

Gibbsmålfunktion för modifiering av optimeringsprocessen i en hybridkvantalgoritm

- En numerisk jämförelse av prestandan för en Gibbsfunktion och väntevärdet av Hamiltonianen som målfunktion i en hybridkvantalgoritm applicerad på Max-Cut problemet.

Kandidatarbete i Teknisk fysik

William Colliander, Pontus Lindgren, Oscar Stommendal,
Elias Svensson & Felix Uddén

KANDIDATARBETE 2023

Gibbsmålfunktion för modifiering av optimeringsprocessen i en hybridkvantalgoritm

- En numerisk jämförelse av prestandan för en Gibbsfunktion och väntevärdet av Hamiltonianen som målfunktion i en hybridkvantalgoritm applicerad på Max-Cut problemet.

William Colliander
Pontus Lindgren
Oscar Stommendal
Elias Svensson
Felix Uddén



CHALMERS

Institutionen för mikroteknologi och nanovetenskap (MC2)
WACQT CHALMERS TEKNISKA HÖGSKOLA
Göteborg, Sverige 2023

Gibbsmålfunktion för modifiering av optimeringsprocessen i en hybridkvantalgoritm

– En numerisk jämförelse av prestandan för en Gibbsfunktion och väntevärdet av Hamiltonianen som målfunktion i en hybridkvantalgoritm applicerad på Max-Cut problemet.

© William Colliander, Pontus Lindgren, Oscar Stommendal, Elias Svensson & Felix Uddén, 2023.

Handledare: Laura García Álvarez & Göran Johansson, institutionen för mikroteknologi och nanovetenskap

Examinator: Per Lundgren, institutionen för mikroteknologi och nanovetenskap

Kandidatarbete 2023

Institutionen för mikroteknologi och nanovetenskap

WACQT

Chalmers tekniska högskola

SE-412 96 Göteborg

Telefon +46 31 772 1000

Omslag: Nivåkurvor för Gibbsfunktionen vid lösning av Max-Cut problemet för en viss graf. Se figur 4.6 för utförligare beskrivning.

Typsatt i L^AT_EX

Göteborg, Sverige 2023

Sammandrag

Denna studie har jämfört en modifierad version av hybridkvantalgoritmen Quantum Approximate Optimization Algorithm (QAOA) med standardversionen. Standardversionen av QAOA använder väntevärdet av energin, $\langle \hat{H}_C \rangle$, som målfunktion medan den modifierade versionen använder en Gibbsfunktion, $-\ln\langle e^{-\eta\hat{H}_C} \rangle$.

Syftet var dels att undersöka och reproducera resultatet från en tidigare studie från Li et al. [1] och dels att undersöka faktorer som inte utforskats i den tidigare artikeln: effekten av olika grafisomorfismer, grafstorlekar och djup på kvantkretsen.

Jämförelsen gjordes utifrån resultaten som erhölls då algoritmen applicerades på olika fall av Max-Cut problemet. Två olika prestandamått användes, framgångssannolikhet och approximationskvot. De olika faktorer som undersöktes var val av slutgiltiga optimerade parametrar, val av Gibbsparametern η , kretsdjup, grafstorlek och grafisomorfism. Alla resultat erhölls genom simulerings av algoritmerna i Python, med kodpaketet Pennylane [2].

Givet resultatet för de relativt små grafer som behandlats, är det inte möjligt att avgöra huruvida den modifierade algoritmen generellt är bättre än den ursprungliga. Det är till och med möjligt att den modifierade versionen är sämre än den ursprungliga, framför allt i fråga om approximationskvot. Det finns dock fall där den modifierade algoritmen överträffar standardversionen, i synnerhet i fråga om framgångssannolikhet.

Nyckelord: *QAOA, Hybridkvantalgoritm, Målfunktion, Max-Cut, Approximationskvot, Framgångssannolikhet, Optimering.*

Abstract

This study has compared a modified version of the hybrid quantum algorithm, Quantum Approximate Optimization Algorithm (QAOA), with the standard version. The standard version of QAOA uses the expectation value of the energy, $\langle \hat{H}_C \rangle$, as the objective function while the modified version uses a Gibbs function, $-\ln\langle e^{-\eta\hat{H}_C} \rangle$.

The aim was partly to investigate and reproduce the results of a previous study by Li et al. [1] and partly to investigate factors not explored in the previous paper: the effect of different graph isomorphisms, graph sizes and depth of the quantum circuit.

The comparison was made based on the results obtained from applying the algorithm to different cases of the Max-Cut problem. Two different performance measures were used, success probability and approximation ratio. The different factors investigated were the choice of final optimized parameters, the choice of the Gibbs parameter η , circuit depth, graph size and graph isomorphism. All results were obtained by simulating the algorithms in Python, using the code package PennyLane [2].

Given the results for the relatively small graphs considered, it is not possible to determine whether the modified algorithm is generally better than the original one. It is even possible that the modified version is worse than the original, especially in terms of approximation ratio. However, there are cases where the modified algorithm outperforms the standard version, particularly in terms of success probability.

Keywords: *QAOA, Hybrid Quantum Algorithm, Objective Function, Max-Cut, Approximation Ratio, Success Probability, Optimization.*

Förord

Vi vill tacka Laura García Álvarez, Göran Johansson och Hanna Linn. Tack till Laura för all uppmuntran och handledning. Tack till Göran för den tekniska förståelsen för kvantdatorer. Tack till Hanna för hjälp med Pennylane och för att texten är läsbar. Arbetet hade inte kunnat slutföras utan er hjälp.

William Colliander, Pontus Lindgren, Oscar Stommendal,
Elias Svensson & Felix Uddén

Göteborg, Maj 2023

Lista över akronymer

Nedan finns en lista över akronymer och begrepp som används flitigt i denna rapport, listade i ordning som de används.

NISQ Noisy Intermediate Scale Quantum	1
QAOA Quantum Approximate Optimization Algorithm	1
AR Approximationskvot (eng. Approximation Ratio)	2
SP Framgångssannolikhet (eng. Success Probability)	2

Nomenklatur

Nedan är nomenklatur för mängder, parametrar, variabler och begrepp som används i detta kandidatarbete.

Mängder

A, B, C	De olika grafmängder som används
---------	----------------------------------

Parametrar

$\vec{\gamma}$	Parameter som används med kostnadshamiltonianen
$\vec{\beta}$	Parameter som används med mixerhamiltonianen

Variabler

η	Variabeln som används i Gibbsfunktionen
p	Kretsens djup
n	Antalet kvantbitar
\hat{H}_C	Kostnadshamiltonianen
\hat{H}_M	Mixerhamiltonianen
$\hat{\sigma}^z$	Pauli-Z-matrisen

Begrepp

$-\ln\langle e^{-\eta\hat{H}_C}\rangle$	Gibbsfunktionen
$\langle\hat{H}_C\rangle$	Standardmålfunktionen

Innehåll

Lista över akronymer	ix
Nomenklatur	x
Figurer	xii
1 Introduktion	1
1.1 Syfte	2
1.2 Avgränsningar	2
2 Teori	3
2.1 Max-Cut	3
2.1.1 Kvantmekanisk beskrivning av Max-Cut	4
2.2 Kvantalgoritmer	4
2.3 Quantum Approximate Optimization Algorithm	5
2.4 Prestandamått	7
2.5 Gibbsfunktionen	7
2.6 Grafteori	8
3 Metod	8
3.1 Valet av parametern η	9
3.2 Valet av slutparametrar	9
3.3 Konvergenskriterier	10
4 Resultat	10
4.1 Framgångssannolikhet	11
4.2 Approximationskvot	14
4.3 Jämförelse av kostnadsfunktionernas energilandskap	16
4.4 Valet av parametern η	19
5 Diskussion	21
5.1 Framtida studier	23
6 Slutsats	24

Figurer

2.1	Max-Cut-problemet: grafens noder har delats upp så att antalet kanter som förbinder nodmängderna maximeras. Uppdelningen gör att fyra kanter binder samman mängderna.	3
2.2	Schematisk bild över hur QAOA fungerar med en Gibbsfunktion som styr-funktion. \hat{H}_C är kostnadshamiltonianen, \hat{H}_M är mixerhamiltonianen, n är antal kvantbitar, p är kretsdjupet, optimeringsparametrarna betecknas $\vec{\gamma}, \vec{\beta}$ och hyperparametern betecknas η	5
2.3	Exempel på hur sannolikheten för små cykler är liten i större grafer. Sannolikheten att nod 3 kopplar till just nod 1 och skapar en loop är $\frac{1}{4}$ gentemot sannolikheten $\frac{3}{4}$ att koppla till en av noderna 4, 5 eller 6 (om grafen är 2-reguljär). Om man hade lagt till fler noder hade denna sannolikhet varit ännu lägre.	8
4.1	SP för graferna i mängderna A, B och C för Gibbs- respektive standardversionen av QAOA. En standardavvikelse är markerad vid varje punkt och markören visar medelvärdet. De olika figurerna är gjorda med olika djup (a) $p = 1$, (b) $p = 2$ och (c) $p = 3$. För dessa är slutparametrarna valda utifrån vilka som gav högst AR, se (1) i avsnitt 3.2.	11
4.2	SP för graferna i mängderna A, B och C för Gibbs- respektive standardversionen av QAOA. En standardavvikelse är markerad vid varje punkt, där markören visar medelvärdet. De olika figurerna är gjorda med olika djup (a) $p = 1$, (b) $p = 3$, (c) $p = 4$ och (d) $p = 5$. För dessa är slutparametrarna valda utifrån vilka som optimeringen konvergerat till, se (2) i avsnitt 3.2.	12
4.3	Sannolikhetsdistributionen för sluttillståndets energi för grafen med 12 noder ur grafmängd A med djup $p = 3$. Markörerna i figuren visar medelvärdet medan felstaplarna visar det högsta respektive lägsta värdet. Pilarna längst upp i figuren markerar den genomsnittliga Max-Cut kostnaden för Gibbsfunktionen respektive standardmålfunktionen. Max-Cut kostnaden är proportionell mot AR. För dessa är slutparametrarna valda utifrån vilka som optimeringen konvergerat till, se (2) i avsnitt 3.2.	13
4.4	AR för graferna i mängderna A, B och C för Gibbs- respektive standardversionen av QAOA. En standardavvikelse är markerad vid varje punkt, där markören visar medelvärdet. De olika figurerna är gjorda med olika djup (a) $p = 1$, (b) $p = 2$ och (c) $p = 3$. För dessa är slutparametrarna valda utifrån vilka som gav högst AR, se (1) i avsnitt 3.2	14
4.5	AR för graferna i mängderna A, B och C för Gibbs- respektive standardversionen av QAOA. En standardavvikelse är markerad vid varje punkt, där markören (notera skillnaden mellan Gibbsversionen och standard) visar medelvärdet. Figurerna är gjorda med olika djup (a) $p = 1$, (b) $p = 3$, (c) $p = 4$ och (d) $p = 5$. För dessa är slutparametrarna valda utifrån vilka som optimeringen konvergerar till, se (2) i avsnitt 3.2	15

- 4.6 (a) och (b) visar landskapet för Gibbsfunktionen respektive standardmålfunktionen för grafen med storlek 16 i grafmängd A. (c) och (d) visar landskapet för Gibbsfunktionen respektive standardmålfunktionen för grafen med storlek sju som inte ingår i grafmängd A, B eller C. Notera i (a) att iterationen indikerad med en etta i övre vänstra hörnet inte konvergerar till ett lokalt minima trots att motsvarande iteration för standardmålfunktionen i (b) konvergerat. I båda standardlandskapen (c) och (d) finns flera iterationer som inte konvergerar till lokala miniman. Trianglarna visar var iterationerna börjar och kryss där de slutar.

4.7 Figuren visar höjdskillnad respektive landskapet för grafen med storlek sju som inte ingår i grafmängd A, B eller C. (a) visar höjdskillnaden för Gibbsfunktionens landskap, alltså största minus minsta värdet. x -axeln visar k där $\eta = \frac{k}{E_{gs}}$ där $E_{gs}=10$. (b) visar Gibbsenergilandskapet med $k = 1$. Värdena i landskapet är omskalade så att det maximala värdet i landskapet förskjuts till noll. Notera att iterationen indikerad med en tvåa inte hinner konvergera till ett lokalt minima.

4.8 Figuren visar, för grafen med storlek 16 i grafmängd A och $p = 1$, hur både SP (med negativt tecken) (a) och Gibbsfunktionen (b) har lokala miniman omkring ($\gamma/\pi \approx -0,825$, $\beta/\pi \approx -0,125$) medan standardkostnadsfunktionen (c) har minima omkring ($\gamma/\pi \approx -0,84$, $\beta/\pi \approx -0,11$). Det tycks alltså finnas en mätbar förskjutning mellan de lokala miniman för SP (med negativt tecken) och Gibbs- relativt standardkostnadsfunktionens minima. Gula och röda trianglar markerar datapunkter som är bland de 0,2 % respektive 0,9 % lägsta värdena för respektive landskap.

4.9 Figuren visar SP för olika värden på k , där $\eta = k/E_{gs}$, med olika djup (a) $p = 1$, (b) $p = 2$ och (c) $p = 3$ för grafmängd A. I (a) visas en inzoomning för nodstorlekar åtta till 16. Slutparametrarna valdes här enligt vad optimeringen konvergerade till och för varje graf gjordes fem initialiseringar med olika par av parametrar utspridda i energilandskapet. Varje markör visar ett medelvärde och felstaplarna motsvarar en standardavvikelse.

4.10 AR kvot och SP kvot för iterationer med samma initialparametrar för Gibbs- och standardversionen. Datatan som visas är för grafen med storlek sju som inte ingår i A, B eller C. Datatan har filtrerats så att enbart de åtta punkter där Gibbs- och standardversionen haft minst avstånd mellan sig används. Som x-axel används k där $\eta = \frac{k}{E_{gs}}$. E_{gs} är grundtillståndsenergin för grafen och k är en vald konstant. För denna graf är $E_{gs} = 10$

4.11 AR kvot och SP kvot för iterationer med samma initialparametrar för Gibbs- och standardversionen. Datatan som visas är för grafen med nodantal tolv i A. Datatan har filtrerats så att enbart de åtta punkter där Gibbs- och standardversionen haft minst avstånd mellan sig används. Som x-axel används k där $\eta = \frac{k}{E_{gs}}$. För denna graf är $E_{gs} = 22$

1 Introduktion

Ett område som är relativt nytt inom vetenskapen är kvantdatorer. De senaste åren har utvecklingen och intresset kring dessa skjutit i höjden, vilket framträder i form av stora investeringar [3], [4] och ökat mediaintresse [5]. Om de uppnår sin fulla potential, kan de kanske ha en betydande roll i utvecklingen av framtidens vetenskap och industri, bland annat inom ekonomi [6] och miljö [7].

