

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université de Mohammed Seddik Ben Yahya

Faculté des Sciences Exactes et Informatique
Département d'Informatique

Mémoire de Master en Informatique

Spécialité: Intelligence Artificielle

Thème

Détection de l'intrication des états quantiques

Encadré par

- Khalfaoui Khaled
- Kerkouche El Hillali

Réalisé par

- Bernou Cheyma
- Tebibel Islam

2021/2022

Remerciements

Nous remercions DIEUX le tout puissant de nous avoir donné le courage et la volonté d'achever ce travail et sans lequel il n'aurait jamais été accompli.

Nous remercions nos encadreurs Dr : Khalfaoui Khaled et Dr Kerkouche El Hillali pour leurs orientations et leurs conseils durant tout un semestre.

Nous voudrions également remercier tous les membres de jury ,pour l'honneur qu'ils nous ont fait en acceptant d'examiner ce travail ainsi que tous les enseignants qui nous ont accompagnés activement le long de nos années d'étude à l'université.

Finalement, un grand merci à tous nos collègues d'études, pour leur témoignage d'amitié et pour l'ambiance de travail qu'ils ont su créer.

Dédicaces

Je dédie ce mémoire :

A mes chers parents mon père et ma mère pour leur patience , leur amour, leur soutien et leur encouragements.

A mes chères sœurs : marwa, safa , aya et douaa.

A mes amies chacun de son nom.

A mon binôme islam ,pour les moments de joie et de peine qu'on à partager ensemble durant toutes la période de nos études.

Enfin, je voudrais dédier ce mémoire à tout personnes ayant participé de loin ou de près à la réalisation de ce travail.

Cheyma

Résumé

L'informatique quantique est un nouveau domaine de recherche visant à développer des technologies basées sur les principes de la théorie quantique. L'objectif principal est de tirer parti de certains concepts quantiques très puissants telle que l'intrication. Cette dernière, bien qu'elle ne soit pas encore entièrement comprise, est la pièce maîtresse de la plupart des protocoles développés. Dans ce travail, nous présentons un algorithme permettant de détecter cette propriété sans avoir recours aux formules mathématiques. Toutes les fonctionnalités nécessaires sont implémentées. Pour des fins illustratifs, des exemples d'application sont donnés.

Abstract

Quantum computing is a new field of research aimed at developing technologies based on the principles of quantum theory. The main objective is to take advantage of some very powerful quantum concepts such as entanglement. The latter, although not yet fully understood, is the centerpiece of most developed protocols. In this work, we present an algorithm to detect this property without resorting to mathematical formulas. All necessary features are implemented. For illustrative purposes, application examples are given.

Sommaire

Introduction générale	4
Chapitre 01 : Calcul quantique	4
1.1. Concepts de base	4
1.2. Opérations quantiques	4
1.3. Circuit quantique	4
Chapitre 02 : Algorithmes quantiques	4
2.1. Téléportation quantique	4
2.2. Factorisation des grands nombres entiers	4
2.3. Correction des erreurs quantiques	4
Chapitre 04 : Détection de l'intrication quantique	4
3.1.	
3.2.	
Chapitre 03 : Simulation quantique	4
3.1. Classes utilisées	4
3.2. Pseudo_Code des méthodes	4
Chapitre 05 : Exemples d'application	4
3.1. Classes Utilisées	
3.2. Pseudo_Code des Méthodes	
Conclusion générale	
Bibliographie	

Introduction Générale

Introduction générale

L'informatique quantique est un domaine de recherche visant à développer des technologies basées sur les principes de la théorie quantique [1,2] tels que le calcul algorithmique, la téléportation et la cryptographie. L'objectif principal est de tirer parti de certains concepts quantiques très puissants, plus particulièrement la superposition, la mesure et l'intrication. Cette dernière, bien qu'elle ne soit pas encore entièrement comprise, est la pièce maîtresse de la plupart des protocoles développés. C'est une ressource clé dans la téléportation quantique [3,4,5,6,7], la distribution de clés quantiques [8,9,10], le codage superdense [11,12,13], le partage de secrets quantiques [14, 15] et bien d'autres applications. Les Qubits intriqués sont intrinsèquement connectés et ont l'étrange façon de s'influencer instantanément quelle que soit la distance qui les sépare. Cela signifie notamment qu'un changement sur l'état d'un seul Qubit, comme sa mesure par exemple, induit le changement des états des autres Qubits même séparées dans l'espace. Cette influence à distance ne peut être expliquée ou interprétée en termes de concepts de la théorie classique.

Dans ce domaine de recherche, d'énormes efforts ont été faits pour mesurer le degré de cette propriété dans un état multipartite en introduisant des paramètres de quantification tels que la concurrence [16], le tangle [17] et l'information quantique de Fisher [18]. Mais, à l'heure actuelle, la détection de l'intrication est encore un problème de calcul difficile, en particulier pour les systèmes de grande dimension [19, 20] et les méthodes proposées ont généralement des coûts de calcul élevés [21, 22].

Un état quantique est dit intriqué s'il ne peut pas être écrit comme un produit tensoriel de deux états partiels partageant ses Qubits. Pour vérifier cette propriété, nous présentons dans ce mémoire un algorithme très récent proposé dans [23]. Les auteurs ont développé un itératif pour tester toutes les configurations possibles. Pour chaque cas, dans un premier temps, une matrice de coefficients est générée à partir de l'état global. Ensuite, deux états potentiels partageant des Qubits d'état globaux sont calculés. Si leur produit tensoriel donne l'état global alors ce dernier est séparable, sinon l'algorithme passe à la configuration suivante. Cette méthode a été discutée avec un aspect géométrique dans [24,25].

Ce mémoire est constitué de cinq chapitres :

- Le premier chapitre donne un aperçu sur les concepts de base du calcul quantique.
- Le deuxième chapitre est consacré aux algorithmes quantiques.
- Dans troisième chapitre, nous présentons l'algorithme de détection de l'intrication. La technique utilisée est expliquée à travers un exemple introductif.
- Au quatrième chapitre, l'implémentation de toutes les fonctionnalités nécessaires sera détaillée en donnant le pseudo-code des fonctions utilisées.
- Enfin, le dernier chapitre est une illustration de cette technique de détection via des exemples d'applications.

Chapitre 01

Calcul Quantique

1 Calcul quantique

Dans ce chapitre, nous introduisons tous les concepts principaux qui nous seront utiles tout au long de ce manuscrit en donnant un aperçu sur les notions mathématiques nécessaires. Nous présentons les états quantiques, l'évolution d'un système quantique ainsi l'opération de mesure. Tous les concepts mathématiques abordés dans ce chapitre sont détaillés dans [1,2].

1.1 Concepts de base :

1.1.1 Qubit :

En information quantique, l'élément le plus élémentaire d'information est le Qubit. A l'inverse du bit classique possédant deux états mutuellement exclusive 0 et 1, un Qubit est en superposition de ces deux états de base. Il est représenté par un vecteur à deux composantes complexes (dimension 2). En notation de Dirac, il est défini comme suit :

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle \quad (1)$$

$$|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad \text{et} \quad |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad , \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad (2)$$

Tel que :

$$\begin{aligned} - \alpha, \beta &\in \mathbb{C} \\ - |\alpha|^2 + |\beta|^2 &= 1 \end{aligned}$$

1.1.2 État quantique :

Un état quantique est un registre à n Qubits.

$$|\psi\rangle \in \underbrace{\mathbb{C}^2 \times \mathbb{C}^2 \dots \times \mathbb{C}^2}_{n \text{ fois}}. \quad (3)$$

Il s'agit d'une superposition donnée par :

$$|\psi\rangle = \sum_{x \in \{0,1\}^n} \alpha_x |x\rangle \quad (4)$$

Les coefficients α_x doivent vérifier la norme suivante :

$$\sum_{x \in \{0,1\}^n} |\alpha_x|^2 = 1 \quad (5)$$

A titre d'exemple, un état quantique à deux Qubits est un vecteur :

$$|\psi\rangle = \alpha_{00} |00\rangle + \alpha_{01} |01\rangle + \alpha_{10} |10\rangle + \alpha_{11} |11\rangle \quad (6)$$

- $\alpha_{ij} \in \mathbb{C}$
- $\sum |\alpha_{ij}|^2 = 1$
- $\left\{ |ij\rangle_{i,j=0,1} \right\}$: La base canonique de cet espace (dimension $2 \times 2 = 2^2$).

