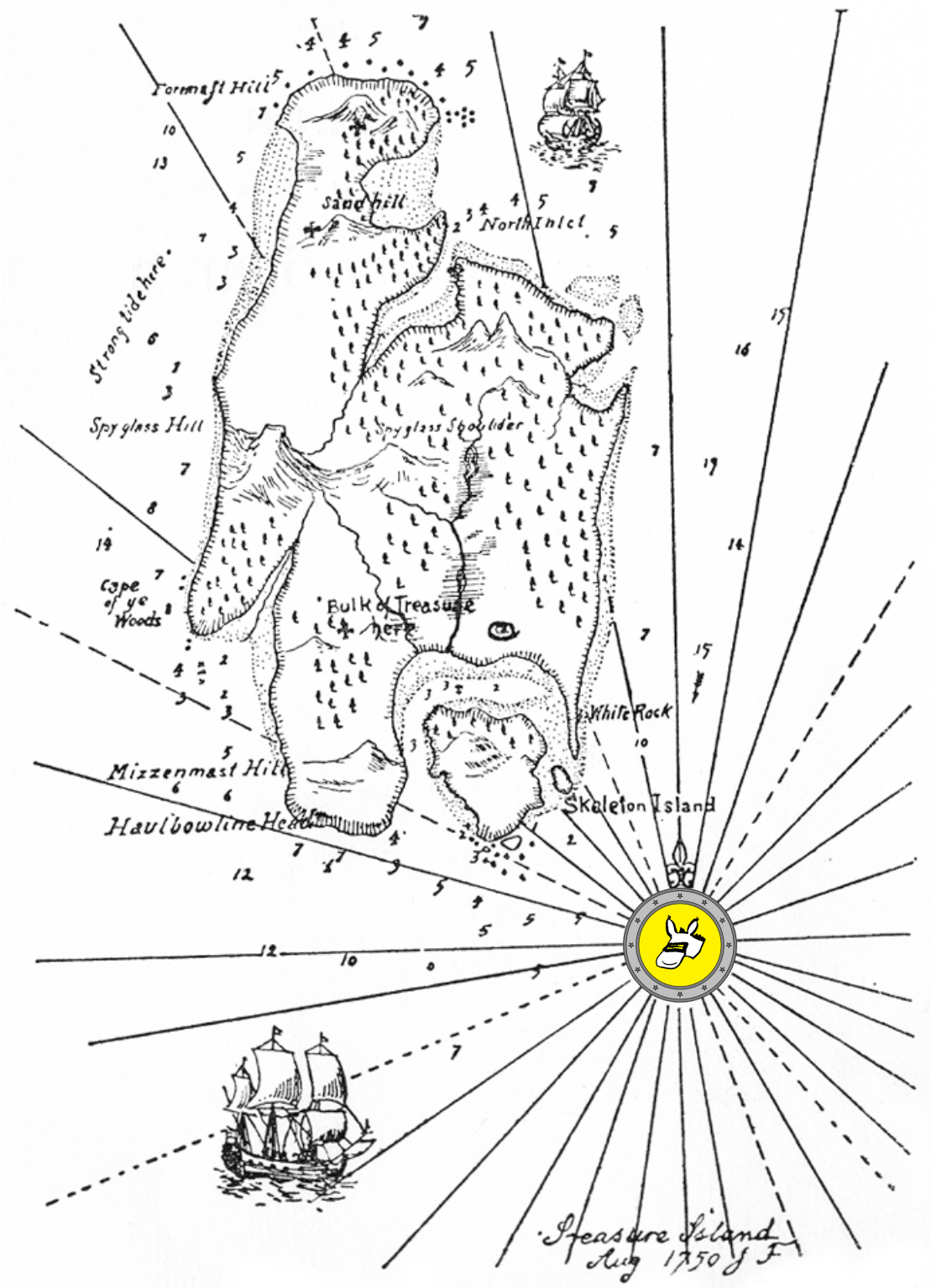
Kvantum Küldetés



Üdvözlünk a *Kvantum Küldetésen*!

Az öthetes küldetésünk célja, hogy elsajátítsd a kvantuminformatika alapjait. A végére meg fogod érteni mik azok a kvantumbitek és kvantumalgoritmusok és, hogy mire is jók. Útközben összebarátkozhatsz Alízzal és Botival, akik 2058-ban élnek és (akárcsak te) tanítás után kvantumszámítógépekkel bütykölnek. Míg a tudósok ma még laboratóriumokban építik a valódi kvantumszámítógépeket, 2058-ra a kvantumszámítógépek mindenhol ott lesznek, még a zsebedben is! Azonban, mint minden technológiát, így a kvantumszámítógépeket sem mindenki használja jó célokra, ezért segítened kell két barátodnak, Alíznak és Botinak, hogy különböző trükköket kiötölve megvédjék magukat a gonosz hacker, Éva ármánykodásaitól. Sok szerencsét a küldetéshez!

Szívélyes kvantum-üdvözlettel,
Maris Ozols & Michael Walter

Tanács az olvasónak

A jegyzetben sok (zöld szöveg dobozba keretezett) gyakorló feladatot és (pirosba keretezett) házi feladatot találtok. A gyakorló feladatok arra szolgálnak, hogy menet közben kipróbáld mennyire sajátítottad el az olvasott anyagot, és mindegyikhez találsz megoldást is a fejezetek végén. A házi feladatokat pedig az online kurzus során hétről-hétre lehet beadni. Néhány feladat „opcionális" jelzéssel van ellátva – úgy gondoljuk, hogy ezek nem feltétlenül szükségesek a kurzus követéséhez, de hasznosak kiegészítő gyakorlásra. Néhány feladatot pedig „csillagos" jelzéssel is elláttunk – ezek egy kicsit nehezebbek a többinél!

Köszönetnyilvánítás

Szeretnénk köszönetet mondani az online kurzus lebonyolításában való közreműködésért segítőinknek: Doutzen Abma, Sebastian Bach, Valerie Bettaque, Amalia Böttger, Milo Camardese, Arjan Cornelissen, Bas Dirkse, Oliver Dorogi, Jari Egbers, Yassine Ferjani, Marten Folkertsma, Koen Groenland, Galina Pass, Philip Verduyn Lunel, Anurudh Peduri, Simon Schmidt, Quinten Tupker, Mees de Vries, Jordi Weggemans, Peter Ypma. Hálásak vagyunk továbbá Craig Gidneynek, a QUIRK kvantumszimulátor megalkotójának, aminek alapján a QUIRKY-t készítettük. Végül szeretnénk köszönetet mondani minden lelkes diáknak, aki részt vett az online kurzuson.

Kvantum Küldetés

Maris Ozols és Michael Walter

2023 November

Tartalomjegyzék

1. Küldetés: A véletlen megzabolázása

Üdvözlünk *A Kvantum Küldetés* első hetében – izgalmas kaland előtt állsz!

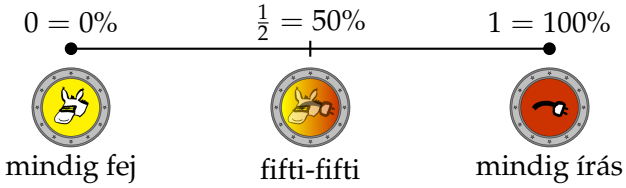
A kvantuminformatika lenyűgöző téma, amely könnyen magával ragadhatja a képzeletet. A fő forrása annak, amiért ennyire lenyűgöző, az a kvantumvilág furcsasága. Azonban ez egyben a fő forrása a zavarodottságunknak is. Valóban, nem nehéz belegabalyodni a kvantumvilág furcsaságaiba vagy eltévedni egy kvantumszámítógép exponenciálisan nagy állapottérében. Hogy elkerüld ezeket a problémákat, először fel kell készülnöd azzal, hogy megismerkedsz a valószínűségek világával. Ha egyszer már a valószínűségek mestere vagy, akkor képes leszel kinyitni az ajtót a kvantumvilág felé is!

Az első küldetés célja, hogy megismerd a valószínűségeket és a valőszínűségi biteket: mik a valőszínűségi bit állapotai, mik az engedélyezett műveletek rajta, és hogyan tudunk információt nyerni egy valőszínűségi bitből mérés által? A második küldetés fő fókusza a kvantumbitek lesznek, amelyek nagyon hasonlítanak a valőszínűségi bitekre.

1.1. Véletlen bitek

A valószínűségek arra szolgálnak, hogy mennyiségi becslést adjunk események bekövetkezési esélyeiről – minél valószínűbb egy esemény, annál nagyobb a valószínűsége. Egy biztosan bekövetkező esemény valószínűsége 1 (valóban, 100% az esélye), míg egy soha be nem következő esemény valószínűsége 0.

Példaként gondolhatunk egy pénzérme feldobására. Ha feldobsz egy érmét és kezeddel lefeded anélkül, hogy ránéznél, két lehetséges esemény van – az érme „fejet” vagy „írást” mutat. Egy *szabályos* érménél a két esemény egyenlő valószínűséggel következik be, tehát mindkettőhöz $\frac{1}{2} = 0,50$ azaz 50% valószínűséget rendelünk (ezt jelöli az 1.1. ábrán). Az érme azonban lehet torzított is, és nagyobb valószínűséggel esik egyik oldalára, mint a másikra. Attól függően, hogy mennyire torzított, elképzelhetünk egy egész spektrumot: egy rendkívül torzított érme mindig „fejet” mutathat, míg egy másik mindig „írást” (lásd és az 1.1. ábra bal és jobb oldalán). Az első érme valószínűsége, hogy „írást” mutat, 0 (mivel mindig „fejet” mutat), míg a másodiké 1.



1.1. ábra. Egy valőszínűségi bit, amely egy véletlenszerű szamár érme állapotát írja le. A „fej” oldal egy szamár fejével, míg az „írás” oldal egy szamár farkával van megjelölve. (Ennek az oka, hogy angolul a „head” és „tail” (farok) kifejezést használják a „fej” és „írás” helyett.)

Mivel nem akarunk különféle formájú és méretű érmék pontos konstrukciójáról gondolkodni, érdemes a pénzfeldobás leírására szolgáló információkat absztrahálni. Ezt úgy érjük el, hogy a „fej” és „írás” eseményeket a bitek 0 és 1 értékeivel társítjuk. Ekkor egy pénzfeldobást két valószínűséggel írhatunk le: A p_0 valószínűséggel, hogy az eredmény 0 („fej”), és a p_1 valószínűséggel, hogy az eredmény 1 („írás”). Egy ilyen bit, amely a két lehetséges értékét meghatározott valószínűségekkel veszi fel, valőszínűségi bitnek nevezik. Ne feledd, hogy a két valószínűség $p_0, p_1 \geq 0$ szükségszerűen 1-re összegződik: $p_0 + p_1 = 1$. Ha az érme, mint a fentebbi példában, szabályos, akkor, mint fent leírtuk, $p_0 = p_1 = \frac{1}{2}$ kell, hogy legyen.

Észrevetted, hogy elég lenne p_0 és p_1 közül csak az egyik valószínűséget megadni, mivel mindkettő megkapható a másikból? A tisztaság érdekében mindig megadjuk mindkettőt, és vektorként írjuk le:

$$p = \begin{pmatrix} p_0 \\ p_1 \end{pmatrix}. \tag{1.1}$$

Ezt a vektort valószínűségi eloszlásnak vagy a valószínűségi bit állapotának nevezzük. Ez a vektor jelölés nemcsak praktikus, hogy az összes valószínűséget egy szép táblázatban ábrázoljuk, hanem lehetővé teszi számunkra egy valószínűségi bit geometriai megjelenítését is. Ezenkívül segít nekünk párhuzamokat látni a valószínűségi bitek és a kvantumbitek között.

Azokat az állapotokat, amelyeknél az eredmény mindig 0 („fej”, ) vagy mindig 1 („írás”, ), determinisztikusnak nevezzük. Ez megfelel a fentebb tárgyalt „rendkívül torzított” érméknek, ahol a pénzfeldobás eredménye determinisztikus, azaz előre meghatározott melyik oldala fog felfelé mutatni az érmének a dobást követően. Az 1.1 egyenletnek megfelelően ezek az állapotok megfelelnek azoknak a valószínűségi eloszlásoknak, ahol $p_0 = 1$, $p_1 = 0$ illetve $p_0 = 0$, $p_1 = 1$. Mivel ezeket az állapotokat gyakran használjuk, érdemes bevezetni a következő rövidítést:

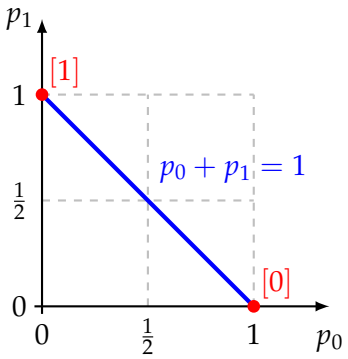
$$[0] = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad [1] = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{1.2}$$

Ez a jelölés kezdetben egy kicsit zavaró lehet, de elképzelheted $[0]$ és $[1]$ mint  és  vagy mint [fej] és [írás], ha úgy tetszik.

Ez a két állapot képezi az összes állapot bázisát. Ez azt jelenti, hogy az összes többi állapotot ezen két állapot lineáris kombinációjaként írhatjuk fel. Konkrétan:

$$\begin{pmatrix} p_0 \\ p_1 \end{pmatrix} = p_0 \begin{pmatrix} 1 \\ 0 \end{pmatrix} + p_1 \begin{pmatrix} 0 \\ 1 \end{pmatrix} = p_0[0] + p_1[1]. \tag{1.3}$$

Mivel az állapotok vektorok, egy kétdimenziós koordináta-rendszerben ábrázolhatjuk őket. A valószínűségi bit lehetséges állapotai pontosan megfelelnek annak a szakasznak, amely a bit determinisztikus állapotainak megfelelő $[0]$ és $[1]$ pontok között van (lásd 1.2. ábra).



1.2. ábra. A kék szakasz megfelel a valószínűségi bit állapotainak.

1.1. Gyakorló feladat: A kék szakasz megértése

Az 1.2. ábra szerint a valószínűségi bit lehetséges állapotai egy szakaszt alkotnak. Gondold át, hogy ez miért van, és próbálj meg válaszolni a következő kérdésekre:

1. Miért fekszenek egy valószínűségi bit lehetséges állapotai egy vonalon?
2. Miért ér véget ez a vonal a koordináta tengelyeken és nem megy tovább?

3. A szakaszon melyik pont felel meg egy szabályos érmének?

1.1.1. Valószínűségek szorzása

Ha feldobsz két érmét, mi a valószínűsége annak, hogy mindkét érme "fej" lesz? Tegyük fel, hogy a két érmét valőszínűségi bitekkel írjuk le

$$a = \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} \qquad b = \begin{pmatrix} b_0 \\ b_1 \end{pmatrix} \tag{1.4}$$

ahol a 0 kimenetel a "fejnek" felel meg, az 1 kimenetel pedig az "írásnak". Ekkor az a érme esetében a "fej" valószínűsége a_0, míg a b érme esetében b_0. (Nem feltételezzük, hogy az érmék tisztességesek, így ezek a valószínűségek nem feltétlenül 50%-ok.) Annak a valószínűsége, hogy mindkét érme egyszerre "fejet" mutat, az az egyes események valószínűségének *összeszorzásával* kapható meg:

$$p_{00} = a_0 b_0. \tag{1.5}$$

Vegyük észre, hogy $p_{00} \leq a_0$ és $p_{00} \leq b_0$, mivel $a_0 \leq 1$ és $b_0 \leq 1$. Ez érthető, hiszen annak a valószínűsége, hogy mindkét érme egyszerre "fejet" mutat, nem lehet nagyobb (sőt, átalában kisebb), mint annak a valószínűsége, hogy az egyik konkrét érme "fejet" mutat. Hasonlóan kiszámíthatjuk az összes többi fej és írás kombináció valószínűségét. A négy esetet az alábbi táblázatban foglaljuk össze:

$$\begin{aligned} p_{00} &= a_0 b_0, & p_{01} &= a_0 b_1, \\ p_{10} &= a_1 b_0, & p_{11} &= a_1 b_1. \end{aligned} \tag{1.6}$$

Két eseményt függetlennek nevezünk, ha azok két különböző forrásból származnak, és az egyik bekövetkezése semmit nem árul el a másik bekövetkezéséről. Általában az ilyen helyzeteket az "és" szóval írjuk le. Például: "az első érme fej *és* a második érme írás". A valószínűségeket összeszorozzuk, ha azt szeretnénk megtudni, hogy két független esemény egyszerre történt-e.

1.2. Gyakorló feladat: Valószínűségek szorzása

Alice unatkozik matekórán, és elkezdi nézni a digitális óráját. Az órájának másodperckijelzője 00 és 59 között mutathat értékeket. Tegyük fel, hogy Alice a következő percen belül valamelyik véletlenszerű pillanatban ránéz az órája másodpercmutatójára.

1. Mekkora a valószínűsége annak, hogy 00-t lát?
2. Mekkora a valószínűsége annak, hogy az utolsó számjegy 0?
3. Mekkora a valószínűsége annak, hogy az első számjegy 0?
4. Indokold meg, hogy a számjegyek miért függetlenek egymástól. Ellenőrizd az 1. kérdésre adott válaszodat a 2. és 3. kérdések valószínűségeinek szorzásával.

1.1.2. Valószínűségek összeadása

Most nézzük a következő, valamivel bonyolultabb problémát. Tegyük fel újra, hogy feldobod az a és b érméket. Mekkora a valószínűsége annak, hogy mindkét érme ugyanazt az eredményt mutatja? Ez kétféleképpen történhet meg – vagy mindkét érme "fej", vagy mindkét érme "írás". Már tudjuk az 1.6. egyenlet alapján, hogy e két egyedi esemény valószínűségei a következők:

$$p_{00} = a_0 b_0, \qquad p_{11} = a_1 b_1.$$

Ekkor annak a valószínűségét, hogy e két esemény egyike bekövetkezik, a valószínűségek *összeadásával* kapjuk meg:

$$p_{00} + p_{11} = a_0 b_0 + a_1 b_1. \tag{1.7}$$

Ha egy kísérlet több különálló kimenetelét összevonod, akkor a valószínűségeik összeadódnak. Az ilyen kombinált eseményeket általában a „vagy" szóval lehet leírni. Például: „mindkét érme fej *vagy* mindkét érme írás". Vigyázat: ez csak akkor működik, ha a lehetőségekben nincs átfedés!

1.3. Gyakorló feladat: Valószínűségek összeadása

Boti is unatkozik matekórán. Észreveszi, hogy Alíz az óráját bámulja, ezért ő is ránéz a saját órájára. Meglepetésére az órája másodperckijelzője 44-t mutat, amit Boti nagyon valószínűtlennek tart. Mekkora a valószínűsége annak, hogy mindkét számjegy ugyanaz, ha Boti egy véletlenszerű pillanatban ránéz az órájára egy percen belül?

Most, hogy már tudod, mikor kell valószínűségeket összeadni és mikor kell szorozni, próbáld meg megoldani az első házi feladatodat!

1.1. Házi feladat: Ellenkező érmék

Alíz két érmét dob fel, melyeket a-nak és b-nek hívunk, a következő valószínűségi eloszlásokkal:

$$a = \begin{pmatrix} 2/3 \\ 1/3 \end{pmatrix}, \qquad b = \begin{pmatrix} 3/4 \\ 1/4 \end{pmatrix}.$$

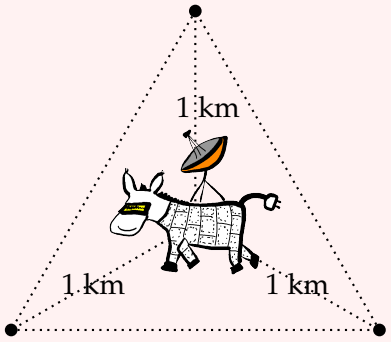
Mekkora a valószínűsége annak, hogy a két érme *ellentétes* eredményt mutat?

1.1.3. Véletlent használó számítások

Van egyáltalán haszna a valőszínűségi biteknek a számításokban? Elsőre úgy tűnhet, hogy nem különösebben hasznosak, mivel egy hagyományos bit 0 és 1 értékei határozott ismeretet képviselnek, míg egy valőszínűségi bit értékei *közelítő* ismeretet (vagy az ismeret *hiányát*) fejezik ki. Miért pazarolnám a számítógépem tárhelyét olyan valőszínűségi bitek tárolására, amelyek az információ hiányát tükrözik, ha helyette tárolhatnám a tényleges információt, még ha az hiányos is? A valőszínűségi bitek előnye, hogy pontosabban képviselik a részleges ismeretet – ha nem tudsz valamit, jobb beismerni, és véletlenszerűen tippelni, mint úgy tenni, mintha biztosan tudnád a helyes választ. Ezt szemlélteti az alábbi probléma, ahol Alice szamárrobotjának döntést kell hoznia anélkül, hogy teljes információval rendelkezne.

1.2. Házi feladat: Alíz szamara

Alíz úgy akarja programozni a szamárrobotját, hogy az önállóan el tudjon menni egy töltőállomásra, és feltöltse magát. Három közeli állomás van, mindegyik 1 km távolságra a szamártól, amelyek egy egyenlő oldalú háromszöget alkotnak, a szamárral a közepén. Alíz szamara csak annyi akkumulátorral rendelkezik, hogy 2,8 km-t tudjon megtenni.



Alíz egy programot fog feltölteni a szamaras robotjára, amely megmondja neki, hova menjen, azonban tudja, hogy gonosz osztálytársa, Éva megpróbálja szabotálni őt. Mivel Éva mindent el tud olvasni, ami a Wi-Fi-n keresztül kerül továbbításra, Éva is láthatja, milyen programot tölt

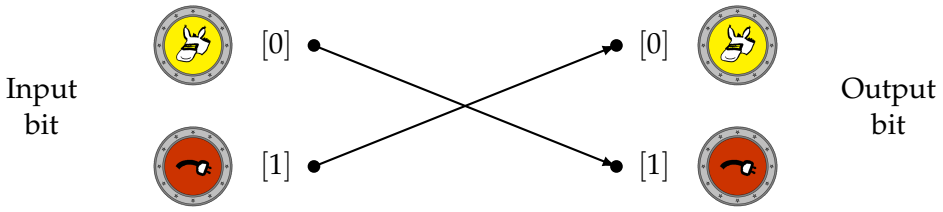
fel Alíz. Ezért, miután a programot feltöltötte, Alíz le fogja csatlakoztatni a szamarát a Wi-Fi-ről, hogy Éva ne tudja követni a mozgását. Miközben a szamár sétál, Éva csak úgy tudja szabotálni, hogy feltöri és letiltja azokat a töltőállomásokat, melyek meglátogatására be van programozva. Éva azonban csak két állomást tud leállítani, mielőtt a behatolását észlelik. Mivel Éva nem tudja követni a szamár mozgását, csak Alíz programja alapján kell döntenie, melyik két töltőállomást tiltja le.

Kérdések:

1. Hány töltőállomást tud meglátogatni a szamár, mielőtt lemerül az akkumulátora?
2. Tegyük fel, hogy Alíz úgy programozza be a szamarat, hogy az egy előre meghatározott sorrendben látogassa az állomásokat. Meg tudja Éva akadályozni, hogy eljusson egy működő töltőállomáshoz? Ne feledd, hogy Éva teljes hozzáféréssel rendelkezik Alíz programjához, így tudja, milyen sorrendben van beprogramozva a szamár az állomások meglátogatására.
3. Tegyük fel, hogy Alíz úgy programozza be a szamarat, hogy az véletlenszerűen döntse el, hova megy. (Bár Éva látja, hogy Alíz ezt programozta be, nem tudja előre megjósolni, milyen döntéseket fog hozni a szamár, miután elindul.) Milyen véletlenszerű stratégiát kellene Alíznak feltöltenie a szamárra, és milyen hackelési stratégiát kellene Évának alkalmaznia ennek ellensúlyozására? Mi a valószínűsége annak, hogy Alíz szamara sikeresen elér egy működő töltőállomást, ha mind Alíz, mind Éva optimális stratégiákat alkalmaznak?

1.2. Műveletek egy valószínűségi biten

Miután az információbiteket vektorokkal írtuk fel, ezeken a biteken végrehajtott műveleteket lineáris transzformációkkal is felírhatjuk, és a lineáris algebra eszközeit használhatjuk. Például vegyük azt a műveletet, amely a szamárérme "fejét" és "írását" cseréli fel:



Ezt a műveletet NOT-ként jelöljük, és matematikailag a következőképpen írjuk fel:

(1.8)

Az 1.2. egyenlet jelölését használva így is írhatjuk:

$$\mathrm{NOT}\,[0] = [1], \qquad \mathrm{NOT}\,[1] = [0]. \tag{1.9}$$

Figyeljük meg, hogy a NOT p rövidítése a NOT(p)-nek – mindkettő azt jelenti, hogy a NOT művelet egy p vektoron hat.[1] Általában nagybetűkkel fogjuk írni a műveleteket, hogy megkülönböztessük őket a számoktól és vektoroktól.

Akárcsak az 1.2. egyenletben, a $[0]$ és $[1]$ vektorok a valószínűségi bit determinisztikus állapotait, a 0-t és az 1-et képviselik. Mivel a NOT művelet kicseréli ezt a két vektort, negálja a

[1] Írhatnánk úgy is, hogy NOT $\cdot\, p$, mivel ez a művelet valójában egy mátrix-vektor szorzásnak felel meg.

bit értékét. Pontosan ezért neveztük "NOT"-nak – ez a logikai negációt jelenti! A NOT művelet egy egyszerű alkalmazása az adatok bevitele a számítógépbe. Ha a számítógéped összes bitje kezdetben 0-ra van állítva, néhányat 1-re változtathatsz az adatok beviteléhez – ez gyakran a számítás első lépése.

Hogyan definiáljuk a NOT műveletet egy valószínűségi biten $p = \binom{p_0}{p_1}$? p_0 valószínűséggel a bit nulla, és egyesre vált. p_1 valószínűséggel a bit egyes, és nullára vált. Így a NOT művelet hatása egy valószínűségi biten egyszerűen:

$$\text{NOT} \begin{pmatrix} p_0 \\ p_1 \end{pmatrix} = \begin{pmatrix} p_1 \\ p_0 \end{pmatrix}. \tag{1.10}$$

Ez különösen az 1.9. egyenlet esetén adja vissza az eredményt, amikor $p_0 = 1$ (és $p_1 = 0$) vagy $p_0 = 0$ (és $p_1 = 1$). A NOT műveletet és az 1.10. egyenletet intuitívan úgy képzelhetjük el, mint egy feldobott, de még meg nem nézett érmét. A NOT műveletnek megfelelően az érmét megfordítjuk (ismét anélkül, hogy megnéznénk).

1.4. Gyakorló feladat: A NOT művelet vizualizálása

Ahogy az 1.2. ábrán látható, a valószínűségi bit összes lehetséges állapota egy szakasznak felel meg. Próbáljuk megvizualizálni, hogyan alakítja át a NOT művelet ezt a szakaszt.

1. Vegyünk egy tetszőleges[a] pontot a (p_0, p_1) koordinátákkal ezen a szakaszon. Hova küldi ezt a pontot a NOT művelet?
2. Hova küldi a szakasz két végpontját?
3. Van-e olyan pont a szakaszon, amely önmagára vetítődik?

[a] Amikor "tetszőleges"-et mondunk, akkor azt értjük alatta, hogy a számításnak p_0 és p_1 bármilyen választására működnie kell. Gyakran az a legjobb, ha minden lépést, beleértve a végső választ is, p_0 és p_1 formájában írjuk le, ezeket ismeretlen számokként kezelve.

1.2.1. Lineáris kiterjesztés

Hogyan definiáljuk az $M\left(\begin{smallmatrix} p_0 \\ p_1 \end{smallmatrix}\right)$ valószínűségi bit, ahol M egy tetszőleges művelet egy biten? Mint korábban, feltételezzük, hogy tudjuk, hogy hogyan hat M a bit két lehetséges értékére, és $M[0]$-t írunk a művelet eredményére, amikor a bemeneti bit nulla, és $M[1]$-et, amikor a bemeneti bit egyes. (A NOT művelet esetében is pontosan ezt a jelölést használtuk az 1.9. egyenletben.) Próbáljuk meg alkalmazni ugyanazt az érvelést, mint fent. p_0 valószínűséggel nulla a bit értéke, amikor $M[0]$-t kapunk a M művelet alkalmazása után, p_1 valószínűséggel pedig egy, amikor helyette $M[1]$-et kapunk. Összességében látható, hogy a helyes definíció

$$M \begin{pmatrix} p_0 \\ p_1 \end{pmatrix} = p_0\, M[0] + p_1\, M[1], \tag{1.11}$$

ahol $p_0\, M[0]$ azt jelenti, hogy az $M[0]$ vektort megszorozzuk a p_0 valószínűséggel.

1.5. Gyakorló feladat: NOT művelet valószínűségi biteken

Mutasd meg, hogy ha M a NOT művelet, akkor az 1.11. egyenlet pontosan az 1.10. egyenletet adja vissza.

Az 1.3. egyenletet használva az 1.11. egyenletet a következő módon is írhatjuk:

$$M\Big(p_0\,[0] + p_1\,[1]\Big) = p_0\, M[0] + p_1\, M[1]. \tag{1.12}$$

Figyeljük meg, hogy a két oldal közötti különbség a műveletek sorrendjében van: a bal oldalon először lineáris kombinációt veszünk, majd alkalmazzuk az M műveletet, míg a jobb oldalon először az M műveletet alkalmazzuk, majd csak ezután vesszük a lineáris kombinációt. Ez az egyenlet nagyon hasonlít a számok körében jól ismert szabályra $a(b+c) = ab + ac$ (a "disztributívitásra").

Ha M egy művelet valószínűségi biteken, amely kielégíti az 1.12. egyenletet, akkor azt mondjuk, hogy M lineáris. Az 1.11. és 1.12. egyenletekben alkalmazott szabályt, amellyel kiterjesztjük M-et bitekről valószínűségi bitekre, linearis kiterjesztésnek nevezzük. Ugyanez az elv a kvantumbitek esetén is jó szolgálatot fog tenni.

1.2.2. Véletlen műveletek

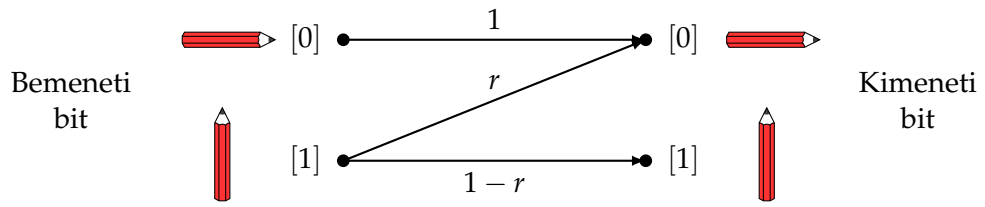
Vegyük észre, hogy az 1.11. egyenlet levezetése során valójában nem feltételeztük, hogy $M[0]$ és $M[1]$ ismét a bit két determinisztikus állapota közül az egyik, $[0]$ vagy $[1]$ volt (bár a NOT művelet esetében ez történt). Ez azt jelenti, hogy az 1.11. egyenlet akkor is működik, ha $M[0]$ vagy $M[1]$ valószínűségi bitek! Ebben az esetben azt mondjuk, hogy M egy véletlenszerű művelet.

Az egyik legegyszerűbb példája a véletlenszerű műveleteknek a következő: Képzeld el, hogy $[0]$ állapotot úgy kódolod, hogy egy ceruzát vízszintesen az asztalra helyezel, és $[1]$ állapotot úgy, hogy függőlegesen állítod fel. Ha óvatosan megütöd az asztalt a kezeddel, a ceruza leeshet, megváltoztatva állapotát $[1]$-ről $[0]$-ra. Azonban, ha már vízszintesen feküdt, az $[0]$ állapota nem változik. Így az asztal megütése *véletlenül visszaállítja* a ceruza állapotát $[0]$-ra; minél erősebben ütsz, annál valószínűbb, hogy visszaállítod.

Matematikailag a véletlen alapú visszaállítás egy $R(r)$ művelettel írható le, amely így definiálható

$$R(r)[0] = [0] = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad R(r)[1] = r[0] + (1-r)[1] = \begin{pmatrix} r \\ 1-r \end{pmatrix}, \tag{1.13}$$

ahol $r \in [0,1]$ a *visszaállítási* valószínűség. E művelet hatását az alábbiak szerint képzelheted el:



A linearitás révén ezt a műveletet minden állapotra kiterjeszthetjük:

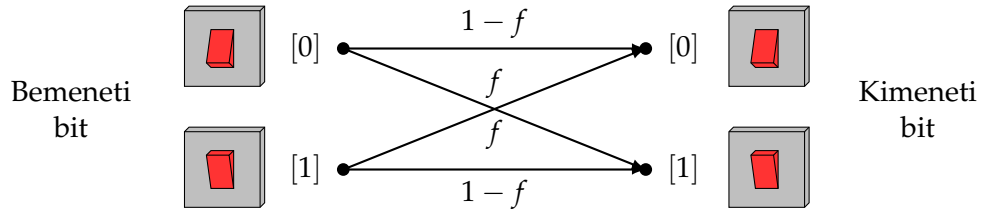
$$\begin{aligned} R(r)\begin{pmatrix} p_0 \\ p_1 \end{pmatrix} &= p_0 R(r)[0] + p_1 R(r)[1] \\ &= p_0 \begin{pmatrix} 1 \\ 0 \end{pmatrix} + p_1 \begin{pmatrix} r \\ 1-r \end{pmatrix} = \begin{pmatrix} p_0 + p_1 r \\ p_1(1-r) \end{pmatrix}. \end{aligned}$$

Ennek az egyenletnek egy speciális esete az0, hogy $R(0)$ egyáltalán nem változtatja meg az állapotot, míg $R(1)$ bármely állapotot $[0]$-ra állít vissza.

Egy másik érdekes példa a véletlenszerű műveletre a véletlen átbillentés, vagy felcserélés művelet $F(f)$, amely f valószínűséggel megfordítja a bemeneti bitet, és $1-f$ valószínűséggel változatlanul hagyja:

$$F(f)[0] = (1-f)[0] + f[1], \qquad F(f)[1] = f[0] + (1-f)[1], \tag{1.14}$$

ahol $f \in [0,1]$ az *átbillentés* valószínűsége. Intuitívabb módon képzeld el, hogy a $[0]$ és $[1]$ egy falon lévő villanykapcsoló két állapotát jelképezik. Ha párnát dobsz a kapcsolóra, csak f valószínűséggel fogod sikeresen eltalálni és átbillenteni, $1-f$ valószínűséggel pedig változatlan marad. Az $F(f)$ működését tehát így képzelheted el:



A következő feladat segít jobban megismerkedni a véletlen átbillentés műveletével.

1.6. Gyakorló feladat: Véletlen átbillentés

$F(f)$ a véletlen átbillentés műveletét jelöli akárcsak az 1.14. egyenletben.

1. Írd le az $F(f)$ $[0]$ és az $F(f)$ $[1]$ vektorokat.
2. Milyen f értéknél működik az $F(f)$ NOT műveletként? Hogyan készíthetünk elő egy valószínűségi bitet egy tetszőleges $\binom{p}{1-p}$ állapotban $[0]$-ból az F segítségével?
3. Terjeszd ki lineárisan az $F(f)$ műveletet valószínűségi bitekre $F(f)\binom{p_0}{p_1}$ kiszámításával.
4. Legyen $\binom{p_0}{p_1}$ egy tetszőleges valószínűségi eloszlás. Mutasd meg, hogy $F(1/2)\binom{p_0}{p_1} = \binom{1/2}{1/2}$.

A feladat utolsó része megmutatja, hogy az $F(1/2)$ mindig az egyenletes eloszlást állítja elő, függetlenül a bemeneti eloszlástól. Ez használható egy elfogulatlan érme feldobásának szimulálására. Valójában a következő feladatban megmutatod, hogy az átbillentési valószínűség f gondos beállításával az $F(f)$ segítségével meg is *változtathatod* egy adott érme torzítottságát.

1.3. Házi feladat: Csokoládé érme

Ma van Boti születésnapja! Mivel szereti a csokoládét, Alíz úgy dönt, hogy készít neki egy csokoládé érmét. Hogy különlegesebb legyen, az érme formájának olyannak kell lennie, hogy ha az asztalon megpörgeted, $q = 5/15$ valószínűséggel érjen földet a S oldala, ami Boti születésnapját, május 15-ét jelképezi. Több különböző formájú érme kipróbálása után Alíznak sikerül olyan csokoládé érmét készítenie, amely a megfelelő valószínűséggel rendelkezik. Nagyon izgatottan az asztalon hagyja, és elrohan a boltba, hogy vegyen egy szép születésnapi üdvözlőlapot.

Sajnos, amikor visszatér, Alíz észreveszi, hogy az érme a napon maradt, és az éle megolvadt. Kipróbálás után Alíz megállapítja, hogy az új valószínűség, hogy S oldala földet ér, $p = 4/15$. Mivel nincs idő a probléma megoldására, Alíz azt írja a születésnapi üdvözlőlapra, hogy miután az érme földet ért, Botinak f valószínűséggel át kell billentenie, és csak akkor fogja a S oldalt a megfelelő q valószínűséggel látni. Segíts Alíznak meghatározni a megfelelő f értéket.

Ötlet: A p, q és f mennyiségekre teljesülnie kell az alábbi egyenletnek: $F(f)\binom{p}{1-p} = \binom{q}{1-q}$.

1.7. Gyakorló feladat: Átbillentés reset és NOT segítségével (opcionális, csillagos)

Hogyan építheted fel az $F(f)$-t az $R(r)$ és NOT műveletek segítségével?

1.3. Egy valószínűségi bit mérése

Ha feldobsz egy szabályos érmét és azonnal letakarod, fogalmad sincs, melyik oldalára esett. Ebben a helyzetben az érme állapotáról szerzett tudásodat az egyenletes eloszlás írja le.

$$= \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix} = \frac{1}{2}[0] + \frac{1}{2}[1]. \tag{1.15}$$

Ha azonban felfeded az érmét és "fejet" látsz, az ismereted frissül a következőre:

$$= [0], \tag{1.16}$$

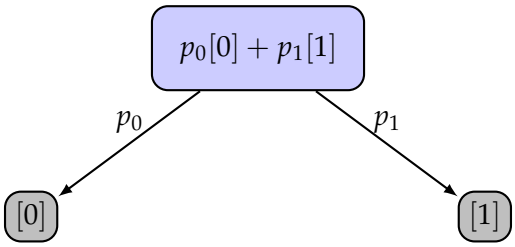
mert most már biztosan tudod, hogy a fej van felül. Azt a folyamatot, amikor egy véletlen érmét felfedünk, hogy meghatározzuk, melyik oldal van felül, mérésnek nevezzük.[2]

Figyeld meg, hogy az 1.15. és 1.16. egyenletekben az érme állapota a mérés előtt és után különbözik. Valóban, a mérés után már nincs kétség afelől, hogy melyik oldal van felül. Most képzeld el, hogy újra letakarod az érmét, miután megmérted. Mi az állapota most? Természetesen a letakarástól nem változik meg.

$$= [0], \tag{1.17}$$

mert már tudod, hogy a fej van felül. Valójában még ha újra meg is méred (megnézed) az érmét, akkor is "fejet" fogsz látni. Hasonlóképpen, ha először "írást" kaptál egy véletlenszerű érme mérésekor, akkor az "írást" fogod kapni, bármennyiszer is méred újra.

Általánosabban, ha van egy valószínűségi bited, amelyet a $\binom{p_0}{p_1}$ eloszlás ír le, a mérés kimenetele p_0 valószínűséggel 0 (vagy "fej") és p_1 valószínűséggel 1 (vagy "írás"):



(1.18)

A valószínűségi bit állapota a mérés *után* már nem $\binom{p_0}{p_1}$, hanem az egyik bázisállapot, $[0] = \binom{1}{0}$ vagy $[1] = \binom{0}{1}$, a mérési eredménytől függően. Például, ha az eredmény 1 (vagy "írás"), az új állapot $[1]$. Általában a mérés *megváltoztatja* az állapotot!

A mérésekkel kapcsolatban fontos megjegyezni, hogy *nem engedik meg*, hogy kinyerd a p_0 és p_1 valószínűségeket – az egyetlen, amit mérési eredményként kapsz, egyetlen bit 0 vagy 1. Továbbá, az eredeti valószínűségi bited $\binom{p_0}{p_1}$ elveszik a mérés után, így nem mérheted meg újra. Ez valójában nagyon természetes. Ha egy érmét pontosan egyszer dobunk fel, akkor egyetlen véletlenszerű eredményt kapunk – de ebből az egyetlen eredményből nem tudhatjuk meg, hogy az érme szabályos vagy torzított volt-e.

Azonban tegyük fel, hogy ugyanazt az érmét sokszor feldobjuk. Ebben az esetben arra számítanánk, hogy az 1 eredmény előfordulásainak aránya nagyjából p_1 lesz. Más szóval,

$$\frac{N_1}{N} \approx p_1, \tag{1.19}$$

[2] Ezt a kifejezést a kvantumszámításból kölcsönöztük, ahol hasonló eljárás létezik.

ahol N a mérések összes száma és N_1 az 1 eredmény előfordulásainak száma. Minél több eredményt gyűjtünk össze, annál jobb lesz a közelítés.[3]

Ez egy eljárást ad számunkra p_1 becslésére. Természetesen, mivel $p_0 + p_1 = 1$, ez p_0 becslését is megadja. Például Alíz ezt az eljárást használhatja, hogy megbecsülje annak valószínűségét, hogy az 1.3. házi feladatbeli torzított érméje (S) vagy (F) oldalát mutatja. Most te is kipróbálhatod ezt magad.

1.4. Házi feladat: Érmefeldobás

1. Keress egy érmét, és rajzolj a két oldalára egy 0 és egy 1 jelet filctollal. Dobd fel az érmét 30 alkalommal, és írd le az eredményeket egy ilyen táblázatba:

Dobások száma: N	1	2	3	4	5	6	7	8	9	…	30
Az N-edik eredmény	1	0	1	0	0	0	1	1	1	…	1

   (A szürke eredmények csak illusztrációk. Helyettesítsd őket a saját eredményeiddel.)

2. Becsüld meg mekkora valószínűséggel ad 1-et az érmed az 1.19. egyenlet segítségével.

3. Érdekes látni, hogyan változik a becslés, ahogy növeled a dobások számát N. Ehhez bővítsd ki az 1. rész táblázatát három további sorral, hogy így nézzen ki:

Dobások száma: N	1	2	3	4	5	6	7	8	…	30
Az N-edik eredmény	1	0	1	0	0	0	1	1	…	1
Összeg N_1	1	1	2	2	2	2	3	4	…	16
Arány: N_1/N	1	1/2	2/3	2/4	2/5	2/6	3/7	4/8	…	16/30
Numerikus értéke	1,00	0,50	0,67	0,50	0,40	0,33	0,43	0,50	…	0,53

   A sorok jelentése a következő: (1) eddigi dobások száma N, (2) a N-edik dobás eredménye, (3) az első N eredmény összege, (4) az 1 eredmény valószínűségének becslése az első N dobás alapján, (5) a becslés numerikus értéke. Nyugodtan használd az *Excel* vagy egy hasonló programot ennek a táblázatnak az elkészítéséhez.

4. Ábrázold a táblázat utolsó sorát a dobások számának N függvényében.

1.4. A QUIRKY szimulátor

A valószínűség törvényei néha ellentmondásosnak tűnhetnek. Szerencsére mindig megpróbálhatod szimulálni a viselkedésüket az otthon (vagy a zsebedben) lévő számítógép segítségével.[4]

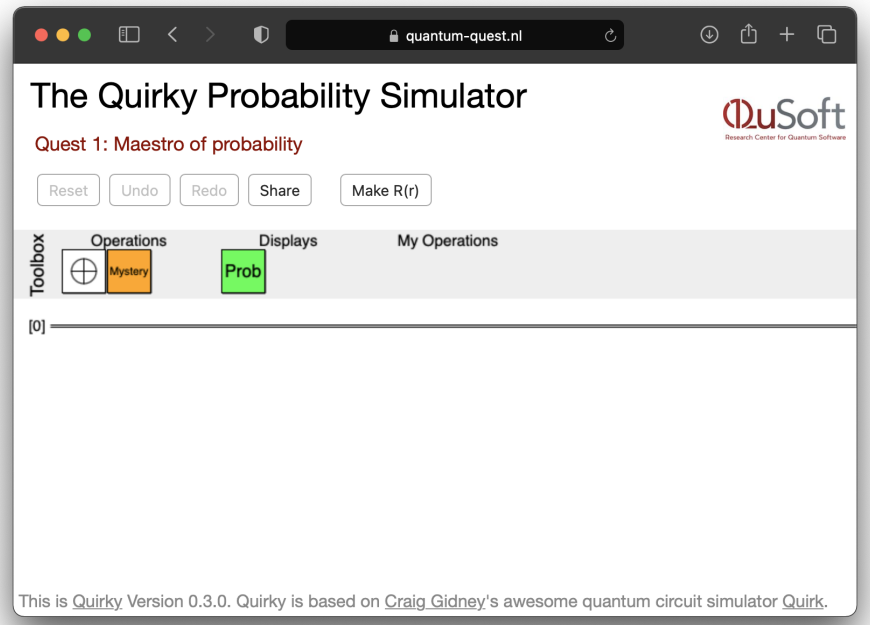
Ebben a kurzusban a QUIRKY nevű szimulátort fogjuk használni. A QUIRKY a Quirk kisebb testvére, egy több funkcióval rendelkező szimulátor, amelyet Craig Gidney fejlesztett a Google-nél. Mivel Craig a kódját nyílt forráskódú licenc alatt tette közzé, a szimulátort a kurzus igényeihez tudtuk igazítani. A QUIRKY egyik legjobb tulajdonsága, hogy közvetlenül a webböngésződben fut – nincs szükség telepítésre! Egyszerűen menj a következő oldalra:

https://www.quantum-quest.org/quirky

és kattints a "Quest 1" gombra. Próbáld ki most – a QUIRKY még a mobiltelefonodon is megnyitható! Amikor először megnyitod a QUIRKY-t, annak az 1.3. ábrán látható módon kell kinéznie.

[3] Milyen jó ez a becslés? Megmutatható, hogy az átlagos hiba nagysága $1/\sqrt{N}$ nagyságrendű, így gyorsan nullára csökken, ha sokszor megismételjük a kísérletet.

[4] Bizonyos mértékig ez még a kvantumszámítógépekre is igaz – de ne szaladjunk ennyire előre…



1.3. ábra.
A QUIRKY első indításkor.

1.4.1. A szimulátor használata

Nézzük át lépésről lépésre a QUIRKY felületét. A tetején találsz egy *menüsor* néhány hasznos paranccsal:

A 'Reset' gomb lehetővé teszi a QUIRKY visszaállítását és az újrakezdést. Az 'Undo' és 'Redo' gombok a parancsok visszavonására és előreléptetésére használhatók. Hasznos tudni, hogy még a szimulátor visszaállítását is vissza lehet vonni. A Share' gomb megnyomásával lehetőség van a munkameneted megosztására másokkal. A 'Make $R(r)$' gombbal később foglalkozunk.

A menü alatt található a *"Toolbox"*, amely az eddig tanult alapműveleteket tartalmazza:

Például az első doboz, $\boxed{\oplus}$, a NOT művelet az 1.2. alfejezetből, amely a $[0]$ állapotot $[1]$-re, és fordítva változtatja. A másik két műveletet hamarosan megvitatjuk. Szerencsére nem kell mindezt megjegyezned – egyszerűen vidd az egeret minden doboz fölé, hogy lásd annak leírását.

A „Toolbox" alatt található a QUIRKY szíve, a *valószínűségi bit*:

A dupla vonal vagy 'vezeték' egy bitnek felel meg, amely az $[0]$ állapotban van inicializálva. Egyszerűen helyezhetsz el műveleteket úgy, hogy az eszköztárból ráhúzod őket a vezetékre. Próbálj most összeállítani a következő egyszerű számítást a QUIRKY-ban:[5]

[5]Ha a jegyzetek digitális verzióját olvasod, egyszerűen kattints bármelyik képre, hogy megnyisd a QUIRKY-t a böngésződben. Minden, amit a képen látsz, automatikusan bekerül! Ha ez nem működik, menj el a https://www.quantum-quest.org/quirky oldalra.

Hogyan tudjuk megjeleníteni egy ilyen számítás eredményét? Mivel általában valószínűségekkel dolgozunk, valójában egy módszert keresünk a *valószínűségek megjelenítésére*. Ezt a zöld dobozzal, amelyen a Prob felirat található az eszköztár Display' részében, érhetjük el. Adjuk hozzá ezt a számításunkhoz, és nézzük meg, mi történik:

Úgy tűnik, hogy a mérési eredmény 100% valószínűséggel 'egy' lesz (vidd az egeret a doboz fölé, hogy megerősítsd a gyanúnkat). Természetesen pontosan ezt várjuk. Az inicializált $[0]$ állapotot a NOT művelet $[1]$ állapottá alakítja, így az eredmény mindig 'egy' lesz az 1.18. egyenlet mérési szabálya szerint.

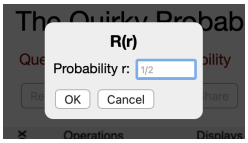
1.8. Gyakorló feladat: Egy művelet eltávolítása

A QUIRKY-ben műveleteket is eltávolíthatsz, egyszerűen úgy, hogy visszahúzod őket a vezetékből az eszköztárba. Távolítsd el a NOT műveletet a számításból, és erősítsd meg, hogy az eredmény most már biztosan nulla' lesz.

1.4.2. Saját műveletek készítése

Eddig csak a $[0]$ és $[1]$ állapotokat tudtuk létrehozni a QUIRKY-ban. Ahhoz, hogy érdekes valószínűségi eloszlást hozzunk létre, használhatjuk az 1.2.2. alfejezetben leírt lenullázó műveletet $R(r)$. Mivel végtelen sok ilyen művelet létezik (egy-egy minden r választására), nem tudtuk mindet hozzáadni az eszköztárhoz. Ehelyett saját lenullázó (reset) műveleteket adhatsz hozzá az eszköztárhoz!

Gyakoroljuk ezt úgy, hogy hozzáadunk egy műveletet, amely $r = \frac{1}{2} = 50\%$-os valószínűséggel nulláz le. Először válaszd a 'Make $R(r)$' opciót a menüsorban. Egy új ablak jelenik meg, ahol beírhatod a szöget:



Írd be a `1/2` értéket, és erősítsd meg a gomb megnyomásával. Gratulálok! Sikeresen hozzáadtad az $R(1/2)$ műveletet az eszköztárhoz, amely most így néz ki:

Az új művelet teszteléséhez építsük fel a következő számítást a QUIRKY-ban:

Gyorsan nézzük meg, hogy ez az eredmény értelmes-e. A $[0] = \binom{1}{0}$ állapotból indultunk. A NOT művelet átbillenti a bitet a $[1] = \binom{0}{1}$ állapotba. Az 1.13. egyenlet szerint az $R(1/2)$ művelet egy bitet az $[1]$ állapotban 50%-os valószínűséggel nulláz le, azaz az állapotot a következőre változtatja:

$$R(1/2)\,[1] = \frac{1}{2}\,[0] + \frac{1}{2}\,[1] = \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix} = \begin{pmatrix} 50\% \\ 50\% \end{pmatrix}.$$

Ez pontosan az, amit a QUIRKY mondott nekünk.

A következő feladatban a QUIRKY-t fogod használni egy bonyolultabb kísérlet végrehajtásához.

1.5. Házi feladat: Kétszeri véletlen lenullázás

1. Építsd fel a következő műveletsort a QUIRKY segítségével: Először készítsd elő az $[1]$ állapotot, majd nulláz le $\frac{1}{4}$ valószínűséggel, végül ismét nullázd le $\frac{2}{3}$ valószínűséggel. Használd a QUIRKY-ban a valószínűség megjelenítőt, hogy meghatározd a mérési eredmények valószínűségét.

2. Indokold meg, hogy a QUIRKY által adott válasz miért helyes.

1.4.3. Egy rejtélyes művelet

Még mindig nem beszéltünk a rejtélyes (angolul „mysterious") narancssárga dobozról. Nevezzük ezt a műveletet M-nek, mivel valóban elég rejtélyes. Hogyan deríthetjük ki, mi történik a dobozban? Első lépésként vizsgáljuk meg az $M\,[0]$ problémáját, azaz, hogy mi lesz az eredménye annak, ha a rejtélyes M műveletet egy $[0]$ állapotú bitre alkalmazzuk. A QUIRKY-ban ez a következő beállításnak felel meg:

Hogyan olvashatjuk le az $M\,[0]$ eredményét? Ezen a ponton jó emlékeztetni magunkat arra, hogy amikor véletlen bitek jelennek meg a természetben, *nem* tudjuk egyszerűen megtekinteni őket és leolvasni a valószínűségeiket. Ehelyett, ahogyan az 1.3. alfejezetben is magyaráztuk, sok mérést kell végeznünk (például sokszor feldobni egy érmét) és a kimenetelekből becsülni a valószínűségeket. A QUIRKY-hoz hasonló szimulátor használatának előnye, hogy nem kell ezekkel a szabályokkal játszanunk – használhatjuk a valószínűség kijelzőt az állapot meghatározásához:

Így megtudjuk, hogy

$$M\,[0] = 0,2\,[0] + 0,8\,[1].$$

Most rajtad a sor!

1.6. Házi feladat: A rejtély felderítése

1. Határozd meg az $M\,[1]$ állapotot.

2. Teljes mértékben meghatározza-e az $M\,[0]$ és $M\,[1]$ a véletlen műveletet M?
   Ha igen, írj le egy képletet az $M\left(\begin{smallmatrix}1/2\\1/2\end{smallmatrix}\right)$-re és ellenőrizd a QUIRKY-ban. Ha nem, magyarázd el, miért.

A következő hetekben meg fogjuk tenni az ugrást a hétköznapi bitekről a kvantumbitekre, és megtanuljuk, hogyan számoljunk velük egyre kifinomultabb módokon. A QUIRKY lesz a megbízható eszközünk, amely új képességekkel bővül, ahogy haladunk előre. Bátorítunk, hogy használd a QUIRKY-t az elmélet tanulmányozására, amit megtanulsz, valamint hogy segítsen megoldani a házi feladataidat.

1.5. A gyakorló feladatok megoldásai

1.1. Gyakorló feladat megoldása

1. Mivel a két esemény egyike biztosan bekövetkezik, a két valószínűség összege szükségszerűen 1. Ez azt jelenti, hogy $p_0 + p_1 = 1$. Ha ezt úgy írod fel, hogy $p_1 = 1 - p_0$, akkor felismered, hogy ez egy olyan egyenes egyenlete, amelynek meredeksége mínusz egy.

2. Ha az egyenes tovább menne, az egyik valószínűség negatív lenne. Mivel a valószínűségek nem lehetnek negatívak, meg kell követelnünk, hogy $p_0 \geq 0$ és $p_1 \geq 0$, ami egyenértékű azzal, hogy a szakasznak a koordináta-tengelyeken kell végződnie.

