Ordinateurs quantiques et futur de la sécurité

Renaud Lifchitz, Econocom digital.security

Lundi de la cybersécurité, 15 avril 2019

Outline (1/2)

Outline (2/2)

Calcul quantique & cryptographie

- 2 approches pour inverser une fonction
- Exemple : CRC-8
- Inverser un chiffrement XOR par oracle
- Menaces quantiques contre la cryptographie actuelle

Cryptographie post-quantique

Présentation de l'auteur

- Expert en sécurité français
  @ Econocom digital.security
- Principales activités:
  - Tests d'intrusion & audits de sécurité
  - Recherche
  - Formations & sensibilisations
- Centres d'intérêt :
  - Securité des protocoles (authentification, cryptographie, fuites d'information, preuves à divulgation nulle de connaissance...)
  - Théorie des nombres (factorisation, tests de primalité, courbes elliptiques...)

Principes du calcul quantique

Principes

Principes quantiques (1/2)

1. Les petites entités physiques (atomes, molécules, photons, électrons, ...) se comportent à la fois comme des particules et des ondes (principe de dualité)
2. Les principales caractéristiques de ces objets (position, spin, polarisation, ...) ne sont pas déterminés à leur création, et ont plusieurs valeurs simultanément (superposition quantique / principe d'incertitude d'Heisenberg)
3. Une interaction ou mesure va faire s'effondrer cette distribution en un endroit et un état stable (décohérence quantique)
4. Par conséquent, copier un état quantique n'est pas possible (théorème de non-clonage)

- Il est malgré tout possible de tirer parti des 3 premiers principes pour calculer très efficacement !

Principes quantiques (2/2)

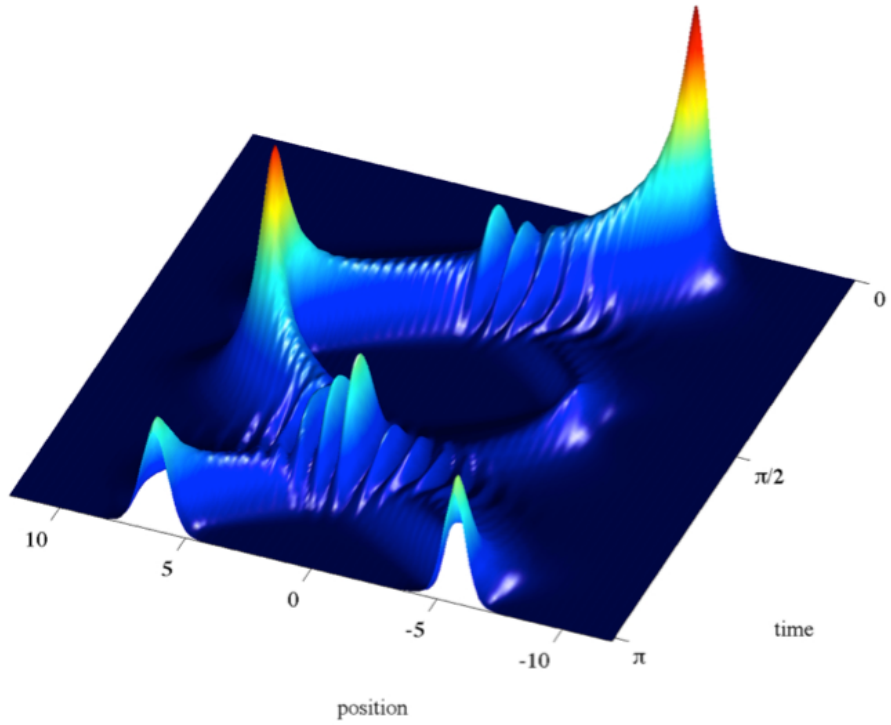


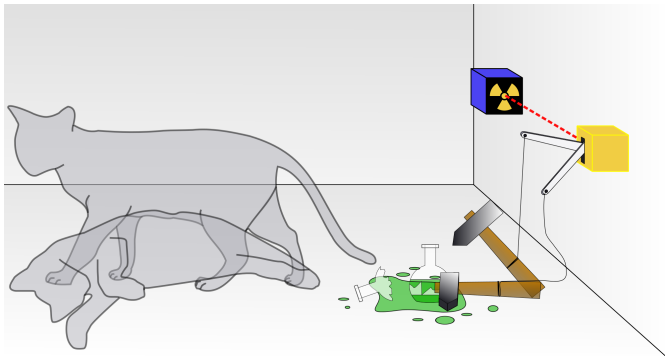
Figure : Position of an atom under quantum conditions across time, sometimes it is 100% determined, sometimes 50% - *Image created by Thomas Fogarty, graduate student from University College Cork in Ireland*

Expériences récentes

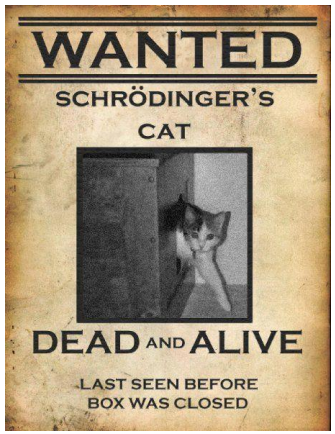
- Interaction instantanée de qubits intriqués - Paradoxe EPR: Eté 2008, Université de Genève, Nicolas Gisin et ses collègues déterminent que la vitesse d'une interaction quantique est d'au moins 10000 fois la vitesse de la lumière en utilisant des photons corrélés à 18 kms de distance (`http://arxiv.org/abs/0808.3316`)

- Quantum teleportation: Septembre 2014, la même équipe réussit une téléportation quantique de 25 kilomètres

Expérience du chat de Schrödinger



- Paradoxe, à travers une expérience de pensée, conçu par le physicien autrichien Erwin Schrödinger en 1935
- Un chat, une bouteille de poison, une source radioactive, et un détecteur de radioactiviité sont placés dans une boîte fermée
- Si le détecteur détecte de la radioactivité, la bouteille est cassée, tuant le chat
- Jusqu'à ouverture de la boîte, le chat est mort **ET** vivant !



Systèmes quantiques commerciaux

- Génaration quantique de nombres aléatoires :
  - ID Quantique “Quantis” génère 4 Mbits/s à 16 MBits/s de nombres aléatoires avec aléa quantique :

