

TDK-dolgozat

Czégel András

Szegedi Tudományegyetem
Természettudományi és Informatikai Kar
Számítógépes Optimalizálás Tanszék

Variációs kvantum optimalizáló algoritmusok

Szerző:

Czégel András

programtervező informatikus

MSc I. évf.

Témavezető:

Gazdag-Tóth Boglárka

Számítógépes Optimalizálás

Tanszék

Absztrakt

A kvantumszámítás napjaink egyik leggyorsabban feltörekvő területe. Bár a kvantumszámítógépekben lévő potenciált matematikailag már a 90-es évek közepén sikerült bizonyítani¹, illetve klasszikus számítógépekkel szembeni előnyét évekkel ezelőtt, 2019-ben demonstrálták², a valóságban még csak kis mértékben sikerült kihasználni az új számítási modellben rejlő lehetőségeket. A NISQ (zajos, kis méretű kvantum-) korszak számítógépeinek előnye leginkább a klasszikus, klasszikusan nehéznek számító kombinatorikus optimalizálási feladatok megoldásán mutatkozik meg. Így, habár intenzív kutatások irányulnak a hardverek fejlesztése, új algoritmusok keresése, fordítók és kontroll rendszerek fejlesztése felé is, a terület jelenlegi gyakorlati felhasználása az optimalizálási problémák köré épül.

Dolgozatom bemutatja az optimalizálásban jelenleg használatban lévő kvantum algoritmusokat elméleti és gyakorlati oldalról (VQE³, QAOA⁴), illetve bemutat egy újabb optimalizáló algoritmust, amely kombinálja a neurális hálók struktúrájában rejlő lehetőségeket a bemutatott kvantum algoritmusok működésével (L-VQE⁵). A dolgozat továbbá tartalmaz egy összehasonlítást az új algoritmus és a régiek között mind matematikai, logikai és experimentális módon is.

Tartalomjegyzék

1	Motiváció	1
2	Dolgozat felépítése, tudományos munka	4
3	Variációs kvantum áramkörök	6
3.1	Kapu alapú kvantumszámítás	6
3.2	Paraméteres kapuk, áramkörök	9
3.3	Hibrid optimalizáló algoritmusok felépítése	11
3.4	Hibrid optimalizálás a gyakorlatban	12
4	Variációs kvantum optimalizálás	16
4.1	Variációs kvantum sajátértékkeresés (VQE)	16
4.2	Quantum Annealing	18
4.3	Kvantum közelítő optimalizáló algoritmus (QAOA)	20
4.4	Rétegzett VQE algoritmus (L-VQE)	23
5	Saját eredmények	25
5.1	QAOA dominanciája az optimalizálásban	25
5.2	L-QAOA felépítése	27
5.3	L-QAOA egyértelmű paraméterezése reguláris gráfokon	29
6	Kitekintés	32

Motiváció

Kvantumszámítás

A kvantumszámítás, mint fogalom a 80-as évek elején jelent meg a köztudatban. Sokan Richard Feynman egy híres kijelentéséhez kötik, amelyben ezt mondta: *a természet nem klasszikus, és ha a természetet szeretnéd szimulálni, akkor jobban teszed, ha a kvantummechanikát használod*. Ezután elméletben meg lett mutatva, hogy lehetséges a kvantummechanika eszközeivel és tulajdonságaival Turing-teljes számítást megvalósítani^{6,7}. Utána több elméleti eredmény született a 90-es évek közepére, amelyek közt matematikailag bizonyítható, szubexponenciális előny is megjelent a klasszikus számítógépekhez képest, amely a perióduskeresés problémáját felhasználva, implementálva kvantumszámításra, a rejtett részcsoporthoz problémát gyorsítja¹.

Ezután több, főként kvadratikusan előnyt nyújtó algoritmus jelent meg, de a következő nagy lépés egy kvantumszámítógép megépítése volt, amely előrelendítette a területet. A 2010-es évek közepére elkezdődött a hardveres verseny is, illetve kontroll rendszerek is épültek, így felhőből, távolról is vezérelhetők ezek az eszközök.

Optimalizálási problémák

Kétféle nagyobb kategóriára bonthatók az optimalizálási problémák, amelyekkel érdemes kvantumszámítógépekkel foglalkozni. Az egyik a kvantumrendszerek struktúrájának optimalizálása. Ez szorosan kapcsolódik Feynman gondolatához, miszerint kvantum rendszert kvantum rendszerrel érdemes szimulálni. Bár a rendszerek nem szimulálva vannak, az optimalizálás bemenetét képző modell kvantum rendszert modellező kvantum rendszer, így klasszikusan exponenciálisan sok információt modellezve képes futni a kvantum optimalizáló algoritmus, jelentős előnyből indulva a klasszikus algoritmusokhoz képest. A másik csoport a kombinatorikus optimalizálási problémáké. Ezt a csoportot tisztán klasszikus problémák alkotják, olyan problémák, amelyek NPO-nehezkek, illetve a gyakorlatban vizsgált problémák eldöntési változata NP-teljes is. Az ilyen problémák egész értékű programként (ILP) vagy vegyes egész értékű programként (MILP) írhatók fel, ame-

lyek átírhatók kvadratikus feltétel nélküli optimalizálási problémára (QUBO), ezek pedig megoldhatók kvantumszámítógépekkel. Ezek a problémák, bár sokszor a gyakorlatban könnyebben megoldhatók, akár MILP solverrel, akár logikailag formalizálva SAT solverrel, általános esetben nehezek, így egy-egy kisebb méretű, illetve minden nagyobb méretű probléma klasszikusan nehéznek mondható.

Mindkét csoport az iparban, gyakorlatban sűrűn használt problémákat tartalmaz, amelyek egyes esetekben hatékonyabban oldhatók meg kvantumszámítógépekkel, vagy jobb közelítést adhatnak rájuk kvantumszámítógépek.

Kvantum optimalizálás

Kvantum optimalizálásra két egymástól teljesen eltérő elvű módszer jelent meg. Az első gyakorlatban használt módszer a quantum annealing, amelynek a fizikai megvalósítása az első kvantumszámítógépnek mondott eszköz volt, ám ez céleszköz, és általános számításokra nem lehetett használni. Az eszközt gyártó cég azóta is foglalkozik a területtel, és a kvantumszámításban egyelőre nagyon ritka módon profitot is termel. A quantum annealing fizikai módszer, Turing géppel nem modellezhető. Egy fizikai jelenség kihasználásáról és kontrollált felhasználásáról van szó.

A másik módszer a parametizált kvantum áramkörök optimalizálása klasszikus módszerekkel (lásd 4.1.1. ábra). Itt az optimalizáló lépéseket egy kvantum áramkör hajtja végre, amely paramétereket kap, majd az eredményt egy klasszikus számítógép értékeli ki, hibát számít, így optimalizálja a kvantum áramkör paramétereit. Ezen algoritmusok végrehajtására általános kvantumszámítógépek is képesek, és algoritmikusan jól analizálhatók, fejleszthetők.

Mindkét módszerben nagy a potenciál, az előbbi gyakorlati felhasználásokban bizonyítja is ezt, utóbbi pedig elméletben nagyon hatékony, és ha a hardver felfejlődik mellé, a gyakorlatban is bizonyítottan hatékony lesz.

Kvantumszámítás gyakorlati felhasználásai jelenleg

Bár maguk a kvantumszámítógépek még⁸ nem nagyok, pár algoritmus már praktikus szerepet is kaphatott. Ezek variációs optimalizáló algoritmusok, amelyek elméletben

nagyon hatékonyak, és gyakorlatban is használhatók már a zajos, kis méretű kvantum (NISQ) korszak eszközein.

A mai világban óriási az igény az optimalizálásra, akár pár ezrelék nyereség dönthet nagyobb vállalatok jövőjéről. Bár gyakorlatban pár probléma kivételével nincs megmutatva, hogy a kvantumszámítógépek előnyt jelentenének, amennyiben ez mégis kiderül, aki nem foglalkozik a területtel most, később óriási hátrányba kerülhet a versenyben.

Ugyanakkor már jelenleg is hatékonyak számító kvantumszámítást alkalmazó optimalizáló algoritmusok alkalmazva vannak a gyógyszeriparban, portfólió optimalizálásban, és már gyártósorok optimalizálását is szervezik kvantumszámítógépekre.

Dolgozat felépítése, tudományos munka

TDK dolgozat motivációja

A dolgozatom legfőbb motivációja a terület jelenlegi állásának szemléltetése, majd ezen keresztül a saját kutatómunkám ismertetése. Bemutatom a kvantum optimalizáló algoritmusok mögötti motivációt, az algoritmusok struktúrális felépítését, előnyeit és hátrányait, illetve a kvantumszámítás jelenlegi gyakorlati felhasználásának lehetőségeit. Emellett a dolgozat a saját kutatási eredményeimet is tartalmazza, amely egy új, hatékony - időközben sajnos mások által publikált - algoritmust mutat be, és annak elméleti és gyakorlati összehasonlítását a jelenleg szélesebb körben használt algoritmusokkal szemben.

Algoritmusok struktúrális felépítése

A kvantumszámítás előnyeit felhasználva közelítő, iteratív és hibrid klasszikus-kvantum optimalizáló algoritmusok születtek. Iteratívak, mert ugyanazt a szubrutint hajtják végre, paramétereket változtatva. Közelítők, mert egyik módszer sem oldja meg direkt módon a feladatot, pontos megoldást sem ad, viszont tetszőlegesen tudja közelíteni az optimumot hardveres hibahatáron kívül. Hibrid algoritmusok, mert 1-1 iteráció során az algoritmus klasszikus és kvantum számításokat is végez: egy klasszikus keret foglalja magába az optimalizálandó kvantumalgoritmust.

Dolgozatomban bemutatom a kapu alapú kvantumszámítást, és az optimalizálásra való felhasználási módját, illetve a kombinatorikus optimalizálási feladatok megoldását az algoritmusokkal, kihasználva a közös tulajdonságaikat.

Jelenleg alkalmazott algoritmusok

A leírt algoritmusok a kvantumszámítás jelenleg egyetlen iparban, gyakorlatban használt megoldásai, ezért fontos szerepük van az egész terület fejlődésében. Egy nagyobb algoritmus családot mutatok be, majd egy általánosabb, illetve egy speciálisabb algoritmust, amely bár a célja és logikai háttere alapján része az algoritmuscsaládnak, mégis teljesen

más elvek mentén hajtja végre feladatát.

Az algoritmusokat részletesen bemutatom, formálisan definiálom, majd példán keresztül szemléltetem NPO-teljes kombinatorikus optimalizálási problémákon.

A főbb tárgyalt módszerek a Quantum Annealing, amely a szimulált hűtés valódi kvantum megvalósításán alapszik, a variációs kvantum sajátértékkeresés (VQE), amely egészértékű programozási feladatok megoldására alkalmas egy klasszikusan hatékony visszavezetés után, illetve a kvantum közelítő optimalizáló algoritmus (QAOA), amely a kettő említett módszer egyfajta kombinációjából inspirálódott, matematikailag bizonyítottan hatékony közelítő algoritmus.

Saját eredmények

A dolgozat saját eredményeket is tartalmaz. A tavalyi év kezdetére felépítettem az L-VQE struktúráját. Ezt tavasz elejére egy csoport feltöltötte arXiv-ra⁵. Ezután azzal foglalkoztam, hogy belássam az L-VQE előnyét a QAOA optimalizálóval szemben, amely jelenleg a kombinatorikus optimalizálás területén a leggyakrabban használt kvantum algoritmus. Ezeket az eredményeket, és a munkámat mutatom be ebben a szekcióban.

Variációs kvantum áramkörök

3.1. Kapu alapú kvantumszámítás

A kvantumszámítás általános célokra használható, Turing-teljes számítási modell⁶. Így elsőre felmerülhet a kérdés, hogy mi más módon lehet még kvantumszámítást megvalósítani, ami a jelenlegi klasszikus számítási modellel konzisztens, mégsem kapu alapú?

