ORSAY
N° d'ordre :

UNIVERSITÉ PARIS-SUD
UFR SCIENTIFIQUE D'ORSAY

THÈSE

présentée pour obtenir

LE GRADE DE DOCTEUR EN SCIENCES
DE L'UNIVERSITÉ PARIS XI ORSAY

Spécialité : INFORMATIQUE

par

Frédéric MAGNIEZ

Sujet :

Auto-test pour les calculs approché et quantique

ou *«Comment tester sans savoir faire ?»*

Soutenue le devant la Commission d'examen :

M.	HARRY BUHRMAN	Rapporteur
M.	ROBERT CORI	
MME.	MARIE-CLAUDE GAUDEL	
M.	PASCAL KOIRAN	Rapporteur
M.	MIKLOS SANTHA	Directeur de thèse
M.	JACQUES STERN	

Table des matières

[2]

Table des figures

Introduction

1. Problématique

Conception et test sont indissociables dans toutes disciplines et plus encore en informatique. Si la nécessité de tester un programme ou plus généralement une machine va de soi, les méthodes pour y parvenir sont beaucoup moins évidentes. Se convaincre de la validité d'une machine est déjà une tâche importante et non triviale, mais convaincre une personne extérieure et non experte est encore plus ardu. En informatique, le problème du test a pris ces dernières années une importance toute particulière. Il vient d'une attente réelle non seulement de l'utilisateur courant en informatique, mais aussi des concepteurs eux-mêmes. La complexité aidant, la fiabilité des programmes est de plus en plus fragile. L'utilisateur suspicieux de cette fiabilité aimerait s'en convaincre de manière autonome et sans avoir à s'investir dans la compréhension du programme. Ce constat s'applique aussi aux concepteurs eux-mêmes qui ne peuvent garantir la qualité de leurs propres programmes sans une série de tests approfondis. Face à cette situation, se dégage la volonté d'une procédure de test simple et fiable, dans le sens qu'elle ne doit faire intervenir que des procédés maîtrisés et moins complexes que ceux invoqués dans la conception même.

Empiriquement, un test consiste en une série d'expériences où la machine est mise à l'épreuve. Se posent alors plusieurs problèmes. Citons ceux du choix des expériences, de leur pertinence, ou encore de la façon de détecter un comportement erroné sur l'une d'entre elles. Pour ce dernier point, une première solution consiste à utiliser une machine de référence correcte. Même si cette méthode est en fin de compte courante, elle n'est pas satisfaisante. Elle suppose l'existence d'une telle machine, ce qui ne fait que contourner le problème. De plus cette procédure de test est alors aussi complexe que la réalisation d'une machine correcte, ce qui n'est pas non plus satisfaisant. Enfin, le choix des expériences est délicat. Si ce choix est aléatoire, alors l'analyse statistique qui en découle ne permet pas *a priori* de déceler les erreurs ponctuelles. Une alternative est de développer une méthodologie formelle du choix des tests. En informatique, cette méthodologie prend en compte la spécification formelle du programme, le source du programme ou seulement l'idée que l'on s'en fait. Une autre solution formelle est la *vérification de programme*. Elle consiste à démontrer la validité du programme à la vue de son source écrit dans un langage approprié. Ces deux procédés dépendent fortement du source, ou d'un certain nombre d'hypothèses faites sur celui-ci. De plus, ces méthodes formelles peuvent s'avérer extrêmement complexes et difficiles même pour de simples programmes. Pour la vérification, l'assistance humaine est même souvent nécessaire.

Une des premières solutions contournant ces handicaps a été proposée au début des années 90 en informatique par Blum et Kannan [BK89]. Ils ont introduit la notion de test de résultat (result checking). Avant de décrire cette notion, précisons que dans ce modèle le programme testé est vu comme une boîte noire. Le testeur ne peut que demander la valeur du programme sur une entrée choisie, et ne peut avoir accès ni à son mécanisme interne, ni à son procédé de conception. Etant donnée une fonction f censée être implémentée par un programme P, et une entrée x fixée, un *testeur de résultat* acceptera avec grande probabilité un programme P s'il calcule partout f, et le refusera avec grande probabilité si $P(x) \neq f(x)$. Dans le cas où $P(x) = f(x)$ mais P diffère de f sur d'autres entrées, la réponse du testeur de résultat

n'est pas spécifiée. Contre toute attente, Blum et Kannan ont montré comment effectuer cette tâche plus efficacement et simplement que le calcul même de la fonction pour une série de problèmes numériques dont le tri, le calcul du rang d'une matrice, et le calcul du plus grand diviseur commun de deux entiers. L'existence de testeurs de résultat efficaces suggère l'utilisation systématique de ces derniers lors de chaque appel à un programme non fiable. Ce procédé ne ralentit que très peu ce programme puisque la complexité de l'ensemble testeur de résultat et programme est alors du même ordre de grandeur que celle du programme initial.

Ces idées ont été reprises et améliorées en auto-test et auto-correction [BLR90, Lip91]. Un *auto-testeur* (self-tester) vérifie si un programme est correct sur la plupart des entrées, et un *auto-correcteur* (self-corrector) transforme un programme correct sur beaucoup d'entrées en un programme probabiliste correct partout. L'existence d'un couple auto-testeur/correcteur peut paraître surprenante. Non seulement il implique l'existence d'un testeur de résultat, mais il permet aussi d'utiliser avec grande fiabilité des programmes erronés. De plus, lorsque le couple auto-testeur/correcteur est suffisamment efficace, la complexité du programme corrigé est du même ordre de grandeur que celle du programme erroné. Ce point est fondamental et suggère encore une implémentation systématique de tels procédés. L'apport supplémentaire par rapport aux testeurs de résultat est qu'ici le couple auto-testeur/correcteur ne se contente pas de détecter une erreur, il la corrige ! Blum, Luby, et Rubinfeld [BLR90] ont développé des couples d'auto-testeurs/correcteurs pour une série de fonctions numériques. La viabilité pratique de ce concept a aussi été éprouvée sur le *bug* concernant la routine de division des premiers processeurs PENTIUM par Blum et Wasserman [BW96]. Ils ont montré comment ce problème aurait pu être détecté et corrigé efficacement par des techniques connues d'auto-test/correction.

L'étape préliminaire à l'auto-correction est donc l'auto-test. En général l'auto-correction est aisée une fois établi un certain degré de fiabilité du programme. Elle peut être vue en terme d'auto-réductibilité aléatoire (random self-reducibility). Plusieurs variantes d'auto-réductibilité existent, et nous n'en présentons ici qu'une variante. Une fonction f définie sur un domaine D fini est *c-aléatoirement auto-réductible* [BLR90], si pour toute entrée $x \in D$, la valeur $f(x)$ peut être calculée à l'aide de $x, a_1, \ldots, a_c$ et $f(a_1), \ldots, f(a_c)$, où $a_1, \ldots, a_c$ sont aléatoirement calculés à partir de x tels que chaque a_i est uniformément distribué sur D. Cette notion a beaucoup d'autres applications en protocoles cryptographiques, preuves interactives, et complexité structurelle. Feigenbaum [Fei93] passe en revue les variantes de cette notion ainsi que ses applications.

Définir le problème de l'auto-test pour une unique fonction est trop restrictif. Il est en fait plus intéressant de le définir pour un ensemble de fonctions, dites *bonnes*. Il s'agit alors de décider si un programme calcule fidèlement une bonne fonction, ou dit autrement s'il vérifie de bonnes propriétés. Cette généralisation correspond à ce qui est fait en pratique. Une fois établi le fait qu'un programme calcule une bonne fonction, il est plus aisé de déterminer si cette fonction est celle voulue. Pour illustration, considérons l'exemple des fonctions linéaires sur $\mathbb{Z}$ ou sur $\mathbb{Z}_n$, l'ensemble des entiers modulo n, *i.e.* les fonctions de la forme $x \mapsto ax$, pour une constante fixée a. Ainsi savoir si un programme calcule une fonction linéaire donnée est une tâche facile lorsqu'il est déjà établi que le programme calcule une fonction linéaire. Le problème de l'auto-test selon Blum, Luby, et Rubinfeld [BLR90] pour le calcul exact peut donc s'énoncer ainsi.

Problème. *Etant donnés $0 \leq \delta_1 < \delta_2 < 1$, et $\mathcal{F}$ un ensemble de fonctions, construire une machine de Turing T probabiliste à oracle simple et efficace telle que pour tout programme P :*

- *s'il existe une fonction $f \in \mathcal{F}$ telle que la proportion des entrées où P ne calcule pas f est inférieure à δ_1, alors T avec oracle P retourne* `BON` *avec grande probabilité ;*
- *si pour toute fonction $f \in \mathcal{F}$ la proportion des entrées où P ne calcule pas f est supérieure à δ_2, alors T avec oracle P retourne* `MAUVAIS` *avec grande probabilité.*

Remarque. Si le but est uniquement de détecter une erreur potentielle du programme, la borne δ_2, dite *seuil de rejet*, est la plus importante et autant fixer $\delta_1 = 0$. Par contre, l'estimation du degré de fiabilité du programme, nécessaire pour effectuer une auto-correction, requiert le choix d'un δ_1, dit *seuil d'acceptance*, aussi près de δ_2 que possible.

Dans cette première formalisation, dire que «P calcule f en x» signifie en calcul exact que $P(x) = f(x)$, mais peut avoir une définition plus relâchée en calcul approché. Si jusqu'ici l'auto-test s'était essayé uniquement à l'informatique classique pour des situations de calcul exact ou approché avec erreur absolue, il est tout à fait raisonnable de l'envisager pour d'autres modèles de calculs et pour le test de toute machine en général. Vérifier le mécanisme interne d'une machine peut nécessiter un équipement rare et complexe. Lorsqu'un individu se trouve face à un programme ou à une machine dont il ne veut ou ne peut étudier le fonctionnement interne, il aimerait pourtant sans aucune assistance et avec peu de temps et de moyens s'auto-convaincre de leur validité !

Dans cette thèse nous nous concentrons sur la notion de l'auto-test. Nous étendons pour la première fois cette notion à toute classe d'objets sans restriction et définissons un modèle général de l'auto-test. Nous essayons et validons ce modèle pour le calcul approché en général et pour le calcul quantique. Nous espérons ainsi exhiber le pouvoir des techniques de l'auto-test à de nouvelles situations de calcul, et répondre en partie à des attentes actuelles tant théoriques que pratiques. Avant de présenter nos résultats, nous définirons dans la prochaine section ces deux notions de calcul, et présenterons l'état de l'art de l'auto-test dans la suivante.

2. Deux modèles de calculs

2.1. Le calcul approché. Le modèle de calcul exact est trop restrictif pour la plupart des situations en calcul numérique. Le calcul approché prend en considération l'impossibilité de calculer exactement un nombre, une solution, ou encore une fonction. De multiples situations motivent cette relâche. Il est impossible de représenter des nombres tels que π avec précision infinie. De plus la représentation même des nombres sur ordinateur implique des erreurs inévitables de précision dues aux arrondis. Enfin, pour de nombreux problèmes numériques, malgré que les solutions exactes ne soient pas explicitement connues, elles peuvent être approchées par des procédés itératifs. Afin de décrire la qualité de ces approximations, un modèle de calcul approprié doit être défini.

En calcul exact, dire qu'un programme P calcule correctement une fonction f sur une entrée x traduit l'égalité $P(x) = f(x)$. En calcul approché, la notion de correction d'un calcul peut être très variée. Elle peut être par exemple liée à l'erreur absolue ou relative.

Dans le cas général, l'erreur maximale tolérée sur un calcul supposé produire la valeur $v \in R$ sur l'entrée $x \in D$ est un *terme d'erreur*, *i.e.* une fonction à valeurs dans $\mathbb{R}_+$, ne dépendant que de x et v. Un tel terme d'erreur est appelé dans ce contexte un *terme d'erreur de calcul*. Plus précisément si les valeurs résultats appartiennent à un espace métrique (R, d), le calcul de la fonction f par le programme P est *correct* en $x \in D$, pour le terme d'erreur de calcul $\varepsilon : D \times R \to \mathbb{R}_+$, si $d(P(x), f(x)) \leq \varepsilon(x, f(x))$, et est *incorrect* si $d(P(x), f(x)) > \varepsilon(x, f(x))$. Ce formalisme permet de modéliser plusieurs modèles de calcul :

– calcul exact :

$$\forall (x, v) \in D \times R, \qquad \varepsilon(x, v) \stackrel{\text{déf}}{=} 0,$$

– calcul approché avec erreur absolue :

$$\forall (x, v) \in D \times R, \qquad \varepsilon(x, v) \stackrel{\text{déf}}{=} \varepsilon_0,$$

– calcul approché avec erreur ne dépendant que de l'entrée :

$$\forall (x, v) \in D \times R, \qquad \varepsilon(x, v) \stackrel{\text{déf}}{=} \varepsilon_1(x),$$

– calcul approché avec erreur relative, s'il existe une norme $\|\cdot\|$ associée à d sur R :

$$\forall (x,v) \in D \times R, \qquad \varepsilon(x,v) \stackrel{\text{déf}}{=} \theta \|v\|,$$

où $\varepsilon_0 \in \mathbb{R}_+$, $\varepsilon_1 : D \to \mathbb{R}_+$, et $\theta \in \mathbb{R}_+$ sont fixés.

2.2. Le calcul quantique. Depuis une dizaine d'années, le calcul quantique est un domaine de recherche très actif. Le problème de la simulation d'un système physique quantique par un ordinateur construit sur les règles de la mécanique classique remonte à Feynman [Fey82]. Selon lui, le pouvoir de calcul d'une machine utilisant des composants quantiques était susceptible d'être plus important que celui d'une machine classique. Cette éventualité constitue un nouveau défi à la version quantitative de la thèse de Church-Turing. Rappelons que cette thèse affirme que toute machine de calcul physiquement réalisable est simulable par une machine de Turing probabiliste dont le temps de calcul est polynomialement borné en fonction de celui de la machine simulée. Les premiers modèles de calcul quantique, la machine de Turing quantique et les circuits quantiques, ont été définis par Deutsch [Deu85, Deu89]. Yao [Yao93] a montré que ces deux modèles étaient polynomialement équivalents.

Historiquement, c'est d'abord en cryptographie que le premier résultat d'informatique quantique est apparu. Il s'agit du protocole de distribution de clefs secrètes de Bennett et Brassard [BB84] qui a été prouvé parfaitement sécuritaire. Ce premier résultat était à la fois surprenant et motivant. Mais le résultat qui a le plus marqué et donné son essor à cette discipline est celui de Shor [Sho94]. Il a en effet expliqué comment des nombres entiers pourraient être factorisés en temps polynomial sur un ordinateur quantique. Ce problème est réputé difficile en informatique classique, c'est pourquoi la sécurité de nombreux protocoles cryptographiques repose sur la difficulté de factoriser. Il va sans dire que la construction effective d'un ordinateur quantique aurait entre autre de sérieuses répercussions en cryptographie. L'autre résultat majeur est celui de Grover [Gro96], qui explique comment rechercher avec une machine de Turing quantique un élément dans une base de données non structurée de taille N en seulement $O(\sqrt{N})$ requêtes, alors qu'une machine classique ou probabiliste nécessite $\Omega(N)$ requêtes. Cette supériorité du calcul quantique a été confirmée par une suite d'articles [DJ92, BB92, BV97, Sim94] exhibant des oracles pour lesquels les machines de Turing quantiques sont superpolynomialement plus puissantes que leurs analogues probabilistes. Enfin récemment la possibilité de téléporter de l'information quantique a suscité beaucoup d'émois dans la communauté scientifique et les médias. La possibilité théorique de la téléportation a été exhibée par Bennett, Brassard, Crépeau, Jozsa, Peres, et Wootters [BBJ$^+$93] et implémentée pratiquement par Bouwmeester, Pan, Mattle, Eibl, Weinfurter, et Zeilinger [BPM$^+$97].

La base du calcul quantique est le *qubit* représentant les états possibles de l'analogue quantique du bit classique. Si un bit classique est un élément de $\{0,1\}$, et un bit probabiliste un élément de $\mathbb{R}^{\{0,1\}}$ à coordonnées positives ou nulles et de somme 1, alors un bit quantique est un élément de l'espace de Hilbert $\mathbb{C}^{\{0,1\}}$ et de norme 1. Un système de n qubits est donc représenté par un vecteur normé de l'espace de Hilbert $\mathbb{C}^{\{0,1\}^n}$. Un circuit quantique opère sur un tel système en tant qu'application linéaire préservant la norme, il s'agit donc d'une transformation unitaire. Un circuit quantique est construit à partir de portes quantiques effectuant des transformations unitaires locales, *i.e.* agissant uniquement sur un nombre constant de qubits. A la fin du calcul, une mesure est effectuée sur un ou plusieurs qubits. Cette mesure correspond à ce que nous appellerons la mesure de von Neumann dans la base de calcul. La mesure d'un qubit est une expérience probabiliste produisant 0 ou 1, de sorte que la probabilité d'observer b est le carré de la norme de la projection orthogonale du vecteur décrivant l'état du système sur l'espace compatible avec cette observation.

Les portes unitaires quantiques agissant sur peu de qubits sont très étudiées pour plusieurs raisons. L'une d'elles étant qu'à ce jour seules les portes à moins de trois qubits ont pu être construites. La possibilité de construire dans l'avenir des portes utilisant un grand nombre

de qubits est incertaine. Les technologies actuelles sont entre autre basées sur les trappes linéaires à ions proposées par Cirac et Zoller [CZ95], et sur des techniques NMR proposées par Gershenfeld et Chuang [GC97] et par Cory, Fahmy et Havel [CFH97]. L'autre raison est qu'il existe des ensembles universels de portes à moins de trois qubits. Un ensemble de portes est universel si n'importe quel circuit peut être approximativement simulé par un autre construit uniquement à partir de ces portes. Le premier ensemble universel constitué de portes agissant sur au plus trois qubits a été exhibé par Deutsch [Deu89]. Après une suite de travaux sur de telles familles [DiV95, Bar95, DBE95, Llo95, BBC$^+$95, Sho96, KLZ96, Kit97], Boykin, Mor, Pulver, Roychowdhury, et Vatan [BMP$^+$99] ont exhibé un ensemble universel fini très simple à base de trois portes usuelles et déjà implémentées. Deux de ces portes agissent sur un seul qubit, et la troisième sur deux qubits. De plus cet ensemble possède la propriété d'être tolérant à l'erreur (*fault-tolerant*), *i.e.* qui résiste à d'éventuelles perturbations physiques.

Nous serons amené à considérer une forme plus générale des états d'un système physique. Effectivement, l'état d'un système peut être plus complexe que ceux définis précédemment, appelés états *purs*. Il peut aussi être *mélangé*, *i.e.* être une distribution de probabilité d'états purs. De tels états mélangés sont par exemple produits suite à la mesure partielle d'un état pur. Les états mélangés d'un système de n qubits sont représentés par des matrices complexes de dimension 2^n particulières appelées *matrices densités*. Entre autre la matrice densité ρ caractérisant l'état d'un système est telle que la probabilité que ce système soit dans l'état pur v est donnée par le réel $v^\dagger \rho v$, où $v^\dagger$ représente le vecteur adjoint associé à v. Un des intérêts majeurs de ce formalisme est qu'il décrit fidèlement les états possibles d'un systèmes ainsi que ses évolutions physiquement autorisées. Ces évolutions sont mathématiquement décrites par des opérateurs agissant sur les matrices densités et appelées *superopérateurs complètement positifs* (CPSO). Dans ce formalisme, les opérateurs unitaires et les mesures sont des CPSO particuliers. Les circuits quantiques ont été redéfinis pour ce modèle par Aharonov, Kitaev, et Nisan [AKN98], qui ont entre autre montré qu'ils étaient polynomialement équivalents aux circuits précédents définis pour des états purs.

3. Travaux antérieurs

3.1. Auto-test exact. Les premiers auto-testeurs ont d'abord été construits pour des programmes définis sur des groupes finis dans un cadre de calcul exact, *i.e.* sans erreur de précision. Dans ce modèle Blum, Luby, et Rubinfeld [BLR90] ont développé plusieurs auto-testeurs pour des fonctions numériques telles que la multiplication modulaire, la multiplication de polynômes et de matrices, l'inversion de matrice, et le calcul de déterminant. Ces auto-testeurs utilisent la possibilité d'auto-tester simplement l'ensemble des homomorphismes entre deux groupes. Il s'agit du test de linéarité [BLR90].

Etudions de plus près le problème du test de linéarité. Etant donné un programme P calculant une fonction entre un groupe abélien G et un autre groupe, le problème est de vérifier si P calcule un certain homomorphisme sur la plupart des entrées de G. Lorsque G désigne l'anneau $\mathbb{Z}_n$ des entiers modulo n, l'ensemble des homomorphismes de $\mathbb{Z}_n \to \mathbb{Z}_n$ sont les applications f pour lesquelles il existe un élément $a \in \mathbb{Z}_n$ tel que $f(x) = ax$, pour tout $x \in \mathbb{Z}_n$. Le calcul de f nécessite intuitivement une multiplication, alors qu'il existe un auto-testeur qui n'utilise que peu d'additions.

Cet auto-testeur est basé sur l'*équation de Cauchy*, ou *de linéarité*, $f(x+y) = f(x)+f(y)$, pour tout $x, y \in G$, qui est satisfaite quand f est un homomorphisme. Il consiste à vérifier que l'équation précédente est satisfaite pour des instances aléatoires. Plus précisément, il vérifie que $P(x+y) = P(x)+P(y)$, pour deux entrées x, y prises uniformément aléatoirement dans G. Si la probabilité que le test échoue est faible, alors P calcule bien un homomorphisme sauf sur une fraction tout aussi faible des entrées. Plus formellement, si $\mathsf{Rej}(P)$ désigne la probabilité de rejet au test, et $\mathsf{Dist}(P, h)$ la fraction des entrées sur lesquelles P ne calcule pas la fonction

h, alors

$$\mathsf{Rej}(P) < 1/6 \implies \mathrm{Inf}\{\mathsf{Dist}(P,h) : h \text{ homomorphisme}\} \leq \frac{\mathsf{Rej}(P)}{2}.$$

Cette caractéristique de l'équation de linéarité est appelée *robustesse*. Ce terme a été introduit par Rubinfeld et Sudan [RS92], et étudié plus généralement pour les propriétés définies à partir d'équations fonctionnelles par Rubinfeld [Rub94]. Pour le cas $G = (\mathbb{Z}_2)^n$, la relation entre distance d'un programme aux fonctions linéaires et probabilité de rejet au test de linéarité a fait l'objet d'une étude plus précise et plus tendue par Bellare, Coppersmith, Håstad, Kiwi, et Sudan [BCH$^+$95]. L'analyse du test repose dans tous les cas sur la possibilité d'auto-corriger P avec peu de questions. Soit g la fonction qui prend en $x \in G$ la valeur majorité des votes $(P(x+y) - P(y))$, pour chacun des votants $y \in G$. Quand la probabilité de rejet au test de linéarité est faible, alors la majorité est en fait une quasi-unanimité, g est un homomorphisme, et g égale P sauf sur une fraction des entrées proportionnelle à la probabilité de rejet au test de linéarité. Plus précisément, si $\mathsf{Rej}(P) \leq \delta$, pour un réel $\delta \geq 0$, alors g satisfait :

$$\forall x \in G, \qquad \Pr_{y \in G}[g(x) \neq P(x+y) - P(y)] \leq 2\delta,$$
$$\Pr_{x \in G}[P(x) \neq g(x)] \leq 2\delta,$$

et si $\delta < 1/6$,

$$\forall x, y \in G, \qquad g(x+y) = g(x) + g(y).$$

D'autres travaux ont suivi construisant de nouveaux auto-testeurs pour de nouvelles fonctions ou améliorant les précédents. Citons les travaux de Rubinfeld et Sudan [RS92] qui ont montré comment auto-tester l'ensemble des fonctions polynomiales de degré borné, de Rubinfeld [Rub94] pour les fonctions vérifiant une propriété additive de la forme $f(x+y) = G(f(x), f(y))$ pour une fonction G fixée, d'Ergün [Erg95] pour les transformées de Fourier rapides, l'évaluation de polynômes, la multiplication de polynômes, et de matrices, et de Ravi Kumar et Sivakumar [KS96] pour les fonctions satisfaisant des équations de récurrences linéaires.

3.2. Preuves interactives et transparentes. Ces résultats ainsi que les techniques développées ont permis d'établir de nouvelles caractérisations probabilistes de certaines classes de complexité traditionnelles à l'aide de classes basées sur l'interaction entre un prouveur (ou plusieurs) de pouvoir de calcul illimité, et un vérifieur probabiliste de complexité en temps polynomiale. Intuitivement le vérifieur décide de l'appartenance d'une entrée x à un langage L en utilisant la puissance du prouveur tout en le contraignant à ne pas tricher. Dans le modèle des preuves interactives, un langage est dit dans la classe de complexité IP [GMR89, Bab85] (Interactive Proofs) lorsqu'il existe un vérifieur telle que : si $x \in L$, alors il existe un prouveur qui convainc toujours le vérifieur ; mais si $x \notin L$, alors pour tout prouveur le vérifieur n'est pas convaincu avec grande probabilité. La classe MIP [FRS88] est définie de manière analogue pour plusieurs prouveurs. Enfin la classe PCP($f(n)$,$g(n)$) [AS92] (Probabilistic Checking of Proofs) est définie pour un prouveur non adaptatif qui correspond alors à une fonction oracle, et un vérifieur utilisant $O(f(n))$ bits aléatoires et demandant à l'oracle $O(g(n))$ bits. Les caractérisations IP=PSPACE [LFKN92, Sha92], MIP=NEXPTIME [BFL91], et PCP($\ln(n), 1$)=NP [AS92, ALM$^+$92] reposent toutes sur la même idée. Une première étape, dite d'arithmétisation, consiste pour le vérifieur à transformer une formule booléenne en un polynôme. Dans une seconde étape, le vérifieur utilise le prouveur pour se convaincre que le polynôme satisfait un certain nombre de propriétés spécifiques équivalant à la satisfaisabilité de la formule initiale. Dans cette dernière étape est utilisé implicitement un auto-test des fonctions polynomiales de degré borné. Ces caractérisations ont d'importantes retombées sur

la non-approximabilité, pour certaines erreurs relatives, de plusieurs problèmes d'optimisation comme entre autre le problème de la clique maximale, ou de tout problème Max-SNP-difficile [AS92, ALM+92, FGL+96]. Pour plus de détails sur le sujet, on pourra consulter les synthèses de Babai [Bab93] et d'Arora [Aro98].

3.3. Test de propriété. Lorsque l'objectif de l'auto-test est plus orienté vers la vérification d'une propriété sur le programme, parler de test de propriété [GGR96] est plus approprié. Etant donnée une propriété fonctionnelle Φ, un *testeur de propriété* pour Φ est en fait un auto-testeur pour la classe des fonctions satisfaisant la propriété Φ.

Goldreich, Goldwasser, et Ron [GGR96, GR97] ont développé de tels auto-testeurs pour les graphes vus soit comme des matrices d'adjacence, *i.e.* des fonctions de $\{1,\ldots,n\}^2 \to \{0,1\}$, où n est le nombre de sommets, soit comme des listes d'incidence, *i.e.* des fonctions de $\{1,\ldots,n\}\times\{1,\ldots,d\} \to \{0,1,\ldots,n\}$, où n est le nombre de sommets et d le degré du graphe. Pour le test de propriété, la première représentation est plus appropriée aux graphes denses, et la deuxième aux graphes de degré borné. Chacune de ces représentations induit une distance entre deux graphes définie comme la fraction des entrées où leurs fonctions caractéristiques diffèrent. Un testeur de propriété pour Φ doit donc décider si un graphe donné est proche, pour l'une des deux distances, d'un autre graphe satisfaisant Φ.

Goldreich, Goldwasser, et Ron [GGR96] ont développé de tels testeurs pour plusieurs propriétés de graphes, représentés par des matrices d'adjacence, comme la k-colorabilité, ρ-clique, ρ-coupe (ρ-CUT), et ρ-bissection. Pour les graphes de degré borné, vus comme des listes d'incidence, Goldreich et Ron [GR97] ont traité des propriétés de graphes comme la connectivité, la k-connectivité de sommets (resp. d'arêtes pour $k = 2, 3$), la planarité, et l'absence de cycle.

Puisque certaines de ces propriétés sont déduites de problèmes d'optimisation comme Max-clique et Max-coupe (Max-CUT), Goldreich, Goldwasser, et Ron [GGR96] ont remarqué que leurs testeurs, pour des graphes représentés des matrices d'adjacence, permettaient d'exhiber des schémas d'approximation très efficaces pour les problèmes d'optimisation associés. Prenons l'exemple du problème Max-coupe et supposons l'existence d'une machine de Turing probabiliste qui, prenant en entrée les réels $\rho \geq 0$ et $\varepsilon > 0$, et l'entier $n \geq 1$, soit un testeur sur les graphes à n sommets pour la propriété ρ-coupe, dont le seuil de rejet est inférieur à $\varepsilon > 0$, et admettant pour complexité en temps une fonction $t(\varepsilon, n)$ indépendante de ρ. Alors cette machine peut être utilisée pour approcher, avec erreur additive εn^2, la taille de la coupe maximale avec une complexité en temps en $O(t(\varepsilon, n)/\varepsilon)$. Puisque cette erreur est relative si le graphe est dense, et qu'un tel testeur existe [GGR96] pour $t(\varepsilon, n) = 2^{\tilde{O}(1/\varepsilon^3)}$[1], Max-coupe peut donc être approché pour les graphes denses en temps constant et avec erreur relative. Les précédents résultats d'Arora, Karger, et Karpinski [AKK95] d'une part, et de Fernandez de la Vega [Fer96] d'autre part, avaient exhibé des schémas d'approximation de complexités en temps respectives $n^{O(1/\varepsilon^2)}$ et $2^{\tilde{O}(1/\varepsilon^2)}n^2$. Les techniques d'auto-test en général ont donc permis d'améliorer significativement ces bornes. Inversement, il est naturel de se demander si des schémas d'approximation peuvent entraîner des testeurs pour les propriétés associées. Actuellement la réponse n'est pas connue, même si l'on observe que la structure de plusieurs schémas induisent naturellement des testeurs.

Le traitement des schémas d'approximation pour les graphes denses est actuellement en plein essor. Suite à plusieurs travaux [AKK95, FK96], Frieze et Kannan [FK99] ont montré comment il était possible d'approcher tout problème d'optimisation Max-SNP pour les graphes denses en temps ne dépendant que de l'erreur relative ε. Plus précisément, cette complexité en temps pour Max-coupe est $2^{\tilde{O}(1/\varepsilon^2)}$, ce qui améliore le schéma précédemment cité [GGR96]. Indépendamment, Alon, Fischer, Krivelevich, et Szegedy [AFKS99] ont résolu le problème

[1] La notation $\tilde{O}$ signifie que les termes logarithmiques ne sont pas pris en compte.

général du test de propriété du type «∃∀» de graphes, vus comme des matrices d'adjacence, en exhibant des testeurs de complexité en temps indépendante de la taille du graphe. Ils ont aussi donné un contre exemple de ce résultat pour une propriété du type «∀∃».

Pour terminer, le test de propriété a été adapté aux langages. La distance entre deux mots de même longueur est ici définie comme la fraction des lettres dont ils diffèrent. Etant donné un langage, le problème est alors de décider si un mot donné de taille quelconque est proche d'un mot du langage. Alon, Krivelevich, Newman, et Szegedy [AKNS99] ont montré qu'il était possible de résoudre ce problème pour tous les langages réguliers en n'examinant qu'un nombre constant de lettres du mot testé. Par contre, ils ont montré que pour certains langages hors contexte, il est nécessaire d'examiner un nombre de lettres proportionnel à la taille du mot.

3.4. Testeur ponctuel. La notion de testeur de résultat à été étendue récemment à celle de testeur ponctuel (spot-checker) par Ergün, Kannan, Ravi Kumar, Rubinfeld, et Viswanathan [EKK$^+$98]. Un *testeur ponctuel* vérifie si un programme P calcule bien une fonction f en une entrée fixée x de la manière suivante : si $P(x) = f(x)$ alors le testeur ponctuel accepte avec grande probabilité ; et si la distance entre $(x, P(x))$ et $(y, f(y))$ est supérieure à un certain $\varepsilon > 0$ fixé pour toute autre entrée y, alors il rejette avec grande probabilité. Ici la distance peut être quelconque. Intuitivement, un testeur ponctuel vérifie si une paire entrée/sortie d'un programme est proche, pour une certaine distance, d'une autre paire entrée/sortie correcte.

Lorsque f représente une propriété et les entrées décrivent les fonctions à tester, alors la notion du test ponctuel généralise celle du test de propriété pour des distances quelconques. Dans ce modèle, Ergün, Kannan, Ravi Kumar, Rubinfeld, et Viswanathan [EKK$^+$98] ont donc entre autre montré comment tester efficacement la propriété d'être triée pour une liste, où la distance entre deux listes de même cardinal est le nombre minimal d'insertions et effacements à effectuer pour aller d'une liste à l'autre divisé par le nombre d'éléments de chaque liste ; ou encore comment tester les propriétés d'associativité ou de groupe sur des lois de composition interne, *i.e.* étant donné un ensemble fini D et un programme $P : D^2 \to D$, leur testeur ponctuel vérifie efficacement si P coïncide avec une loi de composition interne $\circ$ sur D, sauf en une faible fraction de D^2, telle que D muni de $\circ$ possède une structure de groupe.

3.5. Auto-test approché. Supposer qu'un programme travaille sur un domaine clos par addition et effectue un calcul exact sont des hypothèses très fortes. Afin de capturer la nature réelle du calcul numérique, d'autres domaines dit rationnels, ainsi que la notion de calcul approché avec erreur absolue, ont été introduits. Nous appelons domaines rationnels les ensembles finis de la forme $D_{n,s} \stackrel{\text{déf}}{=} \{\frac{i}{s} : i \in \mathbb{Z}, |i| \leq n\}$, pour certains $n \in \mathbb{N}$ et $s \in \mathbb{N}^*$, qui sont une représentation discrète des intervalles réels $[-\frac{n}{s}, \frac{n}{s}]$ avec précision $\frac{1}{s}$. Ces notions ont amené le développement d'auto-testeurs approchés [GLR$^+$91, ABCG93] tolérant des résultats avec une erreur absolue, d'auto-testeurs pour des fonctions définies sur des domaines rationnels [Lip91], et enfin d'auto-testeurs considérant ces deux aspects simultanément [EKR96].

Dans ce dernier cas, le test approché de linéarité consiste à vérifier que $|P(x+y) - P(x) - P(y)| \leq \varepsilon$, pour des éléments x, y pris uniformément aléatoirement dans $D_{n,s}$, et un réel fixé $\varepsilon > 0$. L'analyse de ce test est très similaire à sa variante exacte (*cf.* Paragraphe 3.1) à quelques modifications près. L'opérateur majorité dans la définition de la fonction corrigée g n'est plus adapté. Effectivement, le programme P peut être approximativement partout proche d'une fonction linéaire sans qu'aucun des votes $(P(x + y) - P(y))$ ne coïncident, pour x fixé et y variant dans $D_{n,s}$. Un bon choix consiste à remplacer l'opérateur majorité par l'opérateur médiane. Alors g n'est plus qu'approximativement proche de P sur la plupart des entrées, et qu'approximativement linéaire. Une seconde étape est alors nécessaire pour montrer qu'une fonction satisfaisant $|f(x+y) - f(x) - f(y)| \leq \varepsilon$, pour tout $x, y \in D_{n,s}$, est approximativement proche d'une autre fonction parfaitement linéaire. Cette étape bien connue en mathématiques

consiste à prouver la stabilité locale de l'équation de linéarité pour les termes d'erreur absolue. Lorsque la fonction est définie sur tout un groupe comme $\mathbb{Z}$, il s'agit du problème de la stabilité de l'équation de linéarité plus connue sous le nom de stabilité de Hyers-Ulam. Ce problème de stabilité est dû à Ulam et a été résolu pour la première fois pour des termes d'erreur absolue par Hyers en 1941 [Hye41]. Pour une synthèse de l'étude de la stabilité de Hyers-Ulam, consulter les travaux de Forti [For95], et de Hyers et Rassias [HR92].

3.6. Test quantique. L'historique du test en informatique quantique est succincte et, à ce jour, relativement éloignée des concepts de l'auto-test. Citons cependant trois travaux connectés à notre problématique et représentatifs de l'état de l'art en matière de test quantique.

Chuand et Nielsen [CN97] et Poyatos, Cirac, et Zoller [PCZ97] ont donné des procédures expérimentales pour déterminer certaines propriétés de boîtes noires quantiques. Cependant ces procédures utilisent implicitement un dispositif quantique ayant déjà été testé et caractérisé, et ne peuvent donc pas être utilisées dans une situation d'auto-test.

L'idée de l'auto-test quantique est sous-jacente dans les travaux d'Adleman, Demarrais, et Huand [ADH97]. Ils ont montré comment certaines machines de Turing quantiques, dont l'unique composante quantique consistait en une rotation fixée, pouvaient déterminer elles-mêmes l'angle de cette rotation sans aucune aide extérieure. L'inconvénient de ce test est qu'il repose sur une hypothèse très forte concernant la structure de la machine.

Dans le contexte de la cryptographie quantique, Mayers et Yao [MY98] ont aussi construit une série de tests permettant de décider si une source de photons était utilisable dans le protocole de distribution de clefs secrètes de Bennett et Brassard [BB84] sans aucune connaissance *a priori* de la source.