Fördelarna med en kvantdator relativt en klassisk dator uppkommer på grund av två kvantmekaniska fenomen, superposition och sammanflätning. Till skillnad från klassiska datorer, som processerar data genom klassiska bitar, använder kvantdatorer istället kvantbitar. En kvantbit kan, liksom en klassisk bit, anta värdena 0 och 1. En skillnad är att kvantbitar kan befina sig i ett tillstånd som är en kombination av 0 och 1, en så kallad superposition. Kvantbitar har även möjligheten att samverka med andra kvantbitar, så kallad sammanflätning. Sammanflätning innebär att kvantbitarna inte kan beskrivas med separata vågfunktioner.

Superposition och sammanflätning gör det möjligt att med n kvantbitar hålla reda på och bearbeta sannolikheten för alla de 2^n bitsträngar som kan bildas av n bitar. Det här innebär att en kvantdator, teoretiskt sett, kan bearbeta avsevärt mycket mer information samtidigt jämfört med en klassisk dator. Ett exempel är Peter Shors kvantalgoritm för faktorisering av stora tal, som i teorin kan ge en tidsbesparing i beräkningen relativt klassiska metoder som ökar exponentiellt med talets storlek [8]. Generellt gäller att större kvantmekaniska system är omöjliga att simulera på en klassisk dator, på grund av att vanliga bitar inte är sammanflätade och således behövs i orimliga mängder. En idé är att sådana system ska kunna simuleras med kvantdatorer för tillämpningar inom exempelvis kemi och medicin [9].

Dagens kvantdatorer, så kallade Noisy Intermediate Scale Quantum (NISQ)-datorer har vissa grundläggande begränsningar. Några av dessa är brus, koherenstid och antalet bitar. Brus är störningar i systemet som introduceras då det körs. Det här begränsar hur djupa kretsarna kan göras eftersom bruset ökar med djupet [10]. Koherenstiden begränsar hur länge det går att läsa ut användbar information från kvantbiten och är ungefär $10\ \mu\text{s}$ för supraleddande kvantbitar [11]. Storleken på dagens kvantdatorer är en begränsning eftersom antalet bitar direkt korrelerar till hur stora beräkningar som kan utföras. Begränsningarna gör att exempelvis Shors algoritm inte kan köras för stora tal då algoritmen kräver fler kvantbitar än vad som finns tillgängligt idag. Antalet kvantbitar begränsas av att det är mycket svårt att korrigera fel samt att bitarna interfererar med varandra [12].

I det här arbetet studeras hur kvantdatorer kan användas till att lösa optimeringsproblem, där den optimala lösningen utifrån ett visst antal förutsättningar söks. Dessa typer av problem uppkommer mycket ofta, både i industrin och inom akademien [13]. Det optimeringsproblem som ska studeras i det här arbetet är Max-Cut. Max-Cut går ut på att, given en viss graf, dela in dess noder i två mängder så att antalet kanter som kopplar samman mängderna maximeras. Max-Cut är ett komplext problem som dagens brusiga kvantdatorer teoretiskt sett skulle kunna lösa effektivare än klassiska datorer [14].

Studien kommer att utgå från en av de mest använda optimeringsalgoritmerna för kvantdatorer, Quantum Approximate Optimization Algorithm (QAOA). Denna algoritm har teoretiskt sett potential att kunna lösa optimeringsproblem som Max-Cut mer effektivt än klassiska datorer, vilket gör algoritmen aktuell [14]. QAOA är en så kallad

hybridalgoritm, vilket betyder att den består av en kvantmekanisk del och en klassisk del. Den kvantmekaniska delen tar in ett antal parametrar och ger ett tillstånd för kvantbitarna. Optimeringsproblemet som QAOA ska lösa, beskrivs på ett sätt så att det sökta svaret motsvarar det tillståndet för kvantbitarna som har lägst energi, det så kallade grundtillståndet. Genom att klassiskt optimera parametrarna som används i den kvantmekaniska delen närmar sig tillståndet det sökta grundtillståndet. Målet är inte att hitta det exakta grundtillståndet då detta är svårt och tidskrävande, istället hittas ett tillstånd som är nästan lika bra, det vill säga en kvalitativt lösning.

QAOA består av många olika delar. Dessa skulle kunna modifieras för att förbättra algoritmens prestanda. En sådan del är målfunktionen, alltså den funktion som ska minimeras i optimeringsproblemet. I standardversionen av QAOA är målfunktionen väntevärdet av kvantsystemets energi i slutet av kvantalgoritmen. Enligt Li et al. [1] kan en ändring av målfunktionens utformning ha positiv påverkan på algoritmens prestanda. Den målfunktion som föreslås av Li et al. som ett alternativ till väntevärdet av energin, $\langle \hat{H}_C \rangle$, är en Gibbsfunktion, $-\ln\langle e^{-\eta\hat{H}_C} \rangle$. Den här studien bygger vidare på arbetet av Li et al. genom att analysera Max-Cut problemet för fler testfall.

1.1 Syfte

Arbetet utforskar idén, ursprungligen från Li et al. [1], att användningen av Gibbsfunktionen som målfunktion kan ge en förbättring av QAOA. Det här åstadkoms genom att jämföra prestandamåttet approximationskvot och framgångssannolikhet för standardversionen av QAOA samt den modifierade versionen. Studien syftar även till att undersöka faktorer som inte utforskats i den tidigare artikeln. Dessa faktorer innefattar effekten av olika isomorfismer, grafstorlekar och djup på kvantkretsen. Utöver detta kommer även prestandans beroende på parametern η undersökas. Även valet av de slutgiltiga optimeringsparametrarna berörs.

1.2 Avgränsningar

Studien kommer att undersöka hur den modifierade versionen av QAOA presterar relativt standardversionen när den tillämpas på Max-Cut. Max-Cut undersöks för 4-reguljära grafer, som består av 5 – 16 noder. Studien har satt en övre gräns på 16 noder då större grafer tar mycket lång tid att simulera. Inom mängden av 4-reguljära grafer med samma nodantal finns många icke-isomorfa grafer och tiden ger inte möjlighet att studera alla dessa. Därför begränsas arbetet till tre icke-isomorfa grafer per nodantal där minst tre olika isomorfismer finns.

Vidare finns det en stor mängd prestandamått som skulle studeras för att dra slutsatser om hur den modifierade algoritmen presterar. I detta arbete kommer mestadels Approximationskvot (eng. Approximation Ratio) (AR) och Framgångssannolikhet (eng. Success Probability) (SP) studeras. Det här urvalet har gjorts då det intuitivt kan förväntas att Gibbsfunktionen kommer ge ett bra resultat för SP och ett annat resultat för AR. Se avsnitt 2.5 för detaljer. SP är också något som liknar prestationsmåttet som används av Li et al. [1]. Även andra sätt att karakterisera prestandan berörs, exempelvis en jämförelse av hur Gibbsversionen och standardversionen konvergerar mot lokala miniman. Det görs dock endast kvalitativt.

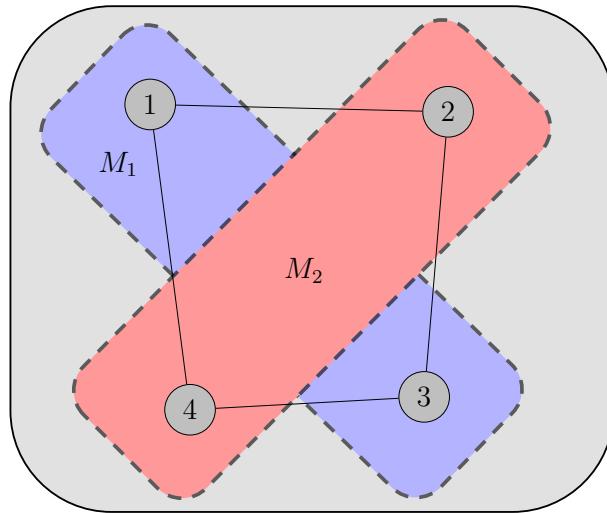
Studien begränsas också till att enbart undersöka hur kvantalgoritmen presterar utan brus, då simuleringar med brus är mycket tidskrävande. Vidare körs algoritmen med djup (se teoriavsnitt 2.3) mellan 1 – 5. Denna begränsning är också nödvändig med avseende på tiden, då ett större djup ökar tiden för en simulering avsevärt. Även den klassiska optimeraren, Gradient Descent, som används i QAOA kommer i studien att vara densamma för samtliga simuleringar. Detta för att begränsa hur mycket som varieras i studien. Gradienten beräknas med avseende på varje optimeringsparameter i optimeringsproblemet.

2 Teori

Kvantalgoritmer, QAOA specifikt i denna studie, har en teoretisk potential för en ökad beräkningshastighet för vissa komplexa problem jämfört med en klassisk dator. QAOAs prestanda är dock beroende på flertalet variabler och underliggande mekanismer som beskrivs i avsnittet nedan. Dessutom förklaras Max-Cut på ett djupare plan. Även prestandamåtten som används kommer att definieras. Sist nämns grafteoretiska resonemang om vad som sker för stora grafer.

2.1 Max-Cut

För att undersöka hur bytet av målfunktionen påverkar prestandan för QAOA användes optimeringsproblemet Max-Cut. Max-Cut är ett välstuderat optimeringsproblem och klassas som NP-fullständigt. Att ett problem har denna klassificering innebär att klassiska datorer idag inte kan lösa problemet på ett effektivt sätt.



Figur 2.1: Max-Cut-problemet: grafens noder har delats upp så att antalet kanter som förbinder nodmängderna maximeras. Uppdelningen gör att fyra kanter binder samman mängderna.

Problemet utgår från en graf $G(N, K)$, där N är dess noder och K dess kanter. Målet med Max-Cut är sedan att dela upp grafens noder i två delmängder, M_1 och M_2 , så att kanterna som förbinder dessa mängder maximeras, se figur 2.1. Figuren innehåller en graf med fyra noder, vilka delas upp i två mängder där antalet kanter som sammanbinder dessa mängder maximeras.

För varje optimeringsproblem finns en kostnadsfunktion som givet en lösning till problemet ger dess kostnad, vilken är ett mått hur bra lösningen är. I denna studie behandlas Max-Cut med oviktade grafer, vars kostnadsfunktion ges av antalet kanter som förbinder noder vilka ingår i olika mängder. Max-Cut kan formuleras matematiskt som ett minimeringsproblem där $|C|$ är antalet kanter som förbinder noder i olika mängder med kostnadsfunktionen:

$$C = - \left[\sum_{i,j \in \text{kanter}} 1 - \delta_{ij} \right]. \quad (2.1)$$

För ekvationen gäller det att i och j indexrar noder som binds samman av en kant samt δ_{ij} är 0, om noderna är i olika nodmängder, och 1 annars. Om noderna hör till olika nodmängder blir bidraget till summan -1 , annars 0. C blir alltså antalet kanter med noder som hör till olika delmängder, fast med negativt tecken.

2.1.1 Kvantmekanisk beskrivning av Max-Cut

För att simulera och hitta lösningar för Max-Cut på en kvantdator krävs en kvantmekanisk beskrivning av problemet. Beskrivningen görs genom att definiera en kostnadshamiltonian, notera likheterna med ekvation 2.1:

$$\hat{H}_C = -\frac{1}{2} \left[\sum_{i,j \in \text{kanter}} 1 - \hat{\sigma}_i^z \hat{\sigma}_j^z \right]. \quad (2.2)$$

För ekvationen gäller det att i och j indexrar noder som binds samman av en kant. Idén är att Paulimatrisserna $\hat{\sigma}_k^z$ ersätter δ_{ij} . Standardvalet är att välja kostnadsfunktionen $C = \langle \hat{H}_C \rangle$, vilket ger samma C som i den klassiska kostnadsfunktionen 2.1. Paulimatrisserna $\hat{\sigma}_k^z$ kan vid mätning av värdet på bit k ge tillbaka egenvärdena -1 eller 1 . När två noder, i och j ingår i olika delmängder (till exempel att bit i har värdet 1 och bit j värdet 0) kommer olika egenvärden att mätas från Paulimatrisserna. Med andra ord förbinder kanten mellan dessa noder de båda delmängderna, vilket översätts till ett totalt bidrag från den kanten till $\langle \hat{H}_C \rangle$ med -1 enligt ekvation 2.2 ovan. Om noderna istället ingår i samma mängd blir bidraget 0, eftersom samma egenvärde då mäts från $\hat{\sigma}_k^z$.

2.2 Kvantalgoritmer

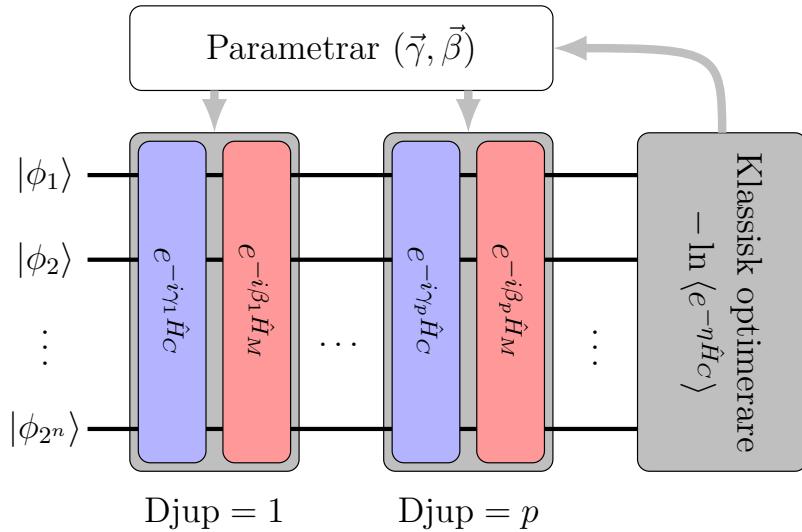
För att förstå QAOA behövs en mer detaljerad förståelse för hur kvantalgoritmer fungerar. I en kvantdator är en metod att använda en elektrons två olika spintillstånd, "upp" och "ner" för att definiera bitvärdena 1 och 0. Precis som i klassiska datorer kan n st bitar representera 2^n olika tal som i sin tur kan representera möjliga lösningar till ett optimeringsproblem. Den stora skillnaden mellan kvantdatorer och vanliga datorer är att n bitar kan representera alla 2^n st tal samtidigt i kvantdatorer. Möjligheten att representera flera tal samtidigt kommer från de kvantmekaniska egenskaperna superposition och sammanflätning. Ett kvanttillstånd med n bitar (exempelvis en samling av n sammanflätade elektroner) kan beskrivas enligt:

$$|\psi\rangle = \sum_{i=1}^{2^n} c_i |i\rangle. \quad (2.3)$$

I ekvationen är i värdet för ett av de 2^n binära talen och $|c_i|^2$ beskriver sannolikheten för att mäta det i :te binära talet. Det finns 2^n stycken c_i och dessa kan behandlas parallellt medan klassiskt endast en bitsträng kan behandlas åt gången. Idén bakom optimering med en kvantalgoritm är att öka värdet på $|c_i|^2$ för i som representerar bra lösningar på optimeringsproblemet och minska $|c_i|^2$ för sämre lösningar. Optimeringen av sannolikhetsamplituderna gör att mestadels tillstånd som representerar en bra lösning på optimeringsproblemet mäts, vilket fölaktligen ger ett lägre värde på kostnadsfunktionen.

