satisfaction comes from the way the answer checks against reality and can be used to make predictions.
– *Paul Halmos*

Plan of the rest of the talk

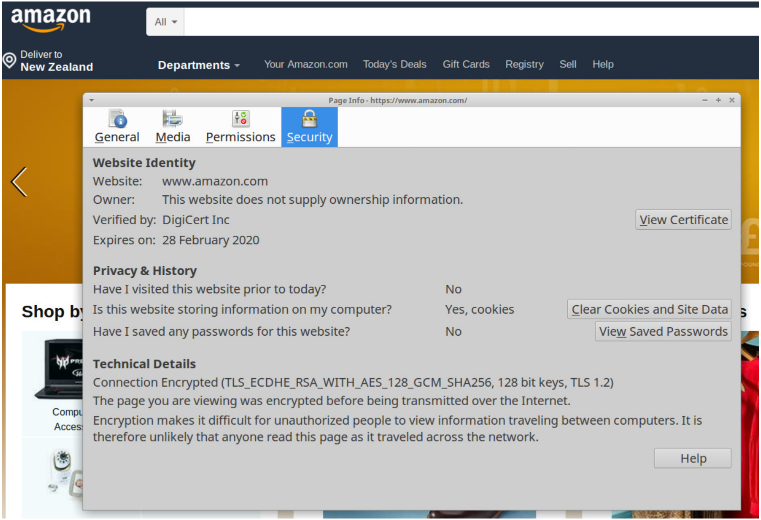
- Cryptography
- Quantum algorithms
- Post-quantum public key cryptography
- Lattices
- Signatures
- Proof of knowledge

Please ask questions

Digital signatures

- Public key cryptography solves the authentication problem: How can I be certain of the sender?
- Automatic software updates:
  "Please install me on your computer. It's OK, I am from Microsoft."
- Normal signatures are no good, because an attacker can cut-and-paste.
- A digital signature on a file is created using the secret and it depends on the file.
  A digital signature can be verified using only the public key.

Internet shopping



Quantum Computing

- Quantum computing was proposed by: Paul Benioff (1980), Yuri Manin (1980), Richard Feynman (1982) and David Deutsch (1985).
- Peter Shor (1994): polynomial-time quantum algorithm for integer factorisation and discrete logs.
- Late 1990s: Breakthrough in quantum computing around "10 years away".
- Dave Wecker (Microsoft) invited talk at PQ Crypto 2018: Microsoft will have a quantum computer suitable for chemistry applications within 5 years and "something of interest to this crowd" in 10 years.

Quantum computer or microbrewery?

Quantum computer or microbrewery?

GIZMODO AU allure Network

SHARE TWEET

Reviews | Smart Home & Gadgets | Mobile | Car Tech | Online | Science & Health | Cameras | Computing | Gaming | Entertainment

Want Gizmodo's email newsletter?

Enter your email address Subscribe

Alert me about competitions & deals

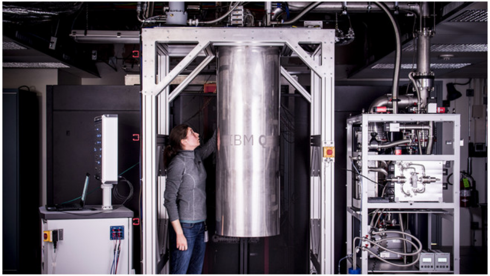
Follow us, subscribe and get in touch!

