République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université de Jijel

Faculté des Sciences et de la Technologie

Département d'Informatique

Mémoire

de fin d'études pour l'obtention du diplôme

de Master en Informatique

Option : *Intelligence Artificielle*

Thème

Correction d'erreur quantique

Encadré par :

Mr KHELFAOUI Khaled

Réalisé par :

Melle KERROUM Manel

Promotion Juin 2018.

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université de Jijel

Faculté des Sciences et de la Technologie

Département d'Informatique

Mémoire

de fin d'études pour l'obtention du diplôme

de Master en Informatique

Option : *Intelligence Artificielle*

Thème

Correction d'erreur quantique

Encadré par :

Mr KHELFAOUI Khaled

Réalisé par :

Melle KERROUM Manel

Promotion Juin 2018.

Remerciements

Le trajet qui m'a amené à cette destination me laisse très sensible à certaines personnes qui méritent quand même un intérêt particulier.

Avant cela je tiens à remercier Dieu le tout puissant de m'avoir donné toute la force et la foi pour en arriver là.

Je remercie mon encadreur Mr KHELFAOUI Khaled pour son assistance et son aide précieuse durant toute la durée de mon travail.

Je remercie très sincèrement et cordialement les membres de jury qui ont accepté l'appartenance à la commission d'évaluation du travail.

Je ne peux pas rester insensible envers nos enseignants qui ont grandement participé durant tout le cycle de ma formation.

Enfin je tiens à remercier toute personne ayant contribué de près ou de loin dans l'élaboration de ce mémoire.

Dédicace

Il m'est impossible de citer tous ceux qui ont contribué à la réalisation de ce travail. De ce fait, je commence par m'en excuser et par leur exprimer mes sincères remerciements.

Arrivé à ce stade, n'est que le fruit du milieu familial qui m'est propice, le fruit de l'équilibre et du sacrifice de mes Parents, Ces êtres chers que Dieu les protège.

Je dédie ce modeste travail à mes chers Parents qui ont été de tout temps, les plus proches, qui n'ont jamais ménagé leurs efforts, leurs encouragements et leur soutien avec abnégation et patience.

A mes chères sœurs Karima, Imane et Selma et mes chers frères Oussama, Yahia, Khaled et sa femme Abir.

A mes Oncles, Tantes et cousins.

En ce moment je ne peux oublier à l'ensemble des amis que j'ai connu pendant mes études et à ceux qui ont prodigué leurs vifs conseils, encouragements et témoigné de leur amitié.

Manel

Résumé

L'informatique quantique est un sous-domaine de l'informatique basé sur les principes de la mécanique quantique. Dans ce mode de calcul, les erreurs sont principalement dues à l'interaction du système quantique avec son environnement. Plutôt que de tenter d'affronter de face cet obstacle inévitable, il est possible d'utiliser des codes correcteurs.

Le but de ce projet est l'étude et l'implémentation de l'algorithme de Shor. Plusieurs exemples d'application sont présentés.

Abstract

Quantum computing is a subdomain of computer science based on principles of quantum mechanics. In this mode of calculation, the errors are mainly due to the interaction of the quantum system with its environment. Rather than trying to face this unavoidable obstacle, it is possible to use correcting codes.

The goal of this project is the study and implementation of the Shor algorithm. Several application examples are presented.

Table Des Matières

Liste Des Figures :

Chapitre 2 :

Chapitre 3 :

Chapitre 4 :

Liste Des Tableaux

Introduction Générale

Due au à la miniaturisation progressive des composants électroniques de l'ordinateur, Richard Feynman, physicien Américain et prix de Nobel, a observé que l'on arrivera à un monde microscopique où les lois de la physique de notre propre échelle ne peuvent plus s'appliquer mais celles de la physique quantique qui s'imposeront [Feynman, 1984]. Il a proposé d'intégrer la physique quantique dans la théorie de l'information et l'informatique et de l'utiliser comme support matériel du calcul. Cette intégration permettrait de tirer profit des phénomènes étranges et surprenants de la physique quantique tels que les superpositions d'états, l'intrication et l'interférence et qui, bien qu'elles semblent défier la logique et le bon sens, offrent des opportunités de calcul infiniment plus rapide que celui d'un calculateur classique.

L'idée de base consiste à réaliser un ordinateur quantique analogue à l'ordinateur classique (machine de Turing) : l'unité d'information classique ''bit '' est remplacé par le '' bit quantique'', les circuits et les portes logiques par des circuits et porte quantiques, et même l'aspect du calcul basé sur les fonctions booléennes est remplacé par un calcul spécifique basé sur les opérations algébrique linéaires.

Dans ce mode calcul, les erreurs sont principalement dues à l'interaction du système quantique avec son environnement. Plutôt que de tenter d'affronter de face cet obstacle inévitable, il est possible d'utiliser des techniques de correction. Généralement, les spécialistes du domaine proposent des adaptations des correcteurs classiques en tirant profit de l'intrication afin de délocaliser sur plusieurs systèmes physiques l'information encodée.

L'objectif de ce travail est d'étudier la correction d'erreurs quantique basée sur la redondance. Plus précisément, il s'agit d'une implémentation de l'algorithme de Shor. C'est une solution qui nécessite l'utilisation huit Qubits supplémentaires. Elle permet la détection et la correction des erreurs de type X, Y et Z. Ce manuscrit est constitué de quatre chapitres :

- Chapitre 1 : Ce chapitre est un résumé des notions mathématiques nécessaires à la compréhension du calcul quantique.
- Chapitre 2 : Dans ce chapitre, nous introduisons les circuits quantiques en donnant un panorama des portes quantiques.
- Chapitre 3 : Ce chapitre est dédié à la présentation de l'algorithme de Shor.
- Chapitre 4 : Dans ce chapitre, nous présentons une implémentation de cet algorithme. Afin de valider notre travail, des exemples d'applications seront présentés.

Enfin, nous terminons par une conclusion générale qui résume l'apport de notre travail et présente quelques perspectives.

Chapitre 1

Concepts de Base :

- Introduction
- Préliminaire
- Qubit
- Intrication quantique
- Conclusion

1. Introduction :

Dans le sens mathématique, la mécanique quantique est une théorie dirigée par un ensemble d'axiomes. Ces derniers constituent des règles qui guident les comportements des systèmes quantiques. Ce chapitre est dédié à la présentation de quelques concepts principaux de cette théorie. Ils nous seront utiles tout au long de ce mémoire.

Comme nous sommes confrontés à un domaine complètement nouveau pour un informaticien, nous commençons d'abord par courte présentation des notions mathématiques nécessaires. Nous tentons ensuite d'expliquer les postulats de la mécanique quantique tout en gardant notre point de vue en tant que des informaticiens. Finalement, nous introduisons quelques phénomènes quantiques qui sont à la base du calcul quantique.

2. Les préliminaires mathématiques

2.1. Espace de Hilbert

Un espace de Hilbert H est un espace vectoriel complexe muni d'un produit scalaire [1], [2]. Dans cet espace, tout système physique isolé est entièrement décrit par un vecteur d'état de norme 1. Il peut subir plusieurs manipulations provoquant des changements d'état. Pour les étudier, les spécialistes utilisent plusieurs notations. La plus efficace est celle proposée par Dirac.

2.2. Notation de Dirac

En mécanique quantique, la notation de Dirac est utilisée afin de faciliter l'écriture des équations[3]. Elle est encore appelée notation ''Bra-Ket'' tel que :

- Bra : est un vecteur ligne.
- Ket : est un vecteur colonne.

- **Vecteur de Ket :**

Dans la notation de Dirac, un état ψ est décrit par une matrice colonne appelée Ket [4] et notée $|\psi\rangle$ tel que :

$$|\psi\rangle = \sum_{i=1}^{n} a_i \, |i\rangle = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} \qquad 1.1$$

$a_i \in \mathbb{C}$

- **Vecteur Bra :**

La matrice ligne obtenue conjugaison complexe des éléments de $|\psi\rangle$ est appelée Bra [4]. Elle est notée $\langle\psi|$ et calculée comme suit :

$$\langle\psi| = \sum_{i=1}^{n} a_i^* \langle i| = (a_1^*, a_2^*, \dots a_n^*) . \qquad 1.2$$

2.3. Produit scalaire Bra_ket

Etant donné un vecteur Bra $\langle\phi|$ et un vecteur Ket $|\psi\rangle$:

$$|\psi\rangle = \sum_{i=1}^{n} a_i \, |i\rangle \qquad 1.3$$

$$\langle\phi| = \sum_{i=1}^{n} b_i^* \langle i| \qquad 1.4$$

Le produit de ces deux vecteurs dans cet ordre noté $\langle\phi|\psi\rangle$ est appelée produit scalaire Bra-ket [5]. C'est un nombre complexe donné par l'équation suivante :

$$\langle \phi | \psi \rangle = \sum_{i=1}^{n} a_i b_i^* \langle i|i\rangle \qquad 1.5$$

2.4. Produit tensorielle

En mécanique quantique, le produit tensoriel est un opérateur très intéressant. Il est utilisé pour combiner deux espaces de vecteur vers un autre plus large. Etant donné deux vecteurs Ket $|\psi\rangle$ et $|\phi\rangle$ tel que :

$$|\psi\rangle=\sum_{i=1}^{n} a_i\ |i\rangle \qquad 1.6$$

$$\langle\phi|=\sum_{i=1}^{n} b_i^*\ \langle i| \qquad 1.7$$

Le produit tensoriel de ces deux vecteurs dans cet ordre noté $|\psi\rangle\otimes|\phi\rangle$ est calculé comme suit :

$$|\psi\rangle\otimes|\phi\rangle = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} \otimes \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix} = \begin{pmatrix} a_1\begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \\ a_2\begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \\ \vdots \\ a_n\begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \end{pmatrix} = \begin{pmatrix} a_1.b_1 \\ a_1.b_2 \\ \vdots \\ a_1.b_m \\ a_2.b_1 \\ \vdots \\ a_{n-1}.b_m \\ a_n.b_1 \\ \vdots \\ a_n b_m \end{pmatrix} \qquad 1.8$$

3. Qubit

En informatique classique, l'information la plus élémentaire dans un ordinateur est le bit (Bynary digit). Il peut ne prendre que deux valeurs possible : 0 ou 1. Par analogie, l'élément de base de l'informatique quantique est appelé ''bit quantique'' ou Qubit [6]. C'est un système décrit dans une base constituée par les deux états fondamentaux 0 et 1 notée $\{|0\rangle, |1\rangle\}$. Donc, un Qbit peut se trouver dans une superposition d'états possibles entre $|0\rangle$ et $|1\rangle$.

Le tableau suivant donne une petite comparaison entre les bits classiques et les bits quantiques [7].

Bit Classique	Bit Quantique
Un bit a toujours une valeur définie.	Pas de valeur définie pour le qubit tant qu'on ne l'observe pas.
Un bit vaut seulement 0 ou 1.	Un qubit peut être dans une superposition de 0 et 1 simultanément.
Un bit peut être copié sans être affecté.	Un qubit dans un état inconnu ne peut être copié
Un bit peut être lu sans affecter sa valeur.	Lire un qubit qui est initialement dans une superposition changera sa valeur.
Lire un bit n'affecte pas un autre	La lecture d'un qubit peut avoir influence sur les autres qubits

Tab 1 .1 : Comparaison de qubit quantique VS le bit classique

3.1. Postulats de la mécanique quantique

3.1.1. Postulat 01 : L'état

Soit H_N l'espace de Hilbert de dimension n. En calcul quantique, tout système physique (ou du moins une représentation abstraite d'un système physique) est décrit par un vecteur d'état (Ket) $|\psi>$ normalisé dans H_N [8].