1.1.3 Évolution d'un état quantique :

Un état quantique de n Qubits évolue par des transformations unitaires U de la forme :

$$\begin{aligned} U: \mathbb{H} &\rightarrow \mathbb{H} \\ |\psi\rangle &\rightarrow |\phi\rangle = U |\psi\rangle \end{aligned}$$

Tel que :

$$U^+U = UU^+ = I_{\mathbb{H}} \quad (7)$$

$$\mathbb{H} = \underbrace{\mathbb{C}^2 \times \mathbb{C}^2 \dots \times \mathbb{C}^2}_{n \text{ fois}} \quad (8)$$

1.2 Opérations quantiques :

Les opérations possibles peuvent être des transformations simples ou contrôlées. En plus, on dispose de la mesure quantique. Elles sont représentées par des portes quantiques et les plus intéressantes sont :

1.2.1 Opérations quantiques simples :

Groupe de Pauli :

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (9)$$

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad (10)$$

$$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \quad (11)$$

$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad (12)$$

Porte Hadamard :

$$H = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{bmatrix} \quad (13)$$

Porte de rotation :

$$R_\theta = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{bmatrix} \quad (14)$$

Porte quantique de permutation :

$$Swap = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (15)$$

Cette transformation permet la permutation des positions de deux Qubits dans un état quantique :

$$- U_{Swap} |\psi\rangle |\phi\rangle = |\phi\rangle |\psi\rangle$$

1.2.2 Opérations quantiques contrôlées :

l'application de ces portes sur des Qubits cibles est conditionnée par l'état d'un ensemble de Qubits de contrôle. La plus simple est la porte CNot.

CNot:

$$CNot = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (16)$$

Cette transformation agit sur le deuxième Qubit selon l'état du premier Qubit comme suit :

$$\begin{aligned} - U_{CNot} |0\rangle |\psi\rangle &= |0\rangle I |\psi\rangle = |0\rangle I |\psi\rangle \\ - U_{CNot} |1\rangle |\psi\rangle &= |1\rangle X |\psi\rangle \end{aligned}$$

Pour un état quantique à n Qubits, la matrice de transformation est de taille $2^n * 2^n$. Elle est calculée par un produit tensoriel de matrices partielles correspondantes aux Qubits. Selon le type d'opération désirée, les Qubits concernés sont manipulés par des matrices partielles spécifiques alors que les autres restent identiques en appliquant la matrice unité.

1.2.3 La mesure quantique :

Le calcul quantique se distingue du calcul classique par la notion de mesure. Cette dernière occupe une place particulière et joue le rôle du complément de l'évolution quantique. Elle est utilisée pour casser la superposition des états quantiques.

Une mesure quantique est définie par un ensemble d'opérateurs $\{M_{(i)}\}$ vérifiant :

$$\sum M_{(i)}^+ M_{(i)} = I_H \quad (17)$$

- (i) est le résultat de la mesure d'un état physique $|\psi\rangle$

Juste avant la mesure, la probabilité d'avoir le résultat (i) est calculée par la formule :

$$\mathbb{P}(i) = \langle \psi | M_i^+ M_i | \psi \rangle \quad (18)$$

$$\sum \mathbb{P}(i) = 1 \quad (19)$$

Juste après la mesure, si le résultat est (k) alors l'état physique $|\psi_k\rangle$ est donné par :

$$\begin{aligned} M_{(k)} : \mathbb{H} &\rightarrow \mathbb{H} \\ |\psi\rangle &\rightarrow |\psi_{(k)}\rangle \end{aligned}$$

Tel que :

$$|\psi_{(k)}\rangle = \frac{M_{(k)} |\psi\rangle}{\sqrt{\langle \psi | M_{(k)}^+ M_{(k)} | \psi \rangle}} \quad (20)$$

1.3 Circuit quantique :

Un calcul quantique sur un état donné $|\psi_e\rangle$ est l'application d'un ensemble d'opérations quantiques et de mesures $\{U_i, M_{(j)}\}$ dans un ordre bien défini afin de générer un état de sortie $|\psi_s\rangle$.

$$\begin{aligned} \{U_i, M_{(j)}\} : \mathbb{H} &\rightarrow \mathbb{H} \\ |\psi_e\rangle &\rightarrow |\psi_s\rangle \end{aligned}$$

Tel que :

$$|\psi_s\rangle = \{U_i, M_{(j)}\} |\psi_e\rangle \quad (21)$$

Ayant en entrée un registre quantique de n Qubits, un circuit quantique est constitué d'un ensemble d'étages successifs. A chaque niveau k, les Qubits sont manipulés par des opérations simples, contrôlées ou de mesure. Le nouveau état $|\psi_k\rangle$ est calculé à partir de l'ancien état $|\psi_{k-1}\rangle$ en appliquant une matrice de transformation de taille $2^n * 2^n$. Ce traitement est réitéré du premier niveau jusqu'au dernier.

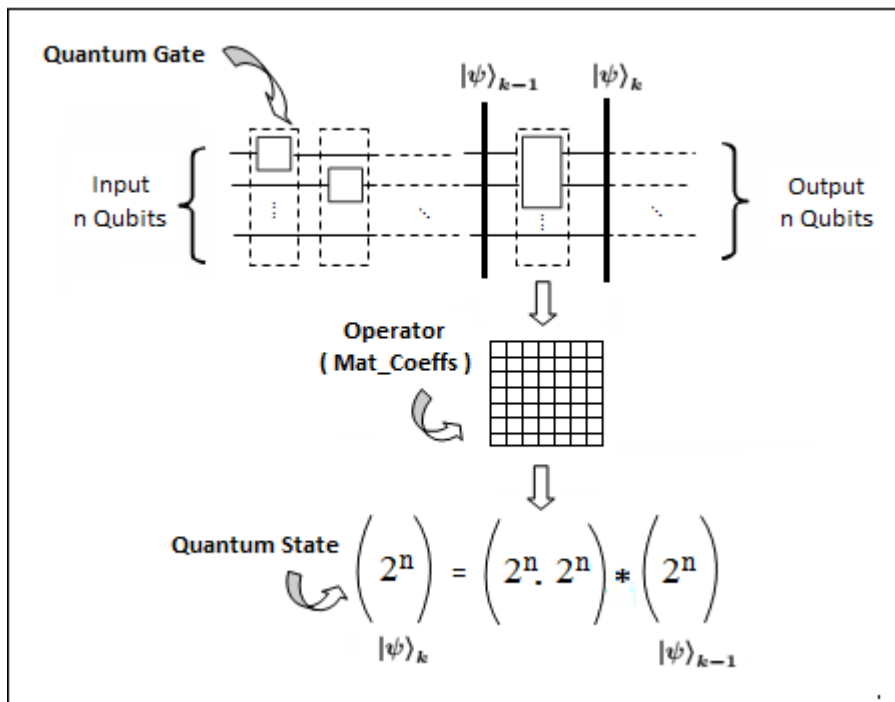


Fig. 1 Calcul Quantique

Chapitre 02

Algorithmes Quantiques

2 Algorithmes quantiques

2.1 Téléportation quantique

La téléportation quantique est une technique qui permet de transférer des états quantiques entre des localités distantes sans l'utilisation de canaux physiques. Elle est basée principalement sur le phénomène d'intrication et les opérations de mesure. Depuis le papier de Bennett et Brassard [26], qui contient le résultat fondamental et original, cet extraordinaire exploit n'a cessé d'être augmenté et parfois changé de forme en rajoutant à son protocole des extensions telles que le contrôle [27], la bi-direction [28], la rotation [29] en plus des augmentations du nombre des Qubits à téléporter, séparés ou intriqués.

