Le calcul quantique pour la sûreté de fonctionnement : une perspective

Mohamed Hibti, Younès Bennani, Zaiou Ahmed, Basarab Matei

Institut pour la Maîtrise des Risques
Sûreté de Fonctionnement - Management - Cindyniques

Le calcul quantique pour la sûreté de fonctionnement : une perspective

1st Mohamed Hibti
PERICLES
EDF R&D
Palaiseau France
mohamed.hibti@edf.fr

2nd Younès Bennani
LIPN
Université Paris Sorbonne
Villetaneuse
younes.bennani@lipn.univ-paris13.fr

3rd Zaiou Ahmed
LIPN/PERICLES
EDF R&D–Université Paris Sorbonne
Palaiseau/Villataneuse
ahmed.zaou@edf.fr/ahmed.zaiou@lipn.univ-paris13.fr

4th Basarab Matei
LIPN
Université Paris Sorbonne
Villataneuse
matei@lipn.univ-paris13.fr

***Résumé*—**Les Etudes Probabilistes de Sûreté semblent faire partie des problèmes à forte combinatoire pouvant être un candidat intéressant pour le calcul quantique. Dans ce papier, nous donnons une perspective de ce que pourraient devenir les algorithmes de calculs dans le domaine des EPS. Différentes classes d'algorithmes quantiques pourraient être envisagées dans ce sens et contribuer non seulement changer notre façon de calculer les métriques de sûreté de fonctionnement dans de grands modèles, mais aussi impacter les applications industrielles supportées par ces méthodes.

***Mots clés*—**Calcul Quantique, Fiabilité, Etudes probabilistes de sûreté

Probabilistic Safety Studies seem to be among the combinatorial problems that can be a good candidate for quantum computing. In this paper, we give a perspective of what could become of the algorithms in the field of PSA. Different classes of quantum algorithms could be envisaged in this direction and may contribute not only to change the way we calculate the main safety metrics using large models but also have an impact on the industrial applications supported by these methods. ***Abstract*—**

***Keywords*—**Quantum Computing, Reliability, Probabilistic Safety Assessment

I. Introduction

L'évaluation quantitative des scénarios ou le calcul de métriques de risque pour les systèmes complexes se heurte souvent à des problèmes d'explosion combinatoire. Pour les modèles statiques, trouver l'ensemble des impliquants premiers est un problème NP-dur. Pour les modèles dynamiques, la question se réduit à un problème d'atteignabilité (cf. [26]) qui est PSPACE-Complet.

Dans le cas classique qui se base sur la réduction Booléenne, on a à résoudre un problème de génération de coupes minimales (les combinaisons de défaillances) ce qui revient à réduire une formule booléenne en forme normale disjonctive (cf. [19]). Dans le second cas, il s'agit d'explorer des séquences accidentelles (avec des comportements dynamiques liés aux phénomènes physiques et aux aspects cinétiques liés aux séquences de conduite, à la récupération de systèmes, et à l'ordre de sollicitation des systèmes de sauvegarde) qui s'apparentent à des parcours de graphes de taille très importante pouvant comporter des circuits et rendant l'évaluation probabiliste des évènements indésirables très compliquée. Les modèles de référence dans l'industrie nucléaire par exemple sont donc des modèles booléens. Il se trouve que l'évolution de leur temps de calcul suit une loi exponentielle. C'est-à-dire que le temps de calcul augmente en fonction de la taille de l'instance (l'arbre maître/master fault tree) de façon exponentielle et dès qu'un seuil est atteint le saut est trop important. En d'autres termes, au cours du développement de modèles, il arrive qu'une petite modification

puisse induire un temps de calcul extravagant (par exemple faire passer le calcul d'une conséquence de quelques dizaines de minutes à quelques jours). Cette situation n'est plus compatible avec un usage industriel des EPS au titre des applications (par exemple dans le cadre des demandes de dérogation). Le calcul quantique est ainsi naturellement un espoir de voir se développer des méthodes tirant profit de la superposition et du parallélisme quantique pour parcourir très rapidement les arbres ou les graphes sous-jacents dans des temps raisonnables.

Dans le cadre de la réduction Booléenne, Ambainis et al. [1] [2] ont démontré qu'évaluer une formule composée de connecteurs `AND` et `OR` de taille N ne prendrait qu'un temps en $N^{1/2} + o(1)$. Donc évaluer une formule booléenne quelconque ne prendrait pas beaucoup plus.

Stacey Jeffery et Shelby Kimmel [17] ont obtenu des résultats similaires en passant par un problème de s-t connectivité dans certains graphes. L'idée ici est de réduire l'évaluation d'une formule Booléenne à un problème de s-t connectivité (qui consiste à décider pour un graphe dirigé si une destination t peut être atteinte à partir d'une source s) pour lequel une borne inférieure pour un algorithme quantique existe en $O(\sqrt{N})$. Belovs et al. [6] ont aussi retrouvé un résultat semblable en utilisant le concept de marche aléatoire, ainsi qu'un modèle de requêtes à un oracle (le terme oracle désigne une sorte de portion d'algorithme sous la forme d'une boite noire). Ces résultats sont des résultats théoriques sans implémentation pratique.

Dans [30], Zaiou et al. en passant par une réduction de coupes minimales en un problème de s-t connectivité ont proposé un algorithme en $O(n)$ avec une implémentation dans un hardware IBM (qiskit [12]).

Dans le cadre des EPS dynamiques, le problème peut être formulé comme un problème d'atteignabilité d'éléments marqués dans un graphe. Plusieurs travaux ont traité ce problème à l'aide de marches quantiques. Krovi et al., ont utilisé les marches aléatoires pour détecter et trouver des éléments marqués dans un graphe. Dans le cas d'éléments multiples, ils ont donné le nombre de pas sur la base d'un temps étendu d'atteinte[1] (extended hitting time). Ambainis et al. [3] ont proposé un algorithme qui permet de détecter la présence d'un élément marqué quadratiquement plus vite que lors d'une marche aléatoire. Belovs a démontré qu'une marche quantique peut détecter la présence d'un élément marqué en $\sqrt{O(WR)}$ étapes pour toute distribution de probabilité sur les nœuds (W étant le poids total du graphe ou encore la somme des poids de ses arêtes et R sa résistance effective[2]).

D'autres algorithmes utilisent la même notion de résistance dans le cadre des problèmes de satisfaction de contraintes pour accélérer l'algorithme [9] qui est un des plus efficaces pour la résolution du problème de satisfiabilité des formules Booléennes (SAT). Des approches hybrides ont été introduites pour l'usage du calcul quantique sur des machines avec un nombre réduit de qubits. On y trouve l'approche Divide and Quantum qui vise à diviser les instances de SAT en cherchant des affectations satisfaisant la formule booléenne sur une boule de rayon r (Promise-Ball-SAT) pour construire ensuite une assignation satisfaisante pour la formule initiale.

II. Calcul quantique

La puissance des ordinateurs quantiques réside dans la densité des données qu'ils sont capables de manipuler. Quand un ordinateur classique peut lire, stocker et manipuler des bits "0" ou "1", un ordinateur quantique peut manipuler des qubits qui sont des superpositions des états "0" et "1" et qui lorsqu'ils sont observés prennent une de ces valeurs.

En mécanique quantique, le postulat numéro 1 stipule que l'état d'un système fermé est complètement décrit par une fonction d'onde associée à un vecteur d'état (ket) $|psi\rangle$, décrit par un vecteur colonne unitaire. Ce dernier appartient à un espace vectoriel complexe muni d'un produit scalaire, connu sous le nom d'espace des états.

