optique quantique TA

Lautaro Labarca
labl2714@usherbrooke.ca

September 25, 2024

Contents

1 Course-1. Qubit code.

L'objectif est de familiariser les étudiants avec la dynamique simple des qubits. Pour ça, on va faire un exercice computationnel. Nous allons esquisser quelques étapes analytiques pour obtenir les formes simples que nous étudierons numériquement, mais aucun détail de ces dérivations n'est fourni. En général, elles sont simples, mais légèrement fastidieuses à reproduire. Si ce n'est pas déjà fait, il est bon de compléter les étapes manquantes. Le chapitre 4 de Nielsen et Chuang [1], le chapitre 2 de Sakurai [2], et la page Wikipédia sur les transformations unitaires couvrent tout le nécessaire pour combler ces lacunes. En guise de spoiler, la semaine prochaine, lors de la révision des devoirs, nous passerons en revue tous les détails analytiques fastidieux, mais nécessaire.

1.1 Configuration de qutip avec Visual Studio Code

Tout d'abord, nous devons installer Python. Allez sur le lien [3]. Utilisez simplement la version recommandée. Pendant que Python s'installe, nous pouvons installer Visual Studio Code [4]. Ensuite, créez simplement un dossier pour le cours, ouvrez-le, et téléchargez-y le notebook utilisé en classe [5]. Créez un environnement virtuel en tapant dans la palette de commandes `Python: Create Environment`. Maintenant, utilisez simplement la palette de commandes pour ouvrir le terminal. Dans le terminal, tapez `pip install qutip`. De la même manière, installez `numpy` (manipulations basiques de tableaux), `matplotlib` (signification évidente), `scipy` (manipulations de tableaux plus efficaces, précises et variées, avec de nombreuses fonctions spécifiques comme l'intégration, les séries, les polynômes, et le fitting), `tqdm` (pour voir des barres de progression). Vous pouvez voir toutes les versions installées avec `pip list`. Vérifiez les compatibilités dans la documentation d'installation de `qutip`, à savoir essentiellement `numpy` < 2.0, `scipy` > 1.8, `python` > 3.9, `matplotlib` $> 1.2.1$. Avec cela, vous êtes prêts à exécuter tous les codes utilisés dans ce cours. Enfin, configurez GitHub Copilot. L'utilité de cet outil c'es top. Avec Visual Studio Code, c'est très facile : il suffit d'installer l'extension et de lier votre compte GitHub étudiant.

Alternativement, créez un compte sur Cocalc, allez dans vos projets, démarrez-en un nouveau et téléchargez le notebook qubit-drive-pulse.ipynb trouvé sur le GitHub [5].

1.2 Pourquoi nous intéressons-nous aux qubits?

Tout d'abord, le qubit est l'objet mathématique le plus simple qui capture les caractéristiques essentielles de la mécanique quantique non relativiste. Avec les qubits, nous pouvons avoir la superposition, c'est-à-dire l'interférence. En particulier, le fait qu'il y ait deux signes est suffisant, voir par exemple [6]. De plus, avec deux qubits ou plus, nous pouvons étudier l'entrelacement, probablement la caractéristique la plus frappante de la mécanique quantique comme l'a posée Einstein et ses collaborateurs [7] (sérieusement, l'article est très facile à lire, allez le lire), ce qui peut conduire à des corrélations non locales entre les particules entrelacées comme le montrent d'abord [8, 9] basées sur les inégalités de Bell [10] (très simples à lire également, c'est fantastique). Plus récemment, les inégalités ont été violées en utilisant des circuits supraconducteurs dans [11]. Il est à noter que certains chercheurs affirment encore que les inégalités de Bell telles que proposées par Bell n'ont pas encore été testées, voir par exemple [12].

Deuxièmement, pour la physique fondamentale, les qubits, en raison de leur simplicité, sont très utiles pour concevoir des expériences mesurant une force ou une interaction désirée. Par exemple, des tests ont été réalisés pour rechercher la matière noire [13]. Cet expérience, comme beaucoup d'autres, est basée sur le schéma montré dans fig. 1. De plus, les qubits (sous forme d'atomes, on mesure les probabilités de transition ; si une seule transition est pertinente, alors les deux niveaux correspondants forment un qubit) sont utilisés dans des expériences mesurant la constante gravitationnelle (je manque d'une référence particulière pour cela) et font partie de propositions testant la nature quantique de la gravité, voir par exemple le dossier [14].

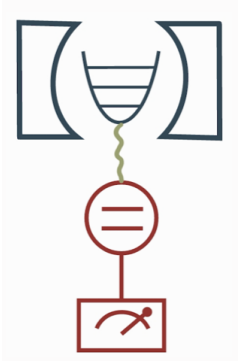

Figure 1: Oscillateur couplé à un qubit. Dans ce cas, le qubit est utilisé pour extraire des informations de l'oscillateur, mais l'inverse est également utilisé.

Troisièmement, les qubits forment la base de l'informatique et de l'information quantiques. Au-delà de la promesse de démontrer que le modèle computationnel le plus fondamental n'est pas la machine de Turing classique, mais l'ordinateur quantique, ils promettent d'améliorer les protocoles cryptographiques. Nous verrons cela dans le chapitre 4.[1]

1.3 Dynamique simple des qubits

En général, la dynamique d'un qubit conservatif est générée par l'Hamiltonien

$$\hat{H}(t) = f(t)\sigma_z + g(t)\sigma_x + h(t)\sigma_y. \tag{1.3.1}$$

Ainsi, formellement, l'évolution est donnée par

$$\hat{U} = \mathcal{T}[i\exp\left\{\int_0^t \mathrm{d}t' f(t')\sigma_z + g(t')\sigma_x + h(t')\sigma_y\right\}] \equiv \exp\{i\alpha(t)\sigma_z\}\exp\{i\beta(t)\sigma_x\}\exp\{i\gamma(t)\sigma_z\}, \tag{1.3.2}$$

où la dernière équivalence est due à la décomposition des rotations, voir [1] chapitre 4. En résumé, toute évolution non dissipative d'un qubit unique est simplement une rotation et nous pouvons la

[1] Il y a eu tant de choses écrites à ce sujet que je préfère ne pas commenter davantage pour le moment. De plus, je manque de temps, donc je ne vais pas inclure d'autres références pour l'instant, mais je pourrais mettre à jour cela à l'avenir.

visualiser dans la sphère de Bloch. Dans le code partagé [5], vous trouverez une animation montrant l'impulsion de $\pi/2$ permettant de préparer $|+\rangle$ à partir de $|0\rangle$.

