

SIMULATEURS CLASSIQUES DE CALCUL QUANTIQUES



Dr Frédéric MAUGUIERE

Ingénieur de recherche, Responsable Scientifique ROMEO

frederic.mauguiere@univ-reims.fr

<http://romeo.univ-reims.fr>



@HPCromeo

#romeoHPC

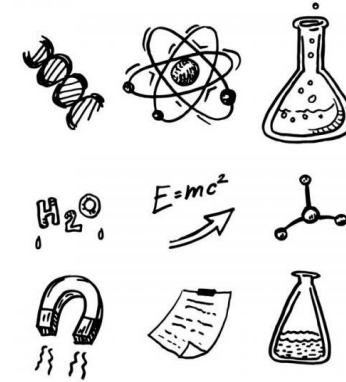
0

Le Centre de Calcul ROMEO :

Supercalculateur



MESoNET
La mésocentre des mésocentres



Recherche

+ new usage



Recherche & Innovation



REIMS - CHAMPAGNE ARDENNE
INSTITUT DE L'INTELLIGENCE ARTIFICIELLE EN SANTÉ



Enseignement



DEEP LEARNING INSTITUTE



HPC Center
ROMEO
Centre de Calcul Régional

Entreprises



EuroHPC
Competence Center



CENTRE DE COMPÉTENCE HPC.HPDA.IA



EUROPEAN TECHNOLOGY PLATFORM FOR HIGH PERFORMANCE COMPUTING

EDIH
Région Grand-Est



Simulations classiques de calculs quantiques



- Est-il possible de simuler le comportement d'un ordinateur quantique avec un ordinateur classique ?
 - ✓ **OUI** mais... ça nécessite beaucoup de mémoire et de temps de calcul
 - Exemple : pour simuler un problème à N qubits il faut stocker 2^N coefficients complexes
 - 1 complexe = 2 réels -> 2 nombres à double précision -> 2×8 octets
 - N=30, il faut ≈ 16 Gb de mémoire
 - ⇒ Besoin d'une machine HPC (beaucoup de mémoire, calcul multi-nœuds, accélérateurs (GPU, vectoriels))
- Comment ?
 - A grands coups de méthodes numériques (principalement de l'algèbre linéaire) : méthodes exactes et approchées
- A quoi ça sert ?
 - ✓ Tester et vérifier l'implémentation d'un algorithme quantique
 - ✓ Comprendre et valider le design d'un hardware quantique
 - ✓ Evaluer la complexité d'un algorithme quantique (est-il meilleur qu'un algo classique ?)

Simulations classiques de calculs quantiques

Quelques Frameworks de développement quantique (liste non exhaustive):

Nom	Développeur	langage	Open source	paradigme	HPC
Qiskit	IBM	python	✓	circuits	MPI, GPU
Cirq	Google	python	✓	circuits	GPU
ACQDP	Alibaba	python	✓	circuits	MPI, GPU
Intel-SQ (qHiPSTER)	Intel	C/C++, python	✓	circuits	MPI
QLM	Eviden (ATOS)	Python	Freeware (myQLM)	circuits, analog, annealing	Appliance (Qaptiva)
Qulacs	FUJITSU	C/C++, python	✓	circuits	MPI, GPU

EVIDEN (Atos) QLM:

Quantum Learning Machine:

- Mémoire : 1.5 To
- CPU : Intel(R) Xeon(R) Platinum 8160M CPU @ 2.10GHz
48 coeurs, 2 threads par coeurs
- Simulation \approx 30 qubits



Environnement de programmation myQLM

<https://atos.net/en/lp/myqlm>

- myQLM (gratuit, possibilité d'installer sur votre machine)
- myQLM Enterprise (payant, options avancées, installé sur la QLM de ROMEO)

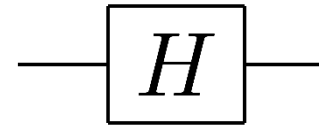
Démo sur QLM:

Construction d'un état intriqué (paire EPR, ou encore état de Bell):

Le but ici est de construire un état intriqué $|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$

