Telefoonverbindingen op het eiland Hau

R. Geel
NLO Ubbo Emmius, Groningen

Samenvatting

We bespreken een onderwerp uit de grafentheorie, dat geschikt lesmateriaal blijkt op te leveren voor (onder andere) Wiskunde A.

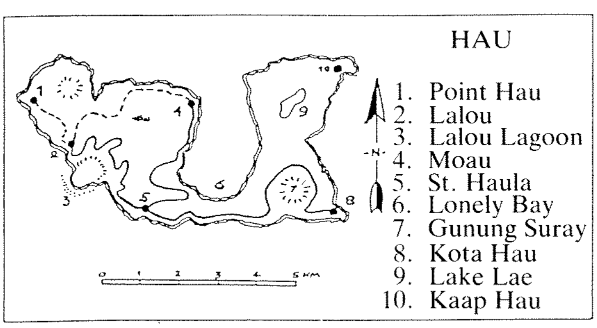
Aan de hand van een min of meer realistisch probleem (er dient met een minimum aan kosten een telefoonnet te worden aangelegd) bespreken we in dit artikel een onderwerp uit de grafentheorie. Het op te lossen probleem wordt in *'Telefoonverbindingen op het eiland Hau'* geformuleerd, waarna we vervolgens een aantal oplossingsmethoden bespreken en enige achtergrondinformatie verschaffen.
Het hier behandelde onderwerp levert niet alleen geschikt lesmateriaal op voor VWO/NLO (bv. een aanvullende opdracht in het kader van Wiskunde A/grafentheorie, waarbij de tekst van *'Telefoonverbindingen'* zonder meer als leerlingentekst kan fungeren), maar leent zich ook voor behandeling op MAVO/HAVO-niveau. In het laatste geval dient de leraar, na kennis te hebben genomen van de verschillende oplossingsmethoden, de tekst naar eigen smaak uit te breiden met aanwijzingen, 'sturende' opdrachten, etc.

Het besproken probleem leidt op natuurlijke wijze tot allerlei probleemoplossende activiteiten van leerlingen/studenten en dwingt tot een nauwgezet formuleren van oplossingsmethoden, aldus een bijdrage leverend tot het 'leren algoritmiseren'.
Daarnaast ontstaat bij het concreet ten uitvoer brengen van de verschillende oplossingsmethoden al snel de behoefte om een computer in te schakelen, zodat de stap naar automatische gegevensverwerking voor de hand ligt. Met behulp van het in *'Het algoritme van Prim in matrixvorm'* beschreven algoritme kan de leraar vrij eenvoudig een programma voor automatische gegevensverwerking schrijven, waarmee de leerlingen dan kunnen werken (overigens is er ook een listing van een BASIC-programma op aanvraag verkrijgbaar).

Om de lezer enigszins een indruk te geven van de prestaties waartoe leerlingen van de bovenbouw HAVO/VWO in staat zijn, wanneer ze met de 'kale' leerlingentekst van *'Telefoonverbindingen'* worden geconfronteerd, doen we in *'Ervaringen met studenten van Ubbo Emmius'* van dit artikel tenslotte verslag van een experiment, dat werd uitgevoerd met eerstejaars wiskundestudenten van de NLO Ubbo Emmius.

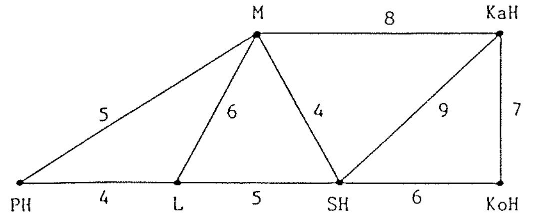
Telefoonverbindingen op het eiland Hau

In de Stille Oceaan ligt het (denkbeeldige) eilandje Hau, waarvan we hieronder een kaart afdrukken:



(Het wegennet op dit eiland was al eerder onderwerp van bespreking, zie [1].)

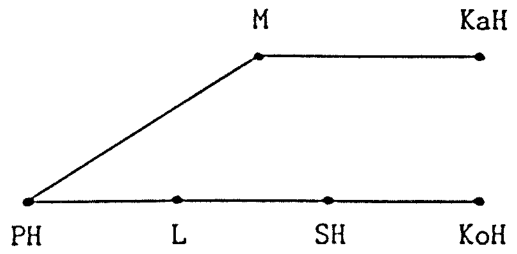
Omdat de toeristenindustrie op Hau langzaam op gang begint te komen, heeft het Eilandbestuur besloten telefoonverbindingen te laten aanleggen tussen de zes plaatsen op het eiland, te weten Point Hau, Lalou, Moau, St. Haula, Kota Hau en Kaap Hau. De vraag is nu, hoe dit telefoonnet het beste kan worden aangelegd.
De mogelijkheden voor het leggen van de telefoonkabels worden weergegeven in het volgende schema:



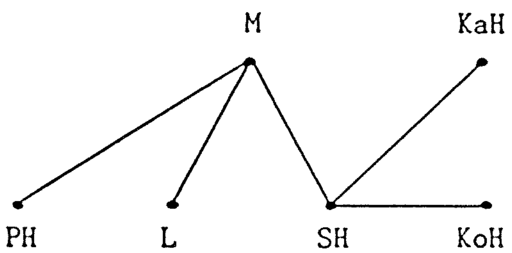
Figuur 1

We herkennen in dit schema een graaf met zes punten (de zes steden) en negen verbindingslijnstukken (ook wel takken genaamd). Een tak tussen twee punten geeft aan, dat het mogelijk is om een directe kabelverbinding te leggen tussen de beide steden; het getal naast een tak stelt de kosten voor (uitgedrukt in een bepaalde geldseenheid) die de aanleg van deze verbinding met zich meebrengt.