En raison de la décomposition des rotations ci-dessus, nous pouvons nous concentrer simplement sur les Hamiltoniens de la forme,

$$\hat{H}(t) = f(t)\sigma_z + g(t)\sigma_x. \tag{1.3.3}$$

L'Hamiltonien ci-dessus est couramment généré dans les circuits supraconducteurs en utilisant un qubit avec une fréquence dépendante du temps (par exemple, en faisant passer un flux magnétique à travers un SQUID), et en excitant le qubit à travers une ligne de charge (la partie $g(t)\sigma_x$). Pour simplifier, concentrons-nous sur $f(t) = \omega_0/2$ constant, ou ω_0 étant la fréquence du qubit. En passant à un cadre tournant à la fréquence ω, nous obtenons dans le cadre d'interaction un Hamiltonien de la forme suivante

$$\hat{H}(t) = \Delta\sigma_z + g(t)\sigma_x. \tag{1.3.4}$$

Notez que $g(t)$ est modifié, mais pour notre étude qualitative, la forme exacte n'est pas importante. En fait, nous simplifierons encore davantage, et nous poserons $\Delta = 0$, pour obtenir

$$\hat{H}(t) = g(t)\sigma_x. \tag{1.3.5}$$

Ensuite, l'évolution unitaire est donnée par

$$\hat{U}(t) = \exp\left[-i\int_0^t \mathrm{d}t' g(t)\sigma_x\right]. \tag{1.3.6}$$

Ainsi, en redimensionnant simplement $g(t) \to 2\pi g(t)$, et en utilisant le fait qu'une rotation autour de l'axe x dans la sphère de Bloch par un angle θ est donnée par

$$\hat{R}_x(\theta) \equiv \exp\{-i\theta\sigma_x/2\}, \tag{1.3.7}$$

nous obtenons

$$\hat{U}(t) = \hat{R}_x(2I), \quad \text{avec} \quad I = \int_0^t \mathrm{d}t' g(t). \tag{1.3.8}$$

Cela signifie que si nous ne considérons que le sous-espace des qubits, seule l'intégrale de l'impulsion, c'est-à-dire son amplitude totale, est importante, et sa forme est sans importance pour la fidélité de l'état final. Cependant, en ajoutant un troisième état, nous verrons que ce n'est plus le cas et que la forme de l'impulsion, et en particulier son profil de fréquence, jouent un rôle crucial.

1.4 Profil de fréquence de l'impulsion et ses effets sur les fuites (leakage)

Ici, je vais développer un exemple avec un qutrit, avec des détails. Le code associé est téléchargé dans [5] et s'appelle qutrit. Cela devrait (légèrement) vous être utile pour l'exercice 1 de votre devoir. Nous pouvons commencer à étudier le problème dans sa forme abstraite la plus générale pour mieux le comprendre. Du fait que l'évolution quantique est linéaire, nous pouvons imaginer le scénario général où l'évolution du qutrit est générée par un Hamiltonien hermitien arbitraire $\hat{H}(t)$, c'est-à-dire

$$\frac{\partial\,|\psi(t)\rangle}{\partial t} = \hat{H}(t)\,|\psi(t)\rangle\,, \tag{1.4.1}$$

sous forme matricielle

$$\begin{pmatrix}\dot{c}_0(t)\\ \dot{c}_1(t)\\ \dot{c}_2(t)\end{pmatrix} = \begin{pmatrix}H_{11}(t) & H_{12}(t) & H_{13}(t)\\ \bar{H}_{12}(t) & H_{22}(t) & H_{23}(t)\\ \bar{H}_{13}(t) & \bar{H}_{23}(t) & H_{33}(t)\end{pmatrix}\begin{pmatrix}c_0(t)\\ c_1(t)\\ c_2(t)\end{pmatrix}, \tag{1.4.2}$$

où les barres représentent la conjugaison. Pour simplifier encore davantage, disons que l'Hamiltonien ne couple que les états voisins, c'est-à-dire $H_{13}(t) = 0$. De plus, simplifions encore notre analyse en supposant que les termes diagonaux sont constants dans le temps et que les termes hors diagonale évoluent de manière harmonique à la fréquence ω_0, c'est-à-dire que la transformée de Fourier de $H_{ij}(t)$ est $\delta(\omega - \omega_0)$. Nous écrivons donc,

$$\begin{pmatrix}\dot{c}_0(t)\\ \dot{c}_1(t)\\ \dot{c}_2(t)\end{pmatrix} = \begin{pmatrix}H_{11} & A_{12}e^{i\omega_0 t} & 0\\ \bar{A}_{12}e^{-i\omega_0 t} & H_{22} & A_{23}e^{i\omega_0 t}\\ & \bar{A}_{23}e^{-i\omega_0 t} & H_{33}\end{pmatrix}\begin{pmatrix}c_0(t)\\ c_1(t)\\ c_2(t)\end{pmatrix}. \tag{1.4.3}$$

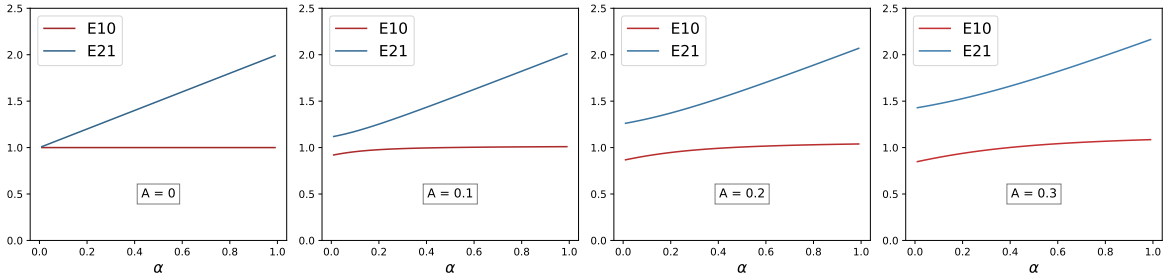

Figure 2: Énergies propres du qutrit en fonction de l'anharmonicité pour différentes amplitudes de contrôle. E10 correspond à $E_1 - E_0$.

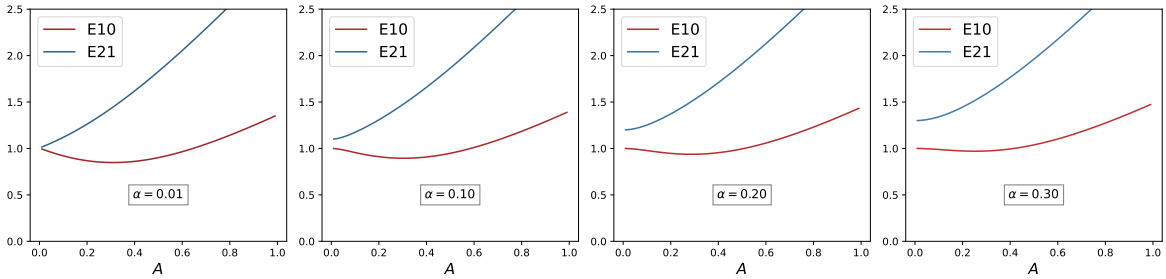

Figure 3: Énergies propres du qutrit en fonction de l'amplitude du contrôle pour différentes anharmonicités. E10 correspond à $E_1 - E_0$.

