



Universiteit
Leiden
The Netherlands

Hamiltongrafen

Brinkman, L.

Citation

Brinkman, L. (2011). *Hamiltongrafen*.

Version: Not Applicable (or Unknown)

License: [License to inclusion and publication of a Bachelor or Master thesis in the Leiden University Student Repository](#)

Downloaded from: <https://hdl.handle.net/1887/3596714>

Note: To cite this publication please use the final published version (if applicable).



UNIVERSITEIT LEIDEN

BACHELORSCRIPTIE

Hamiltongrafen

Filarstructuur en Markov beslissingsketens

Door:
Lucho BRINKMAN

Begeleidster:
Dr. Floske SPIEKSMAN

10 augustus 2011

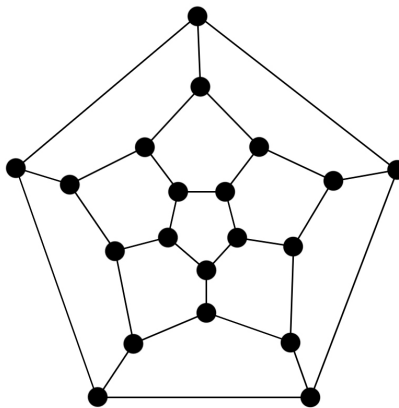
Inhoudsopgave

1	Introductie	2
2	Begrippen en notatie	3
2.1	Grafen	3
2.2	Wandelingen, paden en cykels	3
2.3	Bipartiete grafen en matchings	3
2.4	Lineaire Algebra	4
2.5	Markov beslissingsketens	4
3	Filarstructuur	5
3.1	Incidentiematrix van G	5
3.2	Dataset van Eigenwaarden	6
3.3	Het gemiddelde van de incidentiematrix	8
3.4	De variantie van de incidentiematrix	10
3.5	Hamiltongrafen in de multifilar structuur	11
4	Markov beslissingsketens	15
4.1	Lineaire programmering van Markov beslissingsketens	15
4.2	Convexe combinaties van oplossingen	16
4.3	Experimentele methoden	19
4.4	Verwachte aantal bezoeken in Markov beslissingsketens	20

Hoofdstuk 1

Introductie

In 1857 ontwierp Sir William Rowan Hamilton (1805-1865) het spel 'icosian', dit spel was het begin van één van de moeilijkste problemen in de grafentheorie, het Hamiltongraafprobleem. In icosian zijn er 20 steden, weergegeven als punten, en verbindingen tussen deze steden, weergegeven door lijnen (zie Fig. 1.1). De speler moet nu een route over de lijnen maken waarbij hij elke stad precies één keer bezoekt, en weer uitkomt waar hij begonnen was. In de moderne grafentheorie kan een generaliseerde



Figuur 1.1: Icosian door William Hamilton

versie van het icosian spel als volgt gedefiniëerd worden: *Gegeven een graaf, vind een cykel die elk punt precies één keer bezoekt of bewijs dat dit niet mogelijk is.* Dit wordt het Hamiltongraafprobleem (Of Hamiltoncykelprobleem, HCP) genoemd, een graaf welke een dergelijke cykel, een Hamiltoncykel, bevat wordt een Hamiltongraaf genoemd. Hoewel er algoritmes bestaan om voor willekeurige grafen het HCP op te lossen, is er vooralsnog niet één die dit in polynomiale tijd kan (een algoritme is een polynomiale tijd-algoritme als het aantal stappen waarin het algoritme termineerd kleiner is dan $p(n)$ met p een polynoom en n het aantal knopen van de graaf). Het is duidelijk dat, gegeven een oplossing van het HCP voor een zekere graaf, we deze oplossing in polynomiale tijd op juistheid kunnen controleren. Dus het HCP zit in de complexiteitsklasse \mathcal{NP} (non-deterministisch polynomiaal oplosbare problemen) sterker nog, het blijkt dat het HCP een \mathcal{NP} -hard probleem is. Dit betekent dat als er polynomiaal algoritme voor het HCP gevonden wordt, of als er bewezen wordt dat en dergelijk algoritme niet bestaat, het $\mathcal{P}=\mathcal{NP}$ -probleem opgelost is. Het $\mathcal{P}=\mathcal{NP}$ -probleem is één van de zeven millenium prize problems waarvoor, voor elk van deze problemen, het Clay Mathematics Institute een US\$1.000.000 prijs heeft uitgelooft.

Hoofdstuk 2

Begrippen en notatie

2.1 Grafen

Een graaf G is een koppel (V, E) met V een eindige verzameling en E een deelverzameling van $\{e \subset V : |e| = 2\}$. De elementen van V worden de punten, knopen of knooppunten van G genoemd en de elementen van E worden de lijnen, takken of verbindingen van G genoemd, de twee elementen van een $e \in E$ worden de uiteinden van de tak e genoemd. Een tak $e = \{u, v\}$ zullen we ook wel aangeven met uv . Let op dat er met deze definitie van een graaf zowel dubbele verbindingen tussen punten als takken met twee gelijke uiteinden niet worden toegelaten. Als de knooppuntenverzameling van een graaf G niet benoemd is, zullen we deze aanduiden met $V(G)$, op dezelfde manier noemen we zijn takkenverzameling $E(G)$.

We noemen $|V(G)|$ de orde van G en het aantal takken met een uiteinde v voor een zekere $v \in V(G)$ noemen we de graad van v of $d(v)$. Een graaf waarbij elk punt graad k heeft, is een k -reguliere graaf. De graaf $G[U]$ met $U \subset V(G)$ is de graaf $G = (U, F)$ met $F = \{ij \in E(G) : i, j \in U\}$.

2.2 Wandelingen, paden en cykels

Een wandeling (van lengte k) in een graaf G is een alternerende rij $v_0 e_0 v_1 e_1 \dots e_{k-1} v_k$ met $v_i \in V(G)$ (voor $i = 0, \dots, k$) en $e_i = v_i v_{i+1} \in E(G)$ (voor $i = 0, \dots, k-1$). We zullen deze rij ook wel noteren met $v_0 v_1 \dots v_k$. Als $v_0 = v_k$ dan is de wandeling gesloten. Een wandeling waarbij alle v_i verschillend zijn is een pad en een gesloten wandeling waarbij alle v_i verschillend zijn (behalve v_0 en v_k natuurlijk) is een cykel. Een Hamiltoncykel tot slot, is een cykel van lengte $|V(G)|$.

2.3 Bipartiete grafen en matchings

Een bipartiete graaf G is een graaf welke twee partities $A, B \subset V(G)$ heeft zodat $A \cap B = \emptyset$, $A \cup B = V(G)$ en $E(G[A]) = E(G[B]) = \emptyset$. Voor een willekeurige graaf G noemen een verzameling $M \subset E(G)$ een matching als voor elke twee takken $e_1, e_2 \in M$ met $e_1 \neq e_2$ geldt dat ze geen uiteinde gemeen hebben; $e_1 \cap e_2 = \emptyset$. Een M -alternerend pad is een pad in G dat begint in een tak $e \notin M$ en om de beurt een tak uit $E(G) \setminus M$ en M bezoekt. Let op dat een M -alternerend pad niet alle takken van M hoeft te bevatten.

Een vertex cover van G is een verzameling $U \subset V(G)$ met de eigenschap dat voor elke $ij \in E(G)$ geldt dat $i \in U$ of $j \in U$.

2.4 Lineaire Algebra

Het element van een matrix $M \in \mathbb{C}^{n \times n}$ op de i -de rij en j -de kolom noteren we met $M_{i,j}$ of M_{ij} . De getransponeerde matrix van M noteren we met M^T en \bar{M} is de matrix waarbij $\bar{M}_{ij} = \overline{M_{ij}}$, de complex geconjugeerde van M_{ij} is (voor $i, j = 1, \dots, n$).

Op dezelfde manier noteren we het i -de element van een vector $x \in \mathbb{C}^n$ met x_i . De getransponeerde vector van x noteren we met x^T en \bar{x} is de vector met $\bar{x}_i = \overline{x_i}$ de complex geconjugeerde van x_i (voor $i = 1, \dots, n$).

Met $\langle \cdot, \cdot \rangle$ bedoelen we het standaard inproduct. Dus voor $x, y \in \mathbb{C}^n$ geldt $\langle x, y \rangle = \sum_{i=1}^n x_i \bar{y}_i$ en $\|\cdot\|$ is de standaardnorm ofwel $\|x\| = \sqrt{\langle x, x \rangle}$. De determinant van een matrix M geven we aan met $\det(M)$ en het spoor van M noteren we met $Tr[M]$, dus $Tr[M] = \sum_{i=1}^n M_{ii}$.

We zullen hier vaak het concept van de incidentiematrix van een graaf G gebruiken. Deze noteren we met $A(G)$ of kortweg A en is als volgt gedefiniëerd:

$$A(G)_{ij} = \begin{cases} 1 & \text{als } ij \in E(A(G)) \\ 0 & \text{als } ij \notin E(A(G)). \end{cases}$$

Een stochastische matrix is een $n \times n$ matrix M met $\sum_{j=1}^n M_{ij} = 1$ (voor $i = 1, \dots, n$) en een dubbel stochastische matrix is een matrix M waarvoor zowel M als M^T stochastisch is. Een permutatie matrix is een dubbel stochastische $n \times n$ matrix P met $P_{ij} \in \{0, 1\}$ (voor $i, j = 1, \dots, n$).

2.5 Markov beslissingsketens

Een Markov keten over een eindige toestandsruimte S is een rij van random variabelen X_1, X_2, \dots waarvoor de volgende eigenschap geldt:

$$\mathbb{P}(X_{n+1} = x | X_1 = x_1, X_2 = x_2, \dots, X_n = x_n) = \mathbb{P}(X_{n+1} = j | X_n = i)$$

met $x, x_1, x_2, \dots, x_n \in S$.

Bij een Markov beslissingsketen (MBK) is er niet één Markovketen, maar hebben we te maken met een aantal Markov ketens en moeten we beslissen welke de ‘beste’ is voor een gegeven optimaliteitscriterium. We hebben weer een eindige toestandsruimte $S = \{1, 2, \dots, n\}$, en voor elke $i \in S$ een eindige actieverzameling $H(i)$. Hierin is er voor elke $a \in H(i)$ een opbrengst $r_i(a)$ die wordt toegekend als we actie a kiezen in toestand i en overgangskansen $p_{ij}(a) = \mathbb{P}(X_{t+1} = j | X_t = i, a)$ voor $j \in S$, de kans dat we in de volgende tijdstap in toestand j zitten als we in toestand i actie a kiezen.

Een deterministische stationaire strategie is een vector $\pi \in H(1) \times H(2) \times \dots \times H(n)$ die we kunnen toepassen op de MBK. Merk dat als de strategie vast ligt het resulterende proces opgevat kan worden als een Markov keten.

Hoofdstuk 3

Filarstructuur

In dit hoofdstuk zullen we een aantal opmerkelijke resultaten van Jerzy Filar [1] tegenkomen. Deze sectie zou beschouwd kunnen worden als een inleiding tot zijn onderzoek. Hiervoor zullen we in dit hoofdstuk met G altijd een 3-reguliere samenhangende graaf bedoelen met $|V(G)| = n$ knooppunten.

3.1 Incidentiematrix van G

Ten eerste zullen we de incidentiematrix A van G bekijken en een aantal eigenschappen van de matrix $P(G) = \frac{1}{3}A$ (of P als specificatie niet nodig is) uitlichten. De reden dat we P bekijken in plaats van A is dat P een stochastische matrix is (i.e. elke rij heeft som 1). Herinner namelijk dat elke rij van de incidentiematrix A van een 3-reguliere graaf op precies drie 1-en na, gelijk is aan 0. Nu kunnen we het volgende zeggen over de eigenwaarden van P .

Stelling 3.1.1 *Voor elke eigenwaarde λ van P geldt dat: $\lambda \in [-1, 1] \subset \mathbb{R}$.*

Bewijs Laat $\lambda \in \mathbb{C}$ een willekeurige eigenwaarde van P zijn met bijbehorende eigenvector $x \in \mathbb{C}^n$. Eerst zullen we bewijzen dat $\lambda \in \mathbb{R}$ (we bekijken hierbij \mathbb{R} als deelverzameling van \mathbb{C}). We zien dat:

$$\langle x, Px \rangle = x^T \overline{(Px)} = x^T \overline{P} \overline{x} = x^T P \overline{x} = x^T P^T \overline{x} = (Px)^T \overline{x} = \langle Px, x \rangle.$$

Hier hebben we gebruikt dat P reëelwaardig is $\overline{P} = P$ en dat P symmetrisch is $P^T = P$. Ook zien we dat:

$$\langle \lambda x, x \rangle = (\lambda x)^T \overline{x} = \lambda x^T \overline{x} = \lambda \langle x, x \rangle = \lambda \|x\|^2$$

en

$$\langle x, \lambda x \rangle = (x)^T \overline{\lambda x} = \overline{\lambda} x^T \overline{x} = \overline{\lambda} \langle x, x \rangle = \overline{\lambda} \|x\|^2.$$

Omdat λ een eigenwaarde is van P met eigenvector x geldt nu:

$$\lambda \|x\|^2 = \langle \lambda x, x \rangle = \langle Px, x \rangle = \langle x, Px \rangle = \langle x, x \lambda \rangle = \overline{\lambda} \|x\|^2.$$

Omdat $x \neq 0$ geldt nu dat $\lambda = \overline{\lambda}$. Dus $\lambda \in \mathbb{R} \subset \mathbb{C}$. Om te bewijzen dat $|\lambda| \leq 1$ gebruiken we de volgende claim.