4. Contribution

Notre contribution est de deux types. Tout d'abord nous introduisons ci-dessous un nouveau modèle d'auto-test permettant non seulement le test de programme dans un sens plus général, mais aussi le test de toute machine ou objet sans aucune restriction. Nous validons ce modèle pour différentes notions de calcul approché, et pour le calcul quantique.

4.1. A l'extension du modèle. Nous initions la notion d'auto-test pour une machine, ou objet, quelconque. Cette approche nous permettra d'unifier différentes situations d'auto-test. L'auto-testeur pour un objet d'une classe donnée, est une machine de Turing probabiliste à oracle. L'oracle est construit à partir de l'objet à tester selon une *interface* simulant les expériences dictées par l'auto-testeur et retournant leurs résultats. Intuitivement l'interface doit non seulement être plus simple et efficace que l'objet testé, mais doit aussi être complètement fiable. De plus l'objet testé est une boîte noire dont le contenu est invisible. L'oracle obtenu peut être contrairement à l'accoutumée probabiliste. Si l'objet correspond à une machine probabiliste ou quantique, alors l'observation du résultat d'une expérience ne peut être que probabiliste. Par contre si l'objet est une machine déterministe, l'oracle l'est aussi.

Définissons plus formellement ces notions en posant $\mathcal{D}(R)$ l'ensemble des distributions de probabilité sur un ensemble R.

Définition. Soient Q, R deux ensembles. Un *oracle probabiliste* O à *questions* dans Q et à *réponses* dans R est une application de $Q \to \mathcal{D}(R)$.

Lorsque R est fini, nous dirons qu'un oracle probabiliste O renvoie à la question $q \in Q$ la réponse $r \in R$ selon la probabilité définie par $O(q)$. L'interface est maintenant définie ci-dessous.

Définition. Soient $\mathcal{C}, Q, R$ trois ensembles. Une *interface* $\mathcal{O}$ sur $\mathcal{C}$ à *questions* dans Q et à *réponses* dans R est une application de $\mathcal{C}$ dans l'ensemble des oracles probabilistes à questions dans Q et réponses dans R.

L'oracle probabiliste associé à un élément $f \in \mathcal{C}$ par une interface $\mathcal{O}$ sera noté $\mathcal{O}[f]$.

La notion d'auto-testeur est basée sur un critère de qualité entre l'objet voulu et l'objet testé. Afin de ne pas perdre en généralité, l'auto-testeur sera défini pour des critères de qualité les moins restrictifs possibles que sont les pseudo-distances.

Définition. Une *pseudo-distance* sur D est une application $\mathsf{Dist} : D^2 \to \mathbb{R}_+$ telle que $\mathsf{Dist}(z, z) = 0$, pour tout $z \in D$.

Nous pouvons maintenant énoncer la définition générale de l'auto-testeur pour tout type d'objet. Les définitions précédentes [BLR90, GLR$^+$91, EKR96] s'en trouvent enrichies par la notion d'interface et l'utilisation des pseudo-distances. Le but étant d'obtenir une procédure de test plus simple et efficace que l'objet testé (s'il est correct), Blum et Kannan [BK89] ont tenté de donner une approche formelle de ce critère d'efficacité (*little-oh property*). Leur notion d'efficacité n'est en général valide que pour le test de programme et pour des ensembles réduits à un élément. Nous ne reprenons donc pas cette approche, et considérons l'efficacité comme un critère extérieur à cette définition qui sera précisée pour chaque auto-testeur.

Définition (Auto-testeur)**.** Soient $\mathcal{F} \subseteq \mathcal{C}$ deux ensembles, et $\mathcal{O}$ une interface sur $\mathcal{C}$. Soient Dist_1 et Dist_2 deux pseudo-distances sur $\mathcal{C}$, et $\eta_1, \eta_2 \geq 0$. Un $(\mathsf{Dist}_1, \eta_1; \mathsf{Dist}_2, \eta_2)$-*auto-testeur* de $\mathcal{F}$ sur $\mathcal{C}$ avec l'interface $\mathcal{O}$ est une machine de Turing T probabiliste à oracle telle que pour tout $f \in \mathcal{C}$ et entrée $0 < \gamma < 1$ (*paramètre de confiance*) :

– si $\mathsf{Dist}_1(f, \mathcal{F}) \leq \eta_1$, alors $T^{\mathcal{O}[f]}(\gamma)$ retourne `BON` avec probabilité supérieure à $(1 - \gamma)$;

– si $\mathsf{Dist}_2(f, \mathcal{F}) > \eta_2$, alors $T^{\mathcal{O}[f]}(\gamma)$ retourne `MAUVAIS` avec probabilité supérieure à $(1-\gamma)$.

Remarque. $T^{\mathcal{O}[f]}$ désigne la machine T avec l'oracle probabiliste $\mathcal{O}[f]$ associé à f par $\mathcal{O}$.

Nous validerons cette notion d'auto-test pour nos deux modèles de calculs : le calcul approché en général (Première partie), et le calcul quantique (Deuxième partie).

4.2. Au calcul approché. Dans ce modèle, nous testons des programmes déterministes P, non adaptatifs, et supposés calculer une fonction d'un certain ensemble $\mathcal{F}$. Le programme même pourra donc être vu comme une fonction. L'*interface triviale* est choisie pour la communication entre l'auto-testeur et le programme testé. C'est à dire qu'une question à un programme $P : D \to R$ est une entrée $x \in D$ dont la réponse déterministe associée est $P(x)$.

Dans un premier temps, nous développerons une nouvelle théorie générale de l'auto-test de tels programmes construite sur la notion de test approché de fonction (Chapitre 1). Puis, nous envisagerons plusieurs situations de calcul approché généralisant celles précédemment envisagées dans la littérature. Nous considérerons d'une part les termes d'erreur de calcul ne dépendant que de l'entrée, et d'autre part les termes d'erreur de calcul relatifs, *i.e.* proportionnels à la norme du résultat du calcul.

Pour les premiers termes, nous construirons des auto-testeurs pour les fonctions linéaires (Chapitre 2) et plus généralement pour les fonctions polynomiales de degré borné (Chapitre 3). Ces auto-testeurs ont été préalablement construits dans un travail en commun avec Kiwi et Santha [KMS99]. Ces fonctions sont à valeurs réelles et définies sur des domaines de la forme $D_n \stackrel{\text{déf}}{=} \{i \in \mathbb{Z} : |i| \leq n\}$. Pour la clarté de notre discussion, nous avons choisi ces domaines particuliers qui capturent la notion des domaines rationnels. En effet puisque $D_{n,s} = \frac{1}{s} D_n$, tous nos résultats se généralisent à ces domaines.

Enfin, nous construirons pour les termes d'erreur relatifs, des auto-testeurs pour les fonctions multilinéaires définies sur des domaines de la forme $(D_n)^d$ et à valeurs réelles (Chapitre 4). Cette construction exposée dans un précédent travail [Mag00] est surprenante et répond à certaines questions ouvertes [KMS99, Sec. 5].

Par les travaux de cette partie, nous espérons contribuer à rendre les auto-testeurs plus adaptés aux situations du calcul numérique en général.

4.3. Au calcul quantique. Après avoir défini les notions nécessaires de mécanique quantique (Chapitre 5), nous suivrons une structure analogue à celle du calcul approché. Nous commencerons par développer une théorie originale de l'auto-test des fonctions probabilistes caractérisées par des équations probabilistes (Chapitre 6). Cette méthodologie sera ensuite appliquée aux oracles probabilistes associés aux portes quantiques par l'*interface quantique* maintenant décrite. Intuitivement l'interface quantique est un expérimentateur qui réalise des expériences simples. Ces expériences consistent à construire un circuit dont les composants sont uniquement ceux à tester, puis de mettre en entrée du circuit un état classique donné, et enfin d'observer la sortie du circuit selon la mesure de von Neumann dans la base de calcul. Plus formellement, étant donnés un m-uplet de CPSO $(\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m)$, une question à l'oracle probabiliste $\mathcal{O}[\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m]$ associé à $(\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m)$ par l'interface quantique est, par définition, de la forme (C, w). La quantité w représente un mot de $\{0,1\}^n$ pour un certain entier $n \geq 1$. L'objet C est une description d'un circuit quantique agissant sur n qubits et paramétré par les portes correspondant aux CPSO $\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m$. Le CPSO correspondant au circuit associé est noté $C(\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m)$. A cette question, l'oracle probabiliste $\mathcal{O}[\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m]$ renvoie, par définition, la distribution de probabilité associée à la mesure de von Neumann dans la base de calcul de la sortie du circuit $C(\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m)$ sur l'entrée (classique) w. Un tel oracle probabiliste sera appelé *oracle quantique*, et l'auto-testeur pour ces oracles sera nommé *auto-testeur quantique*.

Après avoir caractérisé plusieurs ensembles de portes quantiques (Chapitre 7), nous élaborerons toute une série d'auto-testeurs quantiques (Chapitre 8). Un de ces auto-testeurs sera pour la famille de portes universelle et tolérante à l'erreur exhibée par Boykin, Mor, Pulver, Roychowdhury, et Vatan [BMP$^+$99]. Etant donné un triplet de portes quantiques, nous pouvons décider avec grande probabilité s'il est suffisamment fiable pour être utilisé comme *boîte à outils*, *i.e.* pour construire (approximativement) n'importe quel circuit quantique. Ces caractérisations et auto-testeurs quantiques ont précédemment été exposés dans un travail en commun avec Dam, Mosca, et Santha [DMMS99].

Ces résultats sont à la fois motivants théoriquement car ils montrent la possibilité de tester classiquement des processus quantiques, et intéressants pratiquement car ils constituent la première batterie de tests classiques permettant d'estimer la fiabilité de portes quantiques sans aucune hypothèse *a priori*.

Première partie

Auto-test pour le calcul approché

CHAPITRE 1

Préliminaires

Dans cette partie, nous considérons le problème de l'auto-test de programme pour le calcul approché. Dans ce contexte, tout programme P est considéré déterministe et associé à la fonction qu'il calcule $x \mapsto P(x)$. Cette fonction est l'oracle déterministe associé au programme par l'interface triviale : la réponse à la question x est $P(x)$.

1. Distance seuil

Rappelons que dans l'introduction, nous avons défini de manière générale la notion de correction d'un calcul pour un terme d'erreur de calcul fixé. Ces termes d'erreur de calcul induisent une pseudo-distance naturelle définissant la proportion des entrées sur lesquelles un programme est incorrect. Avec de tels termes d'erreur de calcul, il est possible de définir la distance seuil.

Définition. Soient D est un ensemble fini, et (R, d) un ensemble métrique. Soient $f, g : D \to R$ deux fonctions, et $\varepsilon : D \times R \to \mathbb{R}_+$ un terme d'erreur de calcul. Alors pour tout réel $\varepsilon \geq 0$, la *ε-distance seuil* de P par rapport à f sur D est définie et notée par

$$\boxed{\varepsilon\text{-}\,\mathsf{Dist}_D(P, f) \stackrel{\text{déf}}{=} \Pr_{x \in D}\left[d(P(x), f(x)) > \varepsilon(x, f(x))\right].}$$

Cette probabilité sera considérée dans la suite uniforme, mais il serait tout à fait possible d'envisager une autre probabilité.

2. Test : robustesse et continuité

Dans la littérature, ces notions ne sont décrites que pour des situations particulières. Nous faisons ici l'effort d'unifier différentes approches dans le cadre le plus général possible. Nous reprenons la démarche de Kiwi [Kiw96] pour l'auto-test de programme en calcul exact, qui consiste à extraire de l'objet informatique qu'est l'auto-testeur, celui mathématique qu'est le test de fonction. Ainsi sont séparés les problèmes algorithmiques de ceux mathématiques.

Kiwi définit un *test exact* comme un triplet $(\mathcal{C}, \mathcal{T}, \mu)$, où $\mathcal{C}$ est un ensemble de fonctions, $\mathcal{T}$ un ensemble d'applications de $\mathcal{C}$ dans l'ensemble $\{\texttt{BON}, \texttt{MAUVAIS}\}$, et μ une distribution de probabilité sur $\mathcal{T}$. Pour nous, la distribution de probabilité μ sera toujours uniforme. Cependant, dans le formalisme qui suit nous laissons indéterminé ce choix. Tester une fonction $f \in \mathcal{C}$ selon le test exact $(\mathcal{C}, \mathcal{T}, \mu)$ signifie effectuer les étapes suivantes :

Test 1.1.

Test-exact(f)
1. Tirer selon μ un élément $t \in \mathcal{T}$.
2. Rejeter si $t(f) = \texttt{MAUVAIS}$.

Nous étendons cette définition aux tests approchés de fonction.

Définition. Un *test approché* est un triplet $(\mathcal{C}, \mathcal{T}, \mu)$, où $\mathcal{C}$ est un ensemble de fonctions, $\mathcal{T}$ un ensemble d'applications de $\mathcal{C}$ dans $\mathbb{R}_+$, et μ une distribution de probabilité sur $\mathcal{T}$.

Un test approché requiert aussi un *terme d'erreur de test*, soit une fonction $\beta : \mathcal{T} \times \mathcal{C} \to \mathbb{R}_+$. Alors tester une fonction $f \in \mathcal{C}$ selon un test approché $(\mathcal{C}, \mathcal{T}, \mu)$ et un terme d'erreur de test $\beta : \mathcal{T} \times \mathcal{C} \to \mathbb{R}_+$ signifie :

Test 1.2.

Test-approché(f, β)
1. Tirer selon μ un élément $t \in \mathcal{T}$.
2. Rejeter si $t(f) > \beta(t, f)$.

La *probabilité de rejet* d'une fonction $f \in \mathcal{C}$ au test approché $(\mathcal{C}, \mathcal{T}, \mu)$ pour le terme d'erreur de test $\beta : \mathcal{T} \times \mathcal{C} \to \mathbb{R}_+$ est notée et définie par

$$\beta\text{- }\mathsf{Rej}(f) \stackrel{\text{déf}}{=} \Pr_{t \in_\mu \mathcal{T}} [t(f) > \beta(t, f)] .$$

Une fonction $f \in \mathcal{C}$ *β-satisfait* (resp. *satisfait*) un test approché $(\mathcal{C}, \mathcal{T}, \mu)$, si $\beta\text{- }\mathsf{Rej}(f) = 0$ (resp. $0\text{- }\mathsf{Rej}(f) = 0$, *i.e.* β est la fonction nulle). Les éléments de $\mathcal{C}$ satisfaisant le test approché $(\mathcal{C}, \mathcal{T}, \mu)$ forment alors l'ensemble *caractérisé* par ce test.

La qualité du test $(\mathcal{C}, \mathcal{T}, \mu)$ pour l'ensemble $\mathcal{F}$ qu'il caractérise s'exprime en terme de robustesse et de continuité. La robustesse a été introduite la première fois par Rubinfeld et Sudan [RS92], et étudiée en calcul exact pour des tests construits sur des équations fonctionnelles par Rubinfeld [Rub94]. Nous étendons cette notion aux tests approchés et introduisons de plus la notion de continuité. Cette dernière existait implicitement mais n'était pas formalisée. Son introduction permet de simplifier l'étude des tests. Afin de ne pas perdre en généralité, nous définissons ces deux notions pour les pseudo-distances car seule l'hypothèse $\mathsf{Dist}(z, z) = 0$ est nécessaire dans notre modèle.

Définition. Le test $(\mathcal{C}, \mathcal{T}, \mu)$, caractérisant $\mathcal{F}$, est (η, δ)-*continue* pour le terme d'erreur de test β et la pseudo-distance Dist, si

$$\forall f \in \mathcal{C}, \qquad \mathsf{Dist}(f, \mathcal{F}) \leq \eta \implies \beta\text{- }\mathsf{Rej}(f) \leq \delta.$$

Définition. Le test $(\mathcal{C}, \mathcal{T}, \mu)$, caractérisant $\mathcal{F}$, est (η, δ)-*robuste* pour le terme d'erreur de test β et la pseudo-distance Dist, si

$$\forall f \in \mathcal{C}, \qquad \beta\text{- }\mathsf{Rej}(f) \leq \delta \implies \mathsf{Dist}(f, \mathcal{F}) \leq \eta,$$

3. Equation fonctionnelle et test

Soit $\mathcal{C}$ un espace de fonctions à valeurs dans un espace métrique (R, d). Une équation fonctionnelle sur $\mathcal{C}$ du type

$$\forall x_1, \ldots, x_k \in D, \qquad G(f, x_1, \ldots, x_k) = 0,$$

induit naturellement un test. Pour chaque $x_1, \ldots, x_k \in D$, soit l'application $t_{x_1,\ldots,x_k} : \mathcal{C} \to \mathbb{R}_+$ définie par

$$t_{x_1,\ldots,x_k}(f) \stackrel{\text{déf}}{=} d(G(f, x_1, \ldots, x_k), 0).$$

Alors ce test est formellement défini par $(\mathcal{C}, \mathcal{T}, \mu)$, avec $\mathcal{T} \stackrel{\text{déf}}{=} \{t_{x_1,\ldots,x_k} : x_1, \ldots, x_k \in D\}$ et μ une distribution de probabilité sur $\mathcal{T}$, *i.e.* sur D^k, que nous prendrons uniforme. Intuitivement, ce test consiste à vérifier sur des valeurs $x_1, \ldots, x_k$ prises uniformément aléatoirement dans D que cette équation est satisfaite, pour le calcul exact, ou approximativement satisfaite, pour le calcul approché. Encore une fois, il est tout à fait possible d'envisager une autre distribution de probabilité. Les termes d'erreur de test considérés dépendent alors en général uniquement des entrées $x_1, \ldots, x_k$ et non de la fonction testée.

Test 1.3.

Test-Equation-fonctionnelle(f, β)
1. Tirer aléatoirement $x_1, \ldots, x_k \in D$.
2. Rejeter si $\|G(f, x_1, \ldots, x_k)\| > \beta(x_1, \ldots, x_k)$.

Rubinfeld [Rub94] a étudié la robustesse de ces tests particuliers dans le cadre du calcul exact, puis avec Ergün et Ravi Kumar [EKR96] pour le calcul approché avec erreur absolue.

Nous emploierons aussi constamment ce parallélisme. Lorsque nous parlerons des robustesse et continuité d'une équation fonctionnelle, nous désignerons celles du test associé à cette équation fonctionnelle.

4. Un auto-testeur générique

Tout test approché $(\mathcal{C}, \mathcal{T}, \mu)$ *réalisable* qui est robuste et continu induit un auto-testeur. La notion de réalisabilité fait ici intervenir l'existence d'une machine de Turing M probabiliste à oracle implémentant le test comme suit. Rappelons avant que M^f désigne la machine M muni de l'oracle f avec l'interface triviale.

Définition. Une machine de Turing M probabiliste à oracle *réalise* le test $(\mathcal{C}, \mathcal{T}, \mu)$ pour le terme d'erreur de test β si

$$\forall f \in \mathcal{C}, \qquad \Pr\left[M^f \text{ retourne } \texttt{BON}\right] = \beta\text{-}\,\mathsf{Rej}(f),$$

où la probabilité du terme de gauche est prise sur les tirages uniformément aléatoires de M.

La plupart des auto-testeurs de la littérature et tout ceux que nous envisagerons en calcul approché sont construits sur le théorème suivant.

Théorème 1.1. *Soit $(\mathcal{C}, \mathcal{T}, \mu)$ un test caractérisant $\mathcal{F}$ et réalisable pour un terme d'erreur de test β par une machine M. Soit $0 < \delta < 1/2$. Si $(\mathcal{C}, \mathcal{T}, \mu)$ est $(\eta_1, \delta/2)$-continue pour β et une pseudo-distance Dist_1, et $(\eta_2, 2\delta)$-robuste pour β et une autre pseudo-distance Dist_2, alors il existe un $(\mathsf{Dist}_1, \eta_1; \mathsf{Dist}_2, \eta_2)$-auto-testeur de $\mathcal{F}$ sur $\mathcal{C}$ qui utilise, pour tout paramètre de confiance $0 < \gamma < 1$, au plus $O(\ln(1/\gamma)/\delta)$[1] itérations de M, incrémentations, comparaisons, et décalages binaires.*

Démonstration. La construction d'un tel auto-testeur utilise les bornes de Chernoff. L'auto-testeur T consiste à répéter N fois M sur le programme à tester P. Après N tests, T calcule la fraction *err* des tests ayant été rejetés. Si $err > \delta$, alors T retourne `MAUVAIS`, et `BON` sinon. Pour faciliter cette estimation, N est choisi égal à une puissance de 2.

N est choisi suffisamment grand tel que si $\beta\text{-}\,\mathsf{Rej}(P) \leq \delta/2$ (resp. $\mathsf{Rej}(P) > 2\delta$), alors $err \leq \delta$ (resp. $err > \delta$) avec probabilité supérieure à $1-\gamma$. Les bornes de Chernoff montrent alors qu'il suffit de choisir N tel que $N \geq 16\ln(2/\gamma)/\delta$ (voir par exemple [McD98, Th. 2.3] ou [BLR93, Cor. 16]). Un tel $N = O(\ln(1/\gamma)/\delta)$ peut être calculé efficacement, de sorte que la complexité totale de l'auto-testeur consiste en au plus N itérations de M, incrémentations, comparaisons, et décalages binaires. De plus, la valeur N ne dépend pas du test mais uniquement de δ et γ.

Montrons que T convient. Supposons d'abord que $\mathsf{Dist}_1(P, \mathcal{F}) \leq \eta_1$. La continuité du test implique que $\beta\text{-}\,\mathsf{Rej}(P) \leq \delta/2$, et donc avec probabilité supérieure à $(1-\gamma)$, la machine $T^P(\gamma)$ calcule $err \leq \delta$ et retourne `BON`.

Supposons maintenant que $\mathsf{Dist}_2(P, \mathcal{F}) > \eta_2$. La contraposée de la robustesse entraîne que $\beta\text{-}\,\mathsf{Rej}(P) > 2\delta$. Donc avec probabilité supérieure à $(1-\gamma)$, la machine $T^P(\gamma)$ calcule $err > \delta$ et retourne `MAUVAIS`. ∎

[1] Ce nombre ne dépend pas du test mais bien uniquement de δ et γ.

5. Robustesse : robustesse approchée et stabilité

Soit $(\mathcal{C}, \mathcal{T}, \mu)$ un test caractérisant un ensemble de fonctions $\mathcal{F}$. Afin de construire un auto-testeur basé sur ce test pour une pseudo-distance donnée, il suffit de trouver un bon terme d'erreur de test rendant le test robuste et continu. Si la continuité ne pose en général pas de problèmes majeurs, la robustesse est elle divisée en deux sous-étapes, la robustesse approchée et la stabilité. Ce découpage est inspiré des travaux d'Ergün, Ravi Kumar, et Rubinfeld [EKR96] pour le calcul approché avec erreur absolue.

Si la robustesse consiste à montrer qu'une fonction ayant une faible probabilité de rejet reste proche d'une fonction satisfaisant le test, la robustesse approchée montre uniquement qu'une telle fonction reste proche d'une fonction β-satisfaisant le test. Puis la stabilité prend le relais en montrant qu'une telle fonction est nécessairement proche d'une autre satisfaisant exactement le test.

La stabilité est en fait une théorie très développée en mathématiques, notamment lorsqu'il s'agit d'équation fonctionnelle. Justement les tests de cette partie sont tous basés sur de telles équations, et plus particulièrement sur l'équation de linéarité et de ses dérivées.

6. Stabilité de l'équation de linéarité

La stabilité de l'équation de linéarité, ou la stabilité de Hyers-Ulam, a été largement étudiée pour les fonctions définies sur un semi-groupe. Hyers [Hye41], motivé par une question d'Ulam, a initié en 1941 une série d'articles. Ulam se demandait si une fonction vérifiant approximativement l'équation de linéarité restait proche ou non d'une fonction la vérifiant exactement. Hyers répond positivement à cette question lorsque l'équation est vérifiée à un terme d'erreur constant près, *i.e.* avec erreur absolue. Beaucoup d'autres réponses positives pour d'autres contextes d'approximation ont été depuis apportés. Pour une discussion des ces résultats on pourra consulter les travaux de Hyers et Rassias [HR92], ou de Forti [For95]. Pour illustrer notre discussion citons un théorème de Rassias [Ras78] qui considère des termes d'erreur de calcul ne dépendant que de l'entrée x et proportionnels à $\|x\|^p$, pour un réel $0 \leq p < 1$ fixé.

Théorème 1.2 ([Ras78])**.** *Soient E_1 un semi-groupe normé et E_2 un espace de Banach. Soit $h : E_1 \to E_2$ une application satisfaisant pour deux réels $\theta \geq 0$ et $0 \leq p < 1$:*

$$\forall x, y \in E_1, \qquad \|h(x+y) - h(x) - h(y)\| \leq \theta \operatorname{Max}\{\|x\|^p, \|y\|^p\}.$$

Alors la fonction $l : E_1 \to E_2$ définie en $x \in E_1$ par

$$l(x) \stackrel{\text{déf}}{=} \lim_{m \to \infty} h(2^m x)/2^m,$$

est bien définie, linéaire et telle que

$$\forall x \in E_1, \qquad \|h(x) - l(x)\| \leq \frac{1}{2 - 2^p} \theta \|x\|^p.$$

Remarquons que terme d'erreur de calcul et terme d'erreur de test sont ici tous les deux construits sur la fonction $\beta(x) \stackrel{\text{déf}}{=} \theta\|x\|^p$, généralisée à deux arguments par $\beta(x, y) \stackrel{\text{déf}}{=} \operatorname{Max}\{\beta(x), \beta(y)\}$. En ce sens, le terme d'erreur de calcul est similaire au terme d'erreur de test intervenant dans l'équation de linéarité approchée. Ceci est dû à la structure particulière de cette équation dans laquelle se propagent linéairement les erreurs de calcul. Les notions de termes d'erreur de calcul et de test s'en trouvent naturellement confondues.

Bien que ces résultats ne s'appliquent en général que pour des fonctions définies sur des semi-groupes, il est possible de les utiliser pour des fonctions définies sur des sous-ensembles de semi-groupes. Une étape intermédiaire consiste alors à étendre une fonction vérifiant localement l'équation de linéarité approchée en une fonction définie sur tout le semi-groupe et vérifiant toujours l'équation de linéarité approchée. La fonction étendue satisfait alors les

hypothèses précédentes, et est donc proche d'une fonction linéaire partout. Il en est de même localement pour la fonction initiale. Initialement basée sur un argument de Skof [Sko83], nous avons développé cette technique pour notre cadre [KMS99].

Mais revenons un instant sur le Théorème 1.2. Lorsque $p = 1$ cet énoncé n'est plus valide. Hyers et Semrl [HS92] ont en effet montré que la fonction $f : \mathbb{R} \to \mathbb{R}$ définie en $x \in \mathbb{R}$ par $f(x) \stackrel{\text{déf}}{=} \theta x \log_2(1 + |x|)$, pour un certain réel $\theta > 0$, satisfaisait $|f(x + y) - f(x) - f(y)| \leq 2\theta \operatorname{Max}\{|x|, |y|\}$, pour tout $x, y \in \mathbb{R}$, mais n'était proche d'aucune fonction linéaire. Ce même contre-exemple reste pertinent pour les sous-domaines D_n. Il n'est donc pas possible de garder un terme d'erreur linéaire dans l'équation de linéarité.

Dans un premier temps, nous avons utilisé l'équation de linéarité pour construire un auto-testeur pour les fonctions linéaires, en considérant des termes d'erreur de calcul β dits valides pour lesquels l'équation reste stable. Ces termes d'erreur sont dans la classe $\Theta(|x|^p)$, pour un certain réel fixé $0 \leq p < 1$. Ils induisent des termes d'erreur de calcul non relatifs mais modélisant un fait important souvent ignoré dans la littérature. La plupart des erreurs d'approximation en calcul numérique dépendent de la taille des entrées. Ces résultats sur l'équation de linéarité servent aussi à définir un auto-testeur pour les fonctions polynomiales de degré borné. Rappelons que le cas de l'erreur absolue, $p = 0$, a été préalablement étudié par Ergün, Ravi Kumar, et Rubinfeld [EKR96] mais sans utiliser les techniques élégantes développées dans la théorie de la stabilité de Hyers-Ulam. Nous développerons cette étude [KMS99] au Chapitre 2 pour les fonctions linéaires et au Chapitre 3 pour les polynômes.

Au Chapitre 4 une autre approche [Mag00] est détaillée. Elle repose cette fois-ci sur une nouvelle caractérisation des fonctions linéaires qui reste stable pour les termes d'erreur linéaires. Il s'agit du cas $p = 1$ du paragraphe précédent. Le grand intérêt de cette méthode est qu'elle permet de construire un auto-testeur avec erreur relative pour les fonctions multilinéaires.

CHAPITRE 2

Fonctions linéaires

L'objectif de ce chapitre est de montrer que le test induit par l'équation de linéarité est robuste pour certains termes d'erreur dits valides (Corollaire 2.1). Ces termes d'erreur définiront simultanément des termes d'erreur de calcul et de test. De là pourra être construit un auto-testeur approché pour l'ensemble $\mathcal{L}$ des fonctions linéaires de $\mathbb{Z} \to \mathbb{R}$ (Théorème 2.3). La preuve de la robustesse sera découpée en deux parties : la stabilité (Théorème 2.1) et la robustesse approchée (Théorème 2.2).

1. Test et termes d'erreur valides

Dans toute ce chapitre nous étudierons le test associé à l'équation de linéarité sur D_{4n} :

$$\forall x, y \in D_{4n}, \qquad f(x+y) - f(x) - f(y) = 0.$$

Ce test caractérise sur D_{8n} l'ensemble des fonctions linéaires $\mathcal{L}$. Ce test est robuste pour des termes d'erreur de calcul et d'erreur construits à partir d'autres termes d'erreur dits valides.

Définition. $\beta : \mathbb{Z} \to \mathbb{R}_+$ est un *terme d'erreur valide de degré* $p \in \mathbb{R}_+$ si β est paire, croissante sur $\mathbb{N}$, et satisfait :

$$\forall \lambda \in [1, +\infty[,\ \forall s \in \mathbb{Z}, \qquad \lambda s \in \mathbb{Z} \implies \beta(\lambda s) \leq \lambda^p \beta(s).$$

Pour fixer les idées, voici deux exemples de termes d'erreur valides :

Exemple. Les termes d'erreur suivant sont valides de degré p :
– $\beta(s) \stackrel{\text{déf}}{=} |s|^p$,
– $\beta(s) \stackrel{\text{déf}}{=} \text{Max}\{a, |s|^p\}$, pour tout réel fixé $a \geq 0$.

Un terme d'erreur à une variable sera étendu à plusieurs variables de la manière suivante. Si $\vec{z} \in \mathbb{Z}^d$, alors

$$\beta(\vec{z}) \stackrel{\text{déf}}{=} \text{Max}\{\beta(z_1), \ldots, \beta(z_d)\}.$$

Chaque terme d'erreur valide β défini un terme d'erreur de calcul ne dépendant que de l'entrée, et donc la distance seuil β-Dist_D peut lui être associée. Il défini aussi un terme d'erreur de test. Le test approché de linéarité suivant lui est alors associé :

Test 2.1.

Test-linéarité(P, β)
1. Tirer aléatoirement $x, y \in D_{4n}$.
2. Rejeter si $|P(x+y) - P(x) - P(y)| > \beta(x, y)$.

2. Stabilité

Cette section est destinée à la stabilité locale de l'équation de linéarité pour les termes d'erreur valides de degré $0 \leq p < 1$. Le résultat suivant va y être démontré.

Théorème 2.1. *Soit β un terme d'erreur valide de degré $0 \leq p < 1$. Soit $g : D_{2n} \to \mathbb{R}$ telle que*

$$\forall x, y \in D_n, \qquad |g(x+y) - g(x) - g(y)| \leq \beta(x, y).$$

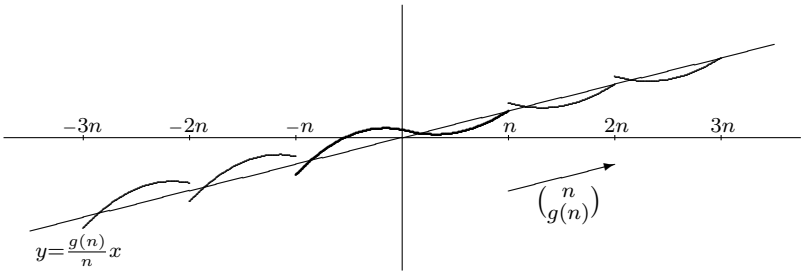

FIG. 2.1. Construction de $h : \mathbb{Z} \to \mathbb{R}$ à partir de $g|_{D_n}$.

Alors la fonction linéaire $l : D_n \to \mathbb{R}$ *définie par* $l(n) \stackrel{\text{déf}}{=} g(n)$ *satisfait*

$$\forall x \in D_n, \qquad |g(x) - l(x)| \leq \frac{1+2^p}{2-2^p}\beta(x).$$

Ce théorème est une contribution à l'étude de la stabilité locale de l'équation de linéarité qui est généralement établie pour des fonctions définies sur des groupes ou plus généralement sur des semi-groupes. La stabilité locale et la stabilité sur un semi-groupe sont connectées par le lemme suivant basé sur un argument dû à Skof [Sko83]. Il permet de prolonger une fonction satisfaisant localement l'équation de linéarité approchée, en une fonction la vérifiant partout (Figure 2.1). Alors un théorème similaire au Théorème 1.2 permet de conclure.

Lemme 2.1. *Soit* β *un terme d'erreur valide de degré* $p \geq 0$. *Soit* $g : D_{2n} \to \mathbb{R}$ *telle que*

$$\forall x, y \in D_n, \qquad |g(x+y) - g(x) - g(y)| \leq \beta(x, y).$$

Alors l'application $h : \mathbb{Z} \to \mathbb{R}$ *définie en* $x \in \mathbb{Z}$ *par*

$$h(x) \stackrel{\text{déf}}{=} \begin{cases} g(x) & \text{si } x \in D_n, \\ h(x-n) + g(n) & \text{si } x > n, \\ h(x+n) - g(n) & \text{si } x < -n; \end{cases}$$

satisfait

$$\forall x, y \in \mathbb{Z}, \qquad |h(x+y) - h(x) - h(y)| \leq (1+2^p)\beta(x, y).$$

En fait, une version plus faible de ce lemme est suffisante pour établir la stabilité locale. Puisque la limite $l(x) \stackrel{\text{déf}}{=} \lim_{m\to\infty} h(2^m x)/2^m$ intervenant au Théorème 1.2 existe toujours et définie nécessairement une fonction linéaire, il suffit de prouver que la fonction l reste proche de h. Ceci est alors garanti par le fait que pour tout $x \in D_n$, $|h(2x) - 2h(x)| \leq (1+2^p)\beta(x)$. Commençons par montrer que l'extension h satisfait cette propriété de doublement.

Lemme 2.2. *Soit* β *un terme d'erreur valide de degré* $p \geq 0$. *Soit* $g : D_{2n} \to \mathbb{R}$ *telle que*

$$\forall x, y \in D_n, \qquad |g(x+y) - g(x) - g(y)| \leq \beta(x, y).$$

Alors l'application $h : \mathbb{Z} \to \mathbb{R}$ *définie en* $x \in \mathbb{Z}$ *par*

$$h(x) \stackrel{\text{déf}}{=} \begin{cases} g(x) & \text{si } x \in D_n, \\ h(x-n) + g(n) & \text{si } x > n, \\ h(x+n) - g(n) & \text{si } x < -n; \end{cases}$$

satisfait

$$\forall x \in \mathbb{Z}, \qquad |h(2x) - 2h(x)| \leq (1+2^p)\beta(x).$$

Démonstration. Par définition de h, il suffit de montrer la propriété pour $x \in D_n$. Effectivement, si $x > n$, $h(2x)-2h(x) = h(2(x-n))-2h(x-n)$, et si $x < -n$, $h(2x)-2h(x) = h(2(x+n)) - 2h(x+n)$.

Supposons d'abord que $x \in D_{n/2}$. Alors $h(2x) = g(2x)$ et $h(x) = g(x)$, donc par hypothèse

$$|h(2x) - 2h(x)| \leq \beta(x).$$

Si $n/2 < x \leq n$, alors $h(2x) = g(2x-n) + g(n)$ et $h(x) = g(x)$. La propriété vient alors par une suite de manipulations élémentaires :

$$\begin{aligned} |h(2x) - 2h(x)| &= |g(2x-n) + g(n) - 2g(x)| \\ &\leq |g(2x) - g(2x-n) - g(n)| + |g(2x) - 2g(x)| \\ &\leq \beta(2x-n, n) + \beta(x), \end{aligned}$$

car g est approximativement linéaire sur D_n par hypothèse. Mais β est un terme d'erreur valide de degré p, donc puisque $0 \leq 2x-n \leq n \leq 2x$, la quantité $\beta(2x-n, n)$ est bornée par $\beta(2x) \leq 2^p\beta(x)$.