Léteznek kifejezetten optimalizálásra kifejlesztett céleszközök, amelyek kvantumszámítást végeznek, és klasszikus optimalizálási feladatokat oldanak meg, ám ezek nem Turing-teljesek, nem használhatók általános számításra, sokan nem tekintik őket számítógépnek, ugyanakkor a feladat ellátására kifejlesztett céleszközként hatékony megoldást jelentenek egy-egy optimalizálási problémára⁹. Ezek kevésbé algoritmikusan definiált fizikai módszerek, ahol kvantummechanikai folyamatok leírása alkotja a formális definíciót és a rendszerek viselkedése rejti a számítási módot, amely a számítástudomány eszközeivel megfoghatatlan. Ugyanakkor a módszer felbontható, diszkretizálható és más módon matematikailag leírható, ezt pedig a QAOA algoritmus felhasználja.

A bemutatott algoritmusokat mind kapu alapú kvantumszámítás valósítja meg. Itt a fizikai kvantumbiteken végrehajtott operációk, mint kapusorozat alkotja a számítás módját. Fizikailag többféle módon megvalósítható általános célú, univerzális kvantumszámítógép. Az architektúrák, és azok kontroll rendszerei teljesen különbözők, ezért a dolgozat nem tér ki a fizikai megvalósításra, csak a számítás mögötti matematikai absztrakcióra.

A kapu alapú kvantumszámítás elemeit a qubitek, több qubites rendszerek, illetve a rájuk ható kapuk alkotják. A qubit egy kétdimenziós, normált komplex vektorként írható le. A fizikai megvalósítás hibái sem részei az algoritmusoknak, ezért a sűrűségmátrix-reprezentáció helyett, algoritmikus leírás szempontjából ekvivalens állapotvektor-reprezentációt használ a dolgozat.

A több qubites rendszert (kezdetben) az egyes qubitek tenzorszorzata reprezentálja, azaz egy n qubites kvantumrendszert egy 2^n dimenziós, normált komplex vektor ír le. Ez a vektor kezdetben felbontható az n qubit tenzorszorzatára. A számítás során kialakul olyan

állapot, amely nem írható fel tenzorszorzat alakban, ezek az összefonódott állapotok.

Tehát két tenzorszorzatként felírható qubit, legyenek

$$|a\rangle = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \quad \text{és} \quad |b\rangle = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

ahol

$$a_1, a_2, b_1, b_2 \in \mathbb{C}, \quad \sqrt{a_1^2 + a_2^2} = 1, \quad \sqrt{b_1^2 + b_2^2} = 1,$$

szuperpozíciójának általános jelölése

$$\begin{bmatrix} a_1 b_1 \\ a_1 b_2 \\ a_2 b_1 \\ a_2 b_2 \end{bmatrix}.$$

A kvantum algoritmus futtatása után egy értéket tudunk leolvasni a rendszerről, az eredményt amit kapni szeretnénk. Ez egy diszkrét valószínűségi eloszlásból való mintavételezésként is felfogható, mégpedig oly módon, hogy az állapotvektor elemei (az egyes bázisállapotok amplitúdói) az egyes események, az elemek négyzete pedig a hozzájuk tartozó valószínűség, tehát $P(M | a) = a_1^2$, ahol M a standard bázisban¹ való mérés operátora. Leolvasáshoz mindig ezt a fajta mérési módszert használok a dolgozatban, az algoritmusok mind ezt használják, tehát standard bázisban (bázisállapotok az n bites bitsorozatok) mérve, az állapotvektor projekcióját veszem a bázisállapotokra, így kapva a valószínűségeket.

A bázisvektorokra ható kapukból álló kvantumszámítás ilyen szintű absztrakciója elrejtje a fizikai jelentését a számításnak, ugyanakkor nagyban leegyszerűsíti a megértését, miközben konzisztens marad a megvalósítással. Amit állapotvektorral ábrázolunk, pontosabban egy diagonális sűrűségmátrix ($\text{diag}(|a\rangle) = \rho$), méréskor pedig az egyes bázisállapotokhoz tartozó mérési operátorok ($M_1, M_2, \dots, M_n$) Hermitikus, pozitív szemidefinit

¹számítási bázis, vagyis a bázisállapotok a $|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ és $|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

mátrixok. Ilyen mérés esetén a valószínűség, hogy a rendszert az i -edik mérési operátor állapotában mérjük, a sűrűségmátrix és a mérési operátor szorzatának nyoma, azaz $P(i) = \text{tr}(\rho M_i)$. A számítási bázis mérési operátorai ortogonális projektorok, amelyek összege az egységmátrix. Ez az ún. PVM (positive-valued measure). A standard bázis ortogonális projektorai az egyes bitsorozatokot ábrázoló állapotvektorok önmagukkal vett külső szorzatai, azaz $M_i = |i\rangle \langle i|$ minden $i \in \{0, 1\}^n = |i_1\rangle \otimes |i_2\rangle \otimes \dots \otimes |i_n\rangle$ bázisállapotra. Sokkal többféle mérési módszer és bázis lehetséges, ugyanakkor ehhez a dolgozathoz elegendő a standard bázis PVM mérése, a Born-szabály használata.

A qubitekre kvantum kapuk hathatnak, amelyeket matematikailag unitér mátrixokkal lehet reprezentálni. Egy braket jelölésben, állapotvektorral reprezentált $|\psi\rangle$ qubitre ható U kvantum kapu jelölése: $U|\psi\rangle$. Ha egy $U_1, U_2 \dots U_n$ kapusorozatot szeretnénk egymás után egy qubiten alkalmazni, akkor matematikailag - mivel a kapuk operátorai nem szükség szerűen kommutálnak - leírva a $U_n U_{n-1} \dots U_2 U_1 |\psi\rangle$ szorzat lesz az eredmény. Több qubites rendszerre ható, megfelelő méretű operátorokkal hasonló módon hathatunk.

A qubiten, mint két bázisállapotú egységén a számításnak egy X (NOT) kapu végrehajtása így néz ki:

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

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

A példa alapján általános qubitre is könnyen belátható, hogy az X kapu az amplitudók felcserélésével logikai negációt hajt végre a kétállapotú rendszeren.

A kvantumszámítás fontos tulajdonsága, hogy reverzibilis, azaz egy kapusorozatot megfordítva, és az operátoraikat invertálva (ami unitér operátorok esetén csak konjugált transzponálást, fizikailag pedig egy egyszerű transzformációt takar) minden egyes lépést "visszaszámíthatunk". Így egy nagyon fontos operátor, a CNOT kapu (kontrollált negálás), amely a XOR műveletet valósítja meg reverzibilis módon úgy, hogy az outputja két qubitből áll, és két input qubitje is van. Az operáció pedig: ha az első qubit állapota 1, akkor

negálja a másodikat.

3.2. Paraméteres kapuk, áramkörök

A qubitek folytonos rendszert alkotnak. Két bázisállapotuk lehet, de ezek között bármilyen értéket felvehetnek. Méréskor ugyan csak az egyik bázisállapotban tudjuk megfigyelni őket, de ezt az eredményt egy jól definiálható, a bázisállapotok amplitúdónégyzeiteiből álló valószínűségi eloszlásból kapjuk.

Így, változtatva, pontosítva az előző pont utolsó bekezdésén: *amennyire* az első qubit értéke 1, *annyira* negálja a másodikat. Ennek a segítségével elérhető például teljesen összefonódott állapot, ahol a (tegyük fel) két qubit állapotát nem tudjuk felírni az egyes qubitek állapotvektorának tenzorszorzataként. Ilyenkor a két qubit állapotára szokás azt is mondani, hogy teljesen korrelálnak, azaz az egyik állapotát megmérve pontosan megismerhetjük a másik állapotát, mérés nélkül is.

A qubitek folytonos értéket vehetnek fel, és a kapuk is követik ezt a mintát. A kapuk unitér mátrixok, azaz forgatást valósítanak meg egy qubites rendszer esetén a kétdimenziós komplex euklideszi térben. Ilyen mátrixból végtelen sok létezhet, és forgatás alapján parametizálhatók. Az egy qubites univerzális kvantum kapu a következő formában írható fel:

$$U(\theta, \phi, \lambda) = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) & -e^{i\lambda} \sin\left(\frac{\theta}{2}\right) \\ e^{i\phi} \sin\left(\frac{\theta}{2}\right) & e^{i(\phi+\lambda)} \cos\left(\frac{\theta}{2}\right) \end{bmatrix}$$

Végtelen sok kaput pontosan megvalósítani egy realiztikus architektúrán sem lehet, és definiálni lehet egy véges sok kapuból álló kapukészletet, amelyre minden univerzális (tetszőleges számú qubitre ható) kapu felbontható. Fizikailag megvalósítva ezt a kapukészletet bármilyen áramkört végre lehet hajtani.

A fenti példában egy qubiten több operáció lett végrehajtva, vagy több qubiten egy. Párhuzamosan több operációt több qubiten az operátorok tenzorszorzatának alkalmazásával lehet leírni, azaz a

$$I|\psi\rangle \otimes X|\phi\rangle = (I \otimes X)(|\psi\rangle \otimes |\phi\rangle)$$

egyenlőség azt írja le, hogy az I (identity, azonosság operátor, egységmátrix) kapuval hatunk az első qubitre, és az X operátorral a másodikra. Tehát a két qubitből álló rendszerre, amelyet egy négydimenziós állapotvektorral lehet reprezentálni, az $(I \otimes X)$ operátorral hatunk, amely mátrixa a négydimenziós mátrix

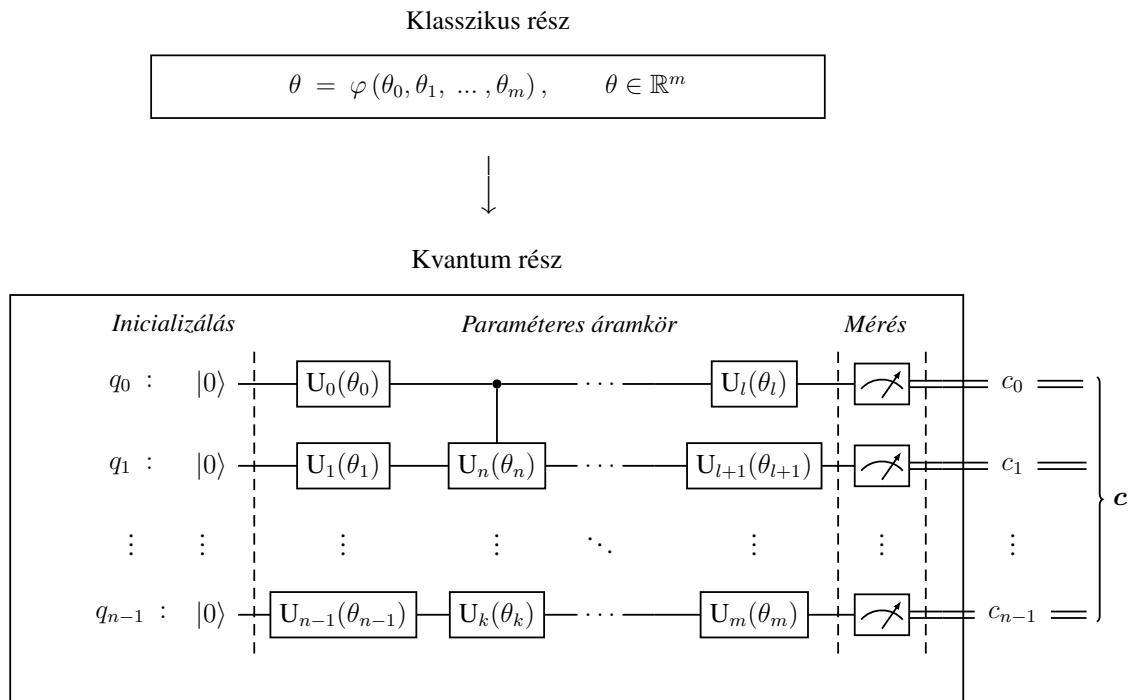
$$\begin{bmatrix} X & I \\ I & X \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Egy paraméteres kvantum algoritmus struktúrális felépítése, amelyben a paraméterek klasszikusan megadható valós számok, a következő módon épül fel:

1. Klasszikus paraméterek meghatározása (a kapuk definiálása).
2. Qubitek kezdőállapotának meghatározása (általában a $|0\rangle \otimes |0\rangle \otimes \dots \otimes |0\rangle = |00\dots 0\rangle$ állapot).
3. Kapuk előre definiált sorozatának alkalmazása a rendszeren.
4. Kapott szuperpozíció megmérése.