CLAIM: Voor $|t| < 1$ geldt:

$$(I - tP)^{-1} = \sum_{k=0}^{\infty} t^k P^k.$$

BEWIJS (CLAIM):

$$(I - tP) \sum_{k=0}^{\infty} t^k P^k = \sum_{k=0}^{\infty} t^k P^k - \sum_{k=0}^{\infty} (tP) t^k P^k = \sum_{k=0}^{\infty} t^k P^k - \sum_{k=1}^{\infty} t^k P^k = I$$

en

$$\left[\sum_{k=0}^{\infty} t^k P^k \right] (I - tP) = \sum_{k=0}^{\infty} t^k P^k + \sum_{k=0}^{\infty} t^k P^k (-tP) = \sum_{k=0}^{\infty} t^k P^k - \sum_{k=1}^{\infty} t^k P^k = I.$$

Deze gelijkheden zijn allen gerechtvaardigd als elk van deze oneindige sommen convergeert. Dit is zeker het geval, want P is stochastisch. Dus voor elke $n \in \mathbb{N}$ geldt dat P^n stochastisch is. Dit betekent dat $p_{ij}^{(n)} \leq 1$ voor elk element $p_{ij}^{(n)}$ van P^n . Bovendien hebben we aangenomen dat $|t| < 1$, dus voor elk element q van $\sum_{n=0}^{\infty} t^n P^n$ geldt:

$$q \leq \sum_{k=0}^{\infty} t^k \cdot 1 < \infty.$$

Dit bewijst de claim.

Stel nu dat $|\lambda| > 1$ dan geldt:

$$0 = \det(P - \lambda I) = \det(-\lambda(I - \lambda^{-1}P)) = (-\lambda)^n \det(I - \lambda^{-1}P).$$

Omdat $(-\lambda)^n \neq 0$ is dit in tegenspraak met de claim. Immers, voor $|\frac{1}{\lambda}| < 1$ heeft $(I - \frac{1}{\lambda}P)$ een inverse en dus is $\det(I - \frac{1}{\lambda}P) \neq 0$. We kunnen nu concluderen dat $\lambda \in [-1, 1] \subset \mathbb{R}$.

□

3.2 Dataset van Eigenwaarden

Om tot één van de belangrijkste bevindingen van het onderzoek van Jerzy Filar te komen, zullen we nu de dataset van eigenwaarden van $P = \frac{1}{3}A$ bekijken. Deze bestaat dus uit n (mogelijk gelijke) waarden uit het interval $[-1, 1]$. Nu definiëren we eerst het gemiddelde en de variantie van een n bij n matrix M met eigenwaarden $\lambda_1(M), \lambda_2(M), \dots, \lambda_n(M)$ respectievelijk, als:

$$\mu(M) = \frac{1}{n} \sum_{i=1}^n \lambda_i(M)$$

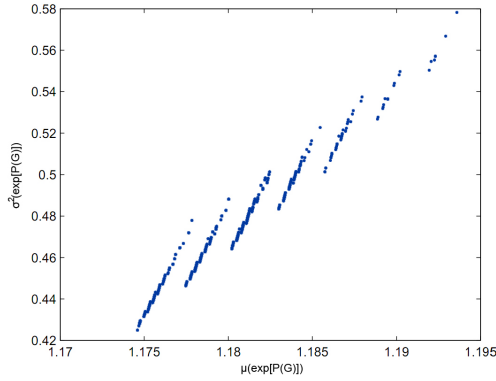
en

$$\sigma^2(M) = \frac{1}{n} \sum_{i=1}^n [\lambda_i(M) - \mu(M)]^2.$$

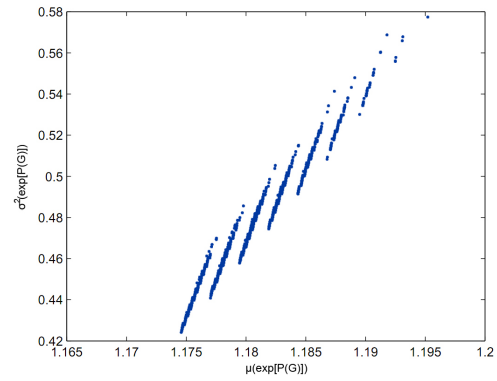
Als we nu $\mu(e^{P(G)})$ en $\sigma^2(e^{P(G)})$ bekijken en deze vergelijken met de $\mu(e^{P(H)})$ en $\sigma^2(e^{P(H)})$ voor elke andere 3-reguliere samenhangende graaf H met $V(H) = n$, dan zullen we een opmerkelijk verband te zien krijgen.

Voorbeeld Op isomorfie na zijn er (volgens [2]) 509 verschillende 3-reguliere samenhangende grafen G met 14 knopen. Als we voor elk van deze grafen $\sigma^2(e^{P(G)})$ uitzetten tegen $\mu(e^{P(G)})$ krijgen we de plot in Fig. 3.1. En als we hetzelfde doen voor de 4060 (volgens [2]) verschillende 3-reguliere samenhangende grafen met 16 knopen, krijgen we Fig. 3.2. We zien hier dus een bijzondere lineaire structuur ontstaan, opgedeeld in een aantal clusters welke filars worden genoemd. Als we verder inzoomen bij Fig. 3.1 zien we (Fig. 3.3) binnen de eerste filar weer dezelfde structuur ontstaan. Als we weer verder inzoomen (Fig. 3.4) zien we dat deze structuur zich herhaalt.

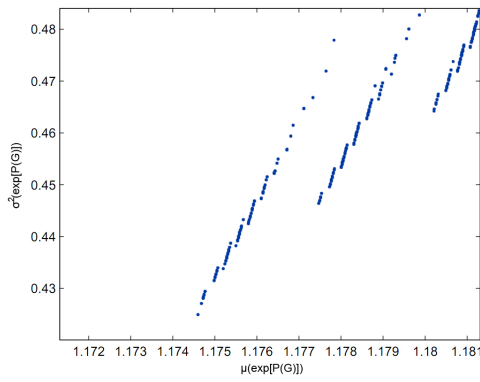
Deze structuur van filars, met daarbinnen zogenaamde subfilars, wordt multifilar genoemd. Een computersimulatie wijst erop dat dit een universeel gedrag is voor 3-reguliere samenhangende grafen van zekere orde n . Wat verder opmerkelijk is aan deze structuur is, dat de locatie van een bepaalde graaf in de multifilar wordt bepaald door het aantal cycli dat hij bevat. Om dit in te zien definiëren we de verzameling k -reguliere samenhangende grafen van orde n : $G_k[n]$. Vervolgens definiëren we de functie $f : G_3[n] \rightarrow \mathbb{N}^n$ door $f(G) = (0, 0, g_3, g_4, \dots, g_n)$ desda G precies g_i cycli van lengte i bevat (voor $i = 3, \dots, n$).



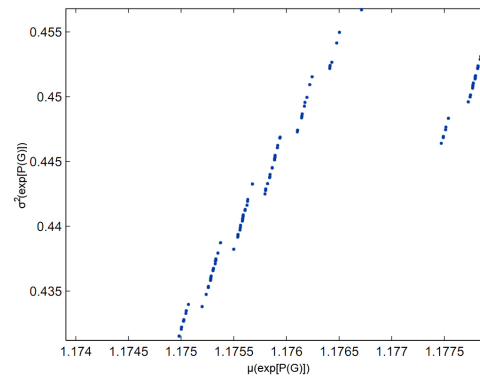
Figuur 3.1: μ - σ^2 plot voor $n = 14$



Figuur 3.2: μ - σ^2 plot voor $n = 16$



Figuur 3.3: Uitvergroting van eerste filar van Fig. 3.1



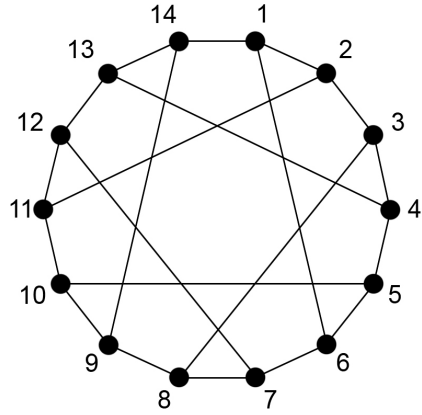
Figuur 3.4: Verdere uitvergroting van eerste filar van Fig. 3.1

Bewering In de multifilar structuur bevindt het punt $(\mu(G), \sigma^2(G))$ zich in de $(f(G)_3 + 1)$ -de filar, daarbinnen in de $(f(G)_4 + 1)$ -de subfilar, daarbinnen in de $(f(G)_5 + 1)$ -de subfilar enzovoorts. Let hierbij op dat niet elke (sub)filar persé bestaat, dus uit een μ - σ -plot zoals Fig. 3.1 en 3.2 kunnen we niet altijd afleiden wat de cyclen van een bepaalde graaf zijn.

Behalve opmerkelijk is dit ook een belangrijke opmerking voor het hamiltongraafprobleem, omdat een graaf een hamiltongraaf is dan en slechts dan als deze een cykel van lengte n bevat.

Voorbeeld Neem de graaf G uit Fig. 3.5. Deze heeft incidentiematrix:

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$



Figuur 3.5: De graaf G met incidentiematrix A

met eigenwaarden $-3, -\sqrt{2}, -\sqrt{2}, -\sqrt{2}, -\sqrt{2}, -\sqrt{2}, -\sqrt{2}, \sqrt{2}, \sqrt{2}, \sqrt{2}, \sqrt{2}, \sqrt{2}, \sqrt{2}$ en 3 . Met behulp van de volgende propositie kunnen we gemakkelijk $\mu(e^{\frac{1}{3}A})$ en $\sigma^2(e^{\frac{1}{3}A})$ uitrekenen.

Propositie 3.2.1 *Voor een symmetrische reële $n \times n$ matrix M met eigenwaarden $\lambda_1, \lambda_2, \dots, \lambda_n$ geldt dat e^M eigenwaarden $e^{\lambda_1}, e^{\lambda_2}, \dots, e^{\lambda_n}$ heeft.*

Bewijs Merk ten eerste op dat -omdat M symmetrisch en reëel is- we de eigenvectoren van M zo kunnen kiezen dat deze een orthonormale basis voor \mathbb{R}^n vormen. Laat x_i een dergelijke eigenvector zijn, behorende bij eigenwaarde λ_i . Dan geldt voor $i = 1, 2, \dots, n$:

$$e^A x_i = \left[\sum_{k=0}^{\infty} \frac{A^k}{k!} \right] x_i = \sum_{k=0}^{\infty} \frac{A^k x_i}{k!} = \sum_{k=0}^{\infty} \frac{\lambda_i^k x_i}{k!} = \left[\sum_{k=0}^{\infty} \frac{\lambda_i^k}{k!} \right] x_i = e^{\lambda_i} x_i.$$

De derde gelijkheide krijgen we door herhaaldelijk $Ax_i = \lambda_i x_i$ toe te passen. De eigenwaarden van e^A zijn dus $e^{\lambda_1}, e^{\lambda_2}, \dots, e^{\lambda_n}$ met bijbehorende eigenvectoren resp. x_1, x_2, \dots, x_n .

□

Dus het gemiddelde en de variantie worden gegeven door:

$$\mu(e^{\frac{1}{3}A}) = \frac{1}{14} \left[e^{-1} + 6e^{-\frac{1}{3}\sqrt{2}} + 6e^{\frac{1}{3}\sqrt{2}} + e^1 \right] \approx 1.1746$$

en

$$\begin{aligned} \sigma^2(e^{\frac{1}{3}A}) &= \frac{1}{14} \left[(e^{-1} - \mu(e^{\frac{1}{3}A}))^2 + 6(e^{-\frac{1}{3}\sqrt{2}} - \mu(e^{\frac{1}{3}A}))^2 + 6(e^{\frac{1}{3}\sqrt{2}} - \mu(e^{\frac{1}{3}A}))^2 + (e^1 - \mu(e^{\frac{1}{3}A}))^2 \right] \\ &\approx 0.4249. \end{aligned}$$

In Fig. 3.3 zien we dat $(\mu(e^{\frac{1}{3}A}), \sigma^2(e^{\frac{1}{3}A}))$ zich in de eerste filar bevindt en daarbinnen in de eerste subfilar. Inderdaad geldt ook dat G precies 0 cyclen van lengte 3 bevat en 0 cyclen van lengte 4.