Het zal duidelijk zijn, dat voor een goed functionerend telefoonnet niet per se alle mogelijke kabelverbindingen in de graaf van figuur 1 gelegd hoeven te worden: van belang is slechts, dat elk van de zes steden (direct of indirect) verbonden is met elk van de andere steden. Aan deze eis voldoen bijvoorbeeld de beide volgende netten:



Figuur 2a



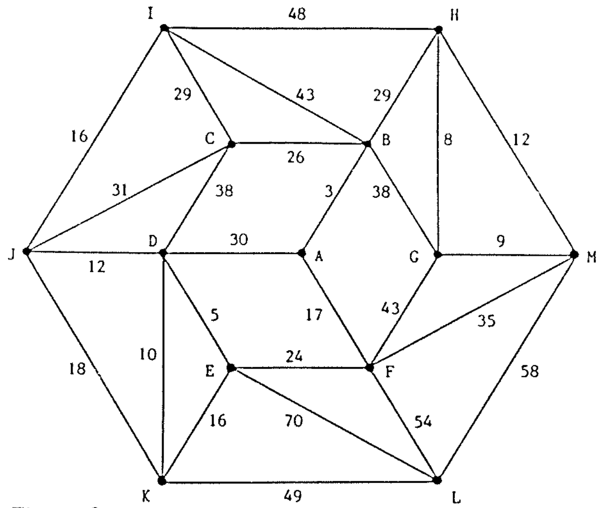
Figuur 2b

De aanlegkosten voor deze netten bedragen 28 resp. 30, zoals men eenvoudig nagaat met behulp van figuur 1.

Het Eilandbestuur ziet zich nu geplaatst voor de volgende vraag:

> Hoe moeten de kabelverbindingen tussen de zes steden gelegd worden, opdat de aanlegkosten van het telefoonnet zo laag mogelijk zijn?

Hierbij is men niet zozeer geïnteresseerd in het *antwoord* op deze vraag (immers, met wat puzzelen in figuur 1 is een oplossing snel gevonden), maar meer in een *systematische oplossingsmethode* die ook in een ingewikkelder situatie, zoals bijvoorbeeld in figuur 3, tot een oplossing leidt.



Figuur 3

Enkele opmerkingen vooraf

In de volgende paragrafen bespreken we een aantal algoritmen voor de bepaling van een goedkoopste telefoonnet. Bij het lezen van deze oplossingsmethoden zult u ongetwijfeld meer dan eens denken: "Dat ligt voor de hand, dat had ik zelf ook kunnen bedenken." Misschien gaat u zelfs verder en denkt u: "Dat zouden mijn leerlingen/studenten ook moeten kunnen bedenken." Teneinde straks in alle objectiviteit vast te kunnen stellen:

a. of de besproken algoritmen inderdaad voor de hand liggend zijn;
b. welke aanwijzingen, deelproblemen, 'sturende' opgaven etc. een leerlingentekst eventueel zou moeten bevatten, voordat u redelijkerwijs van uw leerlingen/studenten mag verwachten dat ze bepaalde algoritmen vinden;

doet u er goed aan om nu eerst, voordat u verder leest, zelf één of meer oplossingsmethoden te bedenken voor het in de vorige paragraaf gestelde probleem.
Waar in het vervolg over 'grafen' wordt gesproken, bedoelen we *enkelvoudige* grafen.
Dit zijn grafen waarin:

1. geen enkel punt door een tak met zichzelf verbonden is;
2. elk tweetal verschillende punten door hoogstens één tak onderling verbonden is.

Twee algoritmen

We beschouwen de graaf van figuur 1. Een dergelijke graaf, waarin elk punt direct of indirect met elk ander punt verbonden is, noemt men een *samenhangende* graaf.
Zoals al eerder werd opgemerkt, hoeft niet elke tak van de graaf te worden gebruikt bij de aanleg van het telefoonnet: we kunnen rustig takken weglaten, als we er maar voor zorgen dat de samenhang van de graaf niet verloren gaat. Een voor de hand liggende methode om een goedkoopste telefoonnet te bepalen, is dan ook de volgende:

> Laat uit de graaf van figuur 1 één voor één takken weg zonder de samenhang aan te tasten.
> Geef hierbij voorrang aan de 'dure' takken.

Door te letten op de zogenaamde *circuits* in de graaf kunnen we eenvoudig nagaan of er nog een tak weggelaten kan worden (en er dus nog bezuinigd kan worden op de aanlegkosten van het telefoonnet) zonder dat de samenhang van de graaf verbroken wordt.

Definitie
Onder een circuit in een graaf verstaan we een 'rondreis' in de graaf, waarbij geen punt of tak meer dan eenmaal gepasseerd wordt (alleen het begin- en eindpunt wordt tweemaal aangedaan).

