

Aljabar Min-Max Plus dan Terapannya

Version 3.0.0

11 September 2015

Subiono



Subiono — Email: subiono2008@matematika.its.ac.id

Alamat: Jurusan Matematika
Institut Teknologi Sepuluh Nopember
Sukolilo Surabaya
Indonesia

Copyright

© 2015 The Author, Subiono.



Kata Pengantar

Alhamdulillahirabbilamin, segala puji hanyalah milikMu ya Allah yang telah memberikan "kebebasan bertanggung jawab" kepada manusia semasa hidupnya untuk suatu kebaikan dalam melaksanakan amanatNya di hamparan bumi yang dihuni beragam manusia. Sholawat dan Salam kepadaMu ya Nabi Muhammad beserta para keluarganya dan para pengikutnya sampai nanti di hari akhir.

Buku ini disusun dengan maksud untuk membantu dan mempermudah mahasiswa dalam mempelajari materi kuliah Aljabar Min-Max Plus. Selain dari pada itu juga dimaksudkan untuk menambah suatu wacana bagi para peminat lainnya dari berbagai disiplin ilmu yang membutuhkannya atau kalangan industri dan perguruan tinggi.

Dalam buku ini diberikan beberapa konsep pengertian dari materi yang disajikan setelah itu diikuti dengan beberapa contoh untuk mempermudah pemahaman, selain itu juga diberikan beberapa contoh aplikasi. Kandungan dari buku ini juga berisi beberapa kajian baru dari hasil-hasil penelitian, baik dilakukan oleh mahasiswa yang dibimbing oleh penulis ataupun oleh penulis dan tim yang tergabung dalam penelitian dalam bidang aljabar min-max plus.

Topik bahasan disajikan dengan penekanan pada "matematika" tetapi tidaklah menjadikan para pemakai lain akan mengalami kesulitan dalam mempelajari buku ini, karena peletakan penekanan aspek matematika dibuat dengan porsi yang seimbang. Sehingga para peminat matematika tetap dapat menikmati dan menggunakan ilmunya terutama dalam Aljabar Min-Max Plus, begitu juga untuk para pemakai yang lainnya diharapkan mendapat tambahan wawasan untuk melihat matematika sebagai alat yang dibutuhkan terutama dalam kajian Aljabar Min-Max Plus untuk menyelesaikan masalah-masalah praktis yang dihadapinya.

Untuk memudahkan pembaca mengikuti alur dari setiap topik bahasan dalam buku ini, diasumsikan pembaca mempunyai bekal pengetahuan "Aljabar" dan "Aljabar Linear" yang memadai sebagai pembanding dari segi Aljabar biasa dengan Aljabar Min-Max Plus.

Persiapan penulisan materi buku ini membutuhkan waktu cukup lama lama, sejak penulis mengajarkan mata kuliah "Sistem Event Diskrit" di jurusan Matematika FMIPA-ITS, Surabaya. Hampir keseluruhan materi dari mata kuliah ini adalah Aljabar Max-Plus,

Aljabar Min Plus dan gabungan keduanya yaitu Aljabar Min-Max Plus penekanannya pada apa yang dinamakan Bipartisi Sistem Min-Max Plus. Beberapa materi disusun dari pengalaman mengajar tsb. Selain itu juga dari kumpulan makalah (paper) penulis dan hasil-hasil dari pembimbingan skripsi dan tesis mahasiswa.

Penulis pada kesempatan ini menyampaikan keaktifan pembaca dalam mengkaji buku ini untuk menyampaikan kritik dan saran guna perbaikan buku ini, sehingga pada versi yang mendatang "mutu buku" yang baik bisa dicapai. Kritik dan saran ini sangat penting karena selain alasan yang telah disebutkan tadi, penulis percaya bahwa dalam sajian buku ini masih kurang dari sempurna bahkan mungkin ada suatu kesalahan dalam sajian buku ini baik dalam bentuk redaksional, pengetikan dan materi yang menyebabkan menjadi suatu bacaan kurang begitu bagus.

Buku ini dapat diperoleh secara gratis oleh siapapun tanpa harus membayar kepada penulis. Hal ini berdasarkan pemikiran penulis untuk kebebasan seseorang mendapatkan suatu bacaan yang tersedia secara bebas dengan maksud "kemanfaatan" dan "kejujuran". Yang dimaksud dengan kemanfaatan adalah bergunanya bacaan ini untuk kemudahan pembaca memperoleh informasi penting yang diperlukannya dan untuk pembelajaran. Sedangkan kejujuran adalah ikatan moral dari pembaca untuk tidak mendistribusi buku ini dengan tujuan yang tidak bermanfaat yang hanya menguntungkan dirinya sendiri.

Penulis menulis buku ini berdasarkan pemikiran "kebebasan menulis" (tidak harus menggunakan media cetak penerbit) dengan asas "kemanfaatan" menggunakan media yang tersaji masa kini. Beberapa alat bantu untuk penulisan buku ini juga didapat secara gratis, yaitu perangkat lunak $\text{\LaTeX}$ untuk Windows yaitu MikTeX 2.9 dan TeXStudio 2.9.4 sebagai salah satu media $\text{\LaTeX}$ editor. Beberapa gambar yang ada dalam buku ini menggunakan perangkat lunak $\text{\LaTeX}$ Draw 3.3.2 yang juga didapat secara gratis. Begitu juga beberapa bahan rujukan didapat secara gratis lewat internet. Selain itu untuk menyelesaikan beberapa contoh yang dibahas digunakan alat bantu perangkat lunak toolbox for maxplus algebra versi terbaru v 3.0.0. Toolbox ini penulis buat dan telah di upload di website ([26]). Tool box bisa di jalankan dengan perangkat lunak Scilab versi 5.5.2, perangkat lunak ini juga didapat dari internet secara gratis. Oleh karenanya sangatlah tidak pantas dan sungguh tidak pantas penulis mengambil keuntungan untuk menjadikan tulisan ini sebagai bahan yang diperjual belikan. Semoga ALLAH melindungi niat ini dan selalu mengingatkan penulis untuk tidak melakukan hal tersebut.

Bila buku ini digunakan sebagai suatu rujukan tuliskan sebagai berikut :

Subiono, "Aljabar Min-Max Plus dan Terapannya, ver. 3.0.0", Jurusan Matematika ITS, Surabaya, 11 September 2015. Akhirnya, dengan segala kerendahan hati penulis mohon kepada Allah semoga penulisan ini bisa berlanjut untuk versi mendatang yang tentunya lebih "baik" dari Versi 3.0.0 yang tersedia saat ini dan semoga benar-benar buku yang tersaji ini bermanfaat bagi pembaca. Selanjutnya aku serahkan kepadaMu ya ALLAH apa yang aku kerjakan ini dan aku mohon dan percaya kepadaMu ya ALLAH untuk melindungi hasil ini dari orang-orang yang mengambil hanya untuk suatu tujuan keuntungan materi sesaat, ingatkanlah orang-orang tersebut untuk tidak melakukannya.

Surabaya, 11 September 2015



Penulis

Daftar Isi

Kata Pengantar	i
1 Pendahuluan	1
1.1 Aljabar Max-Plus	2
1.2 Vektor dan Matriks	8
2 Teori Spektral	21
2.1 Matriks dan Graf	21
2.2 Nilai karakteristik dan Vektor karakteristik	32
2.2.1 Ketunggalan dari vektor waktu-sikel	35
2.2.2 Keujudan vektor waktu sikel untuk matriks tak-tereduksi	37
2.3 Beberapa Algoritma Power	38
2.4 Penyelesaian Persamaan Linear	60
2.4.1 Sub-Penyelesaian Terbesar	60
2.4.2 Analisis Penyelesaian $A \otimes \mathbf{x} = \mathbf{b}$	63
2.4.3 Persamaan $\mathbf{x} = (A \otimes \mathbf{x}) \oplus \mathbf{b}$	71
2.5 <i>Eigenmode</i>	73
2.6 Algoritma Iterasi <i>Policy</i>	86
3 Contoh Aplikasi	95
3.1 Masalah Jadwal Penerbangan Pesawat pada suatu Bandara	95
3.2 Sistem Produksi Sederhana	97
3.3 Penjadwalan Sistem Jaringan Kereta dan Kestabilan	101
3.3.1 Contoh jaringan kereta	102
3.3.2 Pengkajian model yang diharapkan	104
3.3.3 Jadwal keberangkatan	104
3.3.4 Simulasi sistem terhadap keterlambatan	106
3.4 Menentukan Jalur Tercepat	109

4	Pengenalan Petri Nets	111
4.1	Tanda Petri net dan Ruang Keadaan	113
4.2	Dinamika Petri net	115
4.3	Representasi Petri net Menggunakan Matriks	119
4.4	Analisis Model Sistem <i>Event</i> Diskrit Tak Berwaktu	123
4.4.1	<i>Liveness</i> dan <i>Deadlocks</i>	123
4.5	Model sistem antrian	125
4.5.1	Model Aljabar maxplus dari Petri net dengan waktu.	126
4.5.2	Bentuk Petrinet dan Model Dari Sistem Antrian Dengan Adanya Kemungkinan <i>Server Down</i> dan Kapasitas Antrian Takterbatas . .	128
4.5.3	Bentuk Petrinet dan Model dari Sistem Antrian dengan Adanya Kemungkinan <i>Server Down</i> dan Kapasitas Antrian Terbatas	130
4.6	Model Rantai Pasok	133
4.6.1	Penjadwalan Dengan Menggunakan Satu Tanker	140
4.6.2	Penjadwalan Dengan Menggunakan Dua Tanker	142
4.6.3	Model Kapasitas Tanker Sama dan Mempertimbangkan waktu <i>unloading</i> dan <i>loading</i> Produk	143
4.7	Penggunaan <i>Eigenmode</i> Dalam Sistem Antrian	150
5	Pengenalan Sistem-$(min, max, +)$	157
5.1	Sistem Bipartisi- $(min, max, +)$	158
5.2	Sifat Tereduksi dalam Sistem Bipartisi- $(Min, Max, +)$	160
5.3	Nilai karakteristik dan Vektor Karakteristik dalam Sistem Bipartisi- $(Min, Max, +)$	160
	Daftar Pustaka	163

Bab 1

Pendahuluan

Akhir-akhir ini baik dunia industri ataupun dunia akademik tertarik pada teknik untuk memodel, menganalisa dan mengontrol sistem-sistem yang kompleks seperti sistem manufaktur fleksibel, jaringan telekomunikasi, sistem proses parallel, sistem kontrol trafik, sistem logistik dsb. Macam-macam sistem yang telah di sebutkan tadi adalah contoh dari Sistem Event Diskrit (SED). Klas dari SED utamanya memuat sistem buatan manusia yang terdiri dari sejumlah *resources* (misalnya, mesin, kanal-kanal komunikasi, processor,.....) yang dipakai bersama oleh beberapa pengguna (misalnya, macam produk, paket informasi, job,.....) kesemuanya itu berkontribusi untuk mencapai beberapa tujuan bersama (misalnya, produk perakitan, transmisi dari sekumpulan paket informasi, komputasi parallel,.....). Suatu gambaram karakteristik dari SED adalah 'kedinamikannya' yaitu "event-driven" yang bertolak belakang dengan "time-driven". Disini perilaku suatu SED lebih ditentukan oleh event dari pada waktu. Suatu event berkaitan dengan awal atau akhir dari suatu aktifitas. Bila ditinjau suatu sistem produksi, maka event yang mungkin adalah kelengkapan mesin telah menyelesaikan suatu produk, suatu buffer telah kosong dsb. Event terjadi dengan waktu diskrit. Interval diantara event tidak harus identik, bisa deterministik atau stokhastik. Umumnya kedinamikan dari SED dikarakteristikkan oleh 'kesinkronan' dan 'konkurensi'. Sinkronisasi memerlukan ketersediaan dari beberapa resources pada saat yang bersamaan (misalnya, sebelum bisa dirakit suatu produk pada suatu mesin, mesin harus dalam keadaan sedang tidak sibuk ("idle") dan beberapa komponen lain sudah harus tersedia sebelum suatu job tertentu bisa dieksekusi, dalam sistem proses parallel, processor dan semua data input yang diperlukan sudah harus tersedia,.....). Konkurensi tampak ketika pada suatu saat seorang pengguna harus memilih beberapa resources (misalnya, dalam suatu sistem produksi suatu job mungkin dieksekusi pada beberapa mesin yang bisa menangani job tsb. dan saat tersebut mesin-mesin harus dalam keadaan idle; dalam sistem proses parallel suatu "data-driven" dari suatu job mungkin dieksekusi pada beberapa processor yang tersedia pada saat tsb. atau dengan segera processor tsb. akan tersedia.....). Pendekatan aljabar max-plus dapat menentukan dan menganalisa berbagai sifat sistem, tetapi pendekatan hanya bisa diterapkan pada sebagian klas SED yang bisa diuraikan dengan model waktu invarian max-linier. Subklas ini adalah subklas dari waktu

invarian SED deterministik dimana hanya sinkronisasi tanpa kejadian yang konkurensi. Walaupun hanya sinkronisasi saja yang dipertimbangkan dalam aljabar max-plus, hal ini sudah dapat menganalisa perilaku suatu sistem yang ada. Beberapa gambaran konkrit dari pemakaian aljabar max-plus adalah pada suatu jaringan sistem transportasi, hal ini bisa didapat di ([12], [3] dan [25]). Selain itu aljabar max-plus juga dapat digunakan untuk menganalisa kedinamikan sistem pada penjadwalan flow shop ([15]) dan rantai pasok ([21, 22]). Sedangkan pembahasan berkaitan dengan Penjadwalan Jalur Bus Dalam Kota, Sistem Transpotasi yang terintegrasi antara Monorail dan Trem, dan Analisis Jadwal Keberangkatan Pesawat Transit di Bandara dapat dijumpai di [16, 20] dan [19]. Dalam konteks aljabar max-plus sistem model yang terjadi adalah linier dan non-linier pada aljabar biasa. Beberapa penghitungan dalam aljabar maxplus pada contoh-contoh menggunakan maxplus aljabar toolbox versi 3.0.0 ([26]). Masalah teori spektral seperti halnya dalam aljabar biasa, dalam aljabar max-plus sangat penting dimana hal ini berkaitan erat dengan bentuk matriks tak-tereduksi atau tereduksi. Bahasan ini mencakup apa yang dinamakan *eigenmode* tergenerarisasi dan bisa dilihat di [17]. Pada bagian berikutnya dibahas pengertian aljabar max-plus dan beberapa notasi yang digunakan. Pembahasan yang lengkap dan rinci mengenai aljabar max-plus bisa dijumpai di [2] dan [1].

1.1 Aljabar Max-Plus

Dalam bagian ini dibahas beberapa konsep dasar yang akan digunakan untuk membahas sistem linear max-plus waktu-invariant. Pembahasan meliputi *semimodul* $\mathbb{R}_\varepsilon^n$ atas aljabar max-plus $\mathbb{R}_\varepsilon$, sistem persamaan linear max-plus, aljabar max-plus dan pengertian graf berarah.

Pembahasan dimulai dengan pengertian semi ring dan contohnya. Selanjutnya operasi pada $\mathbb{R}_\varepsilon$ diperluas untuk matriks dalam $\mathbb{R}_\varepsilon^{m \times n}$ serta relasi *urutan* didalamnya.

Definisi 1.1.1 *Suatu semiring $(\mathbf{S}, +, \times)$, adalah suatu himpunan takkosong $\mathbf{S}$ disertai dengan dua operasi biner $+$ dan $\times$, yang memenuhi aksioma berikut:*

i) $(\mathbf{S}, +)$ merupakan semigrup komutatif dengan elemen netral $\mathbf{0}$, yaitu $\forall x, y, z \in \mathbf{S}$ memenuhi

$$\begin{aligned} x + y &= y + x \\ (x + y) + z &= x + (y + z) \\ x + \mathbf{0} &= \mathbf{0} + x = x, \end{aligned}$$

ii) $(\mathbf{S}, \times)$ adalah semigrup dengan elemen satuan $\mathbf{1}$, yaitu $\forall x, y, z \in \mathbf{S}$ memenuhi

$$\begin{aligned} (x \times y) \times z &= x \times (y \times z) \\ x \times \mathbf{1} &= \mathbf{1} \times x = x, \end{aligned}$$

iii) sifat penyerapan elemen netral $\mathbf{0}$ terhadap operasi $\times$, yaitu $\forall x \in \mathbf{S}$ memenuhi

$$x \times \mathbf{0} = \mathbf{0} \times x = \mathbf{0}.$$

iv) Operasi $\times$ distributif terhadap $+$, yaitu $\forall x, y, z \in \mathbf{S}$ berlaku

$$\begin{aligned} (x + y) \times z &= (x \times z) + (y \times z), \\ x \times (y + z) &= (x \times y) + (x \times z). \end{aligned}$$

□

Contoh 1.1.1 Diberikan $\mathbb{R}_\varepsilon \stackrel{\text{def}}{=} \mathbb{R} \cup \{\varepsilon\}$ dengan $\mathbb{R}$ adalah himpunan semua bilangan real dan $\varepsilon \stackrel{\text{def}}{=} -\infty$. Pada $\mathbb{R}_\varepsilon$ didefinisikan operasi berikut: $\forall x, y \in \mathbb{R}_\varepsilon$,

$$x \oplus y \stackrel{\text{def}}{=} \max\{x, y\} \quad \text{dan} \quad x \otimes y \stackrel{\text{def}}{=} x + y.$$

Jadi $10 \oplus -10 = \max\{10, -10\} = 10$ dan $-7 \otimes 14 = -7 + 14 = 7$. Selanjutnya ditunjukkan $(\mathbb{R}_\varepsilon, \oplus, \otimes)$ merupakan semiring dengan elemen netral ε dan elemen satuan $e = 0$, karena untuk setiap $x, y, z \in \mathbb{R}_\varepsilon$ berlaku:

$$\begin{aligned} i) \quad x \oplus y &= \max\{x, y\} = \max\{y, x\} = y \oplus x, \\ (x \oplus y) \oplus z &= \max\{\max\{x, y\}, z\} = \max\{x, y, z\} = \max\{x, \max\{y, z\}\} = x \oplus (y \oplus z), \\ x \oplus \varepsilon &= \max\{x, -\infty\} = \max\{-\infty, x\} = \varepsilon \oplus x = x. \end{aligned}$$

$$\begin{aligned} ii) \quad (x \otimes y) \otimes z &= (x + y) + z = x + (y + z) = x \otimes (y \otimes z), \\ x \otimes e &= x + 0 = 0 + x = e \otimes x = x, \end{aligned}$$

$$iii) \quad x \otimes \varepsilon = x + (-\infty) = -\infty = -\infty + x = \varepsilon \otimes x,$$

$$\begin{aligned} iv) \quad (x \oplus y) \otimes z &= \max\{x, y\} + z = \max\{x + z, y + z\} = (x \otimes z) \oplus (y \otimes z), \\ x \otimes (y \oplus z) &= x + \max\{y, z\} = \max\{x + y, x + z\} = (x \otimes y) \oplus (x \otimes z). \end{aligned}$$

□

Selanjutnya untuk lebih ringkasnya, penulisan semiring $(\mathbb{R}_\varepsilon, \oplus, \otimes)$ ditulis sebagai $\mathbb{R}_{\max}$.

Definisi 1.1.2 Bila suatu semiring $(\mathbf{S}, +, \times)$ terhadap operasi $\times$ berlaku $\forall x, y \in \mathbf{S}, x \times y = y \times x$, maka dikatakan semiring komutatif.

□

Definisi 1.1.3 Bila suatu semiring $(\mathbf{S}, +, \times)$ mempunyai sifat **idempoten** terhadap operasi $+$ yaitu untuk setiap x di $\mathbf{S}$ berlaku $x + x = x$, maka dikatakan semiring idempoten atau **dioid**.

□

Contoh 1.1.2 Semiring $\mathbb{R}_\varepsilon$ merupakan semiring komutatif yang sekaligus idempoten, sebab untuk setiap $x, y \in \mathbb{R}_\varepsilon$ berlaku $x \otimes y = x + y = y + x = y \otimes x$ dan $x \oplus x = \max\{x, x\} = x$. □

Definisi 1.1.4 Suatu semiring komutatif $(S, +, \times)$ dinamakan semifield bila setiap elemen x di $S - \{0\}$ mempunyai invers terhadap operasi $\times$, yaitu untuk setiap x di $S - \{0\}$ ada x^{-1} sehingga $x \times x^{-1} = x^{-1} \times x = 1$. □

Contoh 1.1.3 Semiring komutatif $\mathbb{R}_{\max}$ merupakan semifield, sebab untuk setiap $x \in \mathbb{R}$ ada $-x$, sehingga berlaku $x \otimes (-x) = x + (-x) = 0$. □

Dari Contoh 1.1.2 dan 1.1.3 di atas terlihat bahwa $\mathbb{R}_{\max}$ merupakan semifield idempoten. Elemen-elemen di $\mathbb{R}_\varepsilon$ disebut juga dengan *skalar*. Seperti halnya dalam aljabar biasa, prioritas urutan operasi $\otimes$ lebih dulu atas operasi $\oplus$. Misalnya,

$$10 \otimes -7 \oplus 6 \otimes 2$$

mempunyai arti

$$(10 \otimes -7) \oplus (6 \otimes 2) = 3 \oplus 8 = \max\{3, 8\} = 8,$$

perintah dalam Scilab sbb:

```
-->maxplusoplus(maxplusotimes(10,-7),maxplusotimes(6,2))
ans =
    8.
```

sedangkan

$$10 \otimes (-7 \oplus 6) \otimes 2 = 10 \otimes 6 \otimes 2 = 10 + 6 + 2 = 18,$$

perintah dalam Scilab sbb:

```
-->maxplusotimes(maxplusotimes(10,maxplusoplus(-7,6)),2)
ans =
    18
```

Pangkat dalam aljabar max-plus secara biasa diperkenalkan dengan menggunakan sifat asosiatif. Himpunan bilangan asli digabung dengan bilangan nol dinotasikan oleh $\mathbb{N}$ dan didefinisikan untuk $x \in \mathbb{R}_\varepsilon$ dan untuk semua $n \in \mathbb{N}$ dengan $n \neq 0$

$$x^{\otimes n} \stackrel{\text{def}}{=} \underbrace{x \otimes x \otimes \dots \otimes x}_n \quad (1.1)$$

sedangkan untuk $n = 0$ didefinisikan $x^{\otimes n} \stackrel{\text{def}}{=} e (= 0)$. Perhatikan bahwa untuk setiap $n \in \mathbb{N}$, $x^{\otimes n}$ dalam aljabar biasa dibaca sebagai

$$x^{\otimes n} \stackrel{\text{def}}{=} \underbrace{x + x + \dots + x}_n = n \times x.$$

Suatu contoh, misalnya

$$9^{\otimes 2} = 2 \times 9 = 18.$$

Terinspirasi oleh pengertian pangkat ini, dengan cara serupa pangkat negatif dari bilangan real sebagai mana contoh berikut

$$8^{\otimes -2} = -2 \times 8 = -16 = 16^{\otimes -1}.$$

Hal yang sama, akar-akar max-plus diperkenalkan sebagai

$$x^{\otimes \alpha} = \alpha \times x, \quad \text{untuk } \alpha \in \mathbb{R}.$$

Suatu contoh, misalnya

$$9^{\otimes \frac{1}{3}} = \frac{1}{3} \times 9 = 3$$

dan

$$16^{\otimes -\frac{1}{4}} = -\frac{1}{4} \times 16 = -4.$$

Penghitungan pangkat dalam contoh-contoh tsb. bisa dilakukan dengan menggunakan aljabar maxplus toolbox ([26]).

```
-->maxpluspwr(9,2)
ans =
    18.
-->maxpluspwr(8,-2)
ans =
   -16.
// cek apakah maxpluspwr(8,-2)=maxpluspwr(16,-1) sbb:
-->isequal( maxpluspwr(8,-2),maxpluspwr(16,-1))
ans =
    T
-->maxpluspwr(9,1/3)
ans =
     3.
-->maxpluspwr(16,-1/4)
ans =
    - 4.
```

Sebagai mana telah ditunjukkan dalam Contoh 1.1.2 dan 1.1.3 bahwa $\mathbb{R}_\varepsilon$ merupakan semifield idempoten, yaitu semiring komutatif yang idempoten dengan setiap elemen $x \neq \varepsilon$ mempunyai invers $-x$ terhadap operasi $\otimes$. Berikut ini diberikan lagi beberapa contoh dari semiring komutatif yang idempoten.

Contoh 1.1.4

- Aljabar min-plus didefinisikan sebagai $\mathbb{R}_{\min} = (\mathbb{R}_{\varepsilon'}, \oplus', \otimes)$ dimana $\mathbb{R}_{\varepsilon'} = \mathbb{R} \cup \{\varepsilon'\}$ dengan $\varepsilon' \stackrel{\text{def}}{=} +\infty$ dan $x \oplus' y \stackrel{\text{def}}{=} \min\{x, y\}$ untuk semua $x, y \in \mathbb{R}_{\varepsilon'}$. Struktur aljabar min-plus $\mathbb{R}_{\min} = (\mathbb{R}_{\varepsilon'}, \oplus', \otimes)$ isomorpik dengan struktur aljabar max-maxplus $\mathbb{R}_{\max} = (\mathbb{R}_{\varepsilon}, \oplus, \otimes)$. Hal bisa ditunjukkan sebagai berikut: Dikonstruksi suatu pemetaan

$$f : \mathbb{R}_{\varepsilon} \rightarrow \mathbb{R}_{\varepsilon'}$$

dengan $f(x) = -x$ untuk setiap $x \in \mathbb{R}_{\varepsilon}$. Didapat untuk setiap $x, y \in \mathbb{R}_{\varepsilon}$

$$f(x \oplus y) = -(x \oplus y) = -\max\{x, y\} = \min\{-x, -y\} = f(x) \oplus' f(y)$$

dan

$$f(x \otimes y) = -(x + y) = (-x) + (-y) = f(x) \otimes f(y).$$

Juga $f(\varepsilon) = f(-\infty) = -(-\infty) = +\infty$ dan $f(0) = -0 = 0$. Jelas bahwa pemetaan f adalah bijektif. Dengan demikian $\mathbb{R}_{\varepsilon} \cong \mathbb{R}_{\varepsilon'}$.

- Misalkan $\mathbb{R}_{\max, \min} = (\overline{\mathbb{R}}, \oplus', \oplus)$ dengan $\overline{\mathbb{R}} = \mathbb{R} \cup \{\varepsilon, \varepsilon'\}$, $\varepsilon \oplus \varepsilon' = \varepsilon' \oplus \varepsilon = \varepsilon'$ dan $\varepsilon \oplus' \varepsilon' = \varepsilon' \oplus' \varepsilon = \varepsilon$.
- Aljabar max didefinisikan $\mathbb{R}_{\times} = (\mathbb{R}^{\text{pos}}, \oplus, \odot)$, dimana $\mathbb{R}^{\text{pos}} \stackrel{\text{def}}{=} \{x \in \mathbb{R} \mid x \geq 0\}$ dan $\odot \stackrel{\text{def}}{=} \times$. Struktur aljabar dari $\mathbb{R}_{\times} = (\mathbb{R}^{\text{pos}}, \oplus, \odot)$ isomorpik dengan $\mathbb{R}_{\max} = (\mathbb{R}_{\varepsilon}, \oplus, \otimes)$. Hal ini bisa ditunjukkan sebagai berikut. Definisikan pemetaan $f : \mathbb{R}_{\varepsilon} \rightarrow \mathbb{R}^{\text{pos}}$ yang diberikan oleh

$$f(x) \stackrel{\text{def}}{=} e^x, \quad \forall x \in \mathbb{R}_{\varepsilon}.$$

Didapat

1. Untuk setiap $x, y \in \mathbb{R}_{\varepsilon}$ bila $x \leq y$, maka $f(x \oplus y) = f(y) = e^y = e^x \oplus e^y = f(x) \oplus f(y)$. Bila $y \leq x$, maka $f(x \oplus y) = f(x) = e^x = e^x \oplus e^y = f(x) \oplus f(y)$. Dengan demikian untuk setiap $x, y \in \mathbb{R}_{\varepsilon}$ berlaku

$$f(x \oplus y) = f(x) \oplus f(y).$$

2. Untuk setiap $x, y \in \mathbb{R}_{\varepsilon}$, maka $f(x \otimes y) = f(x + y) = e^{x+y} = e^x e^y = f(x) \odot f(y)$.
3. Elemen nol di $\mathbb{R}_{\varepsilon}$ adalah $\varepsilon = -\infty$ dan elemen nol di $\mathbb{R}^{\text{pos}}$ adalah 0. Sedangkan elemen satuan di $\mathbb{R}_{\varepsilon}$ adalah 0 dan elemen satuan di $\mathbb{R}^{\text{pos}}$ adalah 1. Sehingga didapat

$$f(\varepsilon) = e^{\varepsilon} = \frac{1}{e^{\infty}} = 0 \text{ dan } f(0) = e^0 = 1.$$

Terlihat bahwa f adalah pemetaan homomorfisma dari $\mathbb{R}_{\varepsilon}$ ke $\mathbb{R}^{\text{pos}}$. Selanjutnya dapat ditunjukkan bahwa f adalah pemetaan bijektif. Dengan demikian $\mathbb{R}_{\varepsilon} \cong \mathbb{R}^{\text{pos}}$.

- Misalkan himpunan $\mathbf{S}$ takkosong dan $\mathbf{R}$ adalah himpunan dari semua himpunan bagian dari $\mathbf{S}$, maka $(\mathbf{R}, \cup, \cap)$ merupakan semiring komutatif idempoten dengan $X \cup \emptyset = \emptyset \cup X = X, \forall X \in \mathbf{R}$ dan $X \cap \mathbf{S} = \mathbf{S} \cap X = X, \forall X \in \mathbf{R}$. Hal yang sama $(\mathbf{R}, \cap, \cup)$ merupakan semiring komutatif idempoten.

□

Struktur aljabar semiring komutatif idempoten ini berbeda dengan aljabar biasa yang telah banyak dikenal. Hal ini dapat dilihat dalam masalah berikut. Apakah mungkin untuk mendefinisikan elemen invers terhadap operasi $\oplus$ dalam $\mathbb{R}_\varepsilon$? Suatu contoh, apakah mungkin mendapatkan penyelesaian persamaan

$$8 \oplus x = 4? \quad (1.2)$$

Sebagaimana dalam aljabar biasa bila kedua sisi persamaan (1.2) dikurangi dengan 8, didapat penyelesaian

$$x = -8 \oplus 4.$$

Tetapi, apakah mungkin memberikan arti terhadap -8 dalam persamaan diatas ? Apapun hal ini, bila kembali pada persamaan (1.2) didapat persamaan

$$\max\{8, x\} = 4. \quad (1.3)$$

Jelas bahwa tidak akan ada bilangan yang memenuhi persamaan (1.3). Dilain pihak dalam aljabar min-plus, persamaan (1.3) menjadi

$$\min\{8, x\} = 4$$

mempunyai penyelesaian $x = 4$. Selanjutnya bila dipertukarkan bilangan 4 dan 8 dalam persamaan (1.2) didapat $4 \oplus x = 8$. Persamaan ini tidak mempunyai penyelesaian dalam aljabar min-plus. Dari apa yang telah didiskusikan ini, muncul suatu pertanyaan apakah ada suatu semiring khusus sedemikian hingga semua persamaan yang berbentuk persamaan (1.2) mempunyai penyelesaian. Teorema berikut merupakan jawabannya.

Teorema 1.1.1 *Diberikan semiring $\mathbb{R}_{\max} = (\mathbb{R}, \oplus, \otimes)$. Idempoten dari $\oplus$ berakibat bahwa elemen invers terhadap $\oplus$ tidak ada.*

Bukti Misalkan bahwa $a \neq \varepsilon$ mempunyai suatu invers terhadap $\oplus$ yaitu b , didapat

$$a \oplus b = \varepsilon.$$

Tambahkan a pada kedua ruas persamaan, didapat

$$a \oplus a \oplus b = a \oplus \varepsilon.$$

Dengan sifat idempoten, persamaan menjadi

$$a \oplus b = a.$$

hal ini bertentangan dengan kenyataan $a \oplus b = \varepsilon$ dan $a \neq \varepsilon$.

□

1.2 Vektor dan Matriks

Dalam bagian ini dikenalkan matriks atas $\mathbb{R}_\varepsilon$. Himpunan matriks ukuran $n \times m$ dalam aljabar max-plus dinotasikan oleh $\mathbb{R}_\varepsilon^{n \times m}$. Untuk $n \in \mathbb{N}$ dengan $n \neq 0$, didefinisikan

$$\underline{n} \stackrel{\text{def}}{=} \{1, 2, \dots, n\}.$$

Elemen $A \in \mathbb{R}_\varepsilon^{n \times m}$ baris ke- i kolom ke- j dinotasikan oleh $a_{i,j}$ untuk $i \in \underline{n}$ dan $j \in \underline{m}$. Dalam hal ini matriks A ditulis sebagai

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,m} \\ a_{2,1} & a_{2,2} & \dots & a_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \dots & a_{n,m} \end{bmatrix}.$$

Ada kalanya elemen $a_{i,j}$ juga dinotasikan sebagai

$$[A]_{i,j}, \quad i \in \underline{n}, \quad j \in \underline{m}. \quad (1.4)$$

Penjumlahan matriks $A, B \in \mathbb{R}_\varepsilon^{n \times m}$ dinotasikan oleh $A \oplus B$ didefinisikan oleh

$$\begin{aligned} [A \oplus B]_{i,j} &= a_{i,j} \oplus b_{i,j} \\ &= \max\{a_{i,j}, b_{i,j}\}, \end{aligned} \quad (1.5)$$

untuk $i \in \underline{n}$ dan $j \in \underline{m}$.

Contoh 1.2.1 Diberikan matriks

$$A = \begin{bmatrix} 1 & e \\ 2 & 5 \end{bmatrix} \quad \text{dan} \quad B = \begin{bmatrix} -3 & 3 \\ \varepsilon & 10 \end{bmatrix},$$

maka

$$\begin{aligned} [A \oplus B]_{1,1} &= 1 \oplus -3 = \max\{1, -3\} = 1 \\ [A \oplus B]_{1,2} &= e \oplus 3 = \max\{0, 3\} = 3 \\ [A \oplus B]_{2,1} &= 2 \oplus \varepsilon = \max\{2, -\infty\} = 2 \\ [A \oplus B]_{2,2} &= 5 \oplus 10 = \max\{5, 10\} = 10 \end{aligned}$$

Dengan menggunakan notasi matriks didapat

$$A \oplus B = \begin{bmatrix} 1 & 3 \\ 2 & 10 \end{bmatrix}$$

□

perintah dalam Scilab sbb:

```
-->A=[1 0;2 5]
A =
    1.    0.
    2.    5.
-->B=[-3 3;-%inf 10]
B =
   -3.    3.
  -Inf   10.
-->maxplusoplus(A,B)
ans =
    1.    3.
    2.   10.
```

Catatan bahwa, untuk $A, B \in \mathbb{R}_\varepsilon^{n \times m}$ berlaku bahwa $A \oplus B = B \oplus A$, sebab

$$[A \oplus B]_{i,j} = \max\{a_{i,j}, b_{i,j}\} = \max\{b_{i,j}, a_{i,j}\} = [B \oplus A]_{i,j},$$

untuk $i \in \underline{n}$ dan $j \in \underline{m}$.

Untuk $A \in \mathbb{R}_\varepsilon^{n \times m}$ dan skalar $\alpha \in \mathbb{R}_\varepsilon$ perkalian $\alpha \otimes A$ didefinisikan sebagai

$$[\alpha \otimes A]_{i,j} \stackrel{\text{def}}{=} \alpha \otimes a_{i,j} \quad (1.6)$$

untuk $i \in \underline{n}$ dan $j \in \underline{m}$. Sebagai contoh, misalkan matriks A seperti dalam Contoh 1.2.1 dan $\alpha = 3$, maka

$$\begin{aligned} [\alpha \otimes A]_{1,1} &= 3 \otimes 1 = 3 + 1 = 4 \\ [\alpha \otimes A]_{1,2} &= 3 \otimes e = 3 + 0 = 3 \\ [\alpha \otimes A]_{2,1} &= 3 \otimes 2 = 3 + 2 = 5 \\ [\alpha \otimes A]_{2,2} &= 3 \otimes 5 = 3 + 5 = 8 \end{aligned}$$

Dengan menggunakan notasi matriks didapat

$$\alpha \otimes A = \begin{bmatrix} 4 & 3 \\ 5 & 8 \end{bmatrix}$$

dalam Scilab perintahnya:

```
-->alp=3
alp =
    3.
-->A=[1 0;2 5]
A =
    1.    0.
    2.    5.
-->maxplusotimes(alp,A)
```

ans =

4. 3.
5. 8.

Untuk matriks $A \in \mathbb{R}_\varepsilon^{n \times p}$ dan $B \in \mathbb{R}_\varepsilon^{p \times m}$ perkalian matriks $A \otimes B$ didefinisikan sebagai

$$\begin{aligned} [A \otimes B]_{i,j} &= \bigoplus_{k=1}^p a_{i,k} \otimes b_{k,j} \\ &= \max_{k \in \underline{p}} \{a_{i,k} + b_{k,j}\}, \end{aligned} \quad (1.7)$$

untuk $i \in \underline{n}$ dan $j \in \underline{m}$. Perkalian matriks ini serupa dalam perkalian matriks aljabar biasa dimana $+$ diganti dengan $\max$ dan $\times$ dengan $+$.

Contoh 1.2.2 Diberikan matriks

$$A = \begin{bmatrix} 1 & e \\ 2 & 5 \end{bmatrix} \quad \text{dan} \quad B = \begin{bmatrix} -3 & 3 \\ \varepsilon & 10 \end{bmatrix},$$

maka

$$\begin{aligned} [A \otimes B]_{1,1} &= 1 \otimes -3 \oplus e \otimes \varepsilon = \max\{1 + (-3), 0 + (-\infty)\} = -2 \\ [A \otimes B]_{1,2} &= 1 \otimes 3 \oplus e \otimes 10 = \max\{1 + 3, 0 + 10\} = 10 \\ [A \otimes B]_{2,1} &= 2 \otimes -3 \oplus 5 \otimes \varepsilon = \max\{2 + (-3), 5 + (-\infty)\} = -1 \\ [A \otimes B]_{2,2} &= 2 \otimes 3 \oplus 5 \otimes 10 = \max\{2 + 3, 5 + 10\} = 15 \end{aligned}$$

Dengan menggunakan notasi matriks didapat

$$A \otimes B = \begin{bmatrix} -2 & 10 \\ -1 & 15 \end{bmatrix}.$$

□

Perhatikan bahwa perkalian matriks tidak selalu komutatif. Untuk matriks A dan B dalam Contoh 1.2.2 didapat

$$B \otimes A = \begin{bmatrix} 5 & 8 \\ 12 & 15 \end{bmatrix} \neq A \otimes B.$$

Dalam Scilab ketik:

```
-->A=[1 0;2 5]
A =
    1.    0.
    2.    5.
-->B=[-3 3;-%inf 10]
B =
   -3.    3.
  -Inf   10.
```

```
-->maxplusotimes(A,B)
ans =
- 2.    10.
- 1.    15.
-->maxplusotimes(B,A)
ans =
5.      8.
12.     15.
```

cek bahwa $A \otimes B \neq B \otimes A$:

```
-->~isequal(maxplusotimes(A,B),maxplusotimes(B,A))
ans =
T
```

Sifat-sifat berikut berkaitan dengan sifat-sifat elementer perkalian dan penjumlahan matriks dalam aljabar maxplus.

Teorema 1.2.1 *Beberapa sifat berikut berlaku untuk sebarang matriks A, B dan C dengan ukuran yang bersesuaian dan operasi matriks terdefinisi.*

- i) $(A \oplus B) \oplus C = A \oplus (B \oplus C)$
- ii) $(A \otimes B) \otimes C = A \otimes (B \otimes C)$
- iii) $A \otimes (B \oplus C) = (A \otimes B) \oplus (A \otimes C)$
- iv) $(A \oplus B) \otimes C = (A \otimes C) \oplus (B \otimes C)$
- v) $A \oplus A = A$.

Bukti

Akan dibuktikan untuk ii) dan iii) sedangkan bukti yang lainnya mengikuti dari definisi operasi dan sifat-sifat operasi pada $\mathbb{R}_\varepsilon$. Untuk membuktikan ii), ambil sebarang matriks $A \in \mathbb{R}_\varepsilon^{n \times p}$, $B \in \mathbb{R}_\varepsilon^{p \times q}$, $C \in \mathbb{R}_\varepsilon^{q \times m}$. Elemenbaris ke- i kolom ke- j matriks $(A \otimes B) \otimes C$ adalah

$$\begin{aligned}
 [(A \otimes B) \otimes C]_{i,j} &= \bigoplus_{k=1}^q \left(\bigoplus_{l=1}^p a_{i,l} \otimes b_{l,k} \right) \otimes c_{k,j} \\
 &= \bigoplus_{k=1}^q \bigoplus_{l=1}^p a_{i,l} \otimes b_{l,k} \otimes c_{k,j} \\
 &= \bigoplus_{l=1}^p a_{i,l} \otimes \left(\bigoplus_{k=1}^q b_{l,k} \otimes c_{k,j} \right) \\
 &= [A \otimes (B \otimes C)]_{i,j},
 \end{aligned}$$

untuk $i \in \underline{n}$ dan $j \in \underline{m}$. Bukti iii), ambil sebarang matriks $A \in \mathbb{R}_\varepsilon^{n \times p}$ dan $B, C \in \mathbb{R}_\varepsilon^{p \times m}$. Elemenbaris ke- i kolom ke- j matriks $A \otimes (B \oplus C)$ adalah

$$\begin{aligned}
 [A \otimes (B \oplus C)]_{i,j} &= \bigoplus_{k=1}^p a_{i,k} \otimes (b_{k,j} \oplus c_{k,j}) \\
 &= \bigoplus_{k=1}^p (a_{i,k} \otimes b_{k,j} \oplus a_{i,k} \otimes c_{k,j}) \\
 &= \left(\bigoplus_{k=1}^p a_{i,k} \otimes b_{k,j} \right) \oplus \left(\bigoplus_{k=1}^p a_{i,k} \otimes c_{k,j} \right) \\
 &= [(A \otimes B)]_{i,j} \oplus [(A \otimes C)]_{i,j},
 \end{aligned}$$

untuk $i \in \underline{n}$ dan $j \in \underline{m}$. □

Matriks $\varepsilon(n, m)$ menyatakan matriks ukuran $n \times m$ dengan semua elemen sama dengan ε dan matriks $E(n, m)$ adalah matriks ukuran $n \times m$ yang didefinisikan oleh

$$[E(n, m)]_{i,j} \stackrel{\text{def}}{=} \begin{cases} e & \text{untuk } i = j, \\ \varepsilon & \text{untuk } i \neq j. \end{cases}$$

Bila $n = m$, maka matriks $E(n, n)$ dinamakan matriks identitas. Bila dimensi jelas dalam konteks, matriks $\varepsilon(n, m)$ dan $E(n, m)$ cukup ditulis ε dan E . Hal berikut jelas bahwa untuk setiap matriks $A \in \mathbb{R}_\varepsilon^{n \times m}$ memenuhi

$$\begin{aligned}
 A \oplus \varepsilon(n, m) &= \varepsilon(n, m) \oplus A, \\
 A \otimes E(m, m) &= A = E(n, n) \otimes A.
 \end{aligned}$$

Matriks $\varepsilon(m, n)$ dan $E(m, n)$ dalam Scilab, misal $\varepsilon(3, 5)$

```
-->maxpluszeros(3,5)
ans =
  -Inf  -Inf  -Inf  -Inf  -Inf
  -Inf  -Inf  -Inf  -Inf  -Inf
  -Inf  -Inf  -Inf  -Inf  -Inf
```

sedangkan untuk $E(5, 3)$ dan $E(3, 3)$ adalah :

```
-->maxpluseye(5,3)
ans =
   0.  -Inf  -Inf
  -Inf   0.  -Inf
  -Inf  -Inf   0.
  -Inf  -Inf  -Inf
  -Inf  -Inf  -Inf
-->maxpluseye(3,3)
ans =
   0.  -Inf  -Inf
  -Inf   0.  -Inf
  -Inf  -Inf   0.
```

Dalam $\mathbb{R}_\varepsilon^{m \times n}$ penjumlahan matriks $\oplus$ sebagaimana didefinisikan di (1.5) adalah asosiatif, komutatif dan mempunyai elemen nol $\varepsilon(n, m)$. Dalam $\mathbb{R}_\varepsilon^{n \times n}$, perkalian matriks $\otimes$ sebagaimana didefinisikan di (1.7) adalah asosiatif, distributif terhadap $\oplus$ dan mempunyai elemen satuan $E(n, n)$ serta elemen penyerap $\varepsilon(n, n)$ untuk $\otimes$. Dalam pembahasan matriks ini, $(\mathbb{R}_\varepsilon^{n \times n}, \oplus, \otimes)$ adalah semiring idempoten dengan elemen nol ε dan elemen satuan E , tetapi bukan semiring komutatif sebagaimana ditunjukkan dalam Contoh 1.2.2.

Transpose dari suatu elemen $A \in \mathbb{R}_\varepsilon^{m \times n}$ dinotasikan oleh $A^\top$ didefinisikan sebagai $[A^\top]_{i,j} \stackrel{\text{def}}{=} a_{j,i}$, untuk $i \in \underline{n}$ dan $j \in \underline{m}$. Sebagaimana sebelumnya, juga dalam penjumlahan dan perkalian matriks operasi $\otimes$ mempunyai prioritas urutan atas operasi $\oplus$.

Untuk $A \in \mathbb{R}_\varepsilon^{n \times n}$, pangkat ke- k dari A dinotasikan oleh $A^{\otimes k}$ didefinisikan sebagai

$$A^{\otimes k} \stackrel{\text{def}}{=} \underbrace{A \otimes A \otimes \dots \otimes A}_k, \quad (1.8)$$

untuk $k \in \mathbb{N}$ dengan $k \neq 0$ dan $A^{\otimes 0} \stackrel{\text{def}}{=} E(n, n)$. Elemen baris ke- i kolom ke- j dari matriks $A^{\otimes 2}$ adalah

$$[A^{\otimes 2}]_{i,j} = \bigoplus_{r=1}^n a_{i,r} \otimes a_{r,j} = \max_{1 \leq r \leq n} \{a_{i,r} + a_{r,j}\}.$$

Elemen baris ke- i kolom ke- j dari matriks $A^{\otimes 3}$ adalah

$$\begin{aligned} [A^{\otimes 3}]_{i,j} &= \bigoplus_{r_2=1}^n a_{i,r_2} \left(\bigoplus_{r_1=1}^n a_{r_2,r_1} \otimes a_{r_1,j} \right) \\ &= \max_{1 \leq r_1, r_2 \leq n} \{a_{i,r_1} + a_{r_1,r_2} + a_{r_2,j}\}. \end{aligned}$$

Secara umum elemen baris ke- i kolom ke- j dari matriks $A^{\otimes k}$ adalah

$$\begin{aligned} [A^{\otimes k}]_{i,j} &= \bigoplus_{r_{k-1}=1}^n a_{i,r_{k-1}} \dots \left(\bigoplus_{r_1=1}^n a_{r_{k-1},r_1} \otimes a_{r_1,j} \right) \\ &= \max_{1 \leq r_1, \dots, r_{k-1} \leq n} \{a_{i,r_{k-1}} + \dots + a_{r_{k-1},r_1} + a_{r_1,j}\}. \end{aligned}$$

Selanjutnya, untuk $\alpha \in \mathbb{R}_\varepsilon$ dan $A \in \mathbb{R}_\varepsilon^{n \times n}$ elemen baris ke- i kolom ke- j matriks $(\alpha \otimes A)^{\otimes k}$ adalah

$$\begin{aligned} [(\alpha \otimes A)^{\otimes k}]_{i,j} &= \max_{1 \leq r_1, \dots, r_{k-1} \leq n} \{(\alpha + a_{i,r_{k-1}}) + \dots + (\alpha + a_{r_{k-1},r_1}) + (\alpha + a_{r_1,j})\} \\ &= \underbrace{\alpha + \alpha \dots + \alpha}_k + \left(\max_{1 \leq r_1, \dots, r_{k-1} \leq n} \{a_{i,r_{k-1}} + \dots + a_{r_{k-1},r_1} + a_{r_1,j}\} \right) \\ &= \alpha^{\otimes k} \otimes [A^{\otimes k}]_{i,j}. \end{aligned}$$

Sehingga untuk setiap $\alpha \in \mathbb{R}_\varepsilon$ dan $A \in \mathbb{R}_\varepsilon^{n \times n}$ berlaku

$$(\alpha \otimes A)^{\otimes k} = \alpha^{\otimes k} \otimes A^{\otimes k}, \quad \text{untuk } k = 1, 2, \dots \quad (1.9)$$

Selanjutnya untuk setiap $A \in \mathbb{R}_\varepsilon^{n \times n}$ trace dari matriks A dinotasikan oleh $\text{trace}(A)$ didefinisikan sebagai $\text{trace}(A) = \bigoplus_{i=1}^n a_{i,i}$.

Contoh 1.2.3 Diberikan matriks

$$A = \begin{bmatrix} 1 & 2 & \varepsilon \\ \varepsilon & 3 & 4 \\ 5 & \varepsilon & 6 \end{bmatrix}.$$

Maka matriks pangkat berikut

$$A^{\otimes 2} = A \otimes A = \begin{bmatrix} 1 & 2 & \varepsilon \\ \varepsilon & 3 & 4 \\ 5 & \varepsilon & 6 \end{bmatrix} \otimes \begin{bmatrix} 1 & 2 & \varepsilon \\ \varepsilon & 3 & 4 \\ 5 & \varepsilon & 6 \end{bmatrix} = \begin{bmatrix} 2 & 5 & 6 \\ 9 & 6 & 10 \\ 11 & 7 & 12 \end{bmatrix}$$

$$A^{\otimes 3} = A \otimes A^{\otimes 2} = \begin{bmatrix} 1 & 2 & \varepsilon \\ \varepsilon & 3 & 4 \\ 5 & \varepsilon & 6 \end{bmatrix} \otimes \begin{bmatrix} 2 & 5 & 6 \\ 8 & 6 & 10 \\ 10 & 6 & 12 \end{bmatrix} = \begin{bmatrix} 11 & 8 & 12 \\ 15 & 11 & 16 \\ 17 & 13 & 18 \end{bmatrix}$$

dan $\text{trace}(A) = \bigoplus_{i=1}^3 a_{i,i} = \max\{1, 3, 6\} = 6$, $\text{trace}(A^{\otimes 2}) = \bigoplus_{i=1}^3 [A^{\otimes 2}]_{i,i} = \max\{2, 6, 12\} = 12$,
 $\text{trace}(A^{\otimes 3}) = \bigoplus_{i=1}^3 [A^{\otimes 3}]_{i,i} = \max\{10, 10, 18\} = 18$. □

Hasil penghitungan Contoh 1.2.3 dilakukan dalam Scilab sbb:

```
-->A=[1 2 -%inf;-%inf 3 4;5 -%inf 6]
A =
    1.    2.   -Inf
  -Inf    3.    4.
    5.  -Inf    6.
-->a2=maxpluspwr(A,2)
a2 =
    2.    5.    6.
    9.    6.   10.
   11.    7.   12.
-->a3=maxpluspwr(A,3)
a3 =
   11.    8.   12.
   15.   11.   16.
   17.   13.   18.
-->maxplustrace(A)
ans =
    6.
```

```
-->maxplustrace(a2)
ans =
    12.
-->maxplustrace(a3)
ans =
    18.
```

Misalkan $(\mathbf{S}, +, \times)$ adalah semiring komutatif dengan elemen netral 0 dan 1. Semimodul $\mathbf{M}$ atas $\mathbf{S}$ adalah semigrup komutatif $(\mathbf{M}, +)$ bersama operasi perkalian skalar $\bullet : \mathbf{S} \times \mathbf{M} \rightarrow \mathbf{M}$, dituliskan sebagai $(\alpha, x) \mapsto \alpha \bullet x$ yang memenuhi aksioma berikut: $\forall \alpha, \beta \in \mathbf{S}$ dan $\forall x, y \in \mathbf{M}$ berlaku:

- i) $\alpha \bullet (x + y) = \alpha \bullet x + \alpha \bullet y$,
- ii) $(\alpha + \beta) \bullet x = \alpha \bullet x + \beta \bullet x$,
- iii) $\alpha \bullet (\beta \bullet x) = (\alpha \times \beta) \bullet x$,
- iv) $1 \bullet x = x$,
- v) $0 \bullet x = 0$.

Suatu elemen dari suatu semimodul dinamakan **vektor**. Suatu contoh, $\mathbb{R}_\varepsilon^{n \times 1}$ adalah semimodul atas $\mathbb{R}_\varepsilon$, dalam hal ini $\mathbb{R}_\varepsilon^{n \times 1}$ cukup ditulis $\mathbb{R}_\varepsilon^n$. Elemen ke- j dari suatu vektor $\mathbf{x} \in \mathbb{R}_\varepsilon^n$ dinotasikan oleh x_j dan ditulis juga sebagai $[\mathbf{x}]_j$. Vektor di $\mathbb{R}_\varepsilon^n$ dengan semua elemennya sama dengan e dinamakan **vektor satuan** dinotasikan oleh $\mathbf{u}$ ditulis sebagai $[\mathbf{u}]_j = e$ untuk semua $j \in \underline{n}$. Untuk setiap $\alpha \in \mathbb{R}_\varepsilon$ vektor $\alpha \otimes \mathbf{u} = \mathbf{u}[\alpha]$ adalah vektor yang semua elemennya sama dengan α . Untuk setiap $j \in \underline{n}$ kolom ke- j dari matriks satuan $E(n, n)$ dinamakan vektor basis ke- j dari $\mathbb{R}_\varepsilon^n$ dan dinotasikan oleh e_j . Jadi elemen ke- j dari vektor e_j sama dengan e sedangkan elemen lainnya sama dengan ε . Berikut ini diberikan suatu relasi pada suatu himpunan yang berkaitan dengan urutan dalam himpunan tersebut. Pengertian dari relasi ini dan beberapa sifat akan berguna dalam kajian aljabar max-plus $\mathbb{R}_\varepsilon$.

Definisi 1.2.1 Suatu relasi $\leq$ pada suatu himpunan $\mathbf{P}$ dinamakan urutan parsial pada $\mathbf{P}$ bila untuk semua $a, b, c \in \mathbf{P}$ memenuhi

- 1. $a \leq a$, sifat refleksif,
- 2. bila $a \leq b$ dan $b \leq a$, maka $a = b$, sifat antisimetri,
- 3. bila $a \leq b$ dan $b \leq c$, maka $a \leq c$, sifat transitif.

□

Selanjutnya bila berlaku $a \leq b$ atau $b \leq a$, maka a dan b dikatakan **dapat-dibandingkan** (comparable). Penulisan $a \leq b$ juga bisa ditulis $b \geq a$. Bila $a \leq b$ dan $a \neq b$, maka ditulis dengan $a \prec b$. Bila setiap dua elemen dari $\mathbf{P}$ dapat-dibandingkan, maka urutan parsial $\leq$ dinamakan **urutan total**.

Berikut ini diberikan suatu teorema yang berkaitan dengan pengertian urutan parsial pada suatu semigrup komutatif idempoten.