3.3 Het gemiddelde van de incidentiematrix

We bekijken in dit hoofdstuk de matrix $(I - tA)^{-1}$ waarbij A weer de incidentiematrix van een 3-reguliere samenhangende graaf van orde n is, met eigenwaarden $\lambda_1, \lambda_2, \dots, \lambda_n$ en $t \in (0, \frac{1}{3})$. In het bewijs van de volgende propositie zullen we zien dat deze matrix ook daadwerkelijk bestaat.

Propositie 3.3.1 *Als $\lambda_1, \lambda_2, \dots, \lambda_n$ de eigenwaarden van A zijn, dan zijn $(1-t\lambda_1)^{-1}, (1-t\lambda_2)^{-1}, \dots, (1-t\lambda_n)^{-1}$ de eigenwaarden van $(I-tA)^{-1}$.*

Bewijs Omdat $(I-tA)^{-1} = [I-(3t)(\frac{1}{3}A)]^{-1}$ en $t \in (0, \frac{1}{3})$ kunnen we net als in Stelling 3.1.1 gebruiken dat $(I-tA)^{-1} = \sum_{k=0}^{\infty} t^k A^k$. Bovendien is A symmetrisch en reëel, dus kunnen we zijn eigenvectoren zodanig kiezen dat deze een orthonormale basis vormen voor \mathbb{R}^n . Laat x_i (voor $i = 1, 2, \dots, n$) een dergelijke eigenvector van A zijn, behorende bij de eigenwaarde λ_i van A . Dan zien we dat:

$$(I-tA)^{-1}x_i = \left[\sum_{k=0}^{\infty} t^k A^k \right] x_i = \sum_{k=0}^{\infty} t^k (A^k x_i) = \sum_{k=0}^{\infty} t^k (\lambda_i^k x_i) = \left[\sum_{k=0}^{\infty} t^k \lambda_i^k \right] x_i = (1-t\lambda_i)^{-1} x_i.$$

De derde gelijkheid krijgen we door herhaaldelijk $Ax_i = \lambda_i x_i$ toe te passen en de laatste gelijkheid geldt omdat $t\lambda_i < 1$ (Stelling 3.1.1). Dus de eigenwaarden van $(I-tA)^{-1}$ zijn $(1-t\lambda_1)^{-1}, (1-t\lambda_2)^{-1}, \dots, (1-t\lambda_n)^{-1}$ met bijbehorende eigenvectoren resp. x_1, x_2, \dots, x_n . □

Nu we weten dat elke eigenwaarde van $(I-tA)^{-1}$ bepaald wordt door een eigenwaarde van A kunnen we concluderen dat we weer eenzelfde multifilarstructuur krijgen als bij de matrices $e^{\frac{1}{3}A}$. De reden dat we nu deze andere matrix gebruiken is dat dit later goed uitkomt als we het spoor van $(I-tA)^{-1}$ bekijken. De vraag is nu, waarom de gemiddelden van twee incidentiematrices 'dicht' bij elkaar liggen als de bijbehorende grafen een gelijk aantal cycli van lengte 3, 4, ..., j hebben, met $3 \leq j \leq n$. Daarvoor bekijken we de volgende uitdrukking voor het gemiddelde:

$$\mu((I-tA)^{-1}) = \frac{1}{n} \sum_{i=1}^n (1-t\lambda_i)^{-1} = \frac{1}{n} \text{Tr}[(I-tA)^{-1}] = \frac{1}{n} \text{Tr}\left[\sum_{k=0}^{\infty} t^k A^k\right] = \frac{1}{n} \sum_{k=0}^{\infty} t^k \text{Tr}[A^k].$$

Stelling 3.3.2 *Het aantal wandelingen van i naar j van lengte l in een graaf G met incidentiematrix A is gelijk aan $(A^l)_{ij}$.*

Bewijs We gaan dit bewijzen met inductie naar l . Voor $l = 1$ geldt:

$$(A^1)_{ij} = \begin{cases} 1 & \text{als } ij \in E \\ 0 & \text{als } ij \notin E. \end{cases}$$

Dus $(A^1)_{ij}$ is precies het aantal mogelijke wandelingen van i naar j van lengte 1. Neem nu $l > 1$ en stel dat de stelling klopt voor $(l-1)$. Dan geldt:

$$(A^l)_{ij} = (A^{l-1})_{i1}A_{1j} + (A^{l-1})_{i2}A_{2j} + \dots + (A^{l-1})_{in}A_{nj} = \sum_{k=1}^n \mathbf{1}_{\{kj \in E\}} (A^{l-1})_{ik}.$$

Dit is dus het aantal wandelingen van lengte $(l-1)$ van i naar de burens van j . Ofwel het aantal wandelingen van lengte l van i naar j . □

Deze stelling zegt ons onder anderen dat het aantal gesloten wandelingen van lengte l beginnend in i gelijk is aan $(A^l)_{ii}$. Nu kunnen we het gemiddelde dus schrijven als

$$\mu((I-tA)^{-1}) = \frac{1}{n} \sum_{k=0}^{\infty} t^k \text{Tr}[A^k] = \frac{1}{n} \sum_{k=0}^{\infty} t^k p_k,$$

met p_k het aantal gesloten wandelingen van lengte k . Het is niet moeilijk te geloven dat er een verband is tussen het aantal gesloten wandelingen van lengte k en het aantal cykels van lengte k . Als we het aantal cykels van lengte k aangeven met m_k , dan wordt dit verband gegeven door

$$\mu((I - tA)^{-1}) = \frac{1}{n} \sum_{k=0}^{\infty} t^k p_k = H(t) + \frac{2}{n} \sum_{k=3}^n m_k F_k(t)$$

waarbij

$$H(t) = \frac{3\sqrt{1-8t^2} - 1}{2(1-9t^2)}$$

en

$$F_k(t) = \frac{k(1 - \sqrt{1-8t^2})^k}{\sqrt{1-8t^2}[(4t)^k - (1 - \sqrt{1-8t^2})^k]}.$$

Zie [1] voor een volledige uitwerking van deze gelijkheid. In [1] wordt verder opgemerkt dat $F_k(t)$ voor een vaste $t \in (0, \frac{1}{3})$ snel daalt als k groeit. Bijvoorbeeld $F_k(\frac{1}{9}) \leq c10^{-k}$ voor een zekere positieve constante c . Nu zien we dat voor twee grafen G en H de waarden $\mu((I-tA(G))^{-1})$ en $\mu((I-tA(H))^{-1})$ dicht bij elkaar liggen als deze een gelijk aantal cykels van bepaalde lengte hebben. Hierbij telt m_3 veel zwaarder mee dan het aantal cykels van grotere lengte. Als voor G en H de m_3 gelijk is, is m_4 de dominante factor in $\mu((I-tA(G))^{-1}) - \mu((I-tA(H))^{-1})$, daarna m_5 enzovoorts.

3.4 De variantie van de incidentiematrix

Nu rest ons nog om voor $(I - tA)^{-1}$ met $(1 - t\lambda_1)^{-1}, (1 - t\lambda_2)^{-1}, \dots, (1 - t\lambda_n)^{-1}$ zijn eigenwaarden, $t \in (0, \frac{1}{3})$ en A de incidentiematrix van G een zelfde soort resultaat te vinden voor de variantie als we in de vorige sectie voor het gemiddelde vonden. We schrijven daartoe de variantie als

$$\begin{aligned} \sigma^2((I - tA)^{-1}) &= \frac{1}{n} \sum_{i=1}^n [(1 - t\lambda_i)^{-1} - \mu((I - tA)^{-1})]^2 \\ &= \frac{1}{n} \sum_{i=1}^n [(1 - t\lambda_i)^{-1}]^2 + \frac{1}{n} \sum_{i=1}^n [\mu((I - tA)^{-1})]^2 - \frac{2}{n} \sum_{i=1}^n [(1 - t\lambda_i)^{-1} \mu((I - tA)^{-1})] \\ &= \frac{1}{n} \sum_{i=1}^n [(1 - t\lambda_i)^{-1}]^2 + \mu^2((I - tA)^{-1}) - 2\mu^2((I - tA)^{-1}) \\ &= \frac{1}{n} \sum_{i=1}^n [(1 - t\lambda_i)^{-2}] - \mu^2((I - tA)^{-1}). \end{aligned}$$

Merk op dat deze uitdrukking het probleem aanzienlijk versimpeld omdat we $\mu((I - tA)^{-1})$ al weten. Om de term $\frac{1}{n} \sum_{i=1}^n (1 - t\lambda_i)^{-2}$ te vereenvoudigen gebruiken we de volgende propositie.

Propositie 3.4.1 *De eigenwaarden van $(I - tA)^{-2}$ zijn $(1 - t\lambda_1)^{-2}, (1 - t\lambda_2)^{-2}, \dots, (1 - t\lambda_n)^{-2}$ voor de matrix A zoals hierboven met eigenwaarden $\lambda_1, \lambda_2, \dots, \lambda_n$.*

Bewijs De claim uit Stelling 3.1.1 geeft ons dat $(I - tA)^{-2} = [\sum_{k=0}^{\infty} t^k A^k]^2 = \sum_{k,l=0}^{\infty} t^{k+l} A^{k+l}$. Neem x_i (voor $i = 1, 2, \dots, n$) een eigenvector van A als in Propositie 3.3.1, dan zien we dat:

$$(I - tA)^{-2} x_i = \left[\sum_{k,l=0}^{\infty} t^{k+l} A^{k+l} \right] x_i = \sum_{k,l=0}^{\infty} t^{k+l} (A^{k+l} x_i) = \sum_{k,l=0}^{\infty} t^{k+l} (\lambda_i^{k+l} x_i) = \left[\sum_{k,l=0}^{\infty} t^{k+l} \lambda_i^{k+l} \right] x_i = (1 - t\lambda_i)^{-2} v_i.$$

We gebruiken hierin dezelfde argumenten als in Propositie 3.3.1. We concluderen dat $(1 - t\lambda_1)^{-2}, (1 - t\lambda_2)^{-2}, \dots, (1 - t\lambda_n)^{-2}$ de eigenwaarden van $(I - tA)^{-2}$ zijn.

□

Dus:

$$\begin{aligned}
& \sigma^2((I - tA)^{-1}) \\
&= \frac{1}{n} \sum_{i=1}^n [(1 - t\lambda_i)^{-2}] - \mu^2((I - tA)^{-1}) = \frac{1}{n} \text{Tr}[(I - tA)^{-2}] - \mu^2((I - tA)^{-1}) \\
&= \frac{1}{n} \text{Tr}[(I - tA)^{-1}(I - tA)^{-1}] - \mu^2((I - tA)^{-1}) = \frac{1}{n} \text{Tr} \left[\sum_{l=0}^{\infty} t^l A^l \cdot \sum_{l'=0}^{\infty} t^{l'} A^{l'} \right] - \mu^2((I - tA)^{-1}) \\
&= \frac{1}{n} \text{Tr} \left[\sum_{l,l'=0}^{\infty} t^{l+l'} A^{l+l'} \right] - \mu^2((I - tA)^{-1}) = \frac{1}{n} \text{Tr} \left[\sum_{k=0}^{\infty} (k+1) t^k A^k \right] - \mu^2((I - tA)^{-1}) \\
&= \frac{1}{n} \text{Tr} \left[\sum_{k=0}^{\infty} (k+1) t^k A^k \right] - \mu^2((I - tA)^{-1}) = \frac{1}{n} \text{Tr} \left[\frac{d}{dt} \sum_{k=0}^{\infty} t^{k+1} A^k \right] - \mu^2((I - tA)^{-1}) \\
&= \frac{1}{n} \text{Tr} \left[\frac{d}{dt} t(I - tA)^{-1} \right] - \mu^2((I - tA)^{-1}) = \frac{1}{n} \frac{d}{dt} (\text{Tr}[t(I - tA)^{-1}]) - \mu^2((I - tA)^{-1}) \\
&= \frac{1}{n} \frac{d}{dt} (t \text{Tr}[(I - tA)^{-1}]) - \mu^2((I - tA)^{-1}).
\end{aligned}$$

Hierbij is de afgeleide, afhankelijk van het argument, de elementsgewijze differentiatie naar t van een matrix of de standaard differentiatie naar t . Verder krijgen we:

$$\begin{aligned}
\sigma^2((I - tA)^{-1}) &= \frac{1}{n} \frac{d}{dt} (t \text{Tr}[(I - tA)^{-1}]) - \mu^2((I - tA)^{-1}) \\
&= \frac{1}{n} t \frac{d}{dt} (\text{Tr}[(I - tA)^{-1}]) + \frac{1}{n} (\text{Tr}[(I - tA)^{-1}]) - \mu^2((I - tA)^{-1}) \\
&= t \frac{d}{dt} [\mu((I - tA)^{-1})] + \mu((I - tA)^{-1}) - \mu^2((I - tA)^{-1}).
\end{aligned}$$

Substitutie van $\mu((I - tA)^{-1})$ geeft nu

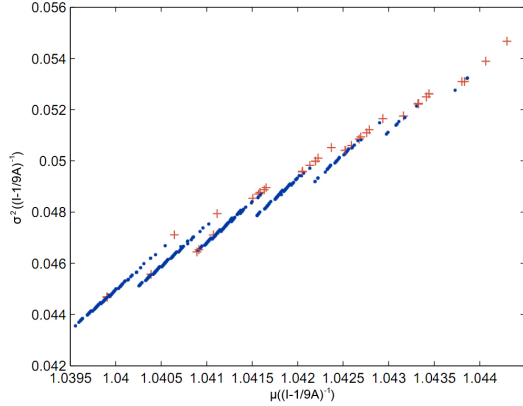
$$\begin{aligned}
\sigma^2((I - tA)^{-1}) &= t \frac{d}{dt} \left[H(t) + \frac{2}{n} \sum_{k=3}^n m_k F_k(t) \right] + H(t) + \frac{2}{n} \sum_{k=3}^n m_k F_k(t) - \left[H(t) + \frac{2}{n} \sum_{k=3}^n m_k F_k(t) \right]^2 \\
&= t \frac{dH}{dt} + \frac{2t}{n} \left[\sum_{k=3}^n m_k \frac{dF_k}{dt} \right] + H(t) + \frac{2}{n} \sum_{k=3}^n m_k F_k(t) - H^2(t) - \left[\frac{2}{n} \sum_{k=3}^n m_k F_k(t) \right]^2 \\
&\quad - \frac{4}{n} H(t) \sum_{k=3}^n m_k F_k(t).
\end{aligned}$$

In [1] wordt aangetoond dat $\frac{dF_k}{dt}$, net als $F_k(t)$, snel daalt als k groter wordt. In het bijzonder geldt $[\frac{dF_k}{dt}]_{t=\frac{1}{9}} \leq c' 10^{-k}$, voor een zekere positieve constante c' . Dus voor een zekere $t \in (0, \frac{1}{3})$, zeg $t = \frac{1}{9}$ geldt, net als bij het gemiddelde (zie vorige sectie), dat het verschil in variantie tussen twee grafen voornamelijk bepaald wordt door de m_k (welke verschillend is voor de grafen) met laagste k .

