The Clean Paper

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

Et kryptografisk bevis kan skjule hemmeligheten sin ved å gjøre den manglende simulatoren vanskelig å bevise

Nullkunnskapsbevis lar noen bevise en påstand uten å avsløre vitnet. Klassisk teori sier at dette, i full forstand, ikke kan oppnås med én melding, uten betrodd oppsett og med perfekt sunnhet. Rahul Ilangos artikkel får ikke denne umuligheten til å forsvinne. Den endrer målet: I stedet for å kreve at en simulator virkelig finnes, krever den at intet valgt bevissystem effektivt kan bevise at simulatoren ikke finnes. Under store antagelser fra beviskompleksitet og kryptografi er det nok til å gjenvinne falsifiserbare, spillbaserte konsekvenser av nullkunnskap, egenskap for egenskap. Poenget er ikke en internettprimitiv som kan kobles rett inn. Det er en bevisteoretisk måte å gjøre matematisk ubevisbarhet om til kryptografisk dekke på.

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

Trikset er ikke å bevise at hemmeligheten er skjult

Begynn med den enkleste versjonen av nullkunnskap.

Alice vil overbevise Bob om at et sudoku har en løsning. Hvis hun sender løsningen, blir Bob overbevist, men oppgaven er ødelagt. Det hun vil ha, er noe merkeligere: et bevis på at det finnes en løsning, uten å avsløre løsningen.

Det er løftet i et nullkunnskapsbevis. Beviseren (Alice) overbeviser verifikatoren (Bob) om at en påstand er sann, samtidig som hun ikke avslører noe utover at påstanden er sann.

Problemet er at dette løftet har en pris. Et vanlig matematisk bevis har to behagelige egenskaper. Det er **én melding**: Du skriver det ned, leverer det og går din vei. Og det har **perfekt sunnhet**: En usann påstand har overhodet ikke noe gyldig bevis. Klassiske umulighetsresultater sier at nullkunnskap må gi avkall på begge egenskapene — og ikke bare de to sammen; hver av dem er utelukket også alene.

For det første trenger et nullkunnskapsbevis en samtale. Hvis Alice sender én enkelt melding, uten et betrodd oppsett som er ordnet på forhånd, bryter nullkunnskapsgarantien sammen — og dette gjelder uansett hvor mye sunnhet du er villig til å ofre i bytte.

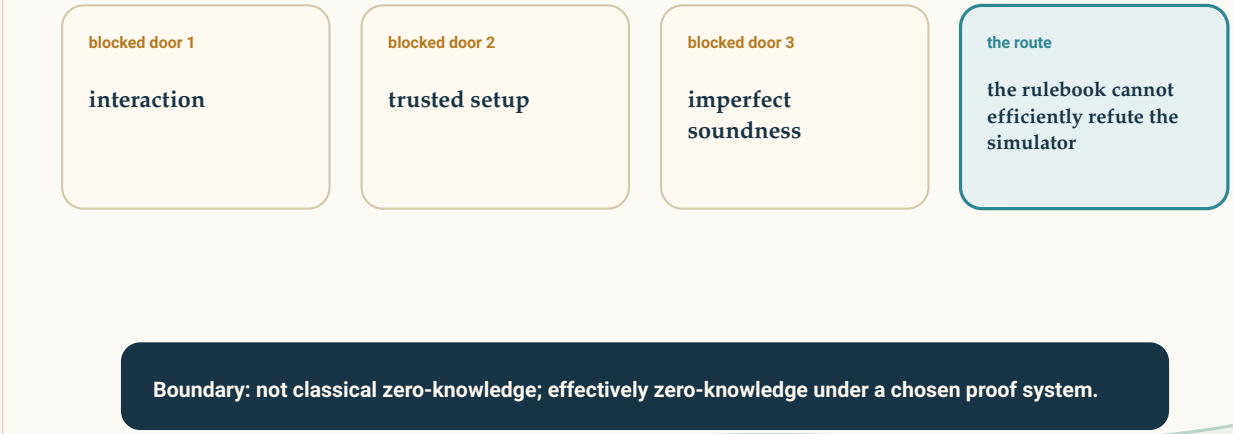
For det andre trenger et nullkunnskapsbevis en liten toleranse for feil. Det viser seg at kravet om perfekt sunnhet i det stille også ødelegger interaksjonen: En verifikator som aldri kan lures, uansett hvilke tilfeldige valg den gjør, kan like gjerne fastsette disse valgene på forhånd — og så snart verifikatoren er forutsigbar, kan Alice svare på alt i én enkelt melding, som nettopp er tilfellet som allerede brøt sammen.

Rahul Ilangos artikkel handler om en vei rundt denne doble muren. Ikke ved å late som om muren ikke finnes, og ikke ved å skape klassisk nullkunnskap i den umulige settingen. Grepet er mer finstemt: å svekke hva «avslører ingenting» betyr, men svekke det på en måte som bevarer sikkerhetsegenskapene kryptografer faktisk kan teste.

Resultatet kalles **effektiv nullkunnskap**.

