응용물리연구실심화실습3 (PHY3075)

2024년 10월 1주차

응용물리학과 2022006971 이민성

Multiple systems

번역

- 소개

  이 수업의 초점은 여러 시스템이 고려될 때 양자 정보의 기초에 관한 것입니다. 이러한 상황은 정보 처리, 특히 고전적 및 양자적 정보 처리의 맥락에서 자연스럽게 발생합니다. 대형 정보 전달 시스템은 종종 비트나 큐비트와 같은 작은 시스템들의 집합을 사용하여 가장 쉽게 구성됩니다.

  이번 수업에서 중요한 간단한 개념은 여러 시스템을 함께 하나의 복합 시스템으로 취급할 수 있다는 것입니다. 이 개념은 이전 수업에서 다룬 논의가 그대로 적용되는 중요한 개념입니다. 실제로 이 아이디어는 양자 상태, 측정 및 연산이 여러 시스템에서 어떻게 작동하는지를 설명하는 데 매우 직접적으로 이어집니다.

  그러나 여러 양자 시스템을 이해하는 데에는 단순히 그것들을 하나의 시스템으로 집합적으로 취급하는 것 이상이 있습니다. 예를 들어, 특정 양자 상태에 있는 여러 양자 시스템을 가지고 있을 때, 그 중 하나(또는 일부 시스템의 하위 집합)만을 선택하여 측정할 수 있습니다. 일반적으로 이는 나머지 시스템들의 상태에 영향을 미치며, 양자 알고리즘과 프로토콜을 분석할 때 이를 정확히 이해하는 것이 중요합니다. 여러 시스템 사이의 상관관계, 특히 얽힘(entanglement)이라는 특정 유형의 상관관계에 대한 이해도 양자 정보와 계산에서 중요합니다.

- 고전 정보 (Classical information)

  이전 수업과 마찬가지로, 우리는 고전 정보에 대한 논의로 시작할 것입니다. 다시 한 번 말하자면, 확률적 설명과 양자적 설명은 수학적으로 유사하며, 고전 정보에서 수학이 어떻게 작동하는지를 인식하는 것은 양자 정보가 왜 그렇게 설명되는지를 이해하는 데 도움이 됩니다.

  데카르트 곱(Cartesian product)을 통한 고전적 상태

  우리는 매우 기초적인 수준에서 시작하여 여러 시스템의 고전적 상태를 살펴볼 것입니다. 간단하게 하기 위해 두 시스템만을 논의하고, 이후에 이를 두 개 이상의 시스템으로 일반화할 것입니다.

  정확히 말하면, X가 고전적 상태 집합이 Σ인 시스템이라고 가정하고, Y가 고전적 상태 집합이 Γ인 두 번째 시스템이라고 가정하겠습니다. 이전 수업에서 이러한 집합을 고전적 상태 집합이라고 지칭했기 때문에 Σ와 Γ는 유한하고 비어 있지 않다는 가정이 있습니다. Σ = Γ일 수도 있지만, 반드시 그렇지는 않으며, 명확성을 위해 서로 다른 이름을 사용하는 것이 유익합니다.

  이제 두 시스템 X와 Y가 나란히 배치되어 있으며, X는 왼쪽에, Y는 오른쪽에 있다고 상상해 보세요. 우리가 선택하면, 이 두 시스템을 마치 하나의 단일 시스템처럼 볼 수 있으며, 이를 (X, Y) 또는 XY로 나타낼 수 있습니다.

  이 복합 시스템 (X, Y)에 대해 물을 수 있는 자연스러운 질문은 “그 고전적 상태는 무엇인가?”라는 것입니다.

  그 답은 (X, Y)의 고전적 상태 집합은 Σ와 Γ의 데카르트 곱이라는 것입니다. 이는 다음과 같이 정의됩니다.

  $$\Sigma \times \Gamma = \Big\{(a, b) \,:\, a \in \Sigma \text{ and } b \in \Gamma\Big\}.$$

  간단히 말해, 데카르트 곱은 한 집합의 원소와 다른 집합의 원소를 함께 하나의 단일 원소로 취급하는 수학적 개념을 정확히 포착한 것입니다.

