The Clean Paper

WHAT THE PAPER SAYS · WHAT IT DOESN'T · WHY IT MATTERS

एक cryptographic proof अपने रहस्य कोइस तरह छिपाता है कि “गायब simulator” कीअनुपस्थिति साबित करनाकठिन होजाए

Zero-knowledge proofs किसीकथन कोबिनाwitness उजागर किए सबित करने देते हैं। Classical theory के अनुसार पूर्ण अर्थ में यह एक message, बिनाtrusted setup और perfect soundness के संभव नहीहै। Rahul Ilango काशोधपत्र उस असंभवताकोमिटानहीं वह लक्ष्य बदलताहै। वास्तविक simulator मौजूद होने की मांग के बजाय वह यह मांगताहै कि चुनौतीपूर्ण proof system यह कुशलतापूर्वक साबित न कर सके कि simulator मौजूद नहीहै। Proof complexity और cryptography से जुड़ीबड़ीमान्यताओं के तहत यह zero-knowledge के falsifiable, game-based परिणामोंको गुण-दर-गुण वापस पाने के लिए पर्याप्त है। यह इंटरनेट में सेंध लगादेने वालाprimitive नहीं बल्कि proof theory काएक तरीकाहै जिसमें गणितीय अप्रमाण्यताcryptographic cover बनतीहै।

Written by Cinzia Vaglio · Cross-checked by Vera Rubin · Edited by Michele Renda · Figures by Nova Vela · AI-assisted, human-reviewed

Paper: *Gödel in Cryptography: Effectively Zero-Knowledge Proofs for NP with No Interaction, No Setup, and Perfect Soundness*, Rahul Ilango, FOCS 2025 / IACR ePrint 2025/1296 · DOI: 10.1109/FOCS63196.2025.00058

चल यह सिद्ध करना नहीं है कि रहस्य छिपा हुआ है

शून्य-ज्ञान के सबसे सरल रूप से शुरू करें।

Alice Bob को यह विश्वास दिलाना चाहती है कि उसके पास puzzle का समाधान है। यदि वह समाधान भेज दे, तो Bob मान जाएगा, लेकिन puzzle खत्म हो जाएगा। उसे कुछ अजीब चाहिए: यह प्रमाण कि समाधान मौजूद है, समाधान बताए बिना।

यही शून्य-ज्ञान प्रमाण का वादा है। प्रूवर (Alice) वेरिफ़ायर (Bob) को विश्वास दिलाता है कि कथन सत्य है, लेकिन कथन के सत्य होने के अलावा कुछ reveal नहीं करता।

समस्या यह है कि इस वादे की कीमत है। सामान्य गणितीय प्रमाण में दो सुविधाजनक विशेषताएँ होती हैं। पहला, वह **एक संदेश** हो सकता है: आप प्रमाण लिखते हैं, देते हैं और बात खत्म। दूसरा, उसमें **पूर्ण soundness** हो सकती है: असत्य कथन का कोई वैध प्रमाण होता ही नहीं। शास्त्रीय impossibility परिणाम बताते हैं कि शून्य-ज्ञान में इन दोनों सुविधाओं को इस रूप में बनाए रखना संभव नहीं— न दोनों को एक साथ, और कुछ परिस्थितियों में न हर एक को अलग से।

पहला, शून्य-ज्ञान प्रमाण को बातचीत चाहिए। यदि Alice एक ही संदेश भेजती है, और पहले से कोई trusted सेटअप arranged नहीं है, तो शून्य-ज्ञान गारंटी collapse हो जाती है—चाहे बदले में आप सहजता जितनी भी कम करने को तैयार हों।

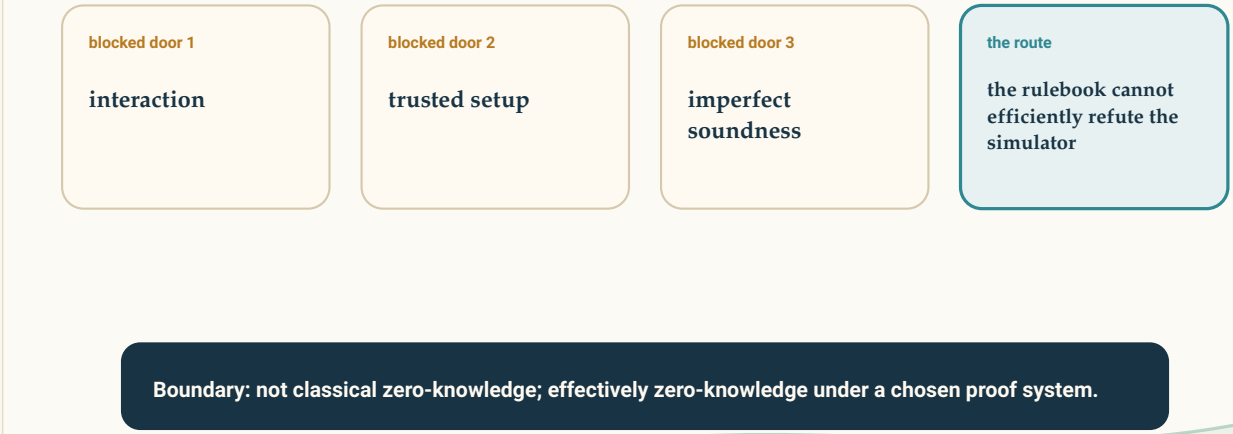
दूसरा, शून्य-ज्ञान प्रमाण को त्रुटि के लिए थोड़ी tolerance चाहिए। पूर्ण सहजता माँगना इंटरैक्शन को भी चुपचाप खत्म कर देता है: ऐसा वेरिफ़ायर जिसे उसके यादृच्छिक विकल्प कुछ भी हो कभी fool नहीं किया जा सकता, वे विकल्प पहले से सुधार ही कर सकता है—और वेरिफ़ायर predictable होते ही Alice सारे उत्तर एक संदेश में दे सकती है, वही मसला जो पहले ही टूट चुका था।

Rahul Ilango का शोधपत्र इस double दीवार के आसपास रास्ता खोजता है। दीवार को नकार कर नहीं, और असंभव परिस्थिति में शास्त्रीय शून्य-ज्ञान पैदा करने का दिखावा करके नहीं। कदम अधिक subtle है: “reveals nothing” का अर्थ कमजोर करना, लेकिन इस तरह कि cryptographers जिन सुरक्षागुण को वास्तव में परीक्षण कर सकते हैं वे बचे रहें।