USB


PCI Express


PCI

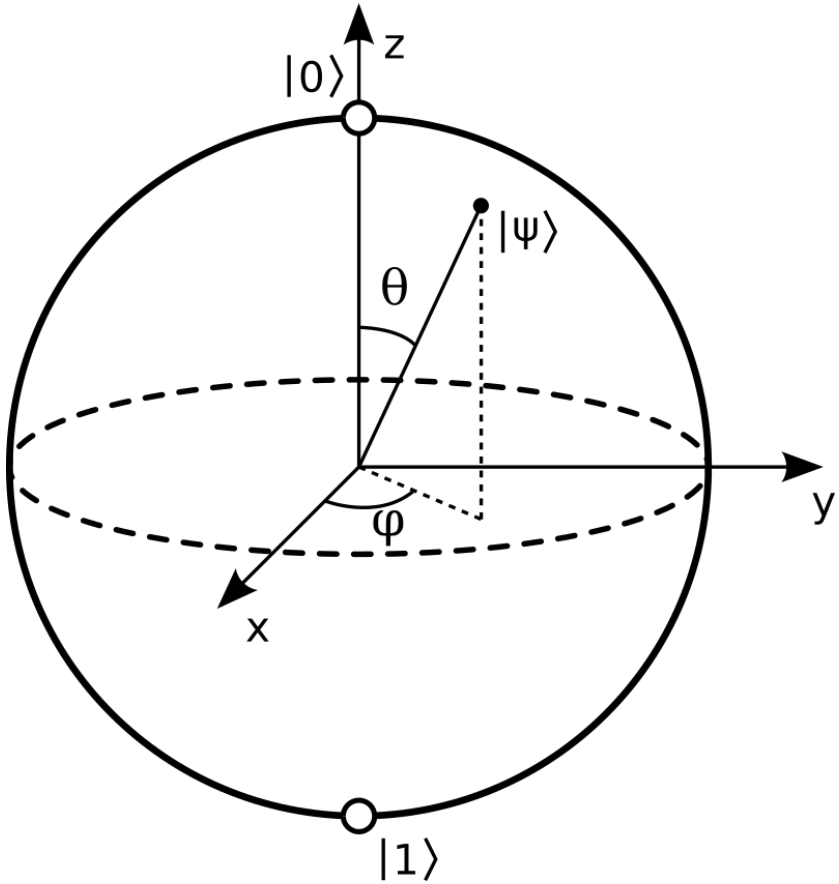
  - Service en ligne “Quantum Random Bit Generator” (QRBG121) : `http://random.irb.hr/`

- Système de chiffrement quantique :
  - ID Quantique “Cerberis” & “Centauris” permettent la distribution de clé quantique (QKD) et du chiffrement jusqu’à 100 Gbps et 100 km :

Représentation des qubits (1/2)

- Les qubits constants 0 and 1 sont représentés $|0\rangle$ et $|1\rangle$
- Ils forment une base à 2 dimensions, e.g. $|0\rangle = \begin{bmatrix}1\\0\end{bmatrix}$ et $|1\rangle = \begin{bmatrix}0\\1\end{bmatrix}$
- Un qubit arbitraire q est une combinaison linéaire des deux états de base : $|q\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{bmatrix}\alpha\\\beta\end{bmatrix}$ where $\alpha \in \mathbb{C}$, $\beta \in \mathbb{C}$
- Quand q est mesuré, la probabilité de mesurer $|0\rangle$ est $|\alpha|^2$ donc $|\alpha|^2 + |\beta|^2 = 1$
- Une combinaison de qubits forme un registre quantique et peut être calculée comme un produit tensoriel : $|10\rangle = |1\rangle \otimes |0\rangle = \begin{bmatrix}0\\0\\1\\0\end{bmatrix}$

Représentation des qubits (2/2)



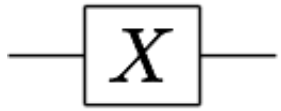
Sphère de Bloch : un qubit peut-être vu comme un vecteur unitaire dans une sphère

Quelques notions sur les portes quantiques

- Pour des raisons thermodynamiques, une porte quantique doit être réversible
- Par conséquent, une porte doit avoir autant d'entrée que de sorties
- Une porte à n qubits peut être représentée par une matrice unitaire 2^nx2^n
- Appliquer une porte quantique à un registre de qubits revient à multiplier à gauche son vecteur par la matrice de la porte
- Une combinaison de portes quantiques peut être calculée par le produit de leur matrice
- En théorie, un circuit quantique n'utilise ou ne dégage aucune énergie ou chaleur

Portes quantiques simples

Porte Pauli-X

Porte Pauli-X	**Nombre de qubits**: 1	**Symbole**: 

Description: Equivalent quantique d'une porte logique "NON". Tourne le qubit autour de son axe X de 180 degrés. $X.X = I$.

Matrice: $X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$

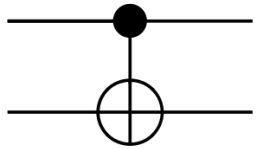
Porte Hadamard

Porte Hadamard	**Nombre de qubits**: 1	**Symbole**: 
Description: Transforme un qubit constant dans une superposition équiprobable de $\|0\rangle$ et $\|1\rangle$.		
Matrice: $H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$		

Porte d'Hadamard

- Pour cette raison, elle est beaucoup utilisée comme première étape d'un algorithme pour travailler en parallèle sur toutes les valeurs possibles des entrées

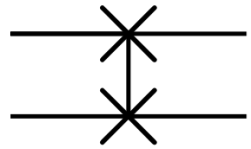
Porte CNOT

Porte CNOT	**Nombre de qubits**: 2	**Symbole**: 

Description: Porte "NON" contrôlée. Le premier qubit est le qubit de contrôle, le second est le qubit cible. Ne change pas le qubit de contrôle et change le qubit cible uniquement si le contrôle vaut $|1\rangle$.

Matrice: $CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

Porte SWAP

Porte SWAP	**Nombre de qubits**: 2	**Symbole**: 
Description: Echange les 2 qubits d'entrée.		
Matrice: $SWAP = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$		

Jeu universel de portes

Un jeu de portes quantiques est appelé **universel** si tous les circuits peuvent être créés avec seulement ces portes :

- En toute généralité : porte d'Hadamard, porte de changement de phase (avec $\theta = \frac{\Pi}{4}$ et $\theta = \frac{\Pi}{2}$) et porte CNOT
- Pour des circuits de logique booléenne :
  porte de Toffoli uniquement

Challenges

Challenges (1/2)

- Qubits et registres de qubits ne peuvent jamais être indépendamment copiés
- En simulation comme en vrai, le nombre de qubits utilisés doit être limité (réutilisation de qubits quand c'est possible)
- Les "shifts" de registres sont coûteux, déplacer les “têtes de lecture” est plus simple
- En pratique, des codes correcteurs d'erreurs quantiques sont souvent utilisés pour le calcul
  (qubits physiques $\neq$ qubits logiques)

Challenges (2/2)

