RESEARCH NOTE 02 · MATHEMATICS & AI

P vs NP: 빨리 검증할 수 있다면 빨리 풀 수도 있는가?

밀레니엄 미해결 문제 P vs NP의 정확한 계산복잡도 정의, NP-완전성과 3-SAT, 증명 기준 및 실험실을 STARGATE EDU 연구 콘텐츠로 제공합니다.

초판 2026.11.22 · 원문 대조·내용 개정 2026.11.23 (KST)

원문 대조 범위 · 2026.11.23

월간 통계 기준월은 비적용입니다. 원문 발표·갱신일과 이번 수집·확인일은 출처 기록에서 구분합니다. 이 페이지의 연구 과제는 앞으로 수행할 계획이며, 수행 증거가 있는 결과와 별도로 읽어 주세요.

1. P와 NP의 엄밀한 뜻

P · POLYNOMIAL TIME

다항시간에 결정할 수 있는 문제

결정문제에 대해 결정론적 계산으로 입력 길이 n의 다항식 시간 $O(n^k)$ 이내에 예/아니오를 판정할 수 있는 언어들의 집합입니다. 여기서 k는 해당 알고리즘에 고정된 상수입니다.

NP · VERIFY IN POLYNOMIAL TIME

증명서를 다항시간에 검증할 수 있는 문제

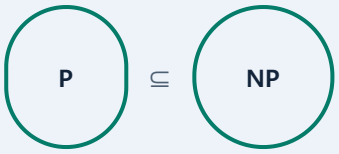
예 답에 대한 다항 길이 증명서(해 후보)를 받아 결정론적으로 다항시간에 확인할 수 있는 결정문제들의 집합입니다. NP는 'not polynomial'의 약자가 아닙니다. 예인 입력에는 적어도 하나의 유효한 증명서가 있고, 아니오인 입력은 어떤 증명서도 검증기를 통과하지 않아야 합니다. [2, 1~2쪽]

$$P \subseteq NP \quad / \quad \text{미해결 질문: } P = NP \ ?$$

핵심은 **최악 사례에 대한 모든 입력**에 적용되는 다항시간 알고리즘이 존재하는지입니다. 실제 입력 몇 개를 빠르게 푸는 것만으로는 결론을 낼 수 없습니다. [1]

2. 3-SAT와 NP-완전성

3-SAT는 각 절(clause)에 리터럴이 최대 3개 들어가는 논리식의 만족 가능 여부를 묻는 문제로, 대표적인 NP-완전 문제입니다. Cook의 정확히 3개 리터럴 정의와 여기서 쓰는 최대 3개 정의는 리터럴 반복을 허용하면 같은 만족 가능성 문제로 옮길 수 있습니다. [2, 4~5쪽] NP-완전 문제는 NP에 속하면서 NP의 모든 문제가 다항시간 환원되는 문제입니다.



NP-완전 문제는 NP의 모든 문제를 다항시간 환원으로 포괄

P = NP를 증명하려면

3-SAT처럼 하나의 NP-완전 문제에 대해 모든 입력에 작동하는 **다항시간 결정 알고리즘·정확성·최악 시간 상한 증명**을 제시하면 충분합니다. 해당 환원으로 모든 NP 문제가 P에 속하게 됩니다.

P ≠ NP를 증명하려면

하나의 NP-완전 결정문제(예: 3-SAT)가 어떠한 결정론적 다항시간 알고리즘으로도 해결되지 않음을 **일반적으로** 증명하면 충분합니다. 특정 탐색 알고리즘이 느리다는 분석은 부족합니다.

