

N. van Splunder
**Optimalisatie van robuustheid in
netwerken**

Bachelorscriptie

30 juni 2017

Scriptiebegeleiders: dr. F.M. Spijksma
dr. J.L. Dorsman



Universiteit Leiden
Mathematisch Instituut

Inhoudsopgave

1	Inleiding	3
2	Effectieve weerstand	4
2.1	Achtergrond en definitie	4
2.2	Berekenen van de effectieve weerstand	5
2.3	Vergelijking met andere maten	6
3	Toevoegen van een tak	8
3.1	Met matrixvegen	8
3.2	Met één matrixinverse	11
3.3	Random wandelaar in een boom	12
3.4	Met de matrix-treestelling	15
4	Verschillende gewichten	18
5	De padgraaf	20
6	Conclusie	23

1 Inleiding

Als in een netwerk een tak uitvalt, kan dat verschillende gevolgen hebben. Het kan betekenen dat er tussen bepaalde knooppunten geen route meer bestaat, maar het kan ook zo zijn dat er voldoende omwegen zijn, waardoor het effect van de verstoring beperkt is. Voor bijvoorbeeld een metronetwerk, of het elektriciteitsnet van een land, is het relevant om te weten hoe ernstig het is als er een tak uitvalt. Om dit te kwantificeren zijn er verschillende maten voor ‘robuustheid’ bedacht, waarbij een hogere robuustheid betekent dat een verstoring minder ernstig is.

In deze scriptie gebruiken we de (totale) effectieve weerstand als maat voor robuustheid. De definitie hiervan geven we in Hoofdstuk 2. De reden dat we juist deze maat bekijken, is dat in dit geval de robuustheid altijd toeneemt als er een tak wordt toegevoegd aan het netwerk (zie Theorem 4.8 in [1]). Daarnaast wordt de effectieve weerstand zowel beïnvloed door het aantal paden tussen knopen als de lengte van de paden. In Paragraaf 2.3 gaan we dieper in op de vergelijking tussen deze maat en andere mogelijke maten voor robuustheid.

We zijn in het bijzonder geïnteresseerd in de situatie dat er al een netwerk ligt, en het nodig is dit robuuster te maken. Hierbij gaan we ervan uit dat er budget is om één tak toe te voegen aan het netwerk. We proberen dus te bepalen welke tak het beste toegevoegd kan worden om de effectieve weerstand zo laag mogelijk te maken, ofwel de robuustheid zo hoog mogelijk te maken.

In Hoofdstuk 2 geven we de achtergrond en definitie van de effectieve weerstand. Ook beargumenteren we waarom we juist de effectieve weerstand als maat voor robuustheid bekijken.

Vervolgens proberen we in Hoofdstuk 3 een manier te vinden om zo snel mogelijk te berekenen welke tak toegevoegd zou moeten worden om de effectieve weerstand te minimaliseren. Dit is relevant omdat sommige netwerken duizenden knopen hebben, en de berekening hierdoor met een inefficiënt algoritme zeer lang zou duren. We kijken hierbij alleen naar de worst case tijdscomplexiteit. Het blijkt dat voor dit probleem algoritmes met worst case tijdscomplexiteit $O(n^4)$ bestaan, waarbij n het aantal knopen is. Een bijzonder geval is de situatie waarin het netwerk een boom is. Dit is interessant omdat veel netwerken in de praktijk op bomen lijken. Een metronetwerk bijvoorbeeld bevat in het algemeen een centraal station, en vertakkingen vanuit dat punt in de verschillende richtingen. Om deze reden bekijken we enkele algoritmes die alleen voor bomen werken. Hierbij blijkt dat er een algoritme bestaat waarmee een gedeelte van de berekening sneller kan worden gedaan, maar dat nog steeds een worst case tijdscomplexiteit van $O(n^4)$ heeft.

In Hoofdstuk 4 bekijken we vervolgens een aantal eigenschappen van het verschil in effectieve weerstand door toevoegen van een tak. We bekijken hierbij welke tak optimaal is bij een hoog of juist een laag gewicht op deze tak.

Ten slotte bekijken we in Hoofdstuk 5 het effect van het toevoegen van een tak in een specifiek soort graaf, de padgraaf. Dit doen we om te bepalen waar de beste tak om toe te voegen ongeveer moet liggen, zodat we niet naar alle takken hoeven te kijken. Voor dit type graaf hebben we een goede benadering gevonden van waar de beste tak om toe te voegen ligt. Dit resultaat kan mogelijk gebruikt worden om ook voor andere soorten netwerken te bepalen welke takken zeker niet optimaal zijn, zodat de hoeveelheid rekenwerk verlaagd kan worden.

2 Effectieve weerstand

In dit hoofdstuk beschrijven we de totale effectieve weerstand, de maat voor robuustheid die we zullen gebruiken. Deze maat wordt beschreven in Paragraaf 3.2 van [2], en is gebaseerd op elektrische netwerken. In Paragraaf 2.1 geven we de natuurkundige achtergrond en de definitie van deze maat. In Paragraaf 2.2 leggen we uit hoe de totale effectieve weerstand van een netwerk eenvoudig te berekenen is. In Paragraaf 2.3 vergelijken we de totale effectieve weerstand met andere maten voor de robuustheid, en leggen we uit waarom we juist deze maat bekijken.

2.1 Achtergrond en definitie

We beschouwen een graaf die normaal, samenhangend en ongericht is. Hierbij betekent normaal dat de graaf geen evenwijdige takken of lussen heeft. Voor elke tak (i, j) kiezen we een weerstand $r_{ij} > 0$, waarbij een hogere weerstand zal leiden tot een lagere robuustheid. Tussen twee knopen a en b voegen we een spanningsbron toe, zodat er een stroom van 1 ampère van a naar b loopt. Laat nu y_{ij} de stroom zijn die door tak (i, j) loopt, voor iedere tak (i, j) . Dan geldt dat $\sum_{j \in N(a)} y_{aj} = 1$ en $\sum_{j \in N(b)} y_{bj} = -1$, waarbij $N(i)$ de verzameling burens van i is.

Volgens de eerste wet van Kirchhoff is de ingaande stroom gelijk aan de uitgaande stroom op alle tussenliggende knooppunten. Dus $\sum_{j \in N(i)} y_{ij} = 0$ voor alle i ongelijk aan a en b . De wet van Ohm zegt dat $U = IR$, waarbij U de spanning is, I de stroomsterkte en R de weerstand. In dit geval houdt dat in dat de spanning, ofwel het potentiaalverschil, tussen twee knopen i en j gelijk is aan $v_i - v_j := r_{ij}y_{ij}$. Deze twee wetten samen geven

$$\sum_{j \in N(i)} \frac{v_i - v_j}{r_{ij}} = \sum_{j \in N(i)} y_{ij} = \begin{cases} 1 & i = a \\ 0 & i \neq a, b \\ -1 & i = b. \end{cases}$$

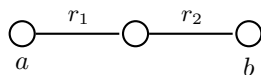
Volgens de Wet van Ohm geldt $v_a - v_b = R_{ab} \cdot 1 = R_{ab}$, met R_{ab} de vervangingsweerstand tussen a en b . Dat wil zeggen dat het potentiaalverschil tussen a en b hetzelfde is in het oorspronkelijke netwerk als in de situatie waarin het netwerk is vervangen door één tak met de weerstand R_{ab} . Deze waarde $v_a - v_b$ is uniek te bepalen uit bovenstaand stelsel vergelijkingen. Aan de hand hiervan definiëren we de totale effectieve weerstand.

Definitie 2.1. De effectieve weerstand R_{ab} is het potentiaalverschil tussen a en b dat ontstaat als we een stroom van 1 ampère door het netwerk van a naar b laten lopen. De totale effectieve weerstand R_{tot} van de graaf definiëren we als de som van de effectieve weerstanden tussen elk tweetal knopen, dus $R_{tot} = \sum_{a < b} R_{ab}$.

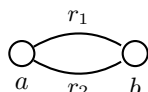
We definiëren de robuustheid nu als 1 gedeeld door de totale effectieve weerstand. De graaf is dus robuust als de totale effectieve weerstand klein is.

Als voorbeeld geven we de effectieve weerstand bij een eenvoudige serie- of parallelschakeling. Deze waarden zijn eenvoudig af te leiden uit bovenstaand stelsel vergelijkingen.