Dans cet exposé, on utilisera les notation de Dirac en bra-kets pour désigner des vecteurs dans un espace de Hilbert. Pour un vecteur v dans l'espace de Hilbert, on utilisera la représentation en colonne vectorielle de v "ket-v".

$$v = \begin{vmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{vmatrix} = |v\rangle$$

on notera le vecteur dual de v

$$\langle v| = \bar{v^T} = \langle v|$$

[1] Le temps d'atteinte étendu de P par rapport à M est $HT^+(P, M) := \lim_{s \to 0} HT(s)$. $HT(s)$ étant le temps d'atteinte interpolé est défini comme étant

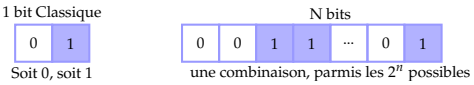
$$HT(s) := \sum_{k=1}^{n-1} \frac{|\langle v_k(s)|U\rangle|^2}{1 - \lambda_k(s)}$$

[2] Voir section VII-E.

où $\bar{v}$ est le conjugué complexe de v
Le produit scalaire entre deux vecteurs de l'espace de Hilbert est défini comme étant:

$$\begin{aligned}\langle u|v\rangle &= \bar{u^T}.v \\ &= [\bar{u_1}\bar{u_2}\dots\bar{u_n}]\begin{vmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{vmatrix} \\ &= \bar{u_1}.v1 + \bar{u_2}v_2 + \dots + \bar{u_n}.v_n\end{aligned}$$

A. *Le qubit*



Dans le monde classique, l'information élémentaire est le bit "0" ou "1" (bit représente un voltage) le qubit est l'équivalent en mécanique quantique.
Le qubit a deux états $|0\rangle$ et $|1\rangle$. Un état quantique d'un qubit est un vecteur dans un espace véctoriel complexe à deux dimensions (voir représentations dan,s la spère de Bloch cf. figure 1). Un état quantique général peut être écrit sous la forme $\alpha|0\rangle+\beta|1\rangle$ qui est une supperposition quantique (combinaison linéaire) des états $|0\rangle$ et $|1\rangle$ avec des amplitudes α pour l'état $|0\rangle$ et β pour $|1\rangle$.

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{bmatrix} \beta \\ \alpha \end{bmatrix} \begin{matrix} \rightarrow |0\rangle\, amplitude \\ \rightarrow |1\rangle\, amplitude \end{matrix}$$

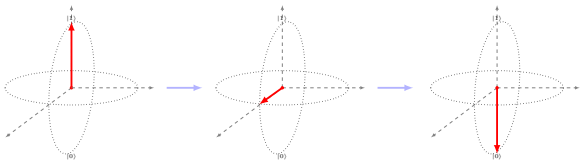


Figure 1. Sphère de Bloch

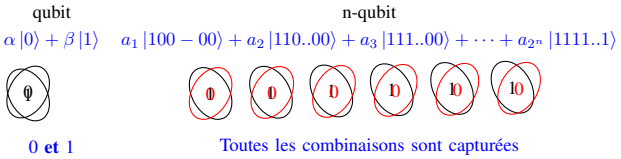
Dans un monde classique 2 bits peuvent encapsuler 2 informations etant donné les combinaisons suivantes :

00
01
10
11

Connaître le premier digit et le second (ce qui fait deux informations) permet de déterminer laquelle des quatre possibilités est réalisée. En substance, 2 bits peuvent nous donner 4 états de façons successives alors que 2 qubits nous les mêmes 4 états "instantanément".
Avec deux spins systems (qubits) on peut encapsuler 4 informations. de manière générale, l'information encapsulée par N-qubits est de l'ordre de 2^N. En effets deux qubits peuvent donner comme résultat à la superposition

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

De manière générale on a pour n-qubits la formule suivante:



Les états quantiques sont manipulés à l'aide de portes quantiques qui servent à définir au sens classique du termes les fonctions que l'on souhaite implémenter. Une séquence de portes logiques quantiques (appliquées à un qubit ou à un ensemble de qubits) constitue un circuit quantique (cf. Figure 2).

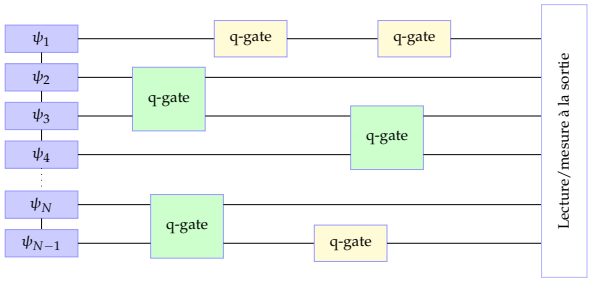


Figure 2. Circuit quantique

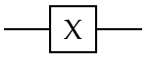
B. *Portes quantiques*

Dans le cas classique, dans un processeur on a des portes universelles (`NOR` ou `NAND`) qui servent à executer n'importe quelle opération logique sur des digits. De la même manière dans le cas quantique, il y a un certain nombre de portes dites universelles et qui permettent de mimer toutes les opérations des autres portes quantiques plus élaborées. Il s'agit des portes `H`, `X`, `T` et `CNOT`.
Lorsqu'on se refère à la sphère de Bloch, les portes quantiques peuvent être représentées par des rotations autour d'un vecteur de la sphère (cf. [21]). D'autre part, ces matrices doivent être unitaires ($U^tU = I$ c'est à dire $U\overline{U^T} = I$ donc préservent les longueurs $\|U|v\rangle\| = \||v\rangle\|$).

1) Porte NON: C'est une généralisation de la porte classique NON ($\neg$) en particulier, $|0\rangle \rightarrow |1\rangle$ et $|1\rangle \rightarrow |0\rangle$ et donc

$$\alpha|0\rangle + \beta|1\rangle \rightarrow \alpha|1\rangle + \beta|0\rangle$$

La représentation dans un circuit quantique est comme suit :

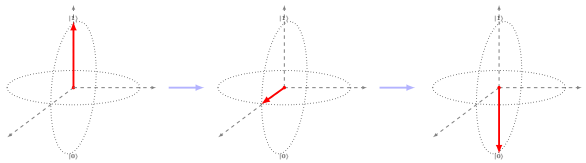


La representation sous forme de matrice est la suivante :

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

$$X|0\rangle = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} = |1\rangle$$

2) Porte de Hadamard: La porte de Hadamard agit sur 1 seul qubit, elle transforme l'état $|0\rangle$ en $\frac{|0\rangle + |1\rangle}{\sqrt{2}}$ et $|1\rangle$ en $\frac{|0\rangle - |1\rangle}{\sqrt{2}}$, ce qui implique qu'une mesure sera equiprobable pour déboucher sur 1 ou 0 (c'est pour cette raison qu'on l'appelle des fois le *fair coin flip*). Cette porte représente une rotation de π sur l'axe des X $(\hat{x} + \hat{z})/\sqrt{2}$.



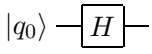
C'est de façon équivalente la combinaison de deux rotations, π sur l'axe des X suivie de $\pi/2$ sur l'axe des Y. Elle est représentée par la matrice de Hadamard suivante

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$|0\rangle \rightarrow \frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

$$|1\rangle \rightarrow \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

La porte Hadamard dans un circuit est représentée par



La porte de Hadamard est une des portes les plus importantes en calcul quantique. Elle semble assez banale mais elle encapsule une propriété très intéressante et très importante en calcul quantique. En effet, en l'appliquant de façon parallèle à n qubits pris initialement dans l'état $|0\rangle$, l'état produit est égal à une superposition de tous les entiers de 0 à 2^{n-1}. Donc, on peut préparer une superposition contenant une nombre exponentiel de termes en utilisant un nompbre polynomial d'opérations. Cette astuce est utilisée dans plusieurs algorithmes quantiques pour charger efficacement en mémoire un registre quantique.