Teorema 1.2.2 *Diberikan suatu semigrup komutatif idempoten $(\mathbf{S}, +)$. Bila pada $\mathbf{S}$ didefinisikan suatu relasi $\leq$ oleh $a \leq b \Leftrightarrow a + b = b$, maka relasi $\leq$ adalah urutan parsial pada $\mathbf{S}$.*

Bukti

Diberikan sebarang elemen a, b dan c di $\mathbf{S}$, maka

1. karena $\mathbf{S}$ idempoten, maka $a + a = a \Leftrightarrow a \leq a$,
2. bila $a \leq b$ dan $b \leq a$, maka $a + b = b$ dan $b + a = a$ dan karena $\mathbf{S}$ komutatif, maka $a + b = b + a = a$, jadi $a = b$,
3. bila $a \leq b$ dan $b \leq c$, maka $a + b = b$ dan $b + c = c$ dan karena $\mathbf{S}$ mempunyai sifat assosiatif, maka $a + c = a + (b + c) = (a + b) + c = b + c = c$, jadi $a \leq c$.

□

Contoh 1.2.4 Dalam $\mathbb{R}_\varepsilon$ relasi $\leq_{\max}$ yang didefinisikan sebagai

$$x \leq_{\max} y \Leftrightarrow x \oplus y = y \quad (1.10)$$

adalah urutan parsial sebab $(\mathbb{R}_\varepsilon, \oplus)$ adalah semigrup komutatif idempoten disertai dengan relasi $\leq_{\max}$ yang diberikan dalam 1.10 dan berdasarkan Teorema 1.2.2, maka relasi $\leq_{\max}$ pada $\mathbb{R}_\varepsilon$ adalah urutan parsial. Selanjutnya untuk setiap $x, y \in \mathbb{R}_\varepsilon$, berlaku

$$x \oplus y = \max\{x, y\} = y \Leftrightarrow x \leq_{\max} y \text{ atau } y \oplus x = \max\{x, y\} = x \Leftrightarrow y \leq_{\max} x.$$

Jadi relasi $\leq_{\max}$ terurut total.

□

Catatan bahwa, relasi $\leq_{\max}$ pada $\mathbb{R}_\varepsilon$ ekuivalen dengan relasi $\leq$ pada $\mathbb{R} \cup \{-\infty\}$, sebab

$$x \leq_{\max} y \Leftrightarrow x \oplus y = y \Leftrightarrow \max(x, y) = y \Leftrightarrow x \leq y.$$

Contoh 1.2.5 Dalam $\mathbb{R}_\epsilon^{m \times n}$ relasi $\leq_{\max}$ yang didefinisikan sebagai

$$A \leq_{\max} B \Leftrightarrow A \oplus B = B \Leftrightarrow [A \oplus B]_{i,j} = [B]_{i,j} \Leftrightarrow [A]_{i,j} \leq_{\max} [B]_{i,j}, \forall i \in \underline{m} \text{ dan } \forall j \in \underline{n} \quad (1.11)$$

adalah urutan parsial sebab $(\mathbb{R}_\epsilon^{m \times n}, \oplus)$ adalah semigrup komutatif idempoten disertai dengan relasi $\leq_{\max}$ yang diberikan dalam 1.11 dan berdasarkan Teorema 1.2.2, maka relasi $\leq_{\max}$ pada $\mathbb{R}_\epsilon^{m \times n}$ adalah urutan parsial. Urutan parsial ini bukan urutan total, sebab untuk dua matriks A dan B masing-masing berukuran 2×2 sebagai mana berikut

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, B = \begin{bmatrix} 0 & 3 \\ 4 & 1 \end{bmatrix}, \text{ maka } A \oplus B = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \oplus \begin{bmatrix} 0 & 3 \\ 4 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 3 \\ 4 & 4 \end{bmatrix}.$$

Terlihat bahwa $A \oplus B \neq B$ dan $B \oplus A \neq A$. □

Dalam Contoh 1.2.5 bila ukuran baris $m = n$ dan ukuran kolom $n = 1$, maka didapat $\mathbb{R}_\epsilon^n$. Sehingga dengan relasi $\leq_{\max}$ pada $\mathbb{R}_\epsilon^n$ yang diberikan oleh

$$\mathbf{x} \leq_{\max} \mathbf{y} \Leftrightarrow \mathbf{x} \oplus \mathbf{y} = \mathbf{y} \Leftrightarrow [\mathbf{x} \oplus \mathbf{y}]_j = [\mathbf{y}]_j \Leftrightarrow [\mathbf{x}]_j \leq_{\max} [\mathbf{y}]_j, \forall j \in \underline{n}$$

juga merupakan relasi urutan parsial yang bukan urutan total, sebab ada $\mathbf{x}, \mathbf{y} \in \mathbb{R}_\epsilon^2$ dengan

$$\mathbf{x} = \begin{bmatrix} 2 \\ -1 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} 0 \\ 3 \end{bmatrix}, \text{ berlaku } \mathbf{x} \oplus \mathbf{y} = \begin{bmatrix} 2 \\ -1 \end{bmatrix} \oplus \begin{bmatrix} 0 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}.$$

Terlihat bahwa $\mathbf{x} \oplus \mathbf{y} \neq \mathbf{y}$ dan $\mathbf{y} \oplus \mathbf{x} \neq \mathbf{x}$.

Sifat berikut mengenai perkalian matriks dengan vektor dalam $\mathbb{R}_\epsilon$ yang dikaitkan dengan relasi urutan parsial.

Teorema 1.2.3 Diberikan matriks $A \in \mathbb{R}_\epsilon^{m \times n}$. Bila vektor $\mathbf{x}, \mathbf{y} \in \mathbb{R}_\epsilon^n$ dengan $\mathbf{x} \leq_{\max} \mathbf{y}$, maka $A \otimes \mathbf{x} \leq_{\max} A \otimes \mathbf{y}$.

Bukti

Untuk sebarang elemen $\mathbf{x}, \mathbf{y} \in \mathbb{R}_\epsilon^n$ dengan $\mathbf{x} \leq_{\max} \mathbf{y}$ berlaku, maka

$$\begin{aligned} \mathbf{x} \oplus \mathbf{y} = \mathbf{y} &\Leftrightarrow A \otimes (\mathbf{x} \oplus \mathbf{y}) = A \otimes \mathbf{y} \\ &\Leftrightarrow (A \otimes \mathbf{x}) \oplus (A \otimes \mathbf{y}) = A \otimes \mathbf{y} \\ &\Leftrightarrow A \otimes \mathbf{x} \leq_{\max} A \otimes \mathbf{y}. \end{aligned}$$

□

Contoh 1.2.6 Diberikan matriks

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \text{ dan vektor } \mathbf{x} = \begin{bmatrix} 5 \\ 6 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} 7 \\ 8 \end{bmatrix}.$$

Jelas bahwa $\mathbf{x} \leq_{\max} \mathbf{y}$ dan

$$A \otimes \mathbf{x} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \otimes \begin{bmatrix} 5 \\ 6 \end{bmatrix} = \begin{bmatrix} 8 \\ 10 \end{bmatrix} \text{ dan } A \otimes \mathbf{y} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \otimes \begin{bmatrix} 7 \\ 8 \end{bmatrix} = \begin{bmatrix} 10 \\ 12 \end{bmatrix}.$$

Terlihat bahwa $A \otimes \mathbf{x} \leq_{\max} A \otimes \mathbf{y}$. □

Perintah dalam Scilab untuk Contoh 1.2.6:

```
-->A=[1 2;3 4]
A =
    1.    2.
    3.    4.
-->x=[5;6]
x =
    5.
    6.
-->y=[7;8]
y =
    7.
    8.
-->maxplusotimes(A,x) <= maxplusotimes(A,y)
ans =
    T
    T
```

Suatu pemetaan f dari $\mathbb{R}_\varepsilon^n$ ke $\mathbb{R}_\varepsilon^n$ dikatakan *affin* bila $f(\mathbf{x}) = A \otimes \mathbf{x} \oplus \mathbf{b}$ untuk beberapa $A \in \mathbb{R}_\varepsilon^{n \times n}$ dan $\mathbf{b} \in \mathbb{R}_\varepsilon^n$. Bila $\mathbf{b} = \varepsilon$, maka f dinamakan *linear*. Suatu relasi berulang $\mathbf{x}(k+1) = f(\mathbf{x}(k))$ untuk $k \in \mathbb{N}$ dinamakan *affin* atau *linear* bila f suatu pemetaan *affin* atau *linear*.

Suatu matriks $A \in \mathbb{R}_\varepsilon^{n \times m}$ dinamakan *reguler* bila disetiap baris A memuat setidaknya satu elemen tidak sama dengan ε . Kereguleran adalah suatu kondisi teknik belaka, bila A tidak reguler, maka A memuat baris redundan dan setiap sistem yang mempunyai model $\mathbf{x}(k+1) = A \otimes \mathbf{x}(k)$ juga bisa dimodelkan oleh versi redundan dari matriks A yang mana semua baris redundan dan kolom terkait diabaikan.

Suatu matriks $A \in \mathbb{R}_\varepsilon^{n \times n}$ dikatakan matriks *segitiga bawah* bila $a_{i,j} = \varepsilon$ untuk $1 \leq i \leq j \leq n$. Matriks A dikatakan matriks *segitiga atas* bila matriks transpose $A^\top$ merupakan matriks segitiga bawah.

Untuk himpunan terhitung (countable), operator max harus dipahami sebagai suatu supremum. Secara formal, misalkan $\{a_i | i \in \mathbb{N}\}$ adalah himpunan terhitung dengan $a_i \in \mathbb{R}_\varepsilon$, maka

$$\bigoplus_{i \geq 0} a_i \stackrel{\text{def}}{=} \bigoplus_{i=0}^{\infty} a_i \stackrel{\text{def}}{=} \sup_{i \geq 0} a_i.$$

Dalam aljabar max-plus mudah diselidiki aturan Fubini, yaitu untuk $\{a_{i,j} \in \mathbb{R}_\varepsilon | i, j \in \mathbb{N}\}$,

$$\bigoplus_{i \geq 0} \bigoplus_{j \geq 0} a_{i,j} = \bigoplus_{j \geq 0} \bigoplus_{i \geq 0} a_{i,j}. \quad (1.12)$$

Bahkan, untuk setiap $k, j \geq 0$, bila $a_{k,j} \leq \bigoplus_{i \geq 0} a_{i,j}$, maka

$$\bigoplus_{j \geq 0} a_{k,j} \leq \bigoplus_{j \geq 0} \bigoplus_{i \geq 0} a_{i,j},$$

untuk $k \geq 0$, sebagai akibat

$$\bigoplus_{k \geq 0} \bigoplus_{j \geq 0} a_{k,j} \leq \bigoplus_{j \geq 0} \bigoplus_{i \geq 0} a_{i,j}.$$

Bab 2

Teori Spektral

Dalam bab ini dibahas teori spektral dari matriks atas semiring $\mathbb{R}_\varepsilon$. Dalam Bagian 2.1 dikaji hubungan diantara graf dan matriks atas semiring $\mathbb{R}_\varepsilon$, dalam Bagian 2.2 dibahas pengertian nilai karakteristik dan vektor karakteristik dari suatu matriks persegi atas semiring $\mathbb{R}_\varepsilon$ selanjutnya dalam Bagian 2.4 dibahas suatu penyelesaian dari persamaan linear dalam $\mathbb{R}_\varepsilon$.

2.1 Matriks dan Graf

Misalkan matriks $A \in \mathbb{R}_{\max}^{n \times n}$, suatu graf berarah dari matriks A adalah $\mathcal{G}(A) = (\mathcal{N}, \mathcal{D})$. Graf $\mathcal{G}(A)$ mempunyai n titik, himpunan semua titik dari $\mathcal{G}(A)$ dinyatakan oleh $\mathcal{N}$. Himpunan semua *arc* (garis) dari graf $\mathcal{G}(A)$ atau pasangan terurut dari beberapa titik di $\mathcal{N}$ dinotasikan oleh $\mathcal{D}$. Suatu garis dari titik j ke titik i *exist* (ada) bila $a_{i,j} \neq \varepsilon$, garis ini dinotasikan oleh (j, i) , dengan demikian $(j, i) \in \mathcal{D}$. Bobot dari garis (j, i) adalah nilai dari $a_{i,j}$ yang dinotasikan oleh $w(j, i) = a_{i,j} \in \mathbb{R}_\varepsilon$. Bila $a_{i,j} = \varepsilon$, maka garis (j, i) tidak ada. Suatu barisan garis $(i_1, i_2), (i_2, i_3), \dots, (i_{l-1}, i_l)$ dari suatu graf dinamakan suatu path. Suatu path dikatakan **elementer** bila tidak ada titik terjadi dua kali dalam path tersebut atau semua titik yang termuat dalam path tersebut berbeda. Suatu sirkuit adalah **path elementer tertutup**, yaitu $(i_1, i_2), (i_2, i_3), \dots, (i_{l-1}, i_1)$. Bobot dari suatu path $p = (i_1, i_2), (i_2, i_3), \dots, (i_{l-1}, i_l)$ dinotasikan oleh $|p|_w$ dan diberikan oleh $|p|_w = w(i_1, i_2) + w(i_2, i_3) + \dots + w(i_{l-1}, i_l) = (a_{i_2, i_1} + a_{i_3, i_2} + \dots + a_{i_l, i_{l-1}})$, sedangkan panjang dari path p atau banyaknya garis dalam path p dinotasikan oleh $|p|_l$. Himpunan semua path dari titik i ke titik j dengan panjang k dinotasikan oleh $P(j, i; k)$. Bobot rata-rata dari path p adalah bobot dari p dibagi oleh banyaknya garis dalam path p , yaitu

$$\frac{|p|_w}{|p|_l} = \frac{(a_{i_2, i_1} + a_{i_3, i_2} + \dots + a_{i_l, i_{l-1}})}{(l-1)}.$$

Sirkuit rata-rata adalah bobot rata-rata dari suatu sirkuit. Sebarang sirkuit dengan sirkuit rata-rata maksimum dinamakan **sirkuit kritis**. Suatu graf dikatakan **strongly connected**

bila suatu path ada untuk setiap titik i ke setiap titik j . Bila graf $\mathcal{G}(A)$ adalah strongly connected, maka matriks A juga dikatakan *irreducible* (tak-tereduksi).

Graf kritis dari $\mathcal{G}(A)$ dinotasikan dengan $\mathcal{G}^c(A) = (\mathcal{N}^c(A), \mathcal{D}^c(A))$ adalah graf yang terdiri dari himpunan titik dan *arc* yang berada pada sirkuit kritis dari graf $\mathcal{G}(A)$. Dalam [1] dijelaskan bahwa suatu titik $i \in \mathcal{N}^c(A)$ dapat disebut sebagai titik kritis. Demikian pula, *subpath* dari suatu sirkuit kritis disebut sebagai *path* kritis.

Graf dan matriks representasi dari graf saling berkaitan satu sama lain. Kondisi dari suatu graf dapat dibaca melalui matriks representasinya, begitu pula sebaliknya. Salah satu contoh keterkaitan tersebut adalah panjang dari suatu *path* dalam graf berhubungan dengan pangkat dari matriks representasinya. Berikut diberikan teorema yang menjelaskan bahwa elemen $[A^{\otimes k}]_{i,j}$ menghasilkan bobot maksimal dari suatu *path* dengan panjang k dari titik j ke titik i , asalkan suatu *path* ada.

Teorema 2.1.1 Diberikan $A \in \mathbb{R}_\varepsilon^{n \times n}$. Untuk setiap $k \geq 1$ berlaku

$$[A^{\otimes k}]_{i,j} = \max\{|p|_w : p \in P(j, i; k)\},$$

dengan $[A^{\otimes k}]_{i,j} = \varepsilon$ pada kasus $P(j, i; k)$ adalah himpunan kosong, yaitu ketika tidak ada *path* dengan panjang k dari j ke i dalam $\mathcal{G}(A)$.

Bukti Pembuktian dilakukan dengan langkah induksi. Misalkan (j, i) adalah sebarang elemen di $\underline{n} \times \underline{n}$. Untuk $k = 1$, *path* pada $P(j, i; k)$ hanya terdiri dari satu *arc* dengan bobot yang diberikan oleh $[A]_{i,j}$. Jika $[A]_{i,j} = \varepsilon$, maka tidak ada *arc* (j, i) di $\mathcal{G}(A)$ dan $P(j, i; k) = \emptyset$.

Misalkan teorema benar untuk $k > 1$. Selanjutnya untuk kasus $k + 1$, perhatikan bahwa untuk $p \in P(j, i; k + 1)$ diasumsikan ada setidaknya satu *path* di $P(j, i; k + 1)$. *Path* p tersebut dapat dipecah menjadi suatu *subpath* dengan panjang k dari titik j ke titik l , dan suatu *subpath* yang terdiri dari satu *arc* dari titik l ke i , atau dalam bentuk simbol dapat ditulis

$$p = \hat{p} \circ (l, i) \quad \text{dengan} \quad \hat{p} \in P(j, l; k).$$

Bobot maksimal dari setiap *path* di $P(j, i; k + 1)$ dapat diperoleh dari

$$\max_{l \in \underline{n}} ([A]_{i,l} + \max\{|\hat{p}|_w : \hat{p} \in P(j, l; k)\}). \quad (2.1)$$

Sesuai hipotesa induksi, menyatakan bahwa

$$\max\{|\hat{p}|_w : \hat{p} \in P(j, l; k)\} = [A^{\otimes k}]_{l,j}$$

dan ekspresi untuk bobot maksimal suatu *path* dari j ke i dengan panjang $(k + 1)$ pada Persamaan (2.1) menjadi

$$\begin{aligned} \max_{l \in \underline{n}} (a_{i,l} + [A^{\otimes k}]_{l,j}) &= \bigoplus_{l=1}^n a_{i,l} \otimes [A^{\otimes k}]_{l,j} \\ &= [A \otimes A^{\otimes k}]_{i,j} \\ &= [A^{\otimes (k+1)}]_{i,j}. \end{aligned}$$

Selanjutnya, untuk kasus $P(j, i; k + 1) = \emptyset$, yaitu tidak ada *path* dengan panjang $k + 1$ dari j ke i . Jelas, hal tersebut menyebabkan untuk setiap titik l , tidak ada *path* dengan panjang k dari j ke l atau tidak ada *arc* dari l ke i (atau kedua kondisi terjadi). Oleh karena itu, $P(j, i; k + 1) = \emptyset$ menyebabkan paling tidak ada satu nilai dari $a_{i,l}$ dan $[A^{\otimes k}]_{l,j}$ sama dengan ε . Sehingga,

$$[A^{\otimes(k+1)}]_{i,j} = \varepsilon$$

terbukti. □

Dari Teorema 2.1.1 didapat, untuk suatu matriks persegi $A \in \mathbb{R}_\varepsilon^{n \times n}$, matriks A^+ didefinisikan sebagai

$$A^+ \stackrel{\text{def}}{=} \bigoplus_{i=1}^{\infty} A^{\otimes i}. \quad (2.2)$$

Catatan bahwa, elemen $[A^{\otimes k}]_{i,j}$ adalah bobot maksimum dari semua *path* dengan panjang k dari titik j ke titik i . Jadi elemen $[A^+]_{i,j}$ adalah bobot maksimum dari *path-path* dengan panjang sebarang dari titik j ke titik i , sehingga didapat

$$[A^+]_{i,j} = \max\{[A^{\otimes k}]_{i,j} \mid k \geq 1\}.$$

Perhatikan bahwa dalam Persamaan (2.2) matriks pangkat $A^{\otimes i}$, $i = 1, 2, \dots, +\infty$. Berikut ini diberikan suatu teorema mengenai A^+ dengan matriks pangkat $A^{\otimes i}$ berhenti untuk $i = n$ dengan n adalah ukuran dari matriks A yaitu banyaknya baris dan banyaknya kolom dari A .

Teorema 2.1.2 *Misalkan $A \in \mathbb{R}_\varepsilon^{n \times n}$ sedemikian hingga setiap sirkuit di $\mathcal{G}(A)$ mempunyai bobot sirkuit rata-rata kurang atau sama dengan 0. Maka*

$$A^+ = \bigoplus_{i=1}^{\infty} A^{\otimes i} = A \oplus A^{\otimes 2} \oplus \dots \oplus A^{\otimes n} \in \mathbb{R}_\varepsilon^{n \times n}. \quad (2.3)$$

Bukti

Karena A berukuran $n \times n$, maka semua *path* di $\mathcal{G}(A)$ dari i ke j dengan panjang lebih besar dari n merupakan setidaknya satu sirkuit dan suatu *path* dari i ke j dengan panjang setidaknya n . Oleh karena sirkuit di $\mathcal{G}(A)$ mempunyai bobot takpositif, didapat

$$[A^+]_{j,i} \leq \max\{[A^{\otimes i}]_{j,i} \mid i \in \underline{n}\}.$$

Hal ini menyimpulkan bahwa

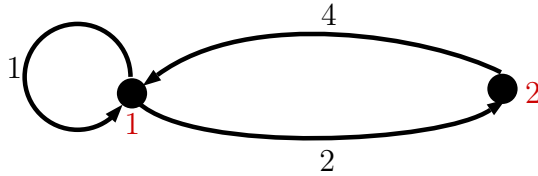
$$A^+ = A \oplus A^{\otimes 2} \oplus \dots \oplus A^{\otimes n}.$$

■

Terdapat dua bentuk graf berdasarkan sifat keterhubungannya. Dua bentuk graf tersebut menjadi bahasan utama dalam bagian ini. Sebelum diberikan definisi dan contoh bentuk-bentuk graf, terlebih dahulu dipahami istilah *reachable* dan *communicate* yang dibutuhkan dalam pendefinisian masing-masing bentuk graf. Untuk $i, j \in \mathcal{N}$, titik i dikatakan *reachable* dari titik j , dinotasikan dengan $j\mathcal{R}i$, jika terdapat suatu *path* dari j ke i . Sedangkan titik i dikatakan *communicate* dengan titik j , dinotasikan dengan $j\mathcal{C}i$, jika dan hanya jika $i = j$ atau titik i *reachable* dari titik j dan titik j *reachable* dari titik i .

Berikut ditunjukkan bahwa relasi $\mathcal{C}$ adalah relasi ekuivalen pada $\mathcal{N}$:

- Relasi $\mathcal{C}$ refleksif sebab untuk setiap $i \in \mathcal{N}$ memenuhi $i = i$ sehingga $i\mathcal{C}i$.
- Relasi $\mathcal{C}$ simetri sebab apabila $i\mathcal{C}j$ berarti titik i *reachable* dari titik j dan titik j *reachable* dari titik i , hal tersebut juga menunjukkan bahwa $j\mathcal{C}i$ untuk $i, j \in \mathcal{N}$.
- Relasi $\mathcal{C}$ transitif sebab untuk $i, j, k \in \mathcal{N}$, apabila $i\mathcal{C}j$ dan $j\mathcal{C}k$ berarti titik i *reachable* dari titik j dan titik j *reachable* dari titik k sehingga titik i *reachable* dari titik k , serta titik k *reachable* dari titik j dan titik j *reachable* dari titik i sehingga titik k *reachable* dari titik i . Karena titik i *reachable* dari titik k dan titik k *reachable* dari titik i maka $i\mathcal{C}k$.



Gambar 2.1: Graf Strongly Connected

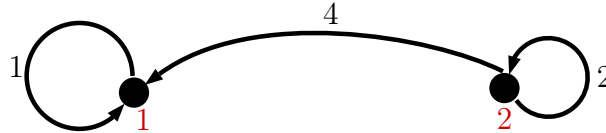
Suatu graf disebut *strongly connected* apabila seluruh titik pada graf tersebut *communicate* satu sama lain, yaitu untuk setiap $i, j \in \mathcal{N}$ memenuhi $i\mathcal{C}j$. Matriks di $\mathbb{R}_{\varepsilon}^{n \times n}$ yang mempunyai representasi graf *strongly connected* disebut sebagai matriks *irreducible* atau matriks **tak-tereduksi**. Lebih lanjut, matriks tak-tereduksi adalah matriks yang tidak dapat dikonstruksi menjadi bentuk matriks segitiga atas. Gambar 2.1 adalah suatu contoh dari graf strongly connected sebab setiap titik di $\mathcal{G}$ *communicate* satu dengan lainnya, yaitu $1\mathcal{C}1$, $1\mathcal{C}2$ dan $2\mathcal{C}1$. Dalam hal ini matriks representasi graf $\mathcal{G}$ dari Gambar 2.1 adalah

$$\begin{aligned}
 A &= \begin{bmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{bmatrix} \\
 &= \begin{bmatrix} 1 & 4 \\ 2 & \varepsilon \end{bmatrix}
 \end{aligned}$$

yang merupakan matriks tak-tereduksi.

Jika ada titik yang tidak *communicate* dengan titik lain dalam suatu graf, maka graf disebut **tidak strongly connected**. Suatu matriks di $\mathbb{R}_{\varepsilon}^{n \times n}$ mempunyai representasi graf tidak *strongly connected* disebut matriks **tereduksi**. Lebih lanjut, matriks tereduksi

adalah matriks yang dapat dikontruksi menjadi bentuk matriks blok segitiga atas, dengan elemen-elemen berupa matriks $\mathcal{E}$ atau matriks tak-tereduksi. Matriks blok segitiga atas tersebut merupakan representasi dari graf tereduksi. Graf tereduksi adalah graf hasil reduksi graf tidak *strongly connected*.



Gambar 2.2: Graf tidak strongly connected

Gambar 2.2 merupakan contoh graf tidak *strongly connected* sebab terdapat titik yang tidak *communicate* dengan titik yang lain, yaitu titik 1 tidak *communicate* dengan titik 2. Hal tersebut terjadi karena titik 2 tidak *reachable* dari titik 1. Adapun matriks representasi dari graf pada Gambar 2.2 adalah

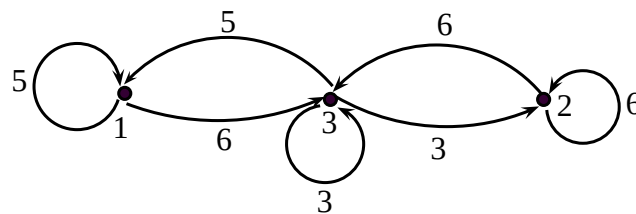
$$\begin{aligned} B &= \begin{bmatrix} b_{1,1} & b_{1,2} \\ b_{2,1} & b_{2,2} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 4 \\ \varepsilon & 2 \end{bmatrix} \end{aligned}$$

yang merupakan matriks tereduksi.

Contoh 2.1.1 Diberikan matriks

$$A = \begin{bmatrix} 5 & \varepsilon & 5 \\ \varepsilon & 6 & 3 \\ 6 & 6 & 3 \end{bmatrix}.$$

Gambar graf dari matriks A ini diberikan dalam Gambar 2.3. Dalam gambar ini ada

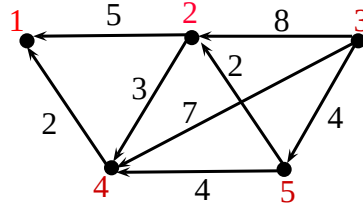


Gambar 2.3: Graf $\mathcal{G}(A)$

lima sirkuit yaitu $(1, 1)$; $(1, 3)$, $(3, 1)$; $(3, 3)$; $(2, 3)$, $(3, 2)$ dan $(2, 2)$. Masing-masing sirkuit mempunyai sirkuit rata-rata $5/1 = 5$, $(6 + 5)/2 = \frac{11}{2}$, $3/1 = 3$, $(6 + 3)/2 = \frac{9}{2}$ dan $6/1 = 6$. Terlihat bahwa sirkuit rata-rata maksimum adalah 6 terjadi pada sirkuit $(2, 2)$. Jadi sirkuit kritis dari graf $\mathcal{G}(A)$ adalah sirkuit $(2, 2)$. Juga tampak bahwa graf $\mathcal{G}(A)$ adalah strongly connected. Jadi matriks A tak-tereduksi. $\square$

Contoh 2.1.2 Diberikan matriks

$$A = \begin{bmatrix} \varepsilon & 5 & \varepsilon & 2 & \varepsilon \\ \varepsilon & \varepsilon & 8 & \varepsilon & 2 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 3 & 7 & \varepsilon & 4 \\ \varepsilon & \varepsilon & 4 & \varepsilon & \varepsilon \end{bmatrix}.$$



Gambar 2.4: Graf $\mathcal{G}(A)$ tanpa sirkuit

Gambar 2.4 adalah graf dari matriks A , terlihat graf $\mathcal{G}(A)$ tidak memuat satupun sirkuit, dengan demikian sirkuit rata-rata maksimum tidak ada. Perhatikan juga bahwa graf $\mathcal{G}(A)$ tidak strongly connected. Jadi matriks A adalah tereduksi. $\square$

Sirkuit rata-rata maksimum dari suatu graf $\mathcal{G}(A)$ adalah suatu bagian penting dari matriks A , sebab sirkuit rata-rata maksimum berkaitan dengan suatu karakteristik dari matriks A . Pada Contoh 2.1.2, graf $\mathcal{G}(A)$ tidak memuat satupun sirkuit. Bagaimanapun hal ini untuk matriks A berukuran yang agak besar tentunya menyelidiki graf $\mathcal{G}(A)$ tidak memuat satupun sirkuit tidaklah mudah. Oleh karena itu akan diberikan suatu sifat yang berkaitan dengan masalah ini. Sifat yang akan dibahas berhubungan dengan matriks pangkat. Oleh karena itu sedikit dibahas ulang mengenai matriks pangkat. Untuk matriks persegi A berukuran $n \times n$, maka

$$A^{\otimes 2} = A \otimes A.$$

Elemen elemen ke- i, j dari $A^{\otimes 2}$ adalah $[A^{\otimes 2}]_{i,j} = \max_k \{a_{i,k} + a_{k,j}\}$ menyatakan bobot maksimum dari semua path dengan panjang 2 dari titik j ke titik i dalam graf $\mathcal{G}(A)$. Secara umum, $[A^{\otimes k}]_{i,j}$ adalah bobot maksimum dari semua path dengan panjang k dari titik j ke titik i dalam graf $\mathcal{G}(A)$.

Teorema 2.1.3 Misalkan matriks A berukuran $n \times n$. Graf $\mathcal{G}(A)$ tidak memuat satupun sirkuit bila dan hanya bila $A^{\otimes k} = \varepsilon(n, n)$, $\forall k \geq n$.

Bukti

Misalkan $\mathcal{G}(A)$ tidak memuat sirkuit. Karena $\mathcal{G}(A)$ mempunyai n titik, maka tidak ada path dengan panjang $k \geq n$. Jadi $A^{\otimes k} = \varepsilon(n, n)$, $\forall k \geq n$. Selanjutnya, misalkan $A^{\otimes k} = \varepsilon(n, n)$, $\forall k \geq n$, yaitu tidak ada path dalam $\mathcal{G}(A)$ dengan panjang $k \geq n$. Karena sirkuit selalu bisa diperluas ke sebarang path yang panjang, hal berakibat bahwa tidak ada sirkuit dalam $\mathcal{G}(A)$.

■

Contoh 2.1.3

Diberikan lagi matriks pada Contoh 2.1.2, maka

$$A^{\otimes 2} = \begin{bmatrix} \varepsilon & 5 & 13 & \varepsilon & 7 \\ \varepsilon & \varepsilon & 6 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 11 & \varepsilon & 5 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{bmatrix}, \quad A^{\otimes 3} = \begin{bmatrix} \varepsilon & \varepsilon & 13 & \varepsilon & 7 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 9 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{bmatrix}$$

dan

$$A^{\otimes 4} = \begin{bmatrix} \varepsilon & \varepsilon & 11 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{bmatrix}, \quad A^{\otimes 5} = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{bmatrix} = \mathbf{\varepsilon}(5, 5).$$

Terlihat bahwa hanya ada 5 pasang titik dengan panjang 2. Misalnya dari titik 3 ke 1 dengan panjang maksimum 13, yaitu $3 \rightarrow 2 \rightarrow 1$. Dan hanya ada 3 pasang titik dengan panjang 3. Misalnya dari titik 3 ke titik 1 dengan panjang maksimum 13 yaitu : $3 \rightarrow 2 \rightarrow 4 \rightarrow 1$. □

Berikut ini diberikan keujudan dari sirkuit rata-rata maksimum untuk matriks persegi tak-tereduksi. Sebelum membahas sifat ini, diberikan notasi $\lambda(A)$ yang menyatakan nilai sirkuit rata-rata maksimum dari suatu matriks persegi A .

Teorema 2.1.4 Bila $A \in \mathbb{R}^{n \times n}$ tak-tereduksi, maka ada $\lambda(A)$ yang diberikan oleh

$$\lambda(A) = \bigoplus_{k=1}^n \frac{\text{tr}(A^{\otimes k})}{k}, \quad \text{dengan} \quad \text{tr}(A) = \bigoplus_{i=1}^n a_{i,i}. \quad (2.4)$$

Bukti

Perhatikan bahwa

$$\text{tr}(A^{\oplus k}) = \bigoplus_{i=1}^n [A^{\otimes k}]_{i,i}$$

menyatakan bobot sirkuit maksimum dari semua sirkuit dengan panjang k dalam $\mathcal{G}(A)$, dengan demikian bila dibagi oleh k merupakan bobot rata-rata maksimum dari sirkuit dengan panjang k . Oleh karena itu untuk $k = 1, 2, \dots, n$ dan bila $\mathcal{C}(A)$ adalah himpunan semua sirkuit elementer dari graf $\mathcal{G}(A)$ didapat

$$\lambda(A) = \max_{p \in \mathcal{C}(A)} \frac{|p|_w}{|p|_l} = \bigoplus_{k=1}^n \frac{\text{tr}(A^{\otimes k})}{k}. \quad \blacksquare$$

Teorema 2.1.4 menunjukkan eksistensi nilai eigen dari matriks tak-tereduksi. Berikut ini, dibuktikan ketunggalan nilai eigen dari matriks tak-tereduksi melalui teorema berikut.

Teorema 2.1.5 Untuk setiap matriks tak-tereduksi $A \in \mathbb{R}_\varepsilon^{n \times n}$ memiliki satu dan hanya satu nilai eigen. Nilai eigen tersebut dinotasikan dengan $\lambda(A)$, merupakan suatu nilai berhingga dan sama dengan sirkuit rata-rata maksimum pada $\mathcal{G}(A)$, yaitu

$$\lambda(A) = \max_{\gamma \in \mathcal{C}(A)} \frac{|\gamma|_w}{|\gamma|_\ell}.$$

Bukti Ambil sebarang sirkuit $\gamma = ((i_1, i_2), (i_2, i_3), \dots, (i_\ell, i_{\ell+1}))$ pada graf $\mathcal{G}(A)$ dengan $|\gamma|_\ell = \ell$ dan $i_{\ell+1} = i_1$. Selanjutnya, karena matriks tak-tereduksi A merupakan representasi dari graf *strongly connected* maka

$$a_{i_{k+1}, i_k} \neq \varepsilon, \quad k \in \underline{\ell}.$$

Misalkan μ adalah suatu nilai eigen berhingga dari matriks A , dan $\mathbf{v}$ adalah suatu vektor eigen yang bersesuaian dengan nilai eigen μ . Karena $A \otimes \mathbf{v} = \mu \otimes \mathbf{v}$, hal tersebut berakibat

$$a_{i_{k+1}, i_k} \otimes v_{i_k} \leq \mu \otimes v_{i_{k+1}}, \quad k \in \underline{\ell}. \quad (2.5)$$

Dalam aljabar konvensional, operasi $\otimes$ dibaca sebagai operasi penjumlahan. Oleh karena itu, untuk $k = \{1, 2, \dots, \ell\}$ Pertidaksamaan (2.5) menjadi

$$\begin{aligned} a_{i_2, i_1} + v_{i_1} &\leq \mu + v_{i_2}, \\ &\vdots \\ a_{i_{\ell+1}, i_\ell} + v_{i_\ell} &\leq \mu + v_{i_{\ell+1}}. \end{aligned}$$

Nilai μ , a_{i_{k+1}, i_k} , dan v_{i_k} untuk $k = \{1, 2, \dots, \ell\}$ adalah elemen-elemen bilangan real. Sehingga apabila masing-masing ruas pada pertidaksamaan-pertidaksamaan di atas dijumlahkan didapat

$$\sum_{k=1}^{\ell} a_{i_{k+1}, i_k} + \sum_{k=1}^{\ell} v_{i_k} \leq \ell \times \mu + \sum_{k=1}^{\ell} v_{i_{k+1}}. \quad (2.6)$$

Karena $i_{\ell+1} = i_1$, maka

$$\sum_{k=1}^{\ell} v_{i_{k+1}} = \sum_{k=1}^{\ell} v_{i_k}.$$

Apabila kedua ruas Pertidaksamaan (2.6) dikurangi dengan $\sum_{k=1}^{\ell} v_{i_k}$ menghasilkan

$$\sum_{k=1}^{\ell} a_{i_{k+1}, i_k} \leq \ell \times \mu. \quad (2.7)$$

Ruas kiri dari Pertidaksamaan (2.7) merupakan bobot dari sirkuit γ dan dapat dinotasikan dengan $|\gamma|_w$, sehingga Pertidaksamaan (2.7) dapat ditulis kembali menjadi

$$|\gamma|_w \leq \ell \times \mu. \quad (2.8)$$

Kedua ruas Pertidaksamaan (2.8) dibagi dengan $|\gamma|_\ell$ menghasilkan sirkuit rata-rata dari γ , yaitu

$$\begin{aligned} \frac{|\gamma|_w}{|\gamma|_\ell} &\leq \frac{\ell \times \mu}{|\gamma|_\ell} \\ &= \frac{\ell \times \mu}{\ell} \\ &= \mu, \end{aligned}$$

yang dapat ditulis kembali sebagai berikut

$$\frac{|\gamma|_w}{|\gamma|_\ell} \leq \mu, \quad \forall \gamma \in \mathcal{C}(A). \quad (2.9)$$

Apabila dihitung maksimum dari pertidaksamaan (2.9) untuk setiap $\gamma \in \mathcal{C}(A)$, diperoleh

$$\max_{\gamma \in \mathcal{C}(A)} \frac{|\gamma|_w}{|\gamma|_\ell} = \mu. \quad (2.10)$$

Dari persamaan (2.10) didapatkan bahwa nilai eigen dapat dicari melalui perhitungan sirkuit rata-rata maksimum dari graf $\mathcal{G}(A)$. Selanjutnya, akan ditunjukkan μ adalah nilai eigen yang tunggal. Misalkan terdapat μ_1 dan μ_2 nilai eigen dari matriks A , artinya

$$A \otimes \mathbf{v} = \mu_1 \otimes \mathbf{v},$$

sehingga

$$\max\{a_{i_{k+1}, i_k} + v_{i_k}\} = \mu_1 + v_{i_{k+1}}, \quad k \in \underline{\ell}, \quad (2.11)$$

dan

$$A \otimes \mathbf{v} = \mu_2 \otimes \mathbf{v},$$

sehingga

$$\max\{a_{i_{k+1}, i_k} + v_{i_k}\} = \mu_2 + v_{i_{k+1}}, \quad k \in \underline{\ell}. \quad (2.12)$$

Apabila ruas kiri Persamaan (2.11) dikurangi dengan ruas kiri Persamaan (2.12), dan ruas kanan Persamaan (2.12) dikurangi dengan ruas kanan Persamaan (2.12) didapatkan

$$0 = \mu_1 - \mu_2.$$

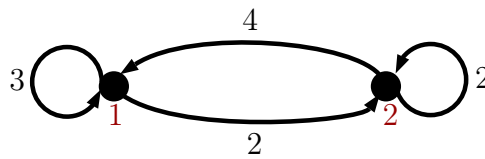
Jadi diperoleh $\mu_1 = \mu_2$ yang menunjukkan nilai eigen dari matriks tak-tereduksi A adalah tunggal.

Berikutnya, akan ditunjukkan bahwa nilai eigen dari matriks tak-tereduksi memiliki nilai berhingga. Misalkan nilai eigen dari matriks tak-tereduksi A adalah $\lambda = \varepsilon$ dan $\mathbf{v}$ adalah vektor eigen yang bersesuaian dengan nilai eigen λ . Maka vektor $\mathbf{v}$ memiliki paling sedikit satu elemen berhingga; misal v_i . Karena matriks tak-tereduksi A representasi dari graf *strongly connected*, maka terdapat baris γ dari matriks A sedemikian hingga $a_{\gamma,i}$ berhingga elemen bilangan real. Selanjutnya, melalui pertidaksamaan berikut

$$\begin{aligned} a_{\gamma,i} \otimes v_i &\leq [A \otimes \mathbf{v}]_{\gamma} \\ &= [\varepsilon \otimes \mathbf{v}]_{\gamma} \\ &= \varepsilon, \end{aligned}$$

didapatkan $a_{\gamma,i} \otimes v_i \leq \varepsilon$. Akan tetapi, ekspresi $a_{\gamma,i} \otimes v_i$ berhingga, sehingga ε tidak dapat menjadi nilai eigen dari matriks tak-tereduksi. Jadi matriks tak-tereduksi memiliki nilai eigen berhingga elemen bilangan real. $\square$

Teorema 2.1.5 membuktikan setiap matriks tak-tereduksi memiliki nilai eigen tunggal berhingga. Nilai eigen tersebut dapat dicari melalui sirkuit rata-rata maksimum dari graf komunikasinya. Meskipun nilai eigen dari matriks tak-tereduksi tunggal, sirkuit rata-rata maksimum dari graf komunikasi belum tentu tunggal. Berikut ini contohnya.



Gambar 2.5: Graf Komunikasi $\mathcal{G}(A)$

Contoh 2.1.4 Misal diberikan graf komunikasi $\mathcal{G}(A)$ seperti pada Gambar 2.5, dengan

$$A = \begin{bmatrix} 3 & 4 \\ 2 & 2 \end{bmatrix}.$$

Karena graf $\mathcal{G}(A)$ adalah graf *strongly connected*, maka A adalah matriks tak-tereduksi. Sehingga matriks A memiliki nilai eigen $\lambda(A)$ tunggal yang dapat dihitung melalui sirkuit

rata-rata maksimumnya, yaitu:

$$\begin{aligned}
 \lambda(A) &= \bigoplus_{k=1}^2 \frac{\text{tr}(A^{\otimes k})}{k} \\
 &= \frac{\text{tr}(A)}{1} \oplus \frac{\text{tr}(A^{\otimes 2})}{2} \\
 &= \frac{3 \oplus 2}{1} \oplus \frac{6 \oplus 6}{2} \\
 &= \frac{3}{1} \oplus \frac{6}{2} \\
 &= 3 \oplus 3 \\
 &= 3.
 \end{aligned}$$

Diperoleh nilai eigen matriks A tunggal, yaitu $\lambda(A) = 3$. Akan tetapi, dari uraian di atas jelas diketahui bahwa sirkuit rata-rata maksimum dari graf komunikasi $\mathcal{G}(A)$ tidak tunggal. Terdapat dua sirkuit rata-rata maksimum dari $\mathcal{G}(A)$, yaitu sirkuit dengan panjang satu dari titik 1 kembali ke titik 1, dan sirkuit dengan panjang dua dari titik 2 kembali ke titik 2 melalui titik 1. □

Dalam Scilab perintah-perintah untuk menentukan sirkuit rata-rata maksimum, lintasan kritis dan menguji kondisi strongly connected dari suatu graf $\mathcal{G}(A)$ adalah sebagai berikut:

```

-->A=[5 -%inf 5;-%inf 6 3;6 6 3]
A =
    5.   -Inf    5.
  -Inf    6.    3.
    6.    6.    3.
// maximum cycle rata-rata dari matriks A (sirkuit rata-rata maksimum)
-->mcm=maxplumcm(A)
mcm =
    6.
// cek graf G(A) strongly connected
-->s = maxplusscg(A)
s =
    T // T menunjukkan true, jadi benar bahwa G(A) strongly connected.
// menentukan lintasan kritis (sirkuit kritis).
-->[1,d,x]=maxplusccir(A)
x =
    2. // lintasan kritis dari titik 2 kembali ke 2
d =
    1. // banyaknya garis dalam lintasan kritis.

```

```

1 =
    6. // maximum cycle rata-rata.
//
// Contoh berikut menunjukkan bahwa sirkuit rata-rata maksimum
// tidak ada, sebab G(A) tidak memuat sirkuit.
-->A = [-%inf    5.  -%inf    2.  -%inf;
        -%inf -%inf    8.  -%inf    2. ;
        -%inf -%inf -%inf -%inf -%inf;
        -%inf    3.    7.  -%inf    4.;
        -%inf -%inf    4.  -%inf -%inf];
-->maxplusmcm(A)
!--error 10000
No any circuit of The Graph G(A)
at line      13 of function maxplusmcm called by :
maxplusmcm(A)
//
// bisa dicek bahwa A pangkat 5 sama dengan matriks nol
//
-->maxpluspwr(A,5)
ans =
    - Inf - Inf - Inf - Inf - Inf
    - Inf - Inf - Inf - Inf - Inf
    - Inf - Inf - Inf - Inf - Inf
    - Inf - Inf - Inf - Inf - Inf
    - Inf - Inf - Inf - Inf - Inf

```

2.2 Nilai karakteristik dan Vektor karakteristik

Pengertian nilai-karakteristik dan vektor-karakteristik yang bersesuaian dari suatu matriks persegi A berukuran $n \times n$ sebagaimana dijumpai dalam aljabar linear biasa juga dijumpai dalam aljabar maxplus, yaitu bila diberikan suatu persamaan:

$$A \otimes \mathbf{x} = \lambda \otimes \mathbf{x}$$

dalam hal ini masing-masing vektor $\mathbf{x} \in \mathbb{R}_\varepsilon^n$ dan $\lambda \in \mathbb{R}$ dinamakan vektor-karakteristik dan nilai-karakteristik dari matriks A dengan vektor $\mathbf{x} \neq (\varepsilon, \varepsilon, \dots, \varepsilon)^\top$. Suatu algoritma untuk memperoleh vektor-karakteristik dan nilai karakteristik dari matriks persegi A bisa ditemui di [8]. Algoritma untuk menentukan nilai-karakteristik dan vektor-karakteristik dari matriks $A \in \mathbb{R}_\varepsilon^{n \times n}$ dilakukan secara berulang dari bentuk persamaan linear

$$\mathbf{x}(k+1) = A \otimes \mathbf{x}(k), \quad k = 0, 1, 2, \dots \quad (2.13)$$

Perilaku periodik dari Persamaan (2.13) baik untuk matriks A yang tak-tereduksi maupun yang tereduksi erat kaitannya dengan apa yang dinamakan *vektor waktu sikel*. Perilaku

periodik dari Persamaan (2.13) berkaitan dengan barisan

$$\{\mathbf{x}(k) \mid k \in \mathbb{N}\},$$

dan untuk keadaan awal $\mathbf{x}(0) = \mathbf{x}_0 \in \mathbb{R}^n$ didapat

$$\mathbf{x}(k) = A^{\otimes k} \otimes \mathbf{x}_0, \quad (2.14)$$

untuk semua $k \geq 0$.

Definisi 2.2.1 Misalkan $\{\mathbf{x}(k) \mid k \in \mathbb{N}\}$ adalah suatu barisan di $\mathbb{R}_\epsilon^n$ dan diasumsikan bahwa untuk semua $j \in \underline{n}$ limit berikut

$$\eta_j \stackrel{\text{def}}{=} \lim_{k \rightarrow \infty} \frac{x_j(k)}{k}$$

ada. Vektor $\boldsymbol{\eta} = [\eta_1, \eta_2, \dots, \eta_n]^\top$ dinamakan vektor waktu-sikel (cycle-time vector) dari barisan $\mathbf{x}(k)$. Bila semua nilai η_j sama, nilai ini dinamakan rata-rata pertumbuhan asimtotik dari barisan $\mathbf{x}(k)$.

Dalam sub-bagian berikutnya ditunjukkan bahwa sekali suatu vektor waktu-sikel ada, maka keberadaannya tidak bergantung pada kondisi awal $\mathbf{x}_0 \in \mathbb{R}^n$. Selanjutnya untuk matriks dalam Persamaan (2.13) yang tereduksi selalu bisa dijadikan suatu bentuk blok matriks segitiga atas. Hal ini dijelaskan sebagai berikut, bila $\mathcal{G} = (\mathcal{N}, \mathcal{D})$ adalah suatu graf tidak *strongly connected*, maka tidak semua titik dari $\mathcal{N}$ *communicate* satu sama lain. Misal diberikan titik $i \in \mathcal{N}$, maka memungkinkan untuk membedakan titik-titik yang *communicate* dengan i dan yang tidak. Sebagaimana telah dibahas dalam Bagian 2.1, relasi $\mathcal{C}$ adalah relasi ekuivalen pada $\mathcal{N}$. Akibatnya, relasi $\mathcal{C}$ dapat mempartisi $\mathcal{N}$ ke dalam kelas ekuivalen yang saling asing. Kelas ekuivalen dari $i \in \mathcal{N}$ adalah himpunan titik-titik dari $\mathcal{N}$ yang *communicate* dengan titik i , dan dinotasikan dengan $[i] \stackrel{\text{def}}{=} \{j \in \mathcal{N} : i\mathcal{C}j\}$ dengan $\mathcal{N} = \bigcup_{i \in \mathcal{N}} [i]$. Karena himpunan $\mathcal{N}$ berhingga, maka $\bigcup_{i \in \mathcal{N}} [i]$ berhingga sehingga

$$N = \bigcup_{i \in \mathcal{N}} [i] = \bigcup_{i \in \underline{q}} [i]. \quad (2.15)$$

Berdasarkan Persamaan (2.15), himpunan $\mathcal{N}$ dipartisi menjadi q bagian yang saling asing, dengan kata lain $\left(\bigcup_{i \in \underline{q}} [i] \right) \cap [j] = \emptyset$ untuk semua $i \neq j$. Kondisi $i\mathcal{R}j$ dapat terjadi untuk beberapa $i \in [i]$ dan $j \in [j]$ dengan $i, j \in \underline{q}$ dan $i \neq j$, namun $j\mathcal{R}i$ tidak dapat terjadi sebab i dan j tidak *communicate*.

Himpunan $\mathcal{N} = \bigcup_{i \in \mathcal{N}} [i]$ juga dapat ditulis $\bigcup_{i \in \underline{q}} \mathcal{N}_i$, dengan $\mathcal{N}_i = \{j \in \mathcal{N} : i\mathcal{C}j\}$. Berdasarkan partisi dari $\mathcal{N}$, maka terdapat subgraf dari $\mathcal{G}$ yaitu $\mathcal{G}_i = (\mathcal{N}_i, \mathcal{D}_i)$, $i \in \underline{q}$ dengan $\mathcal{D}_i \subseteq \mathcal{D}$ adalah himpunan *arc* yang memiliki titik awal dan titik akhir elemen $\mathcal{N}_i$. Jika $\mathcal{D}_i \neq \emptyset$, maka subgraf $\mathcal{G}_i = (\mathcal{N}_i, \mathcal{D}_i)$ adalah suatu subgraf *strongly connected* maksimal.

Dalam graf $\mathcal{G}$ bisa terjadi suatu titik tidak termuat dalam satu atau lebih sirkuit, berarti titik tersebut tidak *communicate* dengan titik lain dan hanya *communicate* dengan dirinya sendiri. Misalkan titik tersebut adalah titik i , maka $[i] = \{i\}$. Karena tidak terdapat *arc* yang menghubungkan i dengan dirinya sendiri, subgraf yang memuat i dinotasikan dengan $([i], \emptyset)$. Lebih lanjut, meskipun subgraf $([i], \emptyset)$ tidak *strongly connected*, $([i], \emptyset)$ akan dianggap sebagai suatu subgraf *strongly connected* maksimal. Oleh karena itu, seluruh subgraf $\mathcal{G}_i = (\mathcal{N}_i, \mathcal{D}_i)$, $i \in \underline{q}$ seperti yang telah dijelaskan sebelumnya merupakan subgraf *strongly connected* maksimal [1].

Selanjutnya, didefinisikan graf tereduksi yang dinotasikan oleh $\tilde{\mathcal{G}} = (\tilde{\mathcal{N}}, \tilde{\mathcal{D}})$, dengan $\tilde{\mathcal{N}} = \{\mathcal{N}_1, \dots, \mathcal{N}_q\}$ dan $(\mathcal{N}_r, \mathcal{N}_s) \in \tilde{\mathcal{D}}$ jika $r \neq s$ dan terdapat *arc* $(k, l) \in \mathcal{D}$ untuk beberapa $k \in \mathcal{N}_r$, $l \in \mathcal{N}_s$, dan $r, s \in \underline{q}$. Oleh karena itu, banyaknya titik pada graf tereduksi sama dengan banyaknya subgraf *strongly connected* maksimal. Graf tereduksi tidak memuat sirkuit, sebab apabila graf tereduksi memuat sirkuit maka dua atau lebih subgraf *strongly connected* maksimal akan terhubung dan membentuk subgraf *strongly connected* maksimal yang lebih besar. Hal tersebut kontradiksi dengan kondisi bahwa subgraf yang ada merupakan subgraf *strongly connected* dan paling maksimal.

Berikutnya, misalkan $A_{i,i}$ merupakan matriks yang diperoleh dengan membatasi A hanya untuk titik-titik pada $\mathcal{N}_i$, untuk setiap $i \in \underline{q}$, maka $[A_{i,i}]_{k,l} = a_{k,l}$ untuk setiap $k, l \in \mathcal{N}_i$. Untuk setiap $i \in \underline{q}$, matriks $A_{i,i}$ merupakan matriks tak-tereduksi atau $A_{i,i} = \varepsilon$. Sehingga setelah dilakukan pelabelan ulang untuk tiap titik pada $\mathcal{G}(A)$, diperoleh matriks blok segitiga atas

$$\begin{bmatrix} A_{1,1} & A_{1,2} & \dots & \dots & A_{1,q} \\ \mathcal{E} & A_{2,2} & \dots & \dots & A_{2,q} \\ \mathcal{E} & \mathcal{E} & A_{3,3} & & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathcal{E} & \mathcal{E} & \dots & \mathcal{E} & A_{q,q} \end{bmatrix}, \quad (2.16)$$

dengan setiap elemen berhingga dari matriks $A_{s,r}$, $1 \leq s < r \leq q$ merupakan bobot *arc* dari suatu titik elemen $\mathcal{N}_r$ ke suatu titik elemen $\mathcal{N}_s$. Dalam hal yang demikian vektor waktu-sikel (*cycle-time vector*) diberikan oleh

$$\boldsymbol{\eta} = \lim_{k \rightarrow \infty} \frac{\mathbf{x}(k)}{k} = \begin{bmatrix} \boldsymbol{\eta}_1 \\ \boldsymbol{\eta}_2 \\ \vdots \\ \boldsymbol{\eta}_q \end{bmatrix},$$

dengan

$$\boldsymbol{\eta}_i = \begin{bmatrix} \eta_i \\ \eta_i \\ \vdots \\ \eta_i \end{bmatrix}$$

dan vektor $\boldsymbol{\eta}_i$ berukuran $q_i \times 1$.

2.2.1 Ketunggalan dari vektor waktu-sikel

Dalam sub-bagian ini akan ditunjukkan bahwa keberadaan $\lim_{t \rightarrow \infty} \mathbf{x}(k)/k$ tidak bergantung pada vektor keadaan awal $\mathbf{x}_0 \in \mathbb{R}^n$. Untuk hal ini terlebih dahulu dikenalkan istilah *norm* yang dinamakan norm- ℓ^∞ . Norm- ℓ^∞ dari suatu vektor $\mathbf{v} \in \mathbb{R}^n$ didefinisikan sebagai maksimum nilai mutlak dari semua komponen $\mathbf{v}$ yang dinotasikan oleh $\|\mathbf{v}\|_\infty$. Jadi untuk setiap $\mathbf{v}$ di $\mathbb{R}^n$,

$$\|\mathbf{v}\|_\infty \stackrel{\text{def}}{=} \max_{i \in \underline{n}} |v_i|,$$

dimana $|\cdot|$ adalah nilai mutlak. Perlu diperhatikan bahwa norm- ℓ^∞ dari suatu vektor di $\mathbb{R}_\varepsilon^n$ bisa bernilai tak-hingga. Hal ini bisa terjadi bila setidaknya satu komponen dari vektor tersebut bernilai ε . Apapun hal ini, bila dipertimbangkan hanya untuk matriks A *reguler* dengan vektor keadaan kondisi awal $\mathbf{x}_0$ berhingga, maka $A \otimes \mathbf{x}_0$ juga vektor yang berhingga. Dengan demikian untuk $k \geq 0$, maka perilaku asimtotik $A^{\otimes k} \otimes \mathbf{x}_0 \in \mathbb{R}^n$ adalah vektor yang berhingga ([1]).