Voorbeeld 2.2. In het geval van een serieschakeling met weerstanden r_1 en r_2 geldt $R_{ab} = r_1 + r_2$.



Voorbeeld 2.3. In het geval van een parallelschakeling met weerstanden r_1 en r_2 geldt $R_{ab} = \frac{1}{\frac{1}{r_1} + \frac{1}{r_2}}$.



2.2 Berekenen van de effectieve weerstand

We kunnen de totale effectieve weerstand berekenen met behulp van de gewogen Laplaciaan-matrix, die we nu zullen definiëren. We definiëren eerst de Laplaciaan in het ongewogen geval, ofwel het geval dat elk van de gewichten gelijk is aan 1.

Definitie 2.4. Zij $G = (V, E)$ een normale, samenhangende, ongerichte graaf. De Laplaciaan L is de matrix gegeven door

$$L(i, j) = \begin{cases} d(i) & i = j \\ -1 & (i, j) \in E \\ 0 & \text{anders,} \end{cases}$$

waarbij $d(i)$ de graad van i is.

De Laplaciaan heeft dus op de diagonaal de graden staan, en verder een -1 op plaats (i, j) als er een tak is tussen i en j , en anders een 0. In het geval dat er gewichten op de takken staan wordt de Laplaciaan als volgt gedefiniëerd.

Definitie 2.5. Zij $G = (V, E)$ een normale, samenhangende, ongerichte graaf, met op iedere tak (i, j) een gewicht w_{ij} . De (gewogen) Laplaciaan L is de matrix gegeven door

$$L(i, j) = \begin{cases} \sum_{k \neq i} w_{ik} & i = j \\ -w_{ij} & (i, j) \in E \\ 0 & \text{anders.} \end{cases}$$

Om de totale effectieve weerstand te bepalen moet het gewicht bij tak (i, j) gekozen worden als $\frac{1}{r_{ij}}$. Er moeten dus gewichten gekozen worden, waarbij een hoger gewicht zal leiden tot een hogere robuustheid. In het geval van een dubbele tak vervangen we deze eerst door een enkele tak met de bijbehorende vervangingsweerstand. Dit heeft hetzelfde effect als het optellen van de bijbehorende gewichten.

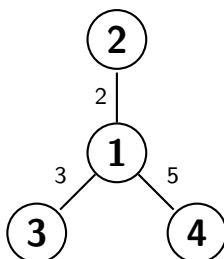
De effectieve weerstand valt nu als volgt te berekenen (zie Stelling 3.3.1 in [2]).

Stelling 2.6. Laat $b \in V$ een knoop uit het netwerk zijn. Noem $L_{[b]}$ de matrix die we krijgen door de b -de rij en kolom weg te laten uit de Laplaciaan. Dan geldt

$$R_{ij} = \begin{cases} L_{[b]}^{-1}(i, i) & j = b, i \neq b \\ L_{[b]}^{-1}(j, j) & j \neq b, i = b \\ L_{[b]}^{-1}(i, i) + L_{[b]}^{-1}(j, j) - L_{[b]}^{-1}(i, j) - L_{[b]}^{-1}(j, i), & i, j \neq b. \end{cases}$$

Uit bovenstaande stelling en Definitie 2.1 volgt nu dat R_{tot} gelijk is aan $n + 1$ keer de som van alle diagonaalelementen, min de som van alle elementen in de matrix, waarbij n het aantal knopen is.

Voorbeeld 2.7. We bekijken de volgende graaf.



De bijbehorende gewogen Laplaciaan is

$$L = \begin{pmatrix} 10 & -2 & -3 & -5 \\ -2 & 2 & 0 & 0 \\ -3 & 0 & 3 & 0 \\ -5 & 0 & 0 & 5 \end{pmatrix}.$$

We kiezen $b = 4$. We laten dus de vierde rij en de vierde kolom weg, en nemen dan de inverse. Dit geeft

$$L_{[b]}^{-1} = \frac{1}{30} \begin{pmatrix} 6 & 6 & 6 \\ 6 & 21 & 6 \\ 6 & 6 & 16 \end{pmatrix}.$$

Hieruit volgt dat $R_{14} = \frac{6}{30} = \frac{1}{5}$, $R_{24} = \frac{21}{30}$ en $R_{34} = \frac{16}{30} = \frac{8}{15}$. Verder geldt

$$\begin{aligned} R_{12} &= \frac{6}{30} + \frac{21}{30} - \frac{6}{30} - \frac{6}{30} = \frac{1}{2}, \\ R_{13} &= \frac{6}{30} + \frac{16}{30} - \frac{6}{30} - \frac{6}{30} = \frac{1}{3}, \\ R_{23} &= \frac{21}{30} + \frac{16}{30} - \frac{6}{30} - \frac{6}{30} = \frac{5}{6}. \end{aligned}$$

De totale effectieve weerstand is dan gelijk aan $\sum_{a < b} R_{ab} = \frac{31}{10}$.

2.3 Vergelijking met andere maten

Naast de totale effectieve weerstand zijn er natuurlijk ook andere maten te bedenken voor de robuustheid. Een voorbeeld is de takspreiding, die is gebaseerd

op het aantal kortste paden dat door elke knoop gaat. Hierbij kan het echter voorkomen dat bij het toevoegen van een tak de robuustheid lager wordt. Een belangrijke eigenschap van de totale effectieve weerstand is dat de robuustheid altijd toeneemt bij het toevoegen van een tak. Een eenvoudigere maat voor robuustheid die deze eigenschap ook heeft, is de gemiddelde afstand tussen de knopen in de graaf. Deze wordt echter alleen bepaald door het kortste pad tussen elk tweetal knopen, en niet door eventuele andere paden. De totale effectieve weerstand wordt zowel door het aantal paden tussen knopen als door de lengte van de paden beïnvloed.

Bovenstaande eigenschappen zorgen dat de totale effectieve weerstand goed overeenkomt met de intuïtie over wat robuustheid betekent. Verder geldt dat de robuustheid behalve voor normale grafen ook gedefinieerd is voor grafen met dubbele takken. Bovendien kunnen we verschillende eigenschappen van het netwerk, zoals de verkeersdruk op de verschillende takken, meemodelleren door gewichten te geven aan de takken. Dit betekent dat de maat toepasbaar is op veel van de netwerken die we in de praktijk tegenkomen. Een nadeel is echter dat de maat niet gedefinieerd is voor gerichte grafen.

Een vergelijking van de effectieve weerstand met andere maten voor robuustheid is te vinden in [1].

3 Toevoegen van een tak

In de praktijk hebben we vaak te maken met de situatie dat er al een netwerk ligt, en het nodig is om aan dit netwerk een tak toe te voegen, bijvoorbeeld omdat het overbelast is. Om deze reden bekijken we welke tak het beste toegevoegd kan worden om de effectieve weerstand zo laag mogelijk te maken.

Het is mogelijk de beste tak om toe te voegen te bepalen met behulp van Stelling 2.6. We kunnen namelijk voor elk knopenpaar v, w de bijbehorende tak toevoegen aan het netwerk, voor deze aangepaste graaf de Laplaciaaninverse bepalen, en de totale effectieve weerstand berekenen met behulp van Stelling 2.6. Dit betekent echter dat er n^2 van zulke inversen berekend moeten worden, waardoor dit geen efficiënt algoritme is. Omdat sommige netwerken, zoals het elektriciteitsnet van een land, in de praktijk duizenden knopen kunnen hebben, is het belangrijk om een efficiënt algoritme te hebben om te bepalen hoe zo'n netwerk het beste uitgebreid kan worden.

In onderstaande paragrafen worden een aantal efficiëntere algoritmes besproken. In Paragraaf 3.1 gebruiken we matrixvegen om de Laplaciaaninverse behorend bij het geval dat er een tak is toegevoegd af te leiden uit die van de oorspronkelijke graaf. Hierbij hebben we alleen de matrices $L_{[w]}^{-1}$ nodig, voor alle knopen w . Dit leidt tot een algoritme met worst case tijdscomplexiteit $O(n^4)$. In Paragraaf 3.2 proberen we iets soortgelijks te doen, maar dan op basis van één matrixinverse. Daarnaast bekijken we het geval dat het gewicht gelijk is aan ∞ , wat ervoor zorgt dat de twee knopen waartussen de nieuwe tak komt worden samengevoegd tot één knoop. Dit blijkt niet tot een sneller algoritme te leiden, omdat de worst case tijdscomplexiteit hierbij minstens $O(n^4)$ is. In Paragraaf 3.3 bekijken we een algoritme dat werkt voor bomen zonder gewichten, en gebruik maakt van een alternatieve interpretatie van de matrixelementen van $L_{[w]}^{-1}$ als het gemiddeld aantal keren dat een random wandelaar in een knoop terugkomt. Hierbij blijkt dat $L_{[w]}^{-1}(i, j)$ in een boom zonder gewichten geheelallig is, en gelijk is aantal takken overlap tussen het pad van i naar w en het pad van j naar w . Ook dit geeft een algoritme met worst case tijdscomplexiteit $O(n^4)$. In Paragraaf 3.4 bekijken we ten slotte een methode die gebruik maakt van de matrix-treestelling, en weer een worst case tijdscomplexiteit van $O(n^4)$ heeft. Ook dit algoritme werkt alleen voor bomen.