2.3 Quantum Approximate Optimization Algorithm

För att utnyttja dagens kvantdatorers potentiella beräkningsfördelar är det nödvändigt att ta fram kvantalgoritmer som fungerar även med till exempel brus. En av dessa är QAOA [15]. QAOA är en hybridkvantalgoritm, vilket som tidigare nämnts innebär att den består av en kvantdel och en klassisk del, se figur 2.2.



Figur 2.2: Schematisk bild över hur QAOA fungerar med en Gibbsfunktion som styrfunktion. \hat{H}_C är kostnadshamiltonianen, \hat{H}_M är mixerhamiltonianen, n är antal kvantbitar, p är kretsdjupet, optimeringsparametrarna betecknas $\vec{\gamma}, \vec{\beta}$ och hyperparametern betecknas η .

Den klassiska delen i QAOA optimerar variablerna för den kvantmekaniska delen för varje iteration. Hur QAOA avbildar ett initialtillstånd till ett sluttillstånd som funktion av optimeringsparametrarna beskrivs matematiskt av:

$$|\Phi(\vec{\beta}, \vec{\gamma})\rangle = \prod_{l=1}^p e^{-i\gamma_l \hat{H}_C} e^{-i\beta_l \hat{H}_M} |\Phi_0\rangle. \quad (2.4)$$

Här är p kretsdjupet, $|\Phi_0\rangle$ initialtillståndet och l index för det aktuella lagret. Det har visats att större p ökar sannolikheten för ett önskevärt utfall hos algoritmen, men att algoritmen även för längsta kretsdjupet $p = 1$ har en viss garanterad prestanda [15]. Det är dock inte relevant att öka djupet till något som är för stort på grund av begränsningar som finns i dagens NISQ datorer. Det gör att vid ett visst p förloras mer på grund av det ökande bruset än vad som tjänas på ökat p [10].

De variabler som optimeras av den klassiska delen då algoritmen körs är $\vec{\beta}$ och $\vec{\gamma}$. Dessa variabler är vektorer och innehåller konstanterna som multipliceras med kostnads- och mixer-hamiltonianen för varje modul i algoritmen. $\vec{\beta}$ och $\vec{\gamma}$ kan tolkas som vinklar vilka bestämmer hur stor vridning som sker i Blochsfären.

De optimala vinklarna kommer erhållas i $\gamma \in [0, 2\pi)$ och $\beta \in [0, \pi)$, bland annat på grund av att Max-Cut-storleken är ett heltal [15]. Den initiala gissningen som görs för dessa parametrar (γ_1, β_1) kommer att påverka det slutgiltiga tillståndet och därmed också resultatet för algoritmen.

Genom att undersöka de optimala parametrarna $(\vec{\gamma}_p^*, \vec{\beta}_p^*)$ för olika djup p , kan observationen att $(\vec{\gamma}_{p+1}^*, \vec{\beta}_{p+1}^*)$ har stora likheter med $(\vec{\gamma}_p^*, \vec{\beta}_p^*)$ göras. Utifrån observationen kan följande initieringsstrategi formas: Börja på djup $p = 1$ och gör en initialgissning för $(\vec{\gamma}_1, \vec{\beta}_1)$. Kör QAOA för att hitta ett lokalt optimalt värde $(\vec{\gamma}_1^L, \vec{\beta}_1^L)$ och extrapolera därefter en initialgissning för djup $p = 2$ enligt:

$$[\vec{\gamma}_{p+1}]_i = \frac{i-1}{p} [\vec{\gamma}_p^L]_{i-1} + \frac{p-i+1}{p} [\vec{\gamma}_p^L]_i. \quad (2.5)$$

I ekvationen ovan är $[\vec{\gamma}_p]_i$ är det i :te elementet i γ -vektorn för djup p . $[\vec{\beta}_{p+1}]_i$ fås genom att byta ut γ mot β i ekvationen ovan. Repetera dessa steg tills det önskade djupet erhållits. Denna strategi kan, på en tid som ökar polynomieligt med djupet p , ge resultat som skulle kräva exponentiellt många försök med slumpmässigt valda initialparametrar att överträffa [16].

Kostnadshamiltonianen \hat{H}_C är den operator som beskriver själva minimeringsproblemet, hur denna härleds beskrivs utförligt under avsnitt 2.1.1. Hamiltonianen används i varje kretsenhet $e^{-i\gamma_k \hat{H}_C}$ där $1 \leq k \leq p$ och γ_k är optimeringsvariabeln. Mätningen som görs i slutet av algoritmen är i en icke-modifierad QAOA väntevärde för kostnadshamiltonianen. I en riktig kvantdator bestäms väntevärdet genom att uppskatta sannolikheterna med upprepade mätningar av bitarnas tillstånd [16]. I en simulerad kvantdator kan dock sannolikheterna fås direkt.

Mixerhamiltonianen \hat{H}_M spelar en viktig roll i QAOA då sannolikheterna utan denna i slutet av algoritmen hade varit samma som för initialtillståndet $|\Phi_0\rangle$. Genom en taylorutveckling kan det visas att sluttillståndet utan mixerhamiltonianen erhålls enligt:

$$|\Phi\rangle \equiv e^{-i\gamma \hat{H}_C} |\Phi_0\rangle = \sum_i^{2^n} e^{-i\gamma E_i} e^{i\theta} \sqrt{p_{0_i}} |i\rangle. \quad (2.6)$$

I ekvationen är i en av alla möjliga bitsträngar, p_{0_i} sannolikheten att mäta bitsträngen i i initialtillståndet, där i har kostnadsfunktionsvärdet E_i . $\gamma = \sum_{l=1}^p \gamma_l$ och γ_l är komponenterna i $\vec{\gamma}$ och θ en godtycklig fas. Från denna ekvation kan sannolikheten för att tillstånd i uppmäts i sluttillståndet bestämmas enligt:

$$p_i = |\langle i | \Phi \rangle|^2 = \left| \langle i | e^{i\theta - i\gamma E_i} \sqrt{p_{0_i}} | i \rangle \right|^2 = \left| e^{i\theta - i\gamma E_i} \sqrt{p_{0_i}} \right|^2 = p_{0_i}. \quad (2.7)$$

I ekvationen har det utnyttjats att $\{|i\rangle\}_{i=1}^{2^n}$ är en egenbas och således ortogonal och normalerad, vilket gör att endast en term i summan bidrar. Basens egenskaper innebär att sannolikheten utan mixerhamiltonian är beroende av $\vec{\gamma}$. Då är optimering inte möjligt. Användandet av en mixerhamiltonian (\hat{H}_M) skapar ett beroende av $\vec{\gamma}$ och $\vec{\beta}$, eftersom \hat{H}_M inte kommuterar med \hat{H}_C .

2.4 Prestandamått

Det finns ett antal olika sätt att jämföra hur den modifierade versionen av QAOA presterar jämfört med standardversionen. I denna studie används prestandamåtten approximationskvoten och framgångssannolikheten. Approximationskvoten (AR) definieras som:

$$\text{AR} = \frac{\langle \hat{H}_C \rangle}{E_{gs}}. \quad (2.8)$$

Här är $\langle \hat{H}_C \rangle$ snittet av Max-Cut-kostnaden hos den kvantmekaniska lösningen och E_{gs} är energin för det optimala tillståndet. Denna beskriver alltså hur nära den optimala lösningen en viss version av QAOA kommer för en given graf. Framgångssannolikheten (SP) definieras som:

$$\text{SP} = \sum_{i=\text{opt}} |\langle i | \psi_{\text{QAOA}} \rangle|^2. \quad (2.9)$$

i indexerar de optimala bitsträngarna och ψ_{QAOA} är tillståndet som algoritmen returnerar i slutet. Framgångssannolikheten är med andra ord sannolikheten att mäta något av de optimala tillstånden.

2.5 Gibbsfunktionen

Enligt Li et al. [1] är idén med Gibbsfunktionen att den ska ge bättre SP än standardversionen av QAOA. Det går att delvis intuitivt förutsäga att Gibbsfunktionen kommer gå från en bra AR till att ge en bra SP när η går från $0 \rightarrow \infty$. Det här kan förstås genom att skriva Gibbsfunktionen som en summa, enligt

$$f_{\text{Gibbs}}(\eta, |\psi\rangle) = -\ln \langle e^{-\eta \hat{H}_C} \rangle = -\ln \sum_{i=1}^{2^n} p_i e^{-\eta E_i}. \quad (2.10)$$

Här är n är antalet noder i grafen, E_i negativa Max-Cut-värden för bitsträng i och p_i sannolikheten att mäta E_i då kvantdatorn befinner sig i tillståndet $|\psi\rangle$. Studien genomför en minimering av Max-Cut värdet, och därför kommer de optimala bitsträngarna ha $E_{gs} = \min_{i \in [1, 2^n]} (E_i)$. Den optimala energin gör att $e^{-\eta E_{gs}}$ är ett stort positivt tal. De optimala tillståndens termer i summan är minst en faktor e^η större än de näst bästa tillståndens termer. Faktorn leder till att då $\eta \rightarrow \infty$ kommer följande ekvation asymptotiskt att gälla:

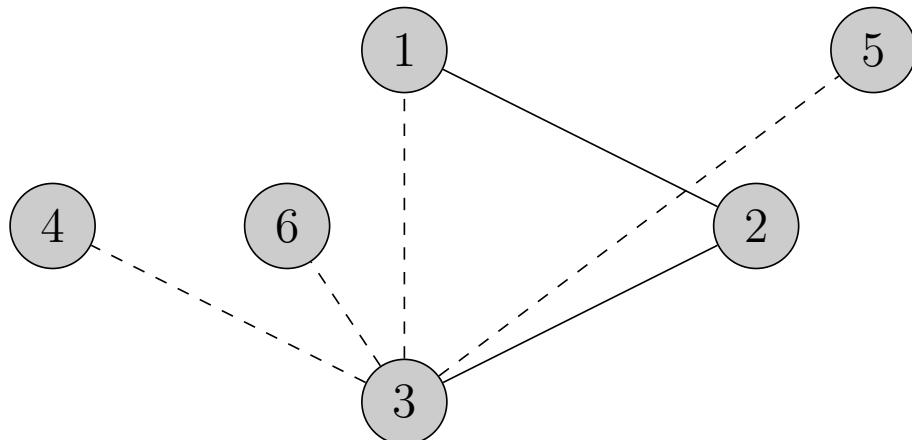
$$f_{\text{Gibbs}}(\eta |\psi\rangle) = -\ln \sum_{i=1}^{2^n} p_i e^{-\eta E_i} \rightarrow -\ln(p_{\text{SP}} e^{-\eta E_{gs}}). \quad (2.11)$$

Eftersom $e^{-\eta E_{gs}}$ är en konstant och $\ln(x)$ monoton kommer $\ln(p_{\text{SP}} e^{-\eta E_{gs}})$ maximeras och därmed f_{Gibbs} minimeras när $p_{\text{SP}} = 1$. Det blir också så att ökande SP ger ett strikt lägre f_{Gibbs} . Det går också att härleda genom en typ av taylorutveckling att $f_{\text{Gibbs}} \rightarrow f_{\text{standard}} = \eta \langle \hat{H}_C \rangle$ (vilket är proportionellt med AR) när $\eta \rightarrow 0$. Dessa resonemang visar på att då η går från 0 till ∞ kommer Gibbsfunktionen att gå från att maximera AR till att enbart försöka maximera SP.

Det här är enligt Li et al. [1] idén med Gibbs, att användandet av en Gibbsfunktion gör att optimeraren får skäl att öka SP på ett annat sätt än om den vanliga kostnadsfunktionen $\langle \hat{H}_C \rangle$, används. En annan fördel som nämns i artikeln är att Gibbsfunktionen bryr sig mindre om värden på p_i för i som inte är optimala, vilket kan förväntas ge ett energilandskap med färre lokala extempunkter.

2.6 Grafteori

Som tidigare nämnts utförs Max-Cut problemet endast på 4-reguljära grafer. En viktig egenskap hos 4-reguljära grafer som är relevant för denna studie är att då nodantalet ökar kommer omgivningen för nästan alla noder se ut som ett träd [17]. För stora grafer är det således osannolikt att det förekommer cykler i grafen med en liten längd l . Ett exempel för 2-reguljära grafer på hur det här kan se ut visas i figur 2.3.



Figur 2.3: Exempel på hur sannolikheten för små cykler är liten i större grafer. Sannolikheten att nod 3 kopplar till just nod 1 och skapar en loop är $\frac{1}{4}$ gentemot sannolikheten $\frac{3}{4}$ att koppla till en av noderna 4, 5 eller 6 (om grafen är 2-reguljär). Om man hade lagt till fler noder hade denna sannolikhet varit ännu lägre.

Eftersom djupet p i kretsen som simuleras motsvarar hur långt bort i grafen som algoritmen ”ser” [18], kommer algoritmen uppfatta allt förutom cykler av storlek $l < p$ som träd. Sannolikheten för existensen av en cykel med $l < p$ kommer som tidigare nämnts minska med antalet noder i grafen, något som gör att algoritmen nästan enbart kommer att se träd för stora grafer [18].

Då graferna lokalt liknar ett visst träd när nodantalet ökar bör lösningarna för olika isomorfismer av en fix storlek bli mer lika varandra. Algoritmen kommer till större och större del ”se” perfekta ”4-vägs” träd, vilka är unika, med p -nivåer. Det totala resultatet för algoritmen erhålls som summan av resultaten över alla delgrafer [18]. Då delgraferna är identiska kommer det summerade totala resultatet bli mer och mer likt för olika isomorfismer.

