İçindekiler

KUANTUM HESAPLAMA

SORU-1: Jitter (Dalga Bozulumu) hakkında bilgi veriniz.

Genelde sinyal işleme konularında geçen bir terim olan jitter (dalga bozulumu), bilgisayar bilimlerinde, ağ (networking), çoklu ortam uygulamaları (multi media) veya resim işleme (image processing) gibi konularda geçmektedir.
Jitter kavramı, kısaca bir sinyalin olması gereken değere göre hatalı dalga değeri vermesidir.

Örneğin yukarıdaki şekilde bir dijital sinyal görülmektedir (resmin üstünde). Bu sinyalin bozulmuş hali resmin ortasında ve bozulmadan kaynaklanan jitter değeri resmin ortasında gösterilmiştir.

Dalgada yaşanan sürekli bir bozulma olmasından dolayı, jitter terimi, faz bozulması veya faz gürültüsü (phase noise) olarak da tanımlanabilir. Dalganın anlık bir noktasında yaşanan gürültüden, bu anlamda farklıdır.

Sinyalde yaşanan ve sürekli olan bu bozulmanın da bir periyodundan (veya frekansından) bahsedilebilir. Dalga bozulumu frekansı (jitter frequency), bu tanıma göre, dalgada yaşanan bozulmaların en büyük değerleri arasındaki mesafedir. Diğer bir deyişle, yukarıdaki şekilde görülen ve bozulum yaşanan dalgaların frekansıdır. Dalga frekansı hesaplanırken, en büyük değerler arasındaki fark alınabileceği gibi en küçük değerler arasındaki fark da alınabilir.

Dalga bozulumunun yaşandığı yere göre farklı isimlendirmelerin kullanılması mümkündür.

Sarnıçlama Dalga Bozulumu (Sampling Jitter): Bu kavram, genelde işaret (sinyal) üzerinde uygulanan çevirimler sırasında ortaya çıkar. DAC (digital to analog converter, dijital verinin analog veriye çevirimi) veya tersi olan ADC (analog to digital converter, analog verinin dijital veriye çevirimi) işlemleri belirli bir zaman almaktadır. O halde sinyal işlenirken, beklenen zamana göre gecikmeli olarak sonuç elde edilecek ve nihayetinde bir dalga bozulumu yaşanacaktır.

Örneğin sarnıçlama yapılan (belirli aralıklarla örnekler alınan, sampling) bir sistemin, ses, ışık veya hız gibi sürekli (conitinous) bir işaret (signal) olduğunu kabul edelim. Bu işaretin belirli zamanlarda değerinin okunarak dijital ortama çevirimi, burada bahsedilen gecikmeler ve kaymalara neticede de sarnıçlama dalga bozulumuna sebep olacaktır.

Paket Dalga Bozulumu: Bilgisayar ağlarında, bazı durumlarda, paketlerin belirli sıklıkta (frequency) iletilmesi beklenir. Bu sıklığın bozulması da bir dalga bozulumu (jitter) olarak kabul edilebilir. Bilgisayar ağlarındaki dalga bozulumu (jitter) aslında başlı başına bir hizmet kalitesi (quality of service) konusudur ve daha çok kabul gören PDV (packet delay variation) terimi altında kullanılmaktadır.

Yukarıda verilen örnekler daha da arttırılabilir. Örneğin bir CD-ROM'dan okuma sırasında, CD üzerindeki verinin aranması sırasında geçen süre, herhangi bir veri transfer yazılımı veya devresinin, veriyi göndermeye başlamasında geçen süre, kuantum kapılarının (qunatum gates), elektron dönüşünden kaynaklanan (spin based) çalışma gecikmesi veya aktarılmak istenen verinin kanal kapasitesinin çok üzerinde olasından dolayı, verinin bir kısmının tıkanıklık (congestion) ile karşılaşması ve bu yüzden beklenen zamandan daha geç transfer edilmesi gibi durumlar birer dalga bozulumu (jitter) örneğidir.

SORU-2: Grover Algoritması (Grover's Algorithm) hakkında bilgi veriniz.

1996 yılında kuantum hesaplamalarının gelişimiyle birlikte, sıralanmamış bir veri tabanı üzerinde arama yapmak üzere geliştirilmiş algoritmadır.

Bilindiği üzere sıralanmamış bir verinin üzerinde arama yapmanın en basit ve en hızlı yolu doğrusal arama (linear search) algoritmasını kullanmaktır. Yani, en kötü ihtimalle, verinin tamamına bakmaktır. Bu algoritmanın büyük-O (growth rate, worst case analysis) değeri O(n)'dir.

Kuantum hesaplamaları, bize bu konuda daha başarılı bir sonuç verebilmektedir. Grover Algoritması sayesinde, sıralı olmayan bir veri kaynağında, veri aramak için, $O(n^{1/2})$, yani n eleman için n'in karekökü kadar elemana bakmak yeterlidir.

Algoritma iki adımda düşünülebilir. İlk aşamada algoritmanın hazırlanması (setup) ikinci aşamada ise algoritmanın çalışmasını inceleyeceğiz.

Hazırlık aşaması (setup)

Algoritmanın kullanılabilmesi için öncelikle algoritmanın çalışacağı problem uzayını tanımlayalım. Problemimiz N adet sıralanmamış verinin içerisinden bir verinin aranarak bulunması olsun. Bu durumda Grover algoritması $\log_2 N$ qubitten oluşan bir H durum uzayı oluşturur. Bu uzayın özelliği N boyutlu olmasıdır. Burada N boyutlu olması ile kastedilen, N adet birbirinden farklı eigen değeri içermesidir. Yani H uzayını aşağıdaki şekilde dirac gösterimiyle ifade edecek olursak

$$\{|1\rangle, |2\rangle, \cdots, |N\rangle\}$$

Uzayda bulunan her eleman için ilgili eigen değeri, aşağıdaki şekilde gösterilebilir:

$$\{\lambda_1, \lambda_2, \cdots, \lambda_N\}$$

Bu uzayda çalışan bir uniform işlem tanımlıyor olalım. Bu işlemi U_ω, ile gösterelim. Bu işlem için basit bir arama kriteri belirlenmesi yeterlidir. Yani uzayımızda yapılacak olan arama veya daha başa dönersek sıralı olmayan veri tabanı üzerinde yapılan aramam kriteri bu kuantum

işlemi U_ω, üzerinde tanımlı olsun. Bu işlemin indis kısmında belirtilen omega değeri (ω) aslında bu işlemin arama sonucunda bulunmasını istediğimiz değerin ω olduğunu belirtir. Diğer bir deyişle, N adet veriyi aradıktan sonra bulmak istediğimiz ver ω sırasında bulunan veri olsun.

Bu verinin eigen durumu (eigenstate) ise |ω> şeklinde gösterilen durumdur. Aynı zamanda eigen değeri (eigen value) ise yukarıdaki tanımlamada λ_ω ile gösterilen değerdir.

Diğer bir deyişle U_ω
işleminin, aşağıdaki özelliği taşımasını istiyoruz*:*

$$U_\omega|\omega\rangle = -|\omega\rangle$$

$$U_\omega|x\rangle = |x\rangle \text{ bütün } x \neq \omega \text{ için}$$

Algoritmanın üzerinde çalıştığı ve algoritma için en kritik nokta olan U_ω, aslında bir kara kutu (Blackbox) olarak düşünülebilir. Bu konuda quantum kapılarından (quantum Gates) tasarlanmış bir devre olarak da düşünülebilir.

Amacımız bu devrenin aradığımız değeri bulması halinde 1 döndürmesi ve bunun dışındaki bütün ihtimallerde 0 döndürmesi şeklinde tanımlanabilir.

Algoritmanın Çalışması

Yukarıdaki hazırlık işlemleri tamamlandıktan sonra, algoritmanın çalışmasına geçebiliriz. Kısaca, hazırlık aşamasında bir adet U_ω, tasarlanmış ve aramanın yapılacağı bütün N değer için eigen halleri (eigenstates) ve eigen değerleri (eigenvalue) çıkarılmıştır.

