暗号技術の現在

ポスト量子暗号への移行と量子暗号

If computers that you build are quantum,
Then spies everywhere will all want 'em.
Our codes will all fail,
And they'll read our email,
Till we get crypto that's quantum, and daunt 'em.

– Jennifer and Peter Shor

To read our E-mail, how mean
of the spies and their quantum machine;
be comforted though,
they do not yet know
how to factorize twelve or fifteen.

– Volker Strassen

Unfortunately, the growth of elliptic curve use has bumped up against the fact of continued progress in the research on quantum computing, which has made it clear that elliptic curve cryptography is not the long term solution many once hoped it would be. Thus, we have been obligated to update our strategy.

-- NSA

I estimate a 1/7 chance of breaking RSA-2048 by 2026 and a 1/2 chance by 2031.

-- Michele Mosca

はじめに

- 暗号には「秘密」がつきものであった。1955年、数学者のJohn Nashは、NSAに対して「鍵の計算に指数関数的時間のかかる」方法を使えば、「誰にも破れない暗号を簡単に作れるようになる」という手紙を送っている。1970年代の初めには、イギリスの情報機関GCHQの科学者たちが、今日の公開キー暗号と同じものを作り上げていた。それはNSAも知っていた。ただ、それらは全て「機密」とされ、世に知られることはなかった。
- なぜ、「機密」にされたかといえば、securityの問題は、第一義的に”National Security”の問題だからということなのだが、なぜそうした技術が、広く実用化されなかったという点に関して言えば、別の問題があったことに気づく。それは、当時のコンピュータには、少なくとも経済的には、こうした複雑な計算を実行する計算能力が十分ではなかったのである。

はじめに

- 現在の暗号化技術の基本が出来上がるのは、1976/1977年のことである。その技術の真価は、80年代のコンピュータのコンシューマライゼーション（PC-AT, Windows, …）を経て、90年代半ばのインターネットのグローバルな拡大、経済活動のグローバル・ネットワーク化、個人のネットワークへの登場を通じて、全面的に開花する。この技術なしでは、今日のネットワークの成功はなかったろう。
- このことは、同時に、暗号化技術が、もはや、もっぱら "National Security"のみにかかわる技術ではなくなったことを意味する。今では、誰もが暗号化技術を必要とし、誰もがそれを、オープンな「標準技術」として利用できる。それは、大きな変化である。
- 暗号化技術は、その時代で利用可能なコンピュータの計算能力、その時代で利用可能な通信基盤に大きく依存している。それは歴史的に変化するものだと僕は考えている。

はじめに

- 現代の暗号化技術の大きな特徴は、その基礎を、コンピュータはある種の数学的問題を、「効率的」には解くことができないという（経験的）事実においていることである。その理論的基礎は、数学的には、「計算複雑性の理論」である。
- 現代の暗号技術の基礎が確立して約20年後の1994年、ベル研究所のPeter Shorは、量子コンピュータを使えば、RSA暗号の基礎である素因数分解問題を、極めて「高速に」解くことができるアルゴリズムを発見する。楕円曲線暗号の基礎である離散対数問題も、このアルゴリズムで「高速に」解ける。
- この発見は、当時、一大センセーションを巻き起こした。しかし、その熱もやがて冷める。というのも、そのアルゴリズムを実現するための量子回路を構築することが、当時の技術では「事実上不可能」であることが、次第に明らかになったからだ。

はじめに

- こうして、まだ存在しない量子コンピュータは、現在の暗号化技術に対する「当面の脅威ではない」という認識が一般には広く共有されることになる。それもある意味では当然である。先にも触れたように、暗号技術は「その時代で利用可能なコンピュータの計算能力、その時代で利用可能な通信基盤に大きく依存している」からである。
- ただ、現代の暗号化技術が、「計算複雑性理論」に依存しているのも事実である。現在のコンピュータでは効率的には計算できないが、量子コンピュータでは効率的に計算可能な複雑性のクラスが存在することを具体的に示した、Shorの発見は、研究者の強い関心を途切れることなく引き続けた。
- 21世紀に入って、量子コンピュータの研究はさらに広がり、Google, IBM, Microsoft, Intelといったベンダーがこの領域に参入するとともに、Shorのアルゴリズムの「実現可能性」

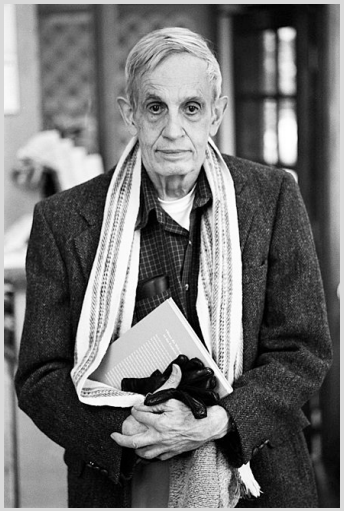
はじめに

- このような状況を受けて、2015年、NSAは次のような重要な決定を公表した。Shorの発見から、約20年後のことであった。「我々は、来るべき量子耐性アルゴリズムへの移行について、早いうちから計画づくりとコミュニケーションを開始することを決定した。我々の最終的な目標は、量子コンピュータの潜在的な能力に対して、コスト効率の良いセキュリティを提供することである。」
- NSAの決定を受けて、NISTは2016年から “Post-Quantum Cryptography”の標準化の策定の作業を開始した。NISTの動向については本セミナーで紹介したいと思う。その計画によれば、早ければ2022年、遅くとも2024年までには、新しい「ポスト量子暗号技術」の標準を決定するという。暗号技術の現在について、最も重要な動きは、このNISTのプロジェクトであると、僕は考えている。

はじめに

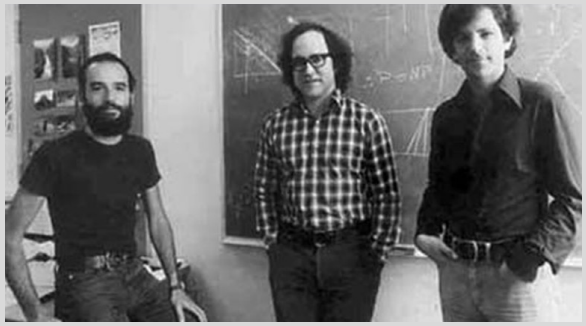
- ただ、それが全てではないだろうと、僕は思う。暗号技術は、時代とともに技術の到達点とともに、変化するものだ。暗号技術の歴史を見れば、ほぼ20年ごとに、大きな変化が起きているのがわかる。
- 一つの問題は、IT技術者にとって、20年とか40年のスパンで技術の未来を考えることが、なかなか難しいことだ。それは誰にとっても同じだと思う。しかし、未来を予測することなしに、未来の技術の準備はできない。
- セミナーの最後に、僕の予想を述べたいと思う。
  現在のコンピュータと量子コンピュータのハイブリッドの時代、また、
  現在の通信技術と量子通信のハイブリッドの時代は、新しい暗号技術を生み出すだろう。そこでは、「計算の複雑さ」ではなく、量子の性質に直接依拠した「量子暗号」が、量子

1955



John Nash

1976/1977



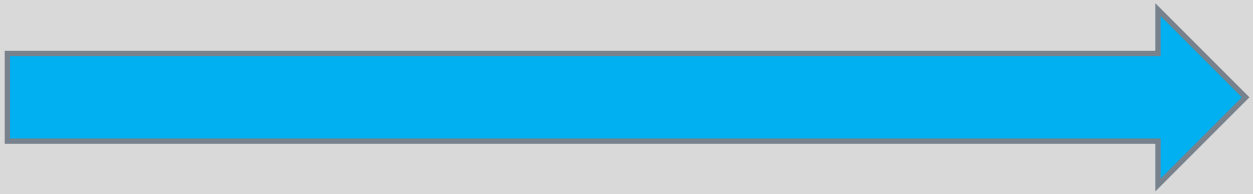
公開キー暗号/ RSA暗号

1994

Peter Shor

2015/2016

Post-Quantum Cryptography



Agenda

- 現代暗号技術成立以前
- 現代の暗号技術の成立
- 暗号と計算複雑性理論
  「計算の難しさ」とは何か

第一部

- 量子コンピュータと暗号
  Shorの素因数分解アルゴリズムの発見
- “Post-Quantum Cryptography”
  標準化の動向

第二部

- 量子暗号 -- 量子の特徴を暗号技術に応用する
- 古典量子ハイブリッドの時代を展望する
- Appendix 量子貨幣

第三部

Agenda 第一部

- 現代暗号技術成立以前
  - 第二次大戦期の暗号技術
  - 言語と暗号
  - 機械翻訳技術と暗号
- 現代の暗号技術の成立
  - 計算の複雑さを暗号に利用する
    理論的に先行したもの -- NashとGödel
  - RSA暗号と公開キー暗号の登場
    素因数分解の難しさと暗号への応用
- 暗号と計算複雑性理論　--「計算の難しさ」とは何か
  - 複雑性理論入門
  - 一方向関数
  - 言語と複雑性

Agenda 第二部

- 量子コンピュータと暗号 -- Shorのアルゴリズムの発見
  - ショアのアルゴリズム – 古典コンピュータ編 概論
  - ショアのアルゴリズム – 量子コンピュータ編 概論
  - ショアのアルゴリズム と複雑性理論
- “Post-Quantum Cryptography” 標準化の動向
  - NSA
    Commercial National Security Algorithm Suite
  - NISTIR 8105
    ”Report on Post-Quantum Cryptography”
  - NISTIR 8240
    “Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process”
  - NIST Presentation at PQCrypt

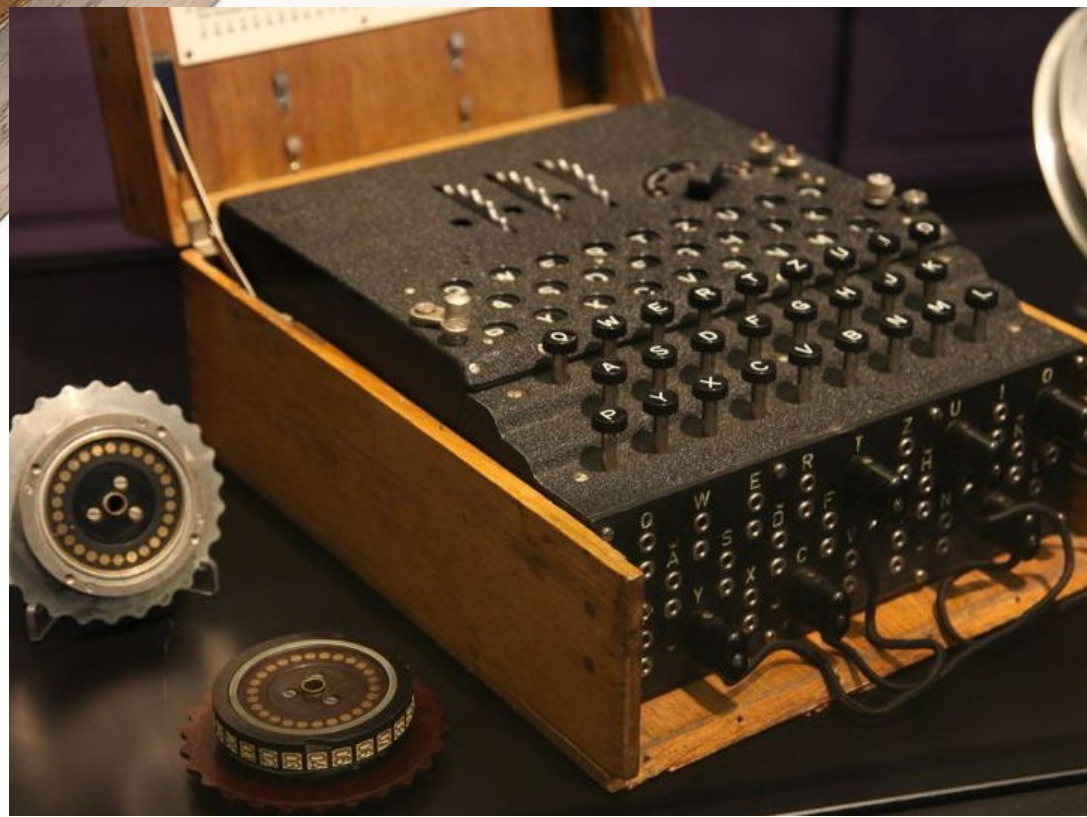
Agenda 第三部

- 量子暗号 -- 量子の特徴を暗号技術に応用する
  - 暗号技術にとって好ましい量子の二つの性質
  - Quantum Key Distribution
    BB84
- 古典量子ハイブリッドの時代を展望する
- Appendix 量子貨幣
  - 秘密キー量子通貨
  - 公開キー量子通貨

現代暗号技術成立以前

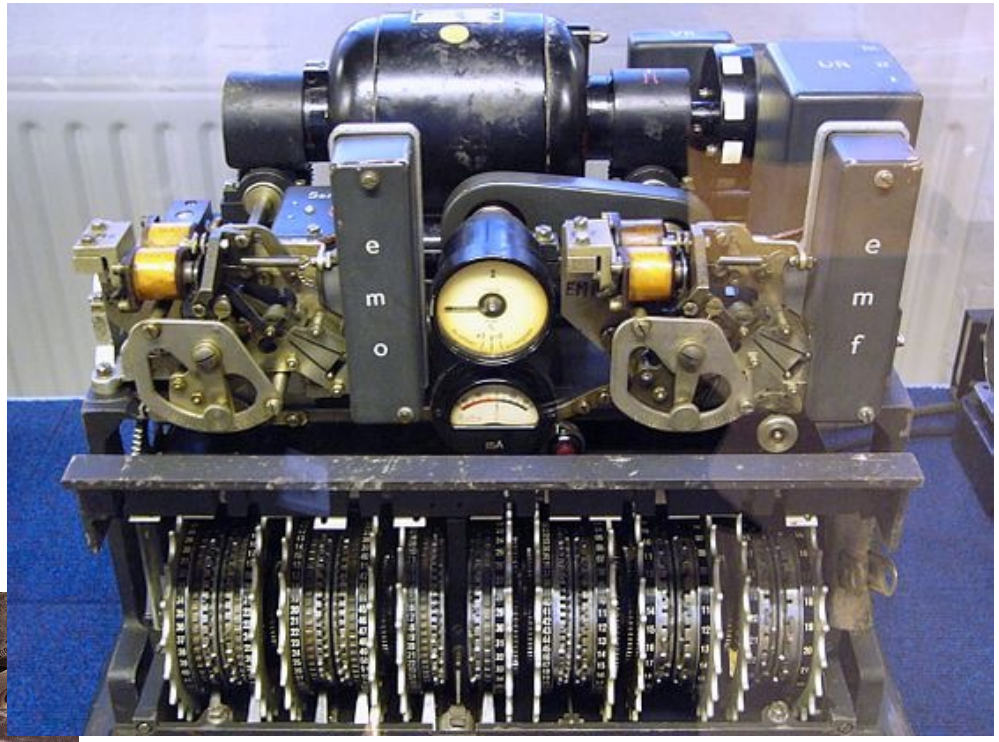
第二次大戦期の暗号技術

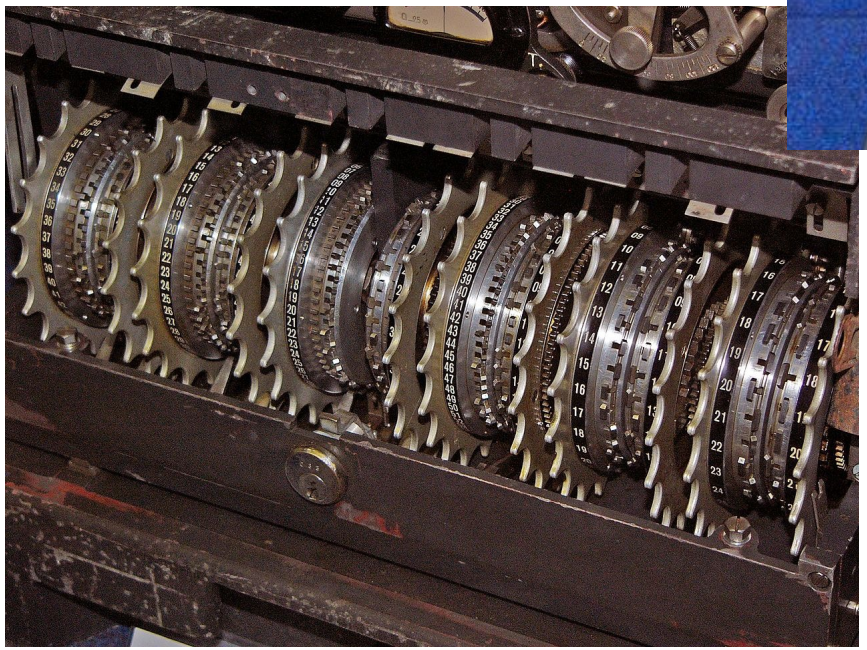
ドイツ
Enigma



プラグボード付き
Enigma

ドイツ
Lorenz SZ42

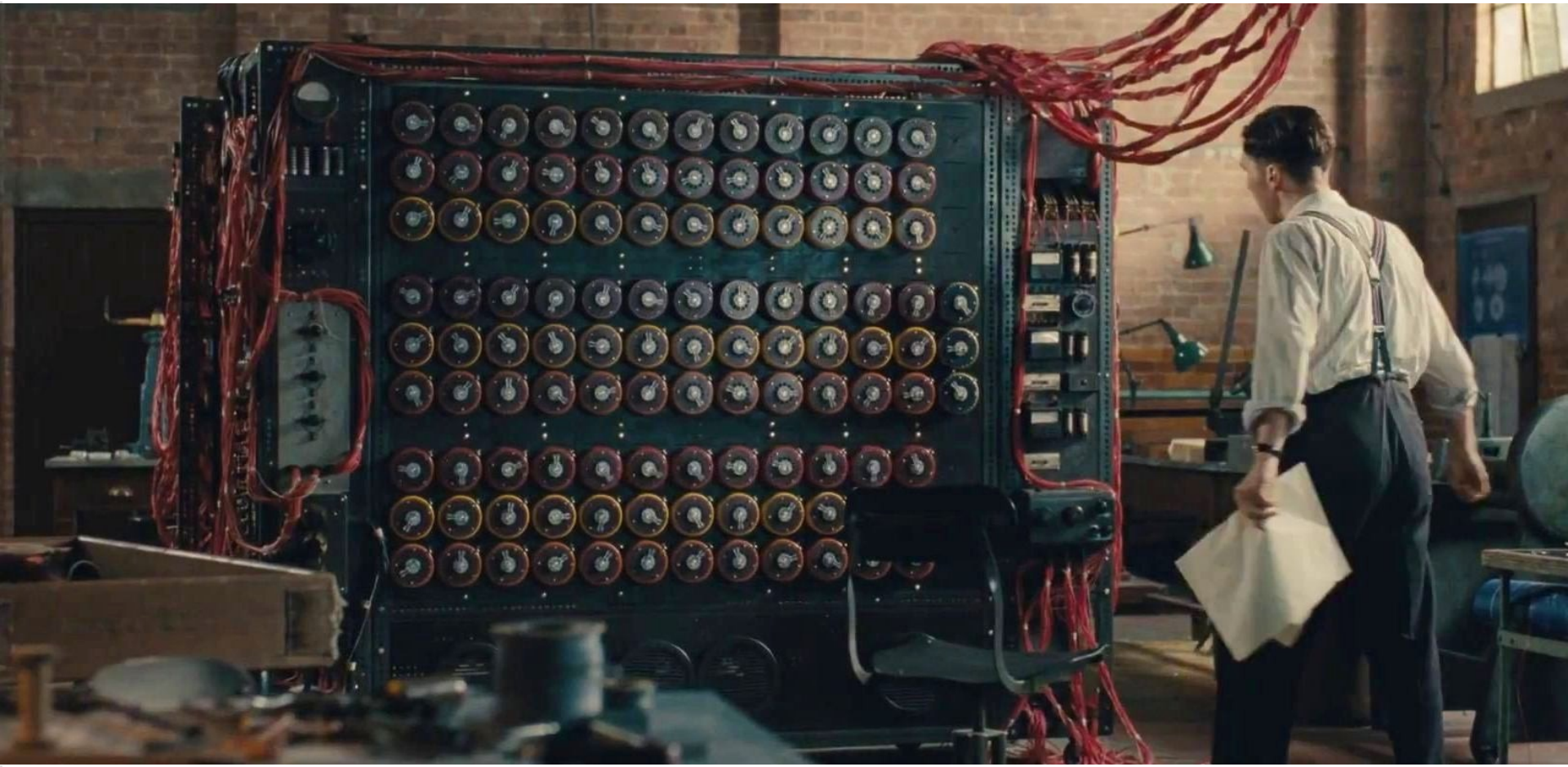




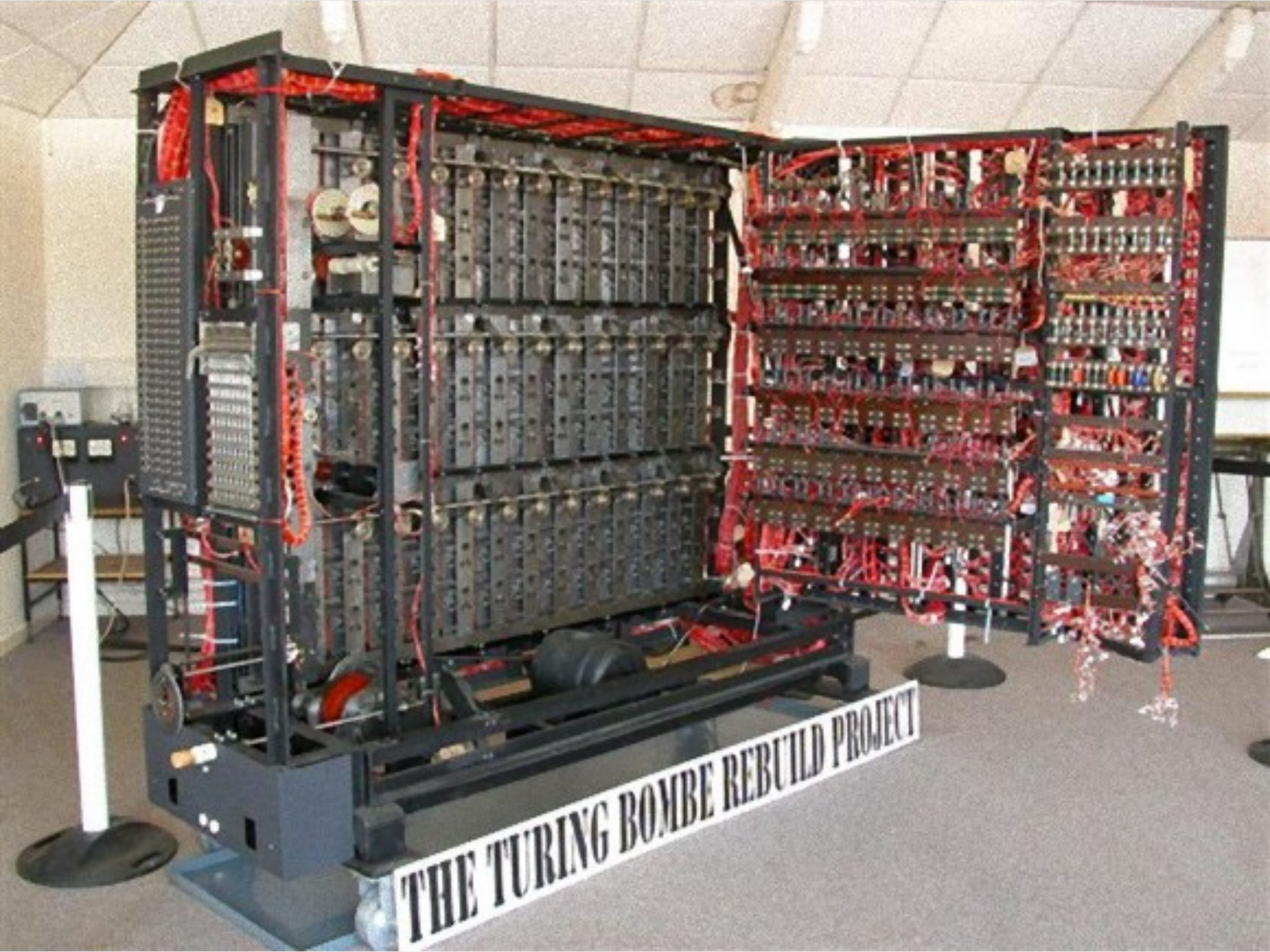
The Lorenz SZ machines had 12 wheels each with a different number of cams (or "pins").

https://en.wikipedia.org/wiki/Lorenz_cipher

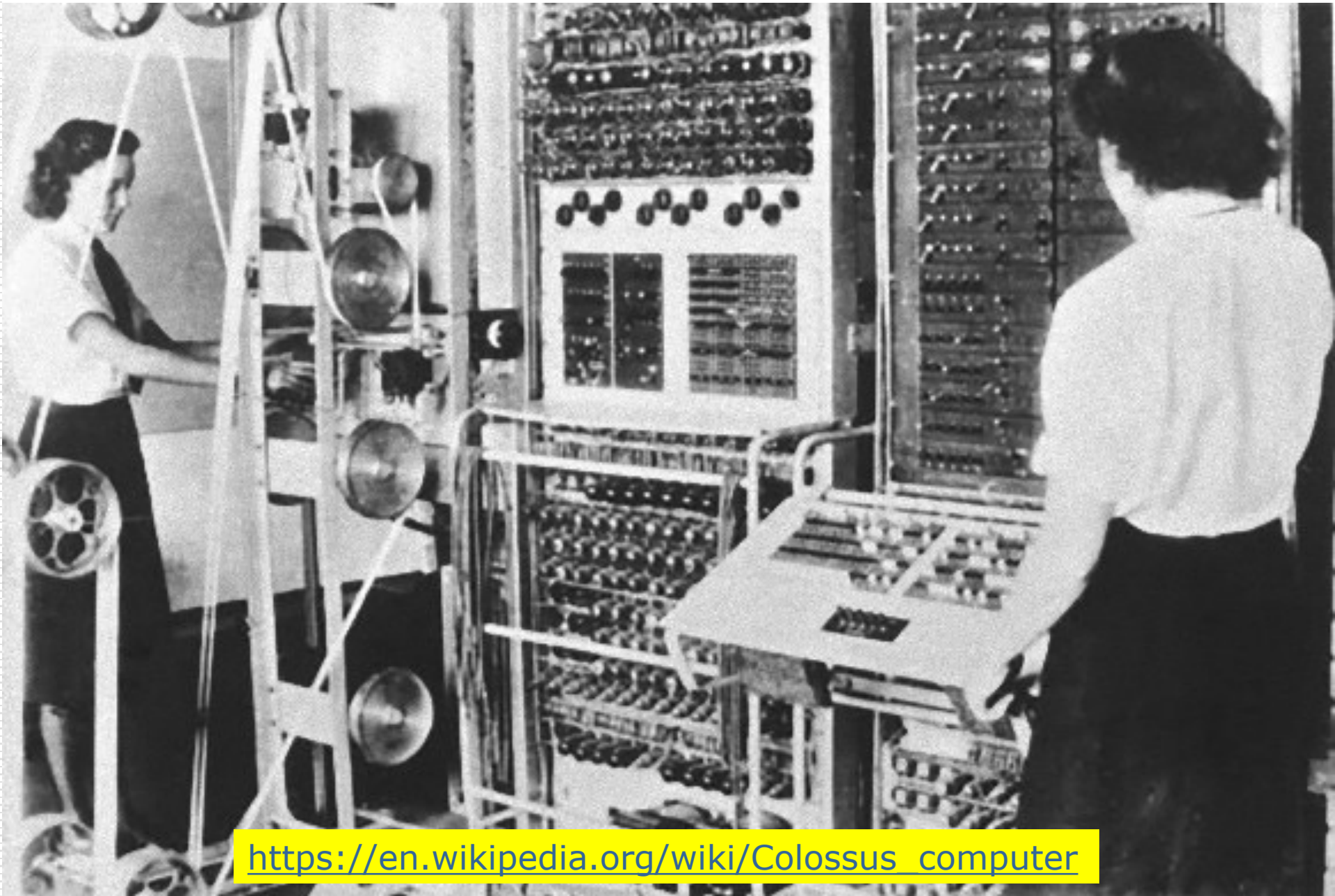
Turing Bombe (=>Enigma)



https://en.wikipedia.org/wiki/Bombe


THE TURING BOMBE REBUILD PROJECT

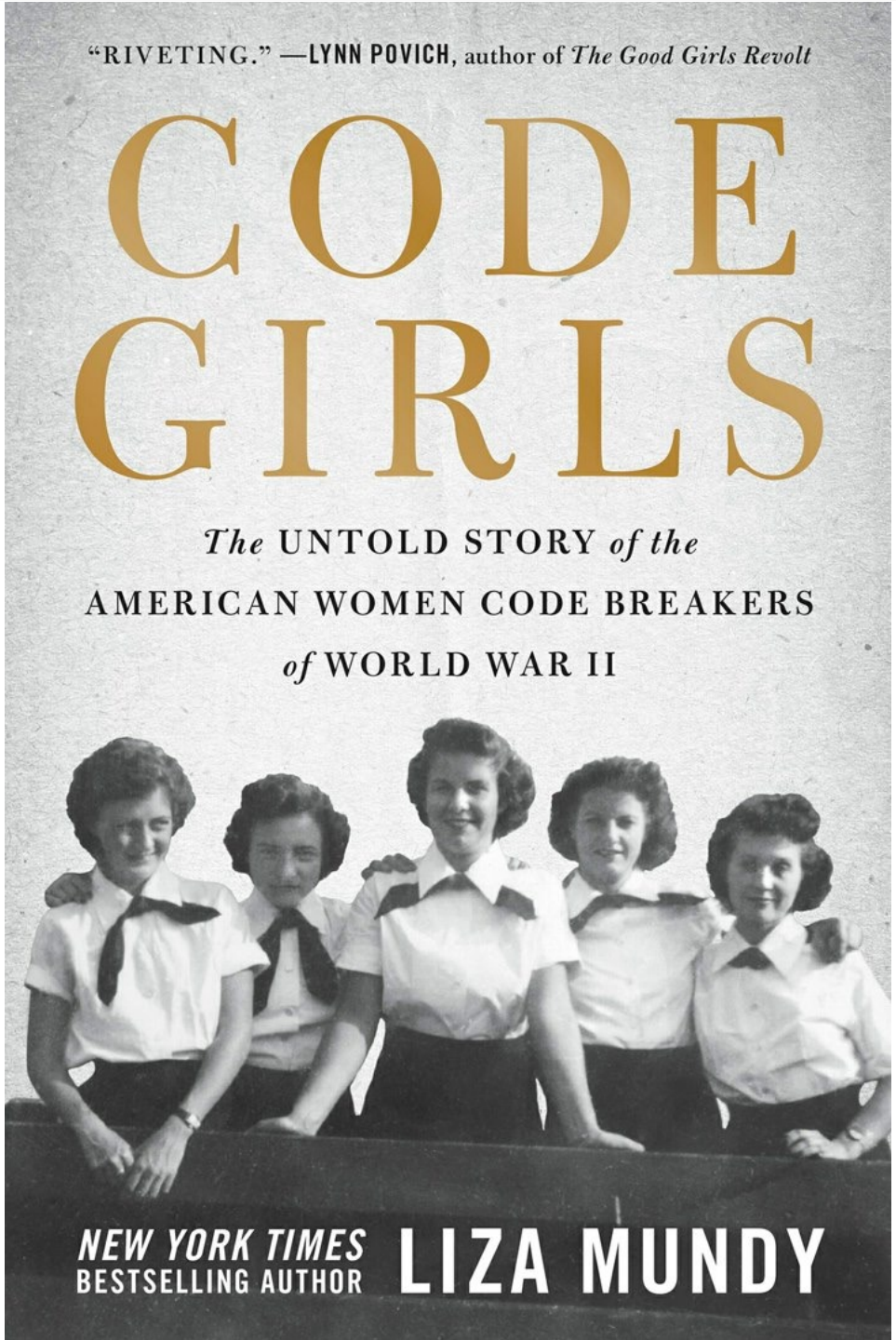
Colossus (=> Lorents SZ42)



日本　Purple暗号

Type B Cipher Machine", codenamed Purple by the United States, was a encryption machine used by the Japanese Foreign Office from February 1939 to the end of World War II.

https://en.wikipedia.org/wiki/Type_B_Cipher_Machine



Genevieve Grotjan

日本のPurple暗号を破る

https://amzn.to/2K1N23g

Code Talker

“コードトーカー(Code talker)とは、
国外にはその言葉を解するものがいな
固有の部族語をコード（暗号）として
前線での無線通信を行うため、アメリカ
軍が使用したアメリカインディアン部族
出身の暗号通信兵である。”

"Were it not for the Navajos, the Marines would never have taken Iwo Jima."

