

# Superposição e paralelismo quânticos:

## possíveis efeitos em algoritmos

Marcos Cuzziol<sup>1</sup>

**Resumo:** Os fenômenos da superposição de estados e do paralelismo tornam a computação quântica fundamentalmente diferente da computação digital. Mas as diferenças não estão restritas à velocidade de processamento. Como veremos abaixo, a qualidade dos resultados do processamento quântico é, potencialmente, inatingível no ambiente digital, independentemente do tempo de processamento necessário. Este artigo tem o objetivo de analisar o princípio dos algoritmos genéticos com a finalidade de ilustrar as diferenças possíveis entre processamentos digital e quântico.

**Palavras-chave:** computação; genoma; quântico; digital; evolução; adaptação.

---

<sup>1</sup> Marcos Cuzziol tem graduação em Engenharia Mecânica pelo Instituto de Ensino de Engenharia Paulista, com o mestrado (intitulado *Games 3D: aspectos de desenvolvimento*) e o doutorado (intitulado *Estados superpostos: proposta de modelo matemático para games 3D*) realizados na ECA-USP. Desenvolvedor de games, programador, sócio fundador da Perceptum Software Ltda., foi gerente do núcleo de Inovação do Instituto Itaú Cultural de 2000 a 2020. É pesquisador colaborador do Instituto de Estudos Avançados da USP. Atua principalmente nos seguintes temas: games, realidade virtual, inteligência e comportamento artificial, arte e tecnologia. Orcid: <https://orcid.org/0009-0005-6108-824X>.

### **Superposition and parallelism: possible effects on algorithms**

**Abstract:** The phenomena of superposition of states and parallelism make quantum computing fundamentally different from digital computing. But the differences are not restricted to processing speed. As we will see below, the quality of quantum processing results is potentially unattainable in the digital environment, regardless of the processing time required. This article analyzes the principle of genetic algorithms in order to illustrate the differences between digital and quantum processing.

**Keywords:** computing; genome; quantum; digital; evolution; adaptation.

## Computação quântica

A unidade básica de dados de um computador digital é o bit, que pode assumir um único valor entre dois possíveis: zero ou um. Em um computador quântico, a unidade é o qubit que, além de zero ou um, pode também assumir os dois valores simultaneamente. Nessa situação, diz-se que um qubit está em superposição de estados, ou coerência. Em um primeiro momento, pode nem parecer tão importante essa característica de se armazenar dois valores simultaneamente em um único qubit. No entanto, para uma sequência de vários qubits, o número de valores simultâneos cresce de forma exponencial. Por exemplo, 2 qubits podem armazenar 4 valores simultaneamente; 8 qubits podem guardar 256 valores simultâneos; 16 qubits, 65.536 valores; e assim por diante. Quando passamos dos 100 qubits, o número de valores simultâneos é comparável, em ordem de grandeza, ao total estimado de átomos do universo. Isso é algo inimaginável na computação digital: ser capaz de armazenar, em uma única variável de pouco mais de 100 unidades de dados, tantos valores quanto o número de átomos do universo inteiro.

Mas de que adianta armazenar tantos valores em uma única variável composta de qubits? Ora, existe outra característica importante que se denomina “paralelismo quântico”. De forma muito simplificada, ao operar sobre qubits, um computador quântico pode agir sobre todos os valores que esses qubits armazenam, de forma simultânea. Na computação digital, se quisermos computar uma série de valores diferentes, precisamos agir sobre eles um de cada vez. Um número muito grande de valores diferentes exigirá um número equivalente de passos de computação. O paralelismo quântico permite, em certas situações, que uma mesma operação (uma conta, por exemplo) seja aplicada sobre todos os valores superpostos, em um único passo de computação.

Se essas duas características, superposição e paralelismo, soam estranhas, é porque elas o são, de fato. Não fazem parte de nossas experiências do dia a dia, embora sejam corriqueiras nos fenômenos quânticos. Mas, como elas poderiam afetar um algoritmo específico?

## Algoritmos genéticos

Apresentados na pesquisa de John Holland em 1975, algoritmos genéticos simulam, em computação digital e de forma simplificada, o processo de adaptação por seleção natural proposto por Charles Darwin. Algoritmos genéticos permitem que soluções “evolam”, ou melhor, que se

adaptem a objetivos específicos. Vejamos como isso acontece utilizando um exemplo de projeto de estrutura mecânica.