Bu aşamada, algoritmanın ulaşmak istediği hedefi aşağıdaki şekilde tanımlayabiliriz:

1. Anlık olarak durum bilgisi s değişkeninde tutulmak üzere:

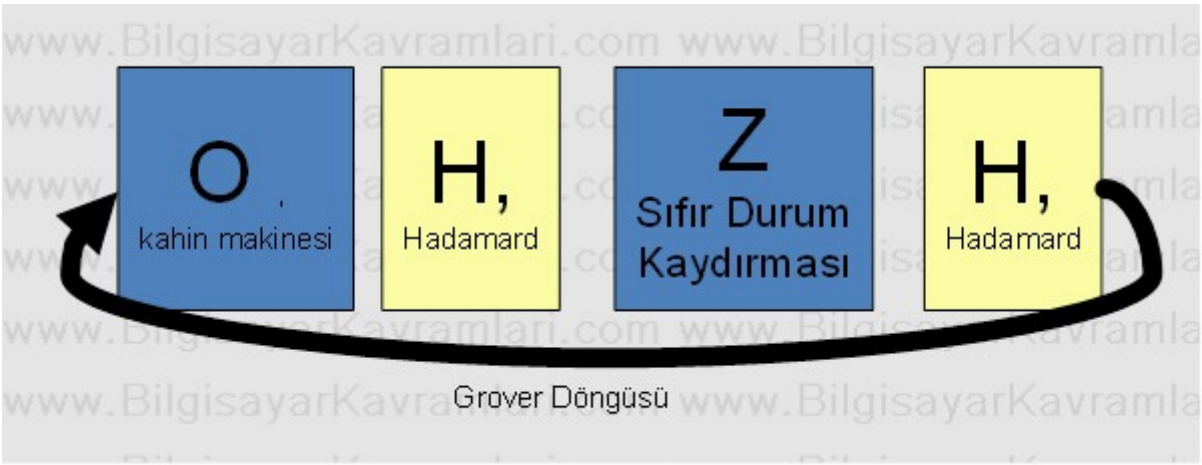
$$|s\rangle = \frac{1}{\sqrt{N}} \sum_{x=1}^{N} |x\rangle$$ değeri hesaplanır

1. Her bir r(N) işlemi, bir grover döngüsü olmak üzere (grover iteration),
    1. U_ω İşlemi uygulanır
    2. U_s İşlemi uygulanır
2. Ω ölçümü yapıldığında, ölçüm sonucu 1′e yakın bir olasılıkla λ_ω olarak bulunacaktır bulunan bu λ_ω değerinden ω değerine ulaşılabilir.

Yukarıdaki algoritmada 1′e yakın olasılık kısmı özel olarak altı çizilmesi gereken kısımdır, bunun sebebi grover algoritmasının kesin sonuç değil olasılıksal sonuç üretmesidir.

Grover Algoritmaının Kuantum Kapıları ile gerçeklenmesi

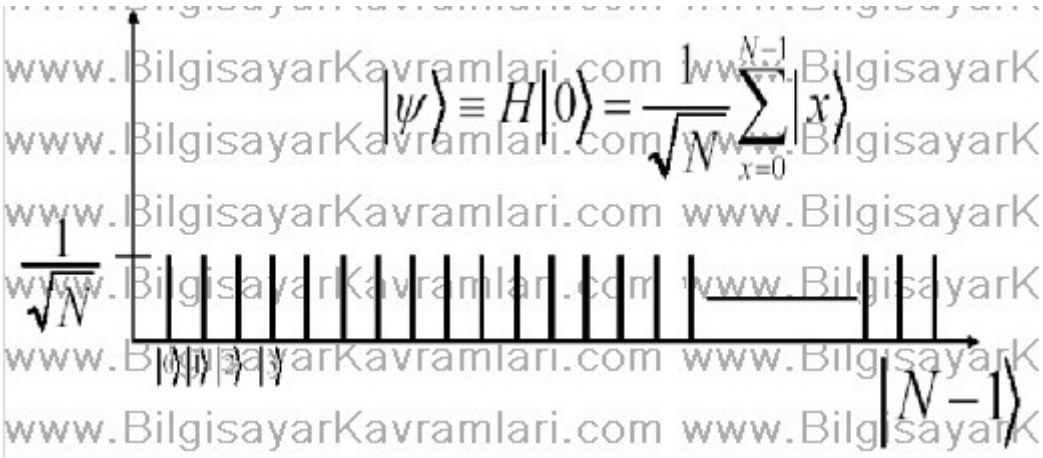
Yukarıdaki algoritmada bulunan 2. adım, yani grover döngüsü (grover iteration) aşağıdaki şekilde temsil edilebilir.



Basit bir kahin makinesi (qunatum oracle machine), elimizde bulunan arama uzayında çalışmaktadır. Bu makine, yukarıdaki algoritmada Uω ile gösterilen dönüşüm işlemidir. Klasik bir kahin makinesinde olduğu gibi, aranan veri için özel olarak tersini alma işlemi yapmaktadır. Diğer bütün verileri olduğu gibi geçirmektedir.

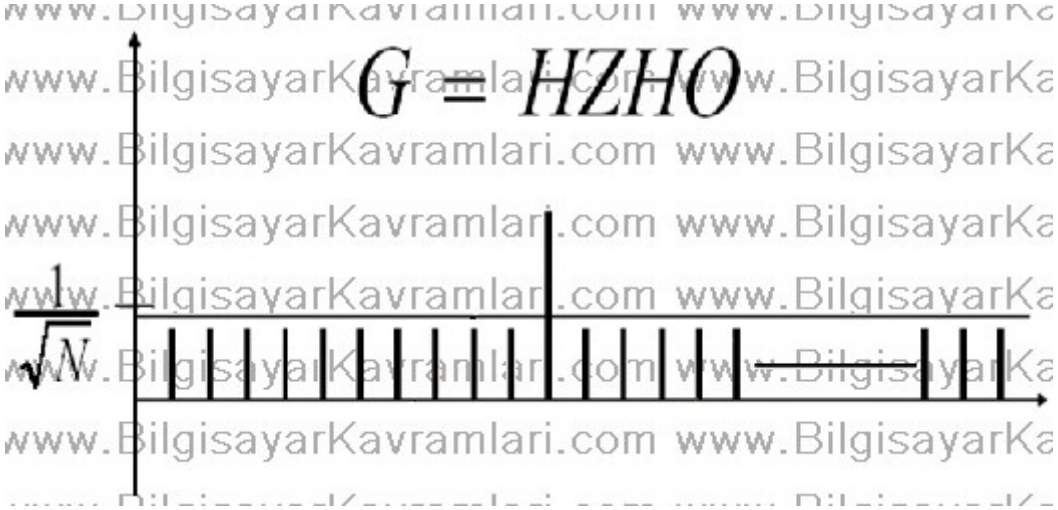
Ayrıca sıfır durum hal kaydırması (Zero state phase shift) devresi ile de Hadamard kapıları arasında, kahin makinesinin işaretlediği kubitin değerini arttırıyoruz. Yani, kahin makinesi n adet kubitten bir tanesini işaretliyor, HZH kombinasyonu ise bu işaretli kubiti arttırıp, diğer kubitlerin değerlerini düşürüyor. Neticede tek bir kubit diğerlerinden daha yüksek değere sahip oluyor. Bu işlem sürekli olarak tekrar edildiğinde de işaretli kubit, diğerlerinden ayırt edilecek kadar (neredeyse, diğerleri 0 ve işaretli kubit ise 1 olacak kadar) belirgin oluyor.

Bunu hayal etmek biraz güç olabilir o yüzden görsel bir şekilde ifade etmeye çalışalım:



Örnek olarak, N adet kubit için yukarıdaki hadamard kapısı sonucunu ele alalım. Buradaki her kubitin genlik değeri (amplitude) düşey eksende gösterilmiştir ve ayrıca ilk grover döngüsü sırasında hepsi eşittir. Yani bütün genlikler eşit bir şekilde çalışmaya başlıyoruz.

İlk döngüden sonra kubitlerin değeri aşağıdaki hali alıyor:



Görüldüğü üzere bütün kubitlerin genliği azalırken, özel olarak seçilmiş olan bir kubitin genliği artmaktadır.

