

양자 미끄럼틀과 사다리(Chutes and Ladders)

Joshua Goings, Ph.D.; Applications Engineer @ IonQ

서론

고전 미끄럼틀과 사다리와 Markov 접근

[Chutes and Ladders \(또는 Snakes and Ladders\) 보드 게임](#)은 우연과 확률 개념을 탐구하기에 훌륭한 방법을 제공합니다. 이 보기에 간단한 게임은 [Markov 체인을 사용하여 모델링할 수 있으며](#), 이를 통해 게임의 구조와 흐름을 수학적으로 이해할 수 있습니다. 이 분석이 기반으로 하는 재미있는 블로그 기사는 [여기](#)에서 찾을 수 있습니다.

"Chutes and Ladders"는 "기억력이 없는" 게임이며, 각 턴의 결과는 이전 턴과 독립적입니다. 각 턴에서 6 면 주사위를 굴러 이동을 결정하며, 여섯 개의 앞에 있는 사각형 중 어떤 하나에 도착할 확률이 동일합니다. 행운이 따르면 사다리의 밑받침대에 도착하여 빨리 올라갈 수 있고, 행운이 따르지 않으면 미끄럼틀의 꼭대기에 도착하여 내려가게 될 수 있습니다. 그러나 이러한 사건들은 이전에 어디에 있었느냐와는 완전히 독립적입니다.

이 기억력이 없는 특성은 Markov 프로세스의 기본 원리인데, Markov 프로세스는 한 상태에서 다른 상태로의 확률적인 전이의 연속입니다. Chutes and Ladders 를 모델링할 때, 게임 보드의 각 사각형을 "상태"로 취급하고 주사위를 굴린 결과에 따라 이러한 상태 사이를 전환합니다. 이것은 각 행-열 교차점이 한 상태에서 다른 상태로 이동할 확률을 나타내는 전이 행렬로 수학적으로 나타낼 수 있습니다. 전이 행렬에서 유도된 통계치에 따르면 가장 짧은 게임은 평균 7 번의 움직임을 갖으며, 약 660 게임 중 한 번 발생합니다. 평균적으로 게임은 39 번의 턴을 가지지만, 확률 분포의 왜곡으로 인해 대부분의 게임은 32 번 이하로 끝나며, 22 번의 움직임이 가장 일반적입니다.

클래식한 Chutes and Ladders 게임의 이러한 확률적 측면을 분석하고 이해하는 것은 양자 버전인 Quantum Chutes and Ladders 를 고려하는 데 훌륭한 출발점을 제공합니다. 이것은 확률 분포, 상태 전이, 그리고 가장 중요한 양자 역학의 "양자 중첩" 개념을 생각하게 됨으로써 우리의 사고 방식을 형성하는 데 도움이 됩니다. 앞으로 나아가면, Quantum Chutes and Ladders 가 단순히 게임이 아니라 양자 역학 원리를 탐구하고 체험하는 놀이터가 됨을 알게 될 것입니다.

양자 보행(Quantum Walks): Markov 프로세스의 양자적 비유

Markov 프로세스의 양자 확장은 양자 보행으로 알려져 있습니다. 양자 보행은 다양한 양자 시스템의 행동을 모델링하는 중요한 도구로 사용되며 알고리즘 검색 문제에 응용됩니다. 고전적인 보행과 달리 양자보행은 가역적이고 유니터리 방식으로 작동합니다. 이러한 본질적인 차이는 상태 간의 전이가 확률적 행렬이 아닌 유니터리 연산자에 의해 지배되는 양자역학의 본질적인 특성으로 인해 발생합니다.

보다 구체적으로, 양자 보행은 마르코프 프로세스와 중요한 차이가 있는데 이는 전이 행렬 대신 유니터리 연산자를 사용하여 고전 시스템에는 없는 가역성을 도입한다는 것입니다. 이러한 유니터리 연산자는 상태 변화를 제어하는 "양자 전이 행렬"을 형성합니다. 또한 양자 보행은 양자 중첩을 활용하여 여러 보드 위치를 동시에 탐색할 수 있어 기존 시스템의 단일 상태 특성과 대조됩니다. 이는 상태 공간의 병렬 탐색으로 이어져 양자 보행을 Markov 프로세스보다 독특하고 복잡하게 만듭니다.