Ces enrichissements des protocoles requièrent la recherche des canaux quantiques correspondants ainsi que les opérations satisfaisant les fonctionnalités demandées. Cela a suscité l'intérêt de la communauté et on voit alors apparaître un nombre important de travaux concernant ce thème de recherche. Dans cette section, on se limite à la présentation du protocole de Brassard.

2.1.1 Protocole original

Supposons que l'émetteur s'appelle Alice et que le récepteur s'appelle Bob. Alice dispose d'un Qubit quelconque $|Q\rangle$:

$$|Q\rangle = \alpha |0\rangle + \beta |1\rangle \quad (22)$$

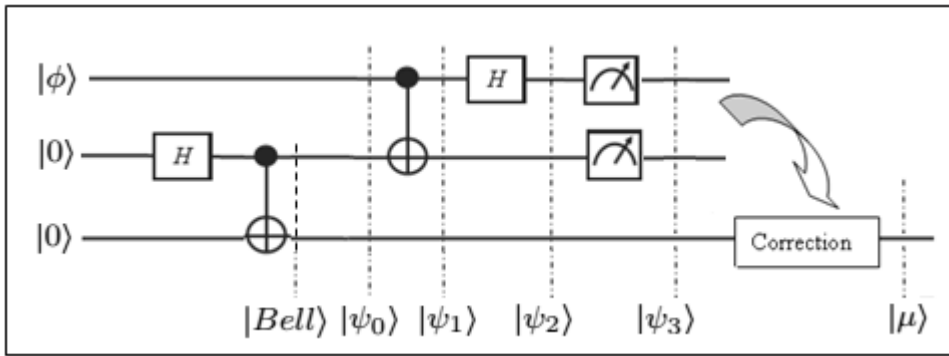


Fig. 2 Protocole original de Brassard

Elle veut transmettre son Qubit à Bob par téléportation quantique. Pour ce faire, Brassard a proposé le circuit quantique suivant :

Ce schéma est constitué des opérations suivantes :

- Fabriquer un état de Bell. C'est une paire de Qubits intriquée utilisée comme canal quantique.
- Partager ces deux Qubits entre Alice et Bob.
- Alice réalise quelques opérations quantiques sur ces deux Qubits.
- Alice réalise une opération de mesure sur ces deux Qubits et envoie le résultat à Bob par un canal classique.
- Bob corrige son Qubit selon le résultat de mesure d'Alice afin de reconstituer le Qubit $|\psi\rangle$.

2.1.2 Simulation du protocole

En supposons que l'état de Bell utilisé comme canal quantique est :

$$|\psi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \quad (23)$$

les deux tables suivantes donnent tous les résultats obtenus en appliquant ce protocole.

Table 1 Résultats du protocole de Brassard : avant la mesure d'Alice

$ Q\rangle$	$\alpha 0\rangle + \beta 1\rangle$
$ Bell\rangle$	$\frac{1}{\sqrt{2}}(00\rangle + 11\rangle)$
$ \psi_0\rangle$	$\frac{1}{\sqrt{2}}(\alpha 000\rangle + \alpha 011\rangle + \beta 100\rangle + \beta 111\rangle)$
$ \psi_1\rangle$	$\frac{1}{\sqrt{2}}(\alpha 000\rangle + \alpha 011\rangle + \beta 101\rangle + \beta 110\rangle)$
$ \psi_2\rangle$	$\frac{1}{2}(\alpha 000\rangle + \beta 001\rangle + \alpha 011\rangle + \beta 010\rangle + \alpha 100\rangle - \beta 101\rangle - \beta 110\rangle + \alpha 111\rangle)$

Remarque : Ce résultat de simulation montre que le Qubit d'Alice est parfaitement téléporté à Bob.

Table 2 Résultats du protocole de Brassard : après la mesure d'Alice

Résultats de mesure d'Alice	$ \psi_3\rangle$	Qubit de Bob ($ \mu\rangle$)	Correction appropriée
$ 00\rangle$	$\alpha 000\rangle + \beta 001\rangle$	$\alpha 0\rangle + \beta 1\rangle$	I
$ 01\rangle$	$\alpha 011\rangle + \beta 010\rangle$	$\beta 0\rangle + \alpha 1\rangle$	X
$ 10\rangle$	$\alpha 100\rangle - \beta 101\rangle$	$\alpha 0\rangle - \beta 1\rangle$	Z
$ 11\rangle$	$-\beta 110\rangle + \alpha 111\rangle$	$\beta 0\rangle - \alpha 1\rangle$	Z X

2.2 Factorisation des grands nombres entiers

En informatique, la factorisation d'un grand nombre entier dans un temps raisonnable est d'une grande importance. Mais, en utilisant le calcul classique, cette tâche reste encore impossible malgré la puissance phénoménale des calculateurs existants. Pour remédier à cet obstacle complexe, Peter Shor a élaboré un algorithme basé sur le calcul quantique permettant de résoudre ce problème dans un temps polynomial [30]. Il a proposé une solution itérative (calcul classique) qui fait appel à un traitement quantique.

2.2.1 Algorithme principale

L'algorithme principal est le suivant tel que les fonctions utilisées sont :

- PGCD(Nbr_01,Nbr_02): Traitement classique qui donne le plus grand commun diviseur de deux entiers Nbr_01 et Nbr_02.
- Puissance(Nbr_01,Nbr_02) : Traitement classique qui donne Nbr_01 à la puissance Nbr_02.
- Nbr_01 Mod Nbr_02 : Traitement classique qui donne le reste de la division de Nbr_01 par Nbr_02.
- Période_Quantique (a , N) : Traitement quantique qui donne la période de la fonction $a^x \text{ Mod } N$.

:

Factorisation Quantique des Nombres Entiers

```

void Factoriser ( N : Entier )
{
  Fin = faux ;
  TQ ( ! Fin )
  {
    Choisir un entier a dans lintervalle [2,N-1] ;
    Si ( PGCD (N,a) != 1 )
      {
        Fact_01 = PGCD (N,a);
        Fact_02 = N / Fact_01;
        Fin = Vrai ;
        Afficher ( Fact_01 , Fact_02);
      }
    Sinon
      {
        r = Priode_Quantique ( a , N );
        Si ( r > 0 )
          {
            Si ( ( r mod 2 ) = 0 )
              {
                Puiss_r = Puissance( a , (r / 2));
                Modulo = ( Puiss_r mod ( N )) ;
                Si ( ( Modulo != 1 ) et ( Modulo != (N-1) ) )
                  {
                    Fact_01 = PGCD( Puiss_r + 1 , N );
                    Fact_02 = PGCD( Puiss_r - 1 , N );
                    Afficher ( Fact_01 , Fact_02);
                    Fin = Vrai ;
                  }
              }
          }
      }
  }
}

```

2.2.2 Transformée de Fourier quantique

Pour calculer la période de la fonction $a^x \text{ Mod } N$, on procède par les étapes suivantes :

- Préparer les termes de la fonction $a^x \text{ Mod } N$ sous forme d'un état quantique.
- Appliquer la transformée de Fourier quantique sur cet état quantique.
- Appliquer une opération de mesure quantique. Le résultat est une fréquence.
- Calculer la période en utilisant les technique des fractions continues. Ce dernier calcul est classique.