Teorema berikut berperanan penting untuk membuktikan bahwa perilaku asimptotik tidak bergantung pada kondisi awal.

Teorema 2.2.1 Misalkan $A \in \mathbb{R}_\varepsilon^{m \times n}$ adalah matriks reguler, maka

$$\|(A \otimes \mathbf{u}) - (A \otimes \mathbf{v})\|_\infty \leq \|\mathbf{u} - \mathbf{v}\|_\infty,$$

untuk sebarang $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$.

Bukti

Perlu diperhatikan bahwa $A \otimes \mathbf{u}, A \otimes \mathbf{v} \in \mathbb{R}^m$ yaitu keduanya adalah berhingga. Diberikan

$$\alpha = \|(A \otimes \mathbf{u}) - (A \otimes \mathbf{v})\|_\infty,$$

pilih $i_0 \in \underline{m}$ yang memenuhi

$$\alpha = |(A \otimes \mathbf{u}) - (A \otimes \mathbf{v})|_{i_0}.$$

Asumsikan bahwa $\alpha = [(A \otimes \mathbf{u}) - (A \otimes \mathbf{v})]_{i_0} \geq 0$, didapat

$$\alpha = \max_{j \in \underline{n}} (a_{i_0, j} + u_j) - \max_{k \in \underline{n}} (a_{i_0, k} + v_k).$$

Jadi dapat dipilih suatu $j_0 \in \underline{n}$ yang memenuhi

$$\alpha = (a_{i_0, j_0} + u_{j_0}) - \max_{k \in \underline{n}} (a_{i_0, k} + v_k).$$

Sehingga untuk $k = j_0$ didapat

$$\begin{aligned}
 \alpha &= (a_{i_0, j_0} + u_{j_0}) - \max_{k \in \underline{n}} (a_{i_0, k} + v_k) \\
 &\leq (a_{i_0, j_0} + u_{j_0}) - (a_{i_0, j_0} + v_{j_0}) \\
 &= u_{j_0} - v_{j_0} \\
 &\leq \max_{j \in \underline{n}} (u_j - v_j) \\
 &\leq \max_{j \in \underline{n}} |u_j - v_j| \\
 &= \|\mathbf{u} - \mathbf{v}\|_\infty.
 \end{aligned}$$

Jadi bila $\alpha = [(A \otimes \mathbf{u}) - (A \otimes \mathbf{v})]_{i_0} \geq 0$, maka $\alpha \leq \|\mathbf{u} - \mathbf{v}\|_\infty$. Juga, dengan cara sama dapat ditunjukkan bahwa $\alpha = [(A \otimes \mathbf{u}) - (A \otimes \mathbf{v})]_{i_0} \leq 0$, maka $\alpha \leq \|\mathbf{u} - \mathbf{v}\|_\infty$. Dengan demikian lengkap sudah bukti. $\square$

Sifat $\|(A \otimes \mathbf{u}) - (A \otimes \mathbf{v})\|_\infty \leq \|\mathbf{u} - \mathbf{v}\|_\infty$, untuk sebarang $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ dinamakan *nonexpansive* dalam norm- ℓ^∞ dari pemetaan

$$\mathbf{u} \in \mathbb{R}_\varepsilon^n \rightarrow A \otimes \mathbf{u} \in \mathbb{R}_\varepsilon^m.$$

Digunakan secara berulang Teorema 2.2.1 untuk suatu matriks persegi A didapat

$$\|(A^{\otimes k} \otimes \mathbf{u}) - (A^{\otimes k} \otimes \mathbf{v})\|_\infty \leq \|\mathbf{u} - \mathbf{v}\|_\infty, \quad (2.17)$$

untuk sebarang $\mathbf{u}, \mathbf{v} \in \mathbb{R}_\varepsilon^n$ dan semua $k \geq 0$. Hal ini berarti bahwa jarak- ℓ^∞ diantara $A^{\otimes k} \otimes \mathbf{u}$ dengan $A^{\otimes k} \otimes \mathbf{v}$ terbatas oleh $\|\mathbf{u} - \mathbf{v}\|_\infty$. Teorema berikut menunjukkan bahwa ke-*nonexpansive*-an berakibat bahwa bila vektor waktu siklus ada setidaknya untuk satu vektor awal berhingga, maka vektor waktu siklus ada untuk sebarang vektor awal yang berhingga dan tidak bergantung pada pemilihan vektor awal. Notasi $\mathbf{x}(k; \mathbf{x}_0)$ menyatakan ketergantungan $\mathbf{x}(k)$ pada nilai awal, yaitu $\mathbf{x}(k) = A^{\otimes k} \otimes \mathbf{x}_0$.

Teorema 2.2.2 Tinjau relasi berulang $\mathbf{x}(k+1) = A \otimes \mathbf{x}(k)$ untuk $k \geq 0$ dengan $A \in \mathbb{R}_\varepsilon^{n \times n}$ dan $\mathbf{x}(0) = \mathbf{x}_0$ sebagai suatu kondisi awal. Bila $\mathbf{x}_0 \in \mathbb{R}^n$ suatu kondisi awal tertentu yang memenuhi $\lim_{k \rightarrow \infty} \mathbf{x}(k; \mathbf{x}_0)$ ada, maka keberadaan limit tersebut ada untuk sebarang kondisi awal $\mathbf{y}_0 \in \mathbb{R}^n$.

Bukti

Asumsikan bahwa $\mathbf{x} \in \mathbb{R}^n$ memenuhi

$$\lim_{k \rightarrow \infty} \frac{\mathbf{x}(k; \mathbf{x}_0)}{k} = \boldsymbol{\eta} \in \mathbb{R}^n.$$

Untuk sebarang $\mathbf{y}_0 \in \mathbb{R}^n$ dan menggunakan ke-*nonexpansive*-an didapat

$$\begin{aligned} 0 &\leq \left\| \frac{\mathbf{x}(k; \mathbf{x}_0)}{k} - \frac{\mathbf{x}(k; \mathbf{y}_0)}{k} \right\|_{\infty} \\ &\leq \frac{1}{k} \left\| (A^{\otimes k} \otimes \mathbf{x}_0) - (A^{\otimes k} \otimes \mathbf{y}_0) \right\|_{\infty} \\ &\leq \frac{1}{k} \|\mathbf{x}_0 - \mathbf{y}_0\|_{\infty}. \end{aligned}$$

Untuk $k \rightarrow \infty$ didapat

$$\begin{aligned} 0 &\leq \lim_{k \rightarrow \infty} \left\| \frac{\mathbf{x}(k; \mathbf{x}_0)}{k} - \frac{\mathbf{x}(k; \mathbf{y}_0)}{k} \right\|_{\infty} \\ &\leq \lim_{k \rightarrow \infty} \left(\frac{1}{k} \right) \|\mathbf{x}_0 - \mathbf{y}_0\|_{\infty} = 0. \end{aligned}$$

Jadi

$$\lim_{k \rightarrow \infty} \left\| \frac{\mathbf{x}(k; \mathbf{x}_0)}{k} - \frac{\mathbf{x}(k; \mathbf{y}_0)}{k} \right\|_{\infty} = 0.$$

Karena masing-masing $\lim_{k \rightarrow \infty} \left\| \frac{\mathbf{x}(k; \mathbf{x}_0)}{k} \right\|_{\infty}$ dan $\lim_{k \rightarrow \infty} \left\| \frac{\mathbf{x}(k; \mathbf{y}_0)}{k} \right\|_{\infty}$ ada didapat

$$0 = \lim_{k \rightarrow \infty} \left\| \frac{\mathbf{x}(k; \mathbf{x}_0)}{k} - \frac{\mathbf{x}(k; \mathbf{y}_0)}{k} \right\|_{\infty} = \lim_{k \rightarrow \infty} \left\| \frac{\mathbf{x}(k; \mathbf{x}_0)}{k} \right\|_{\infty} - \lim_{k \rightarrow \infty} \left\| \frac{\mathbf{x}(k; \mathbf{y}_0)}{k} \right\|_{\infty}.$$

Jadi

$$\eta = \lim_{k \rightarrow \infty} \left\| \frac{\mathbf{x}(k; \mathbf{x}_0)}{k} \right\|_{\infty} = \lim_{k \rightarrow \infty} \left\| \frac{\mathbf{x}(k; \mathbf{y}_0)}{k} \right\|_{\infty}$$

dan η adalah vektor waktu siklus untuk sebarang keadaan awal $\mathbf{y}_0 \in \mathbb{R}^n$. □

2.2.2 Keujudan vektor waktu siklus untuk matriks tak-tereduksi

Akibat dari Teorema 2.2.2 adalah sekali ditentukan vektor waktu siklus ada, maka keberadaannya tidak bergantung pada kondisi awal. Oleh karena itu dalam sub-bagian ini dikaji keujudan (keberadaan) vektor waktu siklus khusus untuk matriks persegi reguler A yang tak-tereduksi. Faktanya, rata-rata pertumbuhan asimptotik untuk suatu keadaan awal yang khusus yaitu $\mathbf{x}_0 = \mathbf{v} \in \mathbb{R}_{\varepsilon}^n$ adalah vektor-eigen dari matriks A untuk nilai-eigen $\lambda \in \mathbb{R}$. Nilai-eigen λ berdasarkan Teorema 2.1.4 dijamin ada (dan tunggal) diberikan oleh

$$\lambda = \lambda(A) = \bigoplus_{i=1}^n \frac{\text{tr}(A^{\otimes i})}{i}.$$

Sehingga dengan melakukan pengulangan relasi pada $\mathbf{x}(k+1) = A \otimes \mathbf{x}(k)$, $k = 0, 1, 2, \dots$ dan untuk $\mathbf{x}(0) = \mathbf{v} \in \mathbb{R}^n$ didapat

$$\mathbf{x}(k) = A^{\otimes k} \otimes \mathbf{x}(0) = \lambda^{\otimes k} \otimes \mathbf{v},$$

untuk semua $k \geq 0$. Dengan demikian untuk sebarang $j \in \underline{n}$ didapat

$$\lim_{k \rightarrow \infty} \frac{x_j(k)}{k} = \lambda, \quad (2.18)$$

terlihat bahwa rata-rata pertumbuhan asimtotik dari $\mathbf{x}(k)$ tepat sama dengan nilai-eigen dari matriks A . Dari berapa yang telah dibahas disimpulkan dalam teorema berikut.

Teorema 2.2.3 Diberikan relasi berulang $\mathbf{x}(k+1) = A \otimes \mathbf{x}(k)$ untuk $k \geq 0$ dengan $A \in \mathbb{R}_{\varepsilon}^{n \times n}$ adalah matriks tak-tereduksi mempunyai nilai-eigen $\lambda \in \mathbb{R}$. Maka untuk semua $j \in \underline{n}$

$$\lim_{k \rightarrow \infty} \frac{x_j(k; \mathbf{x}_0)}{k} = \lambda$$

untuk sebarang kondisi awal $\mathbf{x}(0) = \mathbf{x}_0 \in \mathbb{R}_{\varepsilon}^n$.

Bukti

Misalkan $\mathbf{v}$ adalah suatu vektor-eigen dari A , selanjutnya awali relasi ekivalen dengan $\mathbf{x}_0 = \mathbf{v}$, didapat

$$\lim_{k \rightarrow \infty} \frac{x_j(k; \mathbf{x}_0)}{k} = \lambda$$

untuk semua $j \in \underline{n}$. Hasil ini sesuai dengan Persamaan 2.18 dan berdasarkan Teorema 2.2.2 sekali rata-rata pertumbuhan asimtotik ada, maka hal ini tidak bergantung pada $\mathbf{x}_0$. Dengan demikian lengkap sudah bukti. $\square$

Teorema 2.2.3 menyatakan keujudan vektor waktu sikel $\boldsymbol{\eta} \in \mathbb{R}^n$ dari suatu matriks tak-tereduksi $A \in \mathbb{R}_{\varepsilon}^{n \times n}$ diberikan oleh $[\boldsymbol{\eta}]_j = \lambda$ untuk $j = 1, 2, \dots, n$ dan $\lambda \in \mathbb{R}$ adalah nilai-eigen dari matriks A . Untuk pembahasan yang lebih general berkaitan dengan keujudan vektor waktu sikel $\boldsymbol{\eta} \in \mathbb{R}^n$ dari sebarang matriks reguler $A \in \mathbb{R}_{\varepsilon}^{n \times n}$ diberikan pada pembahasan **Eigenmode** dalam Sub-bagian 2.5.

Apa yang telah dibahas hanya menentukan nilai-eigen dari suatu matriks persegi A yang reguler dan belum membahas bagaimana menentukan vektor-eigennya. Pada pembahasan berikutnya diberikan beberapa algoritma untuk menghitung nilai-eigen dan vektor-eigen dari suatu matriks persegi yang reguler.

2.3 Beberapa Algoritma Power

Algoritma Power dapat digunakan untuk menghitung nilai eigen dan vektor dari suatu matriks persegi.

Algoritma tersebut memiliki prosedur yang sederhana dan mudah dipahami. Algoritma Power pertama kali dikenalkan oleh Olsder dan diberikan contohnya tetapi pada saat itu belum ada teori pengembangannya ([4]). Pengembangan teori algoritma Power berturut-turut dilakukan oleh Braker ([12]) dan Subiono ([3]). Pada bagian ini diberikan beberapa algoritma Power dan contoh-cotoh pembahasan.

Algoritma 2.3.1 ([4])

1. Ambil sebarang vektor awal $\mathbf{x}(0) \neq \mathbf{e}$.
2. Iterasi Persamaan (2.13) sampai ada bilangan bulat p, q dimana $p > q \geq 0$ dan bilangan riil c yang memenuhi $\mathbf{x}(p) = c \otimes \mathbf{x}(q)$.
3. Definisikan sebagai nilai-eigen $\lambda = \frac{c}{p-q}$.
4. Definisikan sebagai (calon) vektor-eigen

$$\mathbf{v} = \frac{1}{p-q} \sum_{i=1}^{p-q} \mathbf{x}(q+i-1).$$

Contoh 2.3.1 Misalkan dalam sistem (2.13) matriks A adalah

$$A = \begin{bmatrix} 3 & 5 \\ 3 & 2 \end{bmatrix} \text{ dan vektor keadaan awal } \mathbf{x}(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Iterasi Persamaan (2.13) didapat

$$\begin{array}{ccc} \mathbf{x}(0), & \mathbf{x}(1), & \mathbf{x}(2) = 8 \otimes \mathbf{x}(0) \quad \dots \\ \downarrow & \downarrow & \downarrow \\ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, & \begin{bmatrix} 5 \\ 3 \end{bmatrix}, & \begin{bmatrix} 8 \\ 8 \end{bmatrix} = 8 \otimes \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \dots \end{array}$$

Dengan demikian dalam Algoritma Power 2.3.1 didapat $p = 2, q = 0$ dan $c = 8$. Jadi sebagai nilai-eigen adalah $\lambda = \frac{c}{p-q} = \frac{8}{2} = 4$. Vektor $\mathbf{v}$ dari Algoritma 2.3.1 diberikan oleh

$$\mathbf{v} = \frac{1}{2}(\mathbf{x}(0) + \mathbf{x}(1)) = \frac{1}{2} \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 5 \\ 3 \end{bmatrix} \right) = \begin{bmatrix} 2\frac{1}{2} \\ 1\frac{1}{2} \end{bmatrix}.$$

Selanjutnya diselidiki apakah $A \otimes \mathbf{v} = \lambda \otimes \mathbf{v}$.

$$A \otimes \mathbf{v} = \begin{bmatrix} 3 & 5 \\ 3 & 2 \end{bmatrix} \otimes \begin{bmatrix} 2\frac{1}{2} \\ 1\frac{1}{2} \end{bmatrix} = \begin{bmatrix} 6\frac{1}{2} \\ 5\frac{1}{2} \end{bmatrix} = 4 \otimes \begin{bmatrix} 2\frac{1}{2} \\ 1\frac{1}{2} \end{bmatrix} = \lambda \otimes \mathbf{v}.$$

□

Berikutnya diberikan dua Algoritma Power yang lainnya dan suatu contoh dimana Aloritma Power 2.3.1 gagal memberikan hasil vektor eigen, sedangkan dua Algoritma Power yang lainnya berhasil menentukan vektor eigen.

Algoritma 2.3.2 ([12])

1. Gunakan Algoritma 2.3.1 untuk menyelidiki bahwa $A \otimes \mathbf{v} = \lambda \otimes \mathbf{v}$.
2. Bila $A \otimes \mathbf{v} = \lambda \otimes \mathbf{v}$, maka $\mathbf{v}$ adalah suatu vektor-eigen dari sistem (2.13) untuk nilai-eigen λ , algoritma berhenti. Bila tidak, definisikan vektor baru $\bar{\mathbf{v}}$ sebagai berikut

$$[\bar{\mathbf{v}}]_i = \begin{cases} [\mathbf{v}]_i & \text{bila } [A \otimes \mathbf{v}]_i = \lambda \otimes [\mathbf{v}]_i, \\ \varepsilon & \text{untuk yang lain.} \end{cases}$$

3. Iterasi lagi Persamaan (2.13) dengan $\mathbf{x}(0) = \bar{\mathbf{v}}$ sampai ada beberapa bilangan bulat $r \geq 0$ yang memenuhi $\mathbf{x}(r+1) = \lambda \otimes \mathbf{x}(r)$. Maka $\mathbf{x}(r)$ adalah suatu vektor-eigen dari sistem Persamaan (2.13) untuk nilai-eigen λ .

Sebelum diberikan suatu contoh penggunaan Algoritma Power 2.3.2, diberikan suatu teorema penunjang untuk satu Algoritma Power yang lainnya.

Teorema 2.3.1 ([8]) Bila untuk sebarang keadaan awal $\mathbf{x}(0) \neq \varepsilon$ sistem Persamaan (2.13) memenuhi $\mathbf{x}(p) = c \otimes \mathbf{x}(q)$ untuk beberapa bilangan bulat p dan q dengan $p > q \geq 0$ dan beberapa bilangan real c dan $\mathbf{x}(q) \in \mathbb{R}^n$, maka

$$\lim_{k \rightarrow \infty} \frac{\mathbf{x}(k)}{k} = \begin{bmatrix} \lambda \\ \lambda \\ \vdots \\ \lambda \end{bmatrix},$$

dengan $\lambda = \frac{c}{p-q}$. Selanjutnya λ adalah suatu nilai karakteristik dari matriks A dengan vektor karakteristik diberikan oleh

$$\mathbf{v} = \bigoplus_{i=1}^{p-q} \left(\lambda^{\otimes (p-q-i)} \otimes \mathbf{x}(q+i-1) \right).$$

Bukti : Misalkan $l = p - q$, didapat

$$\begin{aligned} \lim_{k \rightarrow \infty} \frac{\mathbf{x}(k)}{k} &= \lim_{i \rightarrow \infty} \frac{\mathbf{x}(q+il)}{q+il} = \lim_{i \rightarrow \infty} \frac{c^{\otimes i} \otimes \mathbf{x}(q)}{q+il} \\ &= \left(\lim_{i \rightarrow \infty} \frac{c^{\otimes i}}{q+il} \right) \otimes \left(\lim_{i \rightarrow \infty} \frac{\mathbf{x}(q)}{q+il} \right) \\ &= \left(\lim_{i \rightarrow \infty} \frac{ic}{q+il} \right) \otimes \left(\lim_{i \rightarrow \infty} \frac{\mathbf{x}(q)}{q+il} \right) \\ &= \frac{c}{l} \otimes \mathbf{0} = \frac{c}{p-q} \otimes \mathbf{0} \quad (\text{sebab } \mathbf{x}(q) \in \mathbb{R}^n), \end{aligned}$$

dengan vektor

$$\mathbf{0} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Jadi bila $\lambda = \frac{c}{p-q}$, maka vektor waktu siklus adalah

$$\lim_{k \rightarrow \infty} \frac{\mathbf{x}(k)}{k} = \begin{bmatrix} \lambda \\ \lambda \\ \vdots \\ \lambda \end{bmatrix}.$$

Selanjutnya bila

$$\mathbf{v} = \bigoplus_{i=1}^{p-q} \left(\lambda^{\otimes(p-q-i)} \otimes \mathbf{x}(q+i-1) \right),$$

maka

$$\begin{aligned} A \otimes \mathbf{v} &= A \otimes \left(\bigoplus_{i=1}^{p-q} \left(\lambda^{\otimes(p-q-i)} \otimes \mathbf{x}(q+i-1) \right) \right) \\ &= \bigoplus_{i=1}^{p-q} A \otimes \left(\lambda^{\otimes(p-q-i)} \otimes \mathbf{x}(q+i-1) \right) \\ &= \bigoplus_{i=1}^{p-q} \lambda^{\otimes(p-q-i)} \otimes (A \otimes \mathbf{x}(q+i-1)) \\ &= \bigoplus_{i=1}^{p-q} \lambda^{\otimes(p-q-i)} \otimes \mathbf{x}(q+i) \\ &= \bigoplus_{j=2}^{p-q+1} \lambda^{\otimes(p-q-j+1)} \otimes \mathbf{x}(q+j-1) \\ &= \lambda \otimes \left(\bigoplus_{j=2}^{p-q+1} \lambda^{\otimes(p-q-j)} \otimes \mathbf{x}(q+j-1) \right) \\ &= \lambda \otimes \left(\bigoplus_{j=1}^{p-q} \lambda^{\otimes(p-q-j)} \otimes \mathbf{x}(q+j-1) \right) = \lambda \otimes \mathbf{v}. \end{aligned}$$

Persamaan yang terakhir diperoleh dari

$$\mathbf{x}(p) = \lambda^{\otimes(p-q)} \otimes \mathbf{x}(q),$$

yang berakibat bahwa

$$\lambda^{\otimes -1} \otimes \mathbf{x}(p) = \lambda^{\otimes (p-q-1)} \otimes \mathbf{x}(q).$$

□

Dari hasil Teorema (2.3.1), Algoritma Power berikut dapat digunakan untuk menentukan nilai karakteristik sekaligus vektor karakteristik dari suatu matriks persegi A .

Algoritma 2.3.3 ([8])

1. Mulai dari sebarang vektor awal $\mathbf{x}(0) \neq \boldsymbol{\varepsilon}$.
2. Iterasi persamaan (2.13) sampai ada bilangan bulat $p > q \geq 0$ dan bilangan real c sehingga suatu perilaku periodik terjadi, yaitu $\mathbf{x}(p) = c \otimes \mathbf{x}(q)$.
3. Hitung nilai-karakteristik $\lambda = \frac{c}{p-q}$.
4. Hitung vektor-karakteristik $\mathbf{v} = \bigoplus_{i=1}^{p-q} \left(\lambda^{\otimes (p-q-i)} \otimes \mathbf{x}(q+i-1) \right)$.

Contoh berikut menunjukkan bahwa Algoritma Power 2.3.1 gagal menentukan vektor eigen dari suatu matriks persegi, sedangkan Algoritma Power 2.3.2 dan Algoritma Power 2.3.3 berhasil menentukan vektor eigen matriks tersebut.

Contoh 2.3.2 Misalkan dalam sistem (2.13) matriks A adalah

$$A = \begin{bmatrix} \varepsilon & 3 & \varepsilon & 1 \\ 2 & \varepsilon & 1 & \varepsilon \\ 1 & 2 & 2 & \varepsilon \\ \varepsilon & \varepsilon & 1 & \varepsilon \end{bmatrix} \quad \text{dan vektor keadaan awal } \mathbf{x}(0) = \begin{bmatrix} 0 \\ \varepsilon \\ \varepsilon \\ \varepsilon \end{bmatrix}.$$

Iterasi Persamaan (2.13) didapat

$$\begin{array}{cccccc} \mathbf{x}(0) & \mathbf{x}(1) & \mathbf{x}(2) & \mathbf{x}(3) & \mathbf{x}(4) = 5 \otimes \mathbf{x}(2) & \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \\ \begin{bmatrix} 0 \\ \varepsilon \\ \varepsilon \\ \varepsilon \end{bmatrix} & \begin{bmatrix} \varepsilon \\ 2 \\ 1 \\ \varepsilon \end{bmatrix} & \begin{bmatrix} 5 \\ 2 \\ 4 \\ 2 \end{bmatrix} & \begin{bmatrix} 5 \\ 7 \\ 6 \\ 5 \end{bmatrix} & \begin{bmatrix} 10 \\ 7 \\ 9 \\ 7 \end{bmatrix} & = 5 \otimes \begin{bmatrix} 5 \\ 2 \\ 4 \\ 2 \end{bmatrix} \end{array}.$$

Dengan demikian dalam tiga Algoritma Power didapat $p = 4, q = 2$ dan $c = 5$. Jadi sebagai nilai-eigen adalah $\lambda = \frac{c}{p-q} = \frac{5}{2} = 2\frac{1}{2}$. Vektor $\mathbf{v}$ dari Algoritma 2.3.1 diberikan oleh

$$\mathbf{v} = \frac{1}{2}(\mathbf{x}(2) + \mathbf{x}(3)) = \frac{1}{2} \left(\begin{bmatrix} 5 \\ 2 \\ 4 \\ 2 \end{bmatrix} + \begin{bmatrix} 5 \\ 7 \\ 6 \\ 5 \end{bmatrix} \right) = \begin{bmatrix} 5 \\ 4\frac{1}{2} \\ 5 \\ 3\frac{1}{2} \end{bmatrix}.$$

Selanjutnya diselidiki apakah $A \otimes \mathbf{v} = \lambda \otimes \mathbf{v}$.

$$A \otimes \mathbf{v} = \begin{bmatrix} \varepsilon & 3 & \varepsilon & 1 \\ 2 & \varepsilon & 1 & \varepsilon \\ 1 & 2 & 2 & \varepsilon \\ \varepsilon & \varepsilon & 1 & \varepsilon \end{bmatrix} \otimes \begin{bmatrix} 5 \\ 4\frac{1}{2} \\ 5 \\ 3\frac{1}{2} \end{bmatrix} = \begin{bmatrix} 7\frac{1}{2} \\ 7 \\ 7 \\ 6 \end{bmatrix} \neq \begin{bmatrix} 7\frac{1}{2} \\ 7 \\ 7\frac{1}{2} \\ 6 \end{bmatrix} = 2\frac{1}{2} \otimes \begin{bmatrix} 5 \\ 4\frac{1}{2} \\ 5 \\ 3\frac{1}{2} \end{bmatrix} = \lambda \otimes \mathbf{v}.$$

Jadi vektor $\mathbf{v}$ yang dihasilkan oleh Algoritma 2.3.1 bukan suatu vektor-eigen dari matriks A untuk nilai-eigen $\lambda = 2\frac{1}{2}$.

Sekarang vektor $\mathbf{v}$ dihitung menggunakan Algoritma 2.3.2, hal ini menghasilkan vektor

$$\bar{\mathbf{v}} = \begin{bmatrix} 5 \\ 4\frac{1}{2} \\ \varepsilon \\ 3\frac{1}{2} \end{bmatrix}.$$

Dalam Persamaan (2.13) lakukan iterasi ulang dengan vektor awal $\mathbf{x}(0) = \bar{\mathbf{v}}$ sampai ada beberapa bilangan bulat $r \geq 0$ yang memenuhi $\mathbf{x}(r+1) = \lambda \otimes \mathbf{x}(r)$, didapat

$$\begin{array}{ccccccc} \mathbf{x}(0) & \mathbf{x}(1) & \mathbf{x}(2) & \mathbf{x}(3) = 2\frac{1}{2} \otimes \mathbf{x}(2) \\ \downarrow & \downarrow & \downarrow & \downarrow \\ \begin{bmatrix} 5 \\ 4\frac{1}{2} \\ \varepsilon \\ 3\frac{1}{2} \end{bmatrix} & \begin{bmatrix} 7\frac{1}{2} \\ 7 \\ 6\frac{1}{2} \\ \varepsilon \end{bmatrix} & \begin{bmatrix} 10 \\ 9\frac{1}{2} \\ 9 \\ 7\frac{1}{2} \end{bmatrix} & \begin{bmatrix} 12\frac{1}{2} \\ 12 \\ 11\frac{1}{2} \\ 10 \end{bmatrix} = 2\frac{1}{2} \otimes \begin{bmatrix} 10 \\ 9\frac{1}{2} \\ 9 \\ 7\frac{1}{2} \end{bmatrix} \end{array}.$$

Terlihat bahwa $r = 2$ dan $\mathbf{x}(2)$ adalah vektor-eigen dari A untuk nilai-eigen $\lambda = 2\frac{1}{2}$. Selanjutnya dari hasil iterasi awal yang telah dilakukan dalam Persamaan (2.13) digunakan Algoritma 2.3.3 didapat vektor $\mathbf{v}$ yang diberikan oleh

$$\mathbf{v} = (\lambda \otimes \mathbf{x}(2)) \oplus \mathbf{x}(3) = \begin{bmatrix} 7\frac{1}{2} \\ 4\frac{1}{2} \\ 6\frac{1}{2} \\ 4\frac{1}{2} \end{bmatrix} \oplus \begin{bmatrix} 5 \\ 7 \\ 6 \\ 5 \end{bmatrix} = \begin{bmatrix} 7\frac{1}{2} \\ 7 \\ 6\frac{1}{2} \\ 5 \end{bmatrix}$$

Dapat diselidiki bahwa $\mathbf{v}$ adalah vektor eigen dari A untuk nilai-eigen $\lambda = 2\frac{1}{2}$ sebagai berikut:

$$A \otimes \mathbf{v} = \begin{bmatrix} \varepsilon & 3 & \varepsilon & 1 \\ 2 & \varepsilon & 1 & \varepsilon \\ 1 & 2 & 2 & \varepsilon \\ \varepsilon & \varepsilon & 1 & \varepsilon \end{bmatrix} \otimes \begin{bmatrix} 7\frac{1}{2} \\ 7 \\ 6\frac{1}{2} \\ 5 \end{bmatrix} = \begin{bmatrix} 10 \\ 9\frac{1}{2} \\ 9 \\ 7\frac{1}{2} \end{bmatrix} = 2\frac{1}{2} \otimes \begin{bmatrix} 7\frac{1}{2} \\ 7 \\ 6\frac{1}{2} \\ 5 \end{bmatrix} = \lambda \otimes \mathbf{v}.$$

□

Contoh-contoh berikut untuk menentukan nilai eigen dan vektor eigen dari matriks persegi digunakan Algoritma Power 2.3.3

Contoh 2.3.3 Diberikan matriks tereduksi

$$A = \begin{bmatrix} A_{1,1} & A_{1,2} \\ \boldsymbol{\varepsilon} & A_{2,2} \end{bmatrix}$$

dengan $A_{1,1} = 2$, $A_{1,2} = [2 \ \varepsilon]$ dan

$$A_{2,2} = \begin{bmatrix} 1 & 4 \\ 2 & 2 \end{bmatrix}, \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon \\ \varepsilon \end{bmatrix}.$$

Jelas bahwa matriks $A_{1,1}$ dan $A_{2,2}$ matriks taktereduksi. Dengan demikian matriks A diberikan oleh

$$A = \begin{bmatrix} 2 & 2 & \varepsilon \\ \varepsilon & 1 & 4 \\ \varepsilon & 2 & 2 \end{bmatrix}.$$

Untuk keadaan awal

$$\mathbf{x}(0) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix},$$

didapat evolusi keadaan

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 2 \\ 4 \\ 2 \end{bmatrix}, \begin{bmatrix} 6 \\ 6 \\ 6 \end{bmatrix}, \begin{bmatrix} 8 \\ 10 \\ 8 \end{bmatrix}, \begin{bmatrix} 12 \\ 12 \\ 12 \end{bmatrix}, \dots$$

Terlihat bahwa $\mathbf{x}(2) = 6 \otimes \mathbf{x}(0)$, dalam hal ini $q = 0$, $p = 2$ dan $c = 6$. Jadi nilai karakteristik dari matriks A diberikan oleh

$$\lambda = \frac{c}{p - q} = \frac{6}{2 - 0} = 3$$

dan vektor karakteristiknya adalah

$$\begin{aligned}
 \mathbf{v} &= \bigoplus_{i=1}^{2-0} \lambda^{\otimes(2-0-i)} \otimes \mathbf{x}(0+i-1) \\
 &= \bigoplus_{i=1}^2 3^{\otimes(2-i)} \otimes \mathbf{x}(i-1) \\
 &= (3^{\otimes 1} \otimes \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}) \oplus (3^{\otimes 0} \otimes \begin{bmatrix} 2 \\ 4 \\ 2 \end{bmatrix}) \\
 &= \begin{bmatrix} 3 \\ 3 \\ 3 \end{bmatrix} \oplus \begin{bmatrix} 2 \\ 4 \\ 2 \end{bmatrix} \\
 &= \begin{bmatrix} 3 \\ 4 \\ 3 \end{bmatrix}.
 \end{aligned}$$

Cek hasil yang didapat

$$\begin{bmatrix} 2 & 2 & \varepsilon \\ \varepsilon & 1 & 4 \\ \varepsilon & 2 & 2 \end{bmatrix} \otimes \begin{bmatrix} 3 \\ 4 \\ 3 \end{bmatrix} = \begin{bmatrix} 6 \\ 7 \\ 6 \end{bmatrix} = 3 \otimes \begin{bmatrix} 3 \\ 4 \\ 3 \end{bmatrix}.$$

□

Contoh 2.3.4 Diberikan matriks

$$A = \begin{bmatrix} 5 & \varepsilon & 5 \\ \varepsilon & 6 & 3 \\ 6 & 6 & 3 \end{bmatrix}, \text{ untuk } \mathbf{x}(0) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \text{ didapat } \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 5 \\ 6 \\ 6 \end{bmatrix}, \begin{bmatrix} 11 \\ 12 \\ 12 \end{bmatrix}, \dots$$

Terlihat $q = 1$ dan $p = 2$, sebab

$$\mathbf{x}(2) = 6 \otimes \mathbf{x}(1), \text{ yaitu } \begin{bmatrix} 11 \\ 12 \\ 12 \end{bmatrix} = 6 \otimes \begin{bmatrix} 5 \\ 6 \\ 6 \end{bmatrix} \text{ dan } \lambda = \frac{6}{2-1} = 6.$$

Sesuai langkah 4 dalam Algoritma 2.3.3 didapat vektor-karakteristik $\mathbf{v} = \mathbf{x}(1)$.

□

Misalkan $\mathcal{C}(A)$ adalah himpunan semua sirkuit elementer dari graf $\mathcal{G}(A)$ dan

$$\lambda = \max_{p \in \mathcal{C}(A)} \frac{|p|_w}{|p|_l} \quad (2.19)$$

menyatakan bobot maksimum dari sirkuit rata-rata. Perhatikan lagi matriks A^+ yang didefinisikan oleh persamaan (2.2), biasanya A^+ adalah divergen, untuk kasus ini adalah memungkinkan untuk mempertimbangkan perilaku asimtotik dengan menggunakan matriks A_λ^+ . Bila λ sebagai mana diberikan dalam (2.19), maka matriks A_λ didefinisikan sebagai

$$A_\lambda \stackrel{\text{def}}{=} \lambda^{\otimes -1} \otimes A. \quad (2.20)$$

Dalam hal ini matriks A_λ merujuk sebagai matriks *ternormalkan*. Disini jelas bahwa bobot maksimu sirkuit rata-rata dari $\mathcal{G}(A_\lambda)$ adalah nol. Sehingga berdasarkan Teorema 2.3 didapat

$$A_\lambda^+ = \bigoplus_{i=1}^n A_\lambda^{\otimes i} = A_\lambda \oplus A_\lambda^{\otimes 2} \oplus \dots \oplus A_\lambda^{\otimes n}. \quad (2.21)$$

Selanjutnya graf lintasan kritis dari matriks A dinotasikan oleh $\mathcal{G}^c(A)$. Dalam hal ini graf $\mathcal{G}^c(A)$ tepat sama dengan graf $\mathcal{G}^c(A_\lambda)$ kecuali untuk bobotnya, sehingga untuk semua titik η di graf $\mathcal{G}^c(A_\lambda)$, didapat

$$[A_\lambda^+]_{\eta,\eta} = 0 \quad (2.22)$$

sebab setiap titik di graf kritikal termuat didalam suatu sirkuit dan setiap sirkuit dari graf kritikal mempunyai bobot sama dengan nol. Selanjutnya didefinisikan

$$A_\lambda^* \stackrel{\text{def}}{=} E \oplus A_\lambda^+ = \bigoplus_{i \geq 0} A_\lambda^{\otimes i}. \quad (2.23)$$

Sehingga didapat

$$A_\lambda^+ = A_\lambda \otimes (E \oplus A_\lambda^+) = A_\lambda \otimes A_\lambda^*. \quad (2.24)$$

Misalkan $[B]_{\bullet,k}$ kolom ke- k dari matriks B , sehingga dari Persamaan (2.23) didapat

$$[A_\lambda^*]_{\bullet,\eta} = [E \oplus A_\lambda^+]_{\bullet,\eta}. \quad (2.25)$$

Dari Persamaan (2.25), elemen ke- i dari vektor $[A_\lambda^*]_{\bullet,\eta}$ memenuhi

$$[A_\lambda^*]_{i,\eta} = [E \oplus A_\lambda^+]_{i,\eta} = \begin{cases} \varepsilon \oplus [A_\lambda^+]_{i,\eta} & \text{untuk } i \neq \eta, \\ 0 \oplus [A_\lambda^+]_{i,\eta} & \text{untuk } i = \eta. \end{cases}$$

Dari Persamaan (2.22) dan untuk η adalah titik di $\mathcal{G}^c(A)$, didapat

$$[A_\lambda^+]_{\bullet,\eta} = [A_\lambda^*]_{\bullet,\eta}.$$

Sehingga dengan menggunakan Persamaan (2.24) didapat

$$[A_\lambda \otimes A_\lambda^*]_{\bullet,\eta} = [A_\lambda^*]_{\bullet,\eta}$$

Hal ini memberikan

$$A_\lambda \otimes [A_\lambda^*]_{\bullet,\eta} = [A_\lambda^*]_{\bullet,\eta}$$

atau ekivalen dengan

$$A \otimes [A_\lambda^*]_{\bullet, \eta} = \lambda \otimes [A_\lambda^*]_{\bullet, \eta}.$$

Hal ini menunjukkan bahwa nilai-karakteristik dari matriks A adalah λ dan kolom ke- η dari matriks A_λ^* , merupakan vektor karakteristik dari A untuk semua titik η di graf $\mathcal{G}^c(A)$. Hal ini diberikan dalam teorema berikut.

Teorema 2.3.2 *Jika graf komunikasi $\mathcal{G}(A)$ dari matriks $A \in \mathbb{R}_\varepsilon^{n \times n}$ memiliki sirkuit rata-rata maksimum berhingga λ , maka skalar λ adalah suatu nilai eigen dari A , dan kolom $[A_\lambda^*]_{\bullet, \eta}$ adalah suatu vektor eigen dari A yang bersesuaian dengan λ , untuk setiap titik $\eta \in \mathcal{N}^c(A)$.*

Bukti Diberikan λ sesuai Persamaan (2.19) dengan nilai berhingga, dan matriks ternormalkan A_λ dengan elemen-elemen yang didefinisikan sebagai berikut:

$$[A_\lambda]_{i,j} \stackrel{\text{def}}{=} a_{i,j} - \lambda.$$

Hal ini berakibat sirkuit rata-rata maksimum dari $\mathcal{G}(A_\lambda)$ sama dengan 0, yang memungkinkan berbeda dengan sirkuit rata-rata maksimum dari $\mathcal{G}(A)$. Meskipun demikian, sirkuit kritis dari $\mathcal{G}(A_\lambda)$ tetap sama dengan sirkuit kritis dari $\mathcal{G}(A)$. Berikutnya, dengan mencermati kembali Teorema 3.3.2 dapat didefinisikan

$$A_\lambda^+ \stackrel{\text{def}}{=} \bigoplus_{k=1}^n A_\lambda^{\otimes k}.$$

Sehingga untuk setiap $\eta \in \mathcal{N}^c(A)$,

$$[A_\lambda^+]_{\eta, \eta} = 0. \quad (2.26)$$

Jika didefinisikan

$$A_\lambda^* \stackrel{\text{def}}{=} E \oplus A_\lambda^+,$$

maka kolom ke- η dari matriks A_λ^* adalah

$$[A_\lambda^*]_{\bullet, \eta} = [E \oplus A_\lambda^+]_{\bullet, \eta}, \quad (2.27)$$

dengan

$$[A_\lambda^*]_{i, \eta} = [E \oplus A_\lambda^+]_{i, \eta} = \begin{cases} e \oplus [A_\lambda^+]_{i, \eta} & \text{untuk } i = \eta, \\ \varepsilon \oplus [A_\lambda^+]_{i, \eta} & \text{untuk } i \neq \eta. \end{cases}$$

Persamaan (2.26) dan (2.27) menyebabkan untuk setiap $\eta \in \mathcal{N}^c(A)$,

$$[A_\lambda^+]_{\bullet, \eta} = [A_\lambda^*]_{\bullet, \eta}. \quad (2.28)$$

Selanjutnya, melalui uraian berikut didapatkan

$$\begin{aligned}
 A_\lambda^+ &= A_\lambda \oplus A_\lambda^+ \\
 &= (A_\lambda \otimes E) \oplus (A_\lambda \otimes A_\lambda^+) \\
 &= A_\lambda \otimes (E \oplus A_\lambda^+) \\
 &= A_\lambda \otimes A_\lambda^*,
 \end{aligned}$$

Sehingga Persamaan (2.28) dapat ditulis kembali menjadi

$$\begin{aligned}
 [A_\lambda \otimes A_\lambda^*]_{.,\eta} &= [A_\lambda^*]_{.,\eta} \\
 A_\lambda \otimes [A_\lambda^*]_{.,\eta} &= [A_\lambda^*]_{.,\eta} \\
 -\lambda \otimes A \otimes [A_\lambda^*]_{.,\eta} &= [A_\lambda^*]_{.,\eta}.
 \end{aligned} \tag{2.29}$$

Karena λ berhingga, maka kedua ruas Persamaan (2.29) dapat dioperasikan $\otimes$ dengan λ sehingga didapatkan

$$\begin{aligned}
 \lambda \otimes (-\lambda) \otimes A \otimes [A_\lambda^*]_{.,\eta} &= \lambda \otimes [A_\lambda^*]_{.,\eta} \\
 A \otimes [A_\lambda^*]_{.,\eta} &= \lambda \otimes [A_\lambda^*]_{.,\eta}.
 \end{aligned}$$

Jadi, diperoleh λ adalah nilai eigen dari matriks A dan $[A_\lambda^*]_{.,\eta}$ untuk setiap $\eta \in \mathcal{N}^c(A)$ adalah vektor eigen yang bersesuaian dengan nilai eigen λ . $\square$

Teorema 2.3.2 berlaku untuk graf komunikasi $\mathcal{G}(A)$ dari matriks $A \in \mathbb{R}_\varepsilon^{n \times n}$ strongly conneted atau tidak strongly conneted asalkan graf $\mathcal{G}(A)$ mempunyai sirkuit rata-rata maksimum berhingga yaitu λ . Tetapi dari Teorema 2.1.4 dijamin bahwa nilai λ ada (exist) bila matriks A tak-tereduksi atau ekuivalen dengan graf $\mathcal{G}(A)$ adalah strongly connected. Dalam hal nilai lambda diberikan oleh

$$\lambda = \bigoplus_{k=1}^n \frac{\text{tr}(A^{\otimes k})}{k}, \text{ dengan } \text{tr}(A) = \bigoplus_{i=1}^n a_{i,i}.$$

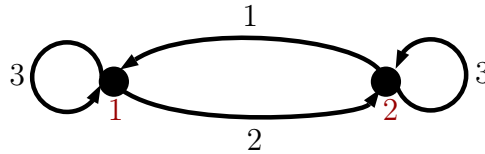
Dari apa yang telah dibahas didapat suatu kesimpulan berikut.

Kesimpulan 2.3.1 *Jika graf komunikasi $\mathcal{G}(A)$ dari matriks $A \in \mathbb{R}_\varepsilon^{n \times n}$ adalah strongly connected, maka ada skalar λ yang merupakan suatu nilai eigen dari A , dan kolom $[A_\lambda^*]_{.,\eta}$ adalah suatu vektor eigen dari A yang bersesuaian dengan λ , untuk setiap titik $\eta \in \mathcal{N}^c(A)$.*

Bukti Dari Teorema 2.1.4 didapat

$$\lambda = \bigoplus_{k=1}^n \frac{\text{tr}(A^{\otimes k})}{k}, \text{ dengan } \text{tr}(A) = \bigoplus_{i=1}^n a_{i,i}.$$

Selanjutnya bukti dapat dilakukan mengikuti hasil Teorema 2.3.2. $\square$

Gambar 2.6: Graf $\mathcal{G}(A)$

Contoh-contoh berikut memberikan gambaran yang kongkrit dari teorema dan kesimpulan yang yang baru saja dibahas.

Contoh 2.3.5 Diberikan graf *strongly connected* $\mathcal{G}(A)$ pada Gambar 2.6. Matriks taktereduksi $A = \begin{bmatrix} 3 & 1 \\ 2 & 3 \end{bmatrix}$ representasi dari graf $\mathcal{G}(A)$ jelas memiliki nilai eigen tunggal yang dapat dihitung melalui formula berikut:

$$\begin{aligned}
 \lambda(A) &= \bigoplus_{k=1}^2 \frac{\text{tr}(B^{\otimes k})}{k} \\
 &= \frac{\text{tr}(B)}{1} \oplus \frac{\text{tr}(B^{\otimes 2})}{2} \\
 &= \frac{3}{1} \oplus \frac{6}{2} \\
 &= 3 \oplus 3 \\
 &= 3.
 \end{aligned}$$

Berdasarkan perhitungan tersebut didapatkan graf kritis dari $\mathcal{G}(A)$ diberikan oleh Gambar 2.7.

Gambar 2.7: Graf $\mathcal{G}^c(A)$

Berikutnya, dihitung vektor eigen yang bersesuaian dengan nilai eigen $\lambda(A) = 3$. Pertama, dilakukan perhitungan A_λ sebagai berikut:

$$\begin{aligned}
 A_\lambda &= -\lambda \otimes A \\
 &= -3 \otimes \begin{bmatrix} 3 & 1 \\ 2 & 3 \end{bmatrix} \\
 &= \begin{bmatrix} 0 & -2 \\ -1 & 0 \end{bmatrix}
 \end{aligned}$$

Selanjutnya, dihitung matriks A_λ^+

$$\begin{aligned} A_\lambda^+ &= \bigoplus_{k=1}^2 A_\lambda^{\otimes k} \\ &= \begin{bmatrix} 0 & -2 \\ -1 & 0 \end{bmatrix} \oplus \begin{bmatrix} 0 & -2 \\ -1 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & -2 \\ -1 & 0 \end{bmatrix}. \end{aligned}$$

Terakhir, dihitung matriks A_λ^* yaitu

$$\begin{aligned} A_\lambda^* &= E \oplus A_\lambda^+ \\ &= \begin{bmatrix} 0 & \varepsilon \\ \varepsilon & 0 \end{bmatrix} \oplus \begin{bmatrix} 0 & -2 \\ -1 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & -2 \\ -1 & 0 \end{bmatrix}. \end{aligned}$$

Berdasarkan graf kritis pada Gambar 2.7 diketahui bahwa baik titik 1 dan titik 2 merupakan elemen dari $\mathcal{N}^c(A)$, sehingga kolom ke-1 dan kolom ke-2 dari matriks A_λ^* merupakan vektor eigen dari matriks A yang bersesuaian dengan nilai eigen $\lambda(A) = 3$, yaitu

$$\mathbf{v}_1 = [A_\lambda^*]_{.,1} = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

dan

$$\mathbf{v}_2 = [A_\lambda^*]_{.,2} = \begin{bmatrix} -2 \\ 0 \end{bmatrix}.$$

Hal ini bisa diselidiki sebagai berikut:

$$A \otimes \mathbf{v}_1 = \begin{bmatrix} 3 & 1 \\ 2 & 3 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ -1 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \end{bmatrix} = 3 \otimes \begin{bmatrix} 0 \\ -1 \end{bmatrix} = \lambda(A) \otimes \mathbf{v}_1$$

dan

$$A \otimes \mathbf{v}_2 = \begin{bmatrix} 3 & 1 \\ 2 & 3 \end{bmatrix} \otimes \begin{bmatrix} -2 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \end{bmatrix} = 3 \otimes \begin{bmatrix} -2 \\ 0 \end{bmatrix} = \lambda(A) \otimes \mathbf{v}_2$$

Perhatikan bahwa vektor eigen $\mathbf{v}_1$ bukan merupakan kombinasi linier dari vektor eigen $\mathbf{v}_2$ yaitu $\mathbf{v}_1 \neq a \otimes \mathbf{v}_2$, $\forall a \in \mathbb{R}$ begitu juga sebaliknya. Sebab bila $\mathbf{v}_1 = a \otimes \mathbf{v}_2$ untuk beberapa $a \in \mathbb{R}$ didapat

$$\begin{bmatrix} 0 \\ -1 \end{bmatrix} = a \otimes \begin{bmatrix} -2 \\ 0 \end{bmatrix}$$

atau ekuivalen dengan $0 = a - 2$ dan $-1 = a$. Hal ini suatu yang tidak mungkin sebab dari satu sisi $a = 2$ di sisi yang lain $a = -1$, yaitu tidak akan pernah 2 sama dengan -1 . $\square$

Contoh berikut, mempertegas bahwa vektor eigen dari suatu matriks persegi tak-tereduksi adalah tidak tunggal.

Contoh 2.3.6

$$A \otimes \mathbf{v} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} = 1 \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

dan

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \otimes \mathbf{v}_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} -1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 1 \otimes \begin{bmatrix} -1 \\ 0 \end{bmatrix}.$$

Terlihat bahwa dua vektor $\mathbf{v}$ dan $\mathbf{v}_1$ adalah vektor karakteristik dari A untuk nilai eigen $\lambda(A) = 1$. Vektor $\mathbf{v}$ bukan merupakan kombinasi linier dari vektor $\mathbf{v}_1$ dan sebaliknya, sebab

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} = a \otimes \begin{bmatrix} -1 \\ 0 \end{bmatrix} \text{ untuk beberapa } a \in \mathbb{R}$$

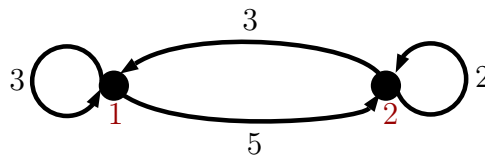
maka didapat $0 = a - 1$ yaitu $a = 1$ dan $0 = a$ suatu hal yang tidak mungkin. Perhatikan juga, vektor

$$\mathbf{v}_2 = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

adalah vektor eigen untuk nilai eigen $\lambda(A) = 1$, sebab

□

Pada Contoh 2.3.5 dan 2.3.6 dua vektor eigen yang telah diperoleh yang satu tidak bisa diperoleh dari yang lainnya dengan mengalikan skalar begitu sebaliknya (kedua vektor eigen tidak sebanding). Contoh berikut berbeda dengan dua contoh yang baru saja dibahas mengenai dua vektor eigen dari suatu matriks tak-tereduksi yaitu sebanding



Gambar 2.8: Graf $\mathcal{G}(B)$

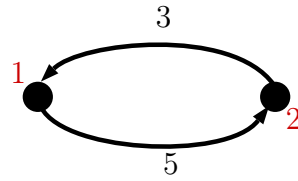
Contoh 2.3.7 Diberikan graf *strongly connected* $\mathcal{G}(B)$ oleh Gambar 2.8. Seperti contoh-contoh sebelumnya, nilai eigen dari matriks matriks representasi graf $\mathcal{G}(B)$, yaitu B dapat dicari dengan menghitung sirkuit rata-rata maksimumnya. Dengan kata lain, nilai eigen dari matriks $B \in \mathbb{R}_{\max}^{2 \times 2}$ dapat dihitung melalui formula berikut:

$$\lambda(B) = \bigoplus_{k=1}^n \frac{\text{tr}(B^{\otimes k})}{k},$$

sehingga untuk $B = \begin{bmatrix} 3 & 3 \\ 5 & 2 \end{bmatrix}$ diperoleh

$$\begin{aligned} \lambda(B) &= \bigoplus_{k=1}^2 \frac{\text{tr}(B^{\otimes k})}{k} \\ &= \frac{\text{tr}(B)}{1} \oplus \frac{\text{tr}(B^{\otimes 2})}{2} \\ &= \frac{3}{1} \oplus \frac{8}{2} \\ &= 4. \end{aligned}$$

Dari perhitungan di atas, didapatkan graf kritis dari $\mathcal{G}(B)$ seperti pada Gambar 2.9.



Gambar 2.9: Graf $\mathcal{G}^c(B)$

Matriks B representasi dari graf $\mathcal{G}(B)$ memiliki vektor eigen yang bersesuaian dengan nilai eigen λ berupa vektor kolom $[B_\lambda^*]_{\cdot, \eta}$ untuk setiap titik $\eta \in \mathcal{N}^c(B)$. Karena kedua titik dari graf $\mathcal{G}(B)$ merupakan elemen $E^c(B)$, maka dapat ditemukan dua vektor eigen melalui langkah berikut:

Pertama, dilakukan perhitungan B_λ yaitu

$$\begin{aligned} B_\lambda &= -\lambda \otimes B \\ &= -4 \otimes \begin{bmatrix} 3 & 3 \\ 5 & 2 \end{bmatrix} \\ &= \begin{bmatrix} -1 & -1 \\ 1 & -2 \end{bmatrix}. \end{aligned}$$

Selanjutnya, dihitung matriks B_λ^+ sebagai berikut

$$\begin{aligned} B_\lambda^+ &= \bigoplus_{k=1}^2 A_\lambda^{\otimes k} \\ &= \begin{bmatrix} -1 & -1 \\ 1 & -2 \end{bmatrix} \oplus \begin{bmatrix} 0 & -2 \\ 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}. \end{aligned}$$

Terakhir, dihitung matriks B_λ^* yaitu

$$\begin{aligned} B_\lambda^* &= E \oplus B_\lambda^+ \\ &= \begin{bmatrix} 0 & \varepsilon \\ \varepsilon & 0 \end{bmatrix} \oplus \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}. \end{aligned}$$

Telah disebutkan bahwa kedua titik dari graf $\mathcal{G}(B)$, yaitu titik 1 dan titik 2 merupakan elemen dari $\mathcal{N}^c(B)$. Sehingga kolom ke-1 dan kolom ke-2 dari matriks B_λ^* adalah vektor-vektor eigen yang bersesuaian dengan nilai eigen $\lambda(B)$. Jadi didapatkan vektor eigen dari matriks B yang bersesuaian dengan nilai eigen $\lambda(B) = 4$ adalah $\mathbf{v}_1 = [B_\lambda^*]_{\bullet,1} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ dan $\mathbf{v}_2 = [B_\lambda^*]_{\bullet,2} = \begin{bmatrix} -1 \\ 0 \end{bmatrix}$. Perhatikan bahwa vektor $\mathbf{v}_1$ merupakan kombinasi linier dari $\mathbf{v}_2$, sebab

$$\mathbf{v}_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 1 \otimes \begin{bmatrix} -1 \\ 0 \end{bmatrix} = 1 \otimes \mathbf{v}_2.$$

□

Berdasarkan Contoh 2.3.7 dapat ditunjukkan bahwa vektor eigen dari matriks tak-tereduksi tidak tunggal. Ketidaktunggalan tersebut karena beberapa vektor eigen didapat dari kombinasi linier dari suatu vektor eigen yang lainnya. Hal ini dapat dinyatakan dalam teorema berikut.