Un qubit $|\phi>$ est un système physique élémentaire dont l'espace d'état est de dimension deux [8]. Il est écrit dans la base canonique $\{|0>, |1>\}$ comme suit :

$$|\phi\rangle = \alpha|0\rangle + \beta|1\rangle \quad \textbf{1.9}$$

Avec

$$\alpha, \beta \in \mathbb{C} \quad \text{et} \quad |\alpha|^2 + |\beta|^2 = 1 \quad \textbf{1.10}$$

L'espace de Hilbert d'un qubit est noté $H_{\{0,1\}}$, ou simplement $\mathbb{C}^2$.

Un registre de n qubits est un vecteur normé $|\psi\rangle$ de $H_{\{0,1\}^n}$ ou encore $\mathbb{C}^{2^n}$. Son état est donné par la formule suivante :

$$|\psi\rangle = \sum_{x\in\{0,1\}^n} \alpha_x |x\rangle \quad \textbf{1.11}$$

Avec

$$\sum_{x\in\{0,1\}^n} |\alpha_x|^2 = 1 \quad \textbf{1.12}$$

3.1.2. Postulat 02 : Evolution

Définition:

Si le transposé conjugué d'une matrice U est égal à l'inverse de la matrice, alors elle est dite unitaire. Notons le transposé conjugué de U par $U^\dagger$ [4] , donc :

$$U^\dagger U = I \quad \textbf{1.13}$$

L'évolution dans le temps d'un système quantique $|\psi\rangle$ de taille n vers un nouveau système $|\psi'\rangle$ est réalisée par application d'un opérateur unitaire U, tel que :

$$|\psi'\rangle = U|\psi\rangle \quad \textbf{1.14}$$

- U est une matrice unitaire des nombres complexes d'une dimension n × n
- L'importance de l'unitarité de U réside dans le fait qu'elle préserve les propriétés de ce produit y compris la normalisation des vecteurs d'états. En plus, Elle implique la réversibilité qui permet de déterminer l'état du vecteur d'entrée étant donné l'état de sortie et l'information sur la nature d'évolution.

3.1.3. Postulat 03 : mesure

Soit $|\psi\rangle = \sum_{i=1}^{n} a_i |i\rangle$ un état quantique de taille n. Selon les définitions précédentes, il peut se trouver dans l'état i avec une probabilité :

$$p(i) = |a_i|^2 \qquad 1.15$$

La mesure de $|\psi\rangle$ donnant l'état $|i\rangle$ est définie par une matrice de transformation notée M_i . Le nouveau état après mesure est donné par la formule suivante [8]:

$$\frac{M_i|\psi\rangle}{\sqrt{\langle\psi| M_i^+ M_i |\psi\rangle}} \qquad 1.16$$

- Tel que : $$p(i) = \langle\psi| M_i^+ M_i |\psi\rangle = |a_i|^2 \qquad 1.17$$

➢ **Remarque :**

- La mesure est un analogue d'une projection d'un vecteur sur une base.
- Avant d'effectuer une mesure sur un système se trouvant dans une superposition, on ne peut pas savoir à priori avec certitude que sera le résultat de la mesure mais on ne peut connaître que la probabilité associée à chacun des états de base.

3.1.4. Postulat 04 : Système composé

Etant donné deux sous-systèmes a et b spécifiés par deux kets $|\psi\rangle$ et $|\phi\rangle$ des deux espaces d'état H^a et H^b .

$$|\psi\rangle=\sum_i a_i|i\rangle \qquad 1.18$$

$$|\phi\rangle=\sum_j b_j|j\rangle \qquad 1.19$$

Le système physique composite est un nouveau état Ket [8][9] de l'espace :

$$H^{ab} = H^a \otimes H^b. \qquad 1.20$$

Il est noté $|\psi\rangle\otimes|\phi\rangle$ et donné par la formule suivante :

$$|\psi'\rangle = |\psi\rangle\otimes|\varphi\rangle=\sum\sum a_i\, b_j\, |i\rangle\otimes|j\rangle \qquad 1.21$$

Ce calcul est basé principalement sur le produit tensoriel. Il est à noter que ce dernier est caractérisé par les axiomes suivants:[9]

- Pour tout $|\psi_1\rangle$,$|\phi_1\rangle \in H_1$ et $|\psi_2\rangle \in H_2$:

$$(|\psi_1\rangle+|\phi_1\rangle)\otimes|\psi_2\rangle= |\psi_1\rangle\otimes|\psi_2\rangle+|\phi_1\rangle\otimes|\psi_2\rangle \qquad 1.22$$

- Pour tout $|\psi_1\rangle \in H_1$ et $|\phi_2\rangle, |\psi_2\rangle \in H_2$:

$$|\psi_1\rangle\otimes(|\phi_2\rangle+|\psi_2\rangle)= |\psi_1\rangle\otimes|\phi_2\rangle+|\psi_1\rangle\otimes|\psi_2\rangle \qquad 1.23$$

Exemples:

$$|0\rangle\otimes\frac{|0\rangle-|1\rangle}{\sqrt{2}}=\frac{|0\rangle\otimes|0\rangle-|0\rangle\otimes|1\rangle}{\sqrt{2}}=\frac{|00\rangle-|01\rangle}{\sqrt{2}}$$

$$\frac{|0\rangle+i|1\rangle}{\sqrt{2}}\otimes\frac{|00\rangle-i|10\rangle}{\sqrt{2}}=\frac{|000\rangle-i|010\rangle+i|100\rangle+|100\rangle}{2}$$

4. Intrication quantique :

En mécanique quantique, l'intrication quantique, ou enchevêtrement quantique, est un phénomène dans lequel deux particules (ou groupes de particules) ont des états quantiques dépendant l'un de l'autre quelle que soit la distance qui les sépare.

Soit un système S décrit par l'état $|\psi>$ et formé de deux sous système S_1et S_2 . On dit que l'état $|\psi>$ est intriqué s'il n'existe pas de vecteurs d'états $|\phi_1\rangle$ et $|\phi_2\rangle$ décrivant les sous systèmes S_1 et S_2 de telle sorte que l'état global $|\psi\rangle$ s'écrive comme produit tensoriel de ces deux états : [10]

$$\boxed{|\psi\rangle \neq |\phi_1\rangle \otimes |\phi_2\rangle \ .} \qquad \textbf{1.24}$$

Dans le cas où un tel produit existe on dit que l'état est séparable.

Le prototype des états intriqués est constitué des états de Bell [11]. Ceux-ci forment une base orthonormée de $C^2 \otimes C^2$.

$$|\psi_{00}\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

$$|\psi_{10}\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle) \qquad \textbf{1.25}$$

$$|\psi_{01}\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)$$

$$|\psi_{11}\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)$$

5. Conclusion :

Ce chapitre était dédié à la présentation des principes du calcul quantique. Nous avons donné un aperçu sur le formalisme mathématique ainsi que les postulats lois contrôlant cette théorie. On a constaté que c'est un domaine spécifique caractérisé par quelques phénomènes très étranges par rapport au calcul classique. Les plus importants sont la superposition, l'intrication et la mesure d'états.

- la superposition: Un état quantique est une combinaison linéaire d'un ensemble états fondamentaux.
- L'intrication: Deux états intriqués constituent une entité indissociable. La transformation d'un état influence l'état de l'autre.
- La mesure : La mesure d'un état superposé d'un ensemble états fondamentaux est une projection sur l'un de ces états. C'est une opération irréversible.

Des physiciens, des informaticiens et des mathématiciens ont montré que ces phénomènes, tels qu'ils sont formulés par la mécanique quantique, peuvent être exploités pour représenter et traiter l'information

L'intrication est une ressource importante dans le traitement quantique de l'information. Elle joue un rôle important dans plusieurs protocoles pour la communication et la cryptographie quantique.

Chapitre 2

Calcule quantique :

- Introduction
- Portes quantiques
- Circuit quantique
- Conclusion

1. Introduction

Dans un traitement quantique, les changements que peut subir un état sont réalisés par des circuits quantiques. C'est une série de portes appliquées dans un ordre bien défini. Une porte quantique est une opération unitaire agissant sur un ou plusieurs Qubits. Le calcul quantique est donc réversible par construction.

Il existe un nombre infini de portes quantiques, qui peuvent toutes s'obtenir par combinaisons de quelques portes élémentaires, constituant un ensemble universel de portes quantiques.

Dans ce chapitre nous introduisons les portes les plus intéressantes en donnant leurs matrices de transformation, leurs représentations graphiques ainsi que les différents types de composition possibles.

2. Porte quantique :

2 .1 . Les portes unaires :

Ces portes agissent sur un seul Qbit. Elles sont caractérisées par des matrices carrées d'ordre 2 [12].

- **Porte Not :**

La porte X est l'équivalent du Not quantique. Sa matrice est donnée comme suit :

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \qquad 2.1$$

Sa table de vérité est la suivante :

Entrée	Sortie
$\|0\rangle$	$\|1\rangle$
$\|1\rangle$	$\|0\rangle$
$\alpha\|0\rangle + \beta\|1\rangle$	$\beta\|0\rangle + \alpha\|1\rangle$

Tab 2.1 : Table de vérité de la porte X

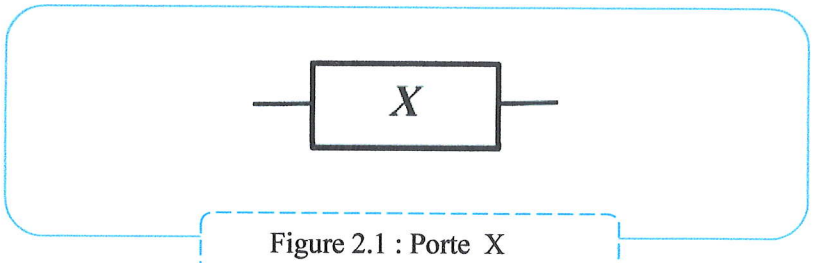


Figure 2.1 : Porte X

- **Porte Y:**

La porte Y peut être décrite sous la forme de la matrice suivante :

$$Y \equiv \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \qquad 2.2$$

La table qui suit donne la table de vérité de la porte Y:

Entrée	Sortie
$\|0\rangle$	$i\|1\rangle$
$\|1\rangle$	$-i\|0\rangle$
$\alpha\|0\rangle + \beta\|1\rangle$	$-\beta i\|0\rangle + \alpha i\|1\rangle$

Tab 2.2 : Table de vérité de la porte **Y**

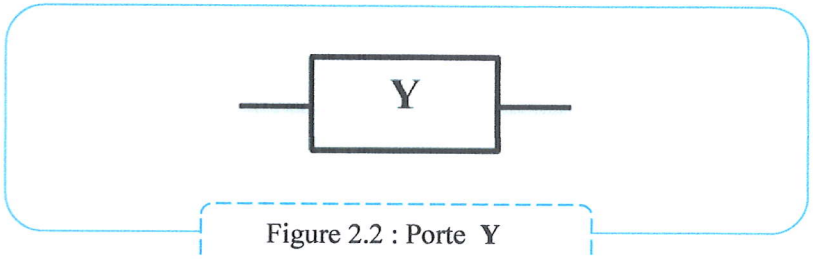


Figure 2.2 : Porte **Y**

- **Les Portes Phases :**

La porte Z est donné par la matrice :

$$Z \equiv \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \qquad 2.3$$

Sa table de vérité est la suivante :

Entrée	Sortie
$\|0\rangle$	$\|0\rangle$
$\|1\rangle$	$-\|1\rangle$
$\alpha\|0\rangle + \beta\|1\rangle$	$\alpha\|0\rangle - \beta\|1\rangle$

Tab 2.3 : Table de vérité de la porte **Z**

Figure 2.3 : Porte **Z**

Remarque : Ces trois portes X, Y et Z sont connue par les matrices de Pauli

- **La porte de Hadamard :**

$$H \equiv \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad 2.4$$

Par la suite sa table de vérité sur la base de calcul est la suivante :