3. Ez a felezőpont, ahol $p_0 = p_1 = \frac{1}{2}$.

1.2. Gyakorló feladat megoldása

1. A számláló 60 különböző értéket vehet fel. Az esélye annak, hogy bármelyik értéket látjuk, $\frac{1}{60}$.

2. Az utolsó számjegy 10 különböző értéket vehet fel. Az esélye annak, hogy bármelyik értéket látjuk, $\frac{1}{10}$.

3. Az első számjegy 6 különböző értéket vehet fel. Az esélye annak, hogy bármelyik értéket látjuk, $\frac{1}{6}$.

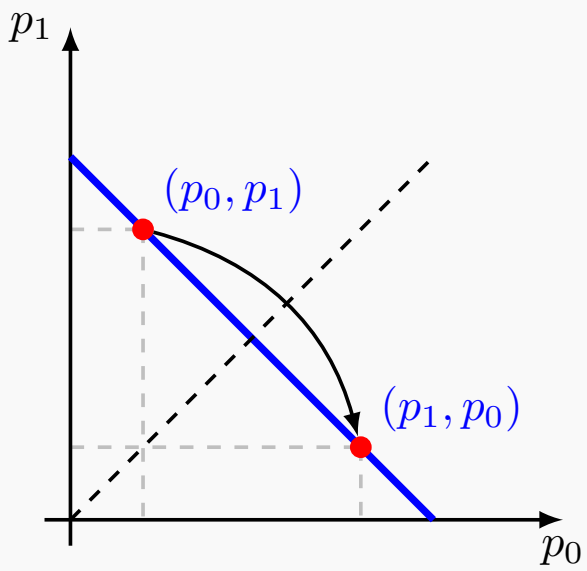
4. Ha csak az első számjegyet látjuk, az utolsó számjegy egyenlő valószínűséggel lehet bármelyik a 10 lehetséges érték közül. Hasonlóképpen, ha csak az utolsó számjegyet látjuk, az első számjegy egyenlő valószínűséggel lehet bármelyik a 6 lehetséges érték közül. Ezért a két számjegy értékei függetlenek. Ellenőrizheted, hogy az esélye annak, hogy 00 valóban $\frac{1}{60}$, ha megszorzod annak valószínűségét, hogy mindkét számjegy nulla:
$$\frac{1}{6} \cdot \frac{1}{10} = \frac{1}{60}.$$

1.3. Gyakorló feladat megoldása

Hat olyan eset van, amikor mindkét számjegy ugyanaz (00-tól 55-ig). Ezek mindegyike $\frac{1}{60}$ valószínűséggel fordul elő. Ezeket az eseteket egyetlen eseménybe csoportosíthatjuk, amelynek valószínűsége a hat egyedi esemény valószínűségének összege:
$$\underbrace{\frac{1}{60} + \frac{1}{60} + \frac{1}{60} + \frac{1}{60} + \frac{1}{60} + \frac{1}{60}}_{\text{6 tag}} = \frac{6}{60} = \frac{1}{10}.$$

1.4. Gyakorló feladat megoldása

1. Az 1.10. egyenlet alapján látszik, hogy a (p_0, p_1) pont koordinátái felcserélődnek, azaz (p_1, p_0)-ra változnak. Itt egy példa arra, hogy ez hogyan néz ki:



   Így a NOT műveletet úgy képzelheted el, mint egy tükrözést a szaggatott vonal körül, amely pontosan a két koordináta-tengely között helyezkedik el.

2. Az 1.2. egyenlet szerint a szakasz két végpontja $(1,0)$ és $(0,1)$ megfelel a $[0]$ és $[1]$ determinisztikus állapotnak. Emlékezz rá, hogy az 1.9. egyenletben láttuk, hogy a NOT művelet ezeket felcseréli.

3. Egy (p_0, p_1) koordinátájú pont helyben marad a NOT művelet után, ha $(p_0, p_1) = (p_1, p_0)$, ami azt jelenti, hogy $p_0 = p_1$. Mivel $p_0 + p_1 = 1$, azt találjuk, hogy $p_0 = p_1 = 1/2$, ami megfelel a $(1/2, 1/2)$ pontnak. Ez az egyetlen pont, amely helyben marad.

1.5. Gyakorló feladat megoldása

Az 1.11. és 1.9. egyenletek használatával azt kapjuk, hogy

$$\text{NOT}\begin{pmatrix} p_0 \\ p_1 \end{pmatrix} = p_0\,\text{NOT}\begin{pmatrix} 1 \\ 0 \end{pmatrix} + p_1\,\text{NOT}\begin{pmatrix} 0 \\ 1 \end{pmatrix} = p_0\begin{pmatrix} 0 \\ 1 \end{pmatrix} + p_1\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} p_1 \\ p_0 \end{pmatrix},$$

ami megegyezik az 1.10. egyenlettel.

1.6. Gyakorló feladat megoldása

1. Az 1.14. egyenletbeli definíció szerint

$$F(f)\,[0] = (1-f)\,\left(\begin{smallmatrix}1\\0\end{smallmatrix}\right) + f\,\left(\begin{smallmatrix}0\\1\end{smallmatrix}\right) = \left(\begin{smallmatrix}1-f\\f\end{smallmatrix}\right),$$
$$F(f)\,[1] = f\,\left(\begin{smallmatrix}1\\0\end{smallmatrix}\right) + (1-f)\,\left(\begin{smallmatrix}0\\1\end{smallmatrix}\right) = \left(\begin{smallmatrix}f\\1-f\end{smallmatrix}\right).$$

2. $F(f)$ biztosan átbillenti a bitet, ha $f = 1$, tehát NOT $= F(1)$. Egy tetszőleges $\left(\begin{smallmatrix}p\\1-p\end{smallmatrix}\right)$ állapot előállításához $[0]$-ból $f = 1-p$ értéket kell választanunk. Valóban, az első fenti egyenletből látható, hogy

$$F(1-p)\,[0] = \begin{pmatrix}1-(1-p)\\1-p\end{pmatrix} = \begin{pmatrix}p\\1-p\end{pmatrix}.$$

3. Az 1.11. egyenlet szerint,

$$F(f)\,\begin{pmatrix}p_0\\p_1\end{pmatrix} = p_0\begin{pmatrix}1-f\\f\end{pmatrix} + p_1\begin{pmatrix}f\\1-f\end{pmatrix} = \begin{pmatrix}p_0(1-f)+p_1 f\\p_0 f + p_1(1-f)\end{pmatrix}.$$

4. Ha $f = 1/2$ értéket behelyettesítjük az előző egyenletbe, akkor kapjuk, hogy

$$F(1/2)\,\begin{pmatrix}p_0\\p_1\end{pmatrix} = \begin{pmatrix}p_0/2+p_1/2\\p_0/2+p_1/2\end{pmatrix} = \frac{1}{2}\begin{pmatrix}p_0+p_1\\p_0+p_1\end{pmatrix} = \frac{1}{2}\begin{pmatrix}1\\1\end{pmatrix}.$$

1.7. Gyakorló feladat megoldása

Két esetet különböztetünk meg:

- $\frac{1}{2} \le f \le 1$: Ebben az esetben $0 \le \frac{1-f}{f} \le 1$. Azt állítjuk, hogy az átbillentés művelet $F(f)$ felépíthető úgy, hogy először alkalmazzuk az $R(\frac{1-f}{f})$ műveletet, majd a NOT-ot, és végül az $R(1-f)$ műveletet. Valóban:

$$R(1-f)\,\mathrm{NOT}\,R(\tfrac{1-f}{f})\,[0] = R(1-f)\,\mathrm{NOT}\,[0] = R(1-f)\,[1] = (1-f)\,[0] + f\,[1]$$

és

$$\begin{aligned}R(1-f)\,\mathrm{NOT}\,R(\tfrac{1-f}{f})\,[1] &= R(1-f)\,\mathrm{NOT}\left(\tfrac{1-f}{f}\,[0] + (1-\tfrac{1-f}{f})\,[1]\right)\\ &= R(1-f)\left((1-\tfrac{1-f}{f})\,[0] + \tfrac{1-f}{f}\,[1]\right)\\ &= (1-\tfrac{1-f}{f})\,[0] + \tfrac{1-f}{f}\,((1-f)\,[0] + f\,[1])\\ &= f\,[0] + (1-f)\,[1].\end{aligned}$$

- $0 \le f \le \frac{1}{2}$: Ez az eset az elsőre redukálható, mivel $F(f)$ ugyanaz, mint először alkalmazni az $F(1-f)$-et, majd a NOT-ot.

2. Küldetés: A qubitek felfedezése

Most, hogy már mestere vagy a valószínűségeknek és a valőszínűségi biteknek, készen állsz arra, hogy megtanuld a kvantumbiteket. A kvantumbitek nagyon hasonlítanak a valőszínűségi bitekhez – valójában csak annyit kell tenned, hogy a valószínűségeket kvantum amplitúdókra cseréled. Ezen a héten megismerkedhetsz egy kvantumbit állapotaival, a rajta megengedett műveletekkel, és azzal, hogyan nyerhetsz ki információt belőle. Emellett kipróbálhatod a QUIRKY *kvantum* verzióját is.

2.1. Kvantumbitek

Napjainkban a bitek az információ alapvető egységei a számítógépekben. Egy bit megvalósításához szükség van egy fizikai dologra, amely két jól megkülönböztethető állapotban lehet, mint például egy kétoldalú érme vagy egy kondenzátor, amely két különböző feszültségszinten tud elektromos töltést tárolni.[6] Az ilyen dolgok viselkedését (és így az általuk kódolt biteket is) fizikai elméletek, mint például a mechanika (érmék esetén) vagy az elektromágnesesség (kondenzátorok esetén) írhatják le.

Azonban, ha igazán apró[7] tárgyakról van szó, ezek az elméletek már nem érvényesek, és egy alapvetőbb elméletet kell használni, amit kvantummechanikának nevezünk. Például, az elektronoknak van egy spin nevű tulajdonságuk, amely (egy érméhez hasonlóan) lehet kétféle állapotban – felfelé vagy lefelé – és így egy elektron egy bit tárolására használható. Azonban, ellentétben egy érmével, az elektron spinje nem csak ezek egyikében lehet, hanem a kettő „szuperpozíciójában" is! Intuitívan, ez némileg hasonlít egy valőszínűségi bitre, amely szintén egy köztes állapotban lehet 0 és 1 között.

Azonban van egy finom különbség a valószínűségek és a „szuperpozíciók" között (lásd az interferenciáról szóló 2.6.1. alfejezetben). Amint látni fogjuk, a kvantummechanika törvényei egy sokkal alapvetőbb információfogalomhoz vezetnek, mint egy bit – egy kvantumbithez vagy qubithoz. A szokásos biteket, megkülönböztetve egzotikusabb kvantum barátaiktól, klasszikus biteknek nevezzük.

A kvantumbiteket egy egyszerű matematikai modellel fogjuk leírni, és nem törődunk azzal, hogyan kellene értelmezni furcsa viselkedésüket, hanem inkább azon gondolkodunk: „Mire lehet őket használni?". Hasonlóképpen, nem foglalkozunk azzal sem, hogy fizikailag hogyan lehetne őket megvalósítani, vagy milyen dolgok használhatók a tárolásukhoz. Azonban, ha kíváncsi vagy erre, röviden megvitatjuk a 2.6.2. alfejezetben, hogyan lehet a fény polarizációját használni egy qubit leírására.

2.1.1. Valószínűségek és amplitúdók

A kvantumbitek nagyon hasonlóak a valőszínűségi bitekhez. Csak két jelentős különbség van:

1. a valószínűségeket amplitúdókra cseréljük (amelyek lehetnek negatívak is),
2. az amplitúdókat a mérés során négyzetre emeljük (míg a valószínűségeket nem).

Rövidesen részletesebben is elmagyarázzuk ezeket a különbségeket, de először nézzük meg egy kvantumbit lehetséges állapotait. Emlékszel, hogyan használtuk az érme két oldalát, hogy megjelöljük a valőszínűségi bit két lehetséges determinisztikus állapotát (lásd 1.1. ábra)? A kvantumszámításban ezt a két állapotot általában $|0\rangle$ és $|1\rangle$ jelöli, hogy megkülönböztessük őket

[6] Lényegében így vannak a bitek eltárolva a számítógépedben, mobiltelefonodban stb.

[7] Az „igazán apró" alatt *nagyon, nagyon aprót* értünk! Ha elektronokat tennél egymás mellé egy sorba, az a mennyiség, amivel elérnéd az 1 cm hosszúságot, hasonló lenne ahhoz a lapmennyiséghez, amit egymásra rakva elérnél a Holdig.

a klasszikus [0] és [1] bitektől. Csakúgy, mint a valószínűségi bitek esetében, egy általános $|\psi\rangle$ kvantumbit állapot egy lineáris kombinációja vagy szuperpozíciója ezen két determinisztikus állapotnak:

$$|\psi\rangle = \psi_0 |0\rangle + \psi_1 |1\rangle . \tag{2.1}$$

Itt a görög betű ψ (ejtsd: „pszí") a kvantumbit állapotának neve (csakúgy, mint ahogy egy valószínűségi bitet p-nek neveztünk). A zárójelek $|\cdot\rangle$ egy úgynevezett „ket" formát alkotnak (az angol *bracket* szóból ered, amely „zárójelet" jelent), amely azt jelzi, hogy egy kvantumállapotról van szó. Összehasonlításként emlékezz vissza az 1.3. egyenletre, ahol egy tetszőleges véletlen p bitet úgy írtunk fel mint

$$p = p_0[0] + p_1[1]. \tag{2.2}$$

Figyeld meg, hogy a 2.1. egyenlet ugyanilyen, kivéve, hogy a p_0 és p_1 valószínűségeket ψ_0 és ψ_1 amplitúdókra cseréltük, és a [0] és [1] klasszikus jelölést $|0\rangle$ és $|1\rangle$ kvantum jelölésre! Azonban van egy nagy különbség: míg a valószínűségek a 2.2. egyenlet szerint így alakulnak:

$$p_0, p_1 \geq 0, \qquad p_0 + p_1 = 1, \tag{2.3}$$

az amplitúdókra ez vonatkozik:

$$\psi_0^2 + \psi_1^2 = 1. \tag{2.4}$$

Ezért aztán $\psi_0^2 \leq 1$ és $\psi_1^2 \leq 1$, és így $\psi_0, \psi_1 \in [-1, 1]$. Ezzel szemben, a valószínűségekre vonatkozó feltételek a 2.3. egyenletben azt jelentik, hogy $p_0, p_1 \in [0, 1]$. A döntő különbség az, hogy az amplitúdók valóban lehetnek negatívak, míg a valószínűségek nem![8]

Csakúgy, mint a valószínűségi biteket, kényelmes a qubit állapotokat is vektorokkal leírni. Teljesen analóg módon az 1.2. egyenlettel, a $|0\rangle$ és $|1\rangle$ determinisztikus qubit állapotokat a két bázisvektorral ábrázoljuk:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

Egy általános $|\psi\rangle$ kvantumállapotot a 2.1. egyenletnek megfelelően így ábrázolunk:

$$|\psi\rangle = \psi_0 \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \psi_1 \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \psi_0 \\ \psi_1 \end{pmatrix}.$$

2.1.2. A qubitek a körvonalon élnek

Figyeld meg, hogy a qubit amplitúdók 2.4. egyenlete mennyire hasonlít az egységkör $x^2 + y^2 = 1$ egyenletére. Fejtsük ki ezt a megfeleltetést részletesebben, mert ez segíteni fog nekünk a kvantumbitek vizualizálásában és intuitívabb megértésében.

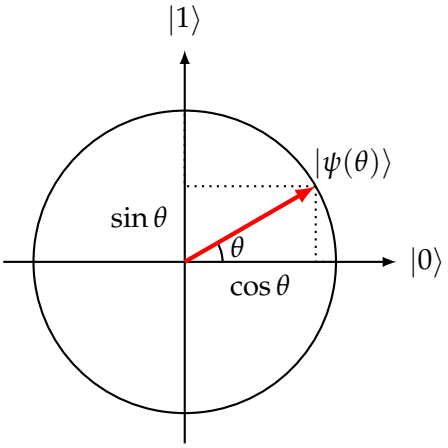
A qubit amplitúdókat kényelmesen paraméterezhetjük a

$$\psi_0 = \cos\theta, \qquad \psi_1 = \sin\theta$$

egyenletekkel, valamely $\theta \in [0, 2\pi)$ szögre. Valójában gyakran hasznos lesz megengedni, hogy a θ tetszőleges valós szám legyen (ami rendben van, feltéve hogy szem előtt tartjuk, hogy bármely két szög, amely 2π egész számú többszörösével különbözik, ugyanazt az amplitúdót eredményezi). Mivel $\cos^2\theta + \sin^2\theta = 1$, garantáltan teljesítjük a 2.4. egyenletet. Ezzel a választással egy általános qubit állapot az alábbi módon néz ki:

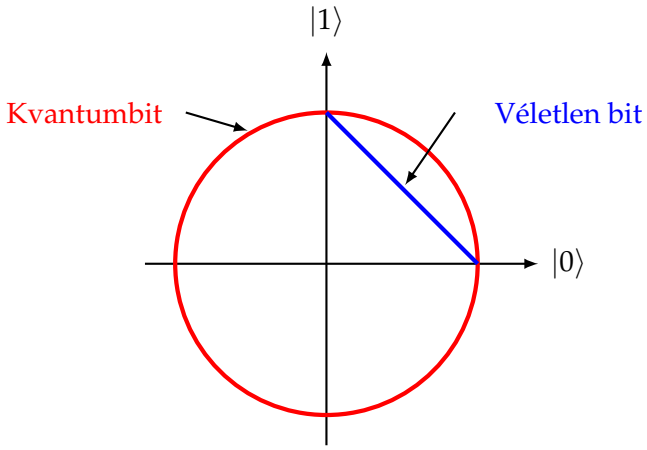
$$|\psi(\theta)\rangle = \cos\theta |0\rangle + \sin\theta |1\rangle = \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}. \tag{2.5}$$

[8] Valójában az amplitúdók lehetnek úgynevezett *komplex számok* is. Erre ebben a kurzusban nem lesz szükségünk, de bátran böngészd az internetet, ha többet szeretnél megtudni erről.



2.1. ábra. A $|\psi(\theta)\rangle$ qubit állapota, mint egy pont az egységkörön.

Ezt úgy lehet elképzelni, mint egy síkbeli egységvektort, amely az origóból indul és θ szöget zár be a vízszintes $|0\rangle$ tengellyel (lásd 2.1. ábra). Speciálisan, $|0\rangle = |\psi(0)\rangle$ és $|1\rangle = |\psi(\frac{\pi}{2})\rangle$. A qubit állapotok halmaza az origó középpontú egységkörnek felel meg. Ezzel szemben, emlékezzünk, hogy a valószínűségi bit összes állapota egy szakaszt alkot, amely összeköti a két koordinátatengelyen található $\binom{1}{0}$ és $\binom{0}{1}$ pontokat, ahogy azt az 1.2. ábrán láttuk. A két halmaz összehasonlítását a 2.2. ábrán láthatod.



2.2. ábra. Egy valószínűségi bit (kék) és egy kvantumbit (piros) állapottere.

2.1. Gyakorló feladat: Állapotok a körön

Tekintsük egy qubit következő két állapotát:

$$|+\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}, \qquad |-\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}.$$

Hol helyezkedik el ez a két állapot a körön? Milyen θ szögeknek felelnek meg?

2.2. Egy kvantumbit mérése

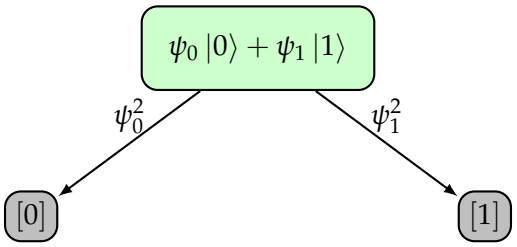
Most már tudjuk, hogy bármely qubit állapot előáll $|\psi(\theta)\rangle$ alakban. Tegyük fel, hogy van egy $|\psi(\theta)\rangle$ állapotú qubited és szeretnéd megtudni θ értékét. Sajnos a kvantummechanika miatt erre nincs lehetőség! Ez nagy problémának tűnik – mire jó egy kvantumszámítógép, ha nem

tudod kinyerni belőle a választ? Nos, lassítsunk egy kicsit! Emlékezz vissza az 1.18. egyenletre, hogy ugyanez igaz a valószínűségi bitekre is – ha egy valószínűségi bitet mérsz, amelynek eloszlása $\binom{p_0}{p_1}$, nem tudod meg p_0 vagy p_1 értékét. Csak egyetlen bitnyi információt kapsz: ez p_0 valószínűséggel 0 és p_1 valószínűséggel 1.

A kvantummérés nagyon hasonló, és az úgynevezett *Born szabály* írja le. Ha van egy qubit valamely $\binom{\psi_0}{\psi_1} = \psi_0 |0\rangle + \psi_1 |1\rangle$ állapotban és megméred, akkor is csak egyetlen bitet kapsz, 0-t vagy 1-et az alábbi valószínűségekkel:

$$p_0 = \psi_0^2, \qquad p_1 = \psi_1^2. \tag{2.6}$$

Bár a négyzetre emelés talán meglepő, vegyük észre, hogy $p_0 + p_1 = \psi_0^2 + \psi_1^2 = 1$. Így a négyzet éppen azt garantálja, hogy $\binom{p_0}{p_1}$ egy tényleges valószínűségi eloszlás, tehát a fenti szabály logikus! A mérés után a qubit eltűnik, és csak egyetlen bit marad, amely a mért eredményt tartalmazza. Más szavakkal, a mérési folyamat egy qubitet egy hagyományos bitre alakít át, amelynek értékét a 2.6. egyenlet szerinti valószínűségek határozzák meg:



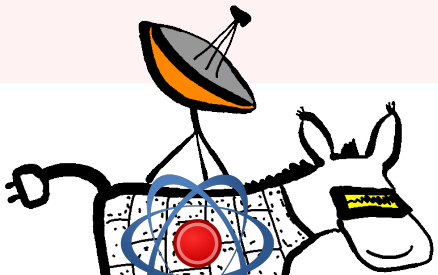
(2.7)

Ahogy az 1.18. és 2.7. egyenletekből látható, a valószínűségi bitek és qubitek mérési szabályai nagyon hasonlóak. Mindkét esetben az eredeti állapot eltűnik, és csak egyetlen bit marad, amelynek véletlenszerű értéke a fenti szabály szerint függ az eredeti állapottól, amelyet megmértél. (Ráadásul, ha többször méred az állapotot, mindig ugyanazt az eredményt kapod, mint először – tehát az ismételt mérések nem adnak további információt az eredeti állapotról.) Az egyetlen különbség az, hogy a qubiteknél az amplitúdókat négyzetre kell emelni a valószínűségek kiszámításához, ahogy az a 2.6. egyenletben szerepel, míg a valószínűségi biteknél közvetlenül kapjuk meg őket, így nincs szükség négyzetre emelésre. Bár ez kis különbségnek tűnhet, jelentős hatással van a megengedett állapotokra, mivel a qubit amplitúdók lehetnek negatívak, míg a valószínűségi bit valószínűségei mindig nemnegatívak (lásd 2.2. ábra).

Nos, valójában van egy másik, még finomabb különbség. Nevezetesen, hogy senki sem tudja előre megjósolni egy kvantummérés kimenetelét. Ez azért finom, mert úgy tűnik, hogy ugyanez igaz a valószínűségi bitekre is. Mi a különbség? Röviden, a válasz az, hogy a valószínűségi bitek véletlenszerűnek tűnnek a róluk való ismeretünk hiánya miatt, míg a kvantumbitek véletlenszerűek akkor is, ha mindent tudunk az állapotukról. Például képzeld el, hogy a barátod feldob egy szabályos érmét, és azonnal letakarja, amint landol. Alapesetben egy ilyen érme állapotát egyenletesen véletlenszerűnek írnád le, ld. 1.15. egyenlet. Azonban, ha nagysebességű kamerával filmeznéd az érmét, lehet, hogy pontosan meg tudnád jósolni, melyik oldalára esett a felvételek alapján. Ebben az értelemben a valószínűségi bitek véletlenszerűsége a tudatlanságunkhoz kapcsolódik. A kvantumbitek esetében azonban a véletlenszerűség alapvető szinten jelenik meg. Bármilyen előzetes ismeretünk is lenne, általánosságban *lehetetlen* tökéletesen megjósolni egy kvantummérés kimenetelét. Másfelől ez azt jelenti, hogy a kvantummérések eredményei jó forrásai lehetnek a véletlenszerűségnek!

2.1. Házi feladat: Véletlen bit generálása kvantumosan

Probléma: Alíz robotszamara megint lemerülőben van, és meg kell találnia az útját egy töltőállomáshoz. Sajnos, ezúttal Éva hackelő ké-



pességei javultak – rájött, hogyan hackelje meg a szamár véletlenszám-generátorát, és újraprogramozza úgy, hogy bármilyen számsort generáljon, amit csak akar! Szerencsére Alíz tud erről, mivel Éva nemrégiben dicsekedett vele egy hacker fórumon. Hogy ellensúlyozza Éva gonosz tervét, Alíz úgy döntött, hogy egy miniatűr, 1 qubites kvantumszámítógépet telepít a robotszamarába. A kvantummérés eredményeinek alapvető kiszámíthatatlanságát kihasználva Alíz egyenletesen véletlen biteket szeretne generálni, amelyeket Éva nem tud kitalálni.

Kérdés: Alíz képes előállítani bármilyen $|\psi(\theta)\rangle$ qubit állapotot, és egyenletesen véletlen bitet akar generálni annak mérésével.

1. Amikor a $|\psi(\theta)\rangle$ állapotot méred, milyen valószínűséggel kapod a 0-t mérési eredményként? Milyen valószínűséggel kapod az 1-et mérési eredményként?

2. Alíz szeretne találni egy θ szöget úgy, hogy mindkét valószínűség $1/2$ legyen. Milyen θ-t válasszon? (Lehet, hogy több lehetőség is van!)

2.3. Kvantumbitek szimulálása QUIRKY segítségével

A kvantumszámítás törvényei meglehetősen furcsák, és a legtöbbünknek nincs kvantumszámítógépe, amellyel kísérletezhetne. Szerencsére a QUIRKY új képességekre tett szert a múlt hét óta, és most már lehetővé teszi számunkra egy *kvantum*bit szimulálását![9] Kezdésként menj az alábbi oldalra:

https://www.quantum-quest.org/quirky

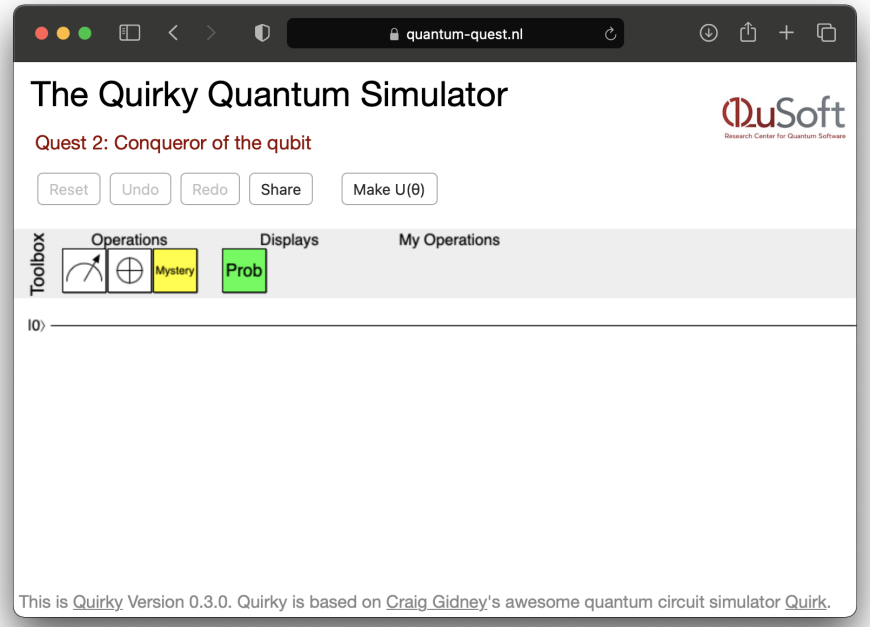
és kattints a „Quest 2" gombra. A böngésződ a 2.3. ábrához hasonlóan fog kinézni.

A fő különbség a múlt héthez képest az, hogy a 'vezeték' most egy *kvantumbit*-nek felel meg, amely a $|0\rangle$ állapotban van inicializálva.

Csakúgy, mint múlt héten, a *Toolbox* olyan műveleteket tartalmaz, amelyeket alkalmazhatunk, ha áthúzzuk őket a Toolbox-ból a vezeték fölé és elengedjük:

Az első doboz,  , lehetővé teszi egy kvantumbit mérését. Menjünk tovább, és építsük meg a következő egyszerű kvantumszámítást a QUIRKY-ben:

[9] Miért akarunk egyáltalán kvantumszámítógépeket építeni, ha ilyen szépen tudjuk őket szimulálni a meglévő számítógépeken? Ennek oka abban rejlik, hogy míg a QUIRKY-hez hasonló szimulátorok jól működnek, ha csak néhány kvantumbited van, viszont amikor a kvantumbitek száma növekszik, gyorsan leállnak. Látni fogjuk, hogy miért van ez így a 4. alfejezetben, a 4.1.1. alfejezeton.



2.3. ábra. QUIRKY a 2. küldetéshez.

Észre fogod venni, hogy az egyvonalas vezeték dupla vonalassá változott. A QUIRKY-ban az egyvonalas vezetékek kvantumbitekre, a dupla vonalas vezetékek pedig hagyományos vagy „klasszikus” bitekre utalnak. Valóban, a 2.2. alfejezet alapján tudjuk, hogy amikor egy kvantumbitet mérünk, az eredmény vagy nulla, vagy egy lesz bizonyos valószínűségekkel, azaz egy valószínűségi bit.

Az eredmények valószínűségeinek megtekintéséhez használhatjuk a múlt hétről már ismert valószínűség kijelzőt Prob. Adjunk hozzá egy ilyet a számításunkhoz, és nézzük meg, mi történik:

Úgy tűnik, hogy a mérési eredmény 100% valószínűséggel nulla lesz (mozgasd az egeret a doboz fölé, hogy megerősítsd a gyanúnkat). Természetesen pontosan ezt várjuk. Amikor $|0\rangle$-t mérünk, az eredmény mindig nulla lesz a 2.7. egyenlet mérési szabályai szerint.

A fejezet hátralévő részében a Toolbox többi dobozáról fogunk beszélni.

2.4. Műveletek egy kvantumbiten

Mielőtt mérnénk egy állapotot, előfordulhat, hogy szeretnénk valamilyen műveletet végrehajtani rajta. De milyen műveleteket végezhetünk egy kvantumbiten? Például, amikor elindítjuk a kvantumszámítógépünket, a kvantumbitje mindig $|0\rangle$ állapotban lesz, így alkalmaznunk kell egy műveletet, hogy létrehozzunk egy érdekes állapotot, például $|\psi(\theta)\rangle$-t. Bármi is legyen a művelet, annak egy másik kvantumbit állapotot kell előállítania kimenetként. Más szóval, a kvantumbit állapotterét önmagára kell leképeznie. Emlékezz vissza a 2.1. ábrára, ahol láttuk, hogy ez az állapottér egy körnek felel meg, tehát olyan módokat keresünk, amelyek a kört önmagára képezik le.

Először nézzük meg a NOT műveletet, amelyet pontosan ugyanúgy definiálhatunk, mint a valószínűségi biteknél az 1.9. egyenletben:

$$\text{NOT}\,|0\rangle = |1\rangle\,, \qquad \text{NOT}\,|1\rangle = |0\rangle\,.$$

Hogyan terjeszthetjük ki a NOT műveletet tetszőleges kvantumbit állapotokra? Ahogy a valószínűségi biteknél tettük az 1.2.1. alfejezetben, itt is a linearitás ötletét fogjuk használni. Ha egy M művelet definiálva van $|0\rangle$-ra és $|1\rangle$-re, akkor tetszőleges kvantumbit állapotra az alábbiak szerint definiálhatjuk:

$$M\Big(\psi_0 |0\rangle + \psi_1 |1\rangle\Big) = \psi_0 M|0\rangle + \psi_1 M|1\rangle . \tag{2.8}$$

Ki is írhatjuk a 2.8. egyenletet explicit módon, a vektoros jelöléssel:

$$M\begin{pmatrix}\psi_0\\ \psi_1\end{pmatrix} = M\left(\psi_0\begin{pmatrix}1\\0\end{pmatrix} + \psi_1\begin{pmatrix}0\\1\end{pmatrix}\right) = \psi_0 M\begin{pmatrix}1\\0\end{pmatrix} + \psi_1 M\begin{pmatrix}0\\1\end{pmatrix}. \tag{2.9}$$

Ahogy korábban már említettük, a matematikában egy M műveletet, amely teljesíti ezt a feltételt, lineárisnak nevezünk, és egy művelet ilyen módon történő kiterjesztését „lineáris kiterjesztésnek" hívjuk. A lényeg az, hogy ha M lineáris, és tudjuk, hogyan hat $|0\rangle$-ra és $|1\rangle$-re, akkor meg tudjuk határozni, hogyan hat tetszőleges kvantumbit állapotokra!

A 2.8. egyenletben csak a $|0\rangle$ és $|1\rangle$ vektorokat vettük figyelembe. Általánosságban azonban igaz, hogy

$$M\Big(a|\psi\rangle + b|\phi\rangle\Big) = a M|\psi\rangle + b M|\phi\rangle \tag{2.10}$$

tetszőleges $|\psi\rangle$, $|\phi\rangle$ vektorokra és a, b számokra. Látod, hogyan következik a 2.10. egyenlet a 2.8. egyenletből?

A kvantummechanika törvényei garantálják, hogy bármely lineáris M művelet lehetséges kvantumbit művelet – feltéve, hogy a teljes kvantumbit állapotteret önmagára képezi le! Ez azt jelenti, hogy minden kvantumbit állapotot (pontot a körön) egy kvantumbit állapotra (pontra a körön) képez le.

A NOT művelet esetében a lineáris kiterjesztés eredménye

$$\mathrm{NOT}\Big(\psi_0|0\rangle + \psi_1|1\rangle\Big) = \psi_0|1\rangle + \psi_1|0\rangle, \quad \text{vagyis} \quad \mathrm{NOT}\begin{pmatrix}\psi_0\\ \psi_1\end{pmatrix} = \begin{pmatrix}\psi_1\\ \psi_0\end{pmatrix}. \tag{2.11}$$

Figyeld meg, hogy a 2.8., 2.9. és 2.11. egyenletek pontosan úgy néznek ki, mint az 1.12. és 1.10. egyenletek – kivéve, hogy most ψ_0 és ψ_1 is lehetnek negatívak. A 2.2. ábra szerint a NOT művelet egy *tükrözést* jelent a 45 fokos tengelyre nézve (ez igaz a valószínűségi bitekre is). Ezt a 2.4. ábra szemlélteti. Nyilvánvaló, hogy a NOT a kvantumbit állapotterét (a kört) önmagára képezi. Így a NOT művelet érvényes művelet egy kvantumbiten.

A QUIRKY-ben a NOT művelet a kvantumbiteken ugyanúgy néz ki, mint a hagyományos biteken, nevezetesen $\boxed{\oplus}$. Próbáld meg most megépíteni a következő kvantumszámítást:

Most úgy tűnik, hogy a mérési eredmény 100% -ban egy lesz. Valóban, az eredeti $|0\rangle$ állapotot a NOT művelet átalakítja $|1\rangle$ állapottá, így az eredmény mindig egy lesz a mérési szabályok szerint a 2.7. egyenletben.

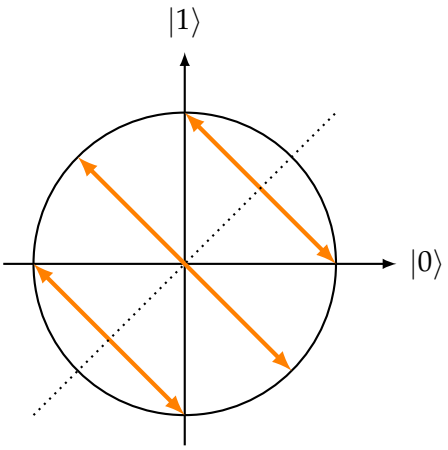
Hasonlóképpen definiálhatunk kvantumbit műveleteket, ha más tengelyek körüli tükrözéseket veszünk figyelembe. Például ilyen a Z művelet, amelyet az alábbiak szerint definiálhatunk:

$$Z|0\rangle = |0\rangle, \qquad Z|1\rangle = -|1\rangle, \tag{2.12}$$

a vízszintes $|0\rangle$-tengely körüli tükrözésnek felel meg. Valóban, ha Z-t lineárisan kiterjesztjük, akkor az tetszőleges kvantumbit állapotra így hat:

$$Z\begin{pmatrix}\psi_0\\ \psi_1\end{pmatrix} = \begin{pmatrix}\psi_0\\ -\psi_1\end{pmatrix},$$

ami bizonyosan kvantumbit állapotokat kvantumbit állapotokra képez le.



2.4. ábra. A kvantumbiteken definiált NOT művelet a 2.11. egyenletben egy 45 fokos (vagy $\pi/4$) tengely körüli tükrözést jelent (szaggatott vonal).

2.2. Házi feladat: A Z művelet

Tekintsük a következő két kvantumbit állapotot:

$$|+\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}, \qquad |-\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}.$$

1. Számold ki $Z|+\rangle$-t és $Z|-\rangle$-t.
2. Ábrázold a Z műveletet grafikusan a körön úgy, mint a 2.4. ábrán.

2.2. Gyakorló feladat: A linearitás nem elegendő (opcionális)

Tekintsd a MAD műveletet, amelyet úgy kapunk, hogy a MAD $|0\rangle = |0\rangle$ és MAD $|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ műveletet lineárisan kiterjesztjük. Keress egy olyan $|\psi\rangle$ állapotot, hogy MAD $|\psi\rangle$ nem érvényes kvantumbit állapot. Így MAD *nem* egy érvényes művelet a kvantumbiteken!

2.4.1. Forgatások

Eddig csak azt tudjuk, hogyan hozzuk létre a $|0\rangle$ és $|1\rangle$ állapotokat a QUIRKY segítségével. A kvantuminformatika nem lenne túl szórakoztató, ha ezek lennének az egyetlen lehetőségeink! Ahhoz, hogy érdekesebb állapotokat hozzunk létre, más kvantumműveleteket kell kitalálnunk.

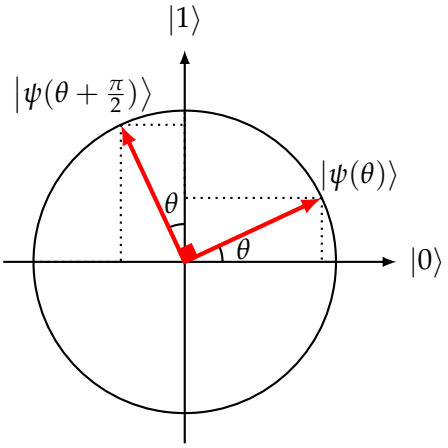
Egy természetes ilyen művelet az, hogy a kört egy rögzített szöggel *elforgatjuk*. Jelöljük a θ szöggel történő forgatást $U(\theta)$-val. Mindig feltételezheted, hogy a szög a $[0, 2\pi)$ tartományban van. Mivel $|0\rangle = |\psi(0)\rangle$ és $|1\rangle = |\psi(\frac{\pi}{2})\rangle$, ez a művelet az alábbiak szerint hat a bázisvektorokra (lásd 2.5. ábra):

$$U(\theta)|0\rangle = |\psi(\theta)\rangle, \qquad U(\theta)|1\rangle = |\psi(\theta + \tfrac{\pi}{2})\rangle. \tag{2.13}$$

Ezt a definíciót vektoros jelöléssel is kiírhatjuk:

$$U(\theta)\begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}\cos\theta\\ \sin\theta\end{pmatrix}, \qquad U(\theta)\begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}-\sin\theta\\ \cos\theta\end{pmatrix}, \tag{2.14}$$

ahol felhasználtuk, hogy $\cos(\theta + \frac{\pi}{2}) = -\sin\theta$ és $\sin(\theta + \frac{\pi}{2}) = \cos\theta$.



2.5. ábra. A $|0\rangle$ és $|1\rangle$ állapotok θ szöggel elforgatva, ld. 2.13. egyenlet.

Ahogy korábban is, most is a linearitást fogjuk használni, hogy $U(\theta)$-t kiterjesszük a bázisvektorokról tetszőleges kvantumbit állapotokra. A következő feladatban meg fogod mutatni, hogy az így kapott $U(\theta)$ művelet valóban forgatásként hat a kvantumbit állapotokra. Ez konkrétan azt jelenti, hogy a művelet kvantumbit állapotokat kvantumbit állapotokra képez le, így $U(\theta)$ érvényes művelet a kvantumbiteken!

2.3. Gyakorló feladat: Egy qubit forgatása

1. Számold ki $U(\alpha)\begin{pmatrix}\psi_0\\ \psi_1\end{pmatrix}$ értékét a 2.8. és 2.13. egyenletek segítségével.

2. Használd $|\psi(\theta)\rangle$ definícióját a 2.5. egyenletben, hogy igazold tetszőleges α és β szögekre, hogy
$$U(\alpha)\,|\psi(\beta)\rangle = |\psi(\alpha+\beta)\rangle\,. \tag{2.15}$$
Ez azt jelenti, hogy $U(\theta)$ forgatásként hat tetszőleges $|\psi(\beta)\rangle$ kvantumbit állapotra.

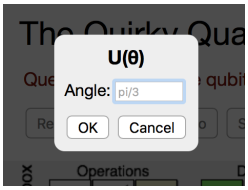
Segítség: A szögek összegére és különbségére vonatkozó trigonometrikus addíciós képletek hasznosak lehetnek:

$$\sin(\alpha\pm\beta) = \sin\alpha\cos\beta \pm \cos\alpha\sin\beta, \qquad \cos(\alpha\pm\beta) = \cos\alpha\cos\beta \mp \sin\alpha\sin\beta. \tag{2.16}$$

Figyeld meg, hogy a 90 fokos (azaz $\pi/2$-vel történő) forgatás nem ugyanaz, mint a tükrözés. Valóban, míg mind a NOT művelet, mind a $U(\pi/2)$ forgatás a $|0\rangle$-t az $|1\rangle$-re képezi le, különböző módon hatnak $|1\rangle$-re:

$$\text{NOT}\,|1\rangle = |0\rangle\,, \qquad U(\pi/2)\,|1\rangle = -\,|0\rangle\,.$$

Hogyan forgathatunk egy kvantumbitet a QUIRKY-ben? Mivel végtelen sok $U(\theta)$ forgatási művelet van, nem tudtuk mindet hozzáadni a Toolboxhoz. Ehelyett hozzáadhatod saját forgatásaidat a Toolboxhoz! Gyakoroljunk egy 30°-os forgatás hozzáadásával. Kezdésként válaszd ki a 'Make $U(\theta)$' lehetőséget a menüsorban. Ekkor megjelenik egy új ablak, ahol megadhatod a szöget:



Írd be a `pi/6` értéket, ami 30°-nak felel meg, és erősítsd meg az OK gomb megnyomásával. Gratulálok! Sikeresen hozzáadtad az $U(\pi/6)$ forgatást a Toolboxhoz, amely most így néz ki:

A forgatás teszteléséhez építsd meg a következő számítást a QUIRKY-ben:

Gondoljuk végig gyorsan, hogy ez az eredmény logikus-e. A $|0\rangle = \binom{1}{0}$ állapotból indultunk ki. A 2.14. egyenlet szerint bármely $U(\theta)$ forgatás a $|0\rangle$-t $|\psi(\theta)\rangle = \binom{\cos(\theta)}{\sin(\theta)}$-ra képezi le. A mi esetünkben $\theta = \pi/6$, és

$$|\psi(\pi/6)\rangle = \begin{pmatrix}\cos(\pi/6)\\ \sin(\pi/6)\end{pmatrix} = \begin{pmatrix}\sqrt{3}/2\\ 1/2\end{pmatrix}.$$

A kvantummérés 2.7. egyenletben leírt szabályait alkalmazva arra a következtetésre jutunk, hogy az 1 kimenetel valószínűsége

$$p_1 = \left(\frac{1}{2}\right)^2 = \frac{1}{4} = 25\%,$$

ami pontosan az, amit a QUIRKY mondott nekünk. A következő feladatban a QUIRKY segítségével hasonló módon teszteld az $U(\theta)$ forgatás hatását a másik bázisvektorra, $|1\rangle$-re.

2.3. Házi feladat: A 30°-os forgatás tesztelése

1. Építsd meg a következő műveletsorozatot a QUIRKY-ben: Először készítsd elő az $|1\rangle$ kvantumbit állapotot, majd forgass ugyanazzal a $\pi/6$ szöggel, és végül mérd meg a kvantumbitet.
2. Használd a QUIRKY 'Probability Display' funkcióját a mérési eredmények valószínűségének meghatározásához. Indokold meg, hogy a QUIRKY által adott válasz helyes.
3. Módosítsd az első kérdésre adott műveletsort úgy, hogy a mérési eredmény nulla valószínűsége 42 százalék legyen.

2.4.2. Kvantumműveletek kompozíciója

Mindig komponálhatunk (azaz egymás után alkalmazhatunk) két adott kvantumbit műveletet, M-et és N-et, hogy új kvantumbit műveletet kapjunk. Valóban, ha $|\psi\rangle$ a bemeneti állapot és először alkalmazzuk az M-et, akkor $M(|\psi\rangle) = M|\psi\rangle$-t kapunk. Ha ezután alkalmazzuk az N-et, a kapott állapot $N(M|\psi\rangle)$ lesz. Ezt az összetett műveletet NM-mel fogjuk jelölni, így

$$NM|\psi\rangle = N(M|\psi\rangle).$$

Vigyázz, nehogy összekeverd a két művelet sorrendjét. Ha az összetett művelet NM, ez azt jelenti, hogy először az M-et alkalmazzuk, és másodszor az N-et! Ez azért van, mert az M a $|\psi\rangle$ mellett áll, tehát először ennek kell hatnia az állapotra.

2.4. Gyakorló feladat: Egy összetett művelet linearitása (opcionális)

Igazold, hogy az NM is lineáris.

Segítség: Használd a 2.10. egyenletet.

Hasonlóképpen három vagy több kvantumbit műveletet is össze tudunk fűzni. Ezt úgy írjuk, hogy ONM és így tovább. Különösképpen, új kvantumbit műveleteket kaphatunk forgatások és tükrözések kompozíciójával. Ezt a későbbiekben, a 2.4.3. alfejezetben tárgyaljuk.

Érdekes megfigyelni, hogy az összes eddig tárgyalt kvantumbit művelet invertálható. Ez azt jelenti, hogy bármely M művelethez létezik egy másik művelet, amelyet M^{-1}-zel (M inverzzel) jelölünk, úgy, hogy ha először M-et, majd M^{-1}-et (vagy fordítva) alkalmazzuk, bármely kvantumbit állapota változatlan marad.[10] Formálisan ezt a következőképpen írhatjuk:

$$M^{-1}M = MM^{-1} = I, \tag{2.17}$$

ahol I az identitás művelet, amely a „triviális" tulajdonsággal rendelkezik

$$I\left|0\right\rangle = \left|0\right\rangle, \qquad I\left|1\right\rangle = \left|1\right\rangle \tag{2.18}$$

(Az I-t definiálhattuk volna $U(0)$-ként, azaz a nulla szögű forgatásként is.) Ezért bármely $\left|\psi\right\rangle$ állapotra $I\left|\psi\right\rangle = \left|\psi\right\rangle$ igaz a linearitás által történő kiterjesztéskor.

Például nézzük meg az U műveletet. Geometriailag egyértelmű, hogy ha először β-val, majd $-\beta$-val forgatunk, akkor egy kvantumbit állapota változatlan marad. Ennek formálisabb megértéséhez csak kétszer kell használnunk a 2.15. egyenletet:

$$U(-\beta)U(\beta)\left|\psi(\alpha)\right\rangle = U(-\beta)\left|\psi(\alpha+\beta)\right\rangle = \left|\psi(\alpha+\beta-\beta)\right\rangle = \left|\psi(\alpha)\right\rangle$$

és hasonlóan megy, ha először $-\beta$-val, majd β-val forgatunk. Ez azt jelenti, hogy az $U(\beta)^{-1}$ inverz művelete egyszerűen $U(-\beta)$:

$$U(\beta)^{-1} = U(-\beta).$$

Hasonlóképpen, mivel a NOT művelet tükrözést jelent, egyértelmű, hogy kétszer alkalmazva egy kvantumbit állapotát változatlanul hagyja. Valóban, az 1.9. egyenletből következik, hogy

$$\mathrm{NOT}\,\mathrm{NOT}\left|0\right\rangle = \mathrm{NOT}\left|1\right\rangle = \left|0\right\rangle \qquad \mathrm{NOT}\,\mathrm{NOT}\left|1\right\rangle = \mathrm{NOT}\left|0\right\rangle = \left|1\right\rangle.$$

A linearitás alapján ez azt jelenti, hogy $\mathrm{NOT}\,\mathrm{NOT}\left|\psi\right\rangle = \left|\psi\right\rangle$ bármely $\left|\psi\right\rangle$ állapotra, tehát a NOT nem csak invertálható, hanem saját maga inverze is, azaz $\mathrm{NOT}^{-1} = \mathrm{NOT}$.

2.5. Gyakorló feladat: Egy összetett művelet inverze

Mutasd meg, hogy ha M és N invertálhatóak, akkor NM is az. Fejezd ki az összetett művelet inverzét, $(NM)^{-1}$-et az N^{-1} és M^{-1} inverzek segítségével.

Valójában az is kimutatható, hogy bármely lineáris művelet, amely a kvantumbit állapotterét önmagára képezi le, szükségszerűen invertálható. Valóban, ez igaz a forgatásokra, $U(\theta)$, és szintén igaz lesz a tükrözésekre, $V(\theta)$, amelyeket a következőkben tárgyalunk. Ez ellentétben áll a valószínűségi bitek műveleteivel, ahol például a véletlen átbillentés művelet, $F(1/2)$, bármely állapotot az egyenletes eloszlásba, $\left(\begin{smallmatrix}1/2\\1/2\end{smallmatrix}\right)$-be, képez le (lásd 1.6. gyakorló feladat), és így nem invertálható.

2.4.3. Tükrözések

Bármely kvantumbit művelet vagy egy forgatás, vagy egy tükrözés. Már ismerjük a legáltalánosabb forgatást, $U(\theta)$-t, amelyet a 2.13. egyenletben definiáltunk. Azonban a tükrözések közül eddig csak kettővel találkoztunk: Z és NOT, ld. 2.11.-2.12. egyenlet. De hogy néz ki a legáltalánosabb tükrözés?

Egy módja annak, hogy bármilyen tükrözést elérjünk, ha veszünk egy rögzített tükrözést (mondjuk a NOT tükrözést) és összekomponáljuk alkalmas forgatásokkal úgy, hogy a tükrözés tengelye a megfelelő mértékben módosuljon. A következő feladatban meg fogod mutatni, hogyan lehet két különböző módon előállítani a Z tükrözést a NOT tükrözésből.

2.4. Házi feladat: Z a NOT-ból

Legyenek Z, NOT és $U(\theta)$ a 2.12., 2.11. és 2.13. egyenletekben definiált kvantumműveletek.

1. Találj egy θ szöget úgy, hogy $Z = U(\theta)\,\mathrm{NOT}\,U(-\theta)$ teljesüljön.
2. Találj egy θ szöget úgy, hogy $Z = \mathrm{NOT}\,U(\theta)$ teljesüljön.

Tudod ábrázolni ezt a két transzformációs sorozatot a körön?

Segítség: Nézd meg a 2.4. ábrát és azt az ábrát, amit a 2.2. házi feladatban készítettél.

Kiderül, hogy valójában bármilyen tükrözést elérhetsz az feladathoz hasonló trükk alkalmazásával. A legáltalánosabb tükrözés a következő formájú:

$$V(\theta) = \mathrm{NOT}\,U(\theta) = U(-\theta)\,\mathrm{NOT}. \tag{2.19}$$

Például egy nagyon hasznos művelet a Hadamard transzformáció, amely a következőképpen hat a bázisállapotokra (lásd 2.6. ábra):

$$H\,|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = |+\rangle\,, \qquad H\,|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = |-\rangle\,. \tag{2.20}$$

Ezt az általános tükrözés következő speciális eseteként kapjuk:

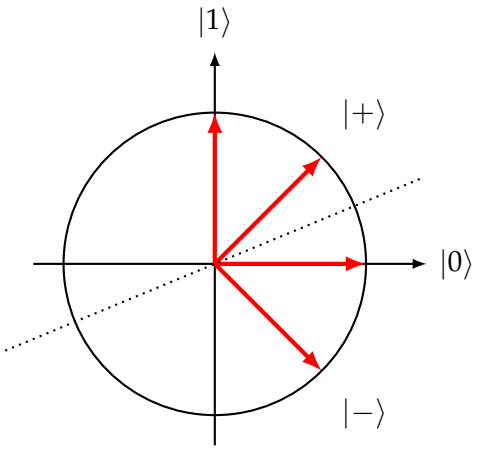
$$H = V(\pi/4). \tag{2.21}$$

Összefoglalva, bármely qubit művelet vagy egy $U(\theta)$ forgatás, vagy egy $V(\theta)$ tükrözés, valamely θ szögre.

2.5. Kvantumállapotok megkülönböztetése

Alíz egy robotszamarak közötti futóversenyt néz, és feljegyzi, hogy a kedvenc szamara nyer-e: 1-est ír, ha ez történik, és 0-t egyébként. Ezt az információt egy qubitbe is kódolhatná: a legáltalánosabb esetben $|\psi(\theta_0)\rangle$ állapotot készít a 0 esetben (ha nem nyer a kedvence), vagy $|\psi(\theta_1)\rangle$ állapotot az 1 esetben (ha a kedvenc szamara nyer). Alíz ezeket az állapotokat úgy tudja egyszerűen létrehozni, hogy $U(\theta_0)$-t vagy $U(\theta_1)$-et alkalmazza $|0\rangle$-ra, ahogy azt a 2.13. egyenletben láttuk. Most tegyük fel, hogy Alíz ezt a qubitet átadja Botinak. Meg tudja-e Boti tippelni, hogy melyik bitérték (0 vagy 1) lett kódolva, pusztán erre az egy qubitra alapozva? Javulnának-e Boti esélyei, ha először elvégezhetne egy forgatást vagy tükrözést? Ezt az elképzelést a következő feladatban gyakorolhatod.

[10] Ez a jelölés a a 2.17. egyenlettel együtt a következőkre emlékeztethet: Ha x egy nem nulla szám, akkor $x^{-1} = \frac{1}{x}$ az inverze, ami azt jelenti, hogy $xx^{-1} = x^{-1}x = 1$.