이 경우, (X, Y)가 고전적 상태 $(a, b) \in \Sigma \times \Gamma$에 있다고 말하는 것은 X가 Σ의 고전적 상태 a에 있고 Y가 Γ의 고전적 상태 b에 있다는 의미입니다.

해당 경우에서, (X, Y)가 고전적 상태 $(a, b) \in \Sigma \times \Gamma$에 있다는 것은 X가 Σ의 고전적 상태 a에 있고 Y가 Γ의 고전적 상태 b에 있다는 것을 의미합니다. 그리고 X의 고전적 상태가 Σ의 a이고 Y의 고전적 상태가 Γ의 b라면, 공동 시스템 (X, Y)의 고전적 상태는 (a, b)입니다.

두 개 이상의 시스템의 경우, 상황은 자연스럽게 일반화됩니다. $X_1, \ldots, X_n$이 각각 고전적 상태 집합 $\Sigma_1, \ldots, \Sigma_n$을 가지는 시스템이라고 가정하면, 양의 정수 n에 대해, n-튜플 $(X_1, \ldots, X_n)$의 고전적 상태 집합을 하나의 단일 시스템으로 보았을 때, 이는 다음과 같은 데카르트 곱입니다.

$$\Sigma_1 \times \cdots \times \Sigma_n = \big\{(a_1, \ldots, a_n) \,:\, a_1 \in \Sigma_1, \,\ldots,\, a_n \in \Sigma_n\big\}.$$

우리는 시스템에 어떤 이름을 붙이든 상관없으며, 원하는 대로 그 순서를 정할 자유가 있습니다. 특히, n개의 시스템이 위와 같다면, 이를 $X_{n-1}, \ldots, X_0$로 이름을 붙이고 그 순서를 이와 같이 정할 수 있습니다. 이는 $(X_{n-1}, \ldots, X_0)$를 의미합니다. 고전적 상태와 고전적 상태 집합에 대해 동일한 패턴을 모방하여, 복합 시스템의 고전적 상태 $(a_{n-1}, \ldots, a_0) \in \Sigma_{n-1} \times \ldots \times \Sigma_0$에 대해 언급할 수 있습니다. 실제로, 이것은 큐비트 명명에 대해 Qiskit에서 사용되는 표준 순서 규칙이며, 우리는 다음 수업에서 양자 회로로 초점을 옮길 때 다시 이 내용을 다룰 것입니다.

문자열로 상태 표현하기

고전적 상태 $(a_1, \ldots, a_n)$를 간결함을 위해 문자열 $a_1 \ldots a_n$로 쓰는 것이 종종 편리합니다. 특히, 고전적 상태 집합 $\Sigma_1, \ldots, \Sigma_n$가 기호나 문자의 집합과 연관된 경우에 매우 일반적입니다.

실제로 문자열의 개념은 컴퓨터 과학에서 근본적으로 중요한 개념이며, 이는 데카르트 곱을 통해 수학적으로 형식화됩니다. 알파벳(alphabet)이라는 용어는 문자열을 형성하기 위해 사용되는 기호 집합을 지칭하는 데 흔히 사용되지만, 알파벳의 수학적 정의는 고전적 상태 집합의 정의와 정확히 동일합니다. 즉, 이는 유한하고 비어 있지 않은 집합입니다.

예를 들어, $X_1, \ldots, X_{10}$이 비트라고 가정하면, 이러한 시스템의 고전적 상태 집합은 모두 동일합니다.

$$\Sigma_1 = \Sigma_2 = \cdots = \Sigma_{10} = \{0, 1\}$$

(집합 {0, 1}은 일반적으로 이진 알파벳(binary alphabet)이라고 불립니다.) 이때, 합성 시스템 $(X_1, \ldots, X_{10})$의 고전적 상태는 총 $2^{10} = 1024$가지가 있으며, 이는 집합

$$\Sigma_1 \times \Sigma_2 \times \cdots \times \Sigma_{10} = \{0, 1\}^{10}.$$

의 원소들입니다.

이 고전적 상태들을 문자열로 쓰면 다음과 같이 보입니다:

$$\begin{gathered} 0000000000 \\ 0000000001 \\ 0000000010 \\ 0000000011 \\ 0000000100 \\ \vdots \\ 1111111111 \end{gathered}$$

예를 들어, 고전적 상태 0001010000의 경우, X_4와 X_6이 상태 1에 있고 나머지 시스템은 모두 상태 0에 있다는 것을 알 수 있습니다.