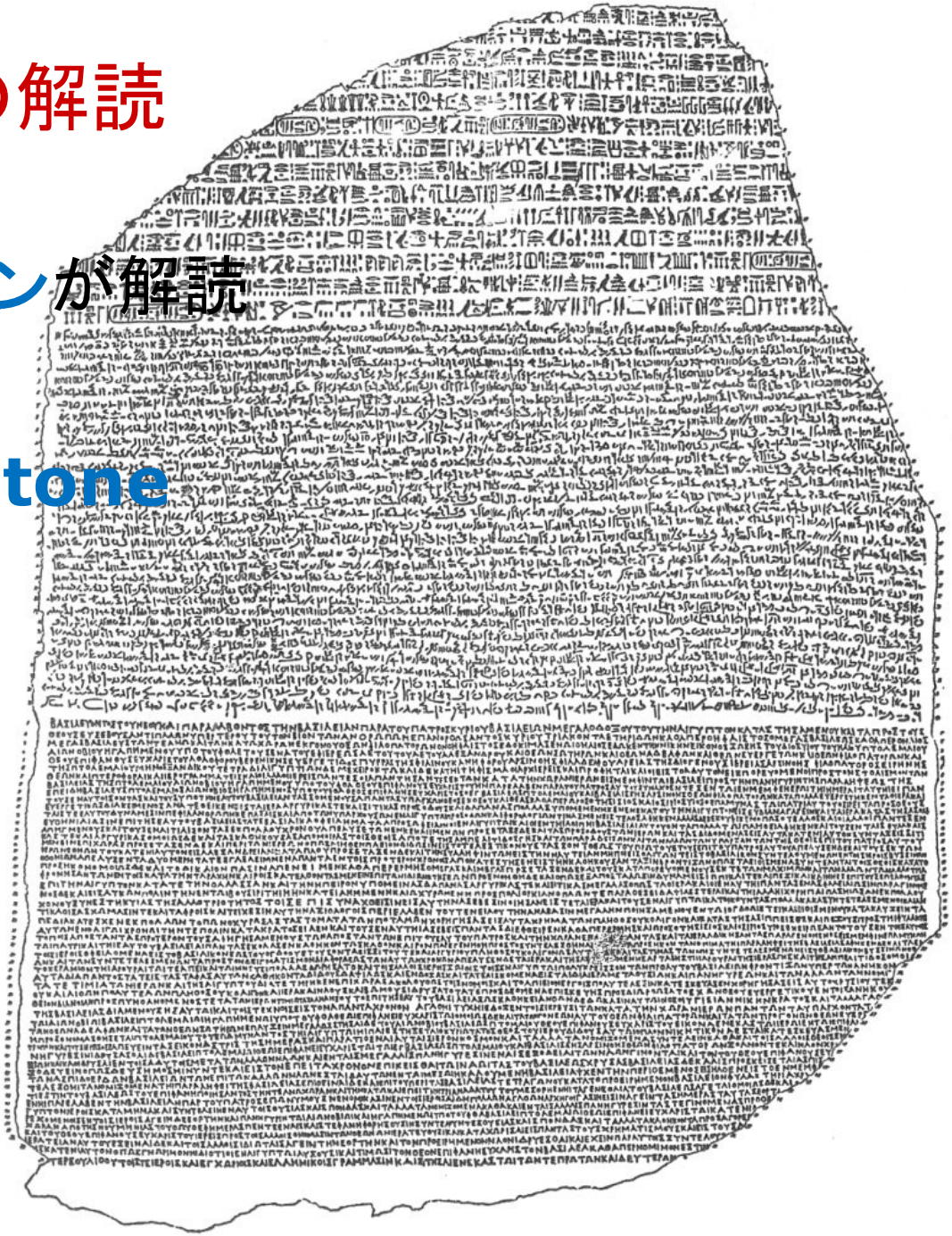
言語と暗号

未知の文字の解読

象形文字の解読

1822年
シャンポリオンが解読

Rosetta Stone
BC 196年



神聖文字

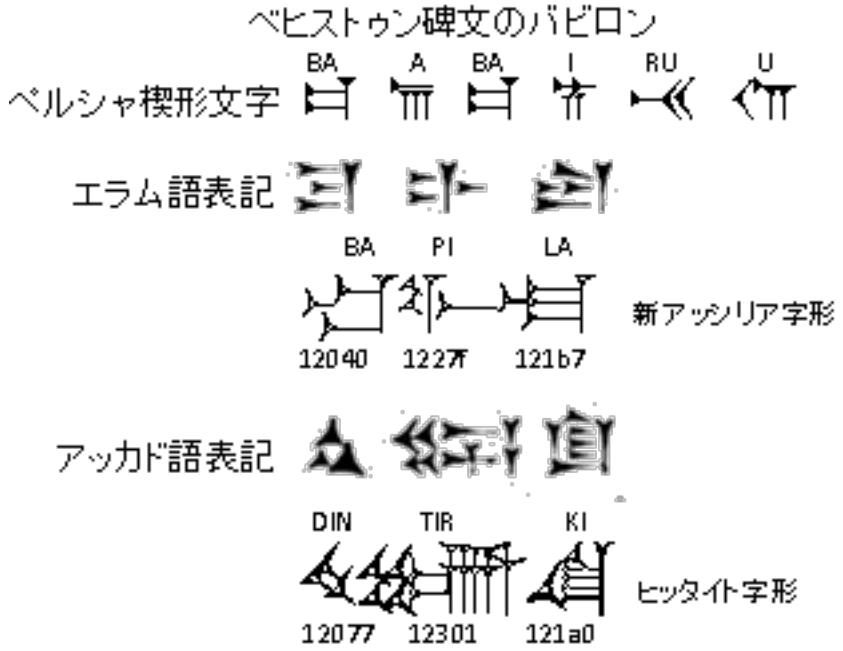
民衆文字

ギリシャ
文字

楔形文字の解読

ベヒストゥン碑文

- コーパス： ベヒストゥン碑文　BC522~
  （エラム語、古代ペルシア語、アッカド語の三つの言語で書かれている）
  解読者：ローリンソンとヒンクス
  1846-1851



線文字Bの解読

- 紀元前1450年から紀元前1375年頃までミュケナイ時代に、ギリシャ本土からエーゲ海諸島の王宮で用いられていた文字。
- 発見者であるイギリスの考古学者アーサー・エヴァンズにより線文字Bと命名された。
- 1953年、イギリスの建築家マイケル・ヴェントリスと言語学者ジョン・チャドウィックによりギリシア語として解読された。

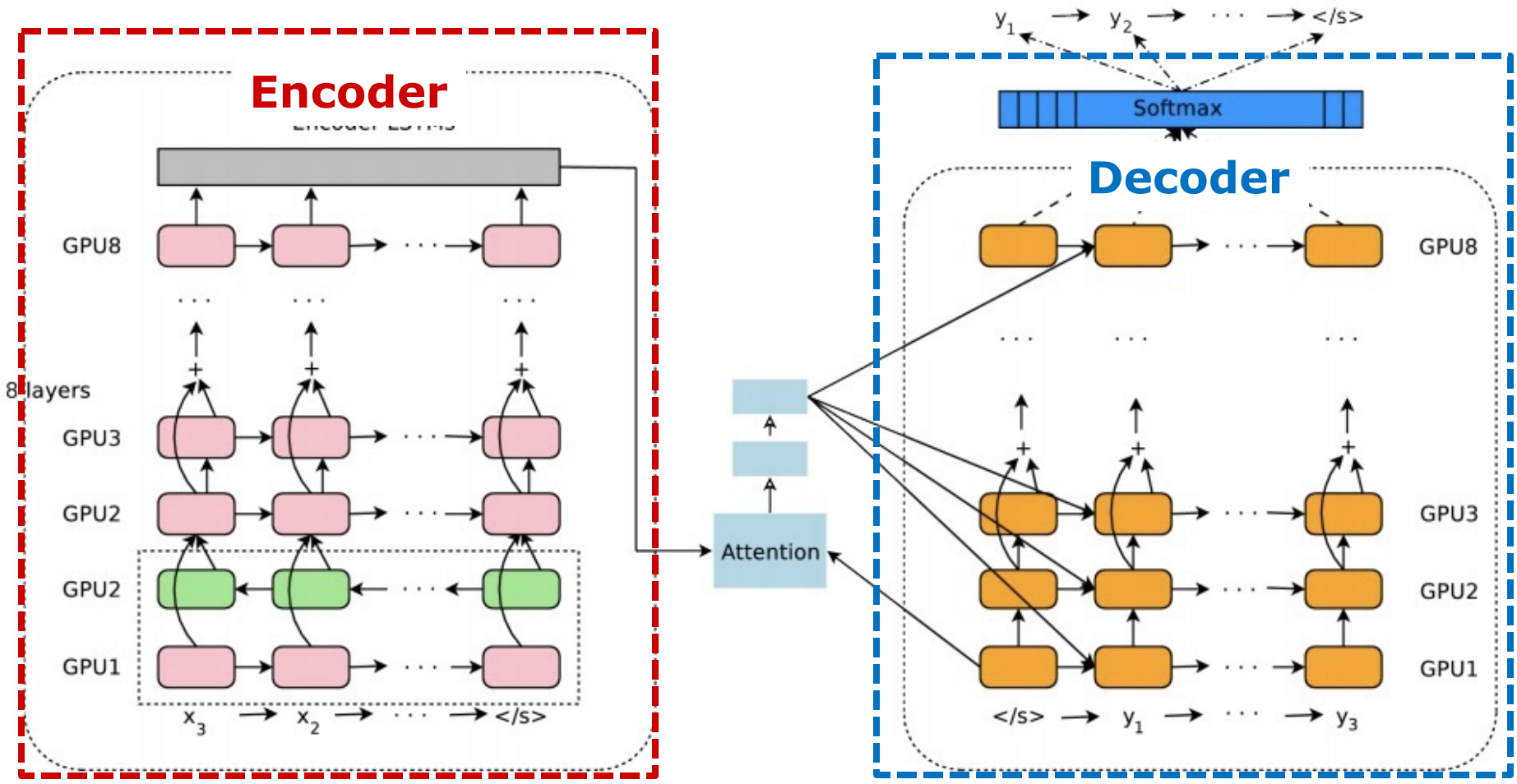
https://goo.gl/4SZhkn

Phaistos diskの解読　2014年

- でも、それとは真逆の取り組みも存在する。クレタのGareth Owensは、たった一つの粘土板にきざまれた、45種類の「文字」で書かれた241文字の「文」の解読に成功したという。
  https://goo.gl/4Ye6Be

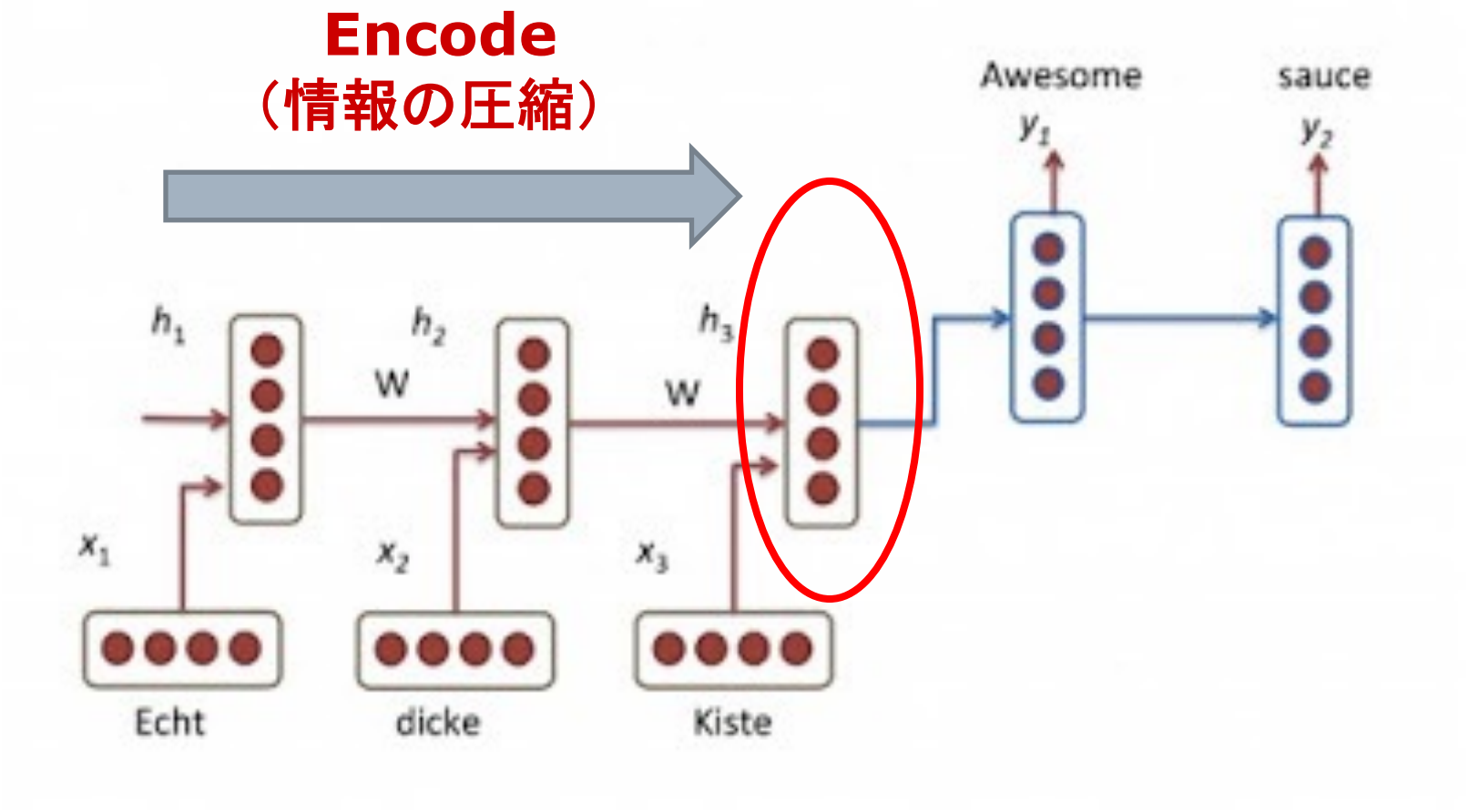
機械翻訳技術と暗号

Encoder / Decoder

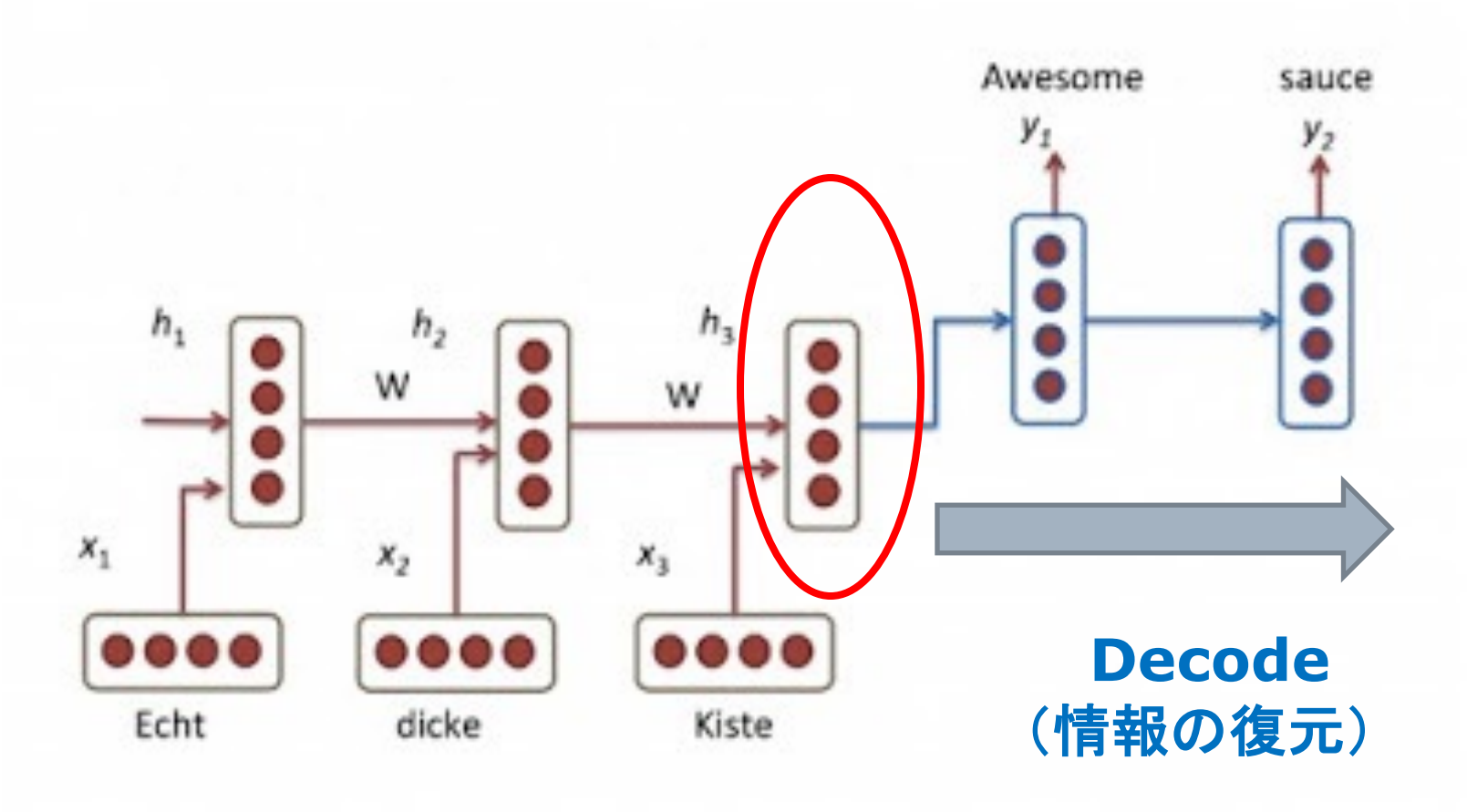


左側に、LSTMを8段重ねにした Encoder LSTMがあり、
右側には、同じくLSTMを8段重ねにした Decoder LSTMがある。

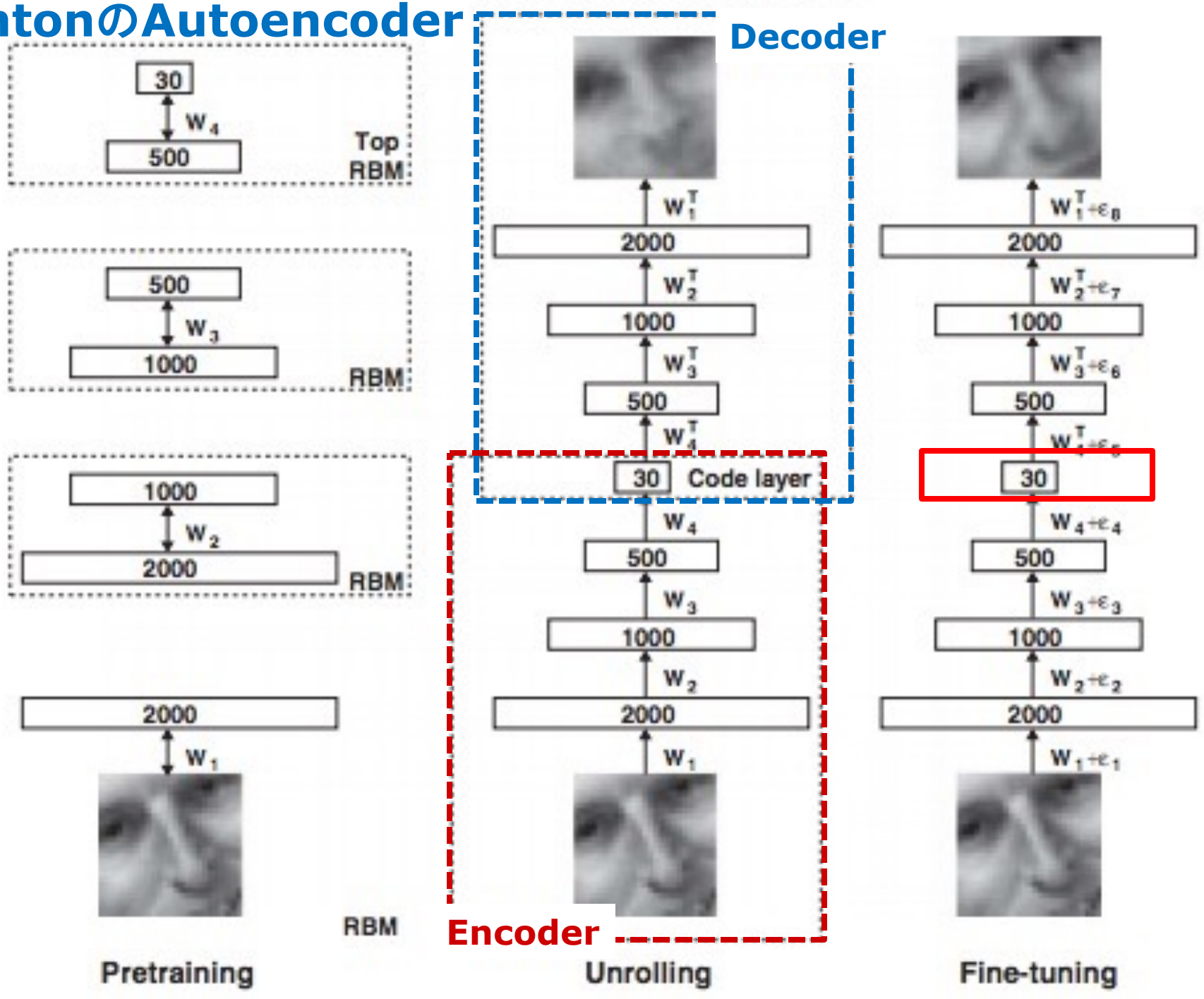
- ここでは、Encoder部が、文章の最後にとるRNNの内部状態h_3が、そのままDecoder部に渡されることが示されている。入力シーケンスの情報のエッセンスが、この内部状態 h_3に凝縮されていると考えればいい。



- AutoencoderのDecoder部が、圧縮された情報から元の情報を復元しようとするように、ここでは、その情報から、**「同じ意味」**を持つ、別の言語の文章を復元しようとする。



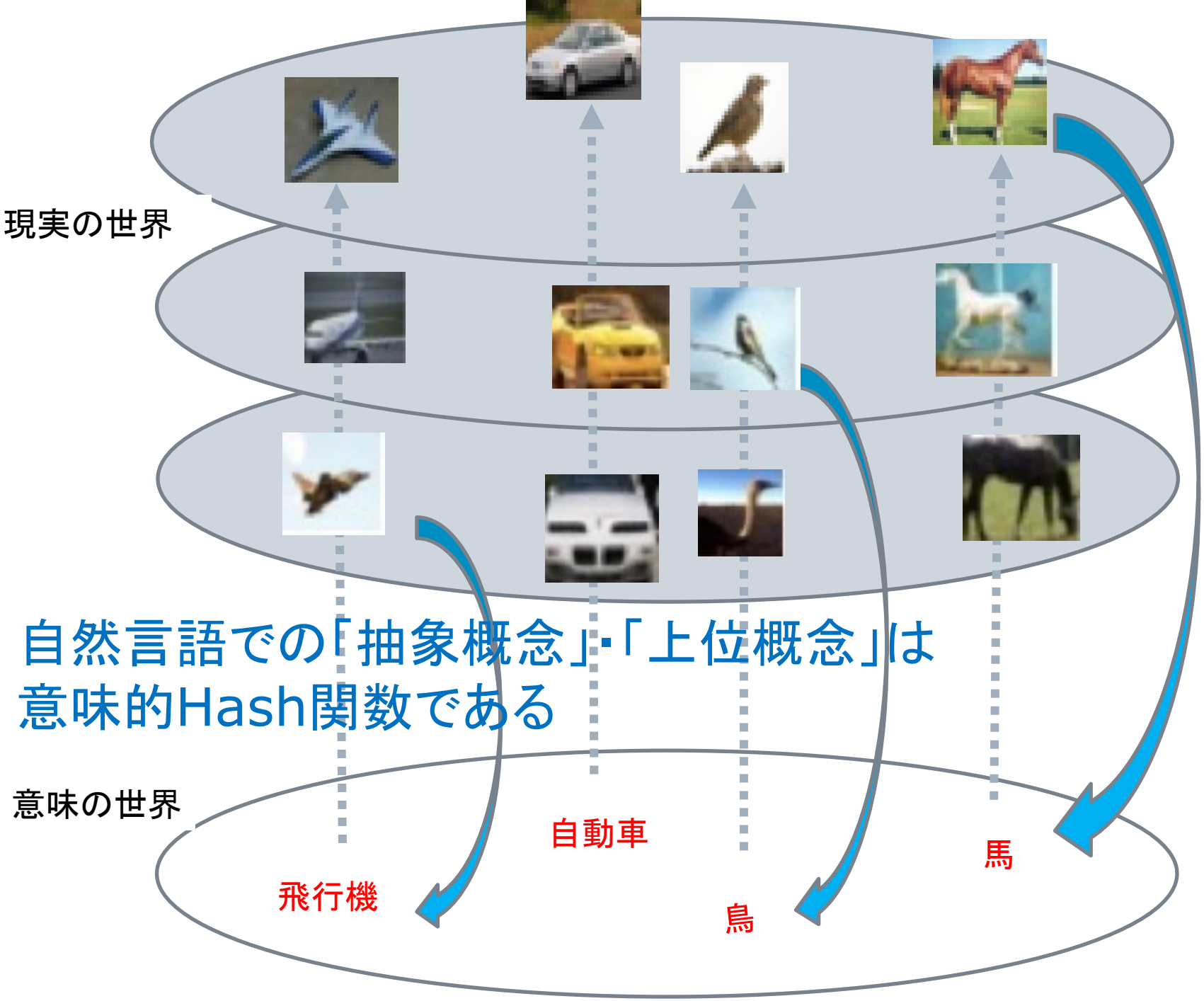
HintonのAutoencoder



Semantic hashing（意味的ハッシング）

- 重要なことは、「画像」と「書籍」では、対象のデータの性質はまるで異なるのだが、Autoencoderは、そのいずれに対しても、高次元のデータを低次元のデータに変換しているということである。別の言葉で言えば、**それは、対象の高次元のデータから、低次元のデータを、元の情報のエッセンスとして取り出しているのである。**

- Hintonは、こうしたAutoencoderの働きを、**Semantic hashing（意味的ハッシング）**と呼んでいる。SHA-1のようなハッシングでは、ハッシュ化されたデータから元のデータを復元することは不可能なのだが、Semantic hashingされたデータは、データの次元は低いものの、元の情報の中核部分を保持している。


現実の世界
自然言語での「抽象概念」・「上位概念」は
意味的Hash関数である
意味の世界
飛行機
自動車
鳥
馬

Deep Learningで暗号解読は可能か？

訓練に利用されたコーパスの規模

- WMT'14の英語（En）<-> フランス語（Fr）データセットには、3,600万の文のペアが含まれている。
- WMT'14の英語(En) <-> ドイツ語（De）データセットには、500万の文のペアが含まれている。
- Googleは、内部に、英語 <-> 日本語（Ja）、英語 <-> 韓国語（Ko）、英語 <-> スペイン語（Es）、英語 <-> ポルトガル語（Pt） 等々の多くのデータセットを持っているが、その規模は、先のWMTのデータセットより、２～3桁大きいという。