2.6. ábra. A Hadamard művelet H egy qubiten egy tükrözésnek felel meg a 22,5 fokos (vagy $\pi/8$ szögű) tengely körül (szaggatott vonal). Az ábrán láthatók még a $|0\rangle$, $|1\rangle$, $|+\rangle$, és $|-\rangle$ állapotok a 2.20. egyenletből.

2.6. Gyakorló feladat: Plusz és mínusz

Képzeld el, hogy kapsz egy qubitet, amely a következő két állapot egyikében van:

$$|+\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}, \qquad |-\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}.$$

Ki szeretnéd találni, hogy melyik állapotban van. Alkalmazhatsz valamilyen forgatást, majd mérhetsz. Milyen forgatást érdemes alkalmaznod, és ekkor milyen valószínűséggel tudod eltalálni a helyes állapotot?

Ha különböző bitértékeket különböző kvantumállapotoknak szeretnél megfeleltetni, ügyelned kell arra, hogy ne használd egyszerre a $|\psi(\theta)\rangle$ és $|\psi(\theta+\pi)\rangle$ állapotokat, mivel ezek az állapotok nem megkülönböztethetők.

2.7. Gyakorló feladat: Megkülönböztethetetlen állapotok

Mutasd meg, hogy a $|\psi(\theta)\rangle$ és $|\psi(\theta+\pi)\rangle = -|\psi(\theta)\rangle$ állapotokat semmilyen módon nem lehet megkülönböztetni. Azaz, függetlenül attól, hogy milyen műveletet alkalmazol, mielőtt megméred a qubitet, a mérési eredmények mindkét esetben mindig ugyanolyan valószínűségűek lesznek.

Érdemes összehasonlítani a 2.6. és 2.7. gyakorló feladatokokat. Amikor két állapot csupán egy előjelben tér el, teljesen megkülönböztethetetlenek, ahogy azt a 2.7. gyakorló feladatban láttuk. Gyakorlati szempontból a $\pm|\psi(\theta)\rangle$ vektorok *ugyanazt* az állapotot írják le. Ezzel szemben a 'relatív' előjelek fontosak, mint a 2.6. gyakorló feladatban, és akár tökéletesen megkülönböztethető állapotokat is eredményezhetnek!

A következő házi feladatban kiderítheted, mi az optimális módja két *tetszőleges* kvantumállapot megkülönböztetésének.

2.5. Házi feladat: Két állapot megkülönböztetése

Legyen θ és θ' két szög. Az egyszerűség kedvéért feltételezzük, hogy $-\frac{\pi}{2} \leq \theta \leq \theta' \leq \frac{\pi}{2}$. Tegyük fel, hogy Éva ad neked egy qubitet, amely vagy $|\psi(\theta)\rangle$ állapotban, vagy $|\psi(\theta')\rangle$ állapotban van, mindkettő 50-50% valószínűséggel. (Például feldobhatna egy szabályos érmét, hogy eldöntse, melyik állapotot adja neked.) A feladatod az, hogy kiderítsd, a

két lehetőség küzül melyik állapotot kaptad. Néhány lépésben megtalálhatod az optimális eljárást:

1. Először alkalmazz egy $U(\phi)$ forgatást valamilyen ϕ szöggel. Melyik két lehetséges állapotot kapod?

2. Ezután mérd meg a qubitet, és értelmezd az eredményt a következőképpen: Ha az eredmény 0, akkor a tipped legyen az, hogy a kapott állapot $|\psi(\theta)\rangle$ volt, egyébként pedig hogy $|\psi(\theta')\rangle$. Mi a valószínűsége annak, hogy ezzel helyesen azonosítod a kapott állapotot? Írj fel egy képletet θ, θ' és ϕ függvényében.

   Ötlet: Először számold ki a siker valószínűségét, feltételezve, hogy az első állapotot kaptad, majd a siker valószínűségét, feltételezve, hogy a második állapotot kaptad, végül idézd fel, hogy valójában 50-50% eséllyel kapod egyik vagy másik állapotot.

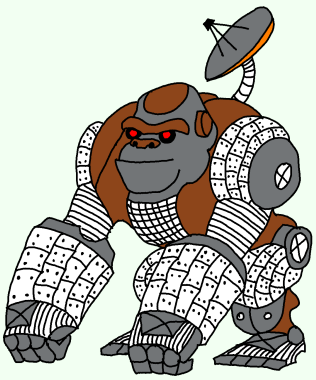
3. Még mindig szabadon megválaszthatod a ϕ forgatási szöget. Mi a siker valószínűsége θ és θ' függvényében, ha a lehető legjobban választod meg ϕ-t?

   Ötlet: Próbáld meg használni a 2.16. egyenlet trigonometrikus azonosságait. Ezek alapján megmutathatod, hogy

$$\sin^2\alpha = \frac{1}{2}\big(1-\cos(2\alpha)\big), \qquad \cos^2\alpha = \frac{1}{2}\big(1+\cos(2\alpha)\big). \tag{2.22}$$

   Ha elakadsz, használhatod a Wolfram Alpha-t is.

2.8. Gyakorló feladat: Törött láb és kar (opcionális, csillagos)



Probléma: Alíz és Boti szeretnek a városuk körüli vadonban barangolni. Hogy könnyebben tudjanak haladni, a rengetegben két nagy gorilla robotot építettek, amik kényelmesen a hátukon tudják őket hordani. Azonban egy szerencsétlen napon Boti robotja véletlenül leesik egy szikláról! Szerencsére Boti néhány zúzódással megússza a zuhanást, de a robotja elég súlyosan megsérül: egy karja, egy lába és a kommunikációs eszköze is eltörik. Botinál nincsenek tartalék alkatrészek, de legalább sikerül rövid időre megjavítania a kommunikációs eszközt. Sajnos ez csak egy bitet vagy egy qubitet tud küldeni, mielőtt végleg elromlik. Boti szeretné közölni Alízzal, hogy melyik lába (bal vagy jobb) és melyik karja (szintén bal vagy jobb) tört el a robotjának, hogy Alíz leszerelhesse a megfelelő alkatrészt a robotjáról és leereszthesse neki a szakadékba. Alíz csak egy végtagot (vagy lábat, vagy kart) tud küldeni neki, mert mindkét robotnak haza kell tudnia menni (amit még három végtaggal meg tudnak tenni). A helyzetet bonyolítja, hogy nincs Alíznál az összes szükséges szerszám, hogy bármely végtagot le tudja szerelni a robotjáról. Boti emlékszik, hogy Alíz vagy a lábakhoz, vagy a karokhoz való szerszámokat vitte magával (de nem mindkettőt), viszont nem emlékszik melyiket.

Négy lehetséges kombinációja van annak, hogy Boti robotjának melyik lába és karja tört el – feltételezheted, hogy mindegyik valószínűsége $1/4$. Hasonlóképpen, Alíz kétféle végtagot tud leszerelni a robotjáról (vagy a lábakhoz, vagy a karokhoz van szerszáma), és feltételezheted, hogy a $1/2$ valószínűséggel vitte magával a megfelelő szerszámokat.

Kérdések:

1. Ha Boti csak egy bitet tud küldeni Alíznak, hogyan döntsön annak értékéről attól függően, hogy a négy lehetséges kombináció közül miképp törtek el a robotjának végtagjai? Hogyan kellene Alíznak értelmeznie az üzenetet, és eldöntenie, hogy a bal vagy a jobb végtagot küldje el? (Emlékezz, hogy Alíz csak lábakat, vagy csak karokat tud küldeni, de Boti nem tudja hogy melyiket.) Ha mindketten az optimális stratégiát használják, mekkora valószínűséggel fogja Alíz helyesen értelmezni Boti üzenetét, és elküldeni a megfelelő végtagot a robotjához?

2. Mi változik, ha Boti egy kvantumbitet küldhet? A helyzettől függően Boti négy állapot közül választhat egyet, és Alíz a nála lévő szerszámtól függően két forgatás közül választhat mielőtt megmérné a kvantumbitet. Mi az optimális közös stratégiájuk, és mekkora valószínűséggel lesz sikeres?

Feltételezheted, hogy Alíz és Boti tudják, hogyan kell értelmezni egymás üzeneteit, mivel előre megbeszélték, mit tesznek majd ha valaha előfordul ez a konkrét vészhelyzet.

2.5.1. Egy másik rejtélyes művelet

Még nem beszéltünk a sárga dobozról a Quirky-ban. Ellentétben a múlt heti rejtélyes művelettel, amely biteken működött, ez a heti rejtélyes doboz kvantumbiteken működik. Nevezzük ezt a rejtélyes kvantumműveletet M-nek. Hogyan tudhatnánk meg, mi zajlik a doboz belsejében? Első lépésként vizsgáljuk meg, hogy mi az $M|0\rangle$. A Quirky-ban ezt az állapotot a következő beállítással hozhatjuk létre:

Az ismeretlen állapot meghatározásának problémáját kvantumállapot tomográfiának nevezzük, mivel különböző mérések elvégzésével szeretnénk egy ismeretlen kvantumállapotot 'kívülről' rekonstruálni. Ez egy alapvető feladat, amellyel a kísérletezők nap mint nap szembesülnek, amikor meg akarják győződni arról, hogy a laboratóriumban előállított kvantumállapot valóban az a kvantumállapot, amelyet létre akartak hozni!

Már jelentős információt nyerhetünk az ismeretlen állapoton végzett mérés segítségével. Ennek megértéséhez írjuk fel

$$M|0\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \end{pmatrix}.$$

Ha végrehajtunk egy mérést, akkor a 2.6. egyenlet szerint az 1 kimenetelt ψ_1^2 valószínűséggel kapjuk meg. Ez azt jelenti, hogy ha a fenti kísérletet sokszor megismételjük, akkor azt várjuk, hogy az 1 kimenetel előfordulási aránya nagyjából ψ_1^2. Ez teljesen analóg ahhoz, ahogyan egy érmét sokszor feldobva és a fejek és írások számát megszámolva becsülhetjük meg a pénzérme tisztaságát, ahogyan a múlt héten megbeszéltük az 1.3. alfejezetben. Ez eljárást biztosít számunkra a ψ_1^2 becslésére. A Quirky-ban egyszerűen használhatjuk a valószínűség kijelzőt a mérés után, hogy meghatározzuk az 1 kimenetel valószínűségét:

Így megállapítjuk, hogy $\psi_1^2 \approx 11,7\%$. Mivel $M|0\rangle$ egy egységvektor, azt is következtethetjük, hogy $\psi_0^2 = 1 - \psi_1^2 \approx 88,3\%$. Azonban az amplitúdók lehetnek negatívak is, így ez csak a ψ_0

és ψ_1 előjeleit határozza meg! Most emlékezzünk vissza a 2.7. gyakorló feladatra, hogy $|\psi\rangle$ és $-|\psi\rangle$ megkülönböztethetetlenek, így csak a teljes előjel figyelembevételével remélhetjük $|\psi\rangle = M|0\rangle$ meghatározását. Tehát két lehetőség marad:

$$\pm\begin{pmatrix}\sqrt{88,3\%}\\ \sqrt{11,7\%}\end{pmatrix}, \quad \pm\begin{pmatrix}\sqrt{88,3\%}\\ -\sqrt{11,7\%}\end{pmatrix}$$

Figyeljük meg, hogy ez a helyzet nagyon hasonló a a 2.6. gyakorló feladathoz, ahol el kellett döntenünk $|+\rangle$ és $|-\rangle$ között. Az utolsó házi feladatban tisztázni fogod a helyzetet és feltárod a rejtélyes doboz belső működését.

2.6. Házi feladat: Rejtvényfejtés

1. Hogyan döntheted el, hogy a két lehetőség közül melyik az eset? Használd a Quirky-t, hogy meghatározd az $M|0\rangle$ kvantumállapotot az előjelig.
2. Hasonlóképpen határozd meg az $M|1\rangle$ kvantumállapotot az előjelig.
3. *Bónusz kérdés:* Az 1. és 2. lépések teljesen meghatározzák a M kvantum műveletet? Ha igen, írj le egy képletet az M-re. Ha nem, hogyan tudhatod meg az M-et?

2.6. Fizikai kitekintés (opcionális)

A *Kvantum Küldetés* fő fókusza a kvantuminformatika *matematikájára* irányul. Azonban, mivel kis kvantumszámítógépeket már építettek laboratóriumokban világszerte, hasznos egy kicsit ismerni a kvantumszámítástechnika *fizikáját* is. Milyen fizikai hatások teszik lehetővé a kvantumszámítógépek működését?

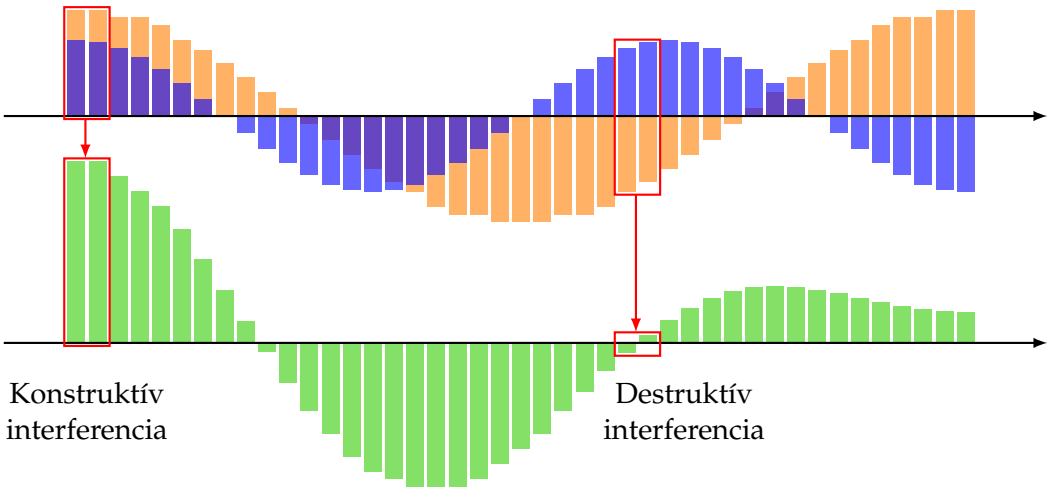
2.6.1. Interferencia

Az egyik legfontosabb fizikai hatás, amit a kvantuminformatika használ, az interferencia – az egymást átfedő hullámok vagy rezgések kölcsönhatása. Az interferenciát megfigyelheted például az egymást keresztező két hajó által keltett vízhullámokon, vagy ha egyszerre dobsz két követ egy nyugodt tóba. Amikor a hullámok erősítik egymást, azt *konstruktív* interferenciának nevezzük, amikor pedig gyengítik egymást, azt *destruktív* interferenciának (lásd 2.7. ábra).

Az interferencia más helyzetekben is fontos szerepet játszik, például a hanghullámok esetében. Egy ismerős példa lehet a zajszűrő fejhallgatókban létrejövő destruktív interferencia. Ezek úgy működnek, hogy rögzítik a háttérzajt, és visszajátsszák azt neked, de ellentétes rezgési irányban. Amikor ez a rögzített hang átfedi az eredeti zajt, kioltják egymást: $1-1=0$. Ha a fejhallgató nem fordítaná meg a rezgés irányát, hanem úgy játszaná vissza a hangot, ahogy érkezik, sokkal hangosabb zajt hallanál: $1+1=2$. Ez gyakorlatilag hallókészülékké változtatná a fejhallgatódat!

A kvantumszámítás egyik fontos különbsége a valószínűségi számításhoz képest, hogy *mindkét* típusú interferenciát – konstruktívat és destruktívat – is képes használni, míg a valószínűségi számítás csak konstruktív interferenciát használhat. Ennek matematikai szemléltetéséhez idézzük fel az $F(1/2)$ valószínűség-átbillentési műveletet és a H Hadamard-műveletet az 1.14. és 2.20. egyenletekből:

$$\begin{aligned} F(1/2)\,[0] &= \frac{1}{2}[0] + \frac{1}{2}[1], & H|0\rangle &= \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle, \\ F(1/2)\,[1] &= \frac{1}{2}[0] + \frac{1}{2}[1], & H|1\rangle &= \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle. \end{aligned} \tag{2.23}$$



2.7. ábra. Két hullám interferenciája: minden pontban a kék és a narancssárga hullám amplitúdói összeadódnak, létrehozva a zöld hullámot. Amikor mindkét amplitúdó előjele azonos, az interferencia *konstruktív*, és még nagyobb amplitúdót kapunk. Amikor az interferáló amplitúdók különböző előjelűek, az interferencia *destruktív*, és sokkal kisebb amplitúdót kapunk.

A négyzetgyököktől eltekintve a két művelet szinte azonos. Azonban vegyük észre, hogy míg $F(1/2)\,[1]$ plusz előjelű, addig $H\,|1\rangle$ mínusz előjelű. Ez apró különbségnek tűnhet, de messzeható következményei lehetnek.

Vizsgáljuk meg e két művelet hatását az egyenletes eloszlásra $\frac{1}{2}\,[0] + \frac{1}{2}\,[1]$ és annak kvantumanalógjára, a plusz állapotra $|+\rangle = \frac{1}{\sqrt{2}}\,|0\rangle + \frac{1}{\sqrt{2}}\,|1\rangle$. Az $F(1/2)$ valószínűség-átbillentési művelet az egyenletes eloszlásra a következőképpen hat:

$$\begin{aligned} F(1/2)\left(\frac{1}{2}\,[0] + \frac{1}{2}\,[1]\right) &= \frac{1}{2}\,F(1/2)\,[0] + \frac{1}{2}\,F(1/2)\,[1] \\ &= \frac{1}{2}\left(\frac{1}{2}\,[0] + \frac{1}{2}\,[1]\right) + \frac{1}{2}\left(\frac{1}{2}\,[0] + \frac{1}{2}\,[1]\right) \\ &= \left(\frac{1}{2}\cdot\frac{1}{2} + \frac{1}{2}\cdot\frac{1}{2}\right)[0] + \left(\frac{1}{2}\cdot\frac{1}{2} + \frac{1}{2}\cdot\frac{1}{2}\right)[1] \\ &= \frac{1}{2}\,[0] + \frac{1}{2}\,[1], \end{aligned}$$

ahol a linearitást és a 2.23. egyenletet használtunk, és összegyűjtöttük a $[0]$ és $[1]$ állapotok valószínűségeit. Figyeljük meg, hogy az $[1]$ állapot valószínűségei mindkét tagban erősítik egymást, így a végső valószínűség $1/2$ lesz. Ez nagyon intuitív: ha egy egyenletesen véletlen bitet átbillentesz, akkor az egyenletesen véletlen marad.

Vizsgáljuk meg most a Hadamard művelet H hatását a plusz állapotra $|+\rangle$:

$$\begin{aligned} H\left(\frac{1}{\sqrt{2}}\,|0\rangle + \frac{1}{\sqrt{2}}\,|1\rangle\right) &= \frac{1}{\sqrt{2}}\,H\,|0\rangle + \frac{1}{\sqrt{2}}\,H\,|1\rangle \\ &= \frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{2}}\,|0\rangle + \frac{1}{\sqrt{2}}\,|1\rangle\right) + \frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{2}}\,|0\rangle - \frac{1}{\sqrt{2}}\,|1\rangle\right) \\ &= \left(\frac{1}{\sqrt{2}}\cdot\frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}}\cdot\frac{1}{\sqrt{2}}\right)|0\rangle + \left(\frac{1}{\sqrt{2}}\cdot\frac{1}{\sqrt{2}} - \frac{1}{\sqrt{2}}\cdot\frac{1}{\sqrt{2}}\right)|1\rangle \\ &= |0\rangle\,. \end{aligned}$$

A számítás majdnem azonos, de az eredmény nagyon különböző – az $|1\rangle$ állapot amplitúdói teljesen kioltják egymást, és csak $|0\rangle$ marad. Ilyen destruktív interferencia lehetetlen valószínűségi biteknél, mert a valószínűségek mindig pozitívak – csak erősíthetik egymást, de soha nem olthatják ki egymást.

Bár a véletlen és kvantumbitek nagyon hasonlóak, ez a példa szemlélteti, hogyan viselkedhetnek különbözően a destruktív interferencia révén. Sok kvantum különlegesség, amellyel a következő hetekben találkozol, valamilyen módon ennek a jelenségnek a következménye. A destruktív interferencia lehetősége pontosan az, ami előnyt biztosít a kvantumszámítógépeknek a klasszikus számítógépekkel szemben – lehetővé teszi a kvantumszámítógép számára, hogy csak a helyes választ adja meg, a helytelen válaszokat pedig a destruktív interferencia révén kioltsa. Ahogy azt részletesebben látni fogjuk az 5.2. alfejezetben, ezt gyakran a Hadamard-kapuval érik el, ami ezért központi szerepet játszik sok kvantumalgoritmusban.

2.6.2. Polarizáció

Most, hogy matematikailag már ismerjük a qubit állapotokat és műveleteket, jó lenne ezeket valami fizikai dologhoz kapcsolni.

Egy qubit fizikai megvalósításai közül az egyik legkézzelfoghatóbb a fény polarizációja. A fény egy elektromágneses hullám, amely egyenes vonalban terjed a térben. Ez a hullám a terjedés irányára merőleges irányban rezeg. Több ilyen irány is lehetséges – egy előre haladó hullám rezeghet balról jobbra vagy fentről lefelé. Ezek a vízszintes és függőleges rezgési módok használhatók a qubit két bázisállapotának fizikai megvalósítására:

$$|\leftrightarrow\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad |\updownarrow\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

Általánosabban az állapot fizikai megvalósítására, egy olyan hullámot is használhatunk, amely θ szögben rezeg a vízszintes tengelyhez képest:

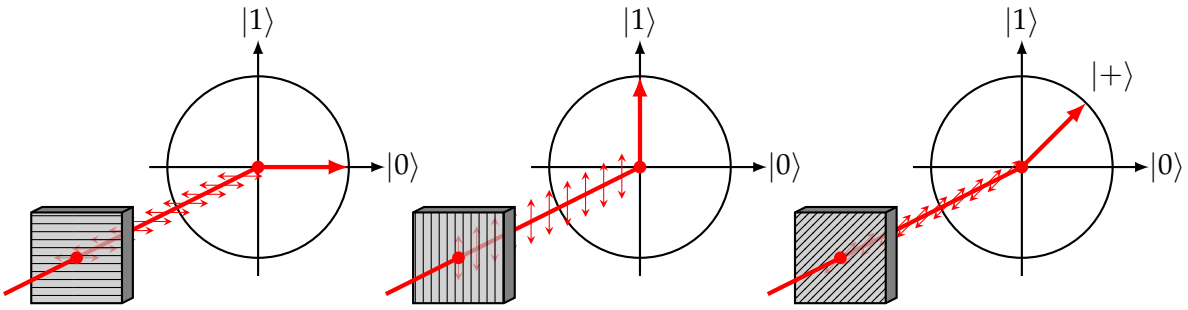
$$|\psi(\theta)\rangle = \cos\theta\,|\leftrightarrow\rangle + \sin\theta\,|\updownarrow\rangle = \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}.$$

Például a 45°-os szögben, a függőleges és vízszintes irányok között rezgő, átlósan polarizált fény az $|\psi(\pi/4)\rangle = |+\rangle$ állapotot valósítja meg. Figyeljük meg, hogy ebben a megvalósításban az elektromágneses hullám rezgési iránya megegyezik annak a vektornak az irányával, amelyet a 2.1.2. alfejezet 2.1. ábráján használtunk a qubit állapotának egységkörön való ábrázolására.

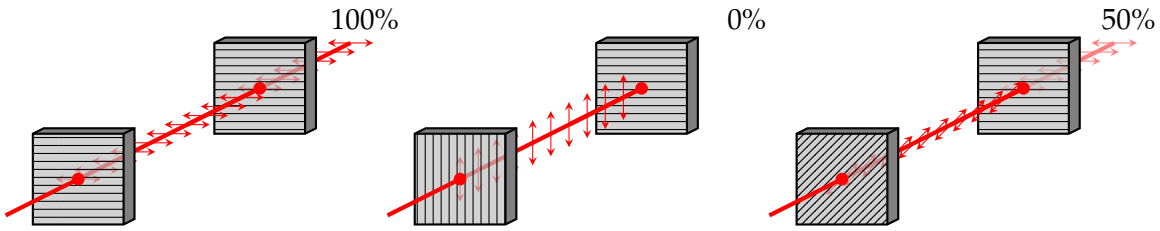
Ezen állapotok előállításához egyszerűen átengedhetünk egy fénysugarat egy polarizátoron, mint amilyen a napszemüvegedben vagy a moziban használt 3D szemüvegben van. A polarizátor csak a hullám egy részét engedi át – azt, amelynek rezgési iránya kompatibilis a polarizátor irányával. A $|\psi(\theta)\rangle$ állapot előállításához egyszerűen megdönthetjük a polarizátort θ szögben a vízszintes tengelyhez képest. Például, a 2.8. ábrán látható, hogyan lehet előállítani a $|0\rangle$, $|1\rangle$ és $|+\rangle$ állapotokat.

A qubit állapotok polarizált fénnyel való megvalósításának egy érdekes tulajdonsága, hogy a $|\psi(\theta)\rangle$ és $|\psi(\theta+\pi)\rangle$ állapotokat ugyanazzal az eljárással állíthatjuk elő – a polarizátor θ szögben való megdöntésével. Ez azt jelenti, hogy ennek a két állapotnak azonosnak kell lennie! Így a polarizáció intuitív magyarázatot ad arra, hogy miért kell a $|\psi\rangle$ és $-|\psi\rangle$ állapotoknak megkülönböztethetetlennek lenniük (lásd 2.7. gyakorló feladat).

A qubitek polarizált fényként való megvalósításánal egy másik előnye, hogy könnyen vizualizálhatjuk a mérést. Tegyük fel, hogy meg akarjuk mérni a $|\psi(\theta)\rangle$ állapotot, hogy meghatározzuk a 0 kimenetel valószínűségét. Ha az állapotot egy θ szögben polarizált fénysugárként kapjuk meg, egyszerűen átengedhetjük egy vízszintes polarizátoron, és megnézhetjük, mennyi fény jut át – ha a fényerő 70%-ra csökkent, akkor a 0 kimenetel valószínűsége 70%. Például, ha a bejövő fénysugár vízszintesen polarizált volt, az összes átjut, míg ha függőlegesen polarizált



2.8. ábra. A vízszintesen, függőlegesen és átlósan polarizált fény használható a $|0\rangle$, $|1\rangle$ és $|+\rangle$ qubit állapotok reprezentálására.



2.9. ábra. A fény vízszintes polarizációjának mértéke meghatározható úgy, hogy átengedjük egy vízszintes polarizátoron, majd megmérjük a fényerejét. A vízszintesen, függőlegesen és átlósan polarizált fény esetében ez 100%, 0% és 50% fényerőt eredményez, ami egybeesik a 0 kimenet megfigyelésének valószínűségével a $|0\rangle$, $|1\rangle$ és $|+\rangle$ állapotok megmérésekor.

volt, semmi sem jut át. Egy átlósan polarizált fénysugár vízszintes polarizátoron való áthaladása pedig 50%-os fényerő-csökkenést eredményez (lásd 2.9. ábra).

2.9. Gyakorló feladat: Polarizációs kísérlet

Ha van otthon egy polarizált napszemüveged, felveheted, és megnézheted a telefonod vagy a számítógéped képernyőjét. Általában a képernyők polarizált fényt bocsátanak ki (amelynek polarizációs iránya az eszköztől függ). Amikor oldalra döntöd a fejed, látnod kell, hogy a képernyő világosabbá vagy sötétebbé válik. El tudod magyarázni, hogy miért van ez így?

A fény polarizációja csak egy példa arra, hogyan lehet egy qubitet megvalósítani a laboratóriumban. Egy másik példa a fényt hordozó részecske, a foton *helyzete* – mivel egy foton a kvantummechanika törvényei szerint viselkedik, egyidejűleg lehet két helyen szuperpozícióban. Ha ezeket a helyeket 0-nak és 1-nek nevezzük, a foton állapota megfeleltethető egy qubitnek. Sok más lehetőség is van: egy szupravezető áramkörben az áram egyszerre folyhat mindkét irányban, egy elektron egyszerre két pályán is lehet egy atom körül, és így tovább. Röviden, bármely kvantummechanikai rendszer amelynek van két különböző állapota, az lehet azok szuperpozíciójában is, így potenciálisan használható egy qubit fizikai megvalósítására.

2.7. A gyakorló feladatok megoldásai

2.1. Gyakorló feladat megoldása

Figyeljük meg, hogy

$$|+\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix} = |\psi(\pi/4)\rangle\,, \qquad |-\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\-1\end{pmatrix} = |\psi(-\pi/4)\rangle\,.$$

Tehát a szögek $\theta = \pm\pi/4$, és a két állapot rendre 45 fokkal felfelé és lefelé helyezkedik el $|0\rangle$-tól.

2.2. Gyakorló feladat megoldása

Legyen $|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, ami egy érvényes kvantumállapot. Mivel MAD a linearitással történő kiterjesztéssel nyerhető, a 2.8. egyenlet szerint

$$\begin{aligned}\mathrm{MAD}\,|\psi\rangle &= \mathrm{MAD}\left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right) = \frac{1}{\sqrt{2}}\mathrm{MAD}\,|0\rangle + \frac{1}{\sqrt{2}}\mathrm{MAD}\,|1\rangle \\ &= \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{2}(|0\rangle + |1\rangle) = \left(\frac{1}{\sqrt{2}} + \frac{1}{2}\right)|0\rangle + \frac{1}{2}|1\rangle\,.\end{aligned}$$

De

$$\left(\frac{1}{\sqrt{2}} + \frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2 = 1 + \frac{1}{\sqrt{2}} \neq 1,$$

tehát $\mathrm{MAD}\,|\psi\rangle$ nem érvényes kvantumállapot.

2.3. Gyakorló feladat megoldása

1. $U(\alpha)$ egy tetszőleges állapoton a következőképpen hat:

$$\begin{aligned}U(\alpha)\begin{pmatrix}\psi_0\\\psi_1\end{pmatrix} &= U(\alpha)\left(\psi_0\begin{pmatrix}1\\0\end{pmatrix} + \psi_1\begin{pmatrix}0\\1\end{pmatrix}\right) = \psi_0\begin{pmatrix}\cos\alpha\\\sin\alpha\end{pmatrix} + \psi_1\begin{pmatrix}-\sin\alpha\\\cos\alpha\end{pmatrix} \\ &= \begin{pmatrix}\psi_0\cos\alpha - \psi_1\sin\alpha\\\psi_0\sin\alpha + \psi_1\cos\alpha\end{pmatrix}.\end{aligned}$$

2. Mivel $|\psi(\beta)\rangle = \begin{pmatrix}\cos\beta\\\sin\beta\end{pmatrix}$,

$$\begin{aligned}U(\alpha)\,|\psi(\beta)\rangle &= U(\alpha)\begin{pmatrix}\cos\beta\\\sin\beta\end{pmatrix} = \begin{pmatrix}\cos\beta\cos\alpha - \sin\beta\sin\alpha\\\cos\beta\sin\alpha + \sin\beta\cos\alpha\end{pmatrix} = \begin{pmatrix}\cos(\alpha+\beta)\\\sin(\alpha+\beta)\end{pmatrix} \\ &= |\psi(\alpha+\beta)\rangle\,.\end{aligned}$$

2.4. Gyakorló feladat megoldása

Vegyünk egy tetszőleges állapotot $\psi_0 |0\rangle + \psi_1 |1\rangle$, először használjuk M linearitását, majd N linearitását:

$$\begin{aligned} NM(\psi_0 |0\rangle + \psi_1 |1\rangle) &= N\big(M(\psi_0 |0\rangle + \psi_1 |1\rangle)\big) \\ &= N\big(\psi_0 M |0\rangle + \psi_1 M |1\rangle\big) = \psi_0 NM |0\rangle + \psi_1 NM |1\rangle . \end{aligned}$$

Az utolsó lépésben a 2.10. egyenletet használtuk.

2.5. Gyakorló feladat megoldása

Mivel $(NM)^{-1} = M^{-1}N^{-1}$, hiszen bármely $|\psi\rangle$ esetén

$$M^{-1}N^{-1}NM |\psi\rangle = M^{-1}(N^{-1}N(M |\psi\rangle)) = M^{-1}(M |\psi\rangle) = M^{-1}M |\psi\rangle = |\psi\rangle$$

és

$$NMM^{-1}N^{-1} |\psi\rangle = N(MM^{-1}(N^{-1} |\psi\rangle)) = N(N^{-1} |\psi\rangle) = NN^{-1} |\psi\rangle = |\psi\rangle .$$

2.6. Gyakorló feladat megoldása

Alkalmazd az $U(-\pi/4)$ műveletet és mérj. Biztosan meg tudod tippelni az állapotot!

2.7. Gyakorló feladat megoldása

Láttuk fentebb, hogy bármely rotációk és tükrözések kombinációja M lineáris. Így ha $M |\psi(\theta)\rangle = |\psi(\theta')\rangle = \binom{\cos\theta'}{\sin\theta'}$, akkor $M\big(-|\psi(\theta)\rangle\big) = -|\psi(\theta')\rangle = \binom{-\cos\theta'}{-\sin\theta'}$. A 2.6. egyenlet alapján a p_0 és p_1 mérési kimenetelek valószínűsége mindkét állapot esetében azonos.

2.9. Gyakorló feladat megoldása

A fejed megdöntésével megváltoztatod a napszemüveged polarizátora és a képernyő által kibocsátott elektromágneses fényhullámok rezgési iránya közötti szöget. Mivel a polarizátoron áthaladó fény mennyisége ettől a szögtől függ, a képernyő világosabbnak vagy sötétebbnek fog tűnni. Hasonlóan, a θ szög megváltoztatása megváltoztatja annak valószínűségét, hogy a $|\psi(\theta)\rangle$ állapot megmérésekor a 0 kimenetelt kapjuk.

2.8. Gyakorló feladat megoldása

Boti alapvetően 2 bitet szeretne küldeni, amelyek jelzik, hogy melyik láb és kar van eltörve. Azonban Alíz csak az egyik bitre kíváncsi, mivel csak egyféle szerszám van nála.

1. Jelöljük mindkét bit lehetséges értékeit L-lel (bal) és R-rel (jobb). Boti például megpróbálhatja a két bit "többségi szavazatát" küldeni, ami a következő kódolásnak felel meg: $LL \mapsto L$, $RR \mapsto R$. A fennmaradó két esetet tetszőlegesen kódolhatja, például $LR \mapsto L$, $RL \mapsto R$. Alíz stratégiája egyszerűen az, hogy azt a végtagot küldi, amelyik megfelel Boti üzenetének (a bal végtagot, ha L-t kapott, és a jobb végtagot, ha R-t kapott). Ekkor a helyes választás valószínüsége

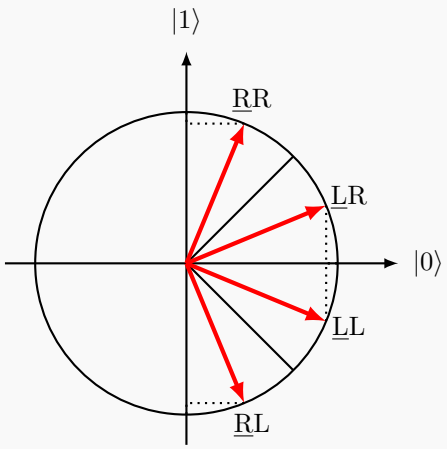
$$\frac{1}{4}\left(1+1+\frac{1}{2}+\frac{1}{2}\right)=\frac{3}{4}=0.75, \tag{2.24}$$

   ahol a zárójeles összegben lévő tagok a sérült robot négy lehetséges állapotának felelnek meg, és Alíz helyes döntésének valószínűségét fejezik ki.

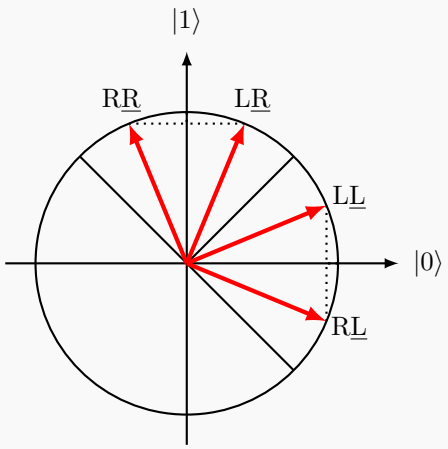
2. Boti például az alábbi qubit állapotok közül küldhet egyet attól függően, hogy melyik végtagok vannak eltörve (LL bal láb és bal kar, LR bal láb és jobb kar, stb.)

$$\begin{aligned}
|LL\rangle &= \cos(\pi/8)\,|0\rangle - \sin(\pi/8)\,|1\rangle \\
|LR\rangle &= \cos(\pi/8)\,|0\rangle + \sin(\pi/8)\,|1\rangle \\
|RR\rangle &= \cos(3\pi/8)\,|0\rangle + \sin(3\pi/8)\,|1\rangle \\
|RL\rangle &= \cos(3\pi/8)\,|0\rangle - \sin(3\pi/8)\,|1\rangle
\end{aligned}$$

   A láb bit visszafejtéséhez Alíz egyszerűen csak megméri a qubitet. A kéz bit visszafejtéséhez Alíz először alkalmazza az $U(\pi/4)$ műveletet és csak utána mér. (Érdemes megjegyezni, hogy Alíz nem tudja mindkét bitet visszafejteni, mivel az eredeti állapot a mérés után elvész.) Könnyen látható, hogy mind a két esetben a siker valószínűsége $\cos^2(\pi/8) = \frac{1}{2} + \frac{1}{2\sqrt{2}} \approx 0,85$. Ez markánsan jobb, mint a klasszikus esetben!



A láb bit visszafejtése



A kéz bit visszafejtése

3. Küldetés: Az összefonódás kibogozása

Az előző küldetésekben megismertük, hogyan viselkedik egy valószínűségi bit és egy qubit. Ezen a héten azt fogjuk megtanulni, mi történik, ha kettő van belőlük. Először két bitet vizsgálunk meg, és megnézzük, milyen állapotokban lehetnek, illetve hogyan lehetnek *korreláltak*. Ezután rátérünk két qubit vizsgálatára, és arra, hogy mit jelent az, ha *összefonódnak*.

3.1. Két valószínűségi bit

Ha két érmét helyezel az asztalra, azok négy lehetséges konfigurációban lehetnek:

Ezek négy lehetséges bitsorozatnak[11] felelnek meg: 00, 01, 10, 11.

Ha van egy valószínűségi bit párod, az állapotukat négy lehetséges determinisztikus állapot valószínűségi eloszlása írja le. Más szóval, az állapotukat négy szám határozza meg: $p_{00}, p_{01}, p_{10}, p_{11} \geq 0$, ahol $p_{00} + p_{01} + p_{10} + p_{11} = 1$. Akárcsak egyetlen bit esetében, ezt felírhatjuk egy vektorként:

$$\begin{pmatrix} p_{00} \\ p_{01} \\ p_{10} \\ p_{11} \end{pmatrix}. \tag{3.1}$$

Bár pontos, ez a leírás elég nehézkes, mert nehéz lehet követni, hogy melyik valószínűség melyik bitkonfigurációhoz tartozik (vajon a p_{10} vagy a p_{01} jön előbb?!), és ez csak nehezebb lesz, ha kettőnél több bitünk van. Sokkal kényelmesebb olyan jelölést használni, ami közvetlenül lehetővé teszi, hogy nyomon kövessük az egyes bitsorozatokhoz rendelt valószínűségeket. Ezért kibővítjük az 1.3. egyenletben bevezetett jelölést, amit egyetlen valószínűségi bitnél használtunk, és a fenti kétbites állapotot a következőképpen írjuk fel:

$$p_{00}[00] + p_{01}[01] + p_{10}[10] + p_{11}[11]. \tag{3.2}$$

Ha szeretnéd, a fenti számokra bármkor gondolhatsz úgy mint a 3.1. egyenletben szereplő vektor 4 koordinátájának az 1.2. egyenlethez hasonló módon:

$$[00] = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \qquad [01] = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \qquad [10] = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \qquad [11] = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}. \tag{3.3}$$

Ennek a jelölésnek egy előnye több bit esetén az, hogy egyszerűen kihagyhatjuk a nulla értékű elemeket. Ezért egyszerűen azt írhatjuk, hogy

$$\frac{1}{2}[00] + \frac{1}{2}[11]$$

az alábbi sokkal hosszabb kifejezés helyett

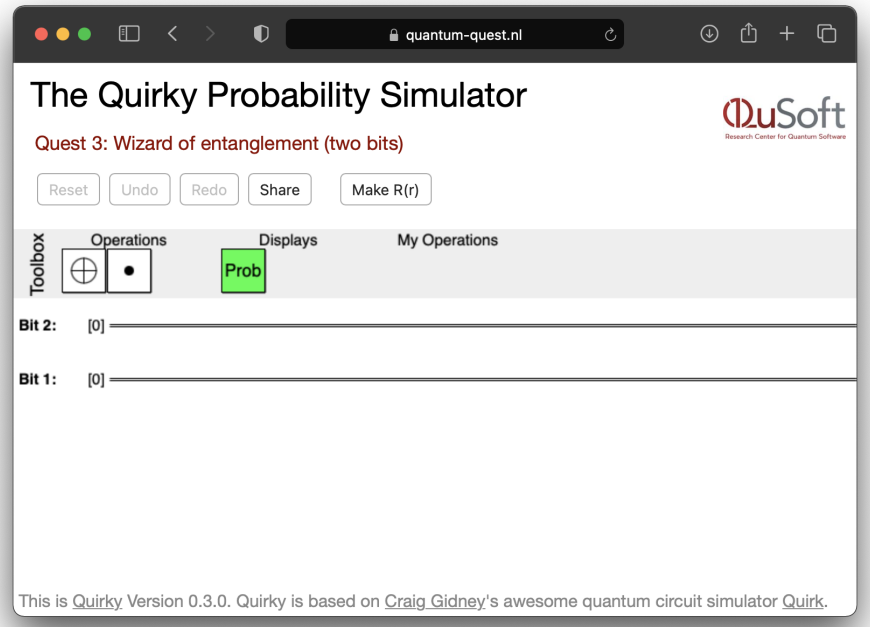
$$\frac{1}{2}[00] + 0\,[01] + 0\,[10] + \frac{1}{2}[11].$$

Ezzel a jelöléssel sokkal könnyebb leírni a méréseket és műveleteket több valószínűségi biten. Két valószínűségi bitet vizsgálhatunk QUIRKY segítségével, amely a múlt hét óta ismét új képességeket kapott. Kezdésként menj a következő oldalra:

[11] A 'sorozat' szimbólumok egymásutánját jelenti (ebben az esetben: bitértékek sorozatát). Ez lesz a kedvenc jelölésünk, amikor több érmével vagy bittel foglalkozunk.

https://www.quantum-quest.org/quirky

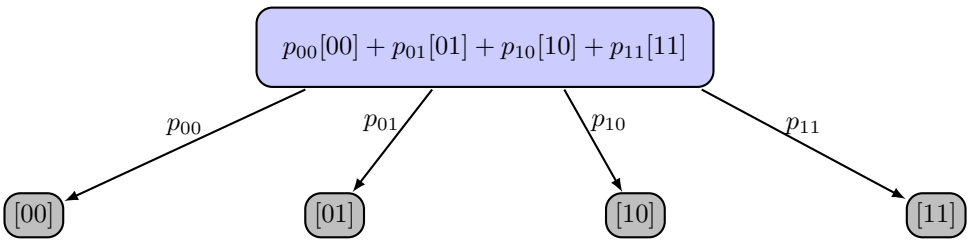
és kattints a „Quest 3”-ra, majd válaszd a „Two Bits” opciót. A böngésződ hasonlóan fog kinézni, mint a 3.1. ábrán. Vedd észre, hogy most *két* vezetékünk van, amelyek két bitnek felelnek meg, és [00] állapotban vannak inicializálva. Talán meglepő módon az első bit az *alsó* vezetéknek, a második bit pedig a *felső* vezetéknek felel meg. Emellett van egy új doboz: [•] (de a rejtélyes doboz eltűnt). Ennek a doboznak a jelentését később tárgyaljuk ebben a fejezetben.



3.1. ábra. QUIRKY a 3. küldetéshez.

3.1.1. Mindkét bit mérése

Két valószínűségi bit mérése (vagy "megfigyelése") ugyanúgy működik, mint az 1.18. egyenletben egyetlen bit esetén. Négy lehetséges eredmény (00, 01, 10, vagy 11) egyikét kapod a megfelelő valószínűséggel:



(3.4)

A mérés után mindkét bit állapota már nem négy lehetőség eloszlása, hanem egyetlen opció, amely megfelel a megfigyelt mérési eredménynek. Hogy hangsúlyozzuk ezt a különbséget, világoskéket használunk a valószínűségi bitekhez, és szürkét a determinisztikusakhoz a mérés után. Hogyan mérhetjük mindkét bitet QUIRKY-ban? Egyszerűen használjuk a valószínűségi kijelzőt, így:

Vedd észre, hogy alapértelmezetten a valószínűségi kijelző *mindkét* vezetékhez csatlakozik, így *mindkét* bit valószínűségeit mutatja. A valószínűségek sorrendje ugyanaz, mint a 3.1. egyenlet egyenletben a 4-vektoros jelölésnél. Nem kell megjegyezned a sorrendet. Egyszerűen vidd az egérkurzort a táblázat fölé, hogy emlékeztesd magad (köszi, Craig!).

Például, ha a két valőszínűségi bit állapota

$$\frac{1}{2}[00] + \frac{1}{2}[11], \tag{3.5}$$

akkor mérési eredményként vagy 00-t, vagy 11-et kapunk, mindkettőt 50% valószínűséggel. Figyeld meg, hogy ez az állapot különleges – ha látjuk, hogy az első bit mérési eredménye 0, azonnal tudjuk, hogy a második bit eredményének is 0-nak kell lennie, és hasonlóan, ha bármelyik eredmény 1. Mivel mindkét bit mérési eredménye mindig egyenlő, a (3.5) állapotú két bitet tökéletesen korreláltnak nevezzük. Alább látni fogjuk, hogyan lehet ilyen állapotokat létrehozni.

3.1.2. Lokális műveletek

Ha két vagy több valőszínűségi bited van, sokféleképpen végezhetsz rajtuk műveleteket. Például, végezhetsz műveletet az összesen egyszerre egy globális művelettel, vagy csak egyen vagy néhányon egy időben egy lokális művelettel. Nézzük először a lokális műveleteket.

Emlékezz a NOT műveletre az 1.2. alfejezet alfejezetből, amely átbillenti a bitet. Mi történik, ha két bitünk van, és csak az elsőn alkalmazzuk a NOT-ot? Ebben az esetben az első bitet kell átbillenteni, míg a másodiknak változatlannak kell maradnia. Ez azt jelenti, hogy az első biten végzett *lokális* NOT művelet, amit NOT_1-gyel jelölünk, a következőképpen működik:

$$\mathrm{NOT}_1\,[00] = [10], \quad \mathrm{NOT}_1\,[01] = [11], \quad \mathrm{NOT}_1\,[10] = [00], \quad \mathrm{NOT}_1\,[11] = [01]. \tag{3.6}$$

Hasonlóképpen, ha csak a második biten alkalmazzuk a NOT-ot, az eredményül kapott NOT_2 művelet a következőképpen működik:

$$\mathrm{NOT}_2\,[00] = [01], \quad \mathrm{NOT}_2\,[01] = [00], \quad \mathrm{NOT}_2\,[10] = [11], \quad \mathrm{NOT}_2\,[11] = [10]. \tag{3.7}$$

Amit most leírtunk, azok lokális NOT műveletek determinisztikus biteken. Hogyan kellene ezeket kiterjesztenünk valőszínűségi bitekre? Emlékezzünk vissza az 1.2.1. alfejezetből, hogy bármely művelet, amely teljesen meghatározott determinisztikus biteken, kiterjeszthető valőszínűségi bitekre *linearisan*. Például a NOT_2 a következőképpen működik két valőszínűségi biten:

$$\begin{aligned}
&\mathrm{NOT}_2\left(p_{00}[00] + p_{01}[01] + p_{10}[10] + p_{11}[11]\right) \\
&\quad = p_{00}[01] + p_{01}[00] + p_{10}[11] + p_{11}[10] \\
&\quad = p_{01}[00] + p_{00}[01] + p_{11}[10] + p_{10}[11],
\end{aligned}$$

ahol az első lépésben a 3.7. egyenletet használtuk, a második lépésben pedig csak átrendeztük a tagokat a bináris sorozatok sorba rendezéséhez. Ezt felírhatod a 4-vektoros jelölésben is, de az valamivel kevésbé intuitív:

$$\mathrm{NOT}_2 \begin{pmatrix} p_{00} \\ p_{01} \\ p_{10} \\ p_{11} \end{pmatrix} = \begin{pmatrix} p_{01} \\ p_{00} \\ p_{11} \\ p_{10} \end{pmatrix}. \tag{3.8}$$

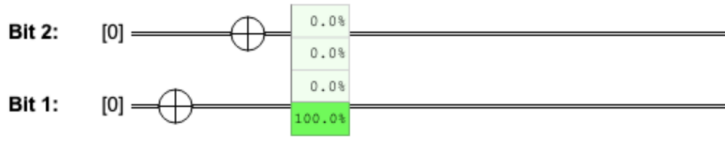
3.1. Gyakorló feladat: NOT_1 a 4-vektoros jelölésben (opcionális)

A 3.8. egyenlet egyenlethez hasonlóan írd fel a NOT_1 művelet hatását két valószínűségi biten a 4-vektoros jelölésben.

Egy egybites művelet alkalmazásához QUIRKY-ban az adott dobozt az első vagy a második vezetékre helyezzük. Például a következő sorozat előállítja a $[10]$ állapotot, és megmutatja a kimeneti valószínűségeket mindkét bit mérésekor:

Ez értelmes, mivel QUIRKY-ban az alsó vezeték felel meg az első bitnek.

Hasonlóképpen, ha először az egyik bitet billentjük át, majd a másikat, az eredmény a $[11]$ állapot lesz:



Nyilvánvaló, hogy a két NOT művelet alkalmazási sorrendje nem számít. Ez azt jelenti, hogy *párhuzamosan* is alkalmazhatjuk őket:

Ugyanígy alkalmazhatunk véletlen műveleteket az egyik biten. Például tegyük fel, hogy az első biten az $R(r)$ műveletet végezzük, amely r valószínűséggel lenullázza a bitet (az 1.13. egyenlet). Mivel $R(r)[0] = [0]$, ezért

$$R(r)_1[00] = [00], \qquad R(r)_1[01] = [01]. \tag{3.9}$$

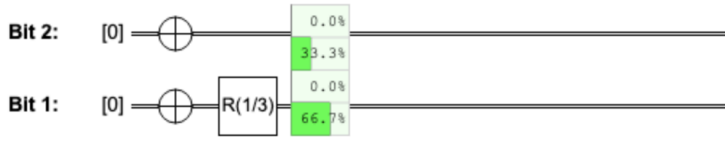
És mivel $R(r)[1] = r[0] + (1-r)[1]$, ezért

$$R(r)_1[10] = r[00] + (1-r)[10], \qquad R(r)_1[11] = r[01] + (1-r)[11]. \tag{3.10}$$

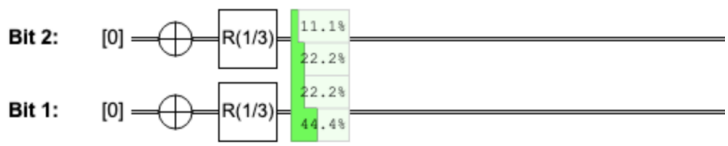
Például, ha előkészítjük a $[11]$ állapotot, és $R(1/3)$-at alkalmazunk az első biten, akkor a következőt kapjuk:

$$R(1/3)_1[11] = \frac{1}{3}[01] + \frac{2}{3}[11], \tag{3.11}$$

amit QUIRKY is megerősít:



Itt van egy még érdekesebb példa, amit az alábbi feladatban vizsgálhatsz meg:



(3.12)

3.1. Házi feladat: $R(r)$ a második biten

1. Írj fel képleteket $R(r)_2$-re a 3.9. és 3.10. egyenletekhez hasonlóan.
2. Magyarázd meg, miért ad QUIRKY helyes választ a 3.12. egyenletben.

3.1.3. Csak egy bit mérése

Ha két valószínűségi bited van és csak az egyiket méred, mi a valószínűsége annak, hogy a két lehetséges kimenet egyikét kapod? A jelölésünk különösen alkalmas ennek kiderítésére. Vegyünk ismét egy általános kétbites véletlen állapotot:

$$p_{00}[00] + p_{01}[01] + p_{10}[10] + p_{11}[11].$$

Ahhoz, hogy megtaláld a 0 kimenet valószínűségét egy bit mérésekor, egyszerűen össze kell adnod az összes olyan tag valószínűségét, ahol a mért bit a kívánt 0 állapotban van; hasonlóan az 1 kimenethez.

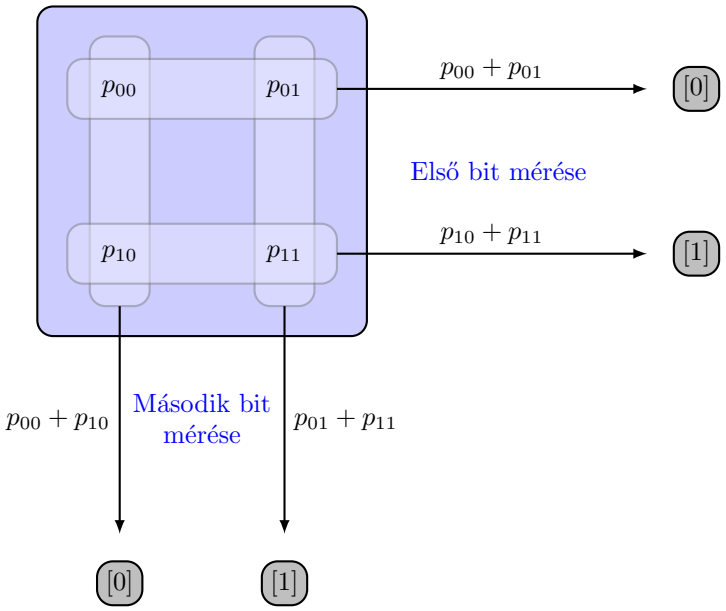
Például annak a valószínűsége, hogy 1-et kapunk az *első* bit mérésekor:

$$p_{10} + p_{11}, \tag{3.13}$$

ami megfelel a 3.4. egyenlet egyenletben szereplő valószínűségeknek, amelyek [10]-hez és [11]-hez vezetnek, azaz a két 1-gyel kezdődő bitsorozathoz. Hasonlóképpen, a 0 kimenet megfigyelésének valószínűsége a *második* bit mérésekor

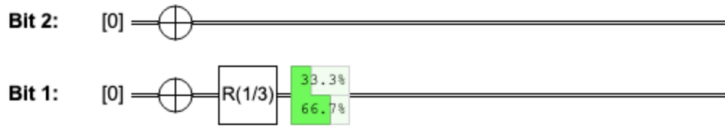
$$p_{00} + p_{10},$$

ami megfelel azoknak a valószínűségeknek, amelyek a nullára végződő két bitsorozathoz vezetnek, vagyis [00]-hoz és [10]-hez. Ezt könnyű kiszámolni, ha a négy valószínűséget egy 2×2-es négyzetbe rendezed, ahogy a 3.2. ábrán látható.