3 Metod

Här beskrivs de val som gjorts av η , slutparametrar och konvergenskriterier. Alla resultat erhölls genom simulerings av algoritmerna i Python, med kodpaketet PennyLane [2]. I den kodade versionen av QAOA användes den klassiska optimeraren Gradient descent [19]. Koden som användes finns som ett GitHub-projekt tillsammans med en beskrivning. Koden är bland annat baserad på ett GitHub-projekt utvecklat på MC2 [20].

3.1 Valet av parametern η

Hyperparametern η i Gibbsfunktionen är en variabel vars värde väljs beroende på problemet som ska lösas. Det här är alltså inte en parameter som ska optimeras av QAOA, utan denna hålls konstant under optimeringen. Genom att explicit uttrycka och minimera den (ovanifrån) begränsade sannolikheten för låga energier för stora η , menar Li et al. att ett optimalt värde på denna ges av:

$$\eta_{opt} = \frac{1}{E_0 - E_{gs}}. \quad (3.1)$$

I ekvationen är E_0 en godtycklig energi som i det aktuella problemets kontext anses vara tillräckligt liten och E_{gs} är grundtillståndets energi [1]. Här motsvarar alltså E_{gs} värdet på den bästa lösningen på Max-Cut för en given graf.

Nämnden i kvoten i ekvation 3.1 kan uttryckas som $C \cdot E_{gs}$, med $C = 1 - \delta$, där $\delta \in [0, 1]$ kommer från att $E_0 = \delta \cdot E_{gs}$. Värdet på δ beskriver alltså inom vilken procentsats av E_{gs} som energin E_0 anses vara låg. Då η :s påverkan på QAOA studeras är det $1/C$ och således värdet på δ som varieras. För att förenkla defintionen av η görs omskrivningen $k = 1/C$. Det ger

$$\eta = \frac{k}{E_{gs}}. \quad (3.2)$$

I enlighet med Li et al. [1] har standard för η valts som $k = 20$ ($\delta = 0,95$). E_0 har alltså ansetts vara låg då denna ligger inom 95 % av grundtillståndets energi. För problemen som lösas i denna studie motsvarar det här värden på η i intervallet 0,8 – 4.

Ett antal simuleringar gjordes för att undersöka vilket η som fungerar bäst för problemen i denna studie. Det här ansågs viktigt då Li et al. [1] menar att valet av η är högst beroende av problemet som ska lösas. Eftersom denna studie undersöker Max-Cut för en annan klass av grafer än Li et al., går det inte med säkerhet att säga om deras val av η också fungerar i detta fall. Om resultaten från denna analys pekar på att andra värden på η hade varit optimala, kan eventuella svaga prestationer för Gibbsfunktionen delvis förklaras.

3.2 Valet av slutparametrar

För både standard- och Gibbsversionen undersöks två sätt att välja de sluttgiltiga optimeringsparametrarna, $(\vec{\gamma}, \vec{\beta})$, i QAOA efter att algoritmen konvergerat. Det första sättet **(1)** är att slutparametrarna väljs till de optimeringsparametrar som under optimeringsprocessen gav den största approximationskvoten. Det andra sättet **(2)** är att välja parametrarna som optimeringen konvergerar till som slutparametrar. Skälet till att dessa val av slutparametrar kan förväntas påverka prestandan, är att Gibbsversionen kan förväntas ge olika resultat för SP och AR, se avsnitt 2.5. Det gör att om Gibbsversionen slumpar sig till att passera en bättre punkt för AR under optimeringsprocessen kommer den inte nödvändigtvis stanna där utan fortsätta till sitt eget minima istället. Därför kommer punkten med bäst AR inte nödvändigtvis vara samma som det Gibbsfunktionen konvergerar till, vilket gör att de olika valen av slutparametrar ger olika resultat.

3.3 Konvergenskriterier

För att minimera onödig tidsåtgång införs vissa konvergenskriterier. Tanken med dessa är att stoppa algoritmen då den konvergerat, och på så sätt undvika beräkningar som inte resulterar i en förbättring av lösningen. I detta arbete har två konvergenskriterier använts. Det första är att förändringen av det genomsnittliga Max-Cut värdet är inom ett förbestämt interval i tio efterföljande steg. Det andra kriteriet består av två subkriterier, varav ett måste vara uppfyllt. Algoritmen behöver ha tagit minst hälften av det maximala antalet tillåtna steg eller så behöver approximationskvoten vara över ett visst värde. Värdena som använts för konvergenskriterierna visas i tabell 3.1.

Tabell 3.1: Värden som används för de två konvergenskriterierna och tillhörande subkriterier.

Kriterium	Värde
Konvergensintervall	$\pm 1\%$
Maximalt antal steg	300
Approximationskvot	90%

För att en viss körning ska räknas som konvergerad måste båda kriterierna vara uppfyllda. Tanken med dubbla kriterier är att algoritmen ska ha hamnat i ett minima samt att minimat ska vara så djupt så att optimeraren inte kan ta sig ur och finna en bättre lösning.

4 Resultat

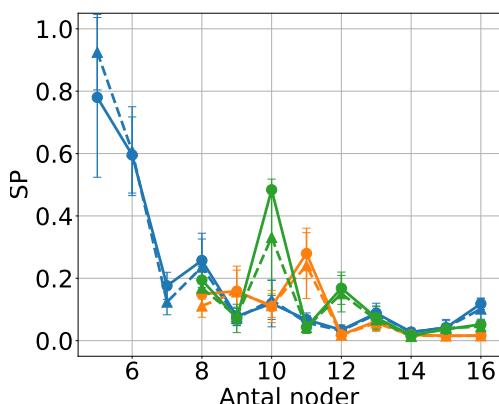
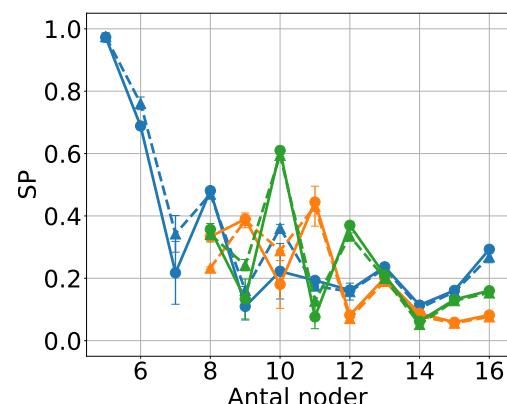
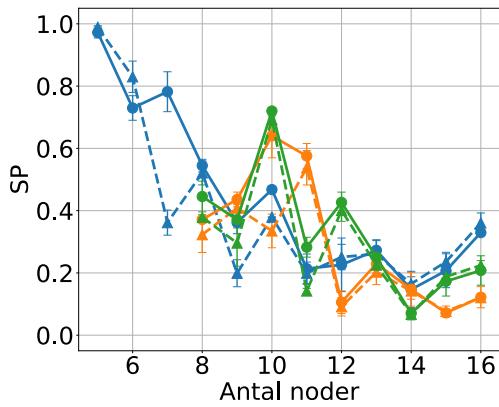
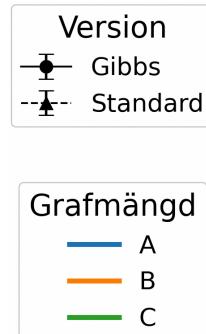
Här presenteras resultatet för framgångssannolikhet och approximationskvot med de två olika sätten att välja slutparametrarna. Se avsnitt 3.2 för detaljer om slutparametrarna. Vidare visas också resultat för energilandskap och val av parametern η .

Simuleringarna av QAOA med både standardversionen och Gibbsversionen för Max-Cut utfördes på tre olika mängder med 4-reguljära grafer i storlekarna fem till 16 noder. För beskrivning av standard- och Gibbsversionen se avsnitt 2.1.1 respektive 2.5. Grafmängderna benämns A, B och C. Graferna av samma storlek mellan de olika mängderna är icke-isomorfa. För grafer med nodantal mindre än åtta finns inte tre icke-isomorfa 4-regulära grafer med samma nodantal, därför har endast grafmängd A simulerats för dessa fall. Parametern η i Gibbsfunktionen valdes i enlighet med Li et al. [1] som $\eta = 1/(0,05 \cdot E_{gs})$, där E_{gs} är grundtillståndets energi. Se avsnitt 3.1 för motivering av valet.

I simuleringarna har det körts 20 initieringar per testfall, alltså 20 olika initialparametrar, med undantag för då olika η studerades. Dessa har varit samma för alla grafstorlekar. Initialparametrarna har i huvudsak erhållits genom latinsk hyperkubisk sampling, en metod för att garantera att de slumpmässiga initialparametrarna spridits ut. Initialparametrarna har använts som gissning för det första lagret; de senare bestäms enligt initialiseringstrategin beskriven i ekvation 2.5.

4.1 Framgångssannolikhet

Först behandlas framgångssannolikheten när de slutgiltiga optimeringsparametrarna valts utifrån vad som ger bäst AR, se (1) i avsnitt 3.2. Figur 4.1 visar att SP för Gibbsrelativt standardversionen är mycket varierat för alla djup. Det går inte att säga att Gibbsfunktionen alltid skulle vara bättre än standardmålfunktionen. Vissa tendenser kan dock anas. För de större graferna är skillnaden mellan Gibbs- och standardversionen genomgående mycket liten för alla grafmängder. SP tycks börja öka med grafstorleken för de största graferna, se grafer med storlek 14 till 16 noder. För de mindre graferna så varierar resultatet mycket. Notera även hur sannolikheten avtar mycket fort när graferna blir större än fem noder.

(a) Djup $p = 1$.(b) Djup $p = 2$.(c) Djup $p = 3$.

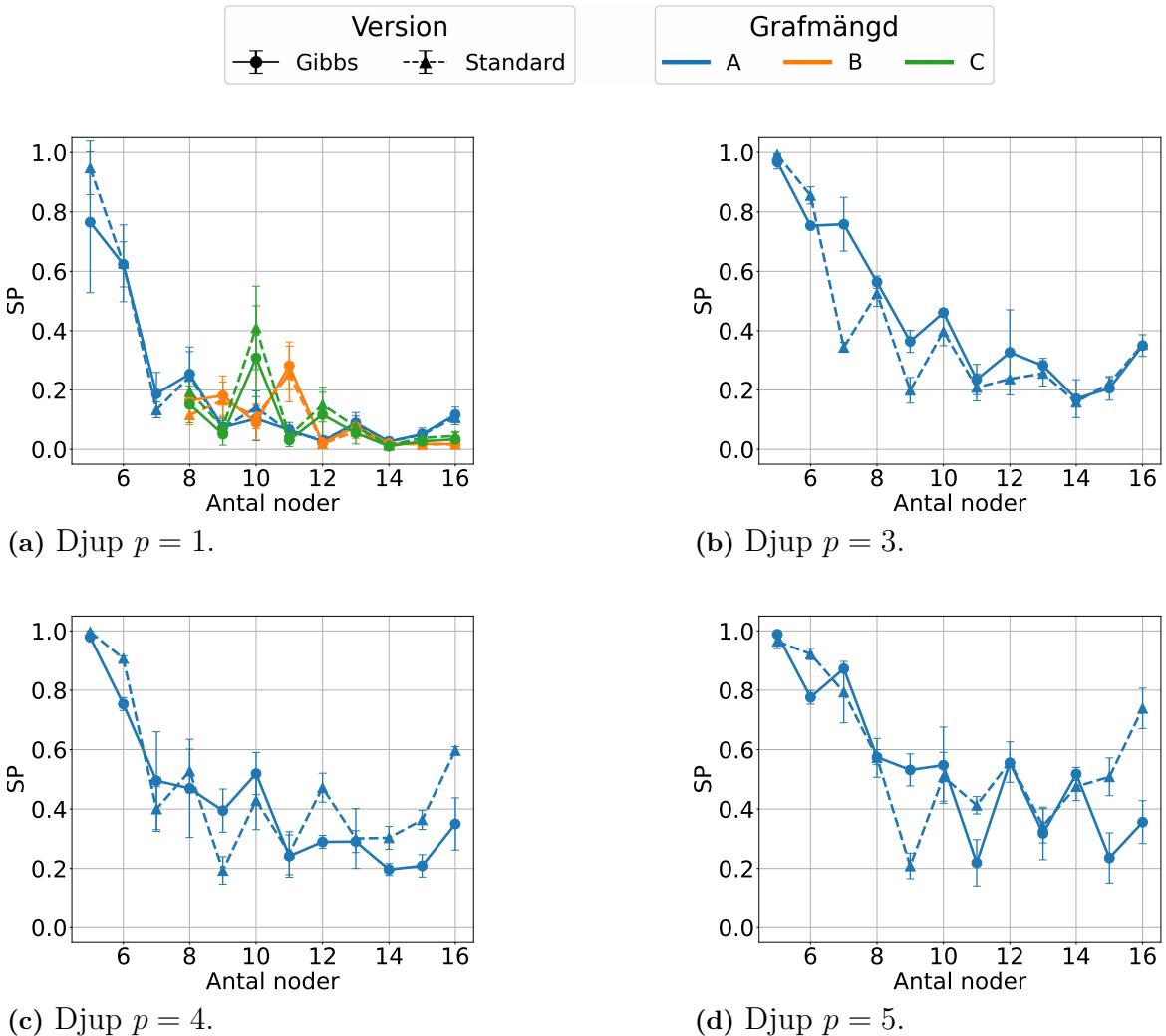
Figur 4.1: SP för graferna i mängderna A, B och C för Gibbs- respektive standardversionen av QAOA. En standardavvikelse är markerad vid varje punkt och markören visar medelvärdet. De olika figurerna är gjorda med olika djup (a) $p = 1$, (b) $p = 2$ och (c) $p = 3$. För dessa är slutparametrarna valda utifår vilka som gav högst AR, se (1) i avsnitt 3.2.

Figur 4.1 visar även på variationer mellan de olika icke-isomorfa grafmängderna, men variationerna tycks minska när djupet ökar. Resultaten i figur 4.1 visar att de olika isomorfismerna kan ha en stor påverkan för hur bra resultat som kan förväntas från

4 Resultat

algoritmen. Skillnaden är extra tydlig för nodantal tio och elva i figur 4.1, medan denna för vissa andra par av icke-isomorfa grafer istället är mycket liten. Exempel på mindre skillnader syns exempelvis i graferna med nodantal 13 till 15.