A proof without leaking the witness

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

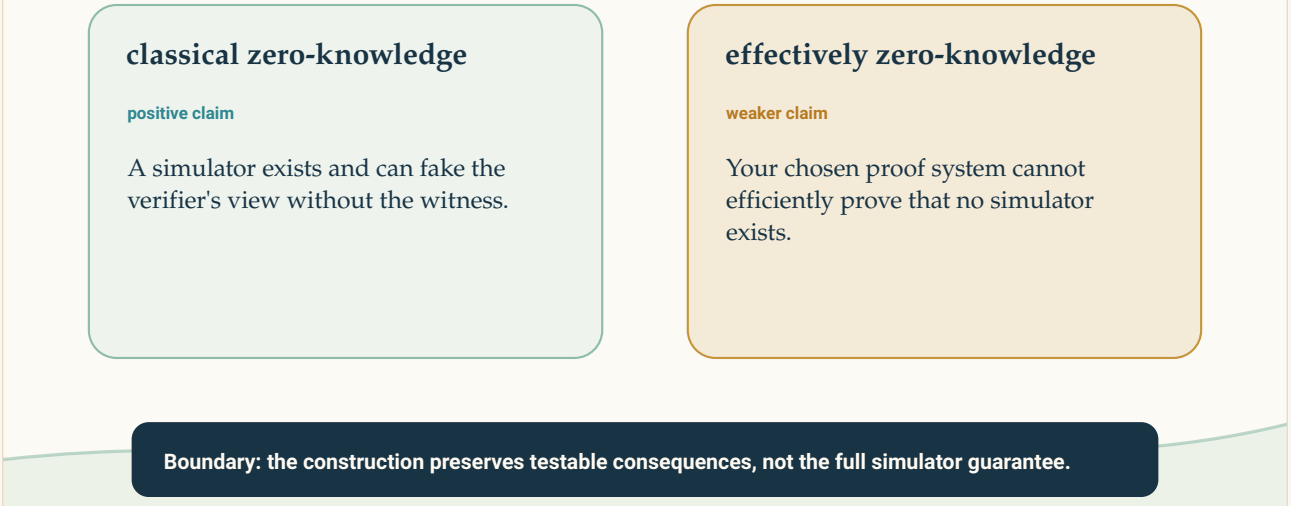


Nullkunnskap er blokkert ved tre dører — interaksjon, betrodd oppsett og ufullkommen sunnhet. Ilangos konstruksjon smetter gjennom en annen: Regelverket kan ikke effektivt gjendrive simulatoren.

Original diagram — The Clean Paper CC BY 4.0

Classical privacy vs effective privacy

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



Klassisk nullkunnskap spør om en simulator finnes; «effektiv nullkunnskap» spør bare om det valgte regelverket ditt effektivt kan bevise at ingen finnes. Det svakere spørsmålet er det som lar konstruksjonen beholde én

melding, intet oppsett og perfekt sunnhet.

Den gamle testen: En simulator finnes

Den klassiske måten å formalisere nullkunnskap på bruker en fiktiv hjelper som kalles en **simulator**.

Tanken er denne: Se for deg Jane, som **ikke** kjenner Alices hemmelighet. Hvis Jane helt på egen hånd kan lage bevis som ser akkurat ut som bevisene Bob ville ha fått fra Alice, har ikke Alices bevis lært Bob noe nytt. Jane kunne allerede etterligne opplevelsen uten Alices hemmelighet.

Klassisk nullkunnskap krever altså en virkelig simulator. Det må finnes en effektiv algoritme som kan lage bevis som ser ekte ut, uten å kjenne hemmeligheten — *vitnet*, på fagspråket; for sudoku er vitnet ganske enkelt det ferdig utfylte rutenettet.

Den definisjonen er kraftfull, men det er også akkurat der den gamle umuligheten rammer. Her er intuisjonen. Et virkelig ikke-interaktivt bevis er bare en streng. Når Bob først har denne strengen, kan han vise den til en annen: Han har fått evnen til å bevise påstanden for andre, noe som allerede høres ut som mer enn «ingenting». De klassiske teoremene skjerper denne intuisjonen til umulighetene ovenfor.