3.1 Met matrixvegen

Een mogelijke manier om de beste tak te bepalen gaat met behulp van matrixvegen. Hierbij maken we gebruik van het feit dat de Laplaciaan van de graaf met tak (v, w) toegevoegd weinig verschilt van die van de originele graaf. Hierdoor kunnen we de Laplaciaaninverse behorend bij de aangepaste graaf afleiden uit die van de originele graaf.

Laat $L_{[w]}$ de Laplaciaan behorend bij de graaf G zijn, zonder de w -de rij en kolom. Dan kunnen we het volgende bewijzen:

Stelling 3.1. *Het totale verschil in weerstand bij het toevoegen van tak (v, w)*

met gewicht σ , waarbij $v < w$, is

$$\frac{\sigma}{1 + \sigma L_{[w]}^{-1}(v, v)} \left(\sum_i L_{[w]}^{-1}(v, i)^2 + \sum_{i < j} (L_{[w]}^{-1}(v, i) - L_{[w]}^{-1}(v, j))^2 \right).$$

Bewijs. Voor het gemak kunnen we de knooppunten hernummeren zodat $v = n-1$ en $w = n$. We schrijven L_G voor de Laplaciaan behorend bij G , en $L_{G'}$ voor die behorend bij G' , de graaf waarbij de tak (v, w) met gewicht σ is toegevoegd. Merk op dat

$$L_{G',[n]}(i, j) = \begin{cases} L_{G,[n]}(i, j) & (i, j) \neq (n-1, n-1) \\ L_{G,[n]}(n-1, n-1) + \sigma & (i, j) = (n-1, n-1). \end{cases}$$

We gaan de matrices $(L_{G,[n]}|I)$ en $(L_{G',[n]}|I)$ vegen. We beginnen door de eerste $n-2$ elementen van de onderste rij tot 0 te vegen, en het $(n-2) \times (n-2)$ -vierkant in de linkerbovenhoek tot de identiteit. Hierbij komt aan de rechterkant de inverse A^{-1} van $L_{G,[n,n-1]}$ te staan. Voor G krijgen we

$$\left(\begin{array}{ccc|c|ccc} 1 & & 0 & & & & 0 \\ & \ddots & & \vec{k} & & A^{-1} & \vdots \\ 0 & & 1 & & & & 0 \\ \hline 0 & \dots & 0 & x & & \vec{b} & 1 \end{array} \right),$$

voor een zekere constante x , vector \vec{b} , en vector \vec{k} . Als we hetzelfde doen voor G' , krijgen we hetzelfde resultaat maar dan met $x + \sigma$ in plaats van x . Merk hierbij op dat $L_{G,[n,n-1]}$ gelijk is aan $L_{G',[n,n-1]}$.

Nu kunnen we verder vegen totdat links de identiteit staat. Voor G krijgen we dan

$$\left(\begin{array}{ccc|c|ccc} 1 & & 0 & 0 & & & \\ & \ddots & & \vdots & & A^{-1} - \vec{k}/x \cdot \vec{b} & -\vec{k}/x \\ 0 & & 1 & 0 & & & \\ \hline 0 & \dots & 0 & 1 & & \vec{b}/x & 1/x \end{array} \right),$$

en voor G' krijgen we hetzelfde, maar dan met x overal vervangen door $x + \sigma$. Nu staat aan de rechterkant $L_{G,[n]}^{-1}$, en deze matrix is symmetrisch omdat $L_{G,[n]}$ dat is. Dus moet gelden dat $\vec{k} = -\vec{b}$.

Laat nu R_{ij} de effectieve weerstand zijn tussen knooppunt i en j in G , en R'_{ij} de effectieve weerstand tussen i en j in G' .

Uit het bovenstaande valt nu af te lezen dat voor $i < j < n-1$ geldt dat

$$\begin{aligned} R_{ij} &= L_{G,[n]}^{-1}(i, i) + L_{G,[n]}^{-1}(j, j) - L_{G,[n]}^{-1}(i, j) - L_{G,[n]}^{-1}(j, i) \\ &= A^{-1}(i, i) - \frac{k_i b_i}{x} + A^{-1}(j, j) - \frac{k_j b_j}{x} - A^{-1}(i, j) + \frac{k_i b_j}{x} - A^{-1}(j, i) + \frac{k_j b_i}{x}, \end{aligned}$$

en voor R'_{ij} krijgen we hetzelfde maar dan met $x + \sigma$ in plaats van x . Dan volgt dat

$$\begin{aligned} R_{ij} - R'_{ij} &= \left(\frac{1}{x} - \frac{1}{x + \sigma} \right) (-k_i b_i - k_j b_j + k_i b_j + k_j b_i) \\ &= \frac{\sigma}{x(x + \sigma)} (b_i^2 + b_j^2 - 2b_i b_j) = \frac{\sigma}{x(x + \sigma)} (b_i - b_j)^2. \end{aligned}$$

Voor $j = n - 1$ en $i < n - 1$ geldt

$$\begin{aligned} R_{ij} - R'_{ij} &= \left(\frac{1}{x} - \frac{1}{x + \sigma} \right) (-k_i b_i + 1 + k_i - b_i) \\ &= \frac{\sigma}{x(x + \sigma)} (b_i - 1)^2. \end{aligned}$$

Ten slotte geldt voor $j = n$ dat

$$R_{ij} = \begin{cases} A^{-1}(i, i) - \frac{k_i b_i}{x} & i < n - 1 \\ \frac{1}{x} & i = n - 1, \end{cases}$$

waaruit volgt dat voor alle $i < j$ geldt dat

$$R_{ij} - R'_{ij} = \begin{cases} b_i^2 \frac{\sigma}{x(x + \sigma)} & j = n, i < n - 1 \\ \frac{\sigma}{x(x + \sigma)} & j = n, i = n - 1 \\ (b_i - 1)^2 \frac{\sigma}{x(x + \sigma)} & j = n - 1, i < n - 1 \\ (b_i - b_j)^2 \frac{\sigma}{x(x + \sigma)} & i < j < n - 1. \end{cases}$$

We schrijven nu voor het gemak L voor $L_{G, [n]}$. Dan valt uit de matrix af te lezen dat

$$\begin{aligned} x &= \frac{1}{L^{-1}(n - 1, n - 1)}, \\ b_i &= L^{-1}(n - 1, i) \cdot x, \quad i < n - 1, \\ 1 &= L^{-1}(n - 1, n - 1) \cdot x. \end{aligned}$$

Als v een andere knoop is dan $n - 1$, kunnen we dezelfde truc toepassen. In dat geval geldt $x = \frac{1}{L^{-1}(v, v)}$, $b_i = L^{-1}(v, i) \cdot x$ voor alle $i \neq v$, en $1 = L^{-1}(v, v) \cdot x$. We krijgen dan

$$R_{ij} - R'_{ij} = \sigma \frac{L^{-1}(v, v)^2}{1 + \sigma L^{-1}(v, v)} \cdot \begin{cases} \left(\frac{L^{-1}(v, i)}{L^{-1}(v, v)} \right)^2 & j = n, i \neq v, n \\ 1 & j = n, i = v \\ \left(\frac{L^{-1}(v, i)}{L^{-1}(v, v)} - \frac{L^{-1}(v, v)}{L^{-1}(v, v)} \right)^2 & j = v, i \neq v, n \\ \left(\frac{L^{-1}(v, i)}{L^{-1}(v, v)} - \frac{L^{-1}(v, j)}{L^{-1}(v, v)} \right)^2 & i < j, i, j \neq v, n. \end{cases}$$