Ett ökat djup ger, från resultatet i figur 4.1 att döma, en ökad SP för båda versionerna av QAOA och alla grafmängder. För vissa grafer ger ett ökat djup en stor förbättring, exempelvis nodstorlek sju respektive tio i mängd A respektive C när djupet ökas från två till tre. Det finns en liten skillnad i hur stor förändringen blir mellan de olika grafmängderna, men här finns det inget tydligt mönster. För vissa grafstorlekar, se till exempel 16 noder i 4.1 (a), ger Gibbsfunktionen ett bättre utfall för alla tre mängder. Samma grafer med djup $p = 3$, som kan ses i figur 4.1 (c), ger dock motsatt resultat. Relationen mellan de olika grafmängdernas SP varierar då djupet ökar. Vilken av grafmängderna som får högst respektive lägst SP skiftar sällan när djupet varierar.

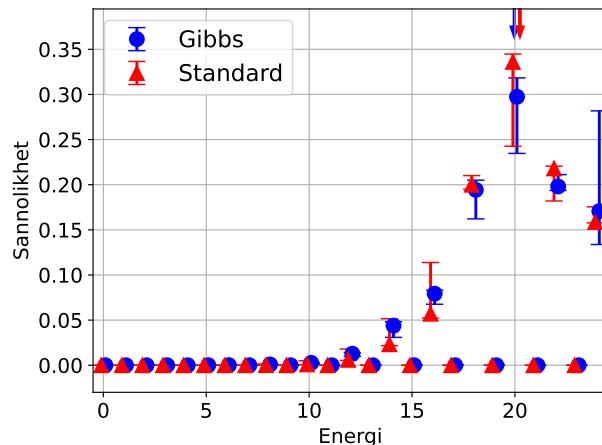


Figur 4.2: SP för graferna i mängderna A, B och C för Gibbs- respektive standardversionen av QAOA. En standardavvikelse är markerad vid varje punkt, där markören visar medelvärdet. De olika figurerna är gjorda med olika djup (a) $p = 1$, (b) $p = 3$, (c) $p = 4$ och (d) $p = 5$. För dessa är slutparametrarna valda utifrån vilka som optimeringen konvergerat till, se (2) i avsnitt 3.2.

Om de parametrar som optimeringen konvergerar till används istället, se (2) i avsnitt 3.2, erhålls liknande men svagt förbättrade resultat som vid det tidigare valet. En förbättring syns för grafen med storlek tolv i grafmängd A i figur 4.1 (c) och 4.2 (b). En av likheterna är den mycket snabba nedgången i SP när grafernäs nodantal ökar, för alla djup, se figur 4.2. Dessutom visar denna figur, precis som figur 4.1, att SP ökar för de största graferna. Intressant är också att standardmålfunktionen tycks prestera betydligt bättre än Gibbefunktionen för större grafstorlekar (15–16 noder) vid de största djpen, se figur 4.2 (c) och (d).

Det har inte gjorts lika många simuleringar med olika grafmängder för det andra sättet att välja slutparametrar, förutom för djup $p = 1$ i figur 4.2 (a). För de grafer som ligger i intervallet 8–12 noder skiljer sig resultaten relativt mycket mellan de olika mängderna, cirka 20%. För de större graferna, 12–16 noder fås ett liknande resultat som i figur 4.1. Det vill säga att de olika icke-isomorfa graferna inte skiljer sig så mycket från varandra, cirka 10%.

Ett resultat som talar till Gibbefunktionens fördel är att denna ger bättre SP än standardmålfunktionen för alla grafer med nodstorlek i intervallet sju till 13 noder i figur 4.2 (b). Det finns intervall i de andra figurerna där Gibbs- och standardversionen skiljer sig åt väsentligt, men dessa innehåller sällan fler än två grafstorlekar.



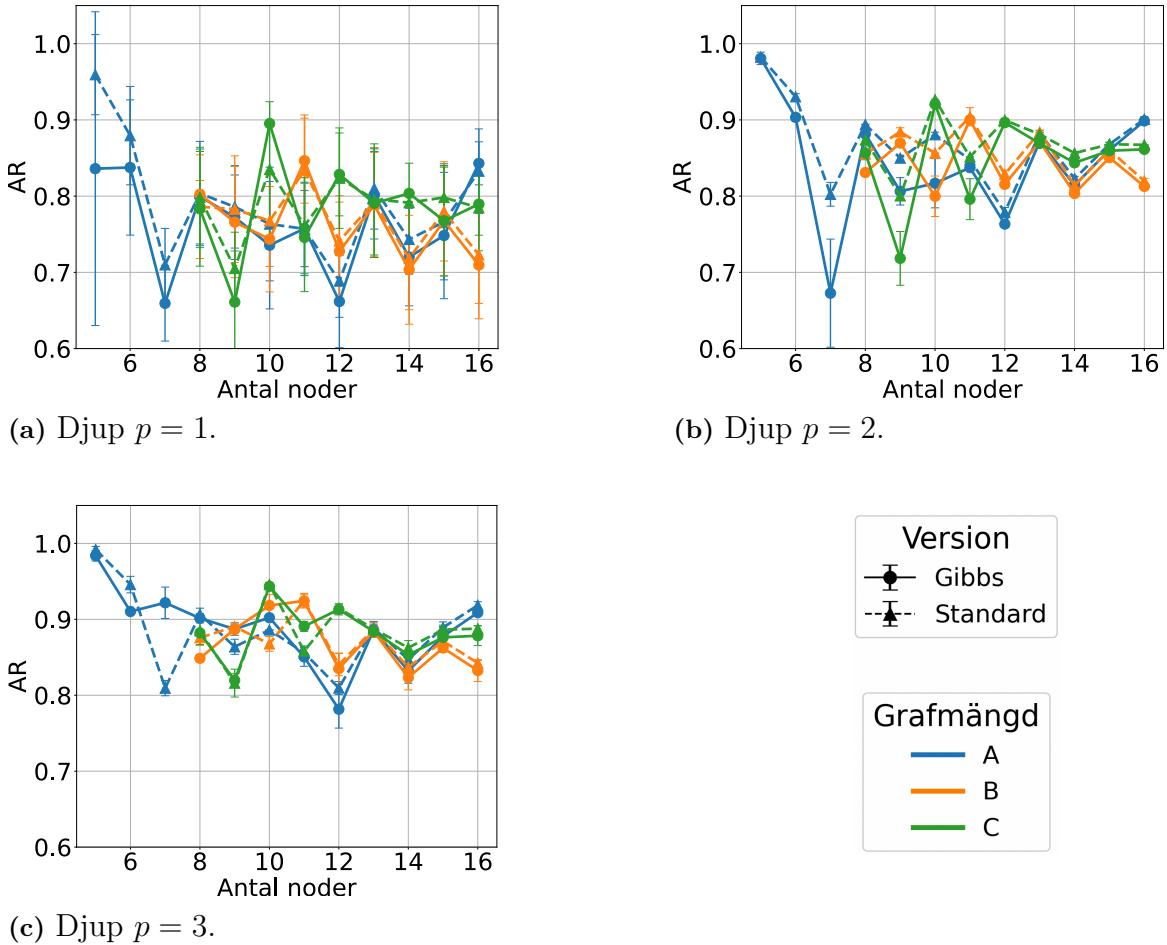
Figur 4.3: Sannolikhetsdistributionen för sluttillståndets energi för grafen med 12 noder ur grafmängd A med djup $p = 3$. Markörerna i figuren visar medelvärdet medan felstapplarna visar det högsta respektive lägsta värdet. Pilarna längst upp i figuren markerar den genomsnittliga Max-Cut kostnaden för Gibbefunktionen respektive standardmålfunktionen. Max-Cut kostnaden är proportionell mot AR. För dessa är slutparametrarna valda utifrån vilka som optimeringen konvergerat till, se (2) i avsnitt 3.2.

För att förstå varför Gibbefunktionen ibland ger bättre utfall undersöktes sannolikhetsfördelningen för det sluttillståndet i figur 4.2 (b). Resultatet visas i figur 4.3. Grafen med tolv noder i mängd A för djup tre valdes då förhållandet mellan Gibbs- och standardversionen ansågs vara representativt för figuren. Ett tydligt mönster i figuren är att sannolikheten att få någon av de udda energinnivåerna alltid är noll för både Gibbs- och standardversionen. Själva distributionerna har endast mindre skillnader, exempelvis att Gibbefunktionen ger högre varians. Punkterna som har störst spridning

är för Gibbsversionen den högsta energin och för standardversionen den som är närmast snittet av energin. Från figur 4.2 (b) och figur 4.3 fås att SP är större för Gibbsfunktionen, men från figur 4.3 fås det motsatta för AR. Alltså har Gibbsfunktions-optimeringen prioriterat en högre SP över en hög AR.

4.2 Approximationskvot

I figur 4.4 visas resultaten då slutparametrarna valdes efter vad som gav bäst AR, se (1) i avsnitt 3.2. Från figuren kan det, precis som för SP, ses att ökande nodantal minskar spridningen mellan de olika grafmängderna, dock följer inte nodantal 16 detta mönster. Likt som för SP minskar även spridningen mellan Gibbs- och standardversionen då nodantalet ökar, vilket syns särskilt vid nodantal 16 för alla djup. Då nodantalet går från fem till sju minskar AR markant. Det enda undantaget är Gibbsversionen för djup tre vid nodantal sju.

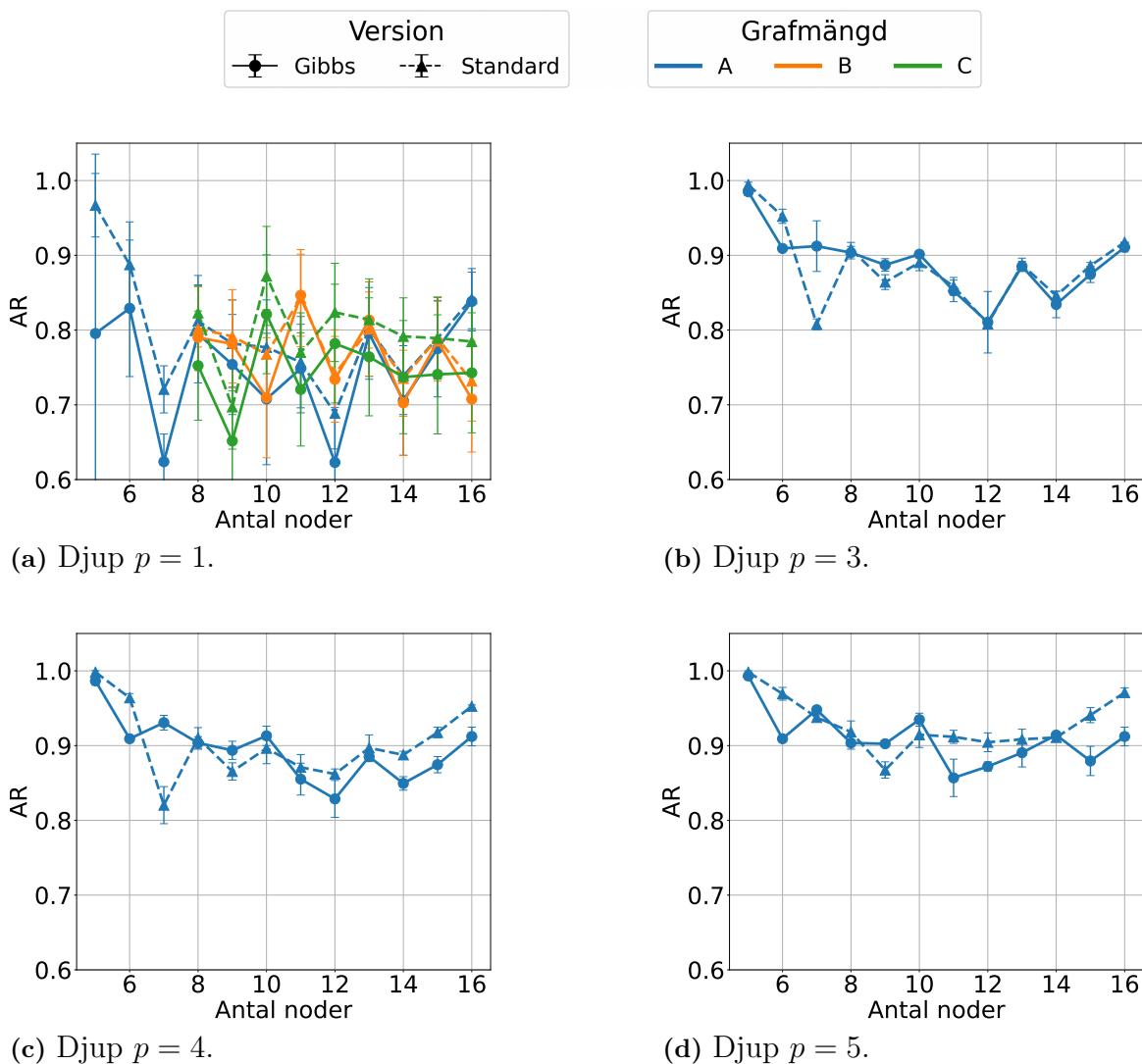


Figur 4.4: AR för graferna i mängderna A, B och C för Gibbs- respektive standardversionen av QAOA. En standardavvikelse är markerad vid varje punkt, där markören visar medelvärdet. De olika figurerna är gjorda med olika djup (a) $p = 1$, (b) $p = 2$ och (c) $p = 3$. För dessa är slutparametrarna valda utifrån vilka som gav högst AR, se (1) i avsnitt 3.2

Från figur 4.4 kan det utläsas att grafmängden, det vill säga isomorfismen, har en stor påverkan på resultatet. Detta syns främst för djup ett, men även för nodantal tolv med djup tre.

Då djupet ökar minskar spridningen generellt för alla grafer, vilket framgår tydligast då djup ett och tre jämförs, se (a) och (c) i figur 4.4. Standardavvikelsen tycks också minska då djupet går över $p = 1$. Det går även att utläsa att AR generellt sett blir bättre för alla nodantal med ökat djup.

Figur 4.4 visar även att standardversionen av QAOA generellt sett ger något bättre AR än Gibbsfunktionens motsvarighet, även om skillnaden vanligen ligger inom en standardavvikelse. Det finns dock några fall där Gibbsfunktionen ger bättre AR, exempelvis nodantal tio för grafmängd B i (c).