Tehát az áramkört előre definiálni kell, a kapuk egymásutánját leírni, mintha önálló függvények sorozatát definiálnánk, majd utána a paraméterekkel a viselkedésüket lehet meghatározni. Ebből kibontakozik az optimalizálási módszer, ugyanis a függvények paramétereinek optimalizálásáról van szó úgy, hogy a végső visszatérési érték az optimalizálandó célfüggvény értéke.

A következő ábrán egy ilyen áramkör felépítése látható. A klasszikus rész értéket ad a θ paramétervektor elemeinek, amelyek segítségével definiálható egy kvantum kapusorozat. Ezután inicializáljuk a qubiteket a kezdőállapotukba, majd végrehajtjuk rajtuk a definiált áramkört. Ezt követi a mérés, amely során a qubitek szuperpozíciójából klasszikus bitek sorozata lesz, eltárolva az eredményt (c vektor).



Ábra 3.2.1: Paraméteres kvantum áramkör felépítése

3.3. Hibrid optimalizáló algoritmusok felépítése

A kvantum áramkörökkel így definiálni lehet a probléma struktúráját, a paraméterekkel pedig előismeretek alapján optimalizálni lehet efelett a probléma felett. A hiba kiszámítását, kvantum áramkör paramétereinek változtatását, optimalizálását klasszikus optimalizáló algoritmus végzi. Ez az algoritmus általában gradiens alapú, és ezt a gradienst analitikusan ki kell tudnia számolni. Ilyen optimalizáló például az SPSA (Simultaneous Perturbation Stochastic Approximation) algoritmus, kifejezetten fizikai rendszerek feletti optimalizálásra.

A feladat struktúráját egy Hamilton operátor kódolja el. Az elkódolás módja változó lehet, ugyanakkor az elkódolás lényege, hogy az operátor legkisebb sajátértéke legyen az optimum, a hozzá tartozó sajátvektor pedig az optimum helye. A Hamilton operátor hermitikus és pozitív szemidefinit, ugyanakkor a mérete exponenciálisan nőhet a probléma méretével, amely klasszikusan nehezíti a probléma megoldását a továbbiakban.

Tehát a probléma minimális sajátérték-keresésre változik ebben az esetben egy exponenciálisan nagy hermitikus, pozitív szemidefinit operátoron. Ez felírható egy kvadratikus alakban, minimalizálási problémaként a következő módon:

$$\min(\langle H \rangle_\psi) \stackrel{\text{def.}}{=} \min(\langle \psi | H | \psi \rangle)$$

Ahol H a Hamilton operátor, ψ pedig a rendszer állapota, amely melletti várható értékét számítjuk az operátornak. Tehát keressük a rendszernek azt az állapotát, amely mellett minimális lesz a várható érték.

A rendszer állapotát pedig az áramkör határozza meg. Kiinduláskor a rendszer a $|00\dots 0\rangle$ állapotban van, ezután egy áramkör hat rá, amelynek nem változik a felépítése, csak a kapuinak a paraméterei. Ha az összes kapu paramétert egy θ vektor gyűjti össze, akkor a feladat az alábbi alakban írható fel:

$$\text{opt} = \min(\langle \psi(\theta) | H | \psi(\theta) \rangle)$$

Ahol $|\psi(\theta)\rangle$ az áramkör eredményeként előálló állapotvektor, azaz a klasszikus optimalizáló feladata megtalálni az optimális θ paramétereket, hogy $|00\dots 0\rangle \xrightarrow{U(\theta)} |\psi\rangle$. Ebből is látható, hogy θ az U áramkör kapuinak paramétereiből álló vektor, amely áramkörrel a $|00\dots 0\rangle$ állapotra hatva minimalizálja az algoritmus a Hamilton operátor kapott állapot melletti várható értékét.

3.4. Hibrid optimalizálás a gyakorlatban

A paraméteres kvantum áramkörökkel való optimalizálás globális, feltétel nélküli, direkt módszer, amely egy diszkrét halmaz felett vett célfüggvényértékek minimumát keresi.

A variációs kvantum optimalizálás standard feladata a QUBO (Quadratic Unconstrained Binary Optimization, azaz kvadratikus feltétel nélküli bináris optimalizálási) probléma megoldása.

$$x^* = \underset{x \in \{0,1\}^n}{\operatorname{argmin}} f(x)$$

ILP feladatról jól meghatározható és hatékonyan számítható konverzió létezik QUBO alakra. Vegyünk egy ILP feladatot az alábbi alakban:

$$\mathbf{x}^{(0)} = \underset{x}{\operatorname{argmin}} \left(\sum_i c_i x_i \right)$$

$$\sum_i A_{ji} x_i + b_j \leq 0, \quad \forall j$$

$$x_i \in \mathbb{Z} \geq 0 \quad \forall i$$

Az egyenlőtlenségek helyett mesterséges változók segítségével egyenletekre térünk át:

$$\mathbf{x}^{(0)} = \underset{x}{\operatorname{argmin}} \left(\sum_i c_i x_i \right)$$

$$\sum_i A_{ji} x_i + s_j + b_j = 0, \quad \forall j$$

$$x_i, s_j \in \mathbb{Z} \geq 0 \quad \forall i, j$$

Ezután tetszőleges pontossággal ábrázolhatjuk binárisan a változókat. A pontosság azért lényeges, mert racionális számokon is bizonyos pontossággal végre lehet hajtani ezt a transzformációt, legyen $\mathbf{z} = (\mathbf{x}, \mathbf{s})$ változó:

$$\mathbf{z}_i = \sum_r^{R_i-1} 2^r \psi_{i_r}$$

ahol R_i azt határozza meg, hogy az i -edik változó hány qubiten legyen elkódolva, és $\psi_{i_r} \in \{0, 1\}$. Ezenkívül legyen T_x és T_s ezt a binarizációt megvalósító transzformáció felső háromszögmátrixa. Az $\mathbf{x}$ változóvektorból és az $\mathbf{s}$ változóvektorból, ahol $x_i, s_i \in \mathbb{Z}$ megkonstruáljuk a változók bináris formáinak reprezentációit tároló Ψ vektort

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{s} \end{pmatrix} = \begin{pmatrix} T_x & 0 \\ 0 & T_s \end{pmatrix} \begin{pmatrix} \Psi_x \\ \Psi_s \end{pmatrix} = T\Psi$$

amelyet a (3.2) modellbe visszahelyettesítünk. Ezután a feltételeket büntető kifejezéseként a célfüggvény tagjaiként írjuk fel, tehát a cél, hogyha valamelyik feltétel nem teljesül, a (minimalizálandó) célfüggvény értéke jelentősen megnövekedjen.

$$f^2(\Psi) = \mathbf{c}^T T_x \Psi_x + p \|AT_x \Psi_x + T_s \Psi_s + \mathbf{b}\|^2 \quad (3.5)$$

ahol p egy a büntetés erősségét jelző konstans szorzó, amely kellően nagy, például $p \geq \mathbf{c}^T \mathbf{x}$, ekkor bármelyik feltétel megsértése esetén mindenképp legalább akkora lesz a célfüggvény értéke, mint a tér megsértett pontjának bármekkora környezetében található fizibilis pontok által elért érték.

Az alakot át lehet rendezni egy egyszerű kvadratikus alakra

$$f^2(\Psi) = \Psi^T \begin{pmatrix} Q_{xx} & Q_{xs} \\ Q_{sx} & Q_{ss} \end{pmatrix} \Psi + p \|\mathbf{b}\|^2 \quad (3.6)$$

ahol

$$\begin{aligned} Q_{xx} &= p \left(T_x^T \left(A^T A + \text{diag} (A^T \mathbf{b} + \mathbf{b}^T A) \right) T_x \right) + \text{diag} (\mathbf{c}) T_x, \\ Q_{xs} &= Q_{sx}^T = p \left(T_x^T A^T T_s \right), \\ Q_{ss} &= p \left(T_s^T T_s + \text{diag} (T_s^T \mathbf{b} + \mathbf{b}^T T_s) \right). \end{aligned} \quad (3.7)$$

A Q mátrix minden komponense hatékonyan számítható. A $p \|\mathbf{b}\|^2$ kifejezés csak egy konstans. Innen tehát elsőfokú tag nincs a feladatban, csak egy kvadratikus alak lett a minimalizálandó függvény, bináris változókkal. A hibrid algoritmusok a kvadratikus alakra találnak megoldást, de az optimum értéke ebből könnyen kiszámítható, hozzáadva a konstans tagot.

Az átalakítások továbbá ekvivalensek végig, így az eredeti ILP feladat megoldását is könnyedén vissza lehet számítani, a QUBO megtalált megoldásából.

A QUBO NP-nehéz probléma, az ILP visszavezethető rá büntetőfüggvény módszerrel. Ezáltal bármilyen LP, ILP probléma is felírható QUBO alakban. Bár klasszikus sejtés

szerint $P \neq NP$ még kvantumszámítógépeken sem, és így LP feladatok QUBO-ra való átírása nem hatékony megoldás, a gyakorlatban ennek ellenére is van felhasználása.

Nagy LP feladatok esetén, ahol a feladatból ritka, vagy lokális (kisebb mátrixok tensorszorzataként előállítható) mátrix írható fel a kvadratikus alakra, hatékony módszer lehet a paraméteres kvantum áramkörös megoldás. Egészértékű problémákra is lehet hatékony megoldó ebben az esetben egy ilyen algoritmus.

Ilyen problémák, például exponenciálisan növekvő méretű feladatok a kémiában jelennek meg, így a gyakorlatban is, már most is felhasználható a paraméteres kvantum algoritmuscsalád.

Variációs kvantum optimalizálás

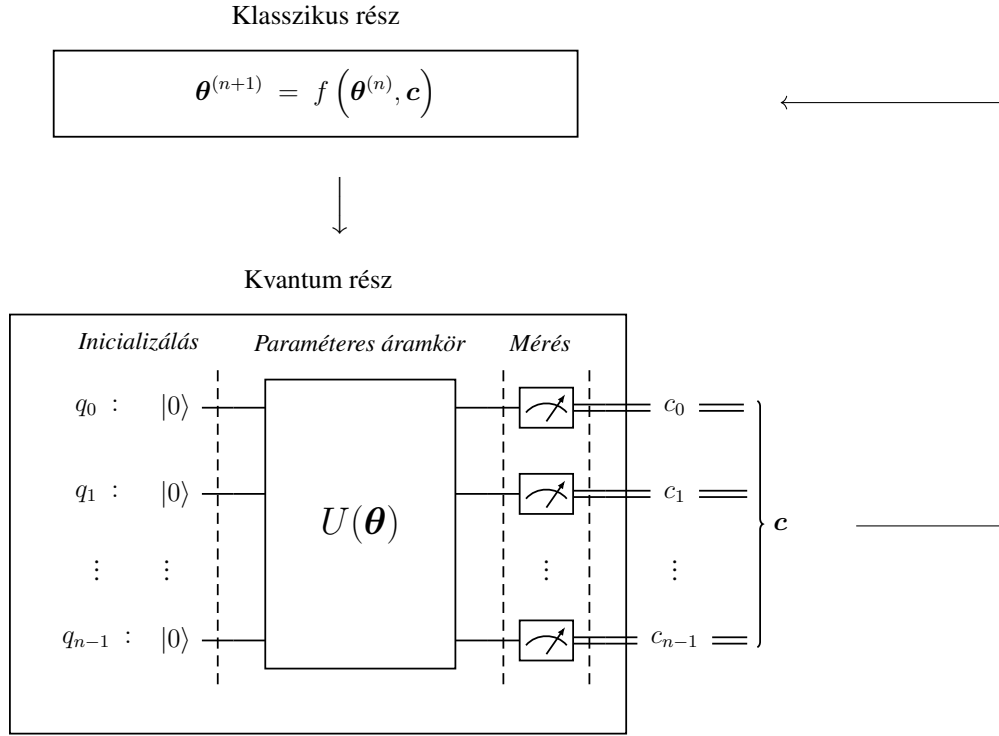
4.1. Variációs kvantum sajátértékkeresés (VQE)

Egy $\langle \psi | H | \psi \rangle$ alakot a H mátrix minimális sajátértékéhez tartozó $|\psi\rangle$ sajátvektora minimalizálja. Mivel a $|\psi\rangle$ kvantumállapot állapotvektora normalizált, ezért a minimális sajátértékhez tartozó sajátvektor egyértelműen meghatározható.¹ Paraméteres kvantum áramkörök klasszikus optimalizálóba ágyazásából előáll egy algoritmus ezen probléma megoldására, ez a variációs kvantum sajátértékkeresés (VQE)^{3,10}.