상태: 공식 미해결

Clay Mathematics Institute는 2026년 11월 23일 조회 당시 P vs NP를 **Unsolved**로 분류하고 있습니다. 아래 데모는 증명서 검증 절차를 배우는 실험이며 난제에 대한 증명이 아닙니다. [1]

---

3. 작은 SAT 문제 실험

다음 논리식에서 x, y, z의 참·거짓 값을 설정하고 '검증'을 누르세요. 조건을 만족하는지 판정은 빠르지만, 검증과 후보 열거의 동작을 비교합니다. 다만 **완전탐색이 유일한 방법이라는 증거는 아닙니다.**

실험 범위: 아래 식은 각 절이 2개 리터럴인 2-CNF입니다. 2-SAT는 다항시간에 풀 수 있으므로 이 예제는 NP-완전 문제군의 어려움을 보여 주는 벤치마크가 아닙니다. 8개 후보 열거는 이 작은 식에 대한 구현 선택이며 일반적인 하한 증명이 아닙니다.

$$(x \vee y) \wedge (\neg x \vee z) \wedge (\neg y \vee \neg z)$$

후보 해 (증명서)

- [ ] x = 참
- [x] y = 참
- [ ] z = 참

만족합니다. 현재 대입은 유효한 증명서입니다.

3개 변수이므로 후보는 $2^3 = 8$개입니다. n개 변수의 단순 완전탐색은 2^n개 후보를 검사하지만, 이 사실만으로 P ≠ NP를 증명할 수 없습니다.

▼ 검산한 8개 대입과 두 개의 만족 해 보기

만족하는 (x,y,z)는 (거짓, 참, 거짓)과 (참, 거짓, 참)입니다.

x	y	z	첫째 절	둘째 절	셋째 절	전체 식
거짓	거짓	거짓	거짓	참	참	거짓
거짓	거짓	참	거짓	참	참	거짓
거짓	참	거짓	참	참	참	참
거짓	참	참	참	참	거짓	거짓
참	거짓	거짓	참	거짓	참	거짓
참	거짓	참	참	참	참	참
참	참	거짓	참	거짓	참	거짓
참	참	참	참	참	거짓	거짓

4. STARGATE 연구 과제

1. **문제·환원 정확성:** 3-SAT→그래프 문제 등 다항시간 환원에서 입력 크기와 계산 비용을 명시적으로 증명합니다.
2. **알고리즘 기준선:** 3-SAT에 완전탐색, DPLL, CDCL(SAT 솔버)을 적용해 최악 사례·경험적 성능을 구분합니다.
3. **증명 장벽 검토:** 상대화(relativization), 자연스러운 증명(natural proofs), 대수화(algebrization) 같은 대표적 장벽을 검토합니다. 이들은 해당 성질을 가진 증명 전략의 한계이며 모든 수학적 접근이 불가능하다는 정리가 아닙니다. 자연스러운 증명 장벽의 일반 회로 하한 논의에는 강한 의사난수 생성기 존재와 같은 복잡도 가정이 붙습니다. [2][4]
4. **Lean 형식화:** 작은 환원 정리나 솔버의 부분 정합성을 정리증명기에 옮겨 검증합니다. 계산 실험과 전역적 하한 증명은 명확히 구별합니다.
5. **산업 적용 분석:** 조합최적화·일정 계획·회로 검증 및 암호 분야에서 최악 복잡도와 현실적인 근사·휴리스틱 성능을 분리해 평가합니다.

> **"해답을 발견하는 속도와, 그 해답이 옳음을 확인하는 속도는 같을까?"**
> **— STARGATE EDU · P vs NP 연구 질문**

DPLL·CDCL 성능 비교와 Lean 환원 형식화는 아직 연구 계획이며 이 페이지에서 구현·실측하지 않았습니다. P = NP가 증명되더라도 다항식 차수나 상수가 매우 크면 실제로 쓸 만한 알고리즘이 즉시 생긴다는 뜻은 아닙니다.

5. 공식 출처

[1] Clay Mathematics Institute — P vs NP (공식 정의 및 미해결 현황)

[2] Clay Mathematics Institute — Stephen Cook, P vs NP 공식 문제 설명

[3] Clay — Millennium Prize Rules

[4] Aaronson·Wigderson — Algebrization: A New Barrier in Complexity Theory (저자 제공 원문 1~2쪽)