3) Porte C-NOT: La porte CNOT (Controlled NOT) opère sur un registre de 2 qubits, une destination (target) et un control qubit elle flippe la destination si et seulement si le control est $|1\rangle$. C'est une porte qui permet "d'implémenter" les conditions `"Si ... alors..."`. C'est une porte qui agit **à condition** que le contrôle soit $|1\rangle$.

Si un qubit peut s'ecrire: $\alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle$, la porte C-NOT agit en le transformant en $\alpha|00\rangle + \beta|01\rangle + \gamma|11\rangle + \delta|10\rangle$:

$$\begin{array}{lcl} |00\rangle & \rightarrow & |00\rangle \\ |01\rangle & \rightarrow & |01\rangle \\ |10\rangle & \rightarrow & |11\rangle \\ |11\rangle & \rightarrow & |10\rangle \end{array}$$

ou encore

Avant		après	
Control	destination	Control	destination
$\|0\rangle$	$\|0\rangle$	$\|0\rangle$	$\|0\rangle$
$\|0\rangle$	$\|1\rangle$	$\|0\rangle$	$\|1\rangle$
$\|1\rangle$	$\|0\rangle$	$\|1\rangle$	$\|1\rangle$
$\|1\rangle$	$\|1\rangle$	$\|1\rangle$	$\|0\rangle$

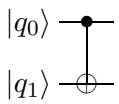
On écrit souvent la porte C-NOT de la façon suivante:

$$|xy\rangle \rightarrow |xy \oplus x\rangle$$

ou sous forme matricielle

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Dans un circuit on présente la porte c-not sous la forme suivante:



4) Calcul quantique: Il est bien connu que dans un monde classique les portes ET et NON peuvent calculer n'importe quelle fonction. Il s'ensuit que les machines de calcul/ordinateurs peuvent être construits en utilisant seulement ces deux portes. On dit que ET et NON sont universelles pour le calcul classique. De la même façon, pour les ordinateurs quantiques les porte CNOT et les portes à 1 seul qubit sont universelles pour construire n'importe quelle opération unitaire sur n-qubits.

C. Marche Quantique

Une marche quantique est l'analogue de la marche aléatoire dans le cas classique. Lorsque l'état courant du marcheur classique suit une distribution de probabilités sur les positions, l'état du marcheur quantique est une superposition des positions. d Une marche quantique à temps continu sur un graphe $G = (V, E)$, où V est l'ensemble des noeuds et E est l'ensemble des arcs est défini comme de la façon suivante: Soit A la matrice d'adjacence $|V| \times |V|$ G avec

$$A_{u,v} = \begin{cases} 1 & \text{if } \{u,v\} \in E \\ 0 & \text{otherwise} \end{cases}$$

et D la matrice des degrés de $G|V|\times|V|$ (pour laquelle v est $degre(v)$), soit $L = D - A$, la matrice laplacienne correspondante.
La marche aléatoire à temps continu sur G est définie par la matrice unitaire.

$$U(t) = e^{-itL}\, U(t) = e^{-itL}$$

où i est le nombre imaginaire de $\mathbb{C}$ et $t \in \mathbb{R}$.
La probabilité d'une marche commençant au noeud u et stoppant à v au temps t est donnée par $|\langle v|U(t)|u\rangle|^2$. Par conséquent, si l'on commence à l'état $|\psi_0\rangle$ et marchant quantiquement pendant un temps t on aboutit à l'état $|\psi_t\rangle = U(t)|\psi_0\rangle$ la mesure va alors localiser la marche à v avec la probabilité $|\langle v|U(t)|\psi_0\rangle|^2$

III. Calcul quantique pour les EPS

Il y a plusieurs classes d'algorithmes quantiques qui diffèrent dans la nature des outils qu'elles convoquent ainsi que des problèmes auxquels elles s'attaquent.
La première est celle qui utilise la version quantique de la transformée de Fourier (L'algorithme est un représentant ainsi que celui de Shor pour la factorisation des nombres entiers qui est au cœur des codes de chiffrement RSA[3]). Une deuxième classe concerne les algorithmes de recherche par exemple dans des grandes bases de données non structurées. L'algorithme de Grover en est un représentant. Et les algorithmes dits de simulation qui résolvent des problèmes en les ramenant à de la simulation de systèmes quantiques.
Plusieurs approches peuvent être considérées pour la résolution du problème d'évaluation des métriques de risque dans les EPS.
Dans ce papier, nous présentons une revue de certaines de ces méthodes. Certaines utilisent les outils de recherche avec un oracle en suivant l'algorithme de Grover ou l'algorithme HHL[4] de résolution de systèmes linéaires d'équations. D'autres se basent sur les marches quantiques, par exemple, pour la recherche des éléments marqués dans un graphe et donc permettent de simuler des processus de Markov. Ces méthodes peuvent servir à la recherche de séquences d'atteinte d'un état de panne à partir d'un état de marche d'un système. Ces algorithmes nécessitent néanmoins un nombre de qubits très important à l'échelle des tailles des problèmes à résoudre. Dans l'état actuel du hardware quantique, il y a une nécessité de mise à l'échelle. Celle-ci peut passer par l'usage des algorithmes hybrides qui permettent de combiner des algorithmes classiques avec la résolution de problèmes combinatoires localement en utilisant des algorithmes quantiques. Ces derniers sont un axe à privilégier dans le sens où les machines quantiques pouvant servir à la résolution sont déjà disponibles ou pourront l'être dans un futur très proche.

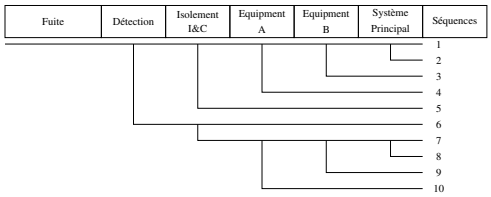
IV. EPS statique

Dans les EPS classiques où l'on utilise des arbres d'évènements pour modéliser les scénarios et des arbres de défaillances pour représenter les défaillances des missions systèmes (FH, I&C ou systèmes classiques) on cherche à retrouver toutes les combinaisons de défaillances pouvant conduire à un évènement redouté (ce qu'on appelle liste des impliquants premiers ou, par abus de langage, liste coupes minimales).
On peut voir qu'un calcul EPS quelconque est d'abord une opération sur une formule booléenne qu'il faudra "aplatir" ou fusionner (on parle souvent

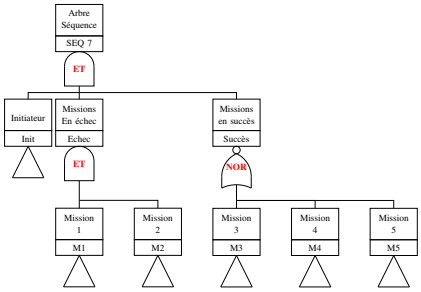
[3]Les codes RSA (Rivest, Shamir et Adelman) une des prelières clés d'encryption, qui semble menacés par les algorithmes quantique. Des travaux sont entrepris pour développer une encryption post-quantique basée sur la factorisation aussi mais à base de matrices, ce qui rend le problème autrement plus difficile.

[4]Au nom de Aram Harrow, Avinatan Hassidim et Seth Lloyd.

de fusion booléenne pour désigner les algorithmes classiques utilisés pour les calculs EPS). Si l'on considère, par exemple, l'arbre suivant :



Une séquence, par exemple 7, peut s'écrire sous la forme de la représentation de l'arbre suivant :



Pour une Conséquence

$$Pr(Ci) = Pr(\bigcup_{Seq \to Ci} Seq)$$

Un calcul EPS consiste à faire de la fusion booléenne sur les arbres obtenus qu'on appelle arbres maîtres (ou *master fault tree*). Ceci revient à "aplatir" des arbres en une forme suivante :

$$\bigvee_{j=1}^{m} \bigwedge_{k=1}^{sj} E_{jk} = (E_{11} \wedge \cdots \wedge E_{1s_1}) \vee \cdots \vee (E_{m1} \wedge \cdots \wedge E_{ms_m})$$

où E_{jk} est soit un *événement de base* (EB), soit une négation d'événement de base. On retrouve les *impliquants* dans les expressions $(E_{f1} \wedge \cdots \wedge E_{fs_f})$.