확률적 상태 (Probabilistic states)

이전 수업에서 다뤘던 것처럼, **확률적 상태**는 시스템의 각 고전적 상태에 확률을 연관시킵니다. 따라서, 여러 시스템의 확률적 상태는 — 마치 하나의 단일 시스템처럼 집합적으로 보았을 때 — 개별 시스템들의 고전적 상태 집합의 데카르트 곱의 각 원소에 확률을 연관시킵니다.

예를 들어, X와 Y가 둘 다 비트라고 가정하면, 각각의 고전적 상태 집합은 $\Sigma = \{0, 1\}$ 그리고 $\Gamma = \{0, 1\}$입니다. 다음은 (X, Y) 쌍의 확률적 상태입니다:

$$\begin{aligned}
\Pr\big((\mathsf{X},\mathsf{Y}) = (0,0)\big) &= 1/2 \\
\Pr\big((\mathsf{X},\mathsf{Y}) = (0,1)\big) &= 0 \\
\Pr\big((\mathsf{X},\mathsf{Y}) = (1,0)\big) &= 0 \\
\Pr\big((\mathsf{X},\mathsf{Y}) = (1,1)\big) &= 1/2
\end{aligned}$$

이 확률적 상태는 X와 Y가 둘 다 랜덤한 비트인 상태로, 각각이 1/2의 확률로 0이거나 1입니다. 그러나 두 비트의 고전적 상태는 항상 일치합니다. 이것은 이 시스템들 간의 상관관계(correlation)의 한 예입니다.

데카르트 곱 상태 집합의 순서 정하기

시스템의 확률적 상태는 확률 벡터로 표현됩니다. 이는 시스템의 기본 고전적 상태 집합과 대응하는 인덱스를 가진 열 벡터입니다.

이와 같은 상황은 여러 시스템에도 적용됩니다. 여러 시스템의 확률적 상태를 데카르트 곱으로 표현하려면, 곱의 원소들의 순서를 결정해야 합니다. X, Y 시스템의 고전적 상태 집합 Σ, Γ가 이미 순서가 정해져 있다고 가정하면, 이를 위한 간단한 규칙이 있습니다: **알파벳 순서**입니다. 보다 정확히 말하면, 각 n-튜플(또는, 동등하게 각 문자열의 기호) 안의 항목들은 왼쪽에서 오른쪽으로 갈수록 중요성이 감소하는 순서로 정렬됩니다.

예를 들어, 이 규칙에 따르면, 데카르트 곱 $\{1, 2, 3\} \times \{0, 1\}$의 순서는 다음과 같습니다:

(1, 0), (1, 1), (2, 0), (2, 1), (3, 0), (3, 1).

n-튜플이 문자열로 쓰이고 이러한 방식으로 정렬되면, 우리는 다음과 같은 익숙한 패턴을 관찰할 수 있습니다. 예를 들어, $\{0, 1\} \times \{0, 1\}$는 00, 01, 10, 11로 정렬되고, $\{0, 1\}^{10}$은 앞서 제안한 대로 정렬됩니다. 또한 $\{0, 1, \ldots, 9\} \times \{0, 1, \ldots, 9\}$는 00부터 99까지 숫자들로 정렬됩니다. 오늘날의 10진수 체계가 동일한 알파벳 순서를 사용하고 있다는 것을 알아차릴 수 있습니다. 물론, 여기서 "알파벳"이라는 단어는 숫자뿐만 아니라 모든 기호 집합을 포함하는 더 넓은 의미를 가집니다.