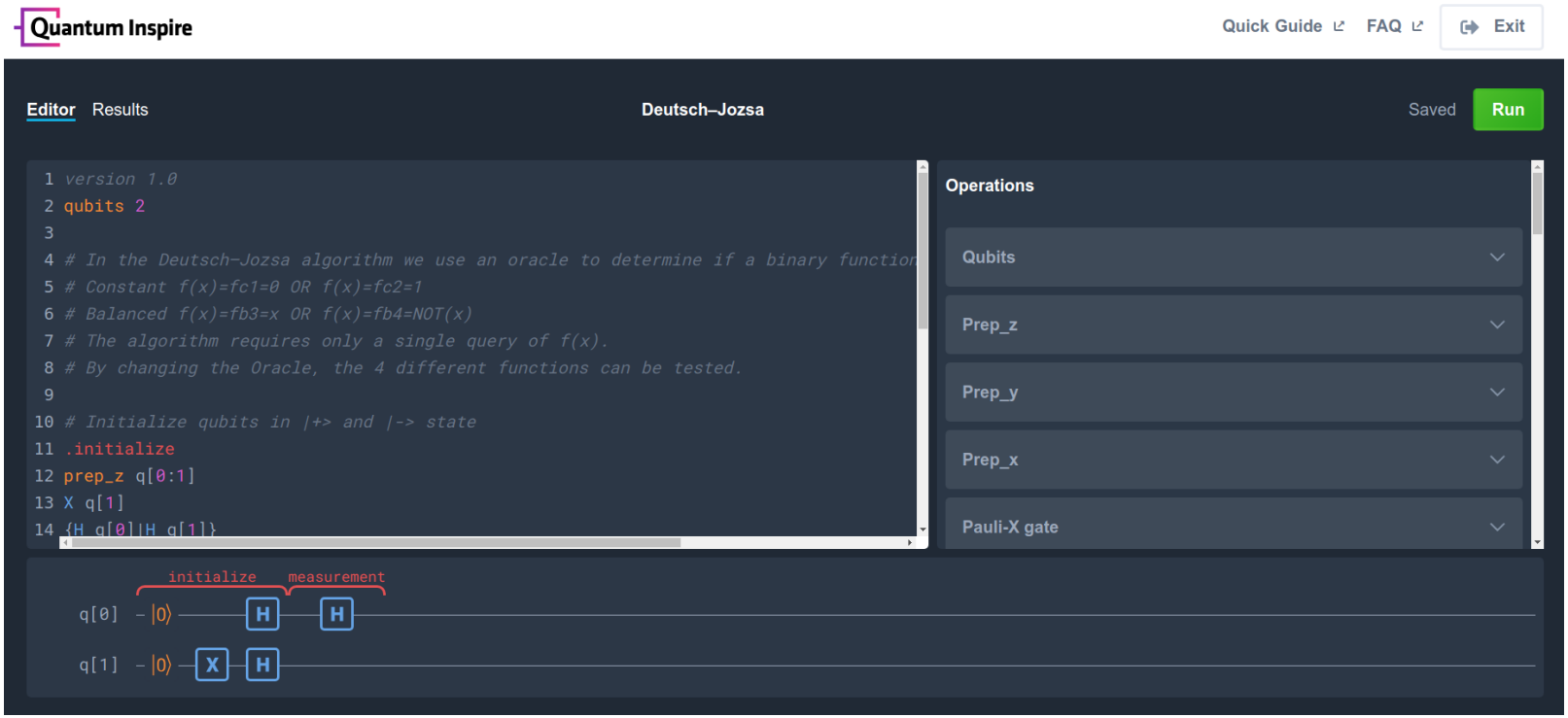
Pour des besoins sérieux :

- Un nombre élevé de qubits
  (environ 50 qubits sont suffisants pour la "suprématie quantique")
- Une bonne fidélité de qubit et de porte (faible taux d'erreur)
- Optionnellement, de la correction d'erreurs

Un grand nombre de qubits n'est pas le plus important, la plupart des algorithmes actuels sont limités par la profondeur de circuit ($\approx$ 20-30 portes) à cause des fidélités imparfaites.