- Googleニューラル機械翻訳では、GPU100個を使って、フルトレーニングには最大1,000万ステップ、収束までには3週間かかることがあるという。

現代の暗号技術の成立

公開キー暗号　1976年
Whitfield Diffie, Martin Hellman
RSA暗号　1977年
Ron Rivest, Adi Shamir, Leonard Adleman

計算の複雑さを暗号に利用する
理論的に先行したもの -- NashとGödel

Nash:「NSAへの手紙」 1955年
Gödel:「フォン･ノイマンへの手紙」 1956年

1950年代に入ると、時代の先鋭な知性たちは、今日の複雑性理論の原型とも言える認識に到達し始める。

スコット・アーロンソンは、複雑性理論の到達点を概観した論文 "N = NP?" https://goo.gl/TZUkgh の冒頭に、ジョン・ナッシュの 1955年のNSA（スノーデンがいた、あのSNAだ！）宛の手紙をあげている。かつては「機密文書」とされていて、いつかの時点で情報公開されたものらしい。

誰にも破れない暗号を作るためには、指数関数的な計算が必要な暗号を作ればいいことを述べている。また、そうすれば、それまでの職人技的な暗号破り（多分、チューリングやファインマンがしていたこと）は「過去」のものになると、明確に自覚している！

ジョン・ナッシュ：1955年のNSA宛の手紙

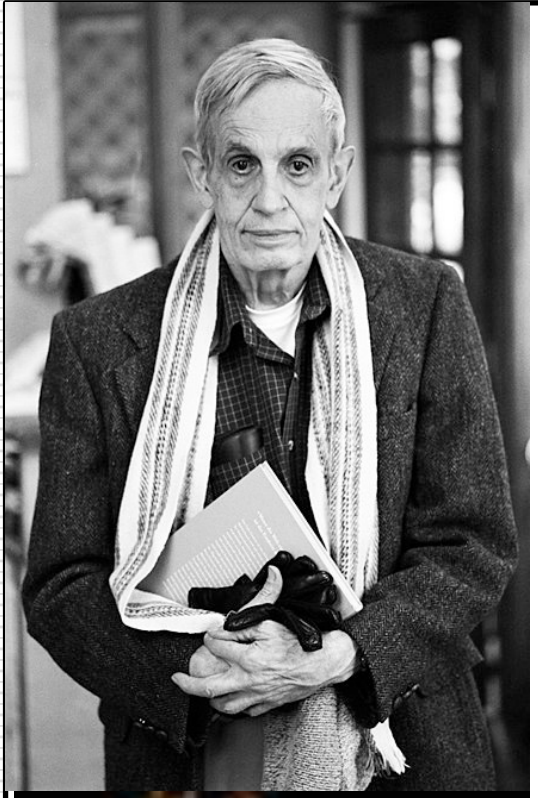
現時点での、私の一般的な予想は次のようなものです。:

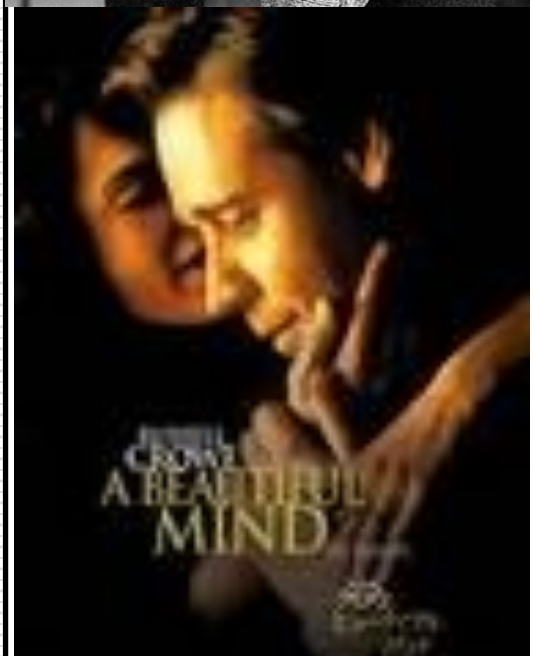
ほとんどすべての十分に複雑なタイプの暗号化にとって、特に、鍵の異なる部分によって与えられた命令が、相互に複雑に相互作用して、それが暗号化の最終的な決定において影響を与えている場合、鍵の計算の平均的な長さは、鍵の長さ、すなわち、鍵の持つ情報の内容に関して、指数関数的に増加します。

もし、この予想が正しいと仮定すれば、この一般的な予想の重要性を理解するのは容易です。それは、実際的には誰も破れない暗号を設計することを、極めて簡単に実行できることを意味します。暗号が洗練されたものになるにつれて、熟練したチームなどによる暗号破りのゲームは、「過去」のものになっていくはずです。

この予想の性質は、たとえ特別な種類の暗号をとっても、私にはそれを証明できないようなものです。また、私は、それが証明されることも期待していません。

https://www.nsa.gov/…/nash-l…/assets/files/nash_letters1.pdf





Now my general conjecture is as follows: For almost all sufficiently complex types of enciphering, especially where the ~~information~~ instructions given by different portions of the key interact complexly with each other in the determination of their ultimate effects on the ~~each~~ enciphering, the mean key computation length increases exponentially with the length of the key, or in other words, with the information content of the key.

The significance of this general conjecture, assuming its truth, is easy to see. It means that it is quite feasible to design ciphers that are effectively unbreakable. ~~and~~ As ciphers become more sophisticated the game of cipher breaking by skilled teams, etc., should become a thing of the past.

The nature of this conjecture is such that I cannot prove it, even for a special type of cipher. Nor do I expect it to be proven. But this

ナッシュの慧眼：計算の複雑性と暗号

- 「鍵の計算の平均的な長さは、鍵の長さ、すなわち、鍵の持つ情報の内容に関して、指数関数的に増加します。」
- 「それは、実際的には誰も破れない暗号を設計することを、極めて簡単に実行できることを意味します。」

暗号理論として：

鍵を破るのに、「指数関数的」時間がかかる暗号は、事実上、誰にも破れない。もちろん、正しい受け取り手が暗号を解読するのに「指数関数的時間」がかかってはいけない。

計算複雑性理論の萌芽として：

「指数関数的時間」とそうではない時間（「多項式時間」という）の区別

Gödel

僕は、ナッシュのNSAへの手紙は知らなかったのだが、有名なのは、1956年にゲーデルが、フォン・ノイマンに宛てた、次の手紙である。ゲーデルが、ほとんど完全に、問題の所在を把握していることがわかる。

我々が、一階の述語論理の全ての式Fに対して、全ての自然数について、長さn（長さ＝記号の数）のFの証明があるかどうかを決定することを可能にするチューリング機械を、簡単に構築することができるのは明らかです。

機械がこれを決定するのに必要とするステップの数をψ(F, n)とし、[特定のFについて] φ(n)＝maxψ(F, n)としましょう。問題は、最適なマシンで、φ(n)がどのくらい急速に大きなものになるかです。

我々は、[nとは独立な定数Kについて] φ(n) ≥ K･n　であることを示すことができます。[φ(n)は、nの一次式より大きいということ。]　実際に、実行時間が Knに比例する（またはKn^2に比例する）マシンが実際にあったとした場合 [「論理式Fに対して長さnのFの証明があるかどうか？」という問題が、nの一次式または二次式に比例する時間で解けるなら]　、これは最大級に重要な結果をもたらすことになります。

つまり、「決定問題」の解決不可能性にもかかわらず、YesかNoかの質問に答えるタイプの問題の場合、数学者の知的努力は完全に機械に置き換えることが可能であることを、明確に示すことができるでしょう。

我々が　実際に行うべきことは、単純に数nを選ぶだけです。そのnが非常に大きくても、もしも、マシンが [「長さnのFの証明があるかどうか？]という質問に] "No"という結果を出した場合には、さらに問題を考える理由はなくなります。

---

https://rjlipton.wordpress.com/the-gdel-letter/

もっとも、これらの50年代半ばの認識は、広く共有されたものではなかったかもしれない。（論文として公開されたものではなく「手紙」だった！）

RSA暗号と公開キー暗号の登場

素因数分解の難しさと暗号への応用

RSA/公開キー暗号の基礎(単純化した例)

ある数Nが、二つの素数p,qの積で表されるとしよう。

例えば、N=99400891の場合、Nは素数 9967と9973の積である。

難しい

N = p x q

99400891 = 9967 x 9973

易しい

この時、Nの素因数を求める計算は難しく、

p, qを掛けてNになることの検証はやさしい。

非常に大きな数Nを取ると、Nを公開したとしても、その素因数p,

例えば、次の220桁の数の素因数を求めるのは難しいRSA Factoring Challenge(2007年に終了)から

RSA-220 =
22601385262034057849416540486101975135080389157197767183211977681094456418179666766085931213065825772506315628866769704480700018111497118630021124879281994874820660701310665866460833279828035603792053919801399464969552 61

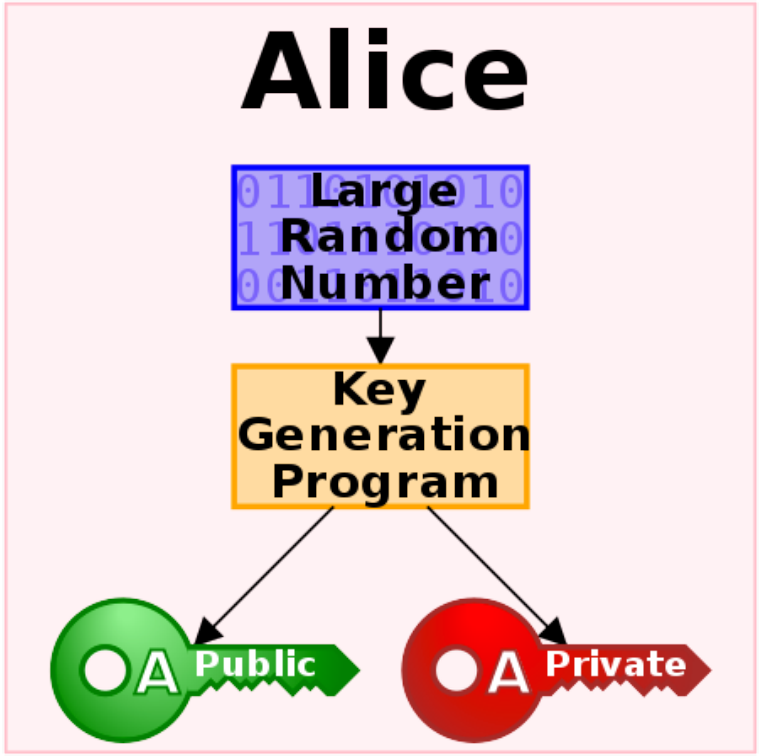

難しい


易しい

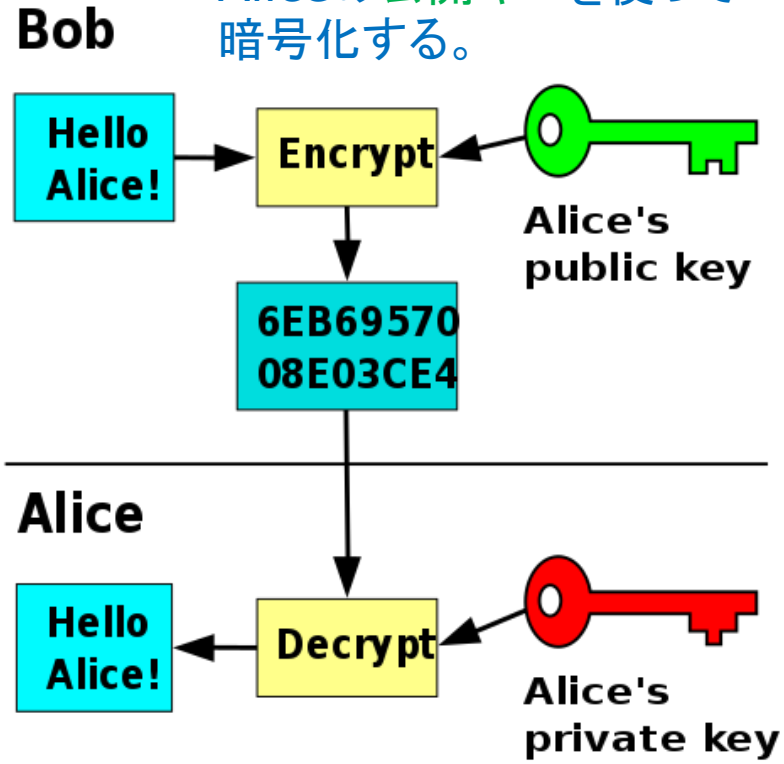
RSA-220 =
68636564122675662743823714992884378001308422399791648446212449933215410614414642667938213644208420192054999687 ×
32929074394863498120493015492129352919164551965362339524626860511692903493094652463337824866390738191765712603

http://bit.ly/2Wl1JoM

公開キー暗号1976年
Whitfield Diffie, Martin Hellman



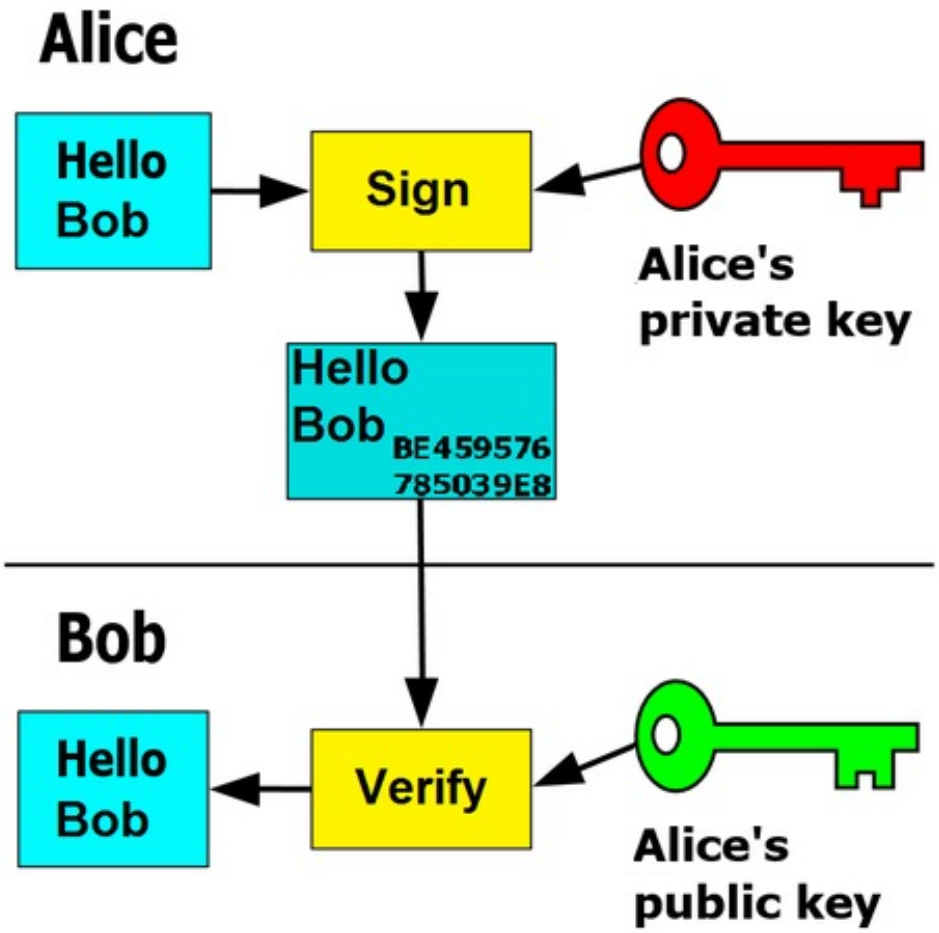
BobはAliceへのメッセージを
Aliceの公開キーを使って
暗号化する。



Aliceは、公開キーと秘密キーの二つの
ペアを生成する。

Aliceは、Bobからのメッセージを
自分の秘密キーを用いて解読する。

電子署名　Digital signatures



Aliceは、自分の送ったメッセージが真正のものであることを示す為に、自分の秘密キーを使って、署名を行う。ハッシュ関数。

Bobは、Aliceからのメッセージが真正であることを確かめる為に、Aliceの公開キーを用いて、署名を検証する

RSA暗号　1977年
Ron Rivest, Adi Shamir, Leonard Adleman

- 大きな桁数の合成数の素因数分解が困難なことを利用。
- まず、適当な正整数 e（通常は小さな数。65537 (= 2^{16} + 1) がよく使われる）を選択する。また、大きな2つの素数 $\{p, q\}$ を生成し、それらの積 n ($=pq$) を求めて、$\{e, n\}$ を平文の暗号化に使用する鍵（公開鍵）とする。
- 2つの素数 {p, q} は、暗号文の復号に使用する鍵（秘密鍵）d の生成にも使用し 秘密に保管する。

$$d = e^{-1} \pmod{(p-1)(q-1)}$$

RSA暗号/公開キー暗号には先行者がいた

機密にされた発見

1970-1974 Ellis-Cocks-Williamson

In 1970, James H. Ellis, a British cryptographer at the UK Government Communications Headquarters(GCHQ), conceived of the possibility of "non-secret encryption", (now called public key cryptography), but could see no way to implement it.[9] In 1973, his colleague Clifford Cocks implemented what has become known as the RSA encryption algorithm, giving a practical method of "non-secret encryption", and in 1974, another GCHQ mathematician and cryptographer, Malcolm J. Williamson, developed what is now known as Diffie–Hellman key exchange. The scheme was also passed to the USA's National Security Agency.[10] With a military focus and low computing power, the power of public key cryptography was unrealised in both organisations:

https://en.wikipedia.org/wiki/Public-key_cryptography

RSA暗号/公開キー暗号には先行者がいた

GCHQ pioneers on birth of public key crypto

Two men who played a role in developing public key cryptography, Clifford Cocks and Ralph Benjamin, talk to ZDNet UK about the invention of the important encryption technique

Public key cryptography is widely used to secure online transactions. The maths behind the technology was invented by UK Government Communications Headquarters scientists in the late 1960s and early 1970s.

The discovery was kept secret to avoid revealing how closely Government Communications Headquarters (GCHQ) was working with the US National Security Agency (NSA) at the time. The breakthrough by GCHQ scientists James Ellis, Clifford Cocks and Matthew Williamson only came to light in 1997, when their work was declassified.

https://zd.net/2wwmaAI

暗号と計算複雑性理論

「計算の難しさ」とは何か？

- 現代の暗号化技術の大きな特徴は、その基礎を、コンピュータはある種の数学的問題を、「効率的」には解くことができないという（経験的）事実においていることである。
- RSA暗号は、素因数分解問題を、現在のコンピュータでは効率的に計算できないことを基礎にし、楕円曲線暗号は、離散対数問題を効率的に解けないことを基礎としている。
- その理論的基礎は、数学的には、「計算複雑性の理論」である。

複雑性理論入門

複雑性理論

「計算の難しさ」を計算に要する「時間」と「メモリー」で評価する

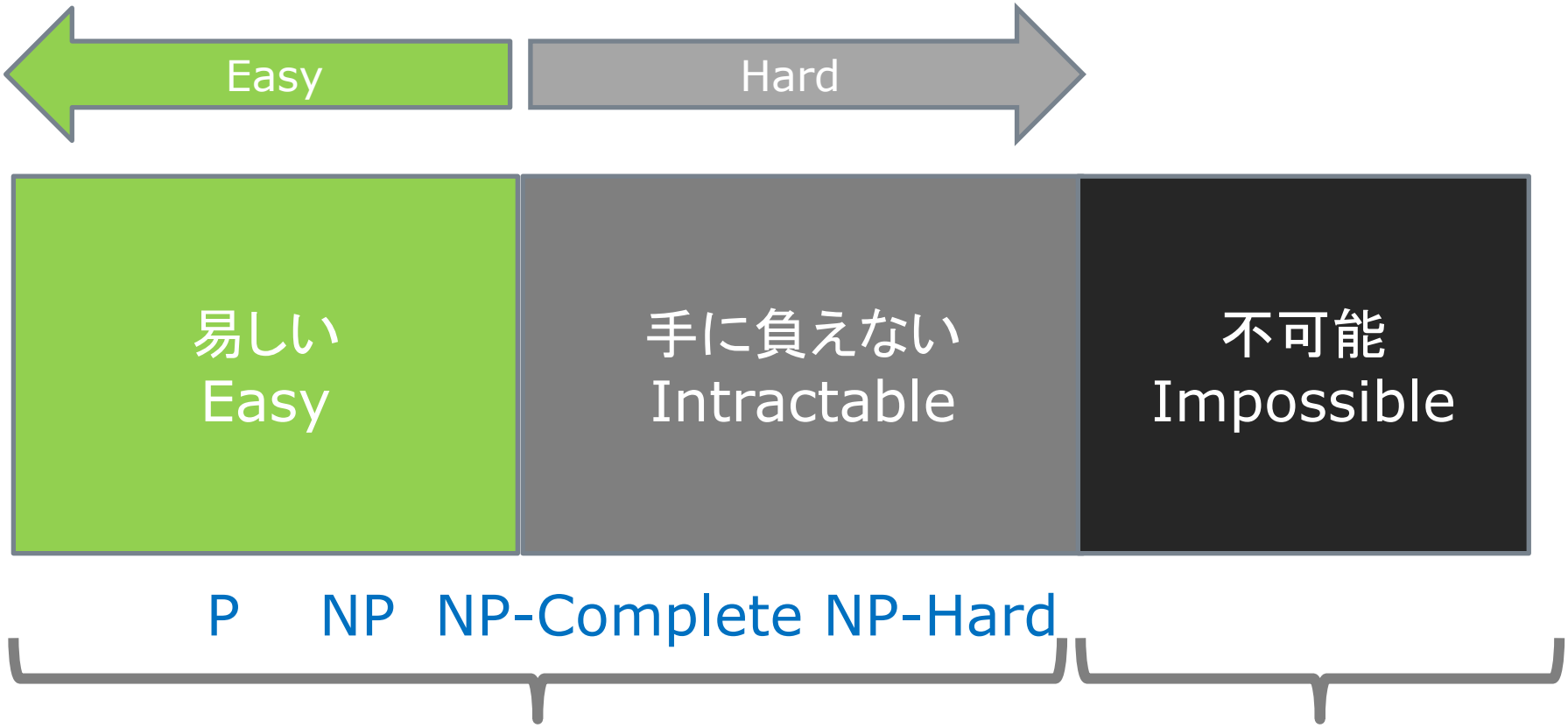
ゲーデル、チューリングらが明らかにしたように、「証明可能＝計算可能」なものには限界があることは、1950年代には、一般に認められるようになった。

ただ、それは、我々の認識の「限界」としては、非常に荒い、かつ、抽象的で原理的な限界を与えるものでしかなかった。

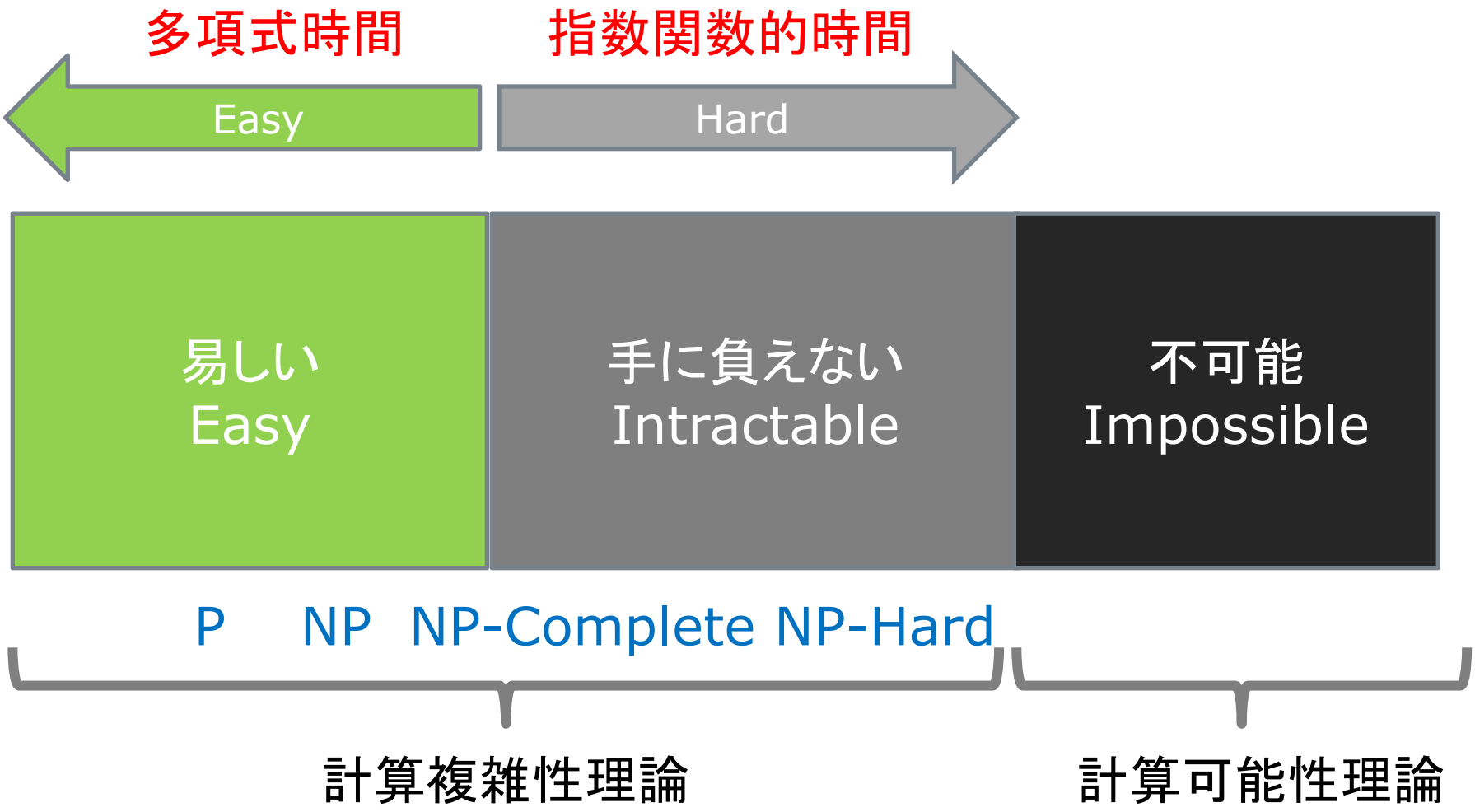
70年代に入ると、「計算の難しさ」に対する新しいアプローチが活発になる。それは、「計算の難しさ」を、計算に要する「時間」と「メモリー」で評価しようというものである。それを「計算複雑性理論」という。

認識の易しさと難しさ

計算の易しさと難しさ


Easy
Hard
易しい
Easy
手に負えない
Intractable
不可能
Impossible
P
NP
NP-Complete
NP-Hard
計算複雑性理論
計算可能性理論

計算の易しさと難しさ


多項式時間
指数関数的時間
Easy
Hard
易しい
Easy
手に負えない
Intractable
不可能
Impossible
P NP NP-Complete NP-Hard
計算複雑性理論
計算可能性理論

「多項式時間」と「指数関数的時間」

計算時間について言えば、このアプローチで重要なのは、「多項式時間」と「指数関数的時間」を区別することだ。

多項式の例：

$$n^3+10n^2+8n+1$$

$$5n^4+9n^3+7n^2+3n+5$$

指数関数の例：

$$3^n+5^n$$

$$2^{2^{2^n}}+3^n+5$$

その上で、計算時間が「多項式時間」で表される計算を「やさしい計算」、指数関数で表される計算を「難しい計算」と考えるのだ。

P問題

入力の文字列の長さをnとするある問題が、nの「多項式時間」で解ける時、この問題を「P問題」という。

「P問題」は、「やさしい計算」問題と考えていい。

計算複雑性の理論で、「P問題」と並んで重要なクラスに「NP問題」がある。「NP問題」は、解くのに「指数関数的時間」がかかる「Not-P問題」ではない。それについては、次に説明する。それでも、

「NP問題」は、「難しい計算問題」と考えていい。

NP問題

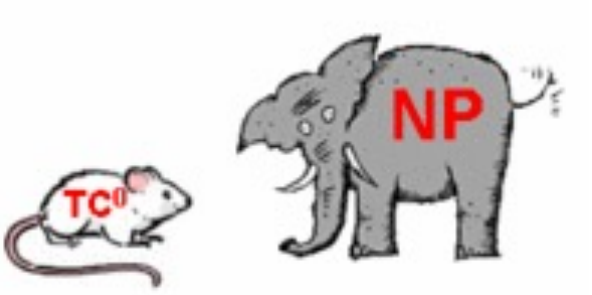
その問題を解くのが「多項式時間」で終わるかどうかはわからなくても、具体的な値で与えられたその問題の「答え」と言われるものが、本当に「正しい答え」であるかどうかを「多項式時間」でチェックできる問題のクラスを、NPという。

NPクラスの説明によく用いられるのは、素因数への分解問題である。

先に見たように、Nは二つの素数の積であることがわかっていたとしても、Nが大きな数になると、Nを素因数に分解すること、すなわち、N＝pqであるような二つの素数 p, q を見つけることは難しくなる。

ただ、N, p, q が具体的に与えられているなら、「Nは、素数pと素数qに、素因数分解される」という主張は、簡単に検証できる。pとqを掛けて、その結果がNに等しいことを示せばいいのだから。掛け算は、もちろん簡単な計算で、多項式時間で終わる。