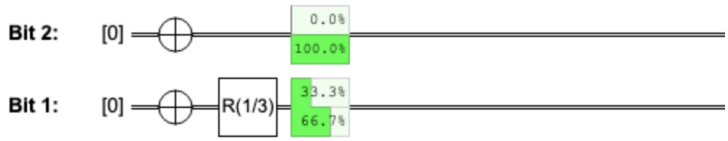


3.2. ábra. A mérési kimenetek valószínűségei, amikor csak egy bitet mérünk két valószínűségi bit közül.

QUIRKY-t is használhatjuk egy bit mérésekor a valószínűségek megjelenítésére. Egyszerűen méretezd át a valószínűségi kijelzőt úgy, hogy csak egy vezetéket fedjen le, például így:



Valójában egyszerre láthatjuk mind az első, mind a második bit mérésének kimeneteli valószínűségeit:



Figyeld meg, hogy az eredmény nagyon intuitív. Mivel a két bit soha nincs „korrelációban", világos, hogy az első bitnek $\frac{1}{3}[0] + \frac{2}{3}[1]$ állapotban, a második bitnek pedig $[1]$ állapotban kell lennie.

3.1.4. A másik bit állapota

A mérés után a mért bit determinisztikus állapotban van, amely megfelel a mérési eredménynek (ahogy egyetlen bit mérésénél is - lásd 1.18. egyenlet). De mi a helyzet a másik bittel, amit nem mértünk? Annak állapota a mérés után általában nem lesz determinisztikus. Például, ha a két valószínűségi bit kezdeti állapota

$$\frac{1}{2}[10] + \frac{1}{2}[11] \tag{3.14}$$

és az első bitet mérjük, az 1 megfigyelésének valószínűsége $1/2 + 1/2 = 1$. Más szóval, az első bit a 3.14. egyenletben determinisztikus, és a két valószínűség valójában csak a második bitet írja le. Ezért intuitíven gondolhatsz erre az állapotra úgy, mint két külön valőszínűségi bit "kombinációjára": $[1]$ és $\frac{1}{2}[0] + \frac{1}{2}[1]$ (később többet beszélünk arról, hogyan lehet két valőszínűségi bitet kombinálni a 3.1.7. alfejezetben). Ezért értelmes, hogy a második bit állapotának egyenletesen véletlennek kell lennie a mérés után, vagyis

$$\frac{1}{2}[0] + \frac{1}{2}[1].$$

Általánosabban, tegyük fel, hogy két valőszínűségi bittel kezdünk tetszőleges állapotban

$$p_{00}[00] + p_{01}[01] + p_{10}[10] + p_{11}[11] \tag{3.15}$$

és mérjük az első bitet. A megmaradt bit állapota általában függ a mérési eredménytől. Például, ha az eredmény 1 volt, a második bit állapotának meghatározásához először összegyűjtjük a 3.15. egyenlet összes olyan tagját, ahol az első bit értéke 1:

$$p_{10}[10] + p_{11}[11].$$

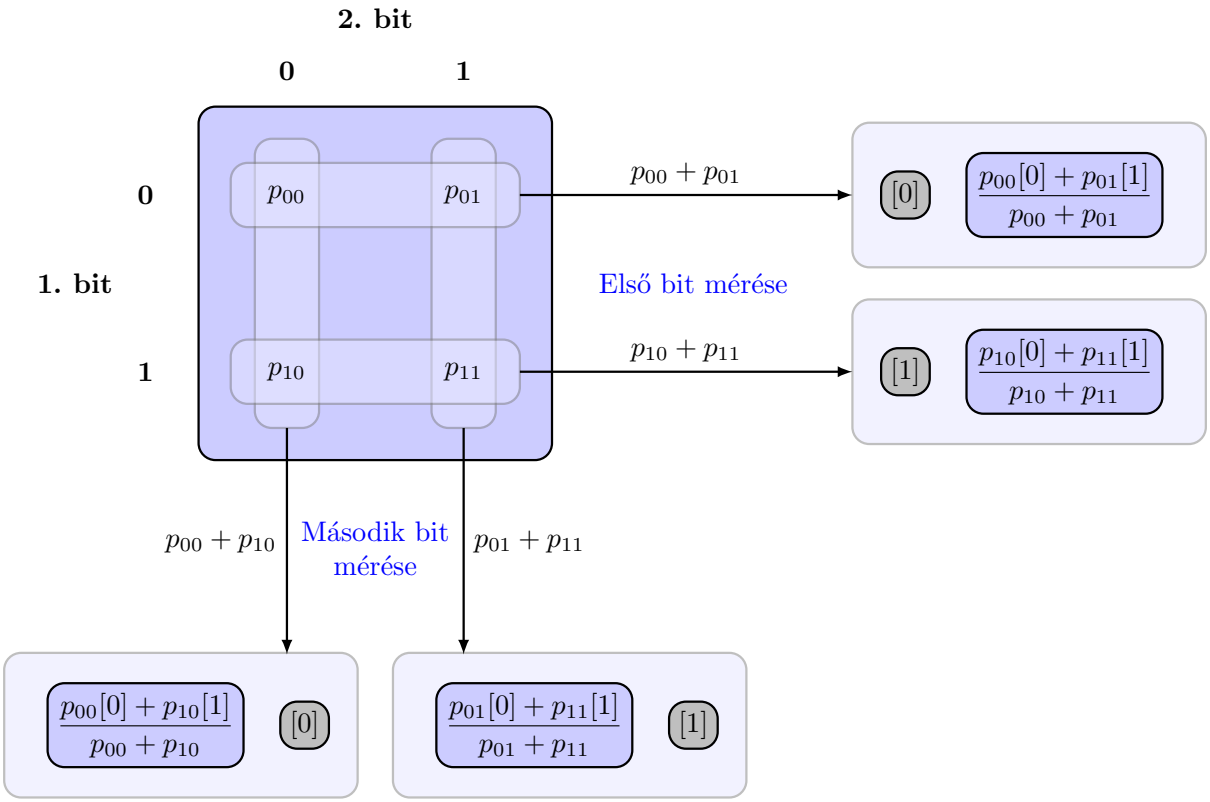
Ezután figyelmen kívül hagyjuk az első bitet, mivel már tudjuk, hogy az értéke 1:

$$p_{10}[0] + p_{11}[1].$$

Végül, mivel ez a két valószínűség nem feltétlenül ad összesen egyet, elosztjuk őket az összegükkel, $p_{10} + p_{11}$-gyel:

$$\frac{p_{10}}{p_{10} + p_{11}}[0] + \frac{p_{11}}{p_{10} + p_{11}}[1]. \tag{3.16}$$

Ez a második bit valószínűségi eloszlása, amikor az első bitet mérjük és 1-es eredményt kapunk (lásd 3.3. ábra jobb oldala). A többi esetet is összefoglalja a 3.3. ábra.



3.3. ábra. Kimeneti valószínűségek és a megmaradt bit állapota két valőszínűségi bit közül az egyik mérésekor. A mérés után a mért bit determinisztikussá válik (szürke), míg a másik bit véletlen marad (világoskék).

Hogy lássuk, ezek a szabályok értelmesek, ellenőrizzük, hogy az első bit, majd a második bit mérése ugyanazokat a valószínűségeket adja-e, mint mindkét bit közvetlen mérése. Például az első bitből 1-et és a második bitből 0-t p_{10} valószínűséggel kapunk. Valóban, a 3.13. egyenlet szerint az első bitből 1-et kapunk $p_{10} + p_{11}$ valószínűséggel, továbbá a 3.16. egyenlet szerint a második bitből 0-t kapunk $p_{10}/(p_{10} + p_{11})$ valószínűséggel. Így a teljes valószínűsége annak, hogy először 1-et kapunk az első bitből, majd 0-t a másodikból

$$(p_{10} + p_{11}) \times \frac{p_{10}}{p_{10} + p_{11}} = p_{10},$$

ami pontosan az, amit a 3.4. egyenlet szerint várunk. A többi eset hasonlóan ellenőrizhető.

3.2. Gyakorló feladat: Alíz érméjének kitalálása

Probléma: Alíznak három érméje van, melyeket u, q, r betűkkel jelölünk, a következő valószínűségi eloszlásokkal:

$$u = \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix}, \qquad q = \begin{pmatrix} 3/4 \\ 1/4 \end{pmatrix}, \qquad r = \begin{pmatrix} 1/3 \\ 2/3 \end{pmatrix}.$$

Alíz a következő érmedobás sorozatot hajtja végre:

1. Feldobja az u érmét.
2. Az eredménytől függően feldobja az egyik másik érmét:
   (0) ha u eredménye 0, feldobja q-t;
   (1) ha u eredménye 1, feldobja r-t.

3. Alíz elmondja barátjának, Botinak a 2. lépésben feldobott érme eredményét (0 vagy 1), de nem árulja el, hogy ez q vagy r dobásából származik-e.

Ebben a helyzetben *két* valószínűségi bit van: Alíz első érmedobása és Alíz második érmedobása (ami pontosan ugyanaz, mint Boti valószínűségi bitje).

Kérdések:

1. Mi Alíz két érmedobásának valószínűségi eloszlása?
2. Mi a valószínűségi eloszlás, amikor Boti megméri a saját valószínűségi bitjét?
3. Ha adott, hogy Boti megméri a bitjét és 0 eredményt kap, melyik a valószínűbb, az hogy Alíz első érméje 0-t vagy az, hogy 1-et mutat? És mi a helyzet, ha Boti eredménye 1?

3.1.5. A SWAP művelet

Most, hogy tudjuk, hogyan manipuláljunk egyedi valószínűségi biteket, szeretnénk műveleteket alkalmazni egyszerre több biten is. Az egyik legegyszerűbb ilyen művelet a SWAP művelet, amely felcseréli a két bitet:

$$\begin{aligned}\text{SWAP}\,[00] &= [00],\\ \text{SWAP}\,[01] &= [10],\\ \text{SWAP}\,[10] &= [01],\\ \text{SWAP}\,[11] &= [11].\end{aligned}$$

A SWAP gyakorlatilag a $[01]$ és $[10]$ karakterláncok cseréjét jelenti, míg a másik két karakterláncot érintetlenül hagyja. A fenti egyenletek tömörebben így írhatók:

$$\text{SWAP}\,[a,b] = [b,a], \tag{3.17}$$

minden $a, b \in \{0,1\}$ esetén, ahol vesszőt használunk a két bit elválasztására. Szokás szerint lineárisan kiterjeszthetjük a SWAP-ot determinisztikus bitekről valószínűségi bitekre:

$$\begin{aligned}&\text{SWAP}\left(p_{00}[00] + p_{01}[01] + p_{10}[10] + p_{11}[11]\right)\\ &\quad= p_{00}[00] + p_{01}[10] + p_{10}[01] + p_{11}[11]\\ &\quad= p_{00}[00] + p_{10}[01] + p_{01}[10] + p_{11}[11].\end{aligned}$$

3.3. Gyakorló feladat: SWAP a 4-vektoros jelölésben (opcionális)

Írd le a SWAP hatását két valószínűségi bitre a 4-vektoros jelölésben.

3.1.6. Vezérelt-NOT művelet

A NOT és SWAP műveleteket együtt lehet használni az egyedi bitek megváltoztatására és átrendezésére. Azonban ezek nem okozzák a bitek egymással való kölcsönhatását. Amit szeretnénk, az egy másik, kifinomultabb művelet, amely megváltoztathatja az egyik bit értékét a másik értékétől függően. A legegyszerűbb és legfontosabb ilyen művelet a CNOT vagy a vezérelt-NOT művelet. Ez átbillenti a cél bitet, ha a vezérlő bit 1-re van állítva.

Amikor az első bit a vezérlő és a második bit a cél, ezt a műveletet $\mathrm{CNOT}_{1\to 2}$-ként írjuk le. Ebben az esetben:

$$\begin{aligned}\mathrm{CNOT}_{1\to 2}\,[00] &= [00],\\ \mathrm{CNOT}_{1\to 2}\,[01] &= [01],\\ \mathrm{CNOT}_{1\to 2}\,[10] &= [11],\\ \mathrm{CNOT}_{1\to 2}\,[11] &= [10].\end{aligned} \tag{3.18}$$

Ez gyakorlatilag a $[10]$ és $[11]$ karakterláncok cseréjét jelenti, miközben a másik két karakterláncot érintetlenül hagyja.

A $\mathrm{CNOT}_{1\to 2}$ egy másik értelmezése az összeadás: összeadja a két bitet (modulo 2) és az eredményt a második bitben tárolja. Ez nagyon tömören így foglalható össze:

$$\mathrm{CNOT}_{1\to 2}\,[a,b] = [a, a\oplus b], \tag{3.19}$$

bármely $a, b \in \{0,1\}$ esetén, ahol a "$\oplus$" jelöli a *modulo 2 összeadást*, és vesszőt használunk a két bit értékeinek elválasztására. A modulo 2 összeadás szabályai a következők:

$$0\oplus 0 = 1\oplus 1 = 0, \qquad 0\oplus 1 = 1\oplus 0 = 1. \tag{3.20}$$

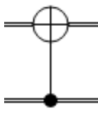
Néha a "$\oplus$" műveletet XOR-nak vagy kizáró VAGY műveletnek is nevezik, mivel az eredmény 1, ha pontosan az egyik bit 1. Mint általában, lineárisan kiterjeszthetjük a $\mathrm{CNOT}_{1\to 2}$-t determinisztikus bitekről valószínűségi bitekre.

Érdekelni fognak minket olyan vezérelt-NOT műveletek is, ahol a *második* bit a vezérlő és az *első* a cél. Analóg módon ezt a műveletet $\mathrm{CNOT}_{2\to 1}$-gyel jelöljük.

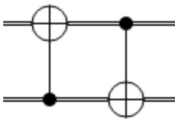
3.4. Gyakorló feladat: Vezérlő és cél felcserélése

1. Írj fel egy a 3.19. egyenlet egyenlethez hasonló képletet a $\mathrm{CNOT}_{2\to 1}$-re.
2. Hogyan lehet a $\mathrm{CNOT}_{2\to 1}$-et megvalósítani SWAP és $\mathrm{CNOT}_{1\to 2}$ használatával?

QUIRKY-ban egy vezérelt-NOT műveletet a következő kétlépéses folyamattal építhetsz fel. Először húzd a [•] dobozt az egyik vezetékre - ez jelöli ki a vezetéket vezérlő bitként. Ezután húzd a [⊕] dobozt a másik vezeték megfelelő helyére, kijelölve azt cél bitként. Megjelenik egy kapcsolat, jelezve, hogy sikeresen létrehoztál egy vezérelt-NOT műveletet. Például, ha az alsó bitet választjuk vezérlőnek és a felső bitet célnak, a következő eredményt kapjuk:



Emlékezve arra, hogy az alsó (!) bit az első bit és a felső bit a második, látjuk, hogy ez megfelel a $\mathrm{CNOT}_{1\to 2}$ műveletnek. Itt egy bonyolultabb példa, ahol először $\mathrm{CNOT}_{1\to 2}$-t, majd $\mathrm{CNOT}_{2\to 1}$-et alkalmazunk:



3.2. Házi feladat: SWAP CNOT**-okból**

Probléma: Majdnem éjfél van, de Boti még mindig a valószínűségi bit számítógépének prototípusát bütyköl-geti, amit másnap az iskolában akar bemutatni. Annyira lefoglalja a véletlenszám-generátor kalibrálása, hogy elfe-lejtette megetetni Ziggy papagáját. Hogy felhívja magára Boti figyelmét, Ziggy lelöki Boti kávéscsészéjét, és a ká-vé kiömlik a különböző műveletek aktiválására szolgáló egyedi billentyűzetére. Boti megrémül, mert a SWAP gomb már nem működik! Szerencsére a CNOT gomb még működik.

Kérdés: Hogyan valósíthatja meg Boti a SWAP műveletet csak vezérelt-NOT műveletek használatával? (Ha be akarod bizonyítani, hogy két művelet azonos, a lineáris kiterjesztés miatt elegendő ezt csak a bázisállapotokra megmutatnod.)

Ötlet: Használj három CNOT műveletet.

3.1.7. Szorzateloszlások

Most beszéljünk részletesebben a kétbites rendszer állapotairól. Tegyük fel például, hogy két valószínűségi bitünk van, $q = q_0[0] + q_1[1]$ és $r = r_0[0] + r_1[1]$. Hogyan építhetünk ebből egy kétbites rendszert? Bár nem ilyen módon fogalmaztuk meg, már tárgyaltuk ezt a kérdést az 1.1.1. alfejezetben, ahol két esemény egyidejű előfordulásának valószínűségét a két egyedi esemény valószínűségének szorzataként definiáltuk. Például annak a valószínűsége, hogy a q és r bitek $[00]$ állapotban vannak, q_0r_0. Hasonlóképpen, annak a valószínűsége, hogy $[01]$ állapotban vannak, q_0r_1. Ha figyelembe vesszük mind a négy lehetőséget, a következőt kapjuk:

$$q_0r_0[00] + q_0r_1[01] + q_1r_0[10] + q_1r_1[11]. \tag{3.21}$$

Más szóval, az együttes állapot négy p_{ab} valószínűsége

$$p_{00}[00] + p_{01}[01] + p_{10}[10] + p_{11}[11]$$

a következő képlettel adható meg:

$$p_{ab} = q_a r_b. \tag{3.22}$$

Ellenőrizheted, hogy az első bit mérésekor kapott eredmények valószínűségeit q adja meg, míg a második bit mérésekor kapott eredmények valószínűségeit r adja meg (ezt megteheted a 3.2. ábra szabályát használva és figyelembe véve, hogy $r_0 + r_1 = 1$ és $q_0 + q_1 = 1$). Azonban a kétbites állapotunknak van egy további különleges tulajdonsága: a bitek értékei független egymástól. Ez azt jelenti, hogy amikor megfigyeljük bármelyik bitet, nem nyerünk információt a másik bitről. Ezt a következő feladatban ellenőrizheted:

3.5. Gyakorló feladat: Független bitek (opcionális)

Tegyük fel, hogy megmérjük az első bitet a a 3.21. egyenletben szereplő állapotban, és jelöljük a kapott eredményt $a \in \{0, 1\}$-gyel. Mutasd meg, hogy a második bit állapota r, függetlenül az első biten mért a eredménytől. Más szóval, a két bit összekapcsolása, majd az első bit mérése egyáltalán nem befolyásolta a második bit állapotát (ahogy az elvárható)!

Vezessünk be egy jelölést, amely világosabbá teszi az állapot speciális szerkezetét. Használ-juk a '$\otimes$' jelet annak a műveletnek a jelölésére, amikor két valószínűségi bitet összekapcsolunk és egyetlen, két bitből álló rendszerként tekintünk rájuk:

$$q \otimes r = \big(q_0[0] + q_1[1]\big) \otimes \big(r_0[0] + r_1[1]\big) \tag{3.23}$$

A '$\otimes$' szimbólumot tenzorszorzatnak vagy *Kronecker-szorzatnak* nevezik. Hogyan alakíthatjuk át ezt a furcsa kifejezést egy tényleges két biten értelmezett eloszlássá, mint a 3.21. egyenletben? Először vegyük észre, hogy determinisztikus bitek esetén a '$\otimes$' művelet egyszerűen a karakterláncok összekapcsolására redukálódik. Például,

$$[0] \otimes [1] = [01]. \tag{3.24}$$

Ez értelmes, mivel egy $[0]$ állapotú bit és egy másik $[1]$ állapotú bit ugyanaz, mint két bit $[01]$ állapotban. Mint mindig, hogy kiterjesszük ezt a szabályt valószínűségi bitekre, a jó barátunktól, a linearitástól kérünk segítséget! A linearitás miatt kibonthatjuk mindkét tagot a 3.23. egyenletben, majd alkalmazhatjuk az összekapcsolási szabályt:

$$\begin{aligned}
&\big(q_0[0] + q_1[1]\big) \otimes \big(r_0[0] + r_1[1]\big) \\
&= q_0 r_0 \big([0] \otimes [0]\big) + q_0 r_1 \big([0] \otimes [1]\big) + q_1 r_0 \big([1] \otimes [0]\big) + q_1 r_1 \big([1] \otimes [1]\big) \\
&= q_0 r_0 [00] + q_0 r_1 [01] + q_1 r_0 [10] + q_1 r_1 [11].
\end{aligned}$$

Vegyük észre, hogy visszakaptuk a 3.21. egyenlet eloszlását. Más szóval, a következő azonosságunk van (3.23) és (3.21) között:

$$\big(q_0[0] + q_1[1]\big) \otimes \big(r_0[0] + r_1[1]\big) = q_0 r_0 [00] + q_0 r_1 [01] + q_1 r_0 [10] + q_1 r_1 [11]. \tag{3.25}$$

Ez azt jelenti, hogy ahogyan definiáltuk a '$\otimes$' tenzorszorzat műveletet, az valóban konzisztens a korábbi érvelésünkkel, miszerint egy kétbites rendszer valószínűségi eloszlását az egyedi bitek valószínűségeinek szorzásával kapjuk, lásd 3.22. egyenlet.

Vegyük észre, hogy a 3.25. egyenlet nagyon hasonlít az összeadás és szorzás disztributív törvényére:

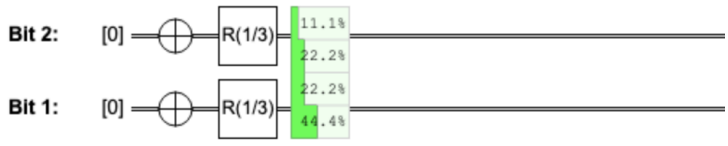
$$(a + b)(c + d) = ac + ad + bc + bd.$$

Az egyetlen különbség az, hogy számok helyett vektoraink vannak, és szorzás helyett az összefűzési szabályt $[a] \otimes [b] = [a, b]$ használjuk. Az összefűzés és a szorzás közötti fő különbség az, hogy az elemek sorrendje fontos az összefűzésnél. Általában $[a, b] \neq [b, a]$, mivel $[a]$ és $[b]$ összefűzése nem ugyanaz, mint $[b]$ és $[a]$ összefűzése. Egyébként ellenőrizheted, hogy vektorjelölést használva a tenzorszorzatot a következőképpen írhatod:

$$\begin{pmatrix} q_0 \\ q_1 \end{pmatrix} \otimes \begin{pmatrix} r_0 \\ r_1 \end{pmatrix} = \begin{pmatrix} q_0 \begin{pmatrix} r_0 \\ r_1 \end{pmatrix} \\ q_1 \begin{pmatrix} r_0 \\ r_1 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} q_0 r_0 \\ q_0 r_1 \\ q_1 r_0 \\ q_1 r_1 \end{pmatrix},$$

ahol a második kifejezés egy blokk-vektor, amelynek mindkét eleme vektor.

A tenzorszorzat gyors módot ad arra, hogy megértsük, mi történik a 3.12. egyenletben, amit itt a kényelem kedvéért megismétlünk:



Vegyük észre, hogy minden bitet függetlenül készítünk elő $\frac{1}{3}[0] + \frac{2}{3}[1]$ állapotban. Ezért mindkét bit együttes állapota:

$$\left(\frac{1}{3}[0] + \frac{2}{3}[1]\right) \otimes \left(\frac{1}{3}[0] + \frac{2}{3}[1]\right) = \frac{1}{9}[00] + \frac{2}{9}[01] + \frac{2}{9}[10] + \frac{4}{9}[11] = \begin{pmatrix} 1/9 \\ 2/9 \\ 2/9 \\ 4/9 \end{pmatrix} \approx \begin{pmatrix} 11,1\% \\ 22,2\% \\ 22,2\% \\ 44,4\% \end{pmatrix},$$

ami összhangban van QUIRKY eredményével.

3.3. Házi feladat: Tenzorszorzat

Találj két olyan q és r valószínűségi bitet, amelyekre

$$q \otimes r = 0,48[00] + 0,32[01] + 0,12[10] + 0,08[11].$$

A tenzorszorzat lehetővé teszi számunkra, hogy tömörebb képleteket írjunk a lokális műveletekre is. Nevezetesen, ha M egy egy biten végzett művelet, akkor

$$M_1([a] \otimes [b]) = M[a] \otimes [b], \qquad M_2([a] \otimes [b]) = [a] \otimes M[b]. \tag{3.26}$$

Ez összhangban van a 3.1.2. alfejezetben tárgyalt képletekkel.

Mivel a fent leírt kétbites eloszlásokat két egybites eloszlás szorzataként kapjuk, $p = q \otimes r$, ezeket szorzatállapotoknak vagy szorzateloszlásoknak nevezzük. Ahogy a 3.5. gyakorló feladatban láttad, a szorzateloszlások természetesen modelleznek olyan helyzeteket, amikor két bit függetlenül jön létre, például két érme feldobásakor. De vajon minden kétbites eloszlás szorzateloszlás? Érdekes módon ez nem így van, ahogy azt a következő részben látni fogjuk.

3.1.8. Korrelált eloszlások

Néhány kétbites eloszlás *nem* szorzateloszlás – nem írhatók fel a 3.21. egyenlet vagy a 3.23. egyenletben szereplő formában, függetlenül attól, milyen q_a és r_b értékeket választunk. Azt mondjuk, hogy egy eloszlás korrelált, ha nem szorzateloszlás. Egy példa korrelált eloszlásra:

$$\frac{1}{2}[00] + \frac{1}{2}[11]. \tag{3.27}$$

Hogy belássuk, ez nem szorzateloszlás, tegyük fel, hogy megmérjük az egyik bitet. Az eredmény a teljesen véletlenszerű lesz, azaz 0 vagy 1, mindkettő 50-50% valószínűséggel. Azonban amint ismerjük az a eredményt, a megmaradt bit állapota teljesen meghatározott – mérése 100% valószínűséggel ugyanazt az eredményt adná. Tehát a megmaradt bit állapota $b = a$, ami függ a másik biten végzett mérés a eredményétől. A 3.5. gyakorló feladatban láttuk, hogy ez nem lehet igaz szorzateloszlásra. Így bebizonyítottuk, hogy a 3.27. egyenlet egy korrelált állapotot ír le. Valójában a két bit *tökéletesen* korrelált, mivel mindkét mérési eredmény teljesen véletlenszerű, de mindig azonos ($a = b$). E tulajdonság miatt azt mondjuk, hogy a 3.27. egyenlet egy tökéletesen korrelált véletlen bit párt ír le.

A korrelált eloszlások természetesen valamilyen kölcsönhatás révén jönnek létre. Például tegyük fel, hogy feldobsz egy szabályos érmét, leírod az eredményt egy papírra, elrejted a papírt egy borítékban, és átadod a borítékot egy barátodnak. A barátod szemszögéből (aki ismeri az előkészítési eljárást, de nem tudja, mi van írva a borítékban lévő papírra), az érméd (1. bit) és a borítékban lévő papír (2. bit) állapotát a következőképpen írhatjuk le:

$$\frac{1}{2} \otimes \boxed{\text{fej}} + \frac{1}{2} \otimes \boxed{\text{írás}}$$

ami nem más, mint egy érdekes módja a 3.27. egyenletben szereplő kétbites állapot felírásának.

Hogyan hozhatunk létre korrelált állapotokat QUIRKY-ban? A lokális műveletek önmagukban nem elegendőek, mivel azok csak szorzatállapotokat tudnak létrehozni. Azonban használhatjuk a vezérelt-NOT műveletet, hogy a két bitet kölcsönhatásba hozzuk, mint a következő QUIRKY számításban:

Miért működik ez? Ezt a következő feladatban kiderítheted:

3.6. Gyakorló feladat: Tökéletesen korrelált véletlen bitek létrehozása

Magyarázd el, miért készíti el a fenti QUIRKY számítás a $\frac{1}{2}[00] + \frac{1}{2}[11]$ állapotot.

Hogy ellenőrizzük, egy tetszőleges kétbites eloszlás

$$p = p_{00}[00] + p_{01}[01] + p_{10}[10] + p_{11}[11]$$

szorzatállapotnak vagy korrelált állapotnak felel-e meg, egyszerűen kiszámolhatjuk a következő mennyiséget:

$$\Delta(p) = p_{00}p_{11} - p_{01}p_{10}. \tag{3.28}$$

Ha $\Delta(p) = 0$, akkor p szorzateloszlás; egyébként korrelált. Például a 3.27. egyenlet egyenletben szereplő állapotra azt találjuk, hogy

$$\Delta\left(\frac{1}{2}[00] + \frac{1}{2}[11]\right) = \frac{1}{2}\cdot\frac{1}{2} - 0\cdot 0 = \frac{1}{4} \neq 0,$$

ami megerősíti, hogy valóban korrelált és nem szorzatállapot. Ha kíváncsi vagy, miért működik ez az egyszerű feltétel, olvashatod az alábbi magyarázatot. De mivel ez nem elengedhetetlen a többi rész megértéséhez, nyugodtan ki is hagyhatod.

Annak bizonyításához, hogy $\Delta(p) = 0$ egyenértékű azzal, hogy p szorzatállapot, két dolgot kell megmutatnunk. Először mutassuk meg, hogy ha p szorzatállapot, akkor $\Delta(p) = 0$. Valóban, ha $p = q \otimes r$, akkor

$$\Delta(p) = q_0r_0q_1r_1 - q_0r_1q_1r_0 = 0.$$

Mi a helyzet a fordított állítással – lehetséges-e, hogy $\Delta(p) = 0$, még akkor is, ha p nem szorzatállapot? Kiderül, hogy ez nem lehetséges! Ennek bizonyításához tegyük fel, hogy $\Delta(p) = 0$, és mutassuk meg, hogy $p = q \otimes r$, ahol q és r valószínűségi bitek. Válasszuk ezt a két bitet a következőképpen:

$$\begin{aligned} q &= q_0[0] + q_1[1] = (p_{00} + p_{01})[0] + (p_{10} + p_{11})[1], \\ r &= r_0[0] + r_1[1] = (p_{00} + p_{10})[0] + (p_{01} + p_{11})[1]. \end{aligned} \tag{3.29}$$

Vegyük észre a 3.3. ábra ábráról, hogy q és r egyszerűen azok az eredményeloszlások, amiket akkor kapnánk, ha az első, illetve a második bitet mérnénk. Most ellenőrizzük, hogy q és r ezen választása valóban a p állapotot eredményezi:

$$\begin{aligned} q \otimes r &= \big((p_{00} + p_{01})[0] + (p_{10} + p_{11})[1]\big) \otimes \big((p_{00} + p_{10})[0] + (p_{01} + p_{11})[1]\big) \\ &= (p_{00} + p_{01})(p_{00} + p_{10})[00] + \cdots \\ &= (p_{00}p_{00} + p_{00}p_{10} + p_{01}p_{00} + \boldsymbol{p_{01}p_{10}})[00] + \cdots \\ &= (p_{00}p_{00} + p_{00}p_{10} + p_{01}p_{00} + \boldsymbol{p_{00}p_{11}})[00] + \cdots \\ &= p_{00}(p_{00} + p_{10} + p_{01} + p_{11})[00] + \cdots \\ &= p_{00}[00] + \cdots \\ &= p. \end{aligned}$$

Itt először a 3.25. egyenletet használtuk, aztán kibontottuk a szorzatot, majd $\Delta(p) = 0$-t használtuk, hogy $\boldsymbol{p_{01}p_{10}}$-t $\boldsymbol{p_{00}p_{11}}$-re cseréljük (lásd 3.28. egyenlet), és végül egyszerűsítettük $p_{00} + p_{01} + p_{10} + p_{11} = 1$-et, mivel p valószínűségi eloszlás. A három tagot, amelyeket "..."-tal rövidítettünk, hasonlóan lehet kezelni. Ki tudod tölteni a kimaradt részleteket önállóan és ellenőrizni az eredményt?

Korábban láttuk a 3.5. gyakorló feladatban, hogy p nem lehet szorzatállapot, ha az első bit mérése befolyásolja a második bit állapotát. Valójában a fordítottja is igaz, ahogy azt a következő házi feladatban megmutathatod.

3.4. Házi feladat: A függetlenséghez szorzat struktúra kell (opcionális)

Tegyük fel, hogy p egy tetszőleges kétbites valószínűségi eloszlás, amelyben a második bit állapota nem függ az első bit mérésének eredményétől. Mutasd meg, hogy egy ilyen p szorzateloszlás. Ezt két lépésben teheted meg:

1. Az első biten végzett mérés eredménye 0 vagy 1 lehet. Használd a 3.3. ábrát, hogy összehasonlítsd a második bit megmaradó állapotát ebben a két esetben, és mutasd meg a következő azonosságokat:

$$\frac{p_{00}}{p_{00}+p_{01}} = \frac{p_{10}}{p_{10}+p_{11}}, \qquad \frac{p_{01}}{p_{00}+p_{01}} = \frac{p_{11}}{p_{10}+p_{11}}.$$

2. Használd ezeket az egyenleteket annak megmutatására, hogy $\Delta(p) = 0$ a 3.28. egyenletből.

3.2. Két kvantumbit

A két kvantumbit leírásának módja nagyon hasonló ahhoz, ahogy két valószínűségi bitet írtunk le. A fő különbség az, hogy amplitúdókat használunk valószínűségek helyett, ami azt jelenti, hogy ezek negatívak lehetnek és más módon normalizáltak (lásd 2.1.1. alfejezet). Egy általános kétqubites kvantumállapot a következőképpen néz ki:

$$|\psi\rangle = \psi_{00}|00\rangle + \psi_{01}|01\rangle + \psi_{10}|10\rangle + \psi_{11}|11\rangle \tag{3.30}$$

ahol $\psi_{ij} \in [-1,1]$ és

$$\psi_{00}^2 + \psi_{01}^2 + \psi_{10}^2 + \psi_{11}^2 = 1.$$

$|a,b\rangle$-t írunk $[a,b]$ helyett, hogy világos legyen, most kvantumbitekkel és nem valószínűségi bitekkel foglalkozunk. Ahogy a 3.3. egyenletben tettük valószínűségi bitek esetén, azonosíthatjuk a négy $|a,b\rangle$ bázisállapotot a négy bázisvektorral:

$$|00\rangle = \begin{pmatrix}1\\0\\0\\0\end{pmatrix}, \qquad |01\rangle = \begin{pmatrix}0\\1\\0\\0\end{pmatrix}, \qquad |10\rangle = \begin{pmatrix}0\\0\\1\\0\end{pmatrix}, \qquad |11\rangle = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}. \tag{3.31}$$

Hasonlóan a 3.1. egyenlethez, a 3.30. egyenletben szereplő általános kétqubites állapot egy 4D-vektorral írható le:

$$\begin{pmatrix}\psi_{00}\\ \psi_{01}\\ \psi_{10}\\ \psi_{11}\end{pmatrix}.$$

Azonban hamar körülményessé válik az ilyen vektorok kezelése. Ezeket már nem is olyan könnyű vizualizálni, ezért főleg a 3.30. egyenletben szereplő $|a,b\rangle$ jelöléssel fogunk dolgozni.

Ennek a jelölésnek egy másik előnye, hogy sokkal könnyebb kvantumrendszereket kombinálni vele. Emlékezzünk vissza a 3.1.7. alfejezetből, hogy a "$\otimes$" tenzorszorzat műveletet használtuk két független valószínűségi bit egy közös kétbites rendszerré való kombinálására. Ugyanez a művelet (csak $[a]$ helyett $|a\rangle$-val) működik qubitekre is. Különösen, ahogy a valószínűségi bitek bázisvektorait kombináltuk a 3.24. egyenletben, ugyanezt megtehetjük qubitekkel is:

$$|0\rangle \otimes |0\rangle = |00\rangle, \qquad |0\rangle \otimes |1\rangle = |01\rangle, \qquad |1\rangle \otimes |0\rangle = |10\rangle, \qquad |1\rangle \otimes |1\rangle = |11\rangle. \tag{3.32}$$

Vedd észre, hogy ez egyszerűen a bitsorozatok összekapcsolásának felel meg. Ez egy alternatív módot ad a 3.31. egyenlet egyenletben szereplő négy kétqubites bázisvektor megtekintésére.

Kiterjeszthetjük a tenzorszorzat műveletet két tetszőleges egyqubites állapot $|\alpha\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle$ és $|\beta\rangle = \beta_0 |0\rangle + \beta_1 |1\rangle$ kombinálására a következő módon:

$$\begin{aligned} |\alpha\rangle \otimes |\beta\rangle &= (\alpha_0 |0\rangle + \alpha_1 |1\rangle) \otimes (\beta_0 |0\rangle + \beta_1 |1\rangle) \\ &= \alpha_0\beta_0 |00\rangle + \alpha_0\beta_1 |01\rangle + \alpha_1\beta_0 |10\rangle + \alpha_1\beta_1 |11\rangle . \end{aligned} \tag{3.33}$$

Az ilyen formájú kétqubites állapotokat szorzatállapotoknak nevezzük. A 3.2.3. alfejezetben tárgyaljuk majd, hogyan lehet ezeket az állapotokat kvantumműveletekkel létrehozni. Fontos megjegyezni, hogy a két valószínűségi bithez hasonlóan nem minden kétqubites állapot szorzatállapot.

3.7. Gyakorló feladat: Tenzorszorzat és szorzatállapotok

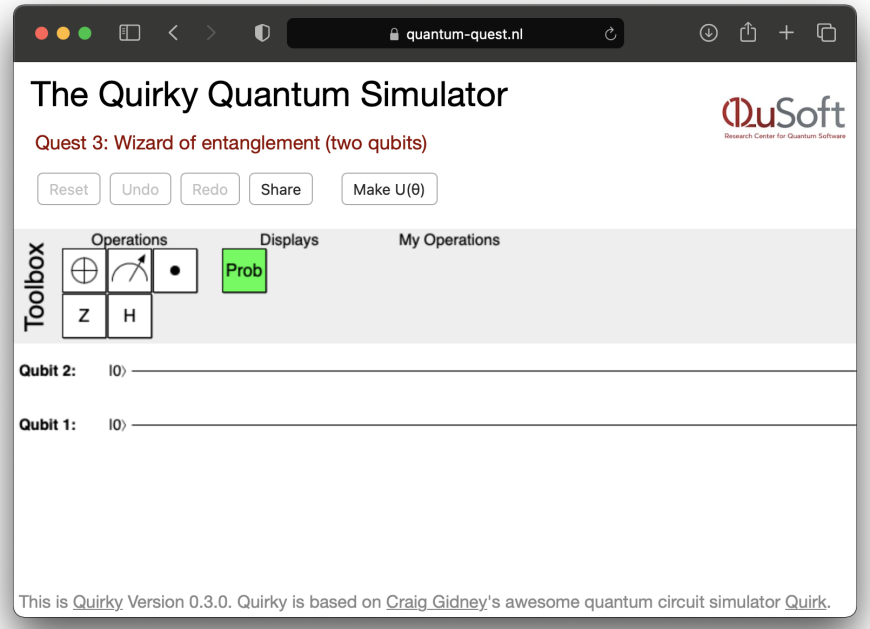
Emlékezz vissza a $|+\rangle$ és $|-\rangle$ állapotokra a 2.1. gyakorló feladatból.

1. Írd fel $|+\rangle \otimes |-\rangle$-t ugyanabban a formában, mint a 3.30. egyenletben.
2. A $\frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$ állapot szorzatállapot?

A következőkben megbeszéljük a két kvantumbit mérésének és manipulálásának szabályait. Bár ezek a szabályok teljesen analógok lesznek a két valószínűségi bit esetével, útközben néhány új és meglepő jelenséget fogunk felfedezni. Szerencsére ezen a héten QUIRKY is segít nekünk felfedezni a két kvantumbit világát! Kezdésként menj a következő oldalra:

https://www.quantum-quest.org/quirky

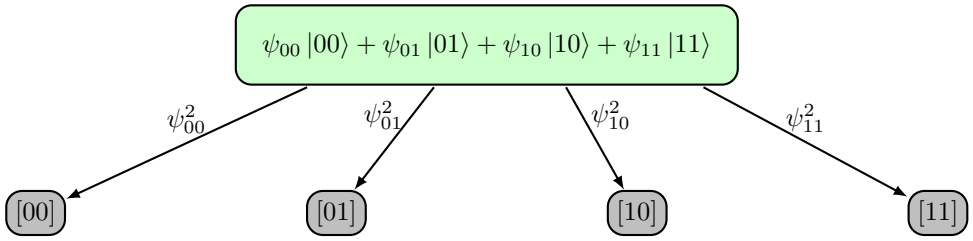
és kattints a "Quest 3"-ra, majd a "Two Qubits"-re. A böngésződ hasonlóan fog kinézni, mint a 3.4. ábrán. A múlt heti QUIRKY-hoz képest most két *kvantumbitünk* van, amelyek $|00\rangle$ állapotban vannak inicializálva. Emellett három új doboz van: $\boxed{Z}$, $\boxed{H}$, és $\boxed{\bullet}$ (és a rejtélyes doboz ismét eltűnt). Ezeket fogjuk tárgyalni a fejezet hátralevő részében.



3.4. ábra. QUIRKY a 3. küldetéshez.

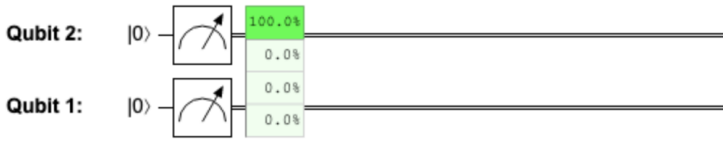
3.2.1. Két qubit mérése

A kétqubites állapotok mérése nagyon hasonlóan van definiálva, mint a 2.7. egyenletben az egyqubites állapotok esetében, azzal a különbséggel, hogy eredményként két bitet kapunk a következő valószínűségekkel:



(3.34)

Itt világoszöldet használunk a kvantumbitekhez és szürkét a mérés után kapott két determinisztikus bithez. Hogyan mérhetünk mindkét qubitet QUIRKY-ban? Egyszerűen hozzáadunk két mérést, egyet-egyet minden qubithez, és használjuk a valószínűségi kijelzőt, ahogy korábban:



3.2.2. Lokális műveletek

Ha két qubited van, egy *lokális művelet* csak az egyikre hat. Bármely egyqubites művelet használható lokális műveletként, amely a két qubit egyikére hat, ahogy azt a valószínűségi bitek esetében is tettük a 3.1.2. alfejezetben. Például, ha visszaemlékszel az egybites NOT műveletre az 1.9. egyenletből, és hogyan alakítottuk át lokális NOT műveletekké a 3.6. és 3.7. egyenletekben, pontosan ugyanezt megtehetjük az egyqubites NOT esetében is. Az eredményül kapott lokális kvantum NOT műveletek nagyon hasonlóak:

$$\begin{aligned}
&\mathrm{NOT}_1\,|00\rangle = |10\rangle\,, \quad \mathrm{NOT}_1\,|01\rangle = |11\rangle\,, \quad \mathrm{NOT}_1\,|10\rangle = |00\rangle\,, \quad \mathrm{NOT}_1\,|11\rangle = |01\rangle\,,\\
&\mathrm{NOT}_2\,|00\rangle = |01\rangle\,, \quad \mathrm{NOT}_2\,|01\rangle = |00\rangle\,, \quad \mathrm{NOT}_2\,|10\rangle = |11\rangle\,, \quad \mathrm{NOT}_2\,|11\rangle = |10\rangle\,.
\end{aligned}$$

Az egyetlen különbség az, hogy a $[ab]$ bitjelölést $|ab\rangle$ qubit jelölésre cseréltük.

Ennek alkalmazásaként tegyük fel, hogy mind a négy lehetséges kétqubites bázisállapotot fel akarjuk építeni $|00\rangle$-ból. Ez mindig megtehető NOT műveletek sorozatával:

$$|00\rangle = |00\rangle\,, \quad |01\rangle = \mathrm{NOT}_2\,|00\rangle\,, \quad |10\rangle = \mathrm{NOT}_1\,|00\rangle\,, \quad |11\rangle = \mathrm{NOT}_2\,\mathrm{NOT}_1\,|00\rangle\,.$$

Vedd észre, hogy az utolsó esetben a két NOT műveletet ellenkező sorrendben is elvégezhettük volna, mivel mindkét sorrend mindkét bit negálásának felel meg. Más szóval, $\mathrm{NOT}_2\,\mathrm{NOT}_1 = \mathrm{NOT}_1\,\mathrm{NOT}_2$.

Lokális művelet alkalmazásához QUIRKY-ban a megfelelő dobozt az első vagy a második vezetékre helyezzük. Például a következő sorozat előkészíti a $|10\rangle$ állapotot és megmutatja a kimeneti valószínűségeket mindkét qubit mérésekor:

Ez értelmes, mivel QUIRKY-ban az alsó vezeték felel meg az első qubitnek.

Ezt a legegyszerűbben a 3.33. egyenletben szereplő tenzorszorzat jelöléssel és linearitással lehet megfogalmazni. Ha U egy tetszőleges egyqubites művelet, akkor U_1-et úgy definiáljuk, mint azt a kétqubites műveletet, amely bármely $|a,b\rangle = |a\rangle \otimes |b\rangle$ bázisvektorra hat, ahol $a,b \in \{0,1\}$, a következőképpen:

$$U_1 |a,b\rangle = U|a\rangle \otimes |b\rangle . \tag{3.35}$$

Hogy világos legyen, a jobb oldal $(U|a\rangle) \otimes |b\rangle$-t jelent, azaz az $U|a\rangle$ állapot és a $|b\rangle$ állapot tenzorszorzatát. Ez intuitív, mivel egyszerűen azt jelenti, hogy U-t az első qubiten alkalmazzuk, és a második qubitet békén hagyjuk.

Hogy U_1-et tetszőleges kétqubites állapotra alkalmazzuk, ezt az előírást *linearitással* terjesztjük ki. Akárcsak az első héten, ez azt jelenti, hogy először $|\psi\rangle$-t a 3.30. egyenlet formájában kifejtjük, majd a műveletet minden bázisvektorra alkalmazzuk. Azaz,

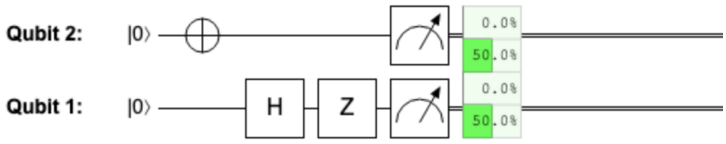
$$U_1 |\psi\rangle = \psi_{00} U_1 |00\rangle + \psi_{01} U_1 |01\rangle + \psi_{10} U_1 |10\rangle + \psi_{11} U_1 |11\rangle$$

és most a 3.35. egyenletet használhatjuk mind a négy tagra. Hasonlóképpen definiáljuk U_2-t:

$$U_2 |a,b\rangle = |a\rangle \otimes U |b\rangle \tag{3.36}$$

és kiterjesztjük linearitással. Ez analóg a 3.26. egyenletben szereplő képletekkel a közönséges bitekre.

A NOT műveleten kívül két fontos kvantumművelet, amelyet állandóan használni fogunk, a Z művelet a 2.12. egyenletből és a Hadamard művelet a 2.20. egyenletből. Mivel olyan fontosak, saját dobozt kaptak QUIRKY-ban, nevezetesen $\boxed{Z}$ és $\boxed{H}$. Például a következő műveletsorozat egy NOT-ot alkalmaz a második (!) qubiten, majd egy Hadamard-ot az első qubiten, és végül egy Z-t, megint az első qubiten:



(3.37)

Milyen állapotot kapunk így? Ennek az állapotnak a matematikai kifejezése:

$$Z_1 H_1 \mathrm{NOT}_2 |00\rangle . \tag{3.38}$$

Vedd észre, hogy ellentétben a grafikus ábrázolással (3.37), ebben a kifejezésben a $|00\rangle$ bemeneti állapot a jobb oldalon van. Ez azt eredményezi, hogy a műveletek sorrendje fordítottnak tűnik. Azonban mind eqrefeq:not-1-h-1-z-measure-measure-chance2, mind a 3.38. egyenlet egyenlet ugyanazt a folyamatot írja le – az első művelet, amit $|00\rangle$-ra alkalmazunk, az NOT_2, aztán H_1, és végül Z_1. Az egyetlen különbség (3.37) és a 3.38. egyenlet között az időirány ábrázolásának konvenciója: balról jobbra halad (3.37)-ben és jobbról balra a 3.38. egyenletben. Sajnos ez a két eltérő konvenció standard a kvantumszámítástechnikában, így nem tehetünk ellene semmit. Egyszerűen óvatosnak kell lenned, amikor QUIRKY képeket egyenletekre fordítasz és fordítva!

3.8. Gyakorló feladat: Jól számol a QUIRKY?

Számold ki a a 3.38. egyenletben szereplő kétqubites állapotot (azaz közvetlenül a mérés előtti állapotot (3.37)-ben). Számold ki a mérési eredmények valószínűségeit és hasonlítsd össze az eredményedet QUIRKY eredményével.

A múlt héten, a 2.4.3. alfejezetben tárgyaltuk, hogy bármely egyetlen qubiten végrehajtott művelet vagy egy $U(\theta)$ forgatás, vagy egy $V(\theta) = \text{NOT}\, U(\theta)$ tükrözés. Mivel tetszőleges forgatásokat tudunk létrehozni QUIRKY-ban (lásd a múlt heti előadás jegyzetének a 2.4.1. alfejezetét), ezért tetszőleges lokális műveleteket alkalmazhatunk bármelyik qubiten QUIRKY használatával.

Valójában a 3.35. egyenlet-3.36. egyenletek szabályai nemcsak bázisvektorokra működnek, hanem tetszőleges szorzatállapotokra is. Azaz, ha $|\alpha\rangle$ és $|\beta\rangle$ tetszőleges egyqubites állapotok, akkor

$$U_1 \left(|\alpha\rangle \otimes |\beta\rangle\right) = U|\alpha\rangle \otimes |\beta\rangle\,, \tag{3.39}$$
$$U_2 \left(|\alpha\rangle \otimes |\beta\rangle\right) = |\alpha\rangle \otimes U|\beta\rangle\,. \tag{3.40}$$

3.9. Gyakorló feladat: Lokális műveletek szorzatállapotokon (opcionális)

Ellenőrizni tudod a 3.39. és a 3.40. egyenletet?

3.2.3. Párhuzamos műveletek

Ha egy műveletet az első qubiten és egy másikat a második qubiten hajtunk végre, akkor e két művelet sorrendje nem számít. Azaz, ha U és V tetszőleges egyqubites műveletek, akkor

$$U_1 V_2 = V_2 U_1. \tag{3.41}$$

Ellenőrizhetjük ezt az intuitív tényt a 3.39. és 3.40. egyenletek felhasználásával. Minden $|a,b\rangle$ bázisállapotra, ahol $a, b \in \{0,1\}$,

$$U_1 V_2 |a,b\rangle = U_1 \left(|a\rangle \otimes V|b\rangle\right) = U|a\rangle \otimes V|b\rangle = V_2 \left(U|a\rangle \otimes |b\rangle\right) = V_2 U_1 |a,b\rangle\,,$$

így a 3.41. egyenlet következik a linearitásból.

Valójában, mivel a két művelet különböző qubiteken hat, akár párhuzamosan is végrehajthatnánk őket! Ez egy új jelölés bevezetését sugallja a 3.41. egyenletben szereplő műveletre:

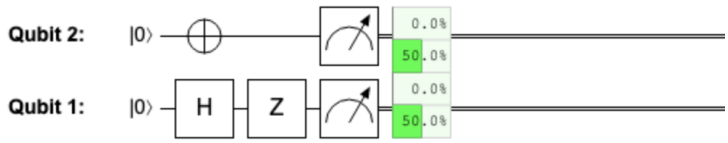
$$U \otimes V.$$

Újra használjuk a tenzorszorzat szimbólumot, amelyet eredetileg két független egyqubites állapot kétqubites állapottá való kombinálására vezettünk be. Ugyanez a jelentés kiterjed a kvantumműveletekre is: $U \otimes V$ jelöli azt a kombinált műveletet, amely U és V alkalmazásából áll egy nagyobb rendszer két különböző alrendszerére. Ez a jelölés különösen kényelmes, mert szépen összejátszik az eredeti tenzorszorzattal állapotokra:

$$(U \otimes V)(|\alpha\rangle \otimes |\beta\rangle) = U|\alpha\rangle \otimes V|\beta\rangle\,. \tag{3.42}$$

Ez az egyenlet egyszerűen azt mondja, hogy ha két független állapotod van, és egy olyan műveletet alkalmazol, amely függetlenül hat mindkét állapotra, akkor végül csak a két művelet mindegyikét alkalmazod a megfelelő állapotra. $U \otimes V$-t két kvantumművelet tenzorszorzatának vagy párhuzamos műveletnek fogjuk nevezni.

QUIRKY lehetővé teszi számunkra, hogy párhuzamosan alkalmazzunk lokális kvantummű-veleteket. Például a 3.37. egyenletben szereplő műveletsort írhattuk volna így:

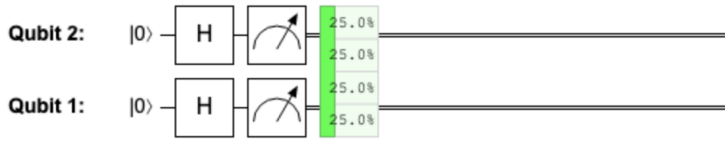


ahol a NOT művelet és a Hadamard művelet most párhuzamosan van alkalmazva.

Nézzünk egy másik példát. Mi történik, ha H-t mindkét qubiten alkalmazzuk? Ezt a műveletet $H \otimes H$-val jelöljük, és a 2.20. és 3.42. egyenletek szerint a következőképpen működik:

$$\begin{aligned}(H \otimes H)\,|00\rangle &= (H\,|0\rangle) \otimes (H\,|0\rangle) \\ &= \left(\frac{1}{\sqrt{2}}\,|0\rangle + \frac{1}{\sqrt{2}}\,|1\rangle\right) \otimes \left(\frac{1}{\sqrt{2}}\,|0\rangle + \frac{1}{\sqrt{2}}\,|1\rangle\right) \\ &= \frac{1}{2}\,|0\rangle \otimes |0\rangle + \frac{1}{2}\,|0\rangle \otimes |1\rangle + \frac{1}{2}\,|1\rangle \otimes |0\rangle + \frac{1}{2}\,|1\rangle \otimes |1\rangle \\ &= \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle).\end{aligned}$$

Az eredményül kapott állapotot két qubit egyenletes szuperpozíciójának nevezzük, mivel egyenlő amplitúdóval tartalmazza mind a négy kétqubites bázisvektort. Ha mérést végzünk ezen az állapoton, mind a négy eredményt egyenlő valószínűséggel kapjuk. Ezt könnyen ellenőrizhetjük QUIRKY segítségével:



3.10. Gyakorló feladat: Szorzatállapot létrehozása

1. Írd fel a $\frac{1}{2}\,(|00\rangle + |01\rangle - |10\rangle - |11\rangle)$ állapotot két egyqubites állapot tenzorszorzataként.
2. Hogyan hozhatod létre ezt az állapotot lokális műveletek sorozatának $|00\rangle$-ra való alkalmazásával?
3. Implementáld a 2. lépésben meghatározott műveletsorozatot QUIRKY-ben.