Teorema 2.3.3 *Untuk setiap matriks tak-tereduksi $A \in \mathbb{R}_\varepsilon^{n \times n}$ memiliki vektor eigen tidak tunggal, yaitu jika $\mathbf{v} \in \mathbb{R}_\varepsilon^n$ adalah vektor eigen yang bersesuaian dengan nilai eigen λ , maka $\alpha \otimes \mathbf{v}$ juga merupakan vektor eigen yang bersesuaian dengan nilai eigen λ untuk sebarang $\alpha \in \mathbb{R}$.*

Bukti Diberikan λ sebagai nilai eigen dari matriks tak-tereduksi A dan $\mathbf{v} \in \mathbb{R}_\varepsilon^n$ adalah vektor eigen yang bersesuaian dengan nilai eigen λ sedemikian hingga

$$A \otimes \mathbf{v} = \lambda \otimes \mathbf{v}. \quad (2.30)$$

Kalikan sebarang skalar $\alpha \in \mathbb{R}$ pada kedua ruas Persamaan (2.30) diperoleh

$$\alpha \otimes A \otimes \mathbf{v} = \alpha \otimes \lambda \otimes \mathbf{v}. \quad (2.31)$$

Karena $\alpha \in \mathbb{R}$ adalah skalar, maka Persamaan (2.31) menjadi

$$\begin{aligned} A \otimes \alpha \otimes \mathbf{v} &= \lambda \otimes \alpha \otimes \mathbf{v} \\ A \otimes (\alpha \otimes \mathbf{v}) &= \lambda \otimes (\alpha \otimes \mathbf{v}). \end{aligned}$$

Diperoleh $\alpha \otimes \mathbf{v} \in \mathbb{R}_\varepsilon^n$ juga merupakan vektor eigen dari matriks A yang bersesuaian dengan nilai eigen λ . Jadi vektor eigen dari matriks tak-tereduksi tidak tunggal. □

Berikutnya, ditunjukkan pula bahwa untuk setiap vektor eigen dari matriks tak-tereduksi hanya memiliki elemen-elemen dengan nilai berhingga.

Teorema 2.3.4 Untuk setiap matriks tak-tereduksi $A \in \mathbb{R}_\varepsilon^{n \times n}$ hanya memiliki vektor-vektor eigen dengan elemen berhingga.

Bukti Misal diberikan nilai eigen dari matriks tak-tereduksi A sama dengan λ . Menurut alur pembuktian Teorema 2.3.2 untuk mendapatkan vektor eigen dari matriks A terlebih dahulu dihitung matriks $A_\lambda \stackrel{\text{def}}{=} -\lambda \otimes A$, yang menunjukkan setiap bobot *arc* dalam graf $\mathcal{G}(A)$ dikurangi dengan λ . Selanjutnya, dilakukan perhitungan matriks $A_\lambda^+ \stackrel{\text{def}}{=} \bigoplus_{k=1}^n A_\lambda^{\otimes k}$. Karena matriks tak-tereduksi A adalah representasi dari graf *strongly connected* $\mathcal{G}(A)$, maka setiap titik pada $\mathcal{G}(A)$ saling terhubung. Hal tersebut berakibat untuk setiap entri matriks A_λ^+ semuanya tidak sama dengan ε . Sehingga untuk setiap entri matriks $A_\lambda^* \stackrel{\text{def}}{=} E \oplus A_\lambda^+$ juga semuanya tidak sama dengan ε . Karena $[A_\lambda^*]_{\cdot, \eta}$ untuk setiap $\eta \in \mathcal{N}^c(A)$ adalah vektor eigen yang bersesuaian dengan nilai eigen λ , maka jelas matriks tak-tereduksi A hanya memiliki vektor-vektor eigen dengan elemen berhingga. $\square$

Menghitung nilai-karakteristik, vektor karakteristik dan A_λ^+ dalam Scilab sebagai berikut:

```
-->A=[5 -%inf 5;-%inf 6 3;6 6 3]
A =
    5.   - Inf    5.
   - Inf    6.    3.
    6.    6.    3.
-->[l,v,d]=maxplusmaxalgol(A) // menghitung nilai-karakteristik dan
d =                               // vektor karakteristik
    1.// nilai dari d=p-q
v =
    5.
    6.
    6. // v adalah vektor-karakteristik
l =
    6. // nilai c dalam algorithma yang merupakan nilai-karakteristik.
// menghitung maksimum rata-rata sikel (lambda)
-->[mcm] = maxplusmcm(A)
mcm =
    6. // lambda (nilai-karakteristik A)
// menentukan lintasan kritis
-->[l,d,x] = maxplusccir(A)
x =
    2. // lintasan kritis
d =
```

```

1. // panjang lintasan kritis
l =
6. // lambda
-->[ap,lam] = maxplusapplus(A)
lam =
6.
ap =
- 1. - 1. - 1.
- 3.  0. - 3.
  0.  0. - 1.
// ap adalah matriks A lambda plus, lam adalah lambda.
// lintasan kritis dari G(A) adalah 2 ke 2, jadi
// kolom ke-2 dari matriks ap adalah vektor-karakteristik dari A
-->isequal(maxplusotimes(A,ap(:,2)),maxplusotimes(lam,ap(:,2)))
ans =
T

```

Intepretasi dari nilai-karakteristik dan vektor-karakteristik dalam Contoh 2.1.1 adalah sebagai berikut: Misalkan ada tiga aktifitas yang beroperasi secara periodik. Diasumsikan bahwa aktifitas ini beroperasi tidak secara bebas dan output dari satu aktifitas menjadi input aktifitas tertentu lainnya. Satu aktifitas hanya bisa dimulai bila beberapa aktifitas tertentu lainnya sudah menyelesaikan pekerjaannya dan mengirimkan hasilnya ke aktifitas yang ditentukan. Jadi satu aktifitas tertentu hanya bisa memulai kegiatannya pada saat yang ke- $(k+1)$ bila beberapa aktifitas yang lainnya telah menyelesaikan kegiatannya pada saat yang ke- k dan mengirimkan hasilnya keaktifitas tsb. Elemen a_{ij} dari matriks A menunjukkan waktu tunggu dari aktifitas i untuk memulai kegiatannya saat yang ke- $(k+1)$ setelah aktifitas j menyelesaikan kegiatannya dan mengirimkan hasilnya ke aktifitas i pada saat yang ke- k . Bila elemen $a_{ij} = \varepsilon$, maka hal ini menunjukkan bahwa aktifitas i tidak bergantung pada aktifitas j .

Selanjutnya evolusi sistem dalam Contoh 2.1.1 diberikan oleh persamaan

$$\mathbf{x}(k+1) = A \otimes \mathbf{x}(k), \quad k = 0, 1, 2, \dots \quad (2.32)$$

Bila dipilih $\mathbf{x}(0)$ adalah vektor-karakteristik dari matriks A , maka sistem (2.32) akan beroperasi secara periodik dengan periode sebesar nilai-karakteristik $\lambda = 6$. Hal ini sesuai dengan bentuk persamaan berikut :

$$\mathbf{x}(k+1) = A \otimes \mathbf{x}(k) = \lambda^{\otimes(k+1)} \otimes \mathbf{x}(0), \quad k = 0, 1, 2, \dots \quad (2.33)$$

Sehingga didapat barisan aktifitas $\mathbf{x}(k)$:

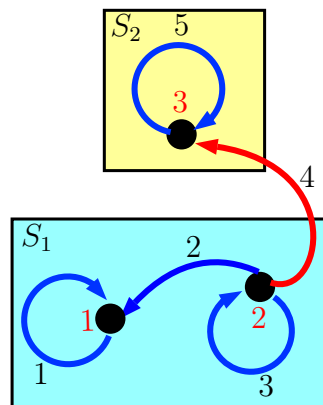
$$\begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 6 \\ 7 \\ 7 \end{bmatrix}, \begin{bmatrix} 12 \\ 13 \\ 13 \end{bmatrix}, \begin{bmatrix} 18 \\ 19 \\ 19 \end{bmatrix}, \dots$$

$$\mathbf{x}(0), \quad \mathbf{x}(1), \quad \mathbf{x}(2), \quad \mathbf{x}(3), \quad \dots$$

Evolusi dari Persamaan 2.33 dapat dilakukan dalam Scilab sebagai berikut:

```
-->A=[5 -%inf 5;-%inf 6 3;6 6 3]
A =
    5.  -Inf    5.
   -Inf    6.    3.
    6.    6.    3.
-->x0=[0;1;1]
x0 =
    0.
    1.
    1.
-->X = maxplussys(A,x0,3)
X =
    0.    6.   12.   18.
    1.    7.   13.   19.
    1.    7.   13.   19.
// untuk menentukan vektor waktu siklus sebagai berikut
-->vec=maxplusctv(A)
vec =
    6.
    6.
    6.
// Terlihat semua komponen vektor waktu siklus
// bernilai sama yaitu 6 yang menunjukkan
// nilai karakteristik dari matriks A.
```

Sebelum mengakhiri subbagian ini diberikan suatu contoh dari matriks A yang tereduksi dan dikaji perilaku semua nilai eigen yang mungkin dan vektor eigennya.



Gambar 2.10: Graf $\mathcal{G}(A)$ dari Contoh 2.3.8

Contoh 2.3.8 Diberikan matriks

$$A = \begin{bmatrix} 1 & 2 & \varepsilon \\ \varepsilon & 3 & \varepsilon \\ \varepsilon & 4 & 5 \end{bmatrix}$$

Hitung nilai-eigen λ dari matriks A dan vektor-eigen $\mathbf{v}$ dari A untuk nilai-eigen λ .

Jawab Maximum cycle mean dari A adalah

$$\lambda(A) = \max \left\{ \text{tr}(A), \frac{1}{2} \text{tr}(A^{\otimes 2}), \frac{1}{3} \text{tr}(A^{\otimes 3}) \right\} = \max \left\{ 5, \frac{10}{2}, \frac{15}{3} \right\} = 5.$$

Untuk menentukan vektor $\mathbf{v}$ adalah vektor eigen dari A untuk nilai eigen $\lambda = 5$ diperlukan matriks

$$A_{\lambda}^* = E \oplus A_{\lambda}^+ = \begin{bmatrix} 0 & -3 & \varepsilon \\ \varepsilon & 0 & \varepsilon \\ \varepsilon & -1 & 0 \end{bmatrix}$$

Dari Gambar 2.3.8 terlihat bahwa lintasan kritis dari graf $\mathcal{G}(A)$ adalah 3 ke 3, maka kolom ke-3 dari matriks $A_{\lambda}^* = \mathbf{v}$ adalah suatu vektor eigen dari A untuk nilai eigen $\lambda = 5$. Hal ini bisa diselidiki sebagai berikut

$$A \otimes \mathbf{v} = \begin{bmatrix} 1 & 2 & \varepsilon \\ \varepsilon & 3 & \varepsilon \\ \varepsilon & 4 & 5 \end{bmatrix} \otimes \begin{bmatrix} \varepsilon \\ \varepsilon \\ 0 \end{bmatrix} = \begin{bmatrix} \varepsilon \\ \varepsilon \\ 5 \end{bmatrix} = 5 \otimes \begin{bmatrix} \varepsilon \\ \varepsilon \\ 0 \end{bmatrix} = \lambda \otimes \mathbf{v}.$$

Dari Gambar 2.3.8, terlihat selain lintasan kritis ada dua siklus. Siklus dari 1 ke 1 dengan rata-rata cycle $\lambda_1 = 1$ didapat $\mathbf{v}_1$ adalah kolom ke-1 dari A_{λ}^* . Vektor $\mathbf{v}_1$ adalah vektor eigen untuk $\lambda_1 = 1$. Hal ini bisa diselidiki sebagai berikut

$$A \otimes \mathbf{v}_1 = \begin{bmatrix} 1 & 2 & \varepsilon \\ \varepsilon & 3 & \varepsilon \\ \varepsilon & 4 & 5 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ \varepsilon \\ \varepsilon \end{bmatrix} = \begin{bmatrix} 1 \\ \varepsilon \\ \varepsilon \end{bmatrix} = 1 \otimes \begin{bmatrix} 0 \\ \varepsilon \\ \varepsilon \end{bmatrix} = \lambda_1 \otimes \mathbf{v}_1.$$

Untuk siklus yang lainnya yaitu dari 2 ke 2 rata-rata cyclenya adalah $\lambda_2 = 3$ dan bukan suatu nilai eigen dari matriks A . Sebab bila merupakan nilai eigen dari matriks A , maka ada vektor $\mathbf{v}_2 \neq \varepsilon$ yang memenuhi $A \otimes \mathbf{v}_2 = \lambda_2 \otimes \mathbf{v}_2$. Misalkan hal ini benar didapat

$$\begin{aligned} A \otimes \mathbf{v}_2 &= \lambda_2 \otimes \mathbf{v}_2 \\ \begin{bmatrix} 1 & 2 & \varepsilon \\ \varepsilon & 3 & \varepsilon \\ \varepsilon & 4 & 5 \end{bmatrix} \otimes \begin{bmatrix} a \\ b \\ c \end{bmatrix} &= 3 \otimes \begin{bmatrix} a \\ b \\ c \end{bmatrix} \\ \begin{bmatrix} (1 \otimes a) \oplus (2 \otimes b) \\ 3 \otimes b \\ (4 \otimes b) \oplus (5 \otimes c) \end{bmatrix} &= \begin{bmatrix} 3 \otimes a \\ 3 \otimes b \\ 3 \otimes c \end{bmatrix} \end{aligned}$$

atau $1 + a \leq 3 + a, 2 + b \leq 3 + a, 3 + b = 3 + b$ didapat $b - a \leq 1$ sebarang dan $4 + b \leq 3 + c, 5 + b \leq 3 + c$. Dari dua pertaksamaan terakhir didapat $b - c \leq -1$ dan $5 \leq 3$ suatu hal yang tidak mungkin. Jadi tidak benar bahwa $\lambda_2 = 3$ adalah suatu nilai eigen dari matriks A . Dengan begitu A hanya mempunyai nilai eigen $\lambda = 5$ dengan vektor eigen $\mathbf{v}$ dan nilai eigen $\lambda_1 = 1$ dengan vektor eigen $\mathbf{v}_1$. Bila diperhatikan graf $\mathcal{G}(A)$ yang disajikan dalam Gambar 2.3.8, graf $\mathcal{G}(A)$ dipartisi menjadi dua bagian graf dalam S_1 dan S_2 . Graf dalam S_1 adalah $\mathcal{G}(A_{1,1})$, dimana

$$A_{1,1} = \begin{bmatrix} 1 & 2 \\ \varepsilon & 3 \end{bmatrix}$$

dan graf dalam S_2 adalah $\mathcal{G}(A_{2,2})$ dimana $A_{2,2} = [5]$. Graf $\mathcal{G}(A_{1,1})$ dan graf $\mathcal{G}(A_{2,2})$ dihubungkan oleh garis dari titik 2 ke 3 dengan bobot $a_{3,2} = 4$. Suatu pertanyaan mendasar adalah bagaimana bila nilai $a_{3,2}$ digantikan pada $a_{2,3}, a_{3,1}$ dan $a_{1,3}$ apakah ada perbedaan yang mendasar mengenai nilai eigen dan vektor eigen yang terkait? Untuk menjawab hal ini misalkan $a_{3,2} = 4$ digantikan pada $a_{2,3}$ dan nilai $a_{3,2} = \varepsilon$, didapat matriks

$$B = \begin{bmatrix} 1 & 2 & \varepsilon \\ \varepsilon & 3 & 4 \\ \varepsilon & \varepsilon & 5 \end{bmatrix}.$$

Graf $\mathcal{G}(B)$ secara prinsip dengan graf $\mathcal{G}(A)$ hampir sama hanya garis yang menghubungkan graf dalam S_1 dan S_2 yang arahnya berlawanan. Maksimum nilai rata-rata siklus dari graf $\mathcal{G}(B)$ sama dengan maksimum nilai rata-rata siklus dari graf $\mathcal{G}(A)$. Jadi $\lambda(B) = \lambda(A) = 5$. Begitu juga dua nilai rata-rata siklus yang lain dari graf $\mathcal{G}(B)$ adalah $\lambda_1 = 1$ dan $\lambda_2 = 3$. Didapat matriks

$$B_{\lambda}^* = \begin{bmatrix} 0 & -3 & -4 \\ \varepsilon & 0 & -1 \\ \varepsilon & \varepsilon & 0 \end{bmatrix}.$$

Dengan cara seperti yang dilakukan sebelumnya dapat diselidiki bahwa vektor $\mathbf{u}_1 = [B_{\lambda}^*]_{\bullet,1}$ adalah vektor eigen dari B untuk nilai eigen $\lambda_1 = 1$ dan vektor $\mathbf{u} = [B_{\lambda}^*]_{\bullet,3}$ adalah vektor eigen dari B untuk nilai eigen $\lambda(B) = 5$. Sedangkan untuk $\lambda_2 = 3$ bukan nilai eigen dari B . Sebab bila merupakan nilai eigen dari matriks B , maka ada vektor $\mathbf{u}_2 \neq \mathbf{0}$ yang memenuhi $B \otimes \mathbf{u}_2 = \lambda_2 \otimes \mathbf{u}_2$. Misalkan hal ini benar didapat

$$\begin{aligned} B \otimes \mathbf{u}_2 &= \lambda_2 \otimes \mathbf{u}_2 \\ \begin{bmatrix} 1 & 2 & \varepsilon \\ \varepsilon & 3 & 4 \\ \varepsilon & \varepsilon & 5 \end{bmatrix} \otimes \begin{bmatrix} a \\ b \\ c \end{bmatrix} &= 3 \otimes \begin{bmatrix} a \\ b \\ c \end{bmatrix} \\ \begin{bmatrix} (1 \otimes a) \oplus (2 \otimes b) \\ (3 \otimes b) \oplus (4 \otimes c) \\ 5 \otimes c \end{bmatrix} &= \begin{bmatrix} 3 \otimes a \\ 3 \otimes b \\ 3 \otimes c \end{bmatrix} \end{aligned}$$

atau $1 + a \leq 3 + a, 2 + b \leq 3 + a, 3 + b = 3 + b, 4 + c \leq 3 + b$ didapat $a \in \mathbb{R}$ sebarang dan $5 + c \leq 3 + b$. Dari pertaksamaan terakhir didapat $5 \leq 3$ suatu hal yang tidak mungkin.

Jadi tidak benar bahwa $\lambda_2 = 3$ adalah suatu nilai eigen dari matriks B . Dengan begitu B hanya mempunyai nilai eigen $\lambda(B) = 5$ dengan vektor eigen $\mathbf{u}$ dan nilai eigen $\lambda_1 = 1$ dengan vektor eigen $\mathbf{u}_1$. Hasil pembahasan matriks A dan B , menunjukkan bahwa memberikan perilaku yang sama.

Berikutnya bila nilai $a_{3,2} = 4$ digantikan pada $a_{1,3}$ dan $a_{3,2} = \varepsilon$, maka didapat matriks

$$C = \begin{bmatrix} 1 & 2 & 4 \\ \varepsilon & 3 & \varepsilon \\ \varepsilon & \varepsilon & 5 \end{bmatrix}.$$

Graf $\mathcal{G}(C)$ mempunyai tiga siklus dengan bobot rata-rata siklus adalah $\lambda_1 = 1, \lambda_2 = 3$ dan maksimum bobot rata-rata siklus $\lambda(C) = 5$. Untuk menyelidiki apakah ketiga nilai bobot rata-rata siklus tersebut adalah nilai eigen dari matriks C , diperlukan matriks berikut

$$C_\lambda^* = \begin{bmatrix} 0 & -3 & -1 \\ \varepsilon & 0 & \varepsilon \\ \varepsilon & \varepsilon & 0 \end{bmatrix}$$

Jelas bahwa vektor $\mathbf{w}_1 = [C_\lambda^*]_{\bullet,1}$ adalah vektor eigen untuk nilai eigen $\lambda_1 = 1$ dari matriks C . Sedangkan vektor $\mathbf{w}_3 = [C_\lambda^*]_{\bullet,3}$ diselidiki sebagai berikut

$$C \otimes \mathbf{w}_3 = \begin{bmatrix} 1 & 2 & 4 \\ \varepsilon & 3 & \varepsilon \\ \varepsilon & \varepsilon & 5 \end{bmatrix} \otimes \begin{bmatrix} -1 \\ \varepsilon \\ 0 \end{bmatrix} = \begin{bmatrix} 4 \\ \varepsilon \\ 5 \end{bmatrix} = 5 \otimes \begin{bmatrix} -1 \\ \varepsilon \\ 0 \end{bmatrix} = \lambda(C) \otimes \mathbf{w}_3.$$

Terlihat bahwa $\lambda(C) = 5$ adalah suatu nilai eigen dari C dengan vektor eigen $\mathbf{w}_3$. Lalu bagaimana dengan $\lambda_2 = 3$ apakah merupakan nilai eigen dari matriks C ? Nilai $\lambda_2 = 3$ adalah nilai eigen dari C , sebab

$$C \otimes \mathbf{w}_2 = \begin{bmatrix} 1 & 2 & 4 \\ \varepsilon & 3 & \varepsilon \\ \varepsilon & \varepsilon & 5 \end{bmatrix} \otimes \begin{bmatrix} -1 \\ 0 \\ \varepsilon \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \\ \varepsilon \end{bmatrix} = 3 \otimes \begin{bmatrix} -1 \\ 0 \\ \varepsilon \end{bmatrix} = \lambda_2 \otimes \mathbf{w}_2.$$

Dari kajian matriks C menunjukkan bahwa semua bobot rata-rata siklus dari graf $\mathcal{G}(C)$ yaitu $\lambda_1 = 1, \lambda_2 = 3$ dan $\lambda(C) = 5$ adalah nilai eigen dari matriks C masing-masing dengan vektor eigen $\mathbf{w}_1, \mathbf{w}_2$ dan $\mathbf{w}_3$.

Pembahasan terakhir adalah nilai $a_{3,2}$ digantikan pada nilai $a_{3,1}$ dan nilai $a_{3,2} = \varepsilon$ didapat matriks

$$D = \begin{bmatrix} 1 & \varepsilon & \varepsilon \\ 2 & 3 & \varepsilon \\ 4 & \varepsilon & 5 \end{bmatrix}.$$

Graf $\mathcal{G}(D)$ hampir sama dengan graf $\mathcal{G}(A)$ hanya garis dari 2 ke 3 diganti garis dari 1 ke 3. Maksimum nilai rata-rata siklus dari graf $\mathcal{G}(D)$ sama dengan maksimum nilai rata-rata siklus dari graf $\mathcal{G}(D)$. Jadi $\lambda(D) = \lambda(A) = 5$. Begitu juga dua nilai rata-rata siklus

yang lain dari graf $\mathcal{G}(D)$ adalah $\lambda_1 = 1$ dan $\lambda_2 = 3$. Jelas bahwa $\lambda(D) = 5$ adalah nilai eigen dari matriks D dengan vektor eigen $[\varepsilon \ \varepsilon \ 0]^\top$. Lalu bagaimana dengan $\lambda_1 = 1$ dan $\lambda_2 = 3$, apakah nilai eigen dari matriks D ? Jawabannya, $\lambda_1 = 1$ dan $\lambda_2 = 3$ bukan nilai eigen dari matriks D . Hal ini bisa ditunjukkan dengan cara yang sama sebagaimana telah dilakukan pada matriks A dan B . Penyelidikan yang lainnya untuk matriks yang grafnya mempunyai bobot rata-rata sikel sama dengan bobot rata-rata sikeldari graf $\mathcal{G}(A)$ adalah matriks A transpose. Hal ini bisa dilakukan sebagaimana telah dilakukan untuk matriks A . $\square$

Pembahasan dan diskusi yang diberikan dalam Contoh 2.3.8 menarik untuk dikaji lebih mendalam. Yaitu kajian mengenai berapa banyak nilai eigen yang mungkin dari suatu matriks persegi A dimana graf $\mathcal{G}(A)$ mempunyai sejumlah berhingga sirkuit.

2.4 Penyelesaian Persamaan Linear

Kekurangan dari aljabar max-plus adalah tidak adanya invers additive, hal ini menyulitkan untuk menyelesaikan sistem persamaan linear seperti $A \otimes \mathbf{x} = \mathbf{b}$. Sebagaimana dalam aljabar biasa penyelesaian persamaan $A \otimes \mathbf{x} = \mathbf{b}$ tidak selalu ada, bila ada hal ini belum tentu tunggal. Pada bagian ini juga akan dibahas bentuk yang lainnya dari persamaan linear.

2.4.1 Sub-Penyelesaian Terbesar

Sebagai motifasi, diberikan persamaan dimensi satu

$$a + x = b, \quad (2.34)$$

dengan a, b adalah bilangan real taknegatif. Jelas bahwa bila $a > b$, maka Persamaan (2.34) tidak mempunyai penyelesaian. Sebaliknya bila $a \leq b$, maka Persamaan (2.34) mempunyai penyelesaian $x = b - a$. Begitu juga untuk persamaan berikut

$$a \oplus x = b, \quad (2.35)$$

dengan $a, b \in \mathbb{R}$. Jelas bahwa bila $a > b$, maka Persamaan (2.35) tidak mempunyai penyelesaian. Sebaliknya bila $a \leq b$, maka Persamaan (2.35) mempunyai penyelesaian $x = b$. Perhatikan bahwa Persamaan (2.34) dapat ditulis dalam bentuk $a \otimes x = b$.

Selanjutnya dibahas a diganti oleh matriks A . Pertama, kasus untuk matriks A tidak harus matriks persegi. Untuk matriks A ini, selalu didapat apa yang dikenal dengan sub-penyelesaian terbesar dari $A \otimes \mathbf{x} = \mathbf{b}$. Sub-penyelesaian terbesar adalah vektor terbesar $\mathbf{x}$ yang memenuhi $A \otimes \mathbf{x} \leq \mathbf{b}$. Penyelesaian ini dinotasikan oleh $\mathbf{x}^*(A, \mathbf{b})$. Sub-penyelesaian terbesar tidak harus merupakan suatu penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$.

$$A = \begin{bmatrix} 1 & \varepsilon \\ 3 & 4 \end{bmatrix}, b = \begin{bmatrix} 1 \\ 2 \end{bmatrix}.$$

Persamaan $A \otimes \mathbf{x} = \mathbf{b}$ tidak punya penyelesaian, sebab bila punya penyelesaian berarti ada $\mathbf{x} = \begin{bmatrix} p \\ q \end{bmatrix}$ sehingga $\begin{bmatrix} 1 & \varepsilon \\ 3 & 4 \end{bmatrix} \otimes \begin{bmatrix} p \\ q \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$. Didapat $p = 0$ dan $\max\{3, 4 + q\} = 2$, terlihat bahwa tidak akan ada $q \in \mathbb{R}_\varepsilon$ sehingga $\max\{3, 4 + q\} = 2$. Jadi $A \otimes \mathbf{x} = \mathbf{b}$ tidak punya penyelesaian. Dilain pihak secara umum, pertaksamaan $A \otimes \mathbf{x} \leq \mathbf{b}$ selalu punya penyelesaian. Hal ini bisa ditunjukkan bahwa untuk $x = \varepsilon$ didapat $A \otimes \mathbf{x} = \varepsilon \leq \mathbf{b}$. Contoh-contoh diatas menjelaskan bahwa $A \otimes \mathbf{x} = \mathbf{b}$ belum tentu punya penyelesaian, dilain pihak $A \otimes \mathbf{x} \leq \mathbf{b}$ selalu punya penyelesaian, bahkan penyelesaian terbesar x yang memenuhi $A \otimes \mathbf{x} \leq \mathbf{b}$ diberikan oleh teorema berikut.

Teorema 2.4.1 Misalkan $A \in \mathbb{R}_\varepsilon^{m \times n}$ adalah suatu matriks yang setiap kolomnya memuat setidaknya satu elemen tidak sama dengan ε dan $\mathbf{b} \in \mathbb{R}_\varepsilon^m$, maka

$$[\mathbf{x}^*(A, \mathbf{b})]_j = \min\{b_i - a_{i,j} \mid i \in \underline{m}, \text{ dan } a_{i,j} > \varepsilon\}.$$

Bukti

Perhatikan bahwa $A \otimes \mathbf{x} \leq \mathbf{b}$ adalah ekivalen dengan masing-masing berikut:

1. untuk semua i dan j , $a_{i,j} + x_j \leq b_i$
2. untuk semua i dan j , $x_j \leq b_i - a_{i,j}$ atau $a_{i,j} = \varepsilon$
3. untuk semua j , $x_j \leq \min\{b_i - a_{i,j} \mid i \in \underline{m} \text{ dan } a_{i,j} > \varepsilon\}$.

Hal ini menjelaskan bahwa $\mathbf{x}$ adalah suatu penyelesaian dari $A \otimes \mathbf{x} \leq \mathbf{b}$ bila dan hanya bila untuk semua j , $x_j \leq \min\{b_i - a_{i,j} \mid i \in \underline{m} \text{ dan } a_{i,j} > \varepsilon\}$. Oleh karena itu, $[\mathbf{x}^*(A, \mathbf{b})]_j = \min\{b_i - a_{i,j} \mid i \in \underline{m} \text{ dan } a_{i,j} > \varepsilon\}$ adalah penyelesaian maksimum dari $A \otimes \mathbf{x} \leq \mathbf{b}$. ■

Teorema 2.4.1 menjelaskan penyelesaian dari $A \otimes \mathbf{x} \leq \mathbf{b}$, sedangkan penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$ bila ada diberikan oleh sifat berikut.

Lemma 2.4.1 Bila suatu penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$ ada, maka sub-penyelesaian terbesar adalah penyelesaiannya.

Bukti

Misalkan $\mathbf{x}'$ adalah suatu penyelesaian maksimum dari $A \otimes \mathbf{x} = \mathbf{b}$, maka $\mathbf{x}$ memenuhi pertaksamaan $A \otimes \mathbf{x} \leq \mathbf{b}$. Jadi haruslah $\mathbf{x}'$ adalah sub-penyelesaian terbesar. Sebagaimanana diketahui bahwa sub-penyelesaian $\mathbf{x}^*(A, \mathbf{b})$ adalah maksimum penyelesaian dari $A \otimes \mathbf{x} \leq \mathbf{b}$. Karena penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$ ada, maka $\mathbf{x}^*(A, \mathbf{b})$ adalah penyelesaiannya. Hal ini menunjukkan bahwa sub-penyelesaian terbesar adalah suatu penyelesaian. ■

Perhatikan bahwa, untuk semua $j \in \underline{n}$, $x_j \leq \min\{b_i - a_{i,j} \mid i \in \underline{m}\}$ dapat diganti oleh: untuk semua $j \in \underline{n}$, $x_j \leq -\max\{-b_i + a_{i,j} \mid i \in \underline{m}\}$ atau untuk semua $j \in \underline{n}$, $-x_j \geq \max\{-b_i + a_{i,j} \mid i \in \underline{m}\}$. Atau secara matriks dalam aljabar max-plus adalah $-\mathbf{x}^\top \geq -\mathbf{b}^\top \otimes A$.

Contoh-contoh berikut menjelaskan hal-hal yang berkaitan dengan Teorema 2.4.1 dan Lemma 2.4.1.

Contoh 2.4.1 Misalkan $A = \begin{bmatrix} 2 & 0 \\ 1 & 3 \end{bmatrix}$, $\mathbf{b} = \begin{bmatrix} 5 \\ 4 \end{bmatrix}$. Dengan menggunakan Teorema 2.4.1, penyelesaian terbesarnya adalah $\mathbf{x} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$. Penyelesaian ini juga memenuhi $A \otimes \mathbf{x} = \mathbf{b}$. Hal ini bisa dicek sebagai berikut:

$$\begin{bmatrix} -5 & -4 \end{bmatrix} \otimes \begin{bmatrix} 2 & 0 \\ 1 & 3 \end{bmatrix} = \begin{bmatrix} -3 & -1 \end{bmatrix} = -\mathbf{x}^\top.$$

Jadi $\mathbf{x} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$. Sehingga didapat $A \otimes \mathbf{x} = \begin{bmatrix} 2 & 0 \\ 1 & 3 \end{bmatrix} \otimes \begin{bmatrix} 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 4 \end{bmatrix} = \mathbf{b}$. □

Contoh 2.4.2 Misalkan $A = \begin{bmatrix} 3 & 2 \\ 1 & 4 \end{bmatrix}$, $\mathbf{b} = \begin{bmatrix} 20 \\ 4 \end{bmatrix}$. Dengan menggunakan Teorema 2.4.1, penyelesaian terbesarnya adalah $\mathbf{x} = \begin{bmatrix} 3 \\ 0 \end{bmatrix}$. Tetapi penyelesaian ini memenuhi $A \otimes \mathbf{x} \neq \mathbf{b}$, jadi $A \otimes \mathbf{x} = \mathbf{b}$ tidak punya penyelesaian. Hal ini bisa dicek sebagai berikut:

$$\begin{bmatrix} -20 & -4 \end{bmatrix} \otimes \begin{bmatrix} 3 & 2 \\ 1 & 4 \end{bmatrix} = \begin{bmatrix} -3 & 0 \end{bmatrix} = -\mathbf{x}^\top.$$

Jadi $\mathbf{x} = \begin{bmatrix} 3 \\ 0 \end{bmatrix}$. Sehingga didapat $A \otimes \mathbf{x} = \begin{bmatrix} 3 & 2 \\ 1 & 4 \end{bmatrix} \otimes \begin{bmatrix} 3 \\ 0 \end{bmatrix} = \begin{bmatrix} 6 \\ 4 \end{bmatrix} \leq \begin{bmatrix} 20 \\ 4 \end{bmatrix} = \mathbf{b}$. □

Teorema 2.4.1 telah diimplementasikan dalam toolbox aljabar max-plus, sehingga Contoh 2.4.1 dan Contoh 2.4.2 dapat dilakukan dalam Scilab 5.1.1 sebagai berikut:

```
// maxplus(A,x)=b punya penyelesaian
-->A=[2 0; 1 3]
A =
2.    0.
1.    3.
-->b=[5;4]
b =
5.
4.
```

```

-->x = maxpluslinsol(A,b)
x =
3.
1.
// Cek maxplusotimes(A,x)==b
-->maxplusotimes(A,x)==b
ans =
T
T
// maxplus(A,x)=b tidak punya penyelesaian
-->A=[3 2; 1 4]
A =
3.    2.
1.    4.
-->b=[20;4]
b =
20.
4.
-->x = maxpluslinsol(A,b)
x =
3.
0.
// Cek maxplusotimes(A,x) <= b
-->maxplusotimes(A,x)<=b
ans =
T
T
// Cek bahwa maxplusotimes(A,x) tidak sama dengan b
-->~isequal(maxplusotimes(A,x)-b,zeros(2,1))
ans =
T

```

2.4.2 Analisis Penyelesaian $A \otimes x = b$

Pada pembahasan sebelumnya telah dibahas tentang persamaan

$$A \otimes x = b. \quad (2.36)$$

Persamaan ini selalu mempunyai penyelesaian suboptimal x yang memenuhi

$$A \otimes x \leq b.$$

Pada bagian ini dibahas beberapa kemungkinan penyelesaian Persamaan (2.36) bila ada yaitu bila mempunyai penyelesaian tunggal dan banyak penyelesaian. Disamping itu juga

dibahas bila Persamaan (2.36) tidak mempunyai penyelesaian. Persamaan (2.36) ditulis ulang dalam bentuk

$$\begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{bmatrix} \otimes \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}$$

atau

$$\begin{aligned} (a_{1,1} \otimes x_1) \oplus (a_{1,2} \otimes x_2) \oplus \cdots \oplus (a_{1,n} \otimes x_n) &= b_1 \\ (a_{2,1} \otimes x_1) \oplus (a_{2,2} \otimes x_2) \oplus \cdots \oplus (a_{2,n} \otimes x_n) &= b_2 \\ &\vdots \\ (a_{m,1} \otimes x_1) \oplus (a_{m,2} \otimes x_2) \oplus \cdots \oplus (a_{m,n} \otimes x_n) &= b_m \end{aligned}$$

atau ditulis dalam notasi baku, harus diselesaikan secara simultan sistem persamaan

$$\begin{aligned} \max \{(a_{1,1} + x_1), (a_{1,2} + x_2), \cdots, (a_{1,n} + x_n)\} &= b_1 \\ \max \{(a_{2,1} + x_1), (a_{2,2} + x_2), \cdots, (a_{2,n} + x_n)\} &= b_2 \\ &\vdots \\ \max \{(a_{m,1} + x_1), (a_{m,2} + x_2), \cdots, (a_{m,n} + x_n)\} &= b_m \end{aligned}$$

Kasus yang pertama dibahas ada suatu penyelesaian dan beberapa elemen dari $\mathbf{b}$ adalah ε . Tanpa menghilangkan keumumannya, persamaan dapat disusun ulang sehingga elemen-elemen yang berhingga disusun dengan urutan yang pertamam didapat

$$\begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{bmatrix} \otimes \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_k \\ \varepsilon \\ \vdots \\ \varepsilon \end{bmatrix}$$

Tulis dalam bentuk baku, didapat

$$\begin{aligned} \max \{(a_{1,1} + x_1), (a_{1,2} + x_2), \cdots, (a_{1,n} + x_n)\} &= b_1 \\ &\vdots \\ \max \{(a_{k,1} + x_1), (a_{k,2} + x_2), \cdots, (a_{k,n} + x_n)\} &= b_k \\ \max \{(a_{k+1,1} + x_1), (a_{k+1,2} + x_2), \cdots, (a_{k+1,n} + x_n)\} &= \varepsilon \\ &\vdots \\ \max \{(a_{m,1} + x_1), (a_{m,2} + x_2), \cdots, (a_{m,n} + x_n)\} &= \varepsilon \end{aligned}$$

Lakukan penomoran ulang pada peubah untuk j sehingga

$$a_{k+1,j}, \dots, a_{m,j} = \varepsilon$$

terjadi pertama, didapat

$$\left[\begin{array}{ccc|c} & A_1 & & A_2 \\ - & - & - & - \\ -\infty & \dots & -\infty & \\ \vdots & \ddots & \vdots & A_3 \\ -\infty & \dots & -\infty & \end{array} \right] \otimes \begin{bmatrix} x_1 \\ \vdots \\ x_l \\ x_{l+1} \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_k \\ -\infty \\ \vdots \\ -\infty \end{bmatrix},$$

dengan matriks A_1 berukuran $k \times l$. Misalkan

$$\bar{\mathbf{b}} = \begin{bmatrix} b_1 \\ \vdots \\ b_k \end{bmatrix} \text{ dan } \bar{\mathbf{x}} = \begin{bmatrix} x_1 \\ \vdots \\ x_l \end{bmatrix}.$$

Catatan bahwa, $A \otimes \mathbf{x} = \mathbf{b}$ mempunyai penyelesaian, maka $x_{l+1} = \dots = x_n = -\infty$ dan $A_1 \otimes \bar{\mathbf{x}} = \bar{\mathbf{b}}$. Jadi, $A \otimes \mathbf{x} = \mathbf{b}$ mempunyai penyelesaian bila dan hanya bila $\bar{\mathbf{x}}$ adalah penyelesaian dari $A_1 \otimes \bar{\mathbf{x}} = \bar{\mathbf{b}}$ dan penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$ adalah

$$\mathbf{x} = \begin{bmatrix} \bar{\mathbf{x}} \\ -\infty \\ \vdots \\ -\infty \end{bmatrix}.$$

Oleh karena itu, penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$ dengan beberapa elemen $\mathbf{b}$ takhingga dapat direduksi kebentuk $A_1 \otimes \bar{\mathbf{x}} = \bar{\mathbf{b}}$ dengan semua elemen dari $\bar{\mathbf{b}}$ berhingga. Jadi pembahasan persamaan $A \otimes \mathbf{x} = \mathbf{b}$ dapat ditekan pada semua elemen $\mathbf{b}$ berhingga. Bila $A \otimes \mathbf{x} = \mathbf{b}$ mempunyai penyelesaian, maka

$$a_{i,j} + x_j \leq b_i, \text{ untuk semua } i \in \underline{m} \text{ dan } j \in \underline{n}.$$

Pertaksamaan ini dapat diperlakukan secara terpisah unuk setiap j , didapat

$$a_{i,1} + x_1 \leq b_i \text{ atau } x_1 \leq b_i - a_{i,1}$$

atau

$$x_1 \leq \min\{(b_1 - a_{1,1}), (b_2 - a_{2,1}), \dots, (b_m - a_{m,1})\}.$$

Jadi bila sistem mempunyai penyelesaian haruslah memenuhi

$$x_1 \leq \min\{(b_1 - a_{1,1}), (b_2 - a_{2,1}), \dots, (b_m - a_{m,1})\}.$$

Dengan demikian penyelesaian $\mathbf{x}$ yang mungkin memenuhi

$$\begin{aligned} x_1 &\leq \min\{(b_1 - a_{1,1}, (b_2 - a_{2,1}), \dots, (b_m - a_{m,1})\} \\ x_2 &\leq \min\{(b_1 - a_{1,2}, (b_2 - a_{2,2}), \dots, (b_m - a_{m,2})\} \\ &\vdots \\ x_n &\leq \min\{(b_1 - a_{1,n}, (b_2 - a_{2,n}), \dots, (b_m - a_{m,n})\}. \end{aligned}$$

Jadi, calon penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$ yang dinotasikan dengan $\bar{\mathbf{x}}$ adalah

$$\bar{\mathbf{x}} = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \\ \vdots \\ \bar{x}_n \end{bmatrix}, \text{ dengan } \begin{cases} x_1 = \min\{(b_1 - a_{1,1}, (b_2 - a_{2,1}), \dots, (b_m - a_{m,1})\} \\ x_2 = \min\{(b_1 - a_{1,2}, (b_2 - a_{2,2}), \dots, (b_m - a_{m,2})\} \\ \vdots \\ x_n = \min\{(b_1 - a_{1,n}, (b_2 - a_{2,n}), \dots, (b_m - a_{m,n})\} \end{cases}$$

Selanjutnya didefinisikan matriks "discrepancy" (ketidaksesuaian) dinotasikan oleh $D_{A,\mathbf{b}}$ dengan

$$D_{A,\mathbf{b}} = \begin{bmatrix} b_1 - a_{1,1} & b_1 - a_{1,2} & \dots & b_1 - a_{1,n} \\ b_2 - a_{2,1} & b_2 - a_{2,2} & \dots & b_2 - a_{2,n} \\ \vdots & \vdots & \vdots & \vdots \\ b_m - a_{m,1} & b_m - a_{m,2} & \dots & b_m - a_{m,n} \end{bmatrix}$$

Catatan bahwa, minimum dari setiap kolom $D_{A,\mathbf{b}}$ adalah elemen dari $\bar{\mathbf{x}}$. Selanjutnya didefinisikan matriks tereduksi ketidaksesuaian $R_{A,\mathbf{b}}$ oleh

$$R_{A,\mathbf{b}} = [r_{i,j}] \text{ dengan } r_{i,j} = \begin{cases} 1, & \text{bila } d_{i,j} = \text{minimum dari kolom ke } j \\ 0, & \text{yang lainnya} \end{cases}$$

Contoh 1

Selesaikan $A \otimes \mathbf{x} = \mathbf{b}$, bila

$$A = \begin{bmatrix} 2 & 3 & 1 \\ 0 & 4 & 6 \\ 3 & 1 & -2 \\ 9 & 6 & 3 \end{bmatrix}, \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \text{ dan } \mathbf{b} = \begin{bmatrix} 6 \\ 10 \\ 5 \\ 11 \end{bmatrix}.$$

Matriks

$$D_{A,\mathbf{b}} = \begin{bmatrix} 4 & 3 & 5 \\ 10 & 6 & 4 \\ 2 & 4 & 7 \\ 2 & 5 & 8 \end{bmatrix} \text{ dan } R_{A,\mathbf{b}} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix},$$

perhatikan bahwa setiap kolom matriks $R_{A,\mathbf{b}}$ hanya terdapat tepat satu elemen bernilai 1. Hal ini mengisyaratkan $A \otimes \mathbf{x} = \mathbf{b}$ hanya mempunyai tepat satu penyelesaian $\bar{\mathbf{x}}$ dengan elemen-elemen adalah minimum dari setiap kolom matrik $D_{A,\mathbf{b}}$, yaitu

$$\bar{\mathbf{x}} = \begin{bmatrix} 2 \\ 3 \\ 4 \end{bmatrix}.$$

Selanjutnya bisa dicek bahwa

$$\begin{aligned} A \otimes \bar{\mathbf{x}} &= \begin{bmatrix} 2 & 3 & 1 \\ 0 & 4 & 6 \\ 3 & 1 & -2 \\ 9 & 6 & 3 \end{bmatrix} \otimes \begin{bmatrix} 2 \\ 3 \\ 4 \end{bmatrix} \\ &= \begin{bmatrix} 6 \\ 10 \\ 5 \\ 11 \end{bmatrix} \\ &= \mathbf{b}. \end{aligned}$$

Matriks $D_{A,\mathbf{b}}$ dan $R_{A,\mathbf{b}}$ penting untuk menentukan perilaku penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$. Telah diketahui bahwa penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$ bila ada yaitu $\mathbf{x}$ dengan elemen x_i diberikan oleh

$$x_j = \min_k \{b_k - a_{k,j}\} = \min_k \{-a_{k,j} + b_k\} = \oplus'_k (-a_{k,j} \oplus b_k),$$

jadi dalam min plus aljabar $\mathbf{x}$ diberikan oleh

$$\mathbf{x} = -A^T \otimes' \mathbf{b}.$$

Dengan demikian didapat

$$\mathbf{x} = \begin{bmatrix} -2 & 0 & -3 & -9 \\ -3 & -4 & -1 & -6 \\ -1 & -6 & 2 & -3 \end{bmatrix} \otimes' \begin{bmatrix} 6 \\ 10 \\ 5 \\ 11 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \\ 4 \end{bmatrix}$$

Contoh 2

Selesaikan $A \otimes \mathbf{x} = \mathbf{b}$, bila

$$A = \begin{bmatrix} 2 & 3 & 1 \\ 0 & 4 & 6 \\ 3 & 1 & -2 \\ 9 & 6 & 3 \end{bmatrix}, \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \text{ dan } \mathbf{b} = \begin{bmatrix} 8 \\ 13 \\ 5 \\ 10 \end{bmatrix}.$$

Matriks

$$D_{A,\mathbf{b}} = \begin{bmatrix} 6 & 5 & 7 \\ 13 & 9 & 7 \\ 2 & 4 & 7 \\ 1 & 4 & 7 \end{bmatrix} \text{ dan } R_{A,\mathbf{b}} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix},$$

perhatikan bahwa setiap kolom matriks $R_{A,\mathbf{b}}$ hanya terdapat setidaknya satu elemen bernilai 1 sedangkan baris ke-2 dan ke-3 terdapat nilai 1 lebih dari satu. Hal ini mengisyaratkan $A \otimes \mathbf{x} = \mathbf{b}$ mempunyai lebih dari satu (tak hingga) penyelesaian $\bar{\mathbf{x}}$ dengan elemen adalah minimum dari setiap kolom matriks $D_{A,\mathbf{b}}$, yaitu

$$\bar{\mathbf{x}} = \begin{bmatrix} 1 \\ 4 \\ 7 \end{bmatrix}.$$

Selanjutnya bisa dicek bahwa

$$\begin{aligned} A \otimes \bar{\mathbf{x}} &= \begin{bmatrix} 2 & 3 & 1 \\ 0 & 4 & 6 \\ 3 & 1 & -2 \\ 9 & 6 & 3 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 4 \\ 7 \end{bmatrix} \\ &= \begin{bmatrix} 8 \\ 13 \\ 5 \\ 10 \end{bmatrix} \\ &= \mathbf{b}. \end{aligned}$$

Bisa dicek bahwa penyelesaian yang lain adalah

$$\bar{\mathbf{x}} = \begin{bmatrix} c_1 \\ c_2 \\ 7 \end{bmatrix},$$

dengan $c_1 \leq 1$ dan $c_2 \leq 4$.

Contoh 3

Selesaikan $A \otimes \mathbf{x} = \mathbf{b}$, bila

$$A = \begin{bmatrix} 2 & 3 & 1 \\ 0 & 4 & 6 \\ 3 & 1 & -2 \\ 9 & 6 & 3 \end{bmatrix}, \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \text{ dan } \mathbf{b} = \begin{bmatrix} 6 \\ 12 \\ 5 \\ 9 \end{bmatrix}.$$

Matriks

$$D_{A,\mathbf{b}} = \begin{bmatrix} 4 & 3 & 5 \\ 12 & 8 & 6 \\ 2 & 4 & 7 \\ 0 & 3 & 6 \end{bmatrix} \text{ dan } R_{A,\mathbf{b}} = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix},$$

perhatikan bahwa tidak semua kolom matriks $R_{A,b}$ setidaknya memuat satu elemen bernilai 1, yaitu baris ke-2 semua elemennya bernilai 0 begitu juga baris ke-3 semua elemennya bernilai 0. Hal ini mengisyaratkan $A \otimes \mathbf{x} = \mathbf{b}$ tidak mempunyai penyelesaian, tetapi hanya mempunyai subpenyelesaian optimal $\bar{\mathbf{x}}$ dengan elemen adalah minimum dari setiap kolom matriks $D_{A,b}$, yaitu

$$\bar{\mathbf{x}} = \begin{bmatrix} 0 \\ 3 \\ 5 \end{bmatrix}.$$

Selanjutnya bisa dicek bahwa

$$\begin{aligned} A \otimes \bar{\mathbf{x}} &= \begin{bmatrix} 2 & 3 & 1 \\ 0 & 4 & 6 \\ 3 & 1 & -2 \\ 9 & 6 & 3 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 3 \\ 5 \end{bmatrix} \\ &= \begin{bmatrix} 6 \\ 11 \\ 4 \\ 19 \end{bmatrix} \\ &\leq \begin{bmatrix} 6 \\ 12 \\ 5 \\ 9 \end{bmatrix} = \mathbf{b}. \end{aligned}$$

Perhatikan bahwa untuk suatu kolom j pada matriks $D_{A,b}$, elemen minimum dari kolom ini adalah penyelesaian maksimum dari sistem pertaksamaan untuk x_j . Dengan mengubah sistem pertaksamaan ini menjadi persamaan, didapat persamaan pada setiap baris pertaksamaan, dengan demikian haruslah ada setidaknya satu minimum di setiap baris $D_{A,b}$, yaitu ada setidaknya satu elemen sama dengan 1 disetiap baris matriks $R_{A,b}$ agar supaya ada penyelesaian. Pada Contoh 1 setiap kolom matriks $R_{A,b}$ hanya terdapat tepat satu elemen bernilai 1, hal ini mengisyaratkan persamaan $A \otimes \mathbf{x} = \mathbf{b}$ mempunyai penyelesaian tunggal. Berikut ini diberikan teorema mengenai beberapa hal yang telah dibahas.

Teorema 2.4.2 *Diberikan persamaan $A \otimes \mathbf{x} = \mathbf{b}$ dengan ukuran A adalah $m \times n$, $\mathbf{x}$ berukuran $n \times 1$ dan $\mathbf{b}$ berukuran $m \times 1$ yang semua elemennya berhingga. Bila suatu baris dari matriks $R_{A,b}$ semua elemennya bernilai 0, maka $A \otimes \mathbf{x} = \mathbf{b}$ tidak punya penyelesaian. Bila setidaknya pada setiap baris matriks $R_{A,b}$ memuat elemen bernilai 1, maka $\bar{\mathbf{x}}$ adalah suatu penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$.*

Bukti

Tanpa menghilangkan kegeneralitasan, misalkan baris ke- k dari matriks $R_{A,b}$ semua elemennya bernilai 0 dan andaikan bahwa $\tilde{\mathbf{x}}$ adalah suatu penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$. Maka

$$\tilde{x}_j \leq \min_l \{b_l - a_{l,j}\} < (b_k - a_{k,j}).$$

Jadi $\tilde{x}_j + a_{k,j} < b_k$ untuk semua j . Dengan demikian $\tilde{\mathbf{x}}$ tidak memenuhi persamaan ke- k . Hal ini bertentangan dengan $\tilde{\mathbf{x}}$ adalah suatu penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$. Jadi $\tilde{\mathbf{x}}$ bukan penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$ atau persamaan $A \otimes \mathbf{x} = \mathbf{b}$ tidak punya penyelesaian. Berikutnya, andaikan $\bar{\mathbf{x}}$ bukan suatu penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$, maka $\bar{x}_j \leq b_k - a_{k,j}$ untuk semua k, j . Jadi

$$\max_j \{a_{k,j} + \bar{x}_j\} \leq b_k$$

dan bila $\bar{\mathbf{x}}$ bukan penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$, maka ada suatu k dengan

$$\max_j \{a_{k,j} + \bar{x}_j\} < b_k$$

hal ini ekuivalen dengan

$$\bar{x}_j < b_k - a_{k,j}, \text{ untuk semua } j.$$

Karena

$$\bar{x}_j = \min \{b_l - a_{l,j}\}, \text{ untuk beberapa } l,$$

maka tidak ada elemen di baris ke- k pada matriks $R_{A,\mathbf{b}}$ yang bernilai 1. Hal ini bertentangan bahwa setiap baris dari matriks $R_{A,\mathbf{b}}$ setidaknya memuat elemen yang bernilai 1. Jadi haruslah $\bar{\mathbf{x}}$ adalah suatu penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$. ■

Salah satu hasil Teorema 2.4.2 adalah eksistensi dari penyelesaian $A \otimes \mathbf{x} = \mathbf{b}$. Eksistensi ini belum menjelaskan bilamana tunggal dan tidak tunggal. Untuk hal ini diperlukan definisi berikut.

Definisi 2.4.1 *Elemen bernilai 1 pada suatu baris $R_{A,\mathbf{b}}$ dinamakan elemen **peubah tetap** bila*

- *bila nilai 1 hanya satu-satunya pada baris tsb. atau*
- *bila nilai tsb. pada kolom yang sama seperti halnya hanya satu-satunya nilai 1.*

*Sisa nilai 1 lainnya dinamakan elemen **slack**.*

Pada Contoh 1 matriks

$$R_{A,\mathbf{b}} = \begin{bmatrix} 0 & \textcircled{1} & 0 \\ 0 & 0 & \textcircled{1} \\ \textcircled{1} & 0 & 0 \\ \textcircled{1} & 0 & 0 \end{bmatrix}$$

semua elemen $\textcircled{1}$ adalah peubah tetap. Persamaan baris pertama menetapkan elemen $x_2 = 3$ dan persamaan baris kedua menetapkan elemen $x_3 = 4$. Persamaan baris ketiga menetapkan elemen $x_1 = 2$, dalam hal ini ketika persamaan baris keempat dicapai semua elemen x sudah dipilih. Setiap elemen-elemen yang telah dipilih ini tidak bisa diubah, bila diubah yang lain akan membentuk pertaksamaan.

Begitu juga pada Contoh 2, matriks

$$R_{A,\mathbf{b}} = \begin{bmatrix} 0 & 0 & \textcircled{1} \\ 0 & 0 & \textcircled{1} \\ 0 & 1 & \textcircled{1} \\ 1 & 1 & \textcircled{1} \end{bmatrix}$$

semua elemen $\textcircled{1}$ adalah peubah tetap sedangkan semua elemen sisa lainnya yang bernilai 1 adalah elemen *slack*. Ada tiga elemen *slack*. Persamaan baris pertama menetapkan elemen $x_3 = 7$. Elemen penyelesaian persamaan baris kedua sudah ditetapkan oleh persamaan baris pertama. Pada persamaan baris ketiga ada dua cara yang mungkin untuk mencapai persamaan yaitu $x_2 = 4$ atau $x_3 = 7$, Tetapi x_3 sudah ditetapkan sebelumnya sama dengan 7. Jadi asalkan $x_2 \leq 4$ tidak akan mengubah persamaan pada baris di atasnya. Dengan cara yang sama pada persamaan baris keempat asalkan $x_1 \leq 1$ tidak akan mengubah persamaan pada baris di atasnya. Dengan demikian, dengan menetapkan $x_3 = 7$ dan untuk $x_2 \leq 4$ dan $x_1 \leq 1$ persamaan semua baris selalu benar.