P=NP?問題



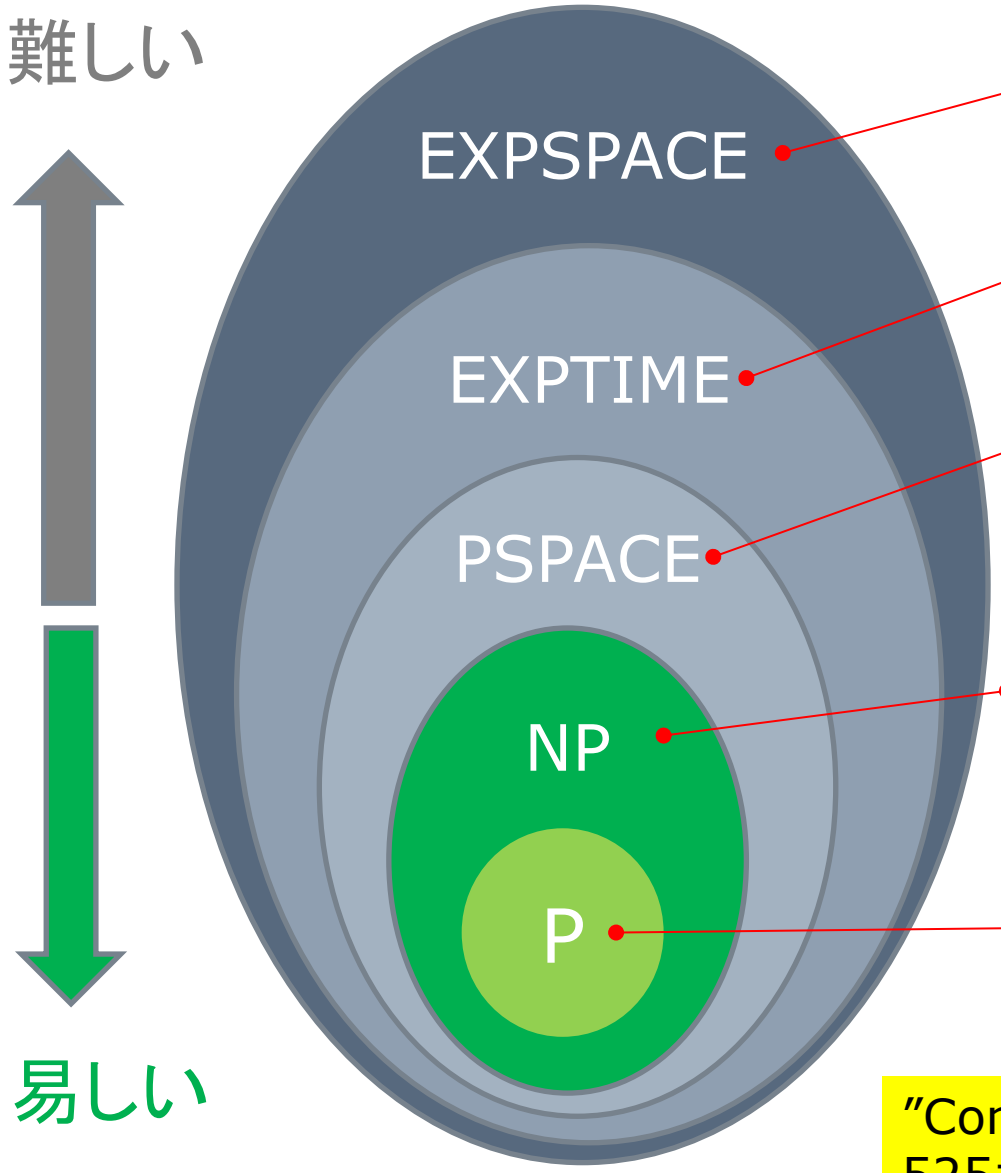
「計算複雑性理論」の中心問題は、このPとNPの関係である。
もしも、PとNPが実際には等しいクラスであることが証明できれば、見かけは「むずかしい計算問題」に見えるNP問題が、実は「やさしい計算」で解けることになる。

多くの数学者は、PとNPは、等しくないと考えている。
ただ、この「P=NP?問題」は、現在に至るも解決されていない。

この問題は、現代数学上の未解決問題の中でも最も重要な問題の一つであり、2000年にクレイ数学研究所のミレニアム懸賞問題の一つとして、この問題に対して100万ドルの懸賞金がかけられている。

現在のようなスタイルで、N = NP? 問題を最初に定式化したのは、バークレーのStephen Cookである。1967年のことだ。

代表的な複雑性クラスの階層



指数関数的なメモリーを使用

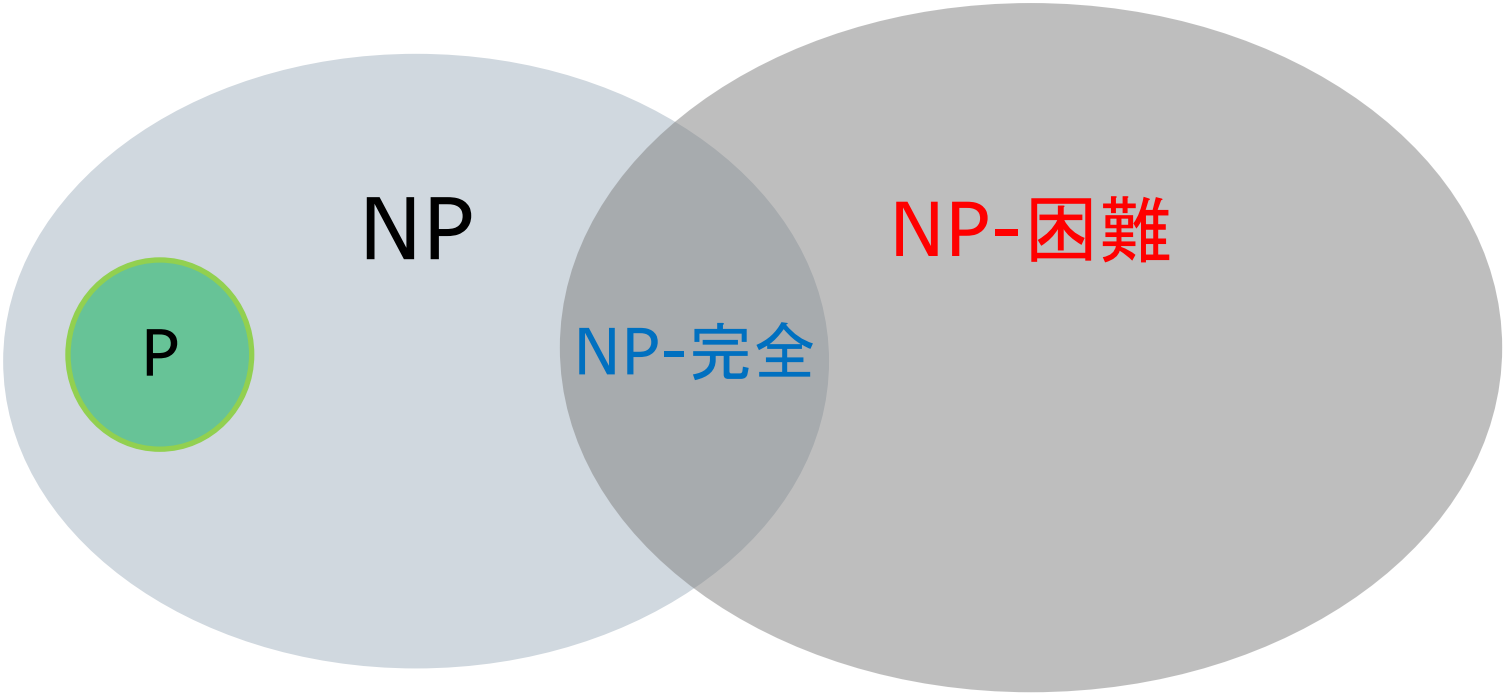
指数関数的な計算時間

多項式関数的なメモリーを使用

答えが正しいかは多項式時間でチェックできる問題

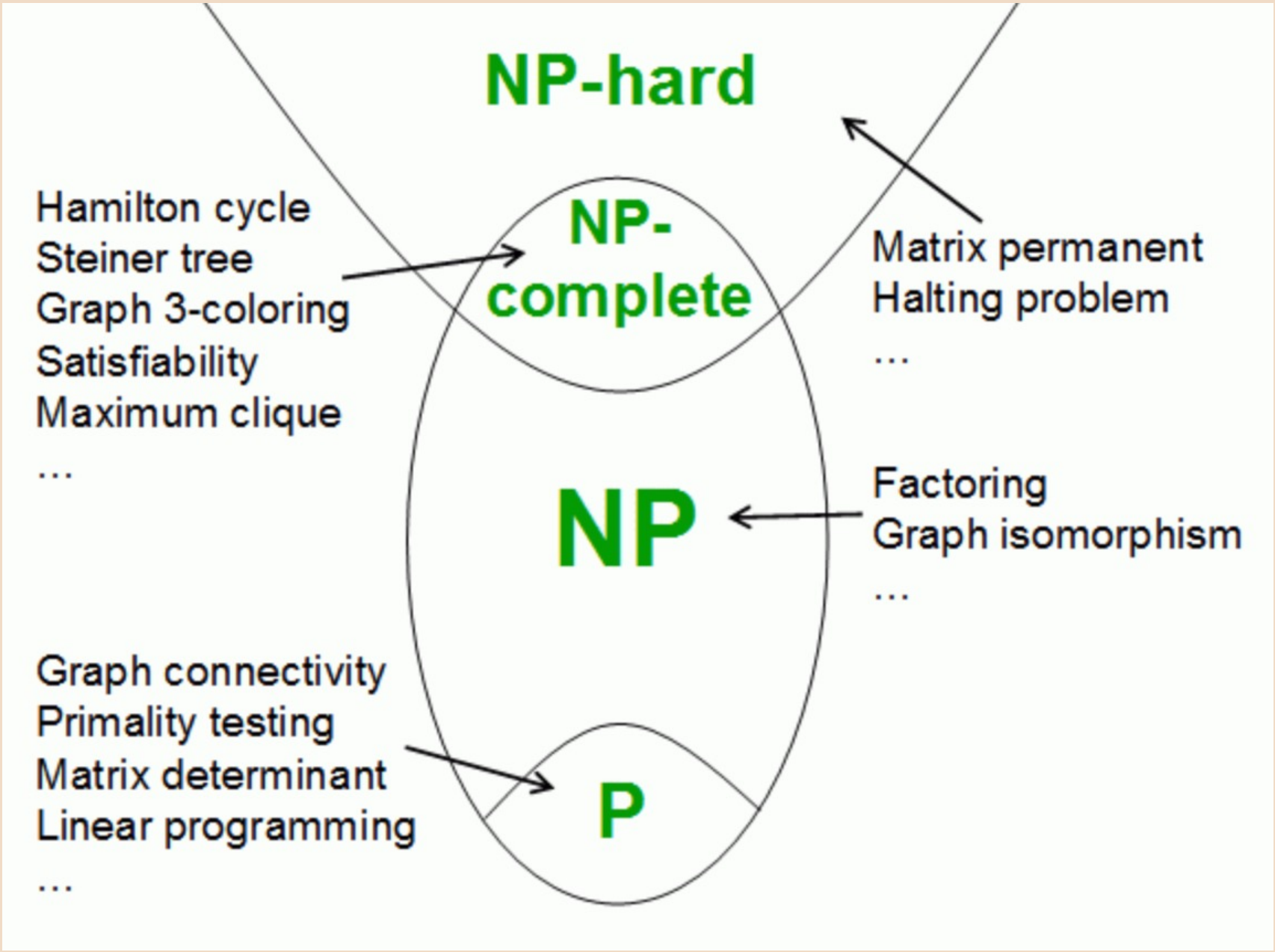
多項式時間で計算できる問題

”Complexity Zoo” http://bit.ly/30FlQxo
525もの複雑性のクラスが収録されている！



P:　多項式時間で解ける問題
NP: 多項式時間で解けるとは限らないが、
　　答えが正しいかは多項式時間でチェックできる問題

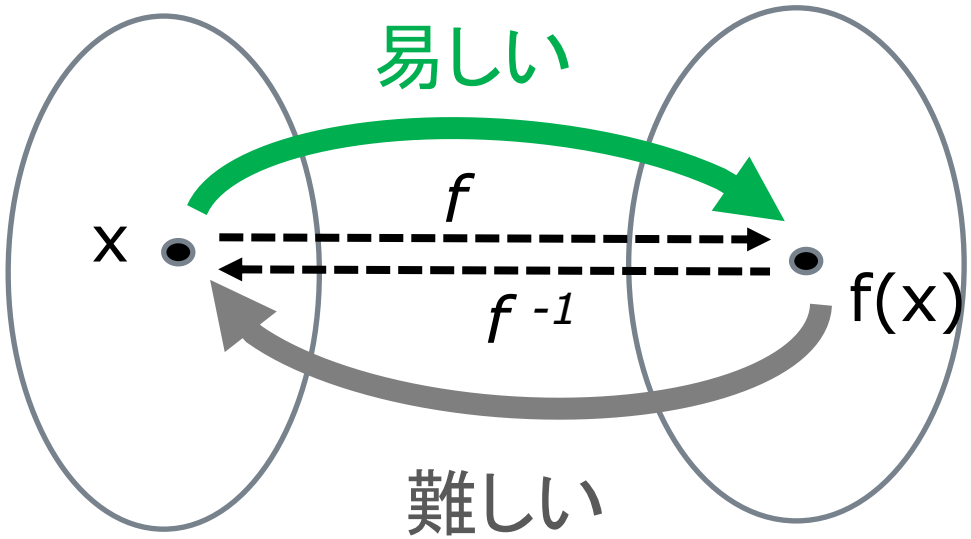
NP-困難: 全てのNP問題が、このクラスの問題に帰着される問題
　　（この帰着は多項式時間で行われなければいけない）
NP-完全: NP問題で、かつ、NP-困難な問題


NP-hard
NP-complete
NP
P
Hamilton cycle
Steiner tree
Graph 3-coloring
Satisfiability
Maximum clique
…
Matrix permanent
Halting problem
…
Factoring
Graph isomorphism
…
Graph connectivity
Primality testing
Matrix determinant
Linear programming
…

一方向関数

一方向関数　One Way Function

関数 y=f(x)は、xからf(x)を計算するのは「易しい」が、逆の計算 yから y=f(x)を満たすxを求めるのが「難しい」時、「一方向関数」という。



p,q が与えられた時
N=pqを計算するのは
易しい

Nが与えられた時
N=pqなる素因数
p,qを見つけるのは
難しい

一方向関数　One Way Function の定義

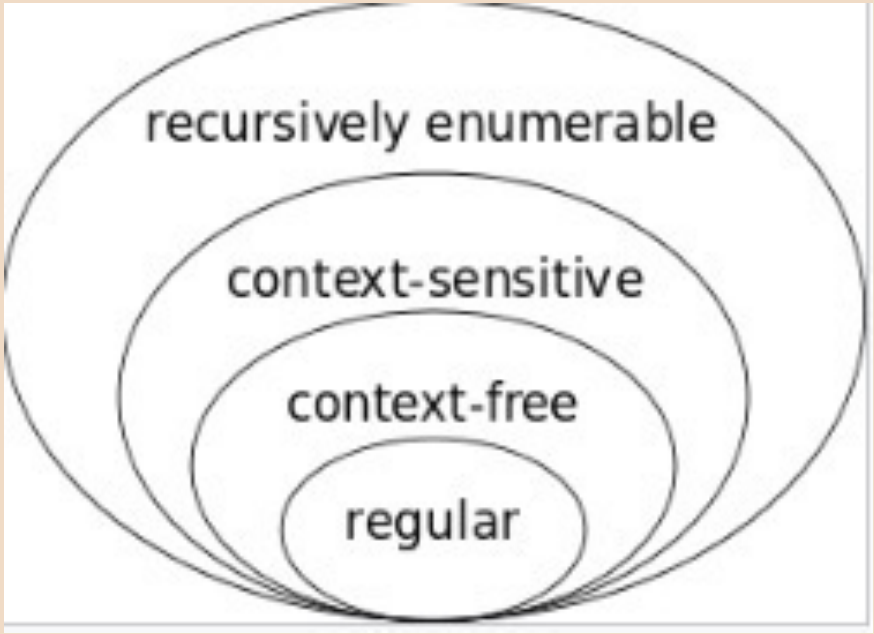
- 一方向関数 f の計算の「易しさ」の定義は、易しい。
  f の計算が、「多項式時間で可能」であるとすれば良い。
- ただ、f^{-1}の計算の「難しさ」の定義は、難しい。
  - f^{-1}の計算に「指数関数的時間がかかる」ということが言えればいいように思うのだが、それを証明するのは簡単ではない。
  - 「NP-完全」のクラスを利用すればいいように思うかもしれないが、暗号として使うには、「最悪の場合」に計算が難しいだけでなく、「平均して」、その計算が難しいものでなければならない。
  - こうした要請に応える一方向関数の「難しさ」の定義があるのだが、それについてはのちに簡単に触れる。
- 実は、「一方向関数が存在する」という命題は、P≠NPを含意する。だから、「一方向関数が存在する」ことは、数学的には、まだ、証明されていないのだ。ただ、ほとんどの数学者は、P≠NPだと信じているので、一方向関数が存在すると信じている。

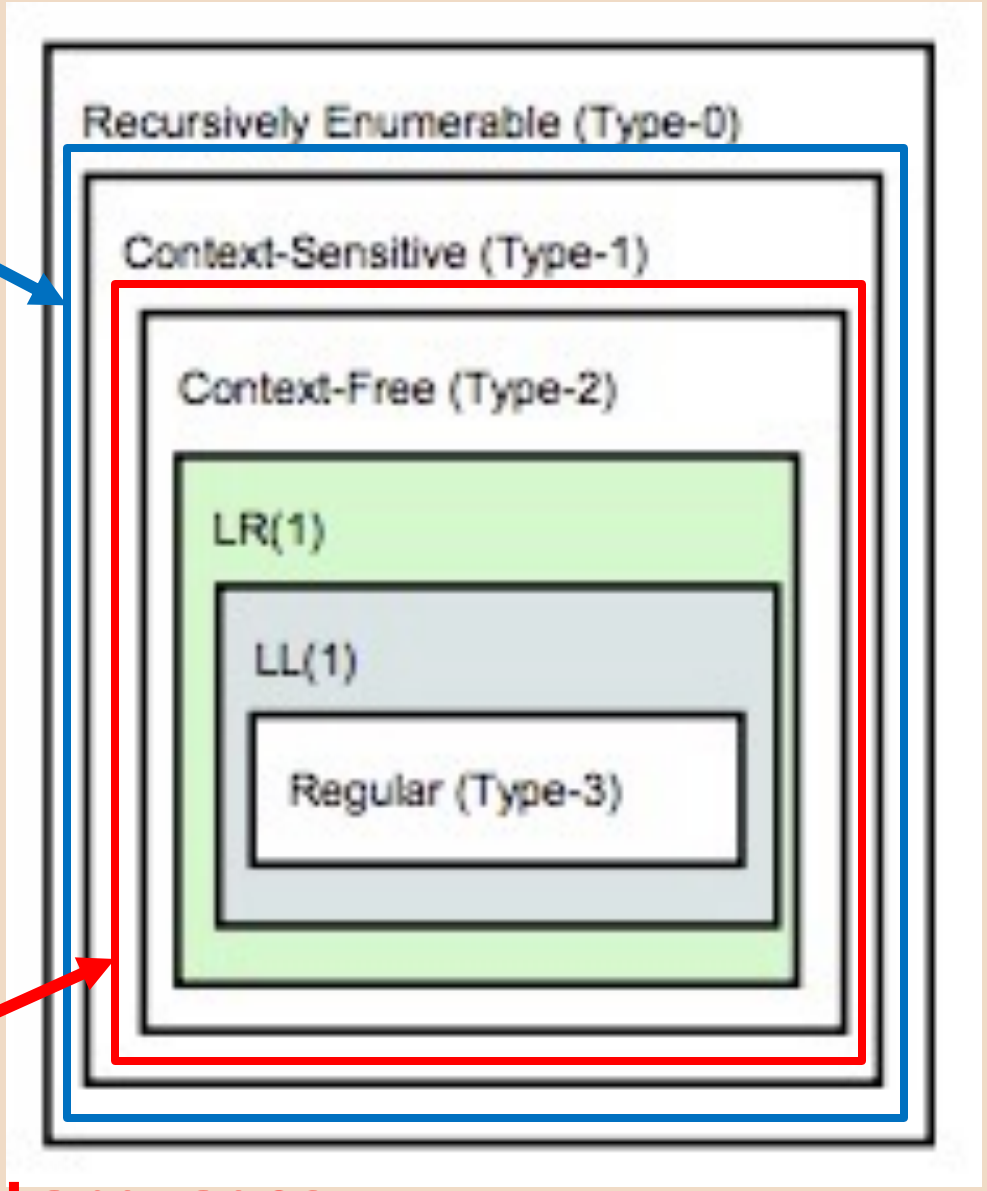
一方向関数　One Way Function の候補

- **素因数分解：**
  N=pqなる素数p,qを求めることのの難しさを利用する。
  これが、RSA暗号の基礎である。
- **離散対数：**
  実数e,x,y が $y=e^x$ を満たす時、
  $x=\log_e y$と書き、xをyの（底eについての）対数という。
  整数e,x,y,p が $y \equiv e^x \pmod p$を満たす時、
  $x \equiv \log_e y \pmod p$と書き、xをyの離散対数という。
  例えば、e=3,x=4,p=17とすれば、$3^4=81$だから、それを17で割った余りは13となるので、$3^4 \equiv 13 \pmod{17}$、よって
  $4 \equiv \log_3 13 \pmod{17}$となる。
  一般に、e,y,pが与えられた時、その離散対数 $\log_e y \pmod p$を求めるのは難しい。これが、楕円曲線暗号の基礎である。

言語と複雑性

Recursive Language
“Merge” is recursive





自然言語？

Mildly Context Sensitive Languages

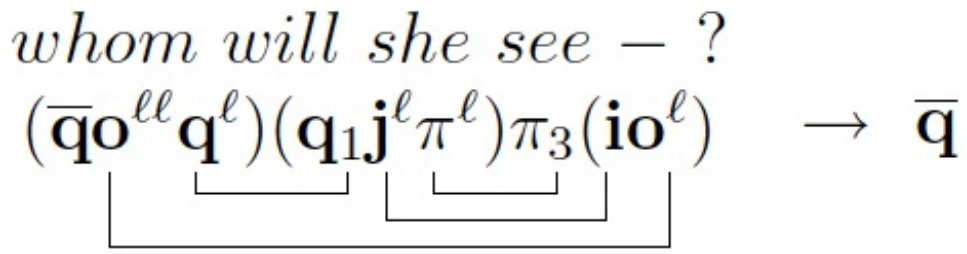
Chomsky Hierarchyに関係する多くの問題は、PSPACE-complete problemsである

- Word problem for context-sensitive language
  Intersection emptiness for an unbounded number of regular languages
- Regular expression star freeness
- Equivalence problem for regular expressions
- Emptiness problem for regular expressions with intersection.
- Equivalence problem for star-free regular expressions with squaring.

https://goo.gl/xZJYF5 https://goo.gl/hXveKm

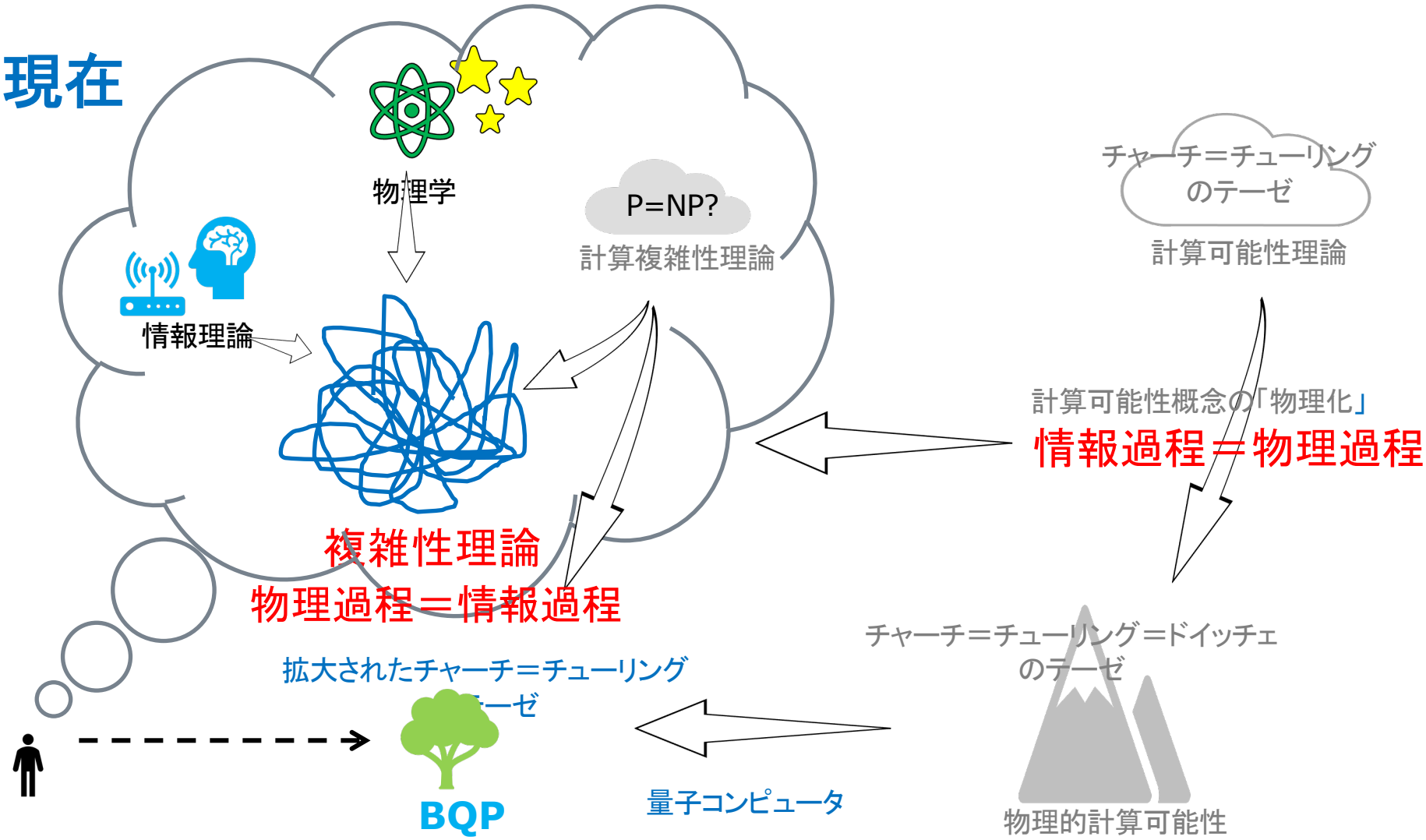
how a sentence such as (3.1) is analyzed in real time?