L'objectif est de construire un **circuit** qui permet à partir d'une formule booléenne géante d'en déduire les impliquants premiers.
En général, dans un monde classique un calcul consiste à partir d'une donnée X vers une sortie $f(X)$. Dans notre cas, du calcul statique EPS, on part d'une formule Booléenne F et on aboutit à une formule Booléenne CND_F (pour forme normale disjonctive de F). Pour obtenir cette dernière, on passe les données de F à travers un **circuit logique** pour obtenir la forme CND_F en sortie. Mais il se trouve que dans un monde classique ce calcul est exponentiel et se comporte de façon assez imprévisible en fonction des instances. En zone de seuil[5], les modèles deviennent très sensibles et l'utilisation des EPS pour décider est donc pénalisée.

Il se trouve que pour chaque circuit digital classique, il existe un équivalent quantique. De plus, on arrive à trouver des **raccourcis très intéressants**.

En calcul quantique, on commence avec une donnée X avec le qubit $|X, 0\rangle$ et on essaye d'arriver à un résultat $|x, f(X)\rangle$.
L'évaluation du calcul Booléen a eu une attention particulière dans le calcul quantique grâce à son statut comme représentant par excellence de la classe des problèmes NP-durs. Dans [1], Ambainis et al. ont démontré qu'évaluer une formule NAND de taille N ne prendrait qu'un temps en $N^{\frac{1}{2}+o(1)}$ c'est-à-dire en gros $\sqrt{N}$.
Donc évaluer une formule booléenne quelconque ne prendrait pas beaucoup plus. En effet, toute fonction Booléenne peut être reformulée en utilisant une combinaison de portes NAND uniquement. Cette propriété est connue sous le non de complétude fonctionnelle.

Jeffery et Kimmel ont obtenu des résultats similaires en passant par un problème de s-t connectivité dans certains graphes (cf. [17]). L'idée ici est de réduire l'évaluation d'une formule Booléenne à un problème de s-t connectivité (qui consiste à décider pour un graphe dirigé si une destination t peut être atteinte à partir d'une source s) pour lequel un algorithme quantique existe en $O(\sqrt{(N)})$. Il est à noter que Zaiou et al. (cf. [30]) ont présenté un algorithme de recherche de coupes minimales d'arbres de défaillances en passant par un problème de s-t connectivité plus naturelle à partir de topologie des systèmes [14]. Andrew M. Childs et al. ont aussi retrouvé un résultat similaire (cf. [8]) en utilisant le concept de marche aléatoire. Leur approche se base sur un modèle de requêtes à un oracle (le terme oracle désigne une sorte de portion d'algorithme sous la forme d'une boite noire).

A titre d'exemple, nous présentons la description de l'algorithme d'Ambainis dans [2].
On suppose donc un arbre construit uniquement avec portes des NAND et dont les feuilles sont des variables x_i. On considère un modèle de requêtes à oracle. Les entrées x_1, ..., x_N sont accessibles via des requêtes en O à une boite noire.

[5] Seuil à partir duquel le temps de calcul fait un saut exponentiel empêchant le calcul de se terminer en temps "raisonnable".

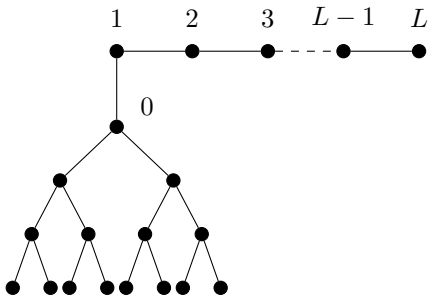


Figure 3. l'arbre correspondant à la formule augmenté avec une queue de longueur L

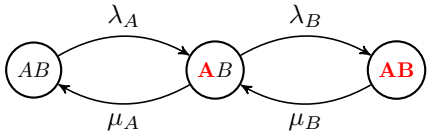
Pour définir O, on représente les états de base comme $|i,z\rangle$ où $i \in 0,1,\ldots,N$. La requête de transformation O_x (où $x=(x_1,\ldots,x_N)$) transforme $|0,z\rangle$ en $|0,z\rangle$ et pour $i \neq 0$ en $(-1)_i^x |i,z\rangle$.
L'algorithme consiste en une succession de requêtes O_x et de transformations arbitraires **non dépendantes de** x_i, et son but est de calculer la valeur de l'arbre en utilisant le moins de requêtes possibles.

Plusieurs équipes avec différentes techniques ont proposé des algorithmes dans ce sens avec des complexités de l'ordre de $O(\sqrt{N})$.
La formule initiale est augmentée d'une queue de longueur L (voir Figure 3).
On considère une marche quantique sur cet arbre. Aux feuilles, les transformations appliquées dépendent de la nature des feuilles $x_i=0$ ou $x_i=1$.

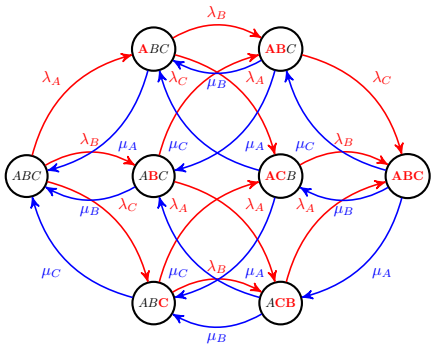
L'état initial est choisi de manière appropriée comme étant l'état quantique $|\psi_{init}\rangle = \sum_i \alpha_i |i\rangle$ consistant en les états $|i\rangle$ de la queue. Il se passe quelque chose de surprenant, quand la formule est évaluée à 0, l'état $|\psi_{init}\rangle$ reste presque inchangé. Quand la formule est évaluée à 1, après $O(N^{\frac{1}{2}+o(1)})$ pas, l'état est complètement différent de $|\psi_{init}\rangle$. Ce qui veut dire que le comportement de la marche ne dépend pas des valeurs d'un x_i particulier, mais de l'évaluation de la formule.

V. EPS DYNAMIQUE

L'un des freins à une EPS dynamique est la difficulté d'envisager une évolution de l'état d'un système dans le temps et ses transitions avec des défaillances stochastiques. Une des principales difficultés est la modélisation des différentes transitions à cause de l'explosion combinatoire et de l'exploration des différentes séquences de façon exhaustive. Les multiples bifurcations des séquences conduisent à des temps d'explorations très longs pour une exploitation industrielle. C'est la raison pour laquelle on cède des approximations de sorte que l'algorithme d'exploration puisse terminer en un temps raisonnable.
Si l'on se place dans le cadre quantique ; les transitions parallèles doivent pouvoir se faire en même temps et les graphes sont parcourus rapidement. D'autre part, si l'on peut bénéficier du raccourci quantique, on peut envisager des calculs dynamiques très poussés. En effet, il est possible d'associer un Hamiltonien H au système ensuite faire des transformations sur H pour simuler le comportement du système ce qui permet d'évaluer quantiquement les différentes grandeurs d'intérêt.
Par exemple, dans le cas où l'on veuille modéliser les transitions d'un système de deux composants redondants A et B. On aboutit au graphe suivant.



avec 3 composants



On voit bien l'explosion combinatoire des transitions qui s'ensuivent. Utiliser des algorithmes classiques ne permet pas de faire le calcul dans des temps raisonnables et réduit l'intérêt de ces méthodes dans le cadre du calcul de fiabilité, ou de l'exploration des séquences dynamiques.
Il existe d'ors et déjà des équivalents de graphes de Markov quantiques. Mais s'il faut aller dans un monde quantique autant chercher des solutions plus innovantes pour profiter de la puissance de ce paradigme.