De plus, avec un simple décalage des énergies, nous pouvons fixer H_{11} à zéro, donc nous avons

$$\begin{pmatrix} \dot{c}_0(t) \\ \dot{c}_1(t) \\ \dot{c}_2(t) \end{pmatrix} = \begin{pmatrix} 0 & A_{12}e^{i\omega_0 t} & 0 \\ \bar{A}_{12}e^{-i\omega_0 t} & H_2 & A_{23}e^{i\omega_0 t} \\ & \bar{A}_{23}e^{-i\omega_0 t} & H_3 \end{pmatrix} \begin{pmatrix} c_0(t) \\ c_1(t) \\ c_2(t) \end{pmatrix}. \tag{1.4.4}$$

Enfin, faisons une simplification supplémentaire en supposant $A_{12} = A_{13} \equiv A \in \mathbb{R}$, afin d'isoler la composante en fréquence du champ de contrôle dans notre analyse. En faisant cela, nous trouvons maintenant les valeurs propres. Pour ce faire, nous résolvons simplement l'équation déterminantielle $\det(H - \lambda I) = 0$, qui dans ce cas se lit :

$$\lambda(H_2 - \lambda)(H_3 - \lambda) + A^2(H_3 - 2\lambda) = 0. \tag{1.4.5}$$

Les valeurs propres ne dépendent alors pas de la fréquence du contrôle, mais seulement de son amplitude et de l'anharmonicité relative entre H_2 et H_3. Il est également à noter que l'équation ci-dessus est un polynôme de degré trois, et bien que nous puissions essayer de trouver des solutions analytiques, cela serait plutôt fastidieux, nous allons donc les résoudre numériquement. Pour ce faire, nous fixons $H_2 = 1$, $H_3 = 1 + \alpha$. Notez que ce choix signifie que $\omega = 1$ est la fréquence d'oscillation entre $|0\rangle$ et $|1\rangle$. De plus, nous fixons $A = 0.2$, ce qui correspond à un faible champ de contrôle, afin d'étudier la structure des vecteurs propres et des énergies propres en fonction de la fréquence du contrôle ω_0 et de l'anharmonicité α.

Dans la fig. 2, nous représentons les valeurs propres en fonction de α pour différentes valeurs de A, et dans la fig. 3, nous les représentons en fonction de A pour différentes valeurs de α. Il est clair qu'en augmentant α, comme attendu, l'écart entre E_{10} et E_{21} augmente. De même, l'augmentation de l'amplitude du contrôle élargit l'écart entre E_{21} et E_{10}. C'est l'écart typique qui apparaît lorsqu'un Hamiltonien est perturbé dans ses éléments non diagonaux de plus proche voisin. Maintenant, disons que nous voulons réaliser une porte, soit de $|0\rangle \to |1\rangle$ ou de $|0\rangle \to |2\rangle$. Quelle fréquence de pilotage devons-nous choisir pour maximiser la fidélité de la porte ? Pour répondre à cette question, nous pouvons simplement résoudre l'évolution sur une durée suffisamment longue, disons $T = 10(2\pi/\omega_0)$, et trouver les valeurs maximales de

$$c_i = |\langle i|\psi(t)\rangle|. \tag{1.4.6}$$

Nous le faisons numériquement en fixant $A = 0.2$ et $\alpha = 0.5A$ dans la fig. 4. De toute évidence, la fréquence des termes de pilotage a une forte influence sur les fidélités atteignables. Il est certain que trouver les fréquences qui maximisent la fidélité est simple numériquement, mais qu'en est-il de prédire cette fréquence uniquement à partir des paramètres du système non piloté et de l'amplitude du pilote A par exemple. Là, dans la fig. 4, nous avons tracé les énergies $E21$ et $E10$, et bien que ces valeurs donnent un ordre de grandeur des pics, elles échouent considérablement en tant qu'outil prédictif. Si vous savez comment prédire les pics, faites-le moi savoir[2]. Nous notons que, naturellement, les profils dépendent de A et α, par exemple en fixant $A = 0.2$, $\alpha = 5A = H_2$, nous obtenons le profil intéressant de la fig. 5.

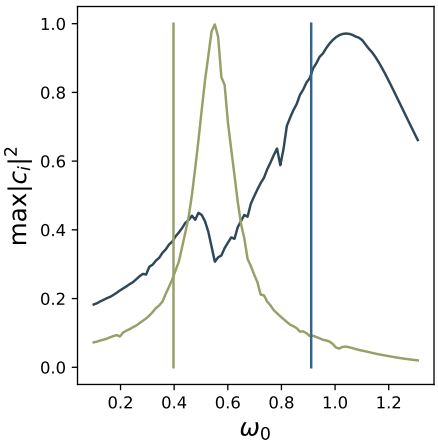

Figure 4: Maximum atteignable c_i en partant de $|0\rangle$ avec $A = 0.2$ et $\alpha = 0.5A$. c_1 en bleu, c_2 en vert. Les lignes verticales correspondent à $E10$ et $E21$.

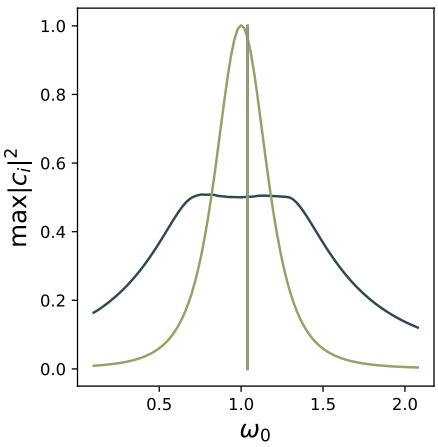

Figure 5: Maximum atteignable c_i en partant de $|0\rangle$ avec $A = 0.2$ et $\alpha = 5A = H_2$. c_1 en bleu, c_2 en vert. Les lignes verticales correspondent à $E10$.

Enfin, nous remarquons qu'il est clair que si le profil de fréquence du ton de pilotage est proche d'une résonance ou de l'autre, cela affectera les fuites. Imaginez une situation où le profil du ton de pilotage correspond à deux deltas de Dirac, l'un à la fréquence qui maximise c_1 (l'autre c_2), alors nous nous attendrions à ce que le maximum atteignable de chacun d'eux soit réduit par rapport au cas idéal. Pour cette raison, l'ingénierie des tons de pilotage est un sujet de recherche qui suscite un grand

[2]Attention, je ne dis pas que cette prédiction est inconnue, je dis simplement que je ne la connais pas pour le moment. Aller simplement dans l'espace de Fourier au niveau des équations pourrait probablement donner suffisamment d'informations pour comprendre.

intérêt. Une approche numérique courante pour trouver des tons de pilotage idéaux est une boucle d'optimisation basée sur la descente de gradient.

2 Course-3. Cryptography.

2.1 Objectif

L'objectif de ce cours est de donner un aperçu de la cryptographie et d'expliquer de manière générale pourquoi la cryptographie quantique pourrait être intéressante. Idéalement, les étudiants seront capables de lire davantage dans la littérature après le cours, ou de revenir à ces notes chaque fois qu'ils auront besoin d'apprendre la cryptographie quantique pour tout problème sur lequel ils pourraient travailler.

Outline of the class (logically makes no sense, but for the impact value in the class seems adequate)

- Objectives of criptography.
- Setup, Alica, Bob, Eve, channels.
- Vernam key (one time pad), intuition on how information is decoded by spies.
- Quantum key distribution.
- No cloning theorem and information gain implies perturbation.
- BB84.
- Classical key distribution.
  - Information reconciliation and privacy amplification.
  - Shannon entropy and mutual information.
  - Error correction, repetition code.

2.2 Introduction à la cryptographie

Ainsi, en cryptographie, nous avons deux objectifs :

- Anonymat : protéger l'émetteur du message.
- Chiffrement : protéger le message.

Un exemple de cryptographie moderne ancienne est la machine Enigma. En réalité, elle a été inventée pour la Première Guerre mondiale, pas même la seconde. L'entrée sur Wikipedia contient beaucoup de détails sur son histoire, c'est une lecture intéressante. Ici, nous nous concentrerons sur la structure mathématique de la cryptographie, et en particulier sur la cryptographie quantique.