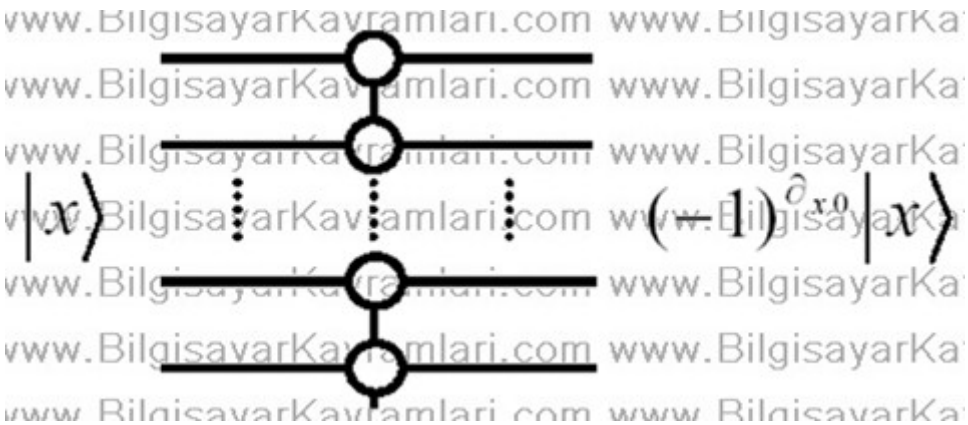
İstatistiksel olarak bu işlemin, n adet kubit için, n değerinin karekökü kadar tekrar edilmesi yeterlidir.

SORU-3: Sıfır Durum Hal Kaydırma Devresi (Zero State Phase Shift Circuit) hakkında bilgi veriniz.

Kuantum devrelerinden birisidir. Genelde Z harfi ile gösterilir. Aşağıdaki dönüşüm işlemini gerçekleştirir:

$$|x\rangle \xrightarrow{Z} (-1)^{\partial_{x0}} |x\rangle$$

Yukarıdaki gösterimde Z harfi ile ifade edilen ve geçiş öncesinde örnek olarak bir |x> girişi alan devre sıfır durum hal kaydırma devresidir (zero state, phase shift circuit)



Örnek olarak bu dönüşüm, yukarıdaki şekilde gösterilmiştir. N adet kubit (qubit) içeren bir girdi için N adet çıktı tasarlanmış ve bu çıktıların ters çevrilmesi işlemi, yani -1 ile çarpılması işlemi, bir delta olasılığına (↱) bağlanmıştır.

Bu olasılıksal değer, aslında devrenin bir kahin makinesi (oracle machine) olarak çalışması anlamına da gelmektedir. Yani hangi kubitin ters çevrileceğine burada karar verilmektedir.

Bu durumu klasik kuantum devrelerinde olduğu üzere, bir matris şeklinde gösterecek olursak, aşağıdaki gösterimi elde ederiz:

$$Z = \begin{bmatrix} -1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

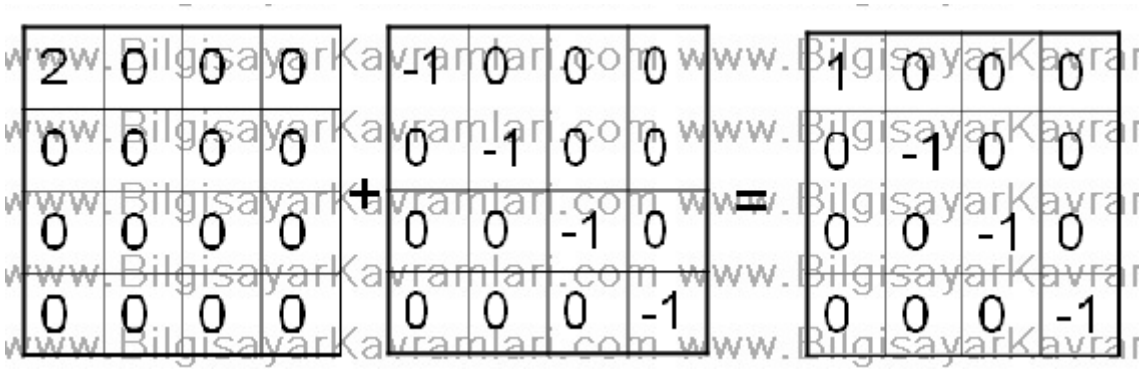
Yukarıdaki bu gösterimde, bütün köşegen değerleri 1 diğer değerler 0 olarak tutulmaktadır. Ancak istatistiksel olarak ters çevirilecek olan bit, ki yukarıdaki örnekte bu bit ilk bit olarak seçilmiştir, -1 olarak dönüşmüş olur.

Bu durumda yukarıdaki matrisi elde etmek için aşağıdaki denklem yazılabilir:

$Z = 2\,|0\rangle\langle 0| - I$

Kısaca, nxn boyutlarında bir sıfırlar matrisi üretilerek bu matrisin köşegeninden birim matris çıkarılacak ve sonuçta yukarıdaki matris elde edilecektir.

Bu durum aşağıdaki şekilde görülebilir:



SORU-4: Kuantum Kahin Makinesi (Quantum Oracle Machine) hakkında bilgi veriniz.

Klasik hesaplama teorisinde (theory of computation) geçen kahin'in (oracle), kuantuma uyarlanmış halidir. Klasik bir kahin makinesi tanımlanırken, bir Turing makinesinin (Turing machine) karar vermeye yarayan özel bir halin olarak belirlenir. Yani aslında soyut bir makinedir ve içeriğiyle çok ilgilenilmez. Tek bilmemiz gereken bir Turing makinesi olduğu

ve bir karar verme mekanizması içerdiğidir. Genelde bir kutu olarak gösterilir ve içi ile ilgilenilmez. Örneğin durma problemi (halting problem) ispatlanmasında kullanılır.

Bu anlamda, örneğin deutsch problemindeki fonksiyonu elinde tutan taraf aslında bir kahin makinesi çalıştırmaktadır.

Kuantum kahin makinesi ise, klasik kahin makinesinin kubitler üzerine uyarlanmış halidir. Diğer bir deyişle kuantum kahin makinesi bir fonksiyonu, süper pozisyon halindeki kubitler üzerinde çalıştırarak bir karar vermeye yarar.

Örneğin aşağıdaki şekilde bir matris dizilimini ele alalım:

$|x\rangle \otimes |d\rangle$

bu dizilimin bir kahin makinesinden geçirilmesi sonucunda aşağıdaki gösterimi elde ederiz:

$|x\rangle \otimes |d \oplus f(x)\rangle$

buradaki f(x) fonksiyonu, kahin makinesinin fonksiyonudur ve belirli bir kubitin değerini işaretler. Bu işaretleme örneğin tersine çevirme olabilir.

Grover algoritması gibi kuantum arama algoritmalarında bu özellik aranan kubitin değerinin tersine çevrilmesi veya aranan kubit için 1 diğer kubitler için 0 döndürmesi şeklinde yorumlanabilir.

Örneğin |x>|d> şeklinde yukarıda tanımladığımız system üzerinde çalışan kahin makinesi, aşağıdaki dönüşümlerden birisini sağlar

- Şayet giren değer çözüm değilse, giren değer değişmeden çıkar:
  $|x\rangle(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle) \xrightarrow{O} |x\rangle(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle)$
- Şayet giren değer çözüm ise, giren değerin tersi çıkar:
  $|x\rangle(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle) \xrightarrow{O} -|x\rangle(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle)$

Yukarıdaki gösterimlerde, |d> gösterimi açılmış ve |0> veya |1> olma ihtimalleri açıkça gösterilmiştir. Yukarıdaki bu iki ihtimali tek bir denklemde modellememiz de mümkündür:

$$|x\rangle(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle) \xrightarrow{O} (-1)^{f(x)}|x\rangle(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle)$$

Bu yeni modeldeki f(x) fonksiyonu, kahin makinesi olarak tanımlanmış olur.

SORU-5: EQP (Exact Quantum Polynomial) hakkında bilgi veriniz.

Bilgisayar bilimlerinde, kuantum hesaplama konusunda kullanılan bir karmaşıklık sınıfıdır. Literatürde tam kuantum polinom zaman (exact qunatum polynomial time) olarak geçmektedir.

Özellikle olasılıksal problemler için %100 başarı ile (yani bütün ihtimalleri eleyerek) sonuç üretme süresinin polinom zamanda olduğunu belirtir.