परिणाम का नाम है **व्यवहारिक रूप से शून्य-ज्ञान**।

A proof without leaking the witness

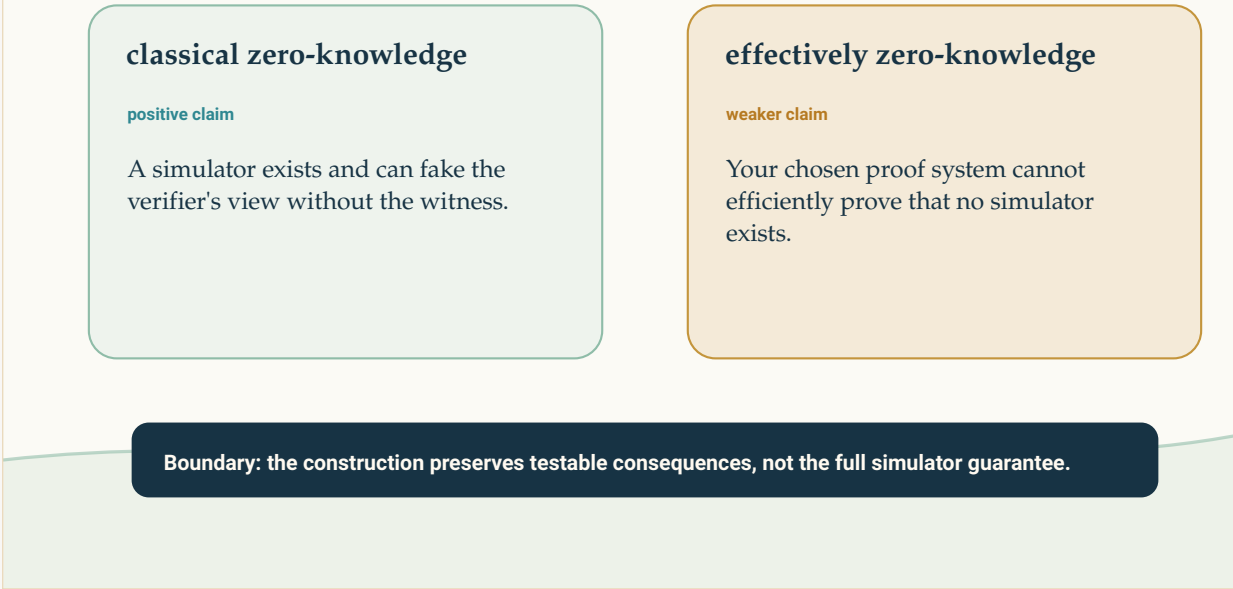
The paper keeps the ordinary-proof shape by weakening what privacy must prove.



शून्य-ज्ञान तीन दरवाज़ोंपर blocked है—इंटरैक्शन, trusted सेटअप और अपूर्ण सहंडनेस। Ilango कीनिर्माण एक अलग दरवाज़े से निकलतीहै: नियम-पुस्तकिासिम्युलेटर कोकुशलतापूर्वक खंडन करनानहींकर सकता।

Classical privacy vs effective privacy

The relaxation is the point: the paper changes what the privacy claim has to establish.



शास्त्रीय शून्य-ज्ञान पूछताहै कि सिम्युलेटर मौजूद है यानहीं "व्यावहारिक रूप से शून्य-ज्ञान" केवल यह पूछताहै कि चुनाहुआ नियम-पुस्तकिाकुशलतापूर्वक साबित कर सकताहै यानहींकि सिम्युलेटर मौजूद नहींहोसकता। यहीकमजोर सवाल निर्माण कोएक

संदेश, no सेटअप और पूर्ण सॉउंडनेस रखने देता है।

पुरावापरीक्षण: सिम्युलेटर मौजूद है

शून्य-ज्ञान को formalize करने का शास्त्रीय तरीका एक fictional helper इस्तेमाल करता है जिसे **सिम्युलेटर** कहते हैं।

विचार यह है: Jane की कल्पना करें, जिसे Alice का रहस्य **नहीं** पता। यदि Jane पूरी तरह अपने दम पर ऐसे प्रमाण उत्पन्न कर सकती है जो Bob को Alice से मिलने वाले प्रमाण जैसे ही दिखें, तो Alice के प्रमाण ने Bob को कुछ नया नहीं सिखाया। Jane Alice के रहस्य के बिना ही वही experience fake कर सकती थी।

इसलिए शास्त्रीय शून्य-ज्ञान actual सिम्युलेटर माँगता है। एक कुशल एल्गोरिदम सच में मौजूद होना चाहिए जो रहस्य—jargon में *विटनेस*—जाने बिना fake-looking प्रमाण बना सके। सुडोकू में विटनेस बस solved ग्रिड है।

यह definition शक्तिशाली है, लेकिन old impossibility ठीक यही कहती है। Intuition यह है: सचमुच non-interactive प्रमाण सिर्फ एक string है। Bob के पास string आने के बाद वह उसे किसी और को दिखा सकता है; उसे कथन दूसरों को सिद्ध करने की ability मिल गई, जो पहले ही "nothing" से अधिक लगती है। शास्त्रीय theorems इसी intuition को ऊपर की impossibilities में sharpen करती हैं।

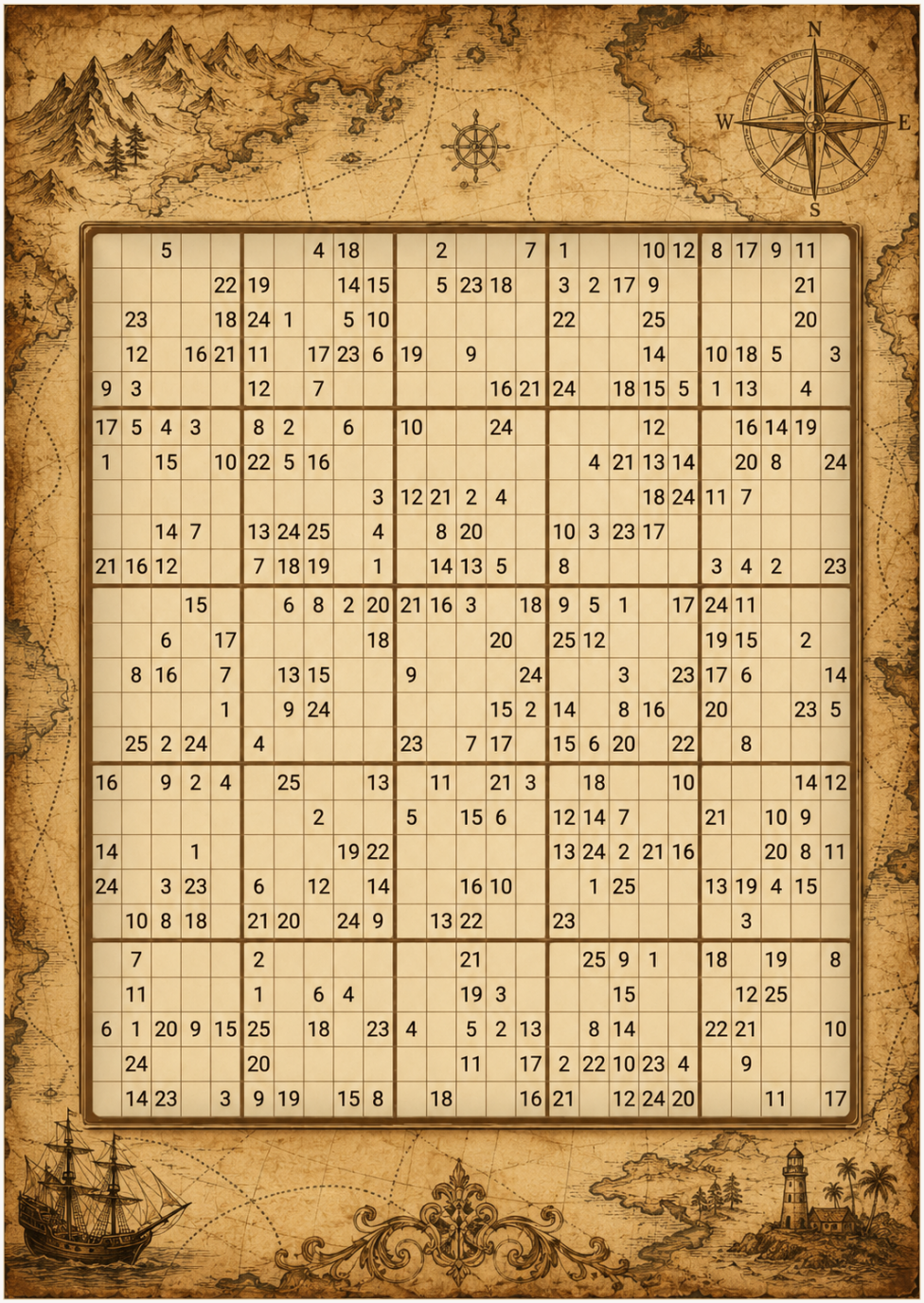
> **इस paper के लिए तीन अनिवार्य properties**
>
> शोधपत्र के title में तीन बाधाएँ हैं:
> **No इंटरैक्शन:** Alice एक प्रमाण string भेजती है। Back-और-forth प्रोटोकॉल नहीं।
> **No सेटअप:** Alice और Bob किसी trusted सामान्य संदर्भ string या पहले से arranged सार्वजनिक randomness पर निर्भर नहीं। "Non-interactive शून्य-ज्ञान" कहलाने वाले कई प्रणालियाँ फिर भी सेटअप इस्तेमाल करते हैं; यहाँ मतलब शून्य सेटअप है।
> **पूर्ण सॉउंडनेस:** असत्य कथन का कोई वैध प्रमाण नहीं। "लगभग कभी accept नहीं होगा" नहीं, वैध प्रमाण अस्तित्व में ही नहीं है।
> सामान्य written mathematics में यही तीन गुण होती हैं—और ऊपर जैसा समझाया, शास्त्रीय शून्य-ज्ञान इन्हें साथ नहीं रख सकता।