The basic criptography setup is depicted in fig. 6. Alice, has some message X, that she shares with Bob through some channel. This channel might be public (everyone has free access), prive (only Alice and Bob have access), or semi-private (Alice and Bob have free access, and some spy Eve has partial access). In theory, we can focus only in the semi-private one, but in practice, as semi-private channels are resource expensive, we also use public channels so the flow is mixed as shown in fig. 8.

La méthode de chiffrement la plus courante de nos jours repose sur des *clés*. Ces clés peuvent être soit privées, soit publiques. Le chiffrement par clé publique est une invention moderne des années 1970. Avant cela, tous les systèmes cryptographiques utilisaient des clés privées. Imaginez qu'Alice veuille envoyer un message à Bob. Pour le chiffrer, Alice utilise une *clé de chiffrement*, et Bob dispose d'une *clé de déchiffrement*. Pour comprendre comment ces systèmes fonctionnent en principe, le mieux est d'observer un exemple. Un exemple commun est le *chiffre de Vernam*, car il est simple et très efficace, parfois appelé *one time pad*. Comme le montre la fig. 7, Alice utilise simplement une clé pour coder un message, et Bob utilise la même clé pour le décoder. De toute évidence, la sécurité d'un tel chiffrement dépend avant tout de la capacité à partager la clé de manière sécurisée entre Alice et Bob. La qualité la plus importante de ce codage est qu'il est prouvablement sécurisé sous les hypothèses suivantes :

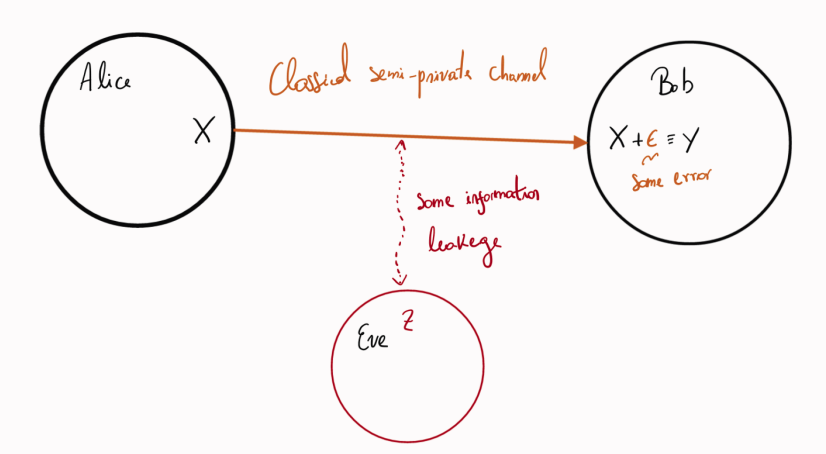

Figure 6: Classical criptography basic setup.

- Les bits de la clé sont générés de manière véritablement aléatoire. Pourquoi ? Un générateur pseudo-aléatoire, comme son nom l'indique, utilise une *graine*. Par exemple, prenons l'heure actuelle sur une horloge. Si un espion découvre la graine et l'algorithme, alors toutes les clés peuvent être reconstruites.
- La longueur de la clé est au moins aussi longue que celle du message. Si la clé se répète, des études statistiques du message peuvent être réalisées, révélant ainsi des informations à un espion (il est d'usage d'appeler l'espion Ève).
- La clé ne doit jamais être réutilisée, ni en partie ni en totalité. Même argument que précédemment.
- Évidemment, la clé doit être conservée complètement secrète.

Cependant, qu'est-ce que cela signifie, « prouvablement sécurisé » ? Nous pouvons utiliser la théorie de l'information pour décrire cette condition mathématiquement de manière rigoureuse. Dans ce qui suit, nous ne nous soucierons pas de la rigueur, mais nous décrirons les éléments et l'intuition de base.

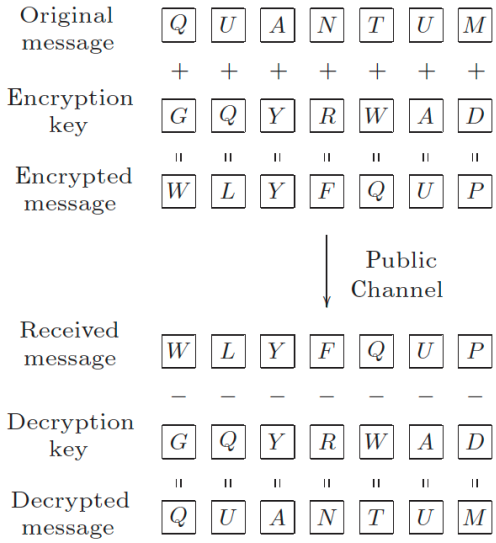

Figure 7: Vernam cipher

Plaçons-nous maintenant dans la position de l'espion, et imaginons que nous essayons de déchiffrer un message simple de 7 lettres comme celui de la fig. 7, que nous écrirons sous la forme $x_1x_2x_3x_4x_5x_6x_7$.

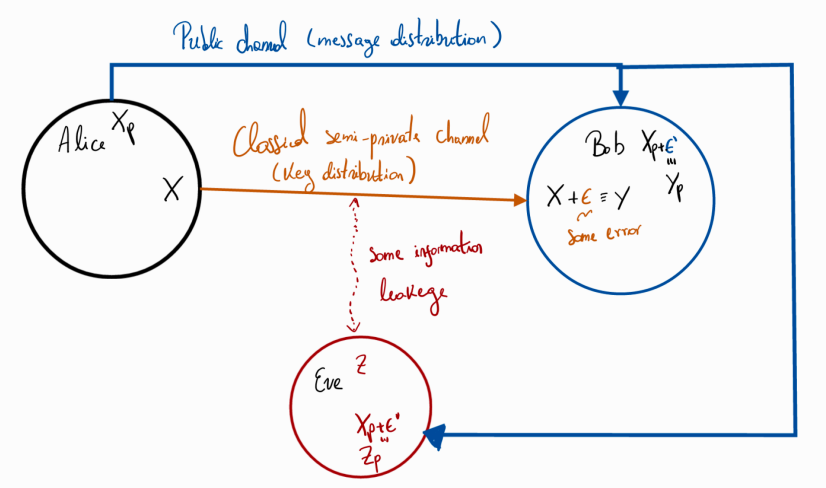

Figure 8: Private key setup.

Pour rendre les choses plus concrètes, disons que nous savons que le message est dans une langue naturelle, concrètement l'anglais. Notre espace d'échantillonnage est donc celui de tous les messages de 7 lettres pouvant être formés en anglais. De plus, on peut imaginer dire que la probabilité que la première lettre soit un a, sur la base d'un simple argument de dénombrement de tous les mots disponibles dans l'espace d'échantillonnage, est donnée par

$$p(x_1 = a) = \frac{\text{tous les messages d'un (deux, trois) mots commençant par a}}{\text{tous les messages de 7 symboles}}, \tag{2.2.1}$$

où l'espace est également inclus comme un symbole possible. En outre, nous disposons d'informations préalables non seulement sur les occurrences, mais aussi sur les corrélations entre ces occurrences, c'est-à-dire que si la première lettre est un a, alors nous avons les probabilités conditionnelles

$$p(x_2 = b|x_1 = a) = \frac{\text{tous les messages d'un (deux, trois) mots qui commencent par a et dont la deuxième lettre est } b}{\text{tous les messages de 7 symboles qui commencent par } a}, \tag{2.2.2}$$

nous pouvons faire de même avec $p(x_3 = c|x_2 = b, x_1 = a)$ et ainsi de suite.