Enfin lorsque $-n \leq x < -n/2$, $h(2x) = h(x+n) - g(n)$ et $h(x) = g(x)$, une manipulation analogue permet de conclure :

$$\begin{aligned} |h(2x) - 2h(x)| &= |g(2x+n) - g(n) - 2g(x)| \\ &\leq |g(2x+n) - g(x+n) - g(x)| + |g(x+n) - g(n) - g(x)| \\ &\leq \beta(x+n, x) + \beta(n, x). \end{aligned}$$

Mais $|x+n| \leq |x|$ et $|x| \leq |n| \leq |2x|$ donc le dernier terme se majore par $\beta(2x) + \beta(x) \leq (1+2^p)\beta(x)$. ∎

Si la fonction h satisfait les conclusions du lemme précédent, alors h reste effectivement proche de la limite introduite au Théorème 1.2. Ce résultat est énoncé pour des fonctions définies sur des sous-domaines de $\mathbb{Z}$ stables par multiplication par 2. Si dans ce chapitre, seul le cas du domaine $\mathbb{Z}$ est utile, le cas plus général servira à la preuve du Théorème 4.2 du Chapitre 4.

Lemme 2.3. *Soient β un terme d'erreur valide de degré $0 \leq p < 1$, et $D \subseteq \mathbb{Z}$ stable par multiplication par 2 (i.e. $x \in D \implies 2x \in D$). Soit $h : D \to \mathbb{R}$ telle que*

$$\forall x \in D, \qquad |h(2x) - h(x)| \leq \beta(x).$$

Si pour tout $x \in D$ la limite $f(x) \stackrel{\text{déf}}{=} \lim_{m\to\infty} h(2^m x)/2^m$ est bien définie, alors f satisfait

$$\forall x \in D, \qquad |f(x) - h(x)| \leq \tfrac{1}{2-2^p}\beta(x).$$

Démonstration. Cette preuve est calquée sur une étape intermédiaire de celle du Théorème 1.2. Nous l'avons généralisée et adaptée à notre situation.

Nous prouvons par récurrence sur l'entier $m \geq 1$ que pour chaque entier $x \in D$,

$$\left|\frac{h(2^m x)}{2^m} - h(x)\right| \leq \frac{1}{2}\beta(x) \sum_{i=0}^{m-1} 2^{i(p-1)}. \tag{2.1}$$

Le cas $m = 1$ de (2.1) correspond à l'hypothèse faite sur h.

Supposons que (2.1) soit satisfaite pour un entier $m \geq 1$. Alors, les hypothèses sur g et de récurrence impliquent :

$$\begin{aligned} \left|\frac{h(2^{m+1}x)}{2^{m+1}} - h(x)\right| &\leq \left|\frac{h(2x)}{2} - h(x)\right| + \frac{1}{2}\left|\frac{h(2^m \cdot 2x)}{2^m} - h(2x)\right| \\ &\leq \frac{1}{2}\beta(x) + \frac{1}{4}\beta(2x) \sum_{i=0}^{m-1} 2^{i(p-1)}. \end{aligned}$$

Mais β est un terme d'erreur valide de degré p, donc $\beta(2x) \leq 2^p\beta(x)$. L'étape $(m+1)$ de la récurrence est alors démontrée.

Donc pour tout entier $m \geq 1$, l'inégalité (2.1) est satisfaite. La preuve est alors achevée en faisant tendre $m \to \infty$. ∎

La preuve du Théorème 2.2 consiste alors simplement à mettre bout à bout les Lemmes 2.2 et 2.3.

3. Robustesse approchée

Dans cette section l'équation de linéarité est prouvée approximativement robuste. Nous utilisons une technique similaire à celle du test de linéarité [BLR90]. Rappelons qu'en calcul exact sur un domaine G à structure de groupe, l'idée est de construire, à l'aide du programme testé P, une fonction g corrigeant P et définie en chaque $x \in G$ par $g(x) \stackrel{\text{déf}}{=} \text{Maj}_{y\in G}\left(P(x+y) - P(y)\right)$, *i.e.* comme la majorité des votes $(P(x+y) - P(y))$ pour tout votant $y \in G$. Lorsque P passe le test de linéarité avec grande probabilité, alors la fonction g est linéaire et proche de P. Une étape intermédiaire consiste à montrer que la majorité est en fait une quasi-unanimité. En calcul approché sur D_n, plusieurs différences majeures apparaissent. Tout d'abord puisque le domaine n'est plus stable par addition, il convient de tester P sur un domaine plus grand : D_{4n}. Puis de part la nature du calcul approché, l'opérateur majorité n'est plus adapté. En effet, P peut être approximativement proche d'une fonction linéaire alors que peu de votes coïncident. Ergün, Ravi Kumar, et Rubinfeld [EKR96] ont montré que l'opérateur médiane défini ci-après pouvait remplacer l'opérateur majorité dans le cas de l'erreur absolue.

Définition. Soient X un ensemble fini et $f : X \to \mathbb{R}$. La *médiane des valeurs prises par f sur X* est notée $\text{Méd}_{x\in X}\left(f(x)\right)$ et définie par

$$\underset{x\in X}{\text{Méd}}\left(f(x)\right) \stackrel{\text{déf}}{=} \text{Inf}\{a \in \mathbb{R} : \Pr_{x\in X}\left[f(x) \geq a\right] \leq 1/2\}.$$

Dans le cas général du terme d'erreur valide, tous les votants ne peuvent être autorisés à voter dans la correction du programme en x. En effet de trop grands votants peuvent introduire de trop grandes erreurs pour de petits x. Toutefois lorsque x est trop petit, les éléments plus petits que x sont en proportion insuffisante pour corriger P. Notre solution prend en compte ces deux problèmes afin de définir un ensemble convenable de votants pour chaque entrée x. La robustesse approchée de l'équation de linéarité s'énonce alors comme suit.

Théorème 2.2. *Soient β un terme d'erreur valide de degré $p \geq 0$, $0 \leq \delta \leq 1$, et $\varphi : \mathbb{Z} \to \mathbb{R}$ la fonction définie en $x \in \mathbb{Z}$ par $\varphi(x) \stackrel{\text{déf}}{=} \text{Max}\{\sqrt{\delta}n, |x|\}$. Soit $P : D_{8n} \to \mathbb{R}$ tel que*

$$\Pr_{x,y\in D_{4n}}\left[|P(x+y) - P(x) - P(y)| > \beta(x,y)\right] \leq \frac{\delta}{384}.$$

Alors la fonction $g : D_{2n} \to \mathbb{R}$ définie en $x \in D_{2n}$ par

$$g(x) \stackrel{\text{déf}}{=} \underset{y\in D_{\varphi(x)}}{\text{Méd}}\left(P(x+y) - P(y)\right),$$

satisfait

$$\Pr_{x\in D_n}\left[|P(x) - g(x)| > \beta(\varphi(x))\right] \leq \frac{\delta}{24},$$

et

$$\forall x, y \in D_n, \qquad |g(x+y) - g(x) - g(y)| \leq 4(2+2^p)\beta(\varphi(x), \varphi(y)).$$

La preuve du théorème est séparée en plusieurs lemmes. Commençons par citer le plus simple que nous nommerons le *Lemme de Contraction.*

Lemme 2.4 (Contraction)**.** *Soient Ω et S deux ensembles finis tels que $S \subseteq \Omega$, et ψ une formule booléenne définie sur Ω. Alors*

$$\Pr_{x\in S}[\psi(x)] \leq \frac{|\Omega|}{|S|}\Pr_{x\in\Omega}[\psi(x)].$$

Le lemme suivant montre que g reste proche de P.

Lemme 2.5. *Sous les hypothèses du Théorème 2.2, la fonction g satisfait*

$$\Pr_{x\in D_{4n}}[|P(x)-g(x)| > \beta(\varphi(x))] \leq \frac{\delta}{96}.$$

Démonstration. Soit $P_{x,y} \stackrel{\text{déf}}{=} P(x+y)-P(x)-P(y)$. Alors la définition de g et l'inégalité de Markov entraînent

$$\begin{aligned}\Pr_{x\in D_{4n}}[|g(x)-P(x)| > \beta(\varphi(x))] &= \Pr_{x\in D_{4n}}\left[|\operatorname*{Méd}_{y\in D_{\varphi(x)}}(P_{x,y})| > \beta(\varphi(x))\right]\\ &\leq 2\Pr_{x\in D_{4n},y\in D_{\varphi(x)}}[|P_{x,y}| > \beta(\varphi(x))].\end{aligned}$$

Alors les minorations $\varphi(x) \geq \varphi(y) \geq |y|$ et $\varphi(x) \geq |x|$, puis le Lemme de Contraction 2.4 concluent la preuve par la suite de majorations :

$$\begin{aligned}&2\Pr_{x\in D_{4n},y\in D_{\varphi(x)}}[|P_{x,y}| > \beta(\varphi(x))]\\ \leq\ & 2\frac{|D_{4n}|^2}{|x,y : x\in D_{4n}, y\in D_{\varphi(x)}|}\Pr_{x,y\in D_{4n}}[|P_{x,y}| > \beta(x,y)]\\ \leq\ & \frac{4\delta}{384}\\ =\ & \frac{\delta}{96}.\end{aligned}$$

■

Afin de montrer l'additivité approchée de g, il convient d'établir que la plupart des votes intervenant dans sa définition restent proches de leur valeur médiane.

Lemme 2.6. *Sous les hypothèses du Théorème 2.2, la fonction g satisfait pour tout $c \in D_{2n}$, et $I \subseteq D_{\varphi(c)}$ tel que $|I| \geq \sqrt{\delta}n+1$,*

$$\Pr_{y\in I}[|g(c)-(P(c+y)-P(y))| > 4\beta(\varphi(c))] < 1/3.$$

Démonstration. Posons encore $P_{x,y} \stackrel{\text{déf}}{=} P(x+y)-P(x)-P(y)$. Alors la définition de g et l'inégalité de Markov induisent comme précédemment la majoration

$$\begin{aligned}&\Pr_{y\in I}[|g(c)-(P(c+y)-P(y))| > 4\beta(\varphi(c))]\\ \leq\ & 2\Pr_{y\in I,z\in D_{\varphi(c)}}[|P_{c+y,z}-P_{c+z,y}| > 4\beta(\varphi(c))]\\ \leq\ & 2\Pr_{y\in I,z\in D_{\varphi(c)}}[|P_{c+y,z}| > 2\beta(\varphi(c))] + 2\Pr_{y\in I,z\in D_{\varphi(c)}}[|P_{c+z,y}| > 2\beta(\varphi(c))].\end{aligned}$$

En observant que dans ces deux dernières probabilités la quantité $\varphi(c)$ majore $|y|$,$|z|$ et $|c|$, on obtient que $2\beta(\varphi(c))$ est plus grand que $\beta(c+y,z)$ et $\beta(c+z,y)$. Alors en utilisant le Lemme de Contraction 2.4, chacune de ces probabilités est indépendamment majorée par

$$\frac{|D_{4n}|^2}{|I|\cdot|D_{\varphi(c)}|}\Pr_{u,v\in D_{4n}}[|P_{u,v}| > \beta(u,v)].$$

La preuve se conclut donc en se souvenant que par hypothèse $\delta/384$ majore ce terme de probabilité et que $|D_{4n}|^2/(|I|\cdot|D_{\varphi(c)}|) < 32/\delta$. ■

Prouvons maintenant le Théorème 2.2.

Démonstration. Tout d'abord le Lemme 2.5 couplé au Lemme de Contraction 2.4 permet de montrer que g est proche de P comme annoncé dans le théorème.

Pour prouver l'additivité approchée de g prenons a et b deux éléments de D_n. Sans perte de généralité supposons que $|a| \leq |b|$. Si $a \geq 0$ (resp. $a < 0$) alors par le Lemme 2.6 il existe, avec probabilité non nulle, un élément $y \in \{-\sqrt{\delta}n, \ldots, 0\}$ (resp. $y \in \{0, \ldots, \sqrt{\delta}n\}$) tel que

$$\begin{aligned} |g(a) - (P(a+y) - P(y))| &\leq 4\beta(\varphi(a)), \\ |g(b) - (P(b+(a+y)) - P(a+y))| &\leq 4\beta(\varphi(b)), \\ |g(a+b) - (P(a+b+y) - P(a+b+y))| &\leq 4\beta(\varphi(a+b)). \end{aligned}$$

Or $|a| \leq |b|$ et $\beta \circ \varphi$ est un terme d'erreur valide de degré p puisque β en est un. Donc $\beta(\varphi(a)) \leq \beta(\varphi(b))$ et $\beta(\varphi(a+b)) \leq 2^p\beta(\varphi(b))$, ce qui donne finalement le résultat avec la combinaison des trois inégalités précédentes. ∎

4. Auto-tester les fonctions linéaires

Les deux sections précédentes impliquent la robustesse du test de linéarité (Test 2.1) pour des termes d'erreur valides de degré $0 \leq p < 1$. Le corollaire suivant, déduit des Théorèmes 2.2 et 2.1, précise cette robustesse. Rappelons que $\mathcal{L}$ désigne l'ensemble des fonctions linéaires, et posons $\beta\text{-}\mathsf{Rej}(P)$ la probabilité de rejet au test de linéarité sur D_{4n}, *i.e.*

$$\beta\text{-}\mathsf{Rej}(P) \stackrel{\text{déf}}{=} \Pr_{x,y \in D_{4n}}\left[|P(x+y) - P(x) - P(y)| > \beta(x,y)\right].$$

Corollaire 2.1. *Soient β un terme d'erreur valide de degré $0 \leq p < 1$, et $0 \leq \delta \leq 1$. Alors pour tout $P : D_{8n} \to \mathbb{R}$,*

$$\beta\text{-}\mathsf{Rej}(P) \leq \frac{\delta}{384} \implies (\tfrac{49}{2-2^p}\beta)\text{-}\mathsf{Dist}_{D_n}(P, \mathcal{L})) \leq \begin{cases} \dfrac{\delta}{24} & \text{si } p = 0, \\ \dfrac{25\sqrt{\delta}}{24} & \text{si } 0 < p < 1. \end{cases}$$

Démonstration. Supposons $\beta\text{-}\mathsf{Rej}(P) \leq \frac{\delta}{384}$. Alors d'après Théorème 2.2, P est proche d'une fonction g définie sur D_{2n} :

$$\Pr_{x \in D_n}\left[|P(x) - g(x)| > \beta(\varphi(x))\right] \leq \frac{\delta}{24},$$

telle que g est de plus approximativement linéaire :

$$\forall x, y \in D_n, \qquad |g(x+y) - g(x) - g(y)| \leq 4(2+2^p)\beta(\varphi(x), \varphi(y)).$$

Alors le Théorème 2.1 permet de montrer que g est proche d'une fonction linéaire l :

$$\forall x \in D_n, \qquad |g(x) - l(x)| \leq 4(2+2^p)\frac{1+2^p}{2-2^p}\beta(\varphi(x)).$$

Donc P est aussi proche de l au sens suivant :

$$\Pr_{x \in D_n}\left[|P(x) - l(x)| > \frac{49}{2-2^p}\beta(\varphi(x))\right] \leq \frac{\delta}{24}.$$

Pour conclure, observons que si $p = 0$ alors $\beta(\varphi(x)) = \beta(x)$, et si $p > 0$ alors

$$\Pr_{x \in D_n}\left[\beta(\varphi(x)) > \beta(x)\right] \leq \sqrt{\delta}.$$

∎

Le test de linéarité est aussi continue. Plus exactement :

Lemme 2.7. *Soit β un terme d'erreur valide de degré $0 \leq p \leq 1$. Alors tout $P : D_{8n} \to \mathbb{R}$ satisfait*

$$\beta\text{-}\mathsf{Rej}(P) \leq 6 \cdot (\beta/4)\text{-}\mathsf{Dist}_{D_{8n}}(P, \mathcal{L}).$$

Démonstration. Soient $\delta \stackrel{\text{déf}}{=} (\beta/4)\text{-}\mathsf{Dist}_{D_{8n}}(P, \mathcal{L})$, et $l \in \mathcal{L}$ telle que $(\beta/4)\text{-}\mathsf{Dist}_{D_{8n}}(P, l) = \delta$. Si x et y sont pris uniformément aléatoirement dans D_{4n}, alors d'après le Lemme de Contraction 2.4, avec probabilité supérieure à $1 - 6\delta$, les inégalités suivantes sont satisfaites :

$$\begin{aligned} |P(x) - l(x)| &\leq \frac{\beta(x)}{4}, \\ |P(y) - l(y)| &\leq \frac{\beta(y)}{4}, \\ |P(x+y) - l(x+y)| &\leq \frac{\beta(x+y)}{4}. \end{aligned}$$

Mais $\beta(x)+\beta(y)+\beta(x+y) \leq 4\beta(x, y)$, et donc la probabilité de rejet au test est bien inférieure à 6δ. ∎

Pour que notre auto-testeur soit simple et efficace, il faut s'assurer que β est facilement calculable. Ceci n'est à priori pas toujours le cas, d'autant plus que $\beta(x)$ est proportionnel à $|x|^p$. Or nous ne pouvons permettre à un auto-testeur d'avoir la puissance de calcul nécessaire pour faire une telle exponentiation. Un moyen de contourner cette difficulté est d'approcher ce terme d'erreur par un autre équivalent. Nous dirons que le terme d'erreur α est (λ, λ')-*équivalent* au terme d'erreur β si $\lambda\beta(x) \leq \alpha(x) \leq \lambda'\beta(x)$, pour tout entier x.

Ainsi par exemple, si calculer $\beta(x) \stackrel{\text{déf}}{=} 2^{k'}|x|^{1/2^k}$ est difficile, le calcul de

$$\alpha(x) \stackrel{\text{déf}}{=} \begin{cases} 0 & \text{si } x = 0, \\ 2^{k'+\lceil \log_2 x \rceil/2^k} & \text{sinon,} \end{cases}$$

est de complexité linéaire en x, et de plus α est $(1, 2)$-équivalent à β.

Pour construire un auto-testeur selon des distantes seuil construites avec un terme d'erreur valide β de degré $0 \leq p < 1$, il est possible d'utiliser éventuellement un terme d'erreur α équivalent à β. Ainsi la robustesse et la continuité du test de linéarité entraînent avec le Théorème 1.1 l'existence de l'auto-testeur suivant.

Théorème 2.3. *Soient β un terme d'erreur valide de degré $0 \leq p < 1$ et α un terme d'erreur (λ, λ')-équivalent à β, où $\lambda, \lambda' > 0$. Alors pour tout $0 < \delta \leq 1/2$ il existe un auto-testeur T qui satisfait sur l'entrée $0 < \gamma < 1$ et pour tout programme $P : D_{8n} \to \mathbb{R}$:*

- *si $(\frac{\lambda\beta}{4})\text{-}\mathsf{Dist}_{D_{8n}}(P, \mathcal{L}) \leq \frac{\delta}{4608}$ alors $T^P(\gamma)$ retourne* `BON` *avec probabilité supérieure à $(1 - \gamma)$,*
- *si $(\frac{49\lambda'}{2-2^p}\beta)\text{-}\mathsf{Dist}_{D_n}(P, \mathcal{L}) > \begin{cases} \frac{\delta}{12} & \text{si } p = 0, \\ \frac{3\sqrt{\delta}}{2} & \text{si } 0 < p < 1, \end{cases}$ alors $T^P(\gamma)$ retourne* `MAUVAIS` *avec probabilité supérieure à $(1 - \gamma)$,*

en utilisant au plus $O(\ln(1/\gamma)/\delta)$ appels à P, comparaisons, décalages binaires, additions, et évaluations de α.

5. Optimalité du test

Le Corollaire 2.1 montre que la distance seuil d'un programme aux fonctions linéaires est majorée, à une constante près, par la racine carrée de sa probabilité de rejet au test de linéarité. Cette racine carrée est surprenante d'autant plus qu'elle n'apparaît pas en calcul exact ou approché avec erreur absolue (cas $p = 0$). Il est donc naturel de se demander si ce

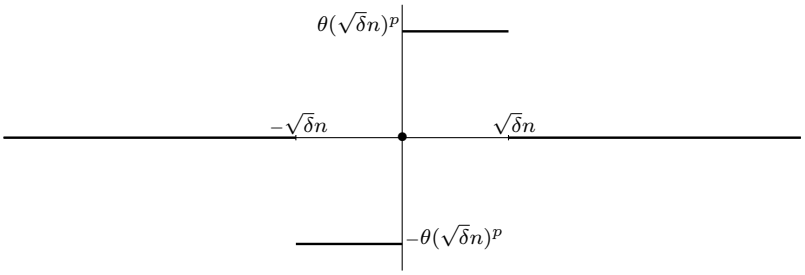

FIG. 2.2. Exemple tendu du test de linéarité.

résultat est optimal. Si aucune hypothèse supplémentaire n'est faite, le Lemme 2.8 montre qu'il l'est à une constante près. En effet, pour chaque terme d'erreur valide $\beta(x) = \theta|x|^p$ de degré $0 < p < 1$, il est possible d'expliciter une fonction P (voir Figure 2.2) pour laquelle le Corollaire 2.1 est tendu à une constante près.

Lemme 2.8. *Soient* $0 < p < 1$, $0 < \delta < 1/4$, *et* $\theta, c > 0$. *Soient* $\beta(x) = \theta|x|^p$, *et* $P : \mathbb{Z} \to \mathbb{R}$ *définie en* $x \in \mathbb{Z}$ *par*

$$P(x) \stackrel{\text{déf}}{=} \begin{cases} -\theta(\sqrt{\delta}n)^p & \text{si } -\sqrt{\delta}n \leq x < 0, \\ \theta(\sqrt{\delta}n)^p & \text{si } 0 < x \leq \sqrt{\delta}n, \\ 0 & \text{sinon.} \end{cases}$$

Alors P *satisfait pour tout* $n' \geq n$

$$\Pr_{x,y \in D_{n'}}[|P(x+y) - P(x) - P(y)| > 2\beta(x,y)] \leq \delta,$$

et de plus

$$(c\beta)\text{-}\mathsf{Dist}_{D_n}(P, \mathcal{L}) \geq \frac{\sqrt{\delta}}{2(\text{Max}\{1, 2c\})^{1/p}}.$$

Démonstration. Tout d'abord, observons que si $|x|$ ou $|y|$ est plus grand que $\sqrt{\delta}n$, alors $|P(x+y) - P(x) - P(y)| \leq 2\beta(x,y)$. Donc si $n' \geq n$, la probabilité de rejet de P au test de linéarité sur $D_{n'}$, pour le terme d'erreur 2β, est bien majorée par δ.

Soit maintenant une fonction linéaire $l \in \mathcal{L}$. Nous allons montrer par l'absurde que l est suffisamment éloignée de P sur D_n. Supposons donc que $(c\beta)\text{-}\mathsf{Dist}_{D_n}(P, l) < \frac{\sqrt{\delta}}{2(\text{Max}\{1,2c\})^{1/p}}$. Posons $d \stackrel{\text{déf}}{=} 1/(\text{Max}\{1, 2c\})^{1/p}$, donc $d \leq 1$, et $a \stackrel{\text{déf}}{=} l(1)$, *i.e.* $l(x) = ax$ pour tout x. Alors montrons que nécessairement

$$|a|n^{1-p} \leq \frac{c\theta}{(1 - d\sqrt{\delta})^{1-p}}. \tag{2.2}$$

Prenons un élément $x \in D_n$ tel que $|x| > \sqrt{\delta}n$. Alors l'écart entre P et l en x est $|P(x) - l(x)| = |0 - l(x)| = |ax|$. L'élément x contribue donc à augmenter la distance seuil entre P et l s'il satisfait $|ax| > c\beta(x)$, *i.e.* $|a||x|^{1-p} > c\theta$. Dans ce cas, tout autre élément $y \in D_n$ satisfaisant $|y| \geq |x|$ contribue aussi dans la distance seuil entre P et l. Par conséquent l'élément $x = \lceil n(1 - d\sqrt{\delta})\rceil$ $(> \sqrt{\delta}n)$ ne doit pas satisfaire $|a||x|^{1-p} > c\theta$, sinon la majoration $(c\beta)\text{-}\mathsf{Dist}_{D_n}(P, l) < d\sqrt{\delta}/2$ serait contredite. Donc nécessairement l'inégalité (2.2) est satisfaite.

L'inégalité (2.2) se réécrit en

$$|a|(\sqrt{\delta}n)^{1-p} \leq c\theta \left(\frac{\sqrt{\delta}}{1 - d\sqrt{\delta}}\right)^{1-p}.$$

Mais comme $d \leq 1$ et $\delta < 1/4$, il vient

$$|a|(\sqrt{\delta}n)^{1-p} < c\theta.$$

Alors pour tout $x \in D_n$ tel que $0 < |x| \leq d\sqrt{\delta}n$ ($\leq \sqrt{\delta}n$) on a :

$$\begin{aligned} |P(x) - l(x)| &= |\theta(\sqrt{\delta}n)^p - a|x|| \\ &\geq \theta(\sqrt{\delta}n)^p - |a||x| \\ &\geq (\theta(d)^{-p} - |a||x|^{1-p})|x|^p \\ &\geq (2c\theta - |a|(\sqrt{\delta}n)^{1-p})|x|^p \\ &> c\beta(x). \end{aligned}$$

Donc P est trop éloigné de l sur une proportion des entrées de D_n supérieure à $\lfloor 2d\sqrt{\delta}n \rfloor/(2n+1)$ qui est minorée $d\sqrt{\delta}/2$. Ce qui contredit l'hypothèse. ■

CHAPITRE 3

Polynômes

Nous allons maintenant utiliser le chapitre précédent pour définir un auto-testeur approché pour l'ensemble des fonctions polynomiales de degré fixé. Pour une considération technique nous noterons ce degré $(d-1)$ pour un certain entier $d \geq 1$ fixé.

1. Un test basé sur l'interpolation polynomiale

Nous utilisons une caractérisation bien connue des polynômes reposant sur l'itération de l'opérateur nabla défini ci-après.

Définition. Pour tout réel t, soit ∇_t l'*opérateur nabla* défini par

$$\nabla_t f(x) \stackrel{\text{déf}}{=} f(x+t) - f(x),$$

pour toute fonction réelle f telle que x et $x+t$ sont dans son domaine de définition.

Pour plus de simplicité, ∇_t^d dénotera l'opérateur ∇_t itéré d fois, et $\nabla_{\vec{t}}$, pour un certain $\vec{t} \in \mathbb{R}^d$, la composition des opérateurs $\nabla_{t_1} \circ \ldots \circ \nabla_{t_d}$.

Les propriétés suivantes de ∇_t sont facilement vérifiables :

$$\nabla_t \text{ est linéaire,} \tag{3.1}$$

$$\nabla_{t_1} \text{ et } \nabla_{t_2} \text{ commutent,} \tag{3.2}$$

$$\nabla_{t_1,t_2} = \nabla_{t_1+t_2} - \nabla_{t_1} - \nabla_{t_2}, \tag{3.3}$$

$$\nabla_t^d f(x) = \sum_{k=0}^{d} (-1)^{d-k} \binom{d}{k} f(x+kt). \tag{3.4}$$

Les polynômes réels de degré au plus $(d-1)$ sont alors caractérisés par l'équation fonctionnelle suivante, dite identité d'*interpolation polynomiale*, :

$$\nabla_t^d f(x) = 0.$$

Plus précisément

Lemme 3.1. *Si $f : \mathbb{Z} \to \mathbb{R}$ est telle que*

$$\forall x, t \in \mathbb{Z}, \qquad \nabla_t^d f(x) = 0,$$

alors f est une fonction polynomiale de degré au plus $(d-1)$.

Pour la preuve de ce lemme, voir par exemple [Cie59] ou [Kem57].

L'utilité de cette identité a été reconnue en auto-test par Rubinfeld et Sudan [RS92]. Grâce à sa simplicité, ils ont pu définir, pour le calcul exact, des auto-testeurs pour les polynômes définis sur des corps finis ou des domaines rationnels finis plus efficaces qu'un de ceux proposés précédemment par Lipton [Lip91].

En utilisant des résultats connus sur la stabilité de cette équation fonctionnelle [AB83], Ergün, Ravi Kumar, et Rubinfeld ont étendu l'auto-testeur de [RS92] au cas du calcul approché avec erreur absolue [EKR96]. L'approche est la suivante. La robustesse approchée de l'équation $\nabla_t^d f(x) = 0$ se montre de manière analogue à la robustesse approchée de l'équation de linéarité. L'étape de la stabilité est plus délicate et consiste à la réduire à celle de la stabilité de la

multilinéarité, *i.e.* que toute fonction satisfaisant approximativement l'équation de linéarité en chacune de ses coordonnées est proche d'une fonction multilinéaire. Nous reprendrons cette même structure pour le cas plus général des termes d'erreur valides [KMS99] afin de montrer la robustesse du test d'interpolation polynomiale. Dans ce qui suit $d \geq 1$ est un entier et μ_d est le plus petit multiple commun des entiers $1, \ldots, d$.

Test 3.1.

Test-Interpolation-polynomiale(P, β)
1. Tirer aléatoirement $x \in D_{d(d+1)^2\mu_d n}$ et $t \in D_{d(d+1)\mu_d n}$.
2. Rejeter si $|\nabla_t^d P(x)| > \beta(x,t)$.

2. Stabilité

2.1. Fonctions multilinéaires. Le théorème suivant établit la stabilité d'un système fonctionnel basé sur l'équation de linéarité. Nous en simplifions grandement la démonstration partielle précédemment publiée dans nos travaux [KMS99], tout en donnant un énoncé plus général. Dans le théorème suivant, nous utilisons la notation $\vec{z}_{z_i=t}$ pour désigner le vecteur $\vec{z}$ dont la i-ème coordonnée a été remplacée par t.

Théorème 3.1. *Soit β un terme d'erreur valide de degré $0 \leq p < 1$. Soit $g : (D_{2n})^d \to \mathbb{R}$ telle que pour chaque $i = 1, \ldots, d$, tout $\vec{z} \in (D_n)^d$ et tout $x, y \in D_n$,*

$$|g(\vec{z}_{z_i=x+y}) - g(\vec{z}_{z_i=x}) - g(\vec{z}_{z_i=y})| \leq \beta(\vec{z}_{z_i=x}, \vec{z}_{z_i=y}).$$

Alors la fonction multilinéaire $l : (D_n)^d \to \mathbb{R}$ définie par $l(n, \ldots, n) \stackrel{\text{déf}}{=} g(n, \ldots, n)$ satisfait

$$\forall \vec{z} \in (D_n)^d, \qquad |g(\vec{z}) - l(\vec{z})| \leq (2d-1)\frac{1+2^p}{2-2^p}\beta(\vec{z}).$$

Démonstration. Fixons la constante $C_p \stackrel{\text{déf}}{=} (1+2^p)/(2-2^p)$. La preuve s'effectue par récurrence sur l'entier d. Le cas $d = 1$ correspond au Théorème 2.1.

Supposons que la propriété soit satisfaite aux rangs 1 et $(d-1)$ pour un entier $d \geq 2$ fixé. Soit alors $g : (D_{2n})^d \to \mathbb{R}$ satisfaisant les hypothèses du théorème. Il existe d'après l'hypothèse de récurrence, deux fonctions réelles l_1 et l_2 définies sur $(D_n)^d$ telles que l_1 est linéaire par rapport à ses $(d-1)$ premières variables, l_2 est linéaire par rapport à sa dernière variable, et pour tout $\vec{z} \in (D_n)^d$:

$$|l_1(\vec{z}) - g(\vec{z})| \leq (2d-3)C_p\beta(\vec{z}), \tag{3.5}$$

$$|l_2(\vec{z}) - g(\vec{z})| \leq C_p\beta(\vec{z}), \tag{3.6}$$

et de plus $l_1(n, \ldots, n, z_d) = g(n, \ldots, n, z_d)$ et $l_2(z_1, \ldots, z_{d-1}, n) = g(z_1, \ldots, z_{d-1}, n)$.

Soit $l : (D_n)^d \to \mathbb{R}$ la fonction multilinéaire définie par $l(n, \ldots, n) \stackrel{\text{déf}}{=} g(n, \ldots, n)$. Cette fonction vérifie $l(., \ldots, ., n) = l_1(., \ldots, ., n)$. Nous allons montrer que l reste proche de g sur $(D_n)^d$. Fixons $\vec{z} \in (D_n)^d$ et $\vec{z'} \stackrel{\text{déf}}{=} \vec{z}_{z_d=n}$. La distance $|g(\vec{z}) - l(\vec{z})|$ est d'abord majorée par l'inégalité triangulaire $|g(\vec{z}) - l_2(\vec{z})| + |l_2(\vec{z}) - l(\vec{z})|$. L'inéquation (3.5) sert à majorer le premier terme. En utilisant la linéarité par rapport à la dernière variable de l et l_2, le deuxième terme se réécrit $\frac{x}{n}|l_2(\vec{z'}) - l(\vec{z'})|$. Mais alors la définition de l implique $l(\vec{z'}) = l_1(\vec{z'})$. Nous avons donc :

$$|g(\vec{z}) - l(\vec{z})| \leq C_p\beta(\vec{z}) + \frac{x}{n}|l_2(\vec{z'}) - l_1(\vec{z'})|.$$

Une inégalité triangulaire permet de majorer $|l_2(\vec{z'}) - l_1(\vec{z'})|$ à l'aide des inégalités (3.5) et (3.6), et nous obtenons :

$$|g(\vec{z}) - l(\vec{z})| \leq C_p\beta(\vec{z}) + C_p\frac{x}{n}(\beta(\vec{z'}) + (2d-3)\beta(\vec{z'})).$$

Mais par définition $\beta(n) \leq (n/x)^p \cdot \beta(x)$, pour tout $x \in D_n$, et donc $(x/n) \cdot \beta(\vec{z'}) \leq \beta(\vec{z})$, ce qui conclut la preuve. ■

2.2. Polynômes. La stabilité de l'identité d'interpolation polynomiale $\nabla_t^d f(x) = 0$ (cas uniforme) utilise la stabilité plus simple de l'équation $\nabla_{\vec{t}} f(0) = 0$ (cas non uniforme), elle-même fondée sur la stabilité des fonctions multilinéaires exposée précédemment. Dans le premier cas l'équation est fondée sur l'itération de l'opérateur nabla pour une même variation, alors que dans le deuxième la variation est arbitraire à chaque pas. Le lien avec les fonctions multilinéaires vient d'une des propriétés de ∇. En effet, puisque $\nabla_{t_1,t_2} = \nabla_{t_1+t_2} - \nabla_{t_1} - \nabla_{t_2}$, l'égalité $\nabla_{\vec{t}} f(0) = 0$ est satisfaite sur $(D_n)^d$ si et seulement si la fonction $g : \vec{t'} \to \nabla_{\vec{t'}} f(0)$, à $(d-1)$ variables, est multilinéaire sur $(D_n)^{(d-1)}$. Cependant le lien entre les cas uniforme et non uniforme introduit des *creux* où nous ne savons actuellement pas si la fonction reste proche d'un polynôme.

Lemme 3.2. *Soient β un terme d'erreur valide de degré $p \geq 0$ et μ_d le plus petit multiple commun des entiers $1, \ldots, d$. Soit $g : D_{d(d+1)\mu_d n} \to \mathbb{R}$ telle que*

$$\forall x, t \in D_{d\mu_d n}, \qquad |\nabla_t^d g(x)| \leq \beta(x, t).$$

Alors la fonction $f : D_{dn} \to \mathbb{R}$ définie en $x \in D_{dn}$ par $f(x) \stackrel{\text{déf}}{=} g(\mu_d x)$, satisfait

$$\forall \vec{t} \in (D_n)^d, \qquad |\nabla_{\vec{t}} f(0)| \leq (d\mu_d)^p 2^d \beta(\vec{t}).$$

Démonstration. Pour tout $\vec{\lambda} \in \{0,1\}^d$ et $\vec{t} \in (D_n)^d$, soient $t'_{\vec{\lambda}} \stackrel{\text{déf}}{=} -\sum_{i=1}^d \lambda_i t_i / i$, $t''_{\vec{\lambda}} \stackrel{\text{déf}}{=} \sum_{i=1}^d \lambda_i t_i$, et $(-1)^{\vec{\lambda}} \stackrel{\text{déf}}{=} (-1)^{\lambda_1 + \ldots + \lambda_d}$. La relation suivante [EKR96, Fact 17] relie les deux types d'itération de l'opérateur nabla :

$$\nabla_{\vec{t}} f(0) = \sum_{\vec{\lambda} \in \{0,1\}^d} (-1)^{\vec{\lambda}} \nabla^d_{t'_{\vec{\lambda}}} f(t''_{\vec{\lambda}}).$$

Or d'après l'hypothèse sur g, et puisque $\mu_d t''_{\vec{\lambda}}, \mu_d t'_{\vec{\lambda}} \in D_{d\mu_d n}$, on déduit que

$$\begin{aligned} |\nabla^d_{t'_{\vec{\lambda}}} f(t''_{\vec{\lambda}})| &= |\nabla^d_{\mu_d t'_{\vec{\lambda}}} g(\mu_d t''_{\vec{\lambda}})| \\ &\leq \beta(\mu_d t''_{\vec{\lambda}}, \mu_d t'_{\vec{\lambda}}) \\ &\leq (d\mu_d)^p \beta(\vec{t}). \end{aligned}$$

La preuve se conclut alors par une inégalité triangulaire. ■

La stabilité du cas non uniforme s'énonce comme suit.

Lemme 3.3. *Soit β un terme d'erreur valide de degré $0 \leq p < 1$. Soit $f : D_{dn} \to \mathbb{R}$ telle que*

$$\forall \vec{t} \in (D_n)^d, \qquad |\nabla_{\vec{t}} f(0)| \leq \beta(\vec{t}).$$

Alors il existe un polynôme $h_{d-1} : D_n \to \mathbb{R}$ de degré au plus $(d-1)$ tel que

$$\forall x \in D_n, \qquad |f(x) - h_{d-1}(x)| \leq \beta(x) \prod_{i=1}^{d-1} (1 + (2i-1)C_p),$$

où $C_p \stackrel{\text{déf}}{=} (1 + 2^p)/(2 - 2^p)$.