IBM's Newest Quantum Computers Are The Most Powerful Of Their Kind

Ryan F. Mandelbaum

Nov 11, 2017, 9:00am· Filed to: ibm

Share



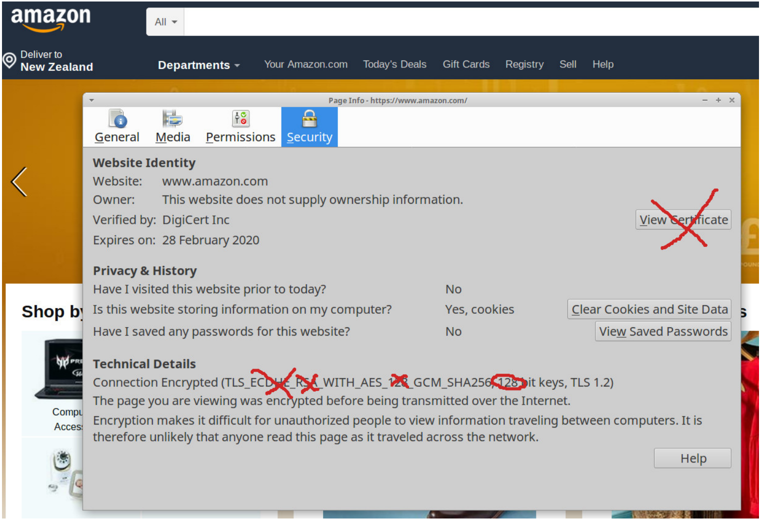
IBM has announced two powerful new quantum computer processors, one client-ready and another in the works.

An IBM quantum computer cryostat (Image: Andy Aaron, IBM via Flickr)

Today's announcement includes both a 20 qubit processor ready for use by its IBM Q clients

Latest Deals

Internet shopping when bad guy has a quantum computer



Post-quantum cryptography (PQC)

- PQC means cryptosystems that can be implemented using *current computing and communication channels*, but are secure against an *adversary with a quantum computer*.
- There is a totally different subject called *quantum cryptography*, which is secure communication using quantum devices.

NIST post-quantum standardisation process

- August 2015: NSA Information Assurance Directorate proposed "a transition to quantum resistant algorithms in the not too distant future".
- February 2016: NIST preliminary announcement of standardization plan.
- November 2017: Submission deadline (69 submissions accepted).
- Mathematical foundation: Lattices, coding-theory, multivariate polynomial systems, hash trees, non-abelian groups, isogenies.
- January 2019: Second round selections announced (26 selected).
- Draft standards expected around 2023-2025.

Hermann Minkowski (1864-1909)

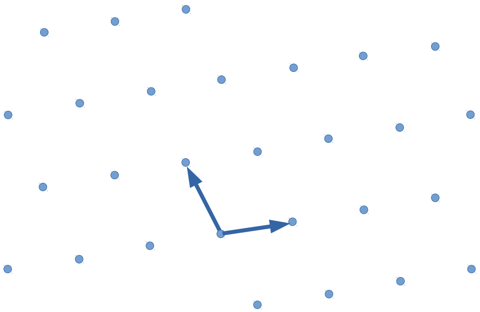
- Showed that special theory of relativity is best understood geometrically as a theory of four-dimensional spacetime (now known as "Minkowski spacetime").
- Pioneered the "geometry of numbers" to prove results in number theory (such as the finiteness of the ideal class group).

[credit: wikipedia]

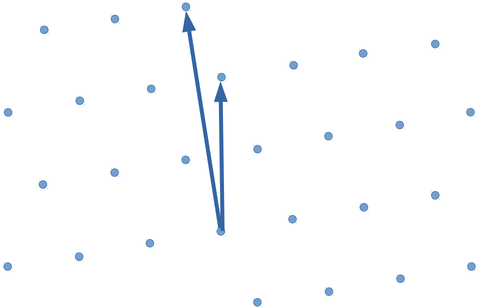
Lattices

- Let $\underline{b}_1, \ldots, \underline{b}_m$ be linearly independent **column** vectors in $\mathbb{R}^m$.
- The set $L = \{\sum_{i=1}^m x_i \underline{b}_i : x_i \in \mathbb{Z}\}$ is a (full rank) lattice. Call its elements **points** or **vectors**.

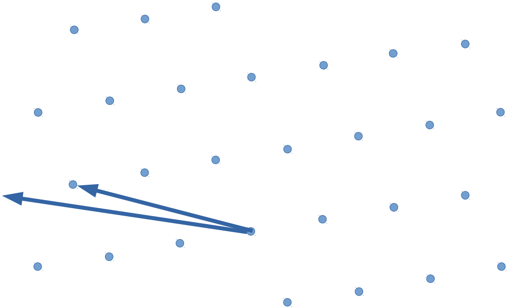
Lattice



Lattice



Lattice



Lattices

- Let $\underline{b}_1, \ldots, \underline{b}_m$ be linearly independent **column** vectors in $\mathbb{R}^m$.
- The set $L = \{\sum_{i=1}^m x_i \underline{b}_i : x_i \in \mathbb{Z}\}$ is a (full rank) lattice. Call its elements **points** or **vectors**.
- The **basis matrix** is the $m \times m$ matrix B whose columns are the vectors $\underline{b}_1, \ldots, \underline{b}_m$.)
- A lattice has many different bases, but the volume $(|\det(B)|)$ is invariant.
- For computational reasons we work with lattices in $\mathbb{Z}^m$.