Maintenant, imaginez que dans le même canal public de 7 symboles, Alice et Bob échangent des messages, et que nous, en tant qu'espions, ne savons pas s'ils mettent à jour leur clé à chaque fois ou s'ils utilisent la même clé de 7 symboles tout le temps. Pour déterminer cela, nous pouvons examiner les occurrences de chaque symbole, leurs positions, et les corrélations entre les symboles, puis les comparer avec nos connaissances a priori. Pour organiser nos informations, nous pouvons classer les occurrences par ordre décroissant, comme par exemple, dans la table 1. Ensuite, la première chose que nous pouvons faire est de vérifier si la distribution résultante est uniforme ou si elle présente un biais. Nous pouvons le faire en utilisant la distance entre les fréquences relatives et la distribution aléatoire. Concrètement, supposons que le nombre total d'échantillons soit N, alors nous définissons la distribution observée comme

$$p(x_1 = W) = \frac{2743}{N}, \tag{2.2.3}$$

et ainsi de suite pour les autres. Évidemment, la distribution aléatoire est

$$p(x_1 = W) = \frac{1}{\text{nombre de symboles}}. \tag{2.2.4}$$

Une mesure valide de la distance entre deux distributions est

$$\Delta(X, Y) = \sum_{\alpha \in D} |p(X = \alpha) - p(Y = \alpha)|, \tag{2.2.5}$$

c'est-à-dire la somme sur tout l'espace des échantillons de la différence en valeur absolue entre la probabilité d'obtenir α dans chacune des distributions. Dans notre cas particulier, nous choisissons D

symbole	occurrences dans x_1
W	2743
Q	2525
B	2321
⋮	⋮

Table 1: Occurrences comptées par l'espionne Eve des symboles dans les messages entre Alice et Bob. Ils utilisent une clé finie, et ainsi, il y a un biais dans le comptage des symboles.

pour être tous les symboles, X pour représenter la distribution des fréquences observées, et Y pour représenter la distribution uniforme. Essentiellement, à mesure que nous collectons de plus en plus de messages entre Alice et Bob, si la distance entre les deux distributions diminue, nous pouvons conclure que la clé n'est pas répétée et qu'ils utilisent soit un générateur véritablement aléatoire soit pseudo-aléatoire. En revanche, si la distance augmente en moyenne, nous pouvons conclure qu'il existe un biais systématique et qu'ils répètent leur clé. Dans le premier cas, notre seule voie de progrès est de déterminer s'ils utilisent ou non un générateur pseudo-aléatoire. Il existe assurément beaucoup de littérature sur la manière de procéder, mais nous n'en parlerons pas, car c'est un domaine de recherche à part entière. Dans le second cas, nous pouvons maintenant essayer de reconstruire cette clé finie $\mathbf{k}$. Comment la trouver ? La réponse à cette question est également un domaine de recherche en soi, mais nous pouvons esquisser un algorithme pour avoir une idée de la manière dont cela pourrait être fait.

Imaginons que nous choisissions simplement une clé au hasard $\mathbf{k_0}$, puis qu'avec cette clé nous transformions les messages observés $\mathbf{x}$ en un message décodé,

$$\mathbf{x} \to \mathbf{k}_0(\mathbf{x}). \tag{2.2.6}$$

Nous pouvons imaginer une fonction $L : \mathbf{k}(\mathbf{x}) \to [0, 1]$, qui mesure simplement la probabilité que les mots obtenus soient effectivement des mots anglais et que le message soit cohérent ou non. Encore une fois, la manière exacte de procéder est un domaine de recherche à part entière qui, en fin de compte, a donné naissance à des IA génératrices de texte comme ChatGPT. Disons simplement ici qu'une telle fonction existe et que nous lui faisons confiance. Ensuite, nous pouvons exécuter une optimisation par descente de gradient sur l'espace des clés et trouver celle qui maximise la fonction de probabilité L. Naturellement, une telle approche présente les inconvénients généraux de l'optimisation sur des espaces de paramètres vastes, mais elle est en général résoluble, de sorte que la sécurité de l'encryption n'est plus garantie. Au-delà de ces considérations mathématiques, il est clair que les protocoles à clé privée reposent sur la capacité à créer des canaux de communication sécurisés entre Alice et Bob pour partager la clé, mais comment pouvons-nous faire cela physiquement ? C'est en répondant à cette question que la cryptographie quantique a vu le jour, et nous allons examiner un exemple concret.

2.3 Key distribution

Comme nous l'avons mentionné, l'élément clé des protocoles à clé privée est la distribution des clés. En pratique, cela est réalisé de manière classique, et il existe plusieurs techniques pour surmonter les imperfections. En principe, nous pouvons également le faire de manière quantique, ce qui est provably sécurisé.

2.3.1 Classical key distribution

Ici, nous mentionnerons très brièvement ce que sont *l'amplification de la confidentialité* et *la réconciliation de l'information*. Donner des descriptions détaillées de cette procédure est hors du cadre de ce cours, mais connaître au moins leur existence semble approprié. Ce qui est le plus important, c'est d'avoir simplement une idée de la façon dont la distribution des clés est effectuée et étudiée dans le cadre classique, et ce qui changera lorsque nous passerons à la cryptographie quantique. Nous dessinons un schéma de l'architecture de la cryptographie classique dans fig. 6. Alice a un message, X, et elle le partage avec Bob à travers un canal semi-privé, c'est-à-dire qu'Eve, l'espionne, peut obtenir seulement une partie des informations envoyées à travers le canal. Comme dans tout processus de transmission d'informations, des erreurs peuvent survenir ; en fait, Bob reçoit le message avec une certaine erreur, et son message devient donc la variable aléatoire $Y = X + \epsilon$.

Maintenant, le canal semi-privé peut être difficile à construire (il peut même s'agir d'une personne), de sorte que l'on cherche à utiliser ces canaux aussi peu que possible, en leur préférant des canaux publics simples où seuls les messages sont envoyés. Seules les clés sont partagées par les canaux semi-privés, comme illustré dans la fig. 8.

Comme chaque canal est soumis à du bruit, imaginons qu'Alice et Bob partagent des chaînes de bits aléatoires corrélées X et Y, et que la corrélation avec une autre chaîne de bits aléatoires Z, qui appartient à Eve, est limitée. Des protocoles existent qui permettent la transmission fidèle d'informations entre Alice et Bob, tout en limitant l'information à laquelle Eve a accès. Mais que signifie limiter l'information d'Eve ? Pour comprendre cela, nous devons au moins savoir ce qu'est l'entropie et l'information mutuelle entre variables aléatoires.