Tehát ha adott a H mátrix, akkor a minimalizálási probléma megfeleltethető egy legkisebb sajátérték keresési problémának. Ahhoz, hogy kvantum algoritmust használjunk a probléma megoldására, ennek a H mátrixnak teljesítenie kell pár tulajdonságot. A kvantum áramkörben ez a H mátrix a fizikai rendszer energiaoperátorának felel meg, hermitikus, pozitív szemidefinit (azaz $A \in \{\mathbb{C}^{n \times n} \mid v^\dagger A v \geq 0 \ \forall v \in \mathbb{C}^n, A_{ij} = A_{ji}^\dagger\}$). Az algoritmus variációs alapú, a kvantum része végzi a függvény kiértékelését, a klasszikus pedig a hiba alapján változtatja a kvantum áramkör paramétereit. Így a klasszikus algoritmustól függ, hogy milyen optimalizáló eljárást használunk, de többnyire klasszikus oldalon egy lokális módszer található. Ugyanakkor a kvantum algoritmus által kihasznált jelenségek, például a szuperpozíció, illetve a mérés, mint nemlineáris mintavételezés egy valószínűségi eloszlásból globálissá teheti a módszert.

A következő ábra a variációs kvantum sajátértékkereső algoritmus felépítését mutatja be. A klasszikus rész kiinduláskor kap egy $\theta^{(n)}$ paramétervektort, majd egy iteráció lépéseit ábrázolja: kvantum áramkör elkészítése a paraméterek alapján, kvantum áramkör inicializálása a $|0\rangle^{\otimes n}$ állapotba, az $U(\theta)$ áramkör végrehajtása, mérés. Mérés után pedig az eredményként kapott klasszikus bitsorozat, és az előző $\theta^{(n)}$ paraméterek alapján meghatározza az új $\theta^{(n+1)}$ paramétereket, ami alapján definiálja az új áramkört, és így tovább. Az f lehet a klasszikus optimalizáló a kvantumáramkör felett, az U pedig a kiértékelő kvantumáramkör.

¹Amennyiben H -nak egy minimális sajátértéke létezik. Ha több, akkor az *egyiket* elegendő megtalálni, amelyhez a sajátvektor szintén egyértelműen meghatározható.

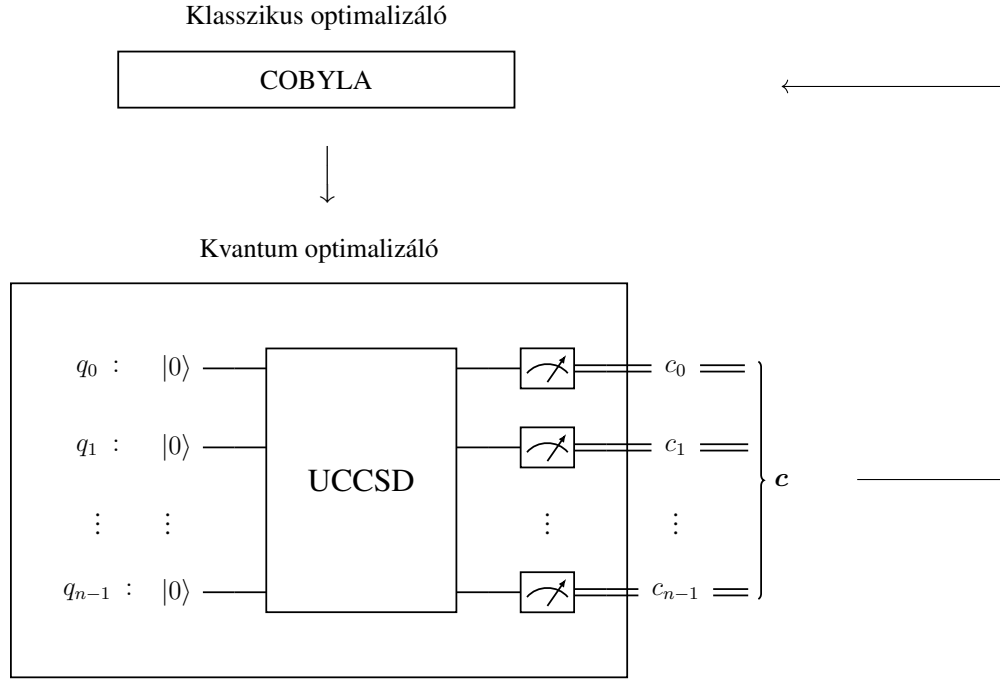


Ábra 4.1.1: A VQE felépítése

Az algoritmus változatait ketté lehet osztani a priori információk alapján. Kvantum algoritmusról van szó, természetes előnyt jelent, ha az optimalizálandó objektum rendelkezik jól meghatározott kvantummechanikai tulajdonságokkal, például ismerjük a rendszer részecskéinek számát és a rendszer struktúráját. Ekkor a probléma dimenziója a kvantumszámítógépen exponenciálisan csökkenthető a klasszikushoz képest. Ilyen problémákon a Hartree-Fock állapotból¹¹ indulva, amely a rendszer egyszerűsített modelljéből közelítő módon származik és klasszikusan könnyedén megtalálható, egy ilyen a priori ismeretet felhasználó parametizált áramkör, például a UCCSD nagyon hatékony. A UCCSD áramkör egy és kétrészecskés összefonódott klaszterekre bontással reprezentálja a rendszert, és ilyen struktúra felett kell már csak egy klasszikus optimalizálónak futnia. Ennek a módszernek hatékonyabb változata is létezik, a rendszerek fizikai struktúrájára optimalizálva.¹²

Ugyanakkor általános szempontok szerint ezt előfeldolgozó lépésként érdemes értelmezni, és bár ez azt jelenti, hogy ilyen problémákon klasszikus optimalizáló algoritmu-

sokkal szemben óriási előnye lehet ennek a hibrid algoritmusnak¹³, ez csak pár problémára igaz, és klasszikus optimalizálási feladatok nincsenek köztük. Emellett kvantum algoritmusok összehasonlítása esetén sem jelent ilyen előfeldolgozó lépés előnyt, ezért a továbbiakban ezt a lépést nem tekintem az algoritmus elemének. A fent tárgyalt algoritmus az egyes részeivel:



Ábra 4.1.2: Példa VQE algoritmusra

A második változat nem rendelkezik előzetes fizikai információval, kizárólag az energiaoperátort kapja meg. Tehát a probléma sokdimenziós komplex tér egy függvényének globális minimumának keresése. A módszer matematikai modelleken ugyanúgy QUBO-ra való transzformáció után használható.

4.2. Quantum Annealing

Az első megvalósított optimalizáló eljárások közé tartozik a Quantum Annealing^{14,15}. Az optimalizálás területén annealing-gel a szimulált hűtés¹⁶ (simulated annealing) eljárás esetében lehet találkozni. Ez egy metaheurisztika közelítő globális optimalizálásra, ahol a

direkt megoldó különböző kezdőpontokból indulva keresheti a megoldást, miközben az új pontok halmaza egyre szűkül egy pont környezete köré. A szűkülés nevezhető "hűtésnek", mert a megoldó egyre kisebbet "ugorhat" az előző kezdőponttól. Az ugrás korrelál egy T paraméterrel, ami a hőmérsékletet szimbolizálja, minél kisebb, annál kisebbet mozgathat az új pont az iterációk között. Amikor $T = 0$, akkor terminál az algoritmus, visszaadva a megtalált pontot, amely közelíti a globális optimumot. A paraméterek T , és ennek csökkenése, azaz egy T_n -t meghatározó "idő" függvény.

A kvantum eljárás a lehetséges megoldások egyenlő szuperpozíciójából indul. Ez lehetséges, ugyanis egyrészt kombinatorikus optimalizálási feladat esetén ez egy véges diszkrét halmaz, másrészt a feladat méretével logaritmikusan sok qubit elegendő a leírásához. A rendszer ezután időfejlődésnek indul, ahol a transzverzális tér erősségétől függően az egyes állapotok kvantum alagút effektussal alakulnak át más állapotokká. A transzverzális tér erőssége megfeleltethető a szimulált hűtés hőmérséklet paraméterének. Az időfejlődést megállítva, megmérve a rendszert az optimum helyét mérhetjük meg.^{17,18}

A módszer hatékonysága több mindentől függ. A transzverzális tér változásától függ, hogy időközben a rendszer marad-e az alapállapotában, illetve mennyire marad benne. Ehhez a feladatot el kell kódolnunk egy fizikai rendszerbe, erre az Ising modell megfelelő, mert viszonylag egyszerű rendszert készítünk vele, amely alkalmas a módszer megvalósítására. A modellt diszkrét változók alkotják, amelyek spinek mágneses dipólusmomentumjaiból állnak. Úgy érdemes elképzelni, mint egy kétdimenziós rácsot, ahol a rácspontok az egyes spinek, amelyek lehetnek fel vagy le állapotban, és a szomszédaikkal való interakcióval írják le a rendszer állapotát. Ha a feladat elkódolása az Ising modellbe helyes, akkor az időfejlődés utáni rendszer alapállapota kódolja el az optimumot. Ugyanakkor a transzverzális tér változásának gyorsításával egyre nagyobb eséllyel kaphatunk más állapotot méréskor.

Az módszer időigényét számítástudományi szempontból nem érdemes vizsgálni, nem műveletsorozatról, algoritmusról van szó, hanem egy fizikai kísérlet időtartamáról. A rendszer méretétől, az Ising modell által meghatározott Hamilton operátor tulajdonságaitól¹⁹ (pl. lokalitás), a transzverzális tér megválasztásától és változásától, illetve az időfejlődés idejétől függően lehet hibát számítani, ezek pedig a többi algoritmussal nem összehasonlíthatók.

Ennek a módszernek, ahol a rendszer végig az alapállapotában marad, adiabatikus kvantumszámítás a neve. Magát a módszert a DWave Systems alkalmazza a gyakorlatban. Az eszközeik tehát univerzális kvantumszámításra nem használható qubit alapú gépek, viszont jelenleg is valós méretű problémákon is alkalmazhatók, több ezer erősen összefüggő qubittel²⁰.

A Quantum Annealing adiabatikus kvantumszámításon alapszik, ami önmagában univerzális kvantumszámítás megvalósítására alkalmas, mert bármilyen állapotátterést meg lehet vele valósítani reverzibilis módon. A lényege, hogy egy H_1 Hamilton operátorral leírható rendszerből (nem energiamegtartó módon) egy H_2 operátorral leírható rendszert állítunk elő, úgy, hogy az energiaszinteket nem keresztezzük. Ezt úgy érdemes elképzelni, hogy megvalósítunk egy bijektív leképezést, ahol ha $(h_{11}, h_{12}, \dots, h_{1n})$ H_1 és $(h_{21}, h_{22}, \dots, h_{2n})$ H_2 rendezett spektruma, akkor a leképezés $h_{11} \mapsto h_{21}, h_{11} \mapsto h_{21}, \dots, h_{1n} \mapsto h_{2n}$ valósítja meg. Ehhez kell az *elég* lassú változás, ekkor ezt az adiabatikus tétel²¹ írja le és bizonyítja.

4.3. Kvantum közelítő optimalizáló algoritmus (QAOA)

A következő tárgyalt algoritmus, a QAOA is adiabatikus kvantumszámításon alapszik, kapu alapú, általános (kvantum)számítógépekkel végrehajtható algoritmus. Ehhez a kvantumrendszer időfejlődését kell kapukkal modellezni, vagyis a rendszer folytonos átalakulását a rendszerre ható diszkrét transzformációkat végrehajtó operátorok sorozatával. Ennek a módszernek a neve Trotterized Annealing, ezt valósítja meg a QAOA.