Computational Problems (Informally)

- Shortest vector problem (SVP): Given a basis matrix B for a lattice $L \subseteq \mathbb{Z}^m$ find a non-zero vector $\underline{v} \in L$ such that $\|\underline{v}\|$ is minimal.
  The norm here is usually the Euclidean norm in $\mathbb{R}^n$, but it can be any norm such as the ℓ_1 norm or ℓ_∞ norm.
- SVP with the ℓ_∞ norm is NP-hard.
- Closest vector problem (CVP): Given a basis matrix B for a full rank lattice $L \subseteq \mathbb{Z}^m$ and an element $\underline{t} \in \mathbb{R}^m$ find $\underline{v} \in L$ such that $\|\underline{v} - \underline{t}\|$ is minimal.
- These problems are believed to be hard for *quantum computers* when dimension m is high.

Short integer solution problem (Ajtai 1996)

- Let q be prime.
- Let A be an $n \times m$ integer matrix, where $m > n \log_2(q)$.
- Let $\underline{x} \in \{-1, 0, 1\}^m$ be a vector.
- Let $\underline{b} \equiv A\underline{x} \pmod{q}$. So $\underline{b} \in \mathbb{Z}_q^n$.
- Given $(A, \underline{b})$, compute $\underline{x}$.
- This is a hard lattice problem.

Solving $\underline{b} \equiv A\underline{x} \pmod{q}$ (optional)

- Let $L = \{\underline{y} \in \mathbb{Z}^m : A\underline{y} \equiv 0 \pmod{q}\}$, which is a lattice.
- Compute any $\underline{z} \in \mathbb{Z}^m$ such that $A\underline{z} \equiv \underline{b} \pmod{q}$.
- Find a close lattice vector $\underline{y} \in L$ to $\underline{z}$.
- Set $\underline{x} = \underline{z} - \underline{y}$, so that $\|\underline{x}\|$ is short.
- Then $A\underline{x} = A\underline{z} - A\underline{y} \equiv \underline{b} \pmod{q}$.

Proving knowledge of a secret

- Can I prove to you that I know a secret, without telling you (or anyone else)?
- For the rest of the talk I describe such an interactive protocol for this lattice problem.
- The main ideas are due to Lyubashevsky (2009, 2012).
- Let $(A, \underline{b})$ be public. Let $\underline{x}$ be a secret short vector such that $A\underline{x} \equiv \underline{b} \pmod{q}$.
- I want to be able to convince you that I know the short vector $\underline{x}$, without telling you $\underline{x}$.
- One can build a digital signature from this interactive protocol.

Interactive protocol to prove knowledge of the solution (toy)

Prover			Verifier
Short vector $\underline{x}$			$(A, \underline{b})$
Choose short $\underline{y} \in \mathbb{Z}^m$			
Set $Y = A\underline{y} \pmod{q}$	$\xrightarrow{Y}$	$\xrightarrow{Y}$	
			Choose small $c \in \mathbb{Z}$
	$\xleftarrow{c}$	$\xleftarrow{c}$	
$\underline{z} = \underline{y} + \underline{x}c$	$\xrightarrow{\underline{z}}$	$\xrightarrow{\underline{z}}$	Check $\Vert\underline{z}\Vert$ short(-ish) and $A\underline{z} \equiv Y + \underline{b}c \pmod{q}$

Interactive protocol to prove knowledge of the solution (toy)

- I need to show there is no forger who can impersonate me.
- The forger knows $(A, \underline{b})$, but does not know $\underline{x}$.
- Treat the forger as an algorithm that takes $(A, \underline{b})$ as input.
- Want to show that if a forger exists then there is an algorithm to find a short $\underline{x} \in \mathbb{Z}^m$ such that $\underline{b} \equiv A\underline{x} \pmod{q}$.