Shannon entropy and mutual information

L'entropie de Shannon [15][3] d'une variable aléatoire X, qui pourrait être un message composé de 7 symboles, est définie comme

$$H(X) = H(p_1, \ldots, p_n) = -\sum_x p_x \log p_x, \tag{2.3.1}$$

où les p_x sont les probabilités d'obtenir un mot particulier. Par convention, on comprend que $0 \log 0 = 0$. Une façon utile de penser à l'entropie est de l'interpréter comme l'information acquise en moyenne lorsque la variable aléatoire X devient connue (c'est-à-dire est mesurée). On gagne le plus d'information lorsque la distribution est uniforme, et le moins lorsqu'elle est une delta. L'entropie de deux (ou n) variables aléatoires est définie de manière analogue,

$$H(X, Y) = -\sum_{x,y} p(x, y) \log p(x, y). \tag{2.3.2}$$

L'entropie possède plusieurs propriétés intéressantes, et nous en listons ici deux :

- C'est une fonction lisse des probabilités uniquement.
- Elle est sous-additive, c'est-à-dire $H(X, Y) \leq H(X) + H(Y)$ avec égalité seulement si X et Y sont indépendants. Cela quantifie le fait que les corrélations limitent les résultats possibles et que nous gagnons des informations sur Y en mesurant X et vice versa.

L'information mutuelle entre deux variables aléatoires est définie comme

$$H(X : Y) = H(X) + H(Y) - H(X, Y). \tag{2.3.3}$$

À partir de la propriété sous-additive mentionnée ci-dessus, nous voyons que cela quantifie simplement à quel point les variables aléatoires X et Y sont corrélées, ou combien d'informations elles partagent en commun. Dans le cas extrême où $Y = X$, $H(X, Y) = H(X)$ et $H(X : Y)$ est simplement égal à $H(X)$.

Maintenant, regardons le schéma de la fig. 9. Minimiser l'information d'Eve signifie minimiser l'information mutuelle $H(X : Z)$ et $H(Y : Z)$, et maximiser la fidélité de transmission de l'information signifie maximiser l'information mutuelle $H(X : Y)$. Pour maximiser l'information mutuelle entre Alice et Bob, $H(X : Y)$, qui n'est pas $H(X)$ à cause du bruit dans le canal, nous pouvons utiliser les techniques et méthodes de *correction d'erreurs*. En résumé, la *correction d'erreurs* consiste à utiliser des ressources physiques redondantes (informations redondantes) pour transmettre (communication, algorithmes) ou préserver (mémoires) un sous-ensemble des informations disponibles dans le système physique. L'exemple le plus élémentaire est le code de répétition.

The repetition code

Imaginez qu'Alice souhaite envoyer un bit à travers un canal bruyant, où l'effet du bruit est

[3] L'article est vraiment bon, si vous avez le temps, allez le lire. For the pedagogical introduction see chapter 11 of [1].

d'inverser le bit de 0 à 1 avec une certaine probabilité p. Le code de répétition consiste simplement à envoyer un groupe de bits physiques, tous initialisés à 0 ou 1, de telle sorte que Bob doit effectuer un comptage majoritaire pour décider quel bit Alice a tenté d'envoyer. Si une seule erreur se produit, trois bits suffisent pour une transmission d'information *tolérante aux fautes*.

$$000 \to \{100, 010, 001\}$$

Ainsi, avec l'utilisation de la correction d'erreurs, Alice et Bob transforment tous deux leurs messages correspondants X et Y en un message corrigé W. Dans l'exemple ci-dessus, pour Alice $000 \to 0$ et pour Bob $100 \to 0$ par exemple. Ce processus de correction des erreurs du canal bruyant est connu sous le nom de *réconciliation d'informations*.

L'étape suivante consiste à réduire l'information mutuelle $H(W : Z)$. Ce processus est appelé *amplification de la confidentialité*. Cela peut être réalisé par l'application de *hash functions*. Une fonction de hash g prend n bits et les transforme en m bits, et possède la propriété suivante. Soient x_1 et x_2 deux chaînes de bits de n bits. Alors, $g(x_1) \neq g(x_2)$ avec une probabilité d'au moins $1/2^m$. C'est une fonction très utile car, en un certain sens, elle sépare l'espace des chaînes de bits en blocs presque séparables, et trouve des applications non seulement en cryptographie mais aussi en tomographie quantique [16]. Donner plus de détails dépasse le cadre de ce cours, mais vous devriez maintenant avoir suffisamment de connaissances pour aller lire davantage sur le sujet.

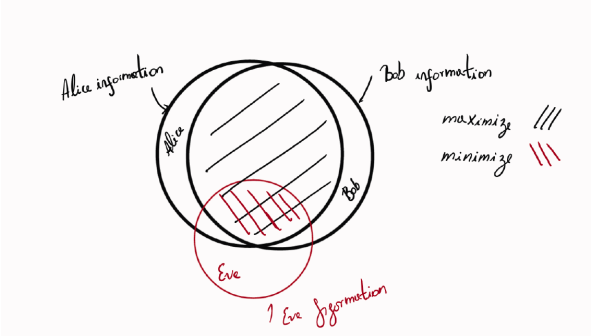

Figure 9: Objectif de la cryptographie

2.3.2 Quantum key distribution

QKD est un protocole *prouvablement* sécurisé, qui permet de créer des clés privées via un canal public. Ensuite, les clés privées peuvent être utilisées comme ci-dessus pour créer un canal de communication classique sécurisé. La condition physique minimale pour que cela soit possible est la capacité de communiquer des qubits via le canal public avec un taux d'erreur en dessous d'un certain seuil. La sécurité d'un tel schéma repose sur deux faits :

- Le théorème de non-clonage, qui est formellement pas unique au quantique, car les ensembles ne peuvent pas non plus être clonés, see [17], you will find further references there.
- Le gain d'information implique une perturbation.

Regardons donc d'abord le théorème de non-clonage.

Théorème de non-clonage

Le théorème de non-clonage stipule qu'il est impossible de réaliser la transformation suivante,

$$|\psi_A\rangle |\psi_B\rangle |\psi_C\rangle \to_{\hat{U}} |\varphi_A\rangle |\psi_B\rangle |\psi_B\rangle . \tag{2.3.4}$$

En d'autres termes, il n'est pas possible de cloner un état quantique arbitraire $|\psi_B\rangle$ via une évolution unitaire. La preuve est simple. Comme le système A représente une ancilla, nous ne pouvons l'ignorer pour capturer la preuve et devons supposer qu'il fait partie de la transformation

unitaire. C'est-à-dire que nous souhaitons prouver qu'il n'existe pas de transformation unitaire $\hat{U}$ telle que

$$|\psi_B\rangle |\psi_C\rangle \rightarrow_{\hat{U}} |\psi_B\rangle |\psi_C\rangle , \tag{2.3.5}$$

pour un état quantique initial inconnu $|\psi_B\rangle$ et tout état $|\psi_C\rangle$ que nous pouvons préparer. Nous pouvons également écrire $U(|\psi\rangle |a\rangle) = |\psi\rangle |\psi\rangle$. Nous allons faire la preuve par contradiction. Supposons qu'un tel unitaire existe. Alors, il fonctionnerait pour deux états particuliers, disons $|\varphi\rangle$ et $|\psi\rangle$, c'est-à-dire

$$\begin{aligned} U(|\varphi\rangle |a\rangle) &= |\varphi\rangle |\varphi\rangle \\ U(|\psi\rangle |a\rangle) &= |\psi\rangle |\psi\rangle . \end{aligned} \tag{2.3.6}$$

En prenant le produit scalaire des deux équations, nous obtenons simplement,

$$\langle\psi|\varphi\rangle = (\langle\psi|\varphi\rangle)^2, \tag{2.3.7}$$

ce qui signifie que $|\psi\rangle$ et $|\varphi\rangle$ sont soit orthogonaux, soit égaux, une contradiction. Par conséquent, nous concluons qu'un tel $\hat{U}$ n'existe pas pour des états arbitraires, mais il peut certainement exister si nous ne nous soucions que d'un ensemble orthonormal d'états (c'est-à-dire une base particulière de l'espace). Il est curieux de voir comment nous pouvons cloner un ensemble de base, c'est-à-dire cloner chaque ket de la base, mais pas leurs combinaisons linéaires (superpositions).

