Credit cards, computationele complexiteit en consistentie uitspraken*

Joost J. Joosten

23 mei 2006

Praag en bier Sinds enkele maanden werk ik als post-doc aan de Czech Academy of Sciences in Praag. Praag is een bijzonder mooie stad en het bier is er goed en goedkoop.

Ik kan er alleen nog steeds niet aan wennen om overdag bier te drinken tijdens de lunch om daarna weer gewoon aan het werk te gaan. Hier echter, hebben bijzonder sterke wiskundigen er geen enkel probleem mee om tijdens de lunch een halve liter bier te drinken om vervolgens, na een kopje koffie, weer moeilijke stellingen te gaan zitten bewijzen. Ik kan er niet aan wennen en heb de hoop opgegeven. Dat betekent dus niet dat ik geen stellingen meer bewijs. Nee, ik bedoel dat ik gewoon geen bier drink tijdens de lunch als mij nog grote rekenpartijen staan te wachten.

Maar ik wilde het niet alleen maar over bier hebben in dit stukje. Dat kan altijd nog. Ik wilde iets kwijt over een mooie stelling die ik laatst tegen kwam. Deze stelling ving mijn aandacht en een tijd lang heb ik het bewijs bestudeerd en enkele verwante resultaten bewezen. In dit stukje wil ik eigenlijk alleen maar de stelling formuleren en een beetje context creëren.

Complexiteit Sinds enkele maanden werk ik niet alleen in een nieuwe stad, waar de standaard maat van een biertje een halve liter is, maar werk ik ook aan een nieuw onderwerp: proof complexity. Proof complexity is een volwassen discipline op zichzelf maar heeft stiekem het ultieme doel om iets te zeggen over computationele complexiteit. En daar heeft iedereen in zijn of haar studententijd wel eens iets over gehoord. Ik geloof dat het probleem of $\mathsf{P} = \mathsf{NP}$ wel aan iedereen bekend is.

Wellicht ten overvloede nog even kort door de bocht, en niet al te precies, wat is P, wat is NP en waarom zouden ze wel of niet gelijk moeten zijn. Zowel P

*Tijdens mijn verblijf in Praag heb ik naast mijn loon van de Academy of Sciences of the Czech Republic, een financiële bijdrage van de Nederlandse Organisatie voor Wetenschappelijk Onderzoek (NWO) ontvangen.

als NP zijn zogeheten complexiteitsklassen. We zeggen dat een probleem in de klasse P is als het aantal rekenstappen dat nodig is om het probleem op te lossen hooguit polynomiaal is in de lengte van (de representatie van) het probleem.

Het is gelukkig niet nodig heel erg precies te zijn met het zeggen wat je met een rekenstap bedoelt. Dit kan zijn een elementaire operatie op een abstracte Turing machine, of een registerbewerking in een CPU van een Macintosh computer. Dit maakt voor de klasse P helemaal niets uit.

Onlangs is bewezen dat het probleem om te bepalen of een natuurlijk getal een priemgetal is of niet in P is. Als je niet slim te werk gaat moet je voor een getal n voor alle getallen m kleiner dan n controleren of m een deler is van n. Dat zijn er dus n veel. En omdat we n met ongeveer $\log(n)$ symbolen weergeven, zijn dit dus ongeveer exponentieel veel rekenstappen en dus is dit algoritme niet in P.

Een probleem is in de klasse NP als we in polynomiale tijd kunnen verifiëren dat een oplossing inderdaad een oplossing is. In dit geval wordt er dus niets gezegd over hoe moeilijk het proces van het zoeken is. Nee, het gaat er alleen maar om dat er in polynomiale tijd gecontroleerd kan worden dat een oplossing inderdaad een oplossing is. De N in NP staat voor non-deterministisch. Een non-deterministische machine gokt random de goeie gok en controleert dan dat dit een oplossing is. Inderdaad, op zo'n non-deterministische machine kunnen we een NP probleem in polynomiale tijd oplossen.