Security of the protocol

Forger $(A, \underline{b})$		Verifier $(A, \underline{b})$
$\xrightarrow{Y}$	$\xrightarrow{Y}$	
		Choose small $c \in \mathbb{Z}$
$\xleftarrow{c}$	$\xleftarrow{c}$	
$\xrightarrow{\underline{z}}$	$\xrightarrow{\underline{z}}$	Check $\|\underline{z}\|$ short(-ish) and $A\underline{z} \equiv Y + \underline{b}c \pmod{q}$

Interactive protocol to prove knowledge of the solution (toy)

- If forger knows what c the verifier will send, they can cheat:
  - Choose a random short vector $\underline{z}$
  - Set $Y = A\underline{z} - \underline{b}c \pmod{q}$
- But since the forger doesn't know c before they send Y, then the protocol should be convincing.
- If c is from a small set then the protocol may need to be repeated many times.
- (Real schemes use matrices or ring elements for $\underline{x}$ and c.)

Security of the protocol

Forger $(A, \underline{b})$			Verifier $(A, \underline{b})$
	$\xrightarrow{Y}$	$\xrightarrow{Y}$	
			Choose small $c \in \mathbb{Z}$
	$\xleftarrow{c}$	$\xleftarrow{c}$	
	$\xrightarrow{\underline{z}}$	$\xrightarrow{\underline{z}}$	Check $\lVert \underline{z} \rVert$ short(-ish) and $A\underline{z} \equiv Y + \underline{b}c \pmod{q}$

Proof that this is a proof of knowledge (attempt 1)

- The verifier gets a vector Y from the forger.
- No matter which small integer c is chosen by the verifier, the forger can respond with a short-ish vector $\underline{z}$ such that $A\underline{z} \equiv Y + \underline{b}c \pmod{q}$.
- We suppose verifier can choose two challenges c_1, c_2 for same Y and get corresponding two responses $\underline{z}_1, \underline{z}_2$.
- We have $A\underline{z}_1 \equiv Y + \underline{b}c_1 \pmod{q}$ and $A\underline{z}_2 \equiv Y + \underline{b}c_2 \pmod{q}$.
- So $A(\underline{z}_1 - \underline{z}_2) \equiv \underline{b}(c_1 - c_2)$.
- Hence $\underline{x} = (\underline{z}_1 - \underline{z}_2)(c_1 - c_2)^{-1} \pmod{q}$ is a solution to the equation $A\underline{x} \equiv \underline{b} \pmod{q}$.
- **Problem:** $\underline{x}$ may not be short.

Second attempt: New computational problem

- New computational problem: Given an $n \times m$ matrix A, find a short (but non-zero) vector $\underline{w}$ such that $A\underline{w} \equiv 0 \pmod{q}$.
- This is also a lattice problem.
- Lyubashevsky showed that if there is a forger for the identification scheme, then there is an algorithm to solve this problem.

Proof that this is a proof of knowledge

- Let A be such a matrix.
- Choose a short vector $\underline{x} \in \mathbb{Z}^m$ and set $\underline{b} \equiv A\underline{x} \pmod{q}$.
- Run forger as before on input $(A, \underline{b})$, to get two responses $\underline{z}_1, \underline{z}_2$ for challenges c_1, c_2 for same Y.
- We have $A\underline{z}_1 \equiv Y + \underline{b}c_1$ and $A\underline{z}_2 \equiv Y + \underline{b}c_2 \pmod{q}$.
- So $A(\underline{z}_1 - \underline{z}_2) \equiv \underline{b}(c_1 - c_2) \equiv (A\underline{x})(c_1 - c_2) \pmod{q}$.
- Since we know $\underline{x}$, we have

$$A(\underline{z}_1 - \underline{z}_2 - \underline{x}(c_1 - c_2)) \equiv 0 \pmod{q}.$$

- Let $\underline{w} = \underline{z}_1 - \underline{z}_2 - \underline{x}(c_1 - c_2)$. So $\underline{w}$ is short and $A\underline{w} \equiv 0 \pmod{q}$.
- **Problem:** $\underline{w}$ may be zero.