Para que uma solução qualquer possa ser processada por um algoritmo genético, ela deve ser expressa por uma sequência numérica. No exemplo de uma estrutura mecânica, essa sequência numérica deve prever dimensões detalhadas de cada pequeno elemento que comporá a estrutura. As variáveis dessa sequência formam um genoma digital, uma codificação capaz de abrigar todas as soluções de possíveis estruturas mecânicas desse genoma. Na natureza, esse seria o genoma de uma espécie, e todos os indivíduos possíveis dessa espécie obedecem à codificação desse genoma. Quando atribuímos valores específicos às variáveis do genoma, obtemos uma solução individual dentro desse genoma, ou seja, um genótipo digital. Na natureza, esse seria um indivíduo daquela espécie.

O algoritmo genético começa por criar uma grande população de soluções (de estruturas mecânicas, no nosso caso) individuais, normalmente geradas de forma aleatória, “sorteando” valores para cada variável do genoma. Muito provavelmente, estruturas geradas de maneira aleatória serão ruins, pouco oferecendo em termos de eficiência estrutural. Mas a população de estruturas é grande, e certamente algumas soluções serão melhores que outras. Faz-se necessário selecionar as melhores estruturas geradas, de modo análogo à seleção dos mais aptos proposta por Darwin.

Isso é feito por uma função de avaliação que atribui notas numéricas para cada estrutura da população. É na função de avaliação que se define o objetivo das soluções propostas pelo algoritmo genético. Como falamos de estruturas mecânicas (lembrando que essas soluções poderiam ter qualquer outra finalidade), digamos que estejamos interessados em obter uma estrutura que apresente a melhor resistência mecânica possível para o menor peso total. Pois bem, nossa função de avaliação precisa dar uma nota para cada estrutura da população, e essa nota será maior para aquelas soluções que tenham maior resistência e menor peso estrutural. Para essa finalidade, a função de avaliação deverá calcular a resistência estrutural e o peso de cada solução individual a partir dos dados dimensionais do genótipo digital. São cálculos trabalhosos, mas bastante viáveis em computação digital (pode-se, inclusive, considerar materiais diferentes e, dessa forma, selecionar o melhor material para aquela estrutura mecânica específica).

As estruturas que recebem as maiores notas são então mantidas, enquanto todas as outras são eliminadas, finalizando o processamento da geração inicial. Mas o trabalho do algoritmo genético está apenas começando. Uma nova população será criada a partir das melhores estruturas, utilizando estratégias baseadas na evolução por seleção natural:

1. Hibridização: o genótipo de uma solução é combinado com o de outra, fazendo com que o genótipo resultante “herde” características dos dois “pais”;

2. Mutação: ao genótipo são aplicadas pequenas modificações aleatórias, permitindo que novas características sejam introduzidas à população;

3. Clonagem: o genótipo de uma solução selecionada é simplesmente copiado, sem alterações.

Gera-se assim uma nova população com um número muito grande de estruturas, que já não são meramente aleatórias como na primeira população, mas que partem das estruturas mais adaptadas. Mais uma vez, aplica-se a função de avaliação a cada indivíduo da população, selecionam-se os mais adaptados, e todo o processo é repetido, dezenas, centenas, milhares de vezes, até que uma solução válida seja encontrada: digamos, uma estrutura com resistência maior que “x” e peso menor que “y”. Nesse ponto, o algoritmo apresenta a solução e para.

Não deve surpreender que estruturas projetadas conforme o descrito acima pareçam mais orgânicas que as projetadas, para as mesmas finalidades, por seres humanos. Também não deve surpreender que soluções de algoritmos genéticos sejam muito mais eficientes que as criadas diretamente por projetistas humanos. Não por acaso, exemplos reais lembram estruturas ósseas (Figura 1).



**Figura 1:** Suporte projetado por algoritmo genético (Henry Dennis/NASA). Disponível em: <<https://shorturl.at/jkyM1>>. Acesso em: 7 set. 2023.