Bilindiği üzere karmaşıklık problem sınıfları tanımlanırken, klasik problemlerin (burada klasik kelimesi, kuantum hesaplaması öncesinde kullanılan Turing makinelerini ifade için kullanılmıştır) karmaşıklıkları P veya NP olarak ikiye ayrılabilir. P sınıfı, polinom zamanda çözüm üretilebilen problemleri gösterirken, NP sınıfı, doğruluğunun polinom zamanda ispatlanabildiği kümeyi ifade eder.

EQP sınıfı ise, kuantum bilgisayarları üzerinde çalışan programlar için (yani kuantum algoritmaları için), klasik algoritmalardaki P sınıfı gibi düşünülebilir. EQP sınıfının hedefi ise içinde ihtimaller bulunan problemleri çözmektir.

Örneğin deutsch problemi bu konuda kullanılabilecek ihtimalli karar problemlerindendir.

SORU-6: Kubit Kopyalanamazlık Teoremi hakkında bilgi veriniz.

Bu teorem, literatürde "no-clonning theorem" olarak geçmektedir. Basitçe elimizde durumu belirsiz (süper pozisyonda) bir kubit bulunuyorsa, bu kubitten ikinci bir kubit (aynı süper pozisyon değerleri ile) elde edilemez.

Bu durumu görmek için klasik olarak verilen devre tasarım örneğini kullanalım. Amacımız, kuantum kapılarını kullanarak (quantum Gates) bir kubit kopyalama devresi tasarlamak olsun.

Bu devre tasarımında klasik CNOT kapısını kullanmak isteyelim (detaylı bilgi için toffoli kapısına bakabilirsiniz)

CNOT devresi, basitçe iki bitlik giriş alıp bu bitlerden ilki 0 ise ikincisini değiştirmez, ilki 1 ise ikincisinin tersini alır (yani ikinci bitin değil kapısına (not gate) konulup konulmayacağı ilk bit tarafından kontrol edilir. Bu yüzden CNOT , control not kapısı ismi verilmiştir)

Elimizde süper pozisyonda bir kübit olsun:

$$|\psi> = \alpha|0> + \beta|1>$$

Bu kübitin CNOT kapısından geçirilmesi sonucunu aşağıdaki şekilde modelleyebiliriz:

$$|\psi>|0> = \alpha|00> + \beta|10>,$$

$$[\alpha|0> + \beta|1>]\,|0> = \alpha|00> + \beta|10>$$

Bu modele göre, sonucun 0 olduğunu biliyorsak (ket notasyonunda |0>, anlam olarak |0> çıkmasını beklediğimiz olarak düşünülebilir), bu durumda CNOT kapısından önce, ilk bit 1 olup ikinci bit 0 olmuş veya ilk bit 0 olup ikinci bit 0 olmuş olabilir. Bu olasılık dağılımı ise formülümüzde gösterilen, α ve β oranlarıdır.

Öyleyse kübitimizin ilk durumundaki süper pozisyon oranlarında yeni bir kübit elde ettik diyebilir miyiz? Çünkü yukarıdaki denklemde çıkan yeni |00> ve |10> ihtimal değerleri, tam olarak giriş kubitimizin ihtimalleri ile aynıdır. Yoksa acaba yukarıdaki işlem kopyalama değil de daha farklı bir işlem midir? Bu sorunun cevabı, yukarıdaki işlemin ne yazık ki bir kopyalama olmadığıdır. Ve yukarıda yapılan işlem aslında |ψ>|ψ> sonucu elde edilmesidir.

Bu durumu daha açık bir şekilde şöyle anlatabiliriz. Kübitin değerinin 1 veya 0 olarak klasik bite çevrilmesi halinde, yukarıdaki düzeneğimizde, sonuç biti 1 veya 0 olarak okunacaktır. Dolayısıyla yukarıdaki düzenekte verilen bir klasik biti aslında bir kübit üzerine yazabildiğimizi göstermiş oluyoruz.

|ψ>|ψ> işleminin sonucunu bulmak istersek aşağıdaki şekilde işlem yapabiliriz: (Buradaki anlam, |ψ> değeri verildiğinde |ψ> değerinin çıkmasını beklememiz anlamındadır.)

$$|\psi>|\psi> = \alpha^2|00> + \alpha\beta|01> + \alpha\beta|10> + \beta^2|11>$$

Yukarıdaki denklemde, elde edilen |00> ve |11> değerlerinin yalnız kalması için, arada bulunan |01> ve |10> değerlerinin yok olması gerekir. Bunun tek yolu da αβ çarpımının 0 olmasıdır.

Ne yazık ki süper pozisyonda olan bir kübit için bu çarpımın 0 olması mümkün olmaz.

Bu örnekten anlaşılacağı üzere, bir kübit kopyalanamaz (no-clonning) ancak elimizdeki 1 veya 0 şeklindeki klasik bitleri, kübit üzerine yazabiliriz.

SORU-7: Deutsch Problemi hakkında bilgi veriniz.

Literatürde deutsch problem olarak geçen bu problem Ali ve Bekir arasında yaşanan bir tahmin problemidir. Basitçe Ali dilediği bir sayıyı seçip (0 veya 1 olarak ikilik tabandaki bir sayı seçecek) Bekir'e yollar. Bekir aldığı bu mesajı bir f(x) fonksiyonuna sokarak sonucu Ali'ye geri yollar.

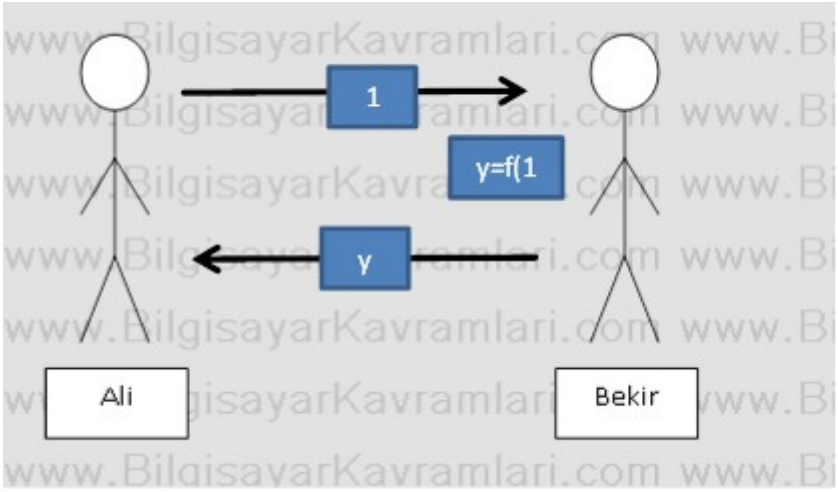
Bekir f(x) fonksiyonunu kendisi seçebilmektedir ve seçtiği bu fonksiyon iki ihtimalden birisi olabilir:

- f(x) fonksiyonu bir sabit fonksiyondur, yani x değerinden bağımsız olarak sürekli 0 veya 1 üreten bir fonksiyon olabilir
- f(x) fonksiyonu eşit miktarda (%50 ihtimalle) 0 ve 1 üreten bir fonksiyondur

Ali yolladığı 1 ve 0 değerlerinin karşılığında aldığı sonuçlara göre, Bekir'in hangi tipte f(x) fonksiyonu kullandığını bulmaya çalışır.

Deutsch problemi, bu bulma işleminin kaç denemede yapılabileceğini sorar.

Bu adımları aşağıdaki şekiller üzerinden açıklamaya çalışalım:



Yukarıdaki şekilde görüldüğü üzere, 1. Adımda, Ali, Bekir'e bir değer yollar (1 veya 0) bu değeri kendi f fonksiyonuna koyan Bekir, sonucu Aliye geri yollar (sonuç yine 1 veya 0 olacaktır).

Yukarıdaki bu işlem istenildiği kadar tekrarlanabilir. Amaç, Ali'nin Bekir'in f fonksiyonunu en kısa sürede tahmin edebilmesidir.

SORU-8: Bell Halleri (Bell States) hakkında bilgi veriniz.