Teorema 2.4.3 *Diberikan persamaan $A \otimes \mathbf{x} = \mathbf{b}$ dengan ukuran A adalah $m \times n$, $\mathbf{x}$ berukuran $n \times 1$ dan $\mathbf{b}$ berukuran $m \times 1$ yang semua elemennya berhingga. Tambahan pula, persamaan $A \otimes \mathbf{x} = \mathbf{b}$ mempunyai penyelesaian. Bila setiap baris dari matriks $R_{A,\mathbf{b}}$ hanya ada satu bernilai 1, maka penyelesaian $A \otimes \mathbf{x} = \mathbf{b}$ tunggal. Bila ada elemen *slack* pada matriks $R_{A,\mathbf{b}}$, maka penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$ adalah tidak tunggal.*

Bukti

Bila disetiap baris $R_{A,\mathbf{b}}$ hanya ada satu elemen bernilai 1, maka disetiap baris $R_{A,\mathbf{b}}$ ada suatu elemen peubah tetap dan tidak ada elemen *slack*. Dengan demikian semua elemen $\mathbf{x}$ tetap, jadi penyelesaian tunggal. Selanjutnya, misalkan $r_{i,j}$ adalah satu elemen *slack* pada $R_{A,\mathbf{b}}$ dan $\bar{\mathbf{x}}$ adalah penyelesaian dari $A \otimes \mathbf{x} = \mathbf{b}$. Karena $r_{i,j}$ bukan elemen peubah tetap, maka tidak ada elemen peubah tetap pada kolom ke- j dari matriks $R_{A,\mathbf{b}}$. Jadi persamaan bisa dicapai tanpa menggunakan elemen $\bar{x}_j$. Dengan demikian walaupun nilai $\bar{x}_j$ menunjukkan nilai maksimum yang mungkin untuk elemen ini, setiap nilai yang lebih kecil atau sama dengan $\bar{x}_j$ tidak mempengaruhi eksistensi persamaan baris yang telah ditetapkan.

■

2.4.3 Persamaan $\mathbf{x} = (A \otimes \mathbf{x}) \oplus \mathbf{b}$

Mengikuti Persamaan (2.2) dan (2.3), secara formal untuk sebarang $A \in \mathbb{R}_\varepsilon^{n \times n}$ didefinisikan

$$A^* \stackrel{\text{def}}{=} E \oplus A^+ = \bigoplus_{i=0}^{\infty} A^{\otimes i}. \quad (2.37)$$

Dari hasil Teorema 2.1.2, jelas bahwa A^* ada untuk setiap matriks persegi A dengan graf $\mathcal{G}(A)$ hanya mempunyai bobot sirkuit takpositip. Catatan bahwa $A^{\otimes n}$ merujuk pada bobot

maksimum dari path dengan panjang n . Jadi path ini memuat setidaknya satu sirkuit. Bila semua sirkuit mempunyai bobot sirkuit takpositip, maka

$$[A^{\otimes n}]_{i,j} \leq \bigotimes_{i=0}^{n-1} [A^{\otimes i}]_{i,j}, \quad i, j \in \underline{n},$$

dan kondisi dalam Teorema 2.1.2, A^* bisa ditentukan sebagai berikut

$$A^* = \bigoplus_{i=0}^{n-1} A^{\otimes i}. \quad (2.38)$$

Penyelesaian dari persamaan $\mathbf{x} = (A \otimes \mathbf{x}) \oplus \mathbf{b}$ diberikan dalam teorema berikut.

Teorema 2.4.4 Misalkan $A \in \mathbb{R}_\varepsilon^{n \times n}$ dan $\mathbf{b} \in \mathbb{R}_\varepsilon^n$. Bila bobot rata-rata sirkuit graf $\mathcal{G}(A)$ kurang dari atau sama 0, maka $\mathbf{x} = A^* \otimes \mathbf{b}$ adalah penyelesaian dari $\mathbf{x} = (A \otimes \mathbf{x}) \oplus \mathbf{b}$. Lagipula, bila bobot sirkuit dalam $\mathcal{G}(A)$ adalah negatip, maka penyelesaiannya tunggal.

Bukti

Akan ditunjukkan bahwa

$$A^* \otimes \mathbf{b} = A \otimes (A^* \otimes \mathbf{b}) \oplus \mathbf{b}.$$

Berdasarkan Teorema 2.1.2, A^* ada, sehingga didapat

$$\begin{aligned} A^* \otimes \mathbf{b} &= \bigoplus_{i=0}^{\infty} A^{\otimes i} \otimes \mathbf{b} \\ &= \left(\bigoplus_{i=1}^{\infty} A^{\otimes i} \otimes \mathbf{b} \right) \oplus (E \otimes \mathbf{b}) \\ &= A \otimes \left(\bigoplus_{i=0}^{\infty} A^{\otimes i} \otimes \mathbf{b} \right) \oplus (E \otimes \mathbf{b}) \\ &= A \otimes (A^* \otimes \mathbf{b}) \oplus \mathbf{b}. \end{aligned}$$

Selanjutnya ditunjukkan ketunggalan penyelesaian, misalkan $\mathbf{x}$ adalah penyelesaian $\mathbf{x} = \mathbf{b} \oplus (A \otimes \mathbf{x})$. Substitusikan $\mathbf{x}$ dalam $\mathbf{b} \oplus (A \otimes \mathbf{x})$, yaitu

$$\mathbf{x} = \mathbf{b} \oplus (A \otimes \mathbf{b}) \oplus (A^{\otimes 2} \otimes \mathbf{x}),$$

ulangi lagi proses ini, sehingga didapat

$$\begin{aligned} \mathbf{x} &= \mathbf{b} \oplus (A \otimes \mathbf{b}) \oplus (A^{\otimes 2} \otimes \mathbf{x}) \oplus (A^{\otimes 3} \otimes \mathbf{x}) \\ &= \mathbf{b} \oplus (A \otimes \mathbf{b}) \oplus \dots \oplus (A^{\otimes (i-1)} \otimes \mathbf{b}) \oplus (A^{\otimes i} \otimes \mathbf{x}) \\ &= \bigoplus_{l=0}^{i-1} (A^{\otimes l} \otimes \mathbf{b}) \oplus (A^{\otimes i} \otimes \mathbf{x}). \end{aligned} \quad (2.39)$$

Elemen-elemen $A^{\otimes i}$ adalah bobot maksimum dari path dengan panjang i . Untuk i cukup besar, setiap path memuat satu atau lebih sirkuit elementer turunan sebagai subpath dan kalau i menuju ∞ banyaknya sirkuit elementer yang terjadi menuju ∞ . Karena sirkuit mempunyai bobot negatif, maka elemen-elemen $A^{\otimes i}$ menuju ε untuk i menuju ∞ , yaitu

$$\lim_{i \rightarrow \infty} A^{\otimes i} \otimes \mathbf{x} = \varepsilon.$$

Jadi, untuk i menuju ∞ Persamaan (2.39) menjadi $\mathbf{x} = A^* \otimes \mathbf{b}$, yang mana untuk ini

$$\lim_{i \rightarrow \infty} \bigoplus_{l=0}^{i-1} (A^{\otimes l} \otimes \mathbf{b}) = \left(\lim_{i \rightarrow \infty} \bigoplus_{l=0}^{i-1} A^{\otimes l} \right) \otimes \mathbf{b} = A^* \otimes \mathbf{b}.$$

■

2.5 Eigenmode

Dalam aplikasi aljabar max-plus, terdapat tiga komponen penting berhubungan dengan matriks. Ketiga komponen tersebut adalah nilai eigen, vektor eigen, dan *eigenmode*. Matriks yang dibahas adalah matriks reguler yaitu matriks yang disetiap baris setidaknya memuat satu elemen tidak sama dengan ε . Pada bagian ini diberikan pengertian *eigenmode* sebagaimana definisi berikut.

Definisi 2.5.1 Suatu pasangan vektor $(\boldsymbol{\eta}, \mathbf{v}) \in \mathbb{R}^n \times \mathbb{R}^n$ disebut *eigenmode tergeneralisasi* dari matriks reguler A jika untuk setiap $k \geq 0$ memenuhi

$$A \otimes (k \times \boldsymbol{\eta} + \mathbf{v}) = (k + 1) \times \boldsymbol{\eta} + \mathbf{v}. \quad (2.40)$$

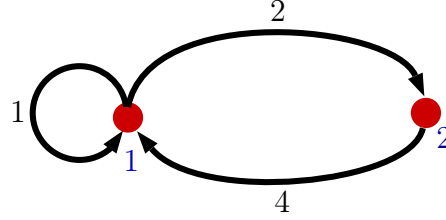
Dalam Persamaan 2.40 bila $k = 0$ didapat

$$A \otimes \mathbf{v} = \boldsymbol{\eta} + \mathbf{v}, \quad (2.41)$$

dan bila semua elemen vektor $\boldsymbol{\eta}$ adalah konstan bernilai $\lambda \in \mathbb{R}$, maka Persamaan 2.41 menjadi

$$A \otimes \mathbf{v} = \lambda \otimes \mathbf{v}. \quad (2.42)$$

Dengan demikian λ adalah suatu nilai eigen dari matriks A yang bersesuaian dengan vektor eigen $\mathbf{v}$. Dalam hal yang demikian vektor $\boldsymbol{\eta}$ adalah perluasan nilai eigen dari matriks A . Dari apa yang telah didiskusikan, terlihat bahwa vektor $\boldsymbol{\eta}$ erat kaitannya dengan vektor *cycle mean*. Jadi bila beberapa elemen dari vektor $\boldsymbol{\eta}$ mempunyai beberapa nilai yang berbeda, maka dipastikan bahwa matriks A dalam Persamaan 2.40 adalah tereduksi atau ekuivalen graf $\mathcal{G}(A)$ tidak *strongly connected*. Tetapi hal ini tidak berlaku sebaliknya. Contoh berikut menjelaskan apa yang telah dibahas berdasarkan Definisi 2.5.1.

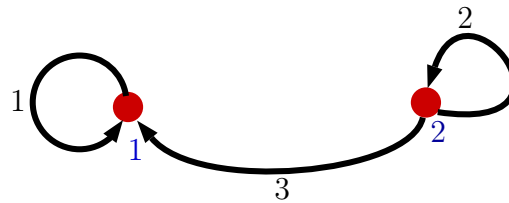
Gambar 2.11: Graf *strongly connected* $\mathcal{G}(A)$

Contoh 2.5.1 Dengan menggunakan matriks reguler $A = \begin{bmatrix} 1 & 4 \\ 2 & \varepsilon \end{bmatrix}$ representasi dari graf pada Gambar 2.11, pasangan vektor $(\boldsymbol{\eta}, \mathbf{v}) \in \mathbb{R}^2 \times \mathbb{R}^2$ yaitu $\left(\begin{bmatrix} 3 \\ 3 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right)$ adalah *eigenmode* tergeneralisasi dari matriks A . Hal tersebut dikarenakan untuk setiap $k \geq 0$, $\boldsymbol{\eta} = \begin{bmatrix} 3 \\ 3 \end{bmatrix}$ dan $\mathbf{v} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ memenuhi

$$A \otimes (k \times \boldsymbol{\eta} + \mathbf{v}) = (k + 1) \times \boldsymbol{\eta} + \mathbf{v}.$$

Contoh yang dibahas ini menunjukkan bahwa graf $\mathcal{G}(A)$ adalah *strongly connected* atau ekuivalen matriks A tak-tereduksi, maka semua komponen vektor $\boldsymbol{\eta}$ bernilai konstan yaitu $\lambda = 3$. □

Contoh 2.5.1 membahas *eigenmode* untuk matriks persegi tak-tereduksi reguler yang menghasilkan semua komponen vektor $\boldsymbol{\eta}$ bernilai konstan yang sama. Untuk matriks persegi tereduksi reguler hasil vektor $\boldsymbol{\eta}$ ada dua kemungkinan yaitu, semua komponen vektornya bernilai konstan yang sama atau beberapa komponen vektornya berbeda. Contoh-contoh berikut menjelaskan hal ini.

Gambar 2.12: Graf tidak *strongly connected* $\mathcal{G}(A)$

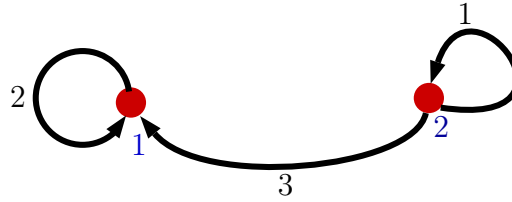
Contoh 2.5.2 Diberikan matriks tereduksi

$$A = \begin{bmatrix} 1 & 3 \\ \varepsilon & 2 \end{bmatrix},$$

dengan representasi graf $\mathcal{G}(A)$ diberikan oleh Gambar 2.12. Maka pasangan vektor $(\boldsymbol{\eta}, \mathbf{v}) \in \mathbb{R}^2 \times \mathbb{R}^2$ yaitu $\left(\begin{bmatrix} 2 \\ 2 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}\right)$ adalah *eigenmode* tergeneralisasi dari matriks A . Hal tersebut dikarenakan untuk setiap $k \geq 0$, $\boldsymbol{\eta} = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$ dan $\mathbf{v} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ memenuhi

$$A \otimes (k \times \boldsymbol{\eta} + \mathbf{v}) = (k + 1) \times \boldsymbol{\eta} + \mathbf{v}.$$

□



Gambar 2.13: Graf tak *strongly connected* $\mathcal{G}(B)$

Contoh 2.5.3 Diberikan matriks tereduksi

$$B = \begin{bmatrix} 2 & 3 \\ \varepsilon & 1 \end{bmatrix},$$

dimana representasi graf tak *strongly connected* $\mathcal{G}(B)$ diberikan oleh Gambar 2.13. Maka pasangan vektor $(\boldsymbol{\eta}, \mathbf{v}) \in \mathbb{R}^2 \times \mathbb{R}^2$ yaitu $\left(\begin{bmatrix} 2 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}\right)$ adalah *eigenmode* tergeneralisasi dari matriks B . Hal tersebut dikarenakan untuk setiap $k \geq 0$, $\boldsymbol{\eta} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$ dan $\mathbf{v} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ memenuhi

$$B \otimes (k \times \boldsymbol{\eta} + \mathbf{v}) = (k + 1) \times \boldsymbol{\eta} + \mathbf{v}.$$

□

Selanjutnya dibahas sifat *eigenmode* dari matriks tak-tereduksi. Pembahasan sifat *eigenmode* dari matriks tak-tereduksi diawali dengan menunjukkan bahwa *eigenmode* dari matriks reguler dapat dipandang sebagai suatu perluasan dari pasangan nilai eigen maupun vektor eigen. Pernyataan tersebut dapat ditulis dalam bentuk lemma berikut.

Lemma 2.5.1 *Bila pasangan vektor $(\boldsymbol{\eta}, \mathbf{v})$ adalah eigenmode tergeneralisasi dari matriks reguler A , maka vektor $\boldsymbol{\eta}$ merupakan perluasan nilai eigen dari matriks A dan vektor $\mathbf{v}$ adalah vektor eigennya. Lebih lanjut, vektor $\boldsymbol{\eta} = \lim_{k \rightarrow \infty} \frac{\mathbf{x}^{(k)}}{k}$, dimana semua elemen vektor $\boldsymbol{\eta}$ bernilai konstan di $\mathbb{R}$.*

Bukti Misalkan pasangan vektor $(\boldsymbol{\eta}, \mathbf{v})$ adalah *eigenmode* tergeneralisasi dari matriks reguler A , maka vektor $\boldsymbol{\eta}$ dan $\mathbf{v}$ memenuhi

$$A \otimes (k \times \boldsymbol{\eta} + \mathbf{v}) = (k + 1) \times \boldsymbol{\eta} + \mathbf{v}. \quad (2.43)$$

Berikutnya, misalkan $\lambda \in \mathbb{R}$ adalah nilai eigen dari matriks reguler A dan vektor eigen yang bersesuaian dengan λ sama dengan $\mathbf{v} \in \mathbb{R}^n$, maka λ dan $\mathbf{v}$ memenuhi

$$A \otimes \mathbf{v} = \lambda \otimes \mathbf{v}. \quad (2.44)$$

Untuk setiap $k \geq 0$, kalikan kedua ruas Persamaan (2.44) dengan $\lambda^{\otimes k}$ diperoleh

$$\begin{aligned} A \otimes \lambda^{\otimes k} \otimes \mathbf{v} &= \lambda^{\otimes k} \otimes \lambda \otimes \mathbf{v} \\ A \otimes \lambda^{\otimes k} \otimes \mathbf{v} &= \lambda^{\otimes (k+1)} \otimes \mathbf{v} \\ A \otimes (k \times \mathbf{u}[\lambda] + \mathbf{v}) &= (k + 1) \times \mathbf{u}[\lambda] + \mathbf{v}, \end{aligned} \quad (2.45)$$

dimana $\mathbf{u}[\lambda]$ adalah vektor dengan ukuran yang sesuai dan semua elemennya bernilai λ . Berdasarkan Persamaan (2.43) dan (2.45) didapatkan vektor $\boldsymbol{\eta} = \mathbf{u}[\lambda]$ merupakan perluasan nilai eigen matriks reguler A , yaitu $\boldsymbol{\eta}$ adalah vektor yang memiliki nilai λ untuk setiap elemennya. Sedangkan vektor $\mathbf{v}$ adalah vektor eigen matriks reguler A yang bersesuaian dengan nilai eigen λ .

Lebih lanjut, dengan mempertimbangkan kembali persamaan rekurensi

$$\mathbf{x}(k + 1) = A \otimes \mathbf{x}(k),$$

dengan $\mathbf{x}_0 = \mathbf{v}$ dan menggunakan definisi *eigenmode* tergeneralisasi didapat

$$\mathbf{x}(k) = k \times \boldsymbol{\eta} + \mathbf{v}.$$

Dengan demikian karena semua elemen vektor $\mathbf{v}$ berhingga, didapat

$$\begin{aligned} \lim_{k \rightarrow \infty} \frac{\mathbf{x}(k)}{k} &= \lim_{k \rightarrow \infty} \frac{k \times \boldsymbol{\eta} + \mathbf{v}}{k} \\ &= \lim_{k \rightarrow \infty} \frac{k \times \boldsymbol{\eta}}{k} + \lim_{k \rightarrow \infty} \frac{\mathbf{v}}{k} \\ &= \lim_{k \rightarrow \infty} \boldsymbol{\eta} + \mathbf{0} \\ &= \boldsymbol{\eta} + \mathbf{0} \\ &= \boldsymbol{\eta}, \end{aligned}$$

dengan vektor $\mathbf{0} = (0 \ 0 \ \dots \ 0)^T$. □

Matriks tak-tereduksi juga merupakan bagian dari matriks reguler. Oleh karena itu, Lemma 2.5.1 juga berlaku untuk kasus matriks tak-tereduksi. Pada Lemma 2.5.1 telah dijelaskan bahwa pasangan vektor $(\boldsymbol{\eta}, \mathbf{v})$ merupakan perluasan dari nilai eigen dan vektor

eigen. Hal tersebut menyebabkan sifat dari *eigenmode* matriks tak-tereduksi dipengaruhi oleh sifat nilai eigen dan vektor eigennya.

Sifat pertama yang dibahas adalah mengenai eksistensi *eigenmode* dari matriks tak-tereduksi. Eksistensi *eigenmode* matriks tak-tereduksi diberikan pada teorema berikut ini.

Teorema 2.5.1 *Jika $A \in \mathbb{R}_\varepsilon^{n \times n}$ adalah matriks tak-tereduksi dengan nilai eigen λ dan vektor eigen $\mathbf{v}$, maka terdapat pasangan vektor $(\boldsymbol{\eta}, \mathbf{v})$ yang merupakan *eigenmode* dari matriks tak-tereduksi A .*

Bukti Teorema 2.1.4 telah membuktikan bahwa jika A adalah matriks tak-tereduksi, maka terdapat λ yang merupakan nilai eigen dari matriks A . Juga, Kesimpulan 2.3.1 membuktikan bahwa jika A adalah matriks tak-tereduksi, maka dapat ditemukan vektor eigen $\mathbf{v}$ yang bersesuaian dengan nilai eigen λ . Karena vektor $\boldsymbol{\eta}$ dalam *eigenmode* merupakan perluasan nilai eigen matriks tak-tereduksi A , yaitu $\boldsymbol{\eta}$ adalah vektor yang memiliki nilai λ untuk setiap elemennya dan vektor $\mathbf{v}$ dalam *eigenmode* merupakan vektor eigen yang bersesuaian dengan nilai eigen λ , maka jelas untuk setiap matriks tak-tereduksi A dapat ditemukan pasangan vektor $(\boldsymbol{\eta}, \mathbf{v})$ yang merupakan *eigenmode* dari matriks tak-tereduksi A . $\square$

Teorema 2.5.1 menunjukkan sifat eksistensi *eigenmode* matriks tak-tereduksi. Adapun sifat lain dari *eigenmode* matriks tak-tereduksi diberikan pada teorema berikut.

Teorema 2.5.2 *Untuk setiap matriks tak-tereduksi $A \in \mathbb{R}_\varepsilon^{n \times n}$ memiliki *eigenmode* yang tidak tunggal.*

Bukti Misalkan matriks tak-tereduksi A memiliki nilai eigen λ , vektor eigen $\mathbf{v}$ yang bersesuaian dengan nilai eigen λ , dan *eigenmode* berupa pasangan vektor $(\boldsymbol{\eta}, \mathbf{v})$. Dalam Teorema 2.1.5 dibuktikan bahwa nilai eigen matriks tak-tereduksi A adalah tunggal yang menyebabkan vektor $\boldsymbol{\eta} = (\lambda \ \lambda \ \dots \ \lambda)^T$ adalah tunggal. Akan tetapi, Teorema 2.3.3 telah membuktikan bahwa vektor eigen $\mathbf{v}$ yang bersesuaian dengan nilai eigen λ dari matriks tak-tereduksi tidak tunggal, yaitu untuk sebarang $\alpha \in \mathbb{R}$, $\alpha \otimes \mathbf{v}$ adalah vektor eigen yang bersesuaian dengan nilai eigen λ . Sehingga *eigenmode* dari matriks tak-tereduksi A juga tidak tunggal yaitu $(\boldsymbol{\eta}, \alpha \otimes \mathbf{v})$ dengan $\alpha \in \mathbb{R}$. $\square$

Pembahasan berikutnya adalah mengenai karakterisasi *eigenmode* dari matriks tereduksi reguler. Sebelum membahas eksistensi *eigenmode* dari matriks tereduksi reguler, diingatkan lagi hasil pembahasan mengenai penyelesaian persamaan nonhomogin

$$\mathbf{x} = (A \otimes \mathbf{x}) \oplus \mathbf{b}. \quad (2.46)$$

yang telah dibahas dalam Subbagian 2.4.3. Berdasarkan Teorema 2.4.4, bila bobot rata-rata sirkuit graf $\mathcal{G}(A)$ kurang dari atau sama dengan 0 maka Persamaan 2.46 mempunyai penyelesaian

$$\mathbf{x} = A^* \otimes \mathbf{b},$$

dengan $A^* \stackrel{\text{def}}{=} E \oplus A^+ = \bigoplus_{i=0}^{\infty} A^{\otimes i}$. Lagipula, bila bobot sirkuit dalam $\mathcal{G}(A)$ adalah negatif, maka penyelesaiannya tunggal.

Adapun persamaan rekurensi nonhomogen merupakan perluasan dari persamaan rekurensi linear $\mathbf{x}(k+1) = A \otimes \mathbf{x}(k)$. Berikut diberikan teorema mengenai persamaan rekurensi nonhomogen.

Teorema 2.5.1 *Diberikan persamaan rekurensi nonhomogen berikut*

$$\mathbf{x}(k+1) = A \otimes \mathbf{x}(k) \oplus \left(\bigoplus_{j=1}^m B_j \otimes \mathbf{u}_j(k) \right), \quad (2.47)$$

dengan matriks $A \in \mathbb{R}_{\varepsilon}^{n \times n}$ adalah matriks tak-tereduksi yang memiliki nilai eigen λ atau $A \in \mathbb{R}_{\varepsilon}$ yaitu $A = \varepsilon$ dengan $\lambda = \varepsilon$, matriks $B_j \in \mathbb{R}_{\varepsilon}^{n \times m_j}$ dengan $m_j \geq 1$ memenuhi $B_j \neq \mathcal{E}$, sedangkan $\mathbf{u}_j(k) \in \mathbb{R}_{\varepsilon}^{m_j}$ memenuhi $\mathbf{u}_j(k) = \tau_j^k \otimes \mathbf{w}_j(k)$, $k \geq 0$ dengan $\mathbf{w}_j \in \mathbb{R}_{\varepsilon}^{m_j}$ dan $\tau_j \in \mathbb{R}$. Maka untuk suatu $\tau = \bigoplus_{j \in \underline{m}} \tau_j$ terdapat bilangan bulat $K \geq 0$ dan vektor $\mathbf{v} \in \mathbb{R}^n$ sedemikian hingga barisan $\mathbf{x}(k) = \mu^{\otimes k} \otimes \mathbf{v}$, dengan $\mu = \lambda \otimes \tau$ memenuhi persamaan rekurensi (2.47) untuk setiap $k \geq K$.

Bukti Pembuktian diberikan dengan mempertimbangkan dua kemungkinan kasus, yaitu $\lambda > \tau$ dan $\lambda \leq \tau$.

Kasus $\lambda > \tau$. Karena matriks A tak-tereduksi, maka A memiliki nilai eigen dan vektor eigen dengan nilai berhingga. Misal ambil vektor eigen $\mathbf{v}$ dari A sedemikian hingga $\mathbf{v} \otimes \lambda > \bigoplus_{j=1}^m B_j \otimes \mathbf{w}_j$. Selanjutnya, ambil $\mu = \lambda > \tau_j$ untuk setiap $j \in \underline{m}$, sedemikian hingga untuk setiap $k \geq 0$ memenuhi

$$\mu \otimes \mathbf{v} \otimes \mu^{\otimes k} = A \otimes \mathbf{v} \otimes \mu^{\otimes k} > \bigoplus_{j=1}^m B_j \otimes \mathbf{w}_j \otimes \mu^{\otimes k} \geq \bigoplus_{j=1}^m B_j \otimes \mathbf{w}_j \otimes \tau_j^{\otimes k}.$$

Karena $\mu \otimes \mu^{\otimes k} = \mu^{\otimes(k+1)}$, maka diperoleh

$$\mathbf{v} \otimes \mu^{\otimes(k+1)} = A \otimes \mathbf{v} \otimes \mu^{\otimes k} > \bigoplus_{j=1}^m B_j \otimes \mathbf{w}_j \otimes \tau_j^{\otimes k}.$$

Jadi persamaan rekurensi (2.47) terpenuhi untuk setiap $k \geq 0$ dengan $\mathbf{x}(k) = \mathbf{v} \otimes \mu^{\otimes k}$, dan $\mathbf{u}_j(k) = \mathbf{w}_j \otimes \tau_j^{\otimes k}$ untuk $j \in \underline{m}$.

Kasus $\lambda \leq \tau$. Misalkan $\tau = \bigoplus_{j \in \underline{m}} \tau_j$ dan nilai maksimum tersebut diperoleh untuk τ ke- r pertama, yang akan terpenuhi dengan penomoran ulang barisan $\mathbf{u}_j(k)$, $j \in \underline{m}$. Selanjutnya, ambil vektor $\mathbf{v}$ yang memenuhi

$$\mathbf{v} = A_{\tau} \otimes \mathbf{v} \oplus \bigoplus_{j=1}^r (B_j)_{\tau} \otimes \mathbf{w}_j, \quad (2.48)$$

dengan A_τ dan $(B_j)_\tau$, $j \in \underline{m}$ didapatkan dari matriks A dan (B_j) dengan semua elemen berhingganya dikurangi dengan τ . Karena $\lambda \leq \tau$, maka graf komunikasi dari A_τ hanya memuat sirkuit dengan bobot tidak positif. Oleh karena itu, dengan Teorema 2.4.4 penyelesaian dari $\mathbf{v}$ ada, kemudian karena seluruh elemen dari $(A_\tau)^*$ berhingga (karena A_τ adalah matriks tak-tereduksi) dan $\bigoplus_{j=1}^r (B_j)_\tau \otimes \mathbf{w}_j$ memuat setidaknya satu elemen berhingga, maka elemen-elemen dari $\mathbf{v}$ berhingga. Oleh karena itu, Persamaan (2.48) menjadi

$$\mathbf{v} \otimes \tau = A \otimes \mathbf{v} \oplus \bigoplus_{j=1}^r B_j \otimes \mathbf{w}_j.$$

Berikutnya, dengan mengambil $\mu = \tau = \tau_j$, $j = 1, 2, \dots, r$, maka untuk setiap $k \geq 0$ memenuhi

$$\mathbf{v} \otimes \mu^{\otimes(k+1)} = A \otimes \mathbf{v} \otimes \mu^{\otimes k} \oplus \bigoplus_{j=1}^r B_j \otimes \mathbf{w}_j \otimes \tau_j^{\otimes k},$$

sehingga

$$\mathbf{v} \otimes \mu^{\otimes(k+1)} \leq A \otimes \mathbf{v} \otimes \mu^{\otimes k} \oplus \bigoplus_{j=1}^m B_j \otimes \mathbf{w}_j \otimes \tau_j^{\otimes k}. \quad (2.49)$$

Akan tetapi, karena $\mu > \tau_j$ untuk $j = r+1, r+2, \dots, m$ terdapat bilangan bulat $K \geq 0$ sebesar yang dibutuhkan sedemikian hingga untuk setiap $k \geq K$ memenuhi

$$\mathbf{v} \otimes \mu^{\otimes(k+1)} \geq \bigoplus_{j=r+1}^m B_j \otimes \mathbf{w}_j \otimes \tau_j^{\otimes k}. \quad (2.50)$$

Dari Pertidaksamaan (2.49) dan (2.50) diperoleh persamaan rekurensi (2.47) terpenuhi untuk setiap $k \geq K$. $\square$

Pada pembahasan sebelumnya, diketahui bahwa matriks tereduksi A dapat disajikan dalam bentuk matriks blok segitiga atas diberikan oleh bentuk 2.16, dengan blok matriks $A_{i,i}$ adalah matriks tak tereduksi sehingga $\lambda_i = \lambda(A_{i,i})$ atau $A_{i,i} = \varepsilon$ sehingga $\lambda_i = \varepsilon$. Selanjutnya, misal diambil vektor $\mathbf{x}(k)$ yang bersesuaian dengan matriks blok segitiga atas (2.16), yaitu

$$\mathbf{x}(k) = \begin{pmatrix} \mathbf{x}_1(k) \\ \mathbf{x}_2(k) \\ \vdots \\ \mathbf{x}_q(k) \end{pmatrix}.$$

Matriks blok segitiga atas dari matriks tereduksi A memenuhi Persamaan rekurensi (2.47), yaitu:

$$\mathbf{x}_i(k+1) = A_{i,i} \otimes \mathbf{x}_i(k) \oplus \bigoplus_{j=i+1}^q A_{i,j} \otimes \mathbf{x}_j(k); \quad i \in \underline{q}, \quad k \geq 0. \quad (2.51)$$

Secara khusus untuk $i = q$, Persamaan (2.51) menjadi $\mathbf{x}_q(k+1) = A_{q,q} \otimes \mathbf{x}_q(k)$. Jika diasumsikan A adalah matriks reguler, maka $A_{q,q}$ juga reguler. Jadi $A_{q,q} \neq \mathcal{E}$, sehingga subgraf *strongly connected* maksimal yang bersesuaian dengan matriks $A_{q,q}$ memiliki himpunan *arc* yang tidak kosong, akibatnya $A_{q,q}$ adalah matriks tak tereduksi. Oleh karena itu, terdapat suatu vektor dengan semua elemen berhingga $\mathbf{v}_q$ dan suatu skalar $\xi_q \in \mathbb{R}$ sedemikian hingga

$$\mathbf{x}_q(k) = \xi_q^{\otimes k} \otimes \mathbf{v}_q$$

memenuhi $\mathbf{x}_q(k+1) = A_{q,q} \otimes \mathbf{x}_q(k)$ untuk setiap $k \geq 0$. Dalam hal ini, $\mathbf{v}_q$ adalah vektor eigen dari matriks $A_{q,q}$ yang bersesuaian dengan nilai eigen $\lambda_q = \lambda(A_{q,q})$, dengan $\xi_q = \lambda_q$.

Adapun untuk $i \in \underline{q}$ secara umum diberikan pada teorema berikut.

Teorema 2.5.2 *Jika dalam Persamaan (2.51) matriks $A_{q,q}$ adalah matriks tak tereduksi, dan untuk $i \in \underline{q-1}$ matriks $A_{i,i}$ adalah matriks tak tereduksi atau $A_{i,i} = \varepsilon$, maka terdapat skalar $\xi_1, \xi_2, \dots, \xi_q \in \mathbb{R}$ dan vektor $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_q$ dengan seluruh elemen vektor berhingga sedemikian hingga*

$$\mathbf{x}_i(k) = \xi_i^{\otimes k} \otimes \mathbf{v}_i, \quad i \in \underline{q}$$

memenuhi Persamaan rekurensi (2.51) untuk setiap $k \geq 0$. Skalar $\xi_1, \xi_2, \dots, \xi_q$ ditentukan dengan

$$\xi_i = \bigoplus_{j \in \mathcal{H}_i} \xi_j \oplus \lambda_i,$$

dengan $\mathcal{H}_i = \{j \in \underline{q} : j > i, A_{i,j} \neq \mathcal{E}\}$.

Bukti Untuk $i = q$ telah dibuktikan terdapat vektor $\mathbf{v}_q$ dengan semua elemen berhingga dan suatu skalar $\xi_q \in \mathbb{R}$ sedemikian hingga $\mathbf{x}_q(k) = \xi_q^{\otimes k} \otimes \mathbf{v}_q$, dengan $\xi_q = \lambda_q$ memenuhi $\mathbf{x}_q(k+1) = A_{q,q} \otimes \mathbf{x}_q(k)$ untuk setiap $k \geq 0$.

Selanjutnya, diasumsikan benar untuk suatu $l+1$, dengan $1 < l+1 \leq q$. Sehingga, terdapat vektor $\mathbf{v}_{l+1}, \dots, \mathbf{v}_q$ dengan seluruh elemen vektor berhingga dan skalar $\xi_{l+1}, \dots, \xi_q \in \mathbb{R}$ sedemikian hingga untuk $k \geq 0$

$$\mathbf{x}_i(k) = \xi_i^{\otimes k} \otimes \mathbf{v}_i, \quad l+1 \leq i \leq q,$$

memenuhi

$$\mathbf{x}_i(k+1) = A_{i,i} \otimes \mathbf{x}_i(k) \oplus \bigoplus_{j=i+1}^q A_{i,j} \otimes \mathbf{x}_j(k), \quad l+1 \leq i \leq q.$$

Kemudian, akan dibuktikan kebenaran pernyataan untuk suatu l , dengan $l \leq i \leq q$. Perhatikan persamaan

$$\mathbf{x}_l(k+1) = A_{l,l} \otimes \mathbf{x}_l(k) \oplus \bigoplus_{j=l+1}^q A_{l,j} \otimes \mathbf{x}_j(k), \quad (2.52)$$

dengan $A_{l,l}$ adalah matriks tak tereduksi atau matriks berukuran 1×1 sama dengan ε . Jika

$$\bigoplus_{j=l+1}^q A_{l,j} \otimes \mathbf{x}_j(k) = \bigoplus_{j \in \mathcal{H}_l} A_{l,j} \otimes \mathbf{x}_j(k),$$

dengan $\mathcal{H}_l = \{j \in \underline{q} : j > l, A_{l,j} \neq \mathcal{E}\}$, maka Persamaan rekurensi (2.52) dapat ditulis

$$\mathbf{x}_l(k+1) = A_{l,l} \otimes \mathbf{x}_l(k) \oplus \bigoplus_{j \in \mathcal{H}_l} A_{l,j} \otimes \mathbf{x}_j(k). \quad (2.53)$$

Persamaan rekurensi (2.53) akan terpenuhi dengan Teorema 2.5.1. Oleh karena itu, dari Teorema 2.5.1 terdapat bilangan bulat $K_l \geq 0$ dan suatu vektor atau skalar $\mathbf{v}_l$ dengan elemen berhingga, sedemikian hingga

$$\mathbf{x}_l(k) = \xi_l^{\otimes k} \otimes \mathbf{v}_l$$

memenuhi Persamaan rekurensi (2.53) untuk setiap $k \geq K_l$, dengan

$$\xi_l = \bigoplus_{j \in \mathcal{H}_l} \xi_j \oplus \lambda_l.$$

Berikutnya, dengan mendefinisikan

$$\mathbf{v}_i := \xi_i^{\otimes K_l} \otimes \mathbf{v}_i, \quad l \leq i \leq q,$$

didapat

$$\mathbf{x}_i(k) = \xi_i^{\otimes K_l} \otimes \mathbf{v}_i, \quad l \leq i \leq q,$$

yang memenuhi

$$\mathbf{x}_i(k+1) = A_{i,i} \otimes \mathbf{x}_i(k) \oplus \bigoplus_{j=i+1}^q A_{i,j} \otimes \mathbf{x}_j(k), \quad l \leq i \leq q.$$

Jadi, dapat dibuktikan kebenaran pernyataan untuk suatu l , dengan $l \leq i \leq q$ sehingga teorema terbukti. $\square$

Teorema 2.5.2 menghasilkan akibat yang menunjukkan eksistensi *eigenmode* dari matriks tereduksi reguler.

Akibat 2.5.1 *Jika $A \in \mathbb{R}_\varepsilon^{n \times n}$ adalah matriks tereduksi reguler, maka terdapat pasangan vektor $(\boldsymbol{\eta}, \mathbf{v}) \in \mathbb{R}^n \times \mathbb{R}^n$ yang merupakan eigenmode tergeneralisasi dari matriks A , sedemikian hingga untuk setiap $k \geq 0$:*

$$A \otimes (k \times \boldsymbol{\eta} + \mathbf{v}) = (k+1) \times \boldsymbol{\eta} + \mathbf{v}.$$

Bukti Pertama, dengan penomoran ulang mengubah Persamaan rekurensi (2.13) menjadi Persamaan rekurensi (2.51). Karena matriks A adalah matriks reguler, blok matriks pada bagian diagonal matriks blok segitiga atas dari A memenuhi kondisi Teorema 2.5.2. Perhatikan bahwa $\mathbf{x}_i(k) = \xi_i^{\otimes k} \otimes \mathbf{v}_i$ yang dihasilkan dari Teorema 2.5.2 juga dapat ditulis sebagai

$$\mathbf{x}_i(k) = k \times \mathbf{u}[\xi_i] + \mathbf{v}_i.$$

Selanjutnya, susun $\mathbf{u}[\xi_i]$ ke atas mulai dari $i = q$ sampai $i = 1$ sehingga terbentuk vektor $\boldsymbol{\eta}$. Begitu pula, susun $\mathbf{v}_i$ ke atas mulai dari $i = q$ sampai $i = 1$ sehingga terbentuk vektor $\mathbf{v}$, sedemikian hingga

$$\mathbf{x}(k) = k \times \boldsymbol{\eta} + \mathbf{v}.$$

Sehingga Persamaan rekurensi

$$\mathbf{x}(k+1) = A \otimes \mathbf{x}(k),$$

untuk $k = 0, 1, 2, \dots$ menjadi

$$(k+1) \times \boldsymbol{\eta} + \mathbf{v} = A \otimes (k \times \boldsymbol{\eta} + \mathbf{v}),$$

untuk $k = 0, 1, 2, \dots$. Jadi pasangan vektor $(\boldsymbol{\eta}, \mathbf{v})$ adalah *eigenmode* dari matriks tereduksi reguler. □

Akibat 2.5.1 menunjukkan eksistensi *eigenmode* dari matriks tereduksi reguler. Sehingga vektor waktu siklus ada untuk setiap matriks tereduksi reguler. Vektor waktu siklus dari matriks A dalam bentuk matriks blok segitiga atas diberikan sebagai berikut

$$\boldsymbol{\eta} = \left(\mathbf{u}^T[\xi_1] \quad \mathbf{u}^T[\xi_2] \quad \dots \quad \mathbf{u}^T[\xi_q] \right)^T.$$

Untuk matriks tereduksi reguler A tidak dalam bentuk matriks blok segitiga atas, vektor waktu siklus adalah dengan mengubah vektor waktu siklus sesuai bentuk matriks blok segitiga atas dari matriks A .

Akibat 2.5.1 menginspirasi penyusunan algoritma *eigenmode* tergeneralisasi untuk matriks tereduksi reguler. Berikut ini langkah-langkah algoritma tersebut.

Algoritma 2.5.1 ALGORITMA EIGENMODE MATRIKS TEREDUKSI REGULER

1. Ambil matriks tereduksi reguler $A \in \mathbb{R}_{\varepsilon}^{n \times n}$.
2. Tentukan bentuk matriks blok segitiga atas dari matriks A .
3. Hitung nilai eigen dan vektor eigen dari blok matriks terakhir pada diagonal utama matriks blok segitiga atas dari matriks A . Misal $A_{q,q}$, maka hitung nilai eigen $\lambda_q = \lambda(A_{q,q})$ dan vektor eigen $\mathbf{v}_q$ yang bersesuaian dengan nilai eigen tersebut. Selanjutnya, ambil $\xi_q = \lambda_q$ dan $i = q$.

4. Hitung nilai eigen $\lambda_{(i-1)}$ dari matriks $A_{(i-1),(i-1)}$.
5. Jika $\lambda_{(i-1)} > \xi_i$ lanjutkan ke langkah 6, jika tidak ke langkah 7.
6. Ambil $\xi_{(i-1)} = \lambda_{(i-1)}$ dan hitung vektor $\mathbf{v}_{(i-1)}$ melalui persamaan berikut:

$$\xi_{(i-1)} \otimes \mathbf{v}_{(i-1)} = A_{(i-1),(i-1)} \otimes \mathbf{v}_{(i-1)} \oplus \bigoplus_{j=i}^q A_{(i-1),j} \otimes \mathbf{v}_j.$$

Kemudian, lanjutkan ke langkah 8.

7. Ambil $\xi_{(i-1)} = \lambda_i$ dan hitung vektor $\mathbf{v}_{(i-1)}$ melalui persamaan berikut:

$$\lambda_i \otimes \mathbf{v}_{(i-1)} = A_{(i-1),(i-1)} \otimes \mathbf{v}_{(i-1)} \oplus \bigoplus_{j=i}^q A_{(i-1),j} \otimes \mathbf{v}_j.$$

Kemudian, lanjutkan ke langkah 8.

8. Jika $i - 1 \neq 1$ kembali ke langkah 4, jika tidak maka selesai.

Jadi didapatkan eigenmode tergeneralisasi dari matriks A adalah pasangan vektor $(\boldsymbol{\eta}, \mathbf{v})$, dengan $\boldsymbol{\eta} = (\xi_1 \ \xi_2 \ \dots \ \xi_q)^T$ dan $\mathbf{v} = (\mathbf{v}_1 \ \mathbf{v}_2 \ \dots \ \mathbf{v}_q)^T$.

Berikut diberikan contoh perhitungan *eigenmode* dari matriks tereduksi reguler.

Contoh 2.5.4 Diberikan matriks tereduksi reguler yang memiliki bentuk matriks blok segitiga atas sebagai berikut:

$$A = \begin{bmatrix} A_{1,1} & A_{1,2} \\ \mathcal{E} & A_{2,2} \end{bmatrix},$$

dengan

$$A_{1,1} = \begin{bmatrix} \varepsilon & 3 \\ 1 & \varepsilon \end{bmatrix}, \quad A_{1,2} = \begin{bmatrix} 2 & \varepsilon \\ \varepsilon & \varepsilon \end{bmatrix}, \quad \mathcal{E} = \begin{bmatrix} \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \end{bmatrix} \quad \text{dan} \quad A_{2,2} = \begin{bmatrix} \varepsilon & 4 \\ 4 & \varepsilon \end{bmatrix}.$$

Berikut dihitung *eigenmode* dari matriks A dengan menggunakan Algoritma 2.5.1. Pertama, dilakukan perhitungan nilai eigen dan vektor eigen untuk $A_{2,2}$ dengan menggunakan Algoritma *Power*. Untuk keadaan awal $\mathbf{x}(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ diperoleh evolusi keadaan

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} 4 \\ 4 \end{bmatrix}, \quad \dots$$

Sehingga diperoleh nilai eigen dari matriks $A_{2,2}$ adalah $\lambda_2 = 4$, dengan vektor eigen yang bersesuaian

$$\begin{aligned} \mathbf{v}_2 &= \bigoplus_{i=1} \left(\lambda^{\otimes(1-i)} \otimes \mathbf{x}(i-1) \right) \\ &= \lambda^{\otimes(0)} \otimes \mathbf{x}(0) \\ &= \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \end{aligned}$$

Kemudian diambil $\xi_2 = \lambda_2 = 4$. Langkah selanjutnya dihitung nilai eigen dari matriks $A_{1,1}$. Dengan menggunakan Algoritma *Power* didapat $\lambda_1 = 2$. Karena $\lambda_1 < \xi_2$, maka $\xi_1 = \lambda_2 = 4$, dan dilakukan perhitungan vektor eigen $\mathbf{v}_1$ sebagai berikut:

$$\begin{aligned} \lambda_2 \otimes \mathbf{v}_1 &= (A_{1,1} \otimes \mathbf{v}_1) \oplus (A_{1,2} \otimes \mathbf{v}_2) \\ 4 \otimes \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} &= \left(\begin{bmatrix} \varepsilon & 3 \\ 1 & \varepsilon \end{bmatrix} \otimes \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \right) \oplus \left(\begin{bmatrix} 2 & \varepsilon \\ \varepsilon & \varepsilon \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right) \end{aligned} \quad (2.54)$$

dari Persamaan (2.54) diperoleh

$$\begin{aligned} 4 + v_1 &= \max\{3 + v_2, 2\} \\ 4 + v_1 &= 2 \\ v_1 &= -2, \end{aligned}$$

dan

$$\begin{aligned} 4 + v_2 &= \max\{1 + v_1, \varepsilon\} \\ 4 + v_2 &= \max\{1 + (-2), \varepsilon\} \\ v_2 &= -5. \end{aligned}$$

Jadi, didapatkan $\mathbf{v}_1 = \begin{bmatrix} -2 \\ -5 \end{bmatrix}$. Oleh karena itu, pasangan vektor $(\boldsymbol{\eta}, \mathbf{v})$ dengan $\boldsymbol{\eta} = (4 \ 4 \ 4 \ 4)^T$ dan $\mathbf{v} = (-2 \ -5 \ 0 \ 0)^T$ adalah *eigenmode* tergeneralisasi dari matriks tereduksi reguler A sebab untuk $k = 0$, memenuhi:

$$A \otimes (0 \times \boldsymbol{\eta} + \mathbf{v}) = \begin{pmatrix} 2 & -1 & 4 & 4 \end{pmatrix}^T = 1 \times \boldsymbol{\eta} + \mathbf{v},$$

untuk $k = 1$, memenuhi:

$$A \otimes (1 \times \boldsymbol{\eta} + \mathbf{v}) = \begin{pmatrix} 6 & 3 & 8 & 8 \end{pmatrix}^T = 2 \times \boldsymbol{\eta} + \mathbf{v},$$

dan seterusnya, vektor $\boldsymbol{\eta}$ dan $\mathbf{v}$ untuk $k = 0, 1, 2, \dots$ memenuhi

$$A \otimes (k \times \boldsymbol{\eta} + \mathbf{v}) = (k + 1) \times \boldsymbol{\eta} + \mathbf{v}.$$

□

Eigenmode dari matriks tereduksi reguler tidak tunggal. Ketidaktunggalan *eigenmode* dari matriks tereduksi reguler disebabkan ketidaktunggalan vektor $\mathbf{v}$.

Teorema 2.5.3 Untuk setiap matriks tereduksi reguler $A \in \mathbb{R}_\varepsilon^{n \times n}$ memiliki *eigenmode* yang tidak tunggal, yaitu jika $(\boldsymbol{\eta}, \mathbf{v})$ adalah *eigenmode* dari matriks A , maka $(\boldsymbol{\eta}, \alpha \otimes \mathbf{v})$ dengan $\alpha \in \mathbb{R}$ juga merupakan *eigenmode* dari matriks A .

Bukti Misalkan $(\boldsymbol{\eta}, \mathbf{v})$ adalah *eigenmode* dari matriks tereduksi reguler $A \in \mathbb{R}_\varepsilon^{n \times n}$ maka untuk setiap $k \geq 0$, $\boldsymbol{\eta}$ dan $\mathbf{v}$ memenuhi

$$A \otimes (k \times \boldsymbol{\eta} + \mathbf{v}) = (k + 1) \times \boldsymbol{\eta} + \mathbf{v}. \quad (2.55)$$

Kalikan sebarang skalar $\alpha \in \mathbb{R}$ pada kedua ruas Persamaan (2.55) diperoleh

$$\alpha \otimes A \otimes (k \times \boldsymbol{\eta} + \mathbf{v}) = \alpha \otimes (k + 1) \times \boldsymbol{\eta} + \mathbf{v}. \quad (2.56)$$

Karena operasi $\otimes$ komutatif, maka Persamaan (2.56) menjadi

$$\begin{aligned} A \otimes (\alpha \otimes k \times \boldsymbol{\eta} + \mathbf{v}) &= (k + 1) \times \boldsymbol{\eta} + \alpha \otimes \mathbf{v} \\ A \otimes (k \times \boldsymbol{\eta} + (\alpha \otimes \mathbf{v})) &= (k + 1) \times \boldsymbol{\eta} + (\alpha \otimes \mathbf{v}). \end{aligned}$$

Diperoleh $(\boldsymbol{\eta}, \alpha \otimes \mathbf{v})$ untuk sebarang $\alpha \in \mathbb{R}$ juga merupakan *eigenmode* dari matriks tereduksi reguler A . □

Pembahasan terakhir untuk karakterisasi *eigenmode* matriks tereduksi reguler adalah mengenai nilai dari elemen vektor dalam *eigenmode*. *Eigenmode* matriks tereduksi reguler memiliki elemen berhingga untuk setiap komponen vektornya.

Teorema 2.5.4 Untuk setiap matriks tereduksi reguler $A \in \mathbb{R}_\varepsilon^{n \times n}$ memiliki *eigenmode* berupa pasangan vektor dengan semua elemen vektor berhingga.

Bukti Misal matriks blok segitiga atas bentuk (2.16) adalah kontruksi dari matriks tereduksi A . *Eigenmode* dari matriks blok segitiga atas hasil kontruksi dari matriks tereduksi A berupa pasangan vektor $(\boldsymbol{\eta}, \mathbf{v})$. Telah dijelaskan bahwa vektor

$$\boldsymbol{\eta} = (\mathbf{u}^T[\xi_1] \quad \mathbf{u}^T[\xi_2] \quad \dots \quad \mathbf{u}^T[\xi_q])^T$$

adalah vektor waktu sikel, dengan ξ_i , $i \in \underline{q}$ adalah elemen bilangan real menurut Teorema 2.5.2. Sehingga vektor $\boldsymbol{\eta}$ terdiri dari elemen berhingga. Berikutnya vektor $\mathbf{v}$ terdiri dari vektor-vektor $\mathbf{v}_i$, $i \in \underline{q}$. Menurut Teorema 2.5.2 seluruh elemen vektor $\mathbf{v}_i$, $i \in \underline{q}$ berhingga yaitu elemen bilangan real. Sehingga vektor $\mathbf{v}$ hanya memuat elemen berhingga. Jadi *eigenmode* dari matriks tereduksi berupa pasangan vektor dengan semua elemen vektor berhingga. □

2.6 Algoritma Iterasi *Policy*

Algoritma yang efisien untuk menghitung komponen eigen dari sistem linier $(\max, +)$ adalah algoritma iterasi *policy* yang diuraikan oleh Cochet-Terrasson [32]. Algoritma didapat dari algoritma iterasi *multichain policy* oleh Howard [33] yang mana telah diketahui secara umum merupakan teori dari Markov. Pada [32] diperkenalkan algoritma generalisasi *eigenmode* yaitu pasangan vektor $(\boldsymbol{\eta}, \mathbf{v})$ dengan $\boldsymbol{\eta}, \mathbf{v} \in \mathbb{R}^n$. Jika ada $M \in \mathbb{R}$ sedemikian hingga untuk $m \in \mathbb{R}$ berlaku

$$m \geq M \implies D^{\otimes m} \otimes \mathbf{v} = \mathcal{A}(D^{\otimes -1}) \otimes D^{\otimes m} \otimes \mathbf{v}, \quad (2.57)$$

dimana D adalah $\text{diag}(\eta_1, \eta_2, \dots, \eta_n)^T$, $D^{\otimes m}$ adalah $\text{diag}(m \times \eta_1, \dots, m \times \eta_n)$ dan

$$\mathcal{A}(\lambda) = \bigoplus_{t=0}^l A_t \lambda^{\otimes t}.$$

Jika $\mathcal{A}$ mempunyai nilai eigen yaitu λ_0 maka Persamaan 2.57 dapat ditulis sebagai

$$\mathbf{v} = \mathcal{A}(\lambda_0^{\otimes -1}) \otimes \mathbf{v}$$

atau

$$\bigoplus_{t=0}^l A_t \otimes \lambda_0^{\otimes -t} \otimes \mathbf{v} = \mathbf{v}.$$

Jika $\mathcal{A}$ tidak punya nilai eigen, maka algoritma iterasi *policy* menghitung vektor $\boldsymbol{\eta} = (\eta_1, \eta_2, \dots, \eta_n)^T$ dimana setiap η_i berhubungan dengan maksimum nilai eigen tergenalisir dari kelas yang mengakses simpul q_i . Seperti halnya vektor eigen tergenalisir $\mathbf{v}$, Persamaan 2.57 dapat ditulis sebagai

$$[D^{\otimes m}]_{j,j} \otimes [\mathbf{v}]_j = \bigoplus_{t=0}^l \bigoplus_{i=1}^n \left(([A_t]_{j,i} - [D^{\otimes t}]_{i,i}) \otimes [D^{\otimes m}]_{i,i} \otimes [\mathbf{v}]_i \right),$$

atau

$$m \times [\boldsymbol{\eta}]_j + [\mathbf{v}]_j = \max_{t=0,1,\dots,l} \max_{i=1,2,\dots,n} (([A_t]_{j,i} - t \times [\boldsymbol{\eta}]_i) + (m \times [\boldsymbol{\eta}]_i)) + [\mathbf{v}]_i \quad (2.58)$$

bagi kedua sisi pada persamaan diatas dengan m dan untuk $m \rightarrow \infty$ didapat

$$[\boldsymbol{\eta}]_j = \max_{(i,t,j) \in E} [\boldsymbol{\eta}]_i,$$

dengan $E = \{(i, t, j) \in \mathcal{N} \times \mathcal{N} \times \mathcal{N} | [A_t]_{j,i} \neq \varepsilon\}$ dengan (i, t, j) sisi dari simpul q_i ke q_j dengan t dapat dianggap sebagai banyaknya kendaraan yang melalui lintasan dari q_i ke q_j dan dalam istilah Petri Nets dinamakan banyaknya *token* dalam suatu *place*. Jika

simpul i mempunyai akses terhadap simpul j maka $[\eta]_i$ sama dengan $[\eta]_j$. Dengan demikian Persamaan 2.58 dapat ditulis sebagai

$$[v]_j = \max_{(i,t,j) \in \bar{E}} (\tau_{j,i} - \mu_{j,i} \times [\eta]_i + [v]_i)$$

dengan $\tau_{j,i}$ dan $\mu_{j,i}$ berturut-turut adalah waktu (bobot) dan banyaknya kendaraan (token) yang melalui lintasan dari i ke j . Sebelum dijelaskan algoritma iterasi *policy*, pertama diberikan asumsi berikut ini.