우리가 어린이 보드 게임의 맥락에서 양자 보행을 탐구하고 있지만, 현실적으로 양자 워크는 여러 유용한 양자 알고리즘에 중요합니다. 양자 보행은 그로버(Grover) 알고리즘과 같은 검색 알고리즘을 포함하여 많은 양자 알고리즘의 기본 구조로 밝혀졌습니다. 이러한 알고리즘은 양자 중첩에 의해 도입된 병렬성을 활용하여 기존 알고리즘보다 더 효율적으로 정렬되지 않은 데이터베이스를 검색합니다. 이러한 관점에서 볼 때, Chutes and Ladders 에서 Quantum Chutes and Ladders 로의 이동은 흥미로운 사례 연구 역할을 합니다. 이는 우리를 고전 확률 및 Markov 프로세스에서 양자 역학, 유니터리 연산 및 양자 보행 영역으로 전환합니다. 이 양자 버전은 더 복잡한 양자 현상과 알고리즘을 이해할 수 있는 길을 열어줍니다.

IonQ Quantum Challenge #1

"Chutes and Ladders" 또는 "Snakes and Ladders"라 불리는 클래식 게임은 흔한 어린이 보드 게임입니다. 이 게임은 "기억력이 없는" 본질을 잡아내기 위해 효과적으로 Markov 체인을 사용하여 모델링할 수 있습니다. 오늘, 우리는 이 간단한 게임을 양자적인 요소로 변형시킬 것입니다. Quantum Chutes and Ladders 에 오신 것을 환영합니다!

이 챌린지에서, 표준 6 면 주사위를 공정한 양자 "동전"으로 대체하고 양자 전이 매트릭스를 구성하여 보드를 이산 양자 랜덤 워크로 모델링함으로써 이 고전 게임을 양자 게임으로 변환하게 됩니다. 이 작업은 양자 개념 입문부터 고급 양자 연구까지 다양한 난이도 수준으로 평가될 것입니다.

Beginner Track

Task 1: 공정한 '양자 동전'을 만들어보세요. 이것은 어떤 연산자와 대응되나요? '네 면짜리' 양자 동전은 어떻게 보이나요? 선택한 양자 SDK 를 사용하여 구현을 제시해주세요. 두 면짜리와 네 면짜리 '양자 동전'이 공정한지 어떻게 확인할 수 있을까요??

Task 2: Chutes 와 Ladders 가 없는 간단한 보드에서 시작해봅시다.

15 (end)	14	13	12
8	9	10	11
7	6	5	4
0 (start)	1	2	3

Figure 1. 간단한 보드.

15 (end)	14	13	12
8	9	10	11
7	6	5	4
0 (start)	1	2	3

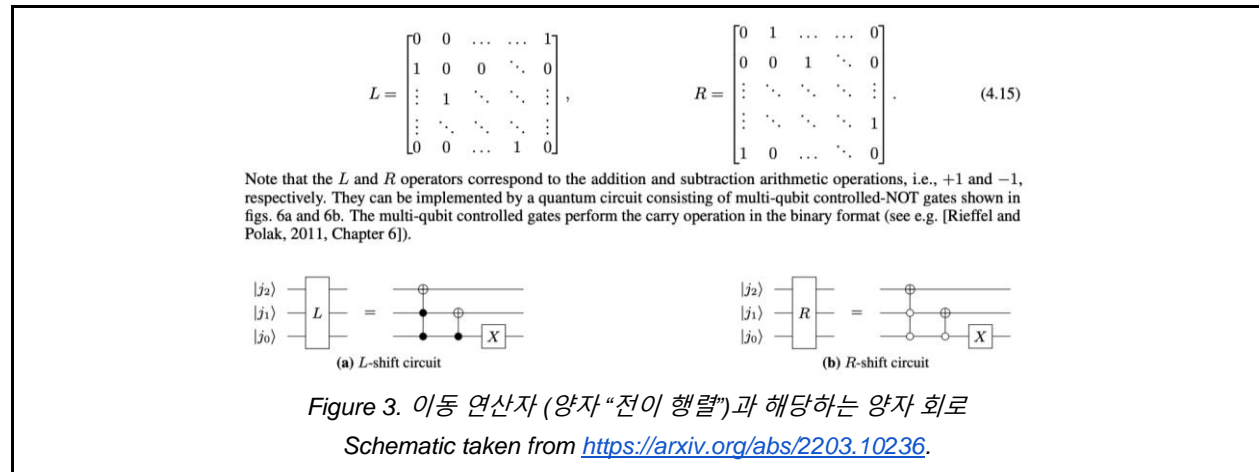
Figure 2. 간단한 보드와 게임 역학. 0,1,2.. 순으로 한 칸씩 앞으로 움직인다.