Entrée	Sortie
$\|0\rangle$	$\frac{1}{\sqrt{2}}(\|0\rangle + \|1\rangle)$
$\|1\rangle$	$\frac{1}{\sqrt{2}}(\|0\rangle - \|1\rangle)$
$\alpha\|0\rangle + \beta\|1\rangle$	$\frac{\alpha+\beta}{\sqrt{2}}\|0\rangle + \frac{\alpha-\beta}{\sqrt{2}}\|1\rangle$

Tab 2.4 : Table de vérité de la porte Hadamard

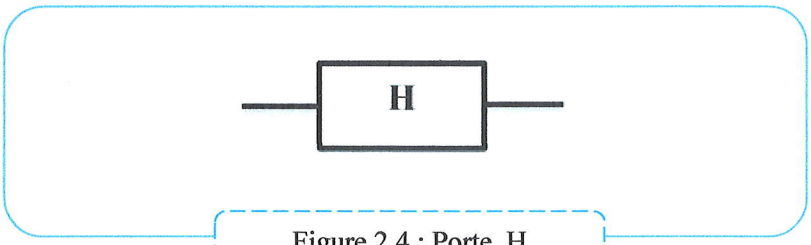


Figure 2.4 : Porte H

- **Les portes de rotation :**

Un Qubit est représenté géométriquement sur une surface d'une sphère dite sphère de Bloch. Cette représentation nous permet de d'appliquer des transformations de rotation basées sur les fonctions cos et sin. Il s'agit de ces matrices rotationnelles :

$$R_x(\gamma) = \begin{bmatrix} \cos\left(\frac{\gamma}{2}\right) & -isin\left(\frac{\gamma}{2}\right) \\ -isin\left(\frac{\gamma}{2}\right) & \cos\left(\frac{\gamma}{2}\right) \end{bmatrix} \quad \textbf{2.5}$$

$$R_y(\gamma) = \begin{bmatrix} \cos\left(\frac{\gamma}{2}\right) & -sin\left(\frac{\gamma}{2}\right) \\ sin\left(\frac{\gamma}{2}\right) & \cos\left(\frac{\gamma}{2}\right) \end{bmatrix} \quad \textbf{2.6}$$

$$R_z(\gamma) = \begin{bmatrix} e^{-i\frac{\gamma}{2}} & 0 \\ 0 & e^{i\frac{\gamma}{2}} \end{bmatrix} \quad \textbf{2.7}$$

$$Ph(\gamma) = \begin{bmatrix} e^{i\gamma} & 0 \\ 0 & e^{i\gamma} \end{bmatrix} \quad \textbf{2.8}$$

2.2. Les portes multi-Qubits :

- **La porte SWAP :**

Cette porte permet la permutation des positions de deux Qbits. Elle est donnée par la matrice suivante :

$$SWAP \equiv \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad 2.9$$

Entrée	Sortie
$\|00\rangle$	$\|00\rangle$
$\|01\rangle$	$\|10\rangle$
$\|10\rangle$	$\|01\rangle$
$\|11\rangle$	$\|11\rangle$
$\alpha\|00\rangle + \beta\|01\rangle + \gamma\|10\rangle + \delta\|11\rangle$	$\alpha\|00\rangle + \gamma\|01\rangle + \beta\|10\rangle + \delta\|11\rangle$

Tab 2.5 : Table de vérité de la porte SWAP

Sa représentation graphique est la suivante :

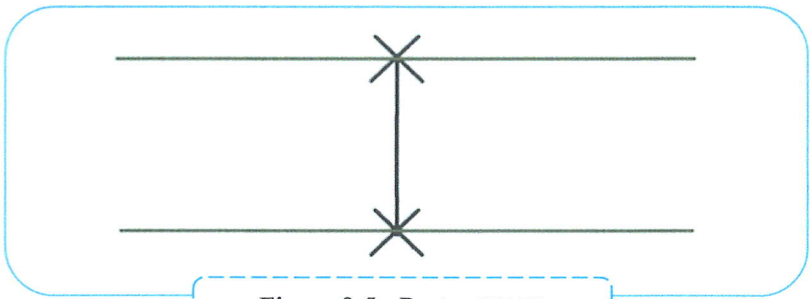


Figure 2.5 : Porte SWAP

- **Portes contrôlées :**

Ces portes agissent généralement sur plusieurs Qubits. Un Qbit de contrôle et des Qubits cibles.

- Si la valeur du Qbit de contrôle satisfait une certaine condition, on applique un transformation sur les Qubits cibles.
- Sinon, rien à faire.

Il est à noter que le Qbit de contrôle reste invariant. La représentation générale d'une porte contrôlée est :

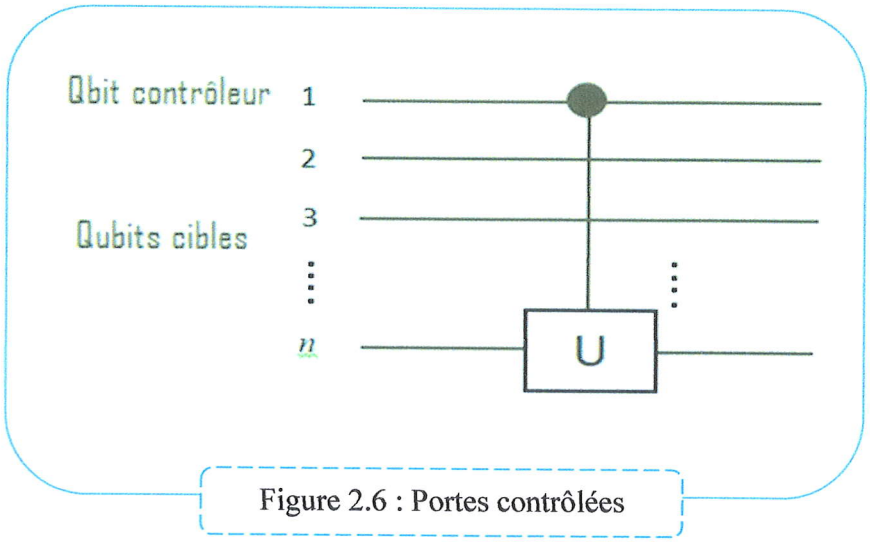
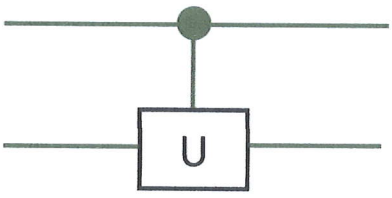


Figure 2.6 : Portes contrôlées

Pour n=2, la matrice de transformation correspondante à une porte contrôlée est donnée par la relation suivante :



$$C_U = I \oplus U$$

- $\oplus$: L'operateur somme directe.

- **Le Not contrôlé (CNot) :**

Le CNOT est un cas particulier des portes contrôlée. C'est l'analogue quantique de la porte XOR classique. Il s'agit d'une transformation X contrôlée :

$$CNOT \equiv I \oplus X \equiv \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad \textbf{2.10}$$

Entrée	Sortie
$\|00\rangle$	$\|00\rangle$
$\|01\rangle$	$\|01\rangle$
$\|10\rangle$	$\|11\rangle$
$\|11\rangle$	$\|10\rangle$
$\boldsymbol{\alpha\|00\rangle + \beta\|01\rangle + \gamma\|10\rangle + \delta\|11\rangle}$	$\alpha\|00\rangle + \beta\|01\rangle + \delta\|10\rangle + \gamma\|11\rangle$

Tab 2.6 : Table de vérité de la porte CNOT

Sa représentation graphique est la suivante :

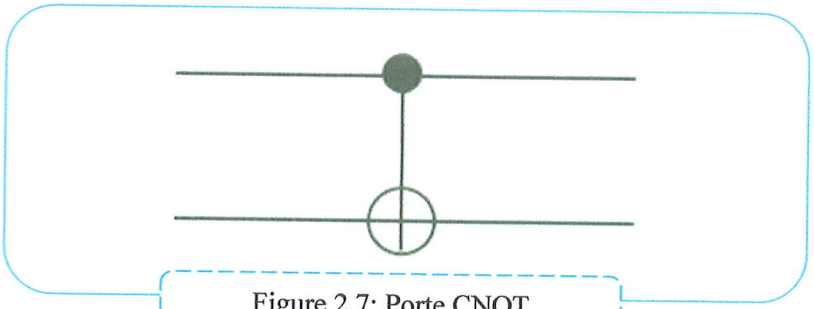


Figure 2.7: Porte CNOT

- **La porte Toffoli :**

Il s'agit d'une porte X contrôlée par deux Qubits de contrôle.

$$TOFFOLI \equiv \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad \textbf{2.11}$$

Entrée	Sortie
$\|000\rangle$	$\|000\rangle$
$\|001\rangle$	$\|001\rangle$
$\|010\rangle$	$\|010\rangle$
$\|011\rangle$	$\|011\rangle$
$\|100\rangle$	$\|100\rangle$
$\|101\rangle$	$\|101\rangle$
$\|110\rangle$	$\|111\rangle$
$\|111\rangle$	$\|110\rangle$
$a_1\|000\rangle + a_2\|001\rangle + a_3\|010\rangle + a_4\|011\rangle + a_5\|100\rangle + a_6\|101\rangle + a_7\|110\rangle + a_8\|111\rangle$	$a_1\|000\rangle + a_2\|001\rangle + a_3\|010\rangle + a_4\|011\rangle + a_5\|100\rangle + a_6\|101\rangle + a_8\|110\rangle + a_7\|111\rangle$

Tab 2.7 : Table de vérité de la porte Toffoli

3. Circuit quantique :

Un circuit quantique manipule un ensemble de Qubits. Il possède autant d'entrées que de sorties. Il est constitué d'un ensemble de portes qui agissent comme des transformations unitaires. Ces dernières peuvent être combinées en série ou en parallèle selon le traitement souhaité.

3.1. Composition en série :

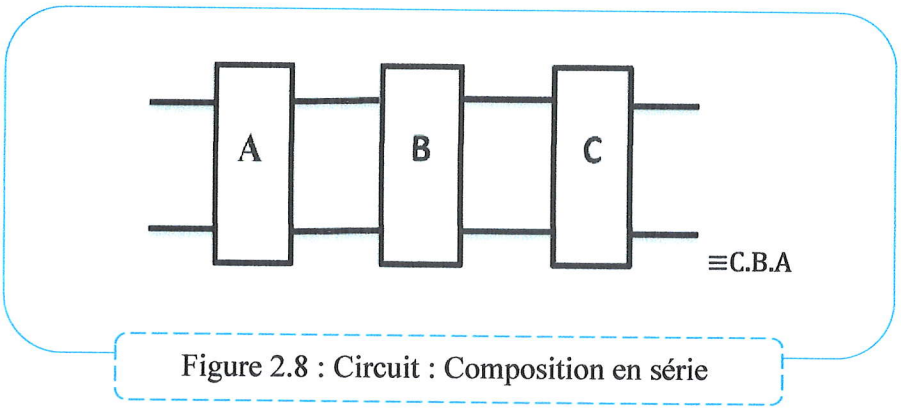
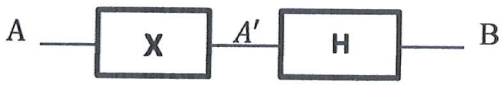


Figure 2.8 : Circuit : Composition en série

- La matrice totale est calculée par le produit cartésien des matrices correspondantes aux portes mais dans l'ordre inverse [13].