Men is het er over het algemeen redelijk over eens dat P de klasse problemen is die in redelijke tijd valt op te lossen. Evenzo is NP de klasse van problemen waarvan de oplossingen in redelijke tijd vallen te verifiëren. Voor moeilijke NP problemen lijkt er echter geen slimmere strategie mogelijk dan al de exponentieel veel (immers verifieerbaar in P) non-deterministische gokken te proberen. Uit de definitie volgt direct dat $\mathsf{P} \subseteq \mathsf{NP}$. Of er ook gelijkheid geldt, dat is de grote vraag.

Goed, ik moet een beetje vaart maken, anders kom ik nooit toe aan die mooie stelling die ik wilde uiteenzetten. Het is wel goed om nog even te vermelden dat bijna iedereen gelooft dat $\mathsf{N} \neq \mathsf{NP}$. Dit geloof is zelfs zo sterk dat verscheidene beschermingsprotocollen op essentiële wijze hierop berusten. Als nu bewezen zou worden dat $\mathsf{P} = \mathsf{NP}$ dan zouden bijvoorbeeld betalingen via internet met creditcards wellicht niet zo goed beschermd zijn als we nu denken dat ze zijn.

Een belangrijk probleem binnen complexiteitstheorie is het *vervulbaarheids probleem* beter bekend als SAT, van *satisfiability*. Het is heel simpel. Gegeven een propositioneel logische formule, kan ik de waarden voor de variabelen zodanig kiezen dat de formule onder die keuze waar is. Het is duidelijk dat dit probleem NP is: gegeven een keuze, dan is het snel te controleren dat dit een goede keuze is. Het probleem SAT is niet alleen NP, maar het is ook NP-volledig zoals dat heet. Dit betekent onder andere dat als we SAT in polynomiale tijd kunnen oplossen, dan kunnen we elk NP probleem in polynomiale tijd oplossen.

Een laatste complexiteitsklasse die ik moet introduceren is die van CoNP. In zekere zin zijn CoNP problemen complementen van NP problemen. Het belangrijkste voorbeeld voor ons van een CoNP-volledig probleem is TAUT, het probleem of een propositioneel logische formule een tautologie is, d.w.z., of die

formule altijd waar is. Het is duidelijk dat een formule een tautologie is dan en slechts dan, als de negatie niet vervulbaar is. Een even belangrijke vraag als $\mathsf{P} = \mathsf{NP}$ is de vraag of $\mathsf{NP} = \mathsf{CoNP}$. Omdat P gesloten is onder complementatie is het duidelijk dat indien $\mathsf{NP} \neq \mathsf{CoNP}$, noodzakelijkerwijs ook $\mathsf{P} \neq \mathsf{NP}$. Dat TAUT CoNP volledig is, betekent onder andere dat indien je kunt laten zien dat TAUT in NP is, dat dan $\mathsf{NP} = \mathsf{CoNP}$.

Ik zal nu wat meer aandacht besteden aan de vraag of $\mathsf{NP} = \mathsf{CoNP}$. Daartoe enige uitweiding over ons CoNP-volledig probleem TAUT. Hoe kun je in het algemeen inzien dat een propositioneel logische formule een tautologie is? Wel, je kan een waarheidstafel maken. Echter, deze worden echter nog al snel onhandelbaar (exponentieel in het aantal variabelen) groot.

In de praktijk geven we vaak een *bewijs* van een formule. Dit bewijs kan zijn in natuurlijke deductie, in Fitch stijl of welk systeem dan ook. We noemen een bewijssysteem P *super* indien er een natuurlijk getal l is zodat iedere tautologie τ een bewijs heeft met lengte $\leq |\tau|^l$. Hier is $|\tau|$ de lengte van τ. Het is heel eenvoudig in te zien (probeer maar) dat, indien er een super bewijssysteem P bestaat, dat dan $\mathsf{NP} = \mathsf{CoNP}$. Het is een niet al te moeilijke stelling van Cook en Reckhow dat het omgekeerde ook geldt. Dat is, indien $\mathsf{NP} = \mathsf{CoNP}$, dan bestaat er een super bewijssysteem.

