

Grover search algorithm 을 이용한 Hamiltonian path 문제 접근법

김범일(고려대학교), 민건식(고려대학교), 허준(고려대학교)*

bik0118@korea.ac.kr, mgs3351@korea.ac.kr, junheo@korea.ac.kr

Approach for Hamiltonian path problem using Grover search algorithm

Kim Bum Il(Korea Univ.), Min Gun Sik(Korea Univ.), *Heo Jun(Korea Univ.)

요약

본 논문에서는 Grover Search algorithm을 이용하여 Hamiltonian path problem의 해결 방안을 제안한다. Hamiltonian path problem을 Satisfiability problem으로 환원하여 Grover Search algorithm에 적용한다. vertex가 3개인 경우의 그래프의 해답을 구하는 시뮬레이션을 Qiskit으로 구현하여 풀어냄으로써 제안한 방식을 검증한다.

I. 서론

Grover Search algorithm은 양자 역학의 superposition과 entanglement를 이용하여 정답을 찾는 알고리즘으로 기존의 디지털 알고리즘에서는 계산 복잡도인 $O(N)$ 에서 $O(\sqrt{N})$ 으로 감소시키는 알고리즘이다[1]. Hamiltonian path problem(HPP)는 방향 그래프나 무방향 그래프에서 그래프의 모든 vertex를 단 한번씩만 통과하는 경로를 찾는 문제이다. HPP와 Satisfiability (SAT) problem은 NP-complete problem으로 서로 환원이 가능하다[2]. SAT problem은 Grover search algorithm을 이용하여 보다 빠르게 답을 찾을 수 있다. 이를 이용하여 본 논문에서는 HPP의 정답을 구하기 위해 SAT problem으로 환원시킨 후 Grover search algorithm에 적용하여 양자 회로로 구성하는 방법을 제시하고, 시뮬레이션을 통해 타당성을 확인한다.

II. 본론

A. Grover search Algorithm

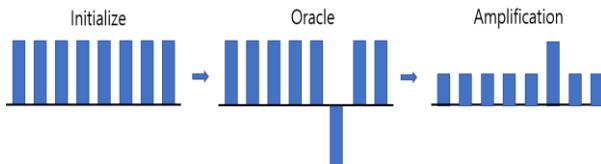


그림 1. Grover search algorithm 의 개요도

그림 1은 Grover search algorithm의 개요도를 나타낸다. Grover search algorithm은 입력들을 중첩 상태로 만드는 Initialize 단계와 중첩된 입력 중에 원하는 입력의 위상을 반전시키는 Oracle, 위상이 반전된 입력을 다시 반전시키고 증폭시키는 Amplification 단계로 구성된다.

Initialize 단계의 경우, Hadamard gate를 이용하여 식(1)과 같은 중첩 상태를 형성한다.

$$|0\rangle \xrightarrow{H} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad (1)$$

Oracle 단계는 다음 식(2)와 같은 Oracle 연산자(O)를 사용하여 연산을 수행한다.

$$O(x) = \begin{cases} 1 & \text{if } |x\rangle = |x_0\rangle \\ 0 & \text{otherwise} \end{cases}$$

식(2)에서 $|x_0\rangle$ 를 정답인 상태, $|x\rangle$ 는 임의의 상태를 의미한다. 따라서, Oracle 연산자는 중첩된 입력 중 찾고자 하는 입력의 위상만 반전시키는 연산을 수행한다. Amplification 단계는 식(3)과 같은 연산을 수행한다.

$$\begin{aligned} A &= 2|\psi\rangle\langle\psi| - I, \\ |\psi\rangle &= \frac{1}{\sqrt{N}} \sum_{x \neq x_0} |x\rangle + \frac{1}{\sqrt{N}}|x_0\rangle \end{aligned} \quad (3)$$

이 연산을 통해 정답인 상태의 위상이 다시 반전되고 증폭되어 정답이 아닌 다른 상태보다 확률이 증가하게 된다. 또한, Grover search algorithm은 반복 수행을 통해 정답 검출 확률을 증가시킬 수 있다. 가장 최적의 반복 횟수의 식은 다음 식(4)와 같다[3].

$$t \leq \frac{\pi}{4} \sqrt{\frac{N}{M}} \quad (4)$$

N은 입력 상태의 개수, M은 답이 되는 상태의 수이다.