É importante notar que o algoritmo genético apenas seleciona boas soluções a partir de um campo de possibilidades muito grande, ou seja, a partir das soluções possíveis dentro do genoma proposto. Mal compa-

rando, seria como selecionar os seres humanos mais adaptados a alguma função tomando por base os indivíduos possíveis dentro do genoma humano. Esse número de possibilidades, embora não seja infinito, é extremamente grande, pois são todas as combinações possíveis dentro do genoma humano, ou, no nosso exemplo, do genoma digital. Um algoritmo genético digital, portanto, não avalia literalmente todas as possibilidades do genoma. Ele parte de uma população inicial, seleciona as mais adaptadas, aplica pequenas modificações e gera nova população; então, seleciona novamente as mais adaptadas e repete o processo até que uma solução aceitável seja encontrada. Não se trata da melhor solução, mas de uma solução aceitável dentro dos parâmetros do algoritmo. Encontrar a “melhor” solução exigiria que todas as possibilidades fossem avaliadas, uma a uma, e isso exigiria tempo demais: dependendo do tamanho do genoma, talvez mais tempo do que toda a idade estimada de nosso universo.

Vimos como um algoritmo genético funciona em computação digital, com resultados surpreendentes. Mas como poderia ser um processo desse tipo que rodasse em um computador quântico?

### **Superposição e paralelismo**

Antes de descrever o processo quântico, cabe alertar que ele ainda não é viável. Os processadores quânticos, em 2023, ainda não têm qubits em número suficiente; existem muitas questões a serem resolvidas para que se possa manter um grande número de qubits em superposição de estados, em coerência; e há também uma questão estatística difícil de controlar: o processo de computação quântica precisa ser repetido algumas vezes, para confirmação dos resultados. No entanto, a tecnologia tem avançado rapidamente, e acho importante tentar antever o que pode vir a ser possível em algumas décadas, talvez antes.

Um genoma quântico seria formado de qubits, não de bits. Ao se atribuir valores, utilizando-se a característica da superposição de estados, o conjunto de qubits armazenaria todas (sim, todas) as soluções possíveis dentro do genoma, simultaneamente. Uma função de avaliação quântica seria aplicada sobre essa única sequência de qubits, que conteria a totalidade de soluções possíveis, e, por meio do paralelismo quântico, indicaria a solução mais bem adaptada dentre todas as possíveis em uma única etapa. Fim. Não haveria necessidade de se gerar populações, nem de se aplicar hibridização, mutação e clonagem, nem mesmo de se economizar capacidade de processamento parando o processo quando uma solução apenas aceitável fosse atingida. E mais: em nosso exemplo, não teríamos

apenas uma boa solução — teríamos a melhor estrutura possível dentro das exigências de resistência e peso. Algo inviável de ser realizado em um computador digital no tempo de existência do universo poderia ser resolvido em poucas instruções de computação quântica.

Fiz questão de descrever os dois processos em certo nível de detalhe para deixar claro que não se trata apenas de maior velocidade de processamento. A velocidade, sem dúvida, pode ser extremamente maior. Entretanto, muito além disso, trata-se de um paradigma de computação diferente, estranho até, que não encontra paralelo na natureza. É difícil até mesmo tentar conceber uma adaptação de espécie tão veloz; por exemplo, algo que não levasse milhares de anos, mas talvez poucos segundos. Além disso, o processo de adaptação natural age sobre relativamente poucos indivíduos por vez, não sobre todos os possíveis, simultaneamente. É desse nível de diferença que estamos tratando quando comparamos a computação digital com a quântica.

### **Algoritmos quânticos**

Atualmente, novas propostas para algoritmos quânticos são desenvolvidas em grande número. Escrever programas para um computador quântico exige, do programador, um modo de pensar muito diferente daquele da computação clássica, digital. Em virtude do estágio atual da computação quântica, é natural que os algoritmos desenvolvidos até aqui sejam até certo ponto experimentais. Muitos trabalhos acadêmicos têm por objetivo desenvolver estratégias para melhorar a eficiência de algoritmos existentes, ou para aproveitar melhor certas características da computação quântica — ou, ainda, para mensurar diferenças entre a computação quântica e a clássica. Um exemplo, dentre muitos possíveis, pode ser encontrado nas referências deste texto: uma comparação numérica entre algoritmos genéticos clássicos e quânticos (Ibarrondo; Gatti; Sanz, 2022).