Het totale verschil in weerstand is dus

$$\frac{\sigma}{1 + \sigma L^{-1}(v, v)} \left(L^{-1}(v, v)^2 + \sum_{i \neq v, n} L^{-1}(v, i)^2 + \sum_{i < j < n} (L^{-1}(v, i) - L^{-1}(v, j))^2 \right).$$

Als $w \neq n$ kunnen we hetzelfde doen. In het algemeen geldt dus dat het verschil in totale effectieve weerstand gelijk is aan

$$\frac{\sigma}{1 + \sigma L_{[w]}^{-1}(v, v)} \left(\sum_i L_{[w]}^{-1}(v, i)^2 + \sum_{i < j} (L_{[w]}^{-1}(v, i) - L_{[w]}^{-1}(v, j))^2 \right).$$

□

We kunnen het verschil in weerstand bij toevoegen van tak (v, w) dus aflezen uit de v -de rij van $L_{[w]}^{-1}$, als $v < w$. In het geval dat $v > w$ geldt echter dat we naar de $(v - 1)$ -de rij van $L_{[w]}^{-1}$ moeten kijken, in plaats van de v -de. De reden is als volgt. In de oorspronkelijke matrix komt de v -de rij overeen met knoop v . Als uit de matrix de w -de rij wordt weggelaten, dan wordt dit de $(v - 1)$ -de rij. Dan is de $(v - 1)$ -de rij dus juist de rij die overeenkomt met knoop v . Om verwarring te voorkomen schrijven we voortaan $L_{[w]}^{-1}(i, j)$ voor het matricelement dat overeenkomt met knoop i en knoop j , in plaats van het (i, j) -de matricelement.

We kunnen nu voor elke tak (v, w) het verschil in effectieve weerstand berekenen, om zo te bepalen welke we het beste toe kunnen voegen. Hiertoe moeten we voor een vaste knoop w een inverse bepalen, waarna we voor iedere knoop v het verschil in weerstand bij toevoegen van (v, w) kunnen berekenen. Dit kan in $O(n^3)$ tijd, want het bepalen van de inverse kan in $O(n^3)$ tijd, en voor iedere v kost het berekenen van het bijbehorende verschil in weerstand $O(n^2)$ tijd. Dit doen voor alle w geeft een tijdscomplexiteit van $O(n^4)$.

3.2 Met één matrixinverse

We gaan proberen iets soortgelijks te doen als in Paragraaf 3.1, maar dan met maar één matrixinverse. We weten al dat het mogelijk is het verschil in weerstand bij toevoegen van tak (v, w) te bepalen met behulp van $L_{[w]}^{-1}$ of $L_{[v]}^{-1}$. We kunnen nu echter proberen dit af te leiden uit $L_{[b]}^{-1}$, voor een willekeurige b , zodat voor elke knopencombinatie maar één matrixinverse nodig is.

Het bepalen van $L_{G', [b]}^{-1}$ met behulp van $L_{G, [b]}^{-1}$ zal echter minstens $O(n^2)$ tijd kosten, vanwege het aantal elementen dat bepaald moeten worden, en omdat de matrices in het algemeen op alle indices verschillen. Dit betekent dat de totale tijdscomplexiteit voor het nagaan van alle knopenparen (v, w) minstens $O(n^4)$ is, dus als we kijken naar de worst case tijdscomplexiteit is dit niet efficiënter dan de methode uit Paragraaf 3.1. Alleen als de matrix een zodanige structuur heeft dat de totale effectieve weerstand te bepalen is zonder alle matricelementen te bepalen zou het sneller kunnen.

Samenvoegen van knopen

Een ander idee is om de graaf wat te vereenvoudigen door het gewicht gelijk te nemen aan ∞ . Dit kunnen we zien als het samenvoegen van de twee knopen waar de tak tussen ligt tot één knoop. Dit geeft dan een resultaat dat correct is voor hoge waarden van σ .

We kunnen dit op twee manieren doen. Laat G de originele graaf zijn en G' de graaf waarbij de knopen v en w zijn samengevoegd. Als we nu $L_{G,[v]}^{-1}$ weten, dan is het mogelijk om hier in $O(n^2)$ tijd $L_{G',[v]}^{-1}$ uit af te leiden. Het bewijs hiervoor gaat met een argument zoals in Stelling 3.1. Hierdoor is het mogelijk om in $O(n^4)$ tijd de beste tak te bepalen om toe te voegen, namelijk door voor elke v de matrix $L_{G,[v]}^{-1}$ te berekenen, en hieruit voor elke w de inverse behorend bij het geval dat knopen v en w zijn samengevoegd te bepalen. Hiermee is dan de totale effectieve weerstand bij samenvoegen van knopen v en w te berekenen, voor alle v en w . Dit is echter nadelig ten opzichte van de methode in Paragraaf 3.1. Het heeft namelijk een net zo hoge tijdscomplexiteit maar levert minder informatie op, omdat het alleen klopt voor hoge waarden van σ . Een andere methode is om uit één matrix $L_{G,[b]}^{-1}$ de matrices behorend bij alle combinaties van knopen te bepalen. Dit probleem komt neer op het bepalen van de $L_{G',[n]}^{-1}$ uit $L_{G,[n]}^{-1}$, waarbij in G' de knopen 1 en 2 zijn samengevoegd. Hierbij geldt dus

$$L_{G,[n]} = \left(\begin{array}{cc|ccc} d_1 & -w_{12} & -w_{13} & \dots & -w_{1,n-1} \\ -w_{12} & d_2 & -w_{23} & \dots & -w_{2,n-1} \\ \hline -w_{13} & -w_{23} & d_3 & & w_{ij} \\ \vdots & \vdots & & \ddots & \\ -w_{1,n-1} & -w_{2,n-1} & w_{ij} & & d_{n-1} \end{array} \right)$$

en

$$L_{G',[n]} = \left(\begin{array}{cc|ccc} d_1 + d_2 - 2w_{12} & -w_{13} - w_{23} & \dots & -w_{1,n-1} - w_{2,n-1} \\ \hline -w_{13} - w_{23} & d_3 & & w_{ij} \\ \vdots & & \ddots & \\ -w_{1,n-1} - w_{2,n-1} & w_{ij} & & d_{n-1} \end{array} \right),$$

waarbij het $(n-2) \times (n-2)$ -blok rechtsonder in beide gevallen dezelfde elementen bevat.

We kunnen nu weer op dezelfde manier als in Stelling 3.1 proberen de inverse van $L_{G',[n]}$ af te leiden uit $L_{G,[n]}$. Ook hier kan dat echter niet sneller dan in $O(n^4)$ tijd, omdat de matrices in het algemeen op alle indices van elkaar verschillen. Ook deze methode is dus nadelig ten opzichte van die in Paragraaf 3.1.

3.3 Random wandelaar in een boom

In de praktijk lijken veel netwerken op bomen. Een metronetwerk bestaat bijvoorbeeld uit een centraal station vanaf waar er vertakkingen in de verschillende richtingen zijn. Om deze reden is het belangrijk het optimaliteitsprobleem voor bomen op te lossen. In het geval dat de graaf een boom is en geen gewichten heeft blijkt dat er een sneller algoritme is dan dat in Paragraaf 3.1, hoewel de worst case tijdscomplexiteit hierbij nog steeds $O(n^4)$ is.

Voor het algoritme hebben we de volgende definitie nodig.

Definitie 3.2. We kunnen $L_{[b]}^{-1}(i, j)$ als volgt opvatten:

$$L_{[b]}^{-1}(i, j) = \tau_{[b]}(i, j) \frac{1}{d(j)},$$

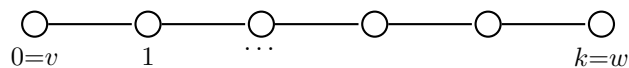
waarbij $\tau_{[b]}(i, j)$ het aantal keren is dat een random wandelaar vanuit i in j terechtkomt, voordat die b bereikt (zie Paragraaf 3.4 in [2]).

We nemen nu aan dat we een boom zonder gewichten hebben, en kiezen v en w vast. We bepalen eerst $L_{[w]}^{-1}(v, v)$. Laat het pad tussen v en w uit k takken bestaan. We noemen $v = 0$, $w = k$ en nummeren de knooppunten op het pad $1, 2, \dots, k - 1$.