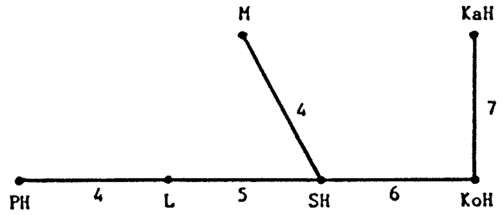
Voorbeelden van circuits in de graaf van figuur 1 zijn o.a. M → L → SH → M en KoH → SH → M → KaH → KoH.

Het zal duidelijk zijn dat, zolang de graaf nog een circuit bevat, het mogelijk is om één willekeurige tak van dat circuit weg te laten zonder dat de samenhang van de graaf verbroken wordt: de punten van het circuit blijven onderling verbonden door de resterende takken van het circuit. Zodra de graaf echter geen enkel circuit meer bevat, gaat met het weglaten van nog een tak de samenhang onherroepelijk verloren.
Conclusie: een goedkoopste telefoonnet heeft in ieder geval de gedaante van een samenhangende graaf zonder circuits.
Een samenhangende graaf zonder circuits wordt een *boom* genoemd.
De grafen in figuur 2a, b zijn voorbeelden van bomen.

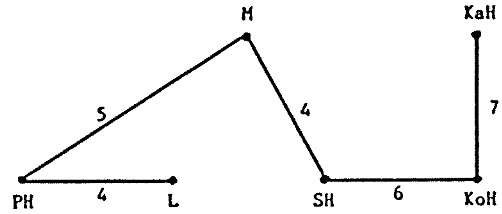
Hiermee zijn we toegekomen aan de formulering van een eerste algoritme voor de bepaling van een goedkoopste telefoonnet.

Algoritme 1 (afbreek-algoritme)
1. Laat van de takken in de graaf, die deel uitmaken van een circuit, de duurste tak weg (maak, indien er meerdere takken in aanmerking komen, een keuze).
2. Herhaal stap 1 totdat de graaf geen circuits meer bevat.
3. De dan ontstane graaf (een boom) stelt een goedkoopste telefoonnet voor.

Toepassing van dit algoritme op de graaf van figuur 1 leidt tot de verwijdering van de takken KaH-SH (9), KaH-M (8), L-M (6) en ófwel M-PH (5) ófwel L-SH (5). De volgende twee telefoonnetten (aanlegkosten: 26) zijn dus het goedkoopst:



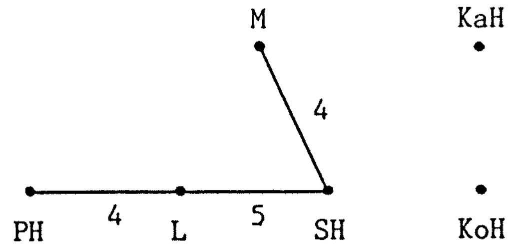
Figuur 4a



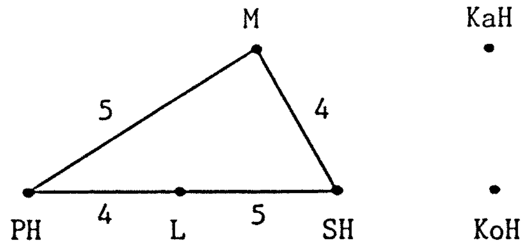
Figuur 4b

In plaats van de graaf van figuur 1 tak voor tak af te breken, is het natuurlijk ook mogelijk om uit te gaan van de zes punten (nog zonder verbindingslijnstukken) en het telefoonnet vervolgens tak voor tak op te bouwen. Een logische strategie is dan de volgende:

> Voeg één voor één takken toe totdat een samenhangende graaf is ontstaan.
> Geef hierbij voorrang aan de 'goedkope' takken, maar voeg geen overbodige takken toe.
> Het begrip 'overbodige tak' verduidelijken we met een voorbeeld.
> Veronderstel, dat we aan de zes punten van de graaf in figuur 1 achtereenvolgens de takken L-PH (4), M-SH (4) en L-SH (5) hebben toegevoegd (zie figuur 5a).



Figuur 5a



Figuur 5b

Lettend op de kosten zouden we dan vervolgens de goedkoopste resterende tak M-PH (5) moeten toevoegen. Dit is echter een overbodige tak, omdat de punten M en PH al indirect met elkaar verbonden zijn. De tak M-PH wordt dus overgeslagen en de volgende tak die wordt toegevoegd, is KoH-SH (6).
Overbodige takken kan men blijkbaar herkennen aan het feit, dat zij met reeds eerder toegevoegde takken een circuit vormen (zie figuur 5b).
Met deze overweging laat zich een tweede algoritme voor de bepaling van een goedkoopste telefoonnet formuleren.

Algoritme 2 (opbouw-algoritme)
1. Teken de punten van de graaf.
2. Voeg aan deze punten de goedkoopste tak toe, die geen circuit vormt met reeds eerder toegevoegde takken (maak, indien er meerdere takken in aanmerking komen, een keuze).
3. Herhaal stap 2 totdat alle punten van de graaf direct of indirect met elkaar verbonden zijn.
4. De dan ontstane graaf (een boom) stelt een goedkoopste telefoonnet voor.