이제 두 비트 시스템의 예로 돌아가서, 확률적 상태는 기본적인 고전적 상태 집합의 열거된 항목에 따라 확률 벡터로 표현됩니다 (이 항목들은 위에서 명시적으로 동일한 순서로 배열된 경우).

$$\begin{pmatrix} \frac{1}{2} \\ 0 \\ 0 \\ \frac{1}{2} \end{pmatrix} \begin{array}{l} \leftarrow \text{probability associated with state 00} \\ \leftarrow \text{probability associated with state 01} \\ \leftarrow \text{probability associated with state 10} \\ \leftarrow \text{probability associated with state 11} \end{array}$$

두 시스템의 독립성

두 시스템의 확률적 상태 중 특별한 유형은 두 시스템이 **독립적**일 때 발생합니다. 직관적으로 말하자면, 두 시스템이 독립적이라는 것은 한 시스템의 고전적 상태를 알게 되어도 다른 시스템과 관련된 확률에 아무런 영향을

미치지 않는다는 것을 의미합니다. 즉, 한 시스템이 어떤 고전적 상태에 있는지를 아는 것이 다른 시스템의 고전적 상태에 대해 전혀 정보를 제공하지 않는 경우입니다.

이 개념을 정확하게 정의하기 위해, X와 Y가 각각 고전적 상태 집합 Σ와 Γ를 가지는 시스템이라고 다시 한 번 가정합시다. 이러한 시스템들의 주어진 확률적 상태에 대해, 두 시스템은 다음 조건을 만족할 때 **독립적**이라고 합니다:

$$\Pr((\mathsf{X},\mathsf{Y}) = (a,b)) = \Pr(\mathsf{X} = a)\Pr(\mathsf{Y} = b)$$

이는 모든 a ∈ Σ와 b ∈ Γ에 대해 성립합니다.

이 조건을 확률 벡터의 용어로 표현하기 위해, (X, Y)의 주어진 확률적 상태가 디랙 표기법으로 다음과 같이 표현되는 확률 벡터로 설명된다고 가정합니다:

$$\sum_{(a,b)\in\Sigma\times\Gamma} p_{ab}|ab\rangle.$$

독립성의 조건 (2)은 다음과 같은 두 개의 확률 벡터가 존재하는 것과 동등합니다:

$$|\phi\rangle = \sum_{a\in\Sigma} q_a|a\rangle \quad \text{and} \quad |\psi\rangle = \sum_{b\in\Gamma} r_b|b\rangle,$$

이는 각각 X와 Y의 고전적 상태에 연관된 확률들을 나타내며, 이때

$$p_{ab} = q_a r_b$$

는 모든 a ∈ Σ와 b ∈ Γ에 대해 성립합니다.

예를 들어, 다음 벡터로 표현된 (X, Y)의 두 비트 쌍의 확률적 상태:

$$\frac{1}{6}|00\rangle + \frac{1}{12}|01\rangle + \frac{1}{2}|10\rangle + \frac{1}{4}|11\rangle$$

는 X와 Y가 독립적인 경우입니다. 구체적으로, 독립성에 필요한 조건은 이 확률 벡터에 대해 성립합니다 :

$$|\phi\rangle = \frac{1}{4}|0\rangle + \frac{3}{4}|1\rangle \quad \text{and} \quad |\psi\rangle = \frac{2}{3}|0\rangle + \frac{1}{3}|1\rangle.$$

예를 들어, 00 항목에 맞추려면, 우리는 1/6 = 1/4 * 2/3이 되어야 하고, 실제로 그러합니다. 다른 항목들도 유사한 방식으로 검증할 수 있습니다.

반면에, 확률적 상태 ρ, 이를 다음과 같이 쓸 수 있습니다:

$$\frac{1}{2}|00\rangle + \frac{1}{2}|11\rangle,$$

이는 X와 Y 사이의 독립성을 나타내지 않습니다. 이를 설명하는 간단한 방법은 다음과 같습니다.

만약 확률 벡터 ∣ϕ⟩와 ∣ψ⟩가 존재한다고 가정하고, 식 (3)과 같이 모든 a와 b에 대해 조건 (4)이 성립한다고 가정하면, 다음과 같은 식이 성립해야 합니다:

$$q_0 r_1 = \Pr\big((\mathsf{X}, \mathsf{Y}) = (0,1)\big) = 0.$$

이는 q_0 = 0이거나 r_1 = 0이어야 한다는 것을 의미합니다. 두 값이 모두 0이 아닌 경우에는 곱 q_0 r_1이 0이 될 수 없기 때문입니다. 이로 인해 q_0 r_0 = 0(즉, q_0 = 0) 또는 q_1 r_1 = 0 (즉, r_1 = 0)이라는 결론에 이르게 됩니다. 그러나 두 가지 식 모두 참일 수 없다는 것을 알 수 있습니다. 왜냐하면 우리는 q_0 r_0 = 1/2이고 q_1 r_1 = 1/2이어야 하기 때문입니다. 따라서 독립성에 필요한 성질을 만족시키는 ∣ϕ⟩와 ∣ψ⟩라는 벡터는 존재하지 않습니다.