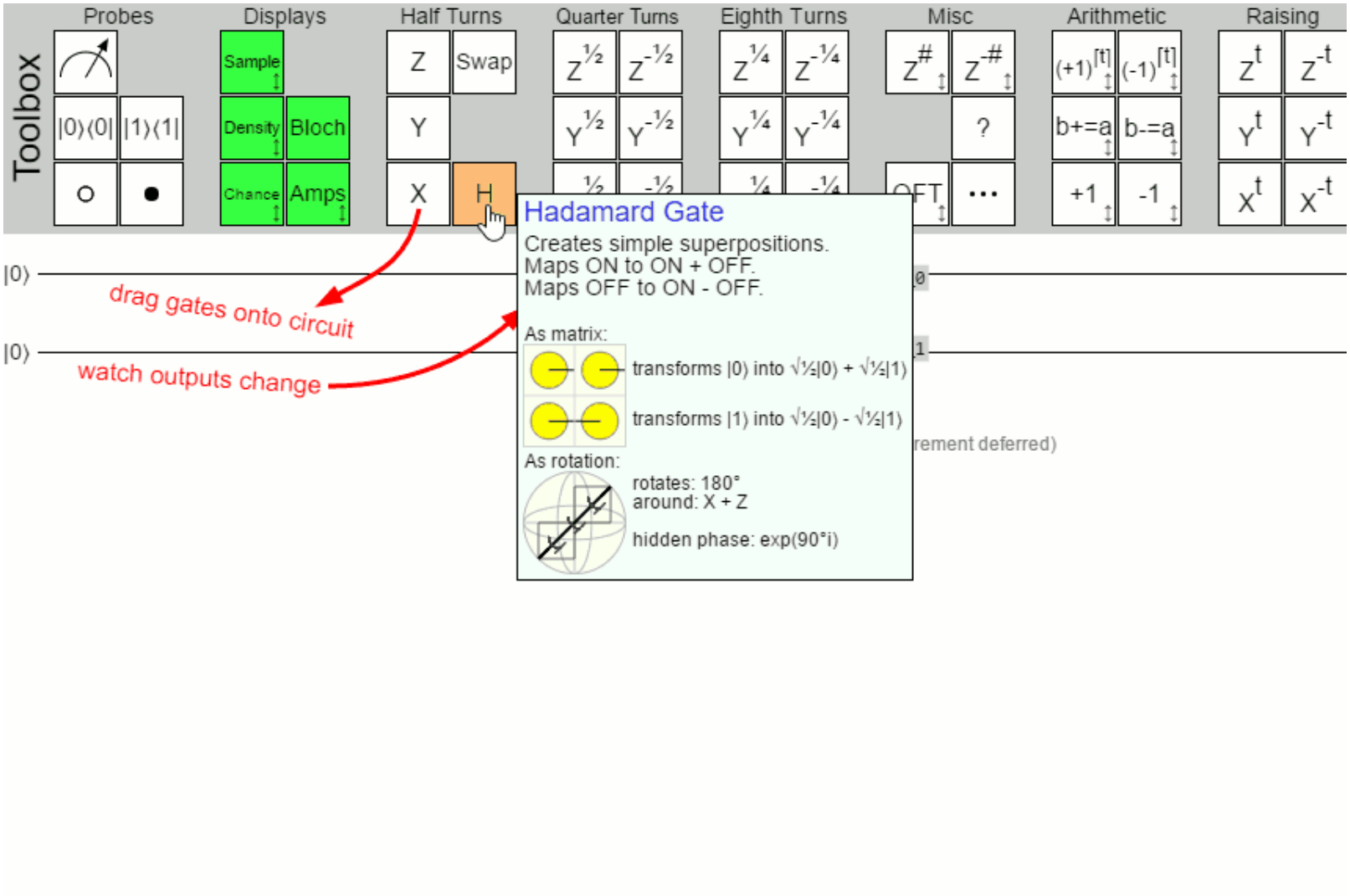
Simulateurs quantiques accessibles

Quantum Inspire



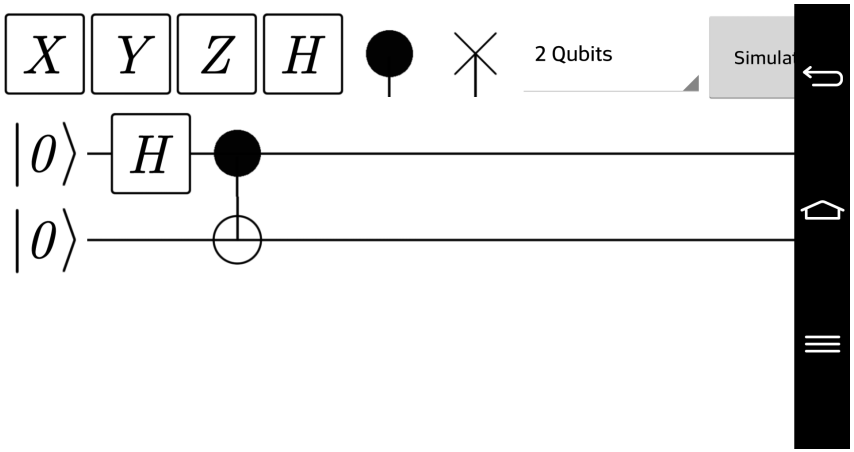
https://www.quantum-inspire.com/

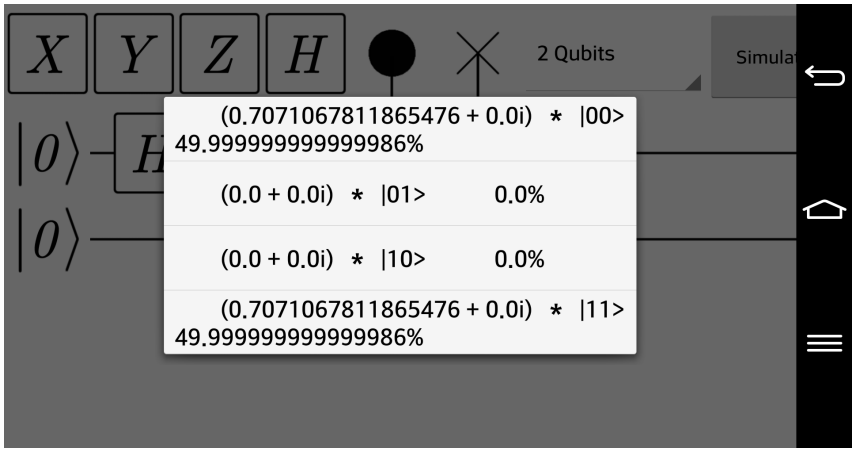
Quirk



http://algassert.com/quirk

Quantum Circuit Simulator (Android)





Conception et simulation d'une intrication quantique de qubits

`https://play.google.com/store/apps/details?id=mert.qcs`

Autres simulateurs quantiques

Une liste plus exhaustive :
`https://quantiki.org/wiki/list-qc-simulators`

Aperçu des puces quantiques accessibles sur le cloud

Public quantum cloud computing services

- Bristol University "Quantum in the Cloud" (`http://www.bristol.ac.uk/physics/research/quantum/engagement/qcloud/`): jusqu'à 2-3 qubits
- Alibaba Quantum Computing Cloud Service (`http://quantumcomputer.ac.cn`): jusqu'à 11 qubits
- IBM "Q Experience" (`https://www.research.ibm.com/ibm-q/technology/devices/`): jusqu'à 14 qubits, 20 qubits pour les clients privés
- Rigetti "Quantum Cloud Services" (`https://www.rigetti.com/qpu`): jusqu'à 19 qubits, 128 qubits à venir
- D-Wave "Leap" (`https://cloud.dwavesys.com/leap/`): jusqu'à 1000 qubits, puce quantique adiabatique, non universel, principalement pour des problèmes d'optimisation

Calcul quantique & cryptographie

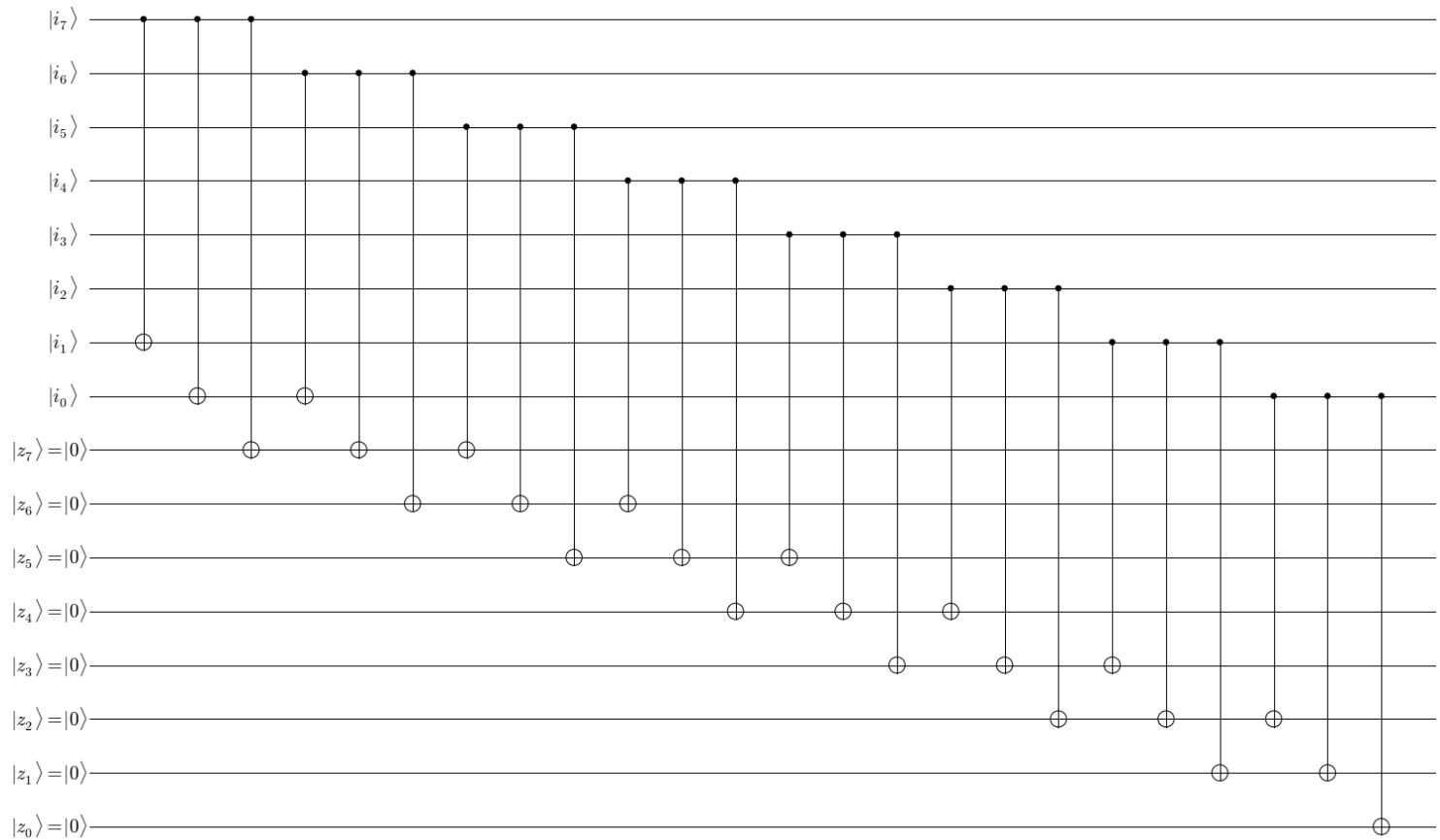
2 approches pour inverser une fonction

2 approches pour inverser une fonction

- Implémenter un circuit et l'exécuter en sens inverse.
  Problèmes :
  - La fonction est souvent non réversible.
    Solution : "envelopper" la fonction (Ajouter des bits d'entrée et de sortie et d'autres techniques simples pour la rendre réversible)
  - Les qubits temporaires sont souvent nombreux
    (mais efficace s'ils sont minoritaires)
- Oracle de Grover: implémenter la fonction directement et demander à un oracle de Grover d'amplifier puis de trouver la valeur recherchée parmi toutes les possibilités!