A felbontást Suzuki-Trotter formulák teszik lehetővé²². Tehát a cél egy $e^{-i\hat{H}t}$ alakú unitér mátrix felbontása, ahol i a komplex egység, t az idő, $\hat{H}$ pedig egy hermitikus energiaoperátor. Tehát a felbontást az első formula építi fel

$$e^{(B+H)} = \lim_{n \rightarrow \infty} \left(e^{\frac{B}{n}} + e^{\frac{H}{n}} \right)^n \quad (\text{S.T. 1})$$

ami átírható egy nagyságrendileg meghatározott hibataggal

$$e^{\delta(B+H)} = e^{\delta B} e^{\delta H} + \mathcal{O}(\delta^2) \quad (\text{S.T. 1})$$

Ezt a fent leírt unitér operátor $e^{-i\hat{H}t}$ esetén alkalmazva megkapjuk, hogy

$$\begin{aligned}
e^{-i\hat{H}t} &= e^{-i(B+H)t} \\
&\simeq (e^{-i(B+H)\delta})^{t/\delta} \\
&= (e^{-i\delta B} e^{-i\delta H})^p
\end{aligned} \tag{S.T. 2}$$

ahol $\hat{H} = B + H$ és $p = \frac{t}{\delta}$. Amikor (S.T. 1)-ben $\delta \rightarrow 0$, akkor a hibatag eltűnik. Azaz itt amikor $p \rightarrow \infty$, akkor a felbontás pontos.

Az algoritmus kapusorozatának definiálásához tehát szükség lesz a H és B Hamilton operátorokra. Ezt a (4.2)-ben definiált módon tesszük meg: H a problémát definiáló Ising Hamilton operátor lesz. B pedig a transzverzális teret leíró operátor. A változásukat δ írja le. A kifejezés tovább bontható:

$$\begin{aligned}
(e^{-i\delta B} e^{-i\delta H})^p &= (e^{-i\beta B} e^{-i\gamma H})^p \\
&= (e^{-i\beta_p B} e^{-i\gamma_p H} \dots e^{-i\beta_1 B} e^{-i\gamma_1 H})
\end{aligned}$$

ahol a δ (szög) szétbontható γ és β komponensekre. Majd ezek tovább bonthatók $\gamma_1 \dots \gamma_p$ és $\beta_1 \dots \beta_p$ szögekre. Az algoritmus itt is a $|+\rangle^{\otimes n} = \left(\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\right)^{\otimes n}$ állapotból, tehát n qubit egyenlő szuperpozíciójából indul. A H operátort például egy QUBO alak mátrixaként kaphatjuk meg, a B operátor pedig lehet² $B = \sum_{i=1}^n X_i$, ahol X_i az i -edik qubitre ható Pauli σ_x operátor. Az algoritmust definiáló kapusorozat tehát a következő:

$$|\psi_p(\gamma, \beta)\rangle = e^{-i\beta_p B} e^{-i\gamma_p H} \dots e^{-i\beta_1 B} e^{-i\gamma_1 H} |+\rangle^{\otimes n}$$

A paraméteres áramkör a globális optimalizálás szempontjából csak egy lokális optimalizálási lépés, amiből megkapott értéket szeretnénk minimalizálni vagy maximalizálni. Visszatérve a QUBO feladathoz, és a VQE definíciójához, azt kapjuk, hogy a problémánk (pl. maximalizálás esetén) tehát

$$\text{opt} = \max(\langle \psi_p(\gamma, \beta) | H | \psi_p(\gamma, \beta) \rangle),$$

ahol a paramétereink:

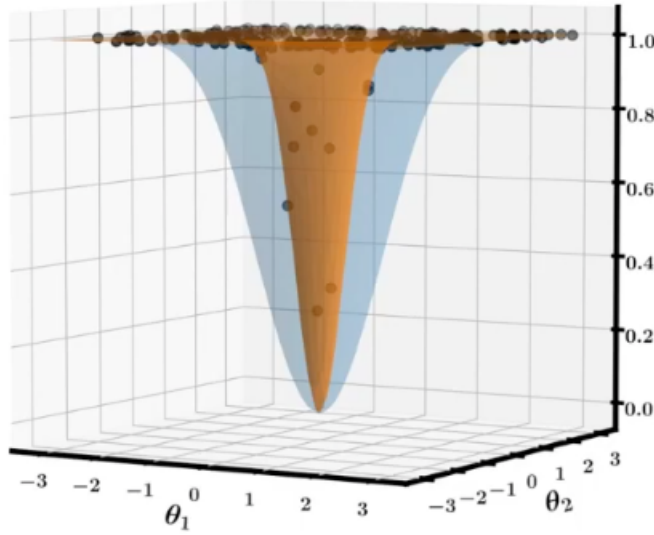
²mert $X|+\rangle = |+\rangle$

1. a p -érték, ami megmondja, hogy az annealing mennyi lépésre bontsa az algoritmus alapján az eredeti időfejlődést,
2. a γ paramétervektor, amely a problémát elkódoló Hamilton operátorra hat,
3. és a β paramétervektor, amely a transzverzális tér operátorára hat.

Minél nagyobb p értéke, úgy nő az algoritmus időigénye, illetve úgy csökken a számítási pontossága is, viszont csökken a képlethiba. Jelenleg közepesen kicsi, való életbeli kémiai problémákon 1-nél nagyobb nem lehet, kis problémákon 2-3 is, de ezek csak “proof of concept” működnek még. A szimulált hűtés példáján, ez olyan, mintha az első lépés után 0-ra állítanánk T -t.

A QAOA-nak akadt egy problémája, ami sok kételyt támasztott vele szemben: ez a barren-plató jelenség²³. Ennek a jelenségnek a megértéséhez a QAOA működését úgy érdemes elképzelni, például egy kétdimenziós függvényen, hogy egy teljesen sík konstansfüggvényből indul, majd a lokális optimumok mentén elkezd mélyülni (nem csak a függvényérték csökken a pontban, a függvény folytonos marad végig). Ezzel üregek keletkeznek. Méréskor az egyes üregek mélysége adja az adott lokális optimumban való mérés valószínűségét. A leírásból kiderült, hogy ∞ p esetén 1 eséllyel mérjük az optimumot. Ez úgy lehetséges geometriailag, hogy a többi pont értéke megegyezik, az optimumhoz tartozó üreg pedig végtelen mély és végtelen szűk. Ugyanakkor a mérés egy mintavételezés, mintha egy golyót ejtenénk a függvényre felülről. Így tehát az esélye, hogy - bár az amplitudó 1 - a rendszert az eredeti feladat optimumában mérjük, végtelenül kicsi. Kicsit realiztikusabban, egy N elemű halmaz feletti kombinatorikus optimalizálási problémánál ez az esély ugyanúgy $1/N$ -t közelíti, mint a kiindulásként vett konstans függvény esetén.

Az ábrán egy θ_1 és θ_2 által paraméterezett kétdimenziós függvényen látható a jelenség. A sötét pontok a mérések általi mintavételezések helyei. A kék rész mutatja az algoritmus egy korábbi állapotát, a narancssárga a mintavételezéskorit. Látható, hogy mintavételezéskor, akármilyen mély az optimum értéke, a helyes mintavételezés esélyét ez jóval csökkenti.



Ábra 4.3.1: Barren-plató vizualizációja Patrick Cole által a 2020 APS March Meetingen

4.4. Rétegzett VQE algoritmus (L-VQE)

A rétegzett VQE egy iteratív megközelítése a VQE-vel való optimalizálásnak⁵. Az ötlet mögötte a paraméterszám k iterációnkénti növelése, ezzel a kvantum áramkör hosszának növelése. Ez úgy tud működni, mint egy mélyneuronháló rétegei, a korábbi iterációkban megtalált paramétereket minimálisan, vagy egyáltalán nem változtatja a későbbi iteráció, inkább csak a saját, újonnan inicializált paramétereit, így egyre kifejezőbb, pontosabb optimalizálót kaphatunk, nagy komplexitásbeli romlás nélkül.

A klasszikus optimalizáló nem változik, a változtatás csak a kvantum algoritmust érinti. Klasszikusan csak a dimenziószám nő iteratíván, amit az optimalizáló adaptál. A kvantum áramkör pedig növeli a paraméterszámot:

$$|\psi_{n+1}(\theta_{1..n}, \theta_{n+1})\rangle \equiv U_{n+1}(\theta_{n+1}) U_{n..1}(\theta_{n..1}) |0\rangle^{\otimes m} \quad (\text{L-VQE})$$

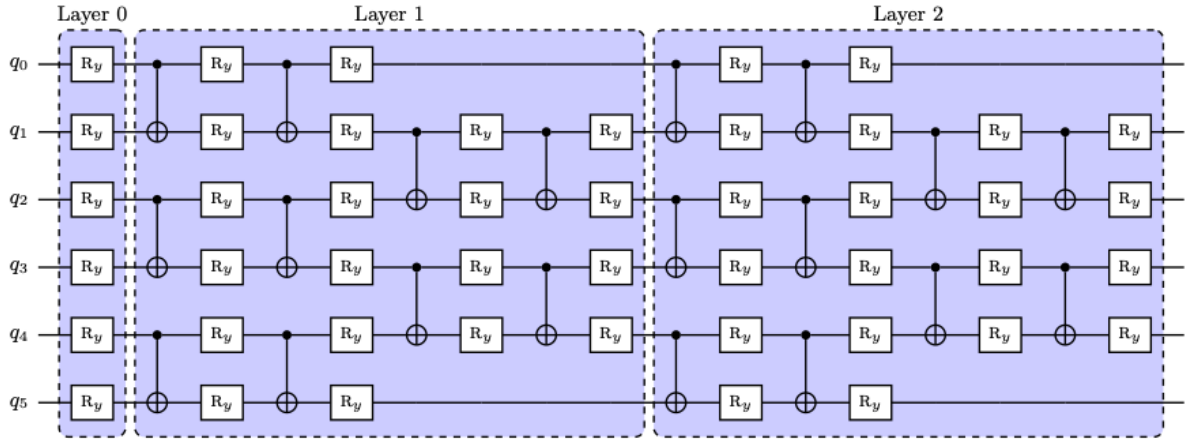
Minden iteráció elején kezdetben

$$U_{n+1}(\theta_{n+1}) = I,$$

így

$$U_{n+1}(\theta_{n+1}) U_{n..1}(\theta_{n..1}) = I U_{n..1}(\theta_{n..1}) = U_{n..1}(\theta_{n..1}),$$

tehát az új iteráció az előző iteráció megoldásából indul.



Ábra 4.4.1: L-VQE áramkörének felépítése⁵

Egy TwoLocal áramkör rétegzett felépítése $R_y(\theta)$ és CNOT kapukkal 6 qubiten a 4.4.1-es ábrának megfelelően alakul. A TwoLocal áramkör lokális, kétbites összefonódások segítségével terjeszt információt a rendszerben, az R_y kapuk pedig a számítási bázisbeli forgatási transzformációk, a θ paraméterük azt határozza meg, "mennyire" negálja a qubitet.

Az algoritmus lépései:

1. Paraméterek beállítása
2. Optimalizálás, amíg a kapott érték változása már elég kicsi
3. Új réteg hozzáadása, inicializálása
4. Ha nem értük el a meghatározott rétegszámot, ugrás a 2. lépésre, ha igen, jelenlegi állapottal visszatérés.

Ennek az algoritmusnak előnye, hogy sokkal kevésbé érzékeny a hardveres zajra, amely meghatározó problémaforrás a jelenlegi kvantumszámítógépek esetén, ezért pedig jelenleg jobb eredményeket is tud elérni egyes problémákon, mint a QAOA. A pontosabb számítás előnye, de ezenkívül kifejezőbb is, mint a sima VQE. A több paraméterrel egyre finomabbra hangolható az optimalizáló, anélkül, hogy az összes paraméterrel egyszerre kellene számítani.

Saját eredmények

5.1. QAOA dominanciája az optimalizálásban

Az általános kvantumszámításban leghatékonyabbnak tartott optimalizáló algoritmus a QAOA. Ennek az okai, hogy az eljárás, amit megvalósít matematikailag definiált, igazolható, pontos, és a hibája is jól definiált. Ráadásul egzakt globális optimalizáló, amely bizonyítottan megtalálja a globális optimumot NP-nehéz optimalizálási problémák esetén is.

A Quantum Annealerek már most is hatékonyak és igazolják a QAOA képességeit is, viszont a QAOA-hoz még nincs elégségesen fejlett hardver, ugyanis mind a rendelkezésre álló qubitek száma, mind a qubitek zajossága, mind a kvantum hibajavítás jelenlegi állása kevés ahhoz, hogy a kvantumszámítógépek előnyét realizálják valós méretű optimalizálási feladatokon. Egy nem túl régi tanulmány szerint 400 logikai qubit lenne nagyjából szükséges ahhoz, hogy hatékonyak mondható legyen az algoritmus²⁴. Jelenleg pár száz qubit felé tart az ipar, a hibajavításhoz viszont pontosabb hardver és nagyon nagy (több-százás) mennyiségű qubitre van szükség - már egy hibajavított, logikai qubithez is.