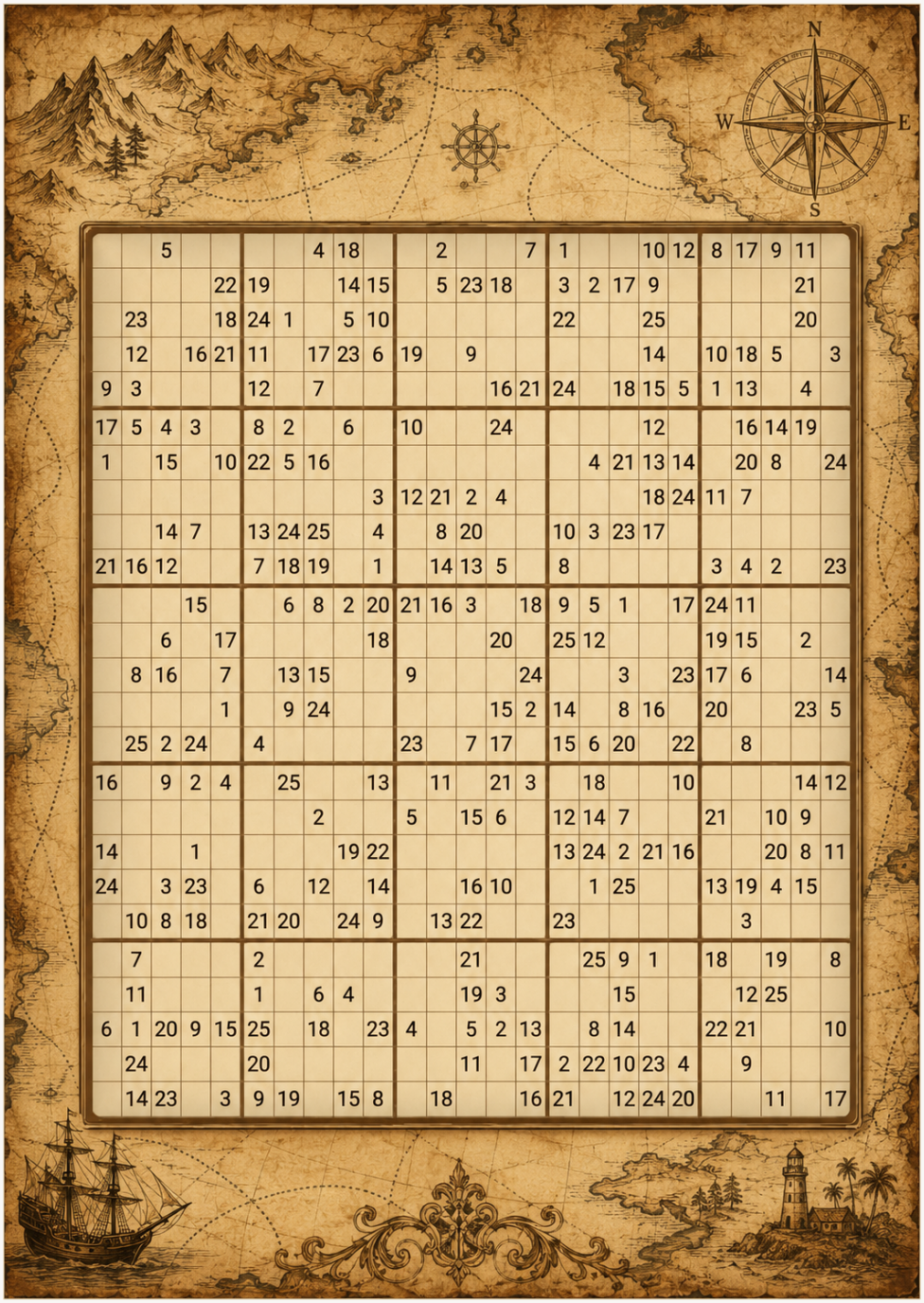
> **De tre egenskapene denne artikkelen insisterer på**
>
> Artikkelens tittel nevner tre begrensninger:
> **Ingen interaksjon:** Alice sender én bevisstreng. Det finnes ingen protokoll med meldinger frem og tilbake.
> **Intet oppsett:** Alice og Bob baserer seg ikke på en betrodd felles referansestreng eller annen offentlig tilfeldighet som er arrangert på forhånd. Mange systemer som kalles «ikke-interaktiv nullkunnskap», er likevel avhengige av oppsett; denne artikkelen mener intet oppsett overhodet.
> **Perfekt sunnhet:** En usann påstand har ikke noe gyldig bevis. Ikke «blir nesten aldri godtatt»; det finnes ikke noe gyldig bevis.
> Disse tre egenskapene er nøyaktig det vanlig, skriftlig matematikk har — og som forklart ovenfor kan klassisk nullkunnskap ikke beholde dem.

En mega-sudokuversjon av forskjellen

Her er en bevisst forenklet måte å få en følelse av forskjellen på.

Ikke bruk et vanlig 9-ganger-9-sudoku i den seriøse delen av analogien. Det er for lite og for endelig: En datamaskin kan ganske enkelt løse det eller bevise at det ikke har noen løsning. Se i stedet for deg en familie av **MegaSudoku(n)**-oppgaver. Skaler opp den vanlige regelen: Velg en blokkstørrelse

n, la $N = n^2$, og bygg et N ganger N-rutenett delt inn i n ganger n-blokker, med N symboler. Vanlig sudoku er bare det bitte lille tilfellet $n = 3$, $N = 9$: et 9-ganger-9-rutenett, 3-ganger-3-blokker og ni symboler. Historien om beviskompleksitet begynner først når n får vokse, og når rutenettet kan ha ekstra gadgeter som får det til å oppføre seg som en SAT-formel utkledd som et sudoku. En SAT-formel er bare en liste over ja/nei-betingelser: Kan du tilordne variablene sann/usann-verdier slik at hver betingelse er oppfylt?



Et 25x25-sudoku: Reglene kan kontrolleres uten å avsløre det ferdige rutenettet — en visuell stedfortreder for et bevis som verifiserer en skjult løsning, vitnet.

Sudoku og SAT: den samme oppgaven i to forkledninger

Påstanden om at et sudoku kan «oppføre seg som en SAT-formel», er ikke en metafor. Oversettelsen går begge veier, og den enkle retningen kan skrives fullt ut.

Fra sudoku til SAT. SAT snakker bare sant/usant, så gi den én boolsk variabel for hver trippel (rad, kolonne, verdi): *x(r,c,v)* betyr «cellen i rad *r*, kolonne *c* inneholder verdien *v*». Et 4-ganger-4-sudoku (2-ganger-2-blokker, verdiene 1–4) trenger 4·4·4 = 64 variabler; det klassiske 9-ganger-9-spillet trenger 729. Hver sudokuregel blir så en gruppe klausuler. (En klausul er en ELLER-kobling av variabler eller negasjonene deres; hele formelen er OG-koblingen av alle klausulene.)

Hver celle inneholder minst én verdi — én klausul per celle:

$$x(1,1,1) \vee x(1,1,2) \vee x(1,1,3) \vee x(1,1,4)$$

Hver celle inneholder høyst én verdi — en «ikke begge»-klausul for hvert verdipar:

$$\neg x(1,1,1) \vee \neg x(1,1,2) \quad \neg x(1,1,1) \vee \neg x(1,1,3)$$ … og så videre for alle seks parene.

Hver rad inneholder hver verdi — for rad 1 og verdien 3: minst én gang,

$$x(1,1,3) \vee x(1,2,3) \vee x(1,3,3) \vee x(1,4,3)$$

og høyst én gang: $\neg x(1,1,3) \vee \neg x(1,2,3)$, og så videre for hvert par av celler i raden.

Kolonner og blokker — identiske grupper; bare cellegruppen endres. For blokken øverst til venstre og verdien 2:

$$x(1,1,2) \vee x(1,2,2) \vee x(2,1,2) \vee x(2,2,2)$$

pluss de parvise «ikke begge»-klausulene.