Exemple : CRC-8

CRC-8 : une implémentation naïve



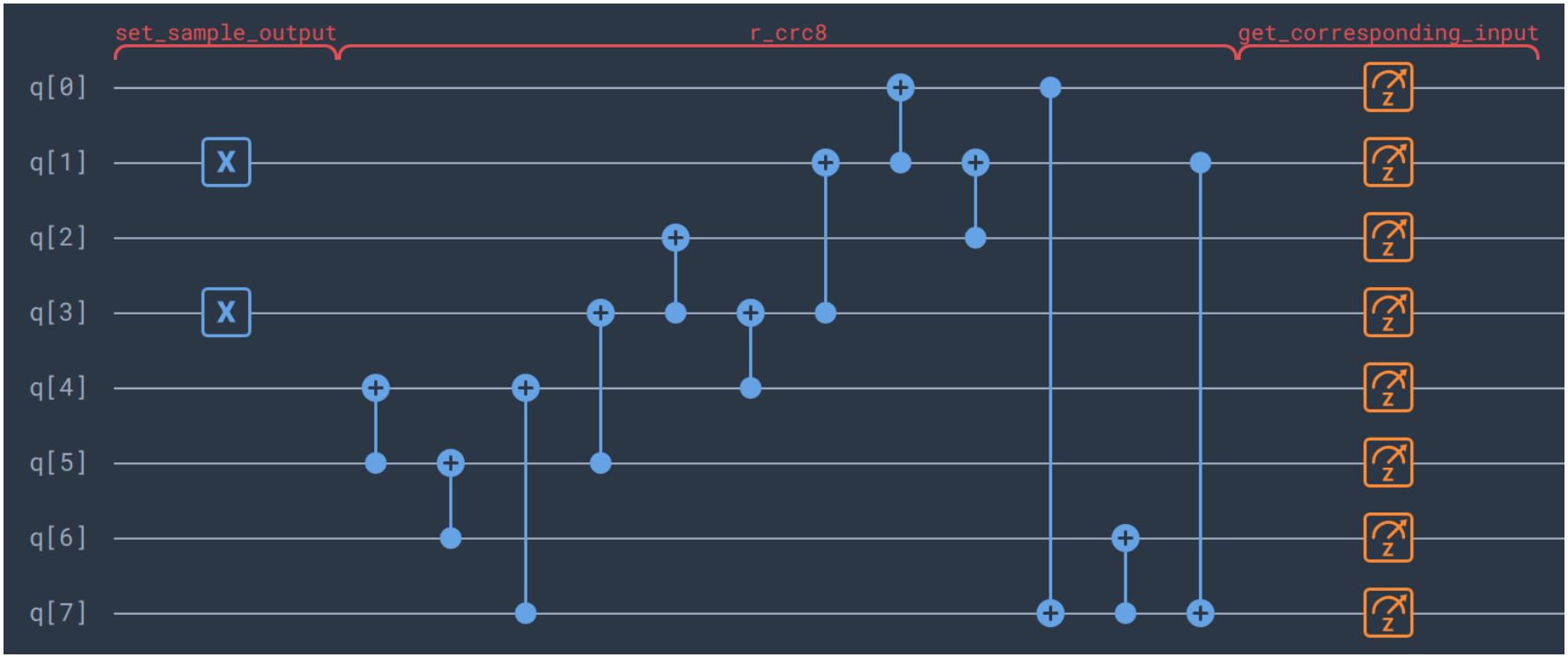
Un circuit CRC-8 quantique avec uniquement des portes CNOT

Implémentation inverse optimale de CRC-8 (1/2)

```
version 1.0
qubits 8
error_model depolarizing_channel, 0.001
.set_sample_output
{X q[1]|X q[3]}
.R_CRC8
CNOT q[5],q[4]
CNOT q[6],q[5]
CNOT q[7],q[4]
CNOT q[5],q[3]
CNOT q[3],q[2]
CNOT q[4],q[3]
CNOT q[3],q[1]
CNOT q[1],q[0]
CNOT q[2],q[1]
CNOT q[0],q[7]
CNOT q[7],q[6]
CNOT q[1],q[7]
.get_corresponding_input
Measure_all
```

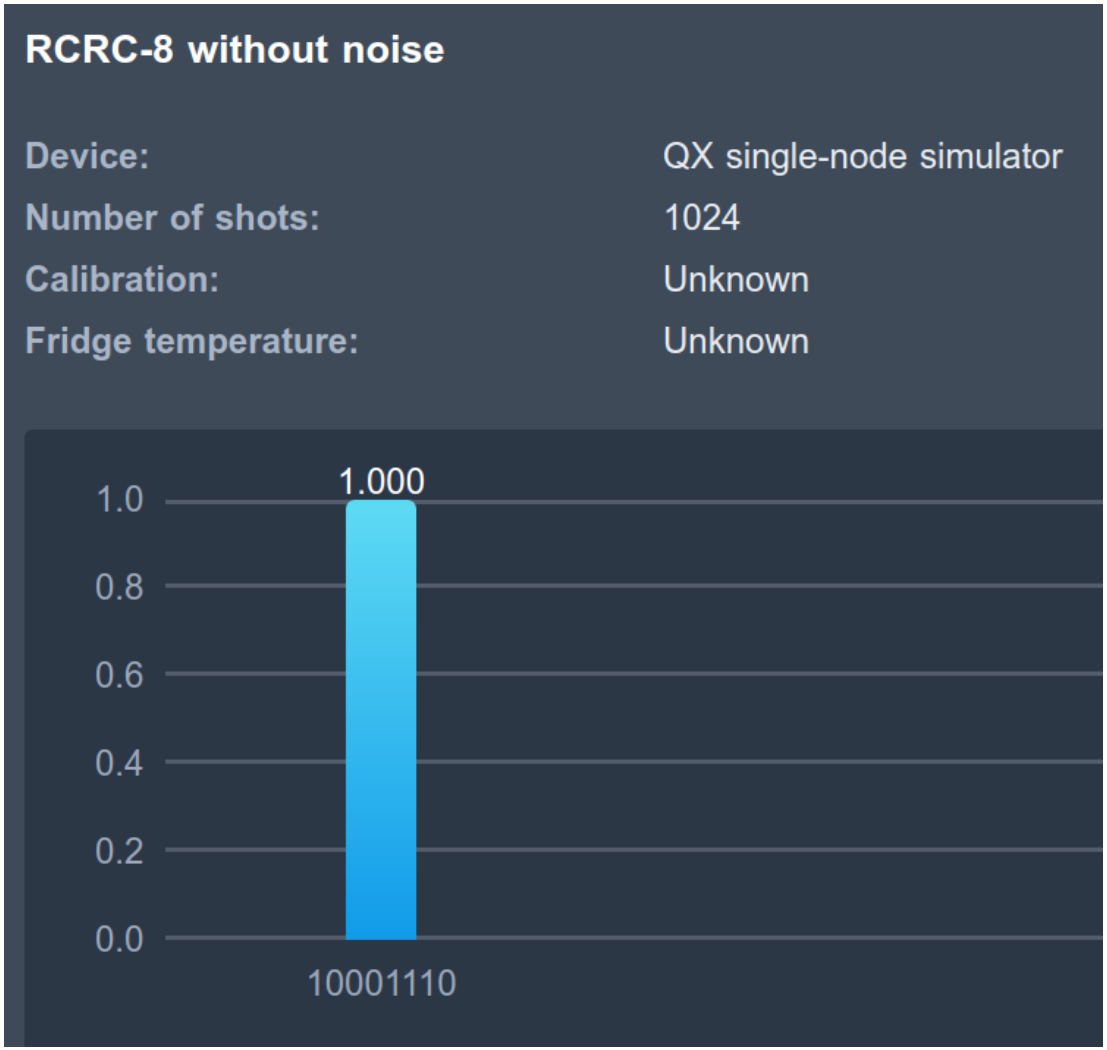
Notre circuit optimal inverse de CRC-8
avec Quantum Inspire

Implémentation inverse optimale de CRC-8 (2/2)