Men overtuige zich ervan, dat dit algoritme achtereenvolgens de takken L-PH (4), M-SH (4), ófwel L-SH (5) ófwel M-PH (5), KoH-SH (6) en KaH-KoH (7) toevoegt en aldus leidt tot de beide bomen van figuur 4a, b.
Algoritme 2 staat in de literatuur bekend als het algoritme van Kruskal.
Voor een bewijs, dat dit algoritme inderdaad tot een goedkoopste telefoonnet leidt, verwijzen we naar [2].

Een andere benadering: alle mogelijkheden nagaan

Het in *'Telefoonverbindingen'* beschreven probleem roept op natuurlijke wijze allerlei vragen op, die ieder op hun beurt weer kunnen leiden tot probleemoplossende activiteiten van leerlingen/studenten. Op enkele van deze vragen zullen we thans nader ingaan: we verschaffen antwoorden en verstrekken enige achtergrondinformatie.

In de eerste plaats kunnen we ons afvragen:

> Hoeveel directe kabelverbindingen moeten er *minstens* gelegd worden, opdat elk van de zes steden (direct of indirect) verbonden is met elk van de andere steden?

of, in grafentheoretische bewoordingen:

> Hoe groot is het minimale aantal takken in een samenhangende graaf met zes punten?

We laten het aan de lezer over om voor zichzelf na te gaan, dat er minimaal $n-1$ geschikt gekozen verbindingslijnstukken nodig zijn om n punten met elkaar te verbinden. Het minimale aantal takken in een samenhangende graaf met zes punten bedraagt dus vijf (zie bv. de beide grafen in figuur 2a, b).
Een goedkoopste telefoonnet bestaat dus uit *minstens* vijf verbindingslijnstukken. Zou een goedkoopste telefoonnet eventueel uit zes of meer verbindingslijnstukken kunnen bestaan? Door wat te spelen in de graaf van figuur 1 krijgen we al snel het volgende vermoeden:

> In een samenhangende graaf met zes punten en n takken ($n>5$) kunnen we altijd $n-5$ takken weglaten zonder de samenhang van de graaf te verstoren.

Ook nu laten we het weer aan de lezer over om voor zichzelf te beredeneren, dat dit vermoeden correct is.
Met behulp van het voorgaande is eenvoudig in te zien, dat een telefoonnet met de laagste aanlegkosten uit *precies* vijf verbindingslijnstukken moet bestaan. Immers, uit een net met meer dan vijf verbindingslijnstukken zouden één of meer lijnstukken weggelaten kunnen worden zonder de samenhang te verbreken: zo'n net kan dus nooit het goedkoopste zijn.

Het volgende algoritme voor de bepaling van een goedkoopste telefoonnet dringt zich thans aan ons op:

Algoritme 3 (opsom-algoritme)
1. Bepaal alle samenhangende deelgrafen van de graaf in figuur 1, die bestaan uit:
   a. de zes genoemde punten;
   b. vijf van de negen getekende takken.
2. Bepaal van de onder 1 gevonden deelgrafen degene met de laagste aanlegkosten.

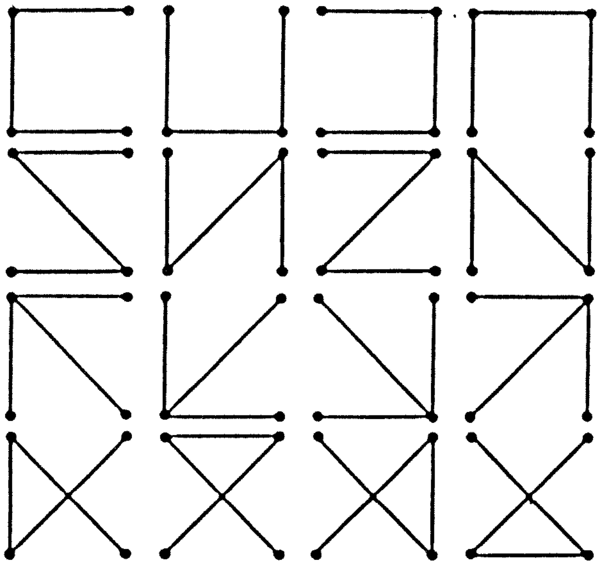
Een deelgraaf van het onder punt 1 beschreven type wordt in het Engelse taalgebied een 'spanning tree' genoemd (letterlijk: een boom die de graaf opspant, d.w.z. alle punten van de graaf bevat). Wij zullen een dergelijke deelgraaf voortaan een *geraamte* noemen. Zo zijn de grafen in figuur 2a, b en in figuur 4a, b geraamten van de graaf in figuur 1.
De bepaling van een goedkoopste telefoonnet komt dan neer op de bepaling van een kleinste geraamte (Engels: 'minimal spanning tree'), d.w.z. een geraamte waarbij de som van de getallen naast de takken zo klein mogelijk is. De grafen in figuur 4a, b zijn de kleinste geraamten van de graaf in figuur 1.

Een interessante vraag naar aanleiding van algoritme 3 is nu: hoeveel geraamten bevat een gegeven (samenhangende) graaf?
Voor een *volledige* graaf (d.w.z. een graaf waarin elk tweetal punten onderling door een tak verbonden is) werd het antwoord op deze vraag in 1889 gegeven door Cayley. Alvorens dit antwoord te openbaren, verstrekken we eerst wat cijfermateriaal. Als n het aantal punten van de (volledige) graaf voorstelt en S(n) het aantal geraamten, dan kunnen we de volgende tabel opstellen:

n	2	3	4	5	6	....	10
S(n)	1	3	16	125	1296	....	10^8

Vraag: Ziet u een verband tussen n en S(n)?