De trykte ledetrådene — den enkleste delen: Hver ledetråd er en klausul med én enkelt variabel. Et trykt 3-tall i øverste venstre hjørne blir klausulen

$$x(1,1,3)$$

OG-koblingen av alt dette er satisfierbar nøyaktig når sudokuen har en løsning — og en satisfierende tilordning *er* løsningen: Les av hvilke x(r,c,v) som er sanne, og fyll ut rutenettet. For et 9-ganger-9-sudoku blir dette 729 variabler og noen tusen klausuler, som en moderne SAT-løser gjør unna på millisekunder. Legg merke til ledetrådklausulen x(1,1,3): Den sier «denne cellen er nøyaktig 3», ikke «alle disse cellene er forskjellige» — den samme asymmetrien som vil gjøre det ekstra trikset for ledetrådceller nødvendig i protokollnoten lenger nede.

Fra SAT til sudoku. Artikkelen trenger den motsatte, vanskeligere retningen: Gitt en *vilkårlig* SAT-formel, bygg et mega-sudoku som har en løsning nøyaktig når formelen har det. Sudokus egne regler kan bare si «alle disse cellene er forskjellige», så vilkårlige logiske betingelser må *bygges* — og det er nettopp det gadgetene er. En gadget er en liten prefabrikert klynge av celler, én per klausul i formelen, der utpekte celler spiller rollen som variabler (symbolet de inneholder, koder sant eller usant) og de interne betingelsene i klyngen er utformet slik at de eneste lovlige utfyllingene svarer til tilordninger som oppfyller klausulen. Dette er standard

håndverk fra bevis for NP-kompletthet; for generalisert sudoku ble det utført av Yato og Seta i 2003.
Til sammen sier de to retningene at N-ganger-N-sudoku og SAT er det samme problemet i ulike forkledninger. Det er dette som gir denne teksten — og forskningsartikkelen — grunnlag for å fortelle en historie om hele NP ved hjelp av rutenett og symboler.

Vitnet er fortsatt lett å se for seg. Alice kjenner en fullstendig, gyldig utfylling av mega-sudokuen. Bob vil bli overbevist om at en slik utfylling finnes, men Alice vil ikke avsløre den. Hvis hun sender hele utfyllingen, blir Bob overbevist, men hemmeligheten er borte.

I den klassiske nullkunnskapsversjonen samhandler Alice og Bob. En gammeldags tankemodell bruker tildekkede brikker. Alice skjuler det løste rutenettet, gir symbolene nye navn i hemmelighet før hver runde og lar Bob inspisere én tilfeldig valgt lokal betingelse: en rad, en kolonne, en blokk eller en gadget. Hvis de åpnede cellene viser bare ulike symboler, får Bob større tillit. Så dekkes alt til igjen, og symbolene får helt nye navn. (Det finnes en komplikasjon: De oppgitte ledetrådene i oppgaven trenger et ekstra triks, fordi omdøpingen også skjuler dem. Noten nedenfor forklarer hvordan de klassiske protokollene løser dette; lekemodellen er tilstrekkelig for det som følger.)

Hvordan de klassiske protokollene egentlig håndterer ledetrådscellene

Omdøpingstrikset har en blindsone. Reglene for rader, kolonner og blokker sier alle «disse cellene er alle forskjellige», og *alle forskjellige* overlever enhver omdøping av symbolene. Men en ledetråd sier «denne cellen inneholder nøyaktig 5», og etter omdøpingen ser Bob bare $\sigma(5)$ — et maskert symbol — uten å kjenne omdøpingen σ. Han kan ikke kontrollere noe. Uten en løsning på dette kunne Alice bevise at *et eller annet* gyldig rutenett finnes, samtidig som hun ser helt bort fra de trykte ledetrådene, noe som ikke beviser noe om *denne* oppgaven. Den klassiske litteraturen har to standardløsninger.
Paletten. Legg til én ekstra rad med N celler i det skjulte rutenettet — en palett som Alice fyller med symbolene 1…N i en fast, offentlig rekkefølge, og som hun deretter omdøper sammen med alt annet, slik at den inneholder $\sigma(1)…\sigma(N)$. Bobs tilfeldige utfordring får nå ett ekstra alternativ. I tillegg til å velge en rad, kolonne, blokk eller gadget som skal åpnes, kan han velge **paletten pluss én ledetrådcelle**. Alice avdekker begge; paletten avslører omdøpingen i denne runden, og Bob kontrollerer at ledetrådcellen viser nøyaktig den omdøpte versjonen av den trykte ledetråden. Dette forblir nullkunnskap fordi Bob bare lærer σ — som trekkes på nytt i hver runde og er verdiløs alene — og verdien i en celle han allerede kjente fra oppgaven. Ingenting om de hemmelige cellene lekker, og en simulator kan etterligne visningen ved å trekke en tilfeldig σ. Protokollen er sunn fordi en Alice som jukser, blir tatt med en fast sannsynlighet per runde, og rundene gjentas til tvilen er neglisjerbar.
Kompilere bort ledetrådene. En mer strukturell variant fjerner den spesielle utfordringen i stedet for å legge den til. I stedet for å *verifisere* ledetrådverdien tvinger man den frem med ulikhetsbetingelser: Knytt ledetrådcellen til hver palettcelle unntatt den som bærer dens egen verdi — «forskjellig fra $\sigma(1)$, forskjellig fra $\sigma(2)$, …, forskjellig fra alt unntatt $\sigma(5)$». Det eneste symbolet cellen lovlig kan inneholde, er ledetrådens. Hver betingelse er nå igjen av

typen «disse to er forskjellige» — invariant under omdøping og kontrollerbar akkurat som en rad. Dette er den samme manøveren som brukes for forhåndsfargede noder i den klassiske protokollen for graffarging, og det er ånden i ordet *gadgeter* ovenfor: I bildet der MegaSudoku er SAT, kompileres ledetrådene til ulikhetsgadgeter som alle andre betingelser.