We kunnen nu met behulp van enkele opmerkingen de situatie vereenvoudigen:

- Vanuit knooppunt 0 heeft de random wandelaar kans $\frac{1}{d(v)}$ om naar knoop 1 te gaan. Als de wandelaar naar een andere knoop gaat, komt hij zeker weer terug in 0. We hoeven dus alleen de graad van knoop 0 te weten, maar kunnen de vertakkingen vanaf dit knooppunt verder buiten beschouwing laten.
- Vanuit een knooppunt $i \in \{1, \dots, k - 1\}$ kan de wandelaar drie dingen doen. Hij kan terug gaan naar knooppunt $i - 1$, doorgaan naar knooppunt $i + 1$, of een andere richting op gaan. In het laatste geval komt de wandelaar zeker terug in knooppunt i , zonder eerst in 0 of in k te komen. Dit geval kunnen we dus negeren. De wandelaar gaat nu, na eventueel een andere richting in te zijn gegaan, met kans $\frac{1}{2}$ uiteindelijk naar $i - 1$, en met kans $\frac{1}{2}$ uiteindelijk naar $i + 1$.
- Aangezien de wandelaar stopt als hij in knooppunt k beland is, kunnen we alle takken vanuit k , behalve die naar $k - 1$, negeren.

Nu houden we de volgende situatie over:



Hierbij hoeven we alleen k en $d(v)$ te weten.

Lemma 3.3. *Er geldt $L_{[w]}^{-1}(v, v) = k$, dat wil zeggen dat de random wandelaar naar verwachting $k \cdot d(v)$ keer in 0 is voordat hij in k eindigt.*

Bewijs. We gebruiken inductie naar k .

Omdat de wandelaar begint in 0, en naar verwachting de $d(v)$ 'de keer dat hij uit 0 vertrekt in 1 uitkomt, is hij naar verwachting $d(v)$ keer in 0 geweest voordat hij in 1 is. Als $k = 1$ klopt de bewering dus.

Zij $k \in \mathbb{Z}_{>0}$ willekeurig en neem aan dat de bewering klopt voor een pad van lengte k . Bekijk nu het geval dat het pad lengte $k + 1$ heeft. We weten dat het naar verwachting $k \cdot d(v)$ keer terugkeren kost om in k te komen, en willen nu weten hoe vaak we terugkeren voor we in $k + 1$ komen. Vanuit knoop k kunnen er drie dingen gebeuren: de wandelaar kan doorlopen naar $k + 1$, hij kan een

stukje teruglopen maar weer in k komen voordat hij 0 bereikt, of hij kan naar 0 lopen zonder eerst weer terug in k te komen. We willen weten wat de kans is op het laatste. Hiertoe definiëren we a_i als de kans dat we vanuit i naar 0 gaan zonder terug naar k te gaan, voor $i = 1, \dots, k-1$. Dan geldt

$$\begin{aligned} a_1 &= \frac{1}{2} + \frac{1}{2}a_2, \\ a_i &= \frac{1}{2}a_{i-1} + \frac{1}{2}a_{i+1} \quad i \in \{2, \dots, k-2\}, \\ a_{k-1} &= \frac{1}{2}a_{k-2}. \end{aligned}$$

Oplossen van dit stelsel vergelijkingen geeft $a_{k-1} = \frac{1}{k}$.

Dit betekent dat de kans om van k naar 0 te gaan, zonder terug te keren in k gelijk is aan $\frac{1}{2} \cdot \frac{1}{k} = \frac{1}{2k}$.

Nu volgt dat de kans dat de wandelaar vanuit k in 0 komt voordat hij in $k+1$ komt gelijk is aan

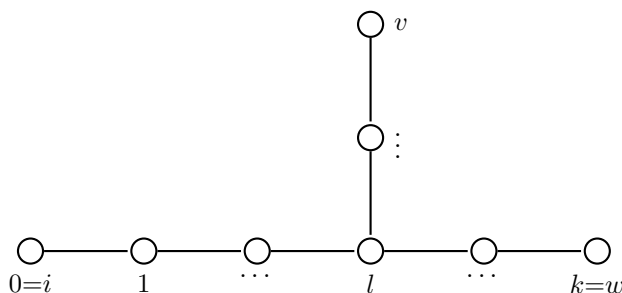
$$\frac{\frac{1}{2k}}{\frac{1}{2k} + \frac{1}{2}} = \frac{1}{k+1}.$$

De kans dat hij in $k+1$ komt voordat hij in 0 komt is dus $\frac{k}{k+1}$. Het verwachte aantal keren dat de wandelaar in 0 is voordat hij in $k+1$ eindigt is dan

$$\frac{k}{k+1} \cdot k \cdot d(v) + \frac{1}{k+1} \cdot \frac{k}{k+1} \cdot 2k \cdot d(v) + \frac{1}{(k+1)^2} \cdot \frac{k}{k+1} \cdot 3k \cdot d(v) + \dots = (k+1)d(v).$$

Met inductie volgt nu dat de wandelaar voor alle k naar verwachting $k \cdot d(v)$ keer in 0 is voor dat hij in k is. \square

Nu kunnen we ook $L_{[w]}^{-1}(v, i)$ bepalen voor alle $i \neq v$. We nummeren het pad van i naar w zoals hierboven. Als het pad van v naar w disjunct is met dat van i naar w , dan komt de wandelaar in w aan voordat hij in i komt. Anders loopt het pad van v naar w vanaf een zeker punt gelijk aan dat van i naar w . Laten we het knooppunt vanaf waar de paden gelijk zijn l noemen. Dan krijgen we de volgende situatie:



Als de wandelaar in knooppunt l is aangekomen, kunnen we met dezelfde redenering als hierboven weer alle takken en knopen buiten het pad van i naar w negeren. Het enige verschil met de situatie hiervoor is dat we nu niet in 0 beginnen, maar op een zeker punt op het pad. Laat nu a_l het aantal keren zijn

dat de wandelaar in i komt voordat hij in w eindigt, als hij in het punt l begint. Dan geldt

$$\begin{aligned} a_0 &= k \cdot d(i), \\ a_l &= \frac{1}{2}a_{l-1} + \frac{1}{2}a_{l+1} \quad l = 2, \dots, k-2, \\ a_{k-1} &= \frac{1}{2}a_{k-2}, \\ a_k &= 0. \end{aligned}$$

Oplossen van dit stelsel geeft $a_l = (k-l)d(i)$ voor alle l . Dan volgt dat $L_{[w]}^{-1}(v, i) = k-l$.

Nu hebben we de volgende stelling bewezen.

Stelling 3.4. *Voor een boom zonder gewichten geldt dat $L_{[w]}^{-1}$ geheeltallig is. In het bijzonder is $L_{[w]}^{-1}(i, j)$ gelijk aan het aantal takken waarop het pad van i naar w overeenkomt met het pad van j naar w .*

Nu kunnen we voor gegeven v en w de waarden $L_{[w]}^{-1}(v, i)$ bepalen door voor alle i te kijken hoeveel takken op het pad tussen i en w overeenkomen met die op het pad tussen v en w . Dit kunnen we doen door Depth First Search te gebruiken vanuit knoop w , en daarbij bij te houden hoeveel takken uit het pad tussen v en w gepasseerd zijn. Dit kan op een boom in $O(n)$ tijd. We kunnen dus in $O(n^3)$ tijd voor alle w de matrices $L_{[w]}^{-1}$ bepalen.

Dit werkt echter alleen voor bomen zonder gewichten. Voor bomen met gewichten kunnen we wel hetzelfde idee toepassen. Het blijkt dan dat in het geval van rationale gewichten de waarden $L_{[w]}^{-1}(i, j)$ ook rationaal worden. Het belangrijkste verschil is dat de kans om vanuit een tak uiteindelijk naar links te gaan niet meer $\frac{1}{2}$ is, maar $\frac{\alpha}{\alpha+\beta}$, waarbij α het gewicht op de linkertak is, en β het gewicht op de rechtertak. Hierdoor gaat het inductiebewijs in Lemma 3.3 niet meer op. Het gevolg is dat voor $l = 1, \dots, k$ een stelsel vergelijkingen zoals in het bewijs van het lemma opgelost moet worden, wat in $O(n^2)$ tijdsstappen kan. Vervolgens moet een stelsel vergelijkingen opgelost worden om de $L_{[w]}^{-1}(v, i)$ te bepalen. Dit doen voor elk paar v, w kost $O(n^4)$ tijd.