Voor een volledige graaf met vier punten staan op de volgende pagina alle zestien geraamten afgebeeld.
Cayley toonde aan, dat het aantal geraamten S(n) van een volledige graaf met n punten gegeven wordt door $S(n) = n^{n-2}$. Voor een tweetal bewijzen van deze formule verwijzen we naar [2].



Het aantal geraamten van een *niet-volledige* graaf kan worden bepaald met behulp van ..... matrixrekening! Hiertoe definiëren we bij een graaf met n punten $P_1, P_2 \ldots, P_n$ op de volgende wijze een $n \times n$-matrix $M = (m_{ij})$:

$$\begin{cases} m_{ii} = \varrho(P_i) \\ m_{ij} = -1 \text{ als } P_i \text{ en } P_j \text{ door een tak verbonden zijn } (i \neq j) \\ m_{ij} = 0 \text{ als } P_i \text{ en } P_j \text{ niet door een tak verbonden zijn } (i \neq j) \end{cases}$$

waarbij $\varrho(P_i)$ de *valentie* van het punt P_i voorstelt, d.w.z. het aantal takken dat in P_i ontspringt.
Voor de graaf van figuur 1 heeft de matrix M bijvoorbeeld de volgende gedaante ($KoH = P_1$, $KaH = P_2$, etc.):

$$\begin{array}{c} \\ \text{KoH} \\ \text{KaH} \\ \text{M} \\ \text{PH} \\ \text{L} \\ \text{SH} \end{array} \begin{array}{c} \begin{array}{cccccc} \text{KoH} & \text{KaH} & \text{M} & \text{PH} & \text{L} & \text{SH} \end{array} \\ \begin{pmatrix} 2 & -1 & 0 & 0 & 0 & -1 \\ -1 & 3 & -1 & 0 & 0 & -1 \\ 0 & -1 & 4 & -1 & -1 & -1 \\ 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & -1 & 3 & -1 \\ -1 & -1 & -1 & 0 & -1 & 4 \end{pmatrix} \end{array}$$

Het aantal geraamten kan nu worden berekend door gebruik te maken van de volgende stelling:

Stelling
Laat G een enkelvoudige, samenhangende graaf zijn met n punten $P_1, P_2, \ldots P_n$ en laat de $n \times n$-matrix M gedefinieerd zijn op de hiervoor beschreven wijze. Laat verder m_{ij} een willekeurig element zijn van M. Dan is het aantal geraamten van G gelijk aan de cofactor van m_{ij}, d.w.z. gelijk aan:

$$(-1)^{i+j}\,|M_{ij}|,$$

waarbij $|M_{ij}|$ de determinant voorstelt van de $(n-1) \times (n-1)$-matrix M_{ij} welke uit M ontstaat door de i^e rij en de j^e kolom te schrappen.

Bewijs
Voor een bewijs van deze stelling verwijzen we naar [3].

Toepassing van de stelling (met bv. $i = j = 1$) op de graaf van figuur 1 leert, dat het aantal geraamten van deze graaf gelijk is aan 55.
Men gaat namelijk eenvoudig na, dat geldt:

$$|M_{11}| = \begin{vmatrix} 3 & -1 & 0 & 0 & -1 \\ -1 & 4 & -1 & -1 & -1 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & -1 & -1 & 3 & -1 \\ -1 & -1 & 0 & -1 & 4 \end{vmatrix} = 55.$$

Vier van deze vijfenvijftig geraamten staan afgebeeld in de figuren 2a, b en 4a, b.

Uit het voorgaande blijkt, dat algoritme 3 slechts van theoretisch belang is: zelfs voor een eenvoudige graaf als die van figuur 1 is het aantal geraamten reeds dusdanig groot, dat we het beter aan de computer kunnen overlaten om alle geraamten op te sporen. Als we echter toch onze toevlucht nemen tot de computer, dan bestaan er wel efficiëntere algoritmen voor de bepaling van een kleinste geraamte dan algoritme 3. Hierover gaat de volgende paragraaf.