Den fysiske protokollen. Kortprotokollen for sudoku i den virkelige verden (Gradwohl, Naor, Pinkas og Rothblum, 2007) bruker ingen omdøping i det hele tatt og avgjør ledetrådene før skjulingen i det hele tatt begynner. For hver celle legger Alice ned tre identiske kort med cellens verdi — med bildesiden ned for hemmelige celler, men **med bildesiden opp for ledetrådceller**, slik at Bob med egne øyne ser at ledetrådene følges før kortene snus. Deretter går ett kort fra hver celle til radens bunke, ett til kolonnens og ett til blokkens; hver bunke blandes og vises, og Bob kontrollerer at den inneholder alle N symboler. Blandingen ødelegger posisjonsinformasjonen (det er nullkunnskapen), men ledetrådene var allerede slått fast da kortene ble lagt ut.

Uansett er lærdommen den samme som denne teksten stadig vender tilbake til: En nullkunnskapsprotokoll er et nøye regnskap over *hvilke* fakta som overlever skjulingen. Omdøping bevarer «alle forskjellige» og visker ut «er lik 5» — derfor må «er lik 5» smugles inn igjen på andre måter.

Dette er ikke protokollen i forskningsartikkelen. Det er tankemodellen for klassisk nullkunnskap:

- Alice og Bob sender meldinger frem og tilbake.
- Bob velger tilfeldige kontroller.
- Alice avslører bare lokal konsistens, ikke hele løsningen.
- Beviset for personvern fungerer ved å vise at Bobs visning kunne ha blitt generert uten Alices hemmelige løsning.

Klassisk nullkunnskap er altså bygd rundt et positivt faktum:

> En simulator finnes virkelig.

Fjern nå de behagelige delene. Alice sender én bevisstreng og går sin vei. Det finnes ikke noe betrodd oppsett, ingen felles tilfeldig streng som er forberedt på forhånd, og Bob må aldri godta en usann oppgave. Dette er settingen klassisk nullkunnskap ikke kan overleve i.

Én figur til trengs før trikset. Fastsett et **regelverk**: et formelt bevissystem, i logikerens betydning — et fast sett aksiomer pluss mekaniske regler for å kontrollere skriftlige matematiske bevis. ZFC, standardaksiomene i matematikken, er det kanoniske eksempelet. Alt herfra uttrykkes relativt til et regelverk valgt på forhånd, og valget er fleksibelt: Konstruksjonen virker for ethvert regelverk du fastsetter, inkludert ZFC.

(En merknad om ord, lånt fra selve forskningsartikkelen: «Bevissystem» betyr her alltid dette regelverket — det formelle systemet som kontrollerer matematiske bevis — aldri meldingene Alice sender. Alices og Bobs maskineri kalles «beviseren og verifikatoren».)

Versjonen i Gödels ånd beholder historien om mega-sudoku, men endrer beviset.

Velg et annet betingelsessystem av samme viste størrelse, og kall det **D**. I historien er S og D to MegaSudoku(n)-oppgaver i samme format. Bak kulissene kan D ha startet som en vanskelig logisk formel av en annen størrelse; om nødvendig kan den fylles ut med harmløse dummybetingelser slik at den passer i det samme rutenettet. D er bygd fra en logisk formel som faktisk er **usatisfierbar**: Det finnes ingen mulig verditilordning som gjør alle betingelsene sanne, akkurat som en ødelagt oppgave ikke har noe lovlig ferdig utfylt rutenett. Et enkelt eksempel ville være en formel som krever både «X er sann» og «X er usann». D har altså ingen gyldig utfylling.

Men D må ikke være en ødelagt oppgave som er *lett å avsløre*. Det enkle eksempelet ovenfor mislykkes med dette: Ethvert regelverk gjendriver «X og ikke-X» på én linje. D må være usann på en måte det valgte regelverket ikke kan bekrefte med et kort argument. Hvis regelverket kunne gjendrive D med et kort bevis, ville historien nedenfor bryte sammen: Den alternative veien som kunne ha produsert bevis uten Alices hemmelighet, kunne formelt utelukkes, og personverngarantien ville forsvinne sammen med den. Derfor velges D fra en familie som det fastsatte regelverket ikke effektivt kan gjendrive: Innenfor dette regelverket finnes det ikke noe kort bevis på at D ikke har noen løsning.

Alices bevis i én melding handler da om en enten/eller-påstand:

> enten har den virkelige mega-sudokuen S en løsning, eller så har lokkemiddelet D en løsning.

Dette er den logiske forbindelsen. D genereres **ikke** på en magisk måte som gjør S sann. Beviset argumenterer ikke «D har ingen løsning, derfor har S en løsning». Det beviser disjunksjonen **S eller D**. Perfekt sunnhet sier at en usann disjunksjon ikke kan ha et gyldig bevis. Siden D er usann i virkeligheten — den har ingen løsning — kan disjunksjonen bare være sann hvis S er sann. Hvis beviset godtas, må S altså ha en løsning. Lokkemiddelet kan ikke gjøre en usann S sann.

Men for den nullkunnskapslignende delen: Spør hva som ville skjedd hvis D *faktisk* hadde en løsning. Denne lokkemiddelløsningen ville fungert som et alternativt vitne. Den ville latt noen produsere bevis uten å kjenne Alices virkelige mega-sudokuløsning — med andre ord en simulator. I virkeligheten har D ingen løsning, så denne veien til en simulator er stengt. Poenget er at regelverket ikke effektivt kan bevise at den er stengt.