B. 그래프

그래프는 $G=(V,E)$ 로 표현된다. 이때, V는 vertex(꼭지점)의 집합, E는 edge의 집합을 의미한다. 임의의 vertex x, y가 있을 때, x와 y를 꼭지점으로 하는 edge가 있다면 $\{x,y\}$ 로 표기하고 x와 y는 인접하다고 말한다.

C. HPP의 SAT problem으로의 환원

SAT problem이란 임의의 변수들과 AND, OR, NOT을 이용하여 만든 논리식이 참값을 갖는 변수 값을 찾는 문제이다. 따라서, 임의의 그래프 G가 주어졌을 때, HPP를 논리식으로 환원하여 논리식이 참이 되는 값을 찾는다면 HPP의 해를 찾을 수 있다.

HPP는 다음과 같은 5가지의 조건을 만족해야 한다.

- ① 모든 vertex를 통과해야 한다.
- ② 같은 vertex는 두 번 이상 통과할 수 없다.
- ③ 경로상의 모든 순서에 vertex가 반드시 할당되어야 한다.
- ④ 경로상 같은 순서에 두 개 이상의 vertex가 할당될 수 없다.
- ⑤ 인접하지 않은 두 vertex는 연속으로 통과할 수 없다.

그래프 G 가 n 개의 vertex를 가지고 $x_i^j (1 \leq i, j \leq n)$ 는 경로상의 vertex를 i , 경로상의 순서 j 를 나타낸다고 정의하면 주어진 조건은 다음의 논리식에 대응된다.

$$(x_i^1 \vee x_i^2 \vee x_i^3 \vee \dots \vee x_i^n) \text{ for all } i \quad ①$$

$$(\neg x_i^j \vee \neg x_i^k) \text{ for all } i, j, k, 1 \leq k \leq n, (j \neq k) \quad ②$$

$$(x_1^j \vee x_2^j \vee x_3^j \vee \dots \vee x_n^j) \text{ for all } j \quad ③$$

$$(\neg x_i^j \vee \neg x_k^j) \text{ for all } i, j, k, 1 \leq k \leq n, (i \neq k) \quad ④$$

$$(\neg x_i^l \vee \neg x_j^{l+1}) \text{ for all } i, j, l \quad ⑤$$

$$\text{such that } \{i, j\} \notin E, 1 \leq l \leq n - 1 \quad ⑤$$

결과적으로 식(4)와 같이 정리된다.

$$\begin{aligned} & \bigwedge_{x_i \in V} (x_i^1 \vee x_i^2 \vee \dots \vee x_i^n) \wedge \bigwedge_{x_i \in V} \bigwedge_{j,k=1, j \neq k}^n (\neg x_i^j \vee \neg x_i^k) \\ & \wedge \bigwedge_{j=1}^n (x_1^j \vee x_2^j \vee \dots \vee x_n^j) \wedge \bigwedge_{x_i, x_k \in V, i \neq k} \bigwedge_{j=1}^n (\neg x_i^j \vee \neg x_k^j) \\ & \wedge \bigwedge_{\{x_i, x_j\} \notin V} \bigwedge_{l=1}^{n-1} (\neg x_i^l \vee \neg x_j^{l+1}) = 1 \end{aligned} \quad (4)$$

D. Oracle 구현 방법

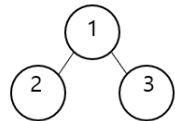


그림 2. Hamiltonian path 시뮬레이션 그래프

그림 2는 제안한 방식을 검증하기 위한 시뮬레이션의 그래프이다. 현재, IBM Q에서 이용할 수 있는 quantum simulator의 한계로 이용할 수 있는 큐비트의 개수가 32개를 초과할 수 없기 때문에 그림2와 같은 간단한 그래프를 이용한다.

AND	OR	NOT
$A \wedge B$	$A \vee B$	$\neg A$
A B	A \boxtimes B	A \boxtimes $\neg A$
B A	B \boxtimes A	B \boxtimes B
0 \oplus	0 \oplus	\boxtimes $A \vee B$
$A \wedge B$	$A \vee B$	$\neg A$

표 1. 양자회로로 구현된 논리 연산 [4]

표 1의 논리 연산을 이용하여 식(4)를 그림 2에 주어진 그래프에 적용하면 다음과 같이 조건식별로 양자회로를 구성할 수 있다.