(3.1)



$$
\begin{aligned}
\textit{whom}: &\quad \overline{\mathbf{q}}\hat{\mathbf{o}}^{\ell\ell}\mathbf{q}^{\ell},\\
\textit{whom will}: &\quad \overline{\mathbf{q}}\hat{\mathbf{o}}^{\ell\ell}\,\underline{\mathbf{q}^{\ell}\mathbf{q}_1}\,\mathbf{j}^{\ell}\pi^{\ell} \rightarrow \overline{\mathbf{q}}\hat{\mathbf{o}}^{\ell\ell}\mathbf{j}^{\ell}\pi^{\ell},\\
\textit{whom will she}: &\quad \overline{\mathbf{q}}\hat{\mathbf{o}}^{\ell\ell}\mathbf{j}^{\ell}\pi^{\ell}\pi_3 \rightarrow \overline{\mathbf{q}}\hat{\mathbf{o}}^{\ell\ell}\mathbf{j}^{\ell},\\
\textit{whom will she see}: &\quad \overline{\mathbf{q}}\hat{\mathbf{o}}^{\ell\ell}\,\underline{\mathbf{j}^{\ell}\,\overline{\mathbf{i}}\,\mathbf{o}^{\ell}} \rightarrow \overline{\mathbf{q}}\,\underline{\hat{\mathbf{o}}^{\ell\ell}\mathbf{o}^{\ell}} \rightarrow \overline{\mathbf{q}}.
\end{aligned}
$$

現在

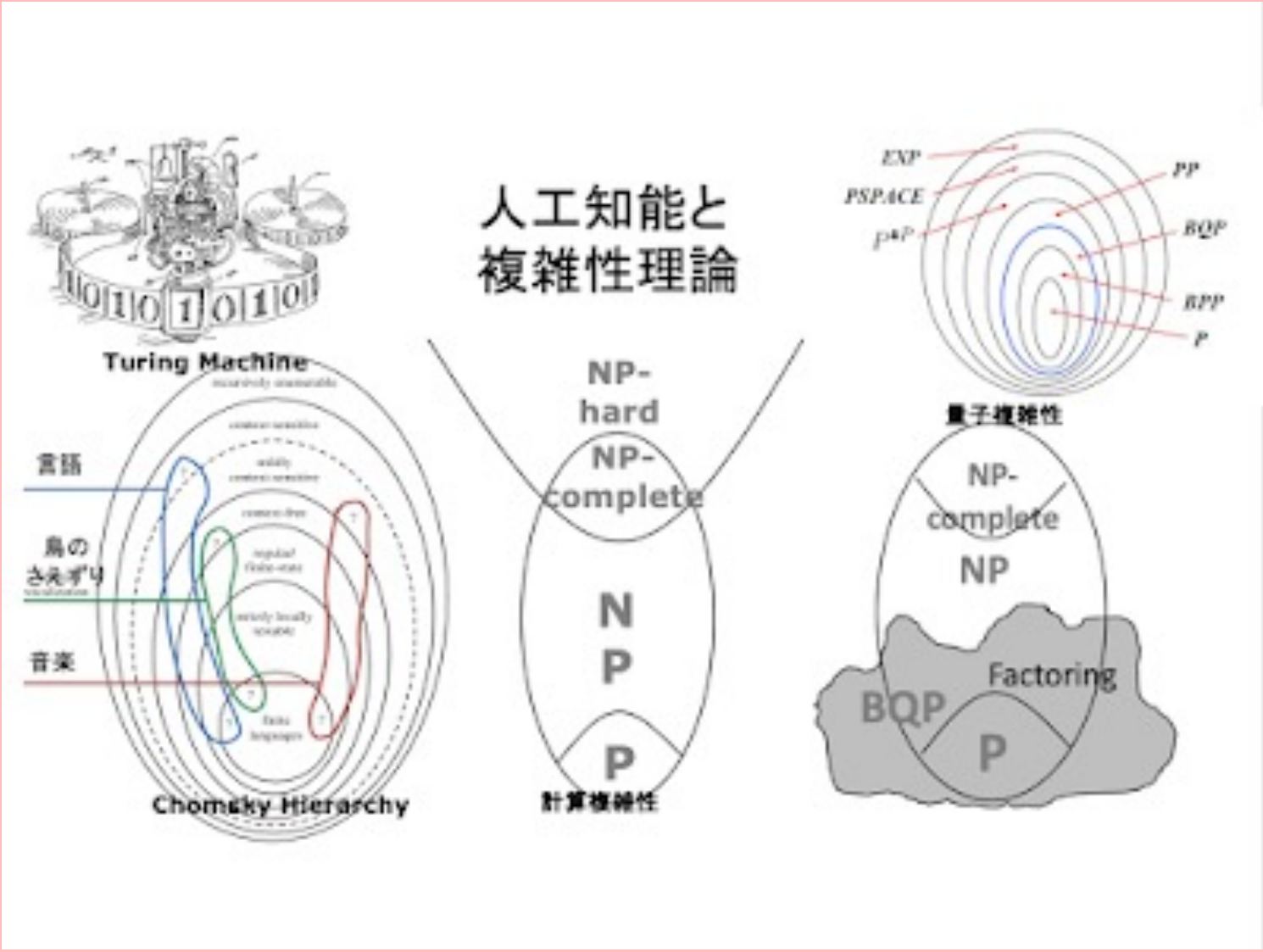


計算の「限界」は、我々の身近の具体的な問題に近づいてきている。物理学の複雑性理論の発展もめざましい。

我々の認識の「限界」の認識が進む一方で、我々の世界の認識が拡大する

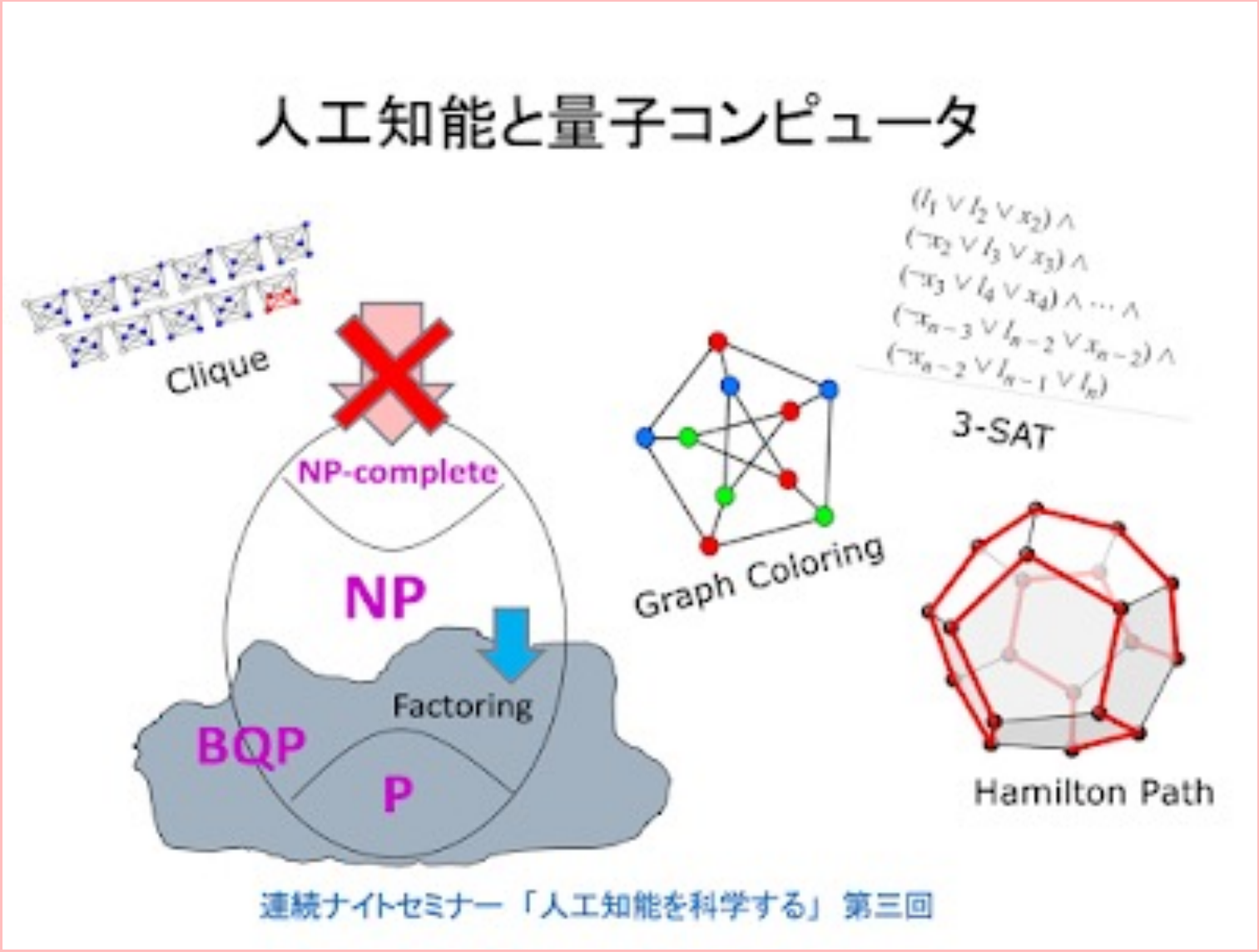
- 複雑性理論は、かつての数学的・抽象的な認識の限界の理論（「チャーチ＝チューリングのテーゼ」）から、物理的・具体的ではあるが、観念の中にしかないものを対象にした限界の理論（「チャーチ＝チューリング＝ドイッチェのテーゼ」）へと発展してきた。かつ、それは、量子コンピュータが現実味を増すにつれて、より具体的な量子複雑性のもとでの限界の理論（「拡大されたチャーチ＝チューリングのテーゼ」）へと変化してきた。
- それは、我々の認識の「限界」の認識が、より一層、精緻で現実的なものに進んだということである。
「我々が知りうることは、限られている。」
- その一方で、そうした限界の認識の深化は、我々の世界の認識が拡大する過程と一体なのだ！
「我々が知りうることは、拡大している。」

参考資料(MaruLabo-角川 2018/09/27)



https://www.marulabo.net/docs/20180927-kadokawa/

参考資料(MaruLabo-角川 2018/11/19)



https://www.marulabo.net/docs/20181119-kadokawa/

参考資料(MaruLabo-角川 2018/10/26)

https://www.marulabo.net/seminartype/kadokawa/

量子コンピュータと暗号

Shorのアルゴリズムの発見

Shorの素因数分解アルゴリズムの発見

- 1994年、Peter Shorは、量子コンピュータを利用すれば、RSA暗号の基礎である素因数分解が、多項式時間で実行できることを、理論的に証明した。
- このアルゴリズムを用いれば、RSAでの素因数分解問題だけでなく、楕円曲線暗号の基礎である離散対数問題も多項式時間で解くことが示され、暗号技術の世界に大きな衝撃を与えた。
- ただ、Shorのアルゴリズムを実行する量子コンピュータの実現が、極めて困難であることがわかり、それが現在の暗号技術への「当面の脅威」ではないことが、むしろ広く認識として共有された。
- 確かに、現在に至るも、Shorのアルゴリズムを用いて、現在利用されている暗号化を破るような量子コンピュータは、いまだ存在しない。
- ただ、こうした認識に大きな変化が現れるのは、2015年になっ

Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer

Peter W. Shor

https://arxiv.org/pdf/quant-ph/9508027.pdf

1996年

ショアのアルゴリズム
– 古典コンピュータ編 概論

いくつかの予備準備
フェルマーの小定理

pが素数の時、pと互いに素な整数aについて、次の式が成り立つ。

$a^p \equiv a \pmod{p}$　　この式は、両辺をaで割って

$a^{p-1} \equiv 1 \pmod{p}$　と同じである。

$2^3 = 8 = 3 \times 2 + 2 \equiv 2 \pmod{3}$

$2^5 = 32 = 5 \times 6 + 2 \equiv 2 \pmod{5}$

$2^7 = 128 = 7 \times 18 + 2 \equiv 2 \pmod{7}$

$2^{11} = 2048 = 11 \times 186 + 2 \equiv 2 \pmod{11}$

対偶をとれば、

aとnが互いに素で、$a^{n-1} \not\equiv 1 \pmod{n}$ ならば、nは素数ではない

ことがわかる。

フェルマー・テスト　確率的素数判定

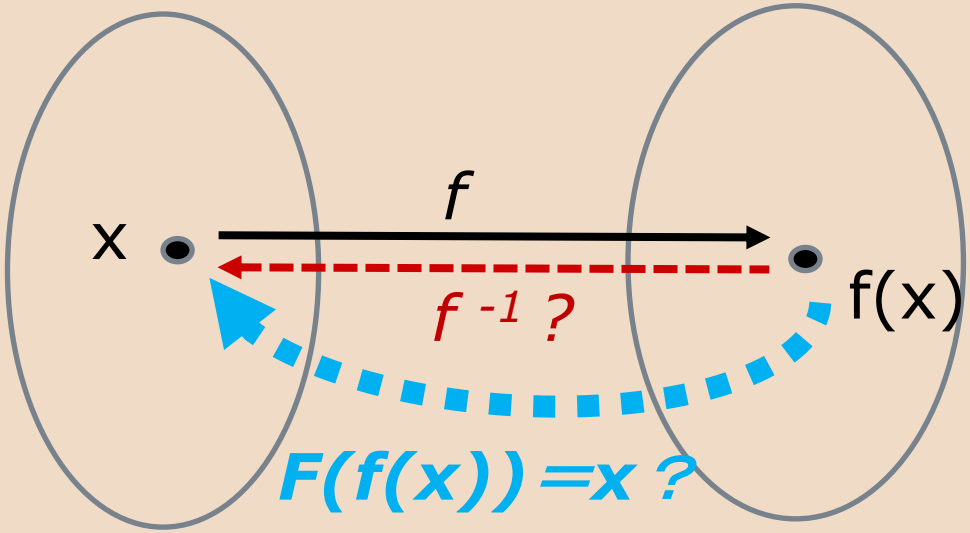
1. n以下の自然数aを一つ選ぶ
2. aとnが互いに素でないなら、nは素数でない
3. $a^{n-1} \not\equiv 1 \pmod{n}$ ならば、nは素数でない
4. そうでないなら、nは素数である確率がある

この時、別のaを選んで、この処理を繰り返す。

十分な回数だけ a を取り替えて繰りかえしても、nがこのテストをパスするなら、その数は実際に素数である可能性が高い。

shorのアルゴリズムも、同じような確率的方法をとる。

一方向関数の「難しさ」の確率的定義（簡略版）



f(x)は既知として、f(x)の値からxの値を推定することを目指す。

あるアルゴリズムがあってそれによるxの推定値をF(f(x))とする。

このxの推定が正しいかは、この値をfに代入して、それがf(x)と等しいかをチェックすれば良い。

f(F(f(x))=f(x) ？

f(x)の値からxの値をランダムに推定するすべてのアルゴリズムFに対して、f(F(f(x))=f(x)が、一定の確率以下でしか成り立たないなら、fは「難しい」と判断できる。

もしFがf^{-1}と完全に一致していれば、100%の確率で、次の式が成り立つ

f(F(f(x))=f(x)

関数a^xの周期（あるいは、群の位数）

$a^r \equiv 1 \pmod N$ なるrが存在する時、この最小のrを（Nについての）aの周期（あるいは位数）という。

例えば、$N = 15, a = 7$ とすると、周期は4である。

$7^0 = 1 \pmod{15}$
$7^1 = 7 \pmod{15}$
$7^2 = 4 \pmod{15}$
$7^3 = 13 \pmod{15}$
$7^4 = 1 \pmod{15}$
$7^5 = 7 \pmod{15}$
$7^6 = 4 \pmod{15}$
$7^7 = 13 \pmod{15}$
$7^8 = 1 \pmod{15}$
$7^9 = 7 \pmod{15}$

周期（位数）を使った素因数分解

xの周期をrとする。rが偶数の時、次のようにかける。

$$
\begin{aligned}
x^r &\equiv 1 \mod N &\iff \\
(x^{r/2})^2 &\equiv 1 \mod N &\iff \\
(x^{r/2}+1)(x^{r/2}-1) &\equiv 0 \mod N &\iff \\
(x^{r/2}+1)(x^{r/2}-1) &= kN \text{ for some } k.
\end{aligned}
$$

すなわち、$x^{r/2}$-1 と$x^{r/2}$ +1は、Nと約数を共有している。

ただ、$x^{r/2}$-1 ≡ 0 mod Nにはならない。rは x^r-1 ≡ 0 mod Nとなる「最小」の r だから。Nは、$x^{r/2}$-1 を割らない。N∤ $x^{r/2}$-1

もし、$x^{r/2}$ +1 ≢ 0 mod N で、Nが $x^{r/2}$ +1も割らないなら gcd($x^{r/2}$ +1,N)かgcd($x^{r/2}$ −1,N)のいずれかが、Nを割ることになる。

周期（位数）を使った素因数分解

例えば、先の、N=15, a=7, r=4の時、

$\gcd(a^{r/2}+1,N)=\gcd(7^2+1,15)=\gcd(50,15)=5$
$\gcd(a^{r/2}-1,N)=\gcd(7^2-1,15)=\gcd(48,15)=3$

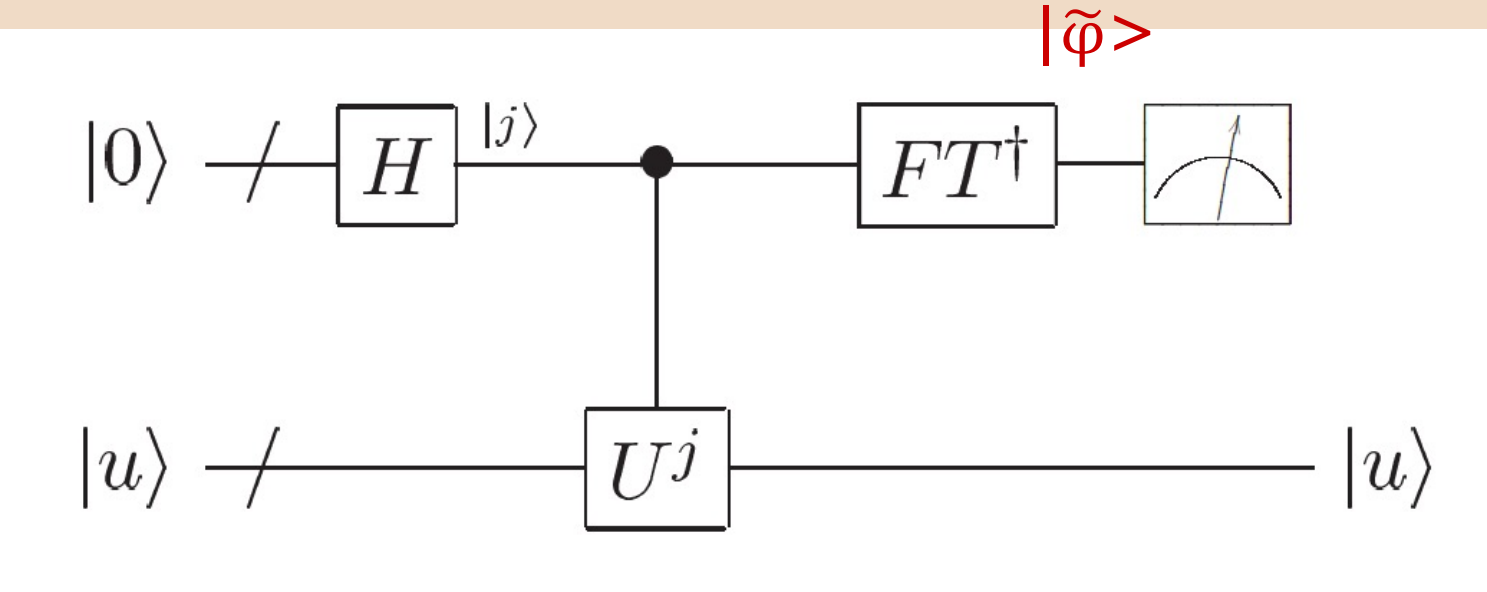
3,5は、15を割る。

周期を使ったショアの素因数分解アルゴリズム

1. a <N なるaをランダムに選ぶ。
2. aとNの最大公約数gcd(a,N)を計算する。（ユークリッドの互除法を用いる） gcd(a,N)が1でなかったら、その数がNの約数である。
3. f(x)＝ a^x mod N なる関数の周期をr とする。$a^r \equiv 1 \bmod N$ すなわち、$a^r - 1$は、Nで割れる。N| $a^r - 1$
（この周期rを求めるのに、量子コンピュータを利用する）
4. rが奇数なら、1.に戻る。
5. rが偶数なら $a^r - 1 = (a^{r/2}+1)(a^{r/2}-1)$と表せる。
6. もし、$a^{r/2} = -1 \bmod N$ なら 1.に戻る。
7. そのいずれでもないなら、gcd($a^{r/2}+1$,N)かgcd($a^{r/2}-1$,N)のいずれかが、Nを割る。

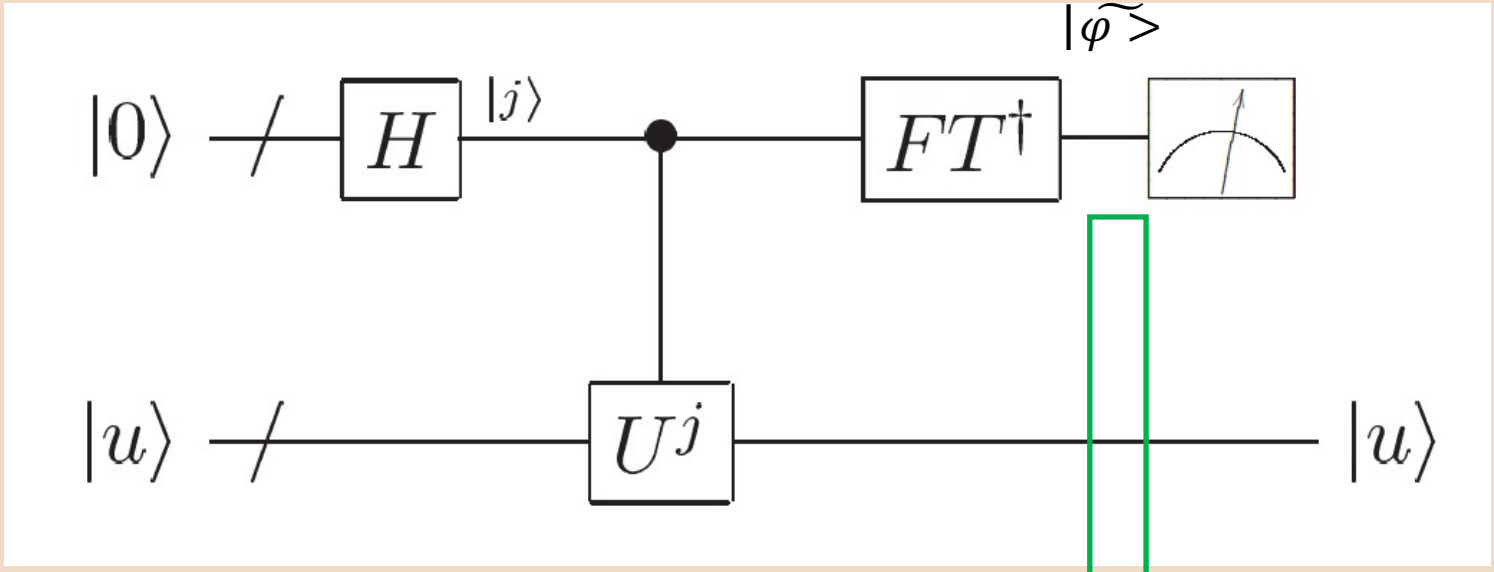
ショアのアルゴリズム
– 量子コンピュータ編 概論

Phase estimator 回路



$$U|u\rangle = e^{2\pi i\varphi}\,|u\rangle$$

すなわち、|u>がUの固有ベクトルで、$e^{2\pi i\varphi}$ がUの固有値になるようなφを求める回路



$$\frac{1}{2^{t/2}} \sum_{j=0}^{2^t-1} e^{2\pi i \varphi j} |j\rangle |u\rangle \rightarrow |\tilde{\varphi}\rangle |u\rangle$$

FT: Fourier 变换

$$|j_1,\ldots,j_n\rangle \rightarrow \frac{\left(|0\rangle + e^{2\pi i 0.j_n}|1\rangle\right)\left(|0\rangle + e^{2\pi i 0.j_{n-1}j_n}|1\rangle\right)\cdots\left(|0\rangle + e^{2\pi i 0.j_1 j_2\cdots j_n}|1\rangle\right)}{2^{n/2}}$$

FT†:逆Fourier 变换

$$\frac{\left(|0\rangle + e^{2\pi i 0.j_n}|1\rangle\right)\left(|0\rangle + e^{2\pi i 0.j_{n-1}j_n}|1\rangle\right)\cdots\left(|0\rangle + e^{2\pi i 0.j_1 j_2\cdots j_n}|1\rangle\right)}{2^{n/2}} \rightarrow |j_1,\ldots,j_n\rangle$$

Phase Estimatorの働き

1. $|0\rangle|u\rangle$ 初期状態

2. $\rightarrow \frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} |j\rangle|u\rangle$ 重ね合わせを作る

3. $\rightarrow \frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} |j\rangle U^j|u\rangle$ ブラックボックスU_jを作用させる

$= \frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} e^{2\pi i j \varphi_u} |j\rangle|u\rangle$ ブラックボックスU_jの作用の結果

4. $\rightarrow |\varphi_u\rangle|u\rangle$ 逆フーリエ変換を作用させる

5. $\rightarrow \varphi_u$ 第一レジスターを測定する

Phase Estimatorを使って、周期を求める

1.	$\|0\rangle\|0\rangle$	初期状態
2.	$\rightarrow \frac{1}{\sqrt{2^t}} \sum_{x=0}^{2^t-1} \|x\rangle\|0\rangle$	重ね合わせを作る
3.	$\rightarrow \frac{1}{\sqrt{2^t}} \sum_{x=0}^{2^t-1} \|x\rangle\|f(x)\rangle$	Uを作用させる
	$\approx \frac{1}{\sqrt{r2^t}} \sum_{\ell=0}^{r-1} \sum_{x=0}^{2^t-1} e^{2\pi i \ell x/r} \|x\rangle\|\hat{f}(\ell)\rangle$	Uの作用の結果
4.	$\rightarrow \frac{1}{\sqrt{r}} \sum_{\ell=0}^{r-1} \|\ell/r\rangle\|\hat{f}(\ell)\rangle$	逆フーリエ変換を作用させる
5.	$\rightarrow \ell/r$	第一レジスターを測定する
6.	$\rightarrow r$	周期 r を求める

Period-finding	$\mathbf{Z}$, +	any finite set	$\{0, r, 2r, \ldots\}$ $r \in G$	$f(x + r) = f(x)$
Order-finding	$\mathbf{Z}$, +	$\{a^j\}$ $j \in Z_r$ $a^r = 1$	$\{0, r, 2r, \ldots\}$ $r \in G$	$f(x) = a^x$ $f(x + r) = f(x)$
Discrete logarithm	$\mathbf{Z}_r \times \mathbf{Z}_r$ + (mod r)	$\{a^j\}$ $j \in Z_r$ $a^r = 1$	$(\ell, -\ell s)$ $\ell, s \in \mathbf{Z}_r$	$f(x_1, x_2) = a^{kx_1 + x_2}$ $f(x_1 + \ell, x_2 - \ell s) = f(x_1, x_2)$
Order of a permutation	$\mathbf{Z}_{2^m} \times \mathbf{Z}_{2^n}$ + (mod 2^m)	$\mathbf{Z}_{2^n}$	$\{0, r, 2r, \ldots\}$ $r \in X$	$f(x, y) = \pi^x(y)$ $f(x + r, y) = f(x, y)$ π = permutation on X
Hidden linear function	$\mathbf{Z} \times \mathbf{Z}$, +	$\mathbf{Z}_N$	$(\ell, -\ell s)$ $\ell, s \in X$	$f(x_1, x_2) =$ $\pi(sx_1 + x_2 \bmod N)$ π = permutation on X
Abelian stabilizer	(H, X) H = any Abelian group	any finite set	$\{s \in H \mid$ $f(s, x) = x,$ $\forall x \in X\}$	$f(gh, x) = f(g, f(h, x))$ $f(gs, x) = f(g, x)$

Shorのアルゴリズムで解ける問題

http://bit.ly/2Ww2Kua

今月6/21 マルレク・サブゼミのお知らせ

3時間で学ぶ
Shorのアルゴリズム入門

“Post-Quantum Cryptography”の流れに、決定的な影響を与えたShorのアルゴリズムの概略を学びます。

- 量子コンピュータではなぜ高速な計算ができるのか？
- 量子コンピュータの出力は、どう取り出せるのか？
- 量子の状態と「位相」
- Shorのアルゴリズムと「位相」の評価
- 「位相」の観測から、関数の「周期」を求める。
- 「周期」から、素因数を求める。

参考資料(マルレク 2018/11/02)

https://www.marulabo.net/docs/20181102-marulec04/

ショアのアルゴリズム と複雑性理論

Shorのアルゴリズムの発見を Shor自身はどう考えていたか？

しかし、コンピュータサイエンスの歴史において、最も重要な問題は、多項式時間またはNP完全問題のいずれかであることがわかっています。したがって、量子コンピュータは、NP完全問題を解くことができない限り、おそらくそれほど広くは役に立たないでしょう。

NP完全問題を効率的に解くことは、理論的コンピュータ科学の「聖杯」です。それが、古典的なコンピュータ上で可能であると期待するのは、非常に少数の人々だけです。量子コンピュータ上で、これらの問題を解決するための多項式時間アルゴリズムを見つけることは、画期的な発見となるでしょう。

量子コンピュータはNP完全問題を解くのに十分に強力ではないといういくつかの弱い徴候があります[Bennett et al。1994]。しかし、私はこの可能性が早々に排除されるべきであるとは思いません。

Shor論文のEnding Remark

BQPクラスの発見

Umesh Vazirani

“Quantum Complexity Theory”
Bernstein ,Vazirani
1993年
https://goo.gl/3y4RSf

拡張されたチャーチ・チューリング・テーゼ

計算可能性の理論がチャーチ-チューリングのテーゼにその基礎を持っていたように、計算の複雑性理論はこのテーゼの現代版の拡張である次のようなテーゼに基づいている。

すなわち、"計算のどんな「合理的な」モデルも確率的チューリング機械で効率的にシミュレートできる。(効率的なシミュレーションとは、シミュレートするマシンの実行時間が多項式によって制限される(上限を持つ)ものである)"

たとえば、単位時間に、任意の長さの語を処理できるコンピュータ、あるいは無限の精度を持つ実数を正確に計算できるコンピュータは、計算の「合理的モデル」ではない。なぜなら、そうしたマシンが物理的に実装できないのは明らかだと思えるからである。

新しいコンピュータは可能か？

有限精度の計算プリミティブしか実装できないと仮定すれば、チューリングマシンモデル（またはそれと等価な多項式時間セルオートマトンモデル）は避けられない選択であると主張されている。

NP≠BPPであるという広く信じられていることを考えれば、このことは、広範囲の重要な計算問題（様々のNP困難な問題）が、コンピュータの能力をはるかに超えたものである示すことになる。

しかし、チューリングマシンは、すべての物理的に実現可能なコンピューティングデバイスとしては不十分なモデルである。その根本的な理由は、チューリングマシンは宇宙の古典的な物理学モデルに基づいているからである。現在の物理理論は、宇宙は量子物理学的であると主張している。我々は、量子物

ファインマンの問題提起

そのようなデバイス、確率論的チューリングマシーンより強力なデバイスの最初の示唆は、10年ほど前のファインマンの論文に現れた。その論文の中で、ファインマンは非常に奇妙な問題を指摘した。

それは指数関数的なスピード低下なしには、確率的チューリングマシン上では、一般的な量子物理システムをシミュレートすることは不可能であるように見えるという問題である。シミュレーションの難しさは、離散的なもので連続系をシミュレートする問題とは関係ない。

我々は、シミュレートされる量子物理システムは離散的なもので、ある種の量子セルオートマトンであると仮定することができる。ファインマンの観察を考慮して、我々は計算複雑性理論の基礎、およびチャーチ・チューリング・テーゼの複雑性理論