Plusieurs approches peuvent être utilisées dans ces sens (par exemple les marches quantiques) pour accélérer de manière significative l'exploration des graphes de Markov.

Une solution de l'équation d'état consiste à trouver les valeurs propres $e^{i\lambda_k}$ de la matrice de transition. C'est l'objectif des approches de l'estimation quantique de phase. Il est donc possible de trouver des algorithmes qui permettent de déterminer l'évolution du système (et donc des séquences de défaillances) sans reposer sur un arsenal d'hypothèses assez fortes[6].

L'algorithme *HHL* pour la résolution des systèmes d'équations est une piste prometteuse. L'idée est d'exploiter l'évolution Hamiltonienne et l'estimation quantique de phase.

HHL permet de résoudre des équations de type $A\vec{x} = \vec{b}$ (ou plutôt dans le cas quantique $A\,|x\rangle = |b\rangle$).

Dans le cas classique la résolution de ce genre d'équation requiert $O(N^3)$, mais dans le cas où la matrice A est creuse, c'est-à-dire pas plus de $s \ll n$ entrée par ligne, on a un calcul en $O(Nlog(N))$. Dans le cas quantique on a besoin de $n = log_2(N)$ qubits pour la construction du vecteur $|b\rangle$ et si la matrice est creuse l'algorithme est en $O(log_2(N))$ étapes sur un ordinateur quantique.

Si on arrive à diagonaliser A alors on peut écrire :

$$A = U \begin{pmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ 0 & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_N \end{pmatrix} U^\dagger$$

et ensuite

$$A^{-1} = U^\dagger \begin{pmatrix} \lambda_1^{-1} & 0 & \dots & 0 \\ 0 & \lambda_2^{-1} & \dots & 0 \\ 0 & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_N^{-1} \end{pmatrix} U$$

On commence par préparer $|b\rangle$ comme un état

$$|b\rangle = \sum_i = 1^N \beta_i \,|V_i\rangle\,;$$

V_i étant les vecteurs[7] propres associés aux λ_i. On ne sais pas encore quelles sont les valeurs de β_i. L'idée est de pouvoir appliquer A^{-1} à b pour obtenir x.

Ensuite on utilise des techniques de simulation Hamiltonienne pour appliquer e^{iAt} à $|b\rangle$ et utiliser l'estimation de phase de Kitaev (cf. [18]) pour estimer les λ_i ($\sum_i = 1^N \beta_i \,|V_i\rangle\,|\lambda_i\rangle$).

Ensuite préparer un nouveau qubit

$$\sum_i = 1^N \beta_i \,|V_i\rangle\,|\lambda_i\rangle\,(\lambda_i^{-1}\,|0\rangle + \sqrt{1-\lambda_i^{-2}}\,|1\rangle$$

et enfin revenir en arrière en inversant la phase de λ_i

$$\sum_i = 1^N \beta_i \,|V_i\rangle\,\lambda_i^{-1}\,|0\rangle + \sum_{i=1}^{N} \beta_i \,|V_i\rangle\,\sqrt{1-\lambda_i^{-2}}\,|1\rangle$$

La dernière étape consiste en l'amplification de la partie $|0\rangle$ pour arriver à $|x\rangle$.

VI. Algorithmes à base de requêtes à la Grover

On va commencer par un petit exemple[8] pour illustrer le fonctionnement de l'algorithme de Grover. Supposons qu'on a une liste de numéros de téléphone et de noms et que l'on veuille trouver le nom d'un numéro spécifique.

On va voir que l'algorithme de Grover peut apporter une accélération quadratique par rapport à un algorithme classique. En effet, dans le cas classique la complexité est en $O(N)$ si N est la taille de la liste alors dans le cas quantique (ici dans Grover) la complexité est en $O(\sqrt{N})$.

L'algorithme de Grover utilise une technique d'amplification d'amplitude[9].

Dans cet exemple, on va considérer quatre numéros, qui peuvent être représentés en utilisant 2 qubits. On cherche le nom associé à $|10\rangle$. De manière générale quand $N = 2^n$ on a besoin de n bits pour coder l'index des N éléments. Par exemple, $8 = 2^3$ on a besoin de 3 bits seulement pour coder 8 éléments $\{000, 010, 011, 001, 100, 110, 101, 111\}$.

- Dans notre exemple, on va superposer les qubits, c'est la première étape de l'algorithme :

  $$a\,|00\rangle + b\,|01\rangle + c\,|10\rangle + d\,|11\rangle$$

  où a, b, c et d sont les amplitudes et initialement $a = b = c = d$.
- La deuxième étape consiste à appliquer un oracle qui permet d'inverser l'amplitude de l'élément recherché dans la direction opposée.

[6] Il faut noter que dans les calculs de fiabilité, il arrive que l'on suppose un certain nombre de postulats qui peuvent être valides sous certaines conditions et qui servent à faciliter les calculs. Mais dans un cadre de plus en plus étendus à cause de l'extension de la couverture des EPS, la validité des hypothèses en question est des fois remise en cause et les calculs doivent en tenir compte (par exemple, les approximations utilisées en présence d'évènements rares).

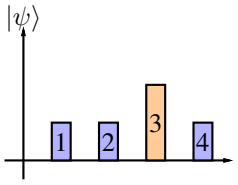
[7] Les vecteurs propres forment une base orthonormée.

[8] Voir computing explained (cf. https://hackernoon.com/quantum-computing-explained-a114999299ca)

[9] En mécanique quantique, on nomme amplitude de probabilité un vecteur composé d'un module et d'une phase, qui peut être représenté par un nombre complexe $e^i\theta$ (deux coordonnées). Le carré du module de cette amplitude est assimilable grosso modo à une probabilité de détection de la particule en un endroit donné.

Donc c qui correspond à notre élément devient négatif et a, b et c restent les mêmes.

- La troisième étape consiste à appliquer une fonction d'amplification qui va amplifier la différence entre chaque amplitude et celles qui sont dans la direction opposée ce qui conduit à une amplification de celle de c qui va rapidement croître au détriment ce celle de a, b ou d.



Maintenant si l'on mesure le qubit du 3ème état, il va retourner la probabilité maximale. Les étapes 2 et 3 peuvent être répétées pour amplifier la probabilité.

1) Tables de vérité: Une des méthodes brutales d'évaluation de la fiabilité d'un système est d'énumérer toutes les combinaisons de défaillances avec l'ensemble des valeurs de vérité associées aux défaillances identifiées.

L'idée est de représenter les vecteurs d'état du système en utilisant les défaillances de ses composants. Chaque vecteur d'état peut être vu comme une chaine binaire qui appliquée à l'arbre de défaillance pourra évaluer le sommet à `VRAI` ou `FAUX`.

Pour des systèmes de taille importante n, il y aura 2^n chaines binaires à évaluer, ce qui est un effort de calcul très important. En utilisant l'algorithme de Grover, on pourrait réduire la complexité du problème. Il faudra donc chercher les chaines marquées qui seraient évaluées à 1.

Comme on l'a vu plus haut (cf. section VI), cela requiert O($\sqrt{\frac{2^n}{M}}$) requêtes pour obtenir une solution sur M possibles, ce qui nous fait $O(\sqrt{2^n})$ requêtes au total, ce qui constitue une accélération quadratique.

Lorsqu'on compare les algorithmes classiques aux algorithmes quantiques, les premiers, avec de l'expérience, ont pu bénéficier d'une bonne exploitation de la structure du problème pour utiliser un certain nombre d'astuces. Les algorithmes quantiques sont plutôt efficaces là où seules des méthodes de force brute paraissent disponibles.