Kuantum işleme (Quantum Computation) konusunda kullanılan ve iki dolanık kubitin (entangled qubit) birbirine göre alabileceği halleri gösterir. Dolanık kubitlerin ikiden fazla olması için kullanılan farklı haller (states) bulunur ancak Bell halleri sadece iki kubit için çalışmaktadır. Bu iki kubitin dolanık olması durumuna ayrıca EPR durumu ismi de verilmektedir. (EPR: Einstein, Podolsky ve Rosen baş harflerinden oluşmaktadır ve 1935 yılında yayınlanan makalelerine atfedilmiştir) Dolanık kubitlerin veri iletişimde kullanımını belirlemekte kullanılır. Bu haller aşağıdaki 4 durumla gösterilebilir.

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |0\rangle_B + |1\rangle_A \otimes |1\rangle_B)$$

$$|\Phi^-\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |0\rangle_B - |1\rangle_A \otimes |1\rangle_B)$$

$$|\Psi^+\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |1\rangle_B + |1\rangle_A \otimes |0\rangle_B)$$

$$|\Psi^-\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |1\rangle_B - |1\rangle_A \otimes |0\rangle_B)$$

Buradaki gösterimde, veri iletişiminde kullanılan Alice ve Bob ikisinden taraflar, A ve B indisleri ile gösterilmiştir. Dolayısıyla, yukarıdaki gösterimde |0>A değeri, Alice için 0 olması veya |1>B gösterimi, Bob için 1 değerinin okunmasını ifade eder (burada kullanılan ket gösterimidir ve detaylı bilgi için Dirac Notasyonu başlıklı yazıyı okuyabilirsiniz)

Yukarıdaki hallerden örneğin $|\Phi^+\rangle$ halini ele alalım.

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |0\rangle_B + |1\rangle_A \otimes |1\rangle_B)$$

Bu halde, Alice tarafından tutulan bilginin 0 veya 1 olması durumu ele alınmıştır. (A indisi içeren |0> ve |1> gösterimleri için). Bu iki ihtimalden (yani 0 veya 1 ihtimallerinden) hangisinin Alice tarafında olduğunu bilmiyor ve rast gele bir değer olarak kabul ediyoruz. (her iki ihtimal için de ½ çarpanı olması hasebiyle (1/2 çarpanının kareler alındığında elde edilecek sonuç olmasından dolayı, denklemde 1 / kök 2 olarak geçmektedir)). $|\Phi^+\rangle$ hali, bize Alice için rast gele bir |0> veya |1> durumunda, Bob için aynı sonucu elde etme ihtimalini verir. Yani Alice için |0> değeri Bob için de oluşuyor ve Alice için |1> değeri Bob için de oluşuyorsa $|\Phi^+\rangle$ durumundan bahsedebiliriz.

Burada Alice için rast gele içerilen bilginin, Bob için aynen ölçülmesi öngörülmüştür. Yani iki adet dolanık kubitin birbiri ile aynı davranması durumu içerilir. Bu durum, veri iletişimi için, örneğin Alice ve Bob ile gösterilen iki tarafın anlaştıkları sonucunu doğurur. Benzer şekilde veri güvenliği açısından da iki tarafın aynı veriyi okuduğu şeklinde yorumlanabilir.

SORU-9: Hadamard Kapısı (Hadamard Gate) hakkında bilgi veriniz.

Hadamard kapıları, kuantum işlemede kullanılan bir kapı türüdür. Kapı basitçe tek kubitlik bir sistemde |1> ve |0> arasında dönüşüm yapmaya yarar.

Bu dönüşümü aşağıdaki şekilde gösterebiliriz. Öncelikle Dirac gösterimindeki kubit değerini hatırlayalım:

$|\psi> = \alpha|0> + \beta|1>$

ψ değeri, yukarıda verilen α değeri kadar 0 ve β değeri kadar 1′dir. Yani bu iki değer arasında bir yerde kabul edilen bir vektördür. Bu vektörün uzunluğunu 1 olarak kabul edersek, Pisagor bağlantısından $|\alpha|^2 + |\beta|^2 = 1$ olmalıdır.

Bu bağlantıda α ve β değerlerini eşit alırsak her değer için ½ olasılık bulunacaktır. Bu durumda yeni kubit değerimiz aşağıdaki şekilde olacaktır.

$$|\psi> = \frac{1}{\sqrt{2}}|0> + \frac{1}{\sqrt{2}}|1>$$

Buradaki karekök alınma sebebi, değerlerin kareleri alındığında ½ sonucunu elde edebilmektir.

Ayrıca Dirac gösteriminden kolon gösterimine çevrilirse

$$|0> = \begin{bmatrix}1\\0\end{bmatrix}$$

$$|1> = \begin{bmatrix}0\\1\end{bmatrix}$$

Şeklinde yazılabilir. Bu yazımdaki değerleri, bir önceki eşitlikte yerine koyarsak:

$$|\psi> = \frac{1}{\sqrt{2}}|0> + \frac{1}{\sqrt{2}}|1>$$ eşitliğini

$$|\psi> = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\0\end{bmatrix} + \frac{1}{\sqrt{2}}\begin{bmatrix}0\\1\end{bmatrix}$$

Şekline geçirmiş oluruz. Denklemi ortak paranteze alıp ilerletirsek aşağıdaki şekilde olasılık değeri 1 elde edilir.

$$|\psi> = \frac{1}{\sqrt{2}}\left(\begin{bmatrix}1\\0\end{bmatrix} + \begin{bmatrix}0\\1\end{bmatrix}\right)$$

$$|\psi> = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix}$$

Yukarıdaki bu kubit gösterimlerini Hadamard kapısından geçirmeyi ve kapının etkisini öğrenmeye çalışalım. Hadamard kapısı, tanım itibariyle aşağıdaki matristen ibarettir.

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

Bu matrisin yukarıdaki Dirac gösteriminde olan kubit ile çarpımını hesaplamamız, kubitin Hadamard kapısından geçmesi sonucunda yaşayacağı etkiyi belirtir.

Dolayısıyla H|0> aşağıdaki şekilde yazılabilir:

$$H|0> = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}|0>$$

$$H|0> = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix}$$

Görüldüğü üzere yukarıdaki şekilde, daha önceden elde ettiğimiz 1 olasılığını görmüş oluruz. Aynı şekilde H|1> dönüşümünü hesaplayabiliriz.

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

$$H|1> = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\begin{bmatrix}0\\1\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\-1\end{bmatrix}$$

Yukarıdaki bu sonuçlar açıldığında dönüşüm daha net bir şekilde görülebilir:

$$H|0> = \frac{1}{\sqrt{2}}|0> + \frac{1}{\sqrt{2}}|1>$$

$$H|1> = \frac{1}{\sqrt{2}}|0> - \frac{1}{\sqrt{2}}|1>$$

Yukarıda görüldüğü üzere aradaki işlem + veya – olarak değişmektedir.

SORU-10: Dolanık Kubitler (Entangled Qubits) hakkında bilgi veriniz.

Kuantum mekaniği üzerinde yapılan çalışmalar göstermiştir ki, dolanık kubitler (entangled qubits) birbiri ile özel bir bağa sahiptir ve bazı kurallara uyarlar. Buradaki bağlantıya çevirim ilişkisi (spin correlation) ismi de verilir.

Basitçe iki kuantum parçacığının aynı anda üretildiğini düşünelim, birisinin yukarı çevirim diğerinin ise aşağı çevirim olduğunu düşünelim. Bu durumda iki kuantum parçacığının dolanık olması söz konusudur ve bu iki parçacığın sürekli olarak ters çevirimde olduğu kabul edilir ve bu duruma çevirim ters ilişkisi (spin anti-correlated) ismi verilir.

Dolanık kubitlerin bir özelliği de bu ilişkilerini her yerde göstermeleridir. Literatürde bu durum için yerel olmayan (non-local) terimi kullanılır. Yani ilişkileri, herhangi bir yerle sınırlı değildir ve kubitlerin ayrı yerlere yollanması durumunda da ilişkileri bozulmaz.

Kuantum haberleşmesi ve güvenli veri iletimi için oldukça önemli olan bu özellik, üretilen iki kübitin farklı yerlerde aynı özelliği göstermesinin yanında, kübitlerden birisinin açılması ve okunması halinde, bu kubitin bozulmasına da dayanmaktadır.

Kubit ölçümleri, kübitin bilgisini bozmaktadır. Bunun anlamı, bir kübit sadece bir kere okunabilir ve bu okumadan sonra kararsız hale gelir.