Dans le cadre de ce memoire, on se milite à la présentation du calcul de la transformée de Fourier quantique. Le circuit permettant de réaliser

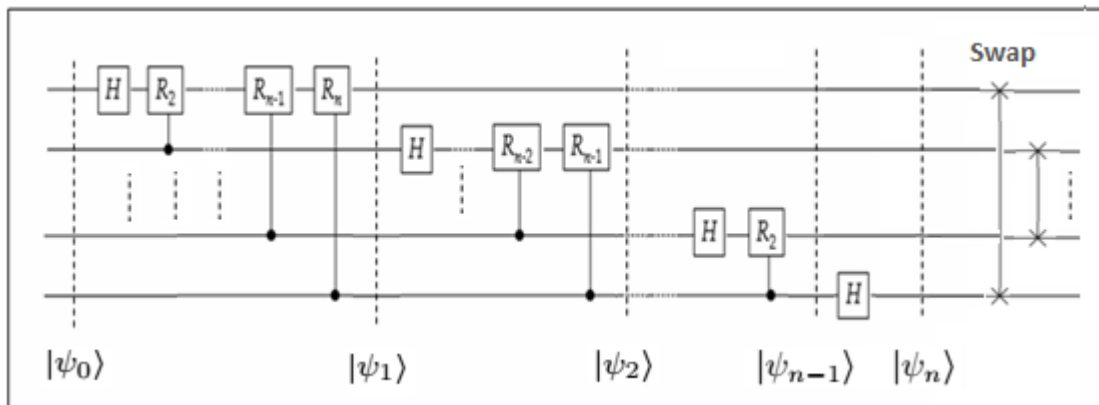


Fig. 3 Circuit de la Transformée de Fourier Quantique

ce traitement est donné dans la figure Fig.3 et le code de simulation correspondant est le suivant :

Simulation de la Transformée de Fourier Quantique

```
void QFT ( Quantum_State Psi )
{
  for ( int i = 0 ; i < Psi.Nbr_Qubits ; i ++ )
  {
    Psi.H(i) ;
    int k = 2 ;
    for ( int j= i + 1 ; j < Psi.Nbr_Qubits ; j++ )
    {
      Psi.C_R ( j , i , k ) ;
      k = k + 1 ;
    }
  }
  for ( int i = 0 ; i < Psi.Nbr_Qubits / 2 ; i++ )
  {
    Psi.Swap ( i , ( Psi.Nbr_Qubits - 1 ) - i ) ;
  }
}
```

2.3 Correction des erreurs quantiques

Afin de protéger les Qubits des effets inévitables liés à l'interaction d'un système quantique avec son environnement, la correction d'erreurs est une tâche indispensable. Une des solutions les plus intéressantes est celle des codes stabilisateurs [31]. Elle est basée sur un encodage préalable du Qubit à protéger. La détection d'erreurs est réalisée en

calculant un ensemble de syndromes. Pour introduire les codes stabilisateurs, prenons un cas très simple. Soit le Qubit $|\psi\rangle$:

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

L'étape d'encodage consiste au rajout de deux Qubits supplémentaires intriqués avec le Qubit $|\psi\rangle$.

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

Supposons maintenant que le seul type d'erreurs qui peuvent affecter le $|\psi_c\rangle$ soit un bit-flip (X). Quatre cas sont possibles :

- $X_1 (\alpha |000\rangle + \beta |111\rangle) = \alpha |100\rangle + \beta |011\rangle$
- $X_2 (\alpha |000\rangle + \beta |111\rangle) = \alpha |010\rangle + \beta |101\rangle$
- $X_3 (\alpha |000\rangle + \beta |111\rangle) = \alpha |001\rangle + \beta |110\rangle$
- Aucune erreur.

La technique basée sur les codes stabilisateurs procède de la manière suivante :

- On définit un ensemble d'opérateurs particuliers U_i dits opérateurs de syndrome tels que :
 - Les états erronés ainsi que l'état sans erreur sont des vecteurs propres de ces opérateurs.
 - Pour chaque cas possible de $|\psi_c\rangle$, les valeurs propres associées aux opérateurs U_i sont différentes des autres cas. Elle sont appelées syndrome.
 - Tous les opérateurs U_i commutent.
- Au cours du calcul quantique, pour vérifier la présence d'erreur, on mesure le syndrome associé à $|\psi_c\rangle$ en utilisant opérateurs U_i . La valeur obtenue permet de déterminer l'état de $|\psi_c\rangle$ parmi les quatre cas précédents sans ambiguïté. Il suffit donc d'apporter la correction appropriée.

Le calcul de la valeur propre associée à chaque opérateur U est réalisé par la technique suivante :

- On ajoute un Qubit auxiliaire initialisé à $H |0\rangle$. L'état global devient :

$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) |\psi_c\rangle$$

- On applique une opération U contrôlée sur les Qubits de $|\psi_c\rangle$ ayant comme Qubit de contrôle le Qubit auxiliaire. On aura donc :

$$\frac{1}{\sqrt{2}}(|0\rangle |\psi_c\rangle + |1\rangle U |\psi_c\rangle)$$

- On applique de nouveau une porte Hadamard sur le Qubit auxiliaire, ce qui donne :

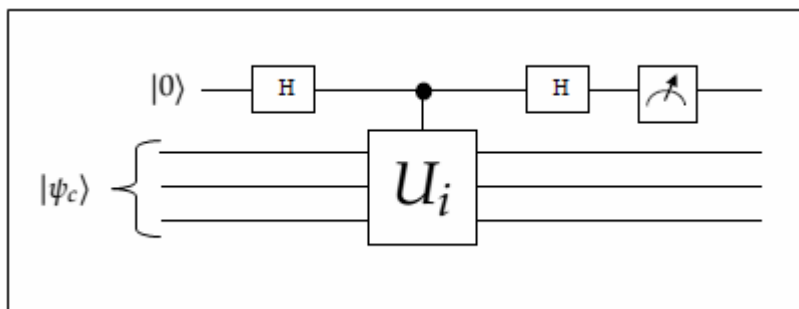


Fig. 4 Circuit de Calcul du Syndrome

$$\frac{1}{2} |0\rangle (|\psi_c\rangle + U |\psi_c\rangle) + \frac{1}{2} |1\rangle (|\psi_c\rangle - U |\psi_c\rangle)$$

- On mesure ensuite le Qubit auxiliaire.
 - Si on obtient 0, la valeur propre est 1.
 - Par contre, si on obtient 1, la valeur propre est -1.

Dans notre cas, nous utilisons les opérateurs de syndrome : $U_1 = Z_1 Z_2$ et $U_2 = Z_1 Z_3$. Le tableau ci-dessous résume les syndromes associés aux quatre cas possibles de $ket\psi_c$.

Table 3 Téléportation quantique contrôlée : Résultats obtenus après la mesure d’Alice

$ \psi_c\rangle$	Valeur propre associé à U_1	Valeur propre associé à U_2
$\alpha 000\rangle + \beta 111\rangle$	+1	+1
$\alpha 100\rangle + \beta 011\rangle$	-1	-1
$\alpha 010\rangle + \beta 101\rangle$	-1	+1
$\alpha 001\rangle + \beta 110\rangle$	+1	-1

Remarques :

- Les deux opérateurs $Z_1 Z_2$ et $Z_1 Z_3$ forment un groupe S, appelé stabilisateur du code.
- $Z_1 Z_2$ et $Z_1 Z_3$ sont appelés générateurs de ce groupe.
- Les opérations quantiques nécessaires dans l’étape de décodage sont les mêmes utilisées dans le processus d’encodage, mais elles doivent être appliquées dans l’ordre inverse.

Chapitre 03

Détection de l'intrication Quantique

3 Détection de l'intrication quantique

L'informatique quantique s'est avérée très efficace dans plusieurs domaines. Outre le parallélisme des opérations, ce nouveau paradigme est doté de mécanismes très puissants tels que l'intrication. Ce phénomène assure une dépendance mutuelle et instantanée des Qubits intriqués quelles que soient les distances qui les séparent. Cela est d'une grande importance dans plusieurs applications. récemment dans [23], les auteurs ont élaboré une technique automatique permettant de vérifier cette propriété sans avoir recours aux formules mathématiques. L'algorithme proposé sera détaillé dans ce chapitre.