Como vimos, computadores quânticos ainda têm desafios a serem vencidos. O número total de qubits e a manutenção deles em coerência são dois desses desafios. Mas computadores quânticos funcionais já existem e tanto a superposição de estados como o paralelismo quântico são corriqueiros neles. Como exemplo, o computador quântico Sycamore, criado pelo Google e utilizado para pesquisas desde 2019, possui 53 qubits que podem, não sem certa dificuldade, ser mantidos em coerência. Resta ainda atingir a escala e a confiabilidade necessárias para que o uso dessa nova forma de computação seja prático. Mas não tenho dúvidas de que isso venha a ser alcançado em alguns anos, talvez na próxima década.

## Além dos algoritmos genéticos

Algoritmos genéticos são processos bastante objetivos, nos quais cada solução é calculada em detalhes segundo critérios de uma função de avaliação. Mas a chamada inteligência artificial trabalha também com algoritmos mais, digamos, “subjetivos”. São processos como o das redes neurais artificiais, muito usadas para reconhecer imagens e voz, que simulam o funcionamento de uma rede de neurônios para reconhecer padrões em dados. Ou como o algoritmo denominado de redes generativas adversárias, no qual uma rede gera soluções enquanto outra tenta definir se as soluções geradas fazem ou não parte de uma grande base de dados reais, que as alimentam. Softwares como Midjourney e Stable Diffusion usam algoritmos semelhantes (Figura 2).



**Figura 2:** Imaginação máquina: engenheiros presenciam a manifestação da primeira consciência artificial. Imagem gerada pelo Stable Diffusion a partir de prompt do autor.

Algoritmos como esses últimos dependem de grandes massas de dados usadas para treinamento, processo conhecido como machine learning, e o que ousos chamar de subjetividade advém exatamente dessa aprendizagem. Por exemplo, se uma rede neural for treinada para reconhecer faces humanas apenas com fotos de pessoas brancas, ela pode ter dificuldades para identificar pessoas negras, asiáticas ou indígenas. Os dados de treinamento delimitam e direcionam os resultados possíveis para esses algoritmos, pois o reconhecimento de padrões depende



fundamentalmente dos dados usados no seu treinamento. Dados novos, que não se encaixam nos padrões apreendidos, deveriam realimentar o treinamento, mas, na prática, isso raramente acontece. Se limitações desse tipo costumam ocorrer conosco, humanos, por que não com simples algoritmos?

É muito difícil prever como a computação quântica poderia afetar algoritmos de machine learning. O mais provável é que novos algoritmos quânticos sejam desenvolvidos sem necessariamente serem baseados em redes neurais ou em redes generativas adversárias. Mas, na mesma linha do exposto para algoritmos genéticos, vejamos algumas possibilidades.

Redes neurais artificiais são, surpreendentemente, algoritmos simples. Cada neurônio pode estar ligado ou desligado, zero ou um, e as sinapses são valores numéricos que conectam os neurônios e funcionam como multiplicadores. Durante o treinamento do algoritmo, definem-se os valores das sinapses para que a rede aprenda com os dados que a alimentam, para que ela reconheça os padrões desejados. Se cada neurônio for representado por um qubit, entretanto, pode-se configurar sobreposições dos estados ligado e desligado, e as sinapses seriam também diversos valores superpostos. O que poderia resultar dessa configuração? No mínimo, a aprendizagem poderia ser muito mais rápida e profunda. No outro extremo... “onisciência” algorítmica, talvez? É difícil prever.

Já para as redes generativas adversárias, temos duas redes que competem entre si, uma gerando resultados (a rede generativa) e outra definindo se esses resultados podem fazer parte de uma base de dados reais (a rede discriminadora). Quando a rede generativa consegue enganar a rede discriminadora, temos um resultado que se parece com algo real, ou melhor, que se parece com os dados reais utilizados como referência pela rede discriminadora. Na computação quântica, já tivemos uma ideia do que uma rede generativa poderia vir a fazer: potencialmente, gerar todas as possibilidades de uma única vez. A rede discriminadora funcionaria como descrito antes mas, talvez, escolhendo de modo muito mais preciso o que parece “real” ou não.

