

NV Sieve on Grover

<https://youtu.be/t6ChS6ljzDY>

저번 발표 이후 진행 사항










- 저번 발표에서 **Overflow**를 대충 해결한 점을 개선

- 당시 시간상 간단하게 큐비트르 2배로 하여 overflow를 해결함으로써 기존의 구현물을 일부 재활용 하였음
- 그러나 이는 자원 낭비이며, 생각해보니 1 큐비트만 더 있어도 범위 커버 가능
- 변경 완료하였으며, 구현 상 다른 점은 딱히 없고 약간의 수정 및 스케일링 거침
- **이를 통해 자원과 큐비트를 많이 아낌**
- 이외에도 최적화 가능 지점을 찾아 봐야 할 것 (우선 나중에..)

- 양자 비용 증가량의 경향을 보기 위한 추가 구현

- R2D2 (Rank=2, Dimension=2)~R4D4까지 Rank 및 Dimension을 증가시키며 오라클 비용 측정

Lab files / KISTI / OF수정 /

Name ▲	Last Modified
 (실행중)R4D4.ipynb	5 days ago
 (완료)R2D2.ipynb	5 days ago
 (완료)R2D3.ipynb	5 days ago
 (완료)R2D4.ipynb	5 days ago
 (완료)R3D2.ipynb	5 days ago
 (완료)R3D3.ipynb	an hour ago
 (완료)R3D4.ipynb	5 days ago
 (완료)R4D2.ipynb	5 days ago
 (완료)R4D3.ipynb	5 days ago

- **Grover**는 일부 실행

- 현재 최대 범위에서의 공격 비용이 궁금해서 R4D4는 Grover를 돌려 두고 학술대회에 다녀왔더니 커널이 끊김
- 실행 시간이 비교적 짧은 R2D2, R3D3 우선 실행 (iteration=1)
 - R2D2 : 1.3초
 - R3D3 : 539.9초
 - R4D4 : 이대로라면 최소 3일 이상..? (그래서 런타임이 끊긴 것으로 보임)

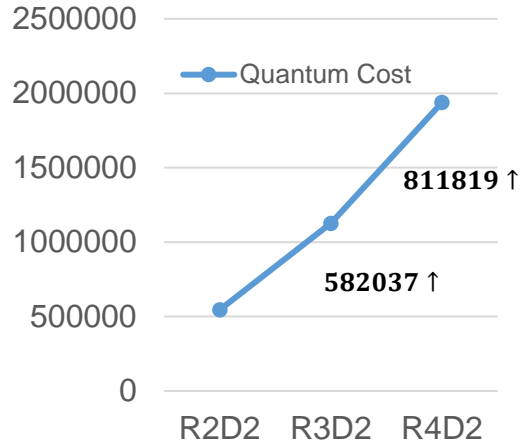
Overflow 해결 방식 수정

- **Overflow 해결 방식을 수정함으로써 양자 비용이 전반적으로 감소**
 - 현재 구현의 R4D4는 큐비트 수를 포함한 모든 양자 비용이 지난 구현의 R2D4 (*ExDim*)보다 적음
 - 오라클을 더욱 최적화 한다면 양자 비용을 더 많이 감소시킬 수 있을 것으로 보임

저번 R2D2									이번 R2D2								
iter=1, default	#CNOT	#1qCliff	#T	T-depth	Full-depth	#Qubit			iter=1, R2D2	#CNOT	#1qCliff	#T	T-depth	Full-depth	#Qubit		
Oracle	477	140	194	582	1580	112			Oracle	291	69	124	396	1126	74		
저번 R3D2									이번 R3D2								
iter=1, rank	#CNOT	#1qCliff	#T	T-depth	Full-depth	#Qubit			iter=1, R3D2	#CNOT	#1qCliff	#T	T-depth	Full-depth	#Qubit		
Oracle	705	203	284	848	2297	160			Oracle	420	90	181	576	1631	105		
저번 R2D4									이번 R2D4								
iter=1, dim	#CNOT	#1qCliff	#T	T-depth	Full-depth	#Qubit			iter=1, R2D4	#CNOT	#1qCliff	#T	T-depth	Full-depth	#Qubit		
Oracle	1613	560	722	2102	5543	344			Oracle	685	224	296	878	2352	179		