Figur 4.5: AR för graferna i mängderna A, B och C för Gibbs- respektive standardversiōnen av QAOA. En standardavvikelse är markerad vid varje punkt, där markören (notera skillnaden mellan Gibbsversionen och standard) visar medelvärdet. Figurerna är gjorda med olika djup (a) $p = 1$, (b) $p = 3$, (c) $p = 4$ och (d) $p = 5$. För dessa är slutparametrarna valda utifår vilka som optimeringen konvergerar till, se (2) i avsnitt 3.2

Figur 4.5 visar resultaten då slutparametrarna valdes på det andra sättet, se (2) i avsnitt 3.2. Likt resultaten i figur 4.4 minskar spridningen mellan Gibbs- och standardversionen för högre nodantal med låga djup. För djup fyra och fem börjar skillnaden istället att växa, precis som för SP i figur 4.2.

De olika grafmängderna ger i figur 4.5 (a), likt figur 4.4, spridda resultat. Spridningen är störst för nodantal nio till tolv, medan skillnaden vid högre nodantal minskar något. I figuren kan det också utläsas att djup ett ger mycket spridda resultat för olika isomorfismer, medan de för högre djup är mer samlade.

Det finns en trend att större djup för högre nodantal minskar skillnaden mellan Gibbs- och standardversionen från figur 4.4, men trenden fortsätter inte för djup fyra och fem i figur 4.5. Här ökar standardversionen istället mycket mer än Gibbsversionen då djupet ökar. Notera den stora skillnaden i standardavvikelse mellan djup ett och övriga. En annan notering är att det finns punkter, så som nodantal sju i mängd A mellan djup ett och två, där Gibbsversionen ökar mer än standardversionen.

I resultaten från figur 4.5 ger standardversionen av QAOA generellt sett bättre AR, precis som i figur 4.4. Med sättet att välja slutparametrar som används i figur 4.5 blev dock skillnaden något större med fördel för standardversionen.

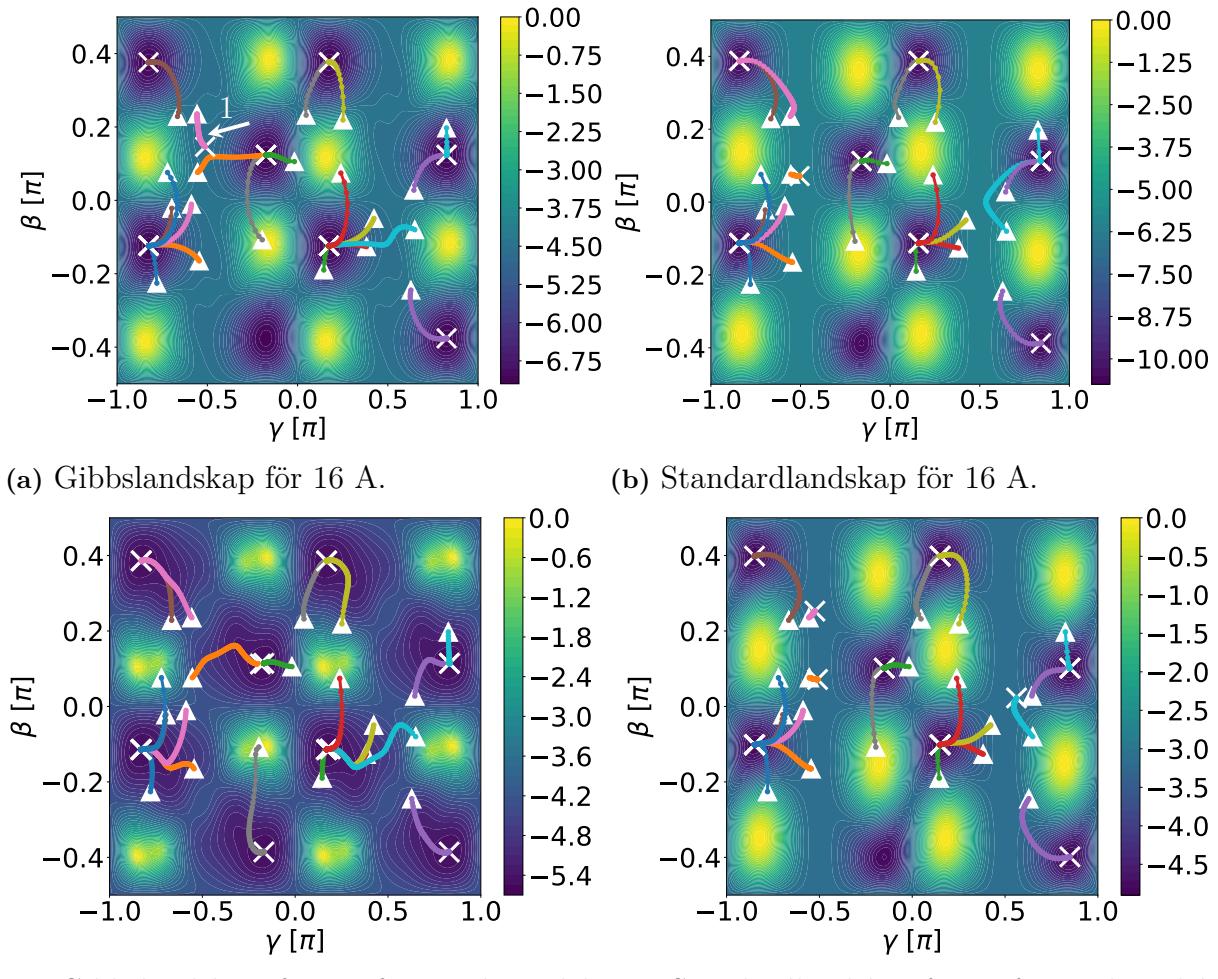
4.3 Jämförelse av kostnadsfunktionernas energilandskap

Här visas de olika “energilandskapen” för kostnadsfunktionerna som funktioner av (γ_1, β_1) . Landskapen framställdes genom att beräkna kostnadsfunktionerna för 10 000 par av (γ_1, β_1) . Paren är de som kan bildas av 100 ekvidistanta punkter på intervallet $[-\pi, \pi]$ för γ_1 och 100 ekvidistanta punkter på intervallet $[-\frac{\pi}{2}, \frac{\pi}{2}]$ för β_1 . Samtliga landskap är skalade så att minima är det optimala värdet och att maximat för respektive landskap är noll i syfte att åskådliggöra höjdskillnader.

Resultaten är skapade för $p = 1$ för grafen med storlek 16 i mängd A och en, tidigare inte använd, graf med storlek sju. I figurerna visas också optimeringsiterationer med Gibbs- respektive standardmålfunktionen som punkter med olika färger. För båda graferna har $\eta = 20/E_{gs}$ använts, där E_{gs} är grundtillståndsenergin för den aktuella grafen.

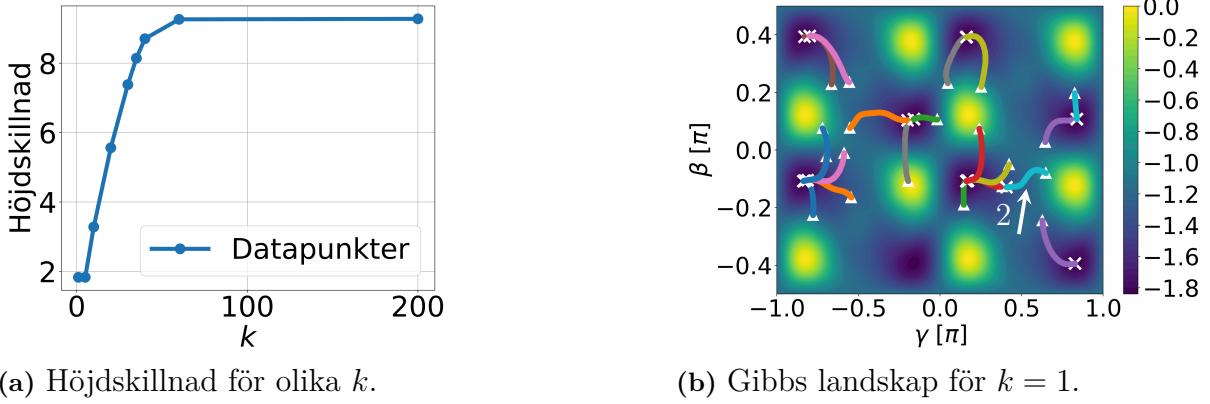
Figur 4.6 visar en utzoomad jämförelse av standard- och Gibbslandskapet. Med förbehåll för att denna typ av visualiseringar bara kan göras för $p = 1$ visar figur 4.6 på ett antal fenomen. Två exempel är att både Gibbsfunktionen och standardmålfunktionen i vissa fall inte tycks konvergera till ett lokalt minima, utan fastnar på en platå och att alla lokala miniman verkar vara ungefär lika djupa. Det verkar alltså inte finnas lokala miniman som inte är globala miniman för $p = 1$. Notera också landskapets tydliga periodicitet, en periodlängd verkar vara cirka π för γ och cirka $\frac{\pi}{2}$ för β .

Figur 4.7 visar vad som händer med landskapet om η varieras. Figuren indikerar att det möjliga finns en koppling mellan η , höjdskillnader i energilandskapet och hur snabbt simuleringen konvergerar. Ett exempel på att mindre höjdskillnad leder till längsammare konvergens är att iterationen indikerad med en tvåa i figur 4.7 (b) inte hinner konvergera. Om figur 4.7 (b) förstoras kan man också ana att steglängden är lägre än i figur 4.6 (c) som är motsvarande landskap för ett högre η . Figur 4.7 (a) tyder på att högre η ger större höjdskillnad och därfor enligt (b) snabbare konvergens. Därför tycks det, med förbehåll för att detta enbart bygger på analys av en specifik graf, finnas en viss tendens att högre η ger större höjdskillnad och därmed snabbare konvergens.

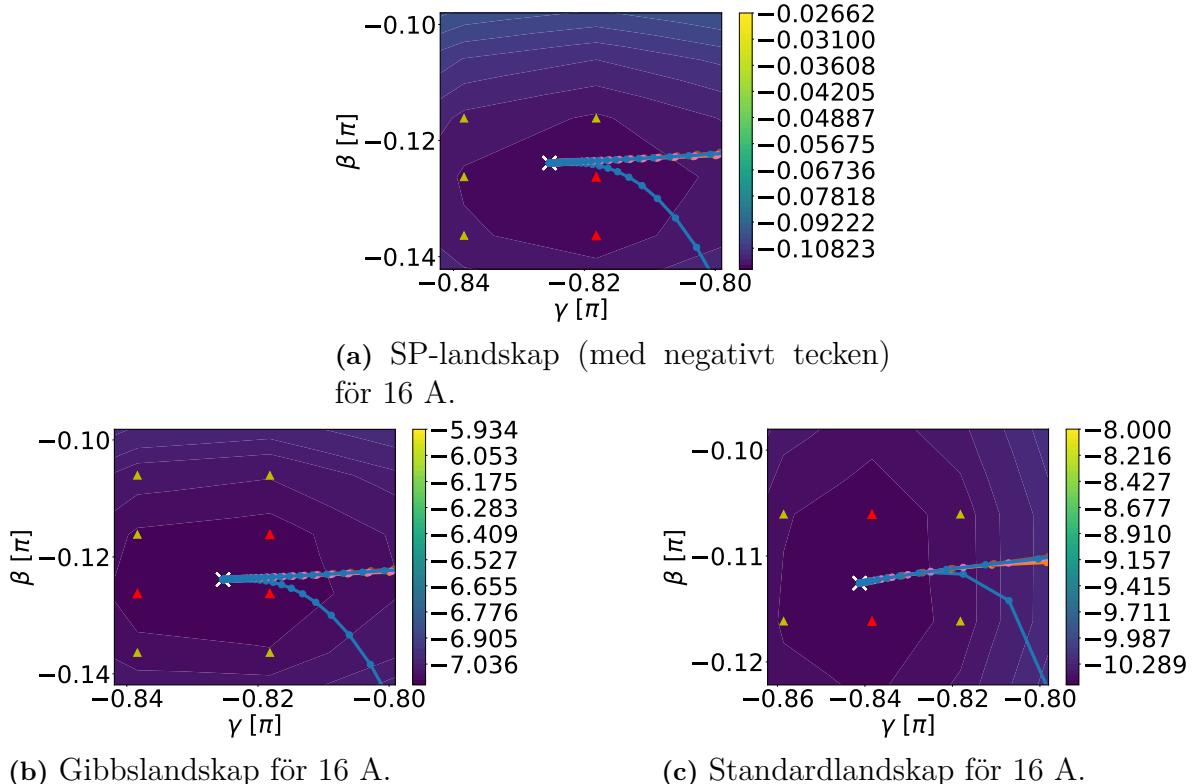


Figur 4.6: (a) och (b) visar landskapet för Gibbsfunktionen respektive standardmålfunktionen för grafen med storlek 16 i grafmängd A. (c) och (d) visar landskapet för Gibbsfunktionen respektive standardmålfunktionen för grafen med storlek sju som inte ingår i grafmängd A, B eller C. Notera i (a) att iterationen indikerad med ett i övre vänstra hörnet inte konvergerar till ett lokalt minima trots att motsvarande iteration för standardmålfunktionen i (b) konvergerat. I båda standardlandskapen (c) och (d) finns flera iterationer som inte konvergerar till lokala miniman. Trianglarna visar var iterationerna börjar och kryss där de slutar.

För att analysera i detalj hur Gibbsversionen respektive standardversionen konvergerar kan området nära ett godtyckligt valt lokalt minima studeras. I figur 4.8 visas en inzoomning vid ett sådant minima för grafen i mängd A med storlek 16. Notera att det lokala minimat är förskjutet mellan Gibbs- och standardversionen. Gibbsversionen tycks alltså inte bara ge en omskalning av energilandskapet utan även en deformation. Jämförs SP-landskapet (med negativt tecken) (a) i figur 4.8 med Gibbslandskapet (b) samt standardlandskapet (c), så framgår det att miniman för SP- och Gibbslandskapen ligger närmare varandra än miniman i SP- och standardlandskapen. Detta innebär att tillståndet Gibbsversionen konvergerar till har ett högre SP än standardversionens slut tillstånd.