फर्क को MegaSudoku से महसूस करें

यह फर्क समझने का जानबूझकर simplified तरीका है।

उपमा के गंभीर हिस्से में सामान्य 9-by-9 सुडोकू न लें। वह बहुत छोटा और finite है: computer उसे हल कर सकता है या सिद्ध कर सकता है कि समाधान नहीं है। इसके बजाय **MegaSudoku(n)** puzzles की परिवार सोचें। Usual नियम को पैमाना करें: रोकना आकार *n* चुनें, $N = n^2$ रखें, फिर *N* by *N* ग्रिड बनाएँ जिसे *n* by *n* blocks में बाँटा गया हो और जिसमें *N* चिह्न हों। सामान्य सुडोकू सिर्फ tiny $n = 3$, $N = 9$ मामला है: 9-by-9 ग्रिड, 3-by-3 blocks और नौ चिह्न। Proof-जटिलता कहानी तभी शुरू होती है जब *n* बढ़ सकता है, और ग्रिड अतिरिक्त गैजेट्स ले सकता है जो उसे सुडोकू के वेश में SAT सूत्र जैसा व्यवहार कराएँ। SAT सूत्र बस yes/no बाधाएँ की list है: क्या variables को सत्य/असत्य assign किया जा सकता है ताकि हर

बग़ाsatisfied हो?



25x25 सुडोकू: उसके नियम finished ग्रिड reveal किए बिनाजाँच किए जासकते हैं—छुपासमाधान, यानीवेरिफ़िकेशन, कोverify करने वाले प्रमाण कादृश्य stand-in।

AI-generated editorial thumbnail — The Clean Paper CC BY 4.0

Sudoku और SAT: एक हीpuzzle दोcostumes में

यह दबाकर सुडोकू “SAT सूत्र जैसाव्यवहार करना” कर सकताहै metaphor नहीं। Translation दोनोंdirections में जातीहै, और आसान दृश्यपूरीलिखीजासकतीहै।

सुडोकू से SAT। SAT केवल सत्य/असत्य बोलता है, इसलिए हर (पंक्ति, स्तंभ, मान) triple के लिए एक boolean variable दें: *x(r,c,v)* का अर्थ "पंक्ति *r*, स्तंभ *c* की कोशिका में मान *v* है।" 4-by-4 सुडोकू (2-by-2 blocks, मान 1–4) को 4·4·4 = 64 variables चाहिए; शास्त्रीय 9-by-9 को 729। हर सुडोकू नियम फिर clauses के batch में बदलता है। (Clause variables या उनकी negations का OR है; पूरी सूत्र सभी clauses का और है।)
हर कोशिका में कम से कम एक मान—हर कोशिका के लिए एक clause:

x(1,1,1) ∨ x(1,1,2) ∨ x(1,1,3) ∨ x(1,1,4)

हर कोशिका में अधिकतम एक मान—हर value-pair के लिए "दोनों नहीं" clause:

¬x(1,1,1) ∨ ¬x(1,1,2) ¬x(1,1,1) ∨ ¬x(1,1,3) … और इसी तरह सभी छह जोड़ियों के लिए।

हर पंक्ति में हर मान—पंक्ति 1 और मान 3 के लिए: कम से कम एक बार,

x(1,1,3) ∨ x(1,2,3) ∨ x(1,3,3) ∨ x(1,4,3)

और अधिकतम एक बार: ¬x(1,1,3) ∨ ¬x(1,2,3), और इसी तरह पंक्ति की हर जोड़ी of कोशिकाएँ के लिए।
Columns और blocks—वैसे ही batches; केवल कोशिकाएँ का समूह बदलता है। शीर्ष-left ब्लॉक और मान 2 के लिए:

x(1,1,2) ∨ x(1,2,2) ∨ x(2,1,2) ∨ x(2,2,2)

साथ में pairwise "नहीं both" clauses।
Printed सुराग—सबसे सरल हिस्सा हर सुराग एकल variable वाला clause है। शीर्ष-left corner में छपे हुए 3 clause बनता है

x(1,1,3)

इन सबका और तभी satisfiable है जब सुडोकू का समाधान हो—और satisfying निर्धारण ही समाधान है: जो x(r,c,v) सत्य हैं उन्हें पढ़ें और ग्रिड भरें। 9-by-9 के लिए 729 variables और कुछ हजार clauses होते हैं, जिन्हें आधुनिक SAT solver milliseconds में हल कर देता है। सुराग clause x(1,1,3) पर ध्यान दें: यह कहता है "यह कोशिका ठीक 3 है," न कि "ये कोशिकाएँ सभी अलग हैं"—यही असमानता आगे प्रूफ़ॉल टिप्पणी में सुराग कोशिकाएँ के लिए अतिरिक्त तर्क की जरूरत पैदा करेगी।
SAT से सुडोकू। शोधपत्र को opposite, कठिन दिशा चाहिए: मनमाना SAT सूत्र से mega-Sudoku बनाएँ जिसका समाधान तभी हो जब सूत्र satisfiable हो। सुडोकू के native नियम केवल "ये कोशिकाएँ सभी अलग हैं" कह सकते हैं, इसलिए मनमाना तर्क कि बाधाएँ को *build* करना पड़ता है—और यही गैजेट्स हैं। Gadget कोशिकाएँ का छोटा pre-fabricated समूह ति होता है, सूत्र के हर clause के लिए एक, जिसमें designated कोशिकाएँ variables का भूमिका निभाती हैं (उनका symbol सत्य/असत्य encode करता है) और समूह ति होता की आंतरिक बाधाएँ ऐसी इंजीनियर्ड होती हैं कि legal fillings ठीक वे assignments हों जो clause satisfy करती हैं। यह NP-completeness प्रमाण की मानक craftsmanship है; समन्वयीकृत सुडोकू के लिए Yato और Seta ने 2003 में इसे किया था।
दोनों directions मिलकर कहते हैं कि N-by-N सुडोकू और SAT एक ही समस्या हैं, अलग costumes में। इसी से यह लेख—और शोधपत्र—grids और चिह्न के जरिए पूरे NP की कहानी कह सकते हैं।