Il est important de tenir compte de cet aspect. En outre, pour des questions de passage à l'échelle, il vaut mieux considérer la structure des problèmes pour envisager des approches hybrides. L'objectif est d'utiliser des algorithmes quantiques seulement là où c'est pertinent (voir section VIII).

Si l'on considère l'exemple des tables de vérité, on peut par exemple réduire le nombre de combinaisons en allant de $N = 2^n$ à $N = 2^{n-k}$ avec k un nombre significatif de sorte que l'algorithme quantique soit appliqué à un problème de taille raisonnable.

Cette approche est utilisée de façon intuitive dans plusieurs algorithmes classiques, dans les outils commerciaux, c'est le processus de modularisation qui sert à réduire considérablement la taille du problème initial pour envisager la réduction Booléenne d'une formule où les littéraux sont des modules ou des négations de modules[10]. Le calcul de ces modules est fait pendant la première phase de l'algorithme, ensuite au cours de celui-ci certains sont tronqués pour n'en garder que ceux au-dessus d'un certain seuil. Une autre approche consiste à utiliser des modules construits en amont du processus d'évaluation et suivant un découpage fonctionnel (voir cf. [15]).

VII. Marches quantiques

Le problème des marches quantiques est étudié de manière assez conséquente dans la littérature sur le calcul quantique. Plusieurs algorithmes l'utilisent d'abord dans le cas classique (marches aléatoires) pour sa capacité à servir à modéliser des phénomènes physiques, en biologie, en finance, en vision par ordinateur (computer vision), mais aussi pour la modélisation des séismes (Cf. [7], [22], [16], [13], [29]). En plus de leur usage dans comme outil algorithmique les marches aléatoires dans le cadre quantique suscitent un intérêt certain d'abord pour la modélisation de phénomènes stochastiques mais aussi pour vérifier ou tester "la nature quantique" des technologies qui vont servir à l'élaboration des ordinateurs quantiques (cf. [28]).

La marche quantique est un cadre qui peut décrire le phénomène de propagation d'une particule (quantique) dans un environnement discret. Ça permet d'expliquer les dynamiques complexes de mouvement de particules dans un système physique, chimique ou biologique[11].

Dans le cas classique —c'est-à-dire dans une marche aléatoire classique— le marcheur décide de façon aléatoire (en jouant pile ou face), quel chemin prendre. Par exemple, dans le graphe suivant, à chaque nœud, il jète sa pièce et suit le chemin de droite

[10]Il faut noter que dans ce cas, quand il s'agit de formules représentant des arbres de séquences ou de conséquences, la négation est gérée partiellement, seulement quand elle se retrouve dans les feuilles. Des traitements spécifiques sont faits à postériori pour supprimer certaines incohérences.

[11]https://physik.uni-paderborn.de/en/silberhorn/research/quantum-networking/quantum-walks/.

ou de gauche en fonction du résultat. Après un certain nombre de pas la probabilité d'atterrir dans une destination finale est donnée par une distribution binomiale avec une selon dispersion proportionnelle à la racine carrée des pas.

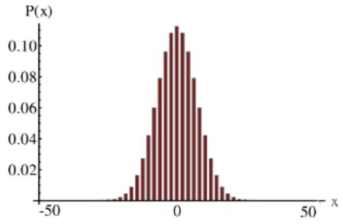
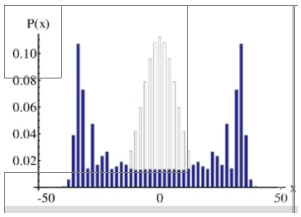


Figure 4. Probabilité d'atterrir dans une destination finale suit une loi binomiale

Dans le cas quantique le "marcheur quantique" (système ici) est décrit par une fonction d'onde. Pendant la marche ou évolution, la fonction se disperse simultanément dans différentes positions laissant le système dans une superposition cohérente de tous les chemins possibles. Ceci résulte des effets d'interférence qui changent de manière drastique la distribution de probabilité finale.
L'évolution quantique est dans ce cas beaucoup plus rapide, l'atteinte des destinations finales est **en moyenne** quadriquement plus rapide que dans le cas classique.



A. Marches quantiques dans des graphes

La question générale qui se pose ici est celle de la convergence des chaines de Markov. Or lorsqu'on considère les marches quantiques, on est dans des opérateurs unitaires réversibles, ce qui empêche d'avoir des propriétés de stationnarité. En gros, les marches quantiques ne convergent vers aucune distribution stationnaire, mais en relaxant certaines contraintes, on peut obtenir une vitesse de dispersion de la marche quantique et comment peut être confinées dans un petit voisinage.
Dans le cas classique, on parle de convergence vers une distribution stationnaire, mais dans le cas quantique, on va plutôt essayer de mesurer le temps de mélange de la chaine de Markov quantique.

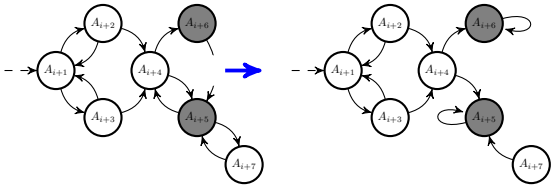


Figure 5. Construction d'une marche analogue

B. Distribution Limite et temps de mélange

Lorsqu'on regarde l'évolution d'une marche quantique en fonction du temps à partir d'un état $|\alpha_0\rangle$. A t la marche quantique est dans un état $|\alpha_0\rangle = U^t |\alpha_0\rangle$. Comme on ne peut raisonner en termes de limite quand $t \longrightarrow \infty$ de $|\alpha_t\rangle$ on va plutôt s' intéresser à la distribution de probabilité sur les nœuds du graphe induite par $|\alpha_t\rangle$.

$$P_t(v|\alpha_0) = \sum_{a \in V} |\, \langle a, v|\alpha_t\rangle \,|^2$$

Cette distribution ne converge pas non plus, mais sa moyenne en temps converge (cf. [20])

$$\bar{P}_t(v|\alpha_0) = \frac{1}{T} \sum_{t=0}^{T-1} P_t(v|\alpha 0)$$

Le temps de mélange de M_ϵ de la chaine de Markov quantique est défini comme étant le nombre de pas nécessaire pour que la distribution moyenne soit $\epsilon -$ proche de la distribution limite partant de l'état initial :

$$M_\epsilon = min\{T|\forall t \geq T, |a, v\rangle : \left\|\pi(.|a, v) - \bar{P}_t(.|a, v)\right\| \leq \epsilon\}.$$

où $P(.|a, v)$ désigne la distribution de probabilité sachant l'état initial $|a, v\rangle$.

C. Recherche d'éléments marqués dans un graphe

Lorsqu'on s'intéresse au problème d'atteinte d'un ensemble de nœuds marqués d'un graphe, ce qui correspond par exemple dans le cas des EPS à l'atteinte de certains états précis (e.g. états de panne). Plusieurs travaux utilisent les marches quantiques pour atteindre ces états dans un temps avec un gain quadratique.
L'approche suivie dans [20] se base sur la construction d'une marche quantique analogue à une chaine de Markov $P(s) = (1-s)P+sP'$ qui interpole entre la marche aléatoire P et la marche absorbante P' dont les transitions sortantes des nœuds marqués (i.e. $n \in M$) par des boucles sur soi (cf. Figure VII-C).
L'idée principale est de considérer un opérateur W(s) qui jouera le rôle de distribution stationnaire dans

le cas quantique. Les vecteurs propres $|\Psi_n(s)\rangle$ (s un paramètre d'interpolation) de $W(s)$ sont utilisés pour les projeter sur les nœuds marqués et non marqués. Si les projections sont larges alors l'algorithme de recherche des éléments marqués réussit avec une grande probabilité dans un temps en $O(\sqrt{HT^+(P,M)})$ où $HT^+(P,M)$ est le temps de mélange étendu.