Exemple



Dans ce circuit :

$$|A'\rangle = X\ |A\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \beta \\ \alpha \end{pmatrix}$$

$$|B\rangle = H\ |A'\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} \beta \\ \alpha \end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix} \beta+\alpha \\ \beta-\alpha \end{pmatrix}$$

La matrice de transfert globale T est donnée par:

$$T = H.X = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}.\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ -1 & 1\end{pmatrix}$$

Tel que : $|B\rangle = T\ |A\rangle$

$$|B\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ -1 & 1\end{pmatrix}\begin{pmatrix}\alpha\\ \beta\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}\beta+\alpha\\ \beta-\alpha\end{pmatrix}$$

3.2. Composition parallèle :

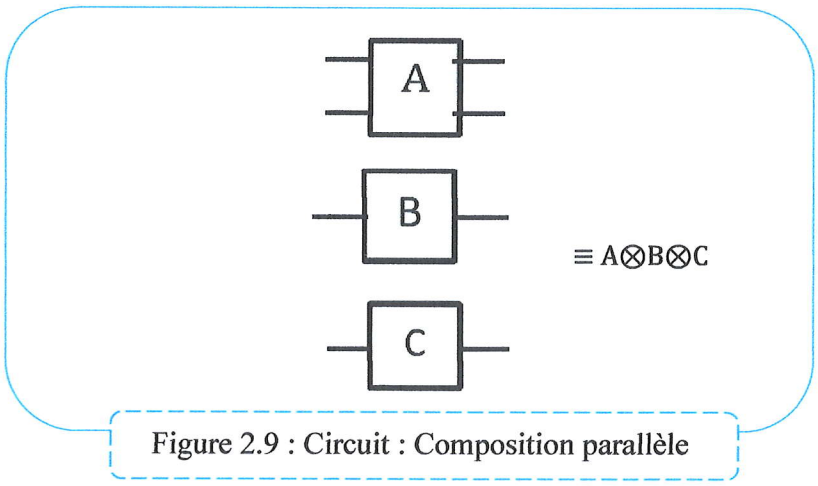
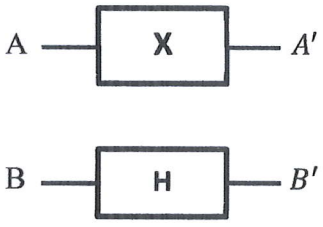


Figure 2.9 : Circuit : Composition parallèle

- Dans ce cas, la matrice globale est obtenue par un produit tensoriel des matrices correspondantes aux portes utilisées [13].

Exemple : soit le circuit suivant **:**



On a: $|A'\rangle = X\ |A\rangle = \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}\begin{pmatrix}\alpha\\ \beta\end{pmatrix} = \begin{pmatrix}\beta\\ \alpha\end{pmatrix}$

$$|B'\rangle = H\ |B\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}\gamma\\ \delta\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}\gamma+\delta\\ \gamma-\delta\end{pmatrix}$$

Le traitement séparé de ces deux Qubits donne comme résultat :

$$|A'B'\rangle = \begin{pmatrix}\beta\\ \alpha\end{pmatrix} \otimes \frac{1}{\sqrt{2}}\begin{pmatrix}\gamma+\delta\\ \gamma-\delta\end{pmatrix} = \begin{pmatrix}\beta\gamma+\beta\delta\\ \beta\gamma-\beta\delta\\ \alpha\gamma+\alpha\delta\\ \alpha\gamma-\alpha\delta\end{pmatrix}$$

Passant maintenant au calcul de la matrice globale T, on a :

$$|A'B'\rangle = T\,|A\,B\rangle$$

Tel que :

$$T = X\otimes H = \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix} \otimes \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}0 & 0 & 1 & 1\\ 0 & 0 & 1 & -1\\ 1 & 1 & 0 & 0\\ 1 & -1 & 1 & 1\end{pmatrix}$$

$$|A\,B\rangle = \begin{pmatrix}\alpha\\ \beta\end{pmatrix} \otimes \begin{pmatrix}\gamma\\ \delta\end{pmatrix} = \begin{pmatrix}\alpha\gamma\\ \alpha\delta\\ \beta\gamma\\ \beta\delta\end{pmatrix}$$

Donc :

$$|A'B'\rangle = T\,|A\,B\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}0 & 0 & 1 & 1\\ 0 & 0 & 1 & -1\\ 1 & 1 & 0 & 0\\ 1 & -1 & 1 & 1\end{pmatrix}\begin{pmatrix}\alpha\gamma\\ \alpha\delta\\ \beta\gamma\\ \beta\delta\end{pmatrix} = \begin{pmatrix}\beta\gamma+\beta\delta\\ \beta\gamma-\beta\delta\\ \alpha\gamma+\alpha\delta\\ \alpha\gamma-\alpha\delta\end{pmatrix}$$

4. Conclusion

Dans un circuit quantique, les portes constituent les briques de base du traitement. Il s'agit d'un ensemble de transformations unitaires et réversibles. Les plus intéressantes sont les portes X, Y et Z ainsi que H.

- Les trois premières sont dites groupe de Pauli. Elles constituent une base de l'espace des matrices unitaires.
- La matrice H est souvent utilisée afin de placer des Qubits dans une superposition.

Les portes quantiques peuvent être combinées en série ou en parallèle. Dans le premier cas, la matrice globale est calculée par un produit matriciel dans le sens inverse des matrices correspondantes aux portes. Alors que dans l'autre cas, elle est obtenue par un produit tensoriel de ces matrices partielles.

Chapitre 3

Correction d'erreur :

- Introduction
- Particularités des erreurs quantiques
- Code de correction à répétition
- Conclusion

1. Introduction

Dans un système quantique, le principal "ennemi" est l'interaction avec l'environnement externe. Cela provoque une grande perturbation des Qubits et engendre des erreurs. Pour remédier à ce problème, les codes correcteurs sont largement utilisés. Ils permettent la détection et la correction correspondante de toute anomalie envisagée.

Dans ce chapitre, nous donnons une petite introduction des codes à répétition. Plus particulièrement, nous détaillons les constructions élémentaires du code de Shor. C'est un processus capable de corriger des erreurs de type X, Y et Z.

2. Particularités des erreurs quantiques :

Dans le cas classique, les techniques de correction d'erreurs sont basées sur l'ajout d'informations redondantes (codage). Cela permet de tester si le message codé a été perturbé et, le cas échéant, de corriger les erreurs. Parmi ces codes, le plus simple est le codage à trois bits. Il consiste à tripler chaque bit d'information :

- $0 \rightarrow 000$.
- $1 \rightarrow 111$.

On suppose que la probabilité d'erreur est suffisament faible pour que la probabilité que deux erreurs surviennent simultanément soit négligeable. Donc après avoir traversé un canal susceptible de créer une erreur, le triplet de bits peut se retrouver avec au plus un bit inversé. La détection de l'erreur se fait en testant si tous les bits sont égaux ou non. En cas d'inégalités, on utilise la règle de la majorité pour rétablir la bonne valeur logique associée au triplet. C'est ce qu'on appelle le décodage.

Mais plusieurs particularités du cas quantique rendent impossible l'adaptation directe des stratégies classiques :

- Tout d'abord la structure du Qubit est différente de celle du bit classique. Alors que dans le cas classique il faut transmettre un 0 ou un 1, dans le cas quantique c'est un état de la forme :

$|\psi> = \alpha|0> + \beta|1>$

C'est-`a-dire deux nombres complexes.

- De plus, du fait du caractère unitaire, certaines opérations classiques sont impossibles à réaliser de manière quantique, par exemple à cause du théorème de non clonage.
- La mesure détruit l'état quantique. En plus, c'est une opération irréversible.
- Les erreurs quantiques possibles sont plus nombreuses que celles classiques. Tandis que classiquement une erreur sur un bit est une simple interversion, une erreur sur un Qubit peut être n'importe quelle transformation unitaire.

Par contre, dans le cadre quantique, les recherches affirment que si nous pouvons corriger à la fois les erreurs X, Y et Z, alors nous pouvons corriger du même coup une très large classe d'erreurs à un Qubit. C'est une conséquence directe des propriétés du groupe de Pauli.

3. Codes de correction à répétition :

3.1. Correction d'erreurs de type X ou Bit-flip [14]

Le circuit de correction d'une erreur de type X est le suivant :

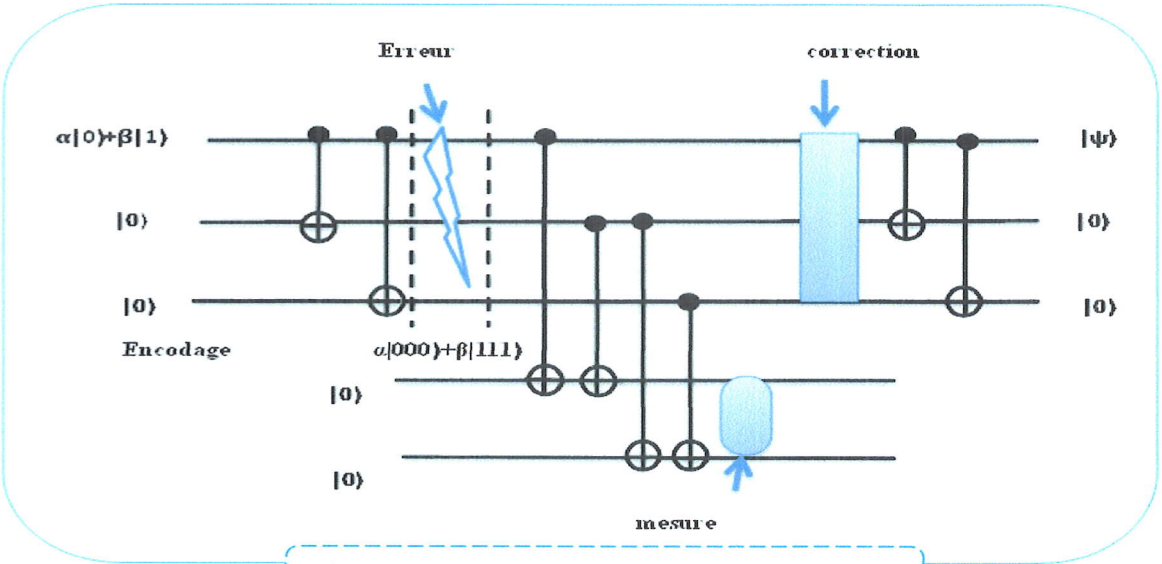


Figure 3.1 : Circuit d'encodage du code bit-flip

Encodage : Soit le Qubit $|\psi\rangle$ défini par : $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$.

L'étape d'encodage est basée sur une redondance spécifique. Elle consiste à l'ajout de deux Qubits intriqués au Qubit initial. Elle est réalisée en appliquant deux portes CNot.

- L'état initial : $|\psi_1\rangle = \alpha|000\rangle + \beta|100\rangle$
- L'application du CNot (1,2) donne l'état :

$$|\psi_2\rangle = \alpha|000\rangle + \beta|110\rangle$$

- L'application du CNot (1,3) donne l'état :

$$|\psi_3\rangle = \alpha|000\rangle + \beta|111\rangle$$

L'encodage d'un Qbit : $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ donne un état quantique $|\psi_3\rangle$ de trois Qubits tel que : $|\psi_3\rangle = \alpha|000\rangle + \beta|111\rangle$

Décodage : Le décodage est constitué des deux étapes suivantes :

- **Etape_01 : Détection d'erreurs :** La détection d'erreurs X est basée sur l'utilisation de deux Qubits supplémentaires initialisés à $|0\rangle$ comme syndrome. Ce dernier est calculé par quatre portes CNot. Dans la suite, nous présentons les quatre cas possibles.