Maar wat is dat eigenlijk in zijn algemeenheid, een bewijssysteem? Als we naar bewijssystemen kijken die we kennen, dan is de belangrijkste eigenschap, dat we in polynomiale tijd kunnen checken dat een bewijs inderdaad een bewijs is, en dat we de conclusie van het bewijs kunnen aflezen. De algemeen aanvaarde definitie van een bewijssysteem is een functie van de verzameling van alle syntactische strings (over het alfabet van b.v. propositielogica) die als bereik alle tautologieën heeft en die bovendien in polynomiale tijd te berekenen is.

Voordat ik naar de consistentie uitspraken ga, moet ik nog één notie behandelen. En dat is de notie van een optimaal bewijssysteem (**opps**, van optimal propositional proof system). Een bewijssysteem P is *optimaal* als voor iedere tautologie de lengte van het kortste bewijs in P hooguit polynomiaal veel langer is dan de lengte van het kortste bewijs in een willekeurig ander bewijssysteem Q. (Per Q mag er een verschillend polynoom gekozen worden.)

Het is vrij eenvoudig in te zien dat, indien P super is, dat dan P ook optimaal is. Indien men dus kan bewijzen dat er geen optimaal bewijssysteem bestaat, dan is dus ook $\mathsf{N} \neq \mathsf{NP}$ bekend. Goed, nu hebben we alles qua propositielogica gehad.

Consistentie uitspraken In de vorige paragraaf had ik het over propositielogica en over bewijssystemen P en Q. In deze paragraaf zal ik spreken over rekenkundige theorieën als T en S. Rekenkundige theoriën zijn theorieën die praten over getallen en die hierover kunnen redeneren. Dit kan op directe wijze zoals bij theorieën als PA, Peano Rekenkunde, maar ook op indirecte wijze zoals bijvoorbeeld in ZFC, Zermelo Fraenkel verzamenlingenleer met het keuze axioma, waar we de getallen kunnen definiëren/interpreteren. Het enige belangrijke wat ik in deze paragraaf moet doen is definiëren wat een *snelle consistentie*

bewijzer (ook wel *facop*, van fast consistency prover) is.

Een theorie T heet *consistent* als T niet bewijst dat $0 = 1$. Een theorie T heet consistent tot n indien alle bewijzen in T die niet meer dan n symbolen bevatten geen bewijzen zijn van $0 = 1$. We schrijven in dit geval $\mathsf{Con}_T(n)$. Merk op dat deze consistentie uitspraken over syntax gaan: we zeggen dat geen enkele string van symbolen, welke een geldig bewijs is, als conclusie de string "0=1"heeft. Van de grote wiskundige/logicus Kurt Gödel hebben we geleerd hoe we deze syntactische uitspraken in rekenkundige theoriën kunnen coderen. Ik zal ook $\mathsf{Con}_T(n)$ schrijven voor deze gecodeerde uitspraken. Dit is dus een uitspraak over getallen die waar is, indien en slechts indien, er inderdaad geen bewijs bestaat van 0=1 met niet meer dan n symbolen.

