INTERVIEW **QUANTUMCOMPUTER**

‘Van een qubit maak je niet even een reservekopietje’

Nu de eerste kleine exemplaren van de quantumcomputer worden gebouwd, is het tijd voor het schrijven van speciale software. Dat is precies waaraan Ronald de Wolf nu werkt.

Ronald de Wolf
FOTO OLIVIER MIDDENDORP

Door onze medewerker
Bruno van Wayenburg

Een nieuw type computer, die binnen seconden problemen uitrekent waar gewone computers miljoenen jaren over zouden doen. Dat is de belofte van de quantumcomputer, die listig gebruik maakt van de tegenintuïtieve wetten van de quantummechanica.

Hoogleraar theoretische informatica Ronald de Wolf is dagelijks bezig met het vervullen van die belofte. Maar in zijn werkkamer in het Amsterdam Science Park probeert hij eerst de hype wat in te dammen die de laatste jaren is ontstaan rond de quantumcomputer. „Veel mensen denken dat een quantumcomputer alle berekeningen sneller laat lopen, maar dat is een grote misvatting”, zegt hij.

De quantumcomputer boekt zijn wonderbaarlijke versnellingen bij een klein aantal heel specifieke rekentaken. Denk aan het kraken van de beveiligingsmethode RSA, die veel wordt gebruikt bij internetbankieren (pincode plus een steeds veranderende cijfercode). Of aan bepaalde logistieke problemen, of simulaties van complexe moleculen.

Sommige van die problemen hebben grote belangstelling van de chemische of logistieke industrie, of van inlichtingendiensten. Vandaar dat natuurkundelaboratoria over de hele wereld, onder andere aan de TU Delft, op zoek zijn naar de ideale fysieke manifestatie van de ‘qubit’, het basisonderdeel van de quantumcomputer.

De qubit is het equivalent van de gewone bit, de 1 of 0, in de gewone computer. Een qubit kan een elektron in een gekoeld diamantkristal zijn, een los gasatoom, of een minuscuul supergeleidend stroompje (zie illustratie en artikel hiernaast over werking van de quantumcomputer en over quantumalgoritmen).

Voorlopig is geen van de qubit-kandidaten al stabiel en manipuleerbaar genoeg om een quantumcomputer te bouwen. Maar als dat op een goede dag lukt, dient zich de volgende vraag aan: hoe schrijf je de algoritmen, ofwel de software, die de wonderbaarlijke snelheid van de quantumcomputer uitbuiten?

Voor het beantwoorden van die vraag krijgen De Wolf, zijn Amsterdamse collega Harry Buhrman en vier andere onderzoekers aan de TU Delft en de Universiteit Leiden een zogeheten Zwaartekracht-subsidie ter waarde van 18,8 miljoen euro voor ‘Quantumsoftware’. Dat geld gaat voor een deel naar quantumapparatuur, maar voor het grootste deel naar toekomstig personeel: universitair docenten, postdocs en promovendi.

Het is een bedrag dat de hoogleraren nu al aantoonbaar in de hoogste staat van activiteit bracht: van vier deelnemende quantum-algoritmische hoogleraren kon alleen Ronald de Wolf tijd maken voor een interview.

U werkt al sinds de jaren negentig aan quantumalgoritmen, wat is er nu veranderd?

„Normaal gesproken proberen we quantumalgoritmen te ontwikkelen voor de quantumcomputer van de verre toekomst, die misschien wel duizenden qubits heeft. Tot nu toe ging het ontwikkelen van fysieke quantumcomputers nogal langzaam. In 1997 al had je al een kleine quantumcomputer, gemaakt van een molecuul. Die had drie qubits, maar werkte niet zo goed. Nu, twintig jaar later, hebben de allerbeste quantumcomputers twintig qubits.

„Maar het lijkt erop dat er nu schot in zit. Een van de beste academische groepen, aan de University of California in Santa Barbara, overgenomen door Google, stelt dat ze eind dit jaar met een quantumcomputer met 50 qubits komen. Het Zwaartekracht-project is vooral bedoeld om te bekijken wat er technisch haalbaar is met zulke kleinere quantumcomputers.”

Wat is een quantumcomputer?

