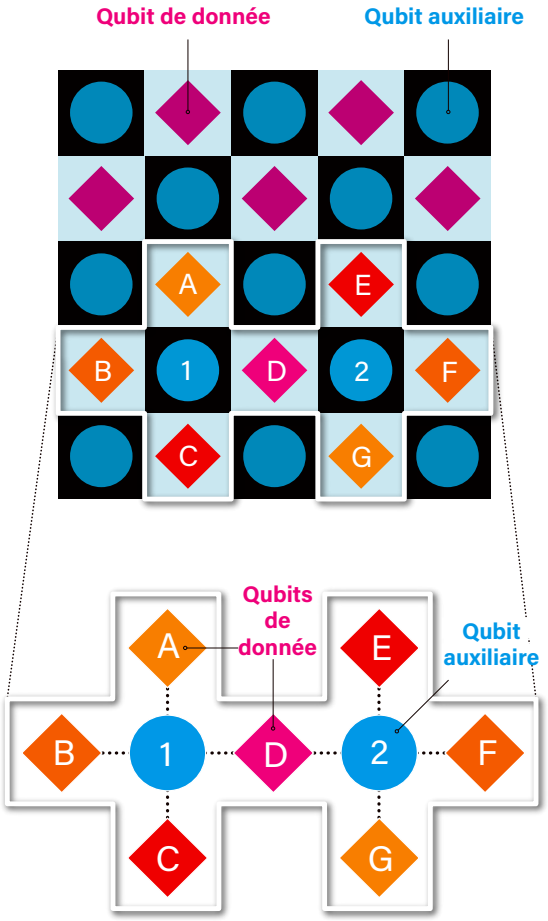
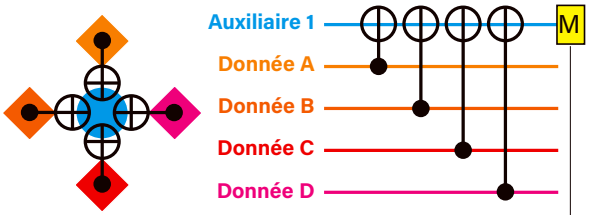
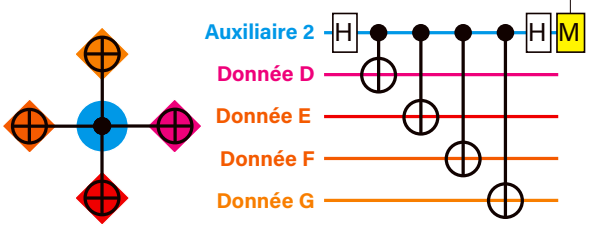
Configuration du code de surface



Détection d'erreur



Si les deux mesures [M] sont égales à 0, le code ne contient pas d'erreur.



Le défi aujourd'hui est donc de trouver des moyens pour améliorer les codes correcteurs afin qu'ils puissent gérer des taux d'erreurs plus élevés, une plus grande variété d'erreurs et s'adapter aux contraintes du matériel.

Les codes correcteurs d'erreurs les plus populaires sont les « codes quantiques topologiques ». Leur origine remonte à 1982, lorsque Frank Wilczek, du MIT (l'institut de technologie du Massachusetts), aux États-Unis, a proposé que l'Univers pouvait contenir une toute nouvelle catégorie de particules. Contrairement aux types connus, dont le spin a une valeur entière (comme le photon) ou demi-entière (comme l'électron), ces particules auraient des valeurs fractionnaires intermédiaires (mais elles n'existeraient que dans des géométries bidimensionnelles). Le physicien les a nommées « anyons » et pensait que les applications pratiques de ces objets étaient probablement lointaines.

DES ANYONS POUR LA CORRECTION

Mais les physiciens ont vite découvert que les anyons n'étaient pas si inutiles que cela. En fait, ils sont liés à des phénomènes du monde réel, dans certains systèmes de matière condensée. Pour achever leur passage de la théorie aux besoins pratiques de la technologie, Alexei Kitaev, de Caltech (l'institut de technologie de Californie), a constaté que les anyons constituaient un outil efficace pour exprimer certains calculs quantiques. Il a alors montré qu'ils avaient une application directe dans des systèmes qui manipulent un grand nombre de particules pour en faire un code quantique.

Dans ces systèmes, les particules sont connectées dans un réseau bidimensionnel où leur état d'énergie le plus bas est fortement intriqué. Les erreurs correspondent à des états d'énergie plus élevée du système, on parle d'excitations. Ces excitations sont représentées par des anyons. Ces idées ont marqué la naissance des codes topologiques (et avec eux, un autre pont entre le calcul quantique et la physique de la matière condensée). Comme le bruit est censé agir localement sur le réseau et que les codes topologiques ont des excitations localisées, ils sont rapidement devenus le système préféré pour protéger les informations quantiques.

Le « code de surface » (ou « code de Kitaev ») et le « code de couleur » sont deux exemples de codes topologiques. Le premier a été développé par Alexei Kitaev et Sergey Bravyi, un de mes collègues chez IBM. Le code de surface est composé des qubits initiaux portant l'information et de qubits auxiliaires, disposés de façon alternée sur un carré bidimensionnel, comme sur un échiquier, les premiers occupant les cases blanches, les seconds les noires.