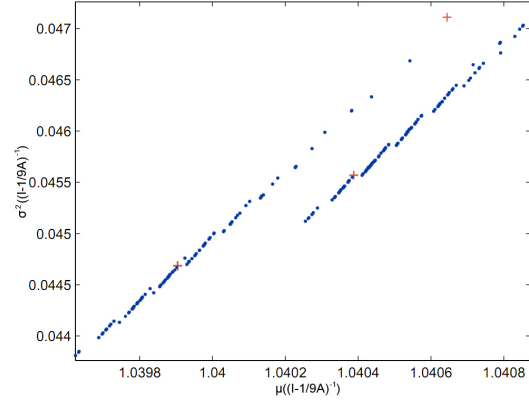
3.5 Hamiltongrafen in de multifilar structuur

In deze sectie zullen we nog een opmerkelijke eigenschap van de multifilar structuur behandelen, één met betrekking tot Hamiltongrafen.

Voorbeeld We bekijken $\mu((I - \frac{1}{9}A)^{-1})$ en $\sigma^2((I - \frac{1}{9}A)^{-1})$ voor de 509 verschillende grafen van orde 14, welke zijn geplot in Fig. 3.6. Hierin zijn de Hamiltongrafen aangeduid met punten en de niet-Hamiltongrafen met een plusteken. Wat opvalt is dat de meeste niet-Hamiltongrafen aan de boven- of onderkant van een filar zitten. Een paar uitzonderingen echter zitten, binnen hun filar, tussen Hamiltongrafen, maar als we verder inzoomen (Fig. 3.7) zien we dat hier een niet-Hamiltongraaf binnen zijn binnenste subfilar altijd helemaal boven- of onderaan zit.



Figuur 3.6: $\mu - \sigma^2$ plot voor $n = 14$ met Hamiltongraaf aanduiding



Figuur 3.7: De eerste twee filars van Fig. 3.6 met Hamiltongraaf aanduiding

Dit blijkt een universeel gedrag te zijn voor niet-Hamiltongrafen.

Vermoeden Voor een niet-Hamiltongraaf G met incidentiematrix A , bevindt $(\mu(A), \sigma^2(A))$ zich -in zijn binnenste subfilar- op de bovenste of onderste zichtbare plek.

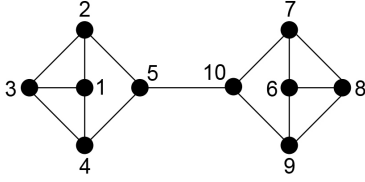
Vooralsnog is het niet duidelijk waarom dit het geval is, maar er is wel (experimenteel) een verband gevonden tussen het soort niet-Hamiltongraaf en of deze aan de boven- of onderkant van zijn binnenste subfilar zit. Daartoe zullen we eerst definiëren wat een bruggraaf is. Een samenhangende graaf G is een bruggraaf als er een $e \in E(G)$ is waarvoor $(V(G), E(G) \setminus \{e\})$ niet samenhangend is. Anders is G een niet-bruggraaf. Merk op dat een bruggraaf zeker geen Hamiltonkring bevat, omdat er maar één pad tussen de twee componenten van $(V(G), E(G) \setminus \{e\})$ bestaat.

Vermoeden Voor een niet-Hamiltongraaf G met incidentiematrix A , bevindt $(\mu(A), \sigma^2(A))$ zich -in zijn binnenste subfilar- op de bovenste zichtbare plek dan en slechts dan als G een bruggraaf is.

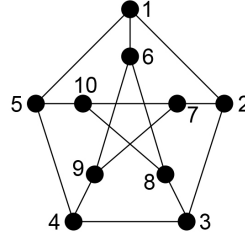
We zullen dit illustreren met een voorbeeld.

Voorbeeld Op isomorfie na, zijn er 19 verschillende 3-reguliere samenhangende grafen van orde 10, hiervan hebben er precies twee geen Hamiltonkring, namelijk G (Fig. 3.8) en G' (Fig. 3.9). G is een bruggraaf en G' is een niet-bruggraaf. De incidentiematrices van G en G' zien er als volgt uit

$$A(G) = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix} \quad A(G') = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}.$$



Figuur 3.8: G , de enige 3-reguliere samenhangende graaf van orde 10 welke een bruggraaf is.

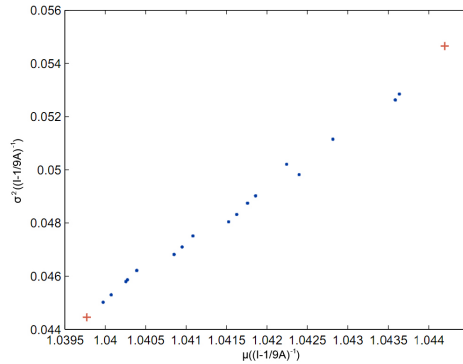


Figuur 3.9: G' , de enige 3-reguliere samenhangende graaf van orde 10 welke een niet-Hamiltongraaf én een niet-bruggraaf is

Bij benadering zijn de eigenwaarden van $A(G)$: $\lambda_1 = -2.4893$, $\lambda_2 = -2$, $\lambda_3 = \lambda_4 = -1$, $\lambda_5 = 0.2892$, $\lambda_6 = \lambda_7 = 0$, $\lambda_8 = 1$, $\lambda_9 = 2.7785$ en $\lambda_{10} = 3$. De eigenwaarden van $A(G')$ zijn: $\lambda'_1 = \lambda'_2 = \lambda'_3 = \lambda'_4 = -2$, $\lambda'_5 = \lambda'_6 = \lambda'_7 = \lambda'_8 = \lambda'_9 = 1$ en $\lambda'_{10} = 3$. Hiermee kunnen we het gemiddelde en de variantie van $(I - \frac{1}{9}A(G))^{-1}$ en $(I - \frac{1}{9}A(G'))^{-1}$ uitrekenen.

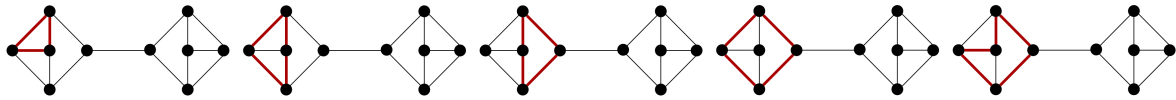
$$\begin{aligned} \mu((I - \frac{1}{9}A(G))^{-1}) &\approx \frac{1}{10} \sum_{i=1}^{10} (1 - \frac{1}{9}\lambda_i)^{-1} \approx 1.0442, \\ \sigma^2((I - \frac{1}{9}A(G))^{-1}) &\approx \frac{1}{10} \sum_{i=1}^{10} [(1 - \frac{1}{9}\lambda_i)^{-1} - 1.0442]^2 \approx 0.0546, \\ \mu((I - \frac{1}{9}A(G'))^{-1}) &\approx \frac{1}{10} \sum_{i=1}^{10} (1 - \frac{1}{9}\lambda'_i)^{-1} \approx 1.0398, \\ \sigma^2((I - \frac{1}{9}A(G'))^{-1}) &\approx \frac{1}{10} \sum_{i=1}^{10} [(1 - \frac{1}{9}\lambda'_i)^{-1} - 1.0398]^2 \approx 0.0445. \end{aligned}$$

In Fig. 3.10 zijn de punten $(1.0442, 0.0546)$ en $(1.0398, 0.0445)$ terug te vinden in de vorm van kruisjes.

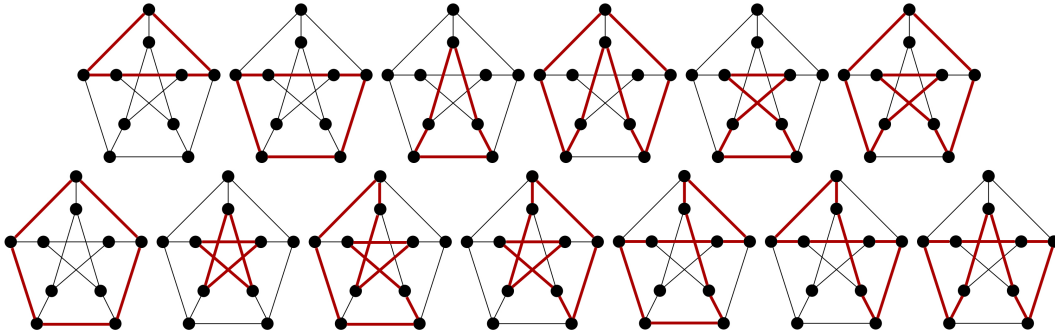


Figuur 3.10: $\mu - \sigma$ plot voor de 19 (op isomorfie na) verschillende grafen van orde 10

Omdat G en G' van relatief lage orde zijn, is het nog te doen om hun cykels te tellen (zie Fig. 3.11 en 3.12). Voor G geldt: $m_3 = 4$, $m_4 = 6$, $m_5 = 4$ en $m_6 = m_7 = m_8 = m_9 = m_{10} = 0$. Voor G' geldt: $m_3 = m_4 = 0$, $m_5 = 12$, $m_6 = 10$, $m_7 = 0$, $m_8 = 15$, $m_9 = 20$ en $m_{10} = 0$. Belangrijker echter, is dat we zien dat G bovenaan zijn binnenste subfilar zit en G' onderaan. Sterker nog, in dit voorbeeld is er geen enkele graaf die boven G zit en geen enkele graaf die onder G' zit.



Figuur 3.11: De verschillende soorten cyclen van de graaf G



Figuur 3.12: De verschillende soorten cyclen van de graaf G'

Een verdere observatie als we Fig. 3.6 bekijken is dat het merendeel van de niet-Hamiltongrafen bruggrafen zijn. Numerieke experimenten wijzen er op dat dit een universeel gedrag is voor 3-reguliere samenhangende grafen van zekere orde, zie tabel 3.1. Wat opvalt aan deze tabel is dat de ratio toeneemt

n	# Grafen	# Niet-H	# Brug	Ratio
10	19	2	1	0.5000
12	85	5	4	0.8000
14	509	35	29	0.8286
16	4060	219	186	0.8493
18	41301	1666	1435	0.8613
20	510489	14498	12671	0.8740
22	7319447	148790	131820	0.8859
24	117940535	1768732	1590900	0.8995

Tabel 3.1: De tweede kolom is het aantal, of isomorfie na, verschillende 3-reguliere samenhangende grafen van orde n , de derde kolom is het aantal niet-Hamiltongrafen daarvan, de vierde kolom is het aantal bruggrafen daarvan en de ratio is $\frac{\#Brug}{\#NietH}$.

met n en als we voor $n = 40$ en $n = 50$ een sample van 1.000.000 3-reguliere samenhangende grafen nemen, zien we dat de ratio hiervoor respectievelijk (ongeveer) 0.9375 en 0.9650 is. Dit leidt tot het volgende vermoeden, waar we dit hoofdstuk mee afsluiten.

Vermoeden Als we het aantal, op isomorfie na, verschillende 3-reguliere samenhangende niet-Hamiltongrafen $N(n)$ noemen, het aantal bruggrafen hiervan $B(n)$ en het aantal niet-bruggrafen hiervan $\overline{B}(n)$ dan:

$$\lim_{n \rightarrow \infty} \frac{B(n)}{N(n)} = \lim_{n \rightarrow \infty} \frac{B(n)}{B(n) + \overline{B}(n)} = 1.$$

Merk op dat dit goed nieuws is omdat we in polynomiale tijd kunnen bepalen of een graaf een bruggraaf is of niet.