두 시스템 간의 독립성을 정의한 이후, 우리는 이제 독립성의 부재를 상관관계의 정확한 정의로 이해할 수 있습니다. 예를 들어, 위에서 주어진 벡터 (5)로 표현된 확률적 상태는 독립적이지 않으며, 따라서 두 비트는 본질적으로 상관관계가 있습니다.

벡터의 텐서곱 (Tensor products of vectors)

앞서 설명한 독립성 조건은 **텐서곱**이라는 개념을 통해 더 간결하게 표현될 수 있습니다. 이 개념은 매우 일반적인 것이며 다양한 수학적 구조에 추상적으로 적용될 수 있지만, 여기서는 간단하고 구체적인 용어로 정의할 수 있습니다. 두 벡터가 주어졌을 때

$$|\phi\rangle = \sum_{a\in\Sigma} \alpha_a |a\rangle \quad \text{and} \quad |\psi\rangle = \sum_{b\in\Gamma} \beta_b |b\rangle,$$

그 텐서곱 ∣ϕ⟩⊗∣ψ⟩는 합성 상태 집합 Σ × Γ 위의 새로운 벡터로 정의되며, 다음과 같습니다:

$$|\phi\rangle \otimes |\psi\rangle = \sum_{(a,b)\in\Sigma\times\Gamma} \alpha_a \beta_b |ab\rangle.$$

동등하게, 벡터 ∣π⟩=∣ϕ⟩⊗∣ψ⟩는 다음 식으로 정의됩니다:

$$\langle ab|\pi\rangle = \langle a|\phi\rangle\langle b|\psi\rangle$$

이는 모든 a ∈ Σ 및 b ∈ Γ에 대해 성립합니다.

우리는 이제 독립성 조건을 재구성하여 합성 시스템 (X, Y)의 확률 벡터 ∣π⟩가 각 하위 시스템 X와 Y에서의 확률 벡터 ∣ϕ⟩ 및 ∣ψ⟩의 텐서곱으로 표현될 수 있어야 한다고 요구할 수 있습니다. 이 경우 ∣π⟩는 **곱 상태 (product state)** 또는 곱 벡터(product vector)라고 불립니다.

우리는 종종 텐서곱 기호(⊗)를 생략하고, ∣ϕ⟩∣ψ⟩와 같이 쓰는 방식으로 텐서곱을 나타냅니다. 이 규칙은 이 문맥에서 텐서곱이 두 벡터의 곱을 취하는 가장 자연스러운 방식이라는 개념을 포착합니다. 덜 일반적이지만, ∣ϕ⟩⊗∣ψ⟩라는 표기도 때때로 사용됩니다.

데카르트 곱의 항목을 정렬하는 알파벳 순서를 사용할 때, 우리는 두 열 벡터의 텐서곱에 대한 다음 사양을 얻습니다.

$$\begin{pmatrix}\alpha_1\\ \vdots\\ \alpha_m\end{pmatrix}\otimes\begin{pmatrix}\beta_1\\ \vdots\\ \beta_k\end{pmatrix}=\begin{pmatrix}\alpha_1\beta_1\\ \vdots\\ \alpha_1\beta_k\\ \alpha_2\beta_1\\ \vdots\\ \alpha_2\beta_k\\ \vdots\\ \alpha_m\beta_1\\ \vdots\\ \alpha_m\beta_k\end{pmatrix}$$

중요한 참고 사항으로, 우리는 표준 기저 벡터의 텐서곱에 대한 다음과 같은 표현을 관찰할 수 있습니다:

∣a⟩⊗∣b⟩=∣ab⟩.