D. Accélaration quadratique pour la recherche d'éléments marqués

Dans [3], un algorithme quantique levant plusieurs hypothèses limitatives sur la nature des éléments en question (un seul élément marqué, ou seulement dans le cas de graphes spécifiques) a été proposé. Deux algorithmes sont en réalité proposés, le premier se base sur des techniques de marche interpolée (cf. VII-C). Le deuxième utilise une combinaison de marche interpolée et un algorithme d'avancement quantique rapide (quantum fast-forwarding).
Dans [4] Simon Apers and Alain Sarlette, montrent qu'il était possible de construire une marche quantique qui à partir d'un état $|v\rangle$ on atteint un état $\epsilon-$proche de $D^t|v\rangle$ où $D=\sqrt{PoP^T}$ est la matrice discriminante [12] et ce dans un temps $O(\sqrt{\frac{1}{\epsilon}log(\frac{1}{\epsilon})})$.

E. Marches quantiques et réseaux électriques

Dans [6] Belovs et al. ont montré que, dans un graphe $G=(V,E)$ pondéré[13], on peut atteindre un élément marqué, s'il existe, en un temps $\sqrt{O(WR)}$ où W est le poid total ($W=\sum_{e\in E} w_e$) et R étant la résistance effective de G[14].
Cet algorithme permet de résoudre le problème de 3-distinctness.

F. Marches quantiques pour les algorithmes de retour arrière

Les algorithmes de retour arrière (backtracking) ont été très utilisés pour résoudre des problèmes satisfaction de contraintes (optimisation, planning ou décision). Ces algorithmes permettent de tester systématiquement l'ensemble des instanciations potentielles d'un ensemble de variables devant satisfaire une assertion donnée et d'en trouver une solution optimale quand c'est possible ou d'utiliser des heuristiques pour trouver une très bonne solution. Le retour arrière vient du fait qu'en commençant dans un certain ordre avec une solution partielle celle-ci est augmentée en en instanciant une variable supplémentaire puis testant les contraintes du problème, et dès qu'une incohérence apparaît en retourne en arrière pour changer récursivement la dernière variable instanciée. En d'autres termes, le retour en arrière ou sur trace est un parcours en profondeur sur l'arbre de décision du problème.
Ces techniques ont été utilisées pour la recherche des coupes minimales ou d'impliquants premiers (cf. [25], [23], et [10]).
Dans [24] Ashley Montanaro, a utilisé la marche quantique de Belovs pour obtenir une accélération quantique dans les algorithmes de retour en arrière (cf. algorithme DPLL pour Davis, Putnam, Logemann et Loveland [9]).
L'algorithme en question trouve une solution (avec une borne d'erreur) à un problème de satisfaction de contraintes en un temps en $O(\sqrt{T}n^{3/2}logn))$ (où T est le nombre de nœuds).

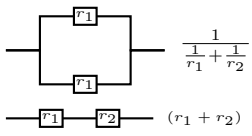
VIII. Approaches hybrides : divide and quantum algorithms

Les approches hybrides sont pour la période actuelle le meilleur compromis "quantique" entre des machines quantiques encore limitées à un nombre réduit de quantum bits et des algorithmes classiques. Ils visent à combiner deux algorithmes, un classique qui permet de "diviser" le problème en instances que les ordinateurs quantiques pourront—vis-à-vis de leur taille— résoudre. Le deuxième consiste à résoudre les instances réduites. Ce sont des approches assez similaires à l'approche divide & conquer.
Dans [31], une approche hybride a été utilisée pour chercher les séquences minimales représentant les scénarios d'échec d'un système en utilisant des exécutions successives d'algorithmes quantiques basés sur la marche quantique.
Pour le cas statique, plusieurs solveurs classiques de 3-SAT[15] sont plus performants que des solveurs

[12](D = P si P est symétrique).

[13]Dont les arcs $e\in E$ ont chacun un poids $w_e\geq 0$.

[14]R peut être déduite en utilisant les mêmes procédures pour déterminer la résistance d'un circuit en utilisant les règles suivantes :



[15]Il s'agit de la restriction du problème SAT aux formules qui sont des formes normales conjonctives avec au plus 3 littéraux (ou exactement 3). Voici un exemple d'une telle forme normale conjonctive :
$(v_1\vee v_2\vee v_3)\wedge(\neg v_1\vee v_2\vee v_4)\wedge(\neg v_1\vee v_2\vee\neg v_5)$ La formule ci-dessus a 3 clauses, 5 variables v_1, v_2, v_3, v_4, v_5 et trois littéraux par clauses. Il s'agit de déterminer si l'on peut attribuer une valeur Vrai ou Faux à chaque variable v_i de façon à rendre toute la formule vraie. (src Wikipedia)

à force-brute. Ils sont en général caractérisés par un temps de calcul de l'ordre de $O^*(2^{\gamma n})$ où γ est une constante telle que $\gamma \in (0,1)$. L'un des meilleurs algorithmes pour résoudre ce problème étant celui de Schöning [27]. Ce dernier initialise une instanciation aléatoire de variables, ensuite trouve de façon répétitive une clause insatisfaite. Ensuite de façon aléatoire sélectionne un littéral et flippe la variable correspondante (un littéral l peut être x ou $\neg x$). L'algorithme termine une fois une instanciation satisfaisante est trouvée ou que le processus a été itéré $O(n)$ fois.

Schöning a démontré que la probabilité de trouver une instanciation satisfaisante est au moins de $(3/4)^n$, qui peut conduire à un algorithme de Monté Carlo avec un temps de calcul en $O^*(2^{\gamma_0 n})$ avec $\gamma_0 := log_2(4/3) \approx 0,415$. Pour accélérer le calcul classique il suffit de réduire ce γ et voir dans quelle mesure des ordinateurs quantiques de petite taille (donc disponibles ou le seront bientôt) peuvent contribuer à le faire.

Dans [11], Dunjko et al. ont présenté un algorithme hybride qui arrive à réduire suffisamment le γ en question et faire de sorte que sur un ordinateur quantique avec M qubits on arrive à résoudre des instances de 3SAT de taille $n \gg M$. L'idée est de considérer un ordinateur quantique avec $M = cn$ qubits où $c \in (0,1)$ est une constante arbitraire. L'algorithme résout le problème 3SAT avec n variables dans un temps de l'ordre de $O^*(2^{(\gamma_0 - f(c) + \epsilon)n})$ où $f(c) > 0$ est une constante et ϵ peut être arbitrairement petit. L'algorithme proposé se base sur le concept " *divide & conquer*", mais c'est plutôt un " *divide & quantum*". En effet, il consiste à considérer un recouvrement d'une instance $x \in \{0,1\}^n$ avec des boules $B_r(x)$ centrées en x (i.e. l'ensemble des chaines binaires y avec une distance de Hamming[16] inférieure à r). L'algorithme prend comme entrée une formule spécifiée avec un ensemble de clauses avec au plus 3 littéraux, un rayon r et un centre x. L'algorithme vérifie certaines conditions de satisfiabilité/non-satisfiabilité (si $r \leq 0$ et $F(x) = 0$ ou si une clause est vide) ou si x est une instanciation satisfaisante. Sinon, dans l'étape suivante, il considère la première clause C insatisfaite et fait appel à $PROMISEBALL(F_{|l=1}, r-1; x)$ pour chaque littéral l de C (où $F_{|l=1}$ désigne la formule obtenue en posant l à 1 (ce qui revient à tronquer toutes celles qui contiennent $\bar{l}$).

Cet algorithme réduit r à chaque étape, dès que r est suffisamment petit on pourra dans ce cas utiliser un algorithme quantique pour résoudre la formule dans les $r-$ boules correspondantes.

