Beregningens grænser

Hvad der til gengæld ikke blot er en teoretisk ide, men har stor konkret betydning for en masse ting i hverdagen, er de fysiske grænser for beregning. Gödel og Turing viste, at der findes formelle grænser for sikker viden. Men der ser ud til at være meget mere håndfaste grænser for, hvor svære opgaver man kan løse af den simple årsag, at de er for komplicerede, at verden er endelig og tiden knap.

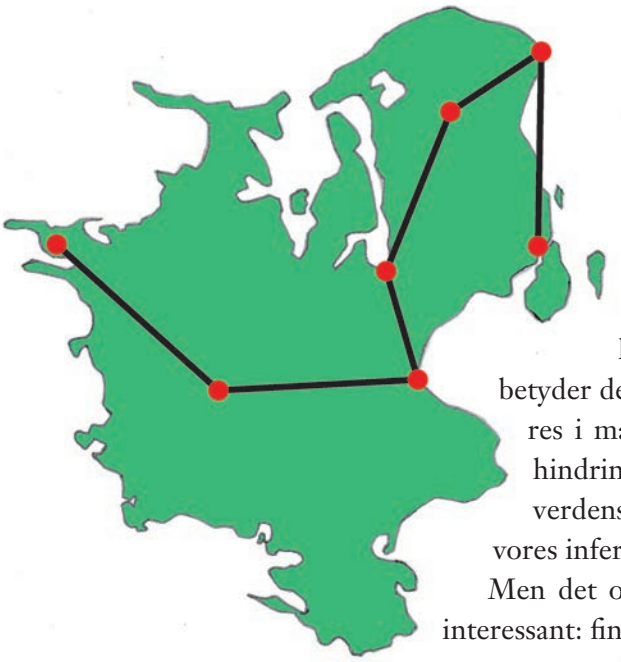
Denne del af historien startede i 1960'erne, hvor matematikere begyndte at spekulere over måder, hvorpå man kan ordne matematiske opgaver i forhold til deres sværhedsgrad. Men hvordan definere sværhedsgrad? Forskerne enedes om at lade sværhedsgraden angive antallet af operationer eller regnetrin, man skal bruge for at komme frem til en løsning af et givent matematisk spørgsmål. Man kunne for eksempel spørge, om tallet $12 \cdot 2$ er nemmere at beregne end tallet 12^2, dvs. om disse regnestykker har forskellige sværhedsgrader. I dag ved man, at det at kvadrere er sværere end at fordoble. En fordobling af et vilkårligt tal med k antal cifre kræver, at man ganger hvert enkelt af de k cifre med to, dvs. antallet af regneskridt er ligefrem proportional med k. En kvadrering derimod kræver normalt, at man ganger hvert ciffer med hvert ciffer, og dermed kommer op på k^2 operationer (for slet ikke at nævne summationen bagefter). Men der findes også endnu sværere regnestykker, som kræver k^3 eller k^4 operationer, og jo sværere de bliver, jo længere tid vil de tage at beregne på en computer. De fleste problemer, man kender fra skolen, er af præcis den beskaffenhed: man kan beregne dem ved hjælp af k^r trin, hvor eksponenten r er et naturligt tal og k antallet af cifre på det tal, man begyndte med. Lægge sammen, gange, dividere, finde rødder, løse ligninger osv. – alle kan de løses med maksimalt k^r operationer (for forskellige værdier af r, vel at mærke). Og i de tilfælde siger man, at de er af polynomiel art eller "af typen P".

Faktisk findes der matematiske problemer af typen P, som stadig tiltrækker forskernes interesse. Blandt andet sorteringsalgoritmer. Alle kender det møjsommelige arbejde med at sortere adresselister, spillekort eller lignende efter en bestemt rangorden. Normalt ville man starte med f.eks. at lede efter den adresse i listen (med k adresser), som starter højest oppe i alfabetet. Til det kræves maksimalt k operationer, idet det jo ikke er sikkert, at man kan finde den rette adresse med det samme. Med den næste adresse kræves der maksimalt k-1 operationer, den næste igen k-2 osv. Det vil sige, at man kan

komme op på $1+2+3+ \ldots + k = \frac{k(k+1)}{2}$ operationer, og da det tal kan sammenlignes med k^2 for store k, er sorteringsalgoritmen af typen k^2. Men sjovt nok kan man gøre det meget hurtigere og få eksponenten helt ned i nærheden af r = 1 takket være kloge hoveder og fiffige programmører.