Démonstration. Prouvons le lemme par récurrence sur l'entier d. Lorsque $d = 1$, par hypothèse, $|f(x) - f(0)| = |\nabla_x f(0)| \leq \beta(x)$, pour tout $x \in D_n$, et donc la fonction constante $h_0(x) \stackrel{\text{déf}}{=} f(0)$ convient.

Supposons maintenant la propriété vraie au rang d pour un certain $d \geq 1$, et démontrons la au rang $(d+1)$. Soit donc $f : D_{(d+1)n} \to \mathbb{R}$ une fonction satisfaisant les hypothèses du

lemme. Par hypothèse, la fonction $G : (D_{2n})^d \to \mathbb{R}$ définie en $\vec{t} \in (D_{2n})^d$ par $G(\vec{t}) \stackrel{\text{déf}}{=} \nabla_{\vec{t}} f(0)$ est approximativement multilinéaire. Pour chaque $i = 1, \ldots, d$, tout $\vec{t} \in (D_n)^d$ et $x, y \in D_n$,

$$\begin{aligned} |G(\vec{t}_{t_i=x+y}) - G(\vec{t}_{t_i=x}) - G(\vec{t}_{t_i=y})| &= |\nabla_{t_1,\ldots,t_{i-1},x,y,t_{i+1},\ldots,t_d} f(0)| \\ &\leq \beta(\vec{t}_{t_i=x}, \vec{t}_{t_i=y}). \end{aligned}$$

D'après le Théorème 3.1, il existe donc une fonction multilinéaire $l : (D_n)^d \to \mathbb{R}$ telle que

$$\forall \vec{t} \in (D_n)^d, \qquad |G(\vec{t}) - l(\vec{t})| \leq (2d-1) C_p \beta(\vec{t}). \tag{3.7}$$

Etendons l à tout $\mathbb{Z}^d$ en préservant sa multilinéarité. Soit alors la fonction $f' : D_{(d+1)n} \to \mathbb{R}$ définie en $x \in D_{(d+1)n}$ par $f'(x) \stackrel{\text{déf}}{=} f(x) - l(x, \ldots, x)/d!$. Nous allons montrer que f' satisfait les conditions de l'hypothèse de récurrence pour un nouveau terme d'erreur valide $\beta' \stackrel{\text{déf}}{=} (1+(2d-1)C_p)\beta$. Pour ce, observons [Djo69, Lemme 2] tout d'abord que la multilinéarité de l entraîne que la fonction $H : \mathbb{Z} \to \mathbb{R}$ définie en $x \in \mathbb{Z}$ par $H(x) \stackrel{\text{déf}}{=} l(x, \ldots, x)$, satisfait

$$\forall x \in D_n,\ \forall \vec{t} \in (D_n)^d, \qquad \nabla_{\vec{t}} H(x) = d! \cdot l(\vec{t}). \tag{3.8}$$

Alors la définition de f', l'inéquation (3.7), et l'équation (3.8) impliquent directement pour tout $\vec{t} \in (D_n)^d$:

$$\begin{aligned} |\nabla_{\vec{t}} f'(0)| &= |\nabla_{\vec{t}} f(0) - \nabla_{\vec{t}} H(0)/d!| \\ &\leq |\nabla_{\vec{t}} f(x) - \nabla_{\vec{t}} f(0)| + |\nabla_{\vec{t}} f(0) - \nabla_{\vec{t}} H(0)/d!| \\ &= |\nabla_{\vec{t},0} f(0)| + |G(\vec{t}) - l(\vec{t})| \\ &\leq (1 + (2d-1)C_p)\beta(\vec{t}). \end{aligned}$$

D'après l'hypothèse, un polynôme $h_{d-1} : D_n \to \mathbb{R}$ de degré au plus $(d-1)$ existe tel que

$$\forall x \in D_n, \qquad |f'(x) - h_{d-1}(x)| \leq \beta(x)(1 + (2d-1)C_p) \prod_{i=1}^{d-1} (1 + (2i-1)C_p).$$

Alors le polynôme h_d de degré au plus d défini par $h_d \stackrel{\text{déf}}{=} h_{d-1} + H$ montre que la propriété est vraie au rang $(d+1)$. ∎

Nous pouvons maintenant établir la stabilité du cas uniforme en mettant bout à bout les Lemmes 3.2 et 3.3. Dans ce théorème la notation kD_n désigne l'ensemble $\{kx : x \in D_n\}$.

Théorème 3.2. *Soient β un terme d'erreur valide de degré $0 \leq p < 1$ et μ_d le plus petit commun diviseur des entiers $1, \ldots, d$. Soit $g : D_{d(d+1)\mu_d n} \to \mathbb{R}$ telle que*

$$\forall x, t \in D_{d\mu_d n}, \qquad |\nabla_t^d g(x)| \leq \beta(x, t).$$

Alors existe un polynôme $h_{d-1} : \mu_d D_n \to \mathbb{R}$ de degré au plus $(d-1)$ tel que

$$\forall x \in \mu_d D_n, \qquad |g(x) - h_{d-1}(x)| \leq \beta(x)(d\mu_d)^p 2^d \prod_{i=1}^{d-1} (1 + (2i-1)C_p),$$

où $C_p = (1+2^p)/(2-2^p)$.

3. Robustesse approchée

La robustesse approchée du test d'interpolation polynomiale peut s'énoncer comme ci-dessous.

Théorème 3.3. *Soient β un terme d'erreur valide de degré $p \geq 0$, μ_d le plus petit multiple commun des entiers $1, \ldots, d$, $0 \leq \delta \leq 1$, et $\varphi : \mathbb{Z} \to \mathbb{R}$ la fonction définie en $x \in \mathbb{Z}$ par $\varphi(x) \stackrel{\text{déf}}{=} \text{Max}\{d\mu_d\sqrt{\delta}n, |x|\}$. Soit $P : D_{2d(d+1)^2\mu_d n} \to \mathbb{R}$ tel que*

$$\Pr_{x \in D_{d(d+1)^2\mu_d n}, t \in D_{d(d+1)\mu_d n}} \left[|\nabla_t^d P(x)| > \beta(x,t) \right] \leq \frac{\delta}{16(d+1)^5}.$$

Alors la fonction $g : D_{(d+1)\mu_d n} \to \mathbb{R}$ définie en $x \in D_{(d+1)\mu_d n}$ par

$$g(x) \stackrel{\text{déf}}{=} \underset{y \in D_{\varphi(x)}}{\text{Méd}} \left(\sum_{i=1}^{d} \alpha_i P(x+it) \right),$$

où $\alpha_i = (-1)^{i+1}\binom{d}{i}$, satisfait

$$\Pr_{x \in \mu_d D_n} [|P(x) - g(x)| > \beta(\varphi(x))] \leq \frac{\mu_d \delta}{4(d+1)^2},$$

et

$$\forall x, t \in D_{d\mu_d n}, \qquad |\nabla_t^d g(x)| \leq (d+1)^3 2^{2(d+1)} \beta(\varphi(x), \varphi(t)).$$

Comme pour la robustesse approchée de l'équation de linéarité, nous séparons ici la preuve en deux lemmes. Tout d'abord montrons que g reste proche de P au sens suivant.

Lemme 3.4. *Sous les hypothèses du Théorème 3.3 et $m \stackrel{\text{déf}}{=} d\mu_d n$, la fonction g satisfait*

$$\Pr_{x \in D_{(d+1)m}} [|P(x) - g(x)| > \beta(\varphi(x))] \leq \frac{\delta}{4(d+1)^4}.$$

Démonstration. Soit $P_{x,t} \stackrel{\text{déf}}{=} \sum_{i=0}^{d} \alpha_i P(x+it)$. La définition de g et l'inégalité de Markov impliquent

$$\Pr_{x \in D_{(d+1)m}} [|g(x) - P(x)| > \beta(\varphi(x), \varphi(t))] \leq 2 \Pr_{x \in D_{(d+1)m}, t \in D_{\varphi(x)}} [|P_{x,t}| > \beta(\varphi(x))].$$

Mais $|P_{x,t}| = |\nabla_t^d P(x)|$, $\varphi(x) \geq \varphi(t) \geq |t|$, $\varphi(x) \geq |x|$, et le Lemme de Contraction 2.4 impliquent que le terme de droite est majorée par

$$\begin{aligned} & 2 \frac{|D_{(d+1)^2 m}| \cdot |D_{(d+1)m}|}{|(x,t) : x \in D_{(d+1)m}, t \in D_{\varphi(x)}|} \Pr_{u \in D_{(d+1)^2 m}, s \in D_{(d+1)m}} [|P_{u,s}| > \beta(u,s)] \\ \leq \; & 2 \frac{2(d+1)\delta}{16(d+1)^5} \\ = \; & \frac{\delta}{4(d+1)^4}. \end{aligned}$$

■

Le prochain lemme établit que la valeur médiane de la définition de g est en accord avec la plupart des votes.

Lemme 3.5. *Sous les hypothèses du Théorème 3.3 et $m \stackrel{\text{déf}}{=} d\mu_d n$, la fonction g satisfait pour tout $c \in D_{(d+1)m}$ et $I \subseteq D_{(d+1)m}$ tel que $|I| \geq \sqrt{\delta}m + 1$,*

$$\Pr_{t' \in I} \left[|g(c) - \sum_{j=1}^{d} \alpha_j P(c + jt')| > (d+1)^2 2^{d+1} \beta(\varphi(c), \varphi(t')) \right] < \frac{1}{2(d+1)}.$$

Démonstration. Soit $P_{x,t} \stackrel{\text{déf}}{=} \sum_{i=0}^{d} \alpha_i P(x+it)$. Comme précédemment, en remplaçant g par sa définition et en utilisant l'inégalité de Markov, le terme de gauche se majore par

$$2 \Pr_{t\in D_{\varphi(c)}, t'\in I}\left[|P_{c,t} - P_{c,t'}| > (d+1)^2 2^{d+1}\beta(\varphi(c),\varphi(t'))\right].$$

Nous utilisons maintenant la relation

$$\sum_{i=0}^{d} \alpha_i P_{c+it',t} = \sum_{j=0}^{d} \alpha_j P_{c+jt,t'},$$

qui se réécrit en

$$P_{c,t} - P_{c,t'} = \sum_{i=1}^{d} \alpha_i P_{c+it',t} - \sum_{j=1}^{d} \alpha_j P_{c+jt,t'},$$

et qui permet de majorer l'expression précédente par

$$\begin{aligned} & 2\sum_{i=1}^{d} \Pr_{t\in D_{\varphi(c)}, t'\in I}\left[|\alpha_i P_{c+it',t}| > (d+1)2^d\beta(\varphi(c),\varphi(t'))\right] \\ + \quad & 2\sum_{j=1}^{d} \Pr_{t\in D_{\varphi(c)}, t'\in I}\left[|\alpha_j P_{c+jt,t'}| > (d+1)2^d\beta(\varphi(c),\varphi(t'))\right]. \end{aligned}$$

Mais $\alpha_k \leq 2^d$, $(d+1)\beta(\varphi(c),\varphi(t)) \geq \beta(c+it',t)$ et $(d+1)\beta(\varphi(c),\varphi(t)) \geq \beta(c+jt,t')$ donc la quantité précédente se majore par :

$$\begin{aligned} & 2\sum_{i=1}^{d} \Pr_{t\in D_{\varphi(c)}, t'\in I}\left[|P_{c+it',t}| > \beta(c+it',t)\right] \\ + \quad & 2\sum_{j=1}^{d} \Pr_{t\in D_{\varphi(c)}, t'\in I}\left[|\alpha_j P_{c+jt,t'}| > \beta(c+jt,t')\right]. \end{aligned}$$

Le Lemme de Contraction 2.4 permet alors de majorer chacun de ces termes de probabilité par

$$\frac{|D_{(d+1)^2 m}| \cdot |D_{(d+1)m}|}{|D_{\varphi(c)}| \cdot |I|} \frac{\delta}{16(d+1)^5} < \frac{\delta}{8(d+1)^2},$$

ce qui conclut la preuve. ∎

Nous pouvons maintenant démontrer le Théorème 3.3.

Démonstration. Tout d'abord le Lemme 3.4 montre avec le Lemme de Contraction 2.4 que g reste proche de P comme annoncé.

Montrons maintenant que g satisfait approximativement l'identité d'approximation. Pour ce, posons $m \stackrel{\text{déf}}{=} d\mu_d n$ et $G_{u,s} \stackrel{\text{déf}}{=} g(u) - \sum_{k=1}^{d} \alpha_k P(u+ks)$. Fixons $x, t \in D_m$. Appliquons le Lemme 3.5 successivement aux cas $c = x$ et $I = \{x + jt' : t' \in D_{\sqrt{\delta}m}\}$, puis $c = x + it$ et $I = \{it' : t' \in D_{\sqrt{\delta}m}\}$, pour $i, j \in \{1, \ldots, d\}$. Nous en déduisons qu'il existe avec probabilité non nulle un élément $t' \in D_{\sqrt{\delta}m}$ tel que pour tout $i, j \in \{1, \ldots, d\}$,

$$|G_{x,t+jt'}| \leq (d+1)^2 2^{d+1}\beta(\varphi(x),\varphi(t+jt'))$$

et

$$|G_{x+it,it'}| \leq (d+1)^2 2^{d+1}\beta(\varphi(x+it),\varphi(it')).$$

Mais $t' \in D_{\sqrt{\delta}m}$, donc les quantités $\beta(\varphi(x),\varphi(t+jt'))$ et $\beta(\varphi(x+it),\varphi(it'))$ sont toutes deux majorées par $(d+1)\beta(\varphi(x),\varphi(t))$. Pour conclure, observons l'égalité

$$\left|\sum_{i=0}^{d} \alpha_i g(x+it)\right| = \left|\sum_{i=1}^{d} \alpha_i G_{x+it,it'} - \sum_{j=1}^{d} \alpha_j G_{x,t+jt'}\right|,$$

et la majoration $|\alpha_i| \leq 2^d$, pour chaque $0 \leq i \leq d$, ce qui entraîne :

$$\begin{aligned}
|\nabla_t^d g(x)| &= \left|\sum_{i=0}^{d} \alpha_i g(x+it)\right| \\
&= \left|\sum_{i=1}^{d} \alpha_i G_{x+it,it'} - \sum_{j=1}^{d} \alpha_j G_{x,t+jt'}\right| \\
&\leq 2^d \sum_{i=1}^{d} |G_{x+it,it'}| + 2^d \sum_{j=1}^{d} |G_{x,t+jt'}| \\
&\leq (d+1)^3 2^{2(d+1)} \beta(\varphi(x), \varphi(t)).
\end{aligned}$$

■

4. Auto-tester les polynômes

Comme pour les fonction linéaires, les résultats précédents conduisent naturellement à un auto-testeur construit sur le test d'interpolation polynomiale (Test 3.1). Sa robustesse est explicitée par le corollaire suivant des Théorèmes 3.3 et 3.2. L'ensemble $\mathcal{P}_{d-1}$ y désigne l'ensemble des fonctions polynomiales de $\mathbb{Z} \to \mathbb{R}$ de degré au plus $(d-1)$, et $\beta\text{-}\mathsf{Rej}(P)$ la probabilité de rejet suivante :

$$\beta\text{-}\mathsf{Rej}(P) \stackrel{\text{déf}}{=} \Pr_{x \in D_{d(d+1)^2\mu_d n}, t \in D_{d(d+1)\mu_d n}} \left[|\nabla_t^d P(x)| > \beta(x,t) \right].$$

Corollaire 3.1. *Soient β un terme d'erreur valide de degré $0 \leq p < 1$, μ_d le plus petit multiple commun des entiers $1, \ldots, d$, et $0 \leq \delta \leq 1$. Alors pour tout $P : D_{2d(d+1)^2\mu_d n} \to \mathbb{R}$,*

$$\beta\text{-}\mathsf{Rej}(P) \leq \frac{\delta}{16(d+1)^5} \implies (C\beta)\text{-}\mathsf{Dist}_{\mu_d D_n}(P, \mathcal{P}_{d-1})) \leq \frac{\mu_d \delta}{4(d+1)^2} + d\sqrt{\delta},$$

où $C \stackrel{\text{déf}}{=} 2^{\Theta(d \ln(d/(2-2^p))}$. Si de plus $p = 0$, alors le terme $d\sqrt{\delta}$ disparaît dans la majoration précédente.

La continuité du même test peut être énoncée ainsi.

Lemme 3.6. *Soient β un terme d'erreur valide de degré $0 \leq p \leq 1$ et μ_d le plus petit multiple commun des entiers $1, \ldots, d$. Alors tout $P : D_{2d(d+1)^2\mu_d n} \to \mathbb{R}$ satisfait*

$$\beta\text{-}\mathsf{Rej}(P) \leq 2(d+1) \cdot \left(\tfrac{\beta}{(d+1)^2 2^d}\right)\text{-}\mathsf{Dist}_{D_{2d(d+1)^2\mu_d n}}.$$

Les preuves du Corollaire 3.1 et du Lemme 3.6 étant structurellement similaires à leurs analogues pour le test de linéarité (Corollaire 2.1 et Lemme 2.7), elles sont omises.

Alors comme pour les fonctions linéaires, un auto-testeur des polynômes de degré au plus $(d-1)$ peut être construit à l'aide du Théorème 1.1.

Théorème 3.4. *Soient β un terme d'erreur valide de degré $0 \leq p < 1$, α un terme d'erreur (λ, λ')-équivalent à β, où $\lambda, \lambda' > 0$, et μ_d le plus petit multiple commun des entiers $1, \ldots, d$. Alors pour tout $0 < \delta \leq 1/2$ il existe un auto-testeur T qui satisfait sur l'entrée $0 < \gamma < 1$ et pour tout programme $P : D_{2d(d+1)^2\mu_d n} \to \mathbb{R}$:*

- *si $\left(\frac{\lambda\beta}{(d+1)^2 2^d}\right)\text{-}\mathsf{Dist}_{D_{2d(d+1)^2\mu_d n}}(P, \mathcal{P}_{d-1}) \leq \frac{\delta}{64(d+1)^6}$ alors $T^P(\gamma)$ retourne* `BON` *avec probabilité supérieure à $(1-\gamma)$,*
- *si $(\lambda' C\beta)\text{-}\mathsf{Dist}_{\mu_d D_n}(P, \mathcal{P}_{d-1}) > \begin{cases} \frac{\mu_d \delta}{2(d+1)^3} & \text{si } p = 0, \\ 2\mu_d\sqrt{\delta} & \text{si } 0 < p < 1, \end{cases}$, alors $T^P(\gamma)$ retourne* `MAUVAIS` *avec probabilité supérieure à $(1-\gamma)$,*

où $C \stackrel{\text{déf}}{=} 2^{\Theta(d \ln(d/(2-2^p))}$, *et en utilisant au plus* $O(d^6 \ln(1/\gamma)/\delta)$ *appels à* P, *comparaisons, décalages binaires, additions, et évaluations de* α.

CHAPITRE 4

Fonctions multilinéaires

Nous présentons dans le dernier chapitre de cette partie une nouvelle caractérisation des fonctions linéaires [Mag00]. Cette caractérisation consiste en une nouvelle équation fonctionnelle basée sur l'équation de linéarité dans laquelle est introduite un terme de dilatation la rendant stable pour les termes d'erreur linéaires. Il s'agit de la première équation fonctionnelle qui non seulement reste stable pour des termes d'erreur valides de degré 1, mais de plus n'utilise que des additions et des multiplications par des puissances de 2. Cette caractérisation permettra ensuite de construire un auto-testeur pour les fonctions multilinéaires avec erreur relative proprement dite. Pour simplifier la discussion, nous supposerons que l'entier n est pair.

1. Une équation de linéarité dilatée

Considérons un instant la caractérisation $f(x) = xf(1)$, pour tout x, des fonctions linéaires. Cette équation utilisant une multiplication peut être rendue robuste pour les termes d'erreur linéaires ainsi :

$$\forall x, \lambda \qquad f(\lambda x) = \lambda f(x). \tag{4.1}$$

La multiplication intervenant dans cette équation fonctionnelle est en général aussi complexe que le calcul de la fonction elle-même, et n'est donc pas directement utilisable en auto-test. Cependant une multiplication peut parfois être aussi simple qu'une addition, *i.e.* de complexité linéaire en la taille des entrées. Cette situation se présente entre autre quand λ est une puissance de 2. L'équation restreinte peut alors s'écrire :

$$\forall x, k \qquad f(2^k x) = 2^k f(x),$$

où k désigne nécessairement un entier. Le problème est maintenant que cette équation ne caractérise plus uniquement les fonctions linéaires. Cependant, elle corrige l'instabilité de l'équation de linéarité lorsqu'elle est intégrée à celle-ci de la manière suivante.

$$\forall x, y, k \qquad f(2^k x + y) = 2^k f(x) + f(y).$$

Avant de préciser quels sont les bons choix des entiers k nous devons définir un processus d'amplification. L'entier $n \geq 0$ étant fixé, pour tout $x \in \mathbb{Z}$, nous noterons k_x l'entier défini par

$$k_x \stackrel{\text{déf}}{=} \begin{cases} 0 & \text{si } x = 0, \\ \text{Min}\{k \in \mathbb{N} : 2^k |x| \geq \frac{n}{2}\} & \text{sinon.} \end{cases}$$

L'application $x \mapsto 2^{k_x} x$ transforme efficacement tout entier x *petit*, *i.e.* tel que $|x| < n/2$, en un entier *grand*, *i.e.* plus grand que $n/2$ en valeur absolue (*cf.* Figure 4.1).

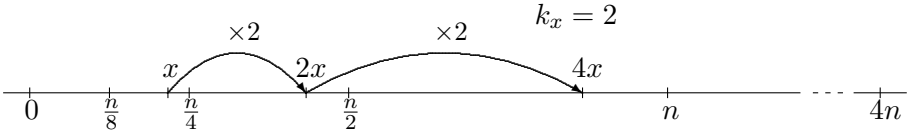

FIG. 4.1. Amplification par l'application $x \mapsto 2^{k_x} x$.

Nous pouvons maintenant définir complètement notre nouvelle équation fonctionnelle, dite *équation de linéarité dilatée*, qui caractérise les fonctions linéaires :

$$\boxed{\forall x, y \in D_{4n}, \qquad f(2^{k_x}x + y) - 2^{k_x} f(x) - f(y) = 0.}$$

Cette équation nous amènera donc à considérer le test suivant paramétré par un réel $\theta > 0$.

Test 4.1.

Test-linéarité-dilatée(P, θ)
1. Tirer aléatoirement $x, y \in D_{4n}$.
2. Rejeter si $|P(2^{k_x}x + y) - 2^{k_x}P(x) - P(y)| > \theta 2^{k_x}|x|$.

Une façon de voir ce test est de le considérer comme un test de linéarité ordinaire avec erreur absolue pour les grands x, auquel est ajouté un terme de dilatation pour les petits x.

2. Robustesse approchée

La robustesse approchée du test de linéarité dilatée s'énonce ainsi.

Théorème 4.1. *Soient $0 \leq \delta \leq 1/512$ et $\theta \geq 0$. Soit $P : D_{8n} \to \mathbb{R}$ tel que*

$$\Pr_{x,y \in D_{4n}} \left[|P(2^{k_x}x + y) - 2^{k_x}P(x) - P(y)| > \theta 2^{k_x}|x| \right] \leq \delta.$$

Alors la fonction $g : D_{2n} \to \mathbb{R}$ définie en $x \in D_{2n}$ par

$$g(x) \stackrel{\text{déf}}{=} \operatorname*{Méd}_{y \in D_{2n} : xy \geq 0} \left(\frac{1}{2^{k_x}} P(2^{k_x}x + y) - P(y) \right),$$

satisfait

$$\Pr_{x \in D_n} [|P(x) - g(x)| > \theta|x|] \leq 32\delta,$$

$$|g(n) + g(-n)| \leq 16\theta n,$$

$$\forall x \in D_{2n}, \quad g(x) = \frac{1}{2^{k_x}} g(2^{k_x}x),$$

et pour tout $x, y \in \{n/2, \ldots, n\}$ (resp. $x, y \in \{-n/2, \ldots, -n\}$)

$$|g(x + y) - g(x) - g(y)| \leq 24\theta n.$$

Pour prouver ce théorème, nous utilisons le même découpage que celui des Théorèmes 2.2 et 3.3.

Lemme 4.1. *Sous les hypothèses du Théorème 4.1, la fonction g satisfait*

$$\Pr_{x \in D_n} [|g(x) - P(x)| > \theta|x|] \leq 32\delta.$$

Démonstration. La preuve du lemme est très similaire à celle des Lemmes 2.5 et 3.4. Posons $P_{x,y} \stackrel{\text{déf}}{=} P(2^{k_x}x + y) - 2^{k_x}P(x) - P(y)$. Alors la technique usuelle permet d'écrire :

$$\begin{aligned} \Pr_{x \in D_n} [|g(x) - P(x)| > \theta|x|] &= \Pr_{x \in D_n} \left[\left| \operatorname*{Méd}_{y \in D_{2n} : xy \geq 0} (P_{x,y}) \right| > \theta 2^{k_x}|x| \right] \\ &\leq 32\delta. \end{aligned}$$

■

Lemme 4.2. *Sous les hypothèses du Théorème 4.1, la fonction g satisfait pour tout $c \in D_{2n}$ tel que $|c| \geq n/2$, et pour tout $I \subseteq \{y \in D_{2n} : |c + y| \geq n/2\}$ non vide,*

$$\Pr_{y \in I} [|g(c) - (P(c + y) - P(y))| > 8\theta n] < 16 \frac{|D_{4n}|}{|I|} \delta.$$

Démonstration. Fixons $c \in D_{2n}$ tel que $|c| \geq n/2$ et reprenons la structure de la preuve du Lemme 2.6. Posons $P_{x,y} \stackrel{\text{déf}}{=} P(2^{k_x}x + y) - 2^{k_x}P(x) - P(y)$ et $D \stackrel{\text{déf}}{=} \{z \in D_{2n} : cz \geq 0\}$. Après avoir observé que $k_{c+t} = 0$ pour tout $t \in D \cup I$, le procédé habituel permet d'écrire :

$$\Pr_{y \in I}\left[|g(c) - (P(c+y) - P(y))| > 8\theta n\right] \leq 2 \Pr_{y \in I, z \in D}\left[|P_{c+y,z} - P_{c+z,y}| > 8\theta n\right].$$

Mais $4n \geq |c + t|$, pour tout $t \in D \cup I$, donc finalement :

$$\begin{aligned} & \Pr_{y \in D}\left[|g(c) - (P(c+y) - P(y))| > 8\theta n\right] \\ \leq\ & 2 \Pr_{y \in I, z \in D}\left[|P_{c+y,z}| > \theta|c+y|\right] \\ & +2 \Pr_{y \in I, z \in D}\left[|P_{c+z,y}| > \theta|c+z|\right] \\ <\ & 16\frac{D_{4n}}{|I|}\delta. \end{aligned}$$

■

Démontrons maintenant le Théorème 4.1.

Démonstration. Le Lemme 4.1 montre que la fonction g reste proche de P. Ensuite la définition de g implique directement $g(2^{k_x}x) = 2^{k_x}g(x)$, pour tout $x \in D_{2n}$.

Montrons que $g(-n)$ et $-g(n)$ sont proches. D'après Lemme 4.2, il existe avec probabilité strictement supérieure à $(1 - 512\delta)$, donc non nulle, un élément $y \in \{0, 1, \ldots, n/2\}$ tel que

$$\begin{aligned} |g(-n) - (P(-n+y) - P(y))| &\leq 8\theta n \\ |g(n) - (P(n + (-n+y)) - P(-n+y))| &\leq 8\theta n. \end{aligned}$$

Alors la distance entre $g(-n)$ et $-g(n)$ est bornée comme voulue suite à une inégalité triangulaire utilisant ces majorations.

Reste à montrer l'additivité de g. Fixons $a, b \in D_n$, tels que $a, b \geq n/2$ (resp. $a, b \leq -n/2$). Le Lemme 4.2 permet alors d'exhiber avec probabilité strictement supérieure à $(1 - 384\delta)$, donc non nulle, un élément $y \in \{0, 1, \ldots, n\}$ (resp. $y \in \{0, -1, \ldots, -n\}$) tel que :

$$\begin{aligned} |g(a+b) - (P(a+b+y) - P(y))| &\leq 8\theta n \\ |g(a) - (P(a+y) - P(y))| &\leq 8\theta n \\ |g(b) - (P(b + (a+y)) - P(a+y))| &\leq 8\theta n. \end{aligned}$$

La preuve s'achève alors par une inégalité triangulaire combinant ces trois majorations. ■

3. Stabilité

Nous nous intéressons ici à un sous-problème de la stabilité de l'équation de linéarité dilatée. Cette stabilité est suffisante pour démontrer la robustesse complète du test de linéarité dilatée avec l'aide de la robustesse approchée précédemment étudiée (Théorème 4.1). Avant d'énoncer le résultat, introduisons la notation D_n^+ pour désigner la partie positive de D_n, *i.e.* $D_n^+ \stackrel{\text{déf}}{=} \{x \in D_n : x > 0\}$.

Théorème 4.2. *Soit $\theta \geq 0$. Soit $g : D_{2n}^+ \to \mathbb{R}$ telle que*

$$\begin{aligned} \forall x, y \in \{n/2, \ldots, n\}, \qquad & |g(x+y) - g(x) - g(y)| \leq \theta n, \\ \forall x \in D_n^+, \qquad & g(x) = \frac{1}{2^{k_x}} g(2^{k_x}x). \end{aligned}$$

alors la fonction linéaire $l : D_n^+ \to \mathbb{R}$ définie par $l(n) \stackrel{\text{déf}}{=} g(n)$ satisfait

$$\forall x \in D_n^+, \qquad |g(x) - l(x)| \leq 5\theta x.$$

Démonstration. La preuve est similaire à celle du Théorème 2.1. Tout d'abord g est prolongé en une fonction h sur l'ensemble d'entiers D défini par $D \stackrel{\text{déf}}{=} \{x \in \mathbb{Z} : x \geq n/2\}$. Après avoir montré que h satisfait approximativement une propriété de doublement, le Lemme 2.3 est utilisé pour conclure que h reste proche d'une fonction linéaire. La propriété de dilatation de g achève alors la preuve.

L'extension h est construite de sorte que g et h coïncident sur l'ensemble $\{n/2, \ldots, n\}$, et que h suive la même direction que la fonction linéaire l définie par $l(n) \stackrel{\text{déf}}{=} g(n)$. Soit donc pour tout $x \in D$:

$$h(x) \stackrel{\text{déf}}{=} \begin{cases} g(x) & \text{si } n/2 \leq x \leq n, \\ h(x - n/2) + g(n)/2 & \text{si } x > n \end{cases}.$$

Montrons alors que pour tout $x \in D$,

$$|h(2x) - 2h(x)| \leq \frac{5\theta n}{2}. \tag{4.2}$$

Puisque $h(x) = h(x-n/2)+g(n)/2$ pour les entiers $x > n$, Il suffit de démonter cette propriété lorsque $n/2 \leq x \leq n$.

Supposons d'abord que $n/2 \leq x \leq 3n/4$. Alors $h(2x) = g(2x-n/2)+g(n)/2$ et $h(x) = g(x)$, donc l'additivité approchée de g entraîne :

$$\begin{aligned} |h(2x) - 2h(x)| &= |g(2x - n/2) + g(n)/2 - 2g(x)| \\ &\leq |g(2x) - g(2x - n/2) - g(n/2)| + \frac{1}{2}|g(n) - 2g(n/2)| + |g(2x) - 2g(x)| \\ &\leq \frac{5\theta n}{2}. \end{aligned}$$

Considérons maintenant le cas $3n/4 < x \leq n$. Alors $h(2x) = g(2x-n)+g(n)$ et $h(x) = g(x)$, donc de manière analogue :

$$\begin{aligned} |h(2x) - 2h(x)| &= |g(2x - n) + g(n) - 2g(x)| \\ &\leq |g(2x) - g(2x - n) - g(n)| + |g(2x) - 2g(x)| \\ &\leq 2\theta n. \end{aligned}$$

Après avoir observer que pour tout $x \in D$, $\lim_{m\to\infty} h(2^m x)/2^m = l(x)$, le Lemme 2.3 entraîne

$$\forall x \in D, \qquad |l(x) - h(x)| \leq \frac{5\theta n}{2}.$$

La preuve est alors terminée puisque pour tout $x \in D_n^+$, la fonction g satisfait $g(x) = g(2^{k_x}x)/2^{k_x}$ et $2^{k_x}x \geq n/2$. ∎

4. Auto-tester les fonctions linéaires (bis)

4.1. Pour des termes d'erreur linéaires. Les résultats des deux sections précédentes nous permettent de construire un auto-testeur pour des distances seuils construites sur des termes d'erreurs de calcul linéaires. Ces termes d'erreurs sont valides de degré 1 dans la terminologie du Chapitre 2. Par conséquent cet auto-testeur comble une lacune du Théorème 2.3 qui ne traite que les termes d'erreur valides de degré $0 \leq p < 1$.

Tout d'abord, la robustesse du test de linéarité dilatée (Test 4.1) découle des Théorèmes 4.1 et 4.2, et est explicitée dans le corollaire suivant. Par abus de notation, $\theta\text{-}\mathsf{Rej}(P)$ y désigne la probabilité de rejet au test de linéarité dilatée sur D_{4n} et paramétré par le réel $\theta \geq 0$:

$$\theta\text{-}\mathsf{Rej}(P) \stackrel{\text{déf}}{=} \Pr_{x,y \in D_{4n}} \left[|P(2^{k_x}x + y) - 2^{k_x}P(x) - P(y)| > \theta 2^{k_x}|x| \right],$$

et $(\theta|x|)\text{-}\mathsf{Dist}(P, \mathcal{L})$ la distance seuil de P à l'ensemble des fonctions linéaires pour le terme d'erreur de calcul $(x, v) \mapsto \theta|x|$.

Corollaire 4.1. *Soient $\theta \geq 0$ et $0 \leq \delta \leq 1/512$. Alors pour tout $P : D_{8n} \to \mathbb{R}$,*

$$\theta\text{-}\mathsf{Rej}(P) \leq \delta \implies (137\theta|x|)\text{-}\mathsf{Dist}_{D_n}(P, l) \leq 32\delta,$$

où la fonction linéaire $l \in \mathcal{L}$ est définie par

$$l(n) \stackrel{\text{déf}}{=} \underset{y \in D_{2n} : y \geq 0}{\text{Méd}} \left(P(n+y) - P(y)\right).$$

Démonstration. Supposons $\theta\text{-}\mathsf{Rej}(P) \leq \delta$. Les Théorèmes 4.1 et 4.2 entraînent l'existence d'une fonction $g : D_{2n} \to \mathbb{R}$ qui reste proche de P :

$$\Pr_{x \in D_n}\left[|P(x) - g(x)| > \theta|x|\right] \leq 32\delta,$$

mais aussi proche sur D_n^+ (resp. D_n^-) d'une fonction linéaire l (resp. l') coïncidant avec g en n (resp $-n$) :

$$\forall x \in D_n^+, \quad |g(x) - l(x)| \leq 120\theta|x|,$$
$$\forall x \in D_n^-, \quad |g(x) - l'(x)| \leq 120\theta|x|,$$

et enfin telle que $g(0) = 0$ et $|g(n) + g(-n)| \leq 16\theta n$. Donc finalement

$$\forall x \in D_n, \qquad |g(x) - l(x)| \leq 136\theta|x|.$$

Observons que d'après la construction de g au Théorème 4.1, la fonction l est bien définie comme annoncée. ∎

La continuité du même test s'établie simplement comme suit en utilisant la même méthode qu'au Lemme 2.7.

Lemme 4.3. *Soit $\theta \geq 0$. Alors tout $P : D_{8n} \to \mathbb{R}$ satisfait*

$$\theta\text{-}\mathsf{Rej}(P) \leq 6 \cdot (\tfrac{\theta|x|}{18})\text{-}\mathsf{Dist}_{D_{8n}}(P, \mathcal{L}).$$

Afin que notre auto-testeur soit efficace, le terme d'erreur $x \to \theta 2^{k_x}|x|$ sera remplacé comme à la Section 4 par le terme d'erreur $(1, 4)$-équivalent

$$x \to \begin{cases} 0 & \text{si } x = 0 \text{ ou } \theta = 0 \\ 2^{\lceil log_2 \theta \rceil + k_x + \lceil log_2(|x|) \rceil} & \text{sinon.} \end{cases}$$

Ainsi robustesse et continuité du test de linéarité dilatée entraînent avec le Théorème 1.1 l'existence d'un auto-testeur pour les fonctions linéaires comme suit.

Théorème 4.3. *Soit $\theta \geq 0$. Alors pour tout $0 < \delta \leq 1/1024$ il existe un auto-testeur T qui satisfait sur l'entrée $0 < \gamma < 1$ et pour tout programme $P : D_{8n} \to \mathbb{R}$:*

- *si $(\frac{\theta|x|}{18})\text{-}\mathsf{Dist}_{D_{8n}}(P, \mathcal{L}) \leq \frac{\delta}{12}$ alors $T^P(\gamma)$ retourne* `BON` *avec probabilité supérieure à $(1 - \gamma)$,*
- *si $(548\theta|x|)\text{-}\mathsf{Dist}_{D_n}(P, \mathcal{L}) \geq 64\delta$ alors $T^P(\gamma)$ retourne* `MAUVAIS` *avec probabilité supérieure à $(1 - \gamma)$,*

en utilisant au plus $O(\ln(1/\gamma)/\delta)$ appels à P, comparaisons, décalages binaires, et additions.