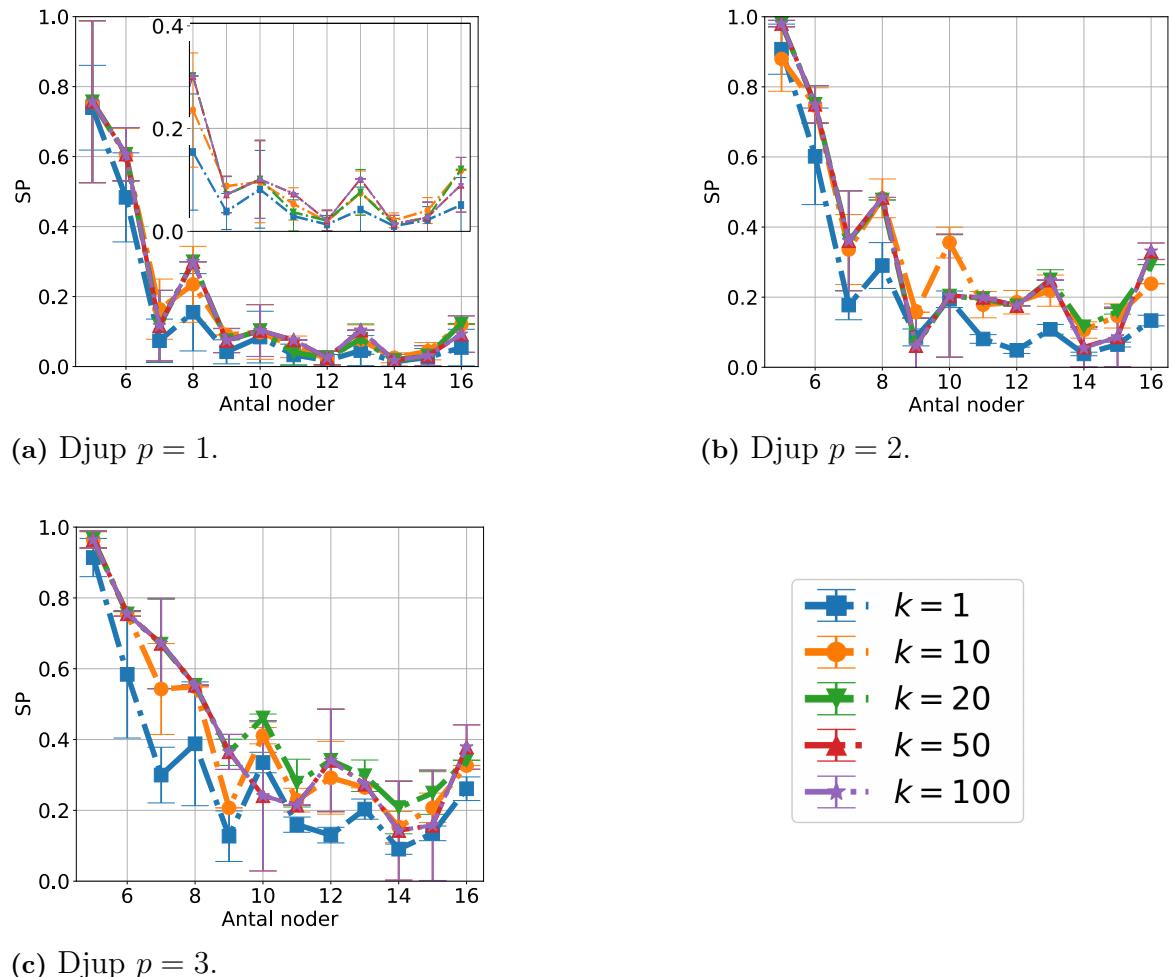
Figur 4.7: Figuren visar höjdskillnad respektive landskapet för grafen med storlek sju som inte ingår i grafmängd A, B eller C. (a) visar höjdskillnaden för Gibbsfunktionens landskap, alltså största minus minsta värdet. x -axeln visar k där $\eta = \frac{k}{E_{gs}}$ där $E_{gs}=10$. (b) visar Gibbsenergilandskapet med $k = 1$. Värdena i landskapet är omskalade så att det maximala värdet i landskapet förskjutits till noll. Notera att iterationen indikerad med en tvåa inte hinner konvergera till ett lokalt minima.



Figur 4.8: Figuren visar, för grafen med storlek 16 i grafmängd A och $p = 1$, hur både SP (med negativt tecken) (a) och Gibbsfunktionen (b) har lokala miniman omkring ($\gamma/\pi \approx -0,825$, $\beta/\pi \approx -0,125$) medan standardkostnadsfunktionen (c) har minima omkring ($\gamma/\pi \approx -0,84$, $\beta/\pi \approx -0,11$). Det tycks alltså finnas en mätbar förskjutning mellan de lokala miniman för SP (med negativt tecken) och Gibbs- relativt standardkostnadsfunktionens minima. Gula och röda trianglar markerar datapunkter som är bland de 0,2 % respektive 0,9 % lägsta värdena för respektive landskap.

4.4 Valet av parametern η

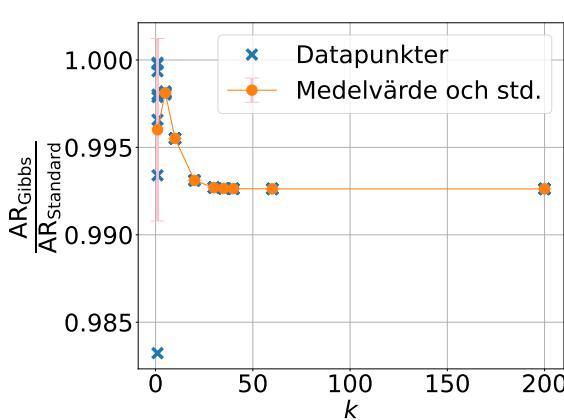
Figur 4.9 visar SP för grafmängd A med olika värden på konstanten $k = 1/C$ och djup. Som standardversion i de andra mätningarna användes $k = 20$ i enlighet med Li et al. [1]. Notera hur ett lågt värde på k ger sämst SP, medan de tre största i huvudsak följer varandra, med undantag för vissa punkter.



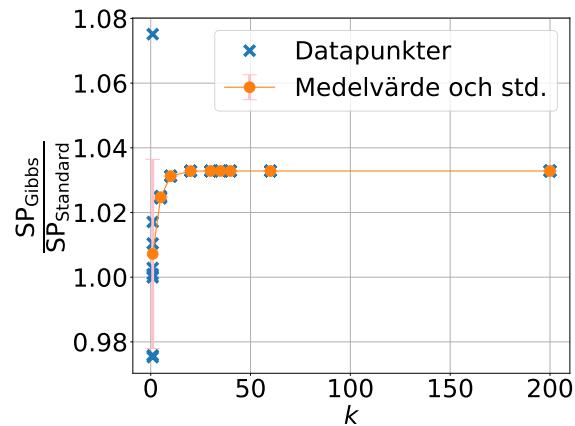
Figur 4.9: Figuren visar SP för olika värden på k , där $\eta = k/E_{gs}$, med olika djup (a) $p = 1$, (b) $p = 2$ och (c) $p = 3$ för grafmängd A. I (a) visas en inzoomning för nodstorlekar åtta till 16. Slutparametrarna valdes här enligt vad optimeringen konvergerade till och för varje graf gjordes fem initialiseringar med olika par av parametrar utspridda i energilandskapet. Varje markör visar ett medelvärde och felstaplarna motsvarar en standardavvikelse.

Tendensen att högre η ger bättre SP visas i mer detalj i figur 4.10 och 4.11. I figurerna har enbart de iterationer av Gibbs- och standardversionen med samma initialparametrar och minst avstånd mellan de slutgiltiga parametrarna använts. Denna filtrering görs för att försöka eliminera resultat där endast en av Gibbs- och standardversionen konvergerat och därmed få en mer rättvis bild för hur η påverkar. SP- och AR-kvoterna mellan Gibbs- och standardversionen analyseras för den tidigare berörda grafen med storlek sju och $p = 1$ samt grafen med storlek tolv i grafmängd A med $p = 3$.

Figur 4.10 visar SP- och AR-kvoterna mellan Gibbs- och standardversionen som funktion av η för grafen med storlek sju som inte ingår i grafmängd A, B eller C. Graferna visar med undantag för det lägsta k -värdet en relativt tydlig trend där SP-kvoten ökar med η medan AR-kvoten minskar.



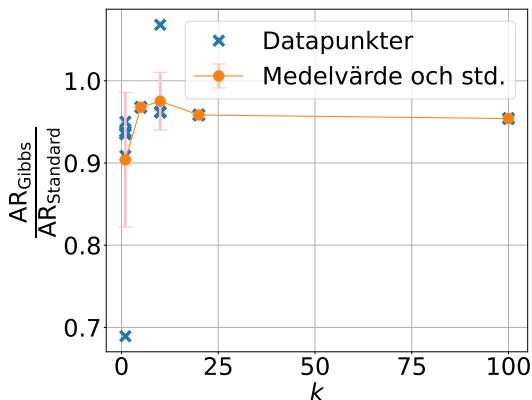
(a) AR kvot för grafen med storlek sju.



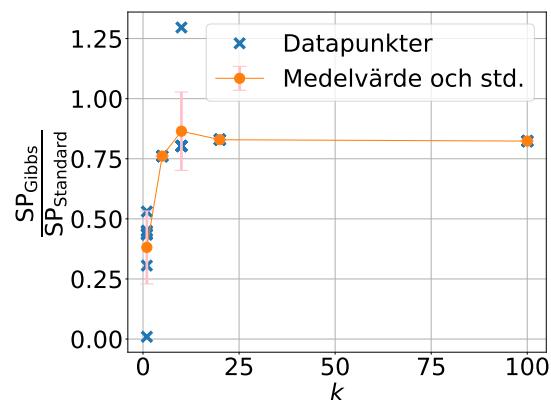
(b) SP kvot för grafen med storlek sju.

Figur 4.10: AR kvot och SP kvot för iterationer med samma initialparametrar för Gibbs- och standardversionen . Dataan som visas är för grafen med storlek sju som inte ingår i A, B eller C. Dataan har filtrerats så att enbart de åtta punkter där Gibbs- och standardversionen haft minst avstånd mellan sig används. Som x-axel används k där $\eta = \frac{k}{E_{gs}}$. E_{gs} är grundtillståndsenergin för grafen och k är en vald konstant. För denna graf är $E_{gs} = 10$.

Figur 4.11 motsvarar figur 4.10 men för grafen med storlek tolv i grafmängd A och $p = 3$. Här tycks mönstret från figur 4.10 inte upprepas, utan högre η ger både bättre SP och AR men mindre avstånd.



(a) AR kvot för grafen med nodantal tolv i grafmängd A.



(b) SP kvot för grafen med nodantal tolv i grafmängd A.

Figur 4.11: AR kvot och SP kvot för iterationer med samma initialparametrar för Gibbs- och standardversionen. Dataan som visas är för grafen med nodantal tolv i A. Dataan har filtrerats så att enbart de åtta punkter där Gibbs- och standardversionen haft minst avstånd mellan sig används. Som x-axel används k där $\eta = \frac{k}{E_{gs}}$. För denna graf är $E_{gs} = 22$.

Från resultatet i figur 4.10 och 4.11 är det oklart om högre η ger bättre eller sämre AR relativt standardversionen. Notera dock att detta enbart är några få exempel. Kopplingen mellan ökande η och högre SP är relativt tydlig baserat på samtliga figurer i avsnittet.

5 Diskussion

Jämförelsen mellan Gibbsfunktionen och standardmålfunktionen kan göras utifrån flera olika perspektiv så som valet av slutparametrar och djupet på kvantkretsen. Här diskuteras skillnader och likheter för de, i denna studie, berörda perspektiven.

Slutparametrar I simuleringsarna användes två olika metoder för att välja de sluttillgängliga optimeringsparametrarna $(\vec{\gamma}, \vec{\beta})$ i QAOA, se (1) och (2) i avsnitt 3.2. De två olika sätten gav olika effekt på SP och AR. För SP går det att se en svag förbättring då slutparametrarna sätts till de som optimeringen konvergerat till, (2), istället för de parametrar som ger bäst AR (1). Denna förbättring betyder dock inte att Gibbsfunktionen alltid presterar bättre än vad standardmålfunktionen gör, utan skillnaden verkar bero på flera faktorer. AR blir generellt sett sämre för denna metod att välja slutparametrar, (2), skillnaden är dock liten. Vidare har Gibbs- och standardlandskapet inte samma optimala parametrar och Gibbsfunktionens miniman tycks bättre överensstämma med maximum för SP, se figur 4.8. Prestandan för de två olika valen av slutparametrar skiljer sig således som förväntat från jämförelsen av energilandskapen.

Prestandamått SP och AR verkar inte vara ekvivalenta; högre SP ger inte alltid bättre AR. Gibbsfunktionen tycks prioritera förbättringar av SP högre än förbättringar av AR. Idén bakom Gibbsfunktionen är att maximera SP, vilket ofta tycks medföra sämre AR. En förklaring är att miniman i energilandskapen för SP och AR verkar vara förskjutna relativt varandra. Ytterligare stöd för att SP och AR inte är ekvivalenta erhålls även från att de två olika valen av slutparametrar ger olika resultat.

Resultaten pekar på att Gibbsfunktionen möjligtvis bör användas om det önskas en hög SP eftersom Gibbsfunktionen då står sig bättre relativt standard. Notera att det finns många fall där standardversionen har bättre SP än Gibbsversionen. Valet av slutparametrar utefter bästa AR inte är optimalt. Detta eftersom det ger sämre SP relativt de faktiska slutparametrarna med Gibbsfunktionen och vanligen sämre AR än standard. Hur avvägningen mellan SP och AR kan se ut visas i sannolikhetsdistributionen i figur 4.3.

Sannolikhetsdistribution Från sannolikhetsdistributionen i figur 4.3 går det att urskilja vissa skillnader mellan Gibbsfunktionen och standardmålfunktionen. Det är tydligt att Gibbsfunktionen prioriterar att öka sannolikheten för den högsta energinivån medan standardversionen är mer normalfordelad kring energisnittet, $\langle \hat{H}_C \rangle$. Kostnaden för att få en högre SP är att Gibbsfunktionen får en lite högre sannolikhet för de lägre energinivåerna. De lägre värdena ger att approximationskvoten blir lite lägre för Gibbsfunktionen än för standard. Notera dock att figur 4.3 endast är baserad på en graf och ett visst djup. Ett intressant mönster i figur 4.3 är att sannolikheten för udda energivärden är noll. Anledningen skulle kunna vara en symmetri hos 4-reguljära grafer, men då inga andra typer av grafer undersökts kan det här inte styrkas.

Djup För ökande djup, där parametrarna interpoleras enligt avsnitt 2.3, har resultatet för alla simuleringar förbättrats. Det här är vad som förväntades utifrån tidigare forskning [15]. I ett fåtal grafer försämras Gibbsfunktionens SP då djupet ökar, exempelvis nodantal sju då djupet ökas från tre till fyra i figur 4.2. Förklaringen skulle kunna vara att algoritmen hamnat närmare ett sämre minimum i energilandskapet, och därmed konvergerade till detta, då djupet ökades. Ett annat alternativ är att algoritmen konvergerade till ett minima för det lägre djupet men inte för det högre.