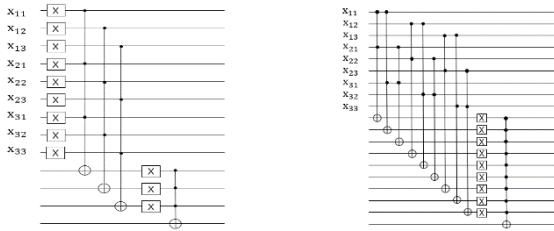


그림 3. 조건식에 대한 양자회로 a) 조건식 ③ b) 조건식 ④

예를 들어 조건식 ③은 다음과 같은 논리식으로 나타난다.

$$(x_1^1 \vee x_2^1 \vee x_3^1) \wedge (x_1^2 \vee x_2^2 \vee x_3^2) \wedge (x_1^3 \vee x_2^3 \vee x_3^3) \quad (5)$$

조건식 ④는 다음과 같은 논리식으로 나타난다.

$$\begin{aligned} & (\neg x_1^1 \vee \neg x_2^1) \wedge (\neg x_1^1 \vee \neg x_3^1) \wedge (\neg x_2^1 \vee \neg x_3^1) \\ & \wedge (\neg x_{12} \vee \neg x_{22}) \wedge (\neg x_{12} \vee \neg x_{32}) \wedge (\neg x_{22} \vee \neg x_{32}) \\ & \wedge (\neg x_{13} \vee \neg x_{23}) \wedge (\neg x_{13} \vee \neg x_{33}) \wedge (\neg x_{23} \vee \neg x_{33}) \end{aligned} \quad (6)$$

조건식 ①의 경우 조건식 ③과, 조건식 ②, ⑤의 경우 조건식 ④과 구성이 유사하므로 약간의 수정을 통해 조건식에 맞는 양자회로가 구성되는 것을 알 수 있다.

E. 예제에 대한 시뮬레이션 결과

Qiskit에서 qasm simulator를 이용하여 식(4)에 따라 12회 반복, 1024회 시뮬레이션을 수행하였다.

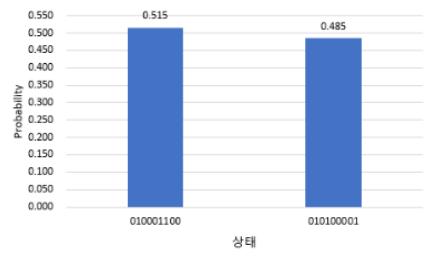


그림 4. 시뮬레이션의 결과

위와 같이 답이 되는 상태는 '010100001'과 '010001100'으로 각각 48.5%와 51.5% 확률로 검출되는 것을 확인할 수 있었다.

III. 결론

본 논문에서는 Grover Search algorithm을 이용하여 HPP의 해결 방안을 제안하였다. HPP를 SAT problem으로 환원하는 방법과 시뮬레이션을 통해 제안한 방식을 검증하였다. 추후에 입력 상태의 수를 감소시킨다면 소요자원을 줄여 개선시킬 수 있을 것으로 보인다.

ACKNOWLEDGMENT

이 논문은 2020년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No. 2019R1A2C2010061)

참고 문헌

- [1] L. K. Grover, "A fast quantum mechanical algorithm for database search," Proceedings of the 28th Annual ACM Symposium Theory of Computing, vol. 212, 1996.
- [2] Richard M. Karp, "Reducibility Among Combinatorial Problems," Complexity of Computer Computations, New York: Plenum, pp. 85–103, 1972.
- [3] M. Boyer, G. Brassard, P. Høyer and A. Tapp, "Tight bounds on quantum searching," Fortschritte Der Physik 46(4–5), pp. 493–505, 1998.
- [4] 하진영, 김태현, 이홍석, 허준, "양자 검색 알고리즘을 이용한 Exactly-1 3-SAT 문제 접근법," 한국통신학회 학술대회논문집, pp. 265–266, 2019.