Het algoritme van Prim

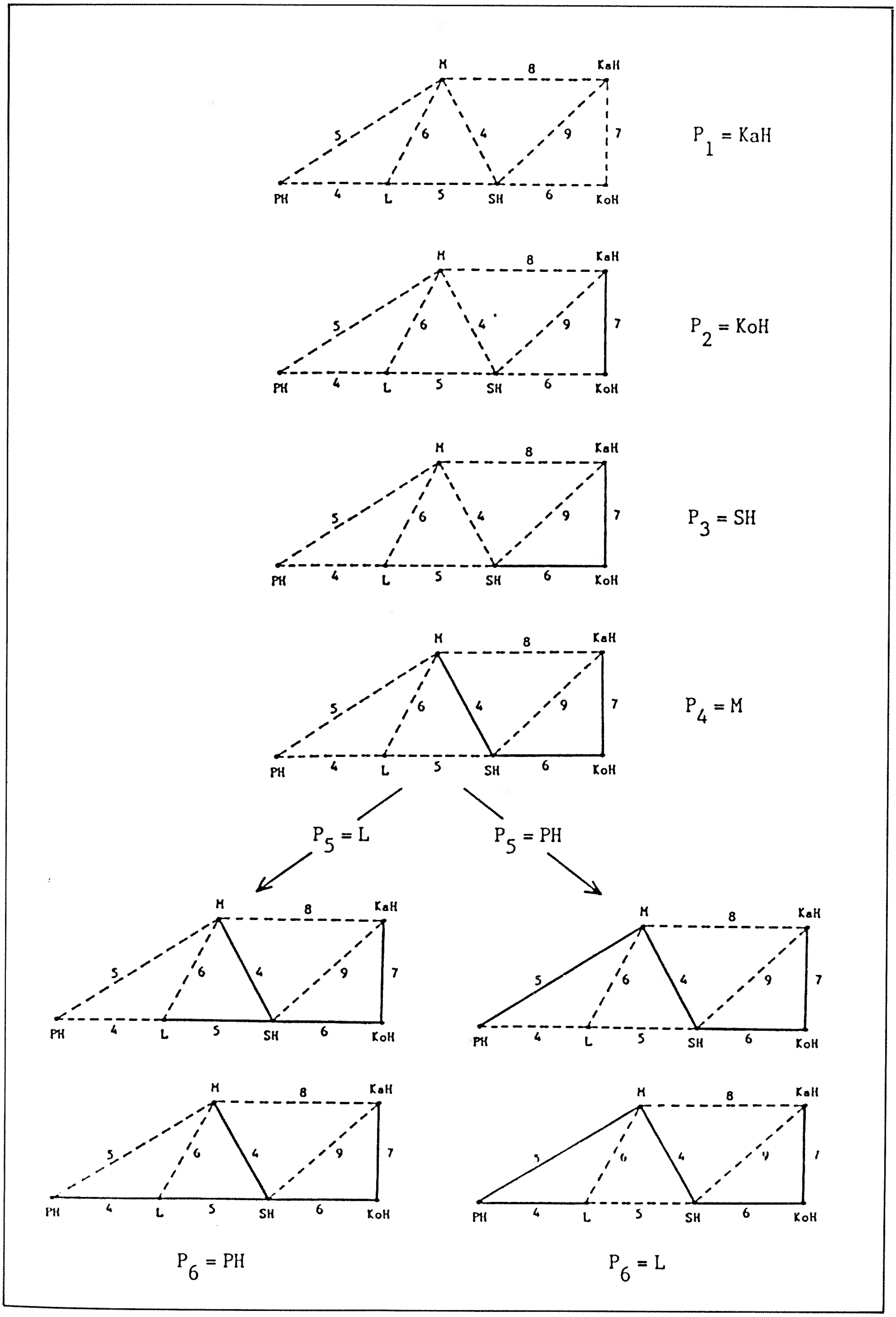
Het nu volgende algoritme voor de bepaling van een kleinste geraamte (in 1957 gepubliceerd door Prim) is vrij eenvoudig om te zetten in een efficiënt werkend computerprogramma en verdient dan ook de voorkeur wanneer we een goedkoopste telefoonnet willen bepalen met behulp van de computer.

Algoritme 4a (algoritme van Prim)
1. Kies een willekeurig punt en noem dit P_1.
2. Bepaal van de resterende punten het punt met de goedkoopste directe verbinding met P_1 (maak, indien er meerdere punten in aanmerking komen, een keuze). Noem dit punt P_2 en teken het bijbehorende goedkoopste verbindingslijnstuk.
3. Bepaal van de resterende punten het punt met de goedkoopste directe verbinding met P_1 of P_2 (maak, indien er meerdere punten in aanmerking komen, en keuze). Noem dit punt P_3 en teken het bijbehorende goedkoopste verbindingslijnstuk.
4. Bepaal van de resterende punten het punt met de goedkoopste directe verbinding met P_1, P_2 of P_3. Noem dit punt P_4, etc. Ga op deze wijze door tot alle punten verbonden zijn.

In figuur 6 (zie volgende pagina) voeren we het algoritme van Prim stap voor stap uit voor de graaf van figuur 1 (voor P_1 werd KaH gekozen). We vinden uiteindelijk weer de beide telefoonnetten van figuur 4a, b.
Men overtuige zich ervan, dat het resultaat van het algoritme van Prim *onafhankelijk* is van de keuze van het startpunt P_1, door het algoritme enkele malen toe te passen met verschillende startpunten.
Voor een bewijs, dat het algoritme van Prim inderdaad tot een kleinste geraamte leidt, verwijzen we naar [4].

Het algoritme van Prim in matrixvorm

Als we het algoritme van Prim willen vertalen in een computerprogramma, kunnen we het beste gebruik maken van de matrixversie van dit algoritme.