4.2. Pour des termes d'erreurs relatifs. Le test de linéarité dilatée peut être modifié pour établir une continuité et une robustesse pour des distances seuil construites sur des termes d'erreur de calcul relatifs. Soit donc le test suivant nommé *test de linéarité relative.*

Test 4.2.

Test-linéarité-relative(P, θ)

1. Tirer aléatoirement $y \in \{0, \ldots, n\}$.
2. Calculer $G = P(n - y) + P(y)$.
3. Calculer $\tilde{\theta} = \theta|G|/n$.
4. Effectuer **Test-linéarité-dilatée**$(ext(P, G), \tilde{\theta})$.

Où la fonction $ext(P,G)$ est définie en $x \in \mathbb{Z}$ par :

$$ext(P,G)(x) \stackrel{\text{déf}}{=} \begin{cases} P(x) & \text{si } x \in D_n, \\ ext(P,G)(x-n)+G & \text{si } x > n, \\ ext(P,G)(x+n)-G & \text{si } x < -n. \end{cases}$$

Posons $\theta\text{-}\,\mathsf{Rej}^r(P)$ la probabilité de rejet à ce test, et $\theta\text{-}\,\mathsf{Dist}^r_{D_n}(P,\mathcal{L})$ la distance seuil de P aux fonctions linéaires sur D_n pour le terme d'erreur relatif $(x,v) \mapsto \theta|v|$. Le test de linéarité relative est drobuste pour cette distance seuil.

Théorème 4.4. *Soient $\theta \geq 0$ et $0 \leq \delta \leq 1/512$. Alors pour tout $P : D_n \to \mathbb{R}$,*

$$\theta\text{-}\,\mathsf{Rej}^r(P) \leq \delta \implies 137\theta\text{-}\,\mathsf{Dist}^r_{D_n}(P,\mathcal{L}) \leq 32\delta.$$

Démonstration. Soit $P : D_n \to \mathbb{R}$ satisfaisant $\theta\text{-}\,\mathsf{Rej}^r(P) \leq \delta$. Nécessairement il existe un élément $y \in D_n$ tel que pour les valeurs $G \stackrel{\text{déf}}{=} P(n-y)+P(y)$ et $\tilde{\theta} \stackrel{\text{déf}}{=} \theta|G|/n$, la fonction $ext(P,G)$ ait une probabilité de rejet inférieure à δ au test **Test-linéarité-dilatée(**$ext(P,G),\tilde{\theta}$**)**. Alors la robustesse de ce test exhibée au Corollaire 4.1 nous apprend l'existence d'une fonction linéaire $l \in \mathcal{L}$ définie par

$$l(n) \stackrel{\text{déf}}{=} \underset{y \in D_{2n} : y \geq 0}{\text{Méd}} \left(P(n+y)-P(y)\right),$$

et satisfaisant $(137\tilde{\theta}|x|)\text{-}\,\mathsf{Dist}_{D_n}(P,l) \leq 32\delta$. Mais un simple calcul montre que $l(n) = G$, et donc

$$17\theta\text{-}\,\mathsf{Dist}^r_{D_n}(P,l) = (17\tilde{\theta}|x|)\text{-}\,\mathsf{Dist}_{D_n}(P,l) \leq 32\delta.$$

■

La continuité du même test s'énonce ainsi.

Lemme 4.4. *Soit $0 \leq \theta \leq 18$. Alors tout $P : D_n \to \mathbb{R}$ satisfait*

$$\theta\text{-}\,\mathsf{Rej}^r(P) \leq 10 \cdot (\tfrac{\theta}{72})\text{-}\,\mathsf{Dist}^r_{D_n}(P,\mathcal{L}).$$

Démonstration. Soient $\delta \stackrel{\text{déf}}{=} (\frac{\theta}{72})\text{-}\,\mathsf{Dist}^r_{D_n}(P,\mathcal{L})$, et $l \in \mathcal{L}$ satisfaisant $(\frac{\theta}{72})\text{-}\,\mathsf{Dist}^r_{D_n}(P,l) = \delta$. Posons $G \stackrel{\text{déf}}{=} P(n-y)+P(y)$, et $\tilde{\theta} \stackrel{\text{déf}}{=} \theta|G|/n$, pour chaque $y \in \{0,\ldots,n\}$.

Le Lemme de Contraction 2.4 permet d'établir que, avec probabilité supérieure à $(1-4\delta)$ prise sur y parcourant $\{0,\ldots,n\}$, la valeur G satisfait $|G-l(n)| \leq \theta|l(n)|/36$.

Lorsque l'inégalité précédente est satisfaite la distance $(\frac{\theta}{36})\text{-}\,\mathsf{Dist}^r_{D_{8n}}(ext(P,G),l)$ se majore aisément par δ. Mais comme $\theta/36 \leq 1/2$, l'inégalité supposée entraîne aussi $|l(n)| \leq 2|G|$, et la majoration de la distance précédente a une variante non relative :

$$(\tfrac{\theta|x|}{18})\text{-}\,\mathsf{Dist}_{D_{8n}}(ext(P,G),l) \leq \delta.$$

Le Lemme 4.3 nous dit alors que la probabilité de rejet au test de linéarité dilatée est inférieure à 6δ.

En conclusion, la probabilité de rejet au test de linéarité relative est inférieure à 10δ. ■

Le Théorème 1.1 nous permet maintenant de construire à partir du test de linéarité relative, un auto-testeur pour les fonctions linéaires avec erreur relative. Pour être efficace, la quantité $\tilde{\theta}$ apparaissant dans le test de linéarité relative sera remplacée par $\tilde{\theta}'$, une puissance de 2 définie par

$$\tilde{\theta}' \stackrel{\text{déf}}{=} \begin{cases} 0 & \text{si } \theta = 0 \text{ ou } G = 0, \\ 2^{\lceil \log_2 \theta \rceil + \lceil \log_2 |G| \rceil - \lfloor \log_2 n \rfloor} & \text{sinon,} \end{cases}$$

qui vérifie $\tilde{\theta} \leq \tilde{\theta}' \leq 8\tilde{\theta}$.

Théorème 4.5. *Soit $0 \leq \theta \leq 18$. Alors pour tout $0 < \delta \leq 1/1024$ il existe un auto-testeur T qui satisfait sur l'entrée $0 < \gamma < 1$ et pour tout programme $P : D_{8n} \to \mathbb{R}$:*

- *si $(\frac{\theta}{72})$- $\mathsf{Dist}^r_{D_n}(P, \mathcal{L}) \leq \frac{\delta}{20}$ alors $T^P(\gamma)$ retourne* `BON` *avec probabilité supérieure à $(1-\gamma)$,*
- *si (137θ)- $\mathsf{Dist}^r_{D_n}(P, \mathcal{L}) \geq 64\delta$ alors $T^P(\gamma)$ retourne* `MAUVAIS` *avec probabilité supérieure à $(1-\gamma)$,*

en utilisant au plus $O(\ln(1/\gamma)/\delta)$ appels à P, comparaisons, décalages binaires, et additions.

5. Auto-tester les fonctions multilinéaires

Nous abordons enfin l'auto-test des fonctions multilinéaires. Nous allons exhiber un test pour ces fonctions construit sur le test de linéarité relative (Test 4.2). Intuitivement ce test vérifie aléatoirement que le programme est approximativement linéaire suivant chacune de ses variables.

Test 4.3.

Test-multilinéarité-relative(P, θ)
1. Tirer aléatoirement $\vec{z} \in (D_n)^d$.
2. Tirer aléatoirement $i \in \{1, \ldots, d\}$.
3. Effectuer **Test-linéarité-relative**$(\tilde{P}^i_{\vec{z}}, \theta)$.

Où la fonction $\tilde{P}^i_{\vec{z}}$ est définie en $t \in D_n$ par

$$\tilde{P}^i_{\vec{z}}(t) \stackrel{\text{déf}}{=} P(z_1, \ldots, z_{i-1}, t, z_{i+1}, \ldots, z_d).$$

La robustesse et la continuité de ce test peuvent se montrer à l'aide de celles de la section précédente. Pour ce, nous utilisons une méthode similaire à celle de Friedl, Hátsági, et Shen [FHS94] pour un scénario d'auto-test de polynômes en calcul exact. En majorant et minorant la distance entre une fonction à d variables et aux fonctions multilinéaires, par ses distances successives aux fonctions linéaires par rapport à leur i-ème variable, pour $i = 1, \ldots, d$. Nous établissons les mêmes encadrements pour notre situation. Avant de les énoncer, posons $\mathcal{L}^d$ l'ensemble des fonctions multilinéaires sur $(D_n)^d$, et $\mathcal{L}^d_i$ l'ensemble des fonctions définies sur $(D^+_n)^d$ et linéaires par rapport à leur i-ème variable.

Lemme 4.5. *Soit $\theta \geq 0$. Alors toute fonction $f : (D_n)^d \to \mathbb{R}$ satisfait*

$$\frac{1}{d}\sum_{i=1}^{d} \theta\text{- }\mathsf{Dist}^r_{(D_n)^d}(f, \mathcal{L}^d_i) \leq \theta\text{- }\mathsf{Dist}^r_{(D_n)^d}(f, \mathcal{L}^d).$$

Démonstration. La preuve est immédiate en remarquant que pour chaque $i = 1, \ldots, d$:

$$\theta\text{- }\mathsf{Dist}^r_{(D_n)^d}(f, \mathcal{L}^d_i) \leq \theta\text{- }\mathsf{Dist}^r_{(D_n)^d}(f, \mathcal{L}^d).$$

■

L'autre borne est par contre plus complexe à démontrer.

Lemme 4.6. *Soit $0 \leq \theta \leq 1/(16d^2)$. Alors toute fonction $f : (D_n)^d \to \mathbb{R}$ satisfait*

$$(4d\theta)\text{- }\mathsf{Dist}^r_{(D_n)^d}(f, \mathcal{L}^d) \leq 2\sum_{i=1}^{d} \theta\text{- }\mathsf{Dist}^r_{(D_n)^d}(f, \mathcal{L}^d_i).$$

Démonstration. Nous prouvons le résultat par récurrence sur d en réutilisant certaines idées de la preuve du Théorème 3.1.

Le cas $d = 1$ est immédiat. Supposons la propriété vraie au rang $(d-1)$ pour un entier $d \geq 2$. Nous allons alors prouver que l'inégalité suivante est vraie, ce qui prouvera avec l'hypothèse de récurrence que la propriété est vraie au rang d :

$$(4d\theta)\text{- }\mathsf{Dist}^r(f, \mathcal{L}^d) \leq (4(d-1)\theta)\text{- }\mathsf{Dist}^r(f, L^{d-1}) + 2(\theta\text{- }\mathsf{Dist}^r(f, \mathcal{L}^d_d)), \tag{4.3}$$

où Dist^r représente pour simplifier $\mathsf{Dist}^r_{(D_n)^d}$, et L^{d-1} l'ensemble des fonctions réelles définies sur $(D_n)^d$ et qui sont linéaires par rapport à leurs $(d-1)$ premières variables.

Posons $\delta_1 \stackrel{\text{déf}}{=} (4(d-1)\theta)\text{-}\,\mathsf{Dist}^r(f, L^{d-1})$ et $\delta_2 \stackrel{\text{déf}}{=} \theta\text{-}\,\mathsf{Dist}^r(f, \mathcal{L}^d_d)$. Soient alors $l_1 \in L^{d-1}$ et $l_2 \in \mathcal{L}^d_d$ telles que $(4(d-1)\theta)\text{-}\,\mathsf{Dist}^r(f, l_1) = \delta_1$, et $\theta\text{-}\,\mathsf{Dist}^r(f, l_2) = \delta_2$. Nous avons clairement la majoration :

$$\Pr_{\vec{z}\in(D_n)^d}\left[\begin{array}{ll} & |l_2(\vec{z}) - P(\vec{z})| > \theta|l_2(\vec{z})| \\ \text{or} & |P(\vec{z}) - l_1(\vec{z})| > (4d-4)\theta|l_1(\vec{z})| \end{array}\right] \leq \delta_1 + \delta_2. \tag{4.4}$$

Cette inégalité implique nécessairement l'existence d'un élément $b \in D_n$ tel que

$$\Pr_{\vec{z}\in(D_n)^d : z_d = b}\left[\begin{array}{ll} & |l_2(\vec{z}) - P(\vec{z})| > \theta|l_2(\vec{z})| \\ \text{or} & |P(\vec{z}) - l_1(\vec{z})| > (4d-4)\theta|l_1(\vec{z})| \end{array}\right] \leq \delta_1 + \delta_2. \tag{4.5}$$

Soit donc $l \in \mathcal{L}^d$ la fonction multilinéaire définie sur $(D_n)^d$ par $l(\cdot, \ldots, \cdot, b) \stackrel{\text{déf}}{=} l_1(\cdot, \ldots, \cdot, b)$. Nous allons montrer que f est proche de l au sens de l'inégalité (4.3).

Tout d'abord, puisque $0 \leq \theta \leq 1/(16d^2)$ on a facilement que :

$$(4d\theta)\text{-}\,\mathsf{Dist}^r(f, l) \leq \Pr_{\vec{z}\in(D_n)^d}\left[\begin{array}{ll} & |P(\vec{z}) - l_2(\vec{z})| > \theta|l_2(\vec{z})| \\ \text{or} & |l_2(\vec{z}) - l(\vec{z})| > (4d-2)\theta|l_2(\vec{z})| \\ \text{or} & |l_2(\vec{z})| > (1+4d\theta)|l(\vec{z})| \end{array}\right].$$

Mais l et l_2 sont linéaires par rapport à leur dernière variable, donc la variable z_d peut arbitrairement fixée à $z_d = b$ dans les deux dernières inégalités sans changer la probabilité globale. En utilisant alors la définition de l_2, nous obtenons :

$$(4d\theta)\text{-}\,\mathsf{Dist}^r(f, l) \leq \delta_2 + \Pr_{\vec{z}\in(D_n)^d : z_d = b}\left[\begin{array}{ll} & |l_2(\vec{z}) - l(\vec{z})| > (4d-2)\theta|l_2(\vec{z})| \\ \text{or} & |l_2(\vec{z})| > (1+4d\theta)|l(\vec{z})| \end{array}\right].$$

En répétant la première manipulation puisque $0 \leq \theta \leq 1/(16d^2)$ et $l(\cdot, \ldots, \cdot, b) = l_1(\cdot, \ldots, \cdot, b)$, nous avons :

$$(4d\theta)\text{-}\,\mathsf{Dist}^r(f, l) \leq \delta_2 + \Pr_{\vec{z}\in(D_n)^d : z_d = b}\left[\begin{array}{ll} & |l_2(\vec{z}) - P(\vec{z})| > \theta|l_2(\vec{z})| \\ \text{or} & |P(\vec{z}) - l_1(\vec{z})| > (4d-4)\theta|l_1(\vec{z})| \\ \text{or} & |l_2(\vec{z})| > (1+4d\theta)|l_1(\vec{z})| \\ \text{or} & |l_1(\vec{z})| > (1+4d\theta)|l_2(\vec{z})| \end{array}\right].$$

Etant donné l'encadrement $0 \leq \theta \leq 1/(16d^2)$, il se trouve que lorsque les deux dernières inégalités ne sont pas satisfaites, alors nécessairement aucune des deux premières ne l'est. Donc ce terme de probabilité ne dépend pas des deux dernières conditions. Alors enfin, l'inégalité (4.5) nous amène à la majoration souhaitée :

$$\begin{aligned}(4d\theta)\text{-}\,\mathsf{Dist}^r(f, l) &\leq \delta_2 + \Pr_{\vec{z}\in(D_n)^d : z_d = b}\left[\begin{array}{ll} & |l_2(\vec{z}) - P(\vec{z})| > \theta|l_2(\vec{z})| \\ \text{or} & |P(\vec{z}) - l_1(\vec{z})| > (4d-4)\theta|l_1(\vec{z})| \end{array}\right] \\ &\leq 2\delta_2 + \delta_1.\end{aligned}$$

■

Alors comme précédemment, la robustesse et la continuité du test de multilinéarité relative peuvent être établies, et elles conduiront ensuite par le Théorème 1.1 à l'auto-testeur suivant.

Théorème 4.6. *Soit $0 \leq \theta \leq O(1/d^2)$. Alors pour tout $0 < \delta \leq O(1)$ il existe un auto-testeur T qui satisfait sur l'entrée $0 < \gamma < 1$ et pour tout programme $P : (D_{8n})^d \to \mathbb{R}$:*

- *si $\theta\text{-}\,\mathsf{Dist}^r_{D_n}(P, \mathcal{L}) \leq \delta$ alors $T^P(\gamma)$ retourne* `BON` *avec probabilité supérieure à $(1-\gamma)$,*
- *si $(C_1\theta)\text{-}\,\mathsf{Dist}^r_{D_n}(P, \mathcal{L}) \geq C_2\delta$ alors $T^P(\gamma)$ retourne* `MAUVAIS` *avec probabilité supérieure à $(1-\gamma)$,*

où $C_1 \stackrel{\text{déf}}{=} O(d)$ et $C_2 \stackrel{\text{déf}}{=} O(d)$, et en utilisant au plus $O(\ln(1/\gamma)/\delta)$ appels à P, comparaisons, décalages binaires, et additions.

Deuxième partie

Auto-test pour le calcul quantique

CHAPITRE 5

Modèle

Cette partie est une introduction au modèle du calcul quantique. Les notions mathématiques de la physique quantique nécessaires pour notre étude sont présentées comme axiomes. Pour une présentation plus physique on pourra consulter les notes de cours en ligne de Preskill [Pre98].

1. L'état quantique

Considérons l'ensemble S des états possibles d'un système classique. S peut être fini lorsque par exemple $S = \{0, 1\}$ pour les états d'un bit, ou $S = \{0, 1\}^n$ pour n bits, mais aussi infini pour les états possibles d'une machine de Turing. Puisque nous nous intéressons aux portes quantiques, S sera supposé fini et pour simplifier nous supposons $S = \{0, 1, \ldots, N-1\}$. Le formalisme qui suit servira à décrire les états possibles de n bits correspondant au cas $S = \{0, 1\}^n$, *i.e.* $N = 2^n$.

Nous considérons maintenant successivement les cas où ce système peut évoluer de manière probabiliste, quantique, ou les deux à la fois. S désigne les états classiques associés à ces évolutions, *i.e.* les états dans lequel un système peut se retrouver après observation.

Le système est dans un *état probabiliste* lorsqu'il est décrit par une distribution de probabilité sur S, soit un vecteur $v \in \mathbb{R}^S$ à coordonnées positives ou nulles et de somme 1 de sorte que la *probabilité d'observer* le système dans un état $i \in S$ est donnée par la coordonnée v_i. Par exemple, un *bit probabiliste* est un élément $(p, 1-p) \in \mathbb{R}^2$, avec $0 \leq p \leq 1$. L'évolution d'un système probabiliste est modélisée par une matrice réelle A de taille N dont les coefficients sont positifs ou nuls et de somme 1 sur chacune des colonnes. Une telle matrice est dite *positive* et *stochastique*. Après l'évolution A le nouvel état v' du système dépend de l'état précédent v par la relation $v' = Av$. Dans le cas de la corrélation de deux systèmes dont les états classiques sont respectivement décrits par S_1 et S_2, l'état probabiliste du système global est décrit par un vecteur $v \in \mathbb{R}^{S_1 \times S_2} = \mathbb{R}^{S_1} \otimes \mathbb{R}^{S_2}$. Intuitivement un état est séparé lorsque les deux systèmes n'interagissent pas, et enchevêtré sinon. Formellement, l'état est *séparé* si $v = v_1 \otimes v_2$, avec $v_1 \in \mathbb{R}^{S_1}$ et $v_2 \in \mathbb{R}^{S_2}$, et *enchevêtré* dans le cas contraire. Si l'évolution de v est elle-même séparée, alors la partie v_1 évolue selon A_1, et la partie v_2 selon A_2, de sorte que l'évolution de v est décrite par la matrice $A_1 \otimes A_2$ satisfaisant $A_1 \otimes A_2(v_1 \otimes v_2) \stackrel{\text{déf}}{=} (A_1 v_1) \otimes (A_2 v_2)$. Ces notions s'étendent pour plusieurs systèmes corrélés.

Lorsqu'un système est en *superposition quantique*, son *état quantique* est décrit par un vecteur normé de l'espace de Hilbert $(\mathbb{C}^S, \|\cdot\|_2)$. Dans la *notation de Dirac*, un tel vecteur est représenté par $|\psi\rangle$, son vecteur adjoint $|\psi\rangle^\dagger$ par $\langle\psi|$, le produit scalaire (resp. extérieur) entre $|\psi\rangle$ et $|\psi'\rangle$ par $\langle\psi|\psi'\rangle$ (resp. $|\psi\rangle\langle\psi'|$). Un *bit quantique*, ou *qubit*, est donc un élément $|b\rangle = \alpha|0\rangle + \beta|1\rangle$, avec $\alpha, \beta \in \mathbb{C}$ tels que $|\alpha|^2 + |\beta|^2 = 1$. L'évolution d'une superposition quantique est décrite par une matrice *unitaire* A de dimension N, *i.e.* qui préserve la norme, ou encore une matrice complexe telle que $A^\dagger A$ soit l'identité. La *mesure de von Neumann* d'un système consiste à transformer son état quantique en un état probabiliste de sorte que la *probabilité d'observer* $|\psi\rangle$ dans un état $i \in S$ après la mesure, est donnée par le carré du module de la coordonné $|\psi\rangle_i$, soit $|\langle i|\psi\rangle|^2$. Implicitement cette mesure est reliée à la *base canonique* $(|i\rangle)_{i \in S}$ de $\mathbb{C}^S$. Une mesure de von Neumann dans une autre base orthonormée est tout à fait envisageable. Alors l'observation de $|\psi\rangle \in \mathbb{C}^S$ par une *mesure de von Neumann*

dans une base orthonormée B de $\mathbb{C}^S$ produit avec probabilité $|\langle\varphi|\psi\rangle|^2$ l'état $|\varphi\rangle \in B$. Lorsque $S = \{0,1\}^n$, la base canonique $(|w\rangle)_{w\in\{0,1\}^n}$ est appelée *base de calcul*. Si $S = S_1 \times S_2$ représente la corrélation de deux systèmes, alors l'état quantique du système global est décrit par un vecteur $|\psi\rangle \in \mathbb{C}^{S_1\times S_2} = \mathbb{C}^{S_1} \otimes \mathbb{C}^{S_2}$. Comme pour les états probabilistes, les *états séparés* sont de la forme $|\psi_1\rangle \otimes |\psi_2\rangle$, et *enchevêtrés* sinon. Une évolution séparée sur S_1 et S_2 est alors de la forme $A_1 \otimes A_2$. Une mesure du premier système est possible. Ce nouvel état est à la fois probabiliste et quantique. Un nouveau modèle est alors nécessaire pour décrire tous les états possibles d'un système physique.

Les états quantiques représentant des superpositions quantiques sont dits *états purs*. L'état d'un système est dit *mélangé* lorsqu'il est une distribution de probabilité d'états purs. Une telle distribution est appelée *mélange*. Physiquement deux distributions de probabilité d'états purs peuvent représenter le même état physique. L'objet mathématique représentant fidèlement tout état physique est la matrice densité.

Définition. Une *matrice densité* ρ de taille N est une matrice complexe de dimension N telle que :
- ρ est hermitienne : $\rho^\dagger = \rho$,
- ρ est positive : $\forall|\psi\rangle \in \mathbb{C}^N, \langle\psi|\rho|\psi\rangle \geq 0$,
- ρ est de trace 1 : $\operatorname{Tr}\rho = 1$.

Tout état pur $|\psi\rangle$ est représenté par la matrice densité $\psi = |\psi\rangle\langle\psi|$, et tout mélange $\{(p_k, |\psi_k\rangle)\}$ par $\psi = \sum_k p_k|\psi_k\rangle\langle\psi_k|$. Par abus de notation, la matrice densité associée à l'état pur $|\psi\rangle$ sera notée ψ. Par exemple, les états purs $e^{i\gamma}|\psi\rangle$, pour $0 \leq \gamma < 2\pi$, ou les mélanges $\{(\frac{1}{2}, |0\rangle), (\frac{1}{2}, |1\rangle)\}$ et $\{(\frac{1}{2}, \frac{|0\rangle+|1\rangle}{2}), (\frac{1}{2}, \frac{|0\rangle-|1\rangle}{2})\}$ ont respectivement la même matrice densité.

Inversement, toute matrice densité ρ de taille N se diagonalise dans une base orthonormée de vecteurs propres $(|e_i\rangle)_{0\leq i<N}$ dont les valeurs propres associées $(\lambda_i)_{0\leq i<N}$ sont des réels positifs ou nuls de somme 1, et représente donc le mélange $\{(\lambda_i, |e_i\rangle) : 0 \leq i < N\}$, soit dit autrement $\rho = \sum_{i=0}^{N-1} \lambda_i |e_i\rangle\langle e_i|$. Les matrices densités représentant des états purs ont une caractérisation algébrique simple : ρ représente un état pur si et seulement si ρ a exactement les deux valeurs propres 0 avec multiplicité $N-1$ et 1 avec multiplicité 1, ou de manière équivalente si $\rho^2 = \rho$.

La matrice densité ρ d'un qubit s'écrit en particulier $\rho = p|0\rangle\langle 0| + (1-p)|1\rangle\langle 1| + \alpha|1\rangle\langle 0| + \alpha^*|0\rangle\langle 1|$, où $0 \leq p \leq 1$ et $\alpha \in \mathbb{C}$ sont tels que $|\alpha|^2 \leq p(1-p)$. Cette matrice densité représente un état pur lorsque son déterminant est nul, *i.e.* $|\alpha|^2 = p(1-p)$. Elle sera notée par la suite $\rho(p,\alpha)$:

$$\rho(p,\alpha) \stackrel{\text{déf}}{=} \begin{pmatrix} p & \alpha^* \\ \alpha & 1-p \end{pmatrix}.$$

Enfin lorsque la discussion portera sur la corrélation de plusieurs systèmes, nous emploierons comme précédemment le qualificatif *séparé* pour désigner un état dont la matrice densité s'écrit comme produit tensoriel des matrices densités de chaque état local, et celui d'*enchevêtré* dans le cas contraire.

2. Superopérateurs

2.1. Généralités. L'évolution de tout système physique est décrite par une transformation spécifique agissant sur les matrices densités représentant les états du système. De telles transformations agissant sur les opérateurs matrices densités sont exactement caractérisées par les superopérateurs complètement positifs, ou CPSO, définis ci-dessous.

Définition.
- Un *superopérateur* de taille N est une transformation linéaire sur $\mathbb{C}^{N\times N}$.
- Un *superopérateur positif* (PSO) est un superopérateur qui conserve l'espace des matrices densités.

– Un *superopérateur complètement positif* (CPSO) $\boldsymbol{G}$ est un superopérateur positif tel que pour tout entier $M \geq 1$, le superopérateur $\boldsymbol{G} \otimes \boldsymbol{I}_M$ est positif, où $\boldsymbol{I}_M$ désigne le superopérateur identité de taille M.

Tout PSO n'est pas nécessairement un CPSO. Par exemple, le PSO *transposition* $\boldsymbol{T}$ agissant sur un *qubit* défini par $\boldsymbol{T}(|i\rangle\langle j|) \stackrel{\text{déf}}{=} |j\rangle\langle i|$, pour $i, j \in \{0, 1\}$ n'est pas un CPSO. En effet, si ρ désigne la matrice densité caractérisant l'état pur $\frac{1}{2}(|0\rangle + |1\rangle)^{\otimes^2}$, alors $\boldsymbol{T} \otimes \boldsymbol{I}_2(\rho)$ admet la valeur propre -1.

2.2. Transformations unitaires. Le calcul quantique est fondé sur la possibilité de construire des CPSO particuliers, dits unitaires, préservant entre autre l'ensemble des états purs. Ces superopérateurs sont construits à partir des matrices unitaires comme suit.

Définition. Un superopérateur $\boldsymbol{G}$ de taille N est *unitaire* s'il existe une matrice unitaire A de taille N telle que

$$\forall V \in \mathbb{C}^{N \times N}, \qquad \boldsymbol{G}(V) = A^\dagger V A.$$

Par abus de notation, le superopérateur unitaire associé à la matrice unitaire A sera noté $\boldsymbol{A}$. Si $|\psi'\rangle$ représente $A|\psi\rangle$, alors le superopérateur unitaire $\boldsymbol{A}$ envoie l'état pur ψ sur l'état pur ψ'. Comme pour la représentation de Dirac des états purs, plusieurs matrices unitaires peuvent définir le même superopérateur unitaire. Si A est une matrice unitaire, alors pour tout réel $0 \leq \gamma < 2\pi$, les matrices unitaires $e^{i\gamma}A$ définissent le même superopérateur unitaire. Il suffit donc de considérer les matrices unitaires de $\mathrm{U}(N)/\mathrm{U}(1)$, où la notation $\mathrm{U}(M)$ désigne l'ensemble des matrices unitaires de taille M, pour tout entier $M \geq 1$.

2.3. Mesures. Les mesures sont une autre classe importante de CPSO (non unitaires). Elles décrivent les transformations physiques correspondant à une observation du système. Etant donnée une base orthonormée B de $\mathbb{C}^N$, la *mesure de von Neumann dans la base* B est le superopérateur $\boldsymbol{M}^B$ de taille N satisfaisant $\boldsymbol{M}^B(V) \stackrel{\text{déf}}{=} \sum_{|\psi\rangle \in B}(\langle\psi|V|\psi\rangle)|\psi\rangle\langle\psi|$, pour toute matrice complexe V de taille N. Nous dirons alors que la *probabilité d'observer* $|\psi\rangle \in B$ sur la matrice densité ρ est $\langle\psi|\rho|\psi\rangle$. Lorsque $B = \{|i\rangle : 0 \leq i < N\}$, la mesure de von Neumann est dite *canonique* et seulement notée $\boldsymbol{M}$. Elle est alors simplement définie par $\boldsymbol{M}(|i\rangle\langle i|) \stackrel{\text{déf}}{=} |i\rangle\langle i|$ et $\boldsymbol{M}(|i\rangle\langle j|) \stackrel{\text{déf}}{=} 0$, si $i \neq j$. Lorsque $S = \{0, 1\}^n$ et $N = 2^n$, la mesure de von Neumann canonique est appelée *mesure de von Neumann dans la base de calcul*. Dans le cas d'un seul qubit, cette mesure envoie la matrice densité $\rho(p, \alpha)$ vers la matrice densité $\rho(p, 0)$.

Une *mesure généralisée* est simplement une mesure de von Neumann précédée de l'application d'un CPSO. Toute mesure généralisée peut donc s'écrire de la forme $\boldsymbol{M} \circ \boldsymbol{G}$, où $\boldsymbol{G}$ est un CPSO quelconque.

3. Sphère de Bloch

L'état du qubit ainsi que ses transformations physiques ont une représentation beaucoup plus naturelle dans $\mathbb{R}^3$. Cette représentation est basée sur l'isomorphisme entre $\mathrm{U}(2)/\mathrm{U}(1)$ et $\mathrm{SO}(3)$, le groupe spécial des rotations sur $\mathbb{R}^3$. Elle nous permet de représenter les états purs comme des points de la sphère unitaire de $\mathbb{R}^3$, les états mélangés comme des points de la boule unitaire de $\mathbb{R}^3$, et les superopérateurs unitaires comme des rotations dans $\mathbb{R}^3$. Les CPSO en général sont alors simplement des transformations affines particulières conservant entre autre la boule unitaire de $\mathbb{R}^3$. Pour une justification de ce qui suit dans cette section, on pourra consulter le cours de Preskill [Pre98].

Plus précisément, la *boule de Bloch* $\mathcal{B}$ (resp. *sphère de Bloch* $\mathcal{S}$) est la boule unité (resp. sphère unité) de l'espace affine euclidien $\mathbb{R}^3$. Chaque point $\overline{u} \in \mathbb{R}^3$ détermine un vecteur de mêmes coordonnées qui sera aussi noté $\overline{u}$ par abus de notation. Le produit scalaire sur $\mathbb{R}^3$ entre deux vecteurs $\overline{u}$ et $\overline{v}$ sera noté $\overline{u} \odot \overline{v}$, et la norme $\|\overline{u}\|$.

Chaque vecteur $\overline{u} \in \mathbb{R}^3$ est aussi caractérisé par sa norme $r \geq 0$, sa latitude $0 \leq \theta \leq \pi$, et sa longitude $0 \leq \varphi < 2\pi$. La *latitude* est l'angle entre l'axe z et le vecteur $\overline{u}$, et la *longitude* l'angle entre l'axe x et la projection orthogonale de $\overline{u}$ sur le plan d'équation $z = 0$. Si $\overline{u} = (x, y, z)$ alors ces paramètres satisfont $x = r\sin\theta\cos\varphi$, $y = r\sin\theta\sin\varphi$, et $z = r\cos\theta$.

Les matrices densités d'un qubit sont envoyées dans $\mathcal{B}$ par la bijection affine

$$\rho(p, \alpha) \mapsto (2\,\mathrm{Re}(\alpha), 2\,\mathrm{Im}(\alpha), 2p - 1),$$

dont l'application inverse est définie par

$$(x, y, z) \mapsto \frac{1}{2}\left(I_2 + x\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} + y\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} + z\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}\right).$$

Les trois matrices intervenant sont appelées *matrices de Pauli*. Ces matrices ont la particularité de former avec l'identité une base réelle des matrices complexes hermitiennes de taille 2. Par abus de notation, le point de $\mathcal{B}$ associé à la matrice densité ρ de taille 2 sera noté $\overline{\rho}$.

Dans ce formalisme, les états purs sont caractérisés dans $\mathcal{B}$ par leur norme.

Proposition 5.1. *Une matrice densité ρ d'un qubit représente un état pur si et seulement si $\|\overline{\rho}\| = 1$.*

De plus, si $0 \leq \theta \leq \pi$ et $0 \leq \varphi < 2\pi$ sont respectivement la latitude et la longitude de $\overline{\psi} \in \mathcal{S}$, alors la matrice densité correspondante décrit l'état pur

$$|\psi\rangle = \cos(\theta/2)|0\rangle + \sin(\theta/2)e^{i\varphi}|1\rangle.$$

La Figure 5.1 illustre cette représentation. En particulier, deux états purs orthogonaux dans $\mathbb{C}^2$ se retrouvent diamétralement opposés dans $\mathbb{R}^3$. La notation suivante sera utilisée pour les six états purs portés par les trois axes de coordonnées x, y, et z :

$$|\zeta_x^+\rangle \stackrel{\text{déf}}{=} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \qquad |\zeta_y^+\rangle \stackrel{\text{déf}}{=} \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle) \qquad |\zeta_z^+\rangle \stackrel{\text{déf}}{=} |0\rangle$$
$$|\zeta_x^-\rangle \stackrel{\text{déf}}{=} \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \qquad |\zeta_y^-\rangle \stackrel{\text{déf}}{=} \frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle) \qquad |\zeta_z^-\rangle \stackrel{\text{déf}}{=} |1\rangle;$$

dont les coordonnées dans $\mathbb{R}^3$ satisfont $\overline{\zeta_x^\pm} = (\pm 1, 0, 0)$, $\overline{\zeta_y^\pm} = (0, \pm 1, 0)$, $\overline{\zeta_z^\pm} = (0, 0, \pm 1)$.

Toute matrice densité ρ d'un qubit pouvant s'écrire $\rho = p|\psi\rangle\langle\psi| + (1-p)|\psi^\perp\rangle\langle\psi^\perp|$, où $|\psi\rangle$ et $|\psi^\perp\rangle$ sont deux vecteurs orthonormés de $\mathbb{C}^2$, sa représentation $\overline{\rho}$ dans $\mathbb{R}^3$ (voir Figure 5.2) s'écrit simplement comme le barycentre des points $\overline{\psi}$ et $\overline{\psi^\perp}$ respectivement pondérés par p et $(1-p)$.

Tout PSO $\boldsymbol{G}$ agissant sur un qubit induit naturellement une transformation affine $\overline{\boldsymbol{G}}$ sur $\mathbb{R}^3$ conservant $\mathcal{B}$ et définie en $\overline{\rho}$ par $\overline{\boldsymbol{G}}(\overline{\rho}) \stackrel{\text{déf}}{=} \overline{\boldsymbol{G}(\rho)}$. De plus l'application $\boldsymbol{G} \mapsto \overline{\boldsymbol{G}}$ est un isomorphisme entre l'ensemble des PSO agissant sur un qubit et l'ensemble des applications affines de $\mathbb{R}^3$ conservant $\mathcal{B}$. Si de plus $\boldsymbol{G}$ est un CPSO, alors $\overline{\boldsymbol{G}}$ vérifie d'autres propriétés. Une de ses propriétés, exposée au Théorème 5.1, sera fondamentale pour notre travail. Elle exprime le fait qu'une transformation affine correspondant à un CPSO peut parfois être uniquement définie par l'image de trois points non alignés, alors que cette situation ne peut jamais se produire pour une application dont on sait uniquement qu'elle est affine. Enfin, $\boldsymbol{G}$ est un CPSO unitaire si et seulement si $\overline{\boldsymbol{G}}$ est une rotation de $\mathbb{R}^3$, *i.e.* l'application $A \mapsto \overline{\boldsymbol{A}}$ est un isomorphisme de $\mathrm{U}(2)/\mathrm{U}(1)$ dans $\mathrm{SO}(3)$.