Het bepalen van de waarden $L_{[w]}^{-1}(v, i)$ is echter nog niet genoeg om de totale effectieve weerstand te bepalen. Deze waarden moeten ook nog ingevuld worden in de formule in Stelling 3.1. Voor elke combinatie van v en w moet het verschil in weerstand moeten berekend, waarvoor ongeveer n^2 kwadraten berekend moeten worden. Het berekenen van het verschil in weerstand bij het toevoegen van elke mogelijke tak kost dus $O(n^4)$ tijd. Bij dit algoritme gaat de eerste stap, het berekenen van de $L_{[w]}^{-1}(v, i)$, dus sneller, maar is de worst case tijdscomplexiteit nog steeds $O(n^4)$.

3.4 Met de matrix-treestelling

Het is mogelijk om de effectieve weerstand na toevoegen van een tak op nog een andere manier te berekenen, die te maken heeft met het aantal opspannende bomen in de graaf. Omdat dit alleen gemakkelijk te berekenen is als de graaf met een toegevoegde tak weinig cycli heeft, bekijken we deze methode alleen

voor bomen.

We zullen eerst R_{ab} op een andere manier schrijven. Hiervoor gebruiken we de matrix-treestelling.

Stelling 3.5. *Zij G een graaf en b een knoop uit de graaf. Als de graaf geen gewichten heeft is het aantal opspannende bomen van G gelijk aan $\det(L_{[b]})$. Als de graaf wel gewichten heeft is $\det(L_{[b]})$ gelijk aan de som van de gewichten van alle opspannende bomen van G . Hierbij is het gewicht van een opspannende boom gedefinieerd als het product van de gewichten op de takken in de boom.*

Uit Stelling 3.1 weten we dat $R_{ab} = L_{[b]}^{-1}(a, a)$, voor elk tweetal knopen a en b . We gebruiken de regel van Cramer om een andere uitdrukking voor R_{ab} te vinden. Laat e_a de vector zijn met op de a -de plek een 1, en verder overal 0. Dan is de a -de element van $L_{[b]}^{-1}e_a$ gelijk aan $L_{[b]}^{-1}(a, a)$. De regel van Cramer zegt dan dat $R_{ab} = \frac{\det(L_{[a,b]})}{\det(L_{[b]})}$. Volgens de matrix-treestelling is $\det(L_{[a,b]})$ gelijk aan het aantal opspannende bomen van de graaf waarbij de knopen a en b zijn samengevoegd tot één knoop. Dit is hetzelfde als het aantal opspannende bomen van G die de tak (a, b) bevatten. Hiermee volgt dat we de effectieve weerstand R_{ab} kunnen berekenen als

$$R_{ab} = \frac{\begin{array}{l} \text{het aantal opspannende bomen van } G \text{ dat } (a, b) \text{ bevat,} \\ \text{of, als deze er niet is, na toevoeging van een tak } (a, b) \end{array}}{\text{het aantal opspannende bomen van } G},$$

als G geen gewichten heeft (zie Paragraaf 3.5 in [2]). Als G wel gewichten heeft nemen we in plaats van het aantal opspannende bomen de som van de gewichten van de opspannende bomen. Voor het gemak bekijken we eerst het geval zonder gewichten.

We willen de effectieve weerstand bepalen in een boom waaraan tak (v, w) is toegevoegd, zodat één cykel is ontstaan. We moeten nu voor elk tweetal knopen i, j de waarde R_{ij} bepalen. Hiertoe moeten we de afstand tussen de knopen in de originele boom weten. Het aantal takken tussen i en j in de originele boom noemen we d_{ij} . Een opspannende boom verkrijgen we door telkens een tak die op een cykel ligt weg te laten totdat er geen cyclen meer zijn. We moeten dus tellen op hoeveel manieren dit kan. In de boom met tak (v, w) toegevoegd ligt één cykel met daarin $d_{vw} + 1$ takken, dus er kan op $d_{vw} + 1$ manieren een opspannende boom worden gevormd. Dus de noemer in de vergelijking is gelijk aan $d_{vw} + 1$. Nu geldt:

- Als (i, j) in de graaf zit en (i, j) op de gecreëerde cykel ligt geldt $R_{ij} = \frac{d_{vw}}{d_{vw}+1}$.
- Als (i, j) in de graaf zit en niet op de cykel ligt geldt $R_{ij} = \frac{d_{vw}+1}{d_{vw}+1} = 1$.
- Als (i, j) niet in de graaf zit en het pad van i naar j niet overlapt met het pad van v naar w geldt $R_{ij} = \frac{d_{ij}(d_{vw}+1)}{d_{vw}+1} = d_{ij}$.
- Als (i, j) niet in de graaf zit en het pad van i naar j op x takken overlapt met het pad van v naar w , moeten twee takken verwijderd worden om een opspannende boom te vormen. Dit kan op twee manieren: we kunnen één

van de x takken waarop de paden overlappen verwijderen, waarna er één cykel overblijft waaruit nog een tak moet worden verwijderd, of we kunnen deze x takken in het netwerk laten blijven, waardoor er uit elk van de twee cycli waarin deze x takken voorkomen een tak verwijderd moet worden. We krijgen nu $R_{ij} = \frac{x(d_{ij}-x+d_{vw}+1-x)+(d_{ij}-x)(d_{vw}+1-x)}{d_{vw}+1} = \frac{d_{ij}(d_{vw}+1)-x^2}{d_{vw}+1}$.

Hiermee valt R_{tot} te berekenen. Om voor één tak (v, w) te kijken wat de effectieve weerstand is na toevoegen van de tak moeten we dus voor elk paar (i, j) bepalen op hoeveel punten het kortste pad tussen i en j en het kortste pad tussen v en w overlappen. Dit kunnen we doen door vanuit elke knoop Depth First Search te gebruiken, en hierbij een teller bij te houden die aangeeft hoeveel takken uit het pad tussen v en w gepasseerd zijn. Het is mogelijk dit te doen in $O(n^2)$ tijd. Om dit voor elk paar (v, w) te doen is dus $O(n^4)$ tijd nodig. Hierna moeten voor elk paar (v, w) de waarden R_{ij} berekend worden, wat ook in $O(n^4)$ tijdstappen kan.

In het geval met gewichten moeten we in de berekening telkens het aantal opspannende bomen vervangen door de som van de gewichten van deze bomen. De opspannende bomen in de berekening verkrijgen we vanuit de graaf waarbij tak (v, w) is toegevoegd door ofwel één tak weg te halen uit de graaf, ofwel twee takken weg te halen uit de graaf en een tak (i, j) met gewicht 1 toe te voegen. Om de gewichten van deze bomen te bepalen berekenen we eerst het product van alle gewichten in de graaf. Om het gewicht van de opspannende bomen te bepalen moeten we dan delen door de gewichten van de weggelaten takken. We moeten dus voor elke tak en elk tweetal takken bepalen wat het gewicht is als die takken weggehaald zijn uit de graaf, wat in $O(n^2)$ tijd kan. Dit betekent dat ook in dit geval de worst case tijdscomplexiteit gelijk is aan $O(n^4)$.

4 Verschillende gewichten

In dit hoofdstuk bekijken we een aantal eigenschappen van het verschil in effectieve weerstand bij toevoegen van een tak. In het bijzonder bekijken we voor vaste w wat de optimale tak (v, w) om toe te voegen is, afhankelijk van of σ hoog of laag is. Hiertoe tonen we eerst aan dat de elementen van $L_{[w]}^{-1}$ niet-negatief zijn.

Lemma 4.1. *Voor alle i, j geldt $L_{[w]}^{-1}(i, j) \geq 0$.*

Bewijs. We zullen bewijzen dat de x en \vec{b} uit het bewijs van Stelling 3.1 niet-negatief zijn. Hieruit volgt het gevraagde.

We kunnen x vinden door de linker-onderdriehoek van $L_{G, [n]}$ tot 0 te vegen. Merk op dat voordat we beginnen met vegen alle diagonaal-elementen positief, en alle andere elementen 0 of negatief zijn. Daarnaast zijn de rij-sommen niet-negatief. We tonen aan dat dit na elke stap in het vegen nog steeds zo is.