Hoofdstuk 4

Markov beslissingsketens

In dit hoofdstuk zullen we een graaf G bekijken als een Markov beslissingsketen in de hoop meer te kunnen zeggen over het wel of niet Hamilton zijn van deze graaf. Eerst zullen we theorie uit [3] toepassen daarna proberen we een polynomiaal algoritme te vinden voor het vinden van Hamiltoncykels.

4.1 Lineaire programmering van Markov beslissingsketens

We nemen een MBK met toestandsruimte $S = \{1, 2, \dots, n\}$, actieverzamelingen $H(i)$ (voor $i \in S$) met opbrengsten $r_i(a)$ en overgangskansen $p_{ij}(a)$ (voor $i, j \in S$ en $a \in H(i)$). Nu kunnen we met behulp van lineaire programmering een strategie vinden waarvoor de gemiddelde opbrengsten minimaal zijn als we de MBK oneindig lang laten doorlopen (gemiddelde opbrengsten over oneindige horizon van een MBK). Het LP-probleem ziet er dan (volgens [3]) als volgt uit:

$$\min \left\{ \sum_{i,a} r_i(a)x_i(a) \left| \begin{array}{l} \sum_{i,a} \{\delta_{ij} - p_{ij}(a)\}x_i(a) = 0, \quad j \in S, \\ \sum_{i,a} x_i(a) = 1, \\ x_i(a) \geq 0, \quad i \in S, \quad a \in H(i) \end{array} \right. \right\}$$

met $\delta_{ij} = 1$ als $i = j$ en $\delta_{ij} = 0$ als $i \neq j$. Wat hier gezegd wordt is dat we in toestand $i \in S$, actie $a \in H(i)$ kiezen met kans $x_i(a) [\sum_{b \in H(i)} x_i(b)]^{-1}$. Nu willen we een Markov beslissingsketen maken aan de hand van een willekeurige samenhangende graaf G . Hiertoe nemen we $V(G)$ als toestandsruimte en voor een knoop $i \in V(G)$ nemen we de actieverzameling $H(i) = \{j : ij \in E(G)\}$ met de volgende overgangskansen $p_{ij}(j) = p_{ij} = 1$ en $p_{ik}(j) = 0$ voor $k \neq j$. Ofwel, de actieverzameling van een knoop bestaat uit de burens van deze knoop. Dit leidt tot het volgende LP-probleem (we schrijven r_{ij} voor $r_i(j)$):

$$\min \left\{ \sum_{i,j} r_{ij}x_{ij} \left| \begin{array}{l} \sum_i x_{ji} = \sum_i x_{ij}, \quad j \in S \\ \sum_{ij} x_{ij} = 1 \\ x_{ij} = 0, \quad ij \notin E(G) \\ x_{ij} \geq 0, \quad i, j \in S \end{array} \right. \right\}.$$

Merk op dat de eerste lineaire beperking ervoor zorgt dat we net zo vaak een knoop verlaten als dat we deze knoop binnenkomen. Vervolgens geeft de tweede beperking ervoor dat x_{ij} de fractie is dat we de actie kiezen die van i naar j gaat van alle actiekeuzes, over alle acties die we over de oneindige horizon kiezen. Het probleem dat nu dus overblijft is de keuze van de opbrengsten r_{ij} en het mogelijk toevoegen van extra (lineaire) beperkingen zodat er een Hamiltonkring geforceerd wordt. Hiertoe zien we dat een Hamiltonkring elke knoop even vaak bezoekt, dus we kunnen de extra beperking $\sum_j x_{ij} = \frac{1}{n}$ toevoegen. Neem vervolgens opbrengsten $r_{i1} = 1$ en $r_{ij} = 0$ (voor $i, j \in S, j \neq 1$) zodat we het

volgende LP-probleem krijgen:

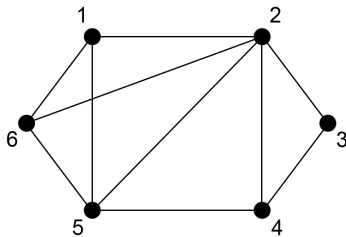
$$\min \left\{ \sum_i x_{i1} \mid \begin{array}{l} \sum_i x_{ji} = \sum_i x_{ij}, \quad j \in S \\ \sum_i x_{ij} = \frac{1}{n}, \quad j \in S \\ x_{ij} = 0, \quad ij \notin E(G) \\ x_{ij} \geq 0, \quad i, j \in S \end{array} \right\}.$$

We kunnen dus als volgt naar dit LP-probleem kijken: we maken oneindige wandelingen over de takken van G waarbij we alle punten even vaak bezoeken. Dus elke Hamiltongraaf is een oplossing voor dit probleem. Er zijn echter twee tekortkomingen aan dit stelsel van lineaire beperkingen, namelijk (1) dat een combinatie van disjuncte cykels (of gesloten wandelingen van lengte 2) een oplossing kan zijn en (2) dat elke convexe combinatie van verschillende oplossingen een oplossing kan zijn. Zoals we nog zullen zien is het deze tweede tekortkoming geen groot probleem, de eerste tekortkoming echter wel.

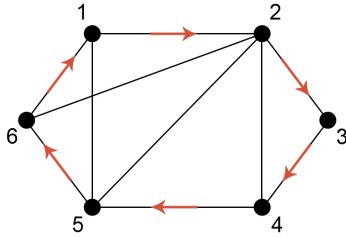
Voorbeeld We bekijken de graaf G zoals in Fig. 4.1. Het LP-probleem ziet er dan als volgt uit:

$$\min \left\{ \begin{array}{l} x_{21} + x_{51} + x_{61} \\ x_{21} + x_{51} + x_{61} \\ x_{12} + x_{32} + x_{42} + x_{52} + x_{62} \\ x_{23} + x_{43} \\ x_{24} + x_{34} + x_{54} \\ x_{15} + x_{25} + x_{45} + x_{65} \\ x_{16} + x_{26} + x_{56} \\ x_{ij} \geq 0, \quad i, j \in V(G) \end{array} \mid \begin{array}{l} x_{12} + x_{15} + x_{16} = x_{21} + x_{51} + x_{61} \\ x_{21} + x_{23} + x_{24} + x_{25} + x_{26} = x_{12} + x_{32} + x_{42} + x_{52} + x_{62} \\ x_{32} + x_{34} = x_{23} + x_{43} \\ x_{42} + x_{43} + x_{45} = x_{24} + x_{34} + x_{54} \\ x_{51} + x_{52} + x_{54} + x_{56} = x_{15} + x_{25} + x_{45} + x_{65} \\ x_{61} + x_{62} + x_{65} = x_{16} + x_{26} + x_{56} \\ x_{21} + x_{51} + x_{61} = \frac{1}{6} \\ x_{12} + x_{32} + x_{42} + x_{52} + x_{62} = \frac{1}{6} \\ x_{23} + x_{43} = \frac{1}{6} \\ x_{24} + x_{34} + x_{54} = \frac{1}{6} \\ x_{15} + x_{25} + x_{45} + x_{65} = \frac{1}{6} \\ x_{16} + x_{26} + x_{56} = \frac{1}{6} \\ x_{ij} \geq 0, \quad i, j \in V(G) \end{array} \right\}.$$

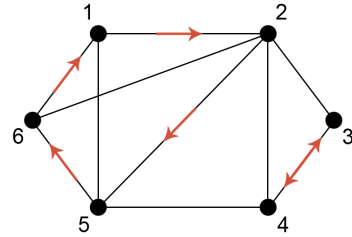
Fig. 4.2 en 4.3 geven twee mogelijke oplossingen van het bovenstaande LP-probleem. De eerste met



Figuur 4.1: De graaf G uit het voorbeeld



Figuur 4.2: Mogelijke oplossing (Hamiltonkring)



Figuur 4.3: Mogelijke oplossing (geen Hamiltonkring)

$x_{12} = x_{23} = x_{34} = x_{45} = x_{56} = x_{61} = \frac{1}{6}$ en de tweede met $x_{12} = x_{25} = x_{56} = x_{61} = x_{34} = x_{43} = \frac{1}{6}$. Een derde mogelijke oplossing van het probleem is een convexe combinatie van deze twee oplossingen, bijvoorbeeld: $x_{12} = x_{34} = x_{56} = x_{61} = \frac{1}{6}$, $x_{43} = x_{25} = \frac{1}{9}$ en $x_{23} = x_{45} = \frac{1}{18}$ ($\frac{1}{3}$ keer de oplossing van Fig. 4.2 en $\frac{2}{3}$ keer de oplossing van 4.3).

Merk op dat de functie die we minimaliseren slechts een formaliteit is, gezien deze al vastgelegd wordt in de beperkingen.

4.2 Convexe combinaties van oplossingen

In deze sectie zullen we bewijzen dat elke oplossing van het LP-probleem uit de vorige sectie een convexe combinatie is van eenduidige oplossingen, daartoe zullen we eerst de volgende stelling bekijken, met

het bewijs uit [5].

Stelling 4.2.1 (KÖNIG-EGERVÁRY) *Laat G een bipartiete graaf zijn. De maximale cardinaliteit van een matching in G is gelijk aan de minimale cardinaliteit van een vertex cover van G .*

Bewijs Laat $A, B \subset V(G)$ de twee partities zijn van $V(G)$ en M een matching van maximale cardinaliteit. Kies voor elke $e \in M$ één van zijn uiteinden, zijn uiteinde in B als er een M -alternerend pad is dat in deze knoop eindigt en anders zijn uiteinde in A , deze verzameling noemen we U .

Nu zullen we bewijzen dat de verzameling U van deze $|M|$ knopen een vertex cover is van G (gezien elke vertex cover van G ook M moet overdekken kan er geen vertex cover zijn met minder dan $|M|$ knopen). Neem een willekeurige tak $ab \in E(G)$, we gaan laten zien dat a of b in U ligt.

Als $ab \in M$ (met $a \in A$ en $b \in B$) dan geldt dat sowieso, dus neem aan dat $ab \notin M$. Omdat M maximaal is er een $a'b' \in M$ met $a = a'$ of $b = b'$. Sterker nog, we kunnen aannemen dat $a = a'$. Immers, als $b = b'$ dan is ab een alternerend pad dat in $b = b'$ begint en hadden we $b' = b$ dus gekozen voor U als uiteinde van $a'b'$. Als nu $a = a' \notin U$, dan $b' \in U$ en is er dus een alternerend pad $P = v_1v_2\dots b'$ dat eindigt in b' . Echter dan is er ook een alternerend pad P' dat eindigt in b . Namelijk $v_1\dots b$ (als $b \in P$) of $v_1\dots b'a'b$ (als $b \notin P$). In beide gevallen moet dit betekenen dat b gematcht is. Anders is P' een alternerend pad waarbij zowel de eerste als laatste tak ongematcht is en is $M' = M \cup (E(P') \setminus M) \setminus (M \cap E(P'))$ een matching van G met $|M'| = |M| + 1$. Dit zou in tegenspraak zijn met de aanname dat M maximaal is.

Dus b is gematcht en P' is een alternerend pad eindigend in b . Dit betekent dat $b \in U$. We concluderen dat elke tak van G een uiteinde heeft in U , dus de minimale cardinaliteit van een vertex cover is gelijk aan $|U| = |M|$. □

Met behulp van deze stelling kunnen we nu de volgende stelling over dubbel stochastische matrices bewijzen.

Stelling 4.2.2 (BIRKHOFF-VON NEUMANN) *Een dubbel stochastische matrix is een convexe combinatie van permutatie matrices.*

Bewijs Laat P een willekeurige dubbel stochastische $n \times n$ matrix zijn. Eerst zullen we voor P bewijzen dat deze n onafhankelijke niet-nul elementen heeft. Dit wil zeggen; er zijn verschillende $i_1, i_2, \dots, i_n \in \{1, 2, \dots, n\}$ en verschillende $j_1, j_2, \dots, j_n \in \{1, 2, \dots, n\}$ zodanig dat $P_{i_1j_1}, P_{i_2j_2}, \dots, P_{i_nj_n} > 0$.

Nu construeren we de bipartiete graaf $G = (V, E)$ van orde $2n$ met partities $R = \{r_1, r_2, \dots, r_n\}$ en $K = \{k_1, k_2, \dots, k_n\}$ en takkenverzameling $E = \{r_i k_j : P_{r_i k_j} > 0\}$ ofwel, elke knoop uit R staat voor een rij van P , elke knoop uit K staat voor een kolom uit P , elke tak staat voor een niet-nul element van P en een matching correspondeert met een verzameling onafhankelijke niet-nul elementen. Verder definiëren de functie $f : E \rightarrow \mathbb{R}_{>0}$ door $f(ij) = P_{ij}$. Dus voor een vaste $i \in V$ geldt $\sum_{j:ij \in E} f(ij) = 1$, omdat P dubbel stochastisch is.