Şayet elimizde dolanık iki kübit bulunuyorsa, bu kubitlerden birisi okunduğunda artık dolanıklık özelliğini yitirir. Dolayısıyla iki kubit üretilip, bu kubitlerden birisinin karşı tarafa yollanması durumunda, yolda bir saldırgan tarafından okunursa artık orijinal kubit ile olan ilişkisi bozulmuş olur. .

Veri iletişimi ile ilgili daha fazla bilgi için güvenli kuantum haberleşmesi konusuna bakabilirsiniz.

Ayrıca dolanık kübitlerin durumları için bell durumları (bell states) konusu okunabilir.

SORU-11: Kuantum Kapıları (Quantum Gates) hakkında bilgi veriniz.

Kuantum kapıları, mantıksal devre tasarımında bulunan klasik kapılara alternatiftir. Amaç, elektronik devrelerin karar mekanizmasında quantum teknolojisini kullanmaktır.

Klasik kapılarda bulunan ve bitlere göre karar vermeye yarayan mekanizmadan farklı olarak kuantum kapılarında, kubitler (qubits) üzerinden karar verilir. Kuantum kapılarının bir özelliği, geri döndürülebilir olmalarıdır (reversible), yani bir girdi için elde edilen sonuç, sonuçtan girdi olarak verildiğinde, girdi geri elde edilebilir.

Bir mantıksal kapının geri döndürülebilir olması, kapının girdisinden elde edilen çıktının tekrar girdi olması halinde, ilk girdinin geri elde edilebilmesidir. Bu karmaşık cümle ile anlatılmak istenen örneğin L kapısı için L(x) = y gibi bir sonuç alınıyorsa, bu kapının tersi olan L' için L'(y) = x sonucunun alınması beklenir. Veya kapının kendisinin ters olması halinde de L(x) = y ve L(y) = x şartlarının aynı anda sağlanması beklenir.

Örneğin klasik değil kapısı (not gate) geri döndürülebilir kapıdır (reversable). Bunu doğruluk çizelgesine (truth table) bakarak kolayca görebiliriz.

Girdi	Çıktı
1	0
0	1

Görüldüğü üzere L(1) = 0 ve L(0)=1 olmakta, dolayısıyla tersi alınabilir bir kapı olmaktadır.

Buna karşılık, geri döndürülebilirlik (reversible) konusunun daha iyi anlaşılabilmesi için, geri döndürülemez bir kapı olan veya kapısını inceleyelim.

Girdi	Çıktı
00	0
01	1
10	1
11	1

Yukarıdaki doğruluk çizelgesinde (truth table) görüldüğü üzere, herhangi bir çıktının, girdiye verilmesi durumunda, girdinin geri elde edilmesi mümkün değildir. Örneğin L(10) = 1 olmakta ama L(1) = 10 olmamaktadır.

Aynı zamanda herhangi bir L' devresi de yukarıdaki tablonun tersini üretemez. Bunun sebebi, 1 çıktısının 01, 10 veya 11 şeklinde geri döndürülme ihtimali olduğu ve 1 çıktısı alındıktan sonra, orijinal girdinin ne olduğunun tahmininin imkânsız olduğudur.

Ve kapısı örneğini ele alarak, bir kapının geri döndürülebilir olması için giriş ve çıkış bitlerinin sayısının aynı olması gerektiğini tahmin edebilirsiniz. Aslında bu durum basitçe güvercin yuvası kaidesi (pigeonholde principle) ile açıklanabilir ve evet bir kapının geri döndürülebilir olması için giriş biti sayısı ile çıkış biti sayısı eşit olmalıdır.

Şayet giriş bitlerinin sayısı ile çıkış bitlerinin sayısı eşit ise, kapının karakterini, yukarıdaki örneklerde olduğu gibi doğruluk çizelgesi (truth table) şeklinde klasik gösterimden farklı olarak gösterebiliriz. Aslında kuantum kapıları (quantum Gates) için vaz geçilmez olan bu gösterim matris gösterimidir.

Örneğin değil kapısını (not gate) ele alalım ve matriste göstermeye çalışalım.

	0	1
0	0	1
1	1	0

Yukarıdaki matris, okunması kolay olsun diye bir satır (en üstteki) ve bir sütun (en soldaki) eklenerek verilmiştir. Bu matriste, satırlar, girdiyi, sütunlar ise çıktıyı tutmaktadır. Yani tablomuzu aşağıdaki şekilde yorumlayabiliriz

	0	1
0	0 girdisi için, 0 çıktısı alınabilir mi?	0 girdisi için, 1 çıktısı alınabilir mi?
1	1 girdisi için 0 çıktısı alınabilir mi?	1 girdisi için 1 çıktısı alınabilir mi?

Yukarıdaki bu sorulara evet veya hayır cevaplarını vererek evet için 1 ve hayır için 0 yerleştiriyoruz. Örneğin değil kapısı (not gate) 0 için 1 sonucu verir ve 0 için 0 sonucu vermez. Dolayısıyla yukarıdaki doğruluk çizelgesinin matris gösterimini aşağıdaki şekilde yapmak yeterlidir.

0	1
1	0

Yukarıdaki bu matrise bakıldığı zaman, bu matrisin doğruluk çizelgesi (truth table) kolaylıkla anlaşılabilir.

Matris gösteriminin kuantum kapıları için kullanılması durumunda, aslında qubit değerlerinin matrise yerleştirilmesinden bahsediliyor demektir.

Örneğin, $\alpha|0> + \beta|1>$ şeklinde yazılan bir kubit gösterimini vektör olarak modellemek istersek

α
β

Şeklinde bir vektör elde edebiliriz. Bu vektörü değil kapısı (not gate) için girdi ve çıktı olarak modellediğimizde, bir qubit için durum aşağıdaki şekilde olur:

$$X\begin{bmatrix}\alpha\\ \beta\end{bmatrix} = \begin{bmatrix}\beta\\ \alpha\end{bmatrix}$$

Görüldüğü üzere, kubitin tersi alınmıştır. Burada dikkat edilecek bir husus, matriste kullanılan α ve β değerlerinin karmaşık sayılar (complex numbers) olduğudur.

Kuantum Kapılarının bir özelliği, bu kapılarda kullanılan matrisin, vahid masfuf (uniter matrix) olmasıdır.

Çok Kullanılan Kuantum Kapıları

Bu bölümde, kuantum kapılarından çok kullanılanlarını anlatacağız. Teorik olarak sonsuz sayıda kuantum kapısı üretilebilir. Ancak buradaki amaç özellikleri bakımından önemli görülen ve literatürde sıkça rastlananları açıklamaktır.

Hadamard Kapısı

Hadamard kapıları, tek kubitli bir sistemde, aşağıdaki dönüşümleri yaparlar.

|0> değerini $\frac{|0\rangle + |1\rangle}{\sqrt{2}}$ olarak

|1> değerini ise $\frac{|0\rangle - |1\rangle}{\sqrt{2}}$ olarak dönüştürür.

Bu durumda, hadamard kapısının matrisi aşağıdaki şekilde olacaktır:

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

Hadamard kapılarının ismi, bu kapılar için kullanılan matrisin bir hadamard matrisi (hadamard matrix) olmasından gelmektedir.

Aslında hadamard matrislerini, değil kapılrının (not gate) karekökü olarak düşünmek de mümkündür. Görüldüğü üzere, elde edilen sonuç bir vahid masfuftur (uniter matrix)

Pauli X kapısı

Pauli X kapıları, kalsik değil kapısının (not gate), kuantum için uyarlanmış halidir. Yani yazının başında anlatılan ve girişi tersine döndürmeye yarayan kapılar olarak düşünülebilir. Bu durumda matrisi aşağıdaki şekilde olacaktır.

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

Aslında bu kapının özelliği Bloch Küresini (Bloch Sphere) X ekseni etrafında pi radyan kadar döndürmesi ve |0> değerini |1> ve |1> değerini |0> yapmasıdır.

Pauli Y kapısı

Pauli X kapısına benzer olarak bu kapı da Bloch Küresi (Bloch Sphere) üzerinde döndürme işlemi yapmaktadır. Ancak bir önceki kapıdan farklı olarak bu defa Y ekseni üzerinde döndürme işlemi yapılır.

$$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

Pauli Z kapısı

Pauli X ve Y kapılarına benzer şekilde Bloch Küresi üzerinde döndürme işlemi yapılır. Bu defa isminden de anlaşılacağı üzere döndürme işlemi Z ekseni üzerinde olur.