Az algoritmus számára elméleti szempontból ugyanakkor ezek a problémák nem fontosak, és ha egyenlő háttérrel, hardvert és tetszőleges számú (probléma méretében polinomiálisan sok) qubitet veszünk figyelembe, akkor már összehasonlítható egyéb kvantum optimalizáló eljárásokkal, amikkel szemben preferálják a QAOA-t.

A QAOA előnye, hogy teljesen fekete dobozként kezeli a függvényt és az adat struktúráját is, semmilyen előzetes tudást nem használ fel róla, ezért például a klasszikus optimalizálási feladatok átírt alakján kvantum algoritmusok közt jól teljesít. Ugyanakkor ha az adat és az optimalizálandó függvény struktúrája ismert, mint például a (4.1) alfejezetben vázolt esetben¹², akkor már egy általánosabb, másmilyen alakú (k-UpUCCSD áramkör) VQE is jobban teljesít.

Az én munkám itt azzal foglalkozott, hogy egy rétegezett VQE algoritmus a priori tudás nélkül, klasszikus (tehát kvantumszámítás szempontjából általános) problémákon teljesíthet-e jobban, mint a jelenleg használható, 1-2 p -értékű QAOA algoritmus. Ennek

a jelentősége az, hogy a QAOA egy speciális VQE, és abban az esetben, ha a rétegezett VQE az adiabatikus áramkör használata nélkül magában jobban teljesít, az indokolatlanná teszi a jelenleg legnépszerűbb optimalizáló algoritmus használatát a most használatban lévő architektúrákon. Mire az elméleti eredményt igazolhattam volna ezt a Layer-VQE-t bemutató cikk⁵ megmutatta, amely tavaly tavasszal került fel az arxiv-ra. A cikkből szeretnék ennek kapcsán kiemelni részleteket.

Először is, a klasszikus értelemben vett időkomplexitás, azaz az operációk száma a qubitek függvényében. Ehhez szükséges kiemelni, hogy a kvantum algoritmusok használatakor az egybites kapuk hibája és végrehajtási ideje nagyságrendekkel kisebb, mint a kétbiteseké, ezért szokás a két (vagy több) qubitre ható kapuk számát venni a kvantum algoritmus komplexitásának. A következő táblázat mutatja, hogy a QAOA kapukomplexitása a p -érték növelésével négyzetesen nő, míg az L-VQE algoritmusé csak lineárisan a rétegek számában (l). Az L-VQE teljes összefonódást generáló TwoLocal áramkörrel lett definiálva, ez egyszerű, de kifejező áramkör, ami tovább optimalizálható előzetes ismeretek és egyes felesleges összefonódást generáló kapuk eldobásával.

	QAOA (p)	L-VQE (l)
Egybites kapuk	$\left(\frac{3}{2}n^2 + \frac{n}{2}\right)p$	$(8n - 4)l + 2n$
Kétbites kapuk	$(5n^2 - 5)p$	$(4n - 2)l$

Táblázat 5.1.1: QAOA és L-VQE kapuinak száma a paraméterfelbontásuk függvényében.

Ezenkívül az optimalizáló algoritmusok pontossága is fontos, variációs kvantum optimalizálókhoz szokás a MAXCUT problémát felhasználni, mert könnyű QUBO alakra hozni, és kis méretű, különböző felépítésű gráfokon jól mérhetők rajta a módszerek kevés qubites eszközökön is. Ezt az összehasonlítást az idézett cikk 3-as ábrája mutatja. Az ábra alapján a QAOA kicsi p -érték esetén kis problémákon is gyengébb pontosságot ér el, mint az L-VQE.

Másik oldalról szeretném megemlíteni a dolgozatban, hogy a MAXCUT problémára létezik 0.878-approximáló klasszikus algoritmus²⁵, amit a QAOA 3-reguláris gráfokon sem tud kicsi p -érték mellett megközelíteni és amit a QAOA-t bemutató eredeti cikk⁴

is belát, így tehát a jelenlegi összehasonlítás még valós életbeli problémákhoz nagyon kis méretű problémákat tartalmaz, és csak a kvantum optimalizálók közti különbségek bemutatására szolgál.

5.2. L-QAOA felépítése

A következő algoritmust építettem fel a tavalyi év első felében, azaz egy rétegezett VQE-t, QAOA áramkörrel, a QAOA logikája mentén, iteratív közelítéssel. A szekcióban található eredmények mind saját kutatásból származó levezetések és adatok, amelyeket a L-VQE⁵ és az L-QAOA²⁶ cikk igazolnak. Ezeket az eredményeim elérése után jelentettek meg, ugyanakkor a cikkeket ezután már nem írtam meg, azonos konklúzióval.

A rétegezett felépítés ötletét tehát nem általános VQE esetén folytattam, hanem a QAOA algoritmus speciális felépítésén. Ez pedig a p -érték iteratív növelését jelentette. Az első ötletem ugyanaz volt, mint ahogy az L-VQE definiálva lett, azaz az új γ_{n+1} és β_{n+1} 0 értéket kap, majd ebből indul a következő iteráció, vagyis

$$\begin{aligned} \left| \psi_{p+1}(\gamma^{(p+1)}, \beta^{(p+1)}) \right\rangle &= e^{-i\beta_{p+1}B} e^{-i\gamma_{p+1}H} e^{-i\beta_p B} e^{-i\gamma_p H} \dots e^{-i\beta_1 B} e^{-i\gamma_1 H} |+\rangle^{\otimes n} \\ &= e^{-i\beta_{p+1}B} e^{-i\gamma_{p+1}H} e^{-i\beta^{(p)} B} e^{-i\gamma^{(p)} H} |+\rangle^{\otimes n} \\ &= e^{-i\beta_{p+1}B} e^{-i\gamma_{p+1}H} \left| \psi_p(\gamma^{(p)}, \beta^{(p)}) \right\rangle, \end{aligned}$$

amely $\gamma_{p+1} = \beta_{p+1} = 0$ esetén

$$\begin{aligned} &= \mathbf{II} \left| \psi_p(\gamma^{(p)}, \beta^{(p)}) \right\rangle \\ &= \left| \psi_p(\gamma^{(p)}, \beta^{(p)}) \right\rangle, \end{aligned}$$

tehát az előző iteráció eredménye.

Az általános, L-VQE ötlete alapján ennek hatékonyabbnak kellene lennie. Ugyanakkor ez a megoldás fizikailag azt jelenti, hogy az időfejlődést szimuláló operátorok transzformációinak szögei (γ és β) megegyeznek az előző iteráció szögeivel, és az új iteráció ezen korrigálhat.

Rövid geometriai magyarázat: ezt érdemes úgy elképzelni, hogy egy konvex sokszög egyik csúcspontjába szeretnénk eljutni a középpontból p szakasz felhasználásával. Az

optimum távolsága d adott, a szakasz d/p hosszú. A szakaszok $p_1, p_2, \dots, p_n$ egymás végpontjában kezdődnek, egy utat alkotnak. Ezeknek a szakaszoknak a szögeit határozza meg az algoritmus. A p növelésekor a szakaszok rövidebbek lesznek, és egy új szakaszt adhatunk hozzá, de ez belátható, hogy újabb, immár $d/(p+1)$ hosszú szakaszként túl sok korrekcióra nem képes. Az analógia nem pontos, a szakaszok hosszát nehezebb leírni, de jó intuíciót ad a problémáról.

Emiatt feltételezhető hogy a kiindulási ponthoz legközelebbi optimum értékénél sokkal jobbat nem tud megtalálni az algoritmus ilyen kezdőértékekkel. Ezen felül MAXCUT problémán, reguláris gráfokon lett tesztelve, ezekre pedig előre meghatározott paraméterekből indulva sokkal jobb közelítést kaphatunk, tehát a szögek a gráf struktúrájától sokkal inkább függenek mint a súlyoktól. Ezen a 0 kezdőértékkel hozzáadott új szög sem segít. Ezt az eredményt júliusban töltötték fel arxiv-ra, novemberben jelent meg, a sejtésnek *Fixed angle conjecture*²⁷ lett a neve.

A következő két táblázat egy 11 csúcsú irányított gráfon futtatott L-QAOA eredménye ilyen módszerrel, illetve összehasonlításképp a standard QAOA utána. A klasszikus, külső optimalizáló mindkét esetben a COBYLA volt, előre nem meghatározott iterációs számmal, addig futtatott, amíg a utolsó két eredmény távolsága nagyobb mint ε .

L (p -érték)	Kiértékelések száma	Energia	γ_1	γ_2	β_1	β_2
1	50	-2.6	0.02	-	1.047	-
1	200	-5.37	0.036	-	1.188	-
1	350	-7.53	0.065	-	1.331	-
2	50	-8.25	0.048	-0.167	1.347	0.066
2	200	-8.45	0.049	-0.205	1.374	0.089
2	350	-8.4691	0.053	-0.222	1.384	0.087
Célfüggvény értéke: 402.0						

Táblázat 5.2.1: L-QAOA futási adatai $p = 2$ -ig

A táblázaton látható, hogy nagyjából 350-350 kiértékelés után kezdett erősen konvergálni a COBYLA célfüggvényértéke, amit az energia értéke mutat. Az energia az iterációk között is javult folyamatosan. Bár a második réteg hozzáadásakor a paraméterek

rendre 0.065, 0, 1.331, 0 voltak, az első 50 kiértékelés után γ_1 szöge jelentősen változott. Már ez is mutatja, hogy nem csak az új paraméterekkel való finomításról van szó, hanem a régiek szignifikáns változásáról is, ami előre jelezheti, hogy a rétegezett megközelítés nem feltétlenül hatékonyabb.

És valóban, a következő táblázaton a standard QAOA eredményei láthatók ugyanezen a gráfon $p = 2$ mellett.

p -érték	Kiértékelések száma	Energia	γ_1	γ_2	β_1	β_2
2	50	-7.89	0.017	-0.091	1.21	0.166
2	200	-8.14	0.031	-0.125	1.265	0.109
2	350	-8.3	0.038	-0.159	1.324	0.097
2	500	-8.438	0.047	-0.198	1.369	0.091
2	650	-8.4689	0.0524	-0.2215	1.383	0.0868
Célfüggvény értéke: 402.0						

Táblázat 5.2.2: Standard QAOA futási adatai $p = 2$ esetén

A táblázatból az látható még, hogy a két módszer nagyjából ugyanannyi iterációszámmal futott le, és az optimum értékére és helyére is nagyon hasonló értékeket talált. Ez a fent leírtakat igazolja, miszerint az L-QAOA ilyen paraméterezéssel ezen a problémán nem hatékonyabb, általános esetben pedig (a következő alfejezetben látható módon) még rosszabb is.

5.3. L-QAOA egyértelmű paraméterezése reguláris gráfokon

Ezt a problémát máshonnan is megvizsgáltam - ha $p \rightarrow \infty$ akkor a QAOA optimális. De a probléma méretétől függően mikor lesz a hibája annyira kicsi, hogy jelentkeznek a fizikai korlátai, és nem tud tovább javulni? Illetve, hogy létezik-e ilyen érték?

A gondolat mögötte, hogy egy optimum lokális környezetéből nem tud már kitalálni az algoritmus, mert ahhoz olyan fázisátalakuláson kellene átmennie, amit a zaj és a fizikai korlátok az algoritmus azon részében már nem engednek.

Ez az L-QAOA-nak is jelentős hátrányt okoz, ugyanis az alapötlet, hogy egy irányba

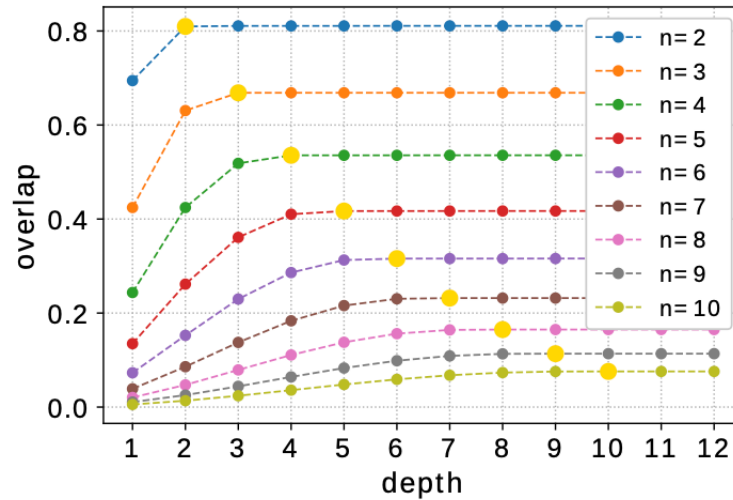
induljon az optimalizálás, és azt finomítsa a globális optimum irányába, pont a lokális optimum felé tereli, még a barren platók elkerülése mellett is.