Pour les réels $-\pi < \alpha \leq \pi$, $0 \leq \theta \leq \pi/2$, et $0 \leq \varphi < 2\pi$, nous noterons $R_{\alpha,\theta,\varphi}$ la transformation unitaire sur $\mathbb{C}^2$ définie par $R_{\alpha,\theta,\varphi}|\psi\rangle \stackrel{\text{déf}}{=} |\psi\rangle$ et $R_{\alpha,\theta,\varphi}|\psi^\perp\rangle \stackrel{\text{déf}}{=} e^{i\alpha}|\psi^\perp\rangle$, où $|\psi\rangle \stackrel{\text{déf}}{=} \cos(\theta/2)|0\rangle + \sin(\theta/2)e^{i\varphi}|1\rangle$ et $|\psi^\perp\rangle \stackrel{\text{déf}}{=} \sin(\theta/2)|0\rangle - \cos(\theta/2)e^{i\varphi}|1\rangle$. Ces transformations parcourent tout $\mathrm{U}(2)/\mathrm{U}(1)$ et correspondent dans $\mathbb{R}^3$ aux rotations d'angle α autour de l'axe

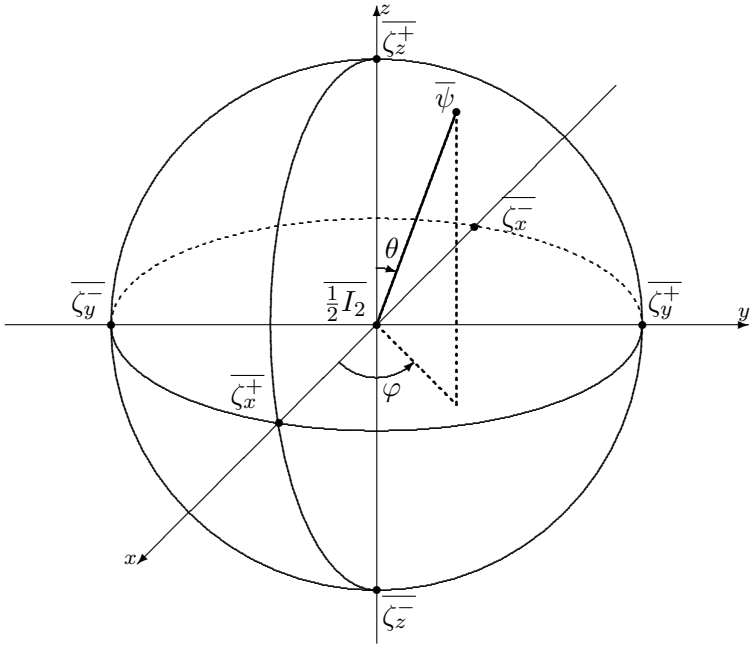

FIG. 5.1. Représentation d'un état pur sur la sphère de Bloch

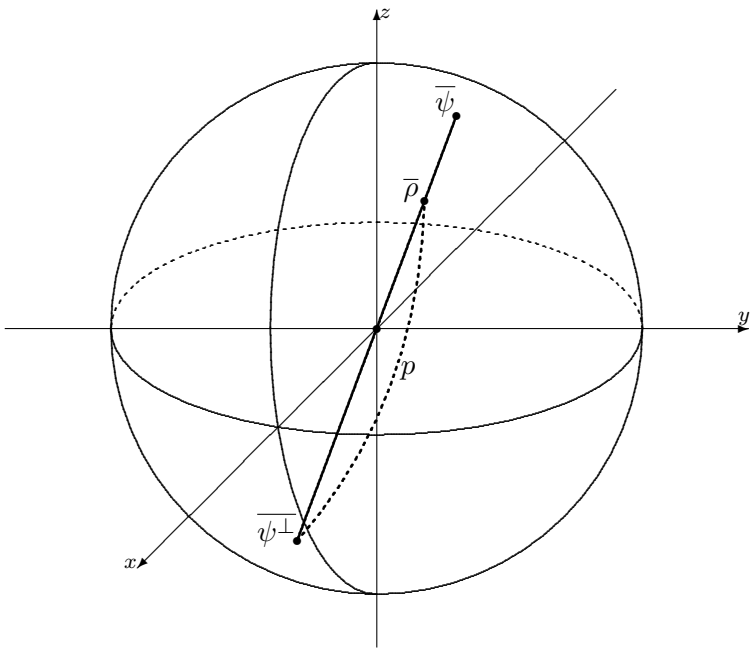

FIG. 5.2. Représentation d'une matrice densité dans la boule de Bloch

coupant $\mathcal{S}$ en $\overline{\psi}$ et $\overline{\psi^\perp}$. Lorsque $\theta = 0$, la transformation $R_{\alpha,0,\varphi}$ ne dépend pas de φ et nous la noterons R_α.

Distinguons quelques CPSO unitaires correspondant à quelques portes bien connues en calcul quantique. Pour tout réel $0 \leq \varphi < 2\pi$, soit NOT_φ la porte négation dans la base $(|0\rangle, e^{i\varphi}|1\rangle)$ définie comme la transformation unitaire de $\mathbb{C}^2$ telle que $\mathrm{NOT}_\varphi|0\rangle \stackrel{\text{déf}}{=} e^{i\varphi}|1\rangle$ et $\mathrm{NOT}_\varphi(e^{i\varphi}|1\rangle) \stackrel{\text{déf}}{=} |0\rangle$, et satisfaisant $\mathbf{NOT}_\varphi = \boldsymbol{R}_{\pi,\pi/2,\varphi}$; et soit H_φ la porte Hadamard dans

la même base telle que $H_\varphi|0\rangle \stackrel{\text{déf}}{=} \frac{|0\rangle+e^{i\varphi}|1\rangle}{\sqrt{2}}$ et $H_\varphi(e^{i\varphi}|1\rangle) \stackrel{\text{déf}}{=} \frac{|0\rangle-e^{i\varphi}|1\rangle}{\sqrt{2}}$, et satisfaisant $\boldsymbol{H}_\varphi = \boldsymbol{R}_{\pi,\pi/4,\varphi}$.

Enfin, le CPSO correspondant à la mesure de von Neumann du qubit dans la base de calcul peut être vu dans $\mathbb{R}^3$ comme la projection orthogonale sur l'axe z. Alors la probabilité d'observer 0 (resp. 1) sur la matrice densité ρ telle que $\overline{\rho} = (x, y, z)$ est $\frac{1+z}{2}$ (resp. $\frac{1-z}{2}$).

4. Norme

Afin de parler d'approximation de CPSO, nous devons définir une distance. Cette distance est définie par rapport à la norme trace, entre autre précédemment utilisée par Aharonov, Kitaev, et Nisan [AKN98] pour l'étude de la propagation des erreurs dans les circuits quantiques. La *norme trace* $\|\cdot\|_1$ est définie sur les matrices complexes de taille $N \geq 1$ par

$$\|V\|_1 \stackrel{\text{déf}}{=} \operatorname{Tr}\sqrt{V^\dagger V},$$

où $\sqrt{V^\dagger V}$ représente la matrice hermitienne positive A telle que $A^2 = V^\dagger V$. Une telle matrice A existe toujours car $V^\dagger V$ est toujours hermitienne positive. On peut aussi définir $\|V\|_1$ comme la somme des modules des valeurs propres de V, pour tout $V \in \mathbb{C}^{N\times N}$. En particulier, si ρ est une matrice densité, alors $\|\rho\|_1 = 1$.

Cette norme capture physiquement très bien la notion de distance entre deux matrices densités, et c'est pourquoi nous l'utiliserons. Etant donnée une mesure de von Neumann, ou généralisée, une matrice densité ρ induit une distribution d de probabilité sur la base de la mesure. La norme trace de la différence de deux matrices densités est la distance en variation maximale entre les deux distributions de probabilité induites, pour toutes les mesures généralisées possibles. Le maximum étant atteint en fait pour une mesure de von Neumann seule. Formalisons ceci en notant $d^{\boldsymbol{M}\circ\boldsymbol{G}}(\rho)$ la distribution de probabilité associée à chaque matrice densité ρ par la mesure généralisée $\boldsymbol{M} \circ \boldsymbol{G}$. La *distance en variation* entre deux distributions d et d' sur $\{0, \ldots, N-1\}$ est définie et notée par

$$|d - d'| \stackrel{\text{déf}}{=} \sum_{i=0}^{N-1} |d_i - d'_i|,$$

où d_i (resp. d'_i) dénote la probabilité de l'événement i par la distribution d (resp. d').

Proposition 5.2. *Soient ρ_1 et ρ_2 deux matrices densités de taille N. Soit $\boldsymbol{M}$ le CPSO de taille N correspondant à la mesure de von Neumann dans la base canonique. Alors*

$$\|\rho_1 - \rho_2\|_1 = \operatorname{Max}\{|d^{\boldsymbol{M}\circ\boldsymbol{G}}(\rho_1) - d^{\boldsymbol{M}\circ\boldsymbol{G}}(\rho_2)| : \boldsymbol{G} \text{ CPSO de taille } N\}.$$

En fait ce maximum est atteint pour la mesure de von Neumann dans la base des vecteurs propres de $(\rho_1 - \rho_2)$ ([AKN98, Lem. 11]).

Dans le cas du qubit, cette norme satisfait la propriété suivante.

Lemme 5.1. *Pour toutes matrices densités d'un qubit $\rho(p, \alpha)$ et $\rho(q, \beta)$,*

$$\|\rho(p,\alpha) - \rho(q,\beta)\|_1 = \|\overline{\rho(p,\alpha)} - \overline{\rho(q,\beta)}\| = 2\sqrt{(p-q)^2 + |\alpha-\beta|^2}.$$

Démonstration. Soit $V \stackrel{\text{déf}}{=} \rho(p,\alpha) - \rho(q,\beta)$. Alors

$$V = \begin{pmatrix} p-q & (\alpha-\beta)^* \\ \alpha-\beta & q-p \end{pmatrix}.$$

Calculons le produit $V^\dagger V$:

$$V^\dagger V = \begin{pmatrix} (p-q)^2 + |\alpha-\beta|^2 & 0 \\ 0 & |\alpha-\beta|^2 + (p-q)^2 \end{pmatrix}.$$

Donc le membre de gauche égale le membre de droit dans la double équation du lemme.

Calculons maintenant le membre du milieu :

$$\begin{aligned}\|\overline{\rho(p,\alpha)} - \overline{\rho(q,\beta)}\| &= \|(2\,\mathrm{Re}(\alpha), 2\,\mathrm{Im}(\alpha), 2p-1) - (2\,\mathrm{Re}(\beta), 2\,\mathrm{Im}(\beta), 2q-1)\| \\ &= 2\sqrt{(p-q)^2 + |\alpha-\beta|^2}.\end{aligned}$$

■

Nous aurons aussi recours à des propriétés plus générales énoncées dans la proposition suivante. La première partie est tirée de [AKN98, Lem. 10], et la deuxième est immédiate en remarquant que si $\|V\|_1$ est la somme des modules des valeurs propres de V, alors $\sqrt{\mathrm{Tr}(V^\dagger V)}$ est la somme des carrés des modules des mêmes valeurs propres.

Proposition 5.3. *Pour tout $V \in \mathbb{C}^{N\times N}$ et $W \in \mathbb{C}^{M\times M}$, la norme trace satisfait*

$$\|V \otimes W\|_1 = \|V\|_1 \|W\|_1 \text{ et } \sqrt{\mathrm{Tr}(V^\dagger V)} \leq \|V\|_1.$$

La norme trace induit une norme d'opérateur sur les superopérateurs. Cette norme notée $\|\cdot\|_\infty$ est définie pour tout superopérateur $\boldsymbol{G}$ par

$$\|\boldsymbol{G}\|_\infty \stackrel{\text{déf}}{=} \mathrm{Sup}\{\|\boldsymbol{G}(V)\|_1 : \|V\|_1 = 1\}.$$

Les CPSO sont contractants pour cette norme (voir [AKN98, Lem. 12]) :

Proposition 5.4. *Tout CPSO $\boldsymbol{G}$ satisfait $\|\boldsymbol{G}\|_\infty = 1$.*

La norme $\|\cdot\|_\infty$ est généralisée pour les k-uplets de superopérateurs par

$$\|(\boldsymbol{G}_1, \ldots, \boldsymbol{G}_k)\|_\infty \stackrel{\text{déf}}{=} \mathrm{Max}\{\|\boldsymbol{G}_1\|_\infty, \ldots, \|\boldsymbol{G}_k\|_\infty\}.$$

Nous noterons aussi $\mathsf{Dist}_\infty(\cdot,\cdot)$ la distance induite par cette norme.

Notons enfin qu'Aharonov, Kitaev, et Nisan [AKN98] ont mis en évidence un inconvénient de la norme $\|\cdot\|_\infty$. La norme d'un superopérateur peut croître lorsqu'il est tensorisé avec l'identité. Ce problème est sérieux lorsque l'étude porte sur des circuits quantiques à plusieurs qubits construits à partir de portes agissant localement sur peu de qubits. Pour résoudre ce problème, ils ont proposé une autre norme, notée $\|\cdot\|_\diamond$ et appelée *norme losange*, qui est multiplicative par rapport au produit tensoriel. Cette norme est définie par $\|\boldsymbol{G}\|_\diamond \stackrel{\text{déf}}{=} \|\boldsymbol{G} \otimes \boldsymbol{I}_N\|_\infty$, pour tout superopérateur $\boldsymbol{G}$ de taille N. La notion d'erreur induite $\|\cdot\|_\infty$ étant physiquement plus intéressante, nous exprimerons nos résultats avec cette norme et ne nous servirons de la norme $\|\cdot\|_\diamond$ que comme outil dans la preuve du Lemme 8.1. Toutefois, notons que ces normes restent équivalentes pour des superopérateurs de taille bornée. Or notre étude portant sur des superopérateurs agissant au plus sur deux qubits, nos résultats restent donc valides pour la norme $\|\cdot\|_\diamond$.

Lemme 5.2. *Pour tout superopérateur $\boldsymbol{G}$ de taille N,*

$$\|T\|_1 \leq \|T\|_\diamond \leq N\|T\|_1.$$

Démonstration. La majoration de gauche fait référence à [AKN98, Lem. 12].

Pour la majoration de droite, rappelons que $\|T\|_\diamond = \|\boldsymbol{G} \otimes \boldsymbol{I}_N\|_\infty$, et prenons un élément $V \in \mathbb{C}^N \otimes \mathbb{C}^N$ tel que $\|V\|_1 = 1$. Décomposons V dans la base canonique, alors :

$$V = \sum_{i,j,k,l=0}^{N-1} \lambda_{ijkl} (|i\rangle\langle j|) \otimes (|k\rangle\langle l|).$$

Mais $\sqrt{\mathrm{Tr}(V^\dagger V)}$, qui est majorée par 1 d'après Proposition 5.3, est aussi égale à la somme $\sum_{ijkl} |\lambda_{ijkl}|^2$. Donc par une inégalité de Cauchy-Schwartz, on déduit la majoration

$$\sum_{ijkl} |\lambda_{ijkl}| \leq N.$$

La preuve se termine en observant que pour chaque i, j, k, l :

$$\begin{aligned}\|(\boldsymbol{G}\otimes\boldsymbol{I}_N)((|i\rangle\langle j|)\otimes(|k\rangle\langle l|))\|_1 &= \|\boldsymbol{G}(|i\rangle\langle j|)\|_1\||k\rangle\langle l|\|_1\\ &\leq \|\boldsymbol{G}\|_\infty.\end{aligned}$$

Donc pour tout $V\in\mathbb{C}^N\otimes\mathbb{C}^N$ tel que $\|V\|_1=1$, nous avons

$$\|(\boldsymbol{G}\otimes\boldsymbol{I}_N)(V)\|_1\leq N\|G\|_\infty.$$

■

5. Propriétés des CPSO

5.1. Propriétés générales aux PSO. Pour le cas du qubit, certaines propriétés des CPSO sont directement reliées à celles de transformations affines de $\mathbb{R}^3$ conservant $\mathcal{B}$ et sont donc aussi valables plus généralement pour les PSO.

Lemme 5.3. *Soit $f:\mathbb{R}^3\to\mathbb{R}^3$ une transformation affine telle que $f(\mathcal{B})\subseteq\mathcal{B}$. Alors f satisfait les deux propriétés suivantes :*

(a) $\forall\overline{u},\overline{v}\in\mathbb{R}^3,\qquad \|f(\overline{u})-f(\overline{v})\|\leq\|\overline{u}-\overline{v}\|$;

(b) *Si f n'est pas constante alors pour tout $\overline{u}\in\mathcal{B}$, $f(\overline{u})\in\mathcal{S}\implies\overline{u}\in\mathcal{S}$.*

Ce lemme se réécrit pour les PSO de la manière suivante.

Lemme 5.4. *Soit $\boldsymbol{G}$ un PSO agissant sur un qubit, et soient ρ et τ deux matrices densités d'un qubit. Alors :*

(a) $\|\boldsymbol{G}(\rho)-\boldsymbol{G}(\tau)\|_1\leq\|\rho-\tau\|_1$;

(b) *Si $\boldsymbol{G}$ n'est pas constant et $\boldsymbol{G}(\rho)$ représente un état pur, alors ρ représente aussi un état pur.*

Remarque. La première propriété est immédiate lorsque $\boldsymbol{G}$ est un CPSO puisqu'alors $\|\boldsymbol{G}\|_\infty=1$.

Prouvons maintenant le premier lemme qui implique le second.

Démonstration. (a) Soient deux éléments distincts $\overline{u},\overline{v}\in\mathbb{R}^3$. Puisque f est une application affine, nous avons

$$f(\overline{u})-f(\overline{v})=\|\overline{u}-\overline{v}\|f\left(\frac{\overline{u}-\overline{v}}{\|\overline{u}-\overline{v}\|}\right).$$

Mais $\frac{\overline{u}-\overline{v}}{\|\overline{u}-\overline{v}\|}$ est de norme 1, et par hypothèse $f(\mathcal{B})\subseteq\mathcal{B}$, donc

$$\|f(\overline{u})-f(\overline{v})\|=\|\overline{u}-\overline{v}\|\times\left\|f\left(\frac{\overline{u}-\overline{v}}{\|\overline{u}-\overline{v}\|}\right)\right\|\leq\|\overline{u}-\overline{v}\|.$$

(b) Prouvons la deuxième propriété par l'absurde. Supposons donc qu'il existe un élément $\overline{u}\in\mathcal{B}-\mathcal{S}$ tel que $\|f(\overline{u})\|=1$. Comme f n'est pas constante, il existe un autre élément $\overline{v}\in\mathcal{B}$ tel que $f(\overline{v})\neq f(\overline{u})$. Pour chaque réel $\varepsilon>0$, soit $\overline{w}_\varepsilon\stackrel{\text{déf}}{=}\overline{u}+\varepsilon(\overline{u}-\overline{v})$. Fixons un $\varepsilon>0$ tel que $\overline{w}_\varepsilon\in\mathcal{B}$. Un tel ε existe car d'après notre hypothèse de travail $\overline{u}$ est à l'intérieur de $\mathcal{B}$. L'affinité de f en $\overline{w}_\varepsilon$ donne

$$f(\overline{w}_\varepsilon)=f(\overline{u})+\varepsilon(f(\overline{u})-f(\overline{v})).$$

Donc en utilisant $\|f(u)\|=1$, la norme de $f(\overline{w}_\varepsilon)$ satisfait

$$\begin{aligned}\|f(\overline{w}_\varepsilon)\|^2 &= 1+2\varepsilon(f(\overline{u})-f(\overline{v}))\odot f(\overline{u})+\varepsilon^2\|f(\overline{u})-f(\overline{v})\|^2\\ &= 1+2\varepsilon(1-f(\overline{v})\odot f(\overline{u}))+\varepsilon^2\|f(\overline{u})-f(\overline{v})\|^2\\ &\geq 1+\varepsilon^2\|f(\overline{u})-f(\overline{v})\|^2\\ &> 1.\end{aligned}$$

Nous avons donc exhibé un élément $\overline{w}_\varepsilon \in \mathcal{B}$ tel que $f(\overline{w}_\varepsilon) \notin \mathcal{B}$, ce qui contredit l'hypothèse $f(\mathcal{B}) \subseteq \mathcal{B}$.

■

5.2. Propriétés spécifiques aux CPSO. Un premier fait surprenant des CPSO est qu'un CPSO admettant un inverse étant aussi un CPSO est alors nécessairement un CPSO unitaire. Une preuve de ce résultat est disponible sur le cours de Preskill [Pre98, Ch. 3, Sec. 8].

Proposition 5.5. *Soient $\boldsymbol{G}$ et $\boldsymbol{H}$ deux CPSO de taille N tels que $\boldsymbol{G} \circ \boldsymbol{H} = \boldsymbol{I}_N$. Alors $\boldsymbol{G}$ et $\boldsymbol{H}$ sont des CPSO unitaires.*

Une transformation affine de $\mathbb{R}^3$ est uniquement définie par les images de quatre points non coplanaires. Etonnement, trois points non alignés peuvent être suffisants si cette transformation correspond à un CPSO agissant sur un qubit. Couplée avec la Proposition 5.5, cette propriété des CPSO est fondamentale pour la suite. Ces deux résultats nous apprennent que si deux CPSO composés agissent comme l'identité en trois points bien choisis, alors ils sont tous les deux unitaires. Ce n'est pas tant le passage de quatre points à trois points qui est important mais plus l'absence de la condition de non-coplanarité. Ainsi même si les itérés des états classiques $\zeta_z^\pm$ par une porte Hadamard sont coplanaires dans $\mathcal{B}$, il sera possible de caractériser une telle porte uniquement en fonction de ses itérés sur les états classiques. Voici donc ce résultat généralisé pour n qubits.

Théorème 5.1. *Soient ρ_1, ρ_2, et ρ_3 trois matrices densités distinctes représentant des états de qubits purs, et telles que le plan dans $\mathbb{R}^3$ contenant les points $\overline{\rho}_1$, $\overline{\rho}_2$, et $\overline{\rho}_3$ passe par le centre de $\mathcal{B}$. Si $\boldsymbol{G}$ est un CPSO agissant comme l'identité sur l'ensemble de n qubits $\{\rho_1, \rho_2, \rho_3\}^{\otimes^n}$ de cardinal 3^n, alors $\boldsymbol{G}$ est l'identité.*

Démonstration. Soit P le plan de $\mathbb{R}^3$ contenant les points $\overline{\rho_1}$, $\overline{\rho_2}$, et $\overline{\rho_3}$. Pour simplifier la preuve, supposons que $\overline{\zeta_z^\pm}$ et $\overline{\zeta_x^\pm}$ sont dans P. Chaque matrice densité ρ d'un qubit telle que $\overline{\rho} \in P$, est en faite une combinaison linéaire des matrices ρ_1, ρ_2, et ρ_3. Par conséquent, la linéarité du superopérateur $\boldsymbol{G}$ entraîne qu'il agit comme l'identité sur toutes les matrices densités de n qubits de l'ensemble $\{\rho : \overline{\rho} \in P\}^{\otimes n}$. Nous allons montrer que $\boldsymbol{G}$ agit comme l'identité sur les matrices densités séparées de n qubits, *i.e.* s'exprimant comme produit tensoriel de matrices densités d'un qubit. Les combinaisons linéaires finies de ces matrices engendrant toutes les autres matrices, le résultat sera alors démontré.

Pour ce, soient pour tout entier k l'ensemble A_k des matrices densité de k qubits séparées, et l'ensemble $B_k \stackrel{\text{déf}}{=} \{\zeta_x^\pm, \zeta_z^\pm\}^{\otimes k}$. Nous allons montrer par récurrence sur k, que pour tout $0 \leq k \leq n$, le CPSO $\boldsymbol{G}$ agit comme l'identité sur $A_k \otimes B_{n-k}$. Le cas $k = 0$ est vrai par hypothèse.

Supposons le résultat vrai pour un entier $0 \leq k \leq n-1$. Fixons alors $\sigma \in A_k$ et $\tau \in B_{n-k-1}$. Pour chaque matrice densité d'un qubit ρ, nous noterons $\tilde{\rho}$ la matrice densité de n qubits $\tilde{\rho} \stackrel{\text{déf}}{=} \sigma \otimes \rho \otimes \tau$. Nous prouvons maintenant que $G(\tilde{\rho}) = \tilde{\rho}$, pour tout $\rho \in A_1$. Pour ce, la possibilité d'écrire la matrice densité Ψ^+ représentant l'état enchevêtré $(|00\rangle + |11\rangle)/\sqrt{2}$, dit EPR, comme produit tensoriel des matrices $\zeta_\cdot^\pm$ sera utilisée :

$$\Psi^+ \quad = \quad \tfrac{1}{2}(\zeta_x^+ \otimes \zeta_x^+ + \zeta_x^- \otimes \zeta_x^- + \zeta_z^+ \otimes \zeta_z^+ + \zeta_z^- \otimes \zeta_z^-) - \tfrac{1}{2}(\zeta_y^+ \otimes \zeta_y^+ + \zeta_y^- \otimes \zeta_y^-).$$

Ceci se généralise immédiatement pour l'état pur $|\mu\rangle \stackrel{\text{déf}}{=} (|\tilde{0}\rangle|\tilde{0}\rangle + |\tilde{1}\rangle|\tilde{1}\rangle)/\sqrt{2}$:

$$\mu \quad = \quad \tfrac{1}{2}(\tilde{\zeta}_x^+ \otimes \tilde{\zeta}_x^+ + \tilde{\zeta}_x^- \otimes \tilde{\zeta}_x^- + \tilde{\zeta}_z^+ \otimes \tilde{\zeta}_z^+ + \tilde{\zeta}_z^- \otimes \tilde{\zeta}_z^-) - \tfrac{1}{2}(\tilde{\zeta}_y^+ \otimes \tilde{\zeta}_y^+ + \tilde{\zeta}_y^- \otimes \tilde{\zeta}_y^-).$$

Appliquons maintenant le CPSO $\boldsymbol{I}_{2^n} \otimes \boldsymbol{G}$ à la matrice densité μ :

$$(\boldsymbol{I}_{2^n} \otimes \boldsymbol{G})(\mu) \quad = \quad \tfrac{1}{2}[\tilde{\zeta}_x^+ \otimes \tilde{\zeta}_x^+ + \tilde{\zeta}_x^- \otimes \tilde{\zeta}_x^- + \tilde{\zeta}_z^+ \otimes \tilde{\zeta}_z^+ + \tilde{\zeta}_z^- \otimes \tilde{\zeta}_z^- - \tilde{\zeta}_y^+ \otimes \boldsymbol{G}(\tilde{\zeta}_y^+) - \tilde{\zeta}_y^- \otimes \boldsymbol{G}(\tilde{\zeta}_y^-)].$$

Posons alors pour tout $|\varphi\rangle$ et $|\varphi'\rangle$ représentant des états purs et orthogonaux de n qubit, $|\Phi^-_{\varphi\varphi'}\rangle \stackrel{\text{déf}}{=} (|\varphi\rangle|\varphi'\rangle - |\varphi'\rangle|\varphi\rangle)/\sqrt{2}$. Les vecteurs $|\Phi^-_{\varphi\varphi'}\rangle$ sons tous dans le cône isotrope de chaque matrice densité de la forme $\psi\otimes\psi$, où ψ représente un état pur de n qubits :

$$\langle\Phi^-_{\varphi\varphi'}|\psi\otimes\psi|\Phi^-_{\varphi\varphi'}\rangle = 0.$$

Inversement, on peut vérifier que toute matrice densité de $2n$ qubits de cône isotrope contenant tous les vecteurs $|\Phi^-_{\varphi\varphi'}\rangle$ est nécessairement de la forme $\psi\otimes\psi$, avec ψ représentant un état pur de n qubits.

Par conséquent, chaque vecteur $|\Phi^-_{\varphi\varphi'}\rangle$ satisfait

$$\langle\Phi^-_{\varphi\varphi'}|(\boldsymbol{I}_{2^n}\otimes\boldsymbol{G})(\mu)|\Phi^-_{\varphi\varphi'}\rangle \;=\; -\tfrac{1}{2}\langle\Phi^-_{\varphi\varphi'}|\tilde{\zeta}^+_y\otimes\boldsymbol{G}(\tilde{\zeta}^+_y)|\Phi^-_{\varphi\varphi'}\rangle - \tfrac{1}{2}\langle\Phi^-_{\varphi\varphi'}|\tilde{\zeta}^-_y\otimes\boldsymbol{G}(\tilde{\zeta}^-_y)|\Phi^-_{\varphi\varphi'}\rangle.$$

Puisque $\boldsymbol{G}$ est un CPSO, chacun des membres «$\langle\cdot|\cdot|\cdot\rangle$» devant être positifs ou nuls, ils sont en fait tous nuls, et donc $\boldsymbol{G}(\tilde{\zeta_y}^{\pm}) = \tilde{\zeta}^{\pm}_y$. Par conséquent, $\boldsymbol{G}$ agit comme l'identité sur les matrices densités $\tilde{\zeta}^{\pm}_z$, $\tilde{\zeta}^{\pm}_x$, et $\tilde{\zeta}^{\pm}_y$, dont les combinaisons linéaires finies génèrent toutes les matrices densités $\tilde{\rho}$. Finalement, $\boldsymbol{G}$ étant linéaire, il agit comme l'identité sur chacune des matrices densités $\tilde{\rho}$. ∎

Ce théorème admet une variante approchée pour le cas du qubit.

Théorème 5.2. *Soient ρ_1 et ρ_2 deux matrices densités distinctes représentant des états de qubit purs, et telles que les vecteurs $\overline{\rho}_1$ et $\overline{\rho}_2$ sont orthogonaux dans $\mathbb{R}^3$. Soit $0 \le \varepsilon \le 1$. Si $\boldsymbol{G}$ est un CPSO agissant sur un qubit tel que $\|\overline{\boldsymbol{G}}(\overline{\rho}) - \overline{\rho}\| \le \varepsilon$, pour $\overline{\rho} \in \{\pm\overline{\rho}_1, \pm\overline{\rho}_2\}$, alors $\|\boldsymbol{G} - \boldsymbol{I}_2\|_\infty \le 121\varepsilon$.*

Démonstration. Comme précédemment nous supposons pour simplifier que $\rho_1 = \zeta^+_x$ et $\rho_2 = \zeta^+_z$. Posons $\rho \stackrel{\text{déf}}{=} \boldsymbol{G}(\zeta^+_y)$ et $(x,y,z) \stackrel{\text{déf}}{=} \overline{\rho}$. Le Lemme 5.4(a) implique $\|\boldsymbol{G}(\zeta^+_z) - \rho\|_1 \le \|\zeta^+_z - \zeta^+_y\|_1 = \sqrt{2}$. Par hypothèse nous savons de plus que $\|\boldsymbol{G}(\zeta^+_z) - \zeta^+_z\|_1 \le \varepsilon$. Par conséquent $\|\zeta^+_z - \rho\|_1 \le \sqrt{2} + \varepsilon$. La même relation peut être écrite pour les trois autres matrices densités ζ^-_z, ζ^+_x, et ζ^-_x. Alors nécessairement, les coordonnées de $\overline{\rho}$ doivent satisfaire les quatre inégalités suivantes.

$$x^2 + y^2 + (z \pm 1)^2 \le (\sqrt{2} + \varepsilon)^2 \le 2 + 4\varepsilon \quad \text{et} \quad (x \pm 1)^2 + y^2 + z^2 \le 2 + 4\varepsilon. \tag{5.1}$$

D'autres restrictions sur (x,y,z) viennent du fait que $\boldsymbol{G}$ est un CPSO. Nous utilisons encore la décomposition de l'état EPR Ψ^+ :

$$\begin{aligned}(\boldsymbol{I}_2\otimes\boldsymbol{G})(\Psi^+) \;=\; & \tfrac{1}{2}(\zeta^+_x\otimes\boldsymbol{G}(\zeta^+_x) + \zeta^-_x\otimes\boldsymbol{G}(\zeta^-_x) + \zeta^+_z\otimes\boldsymbol{G}(\zeta^+_z) + \zeta^-_z\otimes\boldsymbol{G}(\zeta^-_z)) \\ & -\tfrac{1}{2}(\zeta^+_y\otimes\boldsymbol{G}(\zeta^+_y) + \zeta^-_y\otimes\boldsymbol{G}(\zeta^-_y)).\end{aligned}$$

En utilisant les hypothèses du théorème, la projection de cet état sur l'état pur enchevêtré et antisymétrique $|\Phi^-\rangle = (|01\rangle - |10\rangle)/\sqrt{2}$ donne

$$\langle\Phi^-|(\boldsymbol{I}_2\otimes\boldsymbol{G})(\Psi^+)|\Phi^-\rangle \;\le\; 2\varepsilon - \tfrac{1}{2}\langle\Phi^-|\zeta^+_y\otimes\boldsymbol{G}(\zeta^+_y)|\Phi^-\rangle - \tfrac{1}{2}\langle\Phi^-|\zeta^-_y\otimes\boldsymbol{G}(\zeta^-_y)|\Phi^-\rangle.$$

Puisque $\boldsymbol{G}$ est un CPSO, les termes «$\langle\cdot|\cdot|\cdot\rangle$» sont positifs ou nuls, et donc $0 \le \langle\Phi^-|\zeta^+_y\otimes\rho|\Phi^-\rangle \le 4\varepsilon$. Une étape calculatoire permet alors de montrer que cette relation implique que la coordonnée y doit satisfaire $1 - 16\varepsilon \le y \le 1$.

Cette dernière inégalité implique l'autre inégalité $y^2 \ge 1 - 32\varepsilon$, qui combinée avec les autres restrictions en (5.1), entraîne $(x \pm 1)^2 \le 2 + 4\varepsilon - y^2 - z^2 \le 1 + 36\varepsilon$, et similairement $(z \pm 1)^2 \le 1 + 36\varepsilon$. Alors nécessairement les coordonnées x et z de $\overline{\rho}$ satisfont $-18\varepsilon \le x, z \le 18\varepsilon$.

Ces bornes combinées à l'encadrement $1-16\varepsilon \le y \le 1$ donnent finalement $\|\boldsymbol{G}(\zeta^+_y) - \zeta^+_y\|_1 = \sqrt{x^2 + (y-1)^2 + z^2} \le \sqrt{904}\varepsilon$. La même majoration peut être démontrée pour ζ^-_y. Alors le Lemme 5.5 permet de conclure la preuve. ∎

Lemme 5.5. *Soit $\boldsymbol{G}$ un superopérateur agissant sur un qubit. Soit $\varepsilon \geq 0$ tel que $\|\boldsymbol{G}(\rho) - \rho\|_1 \leq \varepsilon$, pour $\rho \in \{\zeta_x^{\pm}, \zeta_y^{\pm}, \zeta_z^{\pm}\}$, alors $\|\boldsymbol{G} - \boldsymbol{I}_2\|_\infty \leq 4\varepsilon$.*

Démonstration. Toute matrice complexe V de taille 2 s'écrit

$$V \stackrel{\text{déf}}{=} \begin{pmatrix} a & b \\ c & d \end{pmatrix} = a\zeta_z^+ + d\zeta_z^- + \frac{b+c}{2}(\zeta_x^+ - \frac{1}{2}(\zeta_z^+ + \zeta_z^-)) + i\frac{b-c}{2}(\zeta_y^+ - \frac{1}{2}(\zeta_z^+ + \zeta_z^-)).$$

Les normes $\|\zeta_\cdot^\pm\|_1$ valant toutes 1, les hypothèses sur G entraînent

$$\|\boldsymbol{G}(V) - V\|_1 \leq 2\varepsilon(|a| + |b| + |c| + |d|).$$

Mais d'après la Proposition 5.3 $\sqrt{\mathrm{Tr}(V^\dagger V)} \leq \|V\|_1$. Or $\mathrm{Tr}(V^\dagger V) = |a|^2 + |b|^2 + |c|^2 + |d|^2$. Une inégalité de Cauchy-Schwarz permet donc d'établir $|a| + |b| + |c| + |d| \leq 2\|V\|_1$, ce qui conclut la preuve. ∎

CHAPITRE 6

Auto-tester les fonctions probabilistes

L'auto-test de portes quantiques, *i.e.* vues comme des CPSO, est relié à l'auto-test d'objets plus généraux que sont les fonctions probabilistes. Dans ce chapitre, des méthodes génériques de construction de tels auto-testeurs vont être exposées avant de revenir à l'auto-test quantique.

1. Caractérisation probabiliste : robustesse et continuité

Dans ce qui suit, une *fonction probabiliste f de D dans S* est une fonction définie sur D et à valeurs dans l'ensemble $\mathcal{D}(S)$ des distributions de probabilité sur S. Avec le formalisme du chapitre précédent, pour chaque $x \in D$, la distribution $f(x)$ est donc un état probabiliste sur S. A partir de maintenant le cas $S = \{0, 1, \ldots, N-1\}$ est considéré. On notera $\Pr[f(x) = i]$ la probabilité d'observer $i \in S$ sur $f(x)$. Alors *évaluer* f en x signifie tirer aléatoirement $i \in S$ selon la distribution de probabilité $f(x)$. Si f est une fonction probabiliste d'une certaine classe $\mathcal{C}$, on veut pouvoir décider si, pour une certaine pseudo-distance, f est proche d'une sous-classe $\mathcal{F}$ de $\mathcal{C}$. Intuitivement, une fonction probabiliste représentera l'oracle probabiliste associé aux portes quantiques testées par l'interface quantique.