Le gain d'information implique une perturbation

Plus concrètement, toute tentative de distinguer entre deux états quantiques non orthogonaux entraîne un gain d'information uniquement au prix d'une perturbation du signal. Prouver cette affirmation est également élémentaire. Supposons qu'Alice communique via un canal public avec Bob, où elle enverra des états quantiques. Prenons deux de ces états possibles qu'elle peut communiquer et qui sont non orthogonaux, disons $|\psi\rangle$ et $|\varphi\rangle$. Maintenant, Eve souhaite obtenir des informations sur ces états. Naturellement, elle possède également son propre état de registre, disons $|z\rangle$. Obtenir des informations sans perturber les états signifie effectuer la transformation suivante,

$$|\psi\rangle |z\rangle \rightarrow |\psi\rangle |v\rangle \tag{2.3.8}$$

$$|\varphi\rangle |z\rangle \rightarrow |\varphi\rangle |v'\rangle , \tag{2.3.9}$$

où évidemment pour qu'Eve puisse obtenir des informations, il doit être vrai que $|v\rangle \neq |v'\rangle$. Maintenant, toutes ces transformations doivent nécessairement être unitaires (rappelez-vous, pour obtenir une dynamique non unitaire, nous devons retracer certaines parties du système, mais si nous ne les retraçons pas, toutes les transformations sont unitaires), ce qui signifie que le produit scalaire est préservé. En prenant le produit scalaire de chaque ligne ci-dessus, nous obtenons,

$$\langle\psi|\varphi\rangle = \langle\psi|\varphi\rangle \langle v|v'\rangle \implies \langle v|v'\rangle = 1. \tag{2.3.10}$$

Ainsi, il est impossible d'obtenir des informations sans perturber l'état.

Ainsi, Alice exploite ce fait et transmet à Bob des états non orthogonaux. En vérifiant les perturbations dans les états transmis, Alice et Bob peuvent établir des limites supérieures sur le bruit ou l'espionnage se produisant dans leur canal de communication. Naturellement, tout comme dans le cas classique, Alice et Bob effectueront une réconciliation de l'information (possiblement en utilisant la correction d'erreurs), et une amplification de la confidentialité (peut-être en utilisant des fonctions de hachage). Examinons maintenant un exemple particulier.

BB84

Alice commence par générer deux chaînes de bits aléatoires, a et b, chacune composée de $(4+\delta)n$ bits. Elle encode ensuite ces chaînes en $(4+\delta)n$ qubits,

$$|\psi\rangle = \otimes_{k=1}^{(4+\delta)n} |\psi_{a_k b_k}\rangle , \tag{2.3.11}$$

où

$$|\psi_{00}\rangle = |0\rangle\,, \tag{2.3.12}$$
$$|\psi_{10}\rangle = |1\rangle\,, \tag{2.3.13}$$
$$|\psi_{01}\rangle = |+\rangle\,, \tag{2.3.14}$$
$$|\psi_{11}\rangle = |-\rangle\,. \tag{2.3.15}$$

En d'autres termes, les bits b déterminent la base dans laquelle le bit a est encodé, soit Z soit X. Si des photons sont utilisés pour transmettre l'information, cela correspond à l'utilisation de la polarisation horizontale et verticale comme bases, ou de la base de polarisation inclinée à 45 degrés, voir fig. 10.

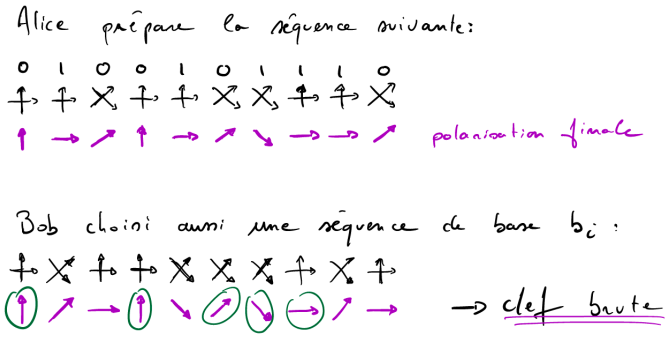

Figure 10: Alice envoie un ensemble de qubits encodés dans deux ensembles de polarisation différents. Mathieu's notes.

Concentrons-nous d'abord sur le protocole sans l'interférence d'Eve, nous discuterons de ce qui se passe lorsqu'elle intervient plus tard. Voir fig. 11 pour un diagramme. Lorsque Bob décide de mesurer, il ne connaît pas la base, mais il génère simplement une chaîne de bits aléatoires b' pour servir de base. Comme $b \neq b'$, tous les bits ne seront pas utiles à la fin, c'est pourquoi des bits redondants δn sont utilisés. Ensuite, Bob annonce qu'il a reçu l'état, et ce n'est qu'alors qu'Alice rend public la base b avec laquelle elle a envoyé les qubits (photons dans ce cas). Avec cette information, Bob peut indiquer à Alice quels bits il a mesurés dans la bonne base. En rendant δ suffisamment grand, il est presque garanti qu'au moins $2n$ mesures ont été correctes. Pour vérifier la présence d'espions, Alice sélectionne n bits aléatoires parmi les $2n$ restants et indique à Bob quels n bits de contrôle elle a choisis. Ensuite, tous deux peuvent rendre publics ces n bits. Si le nombre de bits différents E est supérieur à un certain seuil t, ils abandonnent le protocole. Sinon, ils passent à la distillation de l'information et à l'amplification de la confidentialité avec les n bits secrets restants. Dans fig. 10, la chaîne de bits partagée par Bob et Alice est celle entourée de vert, c'est-à-dire 00011.

Que se passe-t-il lorsque Eve espionne ? Si elle mesure les qubits (photons), elle doit choisir une base pour le faire, et ce choix sera incorrect en moyenne 1/2 du temps, voir fig. 12. La mesure fait s'effondrer le qubit (photon) sur un des états propres de la base choisie (polarisation). Ainsi, lorsque Bob effectue sa mesure, la probabilité qu'il obtienne un bit incorrect, aux positions où Eve a fait un choix de base incorrect, est de 1/2. Par conséquent, la probabilité totale par bit que Bob ait une erreur est $(1/2)^2 = 1/4$. La probabilité que Bob n'ait aucune erreur si Eve a mesuré n bits est $(1/4)^n$. Donc, si un nombre suffisamment grand de bits est utilisé, si Eve essaie de mesurer beaucoup d'entre eux, l'espionnage sera presque certainement détecté. Pour 72 bits, la probabilité de ne pas détecter une erreur est d'environ 10^{-9}, en d'autres termes, en moyenne, dans un cas sur un milliard, Eve ne serait pas détectée. Si Eve choisit de n'en mesurer que quelques-uns, elle n'aura encore qu'une information marginale sur la clé résultante.