„Het idee van de quantumcomputer komt van natuurkundige Richard Feynman, die merkte hoeveel rekenkracht er ging zitten in het rekenen aan quantummechanische systemen, zoals moleculen of elementaire deeltjes. Zijn idee was: als we quantummechanische systemen gebruiken om quantummechanische systemen te simuleren, kunnen we het probleem als oplossing gebruiken. Dat is nog steeds een van de interessante beoogde toepassingen: het simuleren van complexe moleculen is een van de problemen die quantumcomputers veel sneller zouden moeten kunnen dan hun klassieke tegenhangers, de ‘gewone’ computers.

„Lang was het vakgebied een obscure bezigheid voor liefhebbers. Er waren wel quantumalgoritmen, maar die losten nogal kunstmatige problemen op (zie artikel hiernaast). Totdat in 1994 Peter Shor, onderzoeker bij Bell Laboratories, een verbazend snel quantumalgoritme vond voor priemfactorisatie: het vinden van de priemdelers van enorme getallen. Als je twee grote priemgetallen met elkaar vermenigvuldigt, krijg je een nóg groter getal, waarvan het heel lastig is om te bepalen wat de oorspronkelijke priemdelers zijn.

„‘Lastig’ betekent voor computerwetenschappers dat de rekentijd snel uit de hand loopt: als het invoergetal twee keer zo groot wordt, gaat de rekentijd in het kwadraat. Dit is de beruchte exponentiële toename, die er in de praktijk op neerkomt dat grote getallen niet te factoriseren zijn. Op die onmogelijkheid is de versleutelingsmethode RSA gebaseerd, die gebruikt wordt om boodschappen te versleutelen en om bankrekeningen te beschermen.

„Een quantumcomputer, liet Shor zien, zou priemfactorisatie in ‘kwadratische’ rekentijd kunnen oplossen. Dat wil zeggen dat het rekenen maar vier keer langer duurt als het invoergetal twee keer langer wordt. Dat zou in de praktijk betekenen dat de versleutelingsmethode waardeloos wordt. Shors algoritme was het eerste dat een praktisch probleem oploste.”

Zijn onze bankrekeningen nu nog wel veilig?

„Nou, voor een praktische toepassing van Shors algoritme op die schaal is vermoedelijk een quantumcomputer met honderdduizenden of miljoenen qubits nodig. Zover zijn we nog lang niet.”

Is het denkbaar dat partijen in het geheim al quantumcomputers gebruiken om RSA-versleutelde berichten te lezen?

„Denkbaar, maar niet erg waarschijnlijk. De eerste kandidaat zou de Amerikaanse National Security Agency NSA zijn. Uit de lekken van Edward Snowden uit 2013 kunnen we opmaken dat ze in ieder geval toen niet heel veel meer wisten dan de academische wereld. Ook China en Rusland zijn vermoedelijk nog niet zo ver.”

Shors algoritme is dus alleen praktisch omdat wij er met RSA-versleuteling een praktisch probleem van gemaakt hebben. Zijn er ook quantumalgoritmen die van zichzelf nuttig zijn?

„Dat klopt, priemfactoriseren is een heel specifiek probleem waarop bovendien een spectaculaire snelheidswinst geboekt kan worden. Maar in 1996 kwam de computerwetenschapper Lov Grover, ook van Bell Labs, met een quantumalgoritme om iets op te zoeken in een ongesorteerde lijst met gegevens. Dat is een subroutine die onderdeel is van veel andere algoritmen, bijvoorbeeld in navigatiesoftware. Zonder quantumcomputer is de opzoektijd evenredig aan de lengte van de lijst: wordt de lijst vier keer langer, dan duurt het opzoeken ook vier keer langer. Grover's quantumalgoritme doet dat ‘kwadratisch sneller’: bij een vier keer langere lijst duurt het opzoeken maar twee keer langer. Daarmee is Grovers algoritme is eigenlijk het tegenovergestelde van Shors algoritme: het versnelt de rekentijd niet heel spectaculair, maar is wel zeer breed toepasbaar.”

De algoritmen van Shor en Grover zijn inmiddels meer dan twintig jaar oud, is er sindsdien niets gebeurd?

„Er zijn wel allerlei dingen bij gekomen. Veel problemen uit de praktijk zijn optimalisatieproblemen: je wilt de kosten van bedrijfsprocessen of het gebruik van grondstoffen minimaliseren, of je winst of een bepaalde prestatiescore maximaliseren. Dat is echt een industrie: een flinke tak van de informatica is bezig om hier algoritmes voor te vinden. Neem bijvoorbeeld lineair programmeren, een bepaald type optimalisatieproblemen. Vorig jaar liet een paper zien hoe je lineair programmeren met quantumcomputers sneller op kunt lossen, en onze groep heeft daarna weer laten zien dat daar nóg een beetje af kan.”