वटिनेस अभी भी easy to तस्वीर है। Alice को mega-Sudoku का पूर्ण वैध filling पता है। Bob को विश्वास करना है कि filling मौजूद है, लेकिन Alice उसे reveal नहीं करना चाहती। Whole filling भेजने पर Bob convinced हो गा, रहस्य खत्म।

शास्त्रीय शून्य-ज्ञान संस्करण में Alice और Bob आपस में संवाद करते हैं। पुराने ढंग का एक सरल मानसिक मॉडल ढकी

हुई टाइलों का इस्तेमाल करता है। Alice हल किए हुए ग्रिड को छिपाती है, हर दौर से पहले प्रतीकों के नाम गुप्त रूप से बदलती है और Bob को किसी यादृच्छिक स्थानीय बाधा—पंक्ति, स्तंभ, खंड या किसी छोटे गैजेट—की जाँच करने देती है। खोली गई कोशिकाओं में सभी प्रतीक अलग-अलग होते तो Bob का विश्वास बढ़ता है। फिर सब कुछ दोबारा ढक दिया जाता है और अगले दौर के लिए प्रतीकों के नाम फिर बदल दिए जाते हैं। (एक पेच यह है कि पहले से छपे संकेतों के लिए अतिरिक्त तरकीब चाहिए, क्योंकि नाम बदलने से वे भी छिप जाते हैं। नीचे टिप्पणी में शास्त्रीय प्रोटोकॉल का वास्तविक समाधान दिया गया है; आगे की चर्चा के लिए यह सरल तस्वीर पर्याप्त है।)

Classical protocols clue cells को वास्तव में कैसे संभालते हैं

नाम बदलना तरकीब की ब्लाइंड spot है। पंक्ति, स्तंभ और खंड नियम कहते हैं "ये कोशिकाएँ सभी अलग हैं," और *सभी अलग* चिह्न के किसी भी नाम बदलना के बाद भी सत्य रहता है। लेकिन सुराग कहता है "इस कोशिका में ठीक 5 है," और नाम बदलना के बाद Bob केवल $\sigma(5)$—कुछ masked symbol—देखता है, नाम बदलना σ जैसे बनिा। वह जाँच नहीं कर सकता। यदि इसे ठीक न करें तो Alice छपे हुए सुराग ignore करके *some* वैध ग्रिड का प्रमाण दे सकती है, जो *इस* puzzle के बारे में कुछ सिद्ध करना नहीं करता। शास्त्रीय literature में दो मानक repairs हैं।

Palette। छपी ग्रिड में N कोशिकाएँ की अतिरिक्त पंक्ति जोड़ें—palette—जिसे Alice सार्वजनिक स्थिर ऑर्डर में चिह्न 1…N से भरती है और फिर बाकी सबके साथ rename करती है, इसलिए उसमें $\sigma(1)\ldots\sigma(N)$ होते हैं। Bob के यादृच्छिक चुनौती में अब अतिरिक्त option है। पंक्ति, स्तंभ, खंड या गैजेट चुनने के अलावा वह **palette plus एक सुराग कोशिका** चुन सकता है। Alice दो को खोलती है; palette उस दौर की नाम बदलना बताती है और Bob जाँच करता है कि सुराग कोशिका छपे हुए सुराग का ठीक renamed संस्करण दिखाती है। यह शून्य-ज्ञान रहता है क्योंकि Bob केवल σ सीखता है—हर दौर fresh draw, अकेले बेकार—और उस कोशिका की मान जिसे puzzle से पहले ही जानता था। रहस्य कोशिकाएँ के बारे में कुछ रिसता नहीं होता, और सिम्युलेटर यादृच्छिक σ draw करके view fake कर सकता है। साउंडनेस इसलिए है कि cheating Alice स्थिर संभावना से हर दौर पकड़ी जाती है, और rounds तब तक repeat होते हैं जब तक doubt negligible न हो।

सुराग को compile away करना। अधिक संरचनात्मक variant विशेष चुनौती जोड़ने की बजाय उसे हटा देता है। सुराग मान को *verify* करने की बजाय अंतर बाधाएँ से force करें: सुराग कोशिका को palette की हर कोशिका से जोड़ें सिवाय उसकी अपनी मान वाली कोशिका के—"$\sigma(1)$ से अलग, $\sigma(2)$ से अलग, …, $\sigma(5)$ को छोड़कर बाकी सबसे अलग।" तब legally वही symbol बचता है जो सुराग का है। अब हर बाधा फिर "ये दो अलग हैं" type की है—नाम बदलना invariant, पंक्ति की तरह checkable। शास्त्रीय graph-coloring प्रोटोकॉल में pre-colored vertices के लिए यही manoeuvre इस्तेमाल होता है, और ऊपर *गैजेट्स* का spirit यही है: MegaSudoku-as-SAT तस्वीर में सुराग भी बाकी बाधाएँ की तरह inequality गैजेट्स में compile होते हैं।

भौतिक प्रोटोकॉल। सुडोकू का वास्तविक दुनिया card प्रोटोकॉल (Gradwohl, Naor, Pinkas और Rothblum, 2007) नाम बदलना इस्तेमाल नहीं करता और hiding शुरू होने से पहले सुराग settle कर देता है। हर कोशिका के लिए Alice समान मान वाले तीन identical cards रखती है—रहस्य कोशिकाएँ के लिए face-down, लेकिन **सुराग कोशिकाएँ के लिए face-up**, ताकि Bob अपनी आँखों से देख सके कि cards flip होने से पहले सुराग respected हैं। फिर हर कोशिका का एक card पंक्ति packet में, एक स्तंभ packet में और एक खंड packet में जाता है; हर packet shuffle और reveal होता है, और Bob जाँच करता है कि उसमें सभी N चिह्न हैं। Shuffling position जानकारी destroy करती है—यही शून्य-ज्ञान है—लेकिन सुराग dealing समय में पहले ही nail down हो चुके थे।

किसी भी तरीके में सबक वही है जिस पर यह लेख बार-बार लौटता है: शून्य-ज्ञान प्रोटोकॉल इस बात की सावधान bookkeeping है कि hiding के बाद *कौन-से* तथ्य survive करते हैं। नाम बदलना "सभी अलग" बचाती है और "equals 5" मिटाती है—इसलिए "equals 5" को दूसरे तरीके से वापस लाना पड़ता है।