新しい量子複雑性のクラスBQPの発見

本論文では、シミュレーションのオーバーヘッドが多項式で制限される万能量子チューリングマシンの存在を証明する。

本論文では、量子チューリングマシンが古典的な確率的チューリングマシンよりも強力である可能性があるという最初の証拠を提示する。

P　：古典的コンピュータで、多項式時間で計算可能なクラス

BQP: 量子コンピュータで、多項式時間で計算可能なクラス

$$\mathbf{P} \subseteq \mathbf{BQP}$$

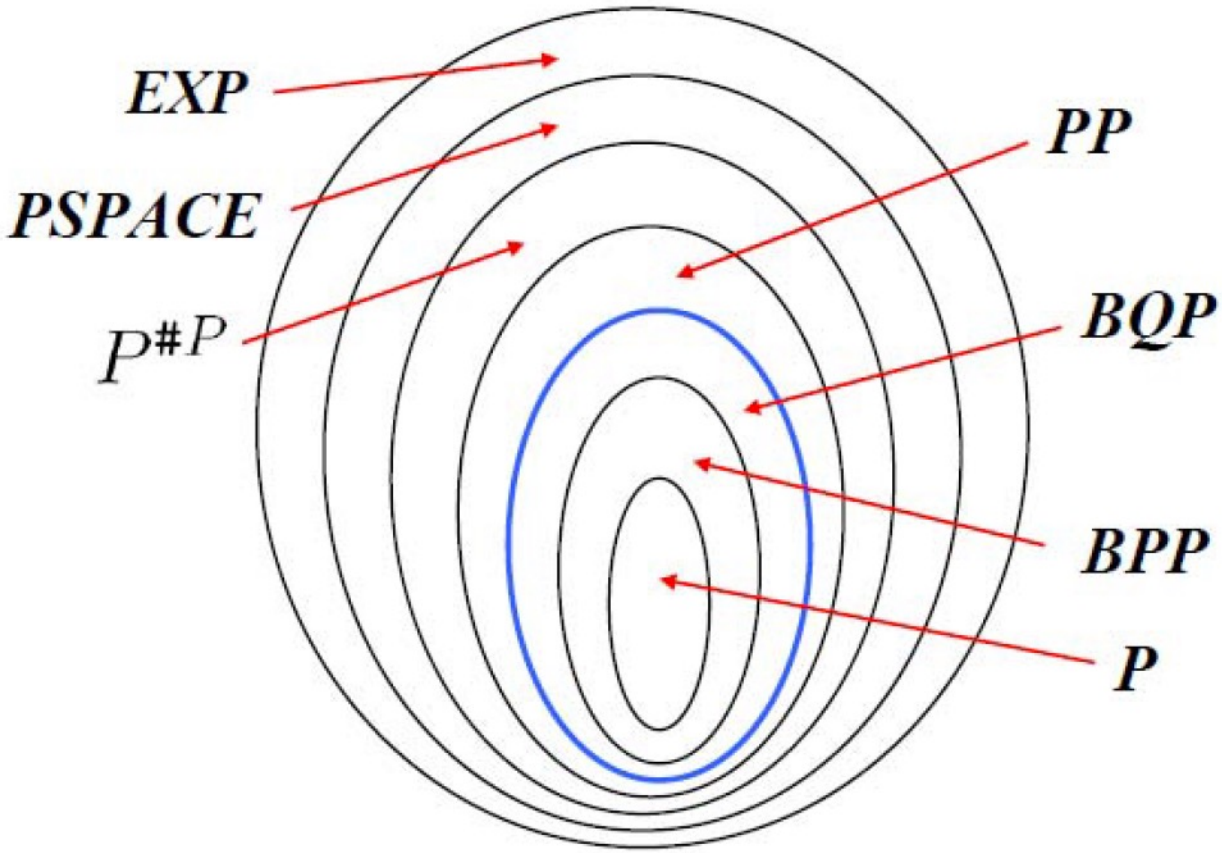
Upper and lower bounds on the power of QTMs

- Theorem 1 **P ⊆ BQP**
- Theorem 2 **BPP ⊆ BQP**
- Theorem 3 **BQP ⊆ PSPACE**
- Theorem 5 **BQP ⊆ $P^{\#P}$**

- **Fourier Sampling and the power of QTMs**

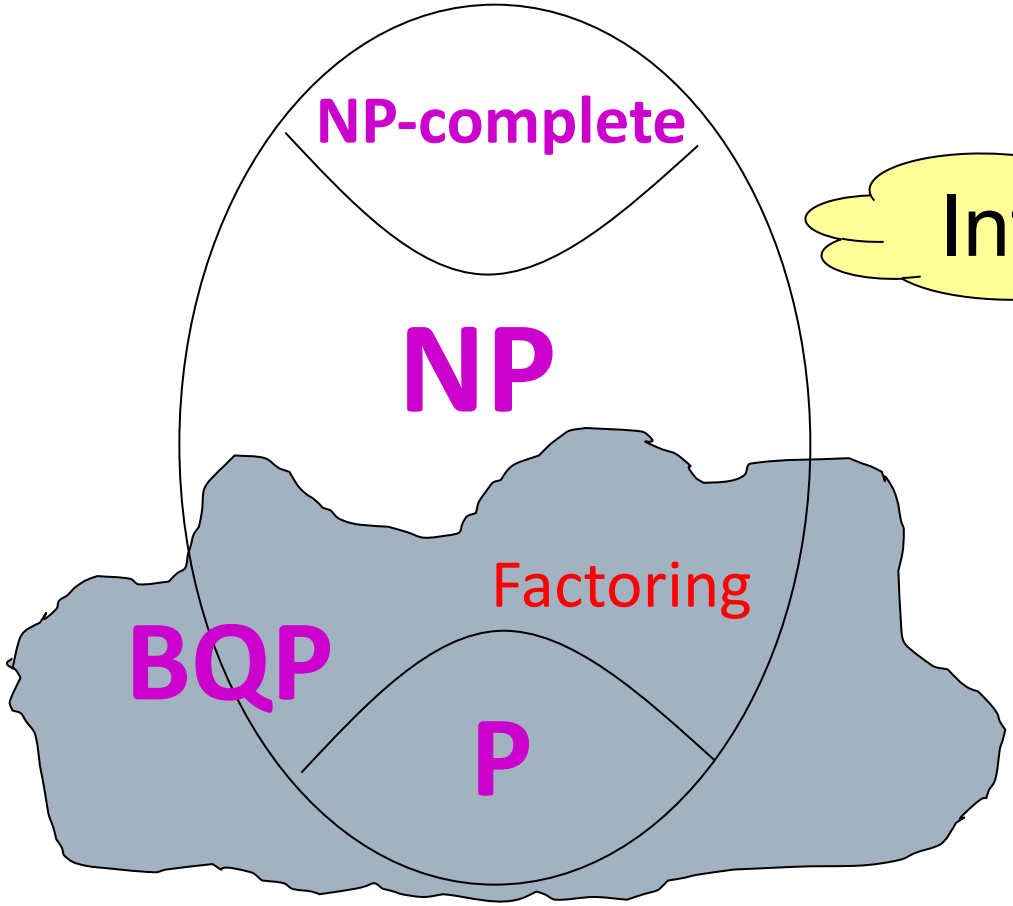
  In this section we give evidence that QTMs are more powerful than bounded error probabilistic TMs. We define the recursive Fourier sampling problem which on input the program for a boolean function takes on value 0 or 1. We show that the recursive Fourier sampling problem is in BQP.

Basic properties of BQP



BQP (Bounded-Error Quantum Polynomial-Time): The class of problems solvable efficiently by a quantum computer, defined by Bernstein and Vazirani in 1993

Shor 1994: Factoring integers is in **BQP**



Manindra Agrawal

PRIMES is in P

Manindra Agrawal,
Neeraj Kayal,
Nitin Saxena

2002年

https://www.cse.iitk.ac.in/users/manindra/algebra/primality_v6.pdf

Since the beginning of complexity theory in the 1960s—when the notions of complexity were formalized and various complexity classes were defined—this problem (referred to as the primality testing problem) has been investigated intensively. It is trivial to see that the problem is in the class co-NP: if n is not prime it has an easily verifiable short certificate, viz., a non-trivial factor of n. In 1974, Pratt observed that the problem is in the class NP too [Pra75] (thus putting it in NP ∩ co-NP).

……

The ultimate goal of this line of research has been, of course, to obtain an unconditional deterministic polynomial-time algorithm for primality testing. Despite the impressive progress made so far, this goal has remained elusive. In this paper, we achieve this.

“Post-Quantum Cryptography” 標準化の動向

- 2015年、NSAは次のような重要な決定を公表した。Shorの発見から、約20年後のことであった。
「我々は、来るべき量子耐性アルゴリズムへの移行について、早いうちから計画づくりとコミュニケーションを開始することを決定した。我々の最終的な目標は、量子コンピュータの潜在的な能力に対して、コスト効率の良いセキュリティを提供することである。」
- NSAの決定を受けて、NISTは2016年から “Post-Quantum Cryptography”の標準化の策定の作業を開始した。その計画によれば、早ければ2022年、遅くとも2024年までには、新しい「ポスト量子暗号技術」の標準を決定するという。

Post-Quantum Cryptography Time Line

Feb 24-26, 2016	NIST Presentation at PQCrypto 2016: *Announcement and outline of NIST's Call for Submissions (Fall 2016)*, *Dustin Moody*
April 28, 2016	NIST releases NISTIR 8105, Report on Post-Quantum Cryptography
Dec 20, 2016	Formal Call for Proposals
Nov 30, 2017	Deadline for submissions
Dec 4, 2017	NIST Presentation at AsiaCrypt 2017: *The Ship Has Sailed: The NIST Post-Quantum Crypto "Competition"*, *Dustin Moody*
Dec 21, 2017	Round 1 algorithms announced (69 submissions accepted as "complete and proper")

Post-Quantum Cryptography Time Line

Apr 11, 2018	NIST Presentation at PQCrypto 2018: *Let's Get Ready to Rumble - The NIST PQC "Competition"*, *Dustin Moody*
April 11-13, 2018	First PQC Standardization Conference - Submitter's Presentations
January 30, 2019	Second Round Candidates announced (26 algorithms) NISTIR 8240
March 15, 2019	Deadline for updated submission packages for the Second Round
August 22-24, 2019	Second PQC Standardization Conference
2020/2021	Round 3 begins or select algorithms
2022/2024	**Draft Standards Available**

NSA
Commercial National Security Algorithm Suite

https://apps.nsa.gov/iaarchive/programs/iad-initiatives/cnsa-suite.cfm

2015/08/15

量子耐性アルゴリズムへの移行のための準備計画

現在のグローバルな環境では、我々の国家とその市民とその利益を守る上で、高速で安全な情報の共有が重要である。強力な暗号化アルゴリズムと安全な標準プロトコルは、我々の国家の安全に貢献し、安全への普遍的な要請と相互運用可能なコミュニケーションに向けた取り組みを助ける死活的に重要なツールである。

現在では、Suite B の暗号化アルゴリズムが 国立標準技術研究所(NIST)によって規定され、機密あるいは非機密の国家安全保障システム（NSS）を保護するソリューションの認可で、NSAの情報保証局で利用されている。以下で、量子耐性アルゴリズムへの移行のための準備計画について告知する。

背景

IAD(The Information Assurance Directorate at the NSA) は、そう遠くない将来、量子耐性アルゴリズムへの移行を開始するであろう。我々は、Suit B を展開した経験に基づいて、来るべき量子耐性アルゴリズムへの移行について、早いうちから計画づくりとコミュニケーションを開始することを決定した。

我々の最終的な目標は、量子コンピュータの潜在的な能力に対して、コスト効率の良いセキュリティを提供することである。我々は、合衆国政府、ベンダー、標準化団体をまたいだパートナーと共に、アルゴリズムの新しいSuitを獲得する明確な計画が存在することを保証するための作業を行なっている。そのアルゴリズムは、オープンで透明性をもって開発され、我々の次の暗号化アルゴリズムSuitの基礎を形作るであろう。

この新しいSuitが開発され、量子耐性アルゴリズムを実装した製品が利用可能になるまで、我々は、現在のアルゴリズムを信頼するであろう。

Suit Bの楕円曲線アルゴリズムへの移行を、まだ行なっていないパートナーならびにベンダーは、現時点で、そのための大きな支出せずに、その代わりに、来るべき量子耐性アルゴリズムへの移行を準備することを、我々は勧めてきた。

すでにスイートBに移行しているベンダーやパートナーにとって、そのために多大な努力を要したことを我々は認識している。その努力に感謝している。National Securityのユーザーのための情報セキュリティを向上させるために協力しながら、進行中の量子コンピュータの脅威に対して、あなたたちの継続的なサポートを我々は期待している。

不幸なことに、楕円曲線の利用の拡大は、量子コンピューティング研究の絶え間ない進歩の事実と衝突するものである。すなわち、量子コンピューティングの研究は、楕円曲線暗号化は、多くの人がかってそうなるだろうと期待したような長期間にわたって有効なソリューションではないことを明らかにした。こうして、我々は、戦略の見直しを余儀なくされてきた。

重要なことは、我々は、ベンダーにSuite Bアルゴリズムの実装を中止するように依頼しているのではなく、また国家安全保障の顧客にこれらのアルゴリズムの使用を中止するよう依頼しているのでもないということである。むしろ、我々は量子の世界の安全の将来に備えて、現時点でベンダーと私たちのユーザーにより多くの柔軟性を与えたいと思う。

楕円曲線プロトコルが使用されることになっているところでは、新しい提案がまだなされていない以上、セキュリティ評価の長い歴史と時間をかけてテストされた実装である Suit B規格が可能な限り最大限に使われるのを勧めるものである。

Guidance

新しいアルゴリズムスイートが開発され製品に実装されるまでの間に実行する緩和策を探しているユーザには、できることがることがいくつかある。

第一に、多くのシステム（特に小規模システム）のアルゴリズムでは、大きな鍵サイズを使用するのが賢明である（以下の表を参照）。

さらに、機密性の高い国家安全保障情報を長い寿命で保護するために、階層化された商用ソリューションを使用しているIADのユーザは、量子耐性保護の層の実装を始めるべきでである。

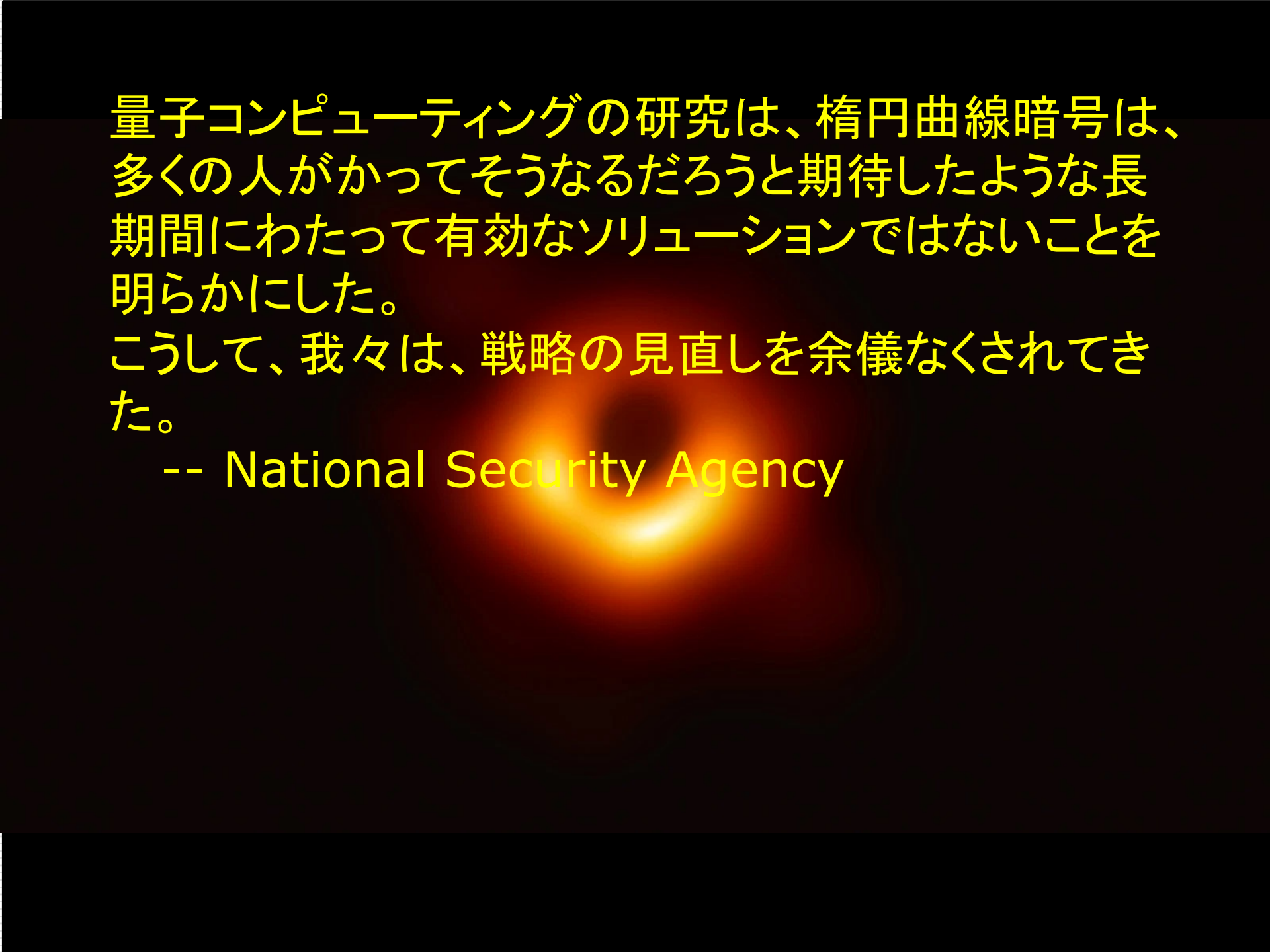
そのような保護は、今日でも、大きな対称鍵および特定の安全なプロトコル標準を使用することによって実装することが可能かもしれない。

Algorithm	Function	Specification	Parameters
Advanced Encryption Standard (AES)	Symmetric block cipher used for information protection	FIPS Pub 197	Use 256 bit keys to protect up to TOP SECRET
Elliptic Curve Diffie-Hellman (ECDH) Key Exchange	Asymmetric algorithm used for key establishment	NIST SP 800-56A	Use Curve P-384 to protect up to TOP SECRET.
Elliptic Curve Digital Signature Algorithm (ECDSA)	Asymmetric algorithm used for digital signatures	FIPS Pub 186-4	Use Curve P-384 to protect up to TOP SECRET.
Secure Hash Algorithm (SHA)	Algorithm used for computing a condensed representation of information	FIPS Pub 180-4	Use SHA-384 to protect up to TOP SECRET.

Diffie-Hellman (DH) Key Exchange	Asymmetric algorithm used for key establishment	IETF RFC 3526	Minimum 3072-bit modulus to protect up to TOP SECRET
RSA	Asymmetric algorithm used for key establishment	NIST SP 800-56B rev 1	Minimum 3072-bit modulus to protect up to TOP SECRET
RSA	Asymmetric algorithm used for digital signatures	FIPS PUB 186-4	Minimum 3072 bit-modulus to protect up to TOP SECRET.

量子コンピューティングの研究は、楕円曲線暗号は、多くの人がかってそうなるだろうと期待したような長期間にわたって有効なソリューションではないことを明らかにした。
こうして、我々は、戦略の見直しを余儀なくされてきた。

-- National Security Agency



NISTIR 8105
”Report on Post-Quantum Cryptography”

https://nvlpubs.nist.gov/nistpubs/ir/2016/NIST.IR.8105.pdf

2016/04/28

NISTIR 8105

Report on Post-Quantum Cryptography

Lily Chen
Stephen Jordan
Yi-Kai Liu
Dustin Moody
Rene Peralta
Ray Perlner
Daniel Smith-Tone

This publication is available free of charge from:
http://dx.doi.org/10.6028/NIST.IR.8105

NIST
National Institute of
Standards and Technology
U.S. Department of Commerce

Abstract

近年、量子コンピュータ – 量子力学的現象を利用して従来のコンピュータでは難しかったり手が出ない数学的問題を解くのに利用するマシン -- について、相当の量の研究がなされてきた。

もしも、大規模な量子コンピュータがいったん構築されれば、それは現在利用されている公開キー暗号の多くを破ることができるだろう。このことは、インターネット上やあらゆるところでのデジタル通信の信頼性と統合性を深刻に傷つけることになるであろう。

ポスト量子暗号（量子耐性暗号とも呼ばれる）の目標は、量子コンピュータ・古典コンピュータの双方に対して安全な暗号システムを開発し、既存の通信プロトコルとネットワークと相互運用できるようにすることである。

この内部レポートは、アメリカ国立標準技術研究所（NIST）の現時点での量子コンピューティングとポスト量子暗号の状態についての理解を共有し、この問題空間に向けて前進しようとするNISTの最初の計画の概要を示すものである。

このレポートはまた、この新しい暗号インフラへの移行がチャレンジングであることを認識している。それゆえ、諸機関が暗号の変化にフォーカスすることの必要性を強調している。

1. Introduction

過去30年間で、公開キー暗号は我々のの世界的な通信デジタルインフラの不可欠な要素となった。これらのネットワークは、携帯電話、インターネットコマース、ソーシャルネットワーク、クラウドコンピューティングなど、私たちの経済、セキュリティ、および生活様式にとって重要な、非常に多くのアプリケーションをサポートしている。この繋がった世界では、個人、企業、および政府が安全にコミュニケーションをする能力が最も重要である。

我々の最も重要な通信プロトコルの多くは、主に3つの主要な暗号化機能、すなわち公開キー暗号、デジタル署名、キー交換に依存している。現在、これらの機能は主にDiffie-Hellmanキー交換、RSA暗号システム、および楕円曲線暗号システムを使用して実装されている。これらの安全性は、素因数分解や離散対数問題などの特定の数論的問題がさまざまな群に対して困難であることに依存している

1994年、Bell研究所のPeter Shorは、物質の物理的性質とエネルギーを利用して計算を行う新しい技術である量子コンピュータが、これらの問題を効率的に解決し、それによってそのような仮定に基づくすべての公開鍵暗号システムを無力にすることを示した。したがって、十分に強力な量子コンピュータは、鍵交換から暗号化、デジタル認証まで、現代のコミュニケーションの多くの形態を危険にさらすだろう。

量子コンピュータが古典的なコンピュータよりも速くある種の問題を解決するために利用され得るという発見は、量子コンピューティングへの大きな関心を呼び起こした。量子複雑度は古典的複雑度と根本的に異なるのか？ 大規模な量子コンピュータはいつ構築されるのか？ 量子暗号攻撃と古典的コンピュータ暗号攻撃の両方に抵抗する方法はあるのか？ 研究者はこれらの疑問に取り組んでいる。

Shorのアルゴリズムが発見されてから20年間で、量子アルゴリズムの理論は著しく発展した。指数関数的な高速化を達成する量子アルゴリズムが、物理シミュレーション、数論、トポロジーに関するいくつかの問題に対して発見された。それにもかかわらず、量子計算による指数関数的な高速化を認められる問題のリストは比較的小さいままである。対照的に、検索や、衝突検出、ブール式の評価に関連する幅広いクラスの問題に対して、より緩やかなスピードアップが開発されている。古典的モデルと量子モデルの実現可能性の間にはギャップがある。

特に、Groverの検索アルゴリズムは、非構造化された検索問題に対して√Nの高速化を提供する。このような高速化によって暗号化技術が時代遅れになることはないが、対称鍵の場合でも、より大きな鍵サイズが必要になるという効果はある。

RSAやAESなどの一般的な暗号化アルゴリズムに対する（大規模）量子コンピュータの影響の概要については、表1を参照されたい。

これらの量子の優位性をどの程度まで押し上げることができるか、また古典的モデルと量子モデルの実現可能性の差がどれほど広いのかは、まだよくわかっていない。

Impact of Quantum Computing on Common Cryptographic Algorithms

Cryptographic Algorithm	Type	Purpose	Impact from large-scale quantum computer
AES	Symmetric key	Encryption	Larger key sizes needed
SHA-2, SHA-3	----------	Hash functions	Larger output needed
RSA	Public key	Signatures, key establishment	No longer secure
ECDSA, ECDH (Elliptic Curve Cryptography)	Public key	Signatures, key exchange	No longer secure
DSA (Finite Field Cryptography)	Public key	Signatures, key exchange	No longer secure

大規模な量子コンピュータが、いつ構築されるだろうかという問いは、複雑なものである。過去には、大規模な量子コンピュータが物理学的な可能性であることについても、今より明確ではなかった。その一方で、現在では、多くの科学者が、それは単なる重要な工学的挑戦に過ぎないと考えている。ある工学者は、次の20年かその程度の年数があれば、現在利用されている公開キーのスキーマを本質的に破る大規模な量子コンピュータを構築するには十分だろうと予想している。

歴史的には、我々の現代的な公開キー暗号のインフラを配置するのに、ほとんど二十年近くかかってきた。それゆえ、量子コンピュータ時代の到来の正確な時間を評価できるか否とにかかわらず、我々は今こそ我々の情報セキュリティシステムが量子コンピューティングに耐性を持つことができるように準備を始めなければならない。

新しい量子耐性プリミティブを利用することによって、私たちの公開鍵インフラが損なわれないままでいることを願って、量子コンピューティングの将来における情報セキュリティの問題に取り組むグローバルなコミュニティが生まれている。

アカデミーの世界では、この新しい科学は「ポスト量子暗号」という名前を付けられている。2006年に始まった一連のカンファレンスPQCryptoを持つ活発な研究分野である。

こうした研究は、特にヨーロッパと日本では、EUプロジェクトのPQCryptoとSAFEcrypto、そして日本のCREST Crypto-Mathプロジェクトを通じて、国の大きな財政支援を受けている。

これらすべての情報を考慮すると、量子耐性技術を開発するための努力が激しさを増していることは明らかである。これらの投資によって示唆される、新しいポスト量子公開鍵暗号を標準化する必要性の緊急性も同様に明らかである。

NIST暗号化標準が業界および世界中の他の標準化団体によって承認されるためには、コミュニティと関わることが重要である。この内部レポートは、量子コンピューティングとポスト量子暗号の現状についてのNISTの現在の理解を共有している。報告書ではまた、前に進むための我々の最初の計画を概説する。

2 An Overview of Quantum-Resistant Cryptography

今日の公開鍵暗号の最も重要な用途は、デジタル署名と鍵確立である。1章で述べたように、大規模な量子コンピュータの構築は、これらの公開鍵暗号システムの多くを安全でないものにするだろう。特に、これには、RSAなどの整数分解の難しさに基づくもの、および離散対数問題の困難さに基づくものが含まれる。対照的に、対称鍵システムへの影響はそれほど劇的ではない（表1を参照）。結果的には、古典的コンピュータと量子コンピュータの両方からの攻撃に対して耐性があると考えられているアルゴリズムの探求は、公開鍵アルゴリズムに集中してきた。この章では、ポスト量子プリミティブが提案されている主なファミリの概要を簡単に説明する。これらのファミリには、格子、コード、および多変量多項式に基づくもの、およびその他の少数のものがある。詳細については、[3、4]を参照されたい。

Lattice-based cryptography

格子問題に基づく暗号システムは、いくつかの理由で新たな関心を集めている。エキサイティングな新しいアプリケーション（完全準同型暗号化、コード難読化、属性ベースの暗号化など）は、ラティス・ベースの暗号化を使用して可能になった。大部分のラティス・ベースの鍵確立アルゴリズムは、比較的単純で効率的、そして高度に並列化可能である。また、いくつかのラティス・ベースのシステムの安全性は、平均的な場合よりも、最悪の場合の難しさの仮定の下では、確かに安全である。他方では、既知の暗号解読技術に対してさえも格子スキームの安全性の正確な推定値を与えることは困難であることが証明されている。

Code-based cryptography

1978年に、McEliece暗号システムが最初に提案され、それ以来破られていない。それ以来、誤り訂正符号に基づく他のシステムが提案されてきた。非常に高速なのだが、ほとんどのコードベースのプリミティブは非常に大きなキーサイズを持つことに悩まされている。キーサイズを減らすために、より新しい変種では、コードにさらに構造を導入しているのだが、追加された構造は、いくつかの提案に基づく攻撃の成功につながっている。コードベースの署名についてはいくつかの提案があるが、コードベースの暗号化は暗号化方式でより成功している。

Multivariate polynomial cryptography

これらのスキームは、有限体上の多変量多項式のシステムを解くことの困難さに基づいている。過去数十年にわたっていくつかの多変量暗号システムが提案されてきたが、その多くは破られている。多変量暗号化方式に対するいくつかの提案がある一方で、歴史的には、多変量暗号化は、署名へのアプローチとしてより成功していた。

Hash-based signatures

ハッシュベースの署名は、ハッシュ関数を使用して構築されたデジタル署名である。 量子攻撃に対しても、それが安全であることは、よく理解されている。より効率的なハッシュベースの署名方式の多くは、署名者が以前に署名されたメッセージの正確な数の記録を保持しなければならず、この記録に誤りがあるとセキュリティが不安定になるという欠点がある。 もう一つの欠点は、限られた数の署名しか作成できないことである。署名の数は、事実上無制限であり、その数を増やすことはできるのだが、これは署名のサイズも大きくする。

Other

上記のファミリーに入らない様々なシステムが提案されてきた。そのような提案の1つは、超特異楕円曲線上の同質性の評価に基づいている。楕円曲線上の離散対数問題は、量子コンピュータ上のShorのアルゴリズムによって効率的に解くことができるのだが、超特異曲線上の同次問題には、同じような量子攻撃が可能であるかは知られていない。

他のいくつかの提案、例えば、conjugacy search問題とbraid groupに関連した問題に基づくものだが、その安全性を確信させる十分な分析はまだ存在しない。

NISTIR 8240

"Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process"

https://nvlpubs.nist.gov/nistpubs/ir/2019/NIST.IR.8240.pdf

2019/01/30

現在の状況
Round2 candidate 開始

- NISTは、2019年1月30日、Post-Quantum Cryptography StandardizationのRound1 のまとめを発表して、Round2への移行を宣言する。
- 正式なドキュメントは、**NISTIR 8240** “Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process” https://nvlpubs.nist.gov/nistpubs/ir/2019/NIST.IR.8240.pdf である。

http://bit.ly/2X76gIb

NISTIR 8240

Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process

Gorjan Alagic
Jacob Alperin-Sheriff
Daniel Apon
David Cooper
Quynh Dang
Carl Miller
Dustin Moody
Rene Peralta
Ray Perlner
Angela Robinson
Daniel Smith-Tone
Computer Security Division
Information Technology Laboratory

Yi-Kai Liu
Applied and Computational Mathematics Division
Information Technology Laboratory

This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8240

January 2019

U.S. Department of Commerce
Wilbur L. Ross, Jr., Secretary

National Institute of Standards and Technology
Walter Copan, NIST Director and Under Secretary of Commerce for Standards and Technology

Abstract

NIST 米国標準技術局は、一般の競争に似た方法を通じて、1つ以上の公開鍵暗号アルゴリズムを選択している。

新しい公開鍵暗号規格では、FIPS 186-4、Digital Signature Standard（DSS）を強化するために、1つ以上の追加のデジタル署名、公開鍵暗号化、および鍵確立アルゴリズムが定義されるだろう。 同様に、特別刊行SP 800-56A Revision 2、離散対数暗号を使用したペアワイズ鍵確立方式、およびSP 800-56B、素因数分解を使用したペアワイズ鍵確立方式も推奨される。

これらのアルゴリズムは、量子コンピュータの出現後を含む、近い将来に機密情報を十分に保護できるようになることを意図している。

2017年11月に、検討のために82の候補アルゴリズムがNISTに提出された。これらのうち、69が最小承認基準と提出要件の両方を満たし、2017年12月20日に第1ラウンドの候補として承認され、NIST ポスト量子暗号標準化プロセスの第1ラウンドの始まりとなった。

このレポートは、第1ラウンドの候補者の一般からのフィードバックと内部レビューに基づいて、評価基準と選択プロセスについて述べている。

そして競争の第2ラウンドに進むための、2019年1月30日に発表された26の候補アルゴリズムの概要をまとめている。

公開鍵暗号化と鍵確立のための第2ラウンドの17のアルゴリズムの候補は、BIKE、Classic McEliece、CRYSTALS-KYBER、FrodoKEM、HAC、LAC、LEDAcrypt（LEDAkemとLEDApkcのマージ）、NewHope、NTRU（NTRUEncrypt / NTRUのマージ） HRSS-KEM、NTRUプライム、NTS-KEM、ROLLO（LAKE / LOCKER / Ouroboros-Rの合併）、Round5（Hila5 / Round2の合併）、RQC、SABRE、SIKE、Three Bears である。

デジタル署名の第2ラウンドの9つの候補は、CRYSTALS-DILITHIUM、FALCON、GeMSS、LUOV、MQDSS、ピクニック、qTESLA、Rainbow、およびSPHINCS +である。

これまでの経過と今後の予定

Feb 24-26, 2016	NIST Presentation at PQCrypto 2016: *Announcement and outline of NIST's Call for Submissions (Fall 2016)*, *Dustin Moody*
April 28, 2016	**NIST releases NISTIR 8105, Report on Post-Quantum Cryptography**
Dec 20, 2016	Formal Call for Proposals
Nov 30, 2017	Deadline for submissions
Dec 4, 2017	NIST Presentation at AsiaCrypt 2017: *The Ship Has Sailed: The NIST Post-Quantum Crypto "Competition"*, *Dustin Moody*
Dec 21, 2017	Round 1 algorithms announced (69 submissions accepted as "complete and proper")

Apr 11, 2018	NIST Presentation at PQCrypto 2018: *Let's Get Ready to Rumble - The NIST PQC "Competition"*, *Dustin Moody*
April 11-13, 2018	First PQC Standardization Conference - Submitter's Presentations
January 30, 2019	**Second Round Candidates announced (26 algorithms) NISTIR 8240**
March 15, 2019	Deadline for updated submission packages for the Second Round
August 22-24, 2019	Second PQC Standardization Conference
2020/2021	**Round 3 begins or select algorithms**
2022/2024	**Draft Standards Available**

今後の予定

Round 2 Submissions

Public-key Encryption and Key-establishment Algorithms

BIKE /Classic McEliece /CRYSTALS-KYBER / FrodoKEM /HQC /LAC /LEDAcrypt /NewHope / NTRU /NTRU Prime /NTS-KEM /ROLLO /Round5 /RQC /SABER /SIKE /Three Bears

17件

Digital Signature Algorithms

CRYSTALS-DILITHIUM /FALCON /GeMSS / LUOV /MQDSS /Picnic /qTESLA /Rainbow / SPHINCS+

9件

http://bit.ly/2W9Jxz3

NIST
Presentation at PQCrypt

NIST
Presentation at AsiaCrypt 2016:

Post-Quantum Cryptgraphy:
NIST's Plan for the Future
https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/pqcrypto-2016-presentation.pdf

2016/02/22-24

Post-Quantum Cryptography: NIST's Plan for the Future

Dustin Moody
Post Quantum Cryptography Team
National Institute of Standards and Technology (NIST)

The sky is falling?