### **O que esperar em um futuro próximo?**

Nos últimos anos, algoritmos digitais se desenvolveram de modo surpreendente. Em especial nas áreas que exigem trabalho cognitivo, como nos exemplos que vimos aqui. Seria difícil imaginar, há meros 20 ou 30 anos, as aplicações que existem hoje, capazes de projetar estruturas, de reconhecer padrões complexos, ou de criar imagens e sons quase reais a partir de meros algoritmos e de grandes volumes de dados. Tal desen-

volvimento acelerado foi possível não só pelo poder de processamento cada vez maior, mas também pelo refinamento de algoritmos progressivamente mais eficientes, que até certo ponto tentam mimetizar processos naturais, como o da evolução e o do funcionamento básico de redes neurais orgânicas. Tão grande é a velocidade com que algoritmos cada vez mais eficientes são criados que mal temos tempo de conhecer suas possibilidades antes que eles sejam suplantados por versões aprimoradas.

Entretanto, algo muito mais revolucionário parece estar no horizonte: a computação quântica. Enquanto o computador digital segue ainda as definições de 1937 da “máquina computadora universal”, de Alan Turing, a computação quântica se baseia em conceitos estranhos como o da superposição de estados e o do paralelismo. Vale repetir: não se trata simplesmente de um tipo de computação mais veloz. Trata-se de um paradigma diferente de computação, que talvez possa realizar processos hoje inimagináveis. Tenho pesquisado sobre a computação quântica já há alguns anos, e não tenho dúvidas de que ela abrirá um campo de possibilidades muito estimulante. Exigirá novas maneiras de pensar, de escrever programas, de criar. Na verdade, isso tudo ela já exige, mesmo em estágio tão inicial. Será uma ferramenta poderosíssima, e talvez venha a decretar o fim da era digital (que, muitos pensávamos, duraria para sempre).

Para encerrar esta perspectiva de um programador (digital) sobre a computação quântica e seus possíveis efeitos em algoritmos, devo mencionar uma questão que, há muitos anos, me fascina: qual nível de controle um programador tem sobre os resultados de seus algoritmos?

Não é uma pergunta óbvia. Sei que, em uma percepção superficial, muitos acreditam que, se um programador escreveu um algoritmo, todos os resultados estariam automaticamente sob seu controle. Nada poderia estar mais longe da verdade... De fato, em um programa muito simples, é esperado que o ser humano que o criou conheça todos os resultados possíveis. Mas, em algoritmos como os dos exemplos deste texto, torna-se absolutamente inviável para um programador prever resultados, ainda que ele tenha escrito todas as instruções seguidas fielmente pelo programa. Exemplos: os programadores de Deep Blue, computador que venceu o então campeão mundial de xadrez em 1997, nunca teriam vencido aquelas partidas se as jogassem contra Garry Kasparov (de fato, eles se surpreendiam com as jogadas do próprio programa que criaram); de modo similar, os programadores do algoritmo genético que gerou a estrutura do exemplo mais acima também não teriam condições de criar uma estrutura mecânica tão eficiente (nem mesmo os melhores projetistas o conseguiriam); e, ainda, os programadores de Midjourney não têm como saber

qual imagem o algoritmo criará com base nos comandos de alguém. Uso o verbo criar ciente do peso que ele tem, pois não consigo imaginar uma palavra melhor para os exemplos aqui discutidos. Certamente não se trata de criação humana — ela é toda baseada em sequência de instruções, de cálculos sobre dados existentes. Em outras palavras, são processos determinísticos: não há “alma” ou “mágica” em um algoritmo. E, ainda assim... eles são capazes de nos superar, de surpreender até mesmo quem os escreveu.

A revolução da computação quântica provavelmente estabelecerá um abismo entre a criação humana de um algoritmo e seus resultados.

### Referências

HILLE, Karl. NASA turns to AI to design mission hardware. *NASA*, 10 fev. 2023. Disponível em: <https://www.nasa.gov/feature/goddard/2023/nasa-turns-to-ai-to-design-mission-hardware>. Acesso em: 14 ago. 2023.

HOLLAND, John. *Adaptation in natural and artificial systems*. Ann Arbor, MI: University of Michigan Press, 1975.

IBARRONDO, Rubén; GATTI, Giancarlo; SANZ, Mikel. Quantum vs. classical genetic algorithms: a numerical comparison shows faster convergence. In: *IEEE Symposium Series on Computational Intelligence (SSCI)*, Singapore, 2022, p. 947–954.