Figuur 6

Hiertoe zetten we de graaf van figuur 1 eerst om in een 6×6-matrix, dus bijvoorbeeld:

$$
\begin{array}{c}
\\ \text{KoH} \\ \text{KaH} \\ \text{M} \\ \text{PH} \\ \text{L} \\ \text{SH}
\end{array}
\begin{array}{c}
\begin{array}{cccccc} \text{KoH} & \text{KaH} & \text{M} & \text{PH} & \text{L} & \text{SH} \end{array} \\
\begin{pmatrix}
- & 7 & - & - & - & 6 \\
7 & - & 8 & - & - & 9 \\
- & 8 & - & 5 & 6 & 4 \\
- & - & 5 & - & 4 & - \\
- & - & 6 & 4 & - & 5 \\
6 & 9 & 4 & - & 5 & -
\end{pmatrix}
\end{array}
$$

Het algoritme luidt dan (let op de analogie met algoritme 4a!):

Algoritme 4b (algoritme van Prim: matrixversie)

1. Kies een willekeurige rij in de matrix, zeg de rij met rijnummer n_1.
2. Schrap de kolom met kolomnummer n_1. Bepaal van de nog niet geschrapte getallen in de rij n_1 het kleinste getal (maak, indien er meerdere getallen in aanmerking komen, een keuze). Laat n_2 het nummer zijn van de kolom waarin dit kleinste getal staat. Noteer de positie van het kleinste getal, dus (n_1, n_2).
3. Schrap de kolom met kolomnummer n_2. Bepaal van de nog niet geschrapte getalllen in de rijen n_1 en n_2 het kleinste getal (maak eventueel een keuze). Laat n_3 het nummer zijn van de kolom waarin dit kleinste getal staat. Noteer de positie van het kleinste getal, dus (n_1, n_3) of (n_2, n_3).
4. Schrap de kolom met kolomnummer n_3. Bepaal van de nog niet geschrapte getallen in de rijen n_1, n_2 en n_3 het kleinste getal, etc.
   Ga op deze wijze door tot alle kolommen geschrapt zijn.

We lichten algoritme 4b thans toe aan de hand van bovenstaande matrix.
Ons startpunt is de rij met rijnummer 2 (d.w.z. het punt KaH).

Kies de rij met nummer 2. Schrap kolom 2.
Het kleinste getal in rij 2 staat op positie (2,1).
Het eerste verbindingslijnstuk van het goedkoopste telefoonnet is dus KaH-KoH.

$$
\begin{array}{cc|cccccc}
 & & \text{KoH} & \text{KaH} & \text{M} & \text{PH} & \text{L} & \text{SH} \\
 & \text{KoH} & - & 7 & - & - & - & 6 \\
\longrightarrow & \text{KaH} & 7 & - & 8 & - & - & 9 \\
 & \text{M} & - & 8 & - & 5 & 6 & 4 \\
 & \text{PH} & - & - & 5 & - & 4 & - \\
 & \text{L} & - & - & 6 & 4 & - & 5 \\
 & \text{SH} & 6 & 9 & 4 & - & 5 & -
\end{array}
$$

Schrap kolom 1. Bekijk de rijen 1 en 2. Het kleinste getal in deze rijen staat op positie (1,6).
Het tweede verbindingslijnstuk van het goedkoopste telefoonnet is dus KoH-SH.

$$
\begin{array}{cc|cccccc}
 & & \text{KoH} & \text{KaH} & \text{M} & \text{PH} & \text{L} & \text{SH} \\
\longrightarrow & \text{KoH} & - & 7 & - & - & - & 6 \\
\longrightarrow & \text{KaH} & 7 & - & 8 & - & - & 9 \\
 & \text{M} & - & 8 & - & 5 & 6 & 4 \\
 & \text{PH} & - & - & 5 & - & 4 & - \\
 & \text{L} & - & - & 6 & 4 & - & 5 \\
 & \text{SH} & 6 & 9 & 4 & - & 5 & -
\end{array}
$$

Schrap kolom 6. Bekijk de rijen 1, 2 en 6. Het kleinste getal in deze rijen staat op positie (6,3).
Het derde verbindingslijnstuk van het goedkoopste telefoonnet is dus SH-M.

$$
\begin{array}{cc|cccccc}
 & & \text{KoH} & \text{KaH} & \text{M} & \text{PH} & \text{L} & \text{SH} \\
\longrightarrow & \text{KoH} & - & 7 & - & - & - & 6 \\
\longrightarrow & \text{KaH} & 7 & - & 8 & - & - & 9 \\
 & \text{M} & - & 8 & - & 5 & 6 & 4 \\
 & \text{PH} & - & - & 5 & - & 4 & - \\
 & \text{L} & - & - & 6 & 4 & - & 5 \\
\longrightarrow & \text{SH} & 6 & 9 & 4 & - & 5 & -
\end{array}
$$

Etc.

We laten het aan de lezer over om na te gaan, dat op deze wijze weer de beide telefoonnetten van figuur 4a, b ontstaan.
Voor een computerprogramma, dat gebruik maakt van de matrixversie van het algoritme van Prim, verwijzen we naar [5].

Ervaringen met studenten van de NLO Ubbo Emmius

Om een indruk te krijgen van de prestaties, waartoe leerlingen van de bovenbouw HAVO/VWO in staat zijn, wanneer ze met de 'kale' leerlingentekst van *'Telefoonverbindingen'* worden geconfronteerd, legden we deze tekst voor aan de eerstejaars wiskundestudenten (lichting 1985) van de NLO Ubbo Emmius.
De studenten (geen van allen bekend met grafentheoretische begrippen of methoden zoals we die bv. tegenkomen bij Wiskunde A) kregen de opdracht om schriftelijk één of meer algoritmen te formuleren voor de constructie van een goedkoopste telefoonnet en om vervolgens met behulp van één der gevonden algoritmen het goedkoopste telefoonnet te bepalen voor de grafen in de figuren 1 en 3.
Er werd gewerkt in groepjes van drie (soms vier) studenten met zoveel mogelijk gelijke vooropleiding (HAVO of VWO). Teneinde wederzijdse beïnvloeding te vermijden, werkten de verschillende groepjes strikt gescheiden.