Generic drawbacks

Comment les protocoles QKD sont-ils vulnérables aux attaques ? Certains types d'attaques sont les mêmes que pour les protocoles de clé privée classiques, notamment :

- Attaque de l'homme du milieu : une personne usurpe l'identité de Bob.

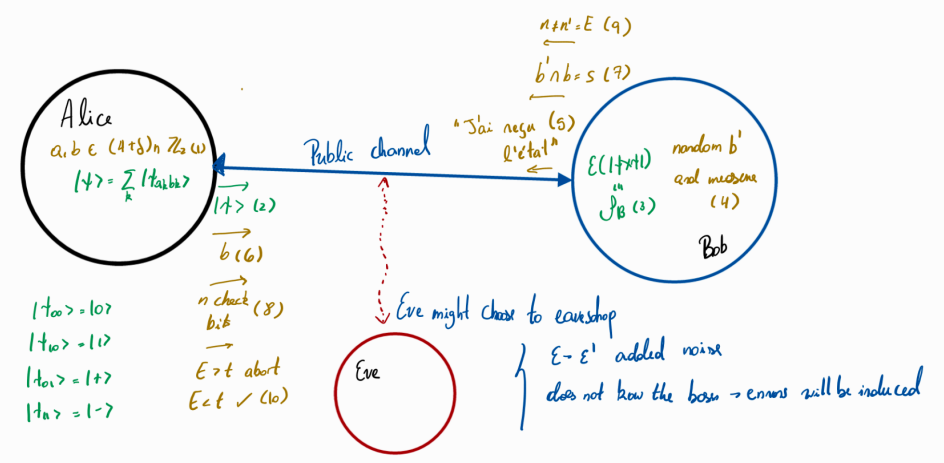

Figure 11: Diagramme pour BB84. Les chiffres entre parenthèses représentent les étapes temporelles du protocole.

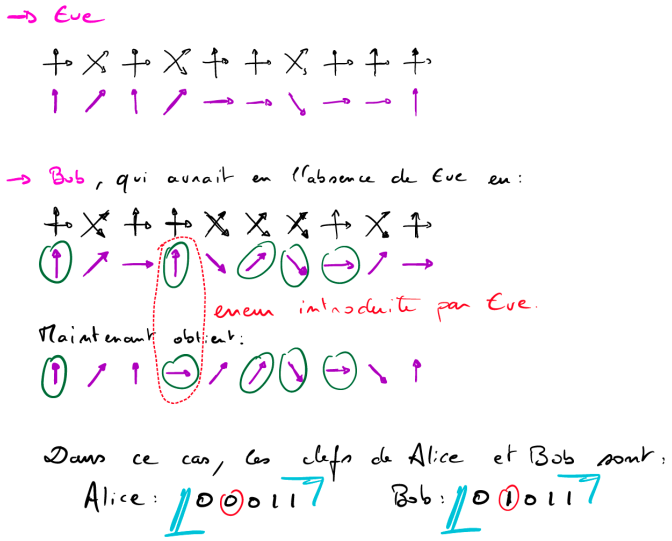

Figure 12: Que se passe-t-il lorsque Eve espionne avec une mesure ?

- Coupure des communications (littéralement couper le câble de fibre optique, par exemple, ce qui n'est pas très subtil).
- Cheval de Troie. Cela consiste à envoyer un signal qui exploite une fonction non idéale de l'équipement utilisé par Alice et Bob. Par exemple, les détecteurs de photons uniques peuvent générer des « post-impulsions », c'est-à-dire des impulsions après la détection, surtout si le signal entrant était fort. Eve peut donc essayer d'exploiter cela en envoyant à Bob une impulsion forte, puis en mesurant la post-impulsion qui revient, afin d'obtenir des informations sur la base que Bob utilise pour mesurer. Cependant, si Bob change constamment de base au hasard, je ne suis pas sûr de l'efficacité de cette méthode pour Eve. Quoi qu'il en soit, c'est un sujet à aborder lorsque l'on travaille vraiment dessus.

Un autre type d'attaque pourrait exploiter le fait que, si des impulsions laser atténuées sont utilisées pour générer les photons, lorsque l'objectif est d'envoyer un seul photon, il existe une probabilité d'en envoyer deux au lieu d'un, ouvrant ainsi une fenêtre pour qu'Eve en stocke un.

People keeps studying this, see for example [18].

References

[1] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge university press, 2010.

[2] Jun John Sakurai and Jim Napolitano. *Modern quantum mechanics*. Cambridge University Press, 2020.

[3] *python.org*. https://www.python.org/downloads/. [Accessed 16-09-2024].

[4] *Getting started with Visual Studio Code — code.visualstudio.com*. https://code.visualstudio.com/docs/introvideos/basics. [Accessed 16-09-2024].

[5] *GitHub - LautaroLabarcaG/optique-quantique: optique-quantique 2024 UdeS — github.com*. https://github.com/LautaroLabarcaG/optique-quantique/tree/main. [Accessed 16-09-2024].

[6] Christopher M Dawson et al. "Quantum computing and polynomial equations over the finite field Z_2". In: *arXiv preprint quant-ph/0408129* (2004).

[7] Albert Einstein, Boris Podolsky, and Nathan Rosen. "Can quantum-mechanical description of physical reality be considered complete?" In: *Physical review* 47.10 (1935), p. 777.

[8] Alain Aspect, Jean Dalibard, and Gérard Roger. "Experimental test of Bell's inequalities using time-varying analyzers". In: *Physical review letters* 49.25 (1982), p. 1804.

[9] Stuart J Freedman and John F Clauser. "Experimental test of local hidden-variable theories". In: *Physical review letters* 28.14 (1972), p. 938.

[10] John S Bell. "On the einstein podolsky rosen paradox". In: *Physics Physique Fizika* 1.3 (1964), p. 195.

[11] Simon Storz et al. "Loophole-free Bell inequality violation with superconducting circuits". In: *Nature* 617.7960 (2023), pp. 265–270.

[12] Andrea Aiello. *Against Bell's Theorem*. 2024. arXiv: 2406.03028 [quant-ph]. URL: https://arxiv.org/abs/2406.03028.

[13] Akash V Dixit et al. "Searching for dark matter with a superconducting qubit". In: *Physical review letters* 126.14 (2021), p. 141302.

[14] *Topical on quantum gravity tests with atoms*. https://smd-cms.nasa.gov/wp-content/uploads/2023/05/45_e8d91f69e93d0cf59de3959b6bb25b55_BiedermannGrantW.pdf. [Accessed 16-09-2024].

[15] Claude Elwood Shannon. "A mathematical theory of communication". In: *The Bell system technical journal* 27.3 (1948), pp. 379–423.

[16] Jordan Cotler and Frank Wilczek. "Quantum overlapping tomography". In: *Physical review letters* 124.10 (2020), p. 100401.

[17] Flavio Del Santo and Nicolas Gisin. *Which features of quantum physics are not fundamentally quantum but are due to indeterminism?* 2024. arXiv: 2409.10601 [quant-ph]. URL: https://arxiv.org/abs/2409.10601.

[18] Brian Pigott et al. *Eavesdropping on the BB84 Protocol using Phase-Covariant Cloning: Experimental Results*. 2024. arXiv: 2409.16284 [quant-ph]. URL: https://arxiv.org/abs/2409.16284.