Proof that this is a proof of knowledge

- Tweak the parameters and the computational assumption, so that that there are many short vectors $\underline{x}'$ such that $\underline{b} \equiv A\underline{x}' \pmod{q}$.
- The forger gets $(A, \underline{b})$, but has no way to know which of the possible vectors $\underline{x}$ we have chosen.
- It can be shown that with non-negligible probability $\underline{w} = \underline{z}_1 - \underline{z}_2 - \underline{x}(c_1 - c_2)$ is non-zero.
- In conclusion: If it is hard to find short non-zero kernel vectors of random integer matrices then it is hard to fake this interactive protocol.

Zero knowledge

- We also have to worry about whether $\underline{z}$ leaks the secret $\underline{x}$.
- Since $\underline{z} = \underline{y} + \underline{x}c$ where c is known, then a statistical analysis might allow an attacker to determine $\underline{x}$.
- This is prevented by taking the entries of $\underline{y}$ to be a discrete Gaussian, and using rejection sampling. (Lyubashevsky 2009)

Discrete Gaussians

- A discrete Gaussian on $\mathbb{Z}^m$ with parameter σ^2 is a distribution such that the probability of $\underline{x} \in \mathbb{Z}^m$ is proportional to

$$\exp(-\|\underline{x}\|^2/(2\sigma^2)).$$

- If $\underline{y}$ and $\underline{x}$ are sampled from continuous Gaussians with parameters (variances) σ_1^2 and σ_2^2 respectively, then $\underline{y} + \underline{x}$ is distributed as a continuous Gaussian with parameter $\sigma_1^2 + \sigma_2^2$.
- This statement is no longer true for discrete Gaussians.
- In our applications, the distribution of $\underline{z} = \underline{y} + \underline{x}c$ is important.

Discrete Gaussians

- Let $\underline{x}$ be sampled from a continuous Gaussian on $\mathbb{Z}^m$ with parameter σ^2 and let X be an $n \times m$ matrix. Then $\underline{y} = X\underline{x}$ has distribution with probability proportional to

$$\exp(-\underline{x}^T X^T X \underline{x}/(2\sigma^2)).$$

- The matrix $X^T X$ is called the Gram matrix.
- If $\underline{x}$ are sampled from a discrete Gaussian with parameter σ^2 then this statement is no longer true.
- Significant focus in cryptography research to get precise estimates of these distributions, and distributions like $\underline{y} + X\underline{x}$ etc.

Mathematical tools that have been introduced to cryptography in recent years

- Sampling algorithms for approximating probability distributions.
- Convolution theorems.
- Algorithms to compute Cholesky decompositions.

Some of my work in this area

- Shi Bai and Steven D. Galbraith, “An Improved Compression Technique for Signatures Based on Learning with Errors”, in J. Benaloh (ed.), CT-RSA 2014, Springer LNCS 8366 (2014) 28–47.
- Shi Bai, Steven D. Galbraith, Liangze Li and Daniel Sheffield, “Improved Combinatorial Algorithms for the Inhomogeneous Short Integer Solution Problem”, Journal of Cryptology, Volume 32, Issue 1 (2019) 35–83.
- Leo Ducas, Steven Galbraith, Thomas Prest and Yang Yu, “Integral matrix sums of squares and lattice Gaussian sampling without floats”, submitted.

What kind of mathematics is this?

- Pure Mathematics?
- Computer Science?
- Applied Mathematics?

Olga Taussky-Todd (1906-1995)

- Trained in algebraic number theory, and later worked on matrix theory and numerical analysis.
- "When people look down on matrices, remind them of great mathematicians such as Frobenius, Schur, Siegel, Ostrowski, Motzkin, Kac etc who made important contributions to the subject. I am proud to have been a torchbearer for matrix theory."

[credit: wikipedia]

What should we be teaching our students?

(Not just for cyber security, but also data science, finance, etc)

- Linear algebra
- Numerical methods
- Probability
- Statistics
- Discrete Mathematics
- Calculus

The End

"Mathematics is more unified than Mathematicians"
– Robbert Dijkgraaf

"I believe that it is vital to counteract these dangerous tendencies by fighting over-specialization and fragmentation of mathematics and by a vigorous effort at building bridges between the diverging mathematical fields"
– Richard Courant

"Mathematics, despite its many subdivisions and their enormous rate of growth is an amazingly unified intellectual structure"
– Paul Halmos