© Jen Christiansen

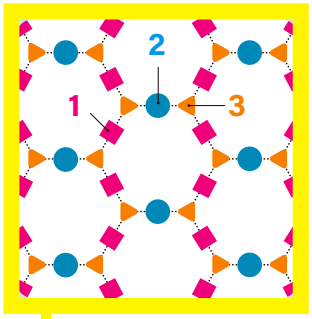
La théorie qui sous-tend les codes de surface est convaincante, mais lorsque nous avons commencé à les mettre en application chez IBM, nous avons rencontré des difficultés. Pour comprendre ces obstacles, il faut d'abord préciser le fonctionnement des qubits que nous utilisons. Il s'agit de transmons, un type de qubits supraconducteurs, qui reposent sur des courants électriques qui oscillent dans un circuit de matériaux supraconducteurs. Le qubit est défini par la superposition des états «0» et«1», correspondant à différentes charges électriques du circuit. Pour effectuer des opérations sur un qubit, nous émettons des impulsions d'énergie à une fréquence microonde spécifique. Lors de sa conception, chaque qubit est réglé pour être réceptif à une fréquence donnée, de sorte à pouvoir atteindre individuellement chaque qubit sans perturber ses voisins. Mais le système n'est pas parfait: la fréquence s'écarte parfois de la valeur prévue; les gammes de fréquences des impulsions se chevauchent de sorte qu'un signal destiné à un qubit modifie de temps en temps la valeur d'un qubit voisin (on parle de diaphonie ou de *crosstalk*). La forte densité de la grille du code de surface, où chaque qubit est connecté à quatre autres qubits, provoque un trop grand nombre de ces collisions de fréquences.

DES ÉCHIQUIERS AUX COLONS DE CATAN

Pour résoudre ce problème, notre équipe a décidé de connecter chaque qubit à un nombre plus faible de voisins. Le résultat est un réseau hexagonal qui ressemble plus à la configuration du jeu de plateau Catan qu'à un plateau d'échecs. Ce système réduit efficacement le *crosstalk*. Mais avec cette nouvelle configuration physique, l'équipe des théoriciens d'IBM a dû développer le code correcteur correspondant.

Ce nouveau code, nommé «code hexagonal lourd», combine certaines caractéristiques du code de surface et d'un autre, le code Bacon-Shor, également conçu pour les réseaux bidimensionnels. La connectivité plus faible des qubits dans notre code signifie que certains qubits doivent servir d'intermédiaires pour identifier les erreurs qui se sont produites, ce qui conduit à des circuits légèrement plus complexes et donc à un seuil où le bruit n'est plus contrôlé un peu plus faible. Nous avons constaté que le compromis entre une diminution du *crosstalk* et un seuil moins bon était satisfaisant.

Il reste un autre problème à résoudre. Les codes qui vivent sur des plans bidimensionnels et qui n'intègrent que les connexions entre voisins les plus proches ont besoin de beaucoup de qubits physiques pour définir un seul qubit logique (ce rapport est désigné par le terme anglais d'*overhead*). Le qubit logique correspond à l'unité d'information intéressante que l'on veut protéger contre les erreurs et les qubits physiques sont les éléments matériels qui opèrent ensemble pour définir le qubit logique et pour garantir qu'il ne contient pas d'erreur. Pour corriger davantage d'erreurs, il faut construire un code plus grand, qui utilise encore plus de qubits physiques pour créer un seul qubit logique. Il est donc nécessaire de manipuler plus de composants physiques pour au final traiter la même quantité de données. Mais l'ajout de matériel rend d'autant plus difficile la maîtrise de qubits de bonne qualité pour ne pas dépasser le seuil d'erreurs.



1. Qubit de donnée
2. Qubit auxiliaire
3. Qubit de signalisation

Les ingénieurs quantiques ont deux options. Première piste, ils acceptent ce large *overhead* – avec un nombre élevé de qubits et de portes logiques – comme le prix à payer pour garder une architecture simple. Et ils s'échinent à comprendre et optimiser les différents facteurs qui y contribuent. Ou alors, ils cherchent un code plus efficace. Par exemple, pour coder plus de qubits logiques à partir d'un nombre plus réduit de qubits physiques, une solution serait de faire interagir des qubits plus éloignés les uns des autres. Pour dépasser la limite du contact aux seuls voisins les plus proches dans le réseau à deux dimensions, il faudrait considérer des réseaux à trois dimensions, ou plus. Notre équipe explore les deux stratégies.

Un ordinateur quantique ne sera vraiment utile que lorsqu'il sera en mesure d'effectuer toutes les opérations de calcul possibles. On oublie souvent ce critère d'universalité, ce qui conduit à de nombreuses confusions sur l'informatique quantique. En d'autres termes, tous les dispositifs que les gens nomment «ordinateurs» quantiques ne sont pas vraiment des ordinateurs. La plupart s'apparentent davantage à des machines à calculer capables de réaliser un nombre limité de tâches.

Le fait de négliger la nécessité d'un calcul universel est également à l'origine de faux espoirs concernant les qubits logiques et la correction d'erreurs. Protéger les informations en mémoire contre les erreurs est un début, mais cela ne résout pas tous les problèmes. Pour obtenir un ordinateur quantique universel, il faut savoir construire un certain nombre de portes logiques différentes. Combinées entre elles, elles sont susceptibles de réaliser toutes les opérations logiques imaginables. Mais ces portes doivent, elles aussi, être robustes contre les erreurs. C'est là que les choses se compliquent.

Certaines portes, appartenant à la catégorie des portes dites «transverses», sont relativement robustes. Quand vous appliquez l'opération d'une porte logique transverse sur un qubit logique, vous devez appliquer cette même opération à chacun des qubits physiques associés. Quand une telle porte opère entre