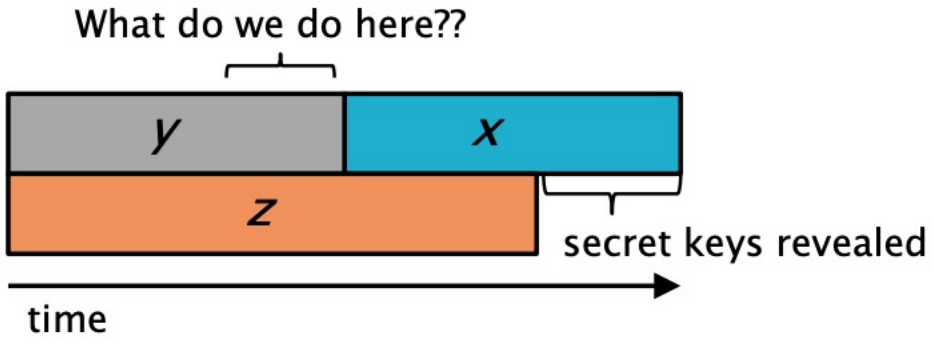
- When will a quantum computer be built?
  - 15 years, $1 billion USD, nuclear power plant (PQCrypto 2014, Matteo Mariantoni)

- Impact:
  - Public key crypto:
    - ~~RSA~~
    - ~~Elliptic Curve Cryptography (ECDSA)~~
    - ~~Finite Field Cryptography (DSA)~~
    - ~~Diffie-Hellman key exchange~~
  - Symmetric key crypto:
    - AES — Need larger keys
    - Triple DES — Need larger keys
  - Hash functions:
    - SHA-1, SHA-2 and SHA-3 — Use longer output

How soon do we need to worry?

- How long does encryption need to be secure (*x* years)
- How long to re-tool existing infrastructure with quantum safe solution (*y* years)
- How long until large-scale quantum computer is built (*z* years)

Theorem (Mosca): If $x + y > z$, then worry



- NSA is transitioning in the "not too distant" future <https://www.iad.gov/iad/programs/iad-initiatives/cnsa-suite.cfm>
- European PQCrypto project
- ETSI work
- IETF - hash-based signature RFC's
- NIST report - <http://csrc.nist.gov/publications/drafts/nistir-8105/nistir_8105_draft.pdf>

Timeline

- Fall 2016 – formal Call For Proposals
- Nov 2017 – Deadline for submissions
- 3-5 years – Analysis phase
  - NIST will report its findings
- 2 years later – Draft standards ready

- Workshops
  - Early 2018 – submitter's presentations
  - One or two during the analysis phase

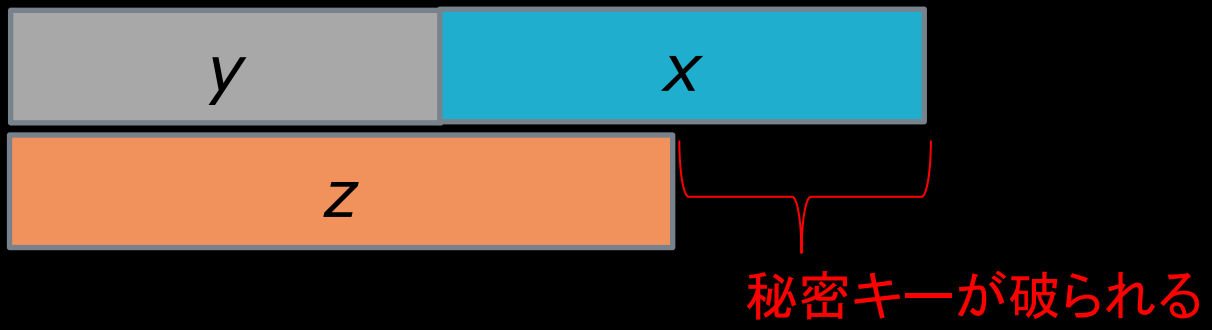
Conclusion

- NIST is calling for quantum-resistant algorithms
  - We see our role as managing a process of achieving community consensus in a transparent and timely manner
  - Different from (but similar to) AES/SHA-3 competitions
- We don't have all the answers
- Wanted: Postdocs, guest researchers at NIST
- We would like public feedback
  - Email: pqc-comments@nist.gov
  - PQC forum: pqc-forum@nist.gov

x年：我々の現在の暗号技術が、何年有効であってほしいか？

y年：量子コンピュータの攻撃に耐える暗号インフラを構築するのに
　　何年必要か？

z年：大規模な量子コンピュータが構築されるまで何年かかるか？



Moscaの定理：

x+y>z なら、暗号システムは破られる

1958年~1986年　28年間有効

1984年~2007年 23年間有効

2004年~　発行中

2024年　発行予定

いつまで有効か？

NIST
Presentation at AsiaCrypt 2017:

The Ship Has Sailed: The NIST Post-Quantum Crypto "Competition"

Dustin Moody

https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/asiacrypt-2017-moody-pqc.pdf

2017/12/04

THE SHIP HAS SAILED

The NIST Post-Quantum Crypto "Competition"

Dustin Moody, NIST

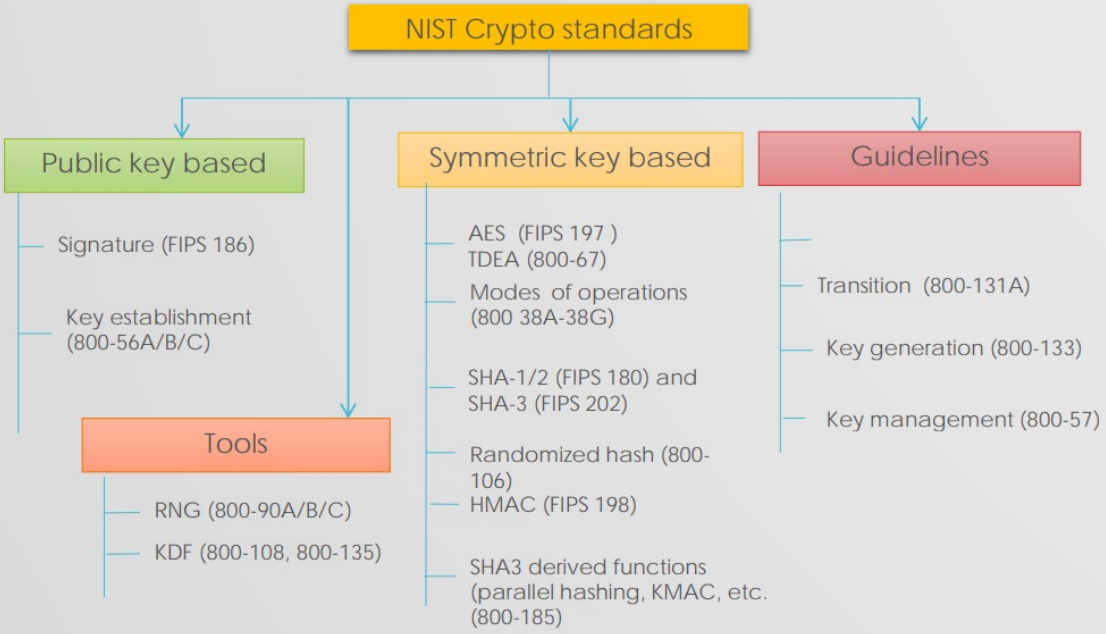
National Institute of Standards and Technology
U.S. Department of Commerce

https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/asiacrypt-2017-moody-pqc.pdf

2017/12/04

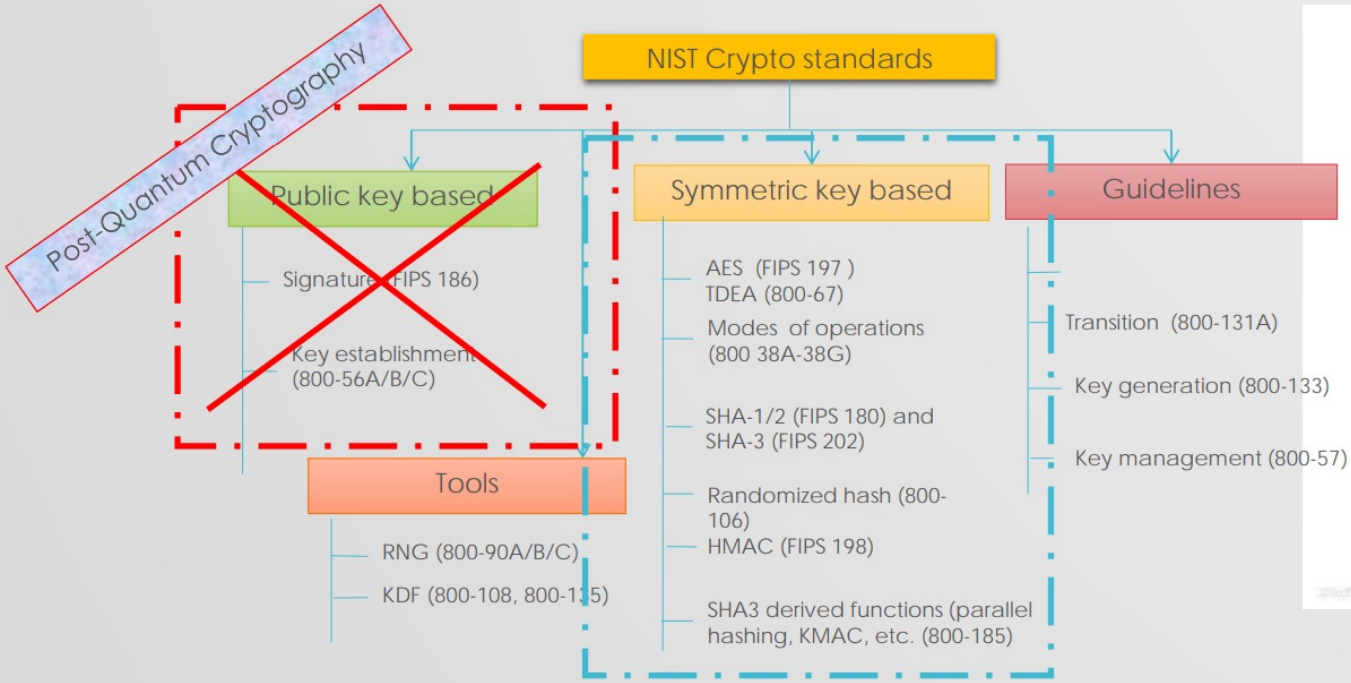
THE SKY IS FALLING?

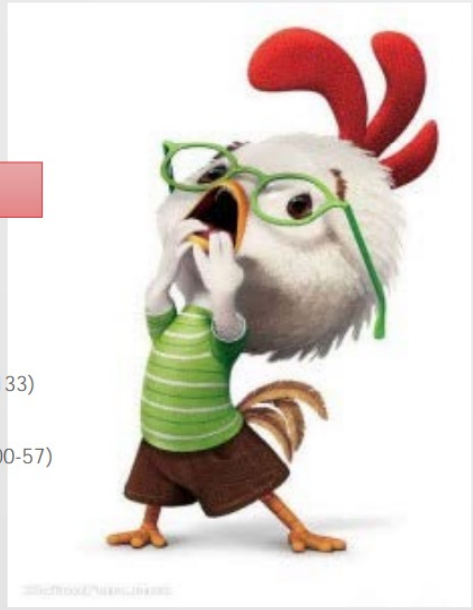
- If a large-scale quantum computer could be built then….



THE SKY IS FALLING?

- If a large-scale quantum computer could be built then….



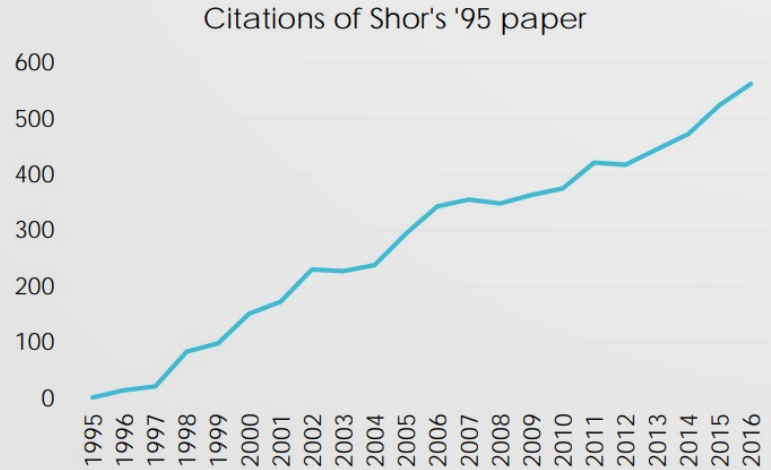


POST-QUANTUM CRYPTOGRAPHY (PQC)

- Cryptosystems which run on classical computers, and are considered to be resistant to quantum attacks
  - Also known as "quantum-safe" or "quantum-resistant"

- PQC needs time to be ready
  - Efficiency
  - Confidence – cryptanalysis
  - Standardization
  - Usability and interoperability

    (IKE, TLS, etc… use public key crypto)



PQC STANDARDIZATION - TOO EARLY?

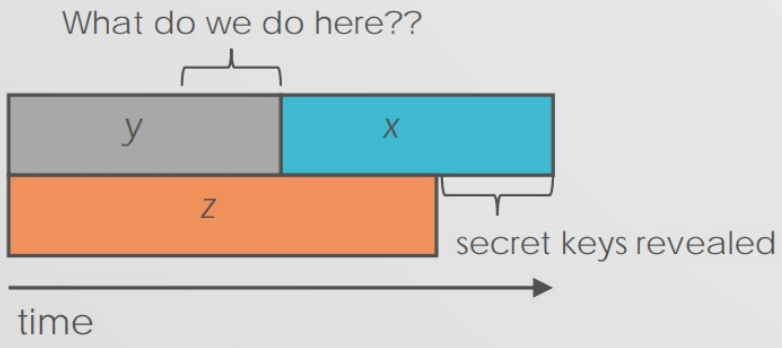
- There has been much debate whether it is too early to look into PQC standardization
- When will a (large-scale) quantum computer be built?
  - "There is a 1 in 7 chance that some fundamental public-key crypto will be broken by quantum by 2026, and a 1 in 2 chance of the same by 2031."

    – Dr. Michele Mosca, U. of Waterloo
- Our experience tells that we need (at least) several years to develop and deploy PQC standards

HOW SOON DO WE NEED TO WORRY?

- How long does your information need to be secure (*x* years)
- How long to re-tool with a quantum safe solution (*y* years)
- How long until a large-scale quantum computer is built (*z* years)

Theorem (Mosca): If $x + y > z$, then worry!



NSA ANNOUNCEMENT

- Aug 2015 - NSA's Information Assurance Directorate updated its list of Suite B cryptographic algorithms
  - "IAD will initiate a transition to quantum resistant algorithms in the not too distant future. Based on experience in deploying Suite B, we have determined to start planning and communicating early about the upcoming transition to quantum resistant algorithms."
- Standardization is the first step towards the transition

THE DECISION TO MOVE FORWARD

- NIST decided **it is the time** to look into standardization
- We see our role as managing a process of achieving community consensus in a **transparent** and **timely** manner
- We do not expect to "pick a winner"
  - Ideally, several algorithms will emerge as 'good choices'

NIST
Presentation at POCrypt 2018:

Let's Get Ready to Rumble
The NIST PQC "Competition"

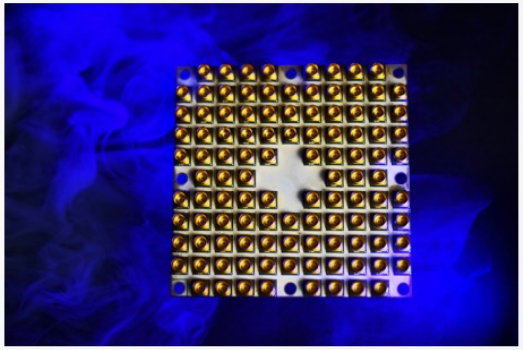
https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/PQCrypto-April2018_Moody.pdf

2018/04/11

Let's Get Ready to Rumble- The NIST PQC "Competition"

Dustin Moody
NIST

The Rise of Quantum Computing

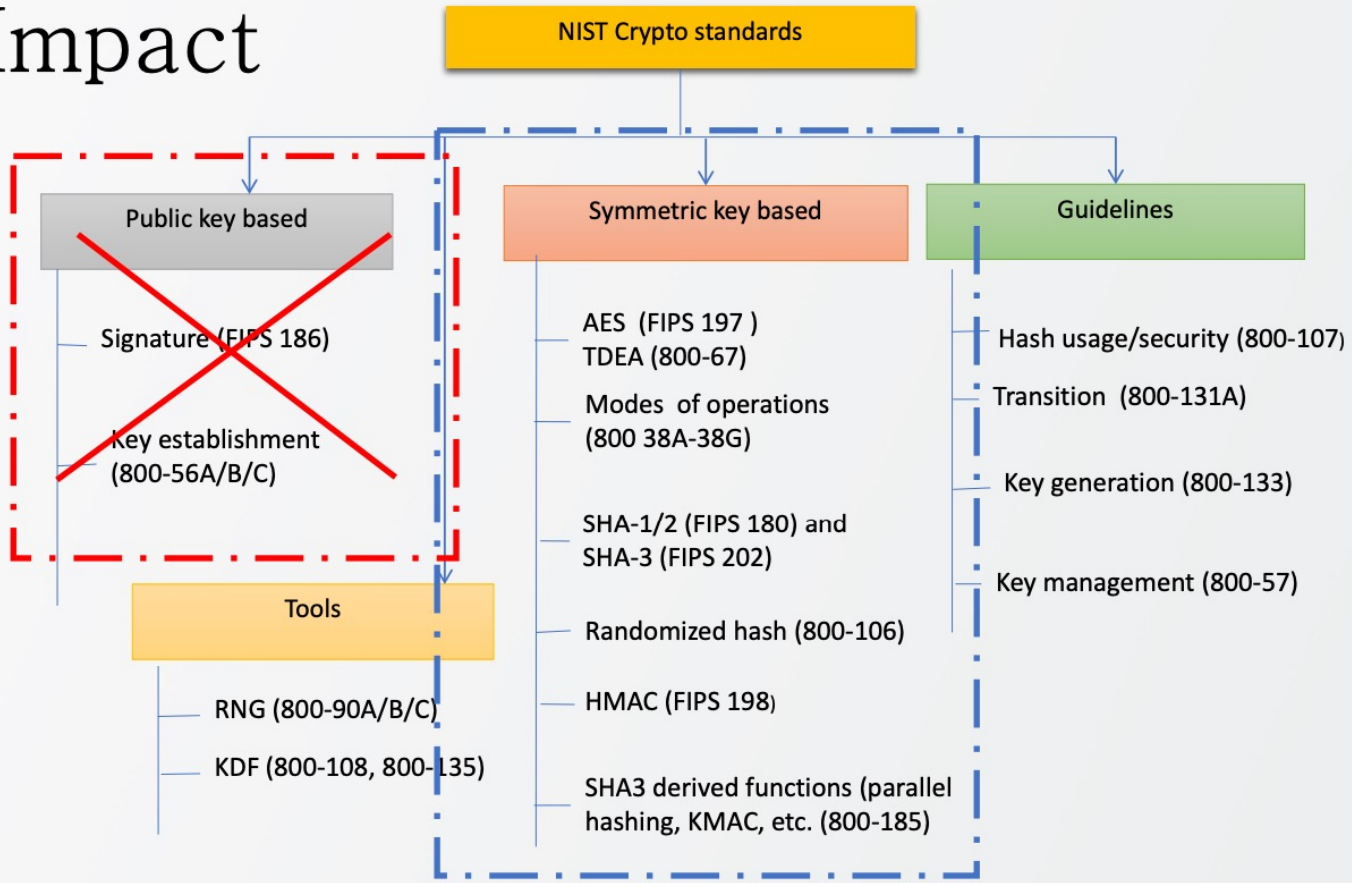


Intel's 49-qubit chip
"Tangle-Lake"
January 2018

Google's 72-qubit chip
"Bristlecone"
March 2018

IBM's 50-qubit
quantum computer
November 2017

The Impact

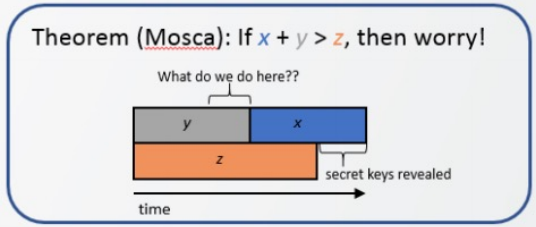

NIST Crypto standards
Public key based
Signature (FIPS 186)
Key establishment (800-56A/B/C)
Tools
RNG (800-90A/B/C)
KDF (800-108, 800-135)
Symmetric key based
AES (FIPS 197)
TDEA (800-67)
Modes of operations (800 38A-38G)
SHA-1/2 (FIPS 180) and SHA-3 (FIPS 202)
Randomized hash (800-106)
HMAC (FIPS 198)
SHA3 derived functions (parallel hashing, KMAC, etc. (800-185)
Guidelines
Hash usage/security (800-107)
Transition (800-131A)
Key generation (800-133)
Key management (800-57)

PQC Standardization – when?

- There had been much debate about whether it is too early to look into PQC standardization
- When will a (large-scale) quantum computer be built?

  - **"There is a 1 in 7 chance that some fundamental public-key crypto will be broken by quantum by 2026, and a 1 in 2 chance of the same by 2031."**

    – Dr. Michele Mosca, (April 2015)



- Our experience tells us that we need (at least) several years to develop and deploy PQC standards

The Decision to Move Forward

- Aug 2015 – NSA statement
  - … "IAD will initiate a transition to *quantum resistant algorithms* in the **not too distant future** …"
- Feb 2016 – NIST Report on PQC (NISTIR 8105)
- Feb 2016 – NIST announcement at PQCrypto

- We see our role as **managing a process** of achieving community consensus in a **transparent** and **timely** manner

- We do not expect to "pick a winner"

Scope

- **Signatures**
  - Public-key schemes for generating/verifying signatures (see FIPS 186-4)
- **Encryption**
  - Key transport from one party to another
  - Exchanging encrypted secret values between two parties to establish shared secret value (see SP 800-56B)
- **Key-establishment (KEMs)**
  - Schemes like Diffie-Hellman key exchange (see SP 800-56A)

Differences with past Competitions

- Post-quantum cryptography is more complicated than AES/SHA-3
  - No silver bullet - each candidate has some disadvantage
  - Not enough research on quantum algorithms to ensure confidence for some schemes
- We do not expect to select just one algorithm
  - Ideally, several algorithms will emerge as "good choices"
- We will narrow our focus at some point
  - This does not mean algorithms are "out"
- Requirements/timeline could potentially change based on developments in the field

Summary

- Post-quantum crypto standardization will be a long journey
- We have seen many complexities, and know more lie ahead
- Be prepared to transition to new algorithms in 10 years
- We will continue to work in an open and transparent manner with the crypto community for PQC standards
- Check out www.nist.gov/pqcrypto
  - Sign up for the pqc-forum for announcements & discussion