In elke stap wordt een positief aantal keer een rij bij een andere rij opgeteld. Immers, om een niet-diagonaal element $a_{i,j}$ gelijk aan 0 te maken, moeten we hier een positief getal bij optellen. Er moet dus een positief aantal keren $a_{j,j}$ bij opgeteld worden. Bij alle andere elementen in de rij worden dan negatieve getallen, of 0, opgeteld. Voor de elementen buiten de diagonaal betekent dit dat ze niet-positief blijven. Omdat de rij-sommen positief waren, en er een positief aantal keren een rij bij een andere is opgeteld, is de nieuwe rij-som ook positief. Hieruit volgt dat het diagonaalelement positief blijft.

Uit het bovenstaande volgt dat x positief is. Daarnaast zijn alle elementen van \vec{b} niet-negatief, omdat we deze verkrijgen door, beginnend met de nul-rij, telkens een positief aantal keren een rij uit I_{n-2} op te tellen. \square

Het bovenstaande volgt ook uit de interpretatie gegeven in Definitie 3.2.

We nemen nu een w vast. We noemen $D_{v,\sigma}$ het totale verschil in weerstand bij toevoegen van tak (v, w) met gewicht σ . Deze waarde hebben we bepaald in Stelling 3.1. Verder noemen we

$$C(v) = \sum_i L_{[w]}^{-1}(v, i)^2 + \sum_{i < j} (L_{[w]}^{-1}(v, i) - L_{[w]}^{-1}(v, j))^2.$$

Als we v vast nemen kunnen we $D_{v,\sigma}$ opvatten als een continue functie van σ . We kunnen nu het volgende bewijzen:

Lemma 4.2. *$D_{v_1,\sigma}$ en $D_{v_2,\sigma}$, het verschil in relatieve weerstand bij toevoegen van tak (v_1, w) resp. (v_2, w) als functie van σ , hebben hooguit één snijpunt met $\sigma > 0$.*

Bewijs. Stel dat $D_{v_1,\sigma} = D_{v_2,\sigma}$. Dan geldt

$$\frac{\sigma}{1 + \sigma L_{[w]}^{-1}(v_1, v_1)} C(v_1) = \frac{\sigma}{1 + \sigma L_{[w]}^{-1}(v_2, v_2)} C(v_2),$$

ofwel

$$\sigma C(v_1)(1 + \sigma L_{[w]}^{-1}(v_2, v_2)) = \sigma C(v_2)(1 + \sigma L_{[w]}^{-1}(v_1, v_1)).$$

Als $\sigma > 0$ volgt hieruit dat $C(v_1)(1 + \sigma L_{[w]}^{-1}(v_2, v_2)) = C(v_2)(1 + \sigma L_{[w]}^{-1}(v_1, v_1))$. Dit is een lineaire vergelijking in σ , en deze heeft dus hooguit één oplossing met $\sigma > 0$. \square

Lemma 4.3. *Stel dat $\frac{C(v_1)}{L_{[w]}^{-1}(v_1, v_1)} > \frac{C(v_2)}{L_{[w]}^{-1}(v_2, v_2)}$. Dan geldt dat $D_{v_1, \sigma} > D_{v_2, \sigma}$ dan en slechts dan als*

$$\sigma > \frac{C(v_2) - C(v_1)}{L_{[w]}^{-1}(v_2, v_2)C(v_1) - L_{[w]}^{-1}(v_1, v_1)C(v_2)}.$$

Bewijs. We willen weten wanneer $D_{v_1, \sigma} > D_{v_2, \sigma}$, ofwel

$$\frac{\sigma}{1 + \sigma L_{[w]}^{-1}(v_1, v_1)} C(v_1) > \frac{\sigma}{1 + \sigma L_{[w]}^{-1}(v_2, v_2)} C(v_2).$$

We delen nu aan beide kanten door σ . Als we dan kruislings vermenigvuldigen en de termen met σ naar de linkerkant halen, krijgen we

$$\sigma(L_{[w]}^{-1}(v_2, v_2)C(v_1) - L_{[w]}^{-1}(v_1, v_1)C(v_2)) > C(v_2) - C(v_1).$$

Omdat $\frac{C(v_1)}{L_{[w]}^{-1}(v_1, v_1)} > \frac{C(v_2)}{L_{[w]}^{-1}(v_2, v_2)}$ geldt dat $L_{[w]}^{-1}(v_2, v_2)C(v_1) - L_{[w]}^{-1}(v_1, v_1)C(v_2) > 0$. Hiermee volgt dat $D_{v_1, \sigma} > D_{v_2, \sigma}$ dan en slechts dan als

$$\sigma > \frac{C(v_2) - C(v_1)}{L_{[w]}^{-1}(v_2, v_2)C(v_1) - L_{[w]}^{-1}(v_1, v_1)C(v_2)}.$$

\square

Lemma 4.4. *Stel dat $C(v_1) > C(v_2)$. Dan geldt voor kleine σ dat $D_{v_1, \sigma} > D_{v_2, \sigma}$.*

Bewijs. Voor alle knopen v geldt dat $D_{v, 0} = 0$. Voor de knoop v die optimaal is voor zeer lage σ moet dus gelden dat $D_{v, \sigma}$ het snelst stijgt bij 0, ofwel dat $\frac{d}{d\sigma} D_{v, \sigma} \Big|_{\sigma=0}$ maximaal is. Er geldt

$$\frac{d}{d\sigma} D_{v, \sigma} \Big|_{\sigma=0} = \frac{C(v)(1 + \sigma L_{[w]}^{-1}(v, v)) + \sigma C(v) \cdot L_{[w]}^{-1}(v, v)}{(1 + \sigma L_{[w]}^{-1}(v, v))^2} \Big|_{\sigma=0} = C(v),$$

dus v is optimaal voor lage σ dan en slechts dan als $C(v)$ maximaal is. Hieruit volgt het gevraagde. \square

Als σ groot is moeten we dus de knoop v kiezen waarvoor $\frac{C(v)}{L_{[w]}^{-1}(v, v)}$ maximaal is, en is het optimaal om de tak (v, w) toe te voegen. Voor kleine σ is het optimaal om de knoop v te kiezen waarvoor $C(v)$ maximaal is, en de tak (v, w) toe te voegen.

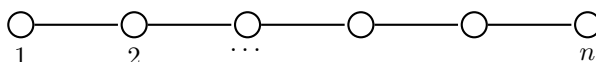
5 De padgraaf

Verschillende soorten netwerken waar we in de praktijk mee te maken hebben bevatten lange paden zonder vertakkingen. Een metronetwerk bijvoorbeeld bestaat uit verschillende metrolijnen, die op slechts een aantal stations bij elkaar komen. Een ander voorbeeld is een wegennetwerk, waarbij op veel plaatsen de mogelijkheid bestaat om een afrit of zijweg toe te voegen, ook als die er nog niet is.

Als we voor dit type graaf zouden kunnen bepalen welke knopencombinaties zeker niet optimaal zijn, geeft dat mogelijk een sneller algoritme voor het vinden van de optimale tak om toe te voegen. Een eerste stap om dit probleem op te lossen is het bekijken van de meest eenvoudige graaf van dit type, de zogenaamde padgraaf.

Definitie 5.1. De padgraaf van lengte n is de graaf met n knopen, waarbij er een tak $(i, i + 1)$ is voor elke knoop $i < n$.

Een padgraaf ziet er als volgt uit:



Laat nu $1 \leq v < w \leq n$. We zoeken v en w zo dat toevoegen van tak (v, w) leidt tot een zo laag mogelijke effectieve weerstand. Hiertoe moeten we de elementen $L_{[w]}^{-1}(v, i)$ bepalen. In Paragraaf 3.3 hebben we beschreven hoe deze bepaald kunnen worden in een boom. Bij deze graaf krijgen we

$$L_{[w]}^{-1}(v, i) = \begin{cases} 0 & i > w, \\ w - v & i \leq v, \\ w - i & v < i < w. \end{cases}$$

Met Stelling 3.1 volgt dan dat het verschil in effectieve weerstand gelijk is aan

$$\begin{aligned} & \frac{1}{1 + w - v} \left(\sum_{1 \leq i \leq v} (w - v)^2 + \sum_{v < i < w} (w - i)^2 + \sum_{1 \leq i \leq v} \sum_{v < j < w} ((w - v) - (w - j))^2 \right. \\ & + \sum_{1 \leq i \leq v} \sum_{w < j \leq n} (w - v)^2 + \sum_{v < i < w} \sum_{w < j \leq n} (w - i)^2 \\ & \left. + \sum_{v < i < w-1} \sum_{i < j < w} ((w - i) - (w - j))^2 \right). \end{aligned}$$

Dit valt te vereenvoudigen tot

$$\begin{aligned} & \frac{1}{1 + w - v} (v(1 + n - w)(v - w)^2 \\ & - \frac{1}{6} (1 + v + n - w)(v - w + 1)(2v - 2w + 1)(v - w) \\ & + \frac{1}{12} (v - w)(v - w + 1)^2 (v - w + 2)). \end{aligned} \tag{1}$$

We zullen nu eerst aantonen dat het voor grotere waarden van n optimaal is als $w = n - v + 1$, ofwel als v en w symmetrisch gekozen zijn.