Asumsi 1 Diasumsikan bahwa matriks polinomial $\mathcal{A}(\gamma) = \bigoplus_{t=0}^l A_t \otimes \gamma^{\otimes t} \in \mathbb{R}_{\varepsilon}^{n \times n}$ mempunyai paling sedikit satu elemen berhingga pada tiap barisnya dan graf dengan bobot waktu A_0 yang dinotasikan sebagai $\mathcal{G}_{TM_0}$ tidak memiliki sirkuit.

Telah diketahui bahwa jika $\mathcal{A}$ memenuhi Asumsi 1 didapat *eigenmode* tergeneralisir (η, v) , maka *cycle time vector* dari $\mathcal{A}$ adalah $\chi(\mathcal{A}) = \eta$. Algoritma iterasi *policy* mengkonstruksi sebuah subgraf dari $\mathcal{G}_{TM}$ sedemikian hingga setiap simpul (j) mempunyai tepat satu simpul (i) dengan (i, t, j) terdapat pada subgraf tersebut. Catatan bahwa subgraf ini memuat paling sedikit satu sirkuit. Himpunan simpul yang terhubung dengan setiap simpul dalam subgraf tersebut berkaitan dengan apa yang dinamakan *policy*. Untuk definisi yang lebih formal diberikan sebagai berikut.

Algoritma iterasi *policy* biasanya berisi dua tahap untuk setiap iterasi ke- k . Bagian pertama, menghitung pasangan (η^k, x^k) dengan

$$\eta^k = [\eta_1^k \quad \eta_2^k \quad \dots \quad \eta_n^k]^\top$$

dan

$$x^k = [x_1^k \quad x_2^k \quad \dots \quad x_n^k]^\top$$

berhubungan dengan *policy* yang diberikan, tahap ini disebut sebagai tahap **penentuan nilai**. Bagian kedua adalah bagian untuk menentukan *policy* yang lebih baik berdasarkan siklus rata-rata dari sirkuit atau berdasarkan vektor x^k yang berdasarkan pada subgraf tersebut, tahap ini disebut sebagai tahap **perbaikan *policy***.

Secara formal, *policy* adalah pemetaan $\pi : V \rightarrow E$ (dengan V himpunan simpul) didefinisikan sebagai

$$\pi(j) = (i, t, j) = p_{j,i}^t \text{ dimana } i \in P_j = \{s | p_{j,s}^t \in E\}$$

untuk semua $j \in V$. Dinotasikan input dari titik pada suatu *policy* dengan $\text{In}(\pi(j))$ dan bobot waktu dari sisi yang menghubungkan kedua simpul diberikan oleh

$$\tau(\pi(j)) := \tau_{j, \text{In}(\pi(j))}$$

dan inisialisasi banyaknya kendaraan diberikan oleh

$$\mu(\pi(j)) := \mu_{j, \text{In}(\pi(j))}.$$

Selanjutnya *policy* π dikaitkan dengan matriks polinomial

$$\mathcal{A}^\pi(\gamma) = \bigoplus_{t=0}^l A_t^\pi \otimes \gamma^{\otimes t},$$

dimana

$$[A_t^\pi]_{j,i} = \begin{cases} \tau(\pi(j)), & i = \text{In}(\pi(j)) \text{ dan } t = \mu(\pi(j)) \\ \varepsilon, & \text{untuk yang lain.} \end{cases}$$

Matriks $A^\pi(\gamma)$ mempunyai tepat satu entri berhingga perbaris yang berhubungan terhadap output simpul dari simpul yang dipilih pada π . Timed marked Graph (TMG) $\mathcal{G}_{TM}^\pi$ yang berhubungan dengan $\mathcal{A}^\pi(\gamma)$ adalah subgraf dari $\mathcal{G}_{TM}$ dengan setiap simpul mempunyai simpul tunggal yang terhubung terhadapnya. Jelas bahwa $\mathcal{G}_{TM}^\pi$ memuat paling sedikit satu sirkuit.

Pada tahap penentuan nilai, dihitung vektor $\boldsymbol{\eta}^k$ dan $\mathbf{x}^k$ berdasarkan *policy* π sedemikian hingga

$$\begin{aligned} [\mathbf{x}^k]_j &= \bigoplus_{i=1}^n \left[\mathcal{A}^\pi(([\boldsymbol{\eta}^k]_i)^{\otimes -1}) \right]_{j,i} \otimes [\mathbf{x}^k]_i \\ &= \bigoplus_{i=1}^n [A_{\mu(\pi(j))}]_{j,i} \otimes ([\boldsymbol{\eta}^k]_i)^{\otimes \mu(\pi(j))} \otimes [\mathbf{x}^k]_i \end{aligned}$$

dan didapat

$$[\mathbf{x}^k]_j = \tau(\pi(j)) - \mu(\pi(j)) \times [\boldsymbol{\eta}^k]_i + [\mathbf{x}]_{\text{In}(\pi(j))} \quad (2.59)$$

untuk semua $j = 1, 2, \dots, n$. Persamaan 2.59 berdasarkan definisi dari *policy*, untuk setiap simpul q_j mempunyai hanya satu simpul yang terhubung dalam $\mathcal{G}_{TM}^\pi$. Persamaan 2.59 diselesaikan untuk $(\boldsymbol{\eta}^k, \mathbf{x}^k)$. Pertama tentukan sirkuit dalam $\mathcal{G}_{TM}^\pi$ berdasarkan *policy* π yang diberikan. Selanjutnya hitung rata-rata siklus $\bar{\eta}$ dari sirkuit yang ditentukan. Pilih sebarang simpul q_i pada sirkuit dan tentukan waktu siklus $[\boldsymbol{\eta}^k]_i = \bar{\eta}$ dan $[\mathbf{x}^k]_i = [\mathbf{x}^{k-1}]_i$. Setiap simpul q_j yang terhubung terhadap *path* dari q_i dalam $\mathcal{G}_{TM}^\pi$ ditentukan waktu siklus $[\boldsymbol{\eta}^k]_j = \bar{\eta}$ dan nilai $[\mathbf{x}^k]_j$ dihitung sesuai Persamaan 2.59. Proses ini dilakukan untuk semua sirkuit dalam *policy* π .

Pada tahap perbaikan *policy*, dilakukan perbaikan *policy* berdasarkan $(\boldsymbol{\eta}^k, \mathbf{x}^k)$ yang telah didapat pada tahap sebelumnya. Pertama, untuk setiap simpul q_j periksa apakah ada sisi $p_{j,i}^t$ dengan input simpul q_i mempunyai waktu siklus yang lebih besar dari $[\boldsymbol{\eta}^k]_i$. Jika ada, *policy* $\pi(j)$ diubah ke sisi $p_{j,i}^t$ yang menghubungkan q_j ke sirkuit dengan siklus rata-rata yang lebih besar. Jika *policy* tidak dapat diperbaiki dengan cara ini maka pilih $\mathbf{x}^k$ yang bisa memperbaiki, yaitu jika ada q_j sedemikian hingga

$$\bigoplus_{i=1}^n (\mathcal{A}^\pi(([\boldsymbol{\eta}^k]_i)^{\otimes -1}))_{j,i} \otimes [\mathbf{x}^k]_i > [\mathbf{x}^k]_j, \quad (2.60)$$

maka *policy* π tidak optimal. Perbaikan *policy* yang ditentukan oleh pengubahan $\pi(j)$ untuk setiap komponen j yang memenuhi 2.60 dan $\pi(j)$ diubah menjadi sisi $p_{j,i}^t$ yang mana

bagian kiri dari Persamaan 2.60 maksimum. Untuk algoritma iterasi *policy* diberikan sebagai berikut.

Algoritma Iterasi Policy

Input : Polinomial matriks $\mathcal{A}$ memenuhi Asumsi 1.

Output : Generalisasi *eigenmode* dari $\mathcal{A}$

1. Inisialisasi

Pilih sebarang *policy* π_1 . Setting $k = 1$ dan $\mathbf{x}^0 = (0, \dots, 0)^\top \in \mathbb{R}^n$. Tentukan pasangan $(\boldsymbol{\eta}^1, \mathbf{x}^1)$ menggunakan tahap 3 dari algoritma

2. Perbaikan Policy

Misal

$$\begin{aligned} J &= \left\{ j \mid \max_{p_{j,i} \in P_j} [\boldsymbol{\eta}^k]_i > [\boldsymbol{\eta}^k]_j \right\}, \\ K(j) &= \arg \max_{p_{j,i} \in P_j} [\boldsymbol{\eta}^k]_i, \quad j = 1, 2, \dots, n, \\ I &= \left\{ j \mid \max_{p_{j,i} \in K(j)} (\tau_{j,i} - \mu_{j,i} [\boldsymbol{\eta}^k]_i + [\mathbf{x}^k]_i) > [\mathbf{x}^k]_j \right\}, \\ L(j) &= \arg \max_{p_{j,i} \in K(j)} (\tau_{j,i} - \mu_{j,i} [\boldsymbol{\eta}^k]_i + [\mathbf{x}^k]_i), \quad j = 1, 2, \dots, n. \end{aligned}$$

jika $I = J = \emptyset$ maka berhenti. Generalisasi *eigenmode* diberikan oleh $(\boldsymbol{\eta}^k, \mathbf{x}^k)$. Lebih lanjut, jika $J \neq \emptyset$. Maka *setting*

$$\pi_{k+1}(j) = \begin{cases} \text{pilih } p_{j,i} \in K(j) & \text{jika } j \in J \\ \pi_k(j) & \text{jika } j \notin J, \end{cases}$$

dan jika tidak ($J = \emptyset$ tetapi $I \neq \emptyset$) *setting*

$$\pi_{k+1}(j) = \begin{cases} \text{pilih } p_{j,i} \in L(j) & \text{jika } j \in I \\ \pi_k(j) & \text{jika } j \notin I. \end{cases}$$

3. Penentuan Nilai

Tentukan sirkuit ξ pada graf $\mathcal{G}_{TM}^{\pi_k}$ dan misal

$$\bar{\eta} = \frac{\sum_{p_{uv} \in \xi} \tau_{uv}}{\sum_{p_{uv} \in \xi} \mu_{uv}}$$

pilih sebarang simpul $q_i \in \xi$ dan *setting* $[\boldsymbol{\eta}^k]_i = \bar{\eta}$, $[\mathbf{x}^k]_i = [\mathbf{x}^{k-1}]_i$. Untuk simpul j yang terhubung terhadap *path* i pada $\mathcal{G}_{TM}^{\pi_k}$, *setting*

$$[\boldsymbol{\eta}^k]_j = \bar{\eta} \quad \text{dan} \quad [\mathbf{x}^k]_j = \tau(\pi(j)) - \mu(\pi(j)) \times [\boldsymbol{\eta}^k]_i + [\mathbf{x}]_{\text{In}(\pi(j))}.$$

Jika ada himpunan takkosong C dari simpul q_i yang takterhubung dengan q_i , maka lakukan kembali tahap 3 menggunakan sirkuit yang berbeda dari sebelumnya.

4. Iterasi *Policy*

Setting $k = k + 1$ dan menuju ke tahap 2.

Contoh 4

Diberikan matriks polinomial

$$\mathcal{A}(\lambda) = A_0 + A_1\lambda$$

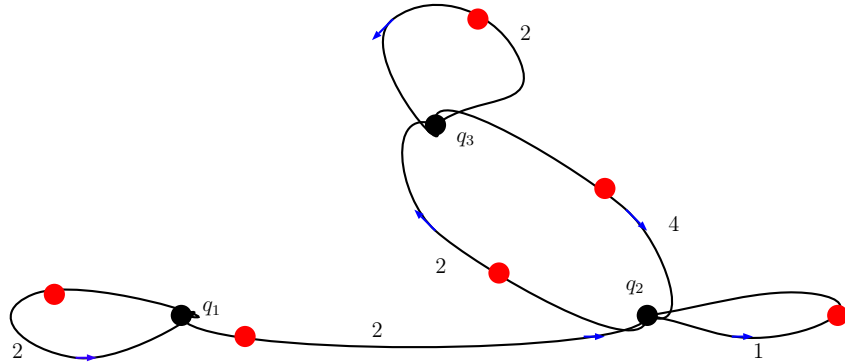
dengan

$$A_0 = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \end{bmatrix}, A_1 = \begin{bmatrix} 2 & 2 & \varepsilon \\ \varepsilon & 1 & 4 \\ \varepsilon & 2 & 2 \end{bmatrix}.$$

Tentukan nilai eigen dan vektor eigen dari matriks polinomial diatas.

Penyelesaian

Sebelum menggunakan algoritma iterasi *policy* terlebih dahulu digambarkan graf $\mathcal{G}_{TM}$ dari matriks polinomial diatas.



Gambar 2.14: Graf dari polinomial pada Contoh 4

selanjutnya diaplikasikan algoritma iterasi *policy* yaitu

Tahap Inisialisasi

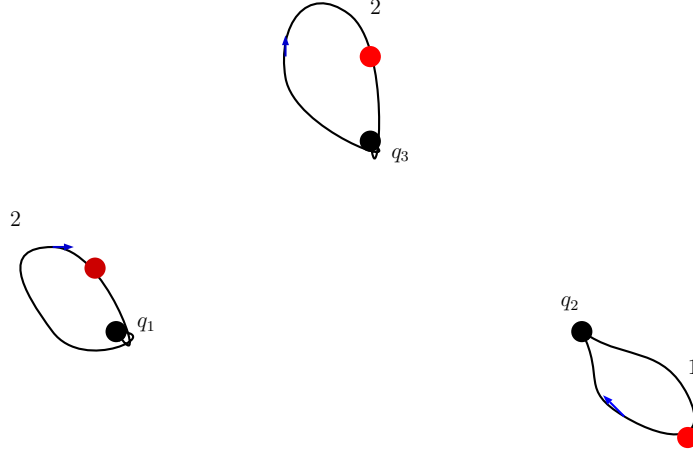
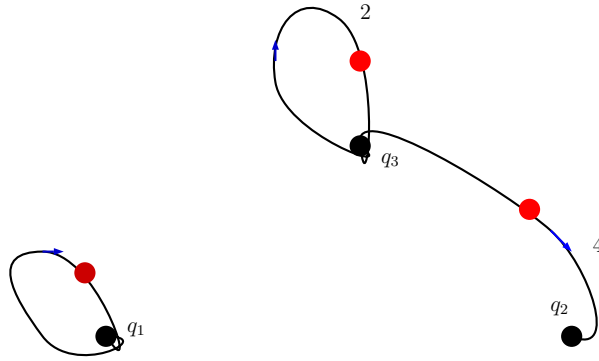
pilih $\mathbf{x}^0 = (0, 0, 0)^T$ dan $\pi_1 = \{p_{11}^1, p_{22}^1, p_{33}^1\}$ dapat digambar graf dari $\mathcal{G}_{TM}^{\pi_1}$ yaitu dapat dilihat pada Gambar 2.15.

Tahap Penentuan Nilai

Ambil siklus $q_1 \rightarrow q_1$ pada $\mathcal{G}_{TM}^{\pi_1}$ sehingga $\bar{\eta} = \frac{2}{1} = 2$ sehingga didapat $x_1^1 = 0$ dan $\eta_1^1 = \bar{\eta} = 2$. Karena tidak ada simpul lain yang terhubung ke q_1 maka ambil siklus lain di $\mathcal{G}_{TM}^{\pi_1}$ yaitu $q_2 \rightarrow q_2$ sehingga didapat $x_2^1 = 0$ dan $\eta_2^1 = 1$ selanjutnya dengan cara yang sama didapat $x_3^1 = 0$ dan $\eta_3^1 = 2$, Dengan demikian pada tahap ini didapatkan $\mathbf{x}^1 = (0, 0, 0)^T, \boldsymbol{\eta}^1 = (2, 1, 2)^T$.

Perbaikan *policy*

Dari Gambar 2.14 dan nilai dari $\mathbf{x}^1$ dan $\boldsymbol{\eta}^1$ didapatkan $J = \{2\}$, $K(1) = \{1\}$, $K(2) = \{3\}$, $K(3) = \{3\}$, $I = \{2\}$, $L(1) = \{1\}$, $L(2) = \{3\}$, $L(3) = \{3\}$. Karena $J \neq \emptyset$ maka


 Gambar 2.15: Graf dari *policy* pada iterasi ke-1

 Gambar 2.16: Graf dari *policy* pada iterasi ke-2

didapat $\pi_2 = \{p_{11}^1, p_{2,3}^1, p_{33}^1\}$ (untuk gambar graf dari $\mathcal{G}_{TM}^{\pi_2}$ dapat dilihat pada Gambar 2.16).

Tahap Penentuan Nilai

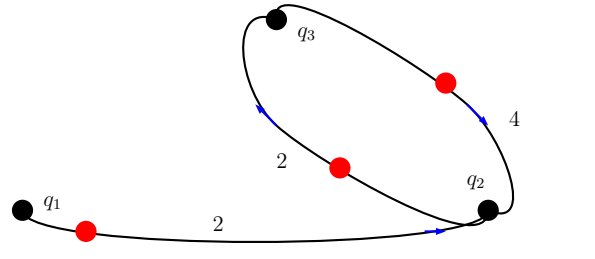
Ambil siklus $q_1 \rightarrow q_1$ pada $\mathcal{G}_{TM}^{\pi_2}$ sehingga $\bar{\eta} = \frac{2}{1} = 2$ sehingga didapat $x_1^2 = 0$ dan $\eta_1^2 = \bar{\eta} = 2$. Karena tidak ada simpul lain yang terhubung ke q_1 maka ambil siklus lain di $\mathcal{G}_{TM}^{\pi_1}$ yaitu $q_3 \rightarrow q_3$ sehingga didapat $x_3^2 = 0$ dan $\eta_3^2 = 2$. Karena q_2 terhubung terhadap q_3 maka didapat $\eta_2^2 = 2$ dan $x_2^3 = \tau(p_{23}^1) - \mu(p_{23}^1)\bar{\eta} + x_3^2 = 4 - 1 \cdot 2 + 0 = 2$. Jadi $\mathbf{x}^2 = (0, 2, 0)^T$, $\boldsymbol{\eta}^1 = (2, 2, 2)^T$.

Perbaikan *policy*

Dari Gambar 2.14 dan nilai dari x^1 dan η^1 didapatkan $J = \emptyset$, $K(1) = \{1, 2\}$, $K(2) = \{2, 3\}$, $K(3) = \{2, 3\}$, $I = \{1, 3\}$, $L(1) = \{2\}$, $L(2) = \{3\}$, $L(3) = \{2\}$. Karena $J = \emptyset$ dan $I \neq \emptyset$ maka didapat $\pi_3 = \{p_{12}^1, p_{23}^1, p_{32}^1\}$ (untuk gambar graf dari $\mathcal{G}_{TM}^{\pi_3}$ dapat dilihat pada Gambar 2.17).

Tahap Penentuan Nilai

Ambil siklus $q_2 \rightarrow q_3 \rightarrow q_2$ pada $\mathcal{G}_{TM}^{\pi_3}$ sehingga $\bar{\eta} = \frac{4+2}{1+1} = 2$ kemudian ambil simpul q_3

Gambar 2.17: Graf dari *policy* pada iterasi ke-3

sehingga didapat $x_3^3 = 0$ dan $\eta_3^3 = \bar{\eta} = 3$. Karena q_1 dan q_2 terhubung terhadap q_3 maka didapat $\eta_1^3 = 3$, $\eta_2^3 = 3$, $x_2^3 = \tau(p_{23}^1) - \mu(p_{23}^1)\bar{\eta} + x_3^3 = 4 - 1 \cdot 3 + 0 = 1$ dan $x_1^3 = \tau(p_{12}^1) - \mu(p_{12}^1)\bar{\eta} + x_2^3 = 2 - 1 \cdot 3 + 1 = 0$. Jadi $\mathbf{x}^2 = (0, 1, 0)^T$, $\boldsymbol{\eta}^1 = (3, 3, 3)^T$.

Perbaiki *policy*

Dari Gambar 2.14 dan nilai dari $\mathbf{x}^1$ dan $\boldsymbol{\eta}^1$ didapatkan $J = \emptyset$, $K(1) = \{1, 2\}$, $K(2) = \{2, 3\}$, $K(3) = \{2, 3\}$, $I = \emptyset$, $L(1) = \{1, 2\}$, $L(2) = \{3\}$, $L(3) = \{2, 3\}$. Karena $J = \emptyset$ dan $I = \emptyset$ maka iterasi berhenti. Sehingga didapat eigenmode tergeneralisir adalah

$$\boldsymbol{\eta} = \begin{bmatrix} 3 \\ 3 \\ 3 \end{bmatrix}, \mathbf{x} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

Algoritma ini telah diimplementasikan pada toolbox Scilab berikut ini contohnya.

```
--> A=[2,2,-%inf;-%inf,1,4;-%inf,2,2];
```

```
-->[eigvalMode,x] = policyIteration(A)
```

Number Of Iteration

3.

x =

2.

3.

2.

eigvalMode =

3.

3.

3.

```
-->\\Atau kamu bisa merubahnya kedalam bentuk polynomial sebagai berikut
```

```
-->A0=maxpluszeros(3,3);

-->A1=[2,2,-%inf;-%inf,1,4;-%inf,2,2];

-->A=A0;

-->A(:, :, 2)=A1
A =

(:, :, 1)

- Inf - Inf - Inf
- Inf - Inf - Inf
- Inf - Inf - Inf
(:, :, 2)

2.      2. - Inf
- Inf    1.    4.
- Inf    2.    2.

-->[eigvalMode,x] = policyIteration(A)

Number Of Iteration

3.
x =

2.
3.
2.
eigvalMode =

3.
3.
3.
```

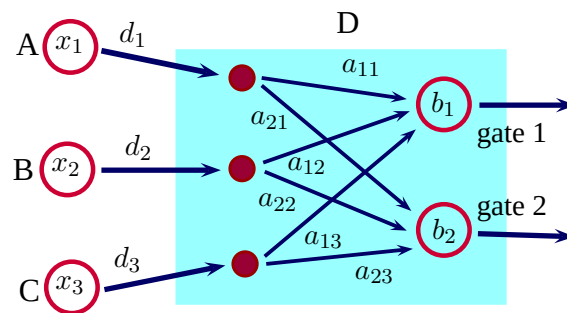

Bab 3

Contoh Aplikasi

Pada bab ini diberikan beberapa contoh aplikasi dari Aljabar maxplus.

3.1 Masalah Jadwal Penerbangan Pesawat pada suatu Bandara

Tiga pesawat penerbangan komersial berangkat masing-masing dari bandara A,B dan C dengan tujuan bandara utama D yang mana dua pesawat terhubung sudah menanti masing-masing di gate 1 dan gate 2. Waktu keberangkatan dari D dan penutupan gate diberikan dan tidak bisa diubah. Durasi waktu transfer diantara tiga kedatangan dan dua keberangkatan pesawat adalah a_{ij} , $i = 1, 2, j = 1, 2, 3$.



Sedangkan lamanya waktu penerbangan dari A ke D adalah d_1 , dari B ke D adalah d_2 dan dari C ke D adalah d_3 . Bila x_1, x_2 dan x_3 masing-masing menyatakan waktu keberangkatan pesawat dari A,B dan C dan b_1, b_2 masing-masing menyatakan waktu penutupan gate 1 dan gate 2, maka buat model aljabar maxplusnya. Selanjutnya bila diketahui $d_1 = 150, d_2 = 120, d_3 = 135, a_{11} = 10, a_{12} = 20, a_{13} = 30, a_{21} = 15, a_{22} = 35, a_{23} = 35$ dan $b_1 = 180, b_2 = 175$, maka hitung x_1, x_2 dan x_3 . Selanjutnya terjemahkan hasil hitungan, yaitu pukul berapa masing-masing pesawat berangkat dari A, B dan C serta pukul berapa gate 1 dan gate 2 ditutup.

Jawab

Model aljabar maxplusnya adalah

$$\begin{aligned}\max\{150 + 10 + x_1, 120 + 20 + x_2, 135 + 30 + x_3\} &= 180 \\ \max\{150 + 15 + x_1, 120 + 35 + x_2, 135 + 35 + x_3\} &= 175\end{aligned}$$

atau

$$A \otimes \mathbf{x} = \mathbf{b}$$

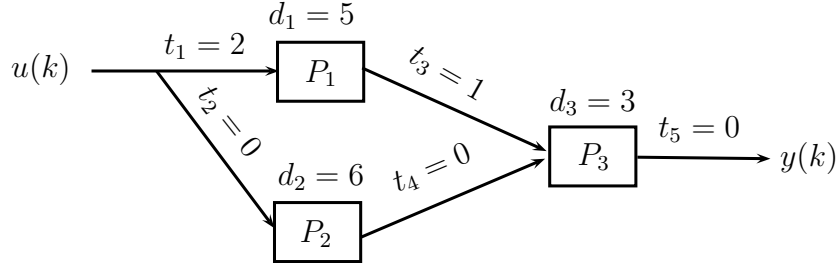
dengan

$$A = \begin{bmatrix} 160 & 140 & 165 \\ 165 & 155 & 170 \end{bmatrix}, \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \text{ dan } \mathbf{b} = \begin{bmatrix} 180 \\ 175 \end{bmatrix}.$$

Untuk menyelesaikan persamaan ini bisa digunakan Scilab sebagai berikut:

```
-->A=[160 140 165;165 155 170]
A =
    160.    140.    165.
    165.    155.    170.
-->b=[180;175]
b =
    180.
    175.
-->[x] = maxpluslinsol(A,b)
x =
    10.
    20.
     5.
-->isequal(maxplusotimes(A,x),b)
ans =
    F
-->maxplusotimes(A,x)<=b
ans =
    T
    T
```

Terlihat bahwa $A \otimes \mathbf{x} \neq \mathbf{b}$, tetapi $A \otimes \mathbf{x} \leq \mathbf{b}$ ini berarti $x_1 = 10, x_2 = 20, x_3 = 5$. adalah penyelesaian suboptimal dari $A \otimes \mathbf{x} \leq \mathbf{b}$. Bila satuan durasi waktu adalah menit, maka salah satu tafsiran $x_1 = 10$ adalah pukul 06.10, $x_1 = 20$ adalah pukul 06.20 dan $x_3 = 5$ adalah pukul 06.05. Dengan demikian $b_1 = 180$ adalah pukul 09.00 dan $b_2 = 175$ adalah pukul 08.55.



Gambar 3.1: Sistem Produksi Sederhana

3.2 Sistem Produksi Sederhana

Diberikan sistem produksi sederhana sebagaimana diberikan dalam Gambar 3.1. Sistem produksi ini terdiri dari 3 unit pemroses P_1, P_2 dan P_3 . Bahan baku dimasukkan pada proses P_1 dan P_2 untuk diproses dan hasilnya dikirim ke P_3 dimana dilakukan perakitan. Waktu proses yang dibutuhkan masing-masing diberikan oleh: $d_1 = 5, d_2 = 6$ dan $d_3 = 3$ satuan waktu. Bila diasumsikan diperlukan $t_1 = 2$ satuan waktu dari bahan baku untuk mencapai P_1 dan dibutuhkan $t_3 = 1$ satuan waktu untuk menyelesaikan produk dari pemroses P_1 untuk mencapai P_3 . Waktu perjalanan lainnya (t_2, t_4 dan t_5) diasumsikan nol. Diantara input dan pemroses terdapat buffer dengan kapasitas yang cukup besar untuk menjamin bahwa buffer tidak akan pernah overflow. Suatu unit pemroses hanya bisa mulai bekerja untuk menghasilkan produk yang baru bila ia telah menyelesaikan proses yang sebelumnya. Diasumsikan pula bahwa setiap unit pemroses sesegera mungkin mulai bekerja bila semua komponen telah tersedia. Didefinisikan:

- $u(k)$ adalah waktu dimana bahan baku dimasukkan ke sistem untuk saat yang ke- $(k + 1)$.
- $x_i(k)$ adalah waktu dimana pemroses yang ke- i mulai aktif saat yang ke- $k, i = 1, 2, 3$.
- $y(k)$ adalah waktu dimana produk selesai saat yang ke- k meninggalkan sistem (ditawarkan ke dunia luar/konsumen).

Selanjutnya ditentukan waktu kapan pemroses P_1 mulai bekerja untuk saat yang ke- $(k + 1)$. Bila dimasukkan bahan baku ke sistem saat yang ke- $(k + 1)$, maka bahan baku ini tersedia sebagai input pemroses P_1 pada waktu $t = u(k) + 2$. Bagaimanapun P_1 hanya dapat mulai bekerja untuk memroses bahan baku tsb. sesegera mungkin bila P_1 telah menyelesaikan proses yang sedang berjalan saat ini, yaitu proses saat yang ke- k . Karena waktu pemrosesan pada P_1 adalah $d_1 = 5$ satuan waktu, maka produk diantara yang ke- k akan meninggalkan P_1 pada saat $t = x_1(k) + 5$. Juga, karena P_1 mulai bekerja sesegera mungkin saat bahan baku telah tersedia dan hasil produk saat yang ke- k sudah meninggalkan P_1 , maka diperoleh:

$$x_1(k + 1) = \max(x_1(k) + 5, u(k) + 2) \quad (3.1)$$

untuk semua $k \in \mathbb{N}_0$, dimana $\mathbb{N}_0$ menyatakan himpunan bilangan bulat taknegatif. Kondisi awal bahwa pemroses P_1 kosong dan "idle" ini menunjukkan bahwa $x_1(0) = \varepsilon$. Oleh karena itu dari (3.1) didapat $x_1(1) = u(0) + 2$. Dengan menggunakan alasan yang serupa, pemroses P_2 dan P_3 mulai bekerja saat yang ke- $(k+1)$ dan produk ditawarkan kekonsumen saat yang ke- k diberikan sebagai berikut:

$$x_2(k+1) = \max(x_2(k) + 6, u(k) + 0) \quad (3.2)$$

$$\begin{aligned} x_3(k+1) &= \max(x_1(k) + 5 + 1, x_2(k) + 6 + 0, x_3(k) + 3) \\ &= \max(x_1(k) + 6, x_2(k) + 6, x_3(k) + 3) \end{aligned} \quad (3.3)$$

$$y(k) = x_3(k) + 3 + 0 \quad (3.4)$$

untuk semua $k \in \mathbb{N}_0$ Kondisi bahwa semua buffer pada saat awal adalah kosong berkenaan dengan $x_1(0) = x_2(0) = x_3(0) = \varepsilon$. Selanjutnya bila ditulis kembali persamaan evolusi dari sistem produksi dengan menggunakan simbol $\oplus$ dan $\otimes$, persamaan (3.1), (3.2), (3.3) dan (3.4) menjadi

$$\begin{aligned} x_1(k+1) &= 5 \otimes x_1(k) \oplus 2 \otimes u(k) \\ x_2(k+1) &= 6 \otimes x_2(k) \oplus u(k) \\ x_3(k+1) &= 6 \otimes x_1(k) \oplus 6 \otimes x_2(k) \oplus 3 \otimes x_3(k) \\ y(k) &= 3 \otimes x_3(k). \end{aligned}$$

Bila persamaan-persamaan evolusi terakhir diatas ditulis kembali ke bentuk matriks aljabar-max-plus, diperoleh

$$\begin{aligned} \mathbf{x}(k+1) &= \begin{bmatrix} 5 & \varepsilon & \varepsilon \\ \varepsilon & 6 & \varepsilon \\ 6 & 6 & 3 \end{bmatrix} \otimes \mathbf{x}(k) \oplus \begin{bmatrix} 2 \\ 0 \\ \varepsilon \end{bmatrix} \otimes u(k) \\ y(k) &= [\varepsilon \ \varepsilon \ 3] \otimes \mathbf{x}(k) \end{aligned}$$

dimana $x(k) = [x_1(k) \ x_2(k) \ x_3(k)]'$ dan notasi $'$ menyatakan transpos. Catatan model diatas adalah model dari persamaan

$$\begin{aligned} \mathbf{x}(k+1) &= A \otimes \mathbf{x}(k) \oplus B \otimes u(k) \\ y(k) &= C \otimes \mathbf{x}(k), \end{aligned}$$

dengan

$$A = \begin{bmatrix} 5 & \varepsilon & \varepsilon \\ \varepsilon & 6 & \varepsilon \\ 6 & 6 & 3 \end{bmatrix}, B = \begin{bmatrix} 2 \\ 0 \\ \varepsilon \end{bmatrix} \text{ dan } C = [\varepsilon \ \varepsilon \ 3]$$

Selanjutnya dikembangkan sistem ini dengan asumsi bahwa bila saat waktu bahan baku dimasukan kesistem ketika saat waktu setelah hasil produk selesai ditawarkan kekonsumen

($u(k) = y(k)$), maka evolusi dari keadaan sistem diberikan oleh:

$$\begin{aligned}
 \mathbf{x}(k+1) &= A \otimes \mathbf{x}(k) \bigoplus B \otimes u(k) \\
 &= A \otimes \mathbf{x}(k) \bigoplus B \otimes y(k) \\
 &= A \otimes \mathbf{x}(k) \bigoplus B \otimes C \otimes \mathbf{x}(k) \\
 &= \bar{A} \otimes \mathbf{x}(k)
 \end{aligned}$$

dimana

$$\bar{A} = A \bigoplus B \otimes C.$$

Untuk menghitung $\bar{A}$ bisa digunakan Aljabar Max-Plus Toolbox dalam Scilab sebagai berikut:

```

-->A=[5 -%inf -%inf;
-->  -%inf 6 -%inf;
-->  6 6 3]
A =
    5.  - Inf  - Inf
   - Inf    6.  - Inf
    6.    6.    3.
-->B=[2;0;-%inf]
B =
    2.
    0.
   - Inf
-->C=[-%inf -%inf 3]
C =
   - Inf  - Inf    3.
-->Abar=maxplusoplus(A,maxplusotimes(B,C))
Abar =
    5.  - Inf    5.
   - Inf    6.    3.
    6.    6.    3.

```

diperoleh nilai

$$\bar{A} = \begin{bmatrix} 5 & \varepsilon & 5 \\ \varepsilon & 6 & 3 \\ 6 & 6 & 3 \end{bmatrix},$$

dalam hal ini tentunya lebih dulu telah diberikan nilai-nilai dari matriks A , B dan C kedalam Scilab. Selanjutnya dikaji perilaku dinamik dari sistem dengan mensimulasikan beberapa keadaan awal. Untuk keadaan awal $\mathbf{x} = [0 \ 0 \ 0]^T$, diperoleh evolusi sistem untuk

$k = 0, 1, 2, \dots, 10$:

$$\mathbf{x} = \begin{matrix} & 0 & 5 & 11 & 17 & 23 & 29 & 35 & 41 & 47 & 53 & 59 \\ 0 & 0 & 6 & 12 & 18 & 24 & 30 & 36 & 42 & 48 & 54 & 60 \\ 0 & 6 & 12 & 18 & 24 & 30 & 36 & 42 & 48 & 54 & 60 \end{matrix}$$

$$y = 3 \quad 9 \quad 15 \quad 21 \quad 27 \quad 33 \quad 39 \quad 45 \quad 51 \quad 57 \quad 63$$

Terlihat keadaan sistem mencapai periodik pada saat $k = 1$ dengan periode sama dengan 6. Perintah dalam Scilab untuk memperoleh $\mathbf{x}$ dan y sebagai berikut

```
-->[X] = maxplussys(Abar,[0;0;0],10)
X =
      column 1 to 8
    0.    5.    11.    17.    23.    29.    35.    41.
    0.    6.    12.    18.    24.    30.    36.    42.
    0.    6.    12.    18.    24.    30.    36.    42.
      column 9 to 11
   47.    53.    59.
   48.    54.    60.
   48.    54.    60.
-->for i=1:11
-->y(:,i)=maxplusotimes(C,X(:,i));
-->end
-->y
y =
      column 1 to 9
    3.    9.    15.    21.    27.    33.    39.    45.    51.
      column 10 to 11
   57.    63.
```

Berikutnya dicoba keadaan awal $\mathbf{x} = [1 \ 2 \ 2]^T$, diperoleh evolusi sistemnya:

$$\mathbf{x} = \begin{matrix} & 1 & 7 & 13 & 19 & 25 & 31 & 37 & 43 & 49 & 55 & 61 \\ 2 & 2 & 8 & 14 & 20 & 26 & 32 & 38 & 44 & 50 & 56 & 62 \\ 2 & 8 & 14 & 20 & 26 & 32 & 38 & 44 & 50 & 56 & 62 \end{matrix}$$

$$y = 5 \quad 11 \quad 17 \quad 23 \quad 29 \quad 35 \quad 41 \quad 47 \quad 53 \quad 59 \quad 65$$

Terlihat mulai awal keadaan sistem sudah periodik dengan periode sama dengan 6. Akhirnya, dicoba lagi dengan keadaan awal $\mathbf{x} = [20 \ 1 \ 23]^T$, didapat evolusi sistemnya:

$$\mathbf{x} = \begin{matrix} & 20 & 28 & 33 & 38 & 44 & 50 & 55 & 61 & 67 & 73 & 79 \\ 1 & 1 & 26 & 32 & 39 & 44 & 50 & 56 & 62 & 68 & 74 & 80 \\ 23 & 23 & 26 & 34 & 38 & 45 & 50 & 56 & 62 & 68 & 74 & 80 \end{matrix}$$

$$y = 26 \ 29 \ 37 \ 41 \ 48 \ 53 \ 59 \ 65 \ 71 \ 77 \ 83$$

Terlihat keadaan sistem mencapai periodik pada saat $k = 6$ dengan periode sama dengan 6. Dari beberapa keadaan awal yang diberikan, bisa disimpulkan bahwa keadaan awal $\mathbf{x} = [1 \ 2 \ 2]^T$ merupakan keadaan yang baik untuk mengawali saat keadaan sistem aktif, yaitu saat waktu awal masing-masing proses P_1, P_2 dan P_3 aktif. Sebab dengan keadaan awal ini, akan diperoleh suatu jadwal dari setiap mesin aktif secara teratur dengan periode sama dengan 6. Suatu pertanyaan muncul, bagaimana menentukan suatu keadaan awal dari sistem supaya memperoleh evolusi suatu keadaan sistem periodik dengan periode yang hingga (finite)? Pertanyaan ini bisa dijawab dengan menggunakan pengertian vektor-karakteristik dan nilai karakteristik dari suatu matriks persegi. Perhatikan bisa dicek bahwa vektor keadaan awal $\mathbf{x} = [1 \ 2 \ 2]^T$ adalah suatu vektor-karakteristik dari matriks $\bar{A}$, sedangkan nilai-karakteristik dari $\bar{A}$ sama dengan 6. Hal ini bisa dicek dalam Silab sebagai berikut.

```
-->z = maxplusisegv(Abar, [1;2;2], 6)
z =
T
```

Terlihat bahwa benar matriks $\bar{A}$ mempunyai nilai karakteristik 6 dengan vektor-karakteristik $[1 \ 2 \ 2]^T$. Suatu hal yang menarik adalah kajian untuk menguji kestabilan dari jadwal yang teratur ini bila terjadi gangguan, misalnya ada satu atau lebih mesin produksi rusak. Hal ini tentunya akan membangkitkan suatu keterlambatan suatu mesin proses yang lain dalam pemrosesan. Akibat gangguan ini, suatu pertanyaan yang mendasar adalah apakah sistem bisa kembali lagi menghasilkan jadwal yang periodik? Dengan kata yang lain apakah sistem stabil? Pertanyaan ini bisa dijawab dengan pengertian berkaitan dengan kestabilan dan mengetahui apa syarat-syaratnya sistem yang dikaji adalah stabil.

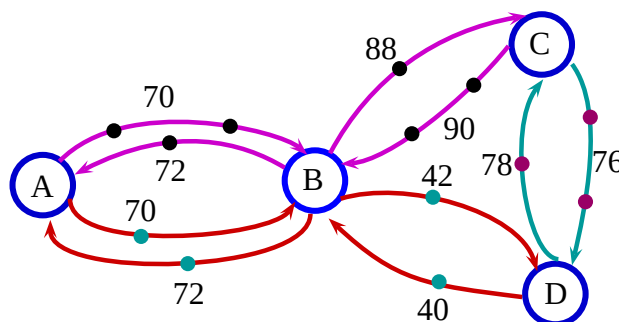
3.3 Penjadwalan Sistem Jaringan Kereta dan Kestabilan

Pada bagian ini diskusikan jaringan transportasi, khususnya jaringan sistem kereta bisa dimodelkan jika diberikan suatu jadwal keberangkatan dari sistem kereta tsb. Model yang dihasilkan bisa digunakan untuk menganalisa perilaku dan ketegaran (robustness) jaringan tsb. Sistem transportasi dipandang sebagai Sistem Event Diskrit dan digunakan aljabar max-plus dalam pemodelan dikerjakan pada abstrasi tingkat tertentu yang akan menghasilkan ciri struktur tertentu jaringan yang bisa dipakai untuk menganalisa dibawah asumsi yang bisa diterima. Alasan utama digunakannya aljabar max-plus disebabkan kemudahannya dalam menanganai proses sinkronisasi. Sistem transportasi adalah suatu contoh dimana sinkronisasi memainkan suatu peranan yang penting. Dalam bidang sain transportasi penggunaan aljabar max-plus adalah relatif baru. Beberapa hasil kerja dari sistem transportasi dengan menggunakan aljabar max-plus bisa dijumpai di [12] dan [3]. Diasumsikan bahwa suatu jaringan kereta beroperasi dengan suatu jadwal keberangkatan

dari semua kereta yang sudah ditentukan secara periodik dengan interval T . Periode T sama untuk semua kereta, yaitu bila keberangkatan kereta i dijadwalkan berangkat pada waktu d_i , maka kereta ini dijadwalkan berangkat pada waktu $d_i + k.T$, $k \in \mathbb{N}$ dimana $\mathbb{N}$ menyatakan himpunan bilangan alam. Semua waktu perjalanan diketahui tetap. Pembatasan ini dimaksudkan bahwa kereta tidak dapat melaju melebihi jadwal yang sudah ditentukan juga tidak memperlambat lajunya sehingga membangkitkan keterlambatan. Asumsi yang lainnya adalah semua waktu perjalanan, dan jadwal disajikan oleh bilangan alam untuk memudahkan analisa dan pemrograman. Hal ini cukup beralasan disebabkan kebanyakan jadwal diberikan dalam menit atau jam. Untuk suatu sistem yang realistis, suatu kondisi dari keberangkatan kereta haruslah memenuhi bahwa kereta sebelumnya yang terjadwal berangkat pada T satuan waktu sudah berangkat lebih dulu. Selain dari pada itu diasumsikan bila suatu kereta terlambat karena sesuatu hal, keterlambatan ini tidak boleh melebihi T . Diasumsikan juga perpindahan penumpang dari satu kereta ke kereta lainya harus terjamin, ini berarti suatu kereta tidak boleh berangkat sebelum kereta tertentu lainya sudah datang. Selajutnya diasumsikan, keberangkatan kereta tidak boleh mendahului jadwal keberangkatan yang sudah ditentukan.

3.3.1 Contoh jaringan kereta

Pada bagian ini dibahas suatu contoh bagaimana menurunkan model jaringan sistem kereta bila diberikan jadwal keberangkatan dari kereta. Dari jadwal yang ada ditentukan banyaknya kereta yang beroperasi pada sistem tsb. disetiap jalur yang ada dengan menggunakan waktu acuan jam 11:58. Selanjutnya dibuat sinkronisasi diantara kereta yang ada pada masing-masing jalur, hal ini dimaksudkan untuk menjamin terjadinya perpindahan penumpang dari suatu kereta dari jalur tertentu ke kereta lainya dengan jalur yang berbeda. Waktu perjalanan dari masing-masing kereta diketahui tetap. Waktu perjalanan tsb. ditentukan dari selisih diantara waktu kedatangan dengan waktu keberangkatan kereta. Misalkan model yang dikaji meliputi 4 stasiun kereta A, B, C dan D yang dihubungkan oleh 3 jalur, seperti ditunjukkan oleh Gambar 3.2. Jalur 1 adalah lintasan kereta dari A ke B ke C, kembali lagi ke A. Jalur 2 adalah lintasan kereta dari A ke B ke D, kembali lagi ke A. Sedangkan jalur 3 adalah lintasan kereta dari C ke D kembali lagi ke C. Angka diatas dan disamping garis panah menunjukkan lamanya waktu perjalanan. Sedangkan jadwal



Gambar 3.2: Jalur Sistem Kereta

keberangkatan kereta, waktu perjalan dan banyaknya kereta pada masing-masing jalur pada saat jam 11:58 sebagai acuan waktu diberikan oleh tabel berikut:

Jalur	dari	ke	jadwal berangkat	lama perjalanan	banyaknya kereta
1	A ₁	B ₁	50	70	2
1	B ₁	C ₁	10	88	1
1	C ₁	B ₁	47	90	2
1	B ₁	A ₁	25	72	1
2	A ₂	B ₂	17	70	1
2	B ₂	D ₁	30	42	1
2	D ₁	B ₂	20	40	1
2	B ₂	A ₂	2	72	1
3	C ₂	D ₂	45	76	2
3	D ₂	C ₂	25	78	1

Disini masing-masing A₁,B₁,C₁,A₂,B₂,C₂ dan D₂ bisa dianggap sebagai platfom pada stasiun A, B, C dan D. Sedangkang aturan sinkronisasi diantara kereta diberikan sebagai berikut:

- Pada jalur 1, kereta ke- $(k + 1)$ yang berangkat dari platfom B₁ ke arah C harus menunggu kedatangan kereta ke- k yang berangkat dari platfom D₁ menuju B. Kereta ke- $(k + 1)$ yang berangkat dari platfom C₁ kearah B harus menunggu kedatangan kereta ke- k yang berangkat dari platfom D₂ menuju C.
- Pada jalur 2, kereta ke- $(k + 1)$ yang berangkat dari platfom B₂ ke arah D harus menunggu kedatangan kereta ke- $(k - 1)$ (sebab ada 2 kereta pada jalur lintasan dari C₁ ke B₁) yang berangkat dari platfom C₁ menuju B. Kereta ke- $(k + 1)$ yang berangkat dari D₂ kearah B harus menunggu kedatangan kereta ke- $(k - 1)$ (sebab ada 2 kereta pada jalur lintasan dari C₂ ke D₂) yang berangkat dari C₂ menuju D.
- Pada jalur 3, kereta ke- $(k + 1)$ yang berangkat dari platfom C₂ ke arah D harus menunggu kedatangan kereta ke- k yang berangkat dari platfom B₁ menuju C. Kereta ke- $(k + 1)$ yang berangkat dari platfom D₂ kearah C harus menunggu kedatangan kereta ke- k yang berangkat dari platfom B₂ menuju D.

Selanjutnya dari informasi jadwal keberangkatan, lamanya waktu perjalanan dan posisi kereta pada saat acuan waktu seperti yang telah ditentukan serta aturan sinkronisasi yang telah diberikan dibuat suatu model sistem jaringan kereta. Bila $x_i(k)$, $i = 1, 2, \dots, 10$ masing-masing adalah keberangkatan kereta yang ke- k dari A₁ ke B₁, dari B₁ ke C₁, dari C₁ ke B₁, dari B₁ ke A₁, dari A₂ ke B₂, dari B₂ ke D₁, dari D₁ ke B₂, dari B₂ ke A₂, dari C₂ ke D₂ dan dari D₂ ke C₂, didapat suatu persamaan yang diberikan oleh bentuk berikut:

$$x(k + 1) = A \otimes x(k), \quad (3.5)$$

dimana $x(k) = (x_1(k), x_2(k), \dots, x_{13}(k))'$ dan

$$A = \begin{bmatrix} . & . & . & 72 & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & 40 & . & . & . & 70 & . & . \\ . & 88 & . & . & . & . & . & . & . & 78 & . & . & . \\ . & . & . & . & . & . & . & . & . & . & . & 90 & . \\ . & . & . & . & 70 & . & . & . & . & . & . & 90 & . \\ . & . & . & . & . & 42 & . & . & . & . & . & . & 76 \\ . & . & . & . & . & . & 40 & . & . & . & . & . & . \\ . & 88 & . & . & . & . & . & . & . & 78 & . & . & . \\ . & . & . & . & . & 42 & . & . & . & . & . & . & 76 \\ 0 & . & . & . & . & . & . & . & . & . & . & . & . \\ . & . & 0 & . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & 0 & . & . & . & . \end{bmatrix}, \quad (3.6)$$

dimana untuk alasan kemudahan dinotasikan ε dengan $.$ $= \varepsilon$. Catatan: total keseluruhan kereta adalah 13, hal ini sama dengan dimensi dari pada x pada (3.5). Peubah keadaan x_{11}, x_{12} dan x_{13} dinamakan peubah keadaan pembantu.

3.3.2 Pengkajian model yang diharapkan

Maksud dari bagian ini adalah mendiskusikan model sistem jaringan yang diharapkan yang dikaitkan dengan realita jadwal yang ada. Seperti yang dibahas pada bagian terdahulu bahwa model yang telah diturunkan yaitu (3.5), tentunya model tsb. memenuhi asumsi yang telah ditetapkan. Jika model tsb. dikaitkan dengan realita jadwal yang ada, diperoleh:

$$\left. \begin{aligned} x(k+1) &= A \otimes x(k) \oplus d(k+1) \\ y(k) &= C \otimes x(k) \\ x(0) &= x_0 \end{aligned} \right\}, \quad (3.7)$$

dimana $x(k) = (x_1(k), x_2(k), \dots, x_{13}(k))'$, A seperti diberikan pada persamaan (3.6), $d(k)$ adalah vektor jadwal keberangkatan kereta yang ke- k dan $C = (O \ \mathcal{E})'$, dimana $O = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)'$ dan $\mathcal{E} = (\varepsilon, \varepsilon, \varepsilon)'$.

3.3.3 Jadwal keberangkatan

Vektor $d(k) \in \mathbb{R}_{\varepsilon}^n$ berisi jadwal keberangkatan semua kereta yang ke- k . Karena kereta dijadwal berangkat modulo T , dalam aljabar max-plus diperoleh hubungan:

$$d(k) = d(0) \otimes T^{\otimes k} \quad (3.8)$$

berlaku untuk semua k . Ini juga bisa ditulis sebagai:

$$d(k) = d(0) + (k.T) \otimes \eta, \quad (3.9)$$

dimana $\eta = (0, 0, \dots, 0)' \in \mathbb{R}_\varepsilon^n$, vektor-satuan dalam aljabar max-plus. Jadi dalam aljabar biasa $d(k) = (d_1(0) + (k.T), d_2(0) + (k.T), \dots, d_n(0) + (k.T))'$. Catatan: sebagaimana telah diketahui dalam fariabel keadaan berisi sebagian fariabel keadaan pembantu. Begitu juga dalam vektor jadwal keberangkatan $d(k)$, berisi jadwal keberangkatan yang sesungguhnya, sedangkan sisanya bukan jadwal keberangkatan yang sesungguhnya. Pada contoh jaringan yang dibahas $d(k)$ diberikan oleh:

$$d(k) = \begin{bmatrix} 50 + 60.k \\ 10 + 60.k \\ 47 + 60.k \\ 25 + 60.k \\ 17 + 60.k \\ 30 + 60.k \\ 20 + 60.k \\ 2 + 60.k \\ 45 + 60.k \\ 25 + 60.k \\ -10 + 60.k \\ -13 + 60.k \\ -15 + 60.k \end{bmatrix}$$

dimana $d_i(k)$, $i = 1, 2, \dots, 10$ merupakan jadwal keberangkatan yang sebenarnya, sedangkan sisanya bukan jadwal yang sebenarnya. Bilangan 60 menunjukkan periodenya (modulo 60), sedangkan -10 berasal dari $50 - 60$, -13 berasal dari $47 - 60$ dan -15 berasal dari $45 - 60$. Jadwal akan bermanfaat bila sistem beroperasi dengan jadwal tsb. Dalam hal ini akan dikatakan jadwal keberangkatan kereta adalah realistik bagi sistem (3.7).

Difinisi 1 Untuk sistem (3.7) jadwal keberangkatan kereta d adalah realistik bila untuk semua $k \geq 0$

$$A \otimes d(k) \leq d(k + 1). \quad (3.10)$$

Dikatakan suatu sistem adalah realistik bila ia mempunyai suatu jadwal keberangkatan kereta yang realistik.

Jadwal keberangkatan kereta yang diberikan pada bagian sebelumnya adalah realistik, sebab:

$$A \otimes d(k) \leq d(k + 1).$$

Difinisi 2 Vektor keterlambatan $z(k)$ untuk $k \geq 0$ didefinisikan sebagai

$$z(k) = x(k) - d(k). \quad (3.11)$$

Maksimum keterlambatan dari keberangkatan kereta ke- k dinotasikan oleh:

$$\|z(k)\|_{\oplus} = \bigoplus_{i=0}^n z_i(k).$$

Berikut ini diberikan suatu teorema yang menyatakan bahwa keterlambatan dari beberapa kereta tidak akan menyebabkan meningkatnya vektor keterlambatan.

Teorema 3.3.1 *Untuk setiap kondisi awal $x(0)$ dalam suatu sistem dengan jadwal keberangkatan yang realistis, maksimum keterlambatan tidak akan meningkat, yaitu untuk semua $k \geq 0$*

$$\|z(k+1)\|_{\oplus} \leq \|z(k)\|_{\oplus}.$$

Berikut ini diberikan pengertian suatu sistem dikatakan stabil.

Definisi 3.3.1 *Suatu sistem dengan jadwal keberangkatan adalah stabil bila setiap keterlambatan awal setelah hingga beberapa keberangkatan berikutnya keterlambatan tidak muncul lagi. Secara formalnya, untuk semua x_0 ada suatu $k(x_0) \in \mathbb{N}$ sedemikian hingga $\|z(k)\|_{\oplus} = 0$, untuk semua $k \geq k(x_0)$.*

Selanjutnya diberikan suatu teorema yang memberikan syarat perlu dan cukup suatu sistem stabil.

Teorema 3.3.2 *Sistem (3.7) adalah stabil bila dan hanya bila $\lambda < T$. Dimana λ adalah nilai karakteristik dari matriks A .*

Teorema (3.3.2), memberikan suatu syarat kapan suatu sistem akan stabil. Pada bahasan berikut ini, akan dibahas lagi sistem yang telah diturunkan pada bagian sebelumnya, terutama dibahas kestabilan sistem bila terjadi beberapa keberangkatan kereta terlambat. Dari hasil yang dikaji bisa diamati keterlambatan yang terjadi besarnya semakin mengecil untuk periode keberangkatan yang berikutnya, sampai pada suatu waktu keberangkatan tertentu tidak terjadi keterlambatan keberangkatan kereta.