A 3.42. egyenlet mutatja, hogy ha párhuzamos műveletet alkalmazunk egy szorzatállapotra, akkor egy másik szorzatállapotot kapunk. Valójában $|00\rangle$-ból kiindulva bármilyen szorzatállapotot megkaphatunk ilyen módon. Ez intuitív értelmezést ad a szorzatállapotoknak: pontosan azok az állapotok, amelyeket tetszőleges párhuzamos művelet alkalmazásával kaphatunk két, $|00\rangle$ állapotban inicializált qubiten.

3.5. Házi feladat: Szorzatállapotok párhuzamos műveletekből

Mutasd meg, hogy minden szorzatállapot előállítható $|00\rangle$-ra alkalmazott párhuzamos kvantumforgatásokkal (azaz $U(\theta) \otimes U(\phi)$ formájú művelettel). Emlékezz a 2.13. egyenletből, hogy $U(\theta)$ jelöli a θ szögű forgatást.

Ötlet: Emlékezz a 2.5. egyenletből, hogy a legáltalánosabb egyqubites állapot $|\psi(\theta)\rangle$ formájú.

A lokális műveletek tárgyalását néhány általános megjegyzéssel zárjuk. Először, nem nehéz ellenőrizni, hogy

$$(U \otimes V)(U' \otimes V') = UU' \otimes VV', \tag{3.43}$$

bármely négy egyqubites műveletre U, U', V, V'. Tudsz rajzolni egy képet, ami szemlélteti, mi történik itt?

Most felhasználhatjuk az I identitásműveletet a 2.18. egyenletből, amely $I|\psi\rangle = |\psi\rangle$-ként működik (ez triviálisan kiterjeszthető I_1-re és I_2-re két qubiten). Önmagában elég haszontalan, de kényelmesen használható a tenzorszorzat jelölésben. Például lehetővé teszi, hogy így írjuk:

$$U_1 = U \otimes I \qquad \text{és} \qquad V_2 = I \otimes V,$$

ami nagyon világossá teszi, hogy például U_1 az U-műveletet alkalmazza az első qubiten és semmit a második qubiten. Például az $U_1 V_2 = V_2 U_1$ azonosság a 3.41. egyenletből most így alakul:

$$(U \otimes I)(I \otimes V) = U \otimes V = (I \otimes V)(U \otimes I) \tag{3.44}$$

ami meglehetősen intuitív.

Talán azon tűnődsz, hogy a műveletek sorrendje számít-e egyáltalán, ha *ugyanarra* a qubitra alkalmazzuk őket? Ha két forgatást tekintünk, akkor a 2.15. egyenletből következik, hogy a sorrend nem fontos, mivel

$$U(\theta)U(\theta') = U(\theta + \theta') = U(\theta' + \theta) = U(\theta')U(\theta).$$

Azonban ha U és V két tetszőleges egyqubites művelet (különösen, ha az egyikük tükrözés), akkor összetételük általában függ a sorrendtől (lásd 3.11. gyakorló feladat alább). Azaz,

$$UV \neq VU.$$

Ez a probléma természetesen akkor is fennáll, ha van egy másik qubit is, de még mindig mindkét műveletet ugyanarra a qubitra alkalmazzuk. Például,

$$(U \otimes I)(V \otimes I) = UV \otimes I \neq VU \otimes I = (V \otimes I)(U \otimes I). \tag{3.45}$$

Ezt így is írhatjuk: $U_1 V_1 \neq V_1 U_1$ (és hasonlóan, amikor mindkét műveletet a második qubiten alkalmazzuk).

A 3.44, 3.45. egyenletek közötti különbség intuitív megértéséhez képzeld el, hogy U jelentése "zoknit húzni" és V jelentése "cipőt húzni". Nyilvánvaló, hogy amikor U-t és V-t ugyanarra a lábra alkalmazzák, különböző eredményeket kapsz a sorrendtől függően! Azonban ha U-t és V-t különböző lábakra alkalmazod (pl. $U \otimes I$-t és $I \otimes V$-t használsz), akkor ugyanazt az eredményt kapod, függetlenül a két művelet sorrendjétől. Mindenesetre, ha megfelelően akarsz öltözni, először $(U \otimes U)$-t, majd $(V \otimes V)$-t kell alkalmaznod.

3.11. Gyakorló feladat: A sorrend fontos

Mutasd meg, hogy $HZ \neq ZH$.

3.2.4. Vezérelt műveletek

A szorzatállapotokon túllépéshez szükségünk van egy olyan műveletre, amely lehetővé teszi a két kvantumbit kölcsönhatását. Akárcsak korábban (valószínűségi bitekre a 2.18. egyenletben), ehhez egy *vezérelt-NOT* műveletet fogunk használni, amelyet teljesen analóg módon definiálunk a 3.18. és 3.19. egyenletekhez:

$$\begin{aligned} \text{CNOT}_{1\to 2}\,|00\rangle &= |00\rangle\,, \\ \text{CNOT}_{1\to 2}\,|01\rangle &= |01\rangle\,, \\ \text{CNOT}_{1\to 2}\,|10\rangle &= |11\rangle\,, \\ \text{CNOT}_{1\to 2}\,|11\rangle &= |10\rangle\,, \end{aligned} \tag{3.46}$$

vagy tömörebben,

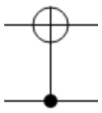
$$\text{CNOT}_{1\to 2}\,|a, b\rangle = |a, a \oplus b\rangle\,. \tag{3.47}$$

Tehát a bázisállapotokon a $\mathrm{CNOT}_{1\to 2}$ művelet átbillenti a második qubitet az első qubit értékétől függően. Definiálhatunk egy $\mathrm{CNOT}_{2\to 1}$ műveletet is, amely a második qubitet használja vezérlőként és az elsőt célként, azaz,

$$\mathrm{CNOT}_{2\to 1}\,|a,b\rangle = |a\oplus b, b\rangle\,. \tag{3.48}$$

Mint általában, ezeket a képleteket linearitással kiterjesztjük tetszőleges kétqubites állapotokra.

QUIRKY-ban ugyanúgy építhetsz vezérelt-NOT műveletet kvantumbitekre, ahogy a közönséges bitekre megtanultad – lásd 3.1.6. , ha nem emlékszel. Például a $\mathrm{CNOT}_{1\to 2}$ művelet kvantumbitekre ugyanúgy néz ki, mint korábban:



Sok dolog, amit valószínűségi bitekre bizonyítottunk, még mindig igaz kvantumbitekre. Például a a 3.2. házi feladatra adott megoldásod éppúgy lehetővé teszi két kvantumbit felcserélését! Egy másik példa erre az a tény, hogy ugyanazt a vezérelt-NOT műveletet kétszer végrehajtani egyenértékű azzal, hogy semmit sem csinálunk. Például $\mathrm{CNOT}_{1\to 2}$ esetében ez azért van így, mert

$$\mathrm{CNOT}_{1\to 2}\,\mathrm{CNOT}_{1\to 2}\,|a,b\rangle = \mathrm{CNOT}_{1\to 2}\,|a,a\oplus b\rangle = |a,a\oplus a\oplus b\rangle = |a,b\rangle$$

mivel $a\oplus a = 0$ bármely $a\in\{0,1\}$ esetén. Következésképpen a vezérelt-NOT művelet önmaga inverze:

$$\mathrm{CNOT}_{1\to 2}^{-1} = \mathrm{CNOT}_{1\to 2} \tag{3.49}$$

ahol M^{-1} az M művelet inverzét jelöli (lásd 2.4.2. alfejezet).

Ha egy kicsit játszadozol QUIRKY-val, észrevehetted, hogy kombinálhatod a $\boxed{\bullet}$-t tetszőleges egyqubites művelettel, nem csak a NOT művelettel. Valóban, definiálhatunk egy vezérelt-U műveletet bármely U egyqubites műveletre. Ezeket $CU_{1\to 2}$ és $CU_{2\to 1}$ jelöli, attól függően, melyik qubit a vezérlő és melyik a cél. Például $CU_{1\to 2}$ a következőképpen van definiálva a négy bázisállapoton:

$$\begin{aligned}
CU_{1\to 2}\,|00\rangle &= |0\rangle\otimes|0\rangle\,,\\
CU_{1\to 2}\,|01\rangle &= |0\rangle\otimes|1\rangle\,,\\
CU_{1\to 2}\,|10\rangle &= |1\rangle\otimes U\,|0\rangle\,,\\
CU_{1\to 2}\,|11\rangle &= |1\rangle\otimes U\,|1\rangle\,.
\end{aligned}$$

Gyorsan ellenőrizheted, hogy $U = \mathrm{NOT}$ esetén visszakapjuk a korábban definiált $\mathrm{CNOT}_{1\to 2}$-t.

3.2.5. Összefonódott állapotok

A 3.33. egyenletben a tenzorszorzatot használtuk egy kétqubites állapot felépítésére két egyqubites állapotból. A 3.2.3. alfejezetben láttuk, hogy ezek a *szorzatállapotok* pontosan azok az állapotok, amelyeket lokális kvantumműveletek $|00\rangle = |0\rangle\otimes|0\rangle$-ra való alkalmazásával lehet létrehozni (amely maga is szorzatállapot). Léteznek azonban olyan kétqubites állapotok is, amelyek *nem* szorzatállapotok. Ezeket az állapotokat összefonódottnak nevezzük, és mint látni fogjuk, nagyon fontosak a kvantuminformatikában.

Hogyan tudjuk meghatározni, hogy egy állapot szorzatállapot-e vagy sem? Bár a kvantumállapotokat amplitúdók és nem valószínűségek határozzák meg, ugyanazt a módszert használhatjuk, amit a 3.1.8. alfejezetben használtunk annak eldöntésére, hogy egy valószínűségi

eloszlás korrelált-e. Adott egy kétqubites állapot (3.30) formában, először a 3.28. egyenlet segítségével kiszámoljuk:

$$\Delta(|\psi\rangle) = \psi_{00}\psi_{11} - \psi_{01}\psi_{10}. \tag{3.50}$$

Ekkor $|\psi\rangle$ akkor és csak akkor szorzatállapot, ha $\Delta(|\psi\rangle) = 0$. Ilyen módon az összefonódott állapotok analógok a korrelált valószínűségi eloszlásokkal.

Egy egyszerű, de fontos példa összefonódott kétqubites állapotra

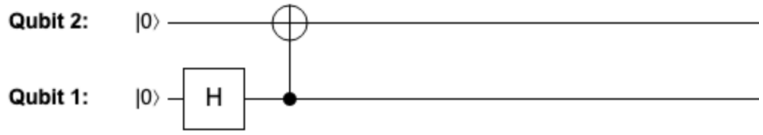
$$\left|\Phi^+\right\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle. \tag{3.51}$$

Ez az állapot analóg a tökéletesen korrelált véletlen bitpárral a 3.27. egyenletből. Összefonódott, mivel

$$\Delta(\left|\Phi^+\right\rangle) = \frac{1}{\sqrt{2}} \cdot \frac{1}{\sqrt{2}} - 0 \cdot 0 = \frac{1}{2} \neq 0.$$

$|\Phi^+\rangle$-t két qubit maximálisan összefonódott állapotának nevezzük (bár e név oka és a sajátos jelölés ezen a ponton még nem túl világos).

Hogyan hozhatunk létre összefonódott állapotokat? Ahogy akkor tettük, amikor korrelált állapotokat akartunk létrehozni két bitből, használhatjuk a vezérelt-NOT műveletet, hogy két kvantumbitet kölcsönhatásba hozzunk. Például a következő műveletsorozat előállítja a maximálisan összefonódott $|\Phi^+\rangle$ állapotot:



Gyorsan ellenőrizzük ezt:

$$\text{CNOT}_{1\to 2}\,(H \otimes I)\,|00\rangle = \text{CNOT}_{1\to 2}\left(\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle\right) = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle.$$

Vedd észre, hogy a CNOT közvetlen alkalmazása $|00\rangle$-ra, vagy bármely más bázisállapotra, *nem* működött volna (lásd 3.46. egyenlet).

3.6. Házi feladat: Egy másik összefonódott állapot

1. Igazold, hogy a $|\psi\rangle = \frac{1}{2}|01\rangle + \frac{\sqrt{3}}{2}|10\rangle$ állapot összefonódott.

   Ötlet: Számold ki a 3.50. egyenletet.

2. Találj egy műveletsorozatot QUIRKY-ban, amely előállítja a $|\psi\rangle$ állapotot.

   Ötlet: Próbálj egy megfelelő szögű forgatást használni.

3. Mik a mérési eredmények valószínűségei, amikor $|\psi\rangle$ mindkét qubitjét méred? Használd QUIRKY-t az eredményed megerősítéséhez.

A 3.51. egyenletben szereplő maximálisan összefonódott állapot egyike egy négy állapotból álló családnak, amelyet Bell-állapotoknak nevezünk. A Bell-állapotokat a következőképpen

definiáljuk:

$$\left|\Phi^+\right\rangle = \frac{1}{\sqrt{2}}\left|00\right\rangle + \frac{1}{\sqrt{2}}\left|11\right\rangle, \tag{3.52}$$

$$\left|\Phi^-\right\rangle = \frac{1}{\sqrt{2}}\left|00\right\rangle - \frac{1}{\sqrt{2}}\left|11\right\rangle, \tag{3.53}$$

$$\left|\Psi^+\right\rangle = \frac{1}{\sqrt{2}}\left|01\right\rangle + \frac{1}{\sqrt{2}}\left|10\right\rangle, \tag{3.54}$$

$$\left|\Psi^-\right\rangle = \frac{1}{\sqrt{2}}\left|01\right\rangle - \frac{1}{\sqrt{2}}\left|10\right\rangle. \tag{3.55}$$

John Steward Bell-ről nevezték el őket, aki elsőként ismerte fel a kvantum-összefonódás figyelemre méltó tulajdonságait. Hogyan hozhatjuk létre ezt a négy Bell-állapotot? Fentebb láttuk, hogy $|\Phi^+\rangle$-t úgy kaphatjuk meg, hogy egy Hadamard és egy CNOT műveletet alkalmazunk a $|00\rangle$ bázisállapotra. Könnyen ellenőrizhető, hogy a másik három Bell-állapot hasonlóan konstruálható, azaz ugyanazt a műveletsorozatot alkalmazva a másik három bázisállapotra. Más szóval, ha definiáljuk

$$U_{\text{Bell}} = \text{CNOT}_{1\to 2}\,(H \otimes I) \tag{3.56}$$

akkor

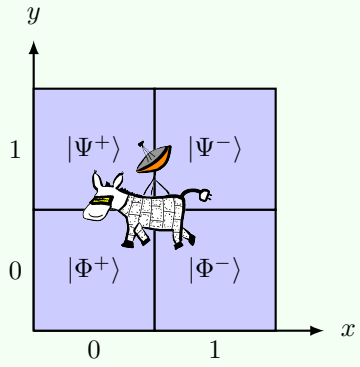
$$\left|\Phi^+\right\rangle = U_{\text{Bell}}\left|00\right\rangle, \qquad \left|\Phi^-\right\rangle = U_{\text{Bell}}\left|10\right\rangle,$$
$$\left|\Psi^+\right\rangle = U_{\text{Bell}}\left|01\right\rangle, \qquad \left|\Psi^-\right\rangle = U_{\text{Bell}}\left|11\right\rangle.$$

3.12. Gyakorló feladat: Bell-állapotok előállítása

Rajzold le hogyan hozhatóak létre a további $|\Phi^-\rangle$, $|\Psi^+\rangle$, és $|\Psi^-\rangle$ Bell-állapotok QUIRKY-ben.

3.13. Gyakorló feladat: Bell-állapotok megkülönböztetése

Alíz szamár robotja eltévedt egy felfedező küldetés során! Sürgősen tudatni akarja Alízzal a pozícióját, hogy az megmenthesse. A szamár az iskolát körülvevő négy negyed egyikében található. Hogy közölje, melyikben, a szamár egy kétqubites kvantumüzenetet $|x, y\rangle$ küld, ahol $x \in \{0, 1\}$ jelzi a helyzete x koordinátáját és $y \in \{0, 1\}$ jelzi az y koordinátáját:



Sajnos Alíz gonosz osztálytársa, Éva zavarja a jelet, és amit Alíz ehelyett kap, az a fent látható négy Bell-állapot egyike. Segíts Alíznak helyesen dekódolni a jelet és megtalálni a szamár robotját! Azaz találj egy műveletsorozatot, amely mind a négy Bell-állapotot visszaképezi a megfelelő $|x, y\rangle$ bázisállapotra.

3.2.6. Összefonódás és korrelációk

Az összefonódott állapotok és a korrelált valószínűségi eloszlások hasonlósága alapján felmerülhet a kérdés, hogyan kapcsolódik egymáshoz ez a két fogalom. Összehasonlításukhoz

beszéljünk általánosabban a kvantumállapotok és a valószínűségi eloszlások kapcsolatáról.

Kezdésként tegyük fel, hogy van egy egyqubites állapotunk $|\psi\rangle = \psi_0 |0\rangle + \psi_1 |1\rangle$, és megmérjük azt. Tudjuk a 2.2. alfejezetből, hogy eredményül egy bitet kapunk, amely vagy nulla, vagy egy, ψ_0^2 és ψ_1^2 valószínűségekkel. Ezt modellezhetjük úgy, mint egy valószínűségi eloszlást

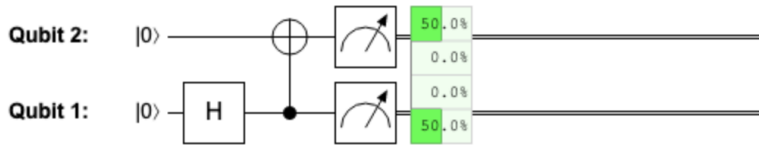
$$\psi_0^2[0] + \psi_1^2[1].$$

Intuitíven ez azt a helyzetet modellezi, amikor megmértük a qubitet, de valójában nem néztük meg az eredményt (ha megnéztük volna, nem egy valőszínűségi bitünk lenne, hanem egy determinisztikus, amely vagy nulla, vagy egy állapotban van).

Ugyanez a logika ugyanúgy működik két qubitra is. Ha megmérünk egy kétqubites állapotot $|\psi\rangle = \psi_{00} |00\rangle + \psi_{01} |01\rangle + \psi_{10} |10\rangle + \psi_{11} |11\rangle$, a mérési eredményeket leírhatjuk a következő valószínűségi eloszlással

$$p = \psi_{00}^2[00] + \psi_{01}^2[01] + \psi_{10}^2[10] + \psi_{11}^2[11]. \tag{3.57}$$

Például, ha előkészítjük és megmérjük a maximálisan összefonódott $|\Phi^+\rangle$ állapotot, egy tökéletesen korrelált véletlen bitpárt kapunk, mint a 3.27. egyenletben. Ezt ellenőrizhetjük QUIRKY segítségével:



Nyilvánvalóan ugyanez igaz, ha ehelyett a $|\Phi^-\rangle$ Bell-állapotot mérjük. (Mi a helyzet a másik két Bell-állapottal, $|\Psi^+\rangle$-szal vagy $|\Psi^-\rangle$-szal? Bármelyikük mérése tökéletesen *antikorrelált* biteket eredményez, amelyeket a $\frac{1}{2}[01] + \frac{1}{2}[10]$ valószínűségi eloszlás ír le.)

Az előző példa nem véletlen volt. Valójában a 3.57. egyenletben szereplő p valószínűségi eloszlás, amelyet egy kétqubites kvantumállapot mérésével kapunk, csak akkor lehet korrelált, ha a megfelelő $|\psi\rangle$ kvantumállapot összefonódott. Ennek belátásához tegyük fel, hogy $|\psi\rangle$ egy szorzatállapot, tehát $\Delta(|\psi\rangle) = 0$. Ekkor p egy szorzateloszlás, mivel

$$\begin{aligned} \Delta(p) &= p_{00}p_{11} - p_{01}p_{10} = \psi_{00}^2\psi_{11}^2 - \psi_{01}^2\psi_{10}^2 \\ &= (\psi_{00}\psi_{11} - \psi_{01}\psi_{10})\,(\psi_{00}\psi_{11} + \psi_{01}\psi_{10}) \\ &= \Delta(|\psi\rangle)\,(\psi_{00}\psi_{11} + \psi_{01}\psi_{10}) = 0. \end{aligned} \tag{3.58}$$

Ez bizonyítja azt az állítást, hogy a korrelált mérési eredmények összefonódást feltételeznek a mért állapotban.

Vegyük észre, hogy általában a kvantumállapotok legalább olyan hasznosak, mint a valőszínűségi bitek, mivel bármely valószínűségi eloszlás elérhető egy megfelelően választott kvantumállapot mérésével. Azaz bármely p valószínűségi eloszláshoz mindig találhatunk egy olyan $|\psi\rangle$ kvantumállapotot, amelynek mérési eredményei p szerint oszlanak el. Például egy kétbites p eloszláshoz egyszerűen választhatjuk a következő állapotot

$$|\psi\rangle = \sqrt{p_{00}}\,|00\rangle + \sqrt{p_{01}}\,|01\rangle + \sqrt{p_{10}}\,|10\rangle + \sqrt{p_{11}}\,|11\rangle\,.$$

Ez arra is rávilágít, hogy az összefonódás általában legalább olyan hasznos, mint a valószínűségi korrelációk, mivel bármely korrelált eloszlás két valőszínűségi biten előállítható valamilyen összefonódott kétqubites állapot mérésével.

3.2.7. Az összefonódás ereje

Valójában az összefonódott kvantumbitek tényleg erősebbek, mint a valószínűségi bitek! A következőkben csak ízelítőt kapunk ebből, de a következő hetekben még sok példát fogunk látni.

Illusztráljuk az összefonódás erejét egy kis történettel, amely a 3.13. gyakorló feladatra épül. Ott segítettél Alíznak dekódolni a szamár robotja pozícióját, amely négy lehetséges hely egyikén volt, két bittel jelölve: $a, b \in \{0, 1\}$. Alíznak nincs ideje felvenni a szamarat, ezért tovább akarja küldeni a pozíciót Botinak. Sajnos Alíznak kevés adatforgalma maradt a kvantum mobilcsomagjában, így csak egyetlen qubitet tud küldeni Botinak. De egy kvantumbit küldése biztosan *nem* lesz elég két bit tökéletes kommunikálásához. Tudjuk ezt, mert Boti egyetlen módja az információ kinyerésére a qubit mérése - de a mérésből csak egyetlen bitet tud meg, és aztán a qubit eredeti állapota eltűnik.

Szerencsére Alíz és Boti előrelátóak voltak, és megosztottak egy $|\Phi^+\rangle$ maximálisan összefonódott állapotot. Ez alatt azt értjük, hogy Alíz birtokolja az első qubitet, Boti pedig a másodikat. Segíthetünk-e Alíznak elküldeni a szamár helyzetét (azaz az a és b két bitet) *egyetlen* qubit továbbításával? Ez valóban lehetséges, és a kulcsfontosságú elem a következő:

3.14. Gyakorló feladat: Egy Bell-állapot átalakítása bármelyik másikba

Mutasd meg, hogy Alíz kizárólag a saját qubitjén végzett lokális műveletekkel át tudja alakítani a $|\Phi^+\rangle$ maximálisan összefonódott állapotot bármelyik másik Bell-állapottá: $|\Phi^-\rangle$, $|\Psi^+\rangle$, vagy $|\Psi^-\rangle$.

Most már világos, hogyan oldhatják meg Alíz és Boti a kihívást. Tegyük fel, hogy Alíz és Boti előzetesen megegyeztek egy leképezésben a két bit $[ab]$ négy lehetséges értéke és a négy Bell-állapot között (mondjuk, $[00]$ megfelel $|\Phi^+\rangle$-nak, $[01]$ megfelel $|\Psi^+\rangle$-nak, és így tovább). A 3.14. gyakorló feladat felhasználásával Alíz először egy műveletet alkalmaz a maximálisan összefonódott állapot rá eső részén, hogy azt a Bell-állapottá alakítsa, amely megfelel a szamár helyzetének. Ezután elküldi a kvantumbitjét Botinak. Most Botinak mindkét kvantumbit a birtokában van, és tudja, hogy azok a négy Bell-állapot egyikében vannak. Így egyszerűen alkalmazhatja ugyanazokat a műveleteket, mint a 3.13. gyakorló feladatban, és mérheti a két qubitet, hogy kiderítse a szamár robot helyzetét.

Az imént leírt eljárás szupersűrű kódolás protokollként ismert, mivel két determinisztikus bitet továbbítunk egyetlen kvantumbit küldésével (egy Alíz és Boti között megosztott maximálisan összefonódott állapot felhasználásának árán). Használhattak volna-e Alíz és Boti ugyanilyen jól valószínűségi biteket összefonódott állapot helyett a kihívás megoldására (pl. egy tökéletesen korrelált bitpárt)? Érdekes módon ez nem lehetséges. Így a szupersűrű kódolás demonstrálja, hogy a kvantum-összefonódás előnyt nyújt a kommunikációs feladatokban.

A következő házi feladatban egy másik híres helyzettel találkozhatsz, amelyben a kvantum-összefonódás egyértelmű előnyt biztosít. A probléma talán egy kicsit ijesztőnek tűnhet - de ez főleg azért van, mert nem tudtunk ellenállni annak, hogy egy kis történetet szőjünk köré. Mint mindig, ne habozz kérdezni a Discordon!

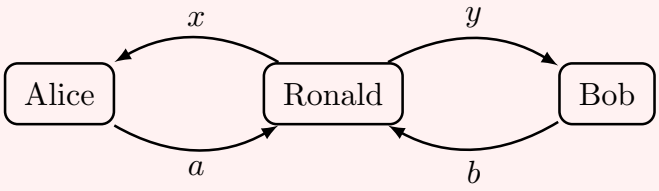
3.7. Házi feladat: Egy összefonódott játék (kihívás)

Alíz és Boti unatkoznak az órán, ezért megkérik kvantummechanika tanárukat, Rolandot, hogy adjon nekik egy kihívást jelentő feladványt. Rövid gondolkodás után Roland elmagyaráz nekik egy érdekes játékot. A játék célja, hogy Alíz és Boti a lehető legjobban együttműködjenek (nem egymás ellen játszanak). Azonban *nem* kommunikálhatnak egymással a játék során! A játék szabályai a következők:

- Kezdésként Roland titokban feldob két szabályos érmét. Megmondja Alíznak az első érmedobás eredményét (x bit) és Botinak a második érmedobás eredményét (y bit).

Ezeket input biteknek nevezzük.

- A bitek fogadása után Alíznak és Botinak egy-egy saját bitet kell válaszolniuk (a és b bitek).
- Alíz és Boti a következő feltétel mellett nyerik meg a játékot: ha $x = y = 1$, akkor nyernek, ha $a \neq b$; egyébként akkor nyernek, ha $a = b$.



x	y	Elfogdási feltétel
0	0	$a = b$
0	1	$a = b$
1	0	$a = b$
1	1	$a \neq b$

A játék kezdete előtt Alíz és Boti röviden összejönnek, hogy megbeszéljék a stratégiájukat. Először azt fontolgatják, hogy egyszerűen két függvényt alkalmaznak $f, g : \{0,1\} \to \{0,1\}$ az input bitjeiken x és y, és a válaszaikat így számolják ki: $a = f(x)$ és $b = g(y)$.

1. Mutasd meg, hogy ebben az esetben 75% valószínűséggel nyerhetik meg a játékot, de ennél nem nagyobbal.

Ezután fontolóra veszik, hogy osztott véletlenszerűséggel koordinálják a játékukat. Boti javasolja, hogy használjanak bonyolultabb f és g függvényeket egy extra bináris argumentummal, és a válaszaikat így számolják ki: $a = f(x, r)$ és $b = g(y, s)$, ahol r és s két véletlen bit, amelyeket együttesen egy kétbites valószínűségi eloszlás ír le.

2. Mutasd meg, hogy még mindig nem nyerhetnek 75%-nál nagyobb valószínűséggel, függetlenül attól, hogy mik az f és g függvények, és mi az r és s bitek együttes eloszlása.

Főhőseink kezdik felismerni, hogy Roland bizonyára egy kvantummechanikai stratégiára gondolt. Alíznak zseniális ötlete támad, és javasolja, hogy osszanak meg egy maximálisan összefonódott $|\Phi^+\rangle$ állapotot a játék kezdete előtt. Azt javasolja, hogy a bitek fogadása után ő elforgatja a qubitjét valamilyen θ_x szöggel (ami az x input bitjétől függ), és megméri, hogy megkapja az a válaszát, míg Boti ehelyett egy másik ω_y szöggel forgat (ami az y input bitjétől függ), majd méri, hogy megkapja a b válaszát.

3. Írd le az állapotot, miután Alíz és Boti elvégezték a forgatásaikat, (3.30) formájában. Erősítsd meg, hogy a játék megnyerésének valószínűsége

$$\frac{1}{4}\left(\cos^2(\theta_0 - \omega_0) + \cos^2(\theta_0 - \omega_1) + \cos^2(\theta_1 - \omega_0) + \sin^2(\theta_1 - \omega_1)\right).$$

Ötlet: Használd a trigonometrikus képleteket a 2.16. és 2.22. egyenletekből.

Alíz és Boti gyorsan rájönnek, hogy $\theta_0 = 0$, $\theta_1 = \pi/4$, és $\omega_0 = \pi/8$ jó választások, de gondban vannak az utolsó szöggel, és az idő gyorsan fogy.

4. Találj egy olyan ω_1 szöget, amellyel 75%-nál nagyobb valószínűséggel nyerik meg a játékot.

A házi feladat első két részében megmutattad, hogy bármely klasszikus biteket használó stratégia legfeljebb 75% valószínűséggel nyerheti meg a játékot. Ez egy példa egy *Bell-egyenlőtlenségre*. A fenti verzió John Clauser, Michael Horne, Abner Shimony és Richard Holt munkájára vezethető vissza, ezért a játékot, amit Alíz és Boti játszik Rolanddal, gyakran *CHSH-*

játéknak nevezik. A kvantummechanikai stratégia, amit a házi feladatban felfedeztél, *megsérti* ezt a Bell-egyenlőtlenséget. Empirikus tény, hogy a Bell-egyenlőtlenségeket a kvantummechanika valóban megsértheti. Ezt először Alain Aspect mutatta ki a 80-as években, és nemrég megerősítették nagyon szigorú feltételek mellett Ronald Hanson és csapata gyönyörű kísérletében a Delfti Műszaki Egyetemen.

Érdekes módon Alíz és Boti nemcsak jobban tudják játszani a CHSH-játékot egy megosztott maximálisan összefonódott állapot használatával, hanem valójában meg tudják győzni Rolandot, hogy kvantumtrükköket kell használniuk ahhoz, hogy ilyen jól játsszák a játékot. Mivel csak valószínűségi stratégiákkal nem nyerhetik meg a játékot 75%-nál nagyobb valószínűséggel, ha valahogyan sikerül gyakrabban nyerniük, akkor az egyetlen lehetséges magyarázat az, hogy valami erősebbet kell használniuk. Valójában további trükkökkel még arról is meg tudják győzni Rolandot, hogy a maximálisan összefonódott állapotot kell használniuk és a forgatásokat a konkrét szögekkel kell alkalmazniuk. Ez lényegében azt jelenti, hogy cáfolhatatlan módon be tudják bizonyítani Rolandnak, hogy kis kvantumszámítógépeik vannak, amelyek képesek manipulálni egyes qubiteket és megosztani összefonódást. Ez egy nagyon fontos megfigyelés, mivel lehetővé teszi, hogy a CHSH-játékot használd annak ellenőrzésére, hogy valaki tényleg épített-e kvantumszámítógépet! Valóban, ha két osztálytársad azt állítaná, hogy mindketten építettek egy-egy kvantumszámítógépet a garázsukban, egyszerűen megkérhetnéd őket, hogy játsszák együtt ezt a játékot ellened. Ha sikerül 75%-nál jelentősen nagyobb valószínűséggel nyerniük, akkor tudnád, hogy valóban nem hazudnak neked, és igazi kvantumszámítógépeik vannak. De ha nem tudnak 75%-nál többet nyerni, akkor könnyen megcáfolhatnád az állításaikat. Hát nem fantasztikus ez?

Ezeken a típusú ellenőrzési eljárásokon jelenleg aktívan dolgoznak kutatók világszerte. Ez egy nagyon fontos probléma, mert különböző nagy cégek, mint az IBM, Google és Microsoft próbálnak kvantumszámítógépet építeni. Ha azt állítják, hogy építettek egyet, valószínűleg jobban hinnél nekik, mint az osztálytársaidnak. Azonban még mindig nagyszerű, ha tényleg ellenőrizni tudod ezt!

3.3. A gyakorló feladatok megoldásai

3.2. Gyakorló feladat megoldása

1. Jelöljük Alíz két érmedobásának valószínűségi eloszlását p-vel. A 3.3. ábrát használjuk a valószínűségek kiszámításához. Tudjuk, hogy Alíz első érmedobása u szerint oszlik el, ami azt jelenti, hogy ha az első bitet mérjük, mindkét kimenetelt $1/2$ valószínűséggel kell kapnunk:

$$p_{00} + p_{01} = \frac{1}{2}, \quad p_{10} + p_{11} = \frac{1}{2}.$$

Továbbá tudjuk, hogy ha az első érmedobás eredménye 0, akkor a második bit állapotát a q érme írja le. Tehát a következőnek kell teljesülnie:

$$\frac{p_{00}[0] + p_{01}[1]}{p_{00} + p_{01}} = \frac{3}{4}[0] + \frac{1}{4}[1],$$

amiből arra következtetünk, hogy $p_{00} = \frac{3}{8}$ és $p_{01} = \frac{1}{8}$. Hasonlóképpen, ha az első érmedobás eredménye 1, akkor a második bit állapotát az r érme írja le, tehát

$$\frac{p_{10}[0] + p_{11}[1]}{p_{00} + p_{01}} = \frac{1}{3}[0] + \frac{2}{3}[1],$$

így $p_{10} = \frac{1}{6}$ és $p_{11} = \frac{2}{6}$. Összességében:

$$p = \frac{3}{8}[00] + \frac{1}{8}[01] + \frac{1}{6}[10] + \frac{2}{6}[11],$$

2. Boti bitjének értéke megegyezik Alíz második érmedobásának eredményével, így egyszerűen követjük a 3.3. ábra ábrát a második bit mérési eredményeinek valószínűségének kiszámításához. Tehát annak a valószínűsége, hogy Boti eredménye 0:

$$p_{00} + p_{10} = \frac{3}{8} + \frac{1}{6} = \frac{13}{24}$$

és annak a valószínűsége, hogy az eredménye 1: $1 - \frac{13}{24} = \frac{11}{24}$. Ezt felírhatjuk valószínűségi eloszlásként:

$$\frac{13}{24}[0] + \frac{11}{24}[1].$$

3. Ismét a 3.3. ábrát használjuk. Ha a második bitet mérjük és az eredmény 0, akkor az első bit állapotát a következő adja:

$$\frac{p_{00}[0] + p_{10}[1]}{p_{00} + p_{10}} = \frac{9}{13}[0] + \frac{4}{13}[1].$$

Így valószínűbb, hogy Alíz első érmedobása 0-t mutat.

Hasonlóképpen, ha a második bit mérésének eredménye 1, akkor az első bit állapotát a következő írja le:

$$\frac{p_{01}[0] + p_{11}[1]}{p_{01} + p_{11}} = \frac{3}{11}[0] + \frac{8}{11}[1].$$

Ebben az esetben valószínűbb, hogy Alíz első érmedobása 1-et mutat.

3.1. Gyakorló feladat megoldása

$$\mathrm{NOT}_1 \begin{pmatrix} p_{00} \\ p_{01} \\ p_{10} \\ p_{11} \end{pmatrix} = \begin{pmatrix} p_{10} \\ p_{11} \\ p_{00} \\ p_{01} \end{pmatrix}.$$

3.3. Gyakorló feladat megoldása

$$\mathrm{SWAP} \begin{pmatrix} p_{00} \\ p_{01} \\ p_{10} \\ p_{11} \end{pmatrix} = \begin{pmatrix} p_{00} \\ p_{10} \\ p_{01} \\ p_{11} \end{pmatrix}.$$

3.4. Gyakorló feladat megoldása

1. $\mathrm{CNOT}_{2\to1}[a,b] = [a \oplus b, b] = [b \oplus a, b]$.
2. Ez elvégezhető egy SWAP alkalmazásával, majd egy $\mathrm{CNOT}_{1\to2}$-vel, és végül egy újabb SWAP-pal. Valóban, a 3.17. és 3.19. egyenleteket használva,

$$\begin{aligned} &\mathrm{SWAP}(\mathrm{CNOT}_{1\to2}(\mathrm{SWAP}[a,b])) = \mathrm{SWAP}(\mathrm{CNOT}_{1\to2}[b,a]) \\ &= \mathrm{SWAP}[b, b \oplus a] = [b \oplus a, b]. \end{aligned}$$

3.5. Gyakorló feladat megoldása

A 3.3. ábra ábra használatával a második bit állapota a mérés után a következőképpen számítható ki:

$$\frac{p_{a0}[0] + p_{a1}[1]}{p_{a0} + p_{a1}} = \frac{q_a r_0[0] + q_a r_1[1]}{q_a r_0 + q_a r_1} = \frac{r_0[0] + r_1[1]}{r_0 + r_1} = r_0[0] + r_1[1] = r,$$

ahol kiejtettük q_a-t és felhasználtuk, hogy $r_0 + r_1 = 1$.

3.6. Gyakorló feladat megoldása

A vezérelt-NOT művelet előtti állapot

$$\left(\frac{1}{2}[0] + \frac{1}{2}[1]\right) \otimes [0] = \frac{1}{2}[00] + \frac{1}{2}[10].$$

A vezérelt-NOT művelet alkalmazása után a következőt kapjuk, hogy

$$\mathrm{CNOT}_{1\to2}\left(\frac{1}{2}[00] + \frac{1}{2}[10]\right) = \frac{1}{2}[00] + \frac{1}{2}[11].$$

3.7. Gyakorló feladat megoldása

1.

$$\begin{aligned}|+\rangle \otimes |-\rangle &= \left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right) \otimes \left(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle\right) \\ &= \frac{1}{2}|00\rangle - \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle - \frac{1}{2}|11\rangle .\end{aligned}$$

2.

$$\frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = |+\rangle \otimes |+\rangle .$$

Tehát ez valóban egy szorzatállapot.

3.8. Gyakorló feladat megoldása

$|00\rangle$ állapottal kezdünk. A második qubiten végzett NOT után $|01\rangle$-et kapunk. Az első qubiten alkalmazott Hadamard ezt $|+\rangle \otimes |1\rangle = \frac{1}{\sqrt{2}}|01\rangle + \frac{1}{\sqrt{2}}|11\rangle$-re alakítja, és a végső Z művelettel a következő kétqubites állapotot kapjuk közvetlenül a mérés előtt:

$$\frac{1}{\sqrt{2}}|01\rangle - \frac{1}{\sqrt{2}}|11\rangle .$$

Tehát vagy 01-et, vagy 11-et kapunk, mindkettőt 50-50% valószínűséggel.

3.9. Gyakorló feladat megoldása

Csak azt mutatjuk meg, hogyan lehet ellenőrizni a 3.39. egyenletet (a másik egyenlet teljesen analóg módon vezethető le). Ehhez írjuk fel $|\alpha\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle$ és $|\beta\rangle = \beta_0|0\rangle + \beta_1|1\rangle$ alakban. Ekkor,

$$\begin{aligned}U_1(|\alpha\rangle \otimes |\beta\rangle) &= U_1(\alpha_0\beta_0|00\rangle + \alpha_0\beta_1|01\rangle + \alpha_1\beta_0|10\rangle + \alpha_1\beta_1|11\rangle) \\ &= \alpha_0\beta_0 U_1|00\rangle + \alpha_0\beta_1 U_1|01\rangle + \alpha_1\beta_0 U_1|10\rangle + \alpha_1\beta_1 U_1|11\rangle \\ &= \alpha_0\beta_0 U|0\rangle \otimes |0\rangle + \alpha_0\beta_1 U|0\rangle \otimes |1\rangle + \alpha_1\beta_0 U|1\rangle \otimes |0\rangle + \alpha_1\beta_1 U|1\rangle \otimes |1\rangle \\ &= \alpha_0 U|0\rangle \otimes (\beta_0|0\rangle + \beta_1|1\rangle) + \alpha_1 U|1\rangle \otimes (\beta_0|0\rangle + \beta_1|1\rangle) \\ &= \alpha_0 U|0\rangle \otimes |\beta\rangle + \alpha_1 U|1\rangle \otimes |\beta\rangle \\ &= (\alpha_0 U|0\rangle + \alpha_1 U|1\rangle) \otimes |\beta\rangle \\ &= U|\alpha\rangle \otimes |\beta\rangle\end{aligned}$$

a 3.33. egyenlet, U_1 definíciója és a linearitás alapján.

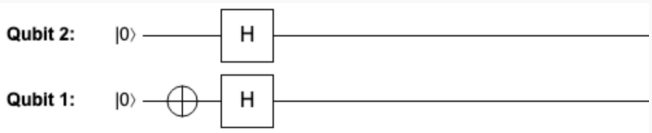
3.10. Gyakorló feladat megoldása

1. Az állapot felírható, mint

$$\frac{1}{2}\left(|00\rangle + |01\rangle - |10\rangle - |11\rangle\right) = \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right) \otimes \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right).$$

2. Ez egyenlő

$$H|1\rangle \otimes H|0\rangle = H\,\mathrm{NOT}\,|0\rangle \otimes H|0\rangle = (H\,\mathrm{NOT} \otimes H)\,|00\rangle.$$

3. 

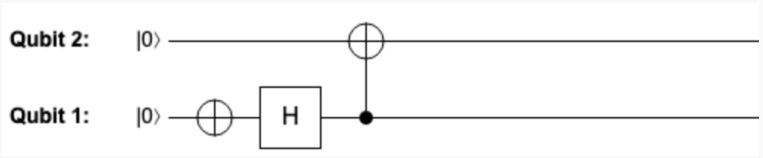
3.11. Gyakorló feladat megoldása

Annak bemutatásához, hogy $HZ \neq ZH$, elegendő ellenőrizni, hogy különböző eredményt adnak, amikor a $|0\rangle$ állapotra alkalmazzuk őket. Valóban:

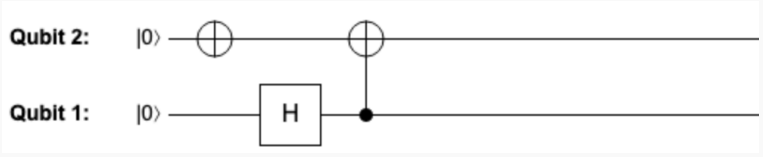
$$HZ|0\rangle = H|0\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle,$$
$$ZH|0\rangle = Z\left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right) = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle.$$

3.12. Gyakorló feladat megoldása

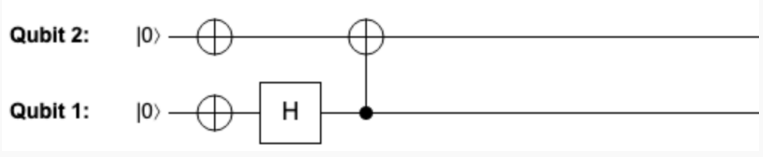
- $|\Phi^-\rangle$:



- $|\Psi^+\rangle$:



- $|\Psi^-\rangle$:



3.13. Gyakorló feladat megoldása

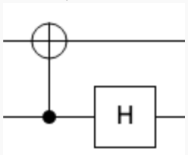
Vegyük észre, hogy ez ugyanaz, mint megfordítani a U_{Bell} műveletet a 3.56. egyenletből. Emlékezzünk, hogy $U_{\text{Bell}} = \text{CNOT}_{1\to 2}\,(H \otimes I)$. Ez egy összetett művelet, így a 2.5. gyakorló feladatból tudjuk, hogy

$$U_{\text{Bell}}^{-1} = (H \otimes I)^{-1}\,\text{CNOT}_{1\to 2}^{-1} = (H^{-1} \otimes I)\,\text{CNOT}_{1\to 2}^{-1}.$$

Emlékezzünk a 3.49. egyenletből, hogy $\text{CNOT}_{1\to 2}$ önmaga inverze, azaz $\text{CNOT}_{1\to 2}^{-1} = \text{CNOT}_{1\to 2}$. Az is igaz, hogy $H^{-1} = H$ (ez valójában minden tükrözésre igaz). Ez azt jelenti, hogy visszafordíthatjuk U_{Bell}-t ugyanazon két művelet alkalmazásával, de fordított sorrendben:

$$U_{\text{Bell}}^{-1} = (H \otimes I)\,\text{CNOT}_{1\to 2}.$$

Azaz először alkalmazzuk $\text{CNOT}_{1\to 2}$-t, majd H-t az első qubiten, ahogy itt:



3.14. Gyakorló feladat megoldása

Vegyük észre a 3.52., 3.53., 3.54. és 3.55. egyenletekből, hogy a Bell-állapotok csak bitflipekben és előjelekben különböznek. Emlékezzünk, hogy egy bitet átbillenthetünk lokális NOT művelettel, és hogy néhány előjelet bevezethetünk lokális Z műveletek alkalmazásával, ahol Z a 2.12. egyenletben van definiálva. A $|\Phi^-\rangle$ létrehozásához Alíz egyszerűen alkalmazhat egy Z műveletet a qubitjén:

$$(Z \otimes I)\,|\Phi^+\rangle = (Z \otimes I)\left(\frac{1}{\sqrt{2}}\,|00\rangle + \frac{1}{\sqrt{2}}\,|11\rangle\right) = \frac{1}{\sqrt{2}}\,|00\rangle - \frac{1}{\sqrt{2}}\,|11\rangle = |\Phi^-\rangle\,.$$

A $|\Psi^+\rangle$ létrehozásához Alíz ehelyett egy NOT műveletet alkalmaz a qubitjén:

$$(\text{NOT} \otimes I)\,|\Phi^+\rangle = (\text{NOT} \otimes I)\left(\frac{1}{\sqrt{2}}\,|00\rangle + \frac{1}{\sqrt{2}}\,|11\rangle\right) = \frac{1}{\sqrt{2}}\,|10\rangle + \frac{1}{\sqrt{2}}\,|01\rangle = |\Psi^+\rangle\,.$$

A $|\Psi^-\rangle$ létrehozásához Alíz először egy NOT műveletet, majd egy Z műveletet alkalmaz a qubitjén:

$$(Z\,\text{NOT} \otimes I)\,|\Phi^+\rangle = (Z \otimes I)\left(\frac{1}{\sqrt{2}}\,|10\rangle + \frac{1}{\sqrt{2}}\,|01\rangle\right) = -\frac{1}{\sqrt{2}}\,|10\rangle + \frac{1}{\sqrt{2}}\,|01\rangle = |\Psi^-\rangle\,.$$

4. Küldetés: A qubitek összehangolása

Az előző héten a két qubit állapotairól és a rajtuk végezhető műveletekről tanultunk. Megismertük az *összefonódott* kvantumállapotokat is, és azt, hogy ezek hogyan használhatók hatékonyabb kommunikációra (pl. két bit átvitelére egyetlen qubit küldésével) és hogyan lehet velük felülmúlni a determinisztikus vagy valószínűségi stratégiákat bizonyos játékokban. Ezen a héten tetszőleges számú qubitet fogunk használni, és elkezdjük összeállítani a sok műveletből álló kvantumáramköröket, hogy érdekes problémákat oldjunk meg velük.

4.1. Kvantumáramkörök

Amikor sok qubited van és sok műveletet végzel rajtuk, könnyen elveszítheted az áttekintést. Ahogy egy zeneszerző kottát ír, hogy pontosan meghatározza, melyik zenésznek mikor milyen hangot kell játszania, úgy a kvantumáramkörök is szemléletes leírást adnak arról, hogy melyik qubiten mikor milyen műveletet kell végrehajtani. A különböző műveletek olyanok, mint a különböző hangjegyek, a különböző qubitek pedig olyanok, mint a különböző zenészek egy zenekarban!

A zenekartól eltérően azonban egyes kvantumműveletek egyszerre két (vagy néha akár három!) qubitet is érintenek. Ez olyan lenne, mintha arra kérnénk a gitárost, hogy pengesse a hegedűt, miközben a hegedűs vonózza a gitárt! Az ilyen kölcsönhatások a zenészek (vagy qubitek) között nagyon szórakoztatóak lehetnek, és sokkal gazdagabb, összetettebb hangzást eredményezhetnek. A gyakorlatban azonban nehéz lehet ezt kivitelezni (képzeld el, ahogy a hegedűsnek át kell rohannia a színpadon, hogy elérje a gitárost!). Ráadásul e hektikus együttműködés során a hegedűs haja beleakadhat a gitáros nagy fülbevalójába, arra kényszerítve őket, hogy a dal hátralévő részét együtt játsszák.

Minél több kölcsönhatást hozol létre a qubitek között, általában annál jobban összefonódnak. A dal végére tipikusan az összes qubit annyira összefonódik egymással, hogy teljesen lehetetlen megkülönböztetni őket. Az egyetlen módja annak, hogy szétválasszuk és valamilyen értelmes állapotba hozzuk őket, az a mérés! Valójában így működik egy általános kvantumszámítás. Formálisabban, egy *kvantumáramkör* három összetevőből áll:

1. egy kezdeti állapot, ami tipikusan minden qubiten $|0\rangle$,
2. kvantumműveletek sorozata, ahol minden művelet általában csak néhány qubiten hat egyszerre (általában egy vagy két qubiten),
3. mérések az információ kiolvasásához (általában minden qubitet megmérünk).

A kvantumáramköröket grafikusan is ábrázolhatjuk ugyanolyan képekkel, amiket már ismerünk a QUIRKY-ből. A kvantumáramkörök tárgyalásánál a műveleteket és méréseket gyakran kvantumkapuknak nevezik (pl. "Hadamard-kapu" a "Hadamard-művelet" helyett). Nézzük meg most közelebbről ezt a három összetevőt, és lássuk, hogyan működnek együtt a qubitek nagy zenekarában.

4.1.1. Több kvantumbit

Az egy és két qubites rendszerekre tanult szabályok természetes módon általánosíthatók több qubites kvantumrendszerekre. Például egy három qubites rendszer tetszőleges állapota így írható fel:

$$\begin{aligned} |\psi\rangle = \psi_{000}\,|000\rangle + \psi_{001}\,|001\rangle + \psi_{010}\,|010\rangle + \psi_{011}\,|011\rangle \\ + \psi_{100}\,|100\rangle + \psi_{101}\,|101\rangle + \psi_{110}\,|110\rangle + \psi_{111}\,|111\rangle\,, \end{aligned} \tag{4.1}$$

ahol $\psi_{ijk} \in [-1, 1]$ és ezeknek az amplitúdóknak a négyzetösszege ismét egy, azaz

$$\psi_{000}^2 + \psi_{001}^2 + \psi_{010}^2 + \psi_{011}^2 + \psi_{100}^2 + \psi_{101}^2 + \psi_{110}^2 + \psi_{111}^2 = 1.$$

Vegyük észre, hogy összesen $8 = 2^3$ amplitúdó van, egy-egy minden három bites bitsorozathoz. A $|\psi\rangle$-re gondolhatunk úgy is, mint egy nyolc elemű vektorra:

$$|\psi\rangle = \begin{pmatrix} \psi_{000} \\ \psi_{001} \\ \psi_{010} \\ \psi_{011} \\ \psi_{100} \\ \psi_{101} \\ \psi_{110} \\ \psi_{111} \end{pmatrix}.$$

Általánosabban, egy n qubites állapotot 2^n amplitúdóval $\psi_{a_1,\ldots,a_n}$ lehet megadni, egyet-egyet minden n bites bitsorozathoz:

$$|\psi\rangle = \psi_{00\ldots00}\,|00\ldots00\rangle + \psi_{00\ldots01}\,|00\ldots01\rangle + \ldots + \psi_{11\ldots11}\,|11\ldots11\rangle \tag{4.2}$$

Itt is minden $\psi_{a_1,\ldots,a_n}$ amplitúdónak a $[-1, 1]$ intervallumban kell lennie, és a négyzeteik összegének egynek kell lennie:

$$\psi_{00\ldots00}^2 + \psi_{00\ldots01}^2 + \ldots + \psi_{11\ldots11}^2 = 1 \tag{4.3}$$

Ha néhány $\psi_{a_1,\ldots,a_n}$ amplitúdó nulla, azokat egyszerűen kihagyhatjuk. Például az ötqubites kvantumállapot

$$\frac{1}{\sqrt{2}}\left(|00000\rangle + |11111\rangle\right)$$

32 amplitúdóval rendelkezik, amelyből 30 nulla.

Mivel 2^n amplitúdó van, $|\psi\rangle$-re gondolhatunk úgy is, mint egy 2^n dimenziós vektorra. Mit jelent geometriailag a 4.3. egyenlet? Egy qubit esetén a 2.1.2. alfejezet-ben láttuk, hogy az állapotok az egységkör pontjainak felelnek meg, azaz egységnyi hosszúságú kétdimenziós vektoroknak. A Pitagorasz-tétel szerint bármely dimenzióban igaz, hogy egy vektor összes komponensének négyzetösszege a vektor hosszának négyzete. Így a 4.3. egyenlet geometriailag azt jelenti, hogy $|\psi\rangle$ egy *egységnyi hosszúságú* vektornak vagy *egységvektornak* felel meg egy 2^n dimenziós térben.

Vegyük észre, hogy az amplitúdók száma nagyon gyorsan nő a qubitek számával. Ez magyarázza, hogy miért válik gyorsan lehetetlenné a kvantumállapotok közvetlen tárolása klasszikus számítógépen. Például egy $n = 300$ qubites kvantumállapot reprezentálásához *több amplitúdóra lenne szükség, mint ahány atom van a megfigyelhető univerzumban*! Emiatt nem lehet 10-nél több qubited a QUIRKY-ben, mert nem szeretnénk, ha a böngésződ kifogyna a memóriából!