We willen nu bewijzen dat de minimale cardinaliteit x van een vertex cover van G gelijk is aan n , want in dat geval is er een matching van G van grootte n (Stelling 4.2.1) en bevat P dus n onafhankelijke niet-nul elementen. Laat $U = \{u_1, u_2, \dots, u_x\}$ een vertex cover van G zijn met cardinaliteit x . Omdat elke tak uit E incident is met een punt uit U geldt:

$$n = \sum_{ij \in E} f(ij) \leq \sum_{i=1}^x \left[\sum_{u_i j \in E} f(u_i j) \right] = x.$$

Bovendien weten we, omdat $|R| = n$, dat $n \geq x$. We concluderen dat $x = n$.

Dus P heeft n onafhankelijke elementen $P_{i_1j_1}, \dots, P_{i_nj_n} > 0$. Laat $\lambda_1 = \min\{P_{i_1j_1}, \dots, P_{i_nj_n}\}$. Dan kunnen we P als volgt schrijven:

$$P = \lambda_1 Q^{(1)} + (P - \lambda_1 Q^{(1)}).$$

Hierbij is $Q^{(1)}$ de permutatie matrix met $Q_{i_1 j_1}^{(1)}, \dots, Q_{i_n j_n}^{(1)} = 1$. Verder geldt, omdat de som van elke rij en van elke kolom van $(P - \lambda_1 Q^{(1)})$ precies gelijk is aan $1 - \lambda_1$, dat $\frac{1}{1 - \lambda_1}(P - \lambda_1 Q^{(1)})$ weer een dubbel stochastische matrix is. Dus we kunnen dit proces herhalen tot we de volgende uitdrukking voor P hebben:

$$P = \lambda_1 Q^{(1)} + \lambda_2 Q^{(2)} + \dots + \lambda_m Q^{(m)}.$$

Merk hierbij op dat $m \leq n^2$ omdat in elke stap tenminste één niet-nul element 0 wordt en dat $\lambda_1 + \lambda_2 + \dots + \lambda_m = 1$ omdat P (dubbel) stochastisch is. Dit bewijst de stelling. □

Met deze stelling kunnen we nu verder kijken naar de oplossing die we krijgen voor het bovenstaande LP-probleem. Daarvoor definiëren we de matrix X als volgt:

$$X_{ij} = \begin{cases} 0 & \text{als } ij \notin E(G) \\ nx_{ij} & \text{anders.} \end{cases}.$$

Omdat $\sum_i x_{ij} = \sum_j x_{ij} = \frac{1}{n}$ betekent dit dat X een dubbel stochastische matrix is. We kunnen X dus schrijven als een convexe combinatie van permutatie matrices die alle een oplossing zijn voor het LP-probleem.

Voorbeeld We nemen weer de graaf G uit het voorbeeld van sectie 4.1 en passen daar het algoritme uit Appendix A op toe. Voor X krijgen we dan de volgende matrix:

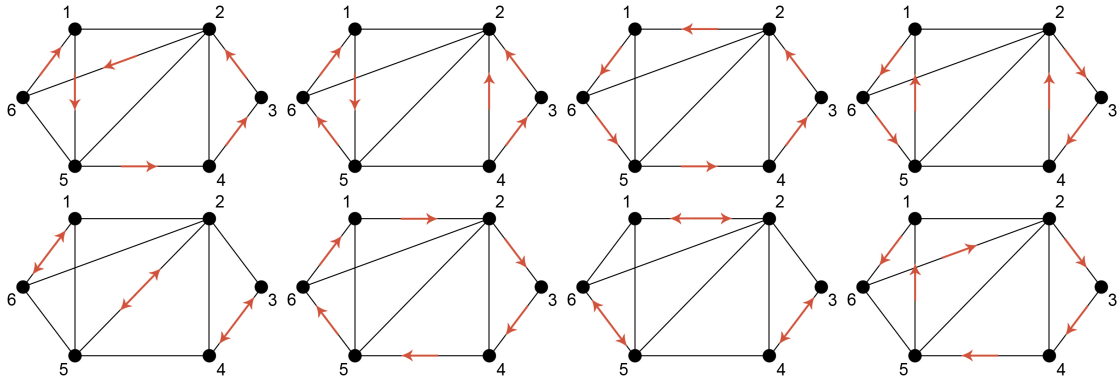
$$X = \begin{pmatrix} 0 & 0.1978 & 0 & 0 & 0.3239 & 0.4784 \\ 0.1973 & 0 & 0.3599 & 0.1322 & 0.1189 & 0.1917 \\ 0 & 0.3599 & 0 & 0.6401 & 0 & 0 \\ 0 & 0.1322 & 0.6401 & 0 & 0.2277 & 0 \\ 0.3235 & 0.1189 & 0 & 0.2277 & 0 & 0.3299 \\ 0.4792 & 0.1912 & 0 & 0 & 0.3296 & 0 \end{pmatrix}.$$

Verder ontbindt het algoritme deze matrix als volgt in een convexe combinatie van permutatie matrices:

$$\begin{aligned} X &= 0.1917 \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} + 0.1321 \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} + 0.0360 \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \\ &+ 0.1322 \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} + 0.1189 \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} + 0.0365 \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \\ &+ 0.1613 \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} + 0.1912 \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}. \end{aligned}$$

Ofwel we hebben nu 8 oplossingen gevonden, die zijn weergegeven in Fig. 4.4. Vier van deze oplossingen corresponderen met een Hamiltonkring. In het totaal zijn er twee verschillende Hamiltonkringen, welke dus beide gevonden zijn.

Dit voorbeeld laat dus zien dat we met het algoritme van Appendix A een oplossing van het HCP kunnen vinden. Echter, als we geen Hamilton cykel vinden betekent dit zeker niet dat zo'n cykel niet bestaat. Dit is natuurlijk een nadeel. Het voordeel is dat het een polynomiale tijd algoritme is, omdat het een maximale matching in een bipartiete graaf in polynomiale tijd kan worden bepaald.



Figuur 4.4: Gevonden antwoorden voor het LP-probleem

4.3 Experimentele methoden

Het blijkt dat de LP oplosser van MATLAB een random oplossing voor een LP-probleem genereert. Dit betekent dat als we herhaaldelijk het algoritme uit Appendix A toepassen op een graaf, we waarschijnlijk steeds een nieuwe oplossing krijgen en ook een nieuwe convexe combinatie van eenduidige oplossingen. Deze ontdekking heeft tot het idee geleid om te onderzoeken hoe vaak we het algoritme moeten doorlopen voordat we met een bepaalde zekerheid kunnen zeggen of een graaf een Hamiltonkring bevat. Dit zou het onderwerp kunnen zijn voor verder onderzoek, maar we hebben hier al een klein begin mee gemaakt. Het algoritme uit Appendix C is dit begin. Hierin maken we een graaf van orde 60, met één Hamiltonkring, vervolgens creëren we met kans p een tak tussen elk tweetal knopen. Op deze manier krijgen we een random graaf (die neigt naar een reguliere graaf). Voor deze graaf tellen we hoe vaak we het algoritme uit Appendix A moeten laten lopen voor deze een Hamiltonkring vindt. In tabel 4.1 staan de resultaten van dit algoritme, we hebben voor p de waarden $0, \frac{1}{10}, \frac{2}{10}, \frac{3}{10}, \frac{4}{10}, \frac{5}{10}, \frac{6}{10}, \frac{7}{10}, \frac{8}{10}$ en $\frac{9}{10}$ genomen en voor elk van deze kansen hebben we de test 10 keer uitgevoerd. We zien dat we het algoritme vaker moeten uitvoeren bij lagere kansen. Bovendien zien we dat alle

	p=0.0	p=0.1	p=0.2	p=0.3	p=0.4	p=0.5	p=0.6	p=0.7	p=0.8	p=0.9
test 1	1	1220	632	79	30	30	71	2	135	1
test 2	7	132	180	15	24	37	19	39	14	35
test 3	1	461	482	21	65	28	62	20	28	25
test 4	1	3164	229	51	107	13	2	21	17	23
test 5	1	2	60	38	162	21	513	3	2	70
test 6	5	611	52	44	40	13	47	23	8	10
test 7	1	632	329	104	29	19	25	19	2	13
test 8	2	202	35	15	103	21	49	30	3	7
test 9	1	1983	185	110	63	11	3	2	9	15
test 10	4	1235	176	179	14	43	18	103	14	32
Gem.	2.4	964.2	236.0	64.6	63.7	23.6	80.9	26.2	23.2	23.1

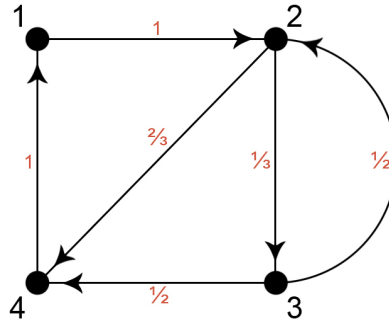
Tabel 4.1: Resultaten van het algoritme uit Appendix C

waarden beduidend kleiner zijn dan $60^2 = 3600$, dit geeft de hoop dat we voor een graaf G na $|G|^2$ keer het (polynomiale) algoritme uit Appendix A te doorlopen, we met behoorlijke zekerheid kunnen zeggen of G een Hamiltongraaf is of niet. Dit is voor verder onderzoek.

4.4 Verwachte aantal bezoeken in Markov beslissingsketens

Het probleem waar we in de vorige sectie tegen aan liepen is, dat we met het gevonden algoritme, naast Hamilton cykels, ook verzamelingen van disjunctie cykels konden vinden. In deze sectie zullen we dat probleem proberen te omzeilen. Dit doen we door te kijken naar het aantal bezoeken van een Markovketen aan een toestand voor absorbtie in een vaste, gegeven toestand (meestal 1). We bekijken dus een Markov keten X_1, X_2, \dots met toestandsruimte $S = \{1, 2, \dots, n\}$ en overgangskansen p_{ij} (voor $i, j \in S$). Verder noteren we het verwachte aantal bezoeken aan j als we in i beginnen, voordat we in het punt 1 komen, met b_{ij} . Dus:

$$b_{ij} = \mathbb{E} \left[\sum_{t=1}^{\infty} 1_{\{X_t=j; X_1, X_2, \dots, X_{t-1} \neq 1\}} \mid X_1 = i \right].$$



Figuur 4.5: Voorbeeld Markov keten

Voorbeeld We bekijken de Markov keten uit Fig. 4.5, met toestandsruimte $S = \{1, 2, 3, 4\}$ en overgangskansen:

$$\begin{aligned} p_{11} = 0, \quad p_{12} = 1, \quad p_{13} = 0, \quad p_{14} = 0, \quad p_{21} = 0, \quad p_{22} = 0, \quad p_{23} = \frac{1}{3}, \quad p_{24} = \frac{2}{3}, \\ p_{31} = 0, \quad p_{32} = \frac{1}{2}, \quad p_{33} = 0, \quad p_{34} = \frac{1}{2}, \quad p_{41} = 1, \quad p_{42} = 0, \quad p_{43} = 0, \quad p_{44} = 0. \end{aligned}$$

Nu kunnen we voor alle $i, j \in S$, b_{ij} uitrekenen.

$$\left(\begin{array}{l} b_{11} = 1 = 1 \\ b_{12} = 1 + \frac{1}{3} \frac{1}{2} + \frac{1}{3} \frac{1}{3} \frac{1}{2} + \dots = \frac{6}{5} \\ b_{13} = \frac{1}{3} + \frac{1}{3} \frac{1}{2} \frac{1}{3} + \frac{1}{3} \frac{1}{2} \frac{1}{3} \frac{1}{2} \frac{1}{3} + \dots = \frac{6}{5} \\ b_{14} = 1 = 1 \end{array} \right. \quad \left. \begin{array}{l} b_{21} = 0 = 0 \\ b_{22} = 1 + \frac{1}{3} \frac{1}{2} + \frac{1}{3} \frac{1}{3} \frac{1}{2} + \dots = \frac{6}{5} \\ b_{23} = \frac{1}{3} + \frac{1}{3} \frac{1}{2} \frac{1}{3} + \frac{1}{3} \frac{1}{2} \frac{1}{3} \frac{1}{2} \frac{1}{3} + \dots = \frac{6}{5} \\ b_{24} = 1 = 1 \end{array} \right) .$$

$$\left(\begin{array}{l} b_{31} = 0 = 0 \\ b_{32} = \frac{1}{2} + \frac{1}{2} \frac{1}{3} \frac{1}{2} + \frac{1}{2} \frac{1}{3} \frac{1}{2} \frac{1}{3} \frac{1}{2} + \dots = \frac{3}{5} \\ b_{33} = 1 + \frac{1}{2} \frac{1}{3} + \frac{1}{2} \frac{1}{3} \frac{1}{2} \frac{1}{3} + \dots = \frac{6}{5} \\ b_{34} = 1 = 1 \end{array} \right. \quad \left. \begin{array}{l} b_{41} = 0 = 0 \\ b_{42} = 0 = 0 \\ b_{43} = 0 = 0 \\ b_{44} = 1 = 1 \end{array} \right) .$$

Als we nu de zogenaamde taboo-overgangskansen p'_{ij} ($i, j \in S$) als volgt definiëren

$$p'_{ij} = \begin{cases} p_{ij}, & \text{als } j \neq 1 \\ 0, & \text{als } j = 1 \end{cases} ,$$

dan kunnen we b_{ij} definiëren aan de hand van de burens van i . We krijgen dan het volgende stelsel lineaire vergelijkingen voor b_{ij} :

$$b_{ij} = \begin{cases} \delta_{ij} + \sum_{k \in S} p'_{ik} b_{kj}, & \text{als } j \neq 1 \\ \delta_{ij}, & \text{als } j = 1 \end{cases} .$$

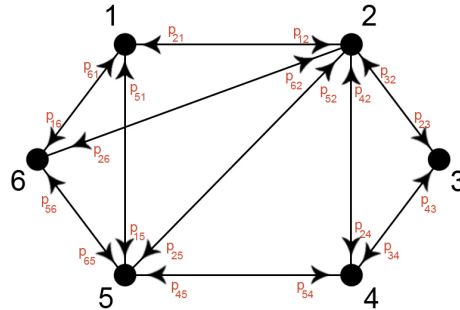
Dit geeft een systeem van n^2 lineaire vergelijkingen met n^2 onbekenden. Omdat de matrix P' gegeven door $P'_{ij} = p'_{ij}$ (voor $i, j \in S$) substochastische is, is er precies één oplossing voor dit systeem.