3.1 Principe de base

Pour plus de clarté, nous préférons d'abord expliquer la technique utilisée à travers un exemple explicatif. Considérons un état quantique $|\psi\rangle$ produit tensoriel de deux états $|\phi\rangle$ et $|\rho\rangle$:

$$|\psi\rangle = |\phi\rangle \otimes |\rho\rangle \quad (24)$$

Au départ on dispose uniquement des coefficients de l'état $|\psi\rangle$ et du nombre de Qubits de l'état $|\phi\rangle$. Prenons l'exemple ou :

- L'état $|\psi\rangle$ est constitué de 5 Qubits
- L'état $|\phi\rangle$ est constitué de 3 Qubits.
- Donc, l'état $|\rho\rangle$ est constitué de 2 Qubits.

Dans le cas le plus général, ces trois états quantique s'écrivent :

$$|\phi\rangle = \alpha_0 |000\rangle + \alpha_1 |001\rangle + \alpha_2 |010\rangle + \alpha_3 |011\rangle + \alpha_4 |100\rangle + \alpha_5 |101\rangle + \alpha_6 |110\rangle + \alpha_7 |111\rangle \quad (25)$$

$$|\rho\rangle = \beta_0 |00\rangle + \beta_1 |01\rangle + \beta_2 |10\rangle + \beta_3 |11\rangle \quad (26)$$

$$\begin{aligned}
|\psi\rangle = & \alpha_0\beta_0 |00000\rangle + \alpha_0\beta_1 |00001\rangle + \alpha_0\beta_2 |00010\rangle + \alpha_0\beta_3 |00011\rangle + \\
& \alpha_1\beta_0 |00100\rangle + \alpha_1\beta_1 |00101\rangle + \alpha_1\beta_2 |00110\rangle + \alpha_1\beta_3 |00111\rangle + \\
& \alpha_2\beta_0 |01000\rangle + \alpha_2\beta_1 |01001\rangle + \alpha_2\beta_2 |01010\rangle + \alpha_2\beta_3 |01011\rangle + \\
& \alpha_3\beta_0 |01100\rangle + \alpha_3\beta_1 |01101\rangle + \alpha_3\beta_2 |01110\rangle + \alpha_3\beta_3 |01111\rangle + \\
& \alpha_4\beta_0 |10000\rangle + \alpha_4\beta_1 |10001\rangle + \alpha_4\beta_2 |10010\rangle + \alpha_4\beta_3 |10011\rangle + \\
& \alpha_5\beta_0 |10100\rangle + \alpha_5\beta_1 |10101\rangle + \alpha_5\beta_2 |10110\rangle + \alpha_5\beta_3 |10111\rangle + \\
& \alpha_6\beta_0 |11000\rangle + \alpha_6\beta_1 |11001\rangle + \alpha_6\beta_2 |11010\rangle + \alpha_6\beta_3 |11011\rangle + \\
& \alpha_7\beta_0 |11100\rangle + \alpha_7\beta_1 |11101\rangle + \alpha_7\beta_2 |11110\rangle + \alpha_7\beta_3 |11111\rangle +
\end{aligned}
\tag{27}$$

A la forme de l'Eq.27, une matrice Mat_Coeffs est associée de taille $2^3 \times 2^2$ dont les coefficients sont notés $\alpha_i\beta_j$ $\{i = 0..7, j = 0..3\}$. Dans cette table :

- Chaque colonne est constituée des coefficients de l'état $|\phi\rangle$ multipliés par le même nombre complexe.
- Chaque ligne est constituée des coefficients de l'état $|\rho\rangle$ multipliés par le même nombre complexe.

Ces deux propriétés ont permis de dégager un critère pratique de séparabilité en prenant le chemin inverse. En effet, les données sont :

- $|\psi\rangle = \sum_{k=0}^{2^n-1} c_k |k\rangle$ tel que n est le nombre des Qubits de $|\psi\rangle$.
- Le nombre des Qubits de $|\phi\rangle$: m.

Pour tester la séparabilité de $|\psi\rangle$, les auteurs ont proposé la technique suivante :

- A partir des coefficients c_k de l'état $|\psi\rangle$, construire la matrice Mat_Coeffs. Its size is $2^m \times 2^{n-m}$.
- Choisir un coefficient non nul c_{ij} de cette matrice.
- Construire un état quantique normalisé $|\phi\rangle$ de m Qubits avec les coefficients de la colonne j.
- Construire un état quantique normalisé $|\rho\rangle$ de (n-m) Qubits avec les coefficients de la ligne i.
- Vérifier l'égalité : $|\psi\rangle = |\phi\rangle \otimes |\rho\rangle$
 - Si cette équation est vérifiée : $|\psi\rangle$ est séparable.
 - Sinon, $|\psi\rangle$ est un état intriqué.

En pratique, comme données, on dispose uniquement de l'état $|\psi\rangle$. Pour tester son intrication, les auteurs ont proposé un algorithme permettant de tester tous les cas de figure possibles en variant la répartition des Qubits de $|\psi\rangle$ entre les deux états $|\phi\rangle$ et $|\rho\rangle$. Il sera détaillé dans la section suivante.

3.2 Algorithme de détection d'intrication

Cet algorithme procède par un traitement itératif (Fig.5), et à chaque itération les opérations réalisées sont :

- Encoder le partage des Qubits de l'état initial $|\psi\rangle$ en utilisant une chaîne de caractères binaire tel que :
 - Les bits de valeur 1 spécifient les Qubits affectés à $|\phi\rangle$.
 - Les bits de valeur 0 spécifient les Qubits affectés à $|\rho\rangle$.
- Réorganisez les Qubits de $|\psi\rangle$ de sorte que les Qubits assignés à $|\phi\rangle$ soient placés au début avant ceux de $|\rho\rangle$.
- Appliquer la technique de vérification de la séparabilité.

L'implémentation de cet algorithme est la suivante tel que les fonctions auxiliaires utilisées sont :

- Rep_Binaire (i , Nbr_Qubits_Psi) : Cette fonction permet de générer la représentation binaire d'un entier i sous forme d'une chaîne de caractères de taille Nbr_Qubits_Psi.
- Nombre_des_Uns(Affectation_Qubits) : Cette fonction calcule le nombre de Qubits attribués à $|\phi\rangle$.
- Rearrangement_Qubits (Psi , Affectation_Qubits , Nbr_Qubits_Phi) : Cette fonction est utilisée pour réorganiser les Qubits de $|\psi\rangle$. Les Qubits affectés à $|\phi\rangle$ sont déplacés au début.
- Est_Séparable(Nouv_Psi , Nbr_Qubits_Phi) : Cette fonction est utilisée pour vérifier le critère de séparabilité de $|\psi\rangle$ avec Nbr_Qubits_Phi est le nombre de Qubits du premier état.

3.3 Implémentation

Pour illustrer cet algorithme, nous avons implémenté une version simplifiée d'un simulateur quantique en langage Java. L'environnement de développement utilisé est NetBeans. Cette application dispose de toutes les fonctionnalités requises dans le cadre de ce travail et les classes principales proposées sont les suivantes :

- Etat_Quantique : Cette classe est dédiée à la manipulation des états quantiques. C'est un tableau d'états de base (Fig.6). Pour un état quantique à n Qubits, ce vecteur test de taille 2^n . Cette classe dispose d'un seul attribut Nbr_Qubits. Elle est munie d'un ensemble de méthodes, les plus intéressantes sont :
 - Constructeurs :
 - Etat_Quantique (Nbr_Qubits_P).
 - Etat_Quantique (Etat_Quantique_P).
 - Mise_A_Jour (Etat_Base_P , Indice_P).