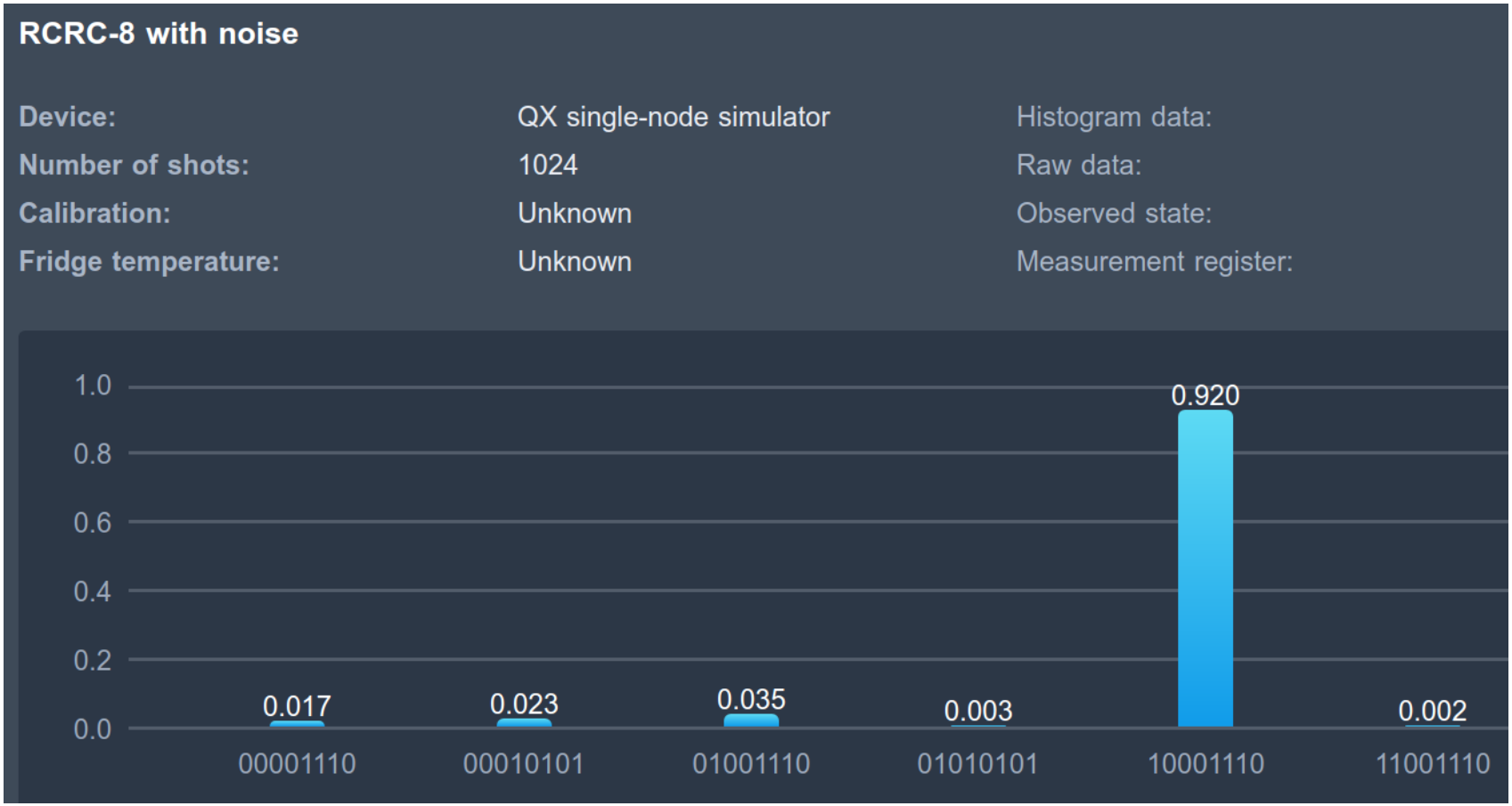
Notre circuit optimal inverse de CRC-8 avec Quantum Inspire

Inversion d'un CRC-8 par calcul quantique (1/4)



Simulation quantique sans bruit sous Quantum Inspire

Inversion d'un CRC-8 par calcul quantique (2/4)



Simulation quantique avec bruit typique sous Quantum Inspire

Inversion d'un CRC-8 par calcul quantique (3/4)

```
def go():
    q = QuantumRegister(8, 'q'); b = ClassicalRegister(8, 'b'); qc1 = QuantumCircuit(q, b)
    qc1.x(q[1]); qc1.x(q[3]); qc1.barrier(q) # Input value
    qc1.cx(q[5], q[4]); qc1.cx(q[6], q[5]); qc1.cx(q[7], q[4])
    qc1.cx(q[5], q[3]); qc1.cx(q[3], q[2]); qc1.cx(q[4], q[3]);
    qc1.cx(q[3], q[1]); qc1.cx(q[1], q[0]); qc1.cx(q[2], q[1]);
    qc1.cx(q[0], q[7]); qc1.cx(q[7], q[6]); qc1.cx(q[1], q[7])
    qc1.barrier(q); qc1.measure(q, b)
    job_sim = execute([qc1,], Aer.get_backend('qasm_simulator'))
    sim_result = job_sim.result(); print("simulation: ",sim_result.get_counts(qc1))
    print("\n(IBMQ Backends)", IBMQ.backends())
    try:
      #least_busy_device = least_busy(IBMQ.backends(simulator=False))
      least_busy_device = IBMQ.get_backend('ibmq_16_melbourne')
      print("Running on current least busy device: ", least_busy_device)
      # running the job
      job_exp = execute([qc1,], backend=least_busy_device, shots=1024)
      interval = 10
      while job_exp.status().name != 'DONE':
        print(job_exp.status().name)
        time.sleep(interval)
      exp_result = job_exp.result()
      d=exp_result.get_counts(qc1)
      print(sorted([(v,k) for k,v in d.items()], reverse=True))
    except ValueError:
        print("All devices are currently unavailable.")
```

Inversion d'un CRC-8 sur puce quantique
(programme, IBM Q 14 Melbourne)

Inversion d'un CRC-8 par calcul quantique (4/4)