Ahogy a 3.33. egyenlet-ben, a tenzorszorzatot „$\otimes$” használhatjuk tetszőleges számú qubit kvantumállapotainak kombinálására. Ha két bázisállapotunk van, a tenzorszorzatukat egyszerűen a bitsorozatok összefűzésével definiáljuk. A 3.32. egyenlet-beli kétqubites eset általánosítása:

$$|a_1, \ldots, a_n\rangle \otimes |b_1, \ldots, b_m\rangle = |a_1, \ldots, a_n, b_1, \ldots, b_m\rangle\,. \tag{4.4}$$

Például,

$$|101\rangle \otimes |01\rangle = |10101\rangle .$$

Általában, ha $|\alpha\rangle$ egy tetszőleges n qubites kvantumállapot és $|\beta\rangle$ egy tetszőleges m qubites kvantumállapot, akkor a tenzorszorzatuk vagy kombinált állapotuk egy $n+m$ qubites állapot. Ennek az állapotnak a kiszámításához egyszerűen "szorzunk" a disztributív törvény használatával, majd minden tagra alkalmazzuk a 4.4. egyenlet-t. Például két maximálisan összefonódott állapot tenzorszorzata a következő:

$$\begin{aligned}
&|\Phi^+\rangle \otimes |\Phi^+\rangle \\
&= \left(\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle\right) \otimes \left(\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle\right) \\
&= \frac{1}{\sqrt{2}}\frac{1}{\sqrt{2}}|00\rangle \otimes |00\rangle + \frac{1}{\sqrt{2}}\frac{1}{\sqrt{2}}|00\rangle \otimes |11\rangle + \frac{1}{\sqrt{2}}\frac{1}{\sqrt{2}}|11\rangle \otimes |00\rangle + \frac{1}{\sqrt{2}}\frac{1}{\sqrt{2}}|11\rangle \otimes |11\rangle \\
&= \frac{1}{2}|0000\rangle + \frac{1}{2}|0011\rangle + \frac{1}{2}|1100\rangle + \frac{1}{2}|1111\rangle .
\end{aligned}$$

4.1. Gyakorló feladat: Bell-állapotok tenzorszorzata

Számítsd ki a 3.53. és 3.55. egyenletek-ben szereplő két Bell-állapot $|\Phi^-\rangle \otimes |\Psi^-\rangle$ tenzorszorzatát.

4.1.2. Műveletek

Milyen kvantummműveleteket tudunk végrehajtani, ha több qubitünk van? Egyrészt alkalmazhatjuk bármelyik egy- vagy kétqubites műveletet, amelyeket a 2. és 3. alfejezetek fejezetekben tárgyaltunk, a többqubites állapot bármely választott qubitjére. Ez a 3.2.2. alfejezet fejezetben leírtak szerint működik.

Például, ha U egy egyqubites művelet, azaz egy forgatás vagy tükrözés, akkor definiálhatunk egy kvantummműveletet, amelyet U_1-gyel jelölünk, és amely az U műveletet alkalmazza egy n-qubites állapot első qubitjére. Ezt a következőképpen definiáljuk a bázisállapotokra:

$$U_1 |a_1, \ldots, a_n\rangle = U |a_1\rangle \otimes |a_2, \ldots, a_n\rangle .$$

Vegyük észre, hogy a tenzorszorzat kombinálja az $U|a_1\rangle$ egyqubites állapotot az $(n-1)$-qubites $|a_2, \ldots, a_n\rangle$ bázisállapottal, hogy egy n qubites állapotot hozzon létre, ahogy szeretnénk. Ahogy szokás, az U_1 műveletet lineárisan kiterjesztjük az általános n qubites kvantumállapotokra. Hasonlóképpen definiáljuk a U_2, U_3, stb. kvantummműveleteket, amelyek az U műveletet a második, harmadik, stb. qubiten alkalmazzák.

4.2. Gyakorló feladat: Egyqubites művelet alkalmazása

Számítsd ki, mi lesz az eredménye, ha a Hadamard-műveletet alkalmazzuk a $|\Phi^+\rangle \otimes |1\rangle$ háromqubites állapot második qubitjére. Más szóval, számítsd ki a $H_2\left(|\Phi^+\rangle \otimes |1\rangle\right)$ kifejezés eredményét. Az eredményt írd fel a 4.1. egyenlet alakjában.

Hasonlóképpen kitalálhatjuk, hogyan lehet egy kétqubites műveletet alkalmazni n qubitből kiválasztott két qubitre. Főleg a vezérelt-NOT műveletekkel fogunk foglalkozni: a $\mathrm{CNOT}_{k\to l}$ művelet, ahol $k \neq l$, az l-edik qubitet (a célqubitet) billenti át a k-adik qubit (a vezérlőqubit) értékétől függően. Matematikailag a bázisállapotokon való hatása a következő:

$$\mathrm{CNOT}_{k\to l} |a_1, \ldots, a_l, \ldots, a_n\rangle = |a_1, \ldots, a_l \oplus a_k, \ldots, a_n\rangle ,$$

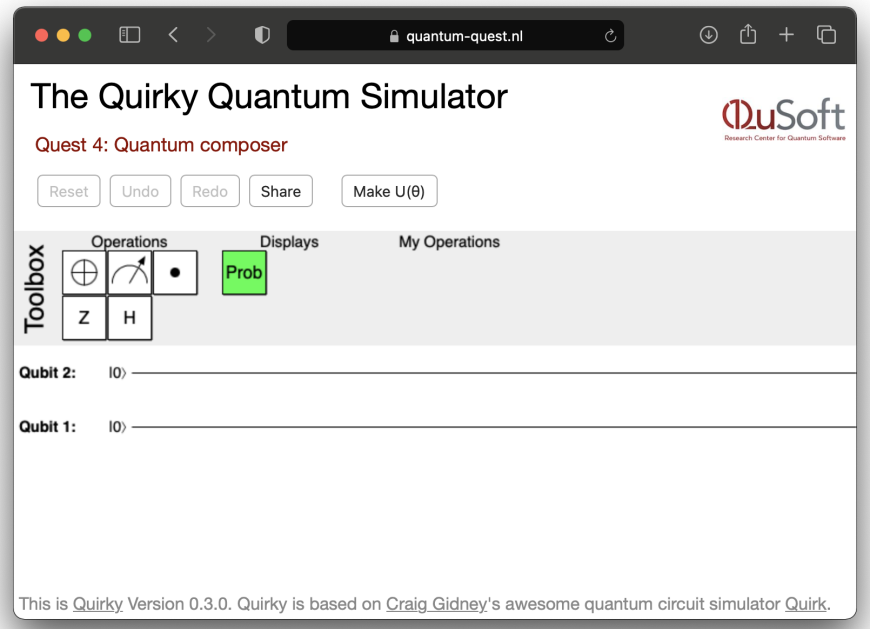
és ezt az előírást lineárisan kiterjesztjük tetszőleges n-qubites állapotokra. Például a $\text{CNOT}_{1\to 3}$ vezérelt-NOT művelet a következőképpen van definiálva négyqubites bázisállapotokra:

$$\text{CNOT}_{1\to 3} |a_1, a_2, a_3, a_4\rangle = |a_1, a_2, a_3 \oplus a_1, a_4\rangle .$$

Hogy néz ki mindez QUIRKY-ben? Lépjünk át a

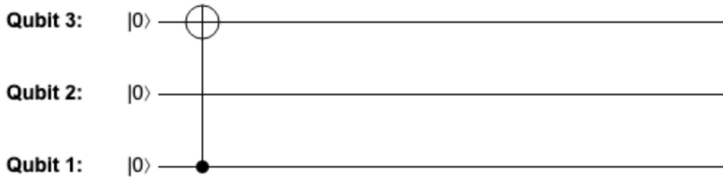
https://www.quantum-quest.org/quirky

oldalra, és kattintsunk a „Quest 4" gombra, hogy megtudjuk. A böngésződ nagyjából úgy fog kinézni, mint a 4.1. ábra.



4.1. ábra. QUIRKY a 4. küldetéshez.

Várj csak, úgy tűnik, mintha QUIRKY pontosan ugyanúgy nézne ki, mint múlt héten!? Azonban amint felveszel egy műveletet a toolboxból, egy új vezeték jelenik meg alul – lehetővé téve, hogy egy további qubiten dolgozz. (Persze korlátoztuk a qubiteknek a számát egy ésszerű számra, amit a klasszikus számítógéped még boldogan tud szimulálni!) Miért nem próbálod ki most, és hozol létre egy $\text{CNOT}_{1\to 3}$ műveletet, ahogy a következő képen látható?



Amikor a kvantumműveletek különböző qubiteken hatnak, párhuzamosan is végrehajthatjuk őket. Ahogy a 3.2.3. ben is, erre a célra újra felhasználjuk a tenzorszorzat szimbólumot. Ha U egy n qubiten végrehajtott kvantumművelet, és V egy m qubiten végrehajtott kvantumművelet, akkor definiálhatunk egy $U \otimes V$ kvantumműveletet $(n+m)$ qubiten, ami mindkét művelet párhuzamos végrehajtásának felel meg. Bázisállapotokon ez így néz ki:

$$(U \otimes V) |a_1, \ldots, a_n, b_1, \ldots, b_m\rangle = U |a_1, \ldots, a_n\rangle \otimes V |b_1, \ldots, b_m\rangle , \tag{4.5}$$

és ezt lineárisan kiterjesztjük tetszőleges állapotokra. A 4.5. egyenletből következik, hogy

$$(U \otimes V)(|\alpha\rangle \otimes |\beta\rangle) = U|\alpha\rangle \otimes V|\beta\rangle\,,$$

de ez csak akkor igaz, ha $|\alpha\rangle$ egy n-qubites állapot és $|\beta\rangle$ egy m-qubites állapot! A következő feladatban a két tenzorszorzat szimbólum *nem* ilyen módon van elrendezve, így *nem használhatod* ezt a szabályt!

4.3. Gyakorló feladat: Nem illeszkedő tenzorszorzatok

Vizsgáld meg a következő háromqubites állapotot: $(\mathrm{CNOT}_{2\to1} \otimes I)(|0\rangle \otimes |\Phi^-\rangle)$.

1. Hogyan tudod felépíteni ezt az állapotot QUIRKYsegítségével? Írd fel az állapotot a 4.1. egyenlet alakjában.
2.

A tenzorszorzatot többször is használhatjuk, hogy fokozatosan egyre nagyobb kvantumműveleteket építsünk fel. Íme három példa különböző számú qubitekre:

1. $I \otimes I \otimes U \otimes I$ ugyanaz a négyqubites művelet, mint U_3,
2. $I \otimes \mathrm{CNOT}_{1\to2} \otimes I \otimes I$ az öt qubiten végrehajtott $\mathrm{CNOT}_{2\to3}$ vezérelt-NOT művelet,
3. $Z \otimes I \otimes X$ az a kvantumművelet, amely Z-t alkalmaz az első qubiten, és párhuzamosan X-et a harmadik qubiten (ezt írhatjuk úgy is, hogy Z_1X_3 vagy X_3Z_1).

4.1.3. A legáltalánosabb kvantumműveletek

Mik a legáltalánosabb műveletek, amiket n qubites kvantumállapotokon alkalmazhatunk? Valójában bármely művelet, amely a következő három tulajdonsággal rendelkezik:

1. lineáris,
2. kvantumállapotokat kvantumállapotokba képez le,
3. invertálható

érvényes kvantumművelet!

4.4. Gyakorló feladat: Toffoli

Definiáljuk a Toffoli-műveletet három qubiten a következőképpen:

$$T|a,b,c\rangle = |a,b,c \oplus ab\rangle$$

a bázisállapotokon (ab az $a,b \in \{0,1\}$ két bit szorzata, és az $\oplus$ műveletet a 3.20. ben definiáltuk), és terjesszük ki lineárisan tetszőleges háromqubites állapotokra. Mutasd meg, hogy T kvantumállapotokat kvantumállapotokba képez le, és hogy T invertálható.

Megjegyzés: T csak akkor fordítja meg a bázisvektor harmadik bitjét, ha az első két bit értéke egy – tehát ez egy fajta "kétszeresen vezérelt" NOT művelet.

A 4.4. gyakorló feladat megmutatja, hogy a Toffoli-művelet egy érvényes háromqubites kvantumművelet. Érdekes módon lehetséges T-t egy- és kétqubites műveletek sorozataként felírni. Valójában ez *bármely* n qubites kvantumműveletre lehetséges – de ezt nem fogjuk tárgyalni ebben a kurzusban, mivel tapasztalt kvantumkomponistának kell lenni ahhoz, hogy megértsük, hogyan lehet ezt megtenni!

4.1.4. Áramköri azonosságok

Amikor egy nagyon bonyolult kvantumáramkörrel dolgozol, jó ismerni néhány trükköt az egyszerűsítéshez. Az ilyen trükkök nemcsak megkönnyíthetik annak megértését, hogy mit csinál az áramkör, hanem hatékonyabbá és így gyorsabban végrehajthatóvá is teszik a kvantumszámítógépen. Nézzünk meg néhány egyszerű példát ilyen trükkökre, amelyek csak egyetlen qubitet érintenek.

Emlékezz vissza a 2.4.3. ból, hogy minden egyqubites művelet vagy forgatás, vagy tükrözés. Hasznos ezeknek a műveleteknek a hatását vizuálisan ábrázolni, felidézve a 2.1.2. ből, hogy a qubit állapotok egy kört alkotnak. Egy azonnali megfigyelés, amit tehetsz, hogy ha bármely rögzített tükrözést kétszer alkalmazol ugyanazon a qubiten, akkor visszajutsz az eredeti állapotba. Valóban, ez vizuálisan is egyértelmű, mivel ugyanazon tengely körüli kétszeri tükrözés mindent visszaállít eredeti állapotába (például ezt láthatod a 2.4. ban a NOT művelet esetén). Ez különösen igaz a Hadamard-műveletre H, amelyről tudjuk, hogy tükrözés a 2.21. egyenlet miatt. Ellenőrizzük ezt a geometriai intuíciót egy kis számítással, és nézzük meg azt is, hogy a Hadamard-kapu hogyan teszi lehetővé a NOT és Z kapuk közötti átalakítást.

4.5. Gyakorló feladat: Z és NOT

Emlékezz vissza a 2.20. ból, hogy a Hadamard-kapu H a következőképpen működik:

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = |+\rangle\,, \qquad H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = |-\rangle\,.$$

1. Ellenőrizd, hogy H ismételt alkalmazása visszajuttat $|0\rangle$ és $|1\rangle$ állapotokba. Azaz, ellenőrizd, hogy
$$H|+\rangle = |0\rangle\,, \qquad H|-\rangle = |1\rangle\,.$$
2. Ellenőrizd, hogy $HZH = \text{NOT}$, ahol Z a 2.12. ben van definiálva.
3. Ellenőrizd, hogy $H\,\text{NOT}\,H = Z$.

Egy másik érdekes kérdés, hogy mi történik, ha két tetszőleges forgatást vagy tükrözést alkalmazunk egymás után? Tudjuk, hogy az eredménynek ismét vagy forgatásnak, vagy tükrözésnek kell lennie. De melyik az, és mi az új szög? Két egymást követő forgatás egyszerűen ugyanaz, mint egyetlen forgatás a két szög összegével, azaz $U(\varphi_2)U(\varphi_1) = U(\varphi_1 + \varphi_2)$. A következő feladat egy szabályt ad arra, hogyan lehet két egymást követő tükrözést egyetlen *forgatássá* egyszerűsíteni.

4.6. Gyakorló feladat: Tükrözések és forgatások (opcionális)

Igazold, hogy két tükrözés szorzata egy forgatás. Azaz mutasd meg, hogy

$$V(\theta_2)V(\theta_1) = U(\theta),$$

valamely θ szögre. Ki tudod fejezni a θ szöget θ_1 és θ_2 függvényében?

Ötlet: Használd a 2.19. t és azt, hogy $U(\varphi_2)U(\varphi_1) = U(\varphi_1 + \varphi_2)$.

A fennmaradó két esetet, amelyek egy forgatást és egy tükrözést tartalmaznak, magad is kidolgozhatod, és ellenőrizheted, hogy mindkettő egy $V(\theta)$ tükrözést eredményez valamilyen θ szögre.

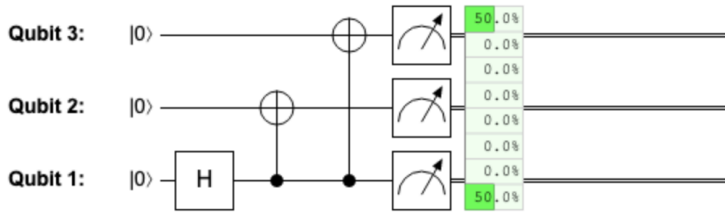
4.1.5. Minden qubit mérése

Miután befejeztük a qubitjeinket átalakító műveletek alkalmazását, szeretnénk valamilyen információt kinyerni. Ahogy korábban is, ennek egyetlen módja a qubitek mérése.

Mik a szabályok egy n qubites kvantumállapot méréséhez? Ha mind az n qubitet megmérjük, akkor eredményként n bitet kapunk, azaz egy $a_1 \dots a_n$ bitsorozatot. Mint korábban, bármely konkrét $a_1 \dots a_n$ kimenet valószínűsége a négyzetgyökös amplitúdó:

$$p_{a_1,\dots,a_n} = \psi^2_{a_1,\dots,a_n}. \tag{4.6}$$

A QUIRKY-ben minden qubitet megmérhetünk, mint korábban, minden qubithez egy mérési dobozt adva. A mérési eredmények valószínűségeinek megtekintéséhez hozzáadhatunk egy valószínűség kijelzőt - de ügyelnünk kell arra, hogy megfelelően átméretezzük, hogy minden vezetékre vonatkozzon. Próbáld meg reprodukálni a következő példát:



A QUIRKY műveletek ezen sorozata előállítja a

$$\frac{1}{\sqrt{2}}|000\rangle + \frac{1}{\sqrt{2}}|111\rangle \tag{4.7}$$

állapotot és megméri mind a három qubitet. Látod, hogyan működik ez?

4.1.6. Csak néhány qubit mérése

Természetesen mérhetünk csak egy részhalmazát is a qubiteknek. (Ezt múlt héten nem is tárgyaltuk két qubit esetén, mivel már így is sok mindenről beszéltünk.) Például tegyük fel, hogy van egy háromqubites kvantumállapotod $|\psi\rangle$, mint a 4.1. egyenlet-ban, de ahelyett, hogy mind a három qubitet mérnéd, csak az első qubitet méred. Mi a valószínűsége p_a annak, hogy az $a \in \{0,1\}$ eredményt kapjuk? Ez egyszerűen az összes olyan valószínűség összege a 4.6. egyenlet-ból, amely egy a-val kezdődő sztringnek felel meg:

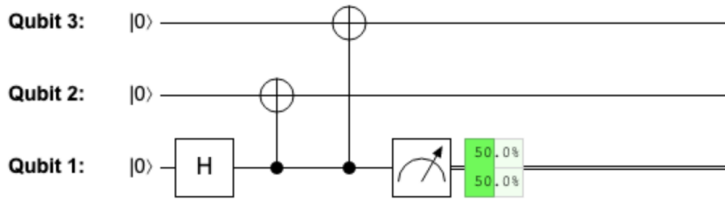
$$p_a = \psi^2_{a00} + \psi^2_{a01} + \psi^2_{a10} + \psi^2_{a11}. \tag{4.8}$$

Például, ha az

$$\frac{1}{\sqrt{8}}|000\rangle + \sqrt{\frac{2}{8}}|010\rangle + \sqrt{\frac{5}{8}}|111\rangle$$

állapot első qubitjét mérjük, akkor $3/8 = 1/8 + 2/8$ valószínűséggel kapjuk a 0 eredményt.

Egyedi qubiteket mérhetünk a QUIRKY-ben egyszerűen úgy, hogy csak egy mérést adunk hozzá ahhoz a vezetékhez, amely érdekel minket. A mérési eredmények valószínűségének megtekintéséhez húzz egy valószínűség kijelzőt az áramkörre. Miért nem próbálod ki most rögtön? Például a következő műveletek sorozata előállítja a 4.7. egyenlet-ben lévő állapotot és csak az első qubitet méri:



(4.9)

Valóban, a 4.8. egyenlet szerint egyenlő valószínűséggel kell kapnunk 0-t és 1-et.

Miután megmérted az első qubitet és valamilyen $a \in \{0,1\}$ eredményt kaptál, mi a második és harmadik qubit kvantumállapota? Követve ugyanazt az eljárást, mint a valószínűségi biteknél a 3.1.3. alfejezet-ben, először összegyűjtjük $|\psi\rangle$ összes olyan tagját, ahol az első qubit abban az állapotban van, ami megfelel az érdekelt eredménynek:

$$\psi_{a00}\,|a00\rangle + \psi_{a01}\,|a01\rangle + \psi_{a10}\,|a10\rangle + \psi_{a11}\,|a11\rangle\,.$$

Ezután elhagyjuk az első qubitet mind a négy tagban, mivel azt már megmértük:

$$\psi_{a00}\,|00\rangle + \psi_{a01}\,|01\rangle + \psi_{a10}\,|10\rangle + \psi_{a11}\,|11\rangle\,.$$

Végül normalizáljuk ezt, hogy érvényes kétqubites állapotot kapjunk. Ehhez olyan c számot keresünk, hogy $\frac{\psi_{a00}}{c}\,|00\rangle + \frac{\psi_{a01}}{c}\,|01\rangle + \frac{\psi_{a10}}{c}\,|10\rangle + \frac{\psi_{a11}}{c}\,|11\rangle$ egy qubit állapot legyen, azaz

$$\left(\frac{\psi_{a00}}{c}\right)^2 + \left(\frac{\psi_{a01}}{c}\right)^2 + \left(\frac{\psi_{a10}}{c}\right)^2 + \left(\frac{\psi_{a11}}{c}\right)^2 = 1.$$

Az általános előjel nem fontos, így egyszerűen használhatjuk a

$$c = \sqrt{\psi_{a00}^2 + \psi_{a01}^2 + \psi_{a10}^2 + \psi_{a11}^2},$$

értéket, ami a 4.8. egyenlet-ben lévő valószínűség négyzetgyöke.

Összefoglalva, ha egy háromqubites állapot első qubitjét méred mint a 4.1. egyenlet-ban, akkor az $a \in \{0,1\}$ eredményt kapod

$$p_a = \psi_{a00}^2 + \psi_{a01}^2 + \psi_{a10}^2 + \psi_{a11}^2 \tag{4.10}$$

valószínűséggel, és a maradék két qubit eredményül kapott $|\psi_a\rangle$ kétqubites állapota

$$|\psi_a\rangle = \frac{\psi_{a00}\,|00\rangle + \psi_{a01}\,|01\rangle + \psi_{a10}\,|10\rangle + \psi_{a11}\,|11\rangle}{\sqrt{\psi_{a00}^2 + \psi_{a01}^2 + \psi_{a10}^2 + \psi_{a11}^2}}. \tag{4.11}$$

Mit jelent ez a (4.9) helyzetben, ahol előállítjuk a $\frac{1}{\sqrt{2}}\,|000\rangle + \frac{1}{\sqrt{2}}\,|111\rangle$ állapotot, majd megmérjük az első qubitet? Ha a mérési eredmény 0 (ami $1/2$ valószínűséggel történik), a maradék két qubit állapota

$$\frac{\frac{1}{\sqrt{2}}\,|00\rangle}{\sqrt{\frac{1}{2}}} = |00\rangle\,.$$

Ha viszont az eredmény 1, akkor hasonlóan a maradék qubitek állapota $|11\rangle$.

Mivel egy qubit mérése sok közül elég trükkös lehet, beszéljünk egy másik módszerről is ennek elvégzésére. Vegyünk ismét egy általános háromqubites $|\psi\rangle$ állapotot, mint a 4.1. egyenlet-ban, és tegyük fel, hogy az első qubitet akarjuk mérni. Átírhatjuk a $|\psi\rangle$ kifejezésében szereplő nyolc tagot a következőképpen:

$$\begin{aligned} |\psi\rangle = \sqrt{p_0}\,|0\rangle \otimes \frac{\psi_{000}\,|00\rangle + \psi_{001}\,|01\rangle + \psi_{010}\,|10\rangle + \psi_{011}\,|11\rangle}{\sqrt{p_0}} \\ + \sqrt{p_1}\,|1\rangle \otimes \frac{\psi_{100}\,|00\rangle + \psi_{101}\,|01\rangle + \psi_{110}\,|10\rangle + \psi_{111}\,|11\rangle}{\sqrt{p_1}}. \end{aligned} \tag{4.12}$$

Ezt most átírhatjuk így:

$$|\psi\rangle = \sqrt{p_0}\,|0\rangle \otimes |\psi_0\rangle + \sqrt{p_1}\,|1\rangle \otimes |\psi_1\rangle\,, \tag{4.13}$$

ahol a p_a-k valószínűségek (nevezetesen a 4.10. egyenlet-ból származók), és a $|\psi_a\rangle$-k kvantumállapotok (a 4.11. egyenlet-ból származó kétqubites állapotok).

Valójában, ha sikerül a kvantumállapotodat a 4.13. egyenlet formájába írni, akkor egyszerűen leolvashatod a mérési eredmények valószínűségeit, és azt is láthatod, hogy mi lesz a maradék két qubit állapota a mérés után. Például,

$$\frac{1}{\sqrt{2}}\,|000\rangle + \frac{1}{\sqrt{2}}\,|111\rangle = \frac{1}{\sqrt{2}}\,|0\rangle \otimes |00\rangle + \frac{1}{\sqrt{2}}\,|1\rangle \otimes |11\rangle\,,$$

Ez megerősíti, hogy a folyamatosan használt példánkban (4.9), mindkét eredmény 1/2 valószínűséggel következik be, és a maradék qubitek állapota vagy $|00\rangle$, vagy $|11\rangle$, az eredménytől függően. Ez a módszer, hogy először csoportosítjuk a tagokat, majd normalizáljuk őket, meglehetősen intuitív és általában nagyon hasznos. Azonban amikor ezt használod, nagyon óvatosnak kell lenned, nehogy elfelejtsd helyesen normalizálni az állapotokat! Azaz, bármilyen $\sqrt{p_a}$ konstansokat húzol ki a két bázisállapot elé a 4.12. és 4.13. egyenletek-ben, azoknak ki kell elégíteniük a $p_0 + p_1 = 1$ feltételt, és a maradék qubiteken lévő $|\psi_0\rangle$ és $|\psi_1\rangle$ állapotoknak megfelelően normalizáltnak kell lenniük, lásd 4.3. egyenlet.

Teljesen hasonlóan járhatunk el, ha háromnál több qubitünk van, vagy ha az első helyett egy másik qubitet akarunk mérni, vagy ha egynél több qubitet akarunk mérni! Például tegyük fel, hogy az első *két* qubitet akarjuk mérni egy általános háromqubites $|\psi\rangle$ állapotból a 4.1. egyenlet-ból. Ekkor a mérési eredmény két bitből áll, a-ból és b-ből, amelyek a következő valószínűségekkel fordulnak elő:

$$p_{a,b} = \psi_{ab0}^2 + \psi_{ab1}^2, \tag{4.14}$$

és a mérés után megmaradó qubit állapota

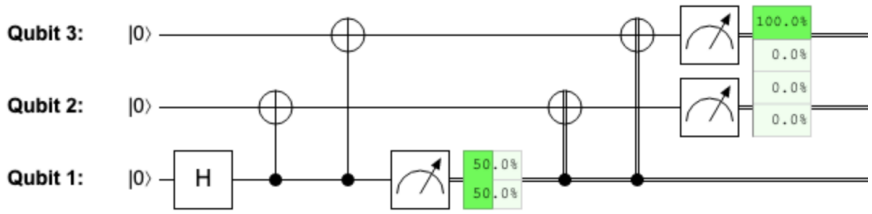
$$|\psi_{a,b}\rangle = \frac{\psi_{ab0}\,|0\rangle + \psi_{ab1}\,|1\rangle}{\sqrt{\psi_{ab0}^2 + \psi_{ab1}^2}}. \tag{4.15}$$

4.7. Gyakorló feladat: Kettő a háromból

Mik az eredmények valószínűségei, ha a 4.7. egyenlet-ben szereplő háromqubites állapot első két qubitjét méred? Használd a QUIRKY-t az eredményed megerősítéséhez.

Ha néhány qubitet mérünk, de nem mindet, gyakran akarjuk használni a mérési eredményeket annak meghatározására, hogy egy műveletet alkalmazni kell-e a maradék qubitekre vagy sem. Például tegyük fel, hogy a (4.9) helyzetben vissza akarod állítani a maradék két qubitet $|00\rangle$ állapotba. Ha a mérési eredmény nulla, nincs szükség semmilyen műveletre. De ha a mérési eredmény egy, akkor tudjuk, hogy a két maradék qubit $|11\rangle$ állapotban van, és vissza szeretnénk állítani őket $|00\rangle$ állapotba. Ezt megtehetjük úgy, hogy mindegyikükre NOT műveletet alkalmazunk. Azonban honnan tudjuk, hogy valóban alkalmaznunk kell-e ezt a műveletet, hiszen ez az első qubit korábbi mérési eredményétől függ. Ezért csak akkor szeretnénk alkalmazni, amikor a mérési eredmény 1. Más szóval, egy vezérelt NOT kaput szeretnénk alkalmazni, ahol a kontroll most egy klasszikus bit (a mérés eredménye), de a cél még mindig kvantum.

A QUIRKY-ben ezt úgy valósíthatjuk meg, ahogy várnád, nevezetesen egy klasszikus bitet használva kontrollként és egy qubitet célként, mint a következő példában:



Itt, miután az első qubitet megmértük, két további vezérelt NOT műveletet alkalmazunk, amelyeket a mérési eredmény vezérel, majd megmérjük a maradék két qubitet. A kép mutatja, hogy valóban sikeresen visszaállítottuk a két qubitet, mivel mérésük 100% valószínűséggel $[00]$-t eredményez.

Ha akarnánk, leírhatnánk ezeket a műveleteket "hibrid" állapotok segítségével, amelyek egy bitből és két qubitből állnak, például,

$$\mathrm{CNOT}_{1\to 2}[a] \otimes |b, c\rangle = [a] \otimes |a \oplus b, c\rangle \,,$$

de itt nem lesz szükségünk erre a formalizmusra.

4.2. Kvantum meglepetések

Most néhány érdekes jelenséget fogunk megvitatni, amelyek akkor merülnek fel, amikor kvantumbitekkel foglalkozunk. A következő szakaszok nagyrészt egymástól függetlenül olvashatók, így nyugodtan kezdd azzal, amelyik a legérdekesebbnek tűnik számodra.

4.2.1. Nincs klónozás

Amikor klasszikus bitünk van, könnyű *másolni* vagy klónozni - egyszerűen megnézzük és lemásoljuk, amit látunk:

$$\begin{aligned}[0] &\mapsto [00],\\ [1] &\mapsto [11].\end{aligned}$$

Klónozhatunk-e kvantumbiteket is?

Tegyük fel egy pillanatra, hogy ez lehetséges. Ez azt jelentené, hogy létezik egy C kvantumművelet, amely bármely $|\psi\rangle$ állapotú qubit és egy friss $|0\rangle$ állapotú qubit esetén így működik:

$$C(|\psi\rangle \otimes |0\rangle) = |\psi\rangle \otimes |\psi\rangle \tag{4.16}$$

hogy egyetlen példányból két példányt állítson elő $|\psi\rangle$-ből. (Miért adjuk meg a friss qubitet? Azért, hogy C-nek ugyanannyi bemeneti qubitje legyen, mint kimeneti.)

Például a klónozó a következőképpen működne a bázisállapotokon:

$$\begin{aligned}C\,|00\rangle &= |00\rangle\,,\\ C\,|10\rangle &= |11\rangle\,.\end{aligned} \tag{4.17}$$

Ahogy könnyen klónozhatunk egy klasszikus bitet, ugyanúgy könnyű olyan kvantumműveletet találni, amely klónozza a bázisállapotokat. Például a $\mathrm{CNOT}_{1\to 2}$ vezérelt NOT művelet a 3.46. egyenlet-ból elvégzi ezt a feladatot.

De létezik-e olyan kvantumművelet, amely tetszőleges ismeretlen qubit állapotot tud klónozni, nem csak bázisállapotot? A következő házi feladatban meg fogod mutatni, hogy ez *nem* lehetséges.

4.1. Házi feladat: Nincs klónozás

Ebben a házi feladatban be akarjuk bizonyítani, hogy nem létezik olyan C kvantumművelet, amely kielégíti a 4.16. egyenlet-t. Egy trükköt fogunk használni, amit *ellentmondásos bizonyításnak* hívunk. Ez azt jelenti, hogy megmutatjuk, ha létezne ilyen C klónozó művelet, akkor ez olyasmit implikálna, amiről tudjuk, hogy helytelen (pl. „$0 = 1$”). Ebből arra következtethetünk, hogy ilyen C nem létezhet.

Kezdjük tehát azzal a feltételezéssel, hogy létezik egy C kvantumművelet, amely kielégíti a 4.16. egyenlet-t. Most kétféleképpen számíthatod ki $C(|+\rangle \otimes |0\rangle)$-t:

1. Először használd a 4.16. egyenlet-t, majd írd az eredményt a 3.30. egyenlet formájában.
2. Először fejtsd ki $|+\rangle \otimes |0\rangle$-t a 3.30. egyenlet formájában, majd használd fel, hogy C lineáris, és végül alkalmazd a 4.16. egyenlet-t.

Ugyanazt az eredményt kapod mindkét esetben? Ha nem, milyen következtetést vonhatsz le ebből?

Ezt a híres eredményt nem-klónozási tételként ismerjük. Ugyanez a következtetés (és valószínűleg az az érv is, amit a 4.1. házi feladat-ben adtál) a valószínűségi bitekre is vonatkozik! Íme egy intuitív magyarázat arra, hogy miért nem másolhatunk sem valószínűségi, sem kvantuminformációt. Ha ez lehetséges lenne, akkor egy ismeretlen p állapotú valószínűségi bit vagy egy ismeretlen $|\psi\rangle$ állapotú qubit esetén először tetszőleges számú másolatot készíthetnénk p-ről és $|\psi\rangle$-ről. Ezeket a másolatokat aztán különböző módokon mérhetnénk, és a kapott adatokat felhasználhatnánk p valószínűségeinek vagy $|\psi\rangle$ amplitúdóinak tetszőleges pontosságú becslésére (ahogy azt a 2.5.1. alfejezet-ben tettük a sárga rejtélyes doboz belső működésének kiderítéséhez). Így egyetlen valószínűségi vagy kvantumbitből tetszőleges mennyiségű információt nyerhetnénk ki. Ez nyilvánvalóan nem lehet lehetséges!

Valóban, ha ez lehetséges lenne, akkor egy nagyon furcsa világban élnénk (sokkal furcsábban, mint amit a kvantummechanika leír)! Képzelj el például egy érmét, amelynek a fej valószínűsége $p = 0,1011010010\ldots$, ahol a bináris számjegyek a Wikipedia teljes tartalmát, valamint az összes YouTube-videót és az interneten található összes macskás képet kódolják. Ha a valószínűségi bitek klónozása lehetséges lenne, egyszer feldobhatnám ezt az érmét, és leírhatnám, milyen eredményt kaptam. Ez egy valószínűségi információ bit, amely p valószínűséggel egyenlő 0-val. Ha ezt a valószínűségi bitet elküldöm neked, és te képes vagy klónozni, akkor annyi másolatot készíthetsz róla, amennyit csak akarsz, majd mindet megmérheted. A mérési eredmények megtekintésével és a kapott nullák számolásával becsülhetnéd a p valószínűséget. Valójában az eredeti bit elegendő számú másolatának előállításával tetszőlegesen jól becsülhetnéd ezt a valószínűséget! Különösen képes lennél p bármely bináris számjegyét kinyerni, és így a p-ben kódolt összes információt is, beleértve a 65535-ös számú macskás képet!

Ennek nyilvánvalóan lehetetlennek kell lennie, mivel különben nem lenne szükségünk USB-meghajtókra, adatközpontokra vagy mobiltelefon-adatkapcsolat fizetésére - egyszerűen tárolhatnánk minden információnkat egyetlen valószínűségi bitben, és ezt a bitet elküldve másnak továbbíthatnánk az összes információt! Ez bizonyosan túl szép ahhoz, hogy igaz legyen...

4.2.2. Egyszeri kulcsú titkosítás

Mielőtt a kvantumállapotok teleportálásáról beszélnénk, hasznos először megérteni egy egyszerűbb eljárást a valószínűségi bitekre, amit egyszeri kulcsú titkosításnak nevezünk. Ez az eljárás lehetővé teszi Alíz számára, hogy titkosítson egy üzenetet és elküldje Botinak úgy, hogy csak Boti érthesse meg az üzenet tartalmát. Azaz, ha bárki más elfogná az üzenetet (például osztálytársuk, Éva), fogalma sem lenne arról, hogy mi az igazi üzenet. Az a tény, hogy ez egyáltalán

lehetséges, kissé meglepő. Valójában milyen előnye van Botinak Évával szemben, hogy képes helyesen dekódolni Alíz üzenetét, míg Évának abszolút fogalma sincs annak tartalmáról?

A trükk az, hogy Alíz és Boti először találkozik egy kávézóban. Vegyenek két érmét, ragasszák össze őket rágógumival, dobják fel az így kapott "dupla érmét", majd válasszák szét újra a két érmét. Alíz és Boti mindketten elveszik az egyik feldobott érmét. Most osztoznak egy pár véletlen biten, amelyet a (3.5)-ból származó állapot ír le, nevezetesen

$$r = \frac{1}{2}([00] + [11]).$$

Gondolhatsz erre úgy, mint egy megosztott titokra! Ráadásul csak Alíz és Boti tudja, mi ez a titok - hogy megtudják, egyszerűen megnézhetik a saját érméjüket (tehát megmérik őket). Mindketten ugyanazt az oldalt fogják látni, és mindkét oldal $1/2$ valószínűséggel fordul elő. Ez egy nagyon jó titok, mivel Éva nem tudja jobban megjósolni, mint vaktában tippelni!

Most nézzük meg, hogyan tudják Alíz és Boti ezt kihasználni. Tegyük fel, hogy Alíznak van egy titkos üzenete $m \in \{0, 1\}$, amit el akar küldeni Botinak. Az összes bitjük állapotát a következő állapot írja le:

$$[m] \otimes r = \frac{1}{2}([m00] + [m11]), \tag{4.18}$$

ahol az első két bit (m és r első fele) Alízé, a harmadik bit (r második fele) pedig Botié.

Az üzenet elküldéséhez Alíznak meg kell néznie a megosztott véletlen bit r saját felét, és

1. ha 0-t lát, akkor elküldi m-et Botinak változatlanul,
2. ha 1-et lát, akkor NOT(m)-et küldi el Botinak.

Képzeljük el, hogy Éva elfogja ezt az üzenetet. Mit lát? m értékétől függetlenül $1/2$ valószínűséggel 0-t és $1/2$ valószínűséggel 1-et fog látni. Ez azért van, mert Alíz $1/2$ valószínűséggel invertálja m-et, ami hatékonyan randomizálja m-et úgy, hogy Éva egyenletesen véletlenszerű bitként látja.

De mi a helyzet Botival? Az ő számára nem tűnik egyenletesen véletlenszerűnek Alíz üzenete? Szerencsére Botinak megvan a megosztott véletlen bit másik fele. Bár eredetileg Alíz üzenete számára is véletlenszerűnek tűnik, dekódolhatja azt pontosan ugyanazzal az eljárással, mint Alíz: megnézi a megosztott véletlen bit saját felét, és

1. ha 0-t lát, akkor Alíz üzenetét változatlanul veszi,
2. ha 1-et lát, akkor egy NOT műveletet alkalmaz Alíz üzenetére.

Összességében Alíz üzenete vagy változatlanul kerül továbbításra, vagy kétszer invertálódik, ami azt jelenti, hogy Boti mindig helyesen érti meg. Azonban Éva szemszögéből $1/2$ valószínűséggel invertálódott, ami azt jelenti, hogy amit ő lát, az egy egyenletesen véletlenszerű bit. Ez tehát egy tökéletesen biztonságos módja annak, hogy Alíz és Boti kommunikáljon!

Értsük meg formálisabban, mi történik itt. Amikor Alíz invertálja az első bitjét, ha a második bitje 1, ez ugyanaz, mintha $\text{CNOT}_{2\to1}$ műveletet alkalmazna a két bitjén. Ezután Alíz elküldi az első bitet Botinak. Amikor Boti dekódolja Alíz üzenetét, amit tesz, az egy $\text{CNOT}_{3\to1}$ művelet alkalmazása (amit megtehet, mivel a harmadik bit mellett most már az első bit is nála van). Az eredményül kapott állapot:

$$\text{CNOT}_{3\to1}\ \text{CNOT}_{2\to1}([m] \otimes r).$$

Ezt a következőképpen értékelhetjük ki:

$$\text{CNOT}_{3\to1}\ \text{CNOT}_{2\to1}\ \frac{1}{2}([m,0,0] + [m,1,1]) = \text{CNOT}_{3\to1}\ \frac{1}{2}([m,0,0] + [\text{NOT}(m),1,1])$$
$$= \frac{1}{2}([m,0,0] + [\text{NOT}(\text{NOT}(m)),1,1]) = \frac{1}{2}([m,0,0] + [m,1,1]) = [m] \otimes r.$$

Tehát az üzenetbit visszakerül az eredeti állapotába, de most már Botinál van.

A fenti egyszeri kulcsú titkosítási protokoll érdekes aspektusa nemcsak az, hogy lehetővé teszi Alíz számára egy determinisztikus $[m]$ üzenet elküldését Botinak, hanem akár egy valószínűségi üzenet küldését is. A linearitásból következik, hogy ha Alíz üzenete ehelyett egy p eloszlású valószínűségi bit, akkor a kezdeti állapot $p \otimes r$, és a végső állapot ismét $p \otimes r$, ahol ezúttal p Botinál van. Azonban Éva szemszögéből a továbbított üzenet továbbra is egyenletesen véletlenszerű. Ennek meglepő része az, hogy egy egyenletesen véletlenszerű bit küldésével Alíz képes titokban továbbítani egy olyan valószínűségi bitet, amelynek eloszlásával talán maga sincs tisztában.

Ez az eljárás nagyon hasonlít a kvantum teleportációhoz, ahol Alíz egy $|\psi\rangle$ qubit állapotot tud továbbítani Botinak két (egy helyett) egyenletesen véletlenszerű bit küldésével. A teleportációhoz egy megosztott maximálisan összefonódott $|\Phi^+\rangle$ állapotot kell használniuk a megosztott véletlen bit r helyett. Mindkét esetben a megosztott erőforrást mérik és így felhasználják az eljárás során. Ezt a következő szakaszban tárgyaljuk.

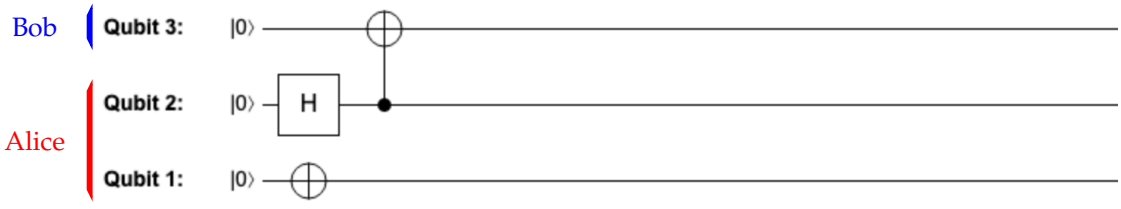
4.2.3. Kvantum teleportáció

Bár nem lehetséges kvantumbiteket klónozni, biztosan *mozgathatunk* kvantumbiteket egyik helyről a másikra. Valóban, minden qubit valamilyen fizikai objektumban vagy részecskében van tárolva, amely hordozza a qubitet. Például, ha Alíz a qubitjét egy foton polarizációjaként tárolja, egyszerűen elküldheti ezt a fotont Botinak. Ami azonban még meglepőbb, az az, hogy egy kvantumbitet egyik helyről a másikra lehet mozgatni véges számú klasszikus bit küldésével (valójában két bit is elegendő). Azért meglepő ez, mert egy általános $|\psi(\theta)\rangle$ qubit állapotot egy tetszőleges θ szög határoz meg, lásd 2.5. egyenlet, ami általában nem kódolható véges számú bitben. E meglepő tulajdonság miatt ezt az eljárást teleportációnak nevezzük. Hamarosan látni fogjuk, hogy összefonódásra van szükségünk ahhoz, hogy működjön!

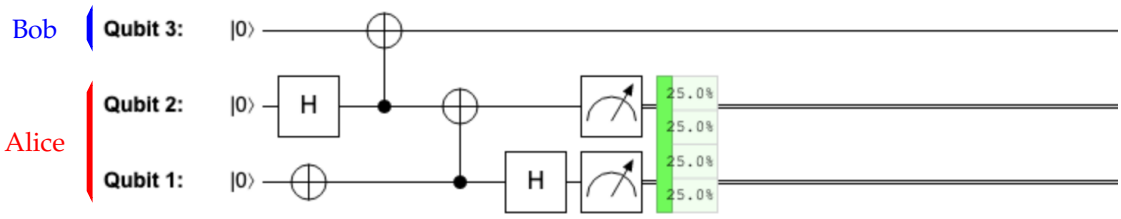
A teleportáció kiindulópontja a következő forgatókönyv. Elképzeljük, hogy Alíznak két qubitje van, Botinak pedig egy. Alíz első qubitje az *üzenetqubit*, amelyet el akar küldeni Botinak, és ez valamilyen tetszőleges $|\psi\rangle$ állapotban van - amelyet Alíz maga sem ismer feltétlenül! A második qubitje és Boti qubitje egy maximálisan összefonódott $|\Phi^+\rangle$ állapotban vannak. Tehát a három qubit a következő állapotban van:

$$|\psi\rangle \otimes |\Phi^+\rangle ,$$

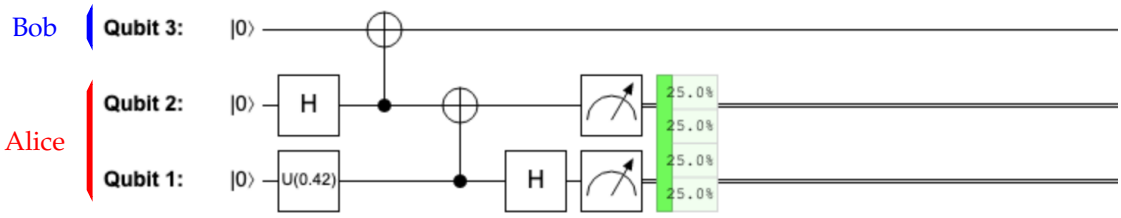
ahol az első két qubit Alízé, az utolsó qubit pedig Botié (emlékezz, hogy a 3.52. egyenlet-ból tudjuk, hogy $|\Phi^+\rangle$ egy kétqubites állapot). A következő QUIRKY áramkör mutatja, hogy ez hogy néz ki abban az esetben, ha Alíz egy $|\psi\rangle = |1\rangle = \text{NOT}\,|0\rangle$ állapotú qubitet akar küldeni:



Mi legyen a következő lépés? Végül is a cél az, hogy Boti megkapja Alíz qubitjét - mivel nem tudjuk klónozni a qubitet, ez azt jelenti, hogy Alíznak valamilyen műveletet kell végrehajtania, ami "megsemmisíti" a saját qubitjét. Ennek jó módja egy mérés végrehajtása. De Alíz két qubitjének egyszerű mérése nem segít rajtuk, mivel tudjuk, hogy egyetlen mérésből nem lehet következtetni $|\psi\rangle$ állapotára. Ez azt jelenti, hogy Alíznak először valamilyen kvantumműveletet kell alkalmaznia mindkét qubitjén, majd meg kell mérnie őket. Kiderül, hogy számára a helyes stratégia az, hogy ugyanazokat a műveleteket hajtsa végre, amelyeket a négy Bell-állapot megkülönböztetésére használtál a 3.13. gyakorló feladat-ben:



Vedd észre, hogy ebben a példában Alíz négy mérési eredménye egyenként 25Itt egy másik példa, amely annak a helyzetnek felel meg, amikor Alíz a $|\psi(0.42)\rangle$ állapotot próbálja teleportálni:



Úgy tűnik, hogy bármi is Alíz üzenetqubitjének állapota, a négy valószínűség mindig ugyanaz. Ez már bíztató, mivel azt jelenti, hogy a mérés egyáltalán nem ad Alíznak információt az üzenetqubitjéről! Emlékezz, hogy azt akarjuk, hogy kezdeti $|\psi\rangle$ állapota teljesen Botinál kössön ki, ami azt jelenti, hogy Alíznak nem szabad semmilyen információt kinyernie vagy megtartania erről az állapotról.

Általánosságban, közvetlenül Alíz mérései előtt az állapot a következőképpen néz ki:

$$(H \otimes I \otimes I)\,(\mathrm{CNOT}_{1\to 2} \otimes I)\,\left(|\psi\rangle \otimes \left|\Phi^+\right\rangle\right)$$

Figyeljünk, mert az utolsó két tenzorszorzat *nincs* egy vonalban (vö. 4.3. gyakorló feladat). Tehát így számoljuk ki:

$$\begin{aligned}
&(H \otimes I \otimes I)\,(\mathrm{CNOT}_{1\to 2} \otimes I)\,\left(|\psi\rangle \otimes \left|\Phi^+\right\rangle\right) \\
&= \frac{1}{\sqrt{2}}\,(H \otimes I \otimes I)\,(\mathrm{CNOT}_{1\to 2} \otimes I)\,(\psi_0\,|000\rangle + \psi_0\,|011\rangle + \psi_1\,|100\rangle + \psi_1\,|111\rangle) \\
&= \frac{1}{\sqrt{2}}\,(H \otimes I \otimes I)\,(\psi_0\,|000\rangle + \psi_0\,|011\rangle + \psi_1\,|110\rangle + \psi_1\,|101\rangle) \\
&= \frac{1}{2}\,(\psi_0\,|000\rangle + \psi_0\,|100\rangle + \psi_0\,|011\rangle + \psi_0\,|111\rangle + \psi_1\,|010\rangle - \psi_1\,|110\rangle + \psi_1\,|001\rangle - \psi_1\,|101\rangle) \\
&= |00\rangle \otimes \frac{\psi_0\,|0\rangle + \psi_1\,|1\rangle}{2} + |01\rangle \otimes \frac{\psi_1\,|0\rangle + \psi_0\,|1\rangle}{2} \\
&+ |10\rangle \otimes \frac{\psi_0\,|0\rangle - \psi_1\,|1\rangle}{2} + |11\rangle \otimes \frac{-\psi_1\,|0\rangle + \psi_0\,|1\rangle}{2}.
\end{aligned}$$

Ez az általános állapot közvetlenül Alíz mérése előtt. Kiszámíthatjuk a mérési eredményeinek valószínűségét a 4.14. egyenlet-ben tárgyaltak szerint. Nevezetesen, az $[ab]$ eredmény $p_{a,b}$ valószínűségének kiszámításához egyszerűen összeadjuk a releváns $|ab0\rangle$ és $|ab1\rangle$ bázisállapotok négyzetezett amplitúdóit. Minden esetben az egyik amplitúdó $\psi_0/2$, a másik pedig $\pm\psi_1/2$, így

négyzetük összege mindig ugyanazt az eredményt adja:

$$
\begin{aligned}
p_{00} &= \left(\frac{\psi_0}{2}\right)^2 + \left(\frac{\psi_1}{2}\right)^2 = \frac{\psi_0^2 + \psi_1^2}{4} = \frac{1}{4}, \\
p_{01} &= \left(\frac{\psi_1}{2}\right)^2 + \left(\frac{\psi_0}{2}\right)^2 = \frac{1}{4}, \\
p_{10} &= \left(\frac{\psi_0}{2}\right)^2 + \left(\frac{-\psi_1}{2}\right)^2 = \frac{1}{4}, \\
p_{11} &= \left(\frac{-\psi_1}{2}\right)^2 + \left(\frac{\psi_0}{2}\right)^2 = \frac{1}{4}.
\end{aligned}
$$

Minden eredmény 25Ez megerősíti, amit korábban megfigyeltünk a QUIRKY-vel.

Alíz mérése után az egyetlen megmaradt kvantumbit Boti qubitje. Jelöljük az állapotát $|\psi'_{a,b}\rangle$-vel, mivel ez függ Alíz mérési eredményétől. Meghatározhatjuk a 4.15. egyenlet segítségével, vagy sokkal egyszerűbben, a 4.13. egyenlet-ban lévő csoportosítási és normalizálási módszerrel. Bárhogy is, az eredmény az, hogy Boti qubitje a következő négy állapot egyikében van:

$$
\begin{aligned}
|\psi'_{00}\rangle &= \psi_0 |0\rangle + \psi_1 |1\rangle, \\
|\psi'_{01}\rangle &= \psi_1 |0\rangle + \psi_0 |1\rangle, \\
|\psi'_{10}\rangle &= \psi_0 |0\rangle - \psi_1 |1\rangle, \\
|\psi'_{11}\rangle &= -\psi_1 |0\rangle + \psi_0 |1\rangle.
\end{aligned}
$$

Amikor $[ab] = [00]$, Boti állapota pontosan megegyezik Alíz eredeti $|\psi\rangle$ állapotával, amit teleportálni akart:

$$
|\psi'_{00}\rangle = |\psi\rangle.
$$

A másik három esetben Boti kvantumállapota kicsit összekeveredett. Ha azonban Alíz elküldi a mérési eredményeit (azaz az a és b két bitet) Botinak, akkor ő megfelelő korrekciós műveletet alkalmazhat az állapota "helyreállítására":

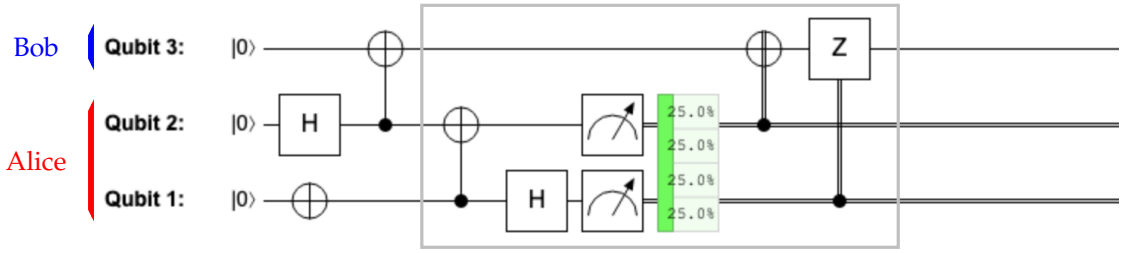
$$
\begin{aligned}
\mathrm{NOT}\,|\psi'_{01}\rangle &= |\psi\rangle, \\
Z\,|\psi'_{10}\rangle &= |\psi\rangle, \\
Z\,\mathrm{NOT}\,|\psi'_{11}\rangle &= |\psi\rangle.
\end{aligned}
$$

Ez a négy eset a következő egyszerű eljárásban foglalható össze Boti számára:

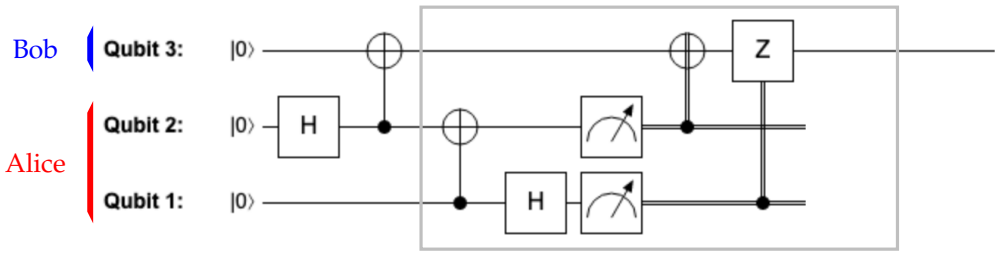
1. nézd meg a b bitet, és ha $b = 1$, akkor alkalmazz NOT-ot,
2. nézd meg az a bitet, és ha $a = 1$, akkor alkalmazz Z-t.