또는, (a,b)(a, b)(a,b)를 문자열 대신 순서쌍으로 쓰면 다음과 같이 쓸 수 있습니다:

∣a⟩⊗∣b⟩=∣(a,b)⟩,

그러나 더 일반적으로는 다음과 같이 씁니다:

∣a⟩⊗∣b⟩=∣a,b⟩,

이는 명확성을 더하거나 모호함을 제거하지 않는 한, 괄호를 제거하는 수학의 관습을 따릅니다.

두 벡터의 텐서곱은 중요한 성질을 가지고 있는데, 이는 쌍선형(bilinear)이라는 것입니다. 즉, 다른 인수를 고정하면 각각의 두 인수에 대해 선형(linear)입니다. 이 성질은 다음 방정식들로 표현될 수 있습니다:

1. 첫 번째 인수에서의 선형성:

$$\big(|\phi_1\rangle+|\phi_2\rangle\big)\otimes|\psi\rangle=|\phi_1\rangle\otimes|\psi\rangle+|\phi_2\rangle\otimes|\psi\rangle$$
$$\big(\alpha|\phi\rangle\big)\otimes|\psi\rangle=\alpha\big(|\phi\rangle\otimes|\psi\rangle\big)$$

2. 두 번째 인수에서의 선형성:

$$|\phi\rangle\otimes\big(|\psi_1\rangle+|\psi_2\rangle\big)=|\phi\rangle\otimes|\psi_1\rangle+|\phi\rangle\otimes|\psi_2\rangle$$
$$|\phi\rangle\otimes\big(\alpha|\psi\rangle\big)=\alpha\big(|\phi\rangle\otimes|\psi\rangle\big)$$

각 쌍의 두 번째 방정식을 고려해 보면, 우리는 스칼라가 텐서곱 내에서 "자유롭게 움직인다"는 것을 알 수 있습니다:

$$\big(\alpha|\phi\rangle\big)\otimes|\psi\rangle=|\phi\rangle\otimes\big(\alpha|\psi\rangle\big)=\alpha\big(|\phi\rangle\otimes|\psi\rangle\big).$$

따라서 단순히 α∣ϕ⟩⊗∣ψ⟩를 쓰거나, 대안적으로 α(∣ϕ⟩⊗∣ψ⟩)라고 써도 모호함 없이 동일한 벡터를 나타냅니다.

세 개 이상의 시스템에서의 독립성과 텐서곱

독립성과 텐서곱의 개념은 세 개 이상의 시스템으로 쉽게 일반화됩니다. 만약 X_1, ..., X_n이 각각 고전적 상태 집합 Σ_1, ..., Σ_n을 가지는 시스템이라면, 합성 시스템 (X_1, ..., X_n)의 확률적 상태는 연관된 확률 벡터가 다음과 같은 형태일 때 곱 상태(product state)라고 합니다:

$$|\psi\rangle = |\phi_1\rangle \otimes \cdots \otimes |\phi_n\rangle$$

여기서 $|\phi1\rangle,\cdots,|\phi n\rangle$은 X_1, ..., X_n의 확률적 상태를 나타내는 확률 벡터입니다.

여기서 텐서곱의 정의는 자연스럽게 일반화됩니다. 즉, 벡터

$$|\psi\rangle = |\phi_1\rangle \otimes \cdots \otimes |\phi_n\rangle$$

은 다음 식으로 정의됩니다:

$$\langle a_1 \cdots a_n|\psi\rangle = \langle a_1|\phi_1\rangle \cdots \langle a_n|\phi_n\rangle$$

이는 모든 $a1\in\Sigma1$, ..., $an\in\Sigma n$에 대해 성립합니다. 세 개 이상의 벡터의 텐서곱을 정의하는 또 다른 방법은 두 벡터의 텐서곱으로 재귀적으로 정의하는 것입니다:

$$|\phi_1\rangle \otimes \cdots \otimes |\phi_n\rangle = \big(|\phi_1\rangle \otimes \cdots \otimes |\phi_{n-1}\rangle\big) \otimes |\phi_n\rangle,$$

여기서 $n\geq3$입니다.

두 벡터의 텐서곱과 유사하게, 세 개 이상의 벡터의 텐서곱은 각 인수에 대해 선형입니다. 이 경우, 세 개 이상의 벡터의 텐서곱을 다중선형(multilinear)이라고 합니다.