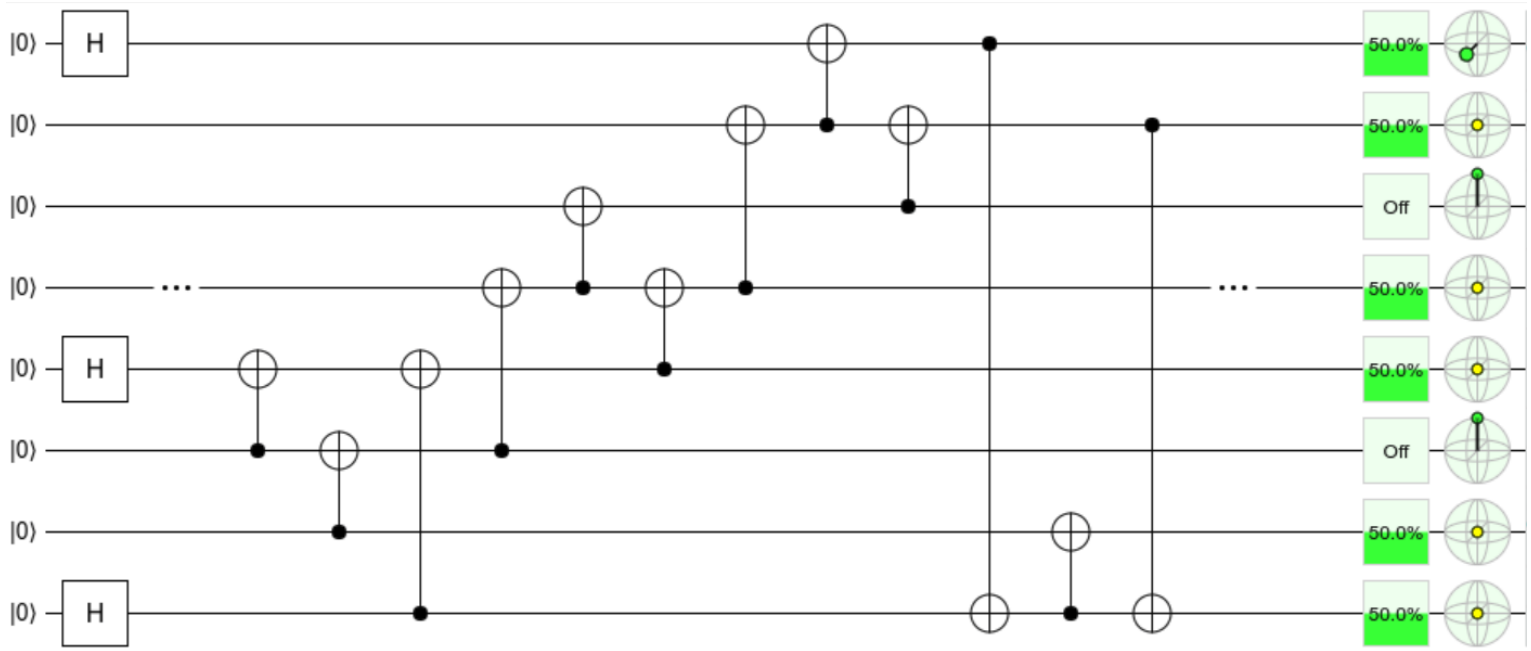
```
go()
```

```
simulation:  {'10001110': 1024}

(IBMQ Backends) [<IBMQBackend('ibmqx4') from IBMQ()>, <IBMQBackend('ibmqx5') from IBMQ()>, <IBMQBackend('ibmqx2') f
rom IBMQ()>, <IBMQBackend('ibmq_16_melbourne') from IBMQ()>, <IBMQBackend('ibmq_qasm_simulator') from IBMQ()>]
Running on current least busy device:  ibmq_16_melbourne
INITIALIZING
RUNNING
RUNNING
[(95, '00000000'), (56, '01000011'), (54, '00100000'), (43, '10001110'), (34, '01000010'), (26, '10000110'), (25,
'01100011'), (25, '00001000'), (24, '11000001'), (24, '10000010'), (24, '00000010'), (24, '00000001'), (23, '110011
01'), (23, '00101000'), (20, '01010011'), (20, '01010010'), (19, '11000101'), (18, '11100001'), (18, '11011100'),
(18, '01001011'), (17, '10101110'), (17, '10001010'), (17, '00000011'), (16, '11000000'), (16, '00001010'), (15, '1
1010000'), (15, '01011111'), (15, '00100001'), (14, '11010001'), (14, '01100010'), (14, '01000111'), (14, '0100000
1'), (14, '00010100'), (14, '00010001'), (14, '00000100'), (13, '11011101'), (13, '10100010'), (13, '10011010'), (1
3, '01001010'), (13, '00011000'), (12, '10010010'), (12, '10000000'), (12, '01101111'), (12, '01010111'), (12, '001
00011'), (11, '11110100'), (11, '11011001'), (11, '11010100'), (11, '01101011'), (11, '00010000'), (10, '1101010
1'), (10, '11001001'), (10, '10110010'), (10, '10011110'), (10, '10010000'), (10, '10001111'), (10, '10001100'), (1
0, '10000111'), (10, '01111111'), (10, '01010000'), (10, '01000000'), (10, '00110001'), (9, '11101100'), (9, '11100
101'), (9, '11100000'), (9, '11000100'), (9, '10100110'), (9, '10100011'), (9, '10011111'), (9, '10010011'), (9, '0
1110111'), (9, '01110010'), (9, '01001111'), (9, '01000110'), (9, '00111100'), (9, '00110000'), (8, '11110001'),
```

Inversion d'un CRC-8 sur puce quantique
(résultats, IBM Q 14 Melbourne)

Inversion de plusieurs CRC-8s avec bits fixés et non fixés



Simulation quantique & résultats avec Quirk: des bits à zéro sont trouvés en entrée pour 8 sorties différentes !
(`https://tinyurl.com/rcrc8multi`)

Inverser un chiffrement XOR par oracle

Inverser un chiffrement XOR par oracle

- Idée: pour une taille de clé donnée, implémenter un chiffrement XOR direct et trouver les clés candidates en minimisant les MSBs non nuls (caractères ASCII non étendu dans le texte original

Menaces quantiques contre la cryptographie actuelle

Menaces quantiques contre la cryptographie symétrique

La principale menace est l'algorithme de Grover :

- Algorithme purement quantique pour chercher parmi N valeurs non triées
- Complexité: $\mathcal{O}(\sqrt{N})$ opérations et $\mathcal{O}(\log N)$ en stockage
- Algorithme probabiliste, itératif et optimal

Défense: doubler toutes les tailles de clés symétriques est suffisant pour échapper à toutes les attaques quantiques futures

Menaces quantiques contre la cryptographie asymétrique

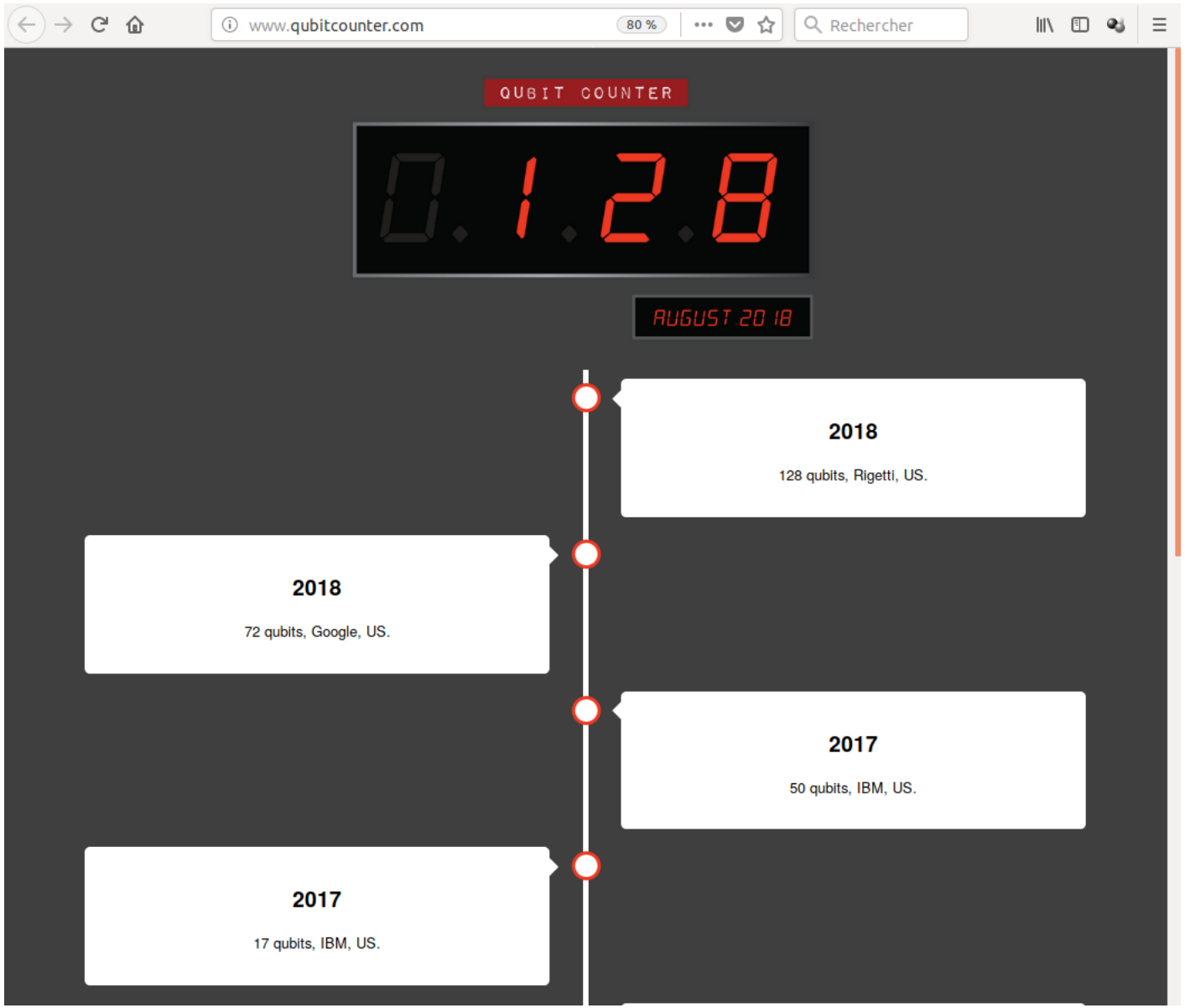
La principale menace est l'algorithme de Shor :