Cas 01 : sans erreur :

$$|\psi_4\rangle = \alpha|000\rangle + \beta|111\rangle$$

- L'ajout des deux Qubits du syndrome donne l'état :

$$|\psi_5\rangle = \alpha|00000\rangle + \beta|11100\rangle$$

- L'application des portes CNot donne les états suivants :

$$C_{[1,4]} \Rightarrow |\psi_6\rangle = \alpha|00000\rangle + \beta|11110\rangle$$

$$C_{[2,4]} \Rightarrow |\psi_7\rangle = \alpha|00000\rangle + \beta|11100\rangle$$

$$C_{[2,5]} \Rightarrow |\psi_8\rangle = \alpha|00000\rangle + \beta|11101\rangle$$

$$C_{[3,5]} \Rightarrow |\psi_9\rangle = \alpha|00000\rangle + \beta|11100\rangle$$

Dans ce cas : $|\psi_9\rangle = (\alpha|000\rangle + \beta|111\rangle)\,|00\rangle$

Cas 02 : Erreur X sur le 1er Qubit :

$$|\psi_4\rangle = \alpha|100\rangle + \beta|011\rangle$$

- L'ajout des deux Qubits du syndrome donne l'état :

$$|\psi_5\rangle = \alpha|10000\rangle + \beta|01100\rangle$$

- L'application des portes CNot donne les états suivants :

$$C_{[1,4]} \Longrightarrow |\psi_6\rangle = \alpha|10010\rangle + \beta|01100\rangle$$

$$C_{[2,4]} \Longrightarrow |\psi_7\rangle = \alpha|10010\rangle + \beta|01110\rangle$$

$$C_{[2,5]} \Longrightarrow |\psi_8\rangle = \alpha|10010\rangle + \beta|01111\rangle$$

$$C_{[3,5]} \Longrightarrow |\psi_9\rangle = \alpha|10010\rangle + \beta|01110\rangle$$

Dans ce cas : $|\psi_9\rangle = (\alpha|100\rangle + \beta|011\rangle)\,|10\rangle$

Cas 03 : Erreur X sur le 2eme Qubit :

$$|\psi_4\rangle = \alpha|010\rangle + \beta|101\rangle$$

- L'ajout des deux Qubits du syndrome donne l'état :

$$|\psi_5\rangle = \alpha|01000\rangle + \beta|10100\rangle$$

- L'application des portes CNot donne les états suivants :

$$C_{[1,4]} \Longrightarrow |\psi_6\rangle = \alpha|01000\rangle + \beta|10110\rangle$$

$$C_{[2,4]} \Longrightarrow |\psi_7\rangle = \alpha|01010\rangle + \beta|10110\rangle$$

$$C_{[2,5]} \Longrightarrow |\psi_8\rangle = \alpha|01011\rangle + \beta|10110\rangle$$

$$C_{[3,5]} \Longrightarrow |\psi_9\rangle = \alpha|01011\rangle + \beta|10111\rangle$$

Dans ce cas : $|\psi_9\rangle = (\alpha|010\rangle + \beta|101\rangle)\,|11\rangle$

Cas 04 : Erreur X sur le 3eme Qubit :

$$|\psi_4\rangle = \alpha|001\rangle + \beta|110\rangle$$

- L'ajout des deux Qubits du syndrome donne l'état :

$$|\psi_5\rangle = \alpha|00100\rangle + \beta|11000\rangle$$

- L'application des portes CNot donne les états suivants :

$$C_{[1,4]} \Rightarrow |\psi_6\rangle = \alpha|00100\rangle + \beta|11010\rangle$$

$$C_{[2,4]} \Rightarrow |\psi_7\rangle = \alpha|00100\rangle + \beta|11000\rangle$$

$$C_{[2,5]} \Rightarrow |\psi_8\rangle = \alpha|00100\rangle + \beta|11001\rangle$$

$$C_{[3,5]} \Rightarrow |\psi_9\rangle = \alpha|00101\rangle + \beta|11001\rangle$$

Dans ce cas : $|\psi_9\rangle = (\alpha|001\rangle + \beta|110\rangle)\,|01\rangle$

- **La mesure :** Dans cette étape, les deux Qubits du syndrome sont mesurés. Les valeurs obtenues sont suffisantes pour la correction.
- **Correction :** On applique l'opérateur unitaire X_i^{-1} sur le Qubit erroné afin de retrouver son état initial. Comme l'opérateur X est unitaire alors : $X_i^{-1} = X_i$

$\|Q_4\ Q_5\rangle$	Correction
$\|00\rangle$	Ne rien faire
$\|01\rangle$	X (3)
$\|10\rangle$	X (1)
$\|11\rangle$	X (2)

Ce qui permet de revenir dans tous les cas à l'état : $|\psi_9\rangle = \alpha|000\rangle + \beta|111\rangle$

- **Décodage :** Le décodage sert à isoler le Qubit initial des deux Qubits d'intrication. Il est réalisé par application de l'inverse des portes utilisées dans l'étape l'encodage.
  - L'état initial : $|\psi_9> = \alpha|000> + \beta|111>$
  - L'application du CNot (1,2) donne l'état :

$$|\psi_{10}> = \alpha|000> + \beta|101>$$

  - L'application du CNot (1,3) donne l'état :

$$|\psi_{11}> = \alpha|000> + \beta|100> = |\psi\ \ >|00>$$

3.2. Correction d'erreurs de type Z ou Phase-flip [14] :

Comme vu dans le chapitre précédent, les portes X, Z et H sont définies par les matrices unitaires suivantes :

$$X \equiv \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad 2.1$$

$$Z \equiv \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad 2.3$$

$$H \equiv \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad 2.4$$

Calculant la matrice correspondante à la transformation : HZH.

$$HZH \equiv \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = X$$

Ce résultat affirme que la correction d'une erreur de type Z se ramène à celle d'une erreur de type X. Il suffit d'ajouter une transformation H à la fin de l'encodage et son inverse (H) avant la détection d'erreur. Donc, le circuit de correction est le suivant :

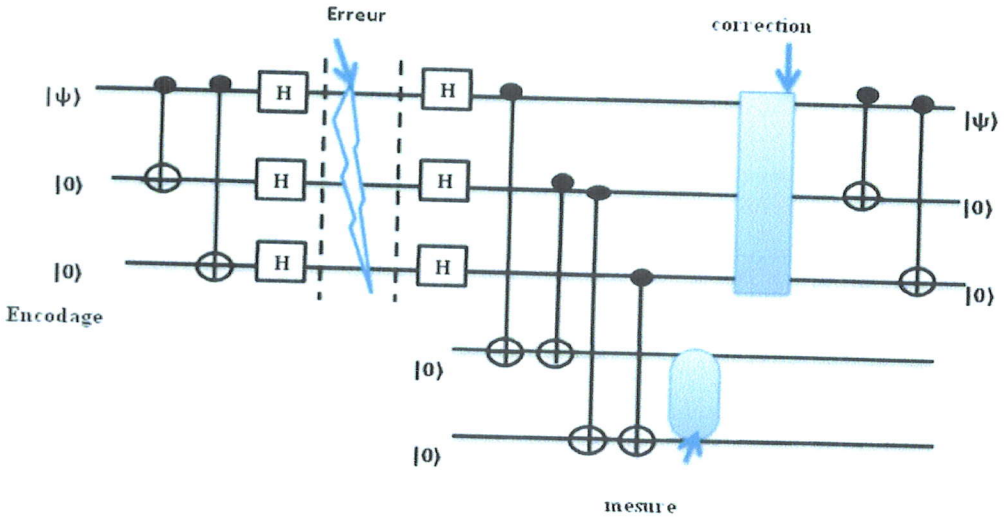


Figure 3. 2 : Circuit d'encodage du code Phase-flip

Encodage : Soit le Qubit $|\psi>$ défini par : $|\psi>= \alpha|0>+ \beta|1>$.

L'étape d'encodage est basée sur une redondance spécifique. Elle consiste à l'ajout de deux Qubits intriqués au Qubit initial. Elle est réalisée en appliquant deux portes CNot.

- L'état initial : $|\psi_1 >= \alpha|000>+ \beta|100>$
- L'application du CNot (1,2) donne l'état :

$$|\psi_2 >= \alpha|000>+ \beta|110>$$

- L'application du CNot (1,3) donne l'état :

$$|\psi_3 >= \alpha|000>+ \beta|111>$$

- Apres l'encodage, l'application des portes H sur les trois Qubits donne les états suivants :

$$|\psi_4\rangle= \frac{1}{\sqrt{2}} \alpha (|0\rangle+|1\rangle)|00\rangle+ \frac{1}{\sqrt{2}} \beta (|0\rangle-|1\rangle) |11\rangle$$

$$|\psi_5\rangle= \frac{1}{2} \alpha (|0\rangle+|1\rangle) (|0\rangle+|1\rangle) |0\rangle + \frac{1}{2} \beta (|0\rangle-|1\rangle) (|0\rangle-|1\rangle) |1\rangle$$

$$|\psi_6\rangle= \frac{1}{2\sqrt{2}} \alpha (|0\rangle+|1\rangle) (|0\rangle+|1\rangle) (|0\rangle+|1\rangle) + \frac{1}{2\sqrt{2}} \beta (|0\rangle-|1\rangle) (|0\rangle-|1\rangle) (|0\rangle-|1\rangle)$$

Décodage : Le décodage est précédé par une application des portes H sur les trois Qubits. Dans la suite, nous présentons les états obtenus dans les quatre cas possibles :

Cas 01 : sans erreur :

- $|\psi_7\rangle= \frac{1}{2\sqrt{2}} \alpha (|0\rangle+|1\rangle) (|0\rangle+|1\rangle) (|0\rangle+|1\rangle) + \frac{1}{2\sqrt{2}} \beta (|0\rangle-|1\rangle) (|0\rangle-|1\rangle) (|0\rangle-|1\rangle)$
- L'application des trois portes H donne l'état suivant :

$$|\psi_8 >= \alpha|000>+ \beta|111>$$

Cas 02 : Erreur Z sur le 1er Qubit :

- $|\psi_7\rangle= \frac{1}{2\sqrt{2}} \alpha (|0\rangle-|1\rangle) (|0\rangle+|1\rangle) (|0\rangle+|1\rangle) + \frac{1}{2\sqrt{2}} \beta (|0\rangle+|1\rangle) (|0\rangle-|1\rangle) (|0\rangle-|1\rangle)$
- L'application des trois portes H donne l'état suivant :

$$|\psi_8\rangle= \alpha|100\rangle+ \beta|011\rangle$$

Cas 03 : Erreur Z sur le 2eme Qubit :

- $|\psi_7\rangle = \frac{1}{2\sqrt{2}} \alpha (|0\rangle+|1\rangle)(|0\rangle-|1\rangle)(|0\rangle+|1\rangle) + \frac{1}{2\sqrt{2}} \beta (|0\rangle-|1\rangle)(|0\rangle+|1\rangle)(|0\rangle-|1\rangle)$
- L'application des trois portes H donne l'état suivant :

$$|\psi_8\rangle = \alpha|010\rangle + \beta|101\rangle$$

Cas 02 : Erreur Z sur le 3eme Qubit :

- $|\psi_7\rangle = \frac{1}{2\sqrt{2}} \alpha (|0\rangle+|1\rangle)(|0\rangle+|1\rangle)(|0\rangle-|1\rangle) + \frac{1}{2\sqrt{2}} \beta (|0\rangle-|1\rangle)(|0\rangle-|1\rangle)(|0\rangle+|1\rangle)$
- L'application des trois portes H donne l'état suivant :

$$|\psi_8\rangle = \alpha|001\rangle + \beta|110\rangle$$

Conclusion:

- Une erreur de type Z sur un Qubit est transformée en une erreur de type X sur le même Qubit.
- Les étapes suivantes sont les mêmes vues dans le traitement des erreurs X.

3.3. Correction d'erreurs de type Y [14] :

De même, Calculant :

$$iZX \equiv \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = Y \qquad 2.2$$

Donc, une erreur Y peut être corrigée par application d'un correcteur d'erreurs Z suivi d'un correcteur d'erreurs X.

3.4. Algorithme de Shor :

Un système quantique pourrait avoir des erreurs de type X, Y et Z. Shor a développé un code de correction universel capable de corriger ces trois types d'erreurs à la fois [14]. L'idée principale est de "concaténer" les deux codes de répétition présentés dans le paragraphe précédent :

- Un correcteur protégeant le Qubit $|\psi\rangle$ des erreurs de type Z. Il utilise deux Qubits supplémentaires.
- Chacun de ces trois Qubits est protégé à son tour par un correcteur des erreurs de type X.