두 시스템의 경우와 마찬가지로, 시스템 X_1, ..., X_n이 독립적일 때 시스템이 곱 상태에 있다고 말할 수 있습니다. 그러나 상호 독립(mutually independent)이라는 용어가 더 정확합니다. 세 개 이상의 시스템의 독립성에는 쌍별 독립(pairwise independence)과 같은 다른 개념이 존재하지만, 이 내용은 지금 다루지 않겠습니다.

표준 기저 벡터의 텐서곱에 관한 이전의 관찰을 일반화하면, 임의의 양의 정수 n과 임의의 고전적 상태 $a1,\cdots,an$에 대해 우리는 다음을 얻습니다:

$$|a_1\rangle \otimes \cdots \otimes |a_n\rangle = |a_1 \cdots a_n\rangle = |a_1, \ldots, a_n\rangle.$$

정리

- 고전 정보

  이전 수업과 마찬가지로, 우리는 고전 정보에 대한 논의로 시작할 것입니다. 확률적 설명과 양자적 설명은 수학적으로 유사하며, 고전 정보에서 수학이 어떻게 작동하는지를 인식하는 것은 양자 정보가 왜 그렇게 설명되는지를 이해하는 데 도움이 됩니다.

- 데카르트 곱을 통한 고전적 상태

  우리는 기본적인 수준에서 여러 시스템의 고전적 상태를 살펴볼 것입니다. 예를 들어 X와 Y가 각각 고전적 상태 집합을 가진 시스템이라고 할 때, 두 시스템의 복합 상태는 각각의 고전적 상태 집합의 데카르트 곱으로 정의됩니다. 이를 통해 여러 시스템을 하나의 시스템으로 볼 수 있습니다.

- 문자열로 상태 표현하기

고전적 상태는 간결하게 문자열로 표현할 수 있습니다. 예를 들어 비트 시스템의 상태 집합은 이진 알파벳으로 구성될 수 있으며, 이는 문자열로 표현됩니다. 이렇게 표현된 고전적 상태는 각 시스템의 상태를 직관적으로 이해하는 데 도움이 됩니다.

- 확률적 상태

  확률적 상태는 각 고전적 상태에 확률을 연관시킵니다. 예를 들어, X와 Y가 각각 비트일 때, 두 비트의 확률적 상태는 각 고전적 상태와 연관된 확률로 표현됩니다. 이는 시스템 간의 상관관계(correlation)를 분석하는 데 중요한 개념입니다.

- 데카르트 곱 상태 집합의 순서 정하기

  확률적 상태는 확률 벡터로 표현되며, 이러한 벡터의 순서는 데카르트 곱 상태 집합의 원소에 대한 알파벳 순서에 따라 정해집니다. 이러한 순서는 여러 시스템 간의 상태를 이해하는 데 중요한 역할을 합니다.

- 두 시스템의 독립성

  두 시스템이 독립적이라는 것은 한 시스템의 상태를 알았을 때 다른 시스템의 상태에 대해 아무런 정보도 제공하지 않는 경우를 말합니다. 이는 확률 벡터 간의 관계로 정의될 수 있으며, 텐서곱을 통해 간결하게 표현됩니다. 독립성을 만족하는 시스템들은 확률적으로 독립적입니다.

- 벡터의 텐서곱

  독립성 조건은 두 벡터의 텐서곱을 통해 더 간결하게 표현될 수 있습니다. 텐서곱은 두 시스템의 상태를 하나로 결합하는 방식으로, 각각의 시스템이 독립적으로 행동할 때 성립합니다. 텐서곱은 각 인수에 대해 선형적이며, 이는 시스템 간의 독립성을 정의하는 데 중요한 역할을 합니다.

- 세 개 이상의 시스템에서의 독립성과 텐서곱

  독립성과 텐서곱은 세 개 이상의 시스템으로 일반화될 수 있습니다. 여러 시스템이 독립적일 때, 이들은 상호 독립(mutually independent)이라고 불리며, 텐서곱을 통해 표현됩니다. 여러 시스템의 독립성은 다중선형(multilinear) 특성을 가지며, 이를 통해 다양한 양자 정보 처리가 가능합니다.