- Algorithme quantique pour la recherche de période formulé en 1994
- Complexité: $\mathcal{O}((\log N)^3)$ opérations et espace de stockage
- Algorithme probabiliste qui trouve la période de la séquence a^k mod N puis des racines de l'unité
- Utilise une QFT, quelques étapes effectuées sur un ordinateur classique
- Casse RSA, DSA, ECDSA, ECDLP de façon efficace!

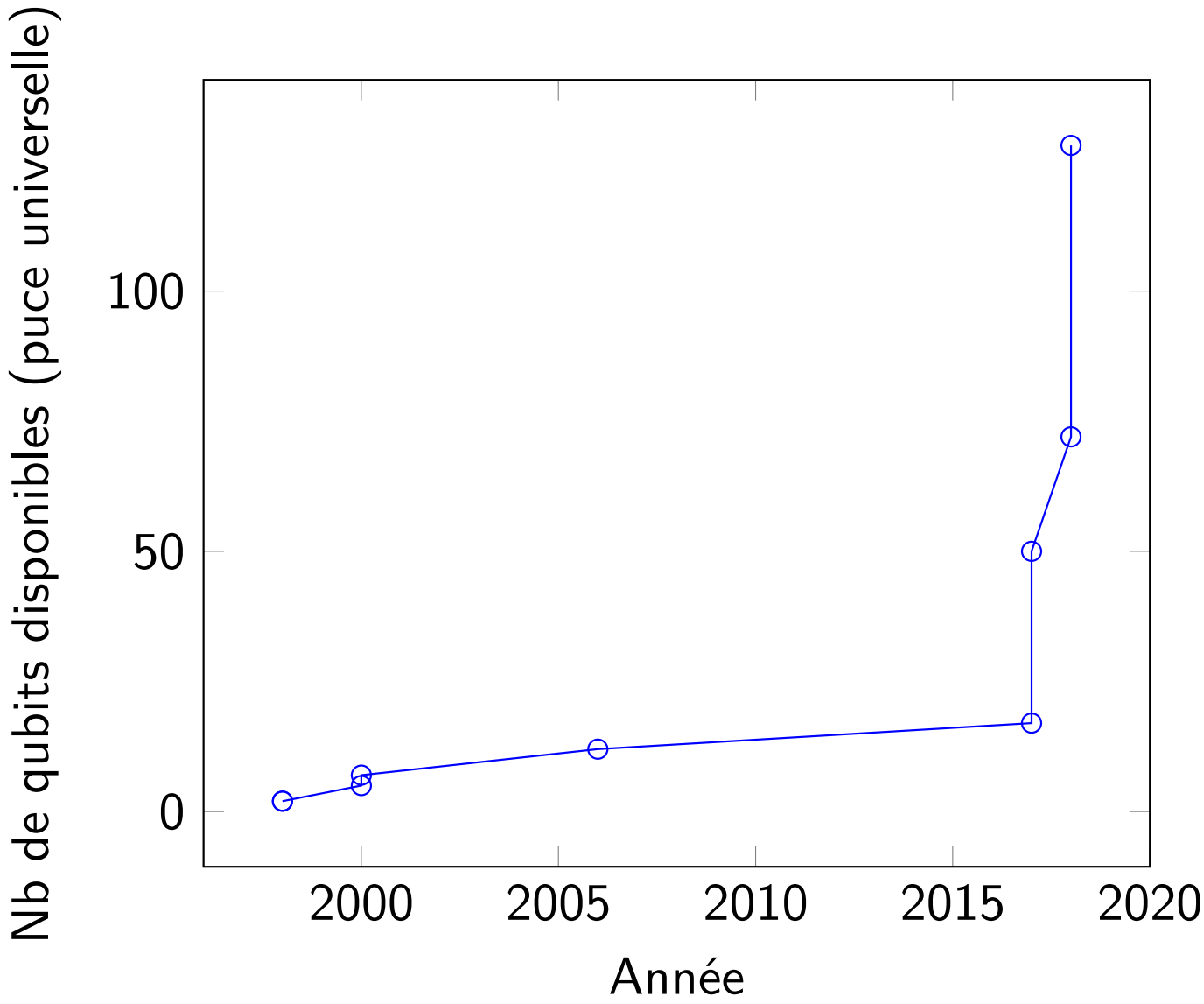
Défense: utiliser de la cryptographie post-quantique

Cryptographie post-quantique

Progrès en nombre de qubits des puces (1/2)



Progrès en nombre de qubits des puces (2/2)



Ressemble à la loi de Moore... ☺

Cryptographie résistante aux algorithmes quantiques (1/3)

Actuellement 6 principales approches :

- "Lattice-based cryptography"
- "Multivariate cryptography"
- "Hash-based cryptography"
- "Code-based cryptography"
- "Supersingular Elliptic Curve Isogeny cryptography"
- "Symmetric Key Quantum Resistance"

Evénement annuel à propos de la cryptographie quantique :
PQCrypto conference
(`https://twitter.com/pqcryptoconf`, 10th edition in 2019)

Cryptographie résistante aux algorithmes quantiques (2/3)

Peu d'algorithmes post-quantiques asymétriques, le plus connu est NTRU, a lattice-based shortest vector problem:

- NTRUEncrypt pour le chiffrement (1996)
- NTRUSign pour la signature électronique

`https://www.onboardsecurity.com/products/ntru-crypto`

Cryptographie post-quantique expérimentée dans Google Chrome

Cryptographie résistante aux algorithmes quantiques (3/3)

Article MISC HS n°13 "Le grand défi du post-quantique" :
`https://connect.ed-diamond.com/MISC/MISCHS-013/Le-grand-defi-du-post-quantique`
(Ludovic Perret & Jean-Charles Faugère)

Merci de votre attention !

Echanges & discussion (30 minutes)

renaud.lifchitz@digital.security