Ezt megvalósíthatjuk egy vezérelt NOT és egy vezérelt Z művelettel, ahol a kontrollok klasszikus bitek. A vezérelt Z műveletet ugyanúgy hozhatod létre, mint a kontrollált NOT műveletet - erről a 3.2.4. alfejezet végén beszéltünk. Hű, ez elég sok munka volt!

Hogyan néz ki mindez a QUIRKY-ben? A végeredmény a következő kvantumáramkör:

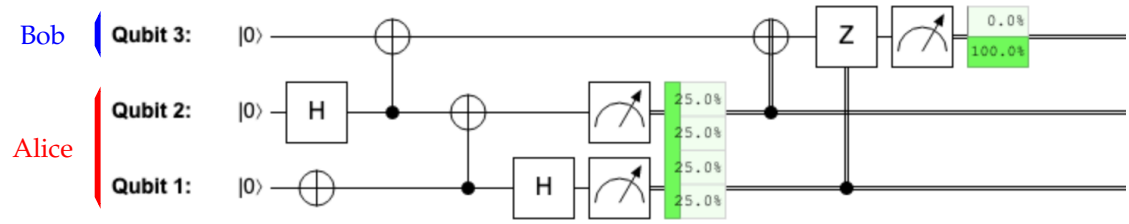


Hozzáadtunk egy szürke dobozt a releváns részhez - a teleportációs áramkörhöz -, hogy elválasszuk az áramkör többi részétől, amely a bemeneti állapotokat hozza létre. Itt van egy kép, amely csak a teleportációs áramkört és a maximálisan összefonódott állapot létrehozását mutatja, Alíz első qubitjének meghatározása nélkül:



Levágtuk Alíz két klasszikus bitjének vezetékeit is, mivel ezek már nem érdekesek, miután Alíz elküldte a két mérési eredményt Botinak. Eltávolítottuk a valószínűségi kijelzőt is, mivel ez nem tényleges kvantumművelet, hanem csak egy módszer az áramkör vizsgálatára a QUIRKY-ben. Így van egy bemeneti qubit Alíz üzenetéhez, két qubit a maximálisan összefonódott állapothoz, és egy kimeneti qubit Boti oldalán. A teleportáció hatása egyszerűen az, hogy egy tetszőleges $|\psi\rangle$ állapotot továbbít a bemeneti qubitről Alíz oldaláról a kimeneti qubitra Boti oldalán. Kulcsfontosságú, hogy a dobozon belül csak klasszikus bitek kerülnek átvitelre Alíztól Botihoz!

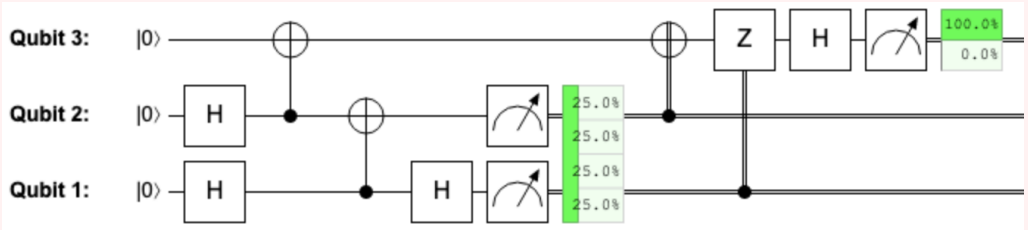
Győződjünk meg róla, hogy nem követtünk el hibát. Mivel azt várjuk, hogy Boti qubitje a $|1\rangle$ állapotba kerül a teleportációs eljárás után, hozzáadhatunk egy egyszerű mérést annak tesztelésére, hogy ez valóban megtörtént-e:



Valóban, 100%-os valószínűséggel kapjuk az 1-es eredményt, ami megerősíti, hogy sikeresen teleportáltuk a $|1\rangle$ állapotot! Most, a $|1\rangle$ nem egy különösen érdekes állapot teleportálásra. Mi a helyzet a $|+\rangle$ állapottal, amely korábban gondot okozott, amikor a klónozásról beszéltünk? A következő házi feladatban először a $|+\rangle$ állapotra, majd tetszőleges egyqubites állapotokra fogod tesztelni a teleportációs áramkört.

4.2. Házi feladat: Teleportáció tesztelése

1. Miért igazolja a következő áramkör, hogy a $|+\rangle$ állapot helyesen lett teleportálva?



2. Hogyan tudnád hasonlóan tesztelni, hogy egy $|\psi(\theta)\rangle$ kvantumállapot helyesen lett-e teleportálva?

A teleportációs áramkör meglehetősen figyelemreméltó. Lehetővé teszi egy qubit küldését Alíztól Botinak mindössze két klasszikus bit átvitelével, feltéve, hogy Alíz és Boti osztoznak egy

maximálisan összefonódott állapoton. Azonban nemcsak az alkalmazások szempontjából hasznos (néhányat alább tárgyalunk), hanem érdekes perspektívát ad a klasszikus és kvantumbitek közötti különbségre is. Emlékezz, hogy a múlt hét végén beszéltünk a szupersűrű kódolásról, amely lehetővé tette két bit küldését egyetlen qubit átvitelével, szintén egy maximálisan összefonódott állapotot használva Alíz és Boti között. Ez azt mutatja, hogy:

Ha korlátlan mennyiségű kvantum összefonódás áll rendelkezésre,
két bit küldése teljesen egyenértékű egy qubit küldésével!

Fontos megjegyezni, hogy sem a teleportáció, sem a szupersűrű kódolás esetén nem tudjuk újrahasználni a maximálisan összefonódott állapotot az eljárás befejezése után - ez az "üzemanyag", amelyet mindkét eljárás elfogyaszt.

4.2.4. Egy pillantás a kvantumhálózatokra

A teleportáció ismételt alkalmazásával kvantumbiteket kommunikálhatunk távoli csomópontok között. Például tegyük fel, hogy Alíz, a robotszamara és Boti a következő helyzetben vannak:

$$\text{Alíz robotszamara} \longleftrightarrow \text{Alíz} \longleftrightarrow \text{Boti},$$

ahol minden „$\leftrightarrow$” nyíl egy maximálisan összefonódott állapotot jelöl. Azaz, három szereplőnk együttes állapota a következő négyqubites állapot:

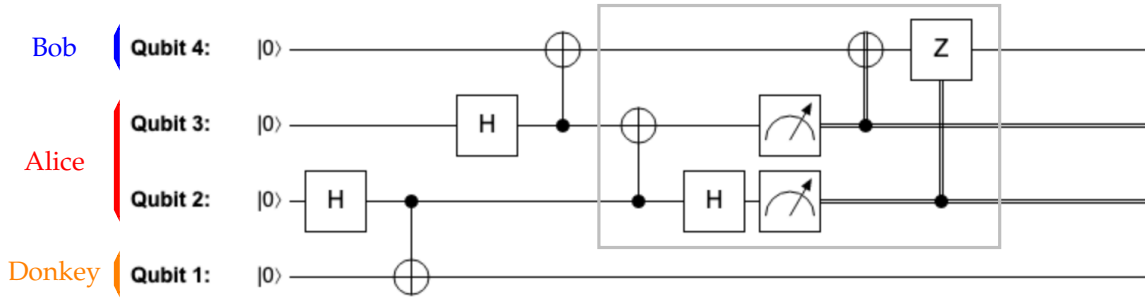
$$\left|\Phi^+\right\rangle \otimes \left|\Phi^+\right\rangle,$$

ahol a középső két qubit Alízé - az első a robottal, a második Botival van összefonódva.

Figyeljük meg, hogy a robotszamár nincs közvetlenül összefonódva Botival! Ennek ellenére, ha kvantumüzenetet kellene küldenie Botinak, ezt meg lehetne tenni a teleportációs eljárás kétszeri futtatásával: először teleportáljuk az üzenetet a robotszamártól Alízhoz (elfogyasztva az első maximálisan összefonódott állapotot), majd Alíztól Botihoz (elfogyasztva a megmaradt maximálisan összefonódott állapotot). Ez hasonló ahhoz, ahogy például a mobiltelefon csatlakozik egy közeli bázisállomáshoz, amely aztán "ismétli" vagy "továbbítja" a jelet egy másik mobiltelefon-toronyhoz (és így tovább). Bár kvantummechanikailag nem másolhatunk egy qubitet a nem-klónozási tétel miatt (lásd 4.1. házi feladat), mégis teleportálhatjuk nagy távolságokra!

Az összefonódás nemcsak teleportációra hasznos, hanem sok más dologra is. Van-e mód arra is, hogy teleportációt használjunk összefonódás létrehozására Alíz robotszamara és Boti között (amit aztán más célokra használhatnának)?

Intuitívan úgy tűnik, Alíznak egyszerűen csak teleportálnia kell az első qubitjét (azt, amelyik a robotszamárral van összefonódva) Botihoz. A QUIRKY-ben ez így nézne ki:



Itt először létrehozzuk a két maximálisan összefonódott állapotot, majd alkalmazzuk ugyanazt a teleportációs áramkört, mint fent (szürke doboz). Intuitívan azt remélhetjük, hogy ez egy maximálisan összefonódott állapotot eredményez a robotszamár és Boti között. A következő házi feladatban megerősítheted, hogy ez valóban így van.

4.3. Házi feladat: Összefonódott qubit teleportálása

Erősítsd meg a QUIRKY használatával, hogy az áramkör végén a szamár qubitje és Boti qubitje a $|\Phi^+\rangle$ maximálisan összefonódott állapotban vannak.

Ugyanígy használhatjuk a teleportációt arra, hogy összefonódást hozzunk létre egyre távolabbi csomópontok között. Például tegyük fel, hogy a következő helyzetben vagyunk:

$$\text{Alíz robotszamara} \longleftrightarrow \text{Alíz} \longleftrightarrow \text{Boti} \longleftrightarrow \text{Boti mókusrobotja.}$$

Ha Alíz először teleportálja az első qubitjét Botihoz, majd Boti ezt követően teleportálja az első qubitjét a mókusrobotjához, ez egy maximálisan összefonódott állapotot eredményez a két robot között. Reméljük, hogy a robotok ezt az összefonódást csak jóindulatú célokra használják!

A nagy távolságokon történő összefonódás létrehozása fontos funkció lesz, amikor megpróbáljuk összekapcsolni a kvantumszámítógépeket egy kis hálózatban, vagy merészen álmodva, egy jövőbeli "kvantuminternetben". Többen közülünk már keményen gondolkodnak azon, hogyan lehet ezt a gyakorlatban megvalósítani, és hogyan lehet a nagy távolságú összefonódást a legjobban felhasználni érdekes alkalmazásokra.

4.2.5. A határozatlansági elv

Az utolsó jelenséghez, amelyet meg szeretnénk tárgyalni, csak egyetlen qubitre lesz szükségünk. Emlékezzünk vissza az egyetlen qubit mérésének szabályaira a 2.6. egyenlet-ból, ahogy a következő képen látható:

Egy $|\psi\rangle = \psi_0 |0\rangle + \psi_1 |1\rangle$ kvantumállapot esetén a két lehetséges kimenet valószínűségei:

$$p_0 = \psi_0^2, \qquad p_1 = \psi_1^2. \tag{4.19}$$

Melyek azok a determinisztikus állapotok, amelyeknél biztosan az egyik kimenetet kapjuk? Ezek olyan állapotok, ahol az egyik valószínűség 100%, a másik 0%. Más szóval, olyan állapotok, ahol az egyik amplitúdó ± 1, a másik nulla. Egy esetleges általános előjeltől eltekintve, amiről tudjuk a 2.7. gyakorló feladat-ból, hogy lényegtelen, csak két ilyen állapot van:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \tag{4.20}$$

azaz a bázisállapotok. Ezek az egyetlen állapotok, amelyeknél tökéletesen megjósolhatjuk a mérési eredményt, így azt mondjuk, hogy nincs bizonytalanság a mérési eredményben.

Mi történik, ha először egy műveletet hajtunk végre a qubiten, majd mérést végzünk? Például tegyük fel, hogy először egy Hadamard-műveletet alkalmazunk, majd mérést végzünk, mint a következő képen:

Itt az egyetlen állapotok, amelyeknél teljesen biztosak vagyunk a mérési eredményben:

$$|+\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \qquad |-\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} \tag{4.21}$$

(egy általános előjeltől eltekintve). Ez azért van, mert a Hadamard-művelet a $|+\rangle$, $|-\rangle$ állapotokat visszaviszi a $|0\rangle$, $|1\rangle$ bázisállapotokba (ezt bemutattad a 4.5. gyakorló feladat-ban); és

ez utóbbi állapotok pontosan azok, amelyeknél teljes bizonyossággal tudjuk a végső mérés eredményét, ahogy fentebb tárgyaltuk. Ezt úgy is láthatjuk, ha kiszámítjuk a két mérési kimenet valószínűségét:

$$q_0 = \frac{(\psi_0 + \psi_1)^2}{2}, \qquad q_1 = \frac{(\psi_0 - \psi_1)^2}{2}. \tag{4.22}$$

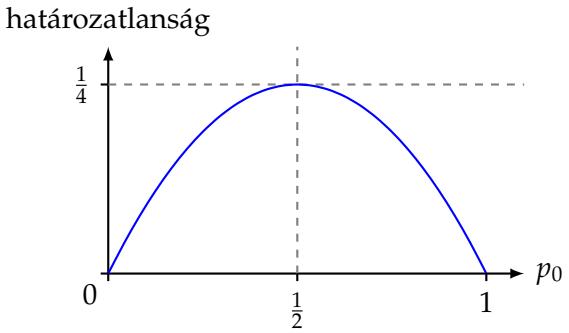
Ha $q_1 = 0$, akkor az állapot $|+\rangle$, míg ha $q_0 = 0$, akkor az állapot $|-\rangle$ (egy általános előjeltől eltekintve).

Ezek mind olyan számítások, amelyeket már többször láttunk - de van egy érdekes megfigyelés, amit eddig még nem tettünk meg. Mivel egyetlen állapot sem jelenik meg *mind* a 4.20. egyenlet-ben, *mind* a 4.21. egyenlet-ben, ez azt jelenti, hogy minden állapotnál legalább az egyik eljárás során lesz valamennyi bizonytalanság a mérési eredményben. Ez az eredmény nem más, mint a híres *Heisenberg-féle határozatlansági elv*!

Tehetjük ezt a megfigyelést kvantitatívabbá? Először szükségünk van egy módszerre, hogy számszerűsítsük a bizonytalanságot vagy "véletlenszerűséget", amit egy $p = \binom{p_0}{p_1}$ valószínűségi eloszlás ad. A következő függvény jó választás:

$$\text{határozatlanság}(p) = p_0(1 - p_0) = p_0 p_1.$$

Ez a $[0, 1/4]$ intervallumba képez értékeket, minimális, ha az egyik kimenet valószínűsége nulla (azaz ha $p_0 = 0$ vagy $p_0 = 1$), és maximális, ha mindkét kimenet egyenlően valószínű (azaz ha $p_0 = p_1 = 1/2$). Itt egy ábra, amely megerősíti ezeket a tulajdonságokat:



Most tegyük fel, hogy egy $|\psi\rangle$ állapottal kezdünk, és a fent leírt két eljárás egyikét hajtjuk végre. Azaz vagy közvetlenül mérjük az állapotot, vagy először alkalmazunk egy Hadamard-műveletet, majd mérünk. A megfelelő bizonytalanságok határozatlanság(p) és határozatlanság(q), ahol a $p = \binom{p_0}{p_1}$ és $q = \binom{q_0}{q_1}$ eloszlásokat a fenti 4.19. és 4.22. egyenletek adja meg. Ekkor:

$$\text{határozatlanság}(p) + \text{határozatlanság}(q) > 0.$$

Valóban, ez az egyenlőtlenség pontosan azt jelenti, hogy nem létezik olyan állapot, amelynél mindkét bizonytalanság egyszerre nulla. A következő házi feladatban egy erősebb eredményt fogsz bemutatni:

4.4. Házi feladat: Bizonytalansági kompromisszum

Mutasd meg, hogy minden $|\psi\rangle$ qubit állapotra:

$$\text{határozatlanság}(p) + \text{határozatlanság}(q) = \frac{1}{4}. \tag{4.23}$$

Továbbá, találj egy olyan $|\psi\rangle$ qubit állapotot, amelyre határozatlanság(p) = határozatlanság(q).

Bónusz kérdés: Állítsd elő ezt az állapotot a QUIRKY segítségével, és erősítsd meg, hogy határozatlanság(p) = határozatlanság(q) a QUIRKY használatával.

[12] A képlet, amit éppen bizonyítottál a 4.4. házi feladat-ben, meglehetősen figyelemreméltó: megmutatja, hogy egyszerű kompromisszum van a két eljárás bizonytalanságai között. Különösen, ha az egyik eljárásnak nulla a bizonytalansága, akkor a másik egyenletesen véletlenszerű eredményt ad![13]

12

[13] Ezt közvetlenül is láthatjuk. Például, ha közvetlenül mérjük a $|+\rangle$ állapotot (egy olyan állapot, amelynek nulla a bizonytalansága a második eljárásban) anélkül, hogy először Hadamard-műveletet végeznénk, egyenlő valószínűséggel kapjuk a 0 és 1 kimeneteket.

4.3. A gyakorló feladatok megoldásai

4.1. Gyakorló feladat megoldása

$$\begin{aligned}
&\left|\Phi^-\right\rangle \otimes \left|\Psi^-\right\rangle \\
&= \left(\frac{1}{\sqrt{2}}\left|00\right\rangle - \frac{1}{\sqrt{2}}\left|11\right\rangle\right) \otimes \left(\frac{1}{\sqrt{2}}\left|01\right\rangle - \frac{1}{\sqrt{2}}\left|10\right\rangle\right) \\
&= \frac{1}{2}\left|0001\right\rangle - \frac{1}{2}\left|0010\right\rangle - \frac{1}{2}\left|1101\right\rangle + \frac{1}{2}\left|1110\right\rangle .
\end{aligned}$$

4.2. Gyakorló feladat megoldása

Először írjuk ki a megadott szorzatállapotot a 4.2. egyenlet alakjában:

$$\left|\Phi^+\right\rangle \otimes \left|1\right\rangle = \frac{1}{\sqrt{2}}\left|001\right\rangle + \frac{1}{\sqrt{2}}\left|111\right\rangle .$$

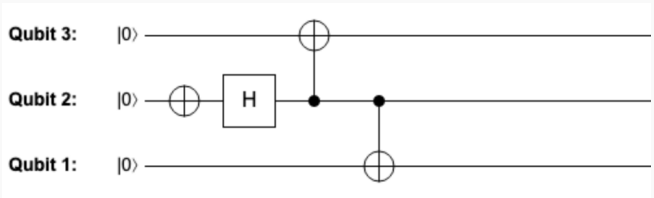
Ebből:

$$\begin{aligned}
&H_2\left(\left|\Phi^+\right\rangle \otimes \left|1\right\rangle\right) = H_2\left(\frac{1}{\sqrt{2}}\left|001\right\rangle + \frac{1}{\sqrt{2}}\left|111\right\rangle\right) = \frac{1}{\sqrt{2}}H_2\left|001\right\rangle + \frac{1}{\sqrt{2}}H_2\left|111\right\rangle \\
&= \frac{1}{\sqrt{2}}\left|0\right\rangle \otimes H\left|0\right\rangle \otimes \left|1\right\rangle + \frac{1}{\sqrt{2}}\left|1\right\rangle \otimes H\left|1\right\rangle \otimes \left|1\right\rangle \\
&= \frac{1}{\sqrt{2}}\left|0\right\rangle \otimes \left|+\right\rangle \otimes \left|1\right\rangle + \frac{1}{\sqrt{2}}\left|1\right\rangle \otimes \left|-\right\rangle \otimes \left|1\right\rangle \\
&= \frac{1}{\sqrt{2}}\left|0\right\rangle \otimes \left(\frac{1}{\sqrt{2}}\left|0\right\rangle + \frac{1}{\sqrt{2}}\left|1\right\rangle\right) \otimes \left|1\right\rangle \; + \; \frac{1}{\sqrt{2}}\left|1\right\rangle \otimes \left(\frac{1}{\sqrt{2}}\left|0\right\rangle - \frac{1}{\sqrt{2}}\left|1\right\rangle\right) \otimes \left|1\right\rangle \\
&= \frac{1}{2}\left|001\right\rangle + \frac{1}{2}\left|011\right\rangle + \frac{1}{2}\left|101\right\rangle - \frac{1}{2}\left|111\right\rangle ,
\end{aligned}$$

ahol a 2.20. egyenlet használtuk a H hatásának kiszámításához az alapállapotokon.

4.3. Gyakorló feladat megoldása

1. A 3.12. gyakorló feladatban láttuk, hogyan készítsünk $\left|\Phi^-\right\rangle$ állapotot. Így a következő áramkör megfelel a célnak:



2. Íme az eredményül kapott állapot:

$$\begin{aligned}
(\mathrm{CNOT}_{2\to 1} \otimes I)(\left|0\right\rangle \otimes \left|\Phi^-\right\rangle) &= (\mathrm{CNOT}_{2\to 1} \otimes I)\left(\frac{1}{\sqrt{2}}\left|000\right\rangle - \frac{1}{\sqrt{2}}\left|011\right\rangle\right) \\
&= \frac{1}{\sqrt{2}}\left|000\right\rangle - \frac{1}{\sqrt{2}}\left|111\right\rangle .
\end{aligned}$$

4.4. Gyakorló feladat megoldása

Legyen

$$|\psi\rangle = \psi_{000}|000\rangle + \psi_{001}|001\rangle + \psi_{010}|010\rangle + \psi_{011}|011\rangle + \psi_{100}|100\rangle + \psi_{101}|101\rangle + \psi_{110}|110\rangle + \psi_{111}|111\rangle$$

egy tetszőleges három-qubit kvantumállapot. A Toffoli-művelet alkalmazásának eredménye:

$$|\psi'\rangle = T|\psi\rangle = \psi_{000}|000\rangle + \psi_{001}|001\rangle + \psi_{010}|010\rangle + \psi_{011}|011\rangle + \psi_{100}|100\rangle + \psi_{101}|101\rangle + \psi_{110}|\mathbf{111}\rangle + \psi_{111}|\mathbf{110}\rangle.$$

A megváltozott két bázisállapotot félkövérrel emeltük ki. Vedd észre, hogy az egyetlen változás az, hogy a $|110\rangle$ és $|111\rangle$ amplitúdói felcserélődtek. Így világos, hogy ha $\sum_{a,b,c=0}^{1} \psi_{a,b,c}^2 = 1$, akkor $\sum_{a,b,c=0}^{1} (\psi'_{a,b,c})^2 = 1$ is teljesül. Tehát a T kvantumállapotokat képez le kvantumállapotokra.

4.5. Gyakorló feladat megoldása

1. Ha H-t alkalmazzuk $|+\rangle$-ra és $|-\rangle$-ra, a következőt kapjuk:

$$H|+\rangle = H\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = \frac{1}{2}\big((|0\rangle + |1\rangle) + (|0\rangle - |1\rangle)\big) = |0\rangle,$$
$$H|-\rangle = H\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = \frac{1}{2}\big((|0\rangle + |1\rangle) - (|0\rangle - |1\rangle)\big) = |1\rangle.$$

2. Az azonosság igazolásához csak azt kell ellenőriznünk, hogy a HZH ugyanúgy hat a bázis vektorokra, mint a NOT (linearitás miatt ez azt jelentené, hogy minden qubit állapotra ugyanúgy hatnak):

$$HZH|0\rangle = HZ\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = H\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = H|-\rangle = |1\rangle,$$
$$HZH|1\rangle = HZ\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = H\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = H|+\rangle = |0\rangle.$$

   Mindkét esetben a HZH invertálja a bitet, tehát ugyanazt a műveletet hajtja végre, mint a NOT.

3. A feladat első része megmutatta, hogy a Hadamard-kapu kétszeri alkalmazása nem csinál semmit: $HH = I$. Így,

$$Z = (HH)Z(HH) = H(HZH)H = H \text{ NOT } H$$

   ahol az utolsó lépés a feladat 2. részéből következik.

4.6. Gyakorló feladat megoldása

1. A 2.15. egyenlet használatával,

$$U(\theta_2)U(\theta_1)\left|\psi(\alpha)\right\rangle = U(\theta_2)\left|\psi(\alpha+\theta_1)\right\rangle = \left|\psi(\alpha+\theta_1+\theta_2)\right\rangle = U(\theta_1+\theta_2)\left|\psi(\alpha)\right\rangle.$$

   Ez bármely $|\psi(\alpha)\rangle$ állapotra igaz, ami azt jelenti, hogy $U(\theta_2)U(\theta_1) = U(\theta)$, ahol $\theta = \theta_1 + \theta_2$. Ez geometriailag is világos: ha először θ_1 szöggel, majd θ_2 szöggel forgatunk, az összességében $\theta = \theta_1 + \theta_2$ szögű forgatásnak felel meg.

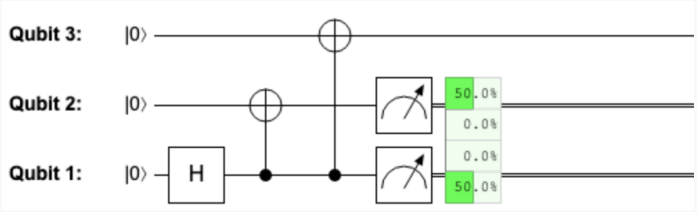
2. Íme egy elegáns megoldás. Emlékezzünk vissza a 2.19. egyenletből, hogy egy általános tükrözést kétféleképpen fejezhetünk ki: $V(\theta) = \mathrm{NOT}\,U(\theta) = U(-\theta)\,\mathrm{NOT}$. Ha $V(\theta_1)$-re az első kifejezést, $V(\theta_2)$-re pedig a második kifejezést használjuk, akkor azt kapjuk, hogy

$$V(\theta_2)V(\theta_1) = U(-\theta_2)\,\mathrm{NOT}\,\mathrm{NOT}\,U(\theta_1) = U(-\theta_2)U(\theta_1) = U(\theta_1-\theta_2),$$

   ahol felhasználtuk azt a tényt, hogy két egymást követő NOT alkalmazása nem csinál semmit, és hogy két egymást követő forgatás egyetlen forgatásnak felel meg, ahol a két szög összeadódik, ahogy fent is láttuk.

4.7. Gyakorló feladat megoldása

A 4.14. egyenlet szerint [00] és [11] eredményt kellene kapnunk, mindkettőt 50% valószínűséggel. Valóban:



5. Küldetés: kvantumalgoritmusok meghódítása

A kvantumszámítógépek tanulmányozásának egyik fő motivációja az a tény, hogy a kvantumszámítógépek néhány problémát sokkal gyorsabban tudnak megoldani, mint a jelenleg használt számítógépeink. A kvantumszámítógépek és a hagyományos számítógépek megkülönböztetésére a klasszikus számítógép általános kifejezést fogjuk használni minden olyan számítási eszközre, amelyet jelenleg használunk. Ez magában foglalja a laptopot vagy asztali számítógépet, de a nagyon kicsi számítógépeket is, mint például az okostelefonodban vagy okosórádban lévőket, valamint a nagyon nagy és erős *szuperszámítógépeket*, amelyek akár egy egész szobát is elfoglalhatnak. Ami ezeket a számítógépeket megkülönbözteti a kvantumszámítógépektől, az nem az, hogy mennyire nagyok, kicsik, lassúak vagy gyorsak, hanem az a tény, hogy belső működésüket a klasszikus fizika írja le (pontosabban az elektromágnesség). Más szóval, a hardverük olyan módon működik, amely leírható a kvantummechanikát megelőző régi fizikai elméletekkel. Ez egy kicsit hasonlít ahhoz, ahogy Mozart és Bach zenéjét *klasszikus zenének* nevezzük. Akárcsak a klasszikus számítógépek, a klasszikus zene sem használja ki teljes mértékben a fejlettebb hangszereket, mint például az elektromos gitárt vagy a szintetizátorokat.

A klasszikus számítógépek hardverének következményeként minden általuk tárolt információ – legyen az kép, hangfájl, videó vagy weblap – bitsorozatokkal, azaz nullák és egyesek hosszú sorozataival van ábrázolva. Ezt az információt olyan szabályok követésével dolgozzák fel, amelyek leírják, hogyan kell ezeket a nullákat és egyeseket módosítani, hogy hasznos választ kapjunk. Ezt az utasítássorozatot algoritmusnak nevezzük. Az algoritmusra úgy gondolhatsz, mint egy receptre – olyan utasítások sorozata, amelyeket ha gondosan követsz, megkapod a kívánt eredményt, például egy csokis brownie-t! Például egy algoritmus két bináris stringet vehet bemenetként, és egy másik bináris stringet állíthat elő kimenetként, amely tartalmazza a két eredeti string összegét (amikor számként értelmezzük őket). Akárcsak a recepteket, az algoritmusokat eredetileg emberek hajtották végre. Valójában a „számítógép” szó régen egy olyan személyre utalt, aki számításokat végzett. Manapság azonban az algoritmusokat valódi számítógépeken futtatják. Mivel a számítógépek általában nem túl okosak, az algoritmusokat rendkívül precíz módon kell leírni számukra. Ez a leírás, egy program, az absztrakt algoritmus konkrét megvalósítása olyan módon, hogy a számítógép megértse. Ehhez valamilyen programozási nyelvet kell használnunk, amelyet a számítógép képes lefordítani elemi műveletekké a nullákon és egyeseken, majd végrehajtani őket a tényleges hardveren.

Amikor egy algoritmust tervezel, beprogramozod a számítógépedbe és futtatod, azt szeretnéd, hogy ésszerű időn belül megkapd a választ. A válasz megszerzéséhez szükséges tényleges idő azonban sok tényezőtől függ:

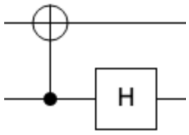
1. milyen gyorsan hajtja végre a számítógéped a különböző elemi utasításokat,
2. a program bemenete merevlemezről, hálózati kapcsolatról vagy közvetlenül a memóriából olvasódik-e,
3. milyen programozási nyelvet használsz a program megírásához (és a programot futtató fordító vagy értelmező verziója),

és így tovább. Ez nagyon megnehezíti a különböző algoritmusok összehasonlítását.

Hogy a különböző algoritmusok összehasonlítása könnyebb és igazságosabb legyen, a számítógép-tudósok nem azt nézik, mennyi ideig tart az algoritmus futtatása egy adott módon konfigurált specifikus számítógépen. Ehelyett megszámolják, hány elemi műveletet hajt végre az algoritmus. Így biztosak lehetnek abban, hogy magát az algoritmust hasonlítják össze, nem pedig azokat a számítógépeket, amelyeken az algoritmusok futnak (valóban, egy jó algoritmus, amely egy nagyon lassú számítógépen fut, rosszabbnak tűnhet, mint egy rossz algoritmus, amely egy nagyon gyors számítógépen fut). Pontosabban, amit a számítógép-tudósok tudni

akarnak, az az, hogy hogyan nő a műveletek száma az algoritmusnak megoldandó probléma *méretével*. Valóban, minél nagyobb mennyiségű adatot kell feldolgoznod, annál tovább tart, függetlenül attól, mennyire jó az algoritmus. Tehát azt szeretnéd tudni, hogy az algoritmusod képes lesz-e még mindig megbirkózni a feladattal, amikor az adatok rendkívül naggyá válnak. A számítógép-tudománynak azt a területét, amely ezt tanulmányozza, számítási komplexitásnak nevezzük.

A kvantumszámítástechnikában arra törekszünk, hogy olyan kvantumalgoritmusokat tervezzünk, amelyek számítási problémákat oldanak meg kvantumállapotok manipulálásával bitsorozatok helyett. Az általunk használt elemi utasítások a kapuk lesznek, mint például a Hadamard-kapu, a vezérelt NOT kapu vagy egy mérés. Kvantumalgoritmusainkat képi formában fogjuk meghatározni egy kvantumáramkör segítségével, amellyel már sok tapasztalatod van a QUIRKY-ban. De használhatnánk szöveges ábrázolást is, ahogy azt egy hagyományos számítógépes programtól várnád. Például a bal oldali áramkört a jobb oldali programszöveg reprezentálhatná:




```
CNOT q1 -> q2;
H q1;
```

ahol a `q1` és `q2` a két qubitet jelöli (emlékezz, hogy az alsó vezeték felel meg az *első* qubitnek). Egy számítási probléma esetén összehasonlíthatjuk az elemi utasítások számát, amelyeket a legjobb ismert kvantum- és klasszikus algoritmusok használnak a megoldásához. Így pontos képet kaphatunk a jövőbeli kvantumszámítógépek által nyújtott előnyökről.

5.1. Beszélgetés orákulumokkal

Ebben a küldetésben több kvantumalgoritmusra is rápillantunk, és megnézzük, hogyan viszonyulnak az ugyanazt a problémát megoldó klasszikus algoritmusokhoz. Mivel általában nagyon nehéz megérteni, hogy hány elemi művelet szükséges egy adott számítási probléma megoldásához, ebben a küldetésben az algoritmus komplexitásának egy egyszerűbb mértékét fogjuk vizsgálni (ezt teszik a számítógép-tudósok is, amikor egy kicsit egyszerűbbé akarják tenni az életüket).

Amikor egy számítógép egy programot futtat, többféle típusú műveletet kell végrehajtania. A leglassabb típusú műveletek azok, amelyeknek adatokhoz kell hozzáférniük. Például amikor a memóriából, a merevlemezről vagy – a legrosszabb esetben – egy másik, interneten keresztül elérhető számítógépről olvasnak. Miután a releváns adatdarabot beolvasta, annak feldolgozása viszonylag gyors lehet. Emiatt nagyjából el tudjuk képzelni, hogy mennyi ideig fog futni egy adott algoritmus, ha csak azokat az utasításokat számoljuk, amelyek adatokhoz férnek hozzá.

Ezt talán ismered is, amikor egy bonyolult weboldalt vagy nagy dokumentumot nyitsz meg. Eltarthat egy ideig, amíg betöltődik, de amint betöltődött, a weboldallal való interakció vagy a dokumentumban való görgetés és egy újabb sor beszúrása általában elég gyors.

Egy másik módja ennek az elgondolásnak, amit ebben a küldetésben használni fogunk, az, hogy az információ, amihez hozzá próbálsz férni, valójában egy másik algoritmus vagy egy szubrutin által jön létre az algoritmusodon belül. Ráadásul ez a szubrutin nagyon lassú. Például lehet, hogy az információt a merevlemezről olvassa be vagy az interneten keresztül fér hozzá. Vagy lehet, hogy nincs is azonnal rendelkezésre álló válasza, és ehelyett nulláról kell előállítania egy nagyon bonyolult számítás elvégzésével. Akárhogy is, ez a szubrutin mindig nagyon sokáig tart, amíg kitalálja a választ, így a lehető legkevesebbszer szeretnéd meghívni. Az ilyen szubrutint orákulumnak nevezzük, mivel nagyon bölcs és minden választ ismer, de egy kicsit lassú is, ezért időbe telik, amíg átgondolja és megadja a választ.

Formálisabban, egy klasszikus orákulum nem más, mint egy $f\colon \{0,1\}^n \to \{0,1\}$ függvény, ahol $\{0,1\}^n$ az összes n-bites bináris string halmazát jelöli. A függvény $x \in \{0,1\}^n$ bemenetére

gondolhatsz kérdésként, a kimenetre, az $f(x)$ bitre pedig válaszként. Minden alkalommal, amikor kiértékeled az f függvényt valamilyen bemenetre, egy olyan kérdést teszel fel, amely megfelel annak a bemenetnek, és egy igen/nem választ kapsz, amely megfelel a függvény értékének.

Például modellezhetjük a merevlemezhez vagy memóriához való hozzáférést egy orákulummal. Mondjuk, ha 4 bit memóriád van, modellezhetjük őket egy $f : \{0,1\}^2 \to \{0,1\}$ függvénnyel, amely egy bináris címet kap, és visszaadja a megfelelő bitet. Például, ha mind a négy bitet meg akarod tudni, négyszer kellene kiértékelned f-et, hogy megkapd a négy értéket: $f(00), f(01), f(10), f(11)$.

Fontos megjegyezni, hogy a mi felállásunkban azok a számítási problémák, amelyeket meg szeretnénk oldani, *nem* az f értékének megtalálásáról szólnak egy adott bemenetre. Valóban, az ilyen problémák triviálisak, mivel megoldhatók az orákulum egyszeri konzultálásával, mert pontosan ezt csinálja az orákulum – megmondja neked a függvény értékét bármilyen általad választott bemenetre. Ezért a minket érdeklő problémák összetettebbek. Amit szeretnénk, az az f függvény valamilyen tulajdonságának meghatározása a lehető legkevesebb kiértékeléssel.

Például tegyük fel, hogy azt szeretnénk tudni, hogy $f(x) = 0$ minden $x \in \{0,1\}^n$ esetén. Ebben az esetben azt tehetnénk, hogy az orákulumtól x véletlenszerű értékeit kérdezzük, amíg nem találunk egyet, amire $f(x) = 1$. Hamarosan más példákat is látni fogunk az ilyen problémákra.

5.1.1. Reverzibilis számítás

Mielőtt teljesen belemerülnénk az izgalmas új kvantumalgoritmusok keresésébe, először győződjünk meg arról, hogy kvantumszámítógépen továbbra is ki tudunk számítani mindent, amit egy hagyományos számítógépen is tudunk. Más szóval, először bizonyosodjunk meg arról, hogy a kvantumszámítógépek valójában nem *kevésbé* erősek, mint a hagyományos (klasszikus) számítógépek! Ez nem teljesen nyilvánvaló, mivel a kvantumszámítógépek működése nagyon különbözik. Különösen az, hogy a kvantumszámítógépen minden művelet reverzibilis vagy invertálható, ahogy azt már említettük a 2.4.2. alfejezet és 4.1.3. alfejezet részekben, de ez általában nem igaz egy hagyományos számítógépre. Ki nem törölt még véletlenül fájlt, vagy felejtette el elmenteni a változtatásokat a dokumentumában, elveszítve az összes munkáját? Ha minden művelet reverzibilis lenne, soha nem kellene aggódnod ilyen triviális dolgok miatt.

Ez felveti a következő kérdést: hogyan láthatjuk, hogy a kvantumszámítógépek mindent ki tudnak számítani, amit a hagyományos számítógépek? Egy módja ennek az, ha megmutatjuk, hogy a reverzibilitás valójában nem korlátozza azt, amit egy hagyományos számítógép ki tud számítani, és így ez nem jelent korlátozást a kvantumszámítógépek számára sem. Más szóval, megmutatjuk, hogy bármely számítás reverzibilissé tehető, és így futtatható kvantumszámítógépen.

Hogy képet kapjunk arról, hogyan lehet ezt megvalósítani, nézzük meg a két bit logikai ÉS függvényének egyszerű példáját. Ha mindkét bit 1, az ÉS 1-et ad eredményül, egyébként 0-t. Tehát az ÉS függvényt a következő függvénytáblával ábrázolhatjuk:

x_1	x_2	$\mathrm{AND}(x_1, x_2)$
0	0	0
0	1	0
1	0	0
1	1	1

(5.1)

A 3.1. alfejezet jelölését használva ezt matematikailag a következő műveletként írhatjuk le két biten:

$$[x_1, x_2] \mapsto [\mathrm{AND}(x_1, x_2)].$$

Nyilvánvaló, hogy ez a művelet nem reverzibilis, mert nincs ugyanannyi kimeneti bitje, mint bemeneti bitje. Valóban, ha csak azt tudod, hogy két bit ÉS művelete 0, akkor nem tudod pontosan meghatározni a két bit állapotát – ahogy a függvénytábla mutatja, három lehetséges opció van.

Hogyan orvosolhatjuk ezt a problémát? Próbáljuk meg megtartani az első bitet, és vezessünk be egy második kimeneti bitet, amelyben a választ tároljuk:

$$[x_1, x_2] \mapsto [x_1, \mathrm{AND}(x_1, x_2)].$$

Ez jobb, mivel most két bitet képezünk le két bitre. De vajon reverzibilis-e? Azaz a kimenetből mindig rekonstruálhatjuk-e a bemenetet? Nos, biztosan rekonstruálhatjuk a bemenet első bitjét, x_1-et, mivel az a kimenetben is megvan. De mi a helyzet x_2-vel? Ha $x_1 = 0$, akkor a 5.1. egyenlet szerint a kimenet $[00]$ – függetlenül x_2 értékétől. Ez azt jelenti, hogy ismét nem tudjuk mindig rekonstruálni a bemenetet, így ez a megközelítés sajnos szintén nem működik.

Ez kezd reménytelennek tűnni. Egyáltalán lehetséges-e az ÉS függvényt reverzibilis módon megvalósítani? Amikor elakadsz, ki kell lépned a dobozból! Ebben az esetben ki mondta, hogy csak két bitre kell korlátozódnunk? Ha mindkét bemeneti bitet megtartjuk, és bevezetünk egy harmadik bitet a válasz tárolására, ez biztosan reverzibilissé teszi a műveletet:

$$[x_1, x_2, 0] \mapsto [x_1, x_2, \mathrm{AND}(x_1, x_2)]. \tag{5.2}$$

Most már nincs probléma a művelet megfordításával – bármilyen kimeneti bitsorozat esetén egyszerűen elfelejthetjük, mit tartalmaz az utolsó bit, és helyettesíthetjük $[0]$-val, hogy visszakapjuk a bemeneti bitsorozatot. Például, ha a kimenet $[111]$, akkor a bemenetnek $[110]$-nak kellett lennie.

Tehát készen vagyunk? Nem olyan gyorsan! Vegyük észre, hogy a 5.2. egyenlet csak *részlegesen* határozza meg a műveletet – ha a bemenet $[111]$, vagy bármely más 1-re végződő bitsorozat, akkor a 5.2. egyenlet *nem* mondja meg, hogyan kell a műveletnek erre a bemenetre hatnia. Mivel a négy 0-ra végződő bemeneti sztringet négy különböző kimeneti sztringre képezzük le, világos, hogy a 5.2. egyenlet-t valamilyen tetszőleges módon kiterjeszthetjük egy három bites reverzibilis műveletre. De van-e valamilyen szisztematikus módja ennek?

Ehhez vegyük észre, hogy a 5.2. egyenlet az utolsó bitet $[0]$-ról $[1]$-re váltja, ha $\mathrm{AND}(x_1, x_2) = 1$. Hasonlóképpen, ha az utolsó bit történetesen $[1]$, egyszerűen definiálhatnánk a műveletünket úgy, hogy visszaváltsa $[0]$-ra, amikor $\mathrm{AND}(x_1, x_2) = 1$. Más szóval, kiterjeszthetjük a 5.2. egyenlet-t *minden* lehetséges bemeneti sztringre a következőképpen:

$$[x_1, x_2, y] \mapsto [x_1, x_2, y \oplus \mathrm{AND}(x_1, x_2)], \tag{5.3}$$

minden $x_1, x_2, y \in \{0, 1\}$ esetén, ahol '$\oplus$' a modulo 2 összeadást jelöli.

A 5.3. egyenlet művelet most már minden lehetséges bemenetre definiálva van. De végre reverzibilis-e? Igen, az! Valójában ez a művelet a saját inverze! Azaz ha kétszer végezzük el a műveletet, visszakapjuk az eredeti bemenetet:

$$\begin{aligned}[x_1, x_2, y] &\mapsto [x_1, x_2, y \oplus \mathrm{AND}(x_1, x_2)] \\ &\mapsto [x_1, x_2, y \oplus \mathrm{AND}(x_1, x_2) \oplus \mathrm{AND}(x_1, x_2)] = [x_1, x_2, y].\end{aligned}$$

A harmadik lépésben felhasználtuk, hogy $a \oplus a = 0$ bármely $a \in \{0, 1\}$ esetén, lásd 3.20. egyenlet. Tehát sikeresen találtunk egy módot az ÉS függvény reverzibilis kiszámítására.

5.1.2. Bit-orákulumok

Ez az ötlet nemcsak a logikai AND függvényre működik, hanem valójában bármely függvényre

$$f\colon \{0, 1\}^n \to \{0, 1\}$$

amely n bitet vesz bemenetként és egyetlen bitet ad vissza. Bármely ilyen f függvényre definiálhatunk egy reverzibilis műveletet $(n+1)$ biten:

$$[x_1, \ldots, x_n, y] \mapsto [x_1, \ldots, x_n, y \oplus f(x_1, \ldots, x_n)] \tag{5.4}$$

minden $x_1, \ldots, x_n, y \in \{0, 1\}$ esetén. Csakúgy, mint a 5.3. egyenlet, ez a művelet is a saját inverze, azaz kétszer alkalmazva egyenértékű azzal, mintha semmit sem csinálnánk.

Megnyugtató, hogy bármely $f : \{0,1\}^n \to \{0,1\}$ függvényt reverzibilisen implementálhatunk, ahogy azt a 5.4. egyenlet mutatja. Először is, ez azt jelenti, hogy bármely klasszikus számítógépen végzett számítás elvégezhető úgy, hogy visszanyerhetjük az eredeti bemenetet. Másodszor, ha a számítást reverzibilissé tettük, kvantumszámítógépen is futtathatjuk. Ez azt jelenti, hogy a kvantumszámítógépek mindent ki tudnak számítani, amit a klasszikus számítógépek! Harmadszor, ez azt jelenti, hogy bármely f függvényt implementálhatunk orákulumként egy kvantumszámítógépen.

Nézzük meg, hogyan is működik ez valójában. A 5.4. egyenlet-ben szereplő művelet kvantumváltozata a következőképpen definiált:

$$U_f \left|x_1, \ldots, x_n, y\right\rangle = \left|x_1, \ldots, x_n, y \oplus f(x_1, \ldots, x_n)\right\rangle \tag{5.5}$$

minden $x_1, \ldots, x_n, y \in \{0, 1\}$ esetén. A 5.5. egyenlet definiálja, hogyan hat U_f az $n+1$ qubit összes bázisállapotára. Mint általában, ezt a definíciót kiterjesztjük egy tetszőleges $n+1$ qubites állapotra a linearitás révén. Mivel U_f egyszerűen permutálja a bázisállapotokat, ellenőrizheted, ahogy a 4.4. gyakorló feladat-ban, hogy ez egy érvényes kvantumműveletet definiál.

A 5.5. egyenlet-ben definiált U_f kvantumműveletet *f bit-orákulumának* nevezzük (a 5.4. egyenlet-ben szereplő függvényt is nevezhetnénk klasszikus bit-orákulumnak, de erre a névre nem lesz szükségünk). Az "orákulum" kifejezés egyszerűen azt jelenti, hogy ennek a műveletnek az alkalmazása olyan, mintha egy mindenható orákulumot kérdeznénk meg, hogy mondja meg nekünk a függvény értékét bármely adott bemenetre. Nem tudjuk pontosan, hogyan van implementálva az orákulum, sem azt, honnan szerzi a válaszát, egyszerűen csak számoljuk, hány kérdést kell feltennünk az orákulumnak, hogy megtudjunk valamilyen tulajdonságot az f függvényről. Sok érdekes algoritmikus probléma modellezhető ilyen módon, ahogy azt a küldetés hátralévő részében látni fogjuk.

Az orákulum koncepciója kicsit hasonlít a "találd ki a számomat" játékhoz. A barátod (az orákulum) kitalál egy számot, és te olyan kérdéseket teszel fel neki, hogy "a számod x"? A barátod minden ilyen kérdésre "igen" vagy "nem" választ ad. Más szóval, a barátod egy olyan függvényt rejt, amely minden bemenetre "nem"-et ad, kivéve egyet (azt a számot, amelyre gondol). Minden kérdéssel, amit felteszel, több információt szerzel arról, hogy melyik szám lehet az. Elgondolkodhatsz azon, hány kérdést kell feltenned a szám meghatározásához. Sőt, mi lenne, ha a kérdéseidet egy kvantum orákulumnak tehetnéd fel, mint például az U_f, a barátod helyett? Kitalálhatnád a választ kevesebb kérdéssel? A hét küldetésében több érdekes példát is látni fogunk, ahol ez valóban így van!

Nézzünk meg néhány példát a bit-orákulumokra. Tegyük fel, hogy f az ÉS függvény. A 5.1. egyenlet szerint az ÉS-t értelmezhetjük modulo 2 szorzásként, mivel $\mathrm{AND}(x_1, x_2) = x_1 x_2$. A megfelelő bit-orákulum

$$U_{\mathrm{AND}} \left|a, b, c\right\rangle = \left|a, b, c \oplus ab\right\rangle$$

nem más, mint a Toffoli-kapu a 4.4. gyakorló feladat-ból. Még $n = 1$ esetén és az $f(x) = x$ függvényre is, amely egyszerűen visszaadja a bemenetét, érdekes eredményt kapunk: minden $a, b \in \{0, 1\}$ esetén

$$U_f \left|a, b\right\rangle = \left|a, b \oplus a\right\rangle .$$

Ez nem más, mint az ismerős vezérelt NOT kapu $\text{CNOT}_{1\to 2}$ a 3.47. egyenlet-ból a 3.2.4. alfejezet-ben! Tehát a bit-orákulum konstrukció reprodukál több érdekes kvantumműveletet, amelyeket korábban kézzel definiáltunk. A következő feladatban megpróbálhatod implementálni az összes többi bit-orákulumot egyetlen bit függvényeire.

5.1. Gyakorló feladat: Bit-orákulum egy egybit függvényhez

Legyen $f : \{0,1\} \to \{0,1\}$ egy függvény egyetlen be- és kimeneti bittel. Egy ilyen függvényt teljesen meghatároznak az $f(0), f(1) \in \{0,1\}$ értékek. Ez két bit, tehát pontosan négy ilyen függvény létezik. Épp most beszéltük meg, hogyan implementáljuk az U_f bit-orákulumot az $f(x) = x$ függvényre. Meg tudod implementálni az U_f bit-orákulumot a másik három függvényre QUIRKY-ben?

5.1.3. Előjel-orákulumok

Mivel a bit-orákulum U_f egy kvantumművelet, nemcsak $|x_1, \ldots, x_n, y\rangle$ bázisállapotokra alkalmazhatjuk, hanem általános kvantumállapotokra is. Miért szeretnénk ilyet tenni? Nos, ha mindig csak "klasszikus" kérdéseket teszünk fel az orákulumnak, akkor kevés esély van kvantum gyorsítás elérésére! Ezt a motivációt szem előtt tartva, vizsgáljuk meg, hogyan viselkedik a bit-orákulum U_f egy tetszőleges $f : \{0,1\}^n \to \{0,1\}$ függvényre, amikor az utolsó regisztert $|-\rangle = (|0\rangle - |1\rangle)/\sqrt{2}$-re állítjuk. Először is, vegyük észre a következő érdekes tényt:

$$\text{NOT}\,|-\rangle = \frac{1}{\sqrt{2}}(\text{NOT}\,|0\rangle - \text{NOT}\,|1\rangle) = \frac{1}{\sqrt{2}}(|1\rangle - |0\rangle) = -\,|-\rangle\,.$$

Azaz, ha invertálunk egy $|-\rangle$ állapotú qubitet, akkor egy előjelet kapunk. Hasonlóképpen kiszámíthatjuk, hogyan hat a bit-orákulum egy $|x_1, \ldots, x_n\rangle \otimes |-\rangle$ alakú állapotra. A linearitás miatt a 5.5. egyenlet alapján:

$$\begin{aligned}
&U_f\left(|x_1, \ldots, x_n\rangle \otimes |-\rangle\right) \\
&= U_f\Big(\frac{1}{\sqrt{2}}\,|x_1, \ldots, x_n, 0\rangle - \frac{1}{\sqrt{2}}\,|x_1, \ldots, x_n, 1\rangle\Big) \\
&= \frac{1}{\sqrt{2}}\,|x_1, \ldots, x_n, f(x_1, \ldots, x_n)\rangle - \frac{1}{\sqrt{2}}\,|x_1, \ldots, x_n, f(x_1, \ldots, x_n) \oplus 1\rangle \\
&= |x_1, \ldots, x_n\rangle \otimes \frac{1}{\sqrt{2}}\big(|f(x_1, \ldots, x_n)\rangle - |f(x_1, \ldots, x_n) \oplus 1\rangle\big) \\
&= (-1)^{f(x_1,\ldots,x_n)}\,|x_1, \ldots, x_n\rangle \otimes |-\rangle\,.
\end{aligned}$$

Másképp megfogalmazva, az utolsó qubitet visszakapjuk a $|-\rangle$ állapotban, de egy általános mínusz előjelet kapunk, amikor $f(x_1, \ldots, x_n) = 1$. Lényegében a következő kvantumműveletet hajtottuk végre az első n qubiten:

$$O_f\,|x_1, \ldots, x_n\rangle = (-1)^{f(x_1,\ldots,x_n)}\,|x_1, \ldots, x_n\rangle\,, \tag{5.6}$$

minden $x_1, \ldots, x_n \in \{0,1\}$ bitsorozatra. Az O_f-et az f előjel-orákulumának nevezzük.

Érdekes módon egy $f\colon \{0,1\}^n \to \{0,1\}$ függvény esetén az előjel-orákulum O_f n qubiten működik, mivel a kimenetet az amplitúdóban tárolja. Ez különbözik a bit-orákulumtól U_f, amely a kimenetet egy további qubitben tárolja, és ezért $n+1$ qubiten működik.

Első ránézésre úgy tűnhet, hogy az előjel-orákulum nem csinál sokat, mivel csak egy általános előjelet vezet be, amikor egy bázisállapotra hat, amiről a 2.7. gyakorló feladat alapján tudjuk, hogy nem figyelhető meg. Azonban, amikor szuperpozícióra alkalmazzuk, akkor az

előjel-orákulum relatív előjeleket vezethet be, így érdekes eredményt kaphatunk. Például $n = 2$ qubit esetén, ha egy általános kétqubites állapotra alkalmazzuk az O_f előjel-orákulumot

$$|\psi\rangle = \psi_{00}|00\rangle + \psi_{01}|01\rangle + \psi_{10}|10\rangle + \psi_{11}|11\rangle$$

akkor a következőt kapjuk:

$$O_f|\psi\rangle = (-1)^{f(0,0)}\psi_{00}|00\rangle + (-1)^{f(0,1)}\psi_{01}|01\rangle + (-1)^{f(1,0)}\psi_{10}|10\rangle + (-1)^{f(1,1)}\psi_{11}|11\rangle$$

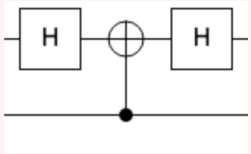
Mint kiderül, az előjel-orákulum O_f valóban hasznos, és gyakran sokkal könnyebben alkalmazható kvantumalgoritmusokban, mint a bit-orákulum U_f, így mostantól nem fogjuk többé használni a bit-orákulumot.

5.2. Gyakorló feladat: Előjel-orákulum egy egybit függvényhez

Emlékezzünk vissza a 5.1. gyakorló feladat-ból, hogy négy olyan $f : \{0,1\} \to \{0,1\}$ függvény létezik, amelynek egy bemeneti és egy kimeneti bitje van. Meg tudod valósítani mindegyikük O_f előjel-orákulumát QUIRKY-ben?