Donc, c'est un correcteur à 9 Qubits. Le schéma suivant donne l'architecture globale de cette solution

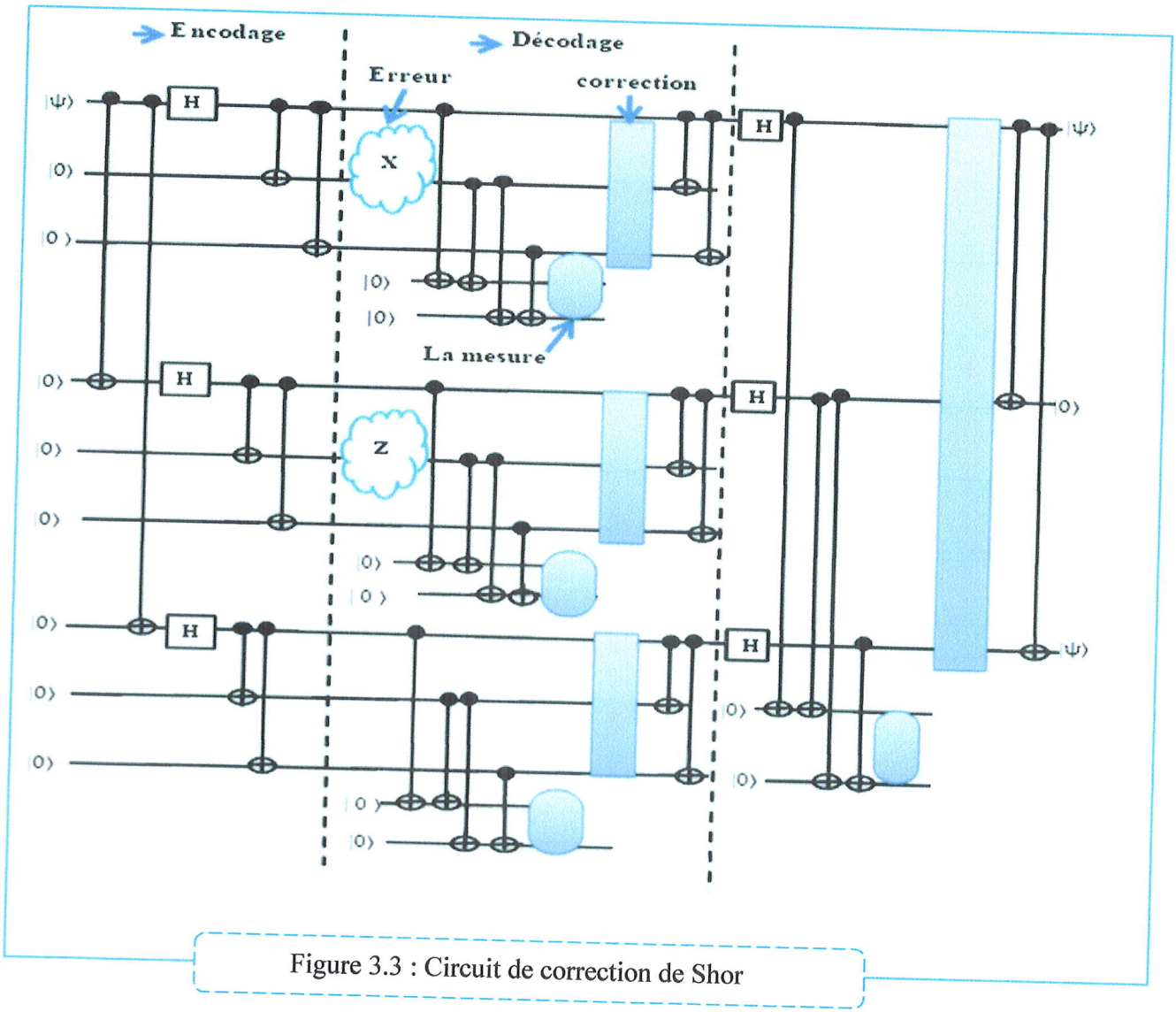


Figure 3.3 : Circuit de correction de Shor

Remarques :

- Déjà avec 9 Qubits, on a constaté que le calcul des états résultats est devenu très difficile. D'où la nécessité d'une implémentation informatique.
- La démonstration de l'efficacité de cette solution est reportée au chapitre suivant. Elle sera détaillée en utilisant notre implémentation.

4. Conclusion

Dans un calculateur quantique, les erreurs sont principalement dues à l'interaction du système quantique avec son environnement. Plutôt que de tenter d'affronter de face cet obstacle inévitable, il est possible d'utiliser des techniques de correction.

De même que dans le cas classique, un code correcteur quantique utilise une procédure de codage/transmission imparfaite/décodage. L'idée la plus simple est l'adaptation directe des stratégies classiques. Mais, plusieurs particularités du cas quantique rendent cette tâche impossible. Le clonage pur et simple est impossible.

Dans ce chapitre, on a présenté l'algorithme de Shor. C'est un correcteur basé sur une redondance implémentée sous forme d'intrication. C'est une solution qui nécessite l'utilisation huit Qubits supplémentaires. Elle permet la détection et la correction des erreurs de type X, Y et Z.

Chapitre 4

Approche proposée :

- Simulation quantique
- Exemples illustratifs

Dans ce chapitre nous présentons quelques exemples d'application en donnant les traces d'exécutions.

1. Simulation quantique :

Pour la démonstration de l'algorithme de Shor, on a développé une version simplifiée d'un simulateur quantique, ne contenant que les fonctionnalités requises, en langage Java. Les classes principales sont les suivantes :

- **La classe Etat_Quantique :** Cette classe dédiée à la manipulation des états quantique. Il s'agit d'un vecteur contenant la combinaison linéaire des états de la base canonique. Pour des fins d'optimisation, les éléments avec des coefficients nuls sont négligés. Cette classe n'a pas d'attributs mais elle est munie des méthodes suivantes :
  - Constructeur (alpha, Beta).
  - Produit_Tensoriel (Etat_Quant_01, Etat_Quant_02).
  - X (Etat_Quant, Pos_QbitCible).
  - Y (Etat_Quant, Pos_QbitCible).
  - Z (Etat_Quant, Pos_QbitCible).
  - H (Etat_Quant, Pos_QbitCible).
  - CNot ((Etat_Quant, Pos_QbitControle, Pos_QbitCible).
  - Mesure(Pos_QbitCible).
  - Afficher ( ).

- **La classe Etat_Base :** Cette classe sert à la manipulation des états élémentaires. Chaque objet est caractérisé par les attributs suivants :
  - Vecteur : chaine de caractère.
  - Param : un nombre réel spécifient le coefficient de cet état élémentaire.

  Les méthodes de cette classe sont :
  - Constructeur (Param, Vect).
  - Afficher ( ).

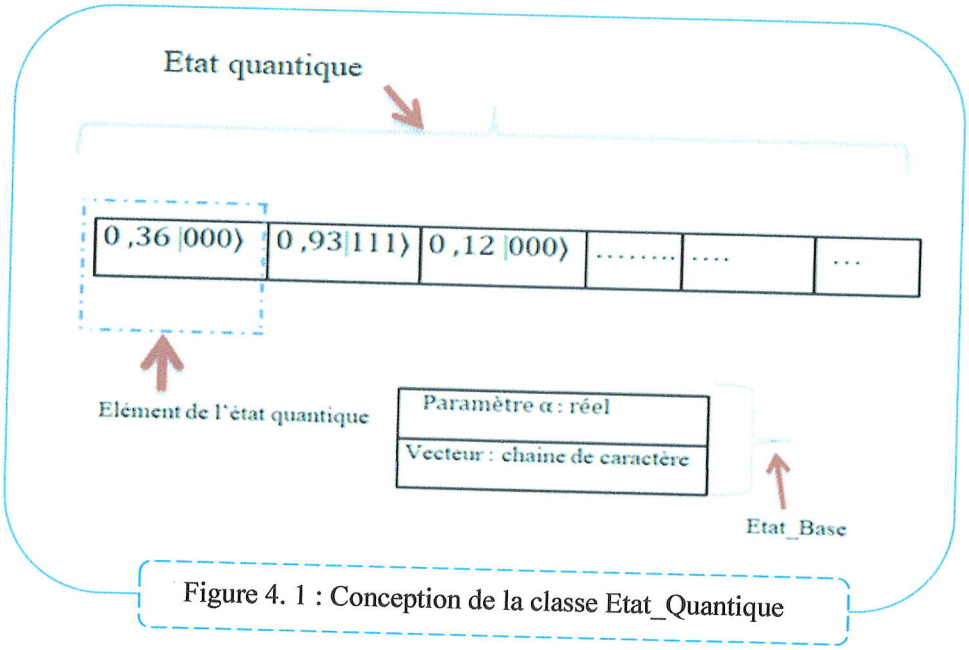


Figure 4. 1 : Conception de la classe Etat_Quantique

- **Pseudo-code des méthodes :**

```
Etat_Quantique Produit_Tensoriel ( Etat_Qt_01 : Etat_Quantique, Etat_Qt_02 : Etat_Quantique )
Debut
    Pour ( i=0 ; i < Taille (Etat_Qt_01 ) ) faire :
        Pour ( j = 0 ; j < Taille ( Etat_Qt_02 ) ) faire :
            Param_R = (Etat_Qt_01 [ i ]. param) * (Etat_Qt_02[ j ]. param) ;
            Vect_R = (Etat_Qt_01 [ i ]. vect) + (Etat_Qt_02 [ j ]. vect) ;
            Etat_Base_R= Etat_Base (Param_R, Vect_R) ;
            Etat_Qt_Resultat .ajouter (Etat_Base_R) ;
        Fin Pour
    Fin Pour
    Return ( Etat_Qt_Resultat) ;
Fin
```

```
Etat_Quantique X (Etat_Quant : Etat_Quantique, Pos_QbitCible : entier)

Debut
    Pos_QbitCible = Pos_QbitCible – 1 ;
   Pour ( i = 0 ; i <  Taille (Etat_Quant)) faire :
        vect_courant  = Etat_Quant [i] . vect ;
         Qbit_cible = vect_courant [Pos_QbitCible] ;
         Si (Qbit_cible = '0')
            nouv_vect = vect_courant [0, Pos_QbitCible];
             nouv_vect = nouv_vect + '1' ;
            nouv_vect =  nouv_vect + vect_courant [Pos_QbitCible + 1 , Taille (vect_courant) ];
         Sinon
              nouv_vect  = vect_courant [0, Pos_QbitCible];
             nouv_vect  = nouv_vect + '0' ;
            nouv_vect =  nouv_vect + vect_courant [Pos_QbitCible + 1, Taille (vect_courant) ];
        Fin Si
       Nouv_EtatBase =  Etat_Base (Etat_Quant [ i ] .param ,  nouv_vect ) ;
       Etat_Qt_Resultat. Ajouter (Nouv_EtatBase);
  Fin Pour

  Return (Etat_Qt_Resultat) ;
Fin
```

```
Etat_Quantique Z (Etat_Quant : Etat_Quantique,  Pos_QbitCible : entier)
Debut
    Pos_QbitCible = Pos_QbitCible – 1 ;
   Pour ( i = 0 ; i <  Taille (Etat_Quant)) faire :
        vect_courant  = Etat_Quant [ i ] . vect ;
         nouv_vect = vect_courant ;

         Qbit_cible = vect_courant [Pos_QbitCible] ;
         Si (Qbit_cible = '0')
             nouv_param = Etat_Quant [ i ] . param ;
        Sinon
             nouv_param = -  Etat_Quant [ i ] . param ;
        Fin Si
       Nouv_EtatBase =  Etat_Base (nouv_param,  nouv_vect) ;
        Etat_Qt_Resultat. Ajouter (Nouv_EtatBase);
   Fin Pour
  Return (Etat_Qt_Resultat) ;
Fin
```

```
Etat_Quantique CNot (Etat_Quant : Etat_Quantique, Pos_QbitControl: entier, Pos_QbitCible : entier)
Debut
    Pos_QbitCible = Pos_QbitCible – 1 ;
    Pos_QbitControle = Pos_QbitControle – 1 ;
  Pour ( i = 0 ; i < Taille (Etat_Quant)) faire :
        nouv_par = Etat_Quant [ i ] . param ;
        vect_courant = Etat_Quant [ i ] . vect ;
        Qbit_Controle = vect_courant [Pos_QbitControl] ;
        Si (Qbit_Controle = '0')
            nouv_vect = vect_courant;
        Sinon
            Qbit_cible= vect_courant [Pos_QbitCible] ;
            Si (Qbit_cible= '0')
                nouv_vect = vect_courant [0, Pos_QbitCible];
                  nouv_vect = nouv_vect + '1' ;
                nouv_vect = nouv_vect + vect_courant [ Pos_QbitCible + 1 , Taille (vect_courant) ];
            Sinon
                nouv_vect = vect_courant [0, Pos_QbitCible];
                nouv_vect = nouv_vect + '1' ;
                nouv_vect = nouv_vect + vect_courant [ Pos_QbitCible + 1 , Taille (vect_courant) ];
            Fin Si
        Fin Si
      Nouv_EtatBase = Etat_Base (nouv_par, nouv_vect) ;
      Etat_Qt_Resultat. Ajouter (Nouv_EtatBase);
  Fin Pour
    Return (Etat_Qt_Resultat) ;
Fin
```

```
Etat_Quantique H (Etat_Quant : Etat_Quantique, Pos_QbitCible : entier)
Debut
    Pos_QbitCible = Pos_QbitCible – 1 ;
    Pour ( i = 0 ; i < Taille (Etat_Quant) ) faire :
        vect_courant = Etat_Quant [ i ] . vect ;
        nouv_vect = vect_courant [0, Pos_QbitCible];
        nouv_vect_0 = nouv_vect + '0' ;
        nouv_vect_1 = nouv_vect + '1' ;
        nouv_vect_0 = nouv_vect + vect_courant [Pos_QbitCible + 1, Taille (vect_courant)];
        nouv_vect_1 = nouv_vect + vect_courant [Pos_QbitCible + 1, Taille (vect_courant) ];
        Qbit_cible = vect_courant [Pos_QbitCible] ;
        Si (QbitCible = '0')
            Nouv_EtatBase_0 = Etat_Base (Etat_Quant [ i ] . par / sqrt(2), nouv_vect_0) ;
            Nouv_EtatBase_1 = Etat_Base (Etat_Quant [ i ] . par / sqrt(2), nouv_vect_1) ;
        Sinon
            Nouv_EtatBase_0 = Etat_Base (Etat_Quant [ i ] . par / sqrt(2), nouv_vect_0) ;
            Nouv_EtatBase_1 = Etat_Base ( - Etat_Quant [ i ] . par / sqrt(2), nouv_vect_1) ;
        Fin Si
        Etat_Qt_Resultat. Ajouter (Nouv_EtatBase_0);
        Etat_Qt_Resultat. Ajouter (Nouv_EtatBase_1);
    Fin Pour
    Return (Etat_Qt_Resultat) ;
Fin
```