En reprenant la démarche générale de l'auto-test, une caractérisation probabiliste basée sur des équations fonctionnelles particulières à la fois robustes et continues entraînent ensuite la possibilité d'auto-tester simplement les fonctions probabilistes.

Définition. Une *équation probabiliste* est une équation de la forme

$$\Pr[f(x) = i] = r, \tag{6.1}$$

où $0 \leq r \leq 1$ est un réel, $i \geq 0$ un entier, et f une variable représentant une fonction probabiliste. Le *terme de probabilité* désigne le membre de gauche de l'équation, et le *terme constant* celui de droite.

Une terminologie relative à ces équations doit maintenant être introduite. Pour tout réel $\eta \geq 0$, une fonction probabiliste f *η-satisfait* (6.1) si $|\Pr[f(x) = i] - r| \leq \eta$. Lorsque $\eta = 0$, on dira seulement que f *satisfait* (6.1). Ces notions de satisfaisabilité sont naturellement étendues à un système fini (E) d'équations probabilistes de sorte que f *η-satisfait* (E) si f η-satisfait chaque équation probabiliste de (E). Etant fixée une classe $\mathcal{C}$ de fonctions probabilistes, l'ensemble $\{f \in \mathcal{C} : f \text{ satisfait } (E)\}$ sera noté $\mathcal{F}_{(E)}(\mathcal{C})$, ou simplement $\mathcal{F}_{(E)}$ lorsque $\mathcal{C}$ est sous-entendu dans le contexte. Enfin, un ensemble $\mathcal{F}$ est *caractérisable* dans $\mathcal{C}$ s'il correspond à $\mathcal{F}_{(E)}(\mathcal{C})$ pour un certain système fini (E) d'équations probabilistes. Dans ce cas, nous dirons que (E) *caractérise* $\mathcal{F}$ dans $\mathcal{C}$. Inversement, un système fini d'équations probabilistes (E) est *satisfaisable* dans $\mathcal{C}$ si $\mathcal{F}_{(E)}(\mathcal{C}) \neq \emptyset$.

Les notions de robustesse et de stabilité définies dans la première partie se généralisent aisément à ces équations.

Définition. Soit (E) un système fini d'équations probabilistes satisfaisable. Le système (E) est (η, δ)-*continue* sur $\mathcal{C}$ pour la pseudo-distance Dist, si

$$\forall f \in \mathcal{C}, \qquad \mathsf{Dist}(f, \mathcal{F}_{(E)}(\mathcal{C})) \leq \eta \implies f \text{ } \delta\text{-satisfait } (E).$$

Définition. Soit (E) un système fini d'équations probabilistes satisfaisable. Le système (E) est (η, δ)-*robuste* sur $\mathcal{C}$ pour la pseudo-distance Dist, si

$$\forall f \in \mathcal{C}, \qquad f \ \delta\text{-satisfait } (E) \implies \mathsf{Dist}(f, \mathcal{F}_{(E)}(\mathcal{C})) \leq \eta.$$

2. Un auto-testeur générique

Dans cette section, nous montrons l'existence générique d'un auto-testeur pour les ensembles de fonctions probabilistes caractérisables. Pour cet auto-testeur l'interface est triviale. Cependant ce résultat s'appliquera à toute classe d'objets dont la classe des oracles probabilistes, pour une certaine interface, est caractérisable.

Théorème 6.1. *Soit (E) un système de d équations probabilistes caractérisant $\mathcal{F}$ sur $\mathcal{C}$. Soit $0 < \delta < 1/2$. Si (E) est $(\eta_1, \delta/2)$-continue sur $\mathcal{C}$ pour une pseudo-distance Dist_1, et $(\eta_2, 2\delta)$-robuste sur $\mathcal{C}$ pour une autre pseudo-distance Dist_2, alors il existe un $(\mathsf{Dist}_1, \eta_1; \mathsf{Dist}_2, \eta_2)$-auto-testeur de $\mathcal{F}$ sur $\mathcal{C}$ qui utilise, pour tout paramètre de confiance $0 < \gamma < 1$, au plus $O(d \ln(d/\gamma)/\delta^2)$ appels à la fonction probabiliste testée, incrémentations, comparaisons, et décalages binaires.*

Démonstration. Le testeur T consiste simplement en l'estimation des termes de probabilités de chaque équation probabiliste de (E) afin de les comparer aux termes constants associés. Cette estimation est effectuée par des évaluations successives de la fonction testée. Les bornes de Chernoff nous garantissent alors que cette estimation est bonne avec grande probabilité.

Plus précisément, soit $f \in \mathcal{C}$ représentant la fonction probabiliste testée. Considérons une équation probabiliste de (E) dont le terme de probabilité pour f vaut p et le terme constant r. Tout d'abord, on demande à T de connaître une approximation rationnelle $\tilde{r}$ de r telle que $|\tilde{r} - r| \leq \delta/4$. Ensuite, un argument de type bornes de Chernoff garantit que T peut calculer une estimation $\tilde{p}$ de p avec probabilité supérieure à $(1 - \gamma/d)$. tel que $|\tilde{p} - p| \leq \delta/4$ en échantillonnant $O(\ln(d/\gamma)/\delta^2)$ fois f (voir par exemple [McD98, Th. 2.1]). Alors T accepte si pour chaque équation probabiliste de (E), l'estimation $\tilde{p}$ satisfait $|\tilde{p} - \tilde{r}| \leq \delta$.

Si $\mathsf{Dist}_1(f, \mathcal{F}) \leq \eta_1$, alors la continuité de (E) implique que f $(\delta/2)$-satisfait (E), *i.e.* $|p - r| \leq \delta/2$, pour chaque équation probabiliste de (E), et $|\tilde{p} - \tilde{r}| \leq \delta$ avec probabilité supérieure à $(1 - \gamma/d)$. Donc T accepte f avec probabilité supérieure à $(1 - \gamma)$.

Si en revanche $\mathsf{Dist}_2(f, \mathcal{F}) > \eta_2$, alors la contraposée de la robustesse de (E) implique que f ne (2δ)-satisfait (E), *i.e.* $|p - r| > 2\delta$ pour une équation probabiliste de (E), et pour cette équation $|\tilde{p} - \tilde{r}| > 3\delta/2 > \delta$ avec probabilité supérieure à $(1 - \gamma/d)$. Donc T n'accepte pas f avec probabilité supérieure à $(1 - \gamma)$. ∎

3. Cas des portes quantiques

Pour auto-tester une porte quantique, vu comme CPSO, l'interface quantique introduite dans le chapitre d'introduction est utilisée. L'oracle probabiliste associé au CPSO $\boldsymbol{G}$ par cette interface est noté $\mathcal{O}[\boldsymbol{G}]$ et appelé oracle quantique. Les CPSO représentant certaines portes quantiques unitaires seront caractérisés par des systèmes finis d'équations probabilistes sur leurs oracles quantiques. Soient $0 \leq r \leq 1$ un réel, v et w deux mots de n bits, et C une description d'un circuit quantique [Deu89, AKN98] agissant sur n qubits et construit uniquement à partir de m paramètres, $\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m$, représentant des CPSO. Afin d'alléger les notations, une équation probabiliste de la forme

$$\Pr\left[\mathcal{O}[\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m](C, w) = v\right] = r,$$

sera représentée par *une équation probabiliste CPSO à m variables*,

$$\Pr^v[(C(\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m)(|w\rangle\langle w|)] = r.$$

La distance entre deux oracles quantiques, l'un défini à partir des CPSO $(\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m)$ et l'autre à partir de $(\boldsymbol{G'}_1, \ldots, \boldsymbol{G'}_m)$, est prise égale à $\mathsf{Dist}_\infty((\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m), (\boldsymbol{G'}_1, \ldots, \boldsymbol{G'}_m))$.

Dans le Chapitre 7, une série de systèmes d'équations probabilistes de CPSO caractérisant des ensembles de portes importants va être exhibée. Après une étude de la continuité et de la robustesse de ces caractérisations, nous pourrons établir l'existence d'auto-testeurs quantiques [DMMS99] pour ces ensembles au Chapitre 8.

CHAPITRE 7

Caractérisation des portes quantiques

Nous nous intéressons maintenant à caractériser certaines portes quantiques unitaires. En particulier, nous caractériserons un ensemble de triplets de portes dont chaque triplet est universel et tolérant à l'erreur.

1. Cas impossibles

Evidemment, le cas idéal consisterait à pouvoir décider avec grande probabilité si une porte donnée est proche de la porte souhaitée. En général ceci est impossible dans notre modèle de test. A chaque CPSO $\boldsymbol{G}$, il est possible d'associer un ensemble $\mathcal{G}$ de CPSO *indistinguables*, *i.e.* tel qu'aucun auto-test ne peut distinguer $\boldsymbol{G}$ d'un élément de $\mathcal{G}$. Pour préciser l'ensemble $\mathcal{G}$ introduisons quelques notations. La base de calcul du qubit, $(|0\rangle, |1\rangle)$, définit implicitement la base de calcul de n-qubit par $(|w\rangle \stackrel{\text{déf}}{=} |w_1\rangle \otimes \ldots \otimes |w_n\rangle : w \in \{0,1\}^n)$, puis celle canonique des matrices densités $(|w\rangle\langle w'| : w, w' \in \{0,1\}^n)$. Changer de base de calcul signifie considérer une nouvelle base orthonormée $(|\psi_0\rangle, |\psi_1\rangle)$ comme base de calcul du qubit. Alors la nouvelle base de calcul pour n qubits est $(|\psi_w\rangle \stackrel{\text{déf}}{=} |\psi_{w_1}\rangle \otimes \ldots \otimes |\psi_{w_n}\rangle : w \in \{0,1\}^n)$, et pour les matrices densités $(|\psi_w\rangle\langle\psi_{w'}| : w, w' \in \{0,1\}^n)$. Intuitivement, $|\psi_0\rangle$ remplace $|0\rangle$, et $|\psi_1\rangle$ remplace $|1\rangle$.

Pour tout CPSO $\boldsymbol{G}$ agissant sur n qubits (resp. matrice densité ρ de n qubits), la notation $\boldsymbol{G}^\varphi$ (resp. ρ^φ) désigne le CPSO $\boldsymbol{G}$ (resp. la matrice densité ρ) exprimé dans la nouvelle base de calcul obtenue par la substitution de $|1\rangle$ par $e^{i\varphi}|1\rangle$ ($|0\rangle$ restant inchangé) ; et $\boldsymbol{G}^*$ (resp. ρ^*) désigne le superopérateur $\boldsymbol{G}$ (resp. la matrice densité ρ) dans lequel chaque coefficient de sa matrice est remplacé par son conjugué. A tout CPSO $\boldsymbol{G}$, est associé l'ensemble $\mathcal{I}(\boldsymbol{G})$ définie par

$$\mathcal{I}(\boldsymbol{G}) \stackrel{\text{déf}}{=} \{\boldsymbol{G}^*, \boldsymbol{G}^\varphi : 0 \leq \varphi < 2\pi\}.$$

Cet ensemble vérifie la propriété annoncée.

Lemme 7.1. *Soient $\boldsymbol{G}$ un CPSO agissant sur n qubits, et $w \in \{0,1\}^n$. Alors pour tout $\boldsymbol{G'} \in \mathcal{I}(\boldsymbol{G})$,*

$$\boldsymbol{M} \circ \boldsymbol{G}(|w\rangle\langle w|) = \boldsymbol{M} \circ \boldsymbol{G'}(|w\rangle\langle w|),$$

où $\boldsymbol{M}$ désigne la mesure de von Neumann dans la base de calcul.

Démonstration. Fixons $v, w \in \{0,1\}^n$, et posons alors $\rho \stackrel{\text{déf}}{=} \boldsymbol{G}(|w\rangle\langle w|)$. Par définition nous avons, pour chaque réel $0 \leq \varphi < 2\pi$,

$$\begin{aligned} \rho^\varphi &= \boldsymbol{G}^\varphi(|w\rangle\langle w|^\varphi), \\ \rho^* &= \boldsymbol{G}^*(|w\rangle\langle w|^*). \end{aligned}$$

Mais les changements de base considérés n'affectent pas les matrices densités de la forme $|t\rangle\langle t|$, pour tout $t \in \{0,1\}^n$, ni les éléments (réels) de la diagonale de toute matrice densité. Donc $|w\rangle\langle w| = |w\rangle\langle w|^\varphi$, $|w\rangle\langle w| = |w\rangle\langle w|^*$, et si $\boldsymbol{M}$ est le CPSO associé à la mesure de von Neumann dans la base de calcul, alors $\boldsymbol{M} = \boldsymbol{M}^\varphi$ et $\boldsymbol{M} = \boldsymbol{M}^*$.

Donc finalement, $\boldsymbol{M}(\rho) = \boldsymbol{M}^\varphi(\rho^\varphi)$ et $\boldsymbol{M}(\rho) = \boldsymbol{M}^*(\rho^*)$, ce qui conclut la preuve. ■

Le lemme précédent a une interprétation géométrique reposant sur l'étude des degrés de liberté de la représentation du qubit dans $\mathcal{B}$.

Puisque seules les matrices densités construites à partir de produits tensoriels de $|0\rangle\langle 0|$ et $|1\rangle\langle 1|$ sont utilisées dans les équations probabilistes de CPSO, nous pouvons considérer que l'axe z est fixé, mais que les axes x et y ne le sont pas. Le choix de l'axe x correspond au choix de l'origine des latitudes, et celui de z au sens positif des rotations. Le sens positif des rotations revient aussi à choisir à laquelle des deux racines du polynôme X^2+1 est égal à «i».

Alors appliquer la transformation $\boldsymbol{G} \mapsto \boldsymbol{G}^{\varphi}$ revient à faire pivoter la boule $\mathcal{B}$ autour de l'axe z d'un angle φ, et appliquer $\boldsymbol{G} \mapsto \boldsymbol{G}^{*}$ à changer de sens positif de rotation autour de l'axe z.

Donnons maintenant une version de ce lemme pour les oracles quantiques, prouvant l'insdistinguabilité des CPSO de $\mathcal{I}(\boldsymbol{G})$.

Lemme 7.2. *Soient $\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m$ m superopérateurs agissant chacun sur un nombre fini de qubits. Alors pour tout $0 \leq \varphi < 2\pi$,*

$$\begin{aligned} \mathcal{O}[\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m] &= \mathcal{O}[\boldsymbol{G}_1^{\varphi}, \ldots, \boldsymbol{G}_m^{\varphi}], \\ \mathcal{O}[\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m] &= \mathcal{O}[\boldsymbol{G}_1^{*}, \ldots, \boldsymbol{G}_m^{*}]. \end{aligned}$$

Démonstration. Soit (C, w) une question à ces oracles quantiques. Nous allons montrer que leurs réponses définissent les mêmes distributions de probabilité. Posons le CPSO $\boldsymbol{G} \stackrel{\text{déf}}{=} C(\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m)$. Alors $\boldsymbol{G}$ satisfait pour tout $0 \leq \varphi < 2\pi$,

$$\begin{aligned} \boldsymbol{G}^{\varphi} &= C(\boldsymbol{G}_1^{\varphi}, \ldots, \boldsymbol{G}_m^{\varphi}), \\ \boldsymbol{G}^{*} &= C(\boldsymbol{G}_1^{*}, \ldots, \boldsymbol{G}_m^{*}). \end{aligned}$$

Le Lemme 7.1 permet de conclure que nécessairement, pour tout $0 \leq \varphi < 2\pi$:

$$\boldsymbol{M} \circ \boldsymbol{G}(|w\rangle\langle w|) = \boldsymbol{M} \circ \boldsymbol{G}^{\varphi}(|w\rangle\langle w|) = \boldsymbol{M} \circ \boldsymbol{G}^{*}(|w\rangle\langle w|).$$

■

Pour le cas des CPSO unitaires agissant sur un qubit, il vient naturellement, avec les observations précédentes, que $\boldsymbol{R}_{\alpha,\theta,0}^{\varphi} = \boldsymbol{R}_{\alpha,\theta,\varphi}$ et $\boldsymbol{R}_{\alpha,\theta,\varphi}^{*} = \boldsymbol{R}_{-\alpha,\theta,\varphi}$. Par conséquent, l'ensemble $\mathcal{I}(\boldsymbol{R}_{\alpha,\theta}, \varphi)$, noté $\mathcal{R}_{\alpha,\theta}$, vérifie

$$\mathcal{R}_{\alpha,\theta} = \{\boldsymbol{R}_{\pm\alpha,\theta,\varphi} : 0 \leq \varphi < 2\pi\}.$$

Pour les portes négations et Hadamard, nous posons

$$\mathcal{N} \stackrel{\text{déf}}{=} \mathcal{R}_{\pi,\pi/2} = \{\mathbf{NOT}_{\varphi} : 0 \leq \varphi < 2\pi\}$$

et

$$\mathcal{H} \stackrel{\text{déf}}{=} \mathcal{R}_{\pi,\pi/4} = \{\boldsymbol{H}_{\varphi} : 0 \leq \varphi < 2\pi\}.$$

2. Portes isolées à un qubit

Nous nous intéressons dans cette section à caractériser les ensembles $\mathcal{R}_{\alpha,\theta}$. Puisque le signe de α ne peut être testé, nous le supposerons encadré par $0 \leq \alpha \leq \pi$. De plus, nous ne considérerons que les superopérateurs unitaires tels que α/π est rationnel. Ce choix est raisonnable puisque ces superopérateurs sont denses dans l'ensemble des superopérateur unitaires agissant sur un qubit. Pour un tel superopérateur, soit $n_\alpha \stackrel{\text{déf}}{=} \text{Min}\{n \in \mathbb{N}\backslash\{0\} : n\alpha = 0 \mod 2\pi\}$. Le cas $n_\alpha = 1$ correspond au superopérateur identité.

Une caractérisation des ensembles $\mathcal{R}_{\alpha,\theta}$ est donnée dans le théorème suivant. Pour $\alpha = \pi$ et $\theta = \pi/4$, il caractérise entre autre l'ensemble $\mathcal{H}$ des portes Hadamard.

Théorème 7.1. *Soient $0 < \alpha \leq \pi$ et $0 < \theta \leq \pi/2$ tels que α/π est rationnel et $(\alpha, \theta) \neq (\pi, \pi/2)$. Alors le système d'équations probabilistes de CPSO suivant caractérise $\mathcal{R}_{\alpha,\theta}$:*

$$\Pr^0[\boldsymbol{G}^{n_\alpha}(|1\rangle\langle 1|)] = 0, \qquad \Pr^0[\boldsymbol{G}^{k}(|0\rangle\langle 0|)] = (1 + z_k(\alpha,\theta))/2, \quad k \in \{1, \ldots, n_\alpha\}; \tag{7.1}$$

où $z_k(\alpha, \theta) = \cos^2\theta + \sin^2\theta\cos(k\alpha)$.

Démonstration. Tout d'abord, observons que chaque CPSO dans $\mathcal{R}_{\alpha,\theta}$ satisfait les équations du théorème puisque la coordonnée en z des points $\overline{\boldsymbol{R}^k_{\alpha,\theta,\varphi}(|0\rangle\langle 0|)}$ est $z_k(\alpha,\theta)$ pour tout $0 \leq \varphi < 2\pi$.

Soit maintenant un CPSO $\boldsymbol{G}$ satisfaisant ces équations. Nous allons prouver que $\boldsymbol{G}$ est un CPSO unitaire. Alors le Lemme 7.3 impliquera que nécessairement $\boldsymbol{G} \in \mathcal{R}_{\alpha,\theta}$.

Par hypothèse sur α et θ, la coordonnée z_1 satisfait nécessairement $z_1(\alpha,\theta) \neq \pm 1$, et donc $\boldsymbol{G}(|0\rangle\langle 0|) \notin \{|0\rangle\langle 0|, |1\rangle\langle 1|\}$. Puisque $z_{n_\alpha}(\alpha,\theta) = 1$, on sait aussi que $\boldsymbol{G}^{n_\alpha}(|0\rangle\langle 0|) = |0\rangle\langle 0|$. Donc $\boldsymbol{G}$ n'est pas constant et le Lemme 5.4(b) implique alors par récurrence descendante que $\boldsymbol{G}(|0\rangle\langle 0|)$ représente un état pur. De plus $\boldsymbol{G}^{n_\alpha+1}(|0\rangle\langle 0|) = \boldsymbol{G}(|0\rangle\langle 0|)$, car $\boldsymbol{G}^{n_\alpha}(|0\rangle\langle 0|) = |0\rangle\langle 0|$. Les matrices densités $|0\rangle\langle 0|$, $|1\rangle\langle 1|$, et $\boldsymbol{G}(|0\rangle\langle 0|)$ représentent donc trois états purs distincts, et $\boldsymbol{G}^{n_\alpha}$ agit comme l'identité sur eux. Par le Théorème 5.1 $\boldsymbol{G}^{n_\alpha}$ est nécessairement l'identité partout. Puis la Proposition 5.5 permet de conclure que $\boldsymbol{G}$ est bien un superopérateur unitaire. ∎

Lemme 7.3. *Soient $0 < \alpha \leq \pi$, $0 < \theta \leq \pi/2$, $-\pi < \alpha' \leq \pi$, $0 \leq \theta' \leq \pi/2$ tels que α/π est rationnel. Si $z_k(\alpha,\theta) = z_k(\alpha',\theta')$ pour tout $k \in \{1,\ldots,n_\alpha\}$, alors $|\alpha'| = \alpha$ et $\theta' = \theta$.*

Démonstration. Par hypothèse, $z_1(\alpha,\theta) \neq 1$ et donc nécessairement $\theta' \neq 0$ et $\alpha' \neq 0$. Alors l'égalité $z_{n_\alpha}(\alpha',\theta') = 1$ entraîne $n_\alpha\alpha' = 0 \mod 2\pi$, *i.e.* il existe un entier $k \in \{1,\ldots,\lfloor n_\alpha/2 \rfloor\}$ tel que $\alpha' = \pm k\alpha$. Or il existe un polynôme de Chebishev T de degré k tel que $\cos(kx) = T(\cos(x))$, pour tout $x \in \mathbb{R}$. Donc l'égalité $z_k(\alpha,\theta) = z_k(\alpha',\theta')$ peut se réécrire en

$$\cos^2\theta + \sin^2\theta x = \cos^2\theta' + \sin^2\theta' T(x),$$

pour tout $x \in \{\cos\alpha, \ldots, \cos(n_\alpha\alpha)\}$. Le cardinal des valeurs que peut prendre x est exactement $1 + \lfloor n_\alpha/2 \rfloor$. Donc l'égalité précédente est en fait une égalité polynomiale en x, et le résultat se déduit par identification des coefficients des polynômes de chaque membre. ∎

Les ensembles $\mathcal{R}_{\alpha,\theta}$ restants pour lesquels α/π est rationnel, sont $\{\boldsymbol{R}_{-\alpha}, \boldsymbol{R}_\alpha\}$, pour $\alpha \in [0,\pi]$, et $\mathcal{N}$. Le lemme suivant établit qu'ils ne sont pas caractérisables isolément.

Lemme 7.4. *Il n'existe pas de systèmes finis d'équations probabilistes de CPSO à une variable caractérisant un des ensembles $\mathcal{R}_\alpha$, pour $\alpha \in [0,\pi]$, ou $\mathcal{N}$.*

Démonstration. Par définition les CPSO de $\mathcal{R}_\alpha$ (resp. $\mathcal{N}$) laissent fixe (resp. échangent) $|0\rangle\langle 0|$ et $|1\rangle\langle 1|$. Donc, suivis d'une mesure de von Neumann dans la base de calcul, ils définissent encore le même oracle quantique. Par conséquent, si $\boldsymbol{M}$ désigne cette mesure, on a pour tout CPSO $\boldsymbol{G} \in \mathcal{R}_\alpha$ (resp. $\boldsymbol{G} \in \mathcal{N}$),

$$\mathcal{O}[\boldsymbol{M} \circ \boldsymbol{G}] = \mathcal{O}[\boldsymbol{G}].$$

Mais $\boldsymbol{M}$ n'étant pas unitaire, il est clair que $\boldsymbol{M} \circ \boldsymbol{G}$ n'est pas dans $\mathcal{R}_\alpha$ (resp. $\mathcal{N}$). ∎

Néanmoins il est possible de les caractériser en même temps qu'une porte Hadamard. La section suivante développe cette approche.

3. Paires de portes à un qubit

L'ensemble $\mathcal{N}$ peut être caractérisé à l'aide de l'ensemble $\mathcal{H}$ de la manière suivante.

Théorème 7.2. *L'ensemble $\{(\boldsymbol{H}_\varphi, \mathbf{NOT}_\varphi) : 0 \leq \varphi < 2\pi\}$ est caractérisé par le système d'équations probabilistes de CPSO à deux variables $(\boldsymbol{F},\boldsymbol{G})$:*

$$\begin{array}{lll} \Pr^0[\boldsymbol{F}(|0\rangle\langle 0|)] = 1/2, & \Pr^0[\boldsymbol{F}^2(|0\rangle\langle 0|)] = 1, & \Pr^0[\boldsymbol{F}^2(|1\rangle\langle 1|)] = 0, \\ \Pr^0[\boldsymbol{G}(|0\rangle\langle 0|)] = 0, & \Pr^0[\boldsymbol{G}(|1\rangle\langle 1|)] = 1, & \Pr^0[\boldsymbol{F} \circ \boldsymbol{G}^2 \circ \boldsymbol{F}(|0\rangle\langle 0|)] = 1, \\ \Pr^0[\boldsymbol{F} \circ \boldsymbol{G} \circ \boldsymbol{F}(|0\rangle\langle 0|)] = 1. & & \end{array} \tag{7.2}$$

Démonstration. Observons d'abord que les paires $(\boldsymbol{F},\boldsymbol{G})$ de $\{(\boldsymbol{H}_\varphi,\mathbf{NOT}_\varphi) : \varphi \in [0,2\pi)\}$ satisfont le système du théorème.

Soient maintenant $\boldsymbol{F}$ et $\boldsymbol{G}$ deux CPSO satisfaisant ce système. Le CPSO $\boldsymbol{F}$ satisfait en particulier le système en (7.1) pour $\alpha = \pi$ et $\theta = \pi/4$, donc d'après le Théorème 7.1 il existe $0 \leq \varphi < 2\pi$ tel que $\boldsymbol{F} = \boldsymbol{H}_\varphi$. Par hypothèse, $\boldsymbol{G}^2$ agit comme l'identité sur les deux matrices densités $|0\rangle\langle 0|$ et $|1\rangle\langle 1|$. De plus $\boldsymbol{H}_\varphi \circ \boldsymbol{G}^2 \circ \boldsymbol{F}(|0\rangle\langle 0|) = |0\rangle\langle 0|$. Appliquons $\boldsymbol{H}_\varphi$ sur chacun des membres de cette dernière égalité. Alors puisque $\boldsymbol{H}_\varphi^2 = \boldsymbol{I}_2$, le CPSO $\boldsymbol{G}^2$ agit aussi comme l'identité sur $\boldsymbol{H}_\varphi(|0\rangle\langle 0|)$. Le Théorème 5.1 nous dit alors que $\boldsymbol{G}^2$ est l'identité, puis la Proposition 5.5 que $\boldsymbol{G}$ est un CPSO unitaire. L'axe de la rotation associée $\overline{\boldsymbol{G}}$ est nécessairement contenu dans le plan d'équation $z = 0$ puisque $\overline{|0\rangle\langle 0|}$ et $\overline{|1\rangle\langle 1|}$ sont échangés par $\overline{\boldsymbol{G}}$. De plus cet axe passe par $\overline{\boldsymbol{H}_\varphi(|0\rangle\langle 0|)}$ car d'après la dernière équation, $\boldsymbol{G}$ laisse invariant $\boldsymbol{H}_\varphi(|0\rangle\langle 0|)$. Le CPSO $\boldsymbol{G}$ est donc bien $\mathbf{NOT}_\varphi$. ∎

Voici maintenant la caractérisation des ensembles $\{\boldsymbol{R}_{-\alpha},\boldsymbol{R}_\alpha\}$, pour $0 \leq \alpha \leq \pi$ à l'aide de $\mathcal{H}$. Lorsque $\alpha = 0$, ce théorème caractérise le CPSO agissant comme l'identité sur un qubit avec les portes Hadamard.

Théorème 7.3. *Soit $0 \leq \alpha \leq \pi$. Si α/π est rationnel, alors l'ensemble $\mathcal{H} \times \{\boldsymbol{R}_{\pm\alpha}\}$ est caractérisé par le système d'équations probabilistes de CPSO à deux variables $(\boldsymbol{F},\boldsymbol{G})$:*

$$\begin{array}{l} \Pr^0[\boldsymbol{F}(|0\rangle\langle 0|)] = 1/2, \quad \Pr^0[\boldsymbol{F}^2(|0\rangle\langle 0|)] = 1, \quad \Pr^0[\boldsymbol{F}^2(|1\rangle\langle 1|)] = 0, \\ \Pr^0[\boldsymbol{G}(|0\rangle\langle 0|)] = 1, \quad \Pr^0[\boldsymbol{G}(|1\rangle\langle 1|)] = 0, \quad \Pr^0[\boldsymbol{F} \circ \boldsymbol{G}^{n_\alpha} \circ \boldsymbol{F}(|0\rangle\langle 0|)] = 1, \\ \Pr^0[\boldsymbol{F} \circ \boldsymbol{G} \circ \boldsymbol{F}(|0\rangle\langle 0|)] = (1+\cos\alpha)/2. \end{array} \tag{7.3}$$

Démonstration. Ce système est bien satisfait par l'ensemble de couples de CPSO $(\boldsymbol{F},\boldsymbol{G})$ dans $\mathcal{H} \times \{\boldsymbol{R}_{\pm\alpha}\}$.

Soient maintenant deux CPSO $\boldsymbol{F}$ et $\boldsymbol{G}$ satisfaisant ce système. De manière analogue à la preuve du Théorème 7.2, il existe un réel $0 \leq \varphi < 2\pi$ tel que $\boldsymbol{F} = \boldsymbol{H}_\varphi$, et $\boldsymbol{G}^{n_\alpha}$ est l'identité. Donc $\boldsymbol{G}$ est un CPSO unitaire. Puisque $\boldsymbol{G}$ laisse invariant $|0\rangle\langle 0|$ et $|1\rangle\langle 1|$, l'axe de sa rotation $\overline{\boldsymbol{G}}$ associée dans $\mathbb{R}^3$ est l'axe z. La dernière équation du système nous apprend que son angle de rotation α' satisfait $\cos\alpha' = \cos\alpha$, *i.e.* $|\alpha'| = \alpha$. ∎

4. La porte c-NOT

Le cas de plusieurs qubits est plus délicat. Nous n'envisagerons que la porte c-NOT. Ce choix est motivé par le fait qu'il suffit de savoir réaliser cette porte plus quelques autres agissant sur un seul qubit pour pouvoir simuler tout circuit quantique unitaire. Nous reviendrons sur ce point lors de la section suivante. Pour chaque $0 \leq \varphi < 2\pi$, l'application c-NOT$_\varphi$ désigne l'unique transformation unitaire de $\mathbb{C}^4$ telle que c-NOT$_\varphi(|0\rangle|\psi\rangle) \stackrel{\text{déf}}{=} |0\rangle|\psi\rangle$ et c-NOT$_\varphi|1\rangle|\psi\rangle \stackrel{\text{déf}}{=} |1\rangle\mathrm{NOT}_\varphi|\psi\rangle$, pour tout $|\psi\rangle \in \mathbb{C}^2$. L'ensemble $\mathcal{I}(\mathbf{c\text{-}NOT}_\varphi)$, pour chaque $0 \leq \varphi < 2\pi$, est l'ensemble $\{\mathbf{c\text{-}NOT}_{\varphi'} : 0 \leq \varphi' < 2\pi\}$. Comme pour l'ensemble $\mathcal{N}$, cet ensemble qui ne peut être testé seul le devient avec une porte Hadamard.

Théorème 7.4. *L'ensemble $\{(\boldsymbol{H}_\varphi, \mathbf{c\text{-}NOT}_\varphi) : 0 \leq \varphi < 2\pi\}$ est caractérisé par le système d'équations probabilistes de CPSO à deux variables $(\boldsymbol{F},\boldsymbol{K})$:*

$$\begin{array}{c} \Pr^0[\boldsymbol{F}(|0\rangle\langle 0|)] = 1/2, \quad \Pr^0[\boldsymbol{F}^2(|0\rangle\langle 0|)] = 1, \quad \Pr^0[\boldsymbol{F}^2(|1\rangle\langle 1|)] = 0, \\ \Pr^{00}[\boldsymbol{K}(|00\rangle\langle 00|)] = 1, \quad \Pr^{01}[\boldsymbol{K}(|01\rangle\langle 01|)] = 1, \\ \Pr^{11}[\boldsymbol{K}(|10\rangle\langle 10|)] = 1, \quad \Pr^{10}[\boldsymbol{K}(|11\rangle\langle 11|)] = 1, \\ \Pr^{00}[(\boldsymbol{I}_2 \otimes \boldsymbol{F}) \circ \boldsymbol{K} \circ (\boldsymbol{I}_2 \otimes \boldsymbol{F})(|00\rangle\langle 00|)] = 1, \\ \Pr^{10}[(\boldsymbol{I}_2 \otimes \boldsymbol{F}) \circ \boldsymbol{K} \circ (\boldsymbol{I}_2 \otimes \boldsymbol{F})(|10\rangle\langle 10|)] = 1, \\ \Pr^{00}[(\boldsymbol{F} \otimes \boldsymbol{I}_2) \circ \boldsymbol{K}^2 \circ (\boldsymbol{F} \otimes \boldsymbol{I}_2)(|00\rangle\langle 00|)] = 1, \\ \Pr^{01}[(\boldsymbol{F} \otimes \boldsymbol{I}_2) \circ \boldsymbol{K}^2 \circ (\boldsymbol{F} \otimes \boldsymbol{I}_2)(|01\rangle\langle 01|)] = 1, \\ \Pr^{00}[(\boldsymbol{F} \otimes \boldsymbol{F}) \circ \boldsymbol{K} \circ (\boldsymbol{F} \otimes \boldsymbol{F})(|00\rangle\langle 00|)] = 1. \end{array} \tag{7.4}$$

Démonstration. Ce système est bien satisfait par les couples de CPSO de l'ensemble annoncé.

Soient maintenant $\boldsymbol{F}$ et $\boldsymbol{K}$ deux CPSO satisfaisant ce système. Puisque $\boldsymbol{F}$ satisfait entre autre le système en (7.1) pour $\alpha = \pi$ et $\theta = \pi/4$, le Théorème 7.1 implique $\boldsymbol{F} = \boldsymbol{H}_\varphi$, pour un certain réel $0 \leq \varphi < 2\pi$. Posons $\rho = \boldsymbol{H}_\varphi(|0\rangle\langle 0|)$. D'après les équations restantes le CPSO $\boldsymbol{K}^2$ agit comme l'identité sur l'ensemble $\{|0\rangle\langle 0|, |1\rangle\langle 1|, \rho\}^{\otimes^2}$. Alors le Théorème 5.1 entraîne $\boldsymbol{K}^2 = \boldsymbol{I}_4$, puis par la Proposition 5.5 que $\boldsymbol{K}$ est un CPSO unitaire.

Nous montrons maintenant que $\boldsymbol{K} = \textbf{c-NOT}_\varphi$. Pour simplifier et quitte à remplacer $|1\rangle$ par $|1'\rangle \stackrel{\text{déf}}{=} e^{i\varphi}|1\rangle$, nous supposons maintenant $\varphi = 0$. Si $K \in \mathrm{U}(4)$ est une transformation unitaire associée à $\boldsymbol{K}$, alors puisque $\boldsymbol{K}$ agit comme l'identité sur $|00\rangle\langle 00|$, il existe un réel $0 \leq \gamma < 2\pi$ tel que $K|00\rangle = e^{i\gamma}|00\rangle$. Puisque $e^{-i\gamma}K$ est aussi une transformation unitaire associée à $\boldsymbol{K}$, nous supposons sans perte de généralités que $\gamma = 0$. Mais par hypothèse $\boldsymbol{K}$ agit aussi comme l'identité sur les matrices densités $|01\rangle\langle 01|$ et $|0\rangle\langle 0| \otimes \rho$. Donc nécessairement la linéarité de K entraîne $K|01\rangle = |01\rangle$.

De même, puisque $\boldsymbol{K}$ agit comme $\textbf{c-NOT}_0$ sur les matrices densités $|10\rangle\langle 10|$, $|11\rangle\langle 11|$, et $|1\rangle\langle 1| \otimes \rho$, il existe un réel $0 \leq \gamma' < 2\pi$ tel que $K|10\rangle = e^{i\gamma'}|11\rangle$ et $K|11\rangle = e^{i\gamma'}|10\rangle$.