यह शोधपत्र का actual प्रोटोकॉल नहीं है। यह शास्त्रीय शून्य-ज्ञान कामानसकि मॉडल है:

- Alice और Bob back-और-forth करते हैं।

- Bob यादृच्छिक जाँच चुनता है।
- Alice केवल स्थानीय consistency reveal करती है, whole समाधान नहीं।
- गोपनीयता का प्रमाण दिखाता है कि Bob का view Alice के रहस्य समाधान के बिना उत्पन्न किया जा सकता था।

इसलिए शास्त्रीय शून्य-ज्ञान positive fact पर built है:

> सिमुलेटर वास्तव में मौजूद है।

अब comfortable हिस्से हटा दें। Alice एक प्रमाण string भेजती है और चली जाती है। Trusted सेटअप नहीं, पहले से shared यादृच्छिक string नहीं, और Bob को असत्य puzzle कभी accept नहीं करना। यही परिस्थिति शास्त्रीय शून्य-ज्ञान survive नहीं कर सकता।

तरकीब से पहले एक और character चाहिए। एक **नियम-पुस्तिका** सुधार करें: logician के अर्थ में औपचारिक प्रमाण-प्रणाली—स्थिर axioms और written गणितीय प्रमाण जाँच करने के यांत्रिक नियम। ZFC, mathematics के मानक axioms, canonical उदाहरण है। आगे सब कुछ पहले से चुने नियम-पुस्तिका के सापेक्ष है, और विकल्प flexible है: निर्माण आपके सुधार किए किसी भी नियम-पुस्तिका के लिए काम करती है, ZFC सहित।

(शोधपत्र से borrowed शब्दकोष टिप्पणी: यहाँ "प्रमाण-प्रणाली" हमेशा इसी नियम-पुस्तिका—गणितीय प्रमाण जाँच करने वाले औपचारिक प्रणाली—का अर्थ है, Alice के भेजे messages का नहीं। Alice और Bob की मशीनरी को "प्रूवर और वेरिफ़ायर" कहा जाता है।)

Gödel-style संस्करण mega-Sudoku कहानी रखता है लेकिन प्रमाण बदलता है।

Displayed आकार का दूसरा बाधा प्रणाली चुनें, उसे **D** कहें। कहानी में S और D समान format के दो MegaSudoku(n) puzzles हैं। Behind scenes D अलग आकार की कठिन तार्किक सूत्र से आया हो सकता है; जरूरत हो तो harmless dummy बाधाएँ से pad करके समान ग्रिड में फिट किया जा सकता है। D ऐसी तार्किक सूत्र से बनाया जाता है जो वास्तव में **unsatisfiable** है: मान की कोई निर्धारण सभी बाधाएँ सत्य नहीं कर सकती, जैसे broken puzzle का legal completed ग्रिड नहीं। Toy उदाहरण ऐसी सूत्र होगी जो एक साथ "X सत्य है" और "X असत्य है" माँगे। इसलिए D का वैध filling नहीं है।

लेकिन D ऐसा broken puzzle नहीं होना चाहिए जिसे *उजागर करना आसान* हो। Toy उदाहरण विफल होता है: कोई भी नियम-पुस्तिका "X और नहीं X" को एक रेखा में खंडन कर देगा। D को ऐसी तरह असत्य होना है कि चुना नियम-पुस्तिका छोटा तर्क से certify न कर सके। यदि नियम-पुस्तिका D को छोटा प्रमाण से खंडन कर सके, नीचे की कहानी collapse होगी: वैकल्पिक मार्ग जो Alice के रहस्य बिना प्रमाण पैदा कर सकता था formally नियम out हो जाएगा, और गोपनीयता गारंटी उसके साथ। इसलिए D ऐसी परिवार से चुना जाता है जिसे स्थिर नियम-पुस्तिका कुशलतापूर्वक खंडन करना नहीं कर सकता: नियम-पुस्तिका के भीतर कोई छोटा प्रमाण नहीं कि D का समाधान नहीं है।

Alice का एक-संदेश प्रमाण फिर either/or कथन के बारे में है:

> या वास्तव कि mega-Sudoku S का समाधान है, या decoy D का समाधान है।

यही तार्किक कड़ी है। D किसी जादुई तरीके से S को सत्य नहीं बनाता। प्रमाण का तर्क यह नहीं है कि "D का समाधान नहीं, इसलिए S का है।" वह disjunction **S or D** को सिद्ध करता है। पूर्ण soundness कहती है कि असत्य disjunction का कोई वैध प्रमाण नहीं हो सकता। चूँकि D वास्तव में असत्य है — उसका कोई समाधान नहीं— disjunction के सत्य होने का एकमात्र रास्ता S का सत्य होना है। इसलिए प्रमाण स्वीकार होता है तो S का समाधान मौजूद होना चाहिए। Decoy, असत्य S को सत्य नहीं बना सकता।

लेकिन शून्य-ज्ञान वाले हिस्से के लिए कल्पना करें कि यदि D का समाधान *होता* तो क्या होता। वह decoy समाधान वैकल्पिक witness बन सकता था। उससे simulator, Alice के वास्तविक mega-Sudoku समाधान को जाने बिना प्रमाण जैसी transcript

बनासकताथा। वास्तवकितामें D कासमग्रम नहीहै, इसलएि यह रास्तासचमुच उपलब्ध नहीं। महत्वपूर्ण बात यह है कि चुनीहुई नयिम-पुस्तकिाकुशलतापूर्वक यह सद्धि नहीकर सकतीकि वह रास्ताबंद है।

इसलएि D के दोनोंक्रियाँहैं। **सहडनेस** के लएि D असत्य है, इसलएि "S or D" कावैध प्रमाण S कोforce करताहै। **Effective शून्य-ज्ञान** के लएि D कठनि to खंडन करनाहै, इसलएि नयिम-पुस्तकिाउस decoy मार्ग कोजल्दीनयिम out नहीकर सकताजोसमिुलेशन संभव बनाता।

इसलएि सुरक्षापरीक्षण अब यह नहीहै:

> क्याहम सद्धि कर सकते हैं कि सम्युिलेटर सच में मौजूद है?

वह बनताहै:

> क्याआपकानयिम-पुस्तकिाकुशलतापूर्वक सद्धि कर सकताहै कि सम्युिलेटर असंभव है?

यदि उत्तर no है, तोsurprisingly मजबूत चीज अनुसरण करतीहै: हर सुरक्षागारंटीजो(a) परीक्षण चलाकर observe की जासकतीहै, और (b) नयिम-पुस्तकिाके भीतर सम्युिलेटर के existence से provably अनुसरण करतीहै, वास्तव में hold करतीहै। उनमें से कसिीपर successful हमलाखुद वहीअनुपस्थति छोटाखंडन बन जाएगा और वह अनुपस्थति छोटाखंडन मौजूद नहीहै। यहीव्यावहारकि रूप से शून्य-ज्ञान का"effective" हसि्साहै।

इसलएि classroom contrast:

शास्त्रीय शून्य-ज्ञान: प्रमाण सुरक्षति हैं क्योंकि सम्युिलेटर exists।

Gödel-style effective शून्य-ज्ञान: observable सुरक्षापरीक्षण के लएि प्रमाण सुरक्षति मानाकएि जाते हैं क्योंकि नयिम-पुस्तकिाकुशलतापूर्वक सद्धि करनानहीकर सकताकि सम्युिलेटर असंभव है।

दूसरादावाऔर कमजोर है। यहीवजह है कि शोधपत्र शास्त्रीय संस्करण में टूटने वालीतीन वशिषताएँ रख सकताहै: एक संदेश, no सेटअप और पूर्ण सहडनेस।

नयापरीक्षण: आप सम्युिलेटर कीअनुपस्थति सद्धि करना नहीकर सकते

Ilango कीrelaxation सवाल बदलतीहै।

शास्त्रीय शून्य-ज्ञान पूछताहै:

> क्यासम्युिलेटर मौजूद है?