Det finns skillnader i hur stor förbättring ett ökat djup ger för både Gibbsfunktionen och standardmålfunktionen. Förbättringen är inte konstant och saknar ett tydligt mönster. Den stora variationen på förbättringen väcker frågor om hur pass generaliserabara resultat simulerade med $p = 5$ är relativt $p = 10$. Att ökat p tycks ge olika effekt tyder på möjligheten att observationen från energilandskapen, om att alla lokala miniman tycks vara globala, inte nödvändigtvis stämmer. Det kan för $p > 1$ förekomma lokala miniman som inte är globala miniman. Att det skulle finnas sådana miniman motsägs inte av våra simulerade data och har stöd av tidigare forskning [21].

Det finns också skillnader i hur standardmålfunktionen presterar relativt Gibbsfunktionen för olika djup. För de större graferna på stora djup, se figur 4.2 (c), (d), 4.5 (c) och (d), är standardmålfunktionen tydligt bättre än Gibbsfunktionen för både AR och SP. Att standardversionen är bättre för de större graferna är något som inte återfinns för lägre djup. Anledningarna till mönstret skulle kunna vara de specifika graferna som används. Det är dock osannolikt att graferna med storlek 14 till 16 i grafmängd A är undantag med fördel till standard. En annan förklaring är att Gibbsfunktionen inte får en lika stor förbättring av ökat p för stora grafer som standard. Det här skulle vara en stor nackdel för Gibbsfunktionen eftersom de flesta verkliga tillämpningar kräver stora p . Ännu en möjlig anledning till skillnaden i prestanda är att Gibbsversionens optimala parametrar inte följer den trend som interpolationsmetoden bygger på, se avsnitt 2.3. Det här skulle medföra en nackdel för Gibbsversionen gentemot standardversionen specifik för denna studie.

Isomorfism Vilken isomorfism grafen har tycks påverka resultatet. Med de olika icke-isomorfa graferna i mängd A, B och C har simuleringarna gett relativt spridda resultat. Isomorfismem har alltså en stor påverkan på utfallet för algoritmen både för Gibbsfunktionen och standard. Det verkar alltså som att de olika isomorfismerna är olika svåra att lösa för algoritmerna. Isomorfism verkar dock påverka SP och AR på liknande sätt. De olika isomorfismerna påverkas också mycket lika av att djupet ökar. Olika isomorfismer är något som Li et al. inte tog hänsyn till då de endast studerade två specifika grafer [1]. Det här kan delvis förklara att resultaten i studien skiljer sig från resultaten i denna studie.

När nodantalet går över 14 ökar skillnaderna mellan de olika isomorfismerna. Den ökade skillnaden gäller både för SP och för AR. Resultatet är inte det som förväntas från teorin som säger att skillnaden mellan olika isomorfismer bör minska när grafstorleken ökar, se avsnitt 2.6. En möjlig orsak till det oväntade resultatet är att graferna behöver ha ett större nodantal än de som undersökts för att effekten av delgrafer ska uppstå.

Icke konvergens Från energilandskapen tycks det som att både Gibbsfunktionen och standardmålfunktionen ibland inte konvergerar för att optimeraren fastnar på en platå. Fenomenet tycks vara vanligare för standardversionen än Gibbsversionen, något som skulle kunna vara en viktig fördel för Gibbsversionen. Om resultaten för $p = 1$ kan generaliseras till högre p är dock oklart.

Val av parametern η För att vara säker på att få en rättvis jämförelse av Gibbsfunktionen och standardmålfunktionen är det viktigt att η väljs rätt. Figurerna 4.9, 4.10 och 4.11 tycks visa på att η måste väljas tillräckligt högt för att få största möjliga SP för Gibbsfunktionen. Resultatet är konsistent med tidigare forskning, i synnerhet är figur tre i Li et al. [1, fig. 3] konsekvent med 4.10 och 4.11, dock användes en något modifierad variant av SP i Li et al. Resultatet är också rimligt sett till teorin då små η gör att SP inte prioriteras lika högt. Större η tycks också ha en koppling till större höjdskillnader i energilandskapet enligt figur 4.7, där högre höjdskillnad tycks ge snabbare konvergens. Det är rimligt att större η ger högre höjdskillnad sett till teorin då större η gör att skillnaderna i energiernas viktning i Gibbsfunktionen ökar. Att högre höjdskillnad ger snabbare konvergens är mycket väntat då steglängden för den använda optimeraren är proportionell med gradienten [19]. Huruvida större η ger bättre eller sämre AR tycks variera.

Mycket höga värden på η tycks baserat på figurerna 4.9, 4.10 och 4.11 inte ge bättre SP, utan SP planar ut för högre η . Det är heller inte uteslutet att SP sjunker något för stora η . Vad detta beror på är inte klarlagt. En praktisk anledning till att inte välja ett mycket stort η är de numeriska problemen som uppstår vid representation av mycket stora flyttal.

Utifrån figur 4.9, 4.10 och 4.11 tycks $k \approx 20$ (där k kommer från $\eta = \frac{k}{E_{gs}}$) vara tillräckligt stort för att ge största möjliga SP. Valet av η som gjordes i studien anses därför vara rimligt. Det är dock möjligt att andra $\eta = \frac{k}{E_{gs}}$ hade kunnat ge något bättre resultat eftersom $k = 20$ inte är optimalt i alla mätpunkter, se figur 4.7.

5.1 Framtida studier

Vidare studier bör både undersöka effekter vid ökat djup i kvantkretsen och större grafer, eftersom det är först då resultaten blir direkt tillämpbara på praktiskt intressanta problem. Studien har använt ett relativt begränsat urval av grafer samt enbart 4-reguljära grafer. En expansion av urvalet är av intresse för att ge en komplett bild av prestandan. Sådana studier hade varit relevanta eftersom resultaten för små grafer och litet djup inte tycks vara generaliseringarbara.

Det hade även varit av intresse att kvantitativt jämföra antalet steg till konvergens för de två algoritmerna. På grund av de konvergenskriterier som används är detta inte möjligt i denna rapport. För en jämförelse av stegen till konvergens skulle kriterierna behöva revideras. Jämförelsen skulle troligtvis även kunna relateras till val av η då en koppling till höjdskillnad i energilanskapsen, och därmed steg till konvergens, tycks finnas.

Ett relevant område att utforska är brus då det är ett stort hinder för dagens kvantdatorer [10]. Det här gör en algoritms bruskänslighet central för dess praktiska användning. Även effekten av att sampla tillstånd, vilket görs i en riktig kvantdator, istället för att använda den kompletta sannolikhetsfördelningen hade kunnat analyseras.

För att förstå när och varför prestandan för Gibbsfunktionen och standardmålfunktionen skiljer sig åt är det av intresse att mäta risken för att Gibbsfunktionen och standardmålfunktionen fastnar på platåer. Risken verkar variera sett till energilandskaps utformning. Framtida studier kan även undersöka huruvida det finns en koppling mellan exempelvis grafens storlek och denna risk.

Under arbetets gång noterades flertalet intressanta mönster som inte direkt kopplade till studiens frågeställning. Dessa mönster skulle kunna undersökas av framtida studier av Max-Cut och QAOA. Exempelvis var sannolikheten att få udda energier för 4-reguljära grafer noll och energilandskapet var mer periodiskt än förväntat.

6 Slutsats

Generellt sett tycks det inte vara statistiskt säkerställt huruvida Gibbsversionen eller standardversionen är att föredra. Vissa tendenser anas – Gibbsversionen tycks stå sig bättre i fråga om SP jämfört med AR. Notera att detta inte nödvändigtvis betyder att SP för Gibbsversionen är bättre än standardversionen. Att Gibbsversionen är bättre på SP än AR är också vad som förväntas teoretiskt. För de högsta djupen och största graferna som simulerades presterade standardversionen bättre än Gibbsversionen.

Generellt sett har den tioprocentiga förbättringen för Gibbsversionen relativt standardversionen, som hittades i Li et al. [1], inte kunnat reproduceras. De simuleringar som gjorts har istället visat på att den förbättring som studien hittat endast gäller för vissa grafer och djup, alltså att resultaten inte är generaliserabara. Notera att den tidigare studien använde ett prestationsmått som skiljer sig något från SP. Skillnaden syns dock bara för större grafer.

De olika metoderna att välja slutparametrar har vissa skillnader. Det skulle vara möjligt att välja mellan att prioritera AR eller SP med Gibbsfunktionen. Att prioritera AR tycks dock vara ointressant, eftersom standard för det mesta ger bättre AR även om denna prioritering görs.

För att Gibbsversionen ska prestera på bästa möjliga sätt i fråga om SP tycks ett rimligt val av η vara $\eta \approx 20/E_{gs}$ där E_{gs} är energin för grundtillståndet. Även något högre värden på η tycks fungera. Detta resultat är konsistent med Li et al. [1]. Skälet till att ett lågt η inte ska väljas är att det inte prioriterar SP lika högt. En annan observation som gjorts var att höga η har en viss koppling till hur snabbt algoritmen konvergerar. Högre η kan dock ge lägre AR och i vissa fall även numeriska problem.

Referenser

- [1] L. Li, M. Fan, M. Coram, P. Riley och S. Leichenauer, "Quantum optimization with a novel Gibbs objective function and ansatz architecture search," *Phys. Rev. Res.*, årg. 2, s. 023074, 2 april 2020. DOI: 10.1103/PhysRevResearch.2.023074.
- [2] V. Bergholm, J. Izaac, M. Schuld m. fl., *PennyLane: Automatic differentiation of hybrid quantum-classical computations*, 2022. DOI: <https://doi.org/10.48550/arXiv.1811.04968>.
- [3] C. Cookson och G. Parker, "UK to invest £2.5bn in quantum computing drive," 2023, Hämtad 2023-5-10. [Online]. Tillgänglig: <https://www.ft.com/content/900ba4ee-6c60-4406-963a-810771c2ad7c>.
- [4] M. Swayne, "India Announces \$730 Million-Plus National Quantum Mission," 2023, Hämtad 2023-5-10. [Online]. Tillgänglig: <https://thequantuminsider.com/2023/04/20/india-announces-730-million-plus-national-quantum-mission/>.
- [5] L. Pautasso, A. Pflanzer och H. Soller, "The current state of quantum computing: Between hype and revolution," 2021, Hämtad 2023-5-10. [Online]. Tillgänglig: <https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/tech-forward/the-current-state-of-quantum-computing-between-hype-and-revolution>.
- [6] M. Elkær, "Ny valuta på väg: Det blir fem kvantkronor, tack," 2023, Hämtad 2023-5-10. [Online]. Tillgänglig: <https://illvet.se/teknik/datorer/ny-valuta-pa-vag-det-blir-fem-kvantkronor-tack>.
- [7] M. Swayne, "Quantum Sensing Method Holds Promise for Improving Greenhouse Gas Detection," 2023, Hämtad 2023-5-10. [Online]. Tillgänglig: <https://thequantuminsider.com/2023/04/07/quantum-sensing-method-holds-promise-for-improving-greenhouse-gas-detection/>.
- [8] "Shor's algorithm," *IBM Quantum*, Hämtad 2023-5-10. [Online]. Tillgänglig: <https://quantum-computing.ibm.com/composer/docs/iqx/guide/shors-algorithm>.
- [9] M. A. Nielsen och I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge University Press, 2010.
- [10] C. Xue, Z.-Y. Chen, Y.-C. Wu och G.-P. Guo, "Effects of quantum noise on quantum approximate optimization algorithm," *Chinese Physics Letters*, årg. 38, nr 3, s. 030302, 2021. DOI: 10.1088/0256-307x/38/3/030302.
- [11] J. Bylander, S. Gustavsson, F. Yan m. fl., "Noise spectroscopy through dynamical decoupling with a superconducting flux qubit," *Nature Physics*, årg. 7, nr 7, s. 565–570, 2011. DOI: 10.1038/nphys1994.
- [12] J. Preskill, "Quantum Computing in the NISQ era and beyond," *Quantum*, årg. 2, s. 79, aug. 2018, ISSN: 2521-327X. DOI: 10.22331/q-2018-08-06-79.
- [13] M. Swayne, "What Are Optimization Problems? How QCI's Unique Quantum Approach Can Help Organizations Solve Complex Problems Across Industries," 2023, Hämtad 2023-5-10. [Online]. Tillgänglig: <https://thequantuminsider.com/2023/03/08/what-are-optimization-problems-how-qcis-unique-quantum-approach-can-help-organizations-solve-complex-problems-across-industries/>.
- [14] G. E. Crooks, "Performance of the Quantum Approximate Optimization Algorithm on the Maximum Cut Problem," 2018. DOI: 10.48550/ARXIV.1811.08419.

- [15] E. Farhi, J. Goldstone och S. Gutmann, “A Quantum Approximate Optimization Algorithm,” 2014. DOI: 10.48550/ARXIV.1411.4028.
- [16] L. Zhou, S.-T. Wang, S. Choi, H. Pichler och M. D. Lukin, “Quantum Approximate Optimization Algorithm: Performance, Mechanism, and Implementation on Near-Term Devices,” *Phys. Rev. X*, årg. 10, s. 021067, 2 juni 2020. DOI: 10.1103/PhysRevX.10.021067.
- [17] B. D. McKay, N. C. Wormald och B. Wysocka, “Short cycles in random regular graphs,” *The Electronic Journal of Combinatorics*, årg. 11, nr 1, 2004. DOI: 10.37236/1819.
- [18] E. Farhi, D. Gamarnik och S. Gutmann, *The Quantum Approximate Optimization Algorithm Needs to See the Whole Graph: Worst Case Examples*, 2020. arXiv: 2005.08747 [quant-ph].
- [19] *qml.GradientDescentOptimizer*, Hämtad 2023-05-09. [Online]. Tillgänglig: <https://docs.pennylane.ai/en/stable/code/api/pennylane.GradientDescentOptimizer.html>.
- [20] *A library for variational quantum algorithms*, hämtad 2023-05-09. [Online]. Tillgänglig: <https://github.com/aqp-mc2-chalmers/vqa>.
- [21] J. Rivera-Dean, P. Huembeli, A. Acín och J. Bowles, *Avoiding local minima in Variational Quantum Algorithms with Neural Networks*, 2021. arXiv: 2104.02955 [quant-ph].

Institutionen för Mikroteknologi och nanovetenskap

CHALMERS UNIVERSITY OF TECHNOLOGY

Gothenburg, Sweden

www.chalmers.se



CHALMERS