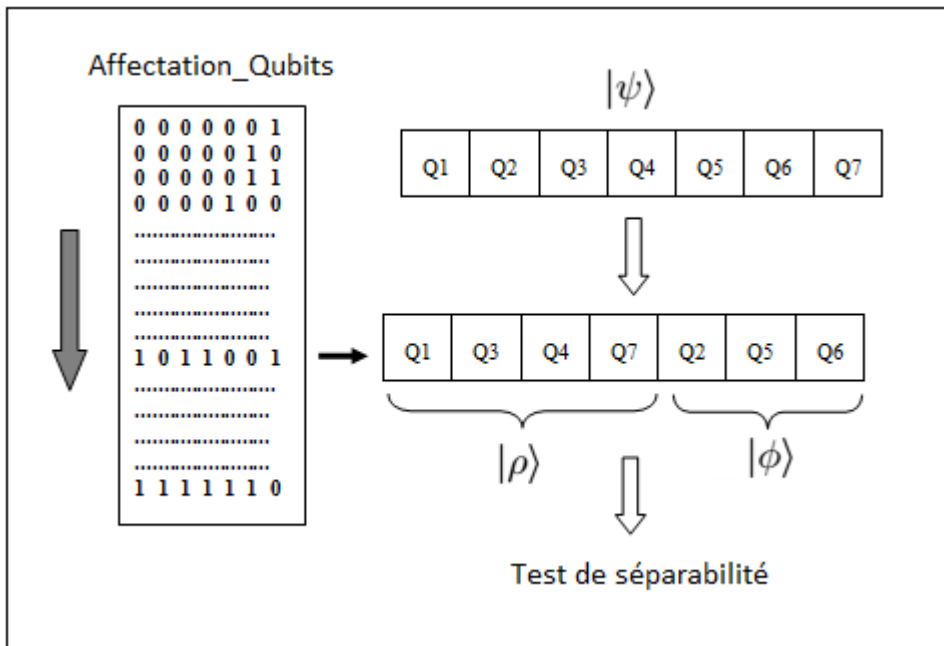


Fig. 5 Quantum Entanglement Detection

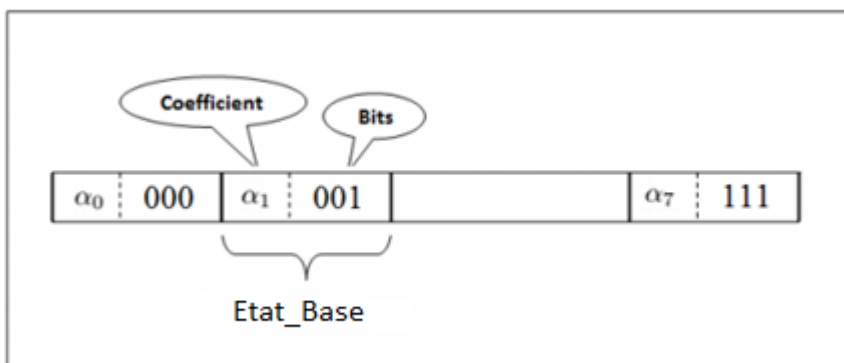


Fig. 6 Structure d'un état quantique

- Produit_Tensoriel (Etat_Quantique_01, Etat_Quantique_02).
- Toutes les méthodes nécessaires à l'implémentation des portes quantiques utilisées : X (Qubit_Cible) , Y (Qubit_Cible), Z (Qubit_Cible), H (Qubit_Cible) et CNot (Qubit_Controle,Qubit_Cible).
- Projection (Projecteur, Nbr_Qubits_Projetés).
- Normaliser().
- Afficher().

- Etat_Base : Cette classe est utilisée pour la création et la manipulation des états élémentaires. Chaque objet de cette classe se caractérise par les attributs suivants :
 - Bits : Chaîne de caractère contenant des bits.
 - Coefficient : Le coefficient associé à cet état élémentaire. Dans le cadre de ce travail, on s'est limité à l'utilisation des réels.
- Les méthodes de cette classe sont :
- Constructeur (Bits_P , Coefficient_P).
 - Afficher ().

Dans ce qui suit, nous présentons toutes les fonctions utilisées dans l'implémentation de l'algorithme de détection de l'intrication en donnant le pseudo-code correspondant.

3.4 La méthode Est_Intriqué :

La fonction Est_Intriqué

```

boolean Est_Intriqu( Etat_Quantique Psi)
{
    boolean Etat_Intriqu = Vrai ;

    int Nbr_Qubits_Psi = Psi.Nbr_Qubits ;

    for ( int i = 1 , i < ( 2 * Pow(Nbr_Qubits_Psi) ) - 1 , i ++ )
    {
        String Affectation_Qubits = Rep_Binaire ( i , Nbr_Qubits_Psi ) ;

        int Nbr_Qubits_Phi = Nombre_des_Uns ( Affectation_Qubits ) ;

        Etat_Quantique Nouv_Psi = Rearrangement_Qubits ( Psi ,
                                                         Affectation_Qubits , Nbr_Qubits_Phi ) ;

        if ( est_Sparable( Nouv_Psi , Nbr_Qubits_Phi )
            { Etat_Intriqu = Faux ; }
        }

    return( Etat_Intriqu );
}

```

3.5 La méthode Rep_Binaire :

La fonction Rep_Binaire

```
String Rep_Binaire ( i , Nbr_Qubits_Psi )
{
    String Resultat = "" ;
    int Cpt_Bits = 0 ;
    int N = i ;
    while ( N != 0 )
    {
        int Reste = N % 2 ;
        Resultat = "" + Reste + Resultat ;
        Cpt_Bits = Cpt_Bits + 1 ;
        N = N / 2 ;
    }
    while ( Cpt_Bits < Nbr_Bit_P )
    {
        Resultat = "0" + Resultat ;
        Cpt_Bits = Cpt_Bits + 1 ;
    }
    return(Resultat);
}
```

3.6 La méthode Nombre_des_Uns :

La fonction Nombre_des_Uns

```
int Nombre_des_Uns ( Affectation_Qubits )
{
    int Cpt_Uns = 0 ;

    int i ;
    for ( i = 0 ; i < Affectation_Qubits.length() ; i ++ )
    {
        if ( Affectation_Qubits( i ) == '1' )
        {
            Cpt_Uns = Cpt_Uns + 1 ;
        }
    }
    return ( Cpt_Uns ) ;
}
```

3.7 La méthode Gen_Mat_coeffs :

La fonction Gen_Mat_coeffs

```
double [ ][ ] Gen_Mat_coeffs ( Nouv_Psi , Nbr_Lignes , Nbr_Colonne )
{
    double [ ][ ] Mat_Coeffs = new double [Nbr_Lignes][Nbr_Colonne];

    int ind_Psi = 0;

    for ( int i = 0 ; i < N ; i ++ )
    {
        for ( int j = 0 ; j < M ; j++)
        {
            Mat_Coeffs[i][j] = Nouv_Psi.Tab_Etats_Base[ind_Psi].Coeff;
            ind_Psi = ind_Psi + 1;
        }
    }

    return( Mat_Coeffs ) ;
}
```

3.8 La méthode `Rearrangement_Qubits` :La fonction `Rearrangement_Qubits`

```
Etat_Quantique Rearrangement_Qubits(Psi, Affectation_Qubits, Nbr_Qubits_Phi)
{
    Etat_Quantique Nouv_Psi = new Etat_Quantique ( Psi ) ;
    int Qubit_Courant = 0 ;   int Cpt_Qubits_Phi = 0 ;

    while ( Cpt_Qubits_Phi < Nbr_Qubits_Phi )
    {
        while ( (Affectation_Qubits(Qubit_Courant) == '1') &&
                (Cpt_Qubits_Phi < Nbr_Qubits_Phi) )
        {
            Cpt_Qubits_Phi = Cpt_Qubits_Phi + 1 ;
            Qubit_Courant = Qubit_Courant + 1 ;
        }

        if( Cpt_Qubits_Phi < Nbr_Qubits_Phi )
        {
            int j = Qubit_Courant + 1 ;
            while (Affectation_Qubits(Qubit_Courant) == '0' )
            {
                j = j + 1 ;
            }
            Nouv_Psi.Swap ( Qubit_Courant , j ) ;

            Affectation_Qubits( Qubit_Courant ) = '1' ;
            Affectation_Qubits( j ) = '0' ;

            Cpt_Qubits_Phi = Cpt_Qubits_Phi + 1 ;
            Qubit_Courant = Qubit_Courant + 1 ;
        }
    }

    return( Nouv_Psi ) ;
}
```