Voorbeeld Als we weer de Markov keten uit het vorige voorbeeld nemen, krijgen we het volgende stelsel vergelijkingen:

$$\left\{ \begin{array}{llll} b_{11} = 1 & = 1 & b_{21} = 0 & = 0 \\ b_{12} = b_{22} & = \frac{6}{5} & b_{22} = 1 + \frac{1}{3}b_{32} + \frac{2}{3}b_{42} & = \frac{6}{5} \\ b_{13} = b_{23} & = \frac{6}{5} & b_{23} = \frac{1}{3}b_{33} + \frac{2}{3}b_{43} & = \frac{6}{5} \\ b_{14} = b_{24} & = 1 & b_{24} = \frac{1}{3}b_{34} + \frac{2}{3}b_{44} & = 1 \\ \\ b_{31} = 0 & = 0 & b_{41} = 0 & = 0 \\ b_{32} = \frac{1}{2}b_{22} + \frac{1}{2}b_{42} & = \frac{3}{5} & b_{42} = 0 & = 0 \\ b_{33} = 1 + \frac{1}{2}b_{23} + \frac{1}{2}b_{43} & = \frac{6}{5} & b_{43} = 0 & = 0 \\ b_{34} = \frac{1}{2}b_{24} + \frac{1}{2}b_{44} & = 1 & b_{44} = 1 & = 1 \end{array} \right\}.$$

Om deze theorie toe te passen op het Hamiltongraaf probleem, maken we van een graaf $G = (V, E)$ een Markov keten met $p_{ij} = 0$ als $ij \notin E$. De overige overgangskansen zijn nog onbekend. Wel kunnen we extra beperkingen toevoegen, namelijk dat $b_{1i} = 1$ voor alle $i \in S$ en $p_{ij} \in \{0, 1\}$ voor alle $ij \in E$. Op deze manier forceren we een Markovketen die met kans 1 een Hamiltonkring volgt, mits deze bestaat. We beginnen namelijk in punt 1, gaan dan met kans 1 naar een volgend punt, die we daarna (met kans 1) niet meer bezoeken. Gezien we elk punt moeten bezoeken leidt dit tot een cykel van lengte $|V(G)|$, een Hamiltoncykel.

Voorbeeld Als voorbeeld nemen we weer de graaf G uit de vorige sectie en maken daar de Markov keten van, zie Fig. 4.6. Merk hierbij op dat de overgangskansen p_{ij} (voor $ij \in E(G)$) nog onbekend



Figuur 4.6: Markov keten bij graaf G

zijn. We krijgen dan het volgende stelsel vergelijkingen met $|V(G)|^2 + 2|E(G)| = 46$ variabelen:

$$\left\{ \begin{array}{ll} b_{11} = 1, & b_{41} = 0 \\ b_{12} = p_{12}b_{22} + p_{15}b_{52} + p_{16}b_{62}, & b_{42} = p_{42}b_{22} + p_{43}b_{32} + p_{45}b_{52} \\ b_{13} = p_{12}b_{23} + p_{15}b_{53} + p_{16}b_{63}, & b_{43} = p_{42}b_{23} + p_{43}b_{33} + p_{45}b_{53} \\ b_{14} = p_{12}b_{24} + p_{15}b_{54} + p_{16}b_{64}, & b_{44} = 1 + p_{42}b_{24} + p_{43}b_{34} + p_{45}b_{54} \\ b_{15} = p_{12}b_{25} + p_{15}b_{55} + p_{16}b_{65}, & b_{45} = p_{42}b_{25} + p_{43}b_{35} + p_{45}b_{55} \\ b_{16} = p_{12}b_{26} + p_{15}b_{56} + p_{16}b_{66}, & b_{46} = p_{42}b_{26} + p_{43}b_{36} + p_{45}b_{56} \\ \\ b_{21} = 0, & b_{51} = 0 \\ b_{22} = 1 + p_{21}b_{12} + p_{23}b_{32} + p_{24}b_{42} + p_{25}b_{52} + p_{26}b_{62}, & b_{52} = p_{51}b_{12} + p_{52}b_{22} + p_{54}b_{42} + p_{56}b_{62} \\ b_{23} = p_{21}b_{13} + p_{23}b_{33} + p_{24}b_{43} + p_{25}b_{53} + p_{26}b_{63}, & b_{53} = p_{51}b_{13} + p_{52}b_{23} + p_{54}b_{43} + p_{56}b_{63} \\ b_{24} = p_{21}b_{14} + p_{23}b_{34} + p_{24}b_{44} + p_{25}b_{54} + p_{26}b_{64}, & b_{54} = p_{51}b_{14} + p_{52}b_{24} + p_{54}b_{44} + p_{56}b_{64} \\ b_{25} = p_{21}b_{15} + p_{23}b_{35} + p_{24}b_{45} + p_{25}b_{55} + p_{26}b_{65}, & b_{55} = 1 + p_{51}b_{15} + p_{52}b_{25} + p_{54}b_{45} + p_{56}b_{65} \\ b_{26} = p_{21}b_{16} + p_{23}b_{36} + p_{24}b_{46} + p_{25}b_{56} + p_{26}b_{66}, & b_{56} = p_{51}b_{16} + p_{52}b_{26} + p_{54}b_{46} + p_{56}b_{66} \\ \\ b_{31} = 0, & b_{61} = 0 \\ b_{32} = p_{32}b_{22} + p_{34}b_{42}, & b_{62} = p_{61}b_{12} + p_{62}b_{22} + p_{65}b_{52} \\ b_{33} = 1 + p_{32}b_{23} + p_{34}b_{43}, & b_{63} = p_{61}b_{13} + p_{62}b_{23} + p_{65}b_{53} \\ b_{34} = p_{32}b_{24} + p_{34}b_{44}, & b_{64} = p_{61}b_{14} + p_{62}b_{24} + p_{65}b_{54} \\ b_{35} = p_{32}b_{25} + p_{34}b_{45}, & b_{65} = p_{61}b_{15} + p_{62}b_{25} + p_{65}b_{55} \\ b_{36} = p_{32}b_{26} + p_{34}b_{46}, & b_{66} = 1 + p_{61}b_{16} + p_{62}b_{26} + p_{65}b_{56} \\ \\ 1 = p_{12} + p_{15} + p_{16}, & 1 = p_{42} + p_{43} + p_{45} \\ 1 = p_{21} + p_{23} + p_{24} + p_{25} + p_{26}, & 1 = p_{51} + p_{52} + p_{54} + p_{56} \\ 1 = p_{32} + p_{34}, & 1 = p_{61} + p_{62} + p_{65} \\ \\ p_{ij} \in \{0, 1\} \quad (ij \in E(G)), \quad b_{ij} \geq 0 \quad (i, j \in V(G)) \end{array} \right. .$$

Stel nu dat de oplossing van dit stelsel geeft dat $p_{i_1j_1}, p_{i_2j_2}, p_{i_3j_3}, p_{i_4j_4}, p_{i_5j_5}$ en $p_{i_6j_6}$ de zes overgangskansen zijn die 1 zijn, dan vormen de pijlen $i_1j_1, i_2j_2, i_3j_3, i_4j_4, i_5j_5$ en i_6j_6 een Hamiltoncykel in de graaf G .

Een groot nadeel van deze methode is dat in eerste instantie zowel de overgangskansen p_{ij} en de verwachtingen b_{ij} onbekend zijn. Dit heeft tot gevolg dat we een stelsel van niet-lineaire vergelijkingen krijgen, dat dus niet in polynomiale tijd oplosbaar is. De conclusie van deze sectie is dat het niet waarschijnlijk is dat deze aanpak, met verwachte bezoeken, een deterministische oplossing gaat geven voor het Hamiltongraaf probleem.

Referenties

- [1] V. Borkar, V. Ejoy, J. Filar en G. Nguyen (in preparation) *Hamiltonian Cycle Problem & Markov Chains*.
- [2] M. Meringer, *GENREG*: <http://www.mathe2.uni-bayreuth.de/markus/reggraphs.html>.
- [3] L.C.M. Kallenberg, *Besliskunde 3*
- [4] L. Peshkin, *Maximum cardinality matching*: <http://www.mathworks.com/matlabcentral/fileexchange/23159-maximum-cardinality-matching>.
- [5] R. Diestel (2010), *Graph Theory*, Springer Heidelberg Dordrecht London New York

Appendices

Appendix A

Dit matlab algoritme leest de incidentie matrix van een graaf in en geeft een oplossing X van het LP-probleem beschreven in secties 4.1 en 4.2 en ontbindt X in een convexe combinatie van permutatie matrices.

```
clear
clc

A=[0,1,0,0,1,1; %incidentiematrix van G
   1,0,1,1,1,1;
   0,1,0,1,0,0;
   0,1,1,0,1,0;
   1,1,0,1,0,1;
   1,1,0,0,1,0];

n=size(A);
n=n(1);

V=zeros(n+1,n.^2); %Matrix voor de gelijkheden
U=zeros(n.^2+n,n.^2); %Matrix voor de ongelijkheden

for i=1:n
    for j=1:n
        if A(i,j)==1
            U((i-1)*n+j,(i-1)*n+j)=-1; % x(j,j)>= 0
            U(n.^2+i,(i-1)*n+j)=-1; % sum(j)[x(i,j)] >= 1/n
            V(i,(i-1)*n+j)=1; % sum(j)[x(i,j)]-sum(j)[x(j,i)] = 0
            V(i,(j-1)*n+i)=-1; % sum(j)[x(i,j)]-sum(j)[x(j,i)] = 0
            V(n+1,(i-1)*n+j)=1; % sum(i,j)[x(i,j)] = 1
        end
    end
end

f=zeros(1,n.^2); %doelfunctie

for i=2:n
    if A(1,i)==1
        f(1,(i-1)*n+1)=1; %minimalizeer: sum(i)[x(i,1)]
    end
end
```

```

    end
end

u=zeros(1,n.^2+n); % x(j,j)>= 0

for i=1:n
    u(1,n.^2+i)=-1/n; % sum(j)[x(i,j)] >= 1/n
end

v=zeros(1,n+1); % sum(j)[x(i,j)]-sum(j)[x(j,i)] = 0
v(1,n+1)=1;% sum(i,j)[x(i,j)] = 1

x=linprog(f,U,u,V,v);

X=zeros(n,n);
for i=1:n
    for j=1:n
        X(i,j)=n*x((i-1)*n+j,1); %X(i,j) = kans als in i, om j te kiezen
    end
end
X0=X;

CMATE=0;
p=zeros(1,1); %vector van lambda waarden uit BirkhoffVonNeumann
P=zeros(1,n); % matrix met elke rij een perm matrix
k=1;
%We stoppen de loop als X0=0 of als de waarden van X0 dusdanig klein zijn
%geworden dat we door numerieke onzekerheid geen perm matrix meer hebben.
while (sum(sum(X0))>0) && (CMATE==0)
    Y=zeros(2*n,2*n);
    %Hier construeren we de bipartiete graaf uit BirkhoffVonNeumann bewijs
    for i=1:n
        for j=1:n
            if (X0(i,j)>0)
                Y(i,n+j)=1;
                Y(n+j,i)=1;
            end
        end
    end
    end
    %cardmatch is een functie welke een max matching vindt
    [MATE]=card_match(Y);
    %Als er geen maximale matching meer gevonden wordt wordt CMATE 1
    for i=1:2*n
        if (MATE(i)==0)
            CMATE=1;
        end
    end
end

if (CMATE==0)
    %Q is de perm matrix uit BirkhoffVonNeumann
    Q=zeros(n,n);
    for i=1:n