D har altså to oppgaver. For **sunnheten** er D usann, så et gyldig bevis på «S eller D» fremtvinger S. For **effektiv nullkunnskap** er D vanskelig å gjendrive, så regelverket kan ikke raskt utelukke lokkemiddelveien som ville ha gjort simulering mulig.

Sikkerhetstesten er dermed ikke lenger:

> Kan vi bevise at en simulator virkelig finnes?

Den blir:

> Kan regelverket ditt effektivt bevise at simulatoren er umulig?

Hvis svaret er nei, følger noe overraskende sterkt: Enhver sikkerhetsgaranti som (a) kan observeres ved å kjøre en test, og (b) beviselig følger — innenfor dette regelverket — av at en simulator finnes,

gjelder faktisk. Et vellykket angrep på noen av dem ville i seg selv utgjøre den manglende korte gjendrivelsen, og den manglende korte gjendrivelsen finnes ikke. Det er den «effektive» delen av effektiv nullkunnskap.

Kontrasten i klasserommet blir dermed:

Klassisk nullkunnskap: Bevisene er sikre fordi en simulator finnes.

Effektiv nullkunnskap i Gödels ånd: Bevisene behandles som sikre for observerbare sikkerhetstester fordi regelverket ikke effektivt kan bevise at simulatoren er umulig.

Den andre påstanden er svakere. Det er også derfor artikkelen kan beholde de tre egenskapene som fikk den klassiske versjonen til å bryte sammen: én melding, intet oppsett og perfekt sunnhet.

Den nye testen: Du kan ikke bevise at simulatoren mangler

Ilangos lemping endrer spørsmålet.

Klassisk nullkunnskap spør:

> Finnes det en simulator?

Effektiv nullkunnskap spør om noe svakere:

> Kan det valgte regelverket ditt effektivt bevise at ingen simulator finnes?

Det kan høres ut som en teknisk unnamanøver, men det er kjerneideen. Konstruksjonen befinner seg i en merkelig tilstand: En simulator finnes faktisk ikke — artikkelen sier dette uttrykkelig — men regelverket du fastsatte, kan ikke effektivt bevise at den ikke finnes. Hvis enhver dårlig konsekvens du bryr deg om, ville kreve en slik gjendrivelse, oppfører systemet seg likevel som nullkunnskap for disse konsekvensene.

Det er her Gödel kommer inn. Ikke som pynt, og ikke som «Gödel gjør krypto sikkert». Forbindelsen er bevisteoretisk. Et regelverk kalles **optimalt** hvis det i en presis forstand er det best mulige: Når et hvilket som helst regelverk kan gjendrive en formel av det relevante slaget med et kort bevis, kan også det optimale regelverket gjøre det, med et bevis som høyst er polynomisk lengre. Krajíček og Pudlák fremsatte i 1989 formodningen om at **intet optimalt bevissystem finnes**: Uansett hvilket regelverk du fastsetter, beviser et annet regelverk en eller annen familie av sanne påstander langt mer konsist. Dette er en av de sentrale åpne formodningene i beviskompleksitet, og det er den endelige, kompleksitetsteoretiske slektningen til Gödels ufullstendighetsteorem: Noen sanne påstander har ikke noe kort bevis i regelverket du fastsatte — ikke fordi de prinsipielt er ubeviselige, men fordi ethvert fastsatt regelverk etterlater noen korte sannheter uten korte bevis.

Artikkelen antar denne formodningen (i en noe sterkere «uendelig ofte»-form, som er standard når formodninger brukes kryptografisk). Gevinsten, gjennom et teorem av Krajíček og Pudlák, er

konkret: For hvert regelverk finnes det en sekvens av formler som virkelig er usatisfierbare, og som regelverket ikke kan gjendrive med korte bevis — og som, avgjørende nok, en effektiv algoritme kan *generere*. Den siste egenskapen, uniformitet, er det som gjør om hele ideen fra en eksistenspåstand til en faktisk algoritme Alice kan kjøre: Lokkemidlene D kommer fra et samlebånd, ikke ut av løse luften.

Det kryptografiske grepet er å sette denne mangelen på beviskraft i arbeid.

Hva konstruksjonen gjør

Her er konstruksjonen i artikkelen, strippet ned til formen.

Fastsett et regelverk — for eksempel ZFC. Under antagelsen fra beviskompleksitet finnes det en effektivt genererbar sekvens av formler som faktisk er usatisfierbare, men regelverket har ikke noe kort bevis på at de er usatisfierbare.

Bygg nå et bevis i én melding med denne formen:

> enten er den virkelige påstanden satisfierbar, eller så er denne spesielle vanskelige formelen satisfierbar.

Den spesielle vanskelige formelen er ikke satisfierbar. Hvis det underliggende bevismaskineriet har perfekt sunnhet, betyr det å godta meldingen derfor fortsatt at den virkelige påstanden er sann. Det gir perfekt sunnhet.

Men for den nullkunnskapslignende sikkerheten kan du forestille deg at den spesielle vanskelige formelen *var* satisfierbar. Da kunne vitnet til formelen brukes til å simulere bevis uten kjennskap til det virkelige vitnet. Formelen er ikke satisfierbar i virkeligheten — men regelverket kan ikke effektivt bevise det. Derfor kan det ikke effektivt bevise at simulatoren er umulig.

Det er hengselet. Systemet skjuler ikke hemmeligheten ved å frembringe en klassisk simulator. For en stor klasse observerbare sikkerhetstester skjuler det hemmeligheten bak regelverkets manglende evne til å bekrefte at simulatoren mangler.