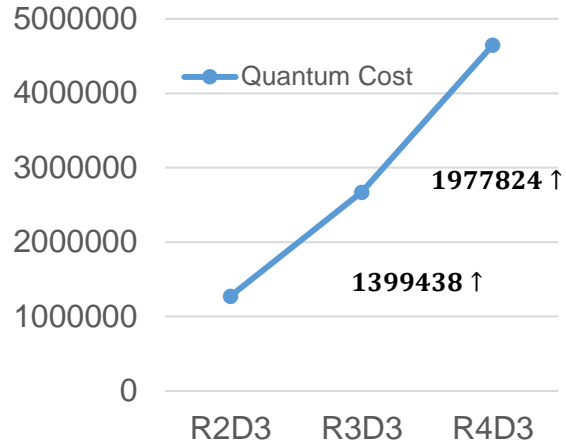
추가 구현들의 Oracle 비용 측정 (그래프)

Quantum Cost



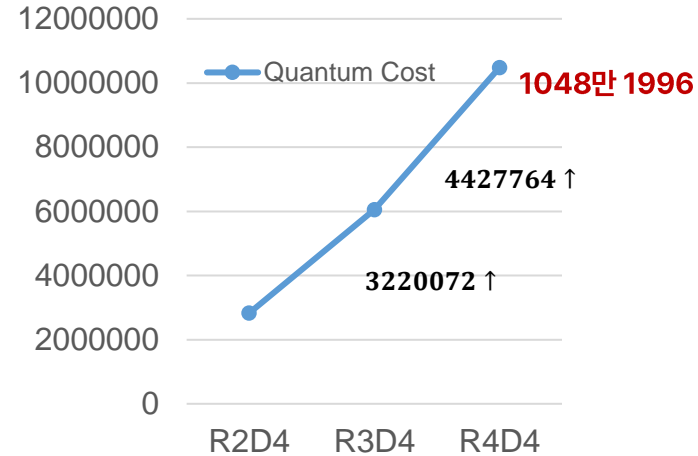
D2에서 랭크 증가

Quantum Cost



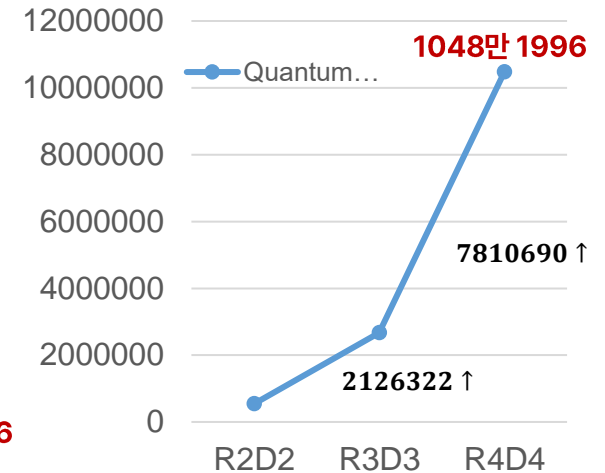
D3에서 랭크 증가

Quantum Cost



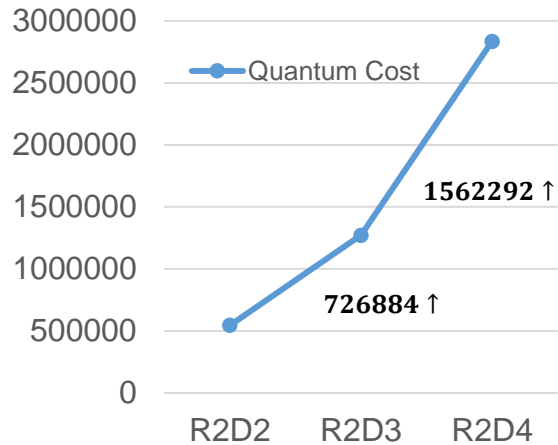
D4에서 랭크 증가

Quantum Cost



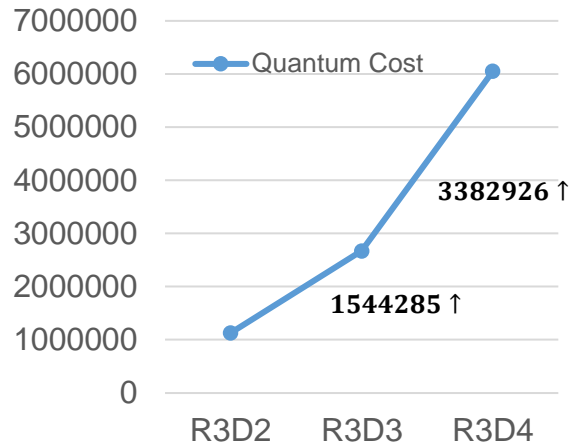
R2D2에서 랭크/디멘션 동시 증가

Quantum Cost



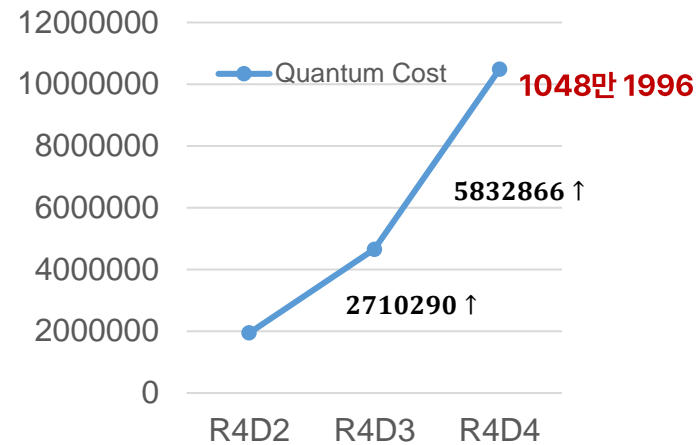
R2에서 디멘션 증가

Quantum Cost



R3에서 디멘션 증가

Quantum Cost



R4에서 디멘션 증가

추가 구현들의 Oracle 비용 측정 (해석)

- 동일하게 1씩 스케일 업 할 때, **Dimension** 증가가 Rank 증가보다 Quantum Cost 증가에 더 많은 영향을 미침
 - 동일 **Dimension**에서 Rank를 늘릴 때, 증가량이 비슷 (1.3~1.4배)
 - 그래프 상에서 약간 linear한 모양새에 더 가까운 듯..
 - 현재 디멘션에 따라 랭크업에 대한 증가량이 커지긴 함
 - 동일 **Rank**에서 **Dimension**을 늘릴 때, 증가량이 크게 증가 (2배 이상)
 - exponential하게 증가할 것으로 보임..
- 당연하지만, Rank / Dimension이 커질수록 Quantum Cost의 증가량도 커짐