여기서 역학은 0 에서 시작하여 15 번째 사각형에 도달하면 게임이 종료된다는 것입니다. 보드(4x4 격자, 16 개 사각형)의 위치를 양자 상태로 어떻게 표현하시겠습니까?

Task 3: 이제 게임 역학을 표현하기 위한 양자 회로를 고안해 보세요. 즉, 양자 보행을 구현합니다. 양자 코인의 결과에 따라 상태를 앞으로 이동시키는 "이동 연산자"와 함께 양자 코인(양면 코인을 사용할 수 있음)을 사용하여 시뮬레이션해야 합니다. 측정하는 것이 아니기 때문에 보드는 이제

중첩 상태가 됩니다. 선택한 양자 SDK 를 사용하여 게임의 N 단계에 대한 게임 플레이를 발전시키는 프로그램을 구현합니다.

힌트 1: 주사위 굴리기와 말을 움직이기 위해서 양자 보행과 "이동" 연산자에서 영감을 얻어보세요 (이 논문의 4.2.1 절 참조). 예를 들어, 왼쪽과 오른쪽 "이동" 연산자는 한 개의 스쿼어를 하나 증가시키거나(또는 하나 감소시키는) 연산자로, 이를 다중 제어 토폴리 게이트로 구현할 수 있습니다.



왼쪽 또는 오른쪽 연산자만 사용하여 단방향 사례를 고려할 수 있습니다. 게이트는 순환 그래프를 모방하기 때문에 최종 상태에 "착륙"하고 "다시 굴림"하면 원래 상태, 즉 정사각형 15 에서 정사각형 0 으로 돌아갈 수 있습니다.

힌트 2: 양자 코인의 상태를 나타내기 위해 추가 큐비트도 고려해야 합니다. 이산 시간 양자 보행에 대한 위키피디아 페이지가 유용할 수 있습니다.

Advanced Track

Task 4: 조금 더 복잡한 상황을 만들어 봅시다. 이제 "미끄럼틀"과 "사다리"를 추가할 것입니다. 양자적으로 동작하기 위해서는 "미끄럼틀"과 "사다리"가 뒤집을 수 있는(유니터리) 연산이어야 하므로, 각각의 "미끄럼틀"과 "사다리"는 상태를 맞바꿀 것입니다. 이전과 마찬가지로 0 번 사각형에서 시작하며, 게임은 15 번 사각형에 도착할 때 끝납니다. 만약 3 번 스쿼어에 도착하면 10 번 스쿼어로 순간이동합니다. 만약 13 번 스쿼어에 도착하면 9 번 스쿼어로 이동하며, 그 반대도 마찬가지입니다.

15 (end)	14	13	12
8	9	10	11
7	6	5	4
0 (start)	1	2	3

Figure 4. 양자 “Chutes”와 “Ladder”가 있는 본. 3 번과 10 번, 9 번과 13 번 사각형에 가면 화살표를 따라 상태가 바뀐다.

힌트 1: "미끄럼틀"과 "사다리"를 구현할 때 두 상태를 맞바꾸는 교환 행렬을 고려해보세요. 또한 여러 사각형을 이동하는 "다중 이동" 연산자로 이것을 구현하는 것도 고려할 수 있습니다. 이를 위해서는 Figure 3 의 도식을 일반화하는 것이 필요할 것입니다.

힌트 2: 유니터리 연산자의 곱은 유니터리이므로, Task 3 에서 얻은 솔루션과 조합할 수 있습니다.

Task 5: 양자 Chutes and Ladders 게임에서 측정의 역할에 대해 논해보세요. 턴과 턴 사이에 상태를 측정하면 어떻게 될까요? 측정하지 않으면 어떻게 되나요? 고전 게임의 "메모리 없는" 특성에 대한 양자적 비유는 무엇입니까?

Task 6: 이제 작업 3 에서 했던 보행과 유사하게 Chute and Ladders "양자 연산자"를 사용하여 게임을 시뮬레이션합니다. 10 걸음 후에 각 사각형에 있을 확률을 비교해 보세요. 이것이 작업 3 의 "Chute and Ladders 연산자"가 없는 양자 게임과 어떻게 비교됩니까?