Hva artikkelen hevder

Hovedteoremet kommer i flere lag. Kjerneresultatet er dette:

Under en kryptografisk standardantagelse — eksistensen av **ikke-interaktive vitneuskillbare bevis**, velstuderte objekter som følger av flere etablerte antagelsespakker — og under den beviskompleksitetsteoretiske formodningen om at **intet (uendelig ofte) optimalt bevissystem finnes**, konstruerer artikkelen, for hvert valg av regelverk, en beviser og verifikator med én melding for NP/SAT, med **perfekt sunnhet** og uten oppsett, som er effektivt nullkunnskap relativt til dette regelverket. (NP/SAT

er den vanlige «vanskeligste fellesnevneren» for oppgavelignende problemer; mega-sudoku er én forkledning det bærer.)

For den bredere påstanden om å bevare falsifiserbare sikkerhetsegenskaper legger artikkelen til enda en standardantagelse, derandomiseringsoppfatningen **P = BPP** (grovt sagt: tilfeldighet gir ikke algoritmer noen vesentlig ekstra kraft).

Oversatt fra teoremspråk:

- Beviset er én melding.
- Det finnes ikke noe betrodd oppsett.
- Usanne påstander kan ikke bevises.
- Beviseren er ikke klassisk nullkunnskap — den har ingen simulator.
- Men enhver falsifiserbar, spillbasert sikkerhetskonsekvens av klassisk nullkunnskap kan oppnås i denne settingen.

«Falsifiserbar» er viktig. Det betyr at en sikkerhetssvikt kan testes ved å kjøre en motstander i et spill. Mange kryptografiske sikkerhetsdefinisjoner har denne formen: Kan motstanderen skille mellom to krypteringer, invertere en funksjon, gjenvinne et vitne eller vinne et bestemt eksperiment? Teoremet gir en beviser for hver falsifiserbar egenskap, én om gangen. En enkelt beviser som har *alle* falsifiserbare egenskaper samtidig, er sannsynligvis umulig — det gamle gjenbruksangrepet («Bob kan vise beviset til andre») er selv en falsifiserbar egenskap, og den svikter faktisk her. Artikkelens forslag er at én beviser plausibelt kan dekke alle *naturlige* falsifiserbare egenskaper — dem som faktisk forekommer i kryptografisk praksis — men denne delen er et betinget teorem som hviler på et uformelt begrep om «naturlig», pluss en uttrykkelig formodning. Garantien er rettet mot observerbare svikt, ikke mot enhver filosofisk eller simuleringsbasert betydning av hemmelighold.

Én konkret korollar er verdt å nevne: Konstruksjonen gir de første ikke-interaktive *vitneskjulende* bevisene med en uniform beviser — «et bevis for en oppgave hjelper deg ikke å finne løsningen», uten interaksjon og uten oppsett — et beskjedent klingende objekt som hadde motstått konstruksjon i flere tiår.

Hva dette ikke sier

Dette er avsnittet som holder teksten ærlig.

Det sier **ikke** at de gamle umulighetsteoremene tok feil. Konstruksjonen unngår dem ved å endre definisjonen.

Det gir **ikke** vanlig, klassisk nullkunnskap uten interaksjon, uten oppsett og med perfekt sunnhet. Artikkelen sier uttrykkelig at den konstruerte beviseren ikke har noen simulator.

Det betyr **ikke** at beviset ikke kan gjenbrukes. Et bevis i én melding kan fortsatt vises til en annen; artikkelen bevarer ikke egenskaper av typen benektbarhet. (Ikke-interaktiv nullkunnskap med betrodd oppsett har den samme begrensningen.)

Det betyr **ikke** at dette er en praktisk protokoll som er klar til utrulling. Dette er kompleksitetsteori og kryptografisk grunnforskning. Resultatet avhenger av store antagelser fra beviskompleksitet og kryptografi, og konstruksjonen handler om hva som er prinsipielt mulig.

Det gjør **ikke** «Gödel» til en magisk sikkerhetsprimitiv. Gödel-forbindelsen går gjennom bevissystemer, optimale bevissystemer og endelige analoger til ufullstendighet. Den nyttige intuisjonen er ikke «ufullstendighet beskytter passordet ditt». Den er: Hvis et regelverk ikke effektivt kan bevise at en simulator er umulig, kan angrep som ville kreve et slikt bevis, blokkeres på sikkerhetsdefinisjonenes nivå.

Hvorfor det likevel er interessant

Kryptografi gjør ofte vanskelighet om til sikkerhet. Faktorisering er vanskelig, så antagelser av RSA-typen blir nyttige. Gitterproblemer er vanskelige, så gitterkryptografi blir nyttig. Her er vanskeligheten mer fremmedartet: ikke «vanskelig å beregne en hemmelighet», men «vanskelig å bevise at et bestemt bevisobjekt ikke kan finnes».

Det er derfor artikkelen føles uvanlig. Den behandler aksiomer og regelverk nesten som kryptografiske ressurser. Den vanlige umuligheten sier at det er en spenning mellom sunnhet og simulering. Ilangos grep er å plassere spenningen bak et bevisteoretisk forheng: Simulatoren mangler, men det formelle systemet kan ikke effektivt avdekke dette fraværet.

For en leser er det overraskende ikke at dette vil erstatte dagens nullkunnskapssystemer. Det vil det sannsynligvis ikke, i hvert fall ikke direkte. Det overraskende er at en begrensning fra matematisk logikk kan brukes konstruktivt: ikke bare som en mur, men som en form for dekke.

Hvor sterke er beleggene?

Dette er en teoremartikkel, så «belegg» betyr noe annet enn i en artikkel om biologi eller astronomi. Spørsmålet er ikke om et eksperiment er replikert. Spørsmålet er om definisjonene, antagelsene og beviskjeden støtter påstanden.

Beviset er formelt, og artikkelen er tydelig om antagelsene sine. Antagelsene er ikke tilfeldige. Ikke-interaktive vitneuskillbare bevis er standardobjekter i kryptografi og følger av flere etablerte antagelsespakker. Formodningen om at intet optimalt bevissystem finnes, er en sentral formodning i beviskompleksitet. P = BPP er en standardoppfatning om derandomisering som bare brukes for det bredere teoremet om falsifiserbare egenskaper.