Men der findes også problemer, som er meget mere krævende end dem af typen P. For eksempel opgaven: find et tal, som går op i 1.050.504.368.559.379, og det må ikke være tallet 1 eller tallet selv. Hvis man vil prøve lykken, må man regne med oddset én ud af 30 millioner, hvilket er en del sværere end at vinde i lotto. Det skyldes, at 1.050.504.368.559.379 kun er et produkt af to tal, nemlig primtallene 18.712.789 og 56.138.311, altså tal, som kun kan deles med sig selv og 1. Der findes ingen metoder, hvormed man konsistent kan løse den slags opgaver hurtigt, end ikke på computere, og gudskelov for det, da krypteringsteknikkernes sikkerhed på bl.a. internettet afhænger af det. Man kalder denne form for matematiske problemer for NP-problemer, hvilket står for "nondeterministic polynomial". Det skyldes, at man endnu ikke har fundet nogen effektiv deterministisk metode, hvormed man kunne løse dem på samme måde, som man kan løse problemer af typen P. Man kan kun gætte og være heldig eller også få et rigtigt surt arbejde. Men ikke desto mindre er de meget vigtige for en række praktiske problemer: hvordan finder man den optimale måde, hvormed industrimaskiner kan operere på et givent område? Hvordan koordineres hjemmeservice bedst? Eller den klassiske: hvordan bruger en handelsrejsende mindst mulig benzin, hvis han pendler mellem x antal forskellige byer? Alle disse problemer er NP-problemer, men måske er de i virkeligheden blot problemer af typen P, hvor man bare endnu ikke kender beregningsmetoden. Ingen kender svaret.

Ikke så få forskere ville give deres højre arm for at kunne svare på spørgsmålet om P = NP, det vil sige, om der findes NP-problemer, som kan reduceres til P, og som derfor kan løses uden brug af held. Afgørende i denne sammenhæng er, at mange NP-problemer er såkaldt "NP-komplette", hvilket betyder, at hvis man først har vist, at ét NP-komplet problem (som f.eks. den handelsrejsende) er af typen P, så er alle andre NP-problemer det også. Hvis ét problem er løst, så er alle de andre det også! Det skyldes, at alle den slags matematiske problemer af klassen NP kan omskrives til en kombination af logiske operatorer (som AND, OR, NOT osv.), og på den måde generaliseres til de såkaldte cookske problemer, der er opkaldt efter den amerikanske matematiker Stephen Cook (f. 1939), som fandt på metoden i 1971.



Det er ikke nemt at finde ud af hvilken rejserute, der er kortest, hvis man skal igennem en række byer. For bare syv byer er antallet af ruter lig $7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1 = 5040$. For 67 byer ville hele universets regnekraft ikke række til at tjekke dem alle sammen.

Hvis et NP-komplet problem kan vises at være af typen P, så betyder det, at held alligevel kan undværes i matematikken, og at den eneste hindring for, at vi mennesker kan løse verdens værste matematiske gåder, er vores inferiøre intellekt.

Men det omvendte spørgsmål er lige så interessant: findes der NP-komplette problemer, som garanteret ikke er af typen P? I så tilfælde ville de filosofiske implikationer være mindst lige så betydningsfulde. Så ville vi endelig vide, at selvom der findes simple løsninger til svære gåder, kan de end ikke principielt findes ad rationel vej. Vi kan kun håbe på heldet. Fremtidens matematiske vil ikke længere være nødvendighed, men kompleksitet og tilfældighed, og troen på den grænseløse erkendelse vil gå tabt endnu engang. Men foreløbig er der ingen, som har bevist hverken det ene eller det andet, for selvom der findes et utal af NP-komplette problemer, har man ikke fundet nogen effektiv løsningsformel for et eneste af dem.

Universet som laptop

Man kan forsøge at komme med et estimat af, hvad man i det hele taget kan regne på, hvis hele universets masse og energi blev brugt til kun det formål. Vi kan med andre ord forestille os, at vi prøver at finde den optimale processorhastighed af hele universet. Til at starte med må vi kende massen af universet. Ifølge Big Bang-teoriens beregninger, som også medregner den del af universet, som ikke kan ses, er den totale masse af universet $6 \cdot 10^{52}$ kg. Massen kan så omregnes til universets totale energi via Einsteins formel $E = mc^2$. Men da vi er interesseret i antallet af bits, som universet kan be-