Alors la dernière équation en (7.4), *i.e.* que $\boldsymbol{K}$ agit comme l'identité sur $\rho \otimes \rho$, entraîne

$$K(|00\rangle + |01\rangle + |10\rangle + |11\rangle) = e^{i\gamma''}(|00\rangle + |01\rangle + |10\rangle + |11\rangle),$$

pour un certain réel $0 \leq \gamma'' < 2\pi$. Donc la linéarité de K entaîne nécessairement $\gamma' = 0$, $\gamma'' = 0$, et $K = \text{c-NOT}_0$. ∎

5. Des portes universelles et tolérantes à l'erreur

L'existence d'ensembles universels de portes est fondamentale à la notion de circuit quantique. Un ensemble de portes est dit *universel* si tout circuit peut-être approximativement construit à partir des portes de cet ensemble. Contrairement aux circuits réversibles classiques, il existe pour les circuits quantiques (réversibles par définition) des ensembles de portes universels agissant au plus sur deux qubits [DiV95, Bar95, DBE95, Llo95, BBC$^+$95]. Un argument [Kit97] utilisant des algèbres et groupes de Lie permet de montrer qu'étant donné un ensemble universel de portes et un entier n, il existe un polynôme p_n tel que tout circuit agissant sur n qubits peut être simulé avec précision ε en utilisant moins de $p_n(\ln(1/\varepsilon))$ portes de l'ensemble universel. De plus la liste de ces portes peut être déterminée par une machine de Turing en temps $p_n(\ln(1/\varepsilon))$.

Mais l'universalité ne garantit pas la réalisabilité physique des circuits quantiques. Il faut de plus être en mesure de corriger tout phénomène d'erreur dû à l'imprécision des portes ou encore à la *décohérence* [Zur91], *i.e.* l'interaction avec l'environnement. Le problème de la protection de l'information quantique est plus difficile que son analogue classique car non seulement les états de base classiques doivent être protégés, mais aussi leurs superpositions. L'ensemble des états possibles de n qubits n'est donc pas seulement infini mais continu. De plus cloner un état étant impossible [WZ82], plusieurs travaux [Unr95, Lan95] ont tenté de justifier l'impossibilité de protéger ou de corriger l'information quantique au cours d'un calcul.

Cependant de récents travaux initiés par Shor [Sho96] tendent à prouver le contraire. Shor a montré comment il était possible de protéger de l'information quantique en utilisant des codes correcteurs quantiques introduits par Shor [Sho95], Calderbank et Shor [CS96], et Steane [Ste96]. En particulier chaque qubit est codé par un bloc de qubits, et les opérations qui devaient lui être appliquées sont alors effectuées sur ce bloc. Ainsi les données restent protégées par ces codes. Plus précisément, Shor montre pour un certain ensemble universel $\mathcal{E}$ de portes, comment réaliser chacune des opérations de $\mathcal{E}$ directement sur les blocs codants, et ce uniquement avec les opérations de $\mathcal{E}$. De tels ensembles sont appelés *universels et tolérants à l'erreur*. Ainsi il montre comment préserver l'information quantique à chaque étape d'un

circuit quantique, pourvu que le taux d'erreur sur chaque qubit reste polylogarithmiquement faible.

Récemment, Aharonov et Ben-Or [AB97] ont rendu ce résultat plus réaliste physiquement, en montrant l'existence d'un seuil d'erreur constant en dessous duquel il est toujours possible de réduire l'erreur globale. De plus, ils ont prouvé que ce seuil existait pour tout ensemble de portes universel [AB99].

Nous allons montrer qu'il est possible de caractériser un triplet de portes universel et tolérant à l'erreur. Ces triplets sont de la forme $(\boldsymbol{H}_\varphi, \boldsymbol{R}_{s\pi/4}, \textbf{c-NOT}_\varphi)$, où $0 \leq \varphi < 2\pi$ et $s = \pm 1$. P. Boykin, T. Mor, M. Pulver, V. Roychowdhury, et F. Vatan [BMP$^+$99] ont montré qu'ils étaient tous universels et tolérants à l'erreur. Un autre avantage de ces triplets est qu'ils correspondent à des portes déjà implémentées physiquement. Leur rôle est donc fondamental, tout comme la possibilité de les tester. Mais avant, les Théorèmes 7.3 et 7.4 nous apprennent qu'il est possible de les caractériser.

Corollaire 7.1. *L'ensemble* $\{(\boldsymbol{H}_\varphi, \boldsymbol{R}_{s\pi/4}, \textbf{c-NOT}_\varphi) : 0 \leq \varphi < 2\pi\,, s = \pm 1\}$ *est caractérisé par le système d'équations probabilistes de CPSO à trois variables* $(\boldsymbol{F}, \boldsymbol{G}, \boldsymbol{K})$ *constitué du système en* (7.3)*, pour* $\alpha = \pi/4$*, et du système en* (7.4)*.*

CHAPITRE 8

Auto-tester des portes quantiques

Les systèmes finis d'équations probabilistes de CPSO ont la particularité d'être toujours continus et robustes. Si la continuité dépend simplement de la taille du circuit invoqué dans chaque équation du système, la robustesse est plus complexe et nécessite une étude plus approfondie du système.

1. Continuité générique

Le *nombre de portes* d'un circuit C est défini comme le nombre de CPSO, apparaissant, éventuellement plusieurs fois, dans le circuit C. Par exemple si $\boldsymbol{G}$ est un CPSO, le circuit C défini par $C(\boldsymbol{G}) \stackrel{\text{déf}}{=} \boldsymbol{G}^3$ est un circuit à 3 portes.

La norme $\|\cdot\|_\diamond$ étant sous-multiplicative par composition et produit tensoriel, elle rend continue tout système d'équations probabilistes de CPSO. Elle peut s'exprimer dans un premier temps comme une version faible d'un résultat d'Aharonov, Kitaev, et Nisan sur le problème de la précision dans un circuit quantique [AKN98, Th. 4].

Proposition 8.1. *Soit $\varepsilon > 0$. Soient C un circuit quantique, et C' le circuit obtenu à partir de C en remplaçant éventuellement chaque porte par une autre à distance inférieure à ε, pour la norme $\|\cdot\|_\diamond$. Si L est le nombre de portes de C, alors les CPSO associés à C et C' sont à distance au plus $L\varepsilon$ pour la norme $\|\cdot\|_\diamond$.*

Puisque les normes $\|\cdot\|_\diamond$ et $\|\cdot\|_\infty$ sont équivalentes, ce résultat s'écrit en terme de continuité pour la distance Dist_∞.

Lemme 8.1. *Soit (E) un système fini d'équations probabilistes de CPSO satisfaisable tel que chaque équation utilise un circuit agissant sur au plus n qubits et dont le nombre de portes est au plus L. Alors pour tout $\eta \geq 0$, le système (E) est $(\eta, 2^n L\eta)$-continue.*

Démonstration. Fixons $\eta \geq 0$. Le système (E) est $(\eta, 2^n L\eta)$-continue si et seulement si chacune de ces équations l'est aussi. Fixons en une de la forme

$$\Pr^v[C(\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m)(|w\rangle\langle w|)] = r.$$

Soit alors $(\boldsymbol{G'}_1, \ldots, \boldsymbol{G'}_m)$ un m-uplet de CPSO à distance inférieure à η, pour la distance Dist_∞, d'un autre m-uplet $(\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m)$ satisfaisant cette équation. L'équivalence des normes $\|\cdot\|_\infty$ et $\|\cdot\|_\diamond$ explicitée au Lemme 5.2, et la Proposition 8.1 entraînent

$$\|C(\boldsymbol{G'}_1, \ldots, \boldsymbol{G'}_m) - C(\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m)\|_\infty \leq 2^n L\eta,$$

et donc

$$\|C(\boldsymbol{G'}_1, \ldots, \boldsymbol{G'}_m)(|w\rangle\langle w|) - C(\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m)(|w\rangle\langle w|)\|_1 \leq 2^n L\eta.$$

L'interprétation statistique de la norme $\|\cdot\|_1$ selon la Proposition 5.2 permet de conclure. ∎

2. Robustesse générique

Une vision algébrique des CPSO et de leur caractérisation par des équations probabilistes permet d'établir la robustesse de tout système fini satisfaisable d'équations probabilistes de CPSO. Le théorème suivant précise ceci.

Théorème 8.1. *Soit (E) un système fini d'équations probabilistes de CPSO satisfaisable. Alors il existe un entier $k \geq 1$ et un réel $C > 0$ tels que, pour tout $\delta \geq 0$, le système (E) est $(C\delta^{1/k}, \delta)$-robuste.*

La preuve faisant intervenir la structure des ensembles semi-algébriques, nous allons introduire quelques notions de géométrie algébrique réelle. Un ensemble *semi-algébrique* (réel) est un sous-ensemble de $\mathbb{R}^m$ défini par

$$X \stackrel{\text{déf}}{=} \{x \in \mathbb{R}^m : Q(x)\},$$

où Q est une combinaison booléenne finie d'expressions de la forme $P(x) > 0$, $P(x) < 0$, ou $P(x) = 0$, avec P un polynôme réel quelconque. Les ensembles semi-algébriques sont stables par réunions finies, intersections finies, et complémentation. Un des résultats fondamentaux sur ces ensembles est qu'ils sont aussi stables par projection. Il s'agit du théorème de Tarski-Seidenberg (voir par exemple [BR90, Th. 2.3.4]). Une conséquence de ce résultat est que l'utilisation des quantificateurs $\exists y \in Y$ et $\forall y \in Y$, pour tout ensemble semi-algébrique réel Y, est autorisée dans la définition d'un ensemble semi-algébrique.

Soit $X \subseteq \mathbb{R}^m$. Une fonction $f : X \to \mathbb{R}^{m'}$ est dite *semi-algébrique* si son graphe est un ensemble semi-algébrique. Naturellement la composition de deux fonctions semi-algébriques l'est aussi. Le théorème de Tarski-Seidenberg implique qu'une fonction définie sur $X \subseteq \mathbb{R}^m$, à valeurs réelles, et de la forme $x \mapsto \mathrm{Inf}\{f(x,y) : (x,y) \in X'\}$ (resp. $x \mapsto \mathrm{Sup}\{f(x,y) : (x,y) \in X'\}$, où $X' \subseteq \mathbb{R}^{m'}$ et $f : X' \to \mathbb{R}$ sont semi-algébriques, reste semi-algébrique (voir par exemple [Hör83, Cor. A.2.4]). En particulier la distance à un compact semi-algébrique est une fonction semi-algébrique continue. Une autre conséquence fondamentale du théorème de Tarski-Seidenberg, pour les fonctions semi-algébriques continues, est l'inégalité de Lojasiewicz (pour une preuve, voir [BR90, Prop. 2.3.11]).

Proposition 8.2 (Inégalité de Lojasiewicz)**.** *Soit $X \subseteq \mathbb{R}^m$ un ensemble compact et semi-algébrique. Soient $f, g : X \to \mathbb{R}$ deux fonctions semi-algébriques et continues. Si pour tout élément $x \in X$,*

$$f(x) = 0 \quad \Longrightarrow \quad g(x) = 0,$$

alors il existe un entier $k \geq 1$ et un réel $C > 0$ tels que pour tout $x \in X$,

$$|g(x)|^k \leq C|f(x)|.$$

Démontrons maintenant le Théorème 8.1, soit la robustesse générique des systèmes finis d'équations probabilistes de CPSO.

Démonstration. L'ensemble K_N des CPSO de taille N est compact et semi-algébrique lorsque $\mathbb{C}$ est identifié à $\mathbb{R}^2$, ce qui sera le cas tout au long de cette preuve. Une manière de le montrer est d'utiliser une des variantes de la représentation de Kraus des CPSO. Pour ce, l'opérateur *trace partielle*, Tr_2, est définie comme l'unique application linéaire sur $\mathbb{C}^{N^2} \otimes \mathbb{C}^{N^2}$ satisfaisant pour chaque $i, j = 1, \ldots, N^2$ et tout $V \in \mathbb{C}^{N^2}$,

$$\mathrm{Tr}_2(V \otimes |i\rangle\langle j|) \stackrel{\text{déf}}{=} \begin{cases} V & \text{si } i = j, \\ 0 & \text{sinon.} \end{cases}$$

Alors l'ensemble K_N des CPSO de taille N est caractérisé par [Pre98, Ch. 3, Sec 3] :

$$K_N = \{\boldsymbol{G} : \exists A \in \mathrm{U}(N^2), \quad \forall V \in \mathbb{C}^{N \times N},\ \boldsymbol{G}(V) = \mathrm{Tr}_2(A(V \otimes I_N)A^\dagger)\}.$$

L'espace $\mathrm{U}(N^2)$ étant compact et semi-algébrique, K_N l'est aussi.

Supposons maintenant que le système (E) est composé de m variables de CPSO agissant respectivement sur $n_1, \ldots, n_m$ qubits. Soient alors $K \stackrel{\text{déf}}{=} K_{2^{n_1}} \times \ldots K_{2^{n_m}}$, et $f : K \to \mathbb{R}$ la fonction faisant correspondre à chaque m-uplet $(\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m) \in K$ le maximum, sur toutes les équations de (E), des valeurs absolues des différences entre chaque terme de probabilité

et terme constant. La fonction f est continue et semi-algébrique car elle s'exprime comme le maximum d'un nombre fini de valeurs absolues de fonctions polynomiales en les coefficients (vus sur $\mathbb{R}$) de chaque $\boldsymbol{G}_i$, pour $i \in \{1, \ldots, m\}$. Enfin, par définition de f nous avons $f^{-1}(0) = \mathcal{F}_{(E)}$.

Puisque K est un compact semi-algébrique, nous avons vu que la fonction $g : K \to \mathbb{R}$ définie par

$$g((\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m)) \stackrel{\text{déf}}{=} \mathsf{Dist}_\infty((\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m), \mathcal{F}_{(E)}),$$

est aussi une fonction continue semi-algébrique.

Puisque, sur K, les zéros de f sont exactement ceux de g, la Proposition 8.2 permet de conclure. ∎

3. Une robustesse explicite

Par une étude minutieuse, il est possible de préciser la robustesse générique du chapitre précédent au cas par cas. Ici nous illustrons cette approche pour le système d'équations en (7.1) caractérisant les portes Hadamard pour $\alpha = \pi$ et $\theta = \pi/4$.

Théorème 8.2. *Pour tout $0 \leq \delta < 1/144$, le système en (7.1) pour $\alpha = \pi$ et $\theta = \pi/4$ est $(2299\sqrt{\delta}, \delta)$-robuste.*

Démonstration. Soit $\boldsymbol{G}$ un CPSO agissant sur un qubit et qui δ-satisfait le système de l'énoncé :

$$|\mathrm{Pr}^0[\boldsymbol{G}(|0\rangle\langle 0|)] - 1/2| \leq \delta, \tag{8.1}$$

$$|\mathrm{Pr}^0[\boldsymbol{G}^2(|0\rangle\langle 0|)] - 1| \leq \delta, \tag{8.2}$$

$$|\mathrm{Pr}^0[\boldsymbol{G}^2(|1\rangle\langle 1|)]| \leq \delta. \tag{8.3}$$

D'abord nous montrons qu'il existe un point $\overline{\rho} \in \mathcal{S}$ de coordonnée z nulle et dont la distance à $\overline{\boldsymbol{G}(|0\rangle\langle 0|)}$ est au plus $10\sqrt{\delta}$. Les inéquations (8.2) et (8.3) impliquent

$$\|\boldsymbol{G}^2(|b\rangle\langle b|) - |b\rangle\langle b|\|_1 \leq 3\sqrt{\delta}, \tag{8.4}$$

pour $b = 0, 1$. Donc $\|\boldsymbol{G}^2(|0\rangle\langle 0|) - \boldsymbol{G}^2(|1\rangle\langle 1|)\|_1 \geq 2 - 6\sqrt{\delta}$, soit encore d'après le Lemme 5.4(a)

$$\|\boldsymbol{G}(|0\rangle\langle 0|) - \boldsymbol{G}(|1\rangle\langle 1|)\|_1 \geq 2 - 6\sqrt{\delta}. \tag{8.5}$$

Mais $\|\boldsymbol{G}(|b\rangle\langle b|)\|_1 \leq 1$, pour $b = 0, 1$, et donc finalement pour $b = 0, 1$:

$$\|\overline{\boldsymbol{G}(|b\rangle\langle b|)}\| \geq 1 - 6\sqrt{\delta} > \frac{1}{2}. \tag{8.6}$$

Posons $\rho(p, \alpha) \stackrel{\text{déf}}{=} \boldsymbol{G}(|0\rangle\langle 0|)$, puis $\tau \stackrel{\text{déf}}{=} \rho(\frac{1}{2}, \alpha)$. Nécessairement $\alpha \neq 0$, sinon l'inéquation précédente n'est pas satisfaite. La contrainte en (8.1) entraîne alors

$$\|\overline{\tau} - \overline{\boldsymbol{G}(|0\rangle\langle 0|)}\| \leq 2\delta. \tag{8.7}$$

Soit ρ la matrice densité définie par $\rho \stackrel{\text{déf}}{=} \rho(\frac{1}{2}, \frac{\alpha}{2|\alpha|})$ satisfaisant aussi $\overline{\rho} = \overline{\tau}/\|\overline{\tau}\|$. La combinaison des deux inéquations (8.6) et (8.7) permet d'écrire

$$\|\boldsymbol{G}(|0\rangle\langle 0|) - \rho\|_1 \leq 10\sqrt{\delta}. \tag{8.8}$$

Le point $\overline{\rho}$ de $\mathcal{S}$ définit un unique réel $0 \leq \varphi < 2\pi$ tel que $\overline{\boldsymbol{H}_\varphi(|0\rangle\langle 0|)} = \overline{\rho}$. L'inégalité (8.5) entraîne

$$\|(\boldsymbol{G} - \boldsymbol{H}_\varphi)(|1\rangle\langle 1|)\|_1 \leq 16\sqrt{\delta}. \tag{8.9}$$

Ensuite puisque $\boldsymbol{H}_\varphi^2$ est l'identité, les inégalités (8.4) et (8.8) entraînent

$$\|(\boldsymbol{G} - \boldsymbol{H}_\varphi)(\boldsymbol{H}_\varphi|0\rangle\langle 0|)\|_1 \leq 13\sqrt{\delta},$$

et les inégalités (8.4) et (8.9),

$$\|(\boldsymbol{G} - \boldsymbol{H}_\varphi)(\boldsymbol{H}_\varphi|1\rangle\langle 1|)\|_1 \leq 19\sqrt{\delta}.$$

En conclusion, $\boldsymbol{H}_\varphi^{-1} \circ \boldsymbol{G}$ agit comme l'identité avec une erreur au plus $19\sqrt{\delta}$ sur les quatre matrices densités $|0\rangle\langle 0|$, $|1\rangle\langle 1|$, $\boldsymbol{H}_\varphi(|0\rangle\langle 0|)$, et $\boldsymbol{H}_\varphi(|1\rangle\langle 1|)$. Le Théorème 5.2 permet alors de conclure. ■

4. Bilan

Nous sommes donc capables en regroupant les caractérisations des Théorèmes 7.1, 7.2, 7.3, et 7.4, les résultats de continuité (Lemme 8.1) et de robustesse (Théorème 8.1), ainsi que l'auto-testeur générique du Théorème 6.1, d'établir l'existence d'auto-testeurs pour toute une série d'ensembles importants de portes. En particulier, nous exhibons un auto-testeur pour la famille de portes universelle et tolérante à l'erreur de Boykin, Mor, Pulver, Roychowdhury, et Vatan [BMP$^+$99].

Théorème 8.3. *Soit $\mathcal{F}$ désignant un des ensembles suivants :*
– $\mathcal{R}_{\alpha,\theta}$, *pour tout* $0 < \alpha \leq \pi$ *et* $0 < \theta \leq \pi/2$ *tels que* α/π *est rationnel et* $(\alpha, \theta) \neq (\pi, \pi/2)$,
– $\{(\boldsymbol{H}_\varphi, \mathbf{NOT}_\varphi) : 0 \leq \varphi < 2\pi\}$,
– $\mathcal{H} \times \{\boldsymbol{R}_{\pm\alpha}\}$, *pour tout* $0 \leq \alpha \leq \pi$ *tel que* α/π *est rationnel,*
– $\{(\boldsymbol{H}_\varphi, \mathbf{c\text{-}NOT}_\varphi) : 0 \leq \varphi < 2\pi\}$,
– $\{(\boldsymbol{H}_\varphi, \boldsymbol{R}_{s\pi/4}, \mathbf{c\text{-}NOT}_\varphi) : 0 \leq \varphi < 2\pi, s = \pm 1\}$.
Il existe un entier $k \geq 1$ et un réel $C > 0$ tels que, pour tout $0 < \delta < 1$, il existe un $(\mathsf{Dist}_\infty, \delta; \mathsf{Dist}_\infty, C\delta^{1/k})$-auto-testeur pour $\mathcal{F}$ qui, pour tout paramètre de confidence $0 < \gamma < 1$, utilise au plus $O(\ln(1/\gamma)/\delta^2)$ appels à l'oracle quantique, incrémentations, comparaisons, et décalages binaires.

Pour les portes Hadamard, le Théorème 8.2 explicite la robustesse des équations la caractérisant, et permet donc de préciser la qualité de l'auto-testeur précédent.

Théorème 8.4. *Pour tout $0 < \delta < 1/288$, il existe un $(\mathsf{Dist}_\infty, \delta/8; \mathsf{Dist}_\infty, 3252\sqrt{\delta})$-auto-testeur pour $\mathcal{H}$ qui, pour tout paramètre de confidence $0 < \gamma < 1$, utilise au plus $O(\ln(1/\gamma)/\delta^2)$ appels à l'oracle quantique, incrémentations, comparaisons, et décalages binaires.*

Bibliographie

[AB83] M. Albert and J. Baker. « Functions with bounded nth difference ». *Ann. Polonici Mathematici*, 43 :93–103, 1983.

[AB97] D. Aharonov and M. Ben-Or. « Fault-tolerant quantum computation with constant error ». In *Proc. 29th STOC*, 1997. Version finale dans [AB99].

[AB99] D. Aharonov and M. Ben-Or. « Fault-tolerant quantum computation with constant error rate ». Soumis à SIAM, 1999.

[ABCG93] S. Ar, M. Blum, B. Codenotti, and P. Gemmell. « Checking approximate computations over the reals ». In *Proc. 25th STOC*, pages 786–795, 1993.

[ADH97] L. Adleman, J. Demarrais, and M. Huand. « Quantum computability ». *SIAM J. Comp.*, 26(5) :1524–1540, 1997.

[AFKS99] N. Alon, E. Fischer, M. Krivelevich, and M. Szegedy. « Efficient testing of large graphs ». In *Proc. 40th FOCS*, pages 656–665, 1999.

[AKK95] S. Arora, D. Karger, and M. Karpinski. « Polynomial time approximation schemes for dense instances of NP-hard problems ». In *Proc. 27th STOC*, pages 284–293, 1995.

[AKN98] D. Aharonov, A. Kitaev, and N. Nisan. « Quantum circuits with miwed states ». In *Proc. 30th STOC*, pages 20–30, 1998.

[AKNS99] N. Alon, M. Krivelich, I. Newman, and M. Szegedy. « Regular languages are testable with a constant number of queries ». In *Proc. 40th FOCS*, 1999.

[ALM+92] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. « Proof verification and intractibility of approximation problems ». In *Proc. 33rd FOCS*, pages 14–23, 1992. Version finale dans [ALM+98].

[ALM+98] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. « Proof verification and intractibility of approximation problems ». *J. ACM*, 45(3) :505–555, 1998.

[Aro98] S. Arora. « The Approximability of NP-hard problems ». In *Proc. 30th STOC*, pages 337–348, 1998.

[AS92] S. Arora and S. Safra. « Probabilistic checkable proofs : A new characterization of NP ». In *Proc. 33rd FOCS*, pages 1–13, 1992.

[Bab85] L. Babai. « Trading group theory for randomness ». In *Proc. 17th STOC*, pages 421–429, 1985.

[Bab93] L. Babai. « Transparent (holographic) proofs ». In *Proc. 10th STACS*, volume 665, pages 525–534. LNCS, 1993.

[Bar95] A. Barenco. « A universal two-bit gate for quantum computation ». In *Proc. Roy. Soc. London*, volume 449 of *A*, pages 679–683, 1995.

[BB84] C. H. Bennett and G. Brassard. « Quantum cryptography : Public key distribution and coin tossing ». In *Proc. IEEE International Conference on Computers, Systems, and Signal Processing*, pages 175–179, 1984.

[BB92] A. Berthiaume and G. Brassard. « The quantum challenge to mag00tural complexity theory ». In *Proc. 7th Structure in Complexity Theory*, pages 132–137, 1992.

[BBC+95] A. Barenco, C. Bennett, R. Cleve, D. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. Smolin, and H. Weinfurter. « Elementary gates for quantum computation ». *Phys. Rev.*, 52 :3457–3467, 1995.

[BBJ+93] C. H. Bennett, G. Brassard, C. Crépeau R. Jozsa, A. Peres, and W. Wootters. « Teleporting an unknown quantum state via dual classical and EPR channels ». *Phys. Rev. Let.*, 70 :1895–1899, 1993.

[BCH+95] M. Bellare, D. Coppersmith, J. Håstad, M. Kiwi, and M. Sudan. « Linearity testing in characteristic two ». In *Proc. 36th FOCS*, pages 432–441, 1995.

[BFL91] L. Babai, L. Fortnow, and C. Lund. « Non-deterministic exponentional time has two-prower interactive protocols ». *Computational Complexity*, 1 :3–40, 1991.

[BK89] M. Blum and S. Kannan. « Designing programs that check their work ». In *Proc. 21st STOC*, pages 86–97, 1989. Version finale dans [BK95].

[BK95] M. Blum and S. Kannan. « Designing programs that check their work ». *J. ACM*, 42(1) :269–291, 1995.

[BLR90] M. Blum, M. Luby, and R. Rubinfeld. « Self-testing/correcting with applications to numerical problems ». In *Proc. 22nd STOC*, pages 73–83, 1990. Version finale dans [BLR93].

[BLR93] M. Blum, M. Luby, and R. Rubinfeld. « Self-testing/correcting with applications to numerical problems ». *J. Comp. and Syst. Sci.*, pages 549–595, 1993.

[BMP+99] P. Boykin, T. Mor, M. Pulver, V. Roychowdhury, and F. Vatan. « On universal and fault-tolerant quantum computing ». Technical Report, Quantum Physics e-Print archive, 1999. `http ://xxx.lanl.gov/abs/quant-ph/9906054`.

[BPM+97] D. Bouwmeester, J. W. Pan, K. Mattle, M. Eibl, H. Weinfurter, and A. Zeilinger. « Experimental quantum teleportation ». *Nature*, 390 :575–579, 1997.

[BR90] R. Benedetti and J.-J. Risler. *Real algebraic and semi-algebraic sets*. Hermann, 1990.

[BV97] E. Bernstein and U. Vazirani. « Quantum complexity theory ». *SIAM J. Comp.*, 26(5) :1411–1473, 1997.

[BW96] M. Blum and H. Wasserman. « Reflections on the Pentium Division Bug ». *IEEE Trans. Comp.*, 45(4) :385–393, April 1996.

[CFH97] D. Cory, A. Fahmy, and T. Havel. « Nuclear magnetic resonance spectroscopy : An experimentally accessible paradigm for quantum computing ». In *Proc. Nat. Acad. Sci. USA*, volume 94, pages 1634–1639, 1997.

[Cie59] Z. Ciesielski. « Some properties of convex functions of higher order ». *Ann. Polon. Math.*, 7 :1–7, 1959.

[CN97] I. L. Chuang and M. A. Nielsen. « Prescription for experimental determination of the dynamics of a quantum black box ». *J. Modern Optics*, 44 :732–744, 1997.

[CS96] A. Calderbank and P. Shor. « Good quantum error-correcting codes exist ». *Phys. Rev.*, 54 :1098–1105, 1996.

[CZ95] I. Cirac and P. Zoller. « Quantum computations with cold trapped ions ». *Phys. Rev. Let.*, 74 :4091–4094, 1995.

[DBE95] D. Deutsch, A. Barenco, and A. Ekert. « Universality in quantum computation ». In *Proc. Roy. Soc. London*, volume 449 of *A*, pages 669–677, 1995.

[Deu85] D. Deutsch. « Quantum theory, the Church-Turing principle and the universal quantum computer ». In *Proc. Roy. Soc. London*, volume 400 of *A*, pages 97–117, 1985.

[Deu89] D. Deutsch. « Quantum computational networks ». In *Proc. Roy. Soc. London*, volume 425 of *A*, pages 73–90, 1989.

[DiV95] D. DiVincenzo. « Two-bit gates are universal for quantum computation ». *Phys. Rev.*, 51 :1015–1022, 1995.

[DJ92] D. Deutsch and R. Jozsa. « Rapid solution of problems by quantum computation ». In *Proc. Roy. Soc. London*, volume 439 of *A*, pages 553–558, 1992.

[Djo69] D. Djoković. « A representation theorem for $(X_1 - 1)(X_2 - 1) \dots (X_n - 1)$ and its applications ». *Ann. Polonici Mathematici*, 22 :189–198, 1969.

[DMMS99] W. van Dam, F. Magniez, M. Mosca, and M. Santha. « Self-testing of universal and fault-tolerant sets of quantum gates ». Technical Report 1235, LRI, 1999.

[EKK+98] F. Ergün, S. Kannan, S. Ravi Kumar, R. Rubinfeld, and M. Viswanathan. « Spot-checkers ». In *Proc. 30th STOC*, pages 259–268, 1998.

[EKR96] F. Ergün, S. Ravi Kumar, and R. Rubinfeld. « Approximate checking of polynomials and functional equations ». In *Proc. 37th FOCS*, pages 592–601, 1996.

[Erg95] F. Ergün. « Testing multivariate linear functions : Overcoming the generator bottleneck ». In *Proc. 27th STOC*, pages 407–416, 1995.

[Fei93] J. Feigenbaum. « Locally random reductions in complexity theory ». In *Adv. in Comp. Complexity Theory*, volume 13 of *DIMACS Series on Disc. Math. and Theoret. Comp. Sci.*, pages 73–98. AMS, 1993.

[Fer96] W. Fernandez de la Vega. « MAX-CUT has a randomized approximation scheme in dense graphs ». *Random Structure and Algorithms*, 8 :187–198, 1996.

[Fey82] R. Feynman. « Simulating physics with computers ». *Internat. J. Theoret. Phys.*, 21 :467–488, 1982.

[FGL+96] U. Feige, S. Goldwasser, L. Lovász, S. Safra, and M. Szegedy. « Interactive proofs and the hardness of approximating cliques ». *J. ACM*, 43(2) :268–292, 1996.

[FHS94] K. Friedl, Z. Hátsági, and A. Shen. « Low-degree tests ». *Proc. 5th SODA*, pages 57–64, 1994.

[FK96] A. Frieze and R. Kannan. « The regularity lemma and approximation schemes for dense problems ». In *Proc. 37th FOCS*, pages 12–20, 1996.

[FK99] A. Frieze and R. Kannan. « Quick approximation to matrices and applications ». *Combinatorica*, 19(2) :175–220, 1999.

[For95] G. L. Forti. « Hyers-Ulam stability of functional equations in several variables ». *Aeq. Mathematicae*, 50 :143–190, 1995.

[FRS88] L. Fortnow, J. Rompel, and M. Sipser. « On the power of multi-prover interactive protocols ». In *Proc. 3rd Structure in Complexity Theory*, pages 156–161, 1988.

[GC97] N. Gershenfeld and I. Chuang. « Bulk spin-resonance quantum computation ». *Science*, 275 :350–356, 1997.

[GGR96] O. Goldreich, S. Goldwasser, and D. Ron. « Property testing and its connection to learning and approximation ». In *Proc. 37th FOCS*, pages 339–348, 1996.

[GLR+91] P. Gemmell, R. Lipton, R. Rubinfeld, M. Sudan, and A. Wigderson. « Self-testing/correcting for polynomials and for approximate functions ». In *Proc. 23rd STOC*, pages 32–42, 1991.

[GMR89] S. Goldwasser, S. Micali, and C. Rackoff. « The knowledge complexity of interactive proofs ». *SIAM J. Comp.*, 18 :186–208, 1989.

[GR97] O. Goldreich and D. Ron. « Property testing in bounded degree graphs ». In *Proc. 37th STOC*, pages 406–415, 1997.

[Gro96] L. Grover. « A fast quantum mechanical algorithm for database search ». In *Proc. 28th STOC*, pages 212–219, 1996.

[Hör83] L. Hörmander. *The analysis of linear partial diffential operators II*. Springer-Verlag, 1983.

[HR92] D. H. Hyers and T. M. Rassias. « Approximate homomorphisms ». *Aeq. Mathematicae*, 44 :125–153, 1992.

[HS92] D. H. Hyers and P. Semrl. « On the behaviour of mappings which do not satisfy Hyers-Ulam stability ». *Proc. AMS*, 144(4) :989–993, April 1992.

[Hye41] D. H. Hyers. « On the stability of the linear functional equation ». *Proc. Nat. Acad. Sci., U.S.A.*, 27 :222–224, 1941.

[Kem57] J. H. B. Kemperman. « A general functional equation ». *Trans. Amer. Math. Soc.*, 86 :28–56, 1957.

[Kit97] A. Kitaev. « Quantum computations : Algorithms and error correction ». *Russian Math. Surveys*, 52 :1191–1249, 1997.

[Kiw96] Marcos Kiwi. « *Probabilistically Checkable Proofs and the Testing of Hadamard-like Codes* ». PhD thesis, Massachusetts Institute of Technology, February 1996.

[KLZ96] E. Knill, R. Laflamme, and W. H. Zurek. « Accuracy threshold for quantum computation ». Technical Report, Quantum Physics e-Print archive, 1996. `http ://xxx.lanl.gov/abs/quant-ph/9611025`.

[KMS99] M. Kiwi, F. Magniez, and M. Santha. « Approximate testing with relative error ». In *Proc. 31st STOC*, pages 51–60, 1999.

[KS96] S. Ravi Kumar and D. Sivakumar. « Efficient self-testing/self-correction of linear recurrences ». In *Proc. 37th FOCS*, pages 602–611, 1996.

[Lan95] R. Landauer. « Is quantum mechanics useful ? ». *Phil. Trans. Roy. Soc. London*, 353 :367–376, 1995.

[LFKN92] C. Lund, L. Fortnow, H. Karloff, and N. Nisan. « Algebraic methods for interactive proof systems ». *J. ACM*, 39(4) :859–868, 1992.

[Lip91] R. Lipton. « New directions in testing ». *Series in Discrete Mathematics and Theoretical Computer Science*, 2 :191–202, 1991.

[Llo95] S. Lloyd. « Almost any quantum logic gate is universal ». *Phys. Rev. Let.*, 75 :346–349, 1995.

[Mag00] F. Magniez. « Multi-linearity self-testing with relative error ». In *Proc. STACS*, 2000.

[McD98] C. McDiarmid. « *Probabilistic Methods for Algorithmic Discrete Mathematics* », Chapitre Concentration, pages 195–248. Springer, 1998.

[MY98] D. Mayers and A. Yao. « Quantum cryptography with imperfect apparatus ». In *Proc. 39th FOCS*, pages 503–509, 1998.

[PCZ97] J. F. Poyatos, J. I. Cirac, and P. Zoller. « Complete characterization of a quantum process : The two-bit quantum gate ». *Phys. Rev. Let.*, 78 :390–393, 1997.

[Pre98] J. Preskill. « Quantum Information and Computation ». `http ://www.theory.caltech. edu/people/preskill/ph229/`, 1998.

[Ras78] T. M. Rassias. « On the stability of the linear mapping in Banach spaces ». *Proc. AMS*, 72(2) :297–300, November 1978.

[RS92] R. Rubinfeld and M. Sudan. « Robust characterizations of polynomials with applications to program testing ». In *Proc. 3rd SODA*, pages 23–32, 1992. Version finale dans [RS96].

[RS96] R. Rubinfeld and M. Sudan. « Robust characterizations of polynomials with applications to program testing ». *SIAM J. Comp.*, 25(2) :23–32, April 1996.

[Rub94] R. Rubinfeld. « On the robustness of functional equations ». In *Proc. 35th FOCS*, pages 288–299, 1994. Version finale dans [Rub94].

[Sha92] A. Shamir. « IP = PSPACE ». *J. ACM*, 39(4) :869–877, 1992.

[Sho94] P. Shor. « Algorithms for quantum computation : Discrete logarithm and factoring ». In *Proc. 26th STOC*, pages 124–134, 1994. Version finale dans [Sho97].

[Sho95] P. Shor. « Scheme for reducing decoherence in quantum computer memory ». *Phys. Rev.*, 52 :2493–2496, 1995.

[Sho96] P. Shor. « Fault-tolerant quantum computation ». In *Proc. 37th FOCS*, pages 56–65, 1996.

[Sho97] P. Shor. « Algorithms for quantum computation : Discrete logarithm and factoring ». *SIAM J. Comp.*, 26(5) :1484–1509, 1997.

[Sim94] D. Simon. « On the power of quantum computation ». In *Proc. 26th STOC*, pages 116–123, 1994. Version finale dans [Sim97].

[Sim97] D. Simon. « On the power of quantum computation ». *SIAM J. Comp.*, 26(5) :1474–1483, 1997.

[Sko83] F. Skof. « Sull'approssimazione delle applicazioni localmente δ-additive ». *Atti Acc. Sci. Torino*, 117 :377–389, 1983. in Italian.

[Ste96] A. Steane. « Multiple particle interference and quantum error correction ». *Proc. Roy. Soc. London*, 452 :2551–2577, 1996.

[Unr95] W. Unruh. « Maintening coherence in quantum computers ». *Phys. Rev.*, 51 :992–997, 1995.

[WZ82] W. Wootters and W. Zurek. « A single quantum cannote be cloned ». *Nature*, 299 :802, 1982.

[Yao93] A. Yao. « Quantum circuit complexity ». In *Proc. 34th FOCS*, pages 352–361, 1993.

[Zur91] W. H. Zurek. « Decoherence and the transition from quantum to classical ». *Physics today*, 10 :36–44, 1991.

Index