3.3.4 Simulasi sistem terhadap keterlambatan

Pada bagian ini diberikan suatu simulasi dari sistem yang dikaji bila beberapa kereta mengalami keterlambatan. Misalkan terjadi keterlambatan keberangkatan sebesar 8 menit masing-masing pada kereta yang berangkat dari A_1 dan C_2 . Dan terjadi keterlambatan sebesar 25 menit dan 18 menit masing-masing pada kereta yang berangkat dari B_1 ke A_1 dan dari B_2 ke A_2 . Dalam hal ini keberangkatan awal karena keterlambatan tsb. diberikan

oleh $x(0)$ dan vektor keterlambatan $z(0)$ diberikan oleh:

$$x(0) = \begin{bmatrix} 58 \\ 10 \\ 47 \\ 50 \\ 17 \\ 30 \\ 20 \\ 20 \\ 53 \\ 25 \\ -10 \\ -13 \\ -15 \end{bmatrix}, \quad z(0) = x(0) - d(0) = \begin{bmatrix} 8 \\ 0 \\ 0 \\ 25 \\ 0 \\ 0 \\ 0 \\ 18 \\ 8 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Selanjutnya dengan menggunakan persamaan (3.7) dan (3.11) diperoleh:

$$x(1) = \begin{bmatrix} 122 \\ 60 \\ 103 \\ 77 \\ 92 \\ 87 \\ 72 \\ 60 \\ 103 \\ 72 \\ 58 \\ 47 \\ 53 \end{bmatrix}, \quad z(1) = \begin{bmatrix} 12 \\ 0 \\ 0 \\ 0 \\ 15 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 8 \\ 0 \\ 8 \end{bmatrix},$$

$$x(2) = \begin{bmatrix} 149 \\ 128 \\ 150 \\ 137 \\ 132 \\ 162 \\ 129 \\ 112 \\ 150 \\ 129 \\ 122 \\ 103 \\ 103 \end{bmatrix}, \quad z(2) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 12 \\ 0 \\ 0 \\ 0 \\ 0 \\ 12 \\ 0 \\ 0 \end{bmatrix},$$

$$x(3) = \begin{bmatrix} 209. \\ 192. \\ 216. \\ 193. \\ 184. \\ 202. \\ 204. \\ 169. \\ 216. \\ 204. \\ 149. \\ 150. \\ 150. \end{bmatrix}, \quad z(3) = \begin{bmatrix} 0. \\ 2. \\ 0. \\ 0. \\ 0. \\ 0. \\ 4. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \end{bmatrix},$$

$$x(4) = \begin{bmatrix} 265. \\ 244. \\ 282. \\ 240. \\ 241. \\ 254. \\ 244. \\ 244. \\ 282. \\ 244. \\ 209. \\ 216. \\ 216. \end{bmatrix}, \quad z(4) = \begin{bmatrix} 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 2. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \end{bmatrix},$$

dan

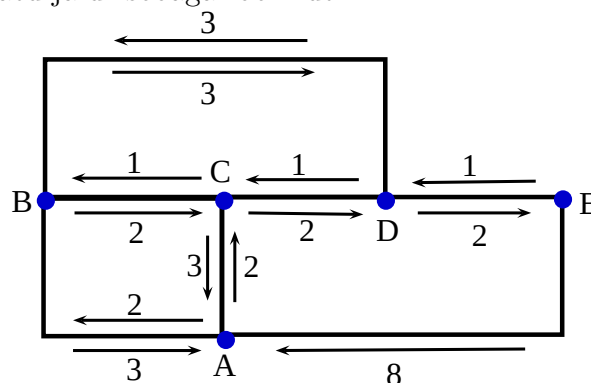
$$x(5) = \begin{bmatrix} 312. \\ 284. \\ 332. \\ 306. \\ 316. \\ 311. \\ 296. \\ 284. \\ 332. \\ 296. \\ 265. \\ 282. \\ 282. \end{bmatrix}, \quad z(5) = \begin{bmatrix} 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \end{bmatrix}$$

Terlihat bahwa akibat keterlambatan awal pada keberangkatan beberapa kereta, keterlambatan keberangkatan berikutnya terlambat sebesar 12 menit terjadi pada lintasan A₁ ke

B_1 . Keterlambatan ini menyebabkan keterlambatan berikutnya sebesar 12 menit terjadi pada lintasan B_2 ke D_1 . Sedangkan keterlambatan sebesar 15 menit pada lintasan A_2 ke B_2 menyebabkan keterlambatan sebesar 2 menit pada keberangkatan dari B_1 ke C_1 dan keterlambatan sebesar 12 menit pada jalur B_2 ke D_1 menyebabkan keterlambatan sebesar 4 menit pada keberangkatan dari D_1 ke B_2 . Keterlambatan sebesar 4 menit pada jalur D_1 ke B_2 menyebabkan keterlambatan keberangkatan sebesar 2 menit pada keberangkatan dari B_2 ke A_2 . Setelah itu untuk $k > 4$ sistem tidak mengalami keterlambatan lagi, ini berarti sistem beroperasi sesuai jadwal keberangkatan ada. Dalam hal ini dikatakan sistem mencapai keadaan stabil untuk $k > 4$. Catatan: nilai karakteristik dari matriks A yang dikaji adalah $\lambda = 56$, sedangkan $T = 60$. Jadi bukanlah hal yang mengherankan bahwa sistem yang dikaji adalah stabil, sebab $\lambda = 56 < T = 60$. Menurut Teorema 2, sistem stabil. Berikut ini diberikan ringkasan hasil apa yang telah dibahas pada keseluruhan bagian ini. Telah dikaji suatu model sistem jaringan kereta yang diturunkan dari jadwal keberangkatan kereta dengan menggunakan aljabar max-plus. Selanjutnya telah ditunjukkan pula bila terjadi keterlambatan dari beberapa kereta pada sistem yang ada keterlambatan ini akan semakin mengecil sampai pada suatu keberangkatan tertentu berikutnya sudah tidak ada keterlambatan lagi. Hal ini menunjukkan sistem kembali beroperasi dengan jadwal keberangkatan sebagaimana bila tidak ada keterlambatan. Dengan kata lain sistem yang dikaji adalah stabil.

3.4 Menentukan Jalur Tercepat

Diberikan gambar suatu jalur sebagai berikut.



Persoalan Lintasan Tercepat merupakan persoalan min plus aljabar. Matriks yang berkaitan dengan masalah ini adalah

$$A = \begin{bmatrix} \infty & 3 & 3 & \infty & 8 \\ 2 & \infty & 1 & 3 & \infty \\ 2 & 2 & \infty & 1 & \infty \\ \infty & 3 & 2 & \infty & 1 \\ \infty & \infty & \infty & 2 & \infty \end{bmatrix}$$

Untuk menentukan waktu tercepat diperlukan matriks A^* yang diberikan oleh

$$\begin{aligned}
 A^* &= A \oplus' A^2 \oplus' A^3 \oplus' A^4 \\
 &= \begin{bmatrix} 5 & 3 & 3 & 4 & 5 \\ 2 & 3 & 1 & 2 & 3 \\ 2 & 2 & 3 & 1 & 2 \\ 4 & 3 & 2 & 3 & 1 \\ 6 & 5 & 4 & 2 & 3 \end{bmatrix}
 \end{aligned}$$

Bab 4

Pengenalan Petri Nets

Petri net dikembangkan pertama kali oleh C.A. Petri pada awal 1960-an. Petri net merupakan salah satu alat untuk memodelkan sistem *event* diskrit selain menggunakan automata yang telah dikenal sebelumnya. Setiap automata dapat diubah menjadi Petri net. Pada Petri net *event* berkaitan dengan transisi. Agar suatu *event* dapat terjadi, beberapa keadaan harus dipenuhi terlebih dahulu. Informasi mengenai *event* dan keadaan ini masing-masing dinyatakan dengan transisi dan *place*. *Place* dapat berfungsi sebagai input atau output suatu transisi. *Place* sebagai input menyatakan keadaan yang harus dipenuhi agar transisi dapat terjadi. Setelah transisi terjadi maka keadaan akan berubah. *Place* yang menyatakan keadaan tersebut adalah output dari transisi.

Definisi 4.0.1 ([29]) *Petri net* adalah 4-tuple (P, T, A, w) dengan

- P : himpunan berhingga *place*, $P = \{p_1, p_2, \dots, p_n\}$,
- T : himpunan berhingga transisi, $T = \{t_1, t_2, \dots, t_m\}$,
- A : himpunan *arc*, $A \subseteq (P \times T) \cup (T \times P)$,
- w : fungsi bobot, $w : A \rightarrow \{1, 2, 3, \dots\}$.

Berdasarkan Definisi 4.0.1 maka himpunan *place* dan transisi tidak harus berupa himpunan berhingga melainkan bisa berupa himpunan takhingga terhitung (*countable sets*). Pada hampir semua kasus yang rumit dapat dimodelkan dengan Petri net yang mempunyai *place* dan transisi berhingga.

Petri net dapat digambarkan sebagai *graph* berarah. *Node* dari *graph* berupa *place* yang diambil dari himpunan *place* P atau transisi yang diambil dari himpunan transisi T . Pada *Petri net graph* diperbolehkan menggunakan beberapa *arc* untuk menghubungkan dua *node* atau ekuivalen dengan memberikan bobot ke setiap *arc* yang menyatakan banyaknya *arc*. Struktur ini dikenal dengan struktur *multigraph*.

Dalam membahas representasi Petri net secara grafik akan digunakan notasi $I(t_j)$ dan $O(t_j)$ yang masing-masing menyatakan himpunan *place* input dan output ke transisi t_j .

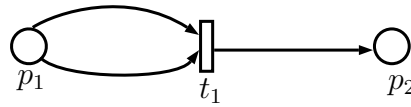
Secara matematis definisi tersebut dapat ditulis menjadi persamaan berikut [29].

$$\begin{aligned} I(t_j) &= \{p_i : (p_i, t_j) \in A\} \\ O(t_j) &= \{p_i : (t_j, p_i) \in A\} \end{aligned} \quad (4.1)$$

Notasi yang sama dapat digunakan untuk mendeskripsikan input dan output transisi untuk *place* p_i sebagai berikut.

$$\begin{aligned} I(p_i) &= \{t_j : (t_j, p_i) \in A\} \\ O(p_i) &= \{t_j : (p_i, t_j) \in A\} \end{aligned} \quad (4.2)$$

Grafik Petri net terdiri dari dua macam *node* yaitu lingkaran dan garis. Lingkaran menyatakan *place* sedangkan garis menyatakan transisi. *Arc* disimbolkan dengan panah yang menghubungkan *place* dan transisi. *Arc* yang menghubungkan *place* p_i ke transisi t_j berarti $p_i \in I(t_j)$. Jika bobot *arc* dari *place* p_i ke transisi t_j adalah k ditulis $w(p_i, t_j) = k$ maka terdapat k *arc* dari *place* p_i ke transisi t_j atau sebuah *arc* dengan bobot k .



Gambar 4.1: Petri net sederhana

Contoh 4.0.1 Perhatikan Petri net pada Gambar 4.1. Terdapat dua *place* pada Petri net tersebut yaitu p_1 dan p_2 ditulis $P = \{p_1, p_2\}$. Untuk menyatakan bahwa terdapat sebuah transisi yaitu t_1 maka ditulis $T = \{t_1\}$. *Arc* dinyatakan dengan pasangan berurutan. Elemen pertama menyatakan asal dan elemen kedua menyatakan tujuan misalnya *arc* dari *place* p_1 ke transisi t_1 ditulis (p_1, t_1) dan (t_1, p_2) menyatakan *arc* dari transisi t_1 ke *place* p_2 . Secara lengkap ditulis $A = \{(p_1, t_1), (t_1, p_2)\}$. Bobot *arc* dari *place* p_1 ke transisi t_1 adalah dua yaitu $w(p_1, t_1) = 2$ dan bobot dari transisi t_1 ke *place* p_2 adalah satu yaitu $w(t_1, p_2) = 1$. Pada contoh ini $I(t_1) = \{p_1\}$ dan $O(t_1) = \{p_2\}$. □

Terlihat pada Contoh 4.0.1 bahwa bobot *arc* dari *place* p_1 ke transisi t_1 adalah 2 dan digambarkan dengan dua buah *arc*. Bobot *arc* dari transisi t_1 ke *place* p_2 adalah satu. Transisi tidak harus mempunyai *place* input dan *place* output seperti transisi t_1 pada Gambar 4.1. Kadang transisi tidak mempunyai *place* input. Ini berarti *event* yang dinyatakan oleh transisi tersebut tidak membutuhkan kondisi untuk dapat terjadi. Transisi t_2 pada Petri net Contoh 4.0.2 tidak mempunyai *place* input. Berikut merupakan penjelasan dari Petri net pada Gambar 4.2 beserta cara identifikasi dan penulisan *place*, transisi, *arc* dan bobotnya.



Gambar 4.2: Petri net dengan Transisi yang Tidak Mempunyai Place Input

Contoh 4.0.2 Petri net pada Gambar 4.2 mempunyai dua *place* dan dua transisi yang masing-masing dapat ditulis $P = \{p_1, p_2\}$ dan $T = \{t_1, t_2\}$. *Arc* dinyatakan dengan pasangan berurutan misalnya *arc* dari *place* p_1 ke transisi t_1 dinotasikan dengan p_1, t_1 . Jumlah *arc* pada Petri net tersebut sebanyak 4 yang ditulis $A = \{(p_1, t_1), (t_1, p_2), (p_2, t_1), (t_2, p_2)\}$. Berikut merupakan bobot pada masing-masing *arc*

$$w(p_1, t_1) = 2, \quad w(t_1, p_2) = w(p_2, t_1) = w(t_2, p_2) = 1 \quad (4.3)$$

Terlihat dari Gambar 4.2 bahwa $I(t_1) = P$, yang menyatakan bahwa semua *place* pada Petri net merupakan input dari transisi t_1 . Jelas bahwa himpunan $O(t_1) = \{p_2\} = O(t_2)$ dan $I(t_2) = \emptyset$ karena tidak ada *place* yang menjadi input dari transisi t_2 . $\square$

Adakalanya berguna untuk membedakan antara Petri net yang *pure* dan *impure*. Petri net disebut *pure* jika tidak ada *place* yang menjadi input dan output untuk suatu transisi. Jika terdapat *place* yang menjadi input dan output untuk transisi tertentu maka Petri net dikatakan *impure*.

Definisi 4.0.2 ([29]) *Petri net dikatakan pure jika tidak mempunyai place yang menjadi input sekaligus output untuk suatu transisi. Secara formal ditulis*

$$\nexists p_i \in P, t_j \in T \ni \{(p_i, t_j), (t_j, p_i)\} \subseteq A \quad (4.4)$$

Jelas bahwa Petri net pada Gambar 4.1 adalah *pure* karena tidak ada *place* yang menjadi input sekaligus output untuk suatu transisi sedangkan Petri net pada Gambar 4.2 adalah *impure*, karena p_2 adalah *place* input dan output untuk transisi t_1 .

4.1 Tanda Petri net dan Ruang Keadaan

Transisi pada Petri net menyatakan *event* pada sistem *event* diskrit dan *place* merepresentasikan kondisi agar *event* dapat terjadi. Diperlukan mekanisme untuk mengindikasikan apakah kondisi telah terpenuhi. *Token* adalah sesuatu yang diletakkan di *place* yang menyatakan terpenuhi tidaknya suatu kondisi. Secara grafik *token* digambarkan dengan *dot* dan diletakkan di dalam *place*. Jika banyaknya *token* besar maka dituliskan dengan angka.

Definisi 4.1.1 ([29]) *Penanda (marking) x pada Petri net adalah fungsi*

$$x : P \rightarrow \{0, 1, 2, \dots\}.$$

Penanda dinyatakan dengan vektor berisi bilangan bulat taknegatif yang menyatakan banyaknya *token* yaitu $\mathbf{x} = [x(p_1), x(p_2), \dots, x(p_n)]^T$. Banyaknya elemen $\mathbf{x}$ sama dengan banyak *place* dalam Petri net. Elemen ke- i pada vektor ini merupakan banyaknya *token* pada *place* p_i , dengan demikian $x(p_i) \in \{0, 1, 2, \dots\}$.

Definisi 4.1.2 ([29]) *Petri net bertanda (marked) adalah 5-tuple $(P, T, A, w, \mathbf{x}_0)$ dimana (P, T, A, w) adalah Petri net dan $\mathbf{x}_0$ adalah penanda awal*

Selanjutnya Petri net bertanda cukup disebut Petri net. Seperti pemodelan sistem pada umumnya, maka harus didefinisikan keadaan (*state*) pada Petri net. Keadaan pada Petri net adalah penanda Petri net.

Definisi 4.1.3 ([29]) *Keadaan (state) Petri net bertanda adalah*

$$\mathbf{x} = [x(p_1), x(p_2), \dots, x(p_n)]^T.$$

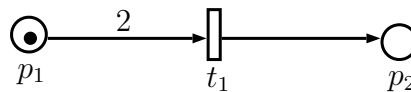
Perhatikan bahwa banyaknya *token* pada *place* adalah sebarang bilangan bulat taknegatif, tidak harus terbatas (*bounded*). Secara umum banyaknya penanda yang mungkin adalah takhingga. Ruang keadaan (*state space*) X pada Petri net bertanda dengan n *place* didefinisikan oleh semua vektor berdimensi n dengan elemen bilangan bulat taknegatif, jadi $X = \{0, 1, 2, \dots\}^n$. Untuk selanjutnya digunakan istilah keadaan dan penanda akibat adanya perubahan tanda.

Jika semua keadaan yang diperlukan sudah terpenuhi maka transisi dapat terjadi. Dalam hal ini keadaan merupakan *place* input dari transisi. Bobot *arc* dari *place* input ke transisi menunjukkan banyaknya *token* minimum di *place* agar transisi *enable*. Jika semua *place* input mempunyai *token* lebih dari atau sama dengan banyaknya *token* minimum yang dibutuhkan maka transisi *enable*. Secara formal didefinisikan sebagai berikut.

Definisi 4.1.4 ([29]) *Transisi $t_j \in T$ dalam Petri net bertanda dikatakan **enable** jika*

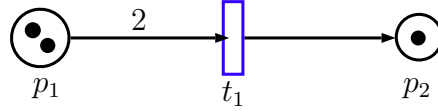
$$x(p_i) \geq w(p_i, t_j), \quad \forall p_i \in I(t_j) \quad (4.5)$$

Gambar 4.3 merupakan contoh transisi yang tidak *enable*. Jelas bahwa $I(t_1) = \{p_1\}$, $x(p_1) = 1$ dan $w(p_1, t_1) = 2$ seperti yang terlihat pada Gambar 4.3 Transisi t_1 tidak *enable* karena $1 = x(p_1) < w(p_1, t_1) = 2$.



Gambar 4.3: Contoh Transisi yang Tidak *Enabled*

Transisi t_1 *enable* jika banyaknya *token* pada *place* p_1 lebih dari atau sama dengan 2. Terlihat pada Petri net Gambar 4.4 bahwa $x(p_1) = 2$ sehingga transisi t_1 dalam Petri net

Gambar 4.4: Contoh Transisi yang *Enabled*

tersebut *enable*. Dari sini dapat disimpulkan agar transisi t_j *enable* maka banyaknya *token* pada *place* p_i paling sedikit sebesar bobot *arc* yang menghubungkan p_i ke t_j . Kenyataan ini sesuai dengan definisi transisi *enable* yang telah dituliskan sebelumnya.

4.2 Dinamika Petri net

Jika Petri net digunakan untuk memodelkan sistem dinamik *event* diskrit, seharusnya Petri net dilengkapi dengan mekanisme yang mirip dengan transisi keadaan (*state transition*) pada automata. Mekanisme ini berupa menjalankan *token* melewati jaringan (*net*) ketika transisi menjadi *enable* dan proses ini mengubah keadaan Petri net.

Hanya transisi *enable* yang dapat *difire*. Transisi *difire* saat *event* yang dinyatakan oleh transisi terjadi. Berikut ini adalah proses yang terjadi pada pemfirean transisi. Semua *token* di *place* input dikurangi/diambil sebanyak bobot *arc* yang menghubungkannya. Berdasarkan Definisi 4.1.4 maka banyaknya *token* di *place* input setelah dikurangi adalah bilangan bulat taknegatif. *Token* di *place* output ditambah sebanyak bobot *arc* yang menghubungkannya.

Definisi 4.2.1 ([29]) Fungsi perubahan keadaan pada Petri net bertanda $(P, T, A, w, \mathbf{x}_0)$ yaitu $f : \{0, 1, 2, \dots\}^n \times T \rightarrow \{0, 1, 2, \dots\}^n$ terdefinisi untuk transisi $t_j \in T$ jika dan hanya jika

$$x(p_i) \geq w(p_i, t_j), \quad \forall p_i \in I(t_j) \quad (4.6)$$

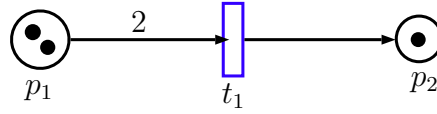
Jika $f(\mathbf{x}, t_j)$ terdefinisi maka ditulis $\mathbf{x}' = f(\mathbf{x}, t_j)$, dimana

$$x'(p_i) = x(p_i) - w(p_i, t_j) + w(t_j, p_i), \quad i = 1, 2, \dots, n \quad (4.7)$$

Kondisi (4.6) menjamin fungsi perubahan keadaan hanya didefinisikan untuk transisi yang *enable*. Fungsi perubahan keadaan didasarkan pada struktur Petri net. Keadaan berikutnya yang didefinisikan pada (4.7) secara eksplisit tergantung dari fungsi bobot pada setiap input dan output pada transisi.

Berdasarkan (4.7), jika p_i adalah *place* input untuk transisi t_j , maka *token* pada *place* p_i berkurang sebanyak bobot *arc* dari p_i ke t_j . Sebaliknya jika p_i adalah *place* output dari transisi t_j maka *token* pada *place* p_i bertambah sebesar bobot *arc* dari t_j ke p_i . Mungkin p_i adalah *place* input dan output dari transisi t_j sehingga menurut (4.7) *token* pada *place* p_i berkurang sebanyak $w(p_i, t_j)$ dan bertambah sebanyak $w(t_j, p_i)$.

Perhatikan bahwa banyaknya *token* pada Petri net bertanda tidak tetap. Jelas dari (4.7) bahwa mungkin $w(t_j, p_i) > w(p_i, t_j)$ sehingga *token* yang ditambahkan pada *place* p_i lebih banyak daripada yang diambil ketika transisi t_j di*fire*. Secara umum *token* pada Petri net mungkin habis setelah beberapa kali pem*fire*an atau banyaknya bertambah menuju takhingga.



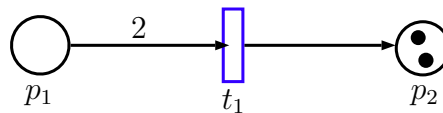
Gambar 4.5: Sebelum Transisi t_1 Di*fire*

Berikut dibahas proses pem*fire*an Petri net pada Gambar 4.5 dan keadaan Petri net setelah terjadi pem*fire*an transisi.

Contoh 4.2.1 Inisialisasi banyaknya *token* pada p_1 dan p_2 masing-masing adalah 2 dan 1 sehingga penanda awal Petri net adalah $\mathbf{x}_0 = [x(p_1), x(p_2)]^T = [2, 1]^T$. Arc dari p_1 ke t_1 mempunyai bobot 2 ditulis $w(p_1, t_1) = 2$.

Jelas bahwa $x(p_1) = 2 \geq w(p_1, t_1)$ sehingga transisi t_1 *enable* dan dapat di*fire*. Untuk mem*fire* transisi t_1 dibutuhkan dua *token* dari *place* p_1 sehingga *token* di *place* tersebut habis dan sebuah *token* ditambahkan di *place* p_2 . Penanda Petri net berubah menjadi $\mathbf{x}' = [0, 2]^T$ setelah transisi t_1 di*fire* dan bisa ditulis $\mathbf{x}' = f(\mathbf{x}_0, t_1)$. Transisi t_1 tidak *enable* karena tidak ada *token* pada *place* p_1 . Pada keadaan ini tidak ada transisi yang *enable* karena Petri net pada Gambar 4.5 mempunyai sebuah transisi yaitu t_1 . $\square$

Keadaan dimana tidak ada transisi yang *enable* seperti yang terlihat pada Gambar 4.6 disebut keadaan *terminal* dan Petri net mengalami *deadlock*. Petri net yang baik seharusnya menghindari terjadinya *deadlock*. Pembahasan mengenai *deadlock* secara lebih terperinci dapat dilihat pada bagian analisis Petri net takberwaktu.



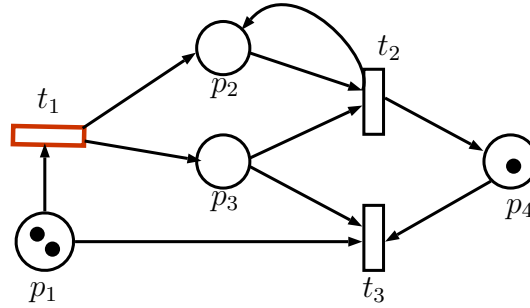
Gambar 4.6: Sesudah Transisi t_1 Di*fire*

Token pada *place* dan bobot *arc* ditulis secara ringkas agar penulisan lebih sederhana. Dalam hal ini digunakan notasi vektor. Untuk menyatakan $[x(p_1), x(p_2), \dots, x(p_n)]^T$ ditulis $x([p_1, p_2, \dots, p_n]^T)$ dan untuk menyatakan $[w(p_1, t_j), w(p_2, t_j), \dots, w(p_n, t_j)]^T$ dapat ditulis $w([p_1, p_2, \dots, p_n]^T, t_j)$ dengan $1 \leq j \leq m$. Pengertian ini dituliskan dalam definisi berikut.

Definisi 4.2.2 Diberikan (P, T, A, w) Petri net. Jika $\{p'_1, p'_2, \dots, p'_{n1}\} \subseteq P$ dan $t \in T$ maka berlaku

$$\begin{aligned} x([p'_1, p'_2, \dots, p'_{n1}]^T) &\stackrel{\text{def}}{=} [x(p'_1), x(p'_2), \dots, x(p'_{n1})]^T \\ w([p'_1, p'_2, \dots, p'_n]^T, t) &\stackrel{\text{def}}{=} [w(p'_1, t), w(p'_2, t), \dots, w(p'_n, t)]^T \end{aligned} \quad (4.8)$$

Untuk memperjelas penggunaan Definisi 4.2.2, berikut dibahas penggunaan notasi vektor pada permasalahan pemfirean transisi. Tujuan penggunaan notasi vektor adalah mempersingkat penulisan.



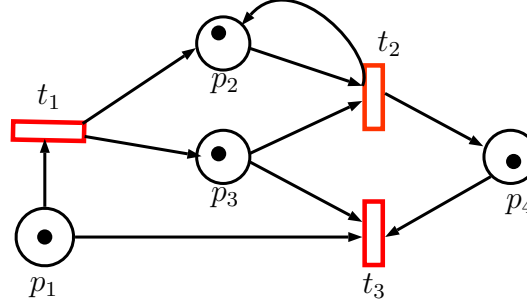
Gambar 4.7: Keadaan Awal Petri net Bertanda

Pada contoh berikut dibahas pemfirean Petri net pada Gambar 4.7. Disini terdapat dua hal yang penting yaitu proses pemfirean dan penggunaan notasi pada Definisi 4.2.2.

Contoh 4.2.2 Gambar 4.7 merupakan keadaan awal Petri net sebagai contoh ilustrasi pemfirean transisi dan perubahan keadaan. Place input dari transisi t_1 adalah p_1 sedangkan place outputnya adalah p_2 dan p_3 yang dapat dinotasikan $I(t_1) = \{p_1\}$ dan $O(t_1) = \{p_2, p_3\}$. Place input dari transisi t_2 adalah p_2 dan p_3 sedangkan p_2 dan p_4 merupakan place outputnya yang dapat ditulis $I(t_2) = \{p_2, p_3\}$ dan $O(t_2) = \{p_2, p_4\}$. Transisi t_3 mempunyai place input p_1, p_3 dan p_4 tetapi tidak mempunyai place output, yaitu $I(t_3) = \{p_1, p_3, p_4\}$ dan $O(t_3) = \emptyset$.

Keadaan awal Petri net adalah $\mathbf{x}_0 = [x(p_1), x(p_2), x(p_3), x(p_4)]^T$ yaitu $\mathbf{x}_0 = [2, 0, 0, 1]^T$. Terlihat pada Gambar 4.7 bahwa satu-satunya transisi yang *enable* adalah t_1 , karena transisi tersebut hanya membutuhkan sebuah *token* dari place p_1 dan *token* pada place p_1 berjumlah dua, yaitu $x(p_1) = 2$. Jelas bahwa $2 = x(p_1) \geq w(p_1, t_1) = 1$ sehingga transisi t_1 memenuhi persamaan (4.6). Transisi t_2 tidak *enable* karena tidak ada *token* pada place p_2 dan p_3 yang dapat ditulis $\mathbf{0} = x([p_2, p_3]^T) < w([p_2, p_3]^T, t_2) = \mathbf{1}$. Notasi $\mathbf{0}$ dan $\mathbf{1}$ menyatakan vektor yang semua elemennya masing-masing 0 dan 1. Jumlah elemen pada vektor tergantung kondisi, pada pertidaksamaan tersebut vektor terdiri dari dua elemen. Transisi t_3 tidak *enable* karena $[2, 0, 1]^T = x([p_1, p_3, p_4]^T) \not\geq w([p_1, p_3, p_4]^T, t_3) = \mathbf{1}$.

Ketika transisi t_1 difire maka sebuah *token* diambil dari place p_1 dan sebuah *token* ditambahkan ke place p_2 dan p_3 seperti pada Gambar 4.8. Persamaan (4.7) juga dapat

Gambar 4.8: Keadaan Petri net Setelah Transisi t_1 Difire

digunakan untuk mengetahui keadaan berikutnya.

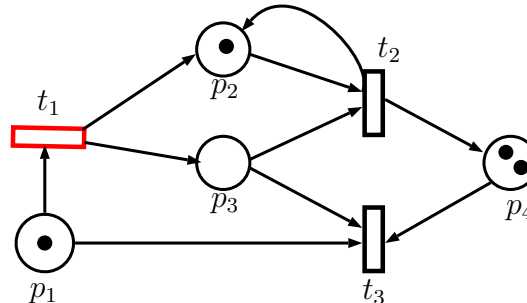
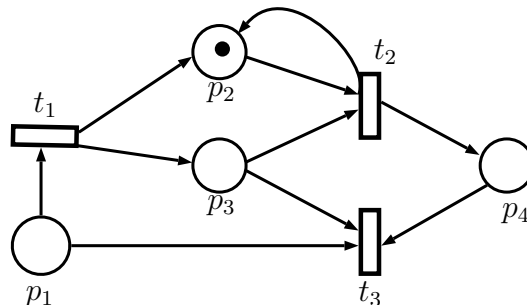
Keadaan Petri net menjadi $\mathbf{x}_1 = \mathbf{1}$ seperti terlihat pada Gambar 4.8. Transisi t_1 *enable* karena ada satu *token* di p_1 , ditulis $x(p_1) = 1 = w(p_1, t_1)$. Kondisi $x([p_2, p_3]^T) = \mathbf{1} = w([p_2, p_3]^T, t_2)$ menyebabkan transisi t_2 *enable*. Pemfirean dapat dilakukan pada transisi t_3 karena terdapat *token* pada *place* p_1 , p_3 dan p_4 , yaitu $x([p_1, p_3, p_4]^T) = \mathbf{1} = w([p_1, p_3, p_4]^T, t_3)$. Jadi pada keadaan ini semua transisi *enable* karena semua *place* sudah terisi *token*.

Selanjutnya misal transisi t_2 difire. Pada setiap *place* input dari transisi t_2 , *token* dikurangi satu. Dalam hal ini pada *place* p_2 dan p_3 sehingga kedua *place* tersebut tidak mempunyai *token*. *Place* output dari transisi t_2 adalah p_2 dan p_4 . Sebuah *token* ditambahkan ke *place* p_2 sehingga *place* p_2 kembali mempunyai sebuah *token*. Ingat bahwa *place* p_2 adalah *place* input sekaligus *place* output dari transisi t_2 yang ditulis $p_2 \in I(t_2) \cap O(t_2)$. Selain *place* p_2 , *place* output dari transisi t_2 adalah *place* p_4 sehingga pada *place* tersebut ditambahkan sebuah *token*. Jumlah *token* yang berada pada *place* p_4 bertambah menjadi dua.

Keadaan Petri net menjadi $\mathbf{x}_2 = [1, 1, 0, 2]^T$ yang ditunjukkan oleh Petri net pada Gambar 4.9. Transisi t_1 tetap *enable* karena *place* p_1 mempunyai sebuah *token* dan *arc* yang menghubungkan *place* p_1 ke transisi t_1 mempunyai bobot satu yang dapat ditulis $x(p_1) = 1 = w(p_1, t_1)$. Tidak adanya *token* pada *place* p_3 menyebabkan transisi t_2 dan t_3 tidak *enable*. Hal ini dapat ditulis sebagai $[1, 0]^T = x([p_2, p_3]^T) \not\geq w([p_2, p_3]^T, t_2) = \mathbf{1}$ dan $[1, 0, 2]^T = x([p_1, p_3, p_4]^T) \not\geq w([p_1, p_3, p_4]^T, t_3) = \mathbf{1}$. Jadi pada keadaan ini satu-satunya transisi yang *enable* adalah t_1 .

Mari kembali ke keadaan $\mathbf{x}_1$ yaitu Petri net pada Gambar 4.8 dimana semua transisi *enable*. Jika sebelumnya transisi yang difire adalah t_1 maka sekarang transisi t_3 difire. *Token* pada setiap *place* input dari transisi t_3 dikurangi satu yaitu pada *place* p_1 , p_3 dan p_4 . Perhatikan bahwa tidak ada *place* output dari transisi t_3 . Keadaan ini juga dinotasikan dengan $\mathbf{x}_2$ karena pemfirean sudah dilakukan dua kali sejak keadaan $\mathbf{x}_0$.

Keadaan Petri net pada Gambar 4.10 menjadi $\mathbf{x}_2 = [0, 1, 0, 0]^T$. Transisi t_1 tidak *enable* karena $0 = x(p_1) < w(p_1, t_1) = 1$ yaitu tidak ada *token* pada *place* p_1 . Kondisi $[1, 0]^T = x([p_2, p_3]^T) \not\geq w([p_2, p_3]^T, t_2) = \mathbf{1}$ menyebabkan transisi t_2 tidak *enable*. Transisi

Gambar 4.9: Keadaan Petri net Setelah Transisi t_1 Difire Kemudian t_2 Gambar 4.10: Keadaan Petri net Setelah Transisi t_1 Difire Kemudian t_3

t_3 juga tidak *enable* karena $\mathbf{0} = x([p_1, p_3, p_4]^T) < w([p_1, p_3, p_4]^T, t_3) = \mathbf{1}$, yaitu semua *place* input dari t_3 tidak mempunyai *token*. Jadi semua transisi tidak *enable* dengan keadaan $\mathbf{x}_2 = [0, 1, 0, 0]^T$. Keadaan dimana Petri net mengalami *deadlock* disebut keadaan *terminal* dari Petri net. □

Urutan pemfirean transisi tidak diberikan pada Petri net. Seperti contoh sebelumnya, pada keadaan $\mathbf{x}_1$ ketiga transisi dapat difire. Tidak adanya data mengenai transisi mana yang difire bukan hal yang mengejutkan karena model sistem *event* diskrit yang dibahas adalah tanpa waktu (*untimed*). Pada model Petri net ini tidak ada informasi terjadinya *event* (transisi). Dengan tidak adanya mekanisme untuk menentukan transisi berikutnya maka harus diteliti setiap kemungkinan urutan pemfirean secara terpisah.

Dengan menggunakan definisi yang telah dibahas pada bagian ini, sulit untuk mengimplementasikan Petri net ke dalam program. Pada pembahasan berikutnya dikaji representasi Petri net dalam matriks yang bertujuan memudahkan implementasi Petri net ke program. Representasi dengan matriks dapat digunakan untuk mendesain Petri net dengan banyaknya *place* dan transisi yang besar. Hal ini sulit dilakukan secara visual.

4.3 Representasi Petri net Menggunakan Matriks

Pada bagian ini dikaji representasi Petri net dalam notasi matriks. Seperti yang telah dikutip pada bagian sebelumnya, hal ini bertujuan untuk memudahkan dalam implemen-

tasi. Petri net dapat direpresentasikan dalam dua matriks yang disebut *backward incidence* dan *forward incidence*. Kedua matriks ini masing-masing berukuran $n \times m$ dengan n adalah banyaknya *place* dan m adalah banyaknya transisi. Elemen matriks ini adalah bilangan bulat taknegatif.

Matriks *backward incidence* dan *forward incidence* masing-masing serupa dengan definisi *place* input dan output pada persamaan (4.1). Elemen pada matriks *backward incidence* merupakan bobot *arc* yang menghubungkan *place* ke transisi. Jika tidak ada *arc* yang menghubungkan *place* ke transisi maka bobot *arc* diisi nol. Jelas bahwa *place* ini adalah *place* input dari transisi. Definisi serupa juga digunakan untuk matriks *forward incidence*. Bedanya elemen pada matriks ini merupakan bobot *arc* yang menghubungkan transisi ke *place* sehingga merupakan *place* output dari transisi.

Definisi 4.3.1 *Matriks backward (forward) incidence yang merepresentasikan Petri net adalah matriks berukuran $n \times m$ dengan elemen baris ke- i kolom ke- j adalah $A_b(i, j) \stackrel{\text{def}}{=} w(p_i, t_j)$ ($A_f(i, j) \stackrel{\text{def}}{=} w(t_j, p_i)$).*

Salah satu kegunaan matriks *backward incidence* adalah menentukan transisi yang *enable*. Perhatikan kembali persamaan (4.5), persamaan itu berlaku hanya untuk *place* input. Jika p_i bukan merupakan *place* input dari transisi t_j yaitu $p_i \notin I(t_j)$ maka bobot *arc* dari *place* p_i ke transisi t_j adalah nol karena tidak ada *arc* yang menghubungkannya, ditulis $w(p_i, t_j) = 0$. Persamaan (4.5) pasti benar karena $x(p_i) \geq 0$. Jadi persamaan (4.5) berlaku untuk semua *place* sehingga dapat ditulis dalam notasi vektor sebagai berikut.

$$\begin{aligned} x([p_1, \dots, p_n]^T) &\geq w([p_1, \dots, p_n]^T, t_j) \\ &= A_b(:, j) \\ &= A_b \mathbf{e}_j \end{aligned} \tag{4.9}$$

dengan $A_b(:, j)$ menyatakan kolom ke- j dari matriks A_b dan $\mathbf{e}_j$ merupakan kolom ke- j matriks identitas berorder m . Terlihat pada persamaan (4.9) transisi t_j *enable* jika vektor keadaan lebih besar atau sama dengan kolom ke- j dari matriks *backward incidence* yaitu $\mathbf{x} \geq A_b(:, j)$. Dengan kata lain penentuan transisi yang *enable* dapat dilakukan dengan mencari kolom dari matriks *backward incidence* yang kurang dari atau sama dengan vektor keadaan. Persamaan (4.9) dapat ditulis lebih ringkas menjadi

$$\mathbf{x} \geq A_b \mathbf{e}_j \tag{4.10}$$

Perhatikan persamaan (4.7) yang mendeskripsikan bagaimana tanda pada *place* berubah ketika suatu transisi *difire*. Selanjutnya akan diturunkan bentuk persamaan vektor dari (4.7) untuk memperoleh keadaan Petri net berikutnya yang dinotasikan $\mathbf{x}' = x'([p_1, p_2, \dots, p_n]^T)$ jika keadaan saat ini adalah $\mathbf{x} = x([p_1, p_2, \dots, p_n]^T)$ dan kenyataan bahwa suatu transisi $t_j \in T$ telah *difire* dengan $1 \leq j \leq m$. Perhatikan bahwa persamaan (4.7) berlaku untuk semua *place* di Petri net. Dengan menggunakan notasi vektor yang telah didefinisikan

sebelumnya maka persamaan (4.7) dapat ditulis kembali menjadi persamaan yang lebih ringkas dan sederhana.

$$x'([p_1, \dots, p_n]^T) = x([p_1, \dots, p_n]^T) - w([p_1, \dots, p_n]^T, t_j) + w(t_j, [p_1, \dots, p_n]^T) \quad (4.11)$$

Berdasarkan Definisi 4.3.1 maka $w([p_1, \dots, p_n]^T, t_j)$ merupakan kolom ke- j matriks *backward incidence* yang dinotasikan $A_b(:, j)$. Notasi $A_f(:, j)$ menyatakan kolom ke- j matriks *forward incidence* yaitu $w(t_j, [p_1, \dots, p_n]^T)$ pada persamaan (4.11). Dengan menggunakan notasi A_f dan A_b maka persamaan (4.11) dapat ditulis menjadi

$$\begin{aligned} x'([p_1, \dots, p_n]^T) &= x([p_1, \dots, p_n]^T) - A_b(:, j) + A_f(:, j) \\ &= x([p_1, \dots, p_n]^T) - A_b \mathbf{e}_j + A_f \mathbf{e}_j \\ &= x([p_1, \dots, p_n]^T) + (-A_b + A_f) \mathbf{e}_j \\ &= x([p_1, \dots, p_n]^T) + A \mathbf{e}_j \end{aligned} \quad (4.12)$$

dengan $\mathbf{e}_j$ adalah kolom ke- j dari matriks identitas berorder m dan $A = A_f - A_b$. Matriks A disebut matriks *combined incidence* atau matriks *incidence*. Elemen matriks ini adalah bilangan bulat yang merupakan selisih bobot *arc place* input dan output yaitu $A(i, j) = w(t_j, p_i) - w(p_i, t_j)$.

Terlihat dari persamaan (4.12) bahwa nilai j menentukan indeks transisi yang difire. Dengan menggunakan notasi vektor maka persamaan (4.12) dapat dinyatakan sebagai

$$\mathbf{x}' = \mathbf{x} + A \mathbf{u} \quad (4.13)$$

dengan $\mathbf{u}$ menyatakan vektor kolom yang mempunyai elemen sebanyak m yaitu diperoleh dari kolom matriks identitas.

Contoh 4.3.1 Petri net yang dibahas disini adalah Petri net pada Gambar 4.5 di Contoh 4.2.1. Terdapat dua *place* dan sebuah transisi sehingga $n = 2$ dan $m = 1$. Matriks *backward incidence* dan *forward incidence* berukuran 2×1 yaitu $A_b = [2, 0]^T$ dan $A_f = [0, 1]^T$. Keadaan awal Petri net dapat ditulis $\mathbf{x}_0 = [2, 1]^T$. Berdasarkan persamaan (4.10) maka transisi t_1 *enable* karena $\mathbf{x}_0 \geq A_b$ sehingga transisi t_1 dapat difire.

Sebelum memfire transisi t_1 dihitung matriks *incidence* terlebih dahulu supaya dapat menggunakan persamaan (4.13) untuk menentukan keadaan berikutnya. Matriks *incidence* merupakan selisih antara *forward incidence* dan *backward incidence* yaitu $A = A_f - A_b = [0, 1]^T - [2, 0]^T = [-2, 1]^T$. Dengan menggunakan persamaan (4.13) diperoleh $\mathbf{x}_1 = [2, 1]^T + [-2, 1]^T = [0, 2]^T$ yang sama dengan keadaan Petri net setelah transisi t_1 difire seperti pada Gambar 4.6. □

Setelah mengetahui proses pencarian transisi yang *enable* dan proses pemfirean transisi yang *enable* pada Contoh 4.2.1, berikutnya dikaji Petri net pada Contoh 4.2.2.

Contoh 4.3.2 Terdapat empat *place* dan tiga transisi pada Petri net ini sehingga $n = 4$ dan $m = 3$. Matriks *backward incidence* dan *forward incidence* masing-masing berukuran 4×3 sebagai berikut.

$$A_b = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}, \quad A_f = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (4.14)$$

Keadaan awal Petri net adalah $\mathbf{x}_0 = [2, 0, 0, 1]^T$. Transisi t_1 *enable* karena $\mathbf{x}_0 \geq A_b(:, 1)$ sedangkan transisi t_2 dan t_3 tidak *enable* karena $\mathbf{x}_0 \not\geq A_b(:, j)$ untuk $j = 2, 3$. Selanjutnya dihitung matriks *incidence* yang dapat digunakan untuk menentukan keadaan berikutnya.

$$A = A_f - A_b = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & -1 \\ 1 & 0 & 0 \\ 1 & -1 & -1 \\ 0 & 1 & -1 \end{bmatrix} \quad (4.15)$$

Untuk menentukan keadaan berikutnya, gunakan persamaan (4.13). Transisi yang *difire* adalah t_1 karena transisi tersebut *enable*.

$$\mathbf{x}_1 = \mathbf{x}_0 + A \mathbf{e}_1 = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} -1 & 0 & -1 \\ 1 & 0 & 0 \\ 1 & -1 & -1 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} -1 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \quad (4.16)$$

Dari persamaan (4.16) diperoleh $\mathbf{x}_1 = [1, 1, 1, 1]^T$ sama seperti Petri net pada Gambar 4.8. Tidak sulit untuk melihat bahwa $\mathbf{x}_1 \geq A_b(:, j)$ dengan $1 \leq j \leq 3$ sehingga semua transisi *enable*. Transisi yang *difire* berikutnya adalah t_2 agar urutan transisi yang *difire* sama dengan Contoh 4.2.2. Cara yang sama digunakan untuk memperoleh keadaan setelah transisi t_2 *difire*.

$$\mathbf{x}_2 = \mathbf{x}_1 + A \mathbf{e}_2 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} + \begin{bmatrix} -1 & 0 & -1 \\ 1 & 0 & 0 \\ 1 & -1 & -1 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 2 \end{bmatrix} \quad (4.17)$$

Keadaan berikutnya adalah $\mathbf{x}_2 = [1, 1, 0, 2]^T$ yang sama dengan keadaan Petri net pada Gambar 4.9. Pada keadaan ini satu-satunya transisi yang *enable* adalah t_1 karena $\mathbf{x}_2 \geq A_b(:, 1)$ dan $\mathbf{x}_2 \not\geq A_b(:, j)$ untuk $j = 2, 3$.

Telah diketahui bahwa pada keadaan $\mathbf{x}_1$, semua transisi *enable*. Jika sebelumnya transisi yang *difire* adalah t_2 maka sekarang dipilih transisi yang lain untuk *difire* yaitu t_3 .

$$\mathbf{x}_2 = \mathbf{x}_1 + A \mathbf{e}_2 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} + \begin{bmatrix} -1 & 0 & -1 \\ 1 & 0 & 0 \\ 1 & -1 & -1 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} + \begin{bmatrix} -1 \\ 0 \\ -1 \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad (4.18)$$

Perhatikan bahwa persamaan (4.18) menyatakan keadaan Petri net pada Gambar 4.10. Pada keadaan ini tidak ada transisi yang *enable* karena $\mathbf{x}_2 \not\geq \mathbf{A}_b(:, j)$ untuk $1 \leq j \leq 3$ sehingga Petri net mengalami *deadlock*. $\square$

Permasalahan yang akan dikaji berikutnya adalah analisis sistem *event* diskrit. Aspek yang dianalisis pada sistem *event* diskrit meliputi keterbatasan (*boundedness*), *liveness*, *deadlock* dan lain-lain. Analisis sistem dapat digunakan untuk menentukan performa sistem *event* diskrit dan menentukan kestabilan sistem.

4.4 Analisis Model Sistem *Event* Diskrit Tak Berwaktu

Analisis model sistem *event* diskrit takberwaktu terdiri dari pengkajian logika dan perilaku kualitatifnya. Saat mendesain sistem *event* diskrit diberikan spesifikasi yang harus dipenuhi. Spesifikasi berupa menghindari keadaan yang "tidak diinginkan" atau mencapai keadaan yang "diinginkan" dalam waktu tertentu. Setelah sistem *event* diskrit sudah didesain maka sistem harus diverifikasi apakah sudah memenuhi spesifikasi yang diberikan.

Langkah pertama adalah mengkategorikan beberapa permasalahan yang berhubungan dengan analisis sistem *event* diskrit. Permasalahan ini berhubungan dengan sistem tanpa memperhatikan representasi yang digunakan misalnya Petri net atau automata. Agar terminologi dan notasi yang digunakan konsisten maka dipilih salah satu representasi sebagai acuan. Pada penelitian ini Petri net digunakan sebagai acuan karena sifatnya yang umum sehingga memudahkan dalam menunjukkan keterbatasan teknik yang dikaji.

Sebelum mengkaji permasalahan yang berhubungan dengan analisis sistem *event* diskrit, terlebih dahulu dibahas teknik analisis yang digunakan. Teknik analisis yang digunakan adalah *coverability tree*. Beberapa permasalahan dapat dianalisis menggunakan teknik ini meskipun teknik ini juga mempunyai kelemahan.

4.4.1 *Liveness* dan *Deadlocks*

Pada kajian sebelumnya sering dijumpai istilah *deadlock* yang secara mudah berarti keadaan dimana tidak ada transisi yang *enable*. *Deadlock* dapat disebabkan persaingan memperoleh *resource*. Ketika semua pihak tidak memperoleh *resource* yang dibutuhkan maka terjadi *deadlock*. *Resource* dalam Petri net biasanya dinyatakan dengan *token* dan pihak yang bersaing memperoleh *token* adalah transisi. Dengan menggunakan keterangan sebelumnya maka dapat disimpulkan bahwa *deadlock* terjadi ketika transisi tertentu atau himpunan transisi pada Petri net tidak dapat *difire*.

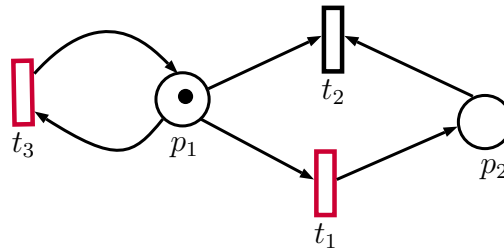
Transisi yang tidak berhubungan dengan *deadlock* disebut *live*. Perhatikan bahwa transisi yang *live* tidak harus *enable*. Istilah *liveness* dapat diartikan dengan transisi yang mungkin *enable*. Idealnya setiap transisi pada Petri net dapat *difire* setelah beberapa *pemfirean*. Hal ini menjamin *deadlock* tidak terjadi.

Definisi 4.4.1 ([29]) Petri net dengan keadaan awal $\mathbf{x}_0$ disebut *live* jika terdapat beberapa *sample path* sedemikian hingga selalu ada transisi yang dapat difire untuk setiap keadaan yang dapat dicapai dari $\mathbf{x}_0$.

Pengujian *liveness* dengan menggunakan Definisi 4.4.1 merupakan pekerjaan yang tidak mudah. Pada beberapa sistem, pengujian *liveness* tidak bisa dilakukan. Cara mengatasi permasalahan ini yaitu menggunakan klasifikasi *liveness*.

Diberikan keadaan awal $\mathbf{x}_0$, berikut merupakan klasifikasi dari *liveness* suatu transisi di Petri net [29].

- *Dead* atau *L0-live*, jika transisi tidak pernah dapat difire dengan keadaan awal ini.
- *L1-live*, jika terdapat beberapa urutan pemfirean dari $\mathbf{x}_0$ sedemikian hingga transisi ini dapat difire paling tidak sekali.
- *L2-live*, jika transisi dapat difire paling tidak sebanyak k kali dengan k adalah bilangan integer positif.
- *L3-live*, jika terdapat takhingga urutan pemfirean dengan pemfirean transisi ini sebanyak takhingga.
- *Live* atau *L4-live*, jika transisi ini *L1-live* untuk setiap kemungkinan keadaan yang dapat dicapai dari $\mathbf{x}_0$.

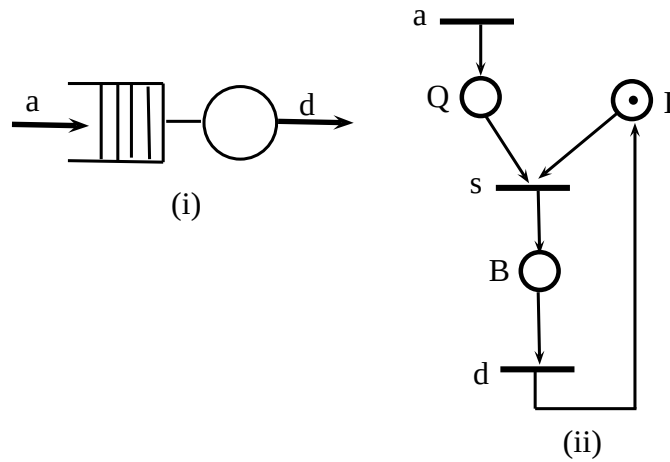


Gambar 4.11: Transisi dengan *Liveness* Berbeda-beda

Pemahaman klasifikasi *liveness* akan menjadi lebih mudah jika diberikan contoh. Berikut merupakan contoh yang menjelaskan tingkat *liveness* setiap transisi pada Petri net di Gambar 4.11. Transisi pada Petri net tersebut akan diklasifikasikan.

Contoh 4.4.1 Transisi t_2 termasuk *dead*. Hal ini disebabkan transisi t_2 dapat difire jika terdapat *token* pada p_1 dan p_2 yang tidak pernah terjadi (jika t_1 difire maka p_2 menerima sebuah *token* sedangkan p_1 kehilangan *token*nya). Transisi t_1 adalah *L1-live* karena transisi tersebut hanya dapat difire sekali. Jika transisi t_1 difire maka semua transisi menjadi *dead* pada keadaan tersebut. Transisi t_3 merupakan *L3-live* karena transisi tersebut dapat difire sebanyak takhingga tetapi transisi itu tidak *L4-live* karena dapat menjadi *dead* pada keadaan setelah pemfirean t_1 .

□



Gambar 4.12: Antrian satu server dan Petri netnya

4.5 Model sistem antrian

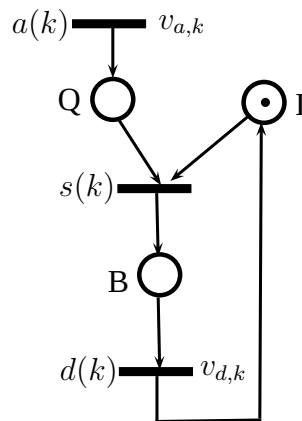
Berikut ini diberikan pengertian dari Petri net yang digunakan dalam pembahasan berikutnya terutama bila Petri net ini dikaitkan dengan waktu. Suatu grap $\mathcal{G} = (E, V)$ dikatakan bipartisi bila himpunan titik-titik V dapat dipartisi menjadi dua himpunan bagian yang saling asing V_1 dan V_2 sedemikian hingga setiap garis di $\mathcal{G}$ menghubungkan suatu titik dari V_1 ke satu titik V_2 begitu juga sebaliknya. Petri net adalah grap bipartisi. Himpunan V dipartisi menjadi dua himpunan bagian P dan T yang masing-masing menyatakan place dan transisi. Suatu place $p \in P$ dan suatu transisi $t \in P$ dapat diinterpretasikan sebagai suatu kondisi dan suatu event dari suatu diskripsi suatu model yang dibahas. Masing-masing kondisi $p \in P$ yang 'terpenuhi' diberi tanda titik (token). Suatu token dalam suatu place mempunyai arti bahwa sepanjang place ini penting bagi suatu transisi yang dihubungkan-nya menyebabkan transisi ini menjadi aktif (event bisa berlangsung). Jadi dengan dua, tiga, empat, ... token, event ini bisa berlangsung dua, tiga, empat, ... kali. Untuk memperjelas pengertian ini diberikan suatu contoh Petri net dari suatu model sistem antrian sederhana dengan satu server. Contoh ini akan dibahas lagi terutama bila Petri net dari sistem antrian sederhana dengan satu server ini dikaitkan dengan waktu (Petri netnya dinamakan Petri net dengan waktu). Selanjutnya diturunkan bentuk modelnya dari Petri net dengan waktu ini dalam aljabar maxplus. Gambar dari sistem antrian sederhana dengan satu server beserta Petri netnya (Petri net tanpa waktu) diberikan sebagai berikut.