व्यावहारकि रूप से शून्य-ज्ञान कुछ और कमजोर पूछताहै:

> क्याआपकाचुनानयिम-पुस्तकिाकुशलतापूर्वक सद्धि कर सकताहै कि कोई सम्युिलेटर मौजूद नहीहै?

यह तकनीकीचाल जैसालग सकताहै, लेकिन मुख्य विचार यहीहै। निर्माण एक अजीब अवस्थामें रहतीहै: सिम्युलेटर वास्तव में **मौजूद नहींहै**—शोधपत्र इस पर स्पष्ट है—लेकिन आपने जोनियम-पुस्तकिासुधार कियिवह कुशलतापूर्वक सिद्ध करनानहींकर सकताकि वह मौजूद नहीं। यदि हर खराब परिणाम जिसकीआपकोपरवाह है ऐसीखंडन माँगे, तोउन नतीजों के लिए प्रणालीशून्य-ज्ञान जैसाव्यवहार करताहै।

यहींGödel आताहै। Decoration के रूप में नहीं और "Gödel crypto कोसुरक्षित बनाताहै" के रूप में नहीं। Connection proof-theoretic है। नियिम-पुस्तकिासर्वोत्तम कहलाताहै यदि सटीक अर्थ में वह सर्वोत्तम संभव हो जब भीकोई नियिम-पुस्तकिा प्रासंगिक kind कीसूत्र कोछोटाप्रमाण से खंडन कर सकताहै, सर्वोत्तम नियिम-पुस्तकिाभीकर सकताहै, अधिकतम polynomially longer प्रमाण के साथ। Krajíček और Pudlák ने 1989 में conjecture कियाकि **कोई सर्वोत्तम प्रमाण-प्रणालीनहींहै**: आप जोनियिम-पुस्तकिासुधार करें, कोई दूसरानियिम-पुस्तकिासत्य कथन कीकिसीपरिवार कोबहुत अधिक succinctly सिद्ध करेगा। यह प्रमाण जटिलताकीकेंद्रीय खुलाconjectures में से है और Gödel incompleteness प्रमेय का finite, जटिलतासैद्धांतिक theoretic cousin है: कुछ सत्य कथन आपके स्थिर नियिम-पुस्तकिामें छोटाप्रमाण नहींरखते—इसलिए नहींकि वे सिद्धांत में unprovable हैं, बल्कि इसलिए कि हर स्थिर नियिम-पुस्तकिाकुछ छोटाtruths कोछोटाप्रमाण के बिनाछोड़ देताहै।

शोधपत्र इस conjecture कोmildly अधिक मजबूत "infinitely often" form में assume करताहै, जोक्रिप्टोग्राफ़िक उपयोग में मानक है। Krajíček और Pudlák के प्रमेय से payoff concrete है: हर नियिम-पुस्तकिाके लिए formulas कीक्रम है जो वास्तव में unsatisfiable हैं, जिनकीछोटाखंडन s नियिम-पुस्तकिानहींदे सकता—और crucially, जिन्हें कुशल एल्गोरिदम *उत्पन्न करना*कर सकताहै। यहीlast गुण, uniformity, विचार कोexistence दावासे actual एल्गोरिदम बनातीहै जिसे Alice चलासकतीहै: decoys D assembly रेखासे आते हैं, पतलावायु से नहीं।

क्रिप्टोग्राफ़िक कदम इसीshortage of प्रमाण शक्ति कोकाम में लगाताहै।

निर्माण क्याकर रहीहै

शोधपत्र कीनिर्माण कोउसकीआकार तक पट्टीकरें।

एक नियिम-पुस्तकिासुधार करें—मान लें ZFC। Proof-जटिलतामान्यताके तहत कुशलतापूर्वक generatable formulas की क्रम है जोवास्तव में unsatisfiable हैं, लेकिन नियिम-पुस्तकिाके पास उनकीunsatisfiability काछोटाप्रमाण नहीं।

अब इस form काएक-संदेश प्रमाण बनाएं:

> यावास्तविक कथन satisfiable है, यायह विशेष कठिन सूत्र satisfiable है।

विशेष कठिन सूत्र satisfiable नहींहै। इसलिए यदि मूल प्रमाण मशीनरीperfectly ध्वनि है, संदेश स्वीकार होने काअर्थ अभीभीवास्तविक कथन सत्य होनाहै। इससे पूर्ण सहंडनेस मिलताहै।

लेकिन शून्य-ज्ञान-like सुरक्षाके लिए imagine करें कि विशेष कठिन सूत्र *satisfiable होती*। तब उसकाविटनेस वास्तविक विटनेस जैसे बिनाप्रमाण simulate करने में इस्तेमाल होसकताथा। Reality में सूत्र satisfiable नहीं—लेकिन नियिम-पुस्तकिाकुशलतापूर्वक सिद्ध करनानहींकर सकताकि ऐसाहै। इसलिए वह कुशलतापूर्वक सिद्ध करनानहींकर सकताकि सिम्युलेटर असंभव है।

यहींhinge है। प्रणालीशास्त्रीय सिम्युलेटर उत्पन्न करके रहस्य hide नहींकरता। Observable सुरक्षापरीक्षण कीबड़ी class के लिए रहस्य कोनियिम-पुस्तकिाकीइस inability के पीछे रखताहै कि वह सिम्युलेटर कीabsence certify नहींकर सकता।

शोधपत्र क्या दावा करता है

मुख्य प्रमेय layers में आता है। मुख्य परिणाम यह है:

एक मानक क्रिप्टोग्राफ़िक मान्यता—**non-interactive वर्टिनेस indistinguishable प्रमाण** का existence, अच्छी तरह studied वस्तुएँ जो कई established मान्यता packages से अनुसरण करते हैं—और proof-जटिलता conjecture कि **कोई (infinitely often) सर्वोत्तम प्रमाण-प्रणाली मौजूद नहीं** के तहत, शोधपत्र हर chosen नियम-पुस्तिका के लिए NP/SAT का एक-संदेश प्रूवर और वेरिफ़ायर construct करता है जिसमें **पूर्ण सॉउंडनेस**, no सेटअप और उस नियम-पुस्तिका के सापेक्ष व्यावहारिक रूप से शून्य-ज्ञान है। (NP/SAT puzzle-like समस्याएँ कामानक "hardest सामान्य denominator" हैं; mega-Sudoku उसका एक costume है।)