$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

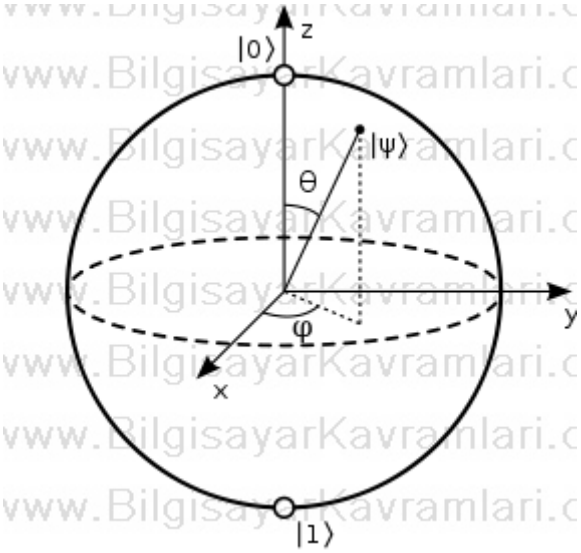
Faz kaydırma kapısı (Phase shift gate)

Bu kapının özelliği, 00, 01 ve 10 için değişiklik yapmamak ama 11 durumu için |1> girdisinin $e^{i\,\Theta}|1\rangle$ girdisine dönüştürmesidir. Yani |1> için, Θ derece döndürme işlemi yapılmaktadır.

$$R(\theta) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{bmatrix}$$

SORU-12: Bloch Küresi (Bloch Sphere) hakkında bilgi veriniz.

Kuantum mekaniğine, Felix Bloch tarafından kazandırılmış, tek kubiti göstermeye yarayan çizimdir. Üç boyutlu bir küredeki herhangi bir nokta, kubitin durumunu göstermektedir. Bu nokta, şayet kürenin yüzeyinde ise, bu durum saf durumdur (pure state) ancak nokta, kürenin içerisinde de olabilir. Bu durumda içsel durum (interior state) olarak adlandırılır.



Yukarıdaki şekilde görüldüğü üzere kürenin merkezine göre eksenler ile yapılabilecek açılar bulunmakta ve bu iki açı Θ ve ϕ sembolleri ile gösterilmektedir.

Ayrıca noktanın merkeze olan uzaklığı ψ sembolü ile gösterilmekte ve bu değer ket gösteriminde (bkz. Dirac Gösterimi (Dirac Notation)) |ψ> olarak tutulmaktadır.

SORU-13: Hadamard Matrisi hakkında bilgi veriniz.

Özellikle matematikte ve bilgisayar bilimlerinin kuantum hesaplama gibi alanlarında kullanılan bir matris (masfuf, matrix) örneğidir. Fransız matematikçi Jaques Hadamard tarafından tasarlanmış ve adıyla anılmıştır. Matrisin en belirgin özelliği, matrisin kare matris olması ve elemanlarının -1 veya +1 değerlerinde olmasıdır. Ayrıca matrisin satırları birbirinden bağımsız olarak diktir (orthogonal). Bunun anlamı, matrisin herhangi iki satırından oluşturulabilecek iki vektörün birbirine dik vektörler olmasıdır (orhogonal vectors).

Hadamard matrisleri, hadamard kodu olarak anılan hata düzeltme kodlamalarında ve hadamard kapıları olarak anılan, kuantum kapılarında kullanılmaları açısından, bilgisayar mühendisliği açısından da önemlidir.

Hadamard Matrislerinin Özellikleri

Hadamard matrisleri için bazı işlemleri ön tanımlı olarak kullanabiliriz. Örneğin bir hadamard matrisinin tersyüzü (transpose) ile çarpımı birim matrisin saklar ile çarpımına dönüştürülebilir.

$$HH^{\mathrm{T}} = nI_n$$

Buradaki sabit değer olan (scalar) n, çarpımda kullanılan hadamard matrisinin boyutudur. Örneğin 5×5 boyutlarında bir matris, kendi tersyüzü ile çarpılırsa n = 5 olacaktır.

SORU-14: Toffoli Kapısı (Toffoli Gate) hakkında bilgi veriniz.

Bilgisayar mühendisliğinin de bir çalışma alanı olan mantıksal devre tasarımı konusunda geçen, ve mucidinin adı ile anılan bir kapı örneğidir. Bu kapının en büyük özelliği evrensel olarak geri döndürülebilir olmasıdır (universally reversable). Literatürde bu kapı için CCNOT (control control not) kapısı ismi de verilmektedir.

Bir mantıksal kapının geri döndürülebilir olması, kapının girdisinden elde edilen çıktının tekrar girdi olması halinde, ilk girdinin geri elde edilebilmesidir. Bu karmaşık cümle ile anlatılmak istenen örneğin L kapısı için L(x) = y gibi bir sonuç alınıyorsa, bu kapının tersi olan L' için L'(y) = x sonucunun alınması beklenir. Veya kapının kendisinin ters olması halinde de L(x) = y ve L(y) = x şartlarının aynı anda sağlanması beklenir.

Örneğin klasik değil kapısı (not gate) geri döndürülebilir kapıdır (reversable). Bunu doğruluk çizelgesine (truth table) bakarak kolayca görebiliriz.

Girdi	Çıktı
1	0
0	1

Görüldüğü üzere L(1) = 0 ve L(0)=1 olmakta, dolayısıyla tersi alınabilir bir kapı olmaktadır.

Buna karşılık, geri döndürülebilirlik (reversible) konusunun daha iyi anlaşılabilmesi için, geri döndürülemez bir kapı olan veya kapısını inceleyelim.

Girdi	Çıktı
00	0
01	1
10	1
11	1

Yukarıdaki doğruluk çizelgesinde (truth table) görüldüğü üzere, herhangi bir çıktının, girdiye verilmesi durumunda, girdinin geri elde edilmesi mümkün değildir. Örneğin L(10) = 1 olmakta ama L(1) = 10 olmamaktadır.

Aynı zamanda herhangi bir L' devresi de yukarıdaki tablonun tersini üretemez. Bunun sebebi, 1 çıktısının 01, 10 veya 11 şeklinde geri döndürülme ihtimali olduğu ve 1 çıktısı alındıktan sonra, orijinal girdinin ne olduğunun tahmininin imkânsız olduğudur.

Ve kapısı örneğini ele alarak, bir kapının geri döndürülebilir olması için giriş ve çıkış bitlerinin sayısının aynı olması gerektiğini tahmin edebilirsiniz. Aslında bu durum basitçe güvercin yuvası kaidesi (pigeonholde principle) ile açıklanabilir ve evet bir kapının geri döndürülebilir olması için giriş biti sayısı ile çıkış biti sayısı eşit olmalıdır.

Şayet giriş bitlerinin sayısı ile çıkış bitlerinin sayısı eşit ise, kapının karakterini, yukarıdaki örneklerde olduğu gibi doğruluk çizelgesi (truth table) şeklinde klasik gösterimden farklı olarak gösterebiliriz. Aslında kuantum kapıları (quantum Gates) için vaz geçilmez olan bu gösterim matris gösterimidir.

Örneğin değil kapısını (not gate) ele alalım ve matriste göstermeye çalışalım.

	0	1
0	0	1
1	1	0

Yukarıdaki matris, okunması kolay olsun diye bir satır (en üstteki) ve bir sütun (en soldaki) eklenerek verilmiştir. Bu matriste, satırlar, girdiyi, sütunlar ise çıktıyı tutmaktadır. Yani tablomuzu aşağıdaki şekilde yorumlayabiliriz

	0	1
0	0 girdisi için, 0 çıktısı alınabilir mi?	0 girdisi için, 1 çıktısı alınabilir mi?
1	1 girdisi için 0 çıktısı alınabilir mi?	1 girdisi için 1 çıktısı alınabilir mi?

Yukarıdaki bu sorulara evet veya hayır cevaplarını vererek evet için 1 ve hayır için 0 yerleştiriyoruz. Örneğin değil kapısı (not gate) 0 için 1 sonucu verir ve 0 için 0 sonucu vermez. Dolayısıyla yukarıdaki doğruluk çizelgesinin matris gösterimini aşağıdaki şekilde yapmak yeterlidir.