Waarom is het bouwen van een quantumcomputer zo moeilijk?

„Een van de problemen is dat echte qubits tamelijk kwetsbaar zijn. De informatie in de qubits kan verloren gaan door ruis, interactie met de buitenwereld. Een probleem is bovendien dat je van een qubit niet gemakkelijk een reservekopietje kunt maken. Als je de quantuminformatie uitleest, vernietig je haar.

„Aanvankelijk leek het erop dat quantumrekenen vanwege deze problemen nooit echt mogelijk zou zijn, maar er bleek een oplossing te zijn: quantumfoutcorrectie. Daarbij sla je een informatie-qubit op in meerdere qubits, 5 of 7 of 20. Dat werkt als een soort geavanceerde reservekopie: als een van de fysieke qubits uit de pas gaat lopen, of beschadigd raakt, is de fout te herstellen met hulp van de andere qubits, zonder dat er quantuminformatie verloren gaat.

„Een consequentie is wel dat je de berekeningen dan ook op deze gecodeerde qubits moet uitvoeren, en dat je dus voortdurend bezig bent met het detecteren en herstellen van fouten. Dat kost behoorlijk wat extra, in aantallen qubits maar ook in rekentijd. Het doel van het zwaartekracht-project is om quantumalgoritmes te vinden die, rekening houdend met al die beperkingen en extra kosten, al iets nuttigs kunnen doen met een quantumcomputer van 50 à 100 fysieke qubits.”

Maar welke rekenproblemen gaan juist níet sneller met quantumcomputers?

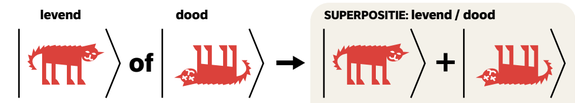
„De meeste problemen zelfs. Het is een misvatting dat de quantumcomputer alles sneller maakt, omdat de quantumcomputer ‘alles tegelijk’ uitrekent met een superpositie van qubits. Als dat zo was, zou het vergelijkbaar zijn met parallelle computers, die met 1.000 computerprocessoren tegelijk rekenen. Maar er is een bottleneck. Je kunt maar een heel klein gedeelte van de hele superpositie uitlezen: één component. Alsof willekeurig 999 van de 1.000 parallelle processoren vlak vóór het uitlezen vernietigd worden.

„De creativiteit van een quantumalgoritme zit hem dan ook vooral in zorgen dat je je superpositie zó manipuleert dat er uit die ene meting toch iets zinnigs komt, iets waarin informatie uit al die andere componenten gecondenseerd is. Dat kan lang niet altijd. Nu het goed gaat, en quantumcomputers in zicht lijken te komen, is er dan ook het gevaar van hype. Er zijn miljarden geïnvesteerd, de ontwikkelingen gaan nu ook wel heel snel, maar de quantumcomputer is geen wondermiddel.”

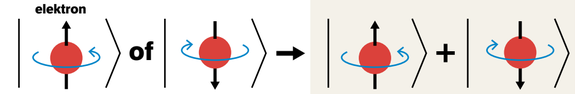
Het is een misvatting dat de quantumcomputer alles sneller maakt.

De quantumcomputer werkt dankzij ‘superposities’

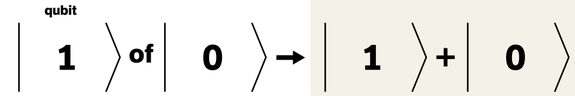
1 Quantummechanica stelt dat voorwerpen in ‘superpositie’ kunnen bestaan. Het beruchtste voorbeeld is Schrödingers kat. Na een quantummechanische moordaanslag met 50 procent slagingskans, is dit arme dier in een superpositie van dood en levend.



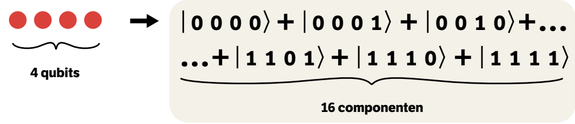
2 Quantumsuperpositie is realistischer bij elementaire deeltjes zoals elektronen, die linksom of rechtsom kunnen draaien. Of in een superpositie van linksom en rechtsom.