परीक्षण-योग्य सुरक्षा गुण बचाए रखना करने के व्यावहारिक दावा के लिए शोधपत्र एक और मानक मान्यता जोड़ता है, derandomization मान्यता **P = BPP**—लगभग, randomness एल्गोरिदम को essential अतिरिक्त शक्ति नहीं देती।

प्रमेय भाषा के बाहर:

- प्रमाण एक संदेश है।
- Trusted सेटअप नहीं है।
- असत्य कथन सिद्ध करना नहीं किए जा सकते।
- प्रूवर शास्त्रीय शून्य-ज्ञान नहीं है—उसका सिमुलेटर नहीं है।
- लेकिन शास्त्रीय शून्य-ज्ञान के हर परीक्षण-योग्य, game-based सुरक्षा परिणाम को इस परिस्थिति में achieve किया जा सकता है।

"परीक्षण-योग्य" महत्वपूर्ण है। इसका अर्थ सुरक्षा failure adversary को खेल में run करके परीक्षण की जा सकती है। कई क्रिप्टोग्राफ़िक सुरक्षा definitions इसी form की हैं: क्या adversary दो encryptions अलग पहचान सकता है, का invert कर सकता है, वर्टिनेस recover कर सकता है या specified प्रयोग जीत सकता है? प्रमेय हर परीक्षण-योग्य गुण के लिए एक प्रूवर देता है, एक at a समय। एक एकल प्रूवर जिसमें हर परीक्षण-योग्य गुण एक साथ होशियार असंभव है—old reusability हमला ("Bob प्रमाण दूसरों को दिखा सकता है") खुद परीक्षण-योग्य गुण है, और यहाँ सच में विफल होती है। शोधपत्र का proposal है कि एकल प्रूवर संभवतः सभी *प्राकृतिक* परीक्षण-योग्य गुण शामिल कर सकता है—जो क्रिप्टोग्राफ़िक practice में वास्तव में आती हैं—लेकिन यह हिस्सा informal "प्राकृतिक" notion और स्पष्ट conjecture पर टिका conditional प्रमेय है। गारंटी observable failures को लक्ष्य करती है, secrecy के हर philosophical या simulation-based अर्थ को नहीं।

एक concrete corollary नाम लेने लायक है: निर्माण uniform प्रूवर के साथ पहले non-interactive *वर्टिनेस hiding* प्रमाण देती है—"puzzle का प्रमाण आपको उसका समाधान खोजने में मदद नहीं करता" no इंटरैक्शन और no सेटअप के साथ—सुनने में मध्यम वस्तु जिसने दशकों तक निर्माण resist की।

यह क्या नहीं कहता

यही section piece कोई मात्रा रखता है।

यह **नहीं** कहता कि old impossibility theorems गलत थे। निर्माण definition बदलकर उनसे बचती है।

यह no इंटरैक्शन, no सेटअप और पूर्ण सॉउंडनेस के साथ सामान्य शास्त्रीय शून्य-ज्ञान **नहीं** देती। शोधपत्र स्पष्ट है

कि constructed प्रूवर कासिम्युलेटर नहीहै।

इसकाअर्थ प्रमाण reuse नहीहोसकतानहीहै। एक-संदेश प्रमाण किसीऔर कोदिखायाजासकताहै; शोधपत्र deniability-style गुण बचाए रखनानहीकरता। (Trusted सेटअप वाले non-interactive शून्य-ज्ञान में भीयहीसीमाहै।)

इसकाअर्थ यह व्यावहारिक प्रूफ़सिस्टम तैनातीकेलिए तैयार नहीहै। यह जटिलतासिद्धांत और क्रिप्टोग्राफ़िक foundations है। परिणाम प्रमाण जटिलताऔर क्रिप्टोग्राफ़ीकीमुख्य मान्यताएँ पर निर्भर है और निर्माण सिद्धांत में क्यासंभव है, उस बारे में है।

यह "Gödel" कोmagic सुरक्षाprimitive **नहीं**बनाता। Gödel connection प्रमाण-प्रणाली, सर्वोत्तम प्रमाण-प्रणाली और incompleteness के finite analogues से है। उपयोगीintuition "incompleteness आपकापassword सुरक्षादेतीहै" नहीं। वह है: यदि नियम-पुस्तिकाकुशलतापूर्वक सिद्ध करनानहींकर सकताकि सिम्युलेटर असंभव है, तोऐसीप्रमाण कीजरूरत वाले हमले सुरक्षाdefinitions के level पर रोकेनाकिए जासकते हैं।

फिर भीयह दिलचस्प क्योंहै

क्रिप्टोग्राफ़ीअक्सर hardness कोसुरक्षामें बदलतीहै। Factoring कठिन है, इसलिए RSA-style मान्यताएँ उपयोगीबनते हैं। Lattice समस्याएँ कठिन हैं, इसलिए lattice क्रिप्टोग्राफ़ीउपयोगीहै। यहाँhardness अजीब है: "रहस्य compute करना कठिन" नहीं, बल्कि "यह सिद्ध करनाकठिन कि कोई खास प्रमाण वस्तु exist नहींकर सकता।"

इसीलिए शोधपत्र unusual लगताहै। वह axioms और नियम-पुस्तिकाएँ कोलगभग क्रिप्टोग्राफ़िक resources कीतरह मानाकरताहै। Usual impossibility सबूतऔर सिमुलेशन के बीच tension कहतीहै। Ilango काकदम इस tension को proof-theoretic curtain के पीछे रखताहै: सिम्युलेटर absent है, लेकिन औपचारिक प्रणालीकुशलतापूर्वक उसकीabsence उजागर करनानहींकर सकता।

पाठक के लिए surprise यह नहींकि यह आज के शून्य-ज्ञान प्रणालियोंको replace करेगा। शायद सीधे नहींकरेगा। Surprise यह है कि गणितीय logic कीसीमाको constructively इस्तेमाल कीजासकतीहै: दीवार कीतरह हीनहीं, शामिल कीतरह भी।

साक्ष्य कितनामजबूत है?

यह प्रमेय शोधपत्र है, इसलिए "साक्ष्य" जीववज्ञान याastronomy शोधपत्र से अलग अर्थ रखताहै। सवाल प्रयोग प्रतिकृति बनीयानहींनहीं। सवाल है definitions, मान्यताएँ और प्रमाण शृंखलादावासमर्थन करते हैं यानहीं।

प्रमाण औपचारिक है और शोधपत्र मान्यताएँ साफ़ बताताहै। मान्यताएँ casual नहीं। Non-interactive वर्टिनेस indistinguishable प्रमाण क्रिप्टोग्राफ़ीके मानक वस्तुएँ हैं और कई established मान्यताpackages से अनुसरण करते हैं। No-सर्वोत्तम-proof-system conjecture प्रमाण जटिलताकीकेंद्रीय conjecture है। P = BPP मानक derandomization मान्यताहै, केवल व्यापक परीक्षण-योग्य-property प्रमेय के लिए।

शोधपत्र यह भीतर्क देताहै कि मान्यताएँ सहीकीमत हैं, मनमानाढाँचानहीं। विपरीत दिशाकाकथन सिद्ध करताहै कि वे मूलतः आवश्यक हैं—यदि ऐसीनिर्माण मौजूद हैं, तोnon-interactive वर्टिनेस indistinguishable प्रमाण exist करने हीचाहिए, और मानक एक-way कार्य मानकर कोई सर्वोत्तम प्रमाण-प्रणालीexist नहींकर सकता। मान्यताएँ "दोनोंओर लाभ वाली" भीहैं: इनमें से किसीकोखंडन करनाअपने आप प्रमाण जटिलता, क्रिप्टोग्राफ़ीयाजटिलतासिद्धांत कीमहत्त्वपूर्ण खोज होगा।