A fenti két táblázat ezt is igazolja, de mielőtt több és nagyobb problémákon kipróbálhattam volna, felkerült arxiv-ra a másik cikk²⁶, amely pontosan ezt írja le, és magyarázza. A munkájukról nem tudtam, nagyjából egy fél éves lépéshátrányban voltam végig ez alapján. Az eredmény viszont igazoló ismét. A cikkben a globális optimummal való összehasonlításképp a mérésre alkalmas célfüggvény maga a pontosság. Ezt két állapot átfedésével lehet mérni, ez pedig matematikailag egy belső szorzat négyzetét jelenti az állapotvektorok között

$$o = |\langle opt | \psi_p(\gamma, \beta) \rangle|^2$$

ahol o az átfedés (overlap), opt az optimumhelyen jelölő állapot vektora (target state), ψ_p pedig a QAOA aktuális futásának eredménye (trial state). Mivel opt és ψ_p egységvektorok, $0 \leq o \leq 1$.

A következő kép azt mutatja, hogy egyes p -értékek esetén meddig nőtt az átfedés. A függőleges oszlop az átfedés mértéke, a vízszintesen p lett ábrázolva, n pedig az a p -érték, amely felett már nem javít új réteg hozzáadása.



Ábra 5.3.1: az L-QAOA szaturációja

A cikk még összehasonlítja a standard QAOA globális megoldásával, ami nagyobb p esetén szigorúan jobb megoldást talál.

A Fixed angle conjecture-ből és a tanítás szaturációjából következik, hogy reguláris gráfokon az L-QAOA-nak optimalizálás szempontjából egyértelműen létezik optimális paraméterezése, és ezzel nem lehet jobb eredményt elérni, mint a standard QAOA használatával.

Ez az eredmény azt jelenti, hogy ismert struktúrájú klasszikus alaphalmazú¹, kombinatorikus optimalizálási feladatokon, habár az L-QAOA elkerüli a standard változat számára problémás barren plató jelenséget, a standard QAOA jobban teljesít. Ennek az optimalizálónak pedig ezek a feladatok jelentik a klasszikus algoritmusok feletti előny lehetőségét, illetve jelenleg a kapu alapú kvantum optimalizálás létjogosultságát a globális optimalizálás területén.

¹Azaz a változók nem kvantumfizikai objektumok

Kitekintés

QAOA megoldásainak klasztereződése

A dolgozatomban során említést teszek arról, hogy reguláris, irányítatlan és súlyozatlan gráfok esetén a QAOA optimális paraméterei a MAXCUT problémán megegyeznek, a gráf egyéb struktúrájától függetlenül. Experimentálisan belátott tény továbbá, hogy a csúcsok átcímkezésére, permutálására sem érzékeny az algoritmus. Tehát ha egy könnyebb problémapéldányon megtalálja az algoritmus az optimumot, akkor a megtalált γ és β paraméterekkel bármilyen, azonos klaszterbe tartozó gráf esetén ugyanezekkel a paraméterekkel azonnal, direkt futtatással megtalálható az optimum helye és értéke. A többféle problémán megfigyelt klasztereződés²⁸ ezen a problémán matematikailag a legjobban struktúrálható.

Kérdéses, hogy az Ising modellbe való elkódolás, illetve a QAOA paramétertere milyen gráfstruktúrák esetén mutatja ezt a jelenséget, és miért. A MAXCUT problémára nagyon sok másik probléma vezethető vissza, így ha pontosan bebizonyítható, gráfok milyen struktúrájára nem érzékeny a paraméterezés, azaz milyen gráfok tartoznak egy klaszterbe a QAOA optimális szögeivel való reprezentálásukban, akkor akár jelentős ipari problémák is könnyedén megoldhatóvá válnának.

Az eredmények nemzetközi konferencián lettek bemutatva áprilisban az Európai Unió Quantum Flagship programjának részeként, munkám folytatásaként nemzetközi kollaborációban kutatom tovább a jelenséget, illetve matematikai hátterét.

Variációs kvantum áramkörök a gépi tanulásban

A VQE algoritmus feladata végső soron univerzális függvényapproximálás - a klasszikus optimalizáló felette csak a VQE eredményét javítja, amit ha minden pontra nézünk, akkor a klasszikus optimalizáló csak egy hibafüggvény-minimalizálást hajt végre.

Egyik oldalról a gépi tanulás felfogható egy ismeretlen függvény közelítésének, egy n változós függvénnyel. Ilyen módon a VQE is felfogható gépi tanulási modellnek, és részben azért, mert jelenlegi gyakorlati felhasználásokat is tud felmutatni a kvantum kémiában²⁹.

A kvantum gépi tanulási modellek is paraméteres áramkörökkel vannak megvalósítva, ugyanis a parametizált áramkörök univerzálisak, így lényegében a paraméterek megválasztásával tetszőleges függvényt ábrázolhatunk valamekkora hibával.

A kvantum gépi tanulás területén a robosztusabb modellek megjelenését a deriválható aktivációs függvények hiánya gátolja meg. Az aktivációs függvények egy fontos tulajdonsága, hogy nemlineárisak, a kvantumszámításban pedig az egyetlen nemlineáris operátor a mérés, ami teljesen elpusztítja a mért állapotot.

Ennek ellenére többféle modell létezik, a legtöbb klasszikus modellek által inspirált kvantum verziók, amelyek működnek, de nem jobban a klasszikus megfelelőiknél. A kvantumszámítás Turing teljes, így az összes klasszikus modell megvalósítható kvantumszámítógépen is, de ez természetesen nem jelent előnyt.

A klasszikus modellek által inspirált modellek nagyrészt perceptron megvalósítások, és neuronhálók. A terület szélességét szemlélítve ezek közül bemutatom a jelentősebbeket pár mondatban.

A legkisebb ilyen a Kvantum Boltzmann Gép³⁰ (Quantum Boltzmann Machine, QBM). Ez a kicsi, 2 réteges modell a QAOA-hoz hasonlóan a transzverzális Ising modellt használja ki, csak a hálós struktúrát és a mögöttes ötletet merítve a klasszikus modellből. Ugyanakkor bármilyen klasszikus feladatot képes ellátni, ahogyan azt az eredeti cikk bemutatja. A cikk bemutatja az Restricted Boltzmann Machine kvantum verzióját is.

Érdekes modellnek tartom a kvantum áramkör Born gépet³¹ (Quantum Circuit Born Machine, QCBM), amely adverzariális natív kvantum modell. A legtöbb QML modellel ellentétben nem klasszikus modell “kvantumositott” implementációja, és olyan problémát használ ki, amelyben a kvantumszámítógépek jelentős előnyt élveznek, mégpedig a felépített/tanult valószínűségi eloszlásból történő mintavételezést.

Ezekon kívül kvantum autoencoder, sőt még kvantum adverzariális háló (QGAN) is létezik például.³² Skálázható neuronháló modellekhez szükséges perceptron megvalósítások ugyanakkor az aktiváció függvény problémájával küzdenek. A különböző megvalósításokat a Microsoft szedte össze még 2016-ban³³, azóta újabb, teljesen új ötletként “tüzelő neuronmodell”³⁴ (spiking neuron) jelent meg tavaly áprilisban, amely új irányba vezetheti a kvantum gépi tanulás területét.

Egyéb problémák

A jelenlegi eredményeket érdemes egy kis fenntartással kezelni. Az egyetlen gyakorlati eredmények valós méretű demonstrálására képes eszköz, módszer a DWave eszközein való kvantum annealing megvalósítása. Az adiabatikus számítást kihasználó optimalizálási és gépi tanulási modellekben matematikailag bizonyítottan nagy a potenciál, amit kizárólag akkor lehet realizálni, ha sikerül robosztusabb, és hibatűrőbb kvantumszámítógépeket építeni.

Még egy szempontot érdemes figyelembe venni: sokszor klasszikus¹ adaton próbáljuk mérni a kvantumoptimalizálók hatékonyságát, illetve ezen összemérni a klasszikus optimalizálókkal. Ez nem egyenlő összehasonlítás. Klasszikus adaton kvantum optimalizálás (CQ optimization) nem olyan hatékony, csakúgy mint a fordított, QC optimalizálás, ahol kvantum adaton klasszikusan próbálunk optimalizálni. A kvantum problémák ábrázolása klasszikusan általában exponenciálisan növeli a probléma méretét. Sokkal hatékonyabb, főleg a jelenlegi, zajos, kis méretű kvantum (NISQ) korszak eszközein a QQ optimalizálás, amelynek előnye is van a QC felett. Ugyanakkor a CC is hatékonyabb, mint a CQ a problémák nagyrészen.

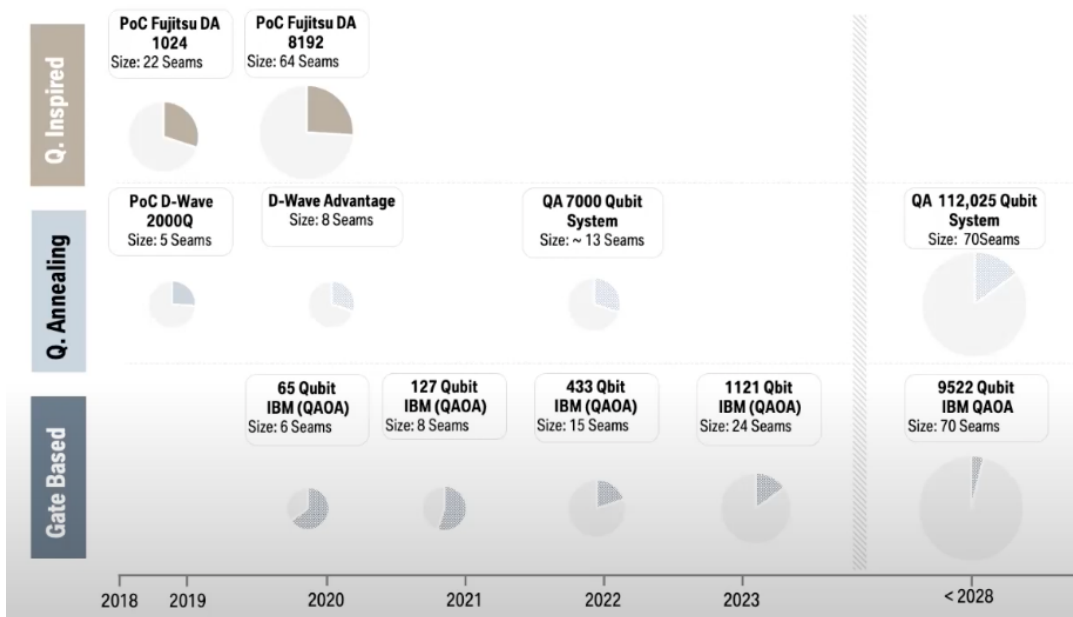
Tehát a jelenlegi problémák: a legtöbb adat klasszikus (adat), ezen kis méretű kvantumszámítógépek egyelőre nem tudnak jelentős eredményeket felmutatni (méret). Ezenkívül a kis méret mellett a zajra még nincs hatékony hibajavítás (zaj), emiatt hosszabb futásidejű kvantum algoritmusokat sem lehet egyelőre megvalósítani (pontosság).