[16]La distance de Hamming (au nom de Richard Hamming), permet de quantifier la différence entre deux séquences de symboles. C'est une distance au sens mathématique du terme. À deux suites de symboles de même longueur, elle associe le nombre de positions où les deux suites diffèrent. Par exemple, la distance de Hamming entre 1011101 et 1001001 est 2, celle entre 2173896 and 2233796 est 3 (cf. Wikipedia).

IX. Conclusion

Dans ce papier, on a présenté une esquisse de ce que pourrait être le calcul EPS dans un cadre quantique. Plusieurs approches semblent adéquates pour les différents cadres d'EPS statique (calcul de coupes minimales ou d'impliquants premiers et réductions Booléenne) et dynamique (méthode des espaces d'états).

Plusieurs algorithmes dans différentes classes d'algorithmes quantiques ont été présentées. D'abord pour la fusion Booléenne avec un certain nombre d'outils basés sur les approches de requêtes à la Grover, ou en utilisant les approches basées sur les marches quantiques. Ces dernières se retrouvent presque partout dans ces calculs combinatoires avec des recherches dans de très grands graphes. Le speedup est en général quadratique.

Pour l'approche dynamique, on peut utiliser plusieurs approches. On peut envisager les marches quantiques bien entendu, mais aussi des variantes de l'algorithme HHL qui apparait plus approprié, mais avec une accelération exponentielle.

Toutes ces approches sont basées sur un usage à l'échelle du nombre de qubits. Il est donc nécessaire dans ce cas de faire une agrégation pour permettre un usage sur des machines quantiques actuelles ou disponibles dans un futur proche.

Parmi les pistes prometteuses, on trouve les approches hybrides qui permettent de combiner la puissance du calcul classique et leur capacité à tirer profit des structures des problèmes qu'elles traitent, mais aussi la force quantique brute ou de bons raccourcis quantiques localement pour accélérer les étapes fastidieuses pour un calcul classique.

Il est plus que nécessaire vu l'actualité de la course au quantique de regarder de plus près certaines de ces approches et de regarder nos problèmes EPS à la lumière de ces nouvelles capacités de calcul.

References

[1] A Ambainis, A M Childs, B W Reichardt, R Spalek, and S Zhang. Any AND-OR Formula of Size N can be Evaluated in time $N^{1/2 + o(1)}$ on a Quantum Computer. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, pages 363–372, 2007.

[2] Andris Ambainis. New Developments in Quantum Algorithms. In *Mathematical Foundations of Computer Science 2010, 35th International Symposium, {MFCS} 2010, Brno, Czech Republic, August 23-27, 2010. Proceedings*, pages 1–11, 2010.

[3] Andris Ambainis, András Gilyén, Stacey Jeffery, and Martins Kokainis. Quadratic speedup for finding marked vertices by quantum walks, 2019.

[4] Simon Apers and Alain Sarlette. Quantum fast-forwarding: Markov chains and graph property testing. *Quantum Information & Computation*, 19:181–213, 2018.

[5] A-C. Baboin. *Calcul Quantique : Algebre Et Geometrie Projective*. PhD thesis, Université de Franche-Comté, 2011.

[6] Aleksandrs Belovs, Andrew M Childs, Stacey Jeffery, Robin Kothari, and Frédéric Magniez. Time-Efficient Quantum Walks for 3-Distinctness. In *Automata, Languages, and Programming - 40th International Colloquium, {ICALP} 2013, Riga, Latvia, July 8-12, 2013, Proceedings, Part {I}*, pages 105–122, 2013.

[7] H C Berg. *Random walks in Biology*. Princeton University Press, 1993.

[8] Andrew M Childs, Richard Cleve, Stephen P Jordan, and David L Yonge-Mallo. Discrete-Query Quantum Algorithm for {NAND} Trees. *Theory of Computing*, 5(1):119–123, 2009.

[9] Martin Davis and Hilary Putnam. A Computing Procedure for Quantification Theory. *J. ACM*, 7(3):201–215, 1960.

[10] David Déharbe, Pascal Fontaine, Daniel Le Berre, and Bertrand Mazure. Computing prime implicants. In *Formal Methods in Computer-Aided Design, {FMCAD} 2013, Portland, OR, USA, October 20-23, 2013*, pages 46–52, 2013.

[11] Y Dunjko, V Ge, and Ignacio Cirac J. Computational Speedups Using Small Quantum Devices. *PHYSICAL REVIEW LETTERS*, 121(250501), 2018.

[12] M D SAJID et al. ANIS. Qiskit: An Open-source Framework for Quantum Computing, 2021.

[13] L Grady. Random Walks for Image Segmentation. *{IEEE} Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1768–1783, 2006.

[14] M Hibti. What if we revisit evaluation of PSA models with network algorithms ? Ans psa 2013 international topical meeting on probabilistic safety assessment and analysis, 2013.

[15] M Hibti, M Hasseni, and N Villatte. EPS Zoomable pour le screening et la prospection. Technical report, EDF R&D, 2020.

[16] J C Hull. *Options, Futures and Other Derivatives*. Prentice Hall, 2005.

[17] Stacey Jeffery and Shelby Kimmel. Quantum Algorithms for Graph Connectivity and Formula Evaluation. *CoRR*, abs/1704.0, 2017.

[18] A Yu. Kitaev. Quantum measurements and the Abelian Stabilizer Problem, 1995.

[19] Knuth D. E. *The Art of Computer Programming, vol 4, Boolean Basics*. Addison-Wesley.

[20] Hari Krovi, Frédéric Magniez, Maris Ozols, and Jérémie Roland. Quantum Walks Can Find a Marked Element on Any Graph. *Algorithmica*, 74:851–907, 2015.

[21] X Lacour. *Information quantique par passage adiabatique: portes quantiques et décohérence*. PhD thesis, Université de bourgogne, 2007.

[22] L D Landau and E M Lifshitz. Statistical Physics, Third Edition,. volume Part 1: Vo, chapter Course of. Butterworth-Heinemann, 1980.

[23] Vasco M Manquinho, Paulo F Flores, João P Marques Silva, and Arlindo L Oliveira. Prime Implicant Computation Using Satisfiability Algorithms. In *9th International Conference on Tools with Artificial Intelligence, {ICTAI} '97, Newport Beach, CA, USA, November 3-8, 1997*, pages 232–239, 1997.

[24] Ashley Montanaro. Quantum-Walk Speedup of Backtracking Algorithms. *Theory of Computing*, 14(1):1–24, 2018.

[25] Clara Pizzuti. Computing Prime Implicants by Integer Programming. In *Proceedings of the 8th International Conference on Tools with Artificial Intelligence*, ICTAI '96, pages 332—-, Washington, DC, USA, 1996. IEEE Computer Society.

[26] Antoine Rauzy. Notes on Computational Uncertainties in Probabilistic Risk/Safety Assessment. *Entropy*, 20(3), 2018.

[27] Schoning, Uwe. A Probabilistic Algorithm for k-SAT and Constraint Satisfaction Problems. *Proceedings of the 40th Annual Symposium on Foundations of Computer Science*, pages 410—-, 1999.

[28] Salvador Elías Venegas-Andraca. Quantum walks: a comprehensive review. *Quantum Information Processing*, 11(5):1015–1106, 2012.

[29] S Ward. Earthquake Simulation by Restricted Random Walks. *Bulletin of The Seismological Society of America - BULL SEISMOL SOC AMER*, 94:2079–2089, 2004.

[30] Ahmed Zaiou, Younès Bennani, Mohamed Hibti, and Basarab Matei. Quantum Approach for Vertex Separator Problem in Directed Graphs. *Encours*, 2022.

[31] Ahmed Zaiou, Younès Bennani, Basarab Matei, and Mohamed Hibti. A quantum learning approach based on Hidden Markov Models for failure scenarios generation. *arXiv preprint arXiv:2204.00087*, 2022.