5.1. Házi feladat: Határozd meg a függvényt az előjel-orákulumából

Tekintsük a következő kétqubites áramkört (mint általában, az alsó vezeték az első qubit):



Melyik $f\colon \{0,1\}^2 \to \{0,1\}$ függvényhez tartozó előjel-orákulum ez?

Ötlet: Használd fel, hogy H NOT $H = Z$, ami következik a 4.5. gyakorló feladat-ból.

Röviden összefoglalva, amit eddig elértünk: A bit-orákulumok segítségével megkérhetünk egy kvantumszámítógépet, hogy értékeljen ki egy $f\colon \{0,1\}^n \to \{0,1\}$ függvényt pontosan ugyanúgy, ahogy egy reverzibilisen működő hagyományos számítógépet kérnénk meg erre (hasonlítsd össze 5.4. és 5.5. egyenletek). Ez azért fontos, mert azt jelenti, hogy nem almát hasonlítunk körtéhez, amikor azt kérdezzük, hány kérdést kell feltennünk a bit-orákulumnak U_f, hogy megtudjunk valamit f-ről, szemben azzal, hányszor kellene kiértékelni f-et egy hagyományos számítógépen, hogy ugyanezt megtudjuk. És mivel épp most mutattuk meg, hogy az előjel-orákulum O_f mindig implementálható a bit-orákulum U_f segítségével, nem tesz különbséget, ha ehelyett az előjel-orákulumnak O_f teszünk fel kérdéseket.

5.2. Kvantumalgoritmusok

Ebben a részben több olyan kvantumalgoritmussal fogunk megismerkedni, amelyek egy számítási problémát sokkal gyorsabban tudnak megoldani, mint bármely klasszikus algoritmus. Az ilyen gyorsítások nagyon meglepőek, mert első ránézésre úgy tűnik, hogy a kvantummechanikának semmi köze nincs a számításokhoz. Ennek ellenére a kvantummechanikai jelenségek, mint például az interferencia, nagyon impresszív számítási gyorsulásokat tesznek lehetővé. Ahogy azt korábban már elmagyaráztuk, olyan számítási környezetben dolgozunk, ahol csak a kérdések számát számoljuk. Vagyis feltételezve, hogy képesek vagyunk kiértékelni egy f függvényt bármilyen bemenetre, hány bemeneten kell kiértékelnünk ahhoz, hogy meghatározzunk valamilyen tulajdonságot f-ről. Másképp fogalmazva, ha hozzáférünk egy orákulumhoz, amely ki tudja értékelni f-et, hányszor kell használnunk ezt az orákulumot ahhoz, hogy meghatározzunk valamilyen tulajdonságot f-ről.

5.2.1. Deutsch algoritmusa

Vasárnap este van. Alíz és Boti épp most fejezték be a kvantumszámítás óra házi feladatait, és épp egy 3D-s filmet készülnek megnézni. Amikor bekapcsolják a holografikus tévékészüléküket, felfedezik, hogy a filmvetítést elhalasztották a Nemzetközi Transzgalaktikus Állomásról érkező váratlan drámai hírek miatt. Szörnyű baleset történt, és egy két legénységi tagot, Hilát és Imant tartalmazó modul levált a fő anyahajóról. Az utolsó üzenet, amit a modulból kaptak, az volt, hogy Iman megsérült és erősen vérzik – sürgősen vérátömlesztésre van szüksége. A helyzetet bonyolítja, hogy Hila és Iman kissé sokkos állapotban vannak és elfelejtették a saját vércsoportjukat – csak arra emlékeznek, hogy mindkettőjüknek A vagy B vércsoportja van. A hírolvasó a nagyközönséghez fordul segítségért, hogy javasoljanak egy módszert, amivel Hila és Iman megállapíthatják, hogy ugyanaz-e a vércsoportjuk, mert ha ez lenne a helyzet, Hila át tudná adni a vérét Imannak, hogy megmentse az életét. Ez azért van, mert Hila és Iman moduljában lévő orvosi készlet tartalmaz egy limfo-transzkódert, amely képes bármelyik vércsoportot az ellenkezőjére átalakítani. Így még ha kiderülne is, hogy a vércsoportjaik nem egyeznek, Hila át tudja alakítani a sajátját az ellenkezőjére a limfo-transzkóder segítségével.

E hírek hallatán Alíz és Boti azonnal elvetik a filmezés tervét, és elkezdenek azon gondolkodni, mit lehetne tenni Hila és Iman megsegítésére. A hírműsor folytatódik néhány további információval. Szerencsére kiderül, hogy a balesetben érintett modul tartalmaz egy adatbázis chipet, amely tárolja Hila és Iman vércsoportját. Ezt modellezhetjük egy $f\colon \{0,1\} \to \{0,1\}$ függvénnyel, ahol

$$f(0) = \begin{cases} 0 & \text{ha Hila vércsoportja A,} \\ 1 & \text{ha Hila vércsoportja B.} \end{cases}$$

$$f(1) = \begin{cases} 0 & \text{ha Iman vércsoportja A,} \\ 1 & \text{ha Iman vércsoportja B.} \end{cases}$$

Amit meg kell határozni, az az, hogy $f(0) = f(1)$ vagy sem!

A megoldás kézenfekvőnek tűnik: Hilának és Imannak egyszerűen kétszer kell lekérdeznie az adatbázist, hogy kiolvassák a saját vércsoportjukat, $f(0)$-t és $f(1)$-et, majd összehasonlítani a két értéket, hogy lássák, ugyanazok-e. Sajnos a baleset részben tönkretette az adatbázis chip vezérlőlogikáját, és a hírolvasó jelenti, hogy az első lekérdezés után az adatbázis chip valószínűleg teljesen ki fog égni.

A két főszereplőnk zsákutcába került. Nyilvánvaló, hogy bármely klasszikus algoritmusnak kétszer kell kiértékelnie f-et ahhoz, hogy meghatározza, vajon $f(0) = f(1)$. Valóban, ha csak $f(0)$ értékét ismered, akkor az, hogy $f(0) = f(1)$, még mindig függ $f(1)$ értékétől, és hacsak nem számítod ki $f(1)$-et is, nem tudnád megmondani, hogy $f(0) = f(1)$. Hasonlóképpen, ha csak $f(1)$-et ismered, nem tudod összehasonlítani $f(0)$-val, hacsak nem tudod $f(0)$ értékét is. Bármilyen stratégiát használsz, mind $f(0)$-t, mind $f(1)$-et ismerned kell ahhoz, hogy meghatározd, vajon $f(0) = f(1)$. Tényleg nincs kiút ebből a helyzetből?

Néhány kézikönyv átlapozása után Boti felfedezi, hogy az adatbázis chipet *kvantum módba* lehet kapcsolni. Ha be van kapcsolva, az adatbázis chip már nem klasszikusan értékeli ki a függvényt, hanem ehelyett az O_f előjel-orákulumot valósítja meg. Lehetne ezt valahogy felhasználni a probléma megoldására? Alíz egy pillanatig gondolkodik rajta, és hirtelen rájön, hogy pontosan erre való Deutsch algoritmusa! Alíz és Boti gyorsan átnéznek néhány számítást, hogy megerősítsék, működik-e, és leülnek, hogy intergalaktikus e-mailt írjanak Hilának és Imannak utasításokkal arról, hogyan oldhatják meg a problémát. Az utasításaik a következők:

1. Készítsetek elő egy qubitet $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$ állapotban.
2. Használjátok az adatbázis chipet kvantum módban az O_f művelet alkalmazásához.
3. Alkalmazzátok a Hadamard-kaput H a kimeneti qubiten, és mérjétek meg.
4. Ha az eredmény 0, akkor Hilának és Imannak ugyanaz a vércsoportja, ellenkező esetben különböző vércsoportjaik vannak.

Vegyük észre, hogy ebben az eljárásban Hila és Iman csak *egyszer* kérdezik le az adatbázis chipet, hogy meghatározzák, ugyanaz-e a vércsoportjuk. Itt van az algoritmus implementációja QUIRKY-ben:

A kép azt mutatja, hogy az eredmény 1, tehát Hilának és Imannak különböző vércsoportjaik vannak.

De miért működik Deutsch algoritmusa? Elemezzük lépésről lépésre. Az első Hadamard-kapu létrehozza a $|+\rangle = H|0\rangle$ állapotot. Ezután alkalmazzuk az O_f előjel-orákulumot, ami a következő állapothoz vezet:

$$\begin{aligned} O_f|+\rangle &= \frac{1}{\sqrt{2}}O_f|0\rangle + \frac{1}{\sqrt{2}}O_f|1\rangle \\ &= \frac{1}{\sqrt{2}}(-1)^{f(0)}|0\rangle + \frac{1}{\sqrt{2}}(-1)^{f(1)}|1\rangle . \end{aligned}$$

A második Hadamard alkalmazása után a következő állapotot kapjuk:

$$\begin{aligned} HO_f|+\rangle &= \frac{1}{\sqrt{2}}(-1)^{f(0)}H|0\rangle + \frac{1}{\sqrt{2}}(-1)^{f(1)}H|1\rangle \\ &= \frac{1}{\sqrt{2}}(-1)^{f(0)}|+\rangle + \frac{1}{\sqrt{2}}(-1)^{f(1)}|-\rangle \\ &= \frac{1}{2}(-1)^{f(0)}(|0\rangle + |1\rangle) + \frac{1}{2}(-1)^{f(1)}(|0\rangle - |1\rangle) \\ &= \frac{(-1)^{f(0)} + (-1)^{f(1)}}{2}|0\rangle + \frac{(-1)^{f(0)} - (-1)^{f(1)}}{2}|1\rangle . \end{aligned} \tag{5.7}$$

Vegyük észre, hogy a két előjel $(-1)^{f(0)}$ és $(-1)^{f(1)}$ összeadódik az első amplitúdóban, de kivonódik a második amplitúdóban. $f(0)$ és $f(1)$ értékeitől függően minden amplitúdónál vagy konstruktív vagy destruktív interferenciát fogunk megfigyelni (lásd 2.6.1. alfejezet). Valójában azt, hogy melyik amplitúdó marad meg, csak az határozza meg, hogy $f(0)$ és $f(1)$ egyenlő-e vagy sem:

$$\begin{aligned} f(0) = f(1): &\qquad HO_f|+\rangle = \pm|0\rangle , \\ f(0) \neq f(1): &\qquad HO_f|+\rangle = \pm|1\rangle . \end{aligned} \tag{5.8}$$

Jó gyakorlat ezt expliciten ellenőrizni:

5.3. Gyakorló feladat: Deutsch algoritmusának ellenőrzése

Emlékezzünk vissza a 5.1. gyakorló feladat-ból, hogy négy $f : \{0,1\} \to \{0,1\}$ függvény létezik. Minden függvényre számítsd ki a $HO_f|+\rangle$ állapotot a 5.7. egyenlet felhasználásával.

Az 5.8. egyenlet mutatja, hogy a végső mérés akkor és csak akkor ad 0 eredményt, ha $f(0) = f(1)$. Tehát Deutsch algoritmusa helyesen határozza meg, hogy $f(0) = f(1)$. Fontos, hogy az $f : \{0,1\} \to \{0,1\}$ függvényt csak egyszer értékeli ki az előjel-orákulum használatával. Ezzel szemben fentebb megbeszéltük, hogy bármely klasszikus algoritmusnak szükségszerűen külön-külön kell kiértékelnie mind az $f(0)$, mind az $f(1)$ függvényértékeket.

Deutsch algoritmusának egy másik értelmezése, hogy kiszámítja az $f(0)$ és $f(1)$ két bit összegét modulo kettő. Ez azért van, mert $f(0) \oplus f(1) = 0$ akkor és csak akkor, ha $f(0) = f(1)$. Valóban, emlékezzünk vissza a 3.20. egyenlet-ből, hogy a modulo kettő összeadás a következőképpen működik:

x_1	x_2	$x_1 \oplus x_2$
0	0	0
0	1	1
1	0	1
1	1	0

(5.9)

Ezért is ismert a modulo kettő összeg a két bit *XOR*-jaként (a "kizáró VAGY" rövidítése), mivel akkor egy, ha pontosan az egyik bit egyes.

5.2.2. Hadamard-transzformáció és interferencia

Bár Deutsch algoritmusa nagyon meglepő, csak nagyon kicsi javulást ér el a legjobb klasszikus algoritmushoz képest, nevezetesen f 1 kiértékelését 2 helyett. Ez hasznos lehet, ha f kiértékelése nagyon hosszú időt vesz igénybe, mondjuk egy évet. De ha csak egy ezredmásodpercig tart, akkor a legtöbb ember nem bánná, ha két ezredmásodpercet kellene várnia a válaszra (és sokkal kevesebbet fizetne érte, mivel nem kellene kvantumszámítógépet használnia)! A kvantumszámítógépek hasznosságának demonstrálásához szeretnénk a számításokat több mint kétszeres tényezővel felgyorsítani.

Nincs remény nagyobb gyorsulás elérésére, ha csak egyetlen bit függvényeit nézzük, mivel ezek teljes mértékben meghatározhatók két kiértékeléssel: $f(0)$ és $f(1)$. Ezért általánosabban fogunk vizsgálni $f\colon \{0,1\}^n \to \{0,1\}$ függvényeket n bemeneti bittel, mivel sokkal több ilyen függvény létezik (valójában 2^{2^n} ilyen van). Ne feledjük, hogy célunk *nem* egy függvény kiértékelése (ezt valóban megtehetjük egyetlen lekérdezéssel az orákulumhoz), hanem inkább a függvény valamely érdekes tulajdonságának megismerése, amely a különböző bemeneteken felvett értékeire vonatkozik. Vannak-e olyan tulajdonságai az n-bites függvényeknek, amelyeket nagyon hatékonyan tanulhatunk meg okos kvantumalgoritmusok használatával?

Deutsch algoritmusának általánosításához figyeljük meg, hogy annak kulcsfontosságú összetevője volt egy Hadamard-kapu H bevezetése az előjel-orákulum előtt és után. Valami nagyon hasonlót tehetünk, ha n-bites függvényünk van. Ebben az esetben az O_f előjel-orákulum egy n qubites kvantumművelet, így egyszerűen alkalmazhatnánk Hadamard-kapukat minden qubiten az orákulum előtt és után. Azt a kvantumműveletet, amely párhuzamosan alkalmaz Hadamard-kapukat az n qubit mindegyikén, Hadamard-transzformációnak nevezzük. Emlékezzünk vissza a 3.2.3. alfejezet-ból, hogy ezt a következőképpen írhatjuk:

$$H \otimes \cdots \otimes H.$$

Nézzük meg először, mi történik, ha a Hadamard-transzformációt egy bázisállapotra alkalmazzuk. A $|0\ldots0\rangle$ esetében az eredmény egyszerűen az összes bázisállapot egyenletes szuperpozíciója:

$$\begin{aligned}(H \otimes \cdots \otimes H)\,|0\ldots0\rangle &= H\,|0\rangle \otimes \cdots \otimes H\,|0\rangle \\ &= \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) \otimes \cdots \otimes \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) \\ &= \frac{1}{\sqrt{2^n}}\left(|0\ldots00\rangle + |0\ldots01\rangle + \ldots + |1\ldots11\rangle\right).\end{aligned}$$

Ez az állapot 2^n bázisállapot szuperpozíciója. Van egy tömörebb módja ennek a felírásnak:

$$(H \otimes \cdots \otimes H)\,|0\ldots0\rangle = \frac{1}{\sqrt{2^n}} \sum_{y_1,\ldots,y_n \in \{0,1\}} |y_1,\ldots,y_n\rangle \tag{5.10}$$

A $\sum_{y_1,\ldots,y_n\in\{0,1\}}$ jelölés azt jelenti, hogy kiszámítjuk a $|y_1,\ldots,y_n\rangle$ összegét az $y_1,\ldots,y_n$ bitek minden lehetséges választására.

Általánosabban, egy tetszőleges bázisállapotra alkalmazva a kimenet mindig az összes 2^n bázisállapot szuperpozíciója, ahol minden amplitúdó egyenlő, kivéve az előjeleket. Annak pontos kiderítése, hogy mik ezek az előjelek, egy kicsit trükkös. Bemelegítésként próbáld ki először az $n = 1$ és $n = 2$ esetekre.

5.4. Gyakorló feladat: Két Hadamard

Emlékezzünk vissza a 2.20. egyenlet-ból, hogy $H\,|x_1\rangle = (|0\rangle + (-1)^{x_1}\,|1\rangle)/\sqrt{2}$, bármely $x_1 \in \{0,1\}$ esetén.

1. Írd fel a $H|x_1\rangle$ állapotot, tetszőleges $x_1 \in \{0,1\}$ esetén, a következő formában:

$$H|x_1\rangle = \frac{1}{\sqrt{2}} \sum_{y_1 \in \{0,1\}} (-1)^{\boxed{???}} |y_1\rangle ,$$

ahol $\boxed{???}$ valamilyen kifejezés, amely $x_1, y_1 \in \{0,1\}$-t tartalmazza. Határozd meg ezt a kifejezést.

2. huÍrd fel az $(H \otimes H)|x_1, x_2\rangle$ állapotot, tetszőleges $x_1, x_2 \in \{0,1\}$ esetén, a következő formában:

$$(H \otimes H)|x_1, x_2\rangle = \frac{1}{2} \sum_{y_1, y_2 \in \{0,1\}} (-1)^{\boxed{???}} |y_1, y_2\rangle ,$$

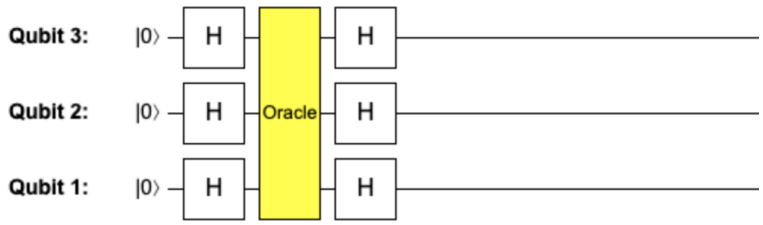
ahol $\boxed{???}$ valamilyen kifejezés, amely $x_1, x_2, y_1, y_2 \in \{0,1\}$-t tartalmazza.
Meg tudod határozni, mi ez a kifejezés?

Rájöttél a megoldásra? Nagyszerű, akkor tovább olvashatsz!

Általánosságban levezetheted a következő képletet, amely leírja az amplitúdók előjelmintázatát, amit akkor kapsz, amikor a Hadamard-transzformációt bármely n-qubites bázisállapotra alkalmazod: Bármely $x_1, \ldots, x_n \in \{0,1\}$ esetén,

$$(H \otimes \cdots \otimes H) \left|x_1, \ldots, x_n\right\rangle = \frac{1}{\sqrt{2^n}} \sum_{y_1,\ldots,y_n \in \{0,1\}} (-1)^{x_1 y_1 + \ldots + x_n y_n} \left|y_1, \ldots, y_n\right\rangle . \tag{5.11}$$

Most egyszerűen általánosíthatjuk Deutsch algoritmusát. Az n-qubites $|0 \ldots 0\rangle$ bázisállapotból kiindulva először alkalmazzuk a Hadamard-transzformációt, majd az O_f előjel-orákulumot a vizsgálni kívánt $f\colon \{0,1\}^n \to \{0,1\}$ függvényre, és végül egy újabb Hadamard-transzformációt. Például $n = 3$ esetén ez a következő QUIRKY áramkörnek felel meg:



Általános n esetén a végső n-qubites állapot matematikai képlete:

$$(H \otimes \cdots \otimes H) O_f (H \otimes \cdots \otimes H) \left|0 \ldots 0\right\rangle . \tag{5.12}$$

Le tudjuk írni ezt az állapotot explicitebben? Mint korábban, lépésről lépésre kiszámíthatjuk ezt. Először alkalmazzuk a Hadamard-transzformációt a csupa nullás bázisállapotra. A 5.10. egyenlet alapján megkapjuk az összes bázisállapot egyenletes szuperpozícióját:

$$(H \otimes \cdots \otimes H) \left|0 \ldots 0\right\rangle = \frac{1}{\sqrt{2^n}} \sum_{x_1,\ldots,x_n \in \{0,1\}} \left|x_1, \ldots, x_n\right\rangle .$$

Ezután alkalmazzuk az O_f előjel-orákulumot:

$$O_f (H \otimes \cdots \otimes H) \left|0 \ldots 0\right\rangle = \frac{1}{\sqrt{2^n}} \sum_{x_1,\ldots,x_n \in \{0,1\}} (-1)^{f(x_1,\ldots,x_n)} \left|x_1, \ldots, x_n\right\rangle .$$

Mit eredményez a végső Hadamard-transzformáció? A linearitás miatt alkalmazhatjuk a 5.11. egyenlet-t minden bázis vektorra, így a következő kifejezést kapjuk a 5.12. egyenlet állapotra:

$$\begin{aligned}
&(H \otimes \cdots \otimes H) O_f (H \otimes \cdots \otimes H) \left|0 \ldots 0\right\rangle \\
&= \frac{1}{\sqrt{2^n}} \sum_{x_1,\ldots,x_n \in \{0,1\}} (-1)^{f(x_1,\ldots,x_n)} \frac{1}{\sqrt{2^n}} \sum_{y_1,\ldots,y_n \in \{0,1\}} (-1)^{x_1 y_1 + \ldots + x_n y_n} \left|y_1, \ldots, y_n\right\rangle \\
&= \sum_{y_1,\ldots,y_n \in \{0,1\}} \frac{1}{2^n} \left(\sum_{x_1,\ldots,x_n \in \{0,1\}} (-1)^{x_1 y_1 + \ldots + x_n y_n} (-1)^{f(x_1,\ldots,x_n)} \right) \left|y_1, \ldots, y_n\right\rangle ,
\end{aligned} \tag{5.13}$$

ahol felcseréltük a két összeget az utolsó egyenlőség eléréséhez. $n = 1$ esetén ez pontosan ugyanaz, mint a 5.7. egyenlet. Általában azonban ez a kifejezés elég nehézkesnek és nehezen értelmezhetőnek tűnik – úgy tűnik, mintha az áramkör a bázisállapotok valamiféle szuperpozícióját számítaná ki, ahol az amplitúdó plusz és mínusz jelek valami furcsa összege!

Próbáljunk meg némi intuíciót szerezni a $|0 \ldots 0\rangle$ amplitúdójának vizsgálatával a 5.13. egyenlet-ban. Ennek a számnak a négyzete annak a valószínűsége, hogy amikor megmérjük az n

qubitet, minden eredmény nulla lesz. Mivel ez az amplitúdó megfelel az $y_1 = \ldots = y_n = 0$-nak, egyszerűen a következőképpen adható meg:

$$\frac{1}{2^n} \sum_{x_1,\ldots,x_n \in \{0,1\}} (-1)^{f(x_1,\ldots,x_n)}.$$

Mit jelent ez? Tegyük fel, hogy N_f olyan bemeneti bitsorozat van, amelyre f nullát ad eredményül, és $2^n - N_f$ olyan bitsorozat, amelyre f egyet ad eredményül. Ekkor,

$$\frac{1}{2^n} \sum_{x_1,\ldots,x_n \in \{0,1\}} (-1)^{f(x_1,\ldots,x_n)} = \frac{N_f - (2^n - N_f)}{2^n} = \frac{2N_f - 2^n}{2^n}.$$

Két érdekes szélsőséges eset van:[14]

- Ha f egy *konstans függvény*, akkor vagy $N_f = 0$ (a csupa-nulla függvényre) vagy $N_f = 2^n$ (a csupa-egy függvényre). Mindkét esetben az amplitúdó $\pm 2^n/2^n = \pm 1$. Mivel az összes amplitúdó négyzetének összege 1 kell, hogy legyen, arra következtetünk, hogy minden más amplitúdónak a 5.13. egyenlet-ban 0-nak kell lennie. Más szóval, amikor f konstans, a 5.13. egyenlet-ban lévő állapot egyszerűen $\pm |0 \ldots 0\rangle$. Ha megmérjük ennek az állapotnak mind az n qubitjét, akkor minden eredmény nulla lesz.

- Ha f egy *kiegyensúlyozott függvény*, ami azt jelenti, hogy ugyanannyi nulla van, mint egyes, akkor $N_f = 2^n/2$ és így az amplitúdó 0. Ez azt jelenti, hogy ha megmérjük a 5.13. egyenlet-ban lévő állapotot, akkor soha nem kaphatunk csupa nulla eredményt. Tehát egy kiegyensúlyozott függvény esetén a mérési eredmények közül legalább az egyik mindig egy lesz.

Figyeljük meg, hogyan felel meg ez a két eset nagyon különböző interferenciamintáknak (lásd 2.6.1. alfejezet). Az első esetben a $|0 \ldots 0\rangle$ amplitúdója ± 1-re erősödik egy erősen fókuszált *konstruktív* interferencia miatt, miközben az összes többi amplitúdó egyidejűleg eltűnik egy nagyon kiterjedt destruktív interferencia miatt. A második esetben a $|0 \ldots 0\rangle$ *destruktív* interferenciát tapasztal, ami miatt máshol nem nulla amplitúdók jelennek meg. A Hadamard-transzformáció szemlélteti az interferencia központi fontosságát a kvantumszámítástechnikában. A következő két szakaszban kulcsfontosságú módon fogjuk ezt felhasználni kvantumalgoritmusok tervezésére nagy számú qubit esetén.

5.2.3. Deutsch-Józsa algoritmus

Hasznosak-e a fenti megfigyelések valamire? Igen, azok! Tegyük fel, hogy $f\colon \{0,1\}^n \to \{0,1\}$ egy ismeretlen függvény, amely vagy konstans, vagy kiegyensúlyozott. Ekkor létezik egy egyszerű kvantumalgoritmus, amely meg tudja határozni, melyik eset áll fenn, mindössze f egyszeri kiértékelésével (azaz O_f egyszeri alkalmazásával).

Ezt az algoritmust Deutsch-Józsa algoritmusnak nevezik, és öt egyszerű lépésben működik:

1. Kezdjünk $|0 \ldots 0\rangle$ állapottal.
2. Alkalmazzuk a Hadamard-transzformációt $H \otimes \cdots \otimes H$.
3. Alkalmazzuk az f függvényhez tartozó O_f előjel-orákulumot.
4. Ismét alkalmazzuk a Hadamard-transzformációt $H \otimes \cdots \otimes H$.

[14] $n = 1$ esetén minden függvény vagy konstans, vagy kiegyensúlyozott. $n > 1$ esetén ez nem igaz (pl. az ÉS függvény sem nem konstans, sem nem kiegyensúlyozott).

5. Mérjük meg az összes qubitet. Ha minden eredmény nulla, az f függvénynek konstansnak kell lennie. Ellenkező esetben kiegyensúlyozottnak kell lennie.

Hogy lássuk, hogyan viszonyul ez a klasszikus algoritmusokhoz, vegyük észre, hogy bármely klasszikus algoritmusnak legalább $\frac{2^n}{2}+1$-szer kell kiértékelnie az f függvényt a legrosszabb esetben. Valóban, tegyük fel, hogy megismerjük a függvényt a bemeneti bitsorozatok felén (azaz $2^n/2$ bemeneten), és minden alkalommal ugyanazt a választ kapjuk. Ekkor még mindig nem következtethetünk arra, hogy a függvény konstans, mivel előfordulhat, hogy a függvény a maradék bemeneti bitsorozatok felén az ellenkező választ adja, miközben még mindig kiegyensúlyozott. Ez tehát a legrosszabb forgatókönyv. Ebben a forgatókönyvben elpazaroltunk $2^n/2$ kérdést, és még mindig nem tanultunk semmi hasznosat. Ahhoz, hogy biztosak legyünk abban, hogy f konstans vagy kiegyensúlyozott, még egy bemeneten ki kell értékelnünk. Összesen ez $\frac{2^n}{2}+1$ kiértékelést jelent f-ből, szemben a kvantum esetben történő 1 kiértékeléssel. Vegyük észre, hogy még mérsékelt értékek esetén is, mint például $n=100$, ez a különbség olyan drámai, hogy nem tudnád f-et ennyi alkalommal kiértékelni ésszerű időn belül (valójában addigra a Nap kimerítené üzemanyagát, és át kellene költöznöd egy másik naprendszerbe).

Összefoglalva: Ha $f\colon \{0,1\}^n \to \{0,1\}$ egy olyan függvény, amely vagy konstans, vagy kiegyensúlyozott, akkor a Deutsch-Józsa algoritmus f *mindössze egy kiértékelésével* meg tudja határozni, melyik eset áll fenn. Ez exponenciálisan jobb, mint bármely klasszikus algoritmus, amelynek a legrosszabb esetben $\frac{2^n}{2}+1$ bemeneten kell kiértékelnie a függvényt.

5.2. Házi feladat: Deutsch-Józsa futtatása

A sárga Oracle doboz a QUIRKY-ben egy olyan függvény előjel-orákulumát implementálja, amely vagy konstans, vagy kiegyensúlyozott. Implementáld a Deutsch-Józsa algoritmust QUIRKY-ben, és használd annak meghatározására, melyik eset áll fenn.

5.2.4. Bernstein-Vazirani algoritmus

Fentebb megbeszéltük, hogyan használhatjuk a Hadamard-transzformációt egy érdekes probléma megoldására. Adott egy $f\colon \{0,1\}^n \to \{0,1\}$ függvény és az ígéret, hogy f vagy konstans, vagy kiegyensúlyozott, a Deutsch-Jozsa algoritmus képes volt meghatározni, melyik eset áll fenn.

Ebben a részben egy másik érdekes problémát fogunk megvitatni, amelyet ugyanennek az eljárásnak egy enyhe változatával lehet megoldani. Mint korábban, most is egy ígérettel kezdünk az ismeretlen f függvényről, amellyel dolgozunk. Ezúttal ahelyett, hogy feltételeznénk, hogy konstans vagy kiegyensúlyozott, feltételezzük, hogy a következő speciális formájú:

$$f(x_1,\ldots,x_n) = x_1a_1 \oplus \ldots \oplus x_na_n, \tag{5.14}$$

ahol $a_1,\ldots,a_n \in \{0,1\}$ egy rögzített bitsorozat, amely meghatározza a függvényt.

Ha $n=1$, akkor csak két ilyen függvény létezik:

- $f(x_1)=0$, amely megfelel $a_1=0$-nak, és
- $f(x_1)=x_1$, amely megfelel $a_1=1$-nek.

Ha $n=2$, akkor már négy ilyen függvény van:

- $f(x_1,x_2)=0$, amely megfelel $[a_1,a_2]=[0,0]$-nak,
- $f(x_1,x_2)=x_2$, amely megfelel $[a_1,a_2]=[0,1]$-nek,
- $f(x_1,x_2)=x_1$, amely megfelel $[a_1,a_2]=[1,0]$-nak, és

- $f(x_1, x_2) = x_1 \oplus x_2$, amely megfelel $[a_1, a_2] = [1, 1]$-nek.

Vegyük észre, hogy mindegyik függvény az x_i változók valamely *részhalmazának* modulo kettő összegét számítja ki. Melyik részhalmaz? Amikor $a_i = 1$, a megfelelő x_i változó beletartozik a részhalmazba.

Általában 2^n választási lehetőség van az $a_1, \ldots, a_n$ bitsorozatra, és így 2^n függvény f létezik a speciális (5.14) formában. Valójában úgy gondolhatunk a 5.14. egyenlet-re, mint egy módszerre, amellyel *elrejtjük* az

$$[a_1, \ldots, a_n]$$

bitsorozatot az f függvényben. Hány kiértékelésre van szükségünk a függvényből, hogy felfedezzük? Mivel

$$\begin{aligned} a_1 &= f(1, 0, \ldots, 0, 0), \\ a_2 &= f(0, 1, \ldots, 0, 0), \\ &\vdots \\ a_n &= f(0, 0, \ldots, 0, 1), \end{aligned}$$

arra következtethetünk, hogy n kiértékelés az f függvényből biztosan elég. Bármely klasszikus algoritmus esetén ez optimális is: minden alkalommal, amikor kiértékeled a függvényt, csak egyetlen bitet ismersz meg. Mivel az ismeretlen $a_1, \ldots, a_n$ bitek teljesen tetszőlegesek és n darab van belőlük, legalább n-szer kell kiértékelned a függvényt, hogy mindet megismerd.

Most látni fogjuk, hogy egy kvantumalgoritmussal sokkal jobban teljesíthetünk. Kezdésként számítsuk ki, hogyan hat a 5.14. egyenlet-ben szereplő függvény előjel-orákuluma a bázisállapotokra:

$$\begin{aligned} O_f \left|x_1, \ldots, x_n\right\rangle &= (-1)^{x_1 a_1 \oplus \ldots \oplus x_n a_n} \left|x_1, \ldots, x_n\right\rangle \\ &= (-1)^{x_1 a_1 + \ldots + x_n a_n} \left|x_1, \ldots, x_n\right\rangle . \end{aligned} \tag{5.15}$$

A második lépésben felhasználtuk, hogy $(-1)^a$ csak a modulo kettőtől függ (azaz attól, hogy a páros vagy páratlan), így nem számít, hogy modulo 2 összeadást ('$\oplus$') vagy a szokásos összeadást használjuk. Hogyan lehet ezt az előjel-orákulumot implementálni? Valójában nem igazán érdekel minket, mivel algoritmusunk fekete dobozként fogja kezelni az orákulumot. De mivel jó gyakorlat, megpróbálhatod kitalálni a következő feladatban.

5.5. Gyakorló feladat: Az előjel-orákulum implementálása (opcionális)

Ebben a feladatban implementálni fogod a (5.14) formájú függvények előjel-orákulumát.

1. Amikor $n = 2$, négy ilyen függvény van, ahogy fentebb tárgyaltuk. Tudsz mindegyikhez találni egy QUIRKY áramkört, amely implementálja az előjel-orákulumot?
2. Magyarázd el szavakkal vagy képekkel, hogyan implementálhatod az előjel-orákulumot az általános esetben (azaz amikor $n \geq 1$ és az $a_1, \ldots, a_n \in \{0, 1\}$ bitek tetszőlegesek).

Most bemutatjuk a Bernstein-Vazirani algoritmust, amely felfedi a rejtett $[a_1, \ldots, a_n]$ bitsorozatot az f előjel-orákulumának egyetlen kiértékelésével:

1. Kezdjünk $|0 \ldots 0\rangle$ állapottal.
2. Alkalmazzuk a Hadamard-transzformációt $H \otimes \cdots \otimes H$.
3. Alkalmazzuk az f függvényhez tartozó O_f előjel-orákulumot.
4. Ismét alkalmazzuk a Hadamard-transzformációt $H \otimes \cdots \otimes H$.

5. Mérjük meg az összes qubitet. A mérési eredmény pontosan az $[a_1, \ldots, a_n]$ bitsorozat.

Az algoritmus azonos a 5.2.3. alfejezet-ben található Deutsch-Jozsa algoritmussal, kivéve az utolsó lépést, amely még egyszerűbb.

5.3. Házi feladat: Bernstein-Vazirani futtatása

A narancssárga Oracle doboz a QUIRKY-ben az (5.14) alakú függvény előjel-orákulumát implementálja $n = 4$ esetén. Implementáld a Bernstein-Vazirani algoritmust QUIRKY-ben, és használd a rejtett $[a_1, a_2, a_3, a_4]$ bitsorozat meghatározására.

Miért működik ez az algoritmus? Most rajtad a sor, hogy elvégezd az elemzést!

5.6. Gyakorló feladat: Bernstein-Vazirani ellenőrzése

Az $f(x_1, x_2) = x_2$ függvény megfelel az $[a_1, a_2] = [0, 1]$ bitsorozatnak, ahogy korábban láttuk.
Mutasd meg, hogy amikor a Bernstein-Vazirani algoritmust futtatod erre a függvényre, a mérési eredmény valóban mindig $[a_1, a_2] = [0, 1]$. Ne csak QUIRKY segítségével ellenőrizd ezt, hanem írd le magad az állapotot minden lépés után.

A következő házi feladatban elemezhetjük az általános esetet:

5.4. Házi feladat: Bernstein-Vazirani ellenőrzése (kihívást jelentő)

Mutasd meg, hogy amikor a Bernstein-Vazirani algoritmust futtatod egy (5.14) alakú függvényre, a mérési eredmény 100%-os valószínűséggel $[a_1, \ldots, a_n]$.

Ötlet: Mivel az algoritmus első négy lépése azonos a Deutsch-Józsa algoritmussal, a mérés előtti állapotot a 5.13. egyenlet adja meg.

5.3. Keresés Groverrel

Miután biztonságosan visszatértek a Földre, Hila és Iman jó barátok lettek Alízzal és Botival. Úgy döntenek, hogy együtt töltik a szilvesztert. Ez a nap egybeesik az éves kvantum lottó sorsolásával is! Tudják, hogy csak egyikük nyerheti meg a fődíjat, de vajon ki lesz az? Ha a négy főszereplőnket bitsorozatokkal címkézzük, mondjuk így:

	x_1	x_2
Alice	0	0
Bob	0	1
Hila	1	0
Iman	1	1

akkor a lottót modellezhetjük egy $f\colon \{0,1\}^2 \to \{0,1\}$ függvénnyel, amely 1-et ad eredményül a nyertesnek megfelelő bitsorozatra. Például, ha $f(1,0) = 1$, akkor Hila az idei lottó nyertese.

Hogyan tudják barátaink meghatározni a nyertest? Klasszikus algoritmussal akár háromszor is ki kell értékelniük a függvényt a nyertes meghatározásához. Valóban, tegyük fel, hogy megtudják, hogy $f(0,1) = 0$ és $f(1,0) = 0$ – még mindig nem tudják, hogy Alíz vagy Iman-e a nyertes! Ősi okokból, amelyeket már rég elfelejtettek, a lottó szabályai csak egyszer engedik meg a függvény kiértékelését. De természetesen a lottó örömmel alkalmazza az O_f előjel-orákulumot bármilyen kétqubites állapotra, amit főszereplőink szeretnének – elvégre ez egy kvantum lottó...

Alíz és barátai összegyűlnek és elkezdenek töprengeni. Egy idő után Boti türelmetlenné válik és javasolja: *„Kövessük egyszerűen a Deutsch-Józsa és Bernstein-Vazirani első néhány lépését – biztosan ugyanaz a trükk még egyszer működni fog..."* A többiek nem igazán tudnak jobb alternatívát, így nekiállnak és előkészítik a következő állapotot:

$$(H \otimes H)(|0\rangle \otimes |0\rangle) = |+\rangle \otimes |+\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle).$$

Ezután átadják az állapotot a kvantum lottónak, amely alkalmazza az O_f előjel-orákulumot és visszaadja az állapotot. Jelölje a és b azt a két bitet, amely a nyertest címkézi, azaz $f(a,b) = 1$ és az összes többi függvényérték nulla. Ekkor a lottó által visszaadott kétqubites állapot a következő:

$$\begin{aligned}
&\frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle - 2\,|a,b\rangle) \\
&= \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle) - |a,b\rangle \\
&= |+\rangle \otimes |+\rangle - |a\rangle \otimes |b\rangle\,,
\end{aligned}$$

ahol az első sorban a $-2\,|a,b\rangle$ az egyik pluszjelet mínuszjelre cseréli. A Hadamard-transzformáció alkalmazása után a következő állapothoz jutnak:

$$\begin{aligned}
&|0\rangle \otimes |0\rangle - H\,|a\rangle \otimes H\,|b\rangle \\
&= |00\rangle - \frac{1}{2}\Big(|0\rangle + (-1)^a\,|1\rangle\Big) \otimes \Big(|0\rangle + (-1)^b\,|1\rangle\Big) \\
&= -\frac{1}{2}\Big(-|00\rangle + (-1)^a\,|10\rangle + (-1)^b\,|01\rangle + (-1)^{a+b}\,|11\rangle\Big)
\end{aligned}$$

Most barátaink összezavarodnak és nem tudják biztosan, mit tegyenek. Hilának támad egy ötlete: *„Nem igazán tetszik nekem a mínuszjel a $|00\rangle$ előtt. Miért nem alkalmazunk egy olyan kvantum-műveletet, ami így néz ki?"*

$$\begin{aligned}
|00\rangle &\mapsto -|00\rangle \\
|01\rangle &\mapsto |01\rangle \\
|10\rangle &\mapsto |10\rangle \\
|11\rangle &\mapsto |11\rangle
\end{aligned} \tag{5.16}$$

Iman csatlakozik: *„Azt hiszem, tudom, hogyan lehet ezt felépíteni egy vezérelt Z művelettel és néhány NOT-tal..."* Pillanatok alatt barátaink a következő állapothoz jutnak:

$$-\frac{1}{2}\Big(|00\rangle + (-1)^a\,|10\rangle + (-1)^b\,|01\rangle + (-1)^{a+b}\,|11\rangle\Big)$$

Rövid idő múlva Alíz rájön: *„Ez a kétqubites állapot felírható tenzorszorzatként!"*

$$-\frac{1}{2}\Big(|0\rangle + (-1)^a\,|1\rangle\Big) \otimes \Big(|0\rangle + (-1)^b\,|1\rangle\Big) = -(H \otimes H)\,|a,b\rangle\,.$$

Boti fellelkesül: *„Aha! Csak egy újabb Hadamard-transzformációt kell alkalmaznunk és megmérni mindkét qubitet..."* Te is ki tudod találni, ki lesz az idei kvantum lottó nyertese?

5.5. Házi feladat: Kvantum lottó

1. Írd le a 5.16. egyenlet-ban szereplő kvantumműveletet egy vezérelt Z művelet és néhány NOT művelet segítségével.

   Ötlet: Emlékezz, hogy a $CZ_{1\to 2}$ vezérelt Z művelet a következőképpen hat a $|x,y\rangle$ bázisállapotokra: Ha $x = 0$, akkor nem csinál semmit. Ha $x = 1$, akkor Z-ként hat a második qubiten. A múlt héten tanultad meg, hogyan lehet ezt felépíteni QUIRKY-ben.

2. Építsd fel QUIRKY-ben azt a kvantumalgoritmus, amelyet Alíz, Boti, Hila és Iman kitalált, és határozd meg az idei kvantum Lottó nyertesét.

A kvantumalgoritmus, amelyet barátaink éppen felfedeztek, a Grover-algoritmus egy speciális esete. Grover algoritmusa a következő problémát oldja meg: Adott egy $f\colon \{0,1\}^n \to \{0,1\}$ függvény orákuluma, megtalálja azokat az $x_1,\ldots,x_n \in \{0,1\}$ értékeket, amelyekre $f(x_1,\ldots,x_n) = 1$. Ezt a problémát úgy is felfoghatjuk, mint egy lottónyertes megtalálását 2^n résztvevő közül, vagy kevésbé prózaian, egy olyan elem megtalálását egy strukturálatlan adatbázisban, amely megfelel valamilyen érdekes tulajdonságnak (strukturálatlan alatt azt értjük, hogy az adatbázis elemei nincsenek rendezve vagy hasonló módon). Ugyanazzal az érvvel, amit korábban említettünk, bármely klasszikus algoritmusnak a legrosszabb esetben 2^n bejegyzés közül az összes, kivéve egyet meg kell néznie, mielőtt befejezné (az átlagos esetben is még mindig a bejegyzések körülbelül felét kell megnézni). Ezzel szemben Grover algoritmusa csak $\sqrt{2^n}$-nel arányos számú alkalommal kell használnia az orákulumot, ami sokkal lassabban növekszik n-nel! Grover algoritmusa egy nagyon sokoldalú eszköz, amely négyzetgyökös gyorsulást ad sok számítási problémára.

5.3.1. Szögnagyítás

Az utolsó kvantumalgoritmus, amit megbeszélünk, egy nagyon fontos szubrutin, amelyet sok más kvantumalgoritmusban használnak (például ez áll Grover algoritmusának középpontjában olyan függvények esetén, amelyeknek több mint $n = 2$ bemeneti bitje van).

A probléma, amit ez a szubrutin megold, először nagyon furcsának tűnhet. Valójában ez egy tisztán kvantumprobléma, amelynek még értelmes klasszikus megfogalmazása sincs. Ennek ellenére természetesen jelenik meg különböző más algoritmusokban, ami nagyon hasznossá teszi a szubrutint. Ez kiemeli, mennyire különbözőek a kvantumalgoritmusok mögött álló ötletek, és hogy új gondolkodásmódokra van szükség új kvantumalgoritmusok feltalálásához!

Továbbá ez egy olyan probléma példája, ahol az orákulumot egynél többször kell konzultálni. Ez különbözik az összes korábbi kvantumalgoritmustól, amelyeket ebben a küldetésben vizsgáltunk, és amelyek csak egyszer használták az orákulumot.

5.6. Házi feladat: Mi a szög? (kihívást jelentő)

Alíz és Boti kétségbe vannak esve. "Nagyon sajnáljuk a kellemetlenséget," mondják neked. "Elkészítettük ezt a gyönyörű $V(\theta)$ tükrözést

$$\theta = +\frac{\pi}{4k} \quad \text{or} \quad \theta = -\frac{\pi}{4k},$$

szöggel, de egyszerűen nem emlékszünk, melyik volt – csak a $k \geq 1$ egész számra emlékszünk!"[a] Ami még tetézi a bajt, hogy a kapu önmegsemmisítő lesz, ha k-nál többször használják.

Tudsz nekik segíteni? A feladatod, ha elfogadod, hogy meghatározd, melyik a két szög közül a helyes, úgy, hogy a $V(\theta)$ kaput legfeljebb k-szor használod.

Ötlet: Két egymást követő tükrözés egy forgatást eredményez. Mit kapsz, ha kombinálod a NOT-ot és $V(\theta)$-t? (Nem kell a 4.6. gyakorló feladat-ben szereplő általános képlet ennek a kérdésnek a megválaszolásához.)

[a] Ha úgy tetszik, gondolhatsz erre a tükrözésre úgy, mint egy orákulumra, amely furcsa módon elrejti az egyik szöget.

5.4. A kvantumos utazásod

Ez az utolsó probléma zárja le *A Kvantum Küldetést*. Hú, már öt hét eltelt – ez a szakkör igazi utazás volt! Reméljük, hogy jól érezted magad az elmúlt hetekben, és sok érdekes matematikát tanultál.

Ha nem tudsz betelni a kvantumszámítástechnikával, ne ess kétségbe. Mostanra már tapasztalt kvantumbit-varázsló vagy, és jól felkészültél arra, hogy önállóan tanulmányozz egy haladóbb könyvet. Miért nem nézel utána, hogy a helyi könyvtáradban van-e példány Michael Nielsen és Isaac Chuang *„Quantum Computation and Quantum Information"* című könyvéből?

5.5. A gyakorló feladatok megoldásai

5.1. Gyakorló feladat megoldása

A másik három függvény az $f = \mathrm{NOT}$ és a két konstans függvény $f(0) = f(1) = 0$ és $f(0) = f(1) = 1$.

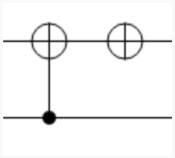
- A NOT függvényre:

$$U_{\mathrm{NOT}} |a, b\rangle = |a, b \oplus \mathrm{NOT}(a)\rangle = |a, b \oplus a \oplus 1\rangle = |a, \mathrm{NOT}(b \oplus a)\rangle ,$$

amit felismerhetünk, hogy egy vezérelt-NOT művelet és egy NOT művelet kompozíciójaként írható fel a második qubiten, azaz

$$U_{\mathrm{NOT}} = (I \otimes \mathrm{NOT})\, \mathrm{CNOT}_{1\to 2}.$$

QUIRKY-ban ez a következőképpen néz ki:



- A csupa nulla függvényre $f(0) = f(1) = 0$:

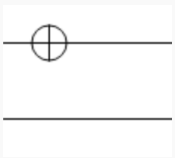
$$U_f |a, b\rangle = |a, b\rangle ,$$

tehát egyáltalán nem kell tennünk semmit:

- A csupa egyes függvényre $f(0) = f(1) = 1$:

$$U_f |a, b\rangle = |a, b \oplus 1\rangle = |a, \mathrm{NOT}(b)\rangle ,$$

tehát csak a második qubitet kell invertálnunk:



5.2. Gyakorló feladat megoldása

Négy függvény van: az „azonosság” függvény $f(x) = x$, a NOT függvény, a csupa nulla függvény és a csupa egyes függvény.

- Az azonosság függvényre $f(x) = x$, az 5.6. egyenlet így néz ki

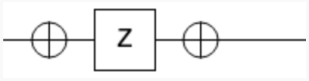
$$O_f |x\rangle = (-1)^x |x\rangle ,$$

tehát ez pontosan a Z kapu:

- A NOT függvényre $f(x) = \mathrm{NOT}(x)$, ezt szeretnénk:

$$O_f |x\rangle = (-1)^{\mathrm{NOT}(x)} |x\rangle = \mathrm{NOT}\, Z\, \mathrm{NOT} |x\rangle ,$$

ami a következő műveletsorozatnak felel meg:



- A csupa nulla függvényre $f(0) = f(1) = 0$:

$$O_f |x\rangle = |x\rangle ,$$

tehát egyáltalán nem kell tennünk semmit:

- A csupa egyes függvényre $f(0) = f(1) = 1$:

$$O_f |x\rangle = - |x\rangle ,$$

amit elérhetünk az első két orákulum egymás utáni alkalmazásával:

Valóban, az első orákulum minusz előjelet ad, ha $x = 1$, míg a második orákulum minusz előjelet ad, ha $x = 0$, tehát mindkét esetben minusz előjelet kapunk:

$$\mathrm{NOT}\, Z\, \mathrm{NOT}\, Z |0\rangle = \mathrm{NOT}\, Z\, \mathrm{NOT} |0\rangle = - |0\rangle ,$$
$$\mathrm{NOT}\, Z\, \mathrm{NOT}\, Z |1\rangle = \mathrm{NOT}\, Z\, \mathrm{NOT}(- |1\rangle) = -\mathrm{NOT}\, Z\, \mathrm{NOT} |1\rangle = - |1\rangle .$$

Az utolsó előtti lépésben a linearitást használtuk a minusz előjel előre hozásához.

5.3. Gyakorló feladat megoldása

Kiértékeljük az 5.7. egyenlet-t mind a négy függvényre.

- $f(x) = x$ esetén:

$$HO_f |+\rangle = \frac{1+(-1)}{2} |0\rangle + \frac{1-(-1)}{2} |1\rangle = |1\rangle .$$

- $f(x) = \mathrm{NOT}(x)$ esetén:

$$HO_f |+\rangle = \frac{-1+1}{2} |0\rangle + \frac{-1-1}{2} |1\rangle = -|1\rangle .$$

- A csupa nulla függvényre:

$$HO_f |+\rangle = \frac{1+1}{2} |0\rangle + \frac{1-1}{2} |1\rangle = |0\rangle .$$

- A csupa egyes függvényre:

$$HO_f |+\rangle = \frac{(-1)+(-1)}{2} |0\rangle + \frac{(-1)-(-1)}{2} |1\rangle = -|0\rangle .$$

5.4. Gyakorló feladat megoldása

1. Itt a $\boxed{???}$ helyén $x_1 y_1$ áll.
2. $n = 2$ esetén,

$$\begin{aligned}
(H \otimes H) |x_1, x_2\rangle &= (H |x_1\rangle) \otimes (H |x_2\rangle) \\
&= \left(\frac{1}{\sqrt{2}} \sum_{y_1 \in \{0,1\}} (-1)^{x_1 y_1} |y_1\rangle \right) \otimes \left(\frac{1}{\sqrt{2}} \sum_{y_2 \in \{0,1\}} (-1)^{x_2 y_2} |y_2\rangle \right) \\
&= \frac{1}{2} \sum_{y_1, y_2 \in \{0,1\}} (-1)^{x_1 y_1} (-1)^{x_2 y_2} |y_1\rangle \otimes |y_2\rangle \\
&= \frac{1}{2} \sum_{y_1, y_2 \in \{0,1\}} (-1)^{x_1 y_1 + x_2 y_2} |y_1, y_2\rangle ,
\end{aligned}$$

   tehát a $\boxed{???}$ helyén $x_1 y_1 + x_2 y_2$ áll.

5.5. Gyakorló feladat megoldása

1. Íme a négy függvény és azok fázis-orákulumai:

   - Ha $[a_1, a_2] = [0,0]$, $O_f |x_1, x_2\rangle = |x_1, x_2\rangle$, tehát a fázis-orákulum egyáltalán nem csinál semmit.
   - Ha $[a_1, a_2] = [0,1]$, $O_f |x_1, x_2\rangle = (-1)^{x_2} |x_1, x_2\rangle$, ami ugyanaz, mint $I \otimes Z$.
   - Ha $[a_1, a_2] = [1,0]$, $O_f |x_1, x_2\rangle = (-1)^{x_1} |x_1, x_2\rangle$, ami ugyanaz, mint $Z \otimes I$.
   - Ha $[a_1, a_2] = [1,1]$, $O_f |x_1, x_2\rangle = (-1)^{x_1+x_2} |x_1, x_2\rangle = (-1)^{x_1}(-1)^{x_2} |x_1, x_2\rangle$, ami ugyanaz, mint $Z \otimes Z$.

   Világos, hogy ez a négy művelet hogyan néz ki QUIRKY-ben.

2. Az általános minta most már egyértelmű. Egy tetszőleges, (5.14) alakú függvényre:

$$O_f |x_1, \ldots, x_n\rangle = (-1)^{x_1 a_1 + \ldots + x_n a_n} |x_1, \ldots, x_n\rangle = (Z^{a_1} \otimes \cdots \otimes Z^{a_n}) |x_1, \ldots, x_n\rangle ,$$

   ahol $Z^1 = Z$ és $Z^0 = I$. Más szóval, a j-edik qubiten akkor és csak akkor alkalmazunk Z kaput, ha $a_j = 1$.

5.6. Gyakorló feladat megoldása

Az 1. lépésben ezzel kezdünk:

$$|00\rangle$$

A 2. lépésben alkalmazzuk a $H \otimes H$-t és ezt kapjuk:

$$\begin{aligned} &\frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) \\ =&\frac{1}{2} (|00\rangle + |01\rangle + |10\rangle + |11\rangle) \end{aligned}$$

A 3. lépésben alkalmazzuk az $f(x_1, x_2) = x_2$ függvény O_f fázis-orákulumát:

$$\frac{1}{2} (|00\rangle - |01\rangle + |10\rangle - |11\rangle)$$

A 4. lépésben ismét alkalmazzuk a $H \otimes H$-t, aminek eredménye:

$$\begin{aligned} &\frac{1}{2}\Big((H \otimes H) |00\rangle - (H \otimes H) |01\rangle + (H \otimes H) |10\rangle - (H \otimes H) |11\rangle\Big) \\ =&\frac{1}{4}\Big((|00\rangle + |01\rangle + |10\rangle + |11\rangle) - (|00\rangle - |01\rangle + |10\rangle - |11\rangle) \\ &+ (|00\rangle + |01\rangle - |10\rangle - |11\rangle) - (|00\rangle - |01\rangle - |10\rangle + |11\rangle)\Big) \\ =&|01\rangle . \end{aligned}$$

Az 5. lépésben mindkét qubitet mérjük, és mindig a $[0,1]$ eredményt kapjuk.