Jelen és jövő

Az előző pontban tárgyalt problémák egyike sem elméleti fal a fejlődés előtt, a terület gyorsan fejlődik, és sokan ebben látják a jövőt. A kvantum információelmélet tavalyi konferenciáján több ipari szereplő is kapott teret, például Johannes Klepsch (BMW), akinek a beszédéből³⁵ egy ábrát emelnék ki:

Az ábrán különböző optimalizálásra használt (nem feltétlenül univerzális számításra képes) eszközök láthatók. Az ábra felső sorát kvantumtechnológiák által inspirált esz-

¹bitsorozatként eltárolt és reprezentált



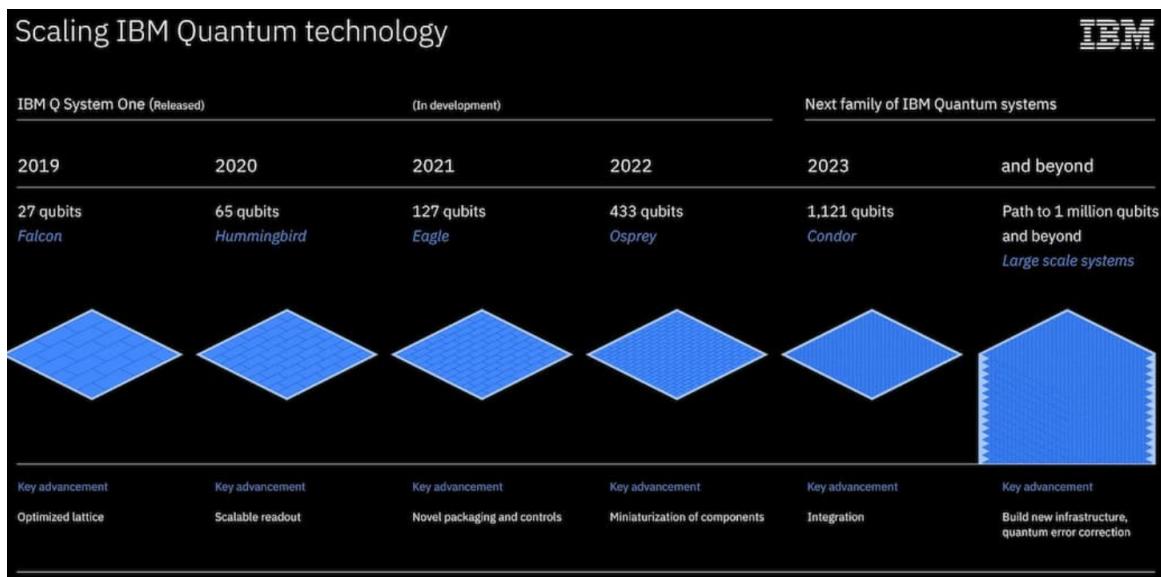
Ábra 6.0.1: Méret és skálázhatóság különböző kvantum technológiák esetén

közök alkotják, a középső sorát Quantum Annealerek, az alsót pedig a kapualapú kvantumszámítás jelenlegi, és a várakozások alapján jövőbeli vezető eszközei. Ez alapján 5 éven belül való életbeli problémákon, gyakorlatban is használható univerzális kvantumszámítógépekkel optimalizálhatunk.

Ezenkívül elméleti téren is sokat fejlődhet még a terület a következő években. A VQE-nek sokféle verziója jelent meg³⁶, a kvantum gépi tanulás nagyon gyorsan fejlődő területté vált, és a hardveres problémákra is folyamatosan jelennek meg ötletek és megoldások, és az anyagi háttere is biztosított a fejlődésnek.

Több startup is indult kvantumos hibajavítás, és fordítók terén, hirtelen nagyon fontossá vált az áramkörök optimális fordítása, és a logikai qubitek megvalósítása. Azaz hibajavított és fordított számítás megvalósítása többféle módon^{37,38}, felépítve egy teljes hardver-szoftver vermet a kvantumszámítógépek fölé.

A fejlődés mértékét szemléltetendő, az IBM rendszeresen publikálja³⁹ a tervezett fejlődési ütemtervét, amit eddig (2021, 2022) meglepő pontossággal sikerült tartaniuk. Fontos megemlíteni, hogy a számítógépek tervezésekor nem csak a qubitek száma (scale), hanem a pontosságuk (accuracy) és számítási kapacitásuk (volume) is ugyanolyan szempont kell legyen.



Ábra 6.0.2: Az IBM tervezett hardveres előrehaladása

Ezenkívül új fejlesztői környezetek is jelennek meg, hogy segítsék a kvantum optimalizálás fejlődését, elérhetővé tegyék bárki számára az akadémiából vagy az iparból.

Az optimalizálás területén a kvantum optimalizálásnak tehát leginkább a kvantum adaton való optimalizálásban, illetve a kombinatorikus optimalizálásban lehet előnye, ami proof of concept szintjén, kis problémákon experimentálisan is bizonyított. Éveken belül pedig mind a kvantumszámítás elméleti, mind a gyakorlati, hardveres részének fejlődése lehetővé teszi a kvantum optimalizálás lehetőségeinek realizálását valós méretű problémák esetén is.

Szeged, 2022. április 10.

Köszönetnyilvánítás

Szeretném megköszönni Gazdag-Tóth Boglárkának az inspirációt, motiváló téma-vezetést, illetve a folyamatos együttműködést a dolgozat megírása során. Köszönöm továbbá Shannon Whitlocknak a segítségét, hogy az interdiszciplináris terület több oldaláról is megismerhettem a dolgozatom témájával foglalkozó embereket, illetve a lehetőséget, hogy a témához kapcsolódó kurzusokon részt vehetek Európaszerte. Szeretnék köszönetet mondani Clara Wassnernek, akinek az eredményei további inspirációt nyújtottak a kutatáshoz, illetve akivel közösen folytatjuk tovább az elkezdett munkát. Emellett szeretném még megemlíteni Bánhelyi Balázs hozzájárulását, aki a Covid bezárás alatt támogatta a kutatómunkámat.

Irodalomjegyzék

- [1] P.W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. *Proceedings 35th Annual Symposium on Foundations of Computer Science*, 1994.
- [2] Google and NASA. Quantum supremacy using a programmable superconducting processor. *Nature*, 574:505–510, 23 Oct. 2019.
- [3] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O’Brien. A variational eigenvalue solver on a quantum processor. *Nature*, (473), 2011.
- [4] Sam Gutmann Edward Farhi, Jeffrey Goldstone. A quantum approximate optimization algorithm. *arXiv*, 14 Nov 2014.
- [5] Xiaoyuan Liu, Anthony Angone, Ruslan Shaydulin, Ilya Safro, Yuri Alexeev, and Lukasz Cincio. Layer vqe: A variational approach for combinatorial optimization on noisy quantum computers. *IEEE Transactions on Quantum Engineering*, 3, 04 Jan. 2022.
- [6] Paul Benioff. The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by turing machines. *Journal of Statistical Physics*, 22:563–591, 1980.
- [7] Paul Benioff. Quantum mechanical hamiltonian models of turing machines. *Journal of Statistical Physics*, 29:515–546, 1982.
- [8] Jay Gambetta. Ibm’s roadmap for scaling quantum technology. *IBM*, 15 Sep 2020.
- [9] Johnson M. and Gildert S. Amin M. Quantum annealing with manufactured spins. *Nature*, 473:194–198, 2011.
- [10] Jarrod R. McClean, Jonathan Romero, and Alán Aspuru-Guzik Ryan Babbush. The theory of variational hybrid quantum-classical algorithms. *New J. Phys*, (18), 14 Sep 2015.

- [11] Charlotte Froese Fischer. General hartree-fock program. *Computer Physics Communications*, 43:355–365, 1987.
- [12] Joonho Lee, William J. Huggins, Martin Head-Gordon, and K. Birgitta Whaley. Generalized unitary coupled cluster wave functions for quantum computation. *Journal of Chemical Theory and Computation*, 15(1):311–324, 2019.
- [13] Abhinav Kandala. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*, 549:242, 2017.
- [14] Bruno Apolloni, Nicolo Cesa-Bianchi, and Diego De Falco. A numerical implementation of quantum annealing. *Stochastic Processes, Physics and Geometry, Proceedings of the Ascona-Locarno Conference*, 1988.
- [15] Bruno Apolloni, Maria C. Carvalho, and Diego De Falco. Quantum stochastic optimization. *Stoc. Proc. Appl.*, 33(2), 1989.
- [16] S. Kirkpatrick, C. D. Gelatt Jr, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598), 1983.
- [17] M. Johnson, M. Amin, and S. et al. Gildert. Quantum annealing with manufactured spins. *Nature*, 10 Apr 2013.
- [18] A.B. Finnila, M.A. Gomez, C. Sebenik, C. Stenson, and J.D. Doll. Quantum annealing: A new method for minimizing multidimensional functions. *Chemical Physics Letters*, 219(5-6), 1994.
- [19] T. Kadowaki and H. Nishimori. Quantum annealing in the transverse ising model. *Phys. Rev. E.*, 58, 1998.
- [20] DWave Systems. Annealer eszközök listája. <https://www.dwavesys.com/solutions-and-products/systems/>, 2022. [Online; látogatva 2022. 04. 01. 17:12].
- [21] M. Born and V. A. Fock. *Phys. Rev. A*, 51(3-4), 1928.

- [22] Yin Sun, Jun-Yi Zhang, Mark S. Byrd, and Lian-Ao Wu. Adiabatic quantum simulation using trotterization. *arXiv*, 2, 1 Jun 2018.
- [23] Vadim N. Smelyanskiy Ryan Babbush Hartmut Neven Jarrod R. McClean, Sergio Boixo. Barren plateaus in quantum neural network training landscapes. *Nature*, 9(4812), 29 Mar 2018.
- [24] G. G. Guerreschi and A. Y. Matsuura. Qaoa for max-cut requires hundreds of qubits for quantum speed-up. *Scientific Reports*, 9(6903), 2019.
- [25] Michel X. Goemans and David P. Williamson. A general approximation technique for constrained forest problems. *Society for Industrial and Applied Mathematics*, 24(2), April 1995.
- [26] E. Campos, D. Rabinovich, V. Akshay, and J. Biamonte. Training saturation in layerwise quantum approximate optimization. *Phys. Rev. A*, 104:L030401, Sep 2021.
- [27] Jonathan Wurtz and Danylo Lykov. Fixed-angle conjectures for the quantum approximate optimization algorithm on regular maxcut graphs. *Phys. Rev. A*, 17 Nov. 2021.
- [28] V. Akshay, D. Rabinovich, E. Campos, and J. Biamonte. Parameter concentrations in quantum approximate optimization. *Phys. Rev. A*, 104:L010401, Jul 2021.
- [29] Daniel Claudino, Jerimiah Wright, Alexander J. McCaskey, and Travis S. Humble. Benchmarking adaptive variational quantum eigensolvers. *Frontiers in Chemistry*, 8, 2020.
- [30] Mohammad H. Amin, Evgeny Andriyash, Jason Rolfe, Bohdan Kulchytskyy, and Roger Melko. Quantum boltzmann machine. *Phys. Rev. X*, 8:021050, May 2018.
- [31] Jin-Guo Liu and Lei Wang. Differentiable learning of quantum circuit born machines. *Phys. Rev. A*, 98:062324, Dec 2018.
- [32] Marcello Benedetti, Erika Lloyd, Stefan Sack, and Mattia Fiorentini. Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, 4, 10 2019.

- [33] Nathan Wiebe, Ashish Kapoor, and Krysta M. Svore. Quantum perceptron models. *arXiv*, 2016.
- [34] Lasse Bjørn Kristensen, Matthias Degroote, Peter Wittek, Alán Aspuru-Guzik, and Nikolaj T. Zinner. An artificial spiking quantum neuron. *npj Quantum Information*, 7(1), 12 Apr 2021.
- [35] Johannes Klepsch. Application of quantum computing in the automotive value chain. <https://www.youtube.com/watch?v=GIy26-s6whY>, 2021. [Online; látogatva 2022. 04. 05. 10:03].
- [36] Jules Tilly, Hongxiang Chen, Shuxiang Cao, Dario Picozzi, Kanav Setia, Ying Li, Edward Grant, Leonard Wossnig, Ivan Rungger, George Booth, and Jonathan Tennyson. The variational quantum eigensolver: a review of methods and best practices. 11 2021.
- [37] Kilian Ender, Roeland ter Hoeven, Benjamin E. Niehoff, Maike Drieb-Schön, and Wolfgang Lechner. Parity quantum optimization: Compiler, 2021.
- [38] Maike Drieb-Schön, Younes Javanmard, Kilian Ender, and Wolfgang Lechner. Parity quantum optimization: Encoding constraints, 2021.
- [39] IBM. Ibm’s roadmap for building an open quantum software ecosystem. <https://research.ibm.com/blog/quantum-development-roadmap>, 2020. [Online; látogatva 2022. 04. 05. 11:32].