	Vooropleiding		Totaal
	HAVO	VWO	
Aantal groepjes	10	6	16
Algoritme 1	–	–	–
Algoritme 2	3	1	4
Algoritme 3	1	–	1
Algoritme 4a	1	–	1
Algoritme 5	2	2	4
Geen algoritme	5	3	8

In de tabel staat vermeld welke algoritmen door de groepjes werden gevonden (sommige groepjes vonden

meer dan één algoritme). De nummers 1, 2, 3 en 4a van de algoritmen corresponderen met de eerder in dit artikel gebruikte nummering (algoritme 1 = afbreek-algoritme, etc.). Op algoritme 5 komen we straks nog terug.
Uit de tabel lezen we o.a. af, dat van de in totaal zestien groepjes er vier waren (drie met vooropleiding HAVO en één met vooropleiding VWO) die algoritme 2 toepasten.

De meeste groepjes vonden via één of ander 'algoritme' wel de correcte oplossingen (voor de graaf van figuur 3 luidt deze oplossing overigens IJDKL, DEFABC, BHGM met 208 als totale aanlegkosten).
Bij onze beoordeling hebben wij echter uitsluitend gelet op de wijze waarop de algoritmen onder woorden werden gebracht: een incorrect of onduidelijk geformuleerd algoritme werd in de tabel opgenomen als zijnde 'geen algoritme'.
Uit de tabel blijkt, dat voor de studenten de algoritmen 2 en 5 het meest voor de hand lagen.
Opvallend was verder, dat geen der groepjes aan de correctheid van de gevonden algoritmen leek te twijfelen: er werden althans geen pogingen gedaan om de algoritmen te rechtvaardigen (daar hadden wij overigens ook niet expliciet om gevraagd!). Toch lijkt het ons zinvol om in klassikaal verband nader in te gaan op dit soort zaken, zonder dat dit nu direct hoeft te leiden tot een volledig bewijs van de correctheid van een algoritme. We lichten dit laatste toe aan de hand van het bij de studenten zo populaire algoritme 5.

Algoritme 5
1. Kies vanuit elk punt in de graaf de goedkoopste tak. Op deze wijze ontstaan één of meer bomen.
2. Verbind deze bomen onderling zo goedkoop mogelijk tot een geraamte.

Passen we algoritme 5 bijvoorbeeld toe op de graaf van figuur 3, dan ontstaan in eerste instantie de drie bomen IJDKL, DE en FABC en HGM. De takken EF en BH vormen vervolgens de goedkoopste verbinding tussen deze bomen.
De eerste stap van algoritme 5 is ook voor een ingewikkelde graaf (zoals die van figuur 3) nog eenvoudig uitvoerbaar, zeker als we dit vergelijken met de algoritmen 1 en 2, waar de takken eerst moeten worden gerangschikt in volgorde van goedkoopte.
Naar aanleiding van de eerste stap van algoritme 5 zou nu in een klassegesprek de volgende vraag, bedoeld als een poging tot rechtvaardiging van het algoritme, aan de orde kunnen komen:

> Weet je zeker, dat het goedkoopste telefoonnet in elk geval de goedkoopste tak vanuit ieder punt moet bevatten?

Hoewel met de beantwoording van deze vraag de correctheid van algoritme 5 nog niet volledig is aangetoond, kan het opwerpen van dergelijke vragen een belangrijke bijdrage leveren tot het 'leren redeneren' van onze leerlingen/studenten.

Tot besluit van dit artikel zullen we nu beredeneren (of had u dat zelf al gedaan?), dat het goedkoopste telefoonnet inderdaad de goedkoopste tak vanuit ieder punt dient te bevatten.
Laat namelijk T een telefoonnet zijn, dat geen gebruik maakt van de goedkoopste tak (zeg: t) vanuit een zeker punt (zeg: P).
Dan kunnen we een goedkoper telefoonnet construeren door eerst de tak t aan T toe te voegen (er ontstaat dan een circuit!) en vervolgens een andere tak vanuit P (namelijk degene die deel uitmaakt van het ontstane circuit) uit T te verwijderen.
Conclusie: het goedkoopste telefoonnet bevat in elk geval de goedkoopste tak vanuit ieder punt.

Literatuur

[1] Lange Jzn, J. de en M. Kindt: *Matrices (HEWET Wiskunde)*, Educaboek, Culemborg, 1984.
[2] Wilson, R.J.: *Introduction to Graph Theory*, Longman, New York, 1985.
[3] Harary, F.: *Graph Theory*, Addison-Wesley, Reading, Mass., 1969.
[4] Ore, O.: *Graphs and Their Uses*, (New Mathematical Library, 10), Random House, New York, 1963.
[5] Wie belangstelling heeft voor een listing van het in *'Het algoritme van Prim in matrixvorm'* genoemde computerprogramma (in BASIC), kan deze aanvragen bij:
Dr. R. Geel,
Lerarenopleiding Ubbo Emmius,
Sektie Wiskunde,
Postbus 2056,
9704 CB Groningen.
Vermeld in uw brief duidelijk uw naam en adres en sluit *f* 1,40 aan postzegels bij.