- 해당 그래프는 Oracle에 대한 비용이며, **실제 Grover 공격 비용은 이보다 2배 이상 커질 것** (Grover 적용 + iteration)
 - Grover 적용으로 인해 약 2배
 - 적절한 iteration이 1보다 클 경우, 공격 비용 증가 (저번 실험에 의하면 iteration이 많아질수록 공격 비용의 증가량이 증가함)
 - 실제로 R2D2, R3D3를 실행해 본 결과 오라클의 약 2배의 자원 필요 (iteration=1인 경우)

약 2배 {	iter=1, R2D2	#CNOT	#1qCliff	#T	T-depth	Full-depth	#Qubit
	Oracle	291	69	124	396	1126	74
	Grover	582	142	248	792	2259	75
약 2배 {	iter=1, R3D3	#CNOT	#1qCliff	#T	T-depth	Full-depth	#Qubit
	Oracle	675	207	284	848	2291	160
	Grover	1350	439	568	1696	4597	161

계획

- 더 큰 Rank 및 Dimension 구현 및 비용 측정

- 이에 대한 결과 그래프 작성
- 검색 비용에 관한 그래프 작성

- Noise

- 노이즈가 첨가된 **fake hardware**가 있으나, 아마 **100 큐비트 전후**로 사용 가능
→ 이전에 사용해봤던 경험에 의하면 **엄청 느림**
- 이 방법 말고는 **게이트에 노이즈를 첨가**할 수 있음
→ 해당 부분은 논문이나 관련 정보를 찾아보고 오류율에 따른 Grover를 위한 비용을 추정해 봐야 할 것으로 보임..

- QPE의 정확도가 떨어질 경우, 적절한 iteration 찾기 어려워짐 → 옳은 솔루션 도출 불가

- **위상 추정 큐비트의 수를 늘리기** / 반복 위상 추정이라는 **Iterative QPE**를 사용한다고 하며, 필요 시 적용 예정

감사합니다.