Stelling 5.2. *Voor grote waarden van n geldt het volgende: het verschil in weerstand bij toevoegen van een tak aan de padgraaf is maximaal als $w = n - v + 1$.*

Bewijs. Laat $w = v + k$, voor een zekere $k > 0$. We tonen eerst aan dat het toevoegen van $(v, v + k)$ optimaal is voor $v = \frac{1}{2}n - \frac{1}{2}k + \frac{1}{2}$, in het geval dat dit een geheel getal is. Voor deze waarde van v geldt dat $w = n - v + 1$.

Om aan te tonen dat het toevoegen van $(v, v + k)$ optimaal is voor $v = \frac{1}{2}n - \frac{1}{2}k + \frac{1}{2}$ vervangen we w door $v + k$ in (1), en nemen de partiële afgeleide naar v . Dit geeft

$$\frac{k^2(-k + n - 2v + 1)}{k + 1}.$$

Deze heeft $\frac{1}{2}n - \frac{1}{2}k + \frac{1}{2}$ als nulpunt. Verder geldt dat de tweede afgeleide naar v gelijk is aan $\frac{-2k^2}{k+1}$. Omdat de tweede afgeleide negatief is volgt dat het nulpunt van de eerste afgeleide een maximum is. Het verschil in weerstand bij toevoegen van $(v, v + k)$ is dus maximaal als $v = \frac{1}{2}n - \frac{1}{2}k + \frac{1}{2}$.

Als v en k beide even of beide oneven zijn, is $\frac{1}{2}n - \frac{1}{2}k + \frac{1}{2}$ geen geheel getal, waardoor dit geen oplossing is. In dit geval zal de optimale waarde voor v liggen bij $v = \frac{1}{2}n - \frac{1}{2}k$ of $v = \frac{1}{2}n - \frac{1}{2}k + 1$. Omdat dit voor grotere waarden van n weinig verschilt van $\frac{1}{2}n - \frac{1}{2}k + \frac{1}{2}$, kunnen we dit verschil negeren. \square

We kunnen dus w gelijk stellen aan $n - v + 1$. Dit invullen in (1) geeft

$$\frac{n(n^3 + n(-12v^2 + 12v - 1) + 8v(2v^2 - 3v + 1))}{12(n - 2v + 2)}.$$

De afgeleide naar v nemen geeft

$$\frac{1}{6}n \left(\frac{3n}{(n - 2v + 2)^2} + n - 8v + 2 \right).$$

Deze functie heeft drie reële nulpunten in termen van v , namelijk

$$v = -\frac{1}{8}\sqrt[3]{n^3 + 6n^2 + 4\sqrt{3}\sqrt{-n^4 - 6n^3 - 8n - 12n + 8}} + \frac{-144n^2 - 576n - 576}{1152\sqrt[3]{n^3 + 6n^2 + 4\sqrt{3}\sqrt{-n^4 - 6n^3 - 8n - 12n + 8}}} + \frac{3(n+2)}{8}, \quad (2)$$

$$v = \frac{1}{16}(1 - i\sqrt{3})\sqrt[3]{n^3 + 6n^2 + 4\sqrt{3}\sqrt{-n^4 - 6n^3 - 8n - 12n + 8}} - \frac{(1 + i\sqrt{3})(-144n^2 - 576n - 576)}{2304\sqrt[3]{n^3 + 6n^2 + 4\sqrt{3}\sqrt{-n^4 - 6n^3 - 8n - 12n + 8}}} + \frac{3(n+2)}{8}, \quad (3)$$

$$v = \frac{1}{16}(1 + i\sqrt{3})\sqrt[3]{n^3 + 6n^2 + 4\sqrt{3}\sqrt{-n^4 - 6n^3 - 8n - 12n + 8}} - \frac{(1 - i\sqrt{3})(-144n^2 - 576n - 576)}{2304\sqrt[3]{n^3 + 6n^2 + 4\sqrt{3}\sqrt{-n^4 - 6n^3 - 8n - 12n + 8}}} + \frac{3(n+2)}{8}. \quad (4)$$

Deze nulpunten bevatten complexe getallen. De reden is dat dit geval de casus irreducibilis is, waarbij van een derdegraads polynoom de reële nulpunten niet weer te geven zijn met n -demachtswortels van reële getallen. Hierdoor is het in dit geval nodig complexe getallen te gebruiken. Bij de wortels in de expressies worden in dit geval de wortels met minimaal argument bedoeld.

Het tweede en derde nulpunt liggen voor grote n boven $\frac{1}{2}n$: als we $\frac{1}{2}n$ van (3) aftrekken, en de limiet naar oneindig nemen krijgen we $\frac{3}{2} - \frac{5}{6n} + O((\frac{1}{n})^2)$. Hetzelfde doen voor (4) geeft $\frac{1}{2} + \frac{1}{6n} + O((\frac{1}{n})^2)$. Omdat we v en w symmetrisch kiezen met $v < w$, geldt $v < \frac{1}{2}n$, en kunnen we deze nulpunten dus negeren.

Het eerste nulpunt verschilt niet veel van $\frac{1}{8}n$: als we $\frac{1}{8}n$ van (2) aftrekken en de limiet naar oneindig nemen krijgen we $\frac{1}{4} + \frac{2}{3n} + O((\frac{1}{n})^2)$.

Om aan te tonen dat het stationaire punt in (2) een maximum is, bekijken we de tweede afgeleide naar v , die gelijk is aan

$$\frac{1}{6}n \left(-8 + \frac{12n}{(2 + n - 2v)^3} \right).$$

Dit is negatief voor $v = \frac{1}{8}n + \frac{1}{4}$. Dus rond (2) is de tweede afgeleide negatief, waardoor er in (2) een maximum moet zijn.

Dit betekent dat $v = \frac{1}{8}n$, $w = n - v + 1$ een goede benadering is voor de optimale tak om toe te voegen. Ook voor netwerken die sterk op een padgraaf lijken kunnen we de optimale tak hiermee benaderen. Dit betekent ook dat we om de optimale tak te vinden in de padgraaf slechts een klein aantal paren rond $(\frac{1}{8}n, \frac{7}{8}n)$ hoeven te bekijken. Voor deze eenvoudige graaf hebben we dus een manier gevonden om de hoeveelheid rekenwerk sterk te verlagen. Mogelijk kan dit resultaat gebruikt worden om iets soortgelijks af te leiden voor andere soorten netwerken.

6 Conclusie

We hebben verschillende algoritmes gevonden om te bepalen welke tak het beste aan een netwerk toegevoegd kan worden zo dat de effectieve weerstand zo laag mogelijk is. Voor algemene netwerken kan dit met worst case tijdscomplexiteit $O(n^4)$, met behulp van een formule die we gevonden hebben via matrixvegen. Voor bomen zonder gewichten hebben we een methode gevonden waarmee een gedeelte van de berekening sneller kan worden gedaan, maar die nog steeds een worst case tijdscomplexiteit van $O(n^4)$ heeft. Deze methode maakt gebruik van een random wandelaar. Daarnaast is er voor bomen een andere methode met tijdscomplexiteit $O(n^4)$, die gebruik maakt van de matrix-treestelling. Voor de padgraaf met n takken blijkt dat de optimale tak om toe te voegen ongeveer bij $(\frac{1}{8}n, \frac{7}{8}n)$ ligt. Mogelijk kan dit resultaat als heuristisch gebruikt worden bij soortgelijke netwerken.

Referenties

- [1] Ellens, W.: *Effective resistance and other graph measures for network robustness*. scriptie, Mathematical Institute, University of Leiden, april 2011.
- [2] Spieksma, F.M.: *Netwerken*, hoofdstuk 3, Storingsgevoeligheid van netwerken, pagina's 41–60. Platform Wiskunde Nederland, 2016.