0	1
1	0

Yukarıdaki bu matrise bakıldığı zaman, bu matrisin doğruluk çizelgesi (truth table) kolaylıkla anlaşılabilir.

Toffoli kapısına gelince, bu kapının doğruluk tablosu ve matrisi aşağıda verilmiştir.

Girdi		**Çıktı**	
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

Yukarıdaki doğruluk çizelgesinin matris hali de aşağıda verilmiştir.

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

Görüldüğü üzere, toffoli kapısı, özel veya (XOR) şeklinde çalışmaktadır ve ilk biti kontrol bitidir. Yani çıktının ikinci biti (doğruluk çizelgesindeki sağdaki bit), XOR sonucu iken, ilk bit Girdinin ilk biti ile aynıdır.

İki girdi için yukarıda verilen doğruluk çizelgesi ve matris gösterimlerinin, 3 giriş için olanı da aşağıdadır.

Giriş	**Çıkış**
0 0 0	0 0 0
0 0 1	0 0 1
0 1 0	0 1 0
0 1 1	0 1 1
1 0 0	1 0 0
1 0 1	1 0 1
1 1 0	1 1 1
1 1 1	1 1 0

Böylelikle, yukarıdaki 3 bit girişin aslında tek bit olan çıkışının başında iki bitlik kontrol bulunmakta ve yazının başında bahsettiğimiz CCNOT yani kontrol kontrol değil (not) kapısı olmaktadır. Bu tablonun matris gösterimi aşağıdaki şekildedir.

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

SORU-15: Çoklu Kubit (Multiple Qubits) hakkında bilgi veriniz.

Kubitler (qubits), yapıları itibariyle klasik bitlerden farklı olarak ikiden fazla durumda olabilirler. Tek bir kubit için anlaşılabilen ve hesaplanabilen bir durum olmasına karşılık birden fazla kubitin aynı anda kullanılabiliyor olması ayrı bir problem doğurur.

Kalsik iki bit için olası durumlar 00, 01, 10 veya 11 durumlarıdır. Ancak kubitler için 4 durumdan fazlasından bahsedilebilir. Dirac gösterimine göre iki kubit için |00>, |01>, |10> veya |11> durumlarından bahsedebiliriz. Bu kubitlerin superpozisyon ihtimali göz önüne alındığında, her ihtimal için bir karmaşık çarpan olacaktır:

| ψ> = a00 |00> + a01 |01> + a10 |10> + a11 |11>

Bu denklemde yazılan a çarpanları, durumlar arasındaki geçiş miktarını belirlemektedir.

Yukarıdaki denklem için bilinen bir x durumu olursa. Yani x = 00 , x = 01 , x = 10 veya x = 11 durumlarından birisinden söz edersek, yukarıdaki kat sayılar bu ihtimaller arasındaki dağılımı belirtir ve |ax|2 olarak hesaplanabilir. Şayet bitlerden birisi hakkında bir kesinlik bulunuyorsa, ihtimal hesaplaması için |a00|2 + |a01|2 denklemi kullanılabilir. Bu hesaplama işlemine normalleştirme (normalization) ismi verilir ve basitçe {0,1} olma durumlarının karesinin hesabı olarak görülebilir.

Olasılık teorisinden bilindiği üzere toplam olasılık 1 olacağı için, yukarıdaki bütün ihtimallerin toplamı aşağıdaki şekilde ifade edilebilir:

$$\sum_{x\in\{0,1\}^2} |a_x|^2 = 1$$

Yukarıdaki bu durumu, ilk bitin 0 olması ihtimali için geri normalleştirmeye sokarsak (re-normalization), bu durumda, aşağıdaki gibi bir denklemle karşılaşırız:

$$|\varphi' > = \frac{a_{00}|00> + a_{01}|01>}{\sqrt{|a_{00}|^2 + |a_{01}|^2}}$$

Yukarıdaki gösterim Bell hali (Bell State) olarak isimlendirilen özel bir durumdur. Bell halinin tam gösterimi aşağıdaki şekildedir:

$$\frac{|00> + |01>}{\sqrt{2}}$$

Aslında Bell hali, kuantum haberleşmesi ve süper yoğun kodlama için temel teşkil eder. Burada dikkat edilmesi gereken bir nokta, iki kubitlik bir sistemde, birinci kubitin ölçülmesi sonucunda, ikinci kubit için ilk kubit ölçümü ile aynı sonucu vermesidir. Dolayısıyla ölçümler birbiri ile bir şekilde ilişkilidir. Bu ilişkiye kuantum hesaplamada dolanık kubitler (entangled qubits) ismi verilir.

SORU-16: Dirac Gösterimi (Dirac Notation) hakkında bilgi veriniz.

Kuantum hesaplamasının gelişmesi ile birlikte, kubit (qubit) kavramını göstermek için bir notasyona ihtiyaç duyulmuştur. Bu ihtiyaç Dirac tarafından geliştirilen bir gösterimle karşılanabilmektedir. Bazı kaynaklarda bra-ket olarak da geçer.

Bra-ket gösterimi < | > şeklinde sembolize edilebilir. Buradaki bra kısmı <| olurken ket kısmı | > olmuş olur. Yani İngilizcedeki parantez anlamına yakın bir kelimeyi parçalara bölerek (aslında barcket kelimesi, İngilizcede parantez anlamına gelir), parantezi iki alt parçada gösterebiliriz.

Bu gösterim basitçe elimizdekiler ve istediklerimizi ayırarak göstermeye yarar.

Elimizde olanları ket kısmına koyuyoruz. Örneğin |p> gösterimi, parçacığın p momentumunda olduğunu ifade etmektedir. Daha farklı belirgin olarak |p=2.1> gösterimi, parçacığın 2.1 momentumuna sahip olduğunu veya |x=2> gösterimi, parçacığın 2 konumunda bulunduğunu ifade eder. Bu anlamda, elimizdeki bilgileri gösteren ket kısmı, aslında başlangıç vektörü veya başlangıç durumu şeklinde de adlandırılabilir.

Öte yandan <| bra gösterimi ise ulaşmak istediğimiz hali, veya beklediğimiz durumu göstermeye yarar. Örneğin <x=1.25| gösterimi bize, parçacığın, 1.25 konumunda bitmesini istediğimizi veya böyle bir beklentimiz olduğunu gösterir. Bu durumda, örneğin <x=1.25 | x=2 > gösterimi, parçacığın 2 konumunda başlayarak 1.25 konumunda bitmesi anlamına gelir.

Genelde mevcut durumu ifade etmek için ket kısmında psi sembolü kullanılır. Örneğin |ψ> gösterimi, mevcut durumun ψ vektörü olduğunu ifade eder.

Kubitler için olası durumlardan iki tanesi 1 ve 0 olma durumudur ki bu durumda kubitler bizim bildiğimiz klasik bitler gibi davranır. Bu durumları göstermek için |0> veya |1> gösterimi kullanılabilir. Elbette unutulmaması gereken bir durum, kubitlerin, klasik bitlerden farklı değerler alabileceğidir. Örneğin kubitler, 0 ve 1 arasındaki herhangi bir doğrusal değeri alabilir.

$|\psi> = \alpha|0> + \beta|1>$

Şeklindeki gösterimde, ψ değeri, yukarıda verilen α değeri kadar 0 ve β değeri kadar 1′dir. Yani bu iki değer arasında bir yerde kabul edilen bir vektördür. Bu vektörün uzunluğunu 1 olarak kabul edersek, Pisagor bağlantısından $|\alpha|^2 + |\beta|^2 = 1$ olmalıdır.

Ket gösterimi, vektörel bir gösterimdir. Diğer bir deyişle, |v> gösterimi aslında [v] şeklinde gösterilebilen bir kolon vektördür.

Bra gösterimi ise satır vektörüdür.

Örneğin ket gösterimi için aşağıdaki şekilde bir vektörden bahsedilebilir:

|ψ> = [a1 a2 a3 … an]

Benzer şekilde bra gösterimi için yukarıdaki bu matrisin tersyüzü (transpoze) alınmıştır denilebilir:

$$<\varphi| = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$$