Artikkelen argumenterer også for at antagelsene er den riktige prisen, ikke et vilkårlig stillas: Den beviser en omvendt påstand som viser at de i hovedsak er nødvendige — hvis konstruksjoner som dette overhodet finnes, må ikke-interaktive vitneuskillbare bevis finnes, og (gitt vanlige enveisfunksjoner) kan intet optimalt bevissystem finnes. Og antagelsene er «vinn-vinn»: Å gjendrive noen av dem ville i seg selv være en banebrytende oppdagelse i beviskompleksitet, kryptografi eller kompleksitetsteori.

Men fordi resultatet er betinget, er tilliten til det også betinget. Hvis disse antagelsene svikter, endres tolkningen av teoremet. Og selv om antagelsene holder, er garantien ikke full klassisk nullkunnskap; den er artikkelens svekkede, bevisteoretiske versjon.

Den rette tilliten er derfor høy til at artikkelen etablerer et sammenhengende, betinget mulighetsresultat; moderat til at antagelsene beskriver den kryptografiske verdenen vi faktisk lever i; og lav når det gjelder umiddelbare praktiske konsekvenser.

Hvorfor det er viktig

Artikkelen åpner en vei som skulle være stengt.

Klassisk teori sier: Full nullkunnskap kan ikke være én melding uten oppsett og kan ikke ha perfekt sunnhet. Ilangos artikkel sier: Hvis vi spør etter konsekvensene av nullkunnskap som kan testes i sikkerhetsspill, og hvis vi lar sikkerhetsdefinisjonen avhenge av hva et regelverk kan eller ikke effektivt kan gjendrive, kan mye av den nyttige oppførselen gjenvinnes — med én melding, intet oppsett og perfekt sunnhet.

Det er ikke en liten definisjonsjustering. Det er en annen måte å tenke på kryptografiske garantier. I stedet for bare å spørre hva som finnes, kan du spørre hva regelverket ditt kan utelukke. I stedet for å behandle ubevisbarhet som en filosofisk plage kan du bruke den som struktur.

Den praktiske verdenen endrer seg kanskje ikke i morgen. Men det gjør det begrepsmessige kartet. Det finnes nå en formell betydning der «ingen kan effektivt bevise at hemmeligheten lekket» kan være sterkt nok til å gjenvinne mange av de spillbaserte beskyttelsene vi ønsket fra «hemmeligheten lekket ikke».

Det er derfor Gödel hører hjemme i tittelen.

Kort oppsummert

Nullkunnskapsbevis lar en beviser overbevise en verifikator om at en påstand er sann, uten å avsløre vitnet. Klassiske umulighetsresultater sier at nullkunnskap ikke kan presses inn i én melding uten oppsett og ikke kan ha perfekt sunnhet. Rahul Ilangos artikkel gjendriver ikke disse umulighetene. Den definerer et svakere begrep, effektiv nullkunnskap: I stedet for å kreve at en simulator virkelig finnes, krever det at et valgt bevissystem — et formelt regelverk som ZFC — ikke effektivt kan bevise at ingen simulator finnes. Under store antagelser fra kryptografi (ikke-interaktive vitneuskillbare bevis) og beviskompleksitet (intet optimalt bevissystem finnes) konstruerer artikkelen bevisere med én melding for NP/SAT, uten oppsett og med perfekt sunnhet, som oppnår de falsifiserbare, spillbaserte konsekvensene av nullkunnskap egenskap for egenskap. Én enkelt beviser som dekker alle «naturlige» slike egenskaper, er en videre, delvis formodningsbasert utvidelse — og å dekke bokstavelig talt enhver falsifiserbar egenskap er sannsynligvis umulig, fordi bevis fortsatt kan gjenbrukes. Resultatet er teoretisk og betinget, ikke en utrullet primitiv, men det viser en ny måte å bruke

bevisteoretisk ubevisbarhet som en kryptografisk ressurs på.

Nøktern vurdering

Hva artikkelen viser: Under de oppgitte antagelsene kan man bygge bevisere for NP/SAT med én melding, uten oppsett og med perfekt sunnhet, som er effektivt nullkunnskap relativt til ethvert valgt bevissystem, og som oppnår hver falsifiserbar, spillbasert konsekvens av klassisk nullkunnskap.

Hva som er plausibelt, men ikke bevist uten vilkår: At de nødvendige antagelsene fra beviskompleksitet og kryptografi holder. De er seriøse, godt studerte antagelser — og artikkelen viser at de i hovedsak er både nødvendige og tilstrekkelige — men de er fortsatt antagelser.

Hva den ikke viser: Klassisk nullkunnskap uten interaksjon, uten oppsett og med perfekt sunnhet; et praktisk system klart til utrulling; benektbarhet eller at bevis ikke kan gjenbrukes; eller at Gödels ufullstendighetsteorem i seg selv sikrer kryptografi.

Hovedbegrensninger: Garantien er en lemping av nullkunnskap; den bredeste versjonen avhenger av flere antagelser; påstandene om én universell beviser er fortsatt delvis formodningsbaserte; og resultatet er først og fremst grunnleggende teori.

Hvor stor tillit bør en allmenn leser ha? Høy tillit til at dette er et viktig, betinget teoriresultat hvis definisjonene godtas. Moderat tillit til at antagelsene fanger virkeligheten. Lav tillit til umiddelbar praktisk utrulling. Den trygge konklusjonen er: Artikkelen bryter ikke umulighetsresultatene for nullkunnskap; den finner en ny bevisteoretisk vei rundt de delene av dem som betyr noe for mange sikkerhetsspill.

Kilder

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`