Het is bijzonder belangrijk om nu te vermelden dat we niet alle theorieën beschouwen in dit artikeltje. Anders, en dit is iets wat ik bewezen heb, is de stelling die ik zo ga noemen gewoonweg niet waar. Van nu af aan zullen alle theoriëen die we beschouwen consistent zijn, en een verzameling axioma's hebben die in polynomiale tijd herkenbaar zijn. Dit wil zeggen, dat je in polynomiale tijd kunt beslissen of een formule een axioma is van de theorie in kwestie, of niet. Nu is het zo, dat alle theorieën die we in het dagelijks wiskundig leven tegenkomen inderdaad een verzameling axioma's hebben die in polynomiale tijd herkenbaar zijn. Ik zal bovendien eisen dat alle theorieën een zekere minimale hoeveelheid standaard rekenkunde bevatten. Voortaan, als ik b.v. schrijf "voor alle T", dan bedoel ik dus een kwantificatie over de zojuist beschreven theorieën.

De centrale definitie is nu de volgende. Een *snelle consistentie bewijzer* of ook wel een **facop**, is een theorie S zodat voor elke andere theorie T, er een natuurlijk getal l bestaat zó dat de bewijzen van $\mathsf{Con}_T(n)$ in S niet meer dan n^l symbolen bevatten.

De stelling De stelling die ik zo wonderlijk vind kan nu heel eenvoudig geformuleerd worden.

Stelling Er bestaat een opps $\Leftrightarrow$ Er bestaat een facop.

Ik wil hier niks over het bewijs kwijt. In plaats daarvan wil ik iets zeggen over waarom ik mij zo verbaasde over deze stelling.

In de eerste plaats vind ik de stelling mooi omdat er twee volledig verschillende gebieden aan elkaar gerelateerd worden. Een opps is iets dat over propositielogica gaat, iets wat je bijna met je handen aan kunt raken. Facops aan de andere kant gaan over rekenkundige theorieën. En dit kan dus gaan over ziekmakend grote kardinaalgetallen, of over vreselijk sterke rekenkundes.

In de tweede plaats, vind ik de stelling mooi omdat ik gewoonweg niet in het bestaan van facops geloof. Het is duidelijk dat elke theorie S inderdaad voor elke theorie T de uitspraken $\mathsf{Con}_T(n)$ kan bewijzen. Immers, S hoeft alleen maar alle mogelijke bewijzen met niet meer dan n symbolen af te gaan en te controleren dat geen van deze bewijzen een bewijs van $0 = 1$ is. Over een eindig alfabet zijn er maar exponentieel veel van deze bewijzen te checken. En wegens onze aannames over S en T kan S ook in polynomiale tijd verifiëren dat

axioma's van T inderdaad axioma's van T zijn. Dit levert dus een exponentiële bovengrens op de lengtes van de bewijzen van $\mathsf{Con}_T(n)$ in S. Maar, hoe kan S nu zo slim zijn dat hij niet alle mogelijkheden af hoeft te gaan. En dan moeten we dus bedenken dat S dit voor *alle* T op slimme wijze moet kunnen. Maar T kan altijd veel en veel sterker zijn dan S. Dus, het lijkt me dat dit gewoon niet moet kunnen.

Meer bier Goed, ik geloof dus niet in het bestaan van facops. Kan ik dat bewijzen dat ze niet bestaan? Als ik dat zou kunnen, dan zou ik dus bewijzen dat er geen opps bestaat. Dientengevolge zou ik dus een bewijs hebben dat er geen super bewijssysteem bestaat en dat dus $\mathsf{CoNP} \neq \mathsf{NP}$ en dus ook $\mathsf{P} \neq \mathsf{NP}$. Dit levert mij dan terloops één miljoen dollar aan prijzengeld op.

Ik geloof dat er in de wereld niemand is die hoop heeft dat hij of zij op korte termijn $\mathsf{P} \neq \mathsf{NP}$ zal bewijzen. Sterker nog, zeer respectabele specialisten in dit gebied raden iedereen af om direct aan dit probleem te werken. Dit wordt als pure tijdverlies beschouwd. Het probleem is gewoon te moeilijk en niemand heeft enig idee hoe het aan te pakken. Leuk hoor, sinds vier maanden ben ik dus werkzaam in dit gebied. Ik denk dat ik nog maar eens een biertje ga drinken.