2. Exemples illustratifs : Supposant que l'état initial du Qubit Psi est le suivant :

$$|Psi\rangle = 0{,}36\,|0> + 0{,}93\,|1>$$

2.1. Correction d'erreurs de type X (Bit-flip)

- Création de l'état initial de Qbit :

$$|Psi\rangle = 0{,}36\,|0> + 0{,}93\,|1>$$

- Ajout des deux Qubits d'intrication:

$$|Psi\rangle = 0{,}36\,|000> + 0{,}93\,|100>$$

- Application de la porte CNot (1, 2):

$$|Psi\rangle = 0{,}36\,|000> + 0{,}93\,|110>$$

- Application de la porte CNot (1, 3):

$$| Psi \rangle = 0{,}36 | 000 > + 0{,}93 | 111 >$$

- **Injecter une Erreur X sur le premier Qubit :**

$$| Psi \rangle = 0{,}36 | 100 > + 0{,}93 | 011 >$$

- Ajout des deux Qubits du syndrome:

$$| Psi \rangle = 0.36 | 10000 > + 0.93 | 01100 >$$

- Application de la porte CNot (1, 4):

$$| Psi \rangle = 0.36 | 10010 > + 0.93 | 01100 >$$

- Application de la porte CNot (2, 4):

$$| Psi \rangle = 0.36 | 10010 > + 0.93 | 01110 >$$

- Application de la porte CNot (2, 5):

$$| Psi \rangle = 0.36 | 10010 > + 0.93 | 01111 >$$

- Application de la porte CNot (3, 5):

$$| Psi \rangle = 0.36 | 10\mathbf{0}1\mathbf{0} > + 0.93 | 01\mathbf{1}1\mathbf{0} >$$

- Mesurer les deux Qubit (Q_4 , Q_5) :

Resultat de mesure = 10, donc erreur de type X1

- La correction : appliquer la porte X(1)

$$| Psi \rangle = 0.36 | 00010 > + 0.93 |11110 >$$

- Suppression des deux Qubits du syndrome:

$$| Psi \rangle = 0.36 | 000 > + 0.93 | 111 >$$

- Application de la porte CNot (1, 2):

$$| Psi \rangle = 0.36 | 000 > + 0.93 | 101 >$$

- Application de la porte CNot (1, 3):

$$| Psi \rangle = 0.36 | 000 > + 0.93 | 100 >$$

$$= (\mathbf{0.36 |0> + 0.93 |1>}) \ |00 >$$

2.2. Correction d'erreurs de type Z (Phase-flip)

- Création de l'état initial de Qbit :

$$|Psi\rangle = 0,36 | 0 > + 0,93 | 1 >$$

- Ajout des deux Qubits d'intrication:

$$|Psi\rangle = 0,36 | 000 > + 0,93 | 100 >$$

- Application de la porte CNot (1, 2):

$$| Psi \rangle = 0,36 | 000 > + 0,93 | 110 >$$

- Application de la porte CNot (1, 3):

$$| Psi \rangle = 0,36 | 000 > + 0,93 | 111 >$$

- Application de la porte H (1):

$$|Psi\rangle = 0,25 | 000 > + 0,25 | 100 > + 0,66 | 011 > -0,66 | 111 >$$

- Application de la porte H (2):

$$|Psi\rangle = 0,18 | 000 > + 0,18 | 010 > + 0,18 | 100 > + 0,18 | 110 > +$$

$$0,46 | 001 > - 0,46 | 011 > -0,46 | 101 > + 0,46 | 111 >$$

- Application de la porte H (3):

$$|Psi\rangle = 0,46 | 000 > -0,20 | 001 > -0,20 | 010 > + 0,46 | 011 > - 0,20 |100 >$$

$$+ 0,46 | 101 > + 0,46 | 110 > -0,20 | 111 >$$

- **Injecter une Erreur Z sur le deuxième Qubit :**

$$|Psi\rangle = 0,46 \mid 000 > -0,20 \mid 001 > +0,20 \mid 010 > - 0,46 \mid 011 > - 0,20 \mid 100 >$$

$$0,46 \mid 101 > - 0,46 \mid 110 > +0,20 \mid 111 >$$

- Application de la porte H (1):

$$|Psi\rangle = 0,18 \mid 000 > + 0,46 \mid 100 > + 0,18 \mid 001 > -0,46 \mid 101 > - 0,18 \mid 010 >$$

$$+ 0,46 \mid 110 > -0,18 \mid 011 > -0,46 \mid 111 >$$

- Application de la porte H (2):

$$|Psi\rangle = 0,25 \mid 010 > + 0,66 \mid 100 > + 0,25 \mid 011 > -0,66 \mid 101 >$$

- Application de la porte H (3):

$$|Psi\rangle = 0,36 \mid 010 > + 0,93 \mid 101 >$$

- Ajout des deux Qubits du syndrome :

$$|Psi\rangle = 0,36 \mid 01000 > + 0,93 \mid 10100 >$$

- Application de la porte CNot (1, 4):

$$|Psi\rangle = 0,36 \mid 01000 > + 0,93 \mid 10110 >$$

- Application de la porte CNot (2, 4):

$$|Psi\rangle = 0,36 \mid 01010 > + 0,93 \mid 10110 >$$

- Application de la porte CNot (2, 5):

$$|Psi\rangle = 0,36 \mid 01011 > + 0,93 \mid 10110 >$$

- Application de la porte CNot (3, 5):

$$|Psi\rangle = 0,36 \mid 01011 > + 0,93 \mid 10111 >$$

- Mesurer les deux Qubit (Q_4 , Q_5) :

Résultat de mesure = 11 , donc erreur de type X2

- La correction : appliquer la porte X(2)

$$| Psi \rangle = 0.36 | 00011 > + 0.93 | 11111 >$$

- Suppression des deux Qubits du syndrome:

$$| Psi \rangle = 0.36 | 000 > + 0.93 | 111 >$$

- Application de la porte CNot (1, 2):

$$| Psi \rangle = 0.36 | 000 > + 0.93 | 101 >$$

- Application de la porte CNot (1, 3):

$$| Psi \rangle = 0.36 | 000 > + 0.93 | 100 >$$

$$= (\mathbf{0.36 |0 > + 0.93 |1>}) \ |00 >$$

2.3. Correction d'erreurs de type X et Z (Shor):

- Création de l'état initial de Qbit :

$$\mathbf{|Psi\rangle = 0,36 | 0 > + 0,93 | 1 >}$$

- Ajout des Qubits d'intrication :

$$|Psi\rangle = 0,36 |000000000> + 0,93 |100000000>$$

- Application de la porte CNot (1, 4):

$$|Psi\rangle = 0,36 |000000000> + 0,93 |100100000>$$

- Application de la porte CNot (1, 7):

$$|Psi\rangle = 0,36 |000000000> + 0,93 |100100100>$$

- Application de la porte H (1):

$$|Psi\rangle = 0,25\ |000000000\rangle + 0,25\ |100000000\rangle + 0,66\ |000100100\rangle - 0,66\ |100100100\rangle$$

- Application de la porte H (4):

$$|Psi\rangle = 0,18\ |000000000\rangle + 0,18\ |000100000\rangle + 0,18\ |100000000\rangle + 0,18\ |100100000\rangle + 0,46\ |000000100\rangle - 0,46\ |000100100\rangle - 0,46\ |100000100\rangle + 0,46\ |100100100\rangle$$

- Application de la porte H (7):

$$|Psi\rangle = 0,46\ |000000000\rangle + 0,20\ |000000100\rangle + 0,20\ |000100000\rangle + 0,46\ |000100100\rangle + 0,20\ |100000000\rangle + 0,46\ |100000100\rangle + 0,46\ |100100000\rangle - 0,20\ |100100100\rangle$$

- Application des portes : CNot (1, 2), CNot (1, 3), CNot (4, 5), CNot (4, 6), CNot (7, 8), CNot (7, 9):

$$|Psi\rangle = 0,46\ |000000000\rangle - 0,20\ |000000111\rangle - 0,20\ |000111000\rangle + 0,46\ |000111111\rangle - 0,20\ |111000000\rangle + 0,46\ |111000111\rangle + 0,46\ |111111000\rangle - 0,20\ |111111111\rangle$$

- **Injecter une Erreur X sur le premier Qubit et une autre erreur de type Z sur le quatrième Qubit :**

$$|Psi\rangle = 0,46\ |100000000\rangle - 0,20\ |100000111\rangle + 0,20\ |100111000\rangle + 0,46\ |100111111\rangle - 0,20\ |011000000\rangle + 0,46\ |011000111\rangle - 0,46\ |011111000\rangle + 0,20\ |011111111\rangle$$

- Ajout des Qubits des syndromes :

$$|Psi\rangle = 0,46\ |10000000000000000\rangle + 0,20\ |10000011100000000\rangle +$$
$$0,20\ |10011100000000000\rangle - 0,46\ |10011111100000000\rangle -$$
$$0,20\ |01100000000000000\rangle + 0,46|01100011100000000\rangle -$$
$$0,46\ |01111100000000000\rangle + 0,20\ |01111111100000000\rangle$$