```

```

        Q(i,MATE(i)-n)=1;
    end
    %y0 is de lambda uit BVN, (I1,I2) de coördinaten van y0 uit X0
    [y0,I2]=min(sum(Q.*X0));
    I1=MATE(n+I2);
    X0=X0-(y0*Q);
    X0(I1,I2)=0;
    p(k,1)=y0;
    for i=1:n
        P(k,i)=MATE(i)-n;
    end
end
k=k+1;
end

%hier bepalen we welke van de permutatiematrices een hamiltongraaf
%weergeeft
HamiltonKringen=zeros(1,n);
sp=size(p);
sp=sp(1);
k=1;
for i=1:sp
    a=zeros(n,1);
    b=1;
    for j=1:n
        a(b)=1;
        b=P(i,b);
    end
    if (sum(a)==n)
        HamiltonKringen(k,:)=P(i,:);
        k=k+1;
    end
end
end
%als het algoritme geen Hamilton kringen gevonden heeft is Hamiltonkringen
%een 0 vector.
p %vector van lambda's uit het bewijs van BVN
P %matrix met elke rij een perm matrix Q uit het bewijs van BVN
HamiltonKringen %de hamiltonkringen van P

```

Appendix B

Dit matlab algoritme bepaalt een maximale matching van een bipartiete graaf (volgens [4]). We gebruiken dit in het algoritme van Appendix A.

```

function [MATE] = card_match(adj2);
% [mate] = card_match(adj) constructs a (Non-weighted) maximum cardinality
% matching
% on a graph represented by ADJ-acency matrix with edge IDs as
% elements

```

```

% OUTPUT: mate(i) = j means edge (i,j)=(j,i) belongs to the matching.
%
% REMARKS:
%     a vertex _v_ is called "outer" when there is an alternating path
% from _v_ to
% an unmatched vertex _u_ that starts with a matched edge.
%
% MATLAB implementation of H. Gabow's labelling scheme explained in JACM 23,
% pp221-34
% by Dr. Leonid Peshkin MIT AI Lab Dec 2003 [inspired by Ed Rothberg C code
% Jun 1985]
% http://www.ai.mit.edu/~pesha
%
%     1 2 3                                4-edge(1,2)
% 1   0 4 5   sample ADJ matrix for the following graph: (2)---(1)---(3)
% 2   4 0 0                                5-edge(1,3)
% 3   5 0 0
% we assume no isolated vertices and un-directed graph (symmetric ADJ)
global MATE LABEL OUTER FIRST Nnds qcount adj break_ties
break_ties = 1;      % need this to debug and have deterministic outcome
adj = adj2;
% step E0 {Initialize} (read the graph)
Nedgs = length(find(adj))/2; % number of edges      V
Nnds = length(adj);      % Number of vertices      U (Rothberg) or V (Gabow)
MATE = zeros(1, Nnds);   %                          and unmatched;
OUTER = zeros(1, Nnds);
FIRST = zeros(1, Nnds); % in a given search, FIRST(v) is the first
qcount = 0; qpctr=0;    % non-outer vertex in P(v) for some outer v

MATE = greedy_match(adj, Nnds); % start off with a greedy matching
% for u = find(mate == 0) % cycle through unmatched nodes
% OUTER = union(OUTER, u) % add to the list of nodes
% FIRST(u) = 0;
%end
% MATE = [0     3     2     6     0     4];
% for i = 1:Nnds, AdjLst{i} = find(adj(i,:)); end % is NO faster then
% find(adj())
neg_ones = -1 * ones(1, Nnds); % speeds up comput a bit
for u = 1:Nnds % step E1 [find unmatched vertex] - outer cycle through
    if (MATE(u) ~= 0) % unmatched vertices
        continue
    end
    LABEL = neg_ones; % all vertices are nonouter
    qcount = qcount+1;
    OUTER(qcount) = u;
    LABEL(u) = 0;
    % step E2 [choose an edge]
    while (qcount ~= qpctr) % while queue is not empty
        qpctr = qpctr+1; x = OUTER(qpctr);
        for y = find(adj(:,x))';
            % y is adjacent to an outer vertex */
            if ((MATE(y) == 0) & (y ~= u)) % step E3 [augment the

```

```

        MATE(y) = x;           % matching] found an augmenting path
        REMATCH(x,y);
        qptr = qcount;
        break; % go to E7
elseif (LABEL(y) >= 0)%step E4[assign edge labels]created blos.
    DOLABEL(x, y, adj(x,y)); % y is outer, we call L
else % step E5 [assign a vertex label] extended
    v = MATE(y); % the search path
    if (LABEL(v) < 0) % if _v_ is nonouter
        LABEL(v) = x;
        FIRST(v) = y;
        qcount = qcount+1; OUTER(qcount) = v;
    end
end
end
end
end
    % step E7 [stop the search]
    qcount = 0;
    qptr = 0;
end

% greedy matching. If a random vertex is unmatched, check all adjacent
% vertices
% to see if any of them are also unmatched. If so, match with a random
% available.
%
function [mate] = greedy_match(adj, Nnds);
global break_ties
mate = zeros(1, Nnds);
if break_ties
    ind = randperm(Nnds);
else
    ind = 1:Nnds;
end
for i = ind
    adj_list = find(adj(:,i)); % very expensive operation ....
    list_size = length(adj_list);
    if list_size > 0
        j = adj_list(ceil(rand*list_size));
        mate(j) = i;
        mate(i) = j;
        adj(i, :) = 0; adj(:, i) = 0; % erase edges from this pair
        adj(j, :) = 0; adj(:, j) = 0; % erase edges from this pair
    end
end
end

% x and y are adjacent and both are outer. Create a blossom. (Rothberg)
%
% assigns the edge label n(xy) to nonouter vertices. Edge xy connects outer
% vertices x and y. DOLABEL sets _join_ to the first nonouter vertex in
% both
% P(x) and P(y), then it labels all nonouter vertices preceding _join_ in

```

```

% P(x) or P(y).
function DOLABEL (x, y, edge)
global MATE LABEL OUTER FIRST qcount
r = FIRST(x);      % step L0
s = FIRST(y);
if ((r*s == 0) | (r == s))      % no vertices can be labeled
    return
end
flag = -edge;
LABEL(r) = flag;
LABEL(s) = flag;
if (s ~= 0)      % step L1 [switch paths]
    temp = r; r = s; s = temp; % swap r <-> s
end
    % step L2 [next nonouter vertex]
r = FIRST(LABEL(MATE(r)));      % r is set to the next nonouter vertex in
if (r ==0) return, end      % P(x) and P(y)
while (LABEL(r) ~= flag)
    LABEL(r) = flag;
    if (s ~= 0)
        temp = r; r = s; s = temp; % step L1: swap paths
    end
    r = FIRST(LABEL(MATE(r)));
    if (r ==0) return, end
end
join = r;
    % step L3 {label vertices in P(x), P(y)}
    % all nonouter vertices between x and _join_, or y and _join_, will be
    % assigned edge labels.
LabelSub(FIRST(x), edge, join); % calls to Gabow's "step L4"
LabelSub(FIRST(y), edge, join);
    % step L5 [update FIRST]
for ndx = 1:qcount
    i = OUTER(ndx); % for each outer vertex i
    if ((FIRST(i) >0) & (LABEL(FIRST(i)) > 0)) % if FIRST(i) is outer
        FIRST(i) = join; %join is now the first nonouter vertex in P(i)
    end
end
return      % step L6

% Make all non-outer vertices in the blossom outer (Rothberg)
% step L4: [Label v]
function LabelSub (v, edge, join)
global MATE LABEL OUTER FIRST qcount
while (v ~= join) & (v ~=0)      % & (v ~= 0) by Pesha LPM
    LABEL(v) = edge;
    FIRST(v) = join;
    qcount = qcount+1; OUTER(qcount) = v;
    v = FIRST(LABEL(MATE(v)));
end

% Augment the matching along the augmenting path defined by LABEL

```



```

% (Rothberg)
%   R(v,w) = REMATCH (Gabow notation)
% "rematches edges in the augmening path. Vertex v is outer.
% Part of path (w)*P(v) is in the augmenting path. It gets rematched by
% REMATCH(v,w)
%   Although REMATCH sets MATE(v) <- w, it does not se MATE(w)<- v. This
% is done in
% step E3 or another call to REMATCH.
function REMATCH(v, w) % This is a recursive routine
global MATE LABEL Nnds adj
t = MATE(v); % step R1:
MATE(v) = w;
if ((t == 0) | (MATE(t) ~= v))
    return; % the path is completely rematched
elseif (LABEL(v) <= Nnds) % % step R2 : rematch a path (v has a vertex
    MATE(t) = LABEL(v); % % label)
    REMATCH(LABEL(v), t);
    return;
else % step R3: rematch two paths (v has an edge label)
    [x, y] = find(triu(adj) == LABEL(v));
    REMATCH(x, y);
    REMATCH(y, x);
    return;
end
end

```

Appendix C

Dit is het algoritme uit sectie 4.3.

```

clear
clc

tabel=zeros(10,11);
for kans=0:9
    for tien=1:10

        n=60; %nxn matrix
        p=0.1*kans; %kans op tak tussen punten
        A=zeros(n,n); %constructie van matrix

        for i=1:n
            for j=(i+1):n
                x=rand(1,1);
                if (x<p)
                    A(i,j)=1;
                    A(j,i)=1;
                end
            end
        end
    end
end

```

```

for i=1:(n-1)
    A(i,i+1)=1;
    A(i+1,i)=1;
end
A(1,n)=1;
A(n,1)=1;

HamiltonKringen=0;
teller=0;

while (sum(HamiltonKringen)==0)
    n=size(A);
    n=n(1);

    V=zeros(n+1,n.^2); %Matrix voor de gelijkheden
    U=zeros(n.^2+n,n.^2); %Matrix voor de ongelijkheden

    for i=1:n
        for j=1:n
            if A(i,j)==1
                U((i-1)*n+j,(i-1)*n+j)=-1; % x(j,j) >= 0
                U(n.^2+i,(i-1)*n+j)=-1; % sum(j)[x(i,j)] >= 1/n
                V(i,(i-1)*n+j)=1; % sum(j)[x(i,j)]-sum(j)[x(j,i)] = 0
                V(i,(j-1)*n+i)=-1; % sum(j)[x(i,j)]-sum(j)[x(j,i)] = 0
                V(n+1,(i-1)*n+j)=1; % sum(i,j)[x(i,j)] = 1
            end
        end
    end

    f=zeros(1,n.^2); %doelfunctie

    for i=2:n
        if A(1,i)==1
            f(1,(i-1)*n+1)=1; %minimalizeer: sum(i)[x(i,1)]
        end
    end

    u=zeros(1,n.^2+n); % x(j,j) >= 0

    for i=1:n
        u(1,n.^2+i)=-1/n; % sum(j)[x(i,j)] >= 1/n
    end

    v=zeros(1,n+1); % sum(j)[x(i,j)]-sum(j)[x(j,i)] = 0
    v(1,n+1)=1;% sum(i,j)[x(i,j)] = 1

    x=linprog(f,U,u,V,v);

    X=zeros(n,n);
    for i=1:n
        for j=1:n

```

```

        X(i, j)=n*x((i-1)*n+j,1); %X(i, j) = kans als in i, om j te kiezen
    end
end
X0=X;

CMATE=0;
p=zeros(1,1); %dit wordt de vector van lambda waarden uit BirkhoffVonNeumann
P=zeros(1,n); %dit wordt de matrix met elke rij een perm matrix
k=1;
%We stoppen de loop als X0=0 of als de waarden van X0 dusdanig klein zijn
%geworden dat we door numerieke onzekerheid geen perm matrix meer hebben.
while (sum(sum(X0))>0) && (CMATE==0)
    Y=zeros(2*n,2*n);
    %Hier construeren we de bipartiete graaf uit BirkhoffVonNeumann bewijs
    for i=1:n
        for j=1:n
            if (X0(i, j)>0)
                Y(i, n+j)=1;
                Y(n+j, i)=1;
            end
        end
    end
    %cardmatch is een functie welke een max matching vindt
    [MATE]=card_match(Y);
    %Als er geen maximale matching meer gevonden wordt wordt CMATE 1
    for i=1:2*n
        if (MATE(i)==0)
            CMATE=1;
        end
    end
    if (CMATE==0)
        %Q is de perm matrix uit BirkhoffVonNeumann
        Q=zeros(n,n);
        for i=1:n
            Q(i, MATE(i)-n)=1;
        end
        %y0 is de lambda uit BVN, (I1, I2) de coördinaten van y0 uit X0
        [y0, I2]=min(sum(Q.*X0));
        I1=MATE(n+I2);
        X0=X0-(y0*Q);
        X0(I1, I2)=0;
        p(k,1)=y0;
        for i=1:n
            P(k, i)=MATE(i)-n;
        end
    end
    k=k+1;
end

%hier bepalen we welke van de permutatiematrices een hamiltongraaf
%weergeeft

```

```

HamiltonKringen=zeros(1,n);
sp=size(p);
sp=sp(2);
k=1;
for i=1:sp
    a=zeros(n,1);
    b=1;
    for j=1:n
        a(b)=1;
        b=P(i,b);
    end
    if (sum(a)==n)
        HamiltonKringen(k,:)=P(i,:);
        k=k+1;
    end
end
teller=teller+1
end
tabel(tien,kans+1)=teller;
tabel
end
end

```