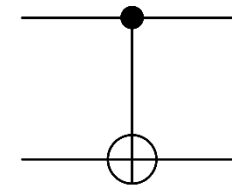
On va avoir besoin d'une porte Hadamard (H) et d'une porte Controlled Not (CNOT)

- Action de H sur 1 qubit : $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$
 $H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$

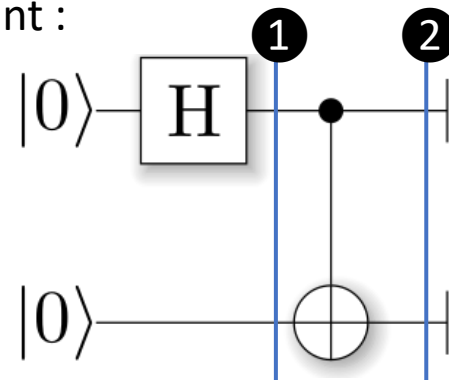


- Action de CNOT sur 2 qubit:

$$\begin{aligned} CNOT|00\rangle &= |00\rangle & CNOT|01\rangle &= |01\rangle \\ CNOT|10\rangle &= |11\rangle & CNOT|11\rangle &= |10\rangle \end{aligned}$$



On va construire le circuit suivant :



① $|\psi_1\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)$

② $|\psi_2\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$

Proba états finaux :

$$P(|00\rangle) = 0,5$$

$$P(|11\rangle) = 0,5$$

Offre de formation:

Session Etat de l'art

Organisation

- Intervention à distance (Teams) + présentiel
- Nombre de participants illimité
- Tout public (pas de prérequis)
- Environ 2:30 + Q&A

Contenu

- Bases de l'informatique quantique
- Technologies de qubits
- Grands défis technologiques
- Différents modèles de programmation
- Ecosystème
- Introduction à la programmation
- Cas d'usage et principaux modèles d'application

Session Basics

MESONET
le mésocentre des mésocentres

CCFR
CENTRE
DE COMPÉTENCE
HPC, HPDA, IA

Organisation

- Intervention à distance (Teams) + présentiel
- Nombre de participants limité
- Tout public (pas de prérequis)
- Durée : 1 ou 2 jours

Contenu

- Introduction à l'informatique quantique
- Présentation de l'émulateur Eviden : Qaptiva™
- Introduction à pyAQASM
- Pratique : codez votre premier programme Quantum impliquant superposition et intrication

- Présentation des outils avancés pour la programmation quantique sur Qaptiva™
- Illustration de réduction de complexité (Bernstein-Vazirani)
- Pratique
- Illustration d'une résolution par recuit d'un problème d'optimisation combinatoire : MaxCut
- Présentation des services de formation Qaptiva™

Offre de formation:

Session Advanced

Organisation






- Intervention en présentiel uniquement
- Nombre de participants limité
- Tout public (prérequis : avoir suivi la session Basics ou être familier des notions abordées)
- Durée : 3 jours

Contenu

- Overview of the Qaptiva environment
- Documentation and tutorials, finance and chemistry libraries
- Discovery of quantum algorithms (Grover algorithm, Shor algorithm, QUBO)
- Connection to QLM for advanced features
 - QLM Emulators
 - Circuit Optimization using QLM features
 - Noise modeling using QLM features
- Overview of specific and thematic trainings, and use case assessment services

Offre de formation:

Agenda des sessions:

-  • 14/12/2023 : session Etat de l'art, en visioconférence
-  • 21/12/2023 : session Basics, en visioconférence (français)
-  • 22/12/2023 : session Basics, en visioconférence (anglais)
-  • 16/01/2024 au 18/01/2024 : sessions Etat de l'art et Basics, en présentiel à Reims (URCA)
-  • 24/01/2024 : session Basics, en présentiel, à Paris (GENCI)
- 28/03/2024-29/03/2024 : session Basics + visite du laboratoire *Lumière, nanomatériaux, nanotechnologies*, Troyes (UTT)
- mai 2024 : session Advanced, en présentiel, Reims

Inscriptions : <https://www.mesonet.fr/>