3 In een quantumcomputer kan een elektron de rol van quantum-bit, of qubit vervullen. Linksom staat dan voor de waarde 1, rechtsom voor 0. De qubit kan ook een superpositie van 1 en 0 aannemen.



4 De wonderbaarlijke vermenigvuldiging van rekenkracht treedt op als meerdere qubits gekoppeld worden. Vier qubits die afzonderlijk ieder in een superpositie van 1 en 0 zijn, kunnen samen een superpositie met zestien (twee tot de vierde macht) componenten vormen: 0000, 0001, 0010, enzovoort, tot en met 1111.



NRC 090917 / RvS, Bruno van Wayenburg

CV
Ronald de Wolf

Ronald Michiel de Wolf (Zaandam, 1973) studeerde **computerwetenschap en filosofie** aan de Erasmus Universiteit Rotterdam.

In 2001 promoveerde hij aan de Universiteit van Amsterdam (UvA) op een proefschrift over quantumcomputers. **Daarna werkte hij aan de universiteit van Berkeley.**

Hij is hoogleraar aan de UvA en onderzoeker bij het Centrum voor Wiskunde en Informatica. De Wolf is een liefhebber van **klassieke muziek** en literatuur.

SUPERPOSITIES

Tegelijk in twee toestanden

Hoe werkt de quantumcomputer? Dankzij het verschijnsel dat iets zich in meerdere toestanden kan bevinden.

De quantumcomputer dankt zijn wonderbaarlijke eigenschappen aan een van de vreemdste verschijnselen in de quantummechanica: ‘superposities’ van meerdere toestanden.

Een quantumbit kan bijvoorbeeld een elektron zijn in een ultra-gekoeld diamantkristal, dat je kunt manipuleren met laserlicht. Het elektron kan linksom draaien, geschreven als respectievelijk |1> en |0> Maar het kan ook in een superpositie zijn: |1>+|0>.

Dit loopt verder uit de hand als je meerdere qubits combineert. Een 4-qubits-quantumcomputer kan zich in een optelsom bevinden van $2^4 = 16$ componenten, genummerd van |0000> tot |1111>. Bij 8 qubits zijn dat al 256 componenten, bij 32 qubits 2^{32}, bijna vijf miljard. Een quantumcomputer kan al die componenten manipuleren met één manipulatie van de superpositie, die in miljoenvoud wordt uitgevoerd. Dat is de quantum-sluiproute die bijna voor niets enorme rekenkracht oplevert.

Het Deutsch-Josza-algoritme, een van de eerste quantumalgoritmen, berekent een eenvoudige maar eigenaardige som. Stel: je hebt een rijtje van 64 enen en nullen, waarvoor geldt: of ze zijn alle 64 hetzelfde (‘gelijk’), óf het zijn 32 enen en 32 nullen (‘gebalanceerd’). Welke van die twee opties is waar: ‘gelijk’ of ‘gebalanceerd’?

Het Deutsch-Josza-algoritme begint met 6 qubits die allemaal |0> zijn, een toestand die geschreven wordt als |000000>. Vervolgens wordt op deze toestand een ‘Hadamard-transformatie’ losgelaten. Die brengt iedere afzonderlijke qubit in een superpositie van |1> en |0>.

Het resultaat is een superpositie van 64 verschillende toestanden: |000000>+|000001> en dan nog 62 componenten tot en met...+ |111111>

Op die superpositie wordt in één ‘vraag-operatie’ het input-rijtje van 64 bits losgelaten. Bijvoorbeeld: voor het 17de getal in het rijtje wordt de 17de component van de superpositie onder handen genomen. Dat is |010001>. Als het 17de getal in het rijtje 0 is, verandert die component niet, maar als dat getal 1 is, wordt de component negatief: -|010001>. Hetzelfde gebeurt met alle andere componenten. Dan volgt opnieuw een Hadamard-transformatie, waarbij alle qubits opnieuw splitsen.

Alleen als het oorspronkelijke rijtje ‘gebalanceerd’ was komt hier de oorspronkelijke superpositie |000000> weer terug. Bij alle andere uitkomsten was het oorspronkelijke rijtje niet gebalanceerd, dus moet het wel ‘gelijk’ geweest zijn. Met een klassieke computer waren er tenminste 32 stappen nodig geweest voor deze conclusie, waar de quantumcomputer maar twee Hadamard-operaties per qubit nodig heeft .