लेकिन परिणाम conditional है, इसलिए विश्वास भीconditional। यदि मान्यताएँ विफल होताहो, प्रमेय कीव्याख्याबदलती

है। और मान्यताएँ hold करने पर भीगारंटीपूराशास्त्रीय शून्य-ज्ञान नहीं, शोधपत्र कारelaxed, proof-theoretic संस्करण है।

इसलिए सही विश्वास: उच्च कि शोधपत्र सुसंगत conditional possibility परिणाम establish करताहै; मध्यम कि मान्यताएँ उस क्रिप्टोग्राफ़िक दुनियाकोdescribe करतीहैं जिसमें हम वास्तव में रहते हैं; immediate व्यावहारिक परिणाम पर कम।

यह क्योंमायने रखताहै

शोधपत्र ऐसारास्ताखोलताहै जिसे बंद मानाजाताथा।

शास्त्रीय सिद्धांत कहतीहै: पूराशून्य-ज्ञान सेटअप बनिाएक संदेश नहींहोसकताऔर perfectly ध्वनि नहींहोसकता। Ilango काशोधपत्र कहताहै: यदि हम शून्य-ज्ञान के उन नतीजोंकीमाँग करें जिन्हें सुरक्षाgames में परीक्षण कियाजा सकताहै, और सुरक्षाdefinition कोनियम-पुस्तिकाक्याकुशलतापूर्वक खंडन कर सकतायानहींकर सकताइस पर निर्भर होनाकरने दें, तोउपयोगीव्यवहार काबड़ाहिस्सावापस पायाजासकताहै—एक संदेश, no सेटअप और पूर्ण सहजनेस के साथ।

यह छोटाdefinitional tweak नहीं। क्रिप्टोग्राफ़िक guarantees सोचने काअलग तरीकाहै। केवल यह मत पूछें कि क्याexists करताहै; पूछें नियम-पुस्तिकाक्यानियम out कर सकताहै। Unprovability कोphilosophical nuisance कीबजाय संरचना कीतरह इस्तेमाल करें।

व्यावहारिक दुनिया कल नहीं बदलेगी। Conceptual मानचित्र बदलती है। अब औपचारिक अर्थ है जिसमें “कोई कुशलतापूर्वक सिद्ध करना नहीं कर सकता कि रहस्य रिसाव हुआ” इतनामजबूत होसकताहै कि “रहस्य रिसाव नहींहुआ” से चहीगई कई game-based protections वापस मिलें।

इसीलिए title में Gödel है।

साफ़ सारांश

शून्य-ज्ञान प्रमाण प्रूवर कोवटिनेस reveal किए बिनावेरिफ़ायर कोकथन सत्य होने काप्रमाण देने देते हैं। शास्त्रीय impossibility परिणाम कहते हैं कि शून्य-ज्ञान कोसेटअप बनिाएक संदेश में squeeze नहींकियाजासकताऔर पूर्ण सहजनेस नहींरख सकता। Rahul Ilango काशोधपत्र इन impossibilities कोखंडन नहींकरता। वह और कमज़ोर notion परिभाषित करताहै, व्यावहारिक रूप से शून्य-ज्ञान: सिम्युलेटर सच में मौजूद होनाजरूरीनहीं, requirement यह है कि chosen प्रमाण-प्रणाली—ZFC जैसाऔपचारिक नियम-पुस्तिका—कुशलतापूर्वक सिद्ध न कर सके कि सिम्युलेटर मौजूद नहीं। क्रिप्टोग्राफ़ीकीमुख्य मान्यताएँ (non-interactive वेरिफ़िनेस indistinguishable प्रमाण) और प्रमाण जटिलता(no सर्वोत्तम प्रमाण-प्रणाली exists) के तहत शोधपत्र NP/SAT के एक-संदेश provers construct करताहै, no सेटअप और पूर्ण सहजनेस के साथ, जोशून्य-ज्ञान के परीक्षण-योग्य, game-based नतीजोंगुण by गुण achieve करते हैं। सभी “प्रकृतिक” गुण शामिल करने वालाएकल पूर्वर आगे कापartly conjectural extension है—और literally हर परीक्षण-योग्य गुण शामिल करनाशायद असंभव है क्योंकि प्रमाण reusable रहते हैं। परिणाम सैद्धांतिक और conditional है, तैनात primitive नहीं, लेकिन proof-theoretic unprovability कोक्रिप्टोग्राफ़िक resource कीतरह इस्तेमाल करने कानयातरीका दिखाताहै।

बिना बढ़ा-चढ़ाकर जाँच

पेपर क्या दिखाता है: Stated मान्यताएँ के तहत NP/SAT के एक-संदेश, no-विश्वास setting में perfectly sound provers बनाए जा सकते हैं जो किसी भी chosen प्रमाण-प्रणाली के संबंध में व्यावहारिक रूप से शून्य-ज्ञान हैं और शास्त्रीय शून्य-ज्ञान के प्रत्येक परीक्षण-योग्य game-based परिणाम को achieve करते हैं।

क्या संभावित लेकिन unconditionally सिद्ध नहीं की जरूरी proof-जटिलता और क्रिप्टोग्राफ़िक मान्यताएँ सही हैं। वे गंभीर, well-studied मान्यताएँ हैं—और शोधपत्र दिखाता है कि वे मूलतः आवश्यक भी हैं और sufficient भी—लेकिन फिर भी मान्यताएँ हैं।

यह क्या नहीं दिखाता No इंटरैक्शन, no सेटअप और पूर्ण सार्वभौमिकता के साथ शास्त्रीय शून्य-ज्ञान; तैनाती-तैयार व्यावहारिक प्रणाली; प्रमाण की deniability या non-reusability; या Gödel incompleteness प्रमेय अपने आप क्रिप्टोग्राफ़ी सुरक्षित करता है।

मुख्य सीमाएँ: गारंटी शून्य-ज्ञान की relaxation है; broadest संस्करण multiple मान्यताएँ पर निर्भर करती है; single-universal-prover दावे partly conjectural हैं; और परिणाम मुख्यतः foundational है।

समान्य पाठक को कितना भरोसा रखना चाहिए? Definitions स्वीकार करने पर यह महत्वपूर्ण conditional सिद्धांत परिणाम है—इस पर उच्च विश्वास। मान्यताएँ वास्तव में capture करती हैं—मध्यम। Immediate व्यावहारिक तैनाती—कम। सुरक्षित takeaway: शोधपत्र शून्य-ज्ञान impossibilities को तोड़ता नहीं; वह उन हिस्सों के आसपास नया proof-theoretic रास्ता खोजता है जो कई सुरक्षा games में मायने रखते हैं।

स्रोत

Ilango, IACR ePrint 2025/1296
https://eprint.iacr.org/2025/1296

FOCS 2025, DOI 10.1109/FOCS63196.2025.00058
https://doi.org/10.1109/FOCS63196.2025.00058