3.9 La méthode est_Separable :

La fonction est_Separable

```

boolean est_Separable( Nouv_Psi , Nbr_Qbits_Phi)
{
    boolean Etat_Separable ;

    int Nbr_Qbits_Rho = Nouv_Psi.Nbr_Qubits - Nbr_Qbits_Phi ;

    int Nbr_Lignes = Puissance ( 2 , Nbr_Qbits_Phi ) ;
    int Nbr_Colonnes = Puissance ( 2 , Nbr_Qbits_Rho ) ;

    double [][] Mat_Coeff = Gen_Mat_coeffs ( Nouv_Psi , Nbr_Lignes ,
    Nbr_Colonnes ) ;

    int Ind_i, ind_j ;

    for ( int i = 0 ; i < Nbr_Lignes ; i ++ )
    {
        for ( int j = 0 ; j < Nbr_Colonnes ; j ++ )
        {
            if ( Mat_Coeffs [i][j] != 0 )
            {
                Ind_i = i ; ind_j = j ;
            }
        }
    }

    Etat_Quantique Phi_Candidat = new Etat_Quantique ( Nbr_Lignes ) ;
    for ( int i = 0 ; i < Nbr_Lignes ; i ++ )
    {
        Phi_Candidat.Tab_Etats_Bases[i].Coeff = Mat_Coeffs [i][Ind_j] ;
    }
    Phi_Candidat.Normaliser ( ) ;

    Etat_Quantique Rho_Candidat = new Etat_Quantique ( Nbr_Colonnes ) ;
    for ( int j = 0 ; j < Nbr_Colonnes ; j ++ )
    {
        Rho_Candidat.Tab_Etats_Bases[j].Coeff = Mat_Coeff [Ind_i][j] ;
    }
    Rho_Candidat.Normaliser ( ) ;

    Etat_Quantique Produit = Produit_Tensoriel ( Phi_Candidat , Rho_Candidat )
    if ( Egalit ( Nouv_Psi , Produit )
    {
        Etat_Separable = Vrai ;
    }

    else
    {
        Etat_Separable = Faux ;
    }

    return( Etat_Separable ) ;
}

```


3.10 La méthode Produit_Tensoriel :

La méthode Produit_Tensoriel

```

Etat_Quantique Produit_Tensoriel(Etat_Quantique EQ_01,Etat_Quantique EQ_02)
{
    Etat_Quantique Resultat;
    Resultat = new Etat_Quantique( EQ_01.Nbr_Qubits+EQ_02.Nbr_Qubits );

    for (int i = 0 ; i < Puiss ( 2 , EQ_01.Nbr_Qubits ) ; i ++ )
    {
        Etat_Base Etat_Base_i = EQ_01.elementAt(i);
        String Bits_i = Etat_Base_i.Bits;
        double Coeff_i = Etat_Base_i.Coefficient ;

        for (int j = 0 ; j < Puiss ( 2 , EQ_02.Nbr_Qubits ) ; j ++ )
        {
            Etat_Base Etat_Base_j = EQ_02.elementAt(j);
            String Bits_j = Etat_Base_j.Bits;
            double Coeff_j = Etat_Base_j.Coefficient ;

            String Bits_i_j = Bits_i + Bits_j ;
            double Coeff_i_j = Coeff_i * Coeff_j ;
            Etat_Base Etat_Base_i_j = new Etat_Base ( Bits_i_j , Coeff_i_j );

            int Ind_i_j = ( i * Puiss(2,EQ_01.Nbr_Qubits) ) + j ;

            Resultat.Mise_A_Jour( Etat_Base_i_j , Ind_i_j ) ;
        }
    }
    return ( Resultat ) ;
}

```

3.11 La méthode Normaliser :

La méthode Normaliser

```
void Normaliser ( )
{
    double Somme_Carrs_Normes = 0 ;
    for (int i = 0 ; i < Puiss ( 2 , this.Nbr_Qbits ) ; i ++ )
    {
        Etat_Base Etat_Base_i = this.elementAt(i);
        double Coeff_i = Etat_Base_i.Coefficient ;

        Somme_Carrs_Normes = Somme_Carrs_Normes + Puiss ( Coeff_i,2) ;
    }

    for (int i = 0 ; i < Puiss ( 2 , EQ_01.Nbr_Qbits ) ; i ++ )
    {
        Etat_Base Etat_Base_i = this.elementAt(i);

        Etat_Base_i.Coeff = Etat_Base_i.Coeff / Somme_Carrs_Normes ;
        Mise_A_Jour ( Etat_Base_i , i ) ;
    }
}
```

Chapitre 04

Exemples d'application

Exemples d'application

4 Exemples d'application

Afin d'illustrer notre implémentation, nous présentons dans cette section quelques exemples d'application. Au début nous commençons par les deux échantillons suivants :

$$|\psi_1\rangle = \frac{1}{2}(|00000\rangle + |01011\rangle + |10101\rangle + |11110\rangle) \quad (28)$$

$$|\psi_2\rangle = \frac{1}{2\sqrt{2}}(|000000\rangle + |001111\rangle + |010101\rangle + |011010\rangle + |100110\rangle + |101001\rangle + |110011\rangle + |111100\rangle) \quad (29)$$

Ces états ont été utilisés comme canaux quantiques de deux protocoles de téléportation [23,33]. Après le test de toutes les combinaisons possibles, les résultats obtenus affirme qu'il s'agit d'états intriqués.

Passons maintenant à l'état suivant :

$$\begin{aligned} |\psi_3\rangle = & -0,136 |00000\rangle - 0,051 |00001\rangle + 0,383 |00010\rangle \\ & + 0,143 |00011\rangle + 0,091 |00100\rangle - 0,257 |00110\rangle \\ & + 0,238 |11000\rangle + 0,089 |11001\rangle + 0,638 |11010\rangle \\ & + 0,238 |11011\rangle - 0,160 |11100\rangle - 0,428ket11110 \end{aligned} \quad (30)$$

Dans ce cas, la machine a détectée la séparabilité de cet état. Les résultats d'exécution affichés sont les suivants :

- Affectation_Qubits : "00101". Cela veut dire que le troisième et le cinquième Qubits appartiennent au premier état alors que les autres sont ceux du deuxième état.
- Pour placer les Qubits du premier état au debut de $|\psi\rangle$, les opérations Swap nécessaires sont : Swap(1,3) et Swap (2,5). L'application de ces deux portes donne l'état :

$$\begin{aligned} |\psi_3\rangle = & -0,136 |00000\rangle + 0,383 |00010\rangle + 0,238 |00101\rangle \\ & + 0,638 |00111\rangle - 0,051 |01000\rangle + 0,143 |01010\rangle \\ & + 0,089 |01101\rangle + 0,238 |01111\rangle + 0,091 |10000\rangle \\ & - 0,257 |10010\rangle - 0,160 |10101\rangle - 0,428 |10111\rangle \end{aligned} \quad (31)$$

Table 4 Mat_Coeffs

-0,136	0	0,383	0	0	0,238	0	0,638
- 0,051	0	0,143	0	0	0,089	0	0,238
0,091	0	- 0,257	0	0	- 0,160	0	- 0,428
0	0	0	0	0	0	0	0

- La matrice des Coefficients Mat_Coeffs générée est la suivante :

- A partir de cette matrice, les deux états potentiels générés sont :

$$|\phi\rangle = 0,638 |00\rangle + 0,238 |01\rangle - 0,428 |10\rangle \quad (32)$$

$$|\rho\rangle = 0,091 |000\rangle - 0,257 |010\rangle - 0,160 |101\rangle - 0,428 |111\rangle \quad (33)$$

- Enfin, la normalisation de ces deux états donne comme résultat :

$$|\phi\rangle = 0,793 |00\rangle + 0,296 |01\rangle - 0,532 |10\rangle \quad (34)$$

$$|\rho\rangle = 0,171 |000\rangle - 0,483 |010\rangle - 0,301 |101\rangle - 0,804 |111\rangle \quad (35)$$

Conclusion Générale

Conclusion générale

L'intrication est un concept fondamental des protocoles quantiques. Il joue un rôle clé dans plusieurs applications, notamment la téléportation quantique où le canal quantique est le support d'interaction entre les différentes parties. Dans ce mémoire, nous nous sommes intéressés par un algorithme permettant de détecter cette propriété de manière informatique. Il s'agit d'un traitement itératif qui procède par un balayage des tous les cas possibles. A chaque itération les opérations réalisées sont :

- Fixer le partage des Qubits de l'état initial entre deux états partiels candidats en utilisant une chaîne de caractères binaire.
- Réorganisez les Qubits de l'état global de sorte que les Qubits assignés au premier état partiel soient placés au début.
- Appliquer un test de séparabilité.