- Application de la porte CNot (1, 10) :

$$|Psi\rangle = 0,46\ |10000000010000000\rangle + 0,20\ |10000011110000000\rangle +$$
$$0,20\ |10011100010000000\rangle -0,46\ |10011111110000000\rangle -$$
$$0,20\ |01100000000000000\rangle + 0,46|01100011100000000\rangle -$$
$$0,46\ |01111100000000000\rangle + 0,20\ |01111111100000000\rangle$$

- Application de la porte CNot (2, 10):

$$|Psi\rangle = 0,46|10000000010000000\rangle + 0,20\ |10000011110000000\rangle +$$
$$0,20\ |10011100010000000\rangle\ -0,46\ |10011111110000000\rangle -$$
$$0,20\ |01100000010000000\rangle + 0,46\ |01100011110000000\rangle -$$
$$0,46\ |01111100010000000\rangle + 0,20\ |01111111110000000\rangle$$

- Application de la porte CNot (2, 11):

$$|Psi\rangle = 0,46\ |10000000010000000\rangle +0,20\ |10000011110000000\rangle +$$
$$0,20\ |10011100010000000\rangle -0,46\ |10011111110000000\rangle -$$
$$0,20\ |01100000011000000\rangle + 0,46\ |01100011111000000\rangle -$$
$$0,46\ |01111100011000000\rangle + 0,20\ |01111111111000000\rangle$$

- Application de la porte CNot (3, 11):

|Psi⟩ = 0,46 |1000000001**1**0000000> + 0,20|1000001111**1**0000000> +

0,20 |1001110001**1**0000000> - 0,46 |1001111111**1**0000000> -

0,20 |0110000001**1**0000000> + 0,46 |0110001111**1**0000000> -

0,46 |0111110001**1**0000000> + 0,20 |0111111111**1**0000000>

- Mesurer les deux Qubit (Q_{10} , Q_{11}) :

Résultat de mesure = 10 , donc erreur de type X1

- La correction : appliquer la porte X(1)

|Psi⟩ = 0,46 |0000000001**1**0000000> + 0,20|0000001111**1**0000000> +

0,20 |0001110001**1**0000000> - 0,46 |0001111111**1**0000000> -

0,20 |1110000001**1**0000000> + 0,46|1110001111**1**0000000> -

0,46 |1111110001**1**0000000> + 0,20|1111111111**1**0000000>

- Application de la porte CNot (4, 12):

|Psi⟩ = 0,46 |0000000001**1**0000000> + 0,20 |0000001111**1**0000000> +

0,20 |0001110001**1**0100000> - 0,46 |0001111111**1**0100000> -

0,20 |1110000001**1**0000000> + 0,46 |1110001111**1**0000000> -

0,46 |1111110001**1**0100000> + 0,20 |1111111111**1**0100000>

- Application de la porte CNot (5, 12):

|Psi⟩ = 0,46 |0000000001**1**0000000> + 0,20 |0000001111**1**0000000> +

0,20 |0001110001**1**0000000> - 0,46 |0001111111**1**0000000> -

0,20 |1110000001**1**0000000> + 0,46 |1110001111**1**0000000> -

0,46 |1111110001**1**0000000> + 0,20 |1111111111**1**0000000>

- Application de la porte CNot (5, 13):

|Psi⟩ = 0,46 |00000000010000000> + 0,20 |00000011110000000> +

0,20 |00011100010010000> -0,46 |00011111110010000> -

0,20 |11100000010000000> + 0,46 |11100011110000000> -

0,46 |11111100010010000> + 0,20 |11111111110010000>

- Application de la porte CNot (6, 13):

|Psi⟩ = 0,46 |00000000010**0**00000> + 0,20 |00000011110**0**00000> +

0,20 |00011100010**0**00000> -0,46 |00011111110**0**00000> -

0,20 |11100000010**0**00000> + 0,46 |11100011110**0**00000> -

0,46 |11111100010**0**00000> + 0,20 |11111111110**0**00000>

- Mesurer les deux Qubit (Q_{12} , Q_{13}) :

Résultat de mesure = 00 , donc pas de correction

- Application de la porte CNot (7, 14):

|Psi⟩ = 0,46 |00000000010000000> +0,20 |00000011110001000> +

0,20 |00011100010000000> -0,46 |00011111110001000> -

0,20 |11100000010000000> + 0,46 |11100011110001000> -

0,46 |11111100010000000> + 0,20 |11111111110001000>

- Application de la porte CNot (8, 14):

|Psi⟩ = 0,46|00000000010000000> + 0,20|00000011110000000> +

0,20 |00011100010000000> -0,46 |00011111110000000> -

0,20 |11100000010000000> + 0,46 |11100011110000000> -

0,46 |11111100010000000> + 0,20 |11111111110000000>

- Application de la porte CNot (8, 15):

|Psi⟩ = 0,46 |0000000010000000> + 0,20|0000011110000100> +

0,20 |0001110001000000> -0,46 |0001111111000100> -

0,20 |1110000001000000> + 0,46|1110001111000100> -

0,46 |1111110001000000> + 0,20|1111111111000100>

- Application de la porte CNot (9, 15):

|Psi⟩ = 0,46 |0000000010000000> + 0,20 |0000011110000000> +

0,20 |0001110001000000> -0,46 |0001111111000000> -

0,20 |1110000001000000> + 0,46 |1110001111000000> -

0,46 |1111110001000000> + 0,20 |1111111111000000>

- Mesurer les deux Qubit (Q_{14} , Q_{15}) :

Résultat de mesure = 00 , donc pas de correction

- Application des portes : CNot (1, 2), CNot (1, 3), CNot (4, 5), CNot (4, 6), CNot (7, 8), CNot (7, 9):

|Psi⟩ = 0,46 |0000000010000000> +0,20 |0000001001000000> +

0,20 |0001000001000000> -0,46 |0001001001000000> -

0,20 |1000000001000000> + 0,46|1000001001000000> -

0,46 |1001000001000000> + 0,20 |1001001001000000>

- Application de la porte H (1):

|Psi⟩ = 0,18 |0000000010000000> + 0,46 |1000000010000000> +

0,18 |0000001001000000> -0,46 |1000001001000000> -

0,18 |0001000001000000> + 0,46 |1001000001000000> -

0,18 |0001001001000000> -0,46 |1001001001000000>

- Application de la porte H (4):

$$|Psi\rangle = 0,25\ |00010000010000000\rangle + 0,66|10000000010000000\rangle +$$

$$0,25\ |00010010010000000\rangle - 0,66\ |10000010010000000\rangle$$

- Application de la porte H (7):

$$|Psi\rangle = 0,36\ |00010000010000000\rangle + 0,93\ |10000010010000000\rangle$$

- Application de la porte CNot (1, 16):

$$|Psi\rangle = 0,36\ |00010000010000000\rangle + 0,93\ |10000010010000010\rangle$$

- Application de la porte CNot (4, 16):

$$|Psi\rangle = 0,36\ |00010000010000010\rangle + 0,93|10000010010000010\rangle$$

- Application de la porte CNot (4, 17):

$$|Psi\rangle = 0,36\ |00010000010000011\rangle + 0,93|10000010010000010\rangle$$

- Application de la porte CNot (7, 17):

$$|Psi\rangle = 0,36\ |00010000010000011\rangle + 0,93|10000010010000011\rangle$$

- Mesurer les deux Qubit (Q_{16} , Q_{17}) :

Résultat de mesure = 11 , , donc erreur de type X2

- La correction : appliquer la porte X(2)

$$|Psi\rangle = 0,36\ |00000000010000011\rangle + 0,93|10010010010000011\rangle$$

- Suppression des deux Qubits des syndromes:

$$|Psi\rangle = 0,36\ |000000000\rangle + 0,93\ |100100100\rangle$$

- Application de la porte CNot (1, 4):

$$|Psi\rangle = 0,36\ |000000000\rangle + 0,93\ |100000100\rangle$$

- Application de la porte CNot (1, 7):

$$|Psi\rangle = 0,36\ |000000000\rangle + 0,9\ 3|100000000\rangle$$

$$= (\mathbf{0,36\ |\ 0\ \rangle + 0,93\ |\ 1\ \rangle})\ |00000000\rangle$$

Conclusion Générale

Les énormes difficultés qu'il y a à préserver les Qbits des influences du milieu environnant représentent l'entrave majeure à la réalisation d'un prototype d'ordinateur quantique. Le problème est si aigu que la solution n'est nullement écartée de l'adoption d'un protocole massif de corrections d'erreurs.

La correction automatique d'erreurs quantiques se fait sur le même principe qu'en théorie classique de l'information en recourant à des Qbits supplémentaires qui créent la redondance nécessaire. Toutefois le problème se complique en théorie quantique du fait que le clonage pur et simple est impossible. On ne peut donc vérifier l'information quantique portée par un état en la comparant à celle contenue dans plusieurs copies.

Dans ce travail, nous nous sommes intéressés à l'algorithme de Shor. Dans un premier temps, nous avons présenté une étude de cette solution. Ensuite, on a proposé une implémentation avec des exemples d'application. L'idée de base est la concaténation de deux codages à trois Qbits basés sur la répétition quantique. Cet algorithme permet la correction des erreurs de types X, Y et Z. Comme ces trois opérateurs, dits groupe de pauli, constituent une base de l'espace des matrices unitaires, n'importe quelle anomalie est donc corrigeable.

En termes de performance, cet algorithme est très efficace dans la correction. Mais, il nécessite l'utilisation de huit Qbits supplémentaires. Des recherches plus récentes ont donné naissance à des solutions plus optimales dites codes stabilisateurs. Elles sont basées principalement sur l'algèbre des opérateurs de Pauli. Cela constitue un sujet très intéressant pour les prochaines promotions.

Bibliographie

[1] Simon Perdrix, « Modèles formels du calcul quantique: ressources, machines abstraites et calcul par mesure», Institut National Polytechnique de Grenoble, Décembre 2006.

[2] A.Oum salem, « Spin transport in nanomaterials application to quantum computers »,thesis of Doctorate,university « Hasiba Ben Bouali »,Chlef,2008 .

[3] H.Talbi, «Algorithmes évolutionnaires quantiques pour le recalage et la segmentation multiobjectif d'images », Thèse de doctorat en sciences, Université Mentouri Constantine, 2009.

[4] Jozeph Gruska, « Quantum computing », McGraw-Hill, 1999.

[5] Emma Strubell, «An Introduction to Quantum Algorithms», COS498 – Chawathe, Spring 2011.

[6] M. Khalil. MEZGHICH, «Approche Quantique pour l'Appariement de Formes», Mémoire de Magister en Informatique, Université Mohamed Khider Biskra, 04 / 02 / 2015.

[7] K.Mahdi, «L'optimisation multi objectif et l'informatique quantique», Mémoire de Magister en Informatique, Université Mentouri- Constantine.

[8] Arun, «Hybrid Quantum Computation», A thesis Submitted for The Degree of Doctor of Philosophy, National University of Singapore 2011..

[9] Aaron .Krahn, «Quantum Computation and Grovers Algorithm», University of Chicago, Research Experiences for Undergraduate, 2012.

[10] Olivier Landon-Cardinal, «Evolution des systèmes quantiques ouverts: décohérence et informatique quantique», Université de Montr´ealKhider, Août, 2009.

[11] Nicolas Macris, «Traitement quantique de l'information», Ecole Polytechnique fédérale de Lausanne, 2013.

[12] Xavier Lacour, «Information Quantique par Passage Adiabatique: Portes Quantiques et décohérence», Université de Bourgogne, 03 octobre 2007.

[13] Dr.Ramazan KOC, «chapter4 Quantum circuits», Université Gaziantep.

[14] Anne Marin, « Borne inférieure sur la capacité d'un canal quantique », Rapport de stage, Centre de recherche Inria-Rocquencourt, 2009.