「RSA-2048暗号は、2026年までにはその1/7が破られ、2031年には1/2が破られるだろう」

-- Michele Mosca

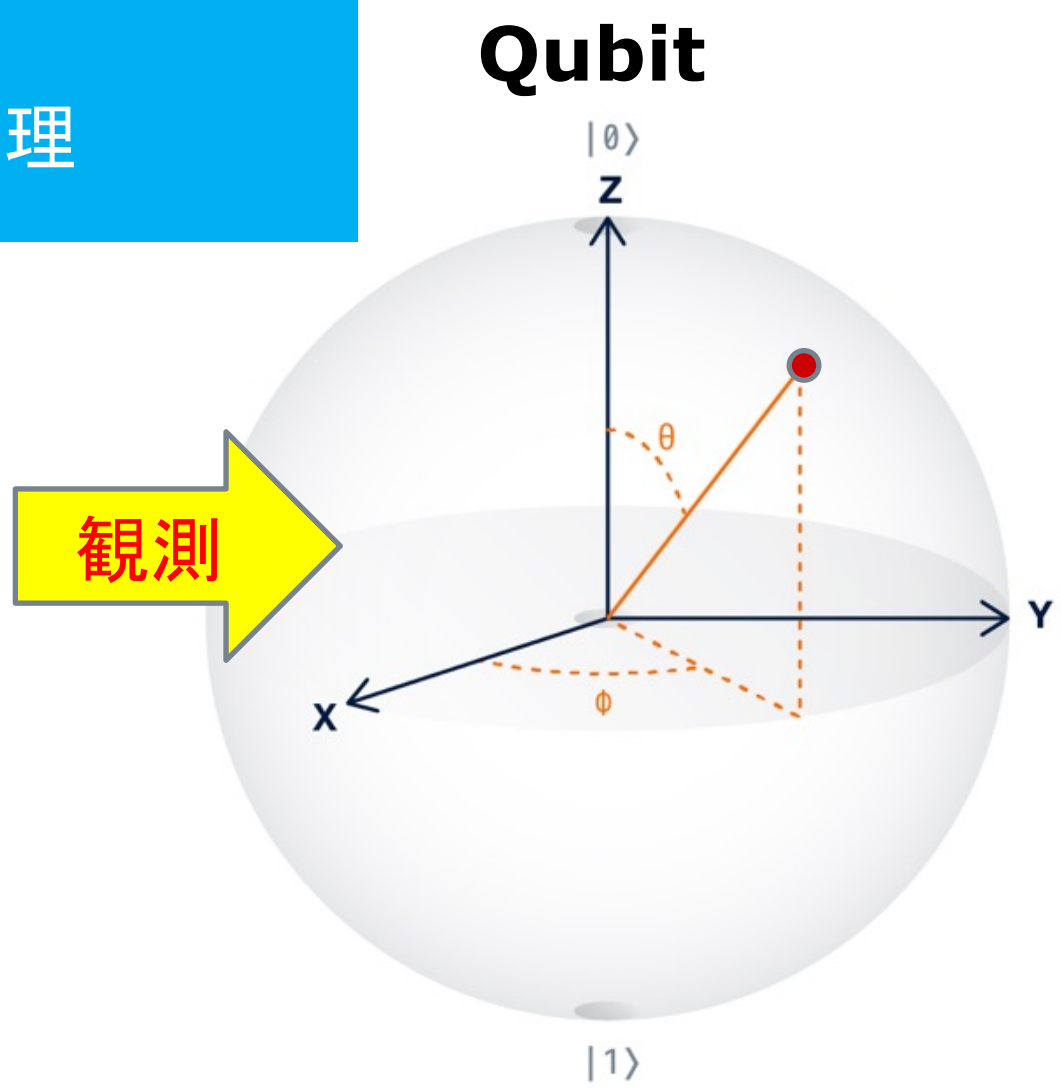
量子暗号

量子の特徴を暗号技術に応用する

暗号技術にとって好ましい量子の二つの性質

観測の原理とNo-Cloning定理

観測の原理

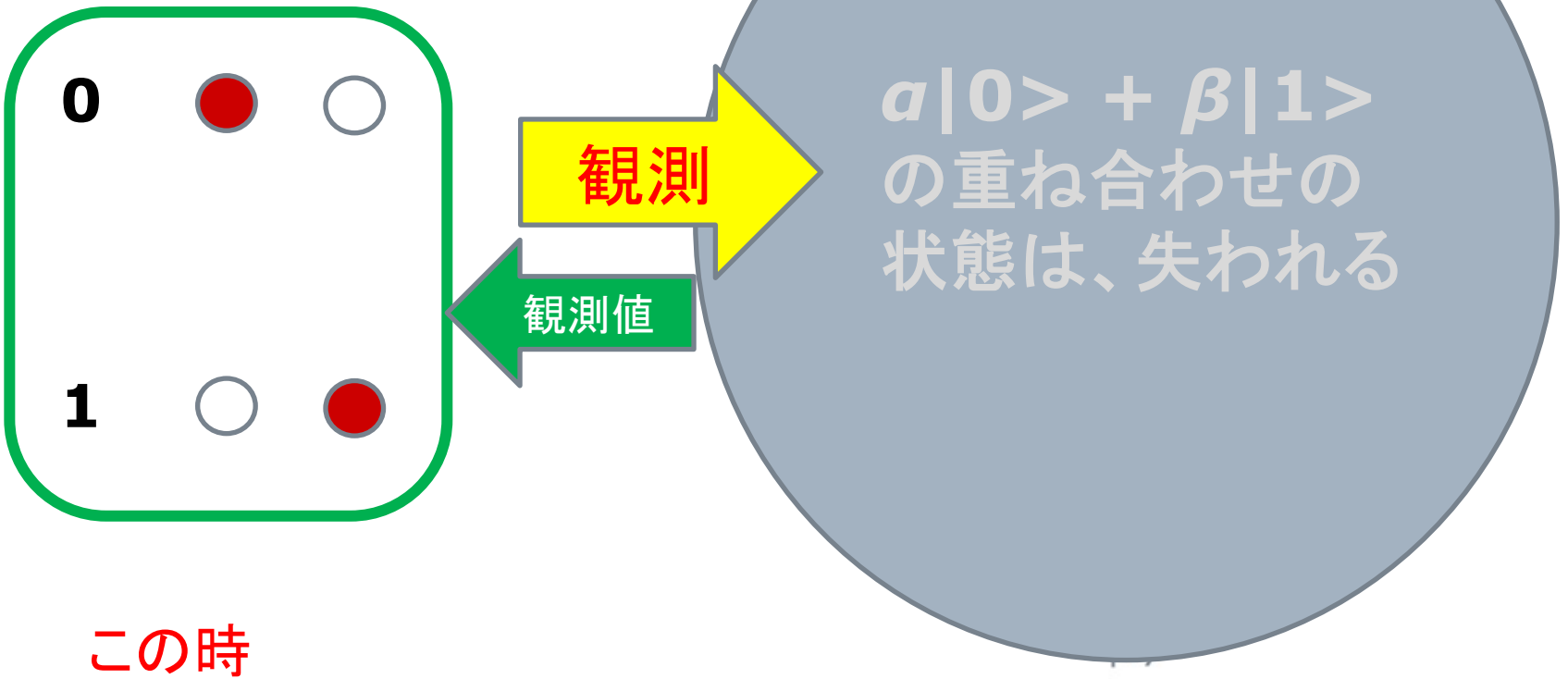


Qubitは、状態 |0>と状態 |1>の重ね合わせの状態を取る

$$|\mathrm{Qubit}\rangle = \alpha|0\rangle + \beta|1\rangle$$

$$|\alpha|^2 + |\beta|^2 = 1\ ;\quad \alpha, \beta \in \mathbb{C}$$

観測を行うと、Qubitの
重ね合わせの状態は
失われ、0か1かの情報
が返る。



この時
0を得る確率は、$|\alpha|^2$で、
1を得る確率は、$|\beta|^2$で、
与えられる。

$\alpha^*\alpha + \beta^*\beta = 1 \ ; \ \ \alpha, \beta \in \mathbb{C}$

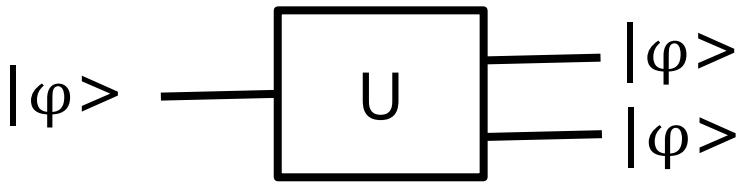
観測と観測による状態の変化

- 量子の状態は、直接には、観測できない。
- また、観測によって、系の状態は、観測前とは異なる状態に変化する。（以前の重ね合わせは失われる。）
  - α|0> + β|1>の重ね合わせの状態は、観測によって|0>または|1>の新しい状態に変化する。（「崩壊する」）
- 新しい状態が観測される確率は、正確に計算することができる。
  - 0を得る確率は、$|\alpha|^2$ で、
    1を得る確率は、$|\beta|^2$ で、与えられる。

No Cloning 定理

ある状態 $|\varphi\rangle=\alpha_0|0\rangle+\alpha_1|1\rangle$に対して、$|\varphi\rangle\otimes|\varphi\rangle$ すなわち自己のコピーをもう一つ生成する回路を考えよう。実は、量子ゲートの世界では、こうしたコピーができないのだ。

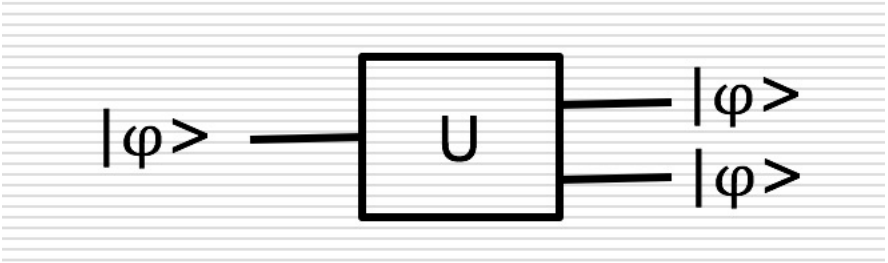
これを、No Cloning 定理と呼ぶ。



こうした回路は存在しない

量子の情報はコピーできない！

No Cloning 定理の証明



先のような回路（ユニタリ変換U）が存在したとする。
その回路は、一般のφだけでなく、|0>, |1> に対しても働くので、

$$U|0\rangle = |00\rangle$$
$$U|1\rangle = |11\rangle$$

$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ とすれば、

$$U|\psi\rangle = \alpha U|0\rangle + \beta U|1\rangle = \alpha|00\rangle + \beta|11\rangle$$

Uは、|ψ>もコピーできるはずなので、

$$U(|\psi\rangle) = |\psi\rangle|\psi\rangle = \alpha^2|00\rangle + \alpha\beta|01\rangle + \alpha\beta|10\rangle + \beta^2|11\rangle$$

しかし、この二つの条件を満たすα, βは、存在しない。

Quantum Key Distribution
BB84

Alice: 古典bitをqubitにエンコードして qubitをBobに送る

- Aliceには、古典bitをqubitにエンコードするのに、次の二つの方法を利用できる。

1. |0>, |1>を基底としてエンコードする。
   この時、古典bit 0 は、qubit |0>に、
   古典bit 1 は、qubit |1>にエンコードされる。
2. |+>, |−>を基底としてエンコードする。
   この時、古典bit 0 は、qubit |+>に、
   古典bit 1 は、qubit |−>にエンコードされる。

- 基底 |0>, |1> と基底 |+>, |−>の関係については、あとで詳しく述べる。

Bob: Aliceからqubitを受け取り、観測して古典bitにデコードする

- Aliceには、qubitを古典bitにデコードするのに、次の二つの方法を利用できる。

1. |0>, |1>を基底としてqubitを観測し、デコードする。
   この時、qubit |0> は、古典bit 0 に、
   qubit |1> は、古典bit 1 にデコードされる。
2. |+>, |−>を基底としてqubitを観測し、デコードする。
   この時、qubit |+> は、古典bit 0に、
   qubit |−> は、古典bit 1にデコードされる。

基底 |0>, |1> と基底 |+>, |−>の関係

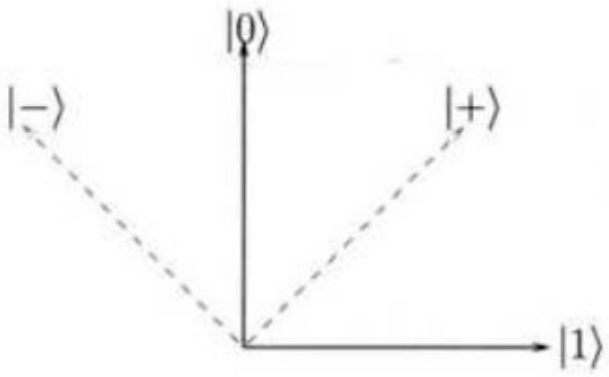
- 基底 |0>, |1> と基底 |+>, |−>の間には、つぎのような関係がある。

1. $|0\rangle = \frac{1}{\sqrt{2}}|+\rangle + \frac{1}{\sqrt{2}}|-\rangle$

   $|1\rangle = \frac{1}{\sqrt{2}}|+\rangle - \frac{1}{\sqrt{2}}|-\rangle$

2. $|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$

   $|-\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$



基底 |0>, |1>を「標準基底」、
基底 |+>, |−>を「アダマール基底」と呼ぶ。

Aliceがエンコードに選んだ基底と Bobがデコードに選んだ基底が同じ場合

- AliceとBobが共に「標準基底」を選んだ場合
  - 古典bitが0の場合、AliceはBobにqubit |0>を送る。
    |0> =1・|0>+0・|1>だから、Bobは100%の確率で|0>を観測するので、0にデコードできる。
  - 古典bitが1の場合、AliceはBobにqubit |1>を送る。
    |1> =0・|0>+1・|1>だから、Bobは100%の確率で|1>を観測するので、1にデコードできる。
- AliceとBobが共に「アダマール基底」を選んだ場合
  - 古典bitが0の場合、AliceはBobにqubit |+>を送る。
    |+> =1・|+>+0・|−>だから、Bobは100%の確率で|+>を観測するので、0にデコードできる。
  - 古典bitが1の場合、AliceはBobにqubit |−>を送る。
    |−> =0・|+>+1・ |−>だから、Bobは100%の確率で|−>を観測するので、1にデコードできる。
- 同じ基底を選んだ時には、正しく情報を伝えられる。

Aliceがエンコードに選んだ基底と Bobがデコードに選んだ基底が異なる場合

- Aliceが「標準基底」、Bobが「アダマール基底」を選んだ場合
  - 古典bitが0の場合、AliceはBobにqubit |0>を送る。
    $|0> = \frac{1}{\sqrt{2}}|+> + \frac{1}{\sqrt{2}}|->$だから、Bobは 1/2の確率で|+>を、同じく 1/2 の確率で|−>を観測するので、0か1か決まらない。
  - 古典bitが1の場合、AliceはBobにqubit |1>を送る。
    $|1> = \frac{1}{\sqrt{2}}|+> - \frac{1}{\sqrt{2}}|->$だから、1/2の確率で|+>を、同じく 1/2 の確率で|−>を観測するので、0か1か決まらない。
- Aliceが「アダマール基底」、Bobが「標準基底」を選んだ場合
  - 古典bitが0の場合、AliceはBobにqubit |+>を送る。
    $|+> = |+> = \frac{1}{\sqrt{2}}|0> + \frac{1}{\sqrt{2}}|1>$だから、 Bobは 1/2の確率で|0>を、同じく 1/2 の確率で|1>を観測するので、0か1か決まらない。
  - 古典bitが1の場合も同様である。
- 異なる基底を選んだ時には、正しく情報が伝えられない。

AliceとBobで、秘密鍵を共有する

- Aliceは、ランダムに 0, 1のビット列を作っておく。
- Aliceは、ランダムに「標準基底」「アダマール基底」を選んで、生成したビット列を1ビットごとにqubitにエンコードしてBobに送る。
- Bobは、ランダムに「標準基底」「アダマール基底」を選んで、送られてきたqubitをビットにデコードする。
- 作業が終わったら、AliceとBobは、それぞれどの基底をエンコード、デコードに使ったかを情報交換する。
- AliceとBobが、エンコードとデコードで同じ基底を使ったビットのみを抜き出して、それを秘密鍵とする。
- AliceとBobが、どのビットで同じ基底を使ったという情報は、第三者に漏れても構わない。同じ基底を使ったというだけでは、そのビットが0だったのか1だったかはわからないからである。

エンコード、デコードの例

Aliceが最初に用意したビット列

Bit	x_1	x_2	x_3	x_4
Value	0	1	1	0

＋：標準基底
Ｘ：アダマール基底

Aliceが選んだ基底とエンコードされたqubit

Classical value	0	1	1	0
Alice's basis	+	×	+	×
Quantum encoding	$\|\psi_1\rangle = \|0\rangle$	$\|\psi_2\rangle = \|-\rangle$	$\|\psi_3\rangle = \|1\rangle$	$\|\psi_4\rangle = \|+\rangle$

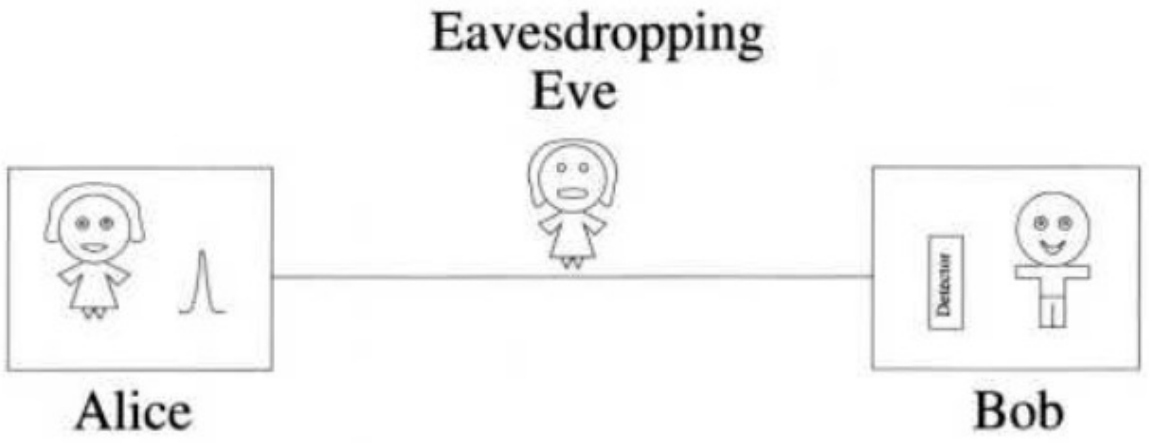
Aliceが選んだ基底とBobの選んだ基底は一致しているか？

Classical value	0	1	1	0
Alice's basis	+	×	+	×
Bob's basis	×	×	+	+
In agreement	No	Yes	Yes	No

両者の基底が一致したビットだけ、この例では、'1', '1' を、両者で共有する。

中間に盗聴者Eveがいた場合

この方式は、AliceとBobの通信の中間に盗聴者がいたとしても、AliceとBobは、そのことに確率的に気づくことができる。以下、それを見よう。



中間に盗聴者Eveがいた場合

- Eveが、AliceとBobが共有する秘密キーの情報を獲得する確率を考える。

1. 共有されるビットは、AliceとBobが同じ基底を使ったビットである。ただ、Eveは、その基底を知らないので、あてずっぽうに二つの基底から一つを選ぶしかない。この時点で、Eveは50%は推測に成功するが、50%は失敗する。
2. Eveは、基底を知らないだけでなく、Aliceが送ったビットが、0なのか1なのかを知らない。ここでも、彼は50%は失敗する。
3. 結局、EveがAliceとBobの基底を正しく推測し、Aliceが送ったビットを正しくBobに送るのに成功する確率は、0.5 x 0.5 =25%にとどまる。

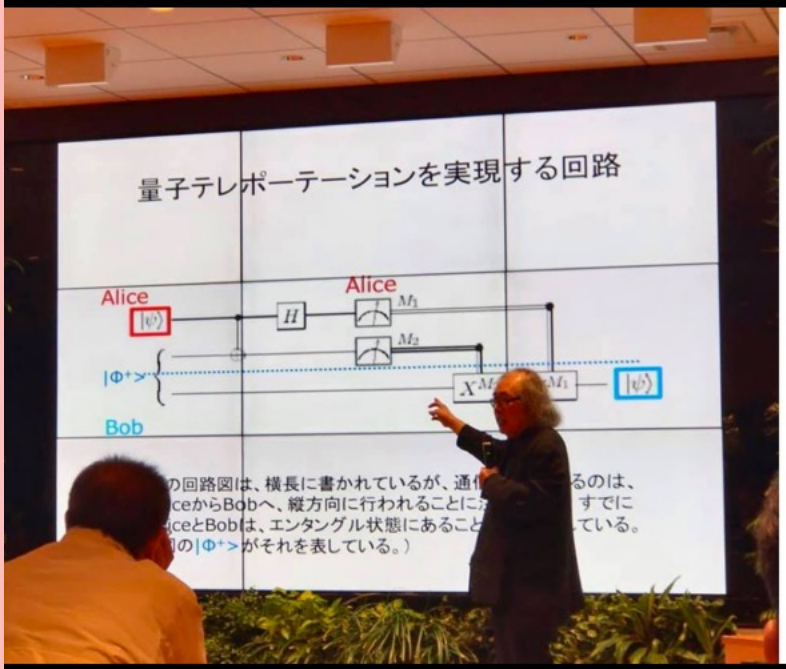
- AliceとBobが、いくつかの秘密キーのビットをチェックすれば、高い確率で、盗聴が行われたことを検知できる。

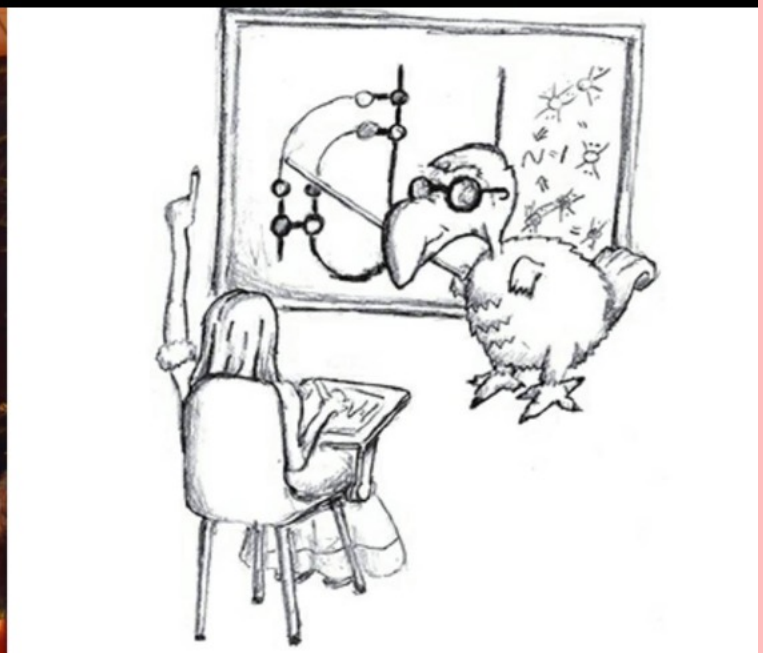
古典量子ハイブリッドの時代を展望する

現在のコンピュータと量子コンピュータのハイブリッド
現在の通信技術と量子通信のハイブリッド

参考資料（マルゼミ 2019/04/12）

紙と鉛筆で学ぶ
量子アルゴリズム演習1





エンタングルメントと
量子テレポーテーションを学ぶ

概要 https://lab-kadokawa79.peatix.com/ 資http://bit.ly/2UPIDGR

参考資料(マルレク 2019/03/25)

概要 https://easyq.peatix.com 資料 https://goo.gl/UEmoSL

来月7/6 ハンズオン予告

「実際の量子コンピュータで学ぶ量子プログラミング入門」

今年に入って、IBMさんが量子コンピュータの商用サービスを開始したことをご存知の方は多いと思います。

同時に、IBMさんが、無償で実際の量子コンピュータにアクセスできるサービスを始めていることは、画期的だと思います。

このハンズオンは、この「実際の量子コンピュータ」を使って、量子プログラミングの初歩を学びます。

Appendix

量子通貨

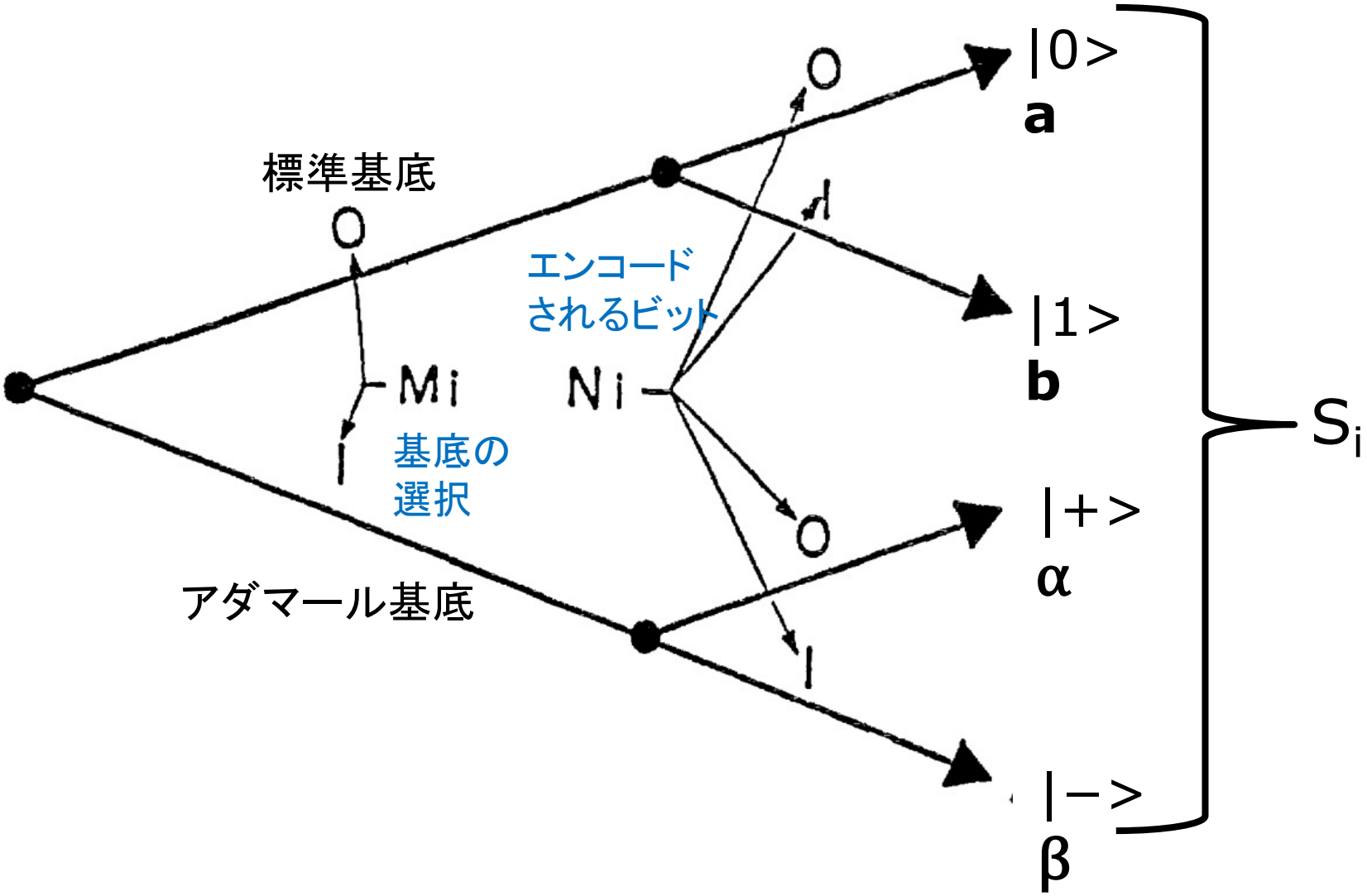
量子通貨の最初のアイデア
1970年 Stephan Wiesner

- 量子通貨の最初のアイデアを出したのは、Stephan Wiesner である。1970年のことである。
  “Conjugate Coding”
  https://dl.acm.org/citation.cfm?id=1008920
- ただ、当時は、彼のアイデアは評価されず、この論文が受理され発表されたのは、1983年になってからであった。
- 彼のこの論文は、先に見た量子暗号 BB84の原理を先取りしたものとしても有名である。

秘密キー量子通貨

量子通貨の最初のアイデア

- 貨幣造幣局は、一つの紙幣に対して、ランダムに生成されたnビットの列 $\{M_i\}_n$, $\{N_i\}_n$ を用意する。
- M_iは、エンコードする基底の選択に対応し、N_iはエンコードされるビットに対応する。
- M_i＝0の時、標準基底で N_iをエンコードし、
  M_i＝1の時、アダマール基底で N_iをエンコードする。
- こうして$\{M_i\}_n$, $\{N_i\}_n$ に対応して生成されたn個のqubit $\{S_i\}_n$を貨幣に封入する。


標準基底
O
Mi
i
基底の
選択
エンコード
されるビット
Ni
O
i
O
i
アダマール基底
|0>
a
|1>
b
|+>
α
|−>
β
Si

- 貨幣造幣局は、持ち込まれた貨幣を、その貨幣に対応づけられた基底 $\{M_i\}_n$ を用いてデコードして、それが$\{N_i\}_n$の通りかチェックすることができる。
- 貨幣偽造者が、 $\{S_i\}_n$ をチェックして、それと同じもの$\{S'_i\}_n$を作ろうとしたとしよう。
- 偽造者は、 $\{M_i\}_n$ を知らないので、特定のS_iをチェックするのに正しい基底を用いる確率は、1/2である。
- 偽造者は、 $\{N_i\}_n$ も知らないので、先に選んだ基底で$\{S'_i\}_n$を作ろうとして、正しい値を用いる可能性は、1/2である。
- 結局、偽造者がS_iから正しくS'_i をコピーする確率は、1/4となり、失敗する確率は、3/4となる。
- 20個のqubitからなる量子貨幣なら、真贋のチェックをパスする可能性は、$(3/4)^{20} < 0.00317$ と極めて低くなる。

Wiesner量子通貨の欠点

- Wiesner量子通貨は、造幣局のみが量子通貨の秘密キーを保有する方式である。次のような欠点がある。

1. この方式では、通貨の真贋を判定できるのは、造幣局のみである。通貨の所有者は、その真贋を判定できない。
2. この方式では、造幣局は、発行する通貨ごとに、その秘密キーを管理しなければならないのだが、発行する通貨が増えるにつれて、その秘密キーデータベースは膨大なものになる。

- 量子通貨を実現する上で最大の問題は、n個のqubitの状態を、通貨の中で維持し続けることである。ただし、ここでは、それが「可能である」という前提で、議論を行っている。

BBBW量子通貨

- Bennett, Brassard, Breidbart, Wiesner によって1982年に提案された方式。先のWiesner量子通貨の、造幣局が発行するすべての通貨の秘密キーデータベースを持つ必要を無くそうというアイデア。
- 造幣局は、$f_k : \{0,1\}^n \rightarrow \{0,1\}^{2n}$ という、pseudorandom function f_kと秘密キーkを一つ持つ。
- 通貨のIDをsとすれば、$f_k(s)$の値は、$|0\rangle, |1\rangle, |+\rangle, |-\rangle$の四つの記号のランダムなn個の並びを返すものと解釈できる。

公開キー量子通貨

公開キー量子通貨

- 公開キー量子通貨は、2009年、Lutomirski, Aaronson, Farhiらによって初めて提案された。

  “Breaking and making quantum money: toward a new quantum cryptographic protocol”
  https://arxiv.org/abs/0912.3825

「公開キー量子通貨方式」とは何か？

- 「公開キー量子通貨方式」は、次の三つの条件を満たすものである。

1. 銀行は通貨を発行できる。
   すなわち、量子通貨の状態を生成する効率的なアルゴリズムが存在すること。
2. 誰もがその真贋をチェックできる。
   通貨を受け取ったものは、誰でも、通貨にダメージを与えることなく、高い確率で通貨の真贋をチェックすることができる効率的なアルゴリズムが存在すること。
3. 誰も（銀行を除いて）、それをコピーできないこと。
   すなわち、銀行以外の誰もが、鑑定者に受け入れられる状態を作り出すのは、指数関数的に小さな確率しかないこと。

Quantum Copy-Protection and Quantum Money

Scott Aaronson

https://www.scottaaronson.com/papers/noclone-ccc.pdf

This paper tackles both questions using the arsenal of modern computational complexity. Our main result is that there exist quantum oracles relative to which publicly-verifiable quantum money is possible, and any family of functions that cannot be efficiently learned from its input-output behavior can be quantumly copy-protected. This provides the first formal evidence that these tasks are achievable. The technical core of our result is a "Complexity-Theoretic No-Cloning Theorem," which generalizes both the standard No-Cloning Theorem and the optimality of Grover search, and might be of independent interest. Our security argument also requires explicit constructions of quantum t-designs.