La vérification de la séparabilité est basée sur une matrice particulière construite à partir des coefficients de l'état global. La procédure de test est la suivante :

- Choisir un coefficient non nul de cette matrice.
- Construire premier un état quantique partiel avec les coefficients de la colonne du coefficient choisit.
- Construire un deuxième état partiel avec les coefficients de la ligne du coefficient choisi.
- Vérifier si l'état global est un produit tensoriel de ces deux états partiels.
 - Si c'est le cas, l'état global est séparable.
 - Sinon, l'état global est intriqué.

Ce projet de fin d'étude fut très intéressant dans notre formation. En plus de l'expérience de développement acquise, il nous a permis une initiation dans un domaine très ambitieux de l'informatique.

Références bibliographiques

References

1. M.A. Nielsen and I.L. Chuang, 'Quantum Computation and Quantum Information', Cambridge University Press, (2010).
2. D. Deutsch, 'Quantum Theory, the Church-Turing Thesis, and the Universal Quantum Computer', Proc. Royal Society of London, Series A, Vol. 400, pp. 97, (1985).
3. C.H. Bennett, G.Brassard, R. Jozsa, A. Peres, W.K. Wootters, 'Teleporting an Unknown Quantum State via Dual Classical and EPR Channels', Phys. Rev. Lett. 70, pp.1895–1899, (1993).
4. S. Hassanpour, M. Houshmand, 'Bidirectional Teleportation of a Pure EPR State by using GHZ States', Quantum Inf. Process. 15(2), pp. 905–912, (2016).
5. Y.R. Sun, N. Xiang, Z. Dou, G. Xu, X.B. Chen, Y.X. Yang, 'A Universal Protocol for Controlled Bidirectional Quantum State Transmission', Quantum Inf. Process. 18(281), (2019).
6. S. Sun, H. Zhang, 'Quantum Double-Direction Cyclic Controlled Communication via a Thirteen-Qubit Entangled State', Quantum Inf. Process, 19(120), (2020).
7. S. Aliloute, A. El Allati, I. El Aouadi, 'Bidirectional Teleportation using Coherent States', Quantum Inf. Process. 20(29), (2021).
8. D. Eleni, L. Hoi-Kwong, Q. Bing and Y. Zhiliang, 'Practical Challenges in Quantum Key Distribution', npj Quantum Information, 2, 16025, (2016).
9. L. Xu, Y. Xin, X. Rong, W. Heqing, L. Hao, W. Zhen, Y. Lixing, F. Xue, L. Fang, C. Kaiyu, H. Yidong and Z. Wei Zhang, 'an Entanglement-Based Quantum Network Based on Symmetric Dispersive Optics Quantum Key Distribution', APL Photonics, Vol. 5, Issue. 7, (2020).
10. L. Zhihui, H. Duo, L. Chengji and G. Feifei, 'the Phase Matching Quantum Key Distribution Protocol with 3-State Systems', Quantum Inf. Process. 20(11), (2021).
11. B. Adriano, E. Artur, 'Dense Coding Based on Quantum Entanglement', Journal of Modern Optics, 42(6),(1995).
12. G. Moreno, R. Nery, C. Gois, R. Rabelo and R. Chaves, 'Semi-device-independent certification of entanglement in superdense coding', Phys. Rev. A, 103,(2021).
13. P. Agrawal and A. Pati, 'Perfect Teleportation and Superdense Coding With W-States', Physics Review A, 74, Article ID 062320, (2006).
14. Y. Long, C. Zhang and Z. Sun, 'Standard (3, 5)-Threshold Quantum Secret Sharing by Maximally Entangled 6-Qubit States', Scientific Reports, Vol. 11, Article Number. 22649, (2021).
15. C.W. Tsai, C.W. Yang and J. Lin, 'Multiparty mediated quantum secret sharing protocol', Quantum Inf. Process. 21(63), (2022).
16. S. Hill, W.K. Wootters, 'Entanglement of a Pair of Quantum Bits', Phys. Rev. Lett, 78(26), 5022, (1997).
17. V. Coffman, J. Kundu and W. K. Wootters, 'Distributed entanglement', Phys. Rev. A 61, 052306, (2000).
18. R. Raj, S. Banerjee and P. K. Panigrahi, 'Remote State Design for Efficient Quantum Metrology with Separable and Non-Teleporting States', Quantum Reports, Vol.3 (1), (2021).
19. S. Gharibian, 'Strong NP-Hardness of the Quantum Separability Problem', Quantum Information and Computation, Vol. 10(3), (2008).

-
20. Y. Huang, 'Computing quantum discord is NP-complete', *New J. Phy*, Vol. 16 (033027), (2014).
 21. Y. Zhou, Q. Zhao, X. Yuan and X. Ma, 'Detecting Multipartite Entanglement Structure with Minimal Resources', *npj Quantum Information*, Vol. 5, Article number. 83, (2019).
 22. J. Li and L. Chen, 'Detection of Genuine Multipartite Entanglement Based on Uncertainty Relations', *Quantum Information Processing*, Vol. 20(6), (2021).
 23. K. Khalfaoui, E.H. Kerkouche, T. Boudjedaa and A. Chaoui, 'Optimized Search for Complex Protocols based on Entanglement Detection', *Quantum Information Processing*, Vol. 21(226), (2022).
 24. V.S. Bhaskara and P.K. Panigrahi, 'Generalized Concurrence Measure for Faithful Quantification of Multipartite Pure State Entanglement Using Lagrange's Identity and Wedge Product', *Quantum Information Processing*, Vol. 16, Article number: 118, (2017).
 25. S. Banerjee and P. K. Panigrahi, 'Quantifying Parallelism of Vectors is the Quantification of Distributed n Party Entanglement', *Journal of Physics A Mathematical and Theoretical*, 53(9), (2020).
 26. C.H. Bennett, G.Brassard, R. Jozsa, A. Peres, W.K. Wootters, 'Teleporting an Unkown Quantum State via Dual Classical and EPR Channels', *Phys. Rev. Lett.* 70, pp.1895–1899, (1993).
 27. T. Gao, F.L. Yan, Z.X. Wang, 'Controlled Quantum Teleportation and Secure Direct Communication', *Chin. Phys.* 14(5), pp. 893–897, (2005).
 28. S. Aliloute, A. El Allati, I. El Aouadi, 'Bidirectional Teleportation using Coherent States', *Quantum Inf. Process.* 20(29), (2021).
 29. S. Sun, H. Zhang, 'Quantum Double-Direction Cyclic Controlled Communication via a Thirteen-Qubit Entangled State', *Quantum Inf. Process*, 19(120), (2020).
 30. P. Shor, 'Algorithms for Quantum Computation: Discrete Logarithms and Factoring', *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, (1994).
 31. R. Laflamme, C. Miquel, J.P. Paz, and W.H. Zurek, 'Perfect Quantum Error Correction Code', *Physical Review Letters*, Vol. 77, pp. 198-201, (1996).
 32. K. Khalfaoui, T. Boudjedaa and E.H. Kerkouche, 'Automatic Design of Quantum Circuits : Generation of Quantum Teleportation Protocols', *Quantum Information Processing*, Vol. 20, Article number: 283, (2021).