Dalam Gambar 4.12 bagian (i), a menyatakan kedatangan pelanggan menuju sistem antrian sedangkan d menyatakan pelanggan telah dilayani oleh server dan meninggalkannya. Sedangkan Gambar 4.12 bagian (ii) merupakan penyajian Petri net dari sistem antrian pada bagian (i). Petri net ini terdiri dari himpunan place $P = \{Q, B, I\}$ dan himpunan transisi $T = \{a, s, d\}$. Masing-masing place Q, B dan I menyatakan kondisi antri, sibuk

dan idel. Sedangkan transisi a , s dan d masing-masing menyatakan event kedatangan, mulai dilayani dan meninggalkan server. Tanda satu token dalam place I menyatakan bahwa sistem antrian dalam keadaan idel. Dalam kondisi ini transisi s tidak bisa aktif sebab walaupun sistem dalam keadaan idel tetapi kondisi di Q kosong/tidak ada yang antri (tidak ada token). Dalam keadaan ini, yang paling mungkin terjadi adalah transisi a yaitu pelanggan datang. Bila hal ini terjadi, maka place Q berisi satu token. Keadaan yang terakhir ini menyatakan bahwa sistem dalam kondisi idel dan sudah ada satu pelanggan yang antri. Oleh karena itu, transisi s bisa terjadi, yaitu server sudah bisa memulai untuk melayani satu pelanggan. Dalam keadaan ini, masing-masing satu token di B dan Q berkurang dan satu token berada di place B (sistem antrian dalam keadaan sibuk).

4.5.1 Model Aljabar maxplus dari Petri net dengan waktu.

Sebagaimana telah disebutkan dalam bagian sebelumnya, model maxplus aljabar dari petri net dengan waktu untuk sistem antrian sederhana akan dibahas dalam bagian ini. Petri net dalam Gambar 4.12 bagian (ii) bila dikaitkan dengan waktu akan berubah. Gambar Petri net berikut merupakan gambar dari Petri net dari sistem antrian sederhana yang dikaitkan dengan waktu.



Gambar 4.13: Gambar Petri net dengan waktu

Dalam Gambar 4.13 ini, bila $a(k)$ menyatakan waktu kedatangan pelanggan saat yang ke- k , $v_{a,k}$ menyatakan lamanya kedatangan pelanggan saat yang ke- k , $s(k)$ menyatakan waktu pelayanan dimulai saat yang ke- k , $d(k)$ menyatakan waktu pelanggan meninggalkan pelayanan saat yang ke- k dan $v_{d,k}$ menyatakan lamanya pelanggan meninggalkan pelayanan

saat yang ke- k , maka didapat

$$\begin{aligned} a(k) &= v_{a,k} + a(k-1), & k &= 1, 2, \dots \\ s(k) &= \max\{a(k), d(k-1)\}, & k &= 1, 2, \dots \\ d(k) &= v_{d,k} + s(k), & k &= 1, 2, \dots \\ &= v_{d,k} + \max\{a(k), d(k-1)\} \\ &= \max\{(v_{d,k} + v_{a,k}) + a(k-1), v_{d,k} + d(k-1)\} \end{aligned}$$

Sehingga dengan menggunakan notasi aljabar maxplus didapat persamaan

$$\begin{bmatrix} a(k) \\ d(k) \end{bmatrix} = \begin{bmatrix} v_{a,k} & c \\ v_{d,k} \otimes v_{a,k} & v_{d,k} \end{bmatrix} \otimes \begin{bmatrix} a(k-1) \\ d(k-1) \end{bmatrix}, \quad (4.19)$$

dimana c dipilih supaya

$$(v_{a,k} \otimes a(k-1)) \oplus (c \otimes d(k-1)) = v_{a,k} \otimes a(k-1), \text{ untuk } k = 1, 2, 3, \dots$$

dan keadaan awal $a(0) = d(0) = 0$.

Persamaan (4.19) adalah bentuk aljabar maxplus dari sistem model antrian dengan satu server dan terlihat bahwa evolusi dari keadaan $a(k)$ dan $d(k)$ bergantung pada nilai-nilai dari $v_{a,k}$ dan $v_{d,k}$ untuk $k = 1, 2, 3, \dots$. Dalam kenyataannya, $v_{a,k}$ dan $v_{d,k}$ adalah barisan bilangan real positif. Sebelum mengakhiri bagian ini, diberikan suatu contoh.

Contoh 4.5.1 Bila diberikan sample path dari $v_{a,k}$ dan $v_{d,k}$ sebagai berikut

$$v_{a,k} = \{0.5, 0.5, 1, 0.5, 2, \dots\}, \quad v_{d,k} = \{1, 1.5, 0.5, 0.5, 1, \dots\}$$

didapat

$$\begin{aligned} \begin{bmatrix} a(1) \\ d(1) \end{bmatrix} &= \begin{bmatrix} 0.5 & \varepsilon \\ 1.5 & 1 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 1.5 \end{bmatrix} \\ \begin{bmatrix} a(2) \\ d(2) \end{bmatrix} &= \begin{bmatrix} 0.5 & \varepsilon \\ 2 & 1.5 \end{bmatrix} \otimes \begin{bmatrix} 0.5 \\ 1.5 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \end{bmatrix} \\ \begin{bmatrix} a(3) \\ d(3) \end{bmatrix} &= \begin{bmatrix} 1 & \varepsilon \\ 1.5 & 0.5 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 3.5 \end{bmatrix} \\ \begin{bmatrix} a(4) \\ d(4) \end{bmatrix} &= \begin{bmatrix} 0.5 & \varepsilon \\ 1 & 0.5 \end{bmatrix} \otimes \begin{bmatrix} 2 \\ 3.5 \end{bmatrix} = \begin{bmatrix} 2.5 \\ 4 \end{bmatrix} \\ \begin{bmatrix} a(5) \\ d(5) \end{bmatrix} &= \begin{bmatrix} 2 & \varepsilon \\ 3 & 1 \end{bmatrix} \otimes \begin{bmatrix} 2.5 \\ 4 \end{bmatrix} = \begin{bmatrix} 4.5 \\ 5.5 \end{bmatrix} \end{aligned}$$

Dalam hal ini $c = \varepsilon$.

□

Dari beberapa pembahasan model antrian ini diberikan beberapa catatan. Model sistem antrian yang dibahas sangat sederhana yaitu antrian hanya dengan satu server, hal ini untuk memberikan suatu ide awal menurunkan model antrian dengan menggunakan aljabar maxplus. Kedepannya kajian bisa diperluas untuk antrian dengan lebih dari satu server, bahkan bila memungkinkan dikaji model antrian yang lebih realistis. Misalnya kapasitas antrian berhingga dan pada saat kondisi tertentu server down sehingga memerlukan perbaikan supaya bisa melayani lagi pelanggannya. Untuk kasus ini Petri net dari antrian bentuknya akan berubah dan bagaimana model aljabar maxplusnya bila memungkinkan. Dalam Contoh 4.5.1 nilai c dipilih ε (konstan), akibat pemilihan ini matriks model menjadi tidak strongly connected. Bila dikehendaki matriks modelnya menjadi strongly connected, maka pemilihan nilai c adalah bervariasi. Nilai c yang bervariasi, nilai-nilai dari $v_{a,k}$ dan $v_{d,k}$ yang merupakan barisan dari bilangan real positif menyebabkan matriks model yang diperoleh erat kaitannya dengan matriks interval dalam aljabar maxplus.

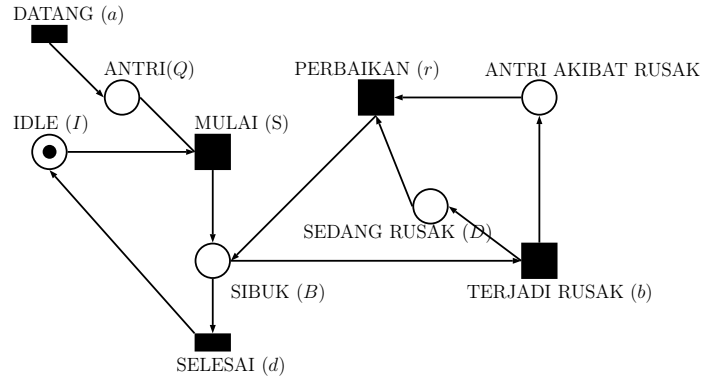
Model antrian selain disajikan dalam bentuk Petri net, bisa disajikan dalam bentuk apa yang dinamakan automata. Kajian yang telah dibahas menginspirasi suatu apa yang dinamakan Aljabar Maxplus Automata lewat pendekatan model antrian. Beberapa pembahasan mengenai Aljabar Maxplus Automata didahului dengan suatu model yang dinamakan 'heap'.

4.5.2 Bentuk Petrinet dan Model Dari Sistem Antrian Dengan Adanya Kemungkinan *Server Down* dan Kapasitas Antrian Takterbatas

Dibandingkan dengan bentuk petrinet dari antrian tanpa melihat adanya kemungkinan terjadi server down, bentuk petrinet dari antrian dengan melihat adanya kemungkinan terjadi server down jauh lebih kompleks. Dalam petrinet ini ditambahkan dua event setelah server mulai melayani pelanggan yaitu *event* terjadinya kerusakan (*server down*) dan proses perbaikan *server down*. Selain itu juga terdapat dua *place* tambahan yaitu *place* sedang terjadinya kerusakan dan *place* tempat antri orang yang gagal dilayani hingga selesai akibat *server down*. Untuk lebih lengkapnya bentuk petrinet dari kasus ini dapat dilihat pada Gambar 4.14.

Selanjutnya akan dilakukan pemodelan dari sistem antrian ini. Pemodelan dilakukan hanya dalam kasus satu server dan terjadinya *server down* maksimal terjadi satu kali dalam melayani satu pelanggan. Sebelum dilakukan pemodelan terlebih dahulu akan dilakukan pendefinisian variabel. Dalam hal ini dilakukan pendefinisian variabel sebagai berikut:

- $a(k)$: kedatangan pelanggan ke- k
- $s(k)$: waktu pelayanan ke- k
- $d(k)$: waktu pelayanan ke- k selesai



Gambar 4.14: Petrinet dari sistem antrian dengan adanya kemungkinan *server down* dan kapasitas antrian takterbatas

- $b(k)$: waktu pelayanan ke- k rusak
- $r(k)$: waktu pelayanan ke- k diperbaiki
- $v_{a,k}$: lamanya kedatangan pelanggan saat yang ke- k
- $v_{d,k}$: lamanya pelanggan dilayani hingga selesai saat yang ke- k (apabila terjadi *server down* nilainya ε)
- $v_{b,k}$: lamanya pelanggan dilayani hingga terjadi *server down* pada saat yang ke- k (apabila tidak terjadi *server down* nilainya ε)
- $v_{r,k}$: lamanya perbaikan server yang mengalami *server down* pada saat yang ke- k (apabila tidak terjadi *server down* nilainya ε)
- $v_{s,k}$: lamanya pelanggan dilayani kembali hingga selesai saat yang ke- k (apabila tidak terjadi *server down* nilainya ε)

dengan pendefinisian variabel ini, didapatkan model sistem dengan antrian satu server

sebagai berikut:

$$\begin{aligned}
a(k) &= v_{a,k} \otimes a(k-1) \\
s(k) &= a(k) \oplus d(k-1) \\
b(k) &= s(k) \otimes v_{b,k} \\
r(k) &= b(k) \otimes v_{r,k} \\
&= s(k) \otimes v_{b,k} \otimes v_{r,k} \\
&= (a(k) \oplus d(k-1)) \otimes v_{b,k} \otimes v_{r,k} \\
&= (v_{b,k} \otimes v_{r,k} \otimes a(k)) \oplus (v_{b,k} \otimes v_{r,k} \otimes d(k-1)) \\
&= (v_{b,k} \otimes v_{r,k} \otimes v_{a,k} \otimes a(k-1)) \oplus (v_{b,k} \otimes v_{r,k} \otimes d(k-1)) \\
d(k) &= v_{d,k} \otimes s(k) \oplus v_{s,k} \otimes r(k) \\
&= (a(k) \oplus d(k-1)) \otimes v_{d,k} \oplus v_{s,k} \otimes r(k) \\
&= (v_{d,k} \otimes a(k)) \oplus (v_{d,k} \otimes d(k-1)) \oplus v_{s,k} \otimes ((v_{b,k} \otimes \\
&\quad v_{r,k} \otimes v_{a,k} \otimes a(k-1)) \oplus (v_{b,k} \otimes v_{r,k} \otimes d(k-1))) \\
&= (v_{d,k} \otimes v_{a,k} \otimes a(k-1)) \oplus (v_{d,k} \otimes d(k-1)) \oplus (v_{s,k} \otimes v_{b,k} \otimes \\
&\quad v_{r,k} \otimes v_{a,k} \otimes a(k-1)) \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k} \otimes d(k-1)) \\
&= ((v_{d,k} \otimes v_{a,k}) \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k} \otimes v_{a,k})) \otimes a(k-1) \oplus \\
&\quad (v_{d,k} \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k})) \otimes d(k-1)
\end{aligned}$$

Dalam bentuk matriks persamaan diatas dapat dituliskan sebagai berikut:

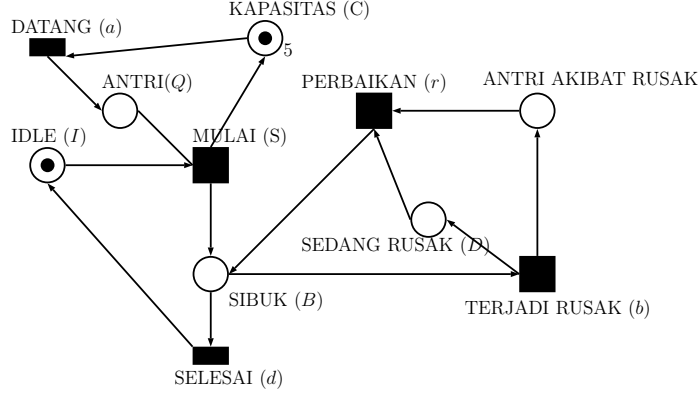
$$\begin{bmatrix} a(k) \\ d(k) \end{bmatrix} = \begin{bmatrix} v_{a,k} & c \\ (v_{d,k} \otimes v_{a,k}) \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k} \otimes v_{a,k}) & v_{d,k} \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k}) \end{bmatrix} \otimes \begin{bmatrix} a(k-1) \\ d(k-1) \end{bmatrix}$$

dengan c dipilih supaya $v_{a,k} \otimes a(k-1) \oplus c \otimes d(k-1) = v_{a,k} \otimes a(k-1)$ dan keadaan awal $a(0) = d(0) = 0$. Dengan demikian nilai trivial dari nilai c adalah ε . Selanjutnya untuk kasus yang seperti ini digunakan nilai trivial.

Simulasi dari contoh ini dapat dilakukan dengan menggunakan toolbox aljabar maxplus yang ada pada Scilab dengan menggunakan fungsi **petriqueue**.

4.5.3 Bentuk Petrinet dan Model dari Sistem Antrian dengan Adanya Kemungkinan *Server Down* dan Kapasitas Antrian Terbatas

Dalam bagian ini sistem antrian dengan kapasitasnya dibatasi sebanyak M dengan $M = 1, 2, \dots$. Petrinet dari sistem ini ditambahkan satu *place* yaitu *place* yang memiliki token sebanyak M . Untuk lebih lengkapnya bentuk petrinet dari kasus ini dapat dilihat pada Gambar 4.15.



Gambar 4.15: Petrinet dari sistem antrian dengan adanya kemungkinan *server down* dan kapasitas antrian terbatas

Untuk model dari sistem ini dibagi menjadi dua kasus yaitu untuk $k = 1, 2, \dots, M$ dan $k = M + 1, M + 2, \dots$, yaitu untuk $k = 1, 2, \dots, M$ berlaku

$$\begin{aligned}
 a(k) &= v_{a,k} \otimes a(k-1) \\
 s(k) &= a(k) \oplus d(k-1) \\
 b(k) &= s(k) \otimes v_{b,k} \\
 r(k) &= b(k) \otimes v_{r,k} \\
 &= s(k) \otimes v_{b,k} \otimes v_{r,k} \\
 &= (a(k) \oplus d(k-1)) \otimes v_{b,k} \otimes v_{r,k} \\
 &= (v_{b,k} \otimes v_{r,k} \otimes a(k)) \oplus (v_{b,k} \otimes v_{r,k} \otimes d(k-1)) \\
 &= (v_{b,k} \otimes v_{r,k} \otimes v_{a,k} \otimes a(k-1)) \oplus (v_{b,k} \otimes v_{r,k} \otimes d(k-1))
 \end{aligned}$$

$$\begin{aligned}
 d(k) &= v_{d,k} \otimes s(k) \oplus v_{s,k} \otimes r(k) \\
 &= (a(k) \oplus d(k-1)) \otimes v_{d,k} \oplus v_{s,k} \otimes r(k) \\
 &= (v_{d,k} \otimes a(k)) \oplus (v_{d,k} \otimes d(k-1)) \oplus v_{s,k} \otimes ((v_{b,k} \otimes \\
 &\quad v_{r,k} \otimes v_{a,k} \otimes a(k-1)) \oplus (v_{b,k} \otimes v_{r,k} \otimes d(k-1))) \\
 &= (v_{d,k} \otimes v_{a,k} \otimes a(k-1)) \oplus (v_{d,k} \otimes d(k-1)) \oplus (v_{s,k} \otimes v_{b,k} \otimes \\
 &\quad v_{r,k} \otimes v_{a,k} \otimes a(k-1)) \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k} \otimes d(k-1)) \\
 &= ((v_{d,k} \otimes v_{a,k}) \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k} \otimes v_{a,k})) \otimes a(k-1) \oplus \\
 &\quad (v_{d,k} \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k})) \otimes d(k-1)
 \end{aligned}$$

Dalam bentuk matriks persamaan diatas dapat dituliskan sebagai berikut:

$$\begin{bmatrix} a(k) \\ s(k) \\ d(k) \end{bmatrix} = \begin{bmatrix} v_{a,k} & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 \\ (v_{d,k} \otimes v_{a,k}) \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k} \otimes v_{a,k}) & \varepsilon & v_{d,k} \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k}) \end{bmatrix} \otimes \begin{bmatrix} a(k-1) \\ s(k-1) \\ d(k-1) \end{bmatrix}$$

Selanjutnya untuk $k = M + 1, M + 2, \dots$ berlaku:

$$\begin{aligned} a(k) &= v_{a,k} \otimes a(k-1) \oplus s(k-M) \\ s(k) &= a(k) \oplus d(k-1) \\ &= (v_{a,k} \otimes a(k-1)) \oplus s(k-M) \oplus d(k-1) \\ b(k) &= s(k) \otimes v_{b,k} \\ r(k) &= b(k) \otimes v_{r,k} \\ &= s(k) \otimes v_{b,k} \otimes v_{r,k} \\ &= (a(k) \oplus d(k-1)) \otimes v_{b,k} \otimes v_{r,k} \\ &= (v_{b,k} \otimes v_{r,k} \otimes a(k)) \oplus (v_{b,k} \otimes v_{r,k} \otimes d(k-1)) \\ &= (v_{b,k} \otimes v_{r,k} \otimes v_{a,k} \otimes a(k-1)) \oplus (v_{b,k} \otimes v_{r,k} \otimes s(k-M)) \oplus (v_{b,k} \otimes v_{r,k} \otimes d(k-1)) \\ d(k) &= v_{d,k} \otimes s(k) \oplus v_{s,k} \otimes r(k) \\ &= (a(k) \oplus d(k-1)) \otimes v_{d,k} \oplus v_{s,k} \otimes r(k) \\ &= (v_{d,k} \otimes a(k)) \oplus (v_{d,k} \otimes d(k-1)) \oplus v_{s,k} \otimes ((v_{b,k} \otimes v_{r,k} \otimes v_{a,k} \otimes a(k-1)) \oplus (v_{b,k} \otimes v_{r,k} \otimes s(k-M)) \oplus (v_{b,k} \otimes v_{r,k} \otimes d(k-1))) \\ &= (v_{d,k} \otimes v_{a,k} \otimes a(k-1)) \oplus (v_{d,k} \otimes s(k-M)) \oplus (v_{d,k} \otimes d(k-1)) \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k} \otimes v_{a,k} \otimes a(k-1)) \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k} \otimes s(k-M)) \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k} \otimes d(k-1)) \\ &= ((v_{d,k} \otimes v_{a,k}) \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k} \otimes v_{a,k})) \otimes a(k-1) \oplus (v_{d,k} \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k})) \otimes d(k-1) \oplus (v_{d,k} \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k})) \otimes s(k-M) \end{aligned}$$

Dalam bentuk matriks persamaan diatas dapat dituliskan sebagai berikut:

$$\mathbf{x}(k) = A_1 \otimes \mathbf{x}(k-1) \oplus A_2 \otimes \mathbf{x}(k-M),$$

dengan $\mathbf{x}$, A_1 dan A_2 adalah sebagai berikut

$$\mathbf{x} = \begin{bmatrix} a \\ s \\ r \end{bmatrix},$$

$$A_1 = \begin{bmatrix} & v_{a,k} & \varepsilon & \varepsilon \\ & v_{a,k} & \varepsilon & 0 \\ (v_{d,k} \otimes v_{a,k}) \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k} \otimes v_{a,k}) & \varepsilon & v_{d,k} \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k}) \end{bmatrix},$$

dan

$$A_2 = \begin{bmatrix} \varepsilon & 0 & \varepsilon \\ \varepsilon & 0 & \varepsilon \\ \varepsilon & v_{d,k} \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k}) & \varepsilon \end{bmatrix}.$$

Secara umum model diatas dapat ditulis dalam bentuk

$$\mathbf{x}(k) = \begin{cases} A \otimes \mathbf{x}(k-1) & , \text{ untuk } k = 1, 2, \dots, M \\ A_1 \otimes \mathbf{x}(k-1) \oplus A_2 \otimes \mathbf{x}(k-M) & , \text{ untuk } k = M+1, M+2, \dots \end{cases}$$

dengan x , A , A_1 dan A_2 diberikan oleh

$$\mathbf{x} = \begin{bmatrix} a \\ s \\ d \end{bmatrix},$$

$$A = \begin{bmatrix} & v_{a,k} & \varepsilon & \varepsilon \\ & \varepsilon & \varepsilon & 0 \\ (v_{d,k} \otimes v_{a,k}) \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k} \otimes v_{a,k}) & \varepsilon & v_{d,k} \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k}) \end{bmatrix},$$

$$A_1 = \begin{bmatrix} & v_{a,k} & \varepsilon & \varepsilon \\ & v_{a,k} & \varepsilon & 0 \\ (v_{d,k} \otimes v_{a,k}) \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k} \otimes v_{a,k}) & \varepsilon & v_{d,k} \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k}) \end{bmatrix}$$

dan

$$A_2 = \begin{bmatrix} \varepsilon & 0 & \varepsilon \\ \varepsilon & 0 & \varepsilon \\ \varepsilon & v_{d,k} \oplus (v_{s,k} \otimes v_{b,k} \otimes v_{r,k}) & \varepsilon \end{bmatrix},$$

Simulasi dari contoh ini dapat dilakukan dengan menggunakan toolbox aljabar max-plus yang ada pada Scilab dengan menggunakan fungsi **petriqueue**.

4.6 Model Rantai Pasok

Rantai pasok atau *supply chain* merupakan suatu sistem yang berkenaan dengan proses produksi, pengiriman, penyimpanan, distribusi dan penjualan produk tersebut. Hal ini bisa berarti bahwa rantai pasok dapat diartikan sebagai proses pengiriman barang atau

jasa dari *supplier* ke *customer*. Tujuan dari rantai pasok adalah untuk memastikan suatu produk berada pada tempat dan waktu yang tepat untuk memenuhi permintaan konsumen tanpa menciptakan stok yang berlebihan atau kekurangan.

Rantai Pasok meliputi 3 bagian:

1. **Upstream Supply Chain.**

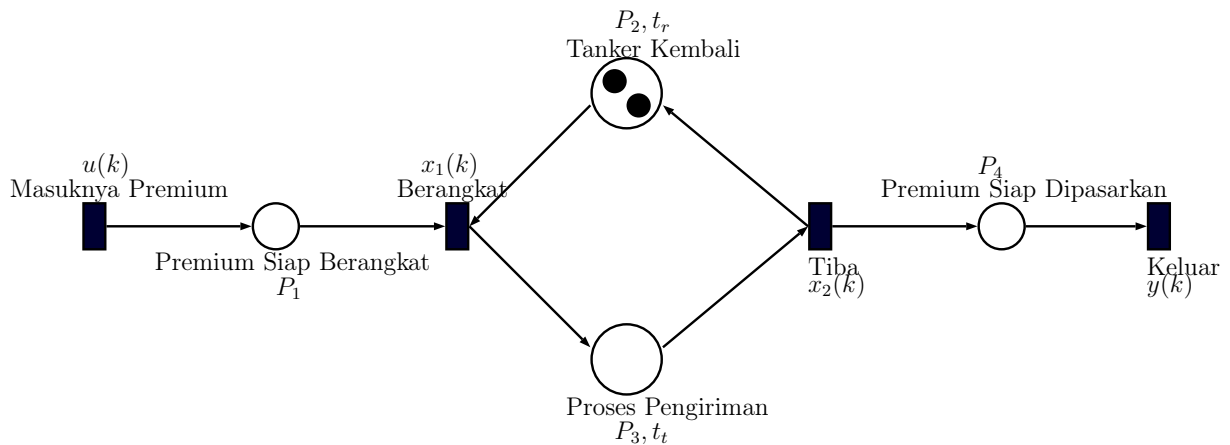
Bagian ini mencakup supplier first-tier dari organisasi (dapat berupa perusahaan manufaktur atau assembling) dan supplier-nya, yang didalamnya telah terbina suatu hubungan/relasi.

2. **Internal Supply Chain.**

Bagian ini mencakup semua proses yang digunakan oleh organisasi dalam mengubah input yang dikirim oleh supplier menjadi output, mulai dari waktu material tersebut masuk pada perusahaan sampai pada produk tersebut didistribusikan di luar perusahaan tersebut.

3. **Downstream Supply Chain.**

Bagian ini mencakup semua proses yang terlibat dalam pengiriman produk pada *customer* akhir.



Gambar 4.16: Petri Net Sistem Transportasi pada Rantai Pasok dari TBBM Tuban ke TBBM Manggis

Pada bagian ini melalui suatu contoh kasus dibahas proses rantai pasok yang terdapat pada industri Pertamina *Supply and Distribution Region III* yaitu pengiriman bahan bakar premium dari Terminal Bahan Bakar Minyak (TBBM) Tuban ke Terminal Bahan Bakar Minyak (TBBM) Manggis, Bali. (Bahasan lengkap dapat dilihat di [21, 23]). Dalam pembahasan ([23]) digunakan data pengiriman solar sedangkan dalam ([21]) digunakan data pengiriman premium. Dalam pembahasan ([23]) dipertimbangkan waktu *loading* dan *unloading* produk serta prioritas kapal tanker yang digunakan. Petrinet pada Gambar 4.16 merepresantasikan sebuah sistem transportasi dimana terdiri dari 2 kendaraan

yang masing-masing memiliki kapasitas pengangkutan tertentu. Kendaraan ini membawa produk dari TBBM Tuban P_1 menuju TBBM Manggis P_4 . Gambar 4.16 menunjukkan waktu masuknya produk BBM jenis premium di TBBM Tuban $u(k)$ yang selanjutnya siap untuk diberangkatkan menuju TBBM Manggis yang mana $u(k)$ bergantung pada waktu permintaan. Hal ini berdasarkan data dari *Pertamina Supply and Distribution Region III* mulai tanggal 20 September 2013 sampai 4 Desember 2013.

Premium ini berada pada kondisi siap untuk diberangkatkan (P_1) dan akan berangkat (x_1) jika terdapat kapal yang sudah siap untuk suatu keberangkatan yang baru (P_2). Selanjutnya terdapat proses pengiriman premium oleh kapal tanker (p_3) dari TBBM Tuban (p_1) menuju TBBM Manggis (p_4) melalui jalur laut. Kapal tanker yang sudah tiba di TBBM Manggis (x_2) dengan membawa premium selanjutnya akan kembali lagi ke TBBM Tuban. Sedangkan premium yang berada di TBBM Manggis sudah siap untuk disalurkan kembali ke wilayah Bali dan Nusa Tenggara ($y(k)$).

Berdasarkan Gambar 4.16 untuk banyaknya tanker adalah n diperoleh suatu sistem persamaan dalam aljabar max-plus sebagai berikut

$$\begin{aligned}x_1(k) &= t_r \otimes x_2(k - n) \oplus u(k) \\x_2(k) &= t_t \otimes x_1(k) \\y(k) &= x_2(k),\end{aligned}$$

dengan t_t Waktu yang digunakan oleh kendaraan (kapal tanker) untuk membawa produk BBM jenis premium sampai ke TBBM Manggis, t_r Waktu yang digunakan oleh kendaraan (kapal tanker) untuk kembali ke TBBM Tuban, $u(k)$ waktu masuknya premium ke- k di TBBM Tuban untuk siap diberangkatkan ke TBBM Manggis, $x_1(k)$ waktu keberangkatan premium ke- k dari TBBM Tuban, $x_2(k)$ waktu kedatangan premium ke- k di TBBM Manggis dan $y(k)$ waktu premium ke- k dipasarkan kembali. Kemudian dalam bentuk persamaan matriks, model diatas dapat dituliskan sebagai berikut

$$\mathbf{x}(k) = A_0 \otimes \mathbf{x}(k) \oplus A_1 \otimes \mathbf{x}(k - n) \oplus B_0 \otimes u(k) \quad (4.20)$$

$$y(k) = C \otimes \mathbf{x}(k). \quad (4.21)$$

dengan

$$\mathbf{x}(k) = \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix}, \quad A_0 = \begin{bmatrix} \varepsilon & \varepsilon \\ t_t & \varepsilon \end{bmatrix}, \quad A_1 = \begin{bmatrix} \varepsilon & t_r \\ \varepsilon & \varepsilon \end{bmatrix}, \quad B_0 = \begin{bmatrix} 0 \\ \varepsilon \end{bmatrix}, \quad \text{dan } C = [\varepsilon \ 0].$$

Selanjutnya dengan mensubstitusi $\mathbf{x}(k)$ pada Persamaan 4.20 didapatkan

$$\begin{aligned}\mathbf{x}(k) &= A_0 \otimes (A_0 \otimes \mathbf{x}(k) \oplus A_1 \otimes \mathbf{x}(k - n) \oplus B_0 \otimes u(k)) \oplus A_1 \otimes \mathbf{x}(k - n) \oplus B_0 \otimes u(k) \\ &= A_0^2 \otimes \mathbf{x}(k) \oplus (E \oplus A_0) \otimes A_1 \otimes \mathbf{x}(k - n) \oplus (E \oplus A_0) \otimes B_0 \otimes u(k).\end{aligned}$$

Kemudian lakukan substitusi ini sebanyak β kali sehingga didapatkan

$$\begin{aligned}\mathbf{x}(k) &= A_0^\beta \otimes \mathbf{x}(k) \oplus (E \oplus A_0 \oplus \dots \oplus A_0^{\beta-1}) \otimes A_1 \otimes \mathbf{x}(k - n) \oplus (E \oplus A_0 \oplus \dots \oplus A_0^{\beta-1}) \\ &\quad \otimes B_0 \otimes u(k).\end{aligned}$$

Lalu untuk β mendekati takhingga didapatkan (lihat definisi A_0^* pada halaman 46):

$$\mathbf{x}(k) = A \otimes \mathbf{x}(k - n) \oplus B \otimes u(k) \quad (4.22)$$

dengan $A = A_0^* \otimes A_1$ dan $B = A_0^* \otimes B_0$.

Agar dapat menghasilkan bentuk persamaan $Y = H \otimes U$, maka dengan mensubstitusikan $\mathbf{x}(k)$ pada Persamaan 4.21 maka didapatkan persamaan baru sebagai berikut

$$\begin{aligned} y(k) &= C \otimes (A \otimes \mathbf{x}(k - n) \oplus B \otimes u(k)) \\ &= C \otimes A \otimes \mathbf{x}(k - n) \oplus C \otimes B \otimes u(k) \end{aligned}$$

Kemudian lakukan substitusi ini sebanyak α kali dengan $\alpha = [k/n]$ yang merupakan *Euclidean division* oleh k dengan n sehingga didapatkan

$$y(k) = C \otimes A^\alpha \otimes \mathbf{x}(k - \alpha \cdot n) \oplus \left(\bigoplus_{i=0}^{\alpha-1} C \otimes A^{\otimes i} \otimes B \otimes u(k - i \cdot n) \right).$$

karena $k - \alpha \cdot n \leq n$ maka didapat

$$y(k) = \bigoplus_{i=0}^{\alpha} C \otimes A^{\otimes i} \otimes B \otimes u(k - i \cdot n).$$

Sehingga untuk banyaknya keberangkatan kapal/banyaknya pesanan n_d dan $l = 1, 2, \dots, n$ maka didapat suatu persamaan

$$Y = H \otimes U \quad (4.23)$$

dengan

$$Y = \begin{bmatrix} y(l) \\ y(l+n) \\ y(l+2n) \\ \vdots \\ y(l+\gamma n) \end{bmatrix}, \quad H = \begin{bmatrix} CB & \varepsilon & \cdots & \varepsilon \\ CAB & CB & \cdots & \varepsilon \\ \vdots & \vdots & \ddots & \vdots \\ CA^\gamma B & CA^{\gamma-1} B & \cdots & CB \end{bmatrix}, \quad U = \begin{bmatrix} u(l) \\ u(l+n) \\ u(l+2n) \\ \vdots \\ u(l+\gamma n) \end{bmatrix}$$

dengan γ adalah suatu bilangan bulat sedemikian hingga $l + \gamma n$ merupakan bilangan bulat terbesar yang kurang atau sama dengan n_d . Kemudian setelah mendapatkan bentuk diatas, maka permasalahan ini dapat diselesaikan dengan menggunakan teori pada Subbagian 2.4.2 penyelesaian $A \otimes \mathbf{x} = \mathbf{b}$.

Selanjutnya diberikan suatu contoh dari permasalahan penjadwalan rantai pasok, yaitu yang berkaitan dengan pengiriman bahan baku minyak dari suatu TBBM ke TBBM yang lain dengan menggunakan transportasi kapal yang tersedia sebanyak dua. Tanggal permintaan dari konsumen diketahui. Dari tanggal permintaan ini sebelum dibentuk model aljabar maxplusnya diubah kedalam bentuk bilangan bulat. Masalahnya adalah kapan tanker pemuat BBM harus berangkat supaya kedatangan dari tanker sesuai dengan tanggal permintaan konsumen,

Contoh 5 Diketahui waktu yang diperlukan tanker mulai dari persiapan BBM hingga BBM sampai ditujuan adalah 5 hari dan waktu yang diperlukan untuk kembali ke Tuban adalah 2 hari dan banyaknya tanker yang tersedia adalah 2 buah. Kemudian diberikan waktu permintaan premium sebagaimana diberikan oleh Tabel 4.1 .

Tabel 4.1: Tabel Tanggal Permintaan Konsumen

No	Tanggal
1	20 September 2013
2	28 September 2013
3	6 Oktober 2013
4	14 Oktober 2013
5	17 Oktober 2013
6	23 Oktober 2013
7	3 November 2013
8	10 November 2013
9	18 November 2013
10	26 November 2013
11	4 Desember 2013

Tentukan jadwal keberangkatan kapal dari tuban ($u(k)$).

Penyelesaian

Tabel permintaan diatas dapat dituliskan kebentuk bilangan bulat yaitu

No	Tanggal
1	20
2	28
3	36
4	44
5	47
6	53
7	64
8	71
9	79
10	87
11	95

kemudian untuk $l = 1$ didapat matriks

$$H = \begin{bmatrix} 5 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 12 & 5 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 19 & 12 & 5 & \varepsilon & \varepsilon & \varepsilon \\ 26 & 19 & 12 & 5 & \varepsilon & \varepsilon \\ 33 & 26 & 19 & 12 & 5 & \varepsilon \\ 40 & 33 & 26 & 19 & 12 & 5 \end{bmatrix}, Y = \begin{bmatrix} 20 \\ 36 \\ 47 \\ 64 \\ 79 \\ 95 \end{bmatrix}$$

sedangkan untuk $l = 2$ didapatkan

$$H = \begin{bmatrix} 5 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 12 & 5 & \varepsilon & \varepsilon & \varepsilon \\ 19 & 12 & 5 & \varepsilon & \varepsilon \\ 26 & 19 & 12 & 5 & \varepsilon \\ 33 & 26 & 19 & 12 & 5 \end{bmatrix}, Y = \begin{bmatrix} 28 \\ 44 \\ 53 \\ 71 \\ 87 \end{bmatrix}$$

Sehingga waktu keberangkatan dari Tuban dalam bentuk bilangan bulat dan konversi ketanggal sebenarnya adalah

No	bilangan bulat	Tanggal Keberangkatan	Tanggal Tiba
1	15	15 September 2013	20 September 2013
2	23	23 September 2013	28 September 2013
3	31	1 Oktober 2013	6 Oktober 2013
4	39	9 Oktober 2013	14 Oktober 2013
5	42	12 Oktober 2013	17 Oktober 2013
6	58	18 Oktober 2013	23 Oktober 2013
7	59	29 Oktober 2013	3 November 2013
8	66	5 November 2013	10 November 2013
9	74	13 November 2013	18 November 2013
10	82	21 November 2013	26 November 2013
11	90	29 Desember 2013	4 Desember 2013

Simulasi dari contoh ini dapat dilakukan dengan menggunakan toolbox aljabar maxplus yang ada pada Scilab dengan menggunakan fungsi **supplychainschedule**.

Selanjutnya akan dibahas mengenai rantai pasok dengan memperhatikan kondisi kapasitas tanker yang tidak selalu memenuhi banyaknya permintaan BBM. Pada keadaan ini diberikan beberapa algoritma untuk menentukan penjadwalannya. Permasalahan dari keadaan ini hampir sama dengan keadaan sebelumnya yaitu menyelesaikan Persamaan 4.23. Perbedaannya adalah terletak pada nilai dari vektor $\mathbf{y}(k)$, kalau keadaannya tanpa melihat kapasitas dari kapal tanker nilai $\mathbf{y}(k)$ dapat langsung didapatkan dari tanggal permintaan sedangkan pada keadaan dengan memperhatikan kapasitas dari kapal tanker nilai $\mathbf{y}(k)$ tidak bisa langsung didapatkan namun terlebih dahulu harus melakukan pengolahan

Algorithm 1 Determine y and n_d where Date Of Demand(Dd), Number of Demand(Nd), Capacity of Tanker(Ct)

Input: n , array Dd, array Nd and array Ct

Output: n_d and array y

procedure

convert Dd to integer

Initialization $n_d = 0$ and $hit=0$

while $hit < \text{length of Dd}$ **do**

hit $\leftarrow$ hit+1

temp $\leftarrow$ Nd[hit]

while temp>0 **do**

temp $\leftarrow$ temp - Ct[mod(n_d, n)+1]

$n_d \leftarrow n_d + 1$

$y[n_d] \leftarrow$ Dd[hit]

end while

end while

return n_d and array y

end procedure

data dari tanggal permintaan, kapasitas tanker dan banyaknya permintaan. Algoritma untuk menentukan nilai dari $y(k)$ dan n_d ini dapat dilihat pada Algoritma 1.

Kemudian perhatikan bahwa jika susunan urutan dari array Ct dirubah maka nilai dari array y juga berubah. Dengan kata lain urutan keberangkatan kapal tanker juga berpengaruh terhadap penjadwalannya. Sehingga dalam penelitian ini dibuat penjadwalan untuk semua kemungkinan dan dipilih jadwal yang terbaik sesuai dengan kriteria tertentu. Algoritma mengenai penentuan jadwal dengan inputan array Ct dapat dilihat pada Algoritma 2.

Nilai u yang didapatkan dari Algoritma 2 masih perlu diproses lagi supaya penjadwalannya lebih optimal. Dalam proses ini digunakan algoritma 3. Algoritma 3 mereduksi banyaknya keberangkatan yang melayani setiap permintaan sedemikian hingga banyaknya jadwal tidak tepat waktu berkurang. Dari proses yang terakhir ini didapatkan jadwal keberangkatan kapal tanker.

Algoritma 1, 2 dan 3 telah diimplementasikan pada bahasa pemrograman Scilab dan telah dimasukkan pada toolbox aljabar max-plus dan petrinet dengan nama fungsinya **supplychainscheduleex**. Pada fungsi ini dibutuhkan inputan t_t , t_r dan data tanggal permintaan, banyaknya permintaan, dan kapasitas kapal dalam bentuk file excel. Hasil dari fungsi ini adalah penjadwalan dari keberangkatan kapal tanker beserta urutannya, tibanya kapal tanker ditempat konsumen dan juga selisih kedatangan kapal tanker dan tanggal pemesanan.

Selanjutnya akan dianalisis hasil penjadwalan dengan menggunakan satu dan dua tanker. Data yang digunakan pada bagian ini adalah data rantai pasok BBM dari TBBM Tuban ke TBBM Manggis.

Algorithm 2 Determine array u **Input:** n, t_r, t_t , array Dd, array Nd and array Ct**Output:** array u **procedure**Determine y and n_d

▷ using Algorithm 1

for l in $\{1, 2, \dots, n\}$ **do**Determine matrix Y_l and H_l

▷ using Equation 4.23

Determine U_l

▷ using Equation ??

end for**return** u **end procedure**

Algorithm 3 Determine component of departure u will be removed in every demand where Number of Demand(Nd), Capacity of Tanker(Ct) and time difference of arrival and demand (diff)

Input: Nd, Ct and diff**Output:** Array del consist of component departure will be removed**procedure** $nc \leftarrow$ length of CtInitialization del $\leftarrow$ number 1 to nc and small $\leftarrow$ infinityperpossible $\leftarrow$ permutation of dellperposs $\leftarrow$ row size of perpossible**for** i in $\{1, 2, \dots, nc\}$ **do**possible $\leftarrow$ perpossible(1:factorial(nc-i):lperposs,:)lpossible $\leftarrow$ row size of possible**for** j in $\{1, 2, \dots, lpossible\}$ **do****if** $\text{sum}(\text{Ct}(\text{possible}(j,:))) \geq \text{Nd}$ and $\text{sum}(\text{diff}(\text{possible}(j,:))) < \text{small}$ **then**

temp=possible(j,:);

end if**end for****end for**

remove temp from del

return del**end procedure**

4.6.1 Penjadwalan Dengan Menggunakan Satu Tanker

Pada bagian ini akan dibandingkan penjadwalan untuk beberapa kemungkinan kondisi dari rantai pasok yang berkaitan dengan kapasitas tanker. Banyaknya tanker yang dibahas pada bagian ini adalah satu. Data yang diperoleh terdiri dari 11 permintaan. Dari 11 permintaan

ini dijadwalkan untuk duabelas kasus yang mungkin yaitu kapasitas tanker melebihi semua permintaan dan kapasitas tanker melebihi semua permintaan kecuali permintaan pertama, kedua, ketiga sampai dengan permintaan ke-11. Penyajian perbandingan jadwal yang tidak tepat waktu dari 12 kasus dapat dilihat pada Tabel 4.2.

Tabel 4.2: Perbandingan jadwal yang tidak tepat waktu untuk banyaknya tanker satu

Kapasitas tanker kurang dari permintaan ke-	Banyaknya keberangkatan	Total Banyaknya hari tidak tepat waktu	Total permintaan tidak tepat waktu
tidak ada	11	15	5
1	12	24	6
2	12	32	6
3	12	40	6
4	12	48	6
5	12	51	6
6	12	57	6
7	12	40	7
8	12	47	8
9	12	46	9
10	12	44	10
11	12	41	10

Tabel 4.3: Tabel jadwal satu tanker untuk kasus kapasitas tanker melebihi semua permintaan kecuali permintaan pertama

Keberangkatan ke-	Keberangkatan	Tiba	Tanggal Permintaan
1	06-09-2013	11-09-2013	20-09-2013
2	13-09-2013	18-09-2013	20-09-2013
3	20-09-2013	25-09-2013	28-09-2013
4	27-09-2013	02-10-2013	06-10-2013
5	04-10-2013	09-10-2013	14-10-2013
6	11-10-2013	16-10-2013	17-10-2013
7	18-10-2013	23-10-2013	23-10-2013
8	29-10-2013	03-11-2013	03-11-2013
9	05-11-2013	10-11-2013	10-11-2013
10	13-11-2013	18-11-2013	18-11-2013
11	21-11-2013	26-11-2013	26-11-2013
12	29-11-2013	04-12-2013	04-12-2013

Perhatikan Tabel 4.2, pada kasus kapasitas kapal tanker melebihi semua permintaan didapatkan penjadwalan yang tidak tepat waktu sebanyak 15 hari. Dimana dalam hal ini kedatangan yang tidak tepat waktu terjadi pada permintaan pertama, kedua, ketiga, keempat, dan kelima. Lebih lanjut perhatikan tabel pada baris dua sampai dengan baris

tujuh, terlihat bahwa banyaknya permintaan tidak tepat waktu semuanya sama yaitu enam permintaan dan untuk baris selanjutnya banyaknya permintaan yang tidak tepat waktu semakin bertambah. Hal ini terjadi karena ketika permintaan ke- k yang melebihi kapasitas kapal maka untuk penjadwalan permintaan sebelumnya akan menyesuaikan dengan kondisi ini sedangkan untuk permintaan selanjutnya tidak terpengaruh. Untuk contoh penjadwalan salah satu kasus dapat dilihat pada Tabel 4.3.

4.6.2 Penjadwalan Dengan Menggunakan Dua Tanker

Pada bagian ini akan dibandingkan penjadwalan untuk beberapa kemungkinan kondisi dari rantai pasok yang berkaitan dengan kapasitas tanker. Banyaknya tanker yang dibahas pada bagian ini adalah dua. Dengan melihat penjadwalan pada bagian satu dengan kasus dimana semua permintaan tidak melebihi kapasitas kapal tanker masih ada kedatangan yang tidak tepat waktu yaitu pada permintaan satu sampai dengan lima. Maka dari itu pada bagian ini diasumsikan tanker pertama memiliki kapasitas yang melebihi dari semua permintaan. Dengan asumsi ini dijadwalkan untuk 6 kasus yaitu kapasitas dari tanker kedua kurang dari semua permintaan dan kapasitas dari tanker kedua kurang dari semua permintaan kecuali permintaan pertama, kedua, ketiga, keempat, dan kelima. Penyajian perbandingan jadwal yang tidak tepat waktu dari 6 kasus ini dapat dilihat pada Tabel 4.4.

Tabel 4.4: Perbandingan jadwal yang tidak tepat waktu untuk banyaknya tanker dua

Kapasitas tanker kedua lebih banyaknya dari permintaan ke-	Banyaknya keberangkatan	Total Banyaknya hari tidak tepat waktu	Jadwal tidak tepat waktu
tidak ada	11	15	5
1	11	13	4
2	11	10	3
3	11	6	2
4	11	1	1
5	11	0	0

Perhatikan Tabel 4.4, terlihat bahwa mulai dari baris pertama hingga terakhir banyaknya permintaan yang tidak tepat waktu semakin berkurang dan akhirnya pada kasus ke-5 tidak ada jadwal yang tidak tepat waktu. Hal ini terjadi karena ketika kapasitas kapal kedua melebihi permintaan ke- k maka untuk penjadwalan permintaan sebelumnya akan dipengaruhi oleh kondisi ini sedangkan untuk jadwal selanjutnya tidak dipengaruhi. Selanjutnya contoh penjadwalan untuk kasus kapasitas kapal tanker kedua kurang dari semua permintaan kecuali permintaan kelima dapat dilihat pada Tabel 4.5.

Perhatikan bahwa dengan mengatur kapal tanker yang beroperasi dan tanggal keberangkatan kapal tanker sebagaimana diberikan pada kolom ke-2 dan ke-3 dari Tabel 4.5, maka kolom ke-4 dan ke-5 dari Tabel 4.5 semua baris ke-1 sampai ke-11 mempunyai nilai yang sama. Ini berarti semua kedatangan kapal tanker sesuai dengan permintaan dari *costumer*.

Tabel 4.5: Tabel Jadwal dua Tanker untuk kasus kapasitas kapal tanker kedua kurang dari semua permintaan kecuali permintaan kelima

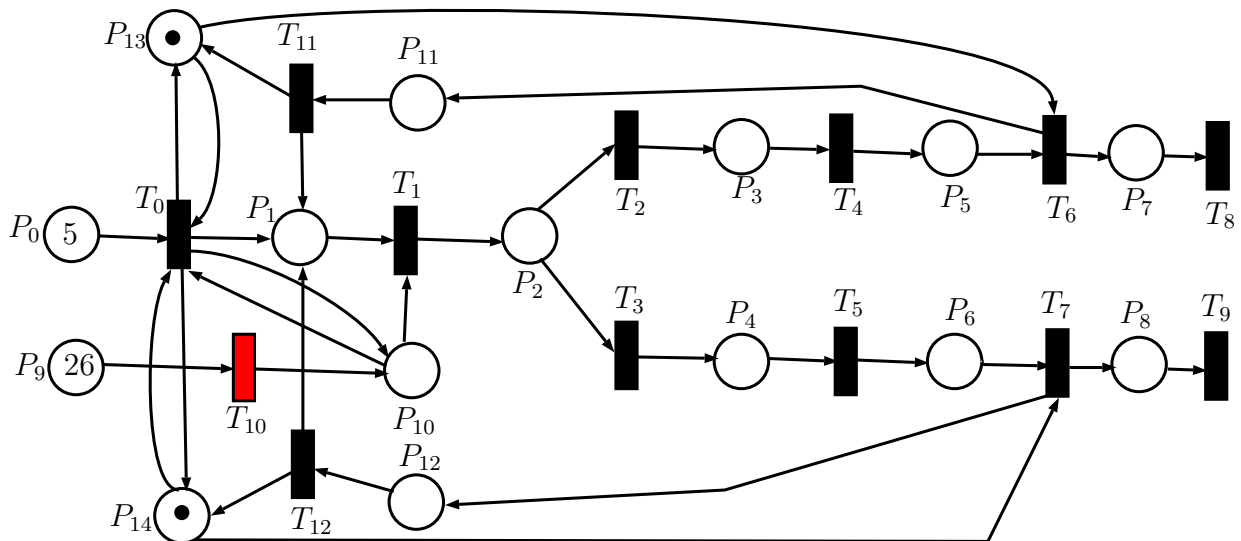
Permintaan ke-	kapal tanker yang beroperasi	Tanggl Keberangkatan	Tanggal Tiba	Tanggal Permintaan
1	pertama	15-09-2013	20-09-2013	20-09-2013
2	pertama	23-09-2013	28-09-2013	28-09-2013
3	pertama	01-10-2013	06-10-2013	06-10-2013
4	pertama	09-10-2013	14-10-2013	14-10-2013
5	kedua	12-10-2013	17-10-2013	17-10-2013
6	pertama	18-10-2013	23-10-2013	23-10-2013
7	pertama	29-10-2013	03-11-2013	03-11-2013
8	pertama	05-11-2013	10-11-2013	10-11-2013
9	pertama	13-11-2013	18-11-2013	18-11-2013
10	pertama	21-11-2013	26-11-2013	26-11-2013
11	pertama	29-11-2013	04-12-2013	04-12-2013

4.6.3 Model Kapasitas Tanker Sama dan Mempertimbangkan waktu *unloading* dan *loading* Produk

Dalam pembahasan terdahulu telah dibahas masalah rantai pasok pengiriman BBM premium menggunakan 2 kapal tanker pengiriman ke *costumer*. Dalam subbagian ini dibahas model aljabar max-plus dari rantai pasok pengiriman BBM solar dengan menggunakan 5 kapal tanker untuk mengirim BBM solar ke pelanggan (*costumer*). Materi pembahasan dalam subbagian ini diambil dari [23].

Rantai pasok dalam pembahasan ini adalah distribusi solar dari TBBM Tuban sebagai pemasok menuju 2 pelanggan yaitu TBBM Manggis di Bali dan Tanjung Wangi di Banyuwangi. Pengiriman Solar dilakukan melalui jalur laut dengan menggunakan kapal tanker milik PT. Pertamina. Adapun kapal tanker yang bisa digunakan untuk pengiriman pasokan solar di wilayah kerja PT. Pertamina tersebut sebanyak 5 dengan 3 kapal tanker berkapasitas 22.000 Kl dan 2 lainnya 25.000 Kl. Kelima kapal tanker tersebut adalah Fastron ex Tuban, Maiden, Pematang ex Tuban, Ae Pioner ex Tuban dan MT Pematang ([22]). Pada model pertama ini kapasitas dari 5 kapal tanker dianggap sama yakni masing-masing sebesar 22.000 Kl. Hal ini disebabkan karena berdasarkan data yang diperoleh menunjukkan bahwa volume permintaan dari kastomer kurang dari atau sama dengan 22.000 Kl sehingga seluruh kapal tanker dapat digunakan untuk melakukan pengiriman solar. Pada model ini dibuat pengaturan prioritas penggunaan kapal tanker yaitu diprioritaskan melakukan pengiriman solar menggunakan kapal tanker yang telah kembali dari pengiriman sebelumnya sehingga diharapkan jumlah kapal tanker yang beroperasi dapat diminimumkan. Alur pengiriman solar diawali dengan penetapan waktu dan volume permintaan dari TBBM Manggis dan Tanjung Wangi. Waktu permintaan dalam satuan menit, jam, tanggal, bulan dan tahun. Sedangkan volume permintaan dalam satuan Kilo liter (Kl). Selanjutnya TBBM Tuban mempersiapkan solar yang akan dikirim. Proses

diawali dengan persiapan kapal tanker. Kemudian dilanjutkan dengan proses loading. Proses loading adalah pekerjaan pengangkutan BBM solar dari gudang ke kapal tanker ([22]). Kemudian dilanjutkan dengan keberangkatan kapal tanker menuju TBBM customer yaitu TBBM Manggis di Bali atau Tanjung Wangi di Banyuwangi. Setelah sampai di TBBM pelanggan selanjutnya dilakukan proses *unloading*. Proses *unloading* adalah pekerjaan yang dilakukan pada saat kapal tanker sampai pada pelanggan yaitu pekerjaan pengangkutan BBM solar dari kapal tanker ke tangki (gudang) penyimpanan pelanggan ([22]). Selanjutnya solar diterima oleh TBBM pelanggan dan kapal tanker kembali ke TBBM Tuban untuk melakukan pengiriman selanjutnya. Rantai pasok distribusi solar ini dimodelkan menggunakan Petri Net dengan 14 *place* dan 12 *transisi*. Terdapat 5 *token* pada *place* P_0 menunjukkan banyaknya kapal tanker yang bisa digunakan untuk pengiriman solar yaitu sebanyak 5 kapal tanker. Pada *place* P_9 terdapat 26 *token* menunjukkan solar siap dikirim untuk memenuhi 26 permintaan dari pelanggan. *Token* pada *place* P_9 ditulis dengan angka karena jumlahnya besar yaitu 26. Sedangkan pada *place* P_{13} dan P_{14} masing-masing memuat 1 *token* yang menunjukkan kapal tanker baru siap digunakan karena tidak ada kapal tanker yang kembali dari TBBM pelanggan. Jika tidak ada *token* di *place* P_{13} menunjukkan bahwa ada kapal tanker yang kembali dari TBBM Manggis dan jika tidak ada *token* di *place* P_{14} menunjukkan bahwa ada kapal tanker yang kembali dari Tanjung wangi. Keadaan awal model Petri Net dapat dilihat pada Gambar 4.17.



Gambar 4.17: Petri net dengan Kapasitas Tanker dianggap sama

Penjelasan Gambar 4.17:

Place:

P_0 : Kapal tanker

P_1 : Kapal tanker siap

P_2 : Proses *loading* P_3 : Perjalanan ke TBBM Manggis

- P_4 : Perjalanan ke TBBM Tanjung Wangi
 P_5 : Proses *unloading* di TBBM Manggis
 P_6 : Proses *unloading* di TBBM Tanjung Wangi
 P_7 : Produk diterima di TBBM Manggis
 P_8 : Produk diterima di TBBM Tanjung Wangi
 P_9 : Produk solar (26 permintaan)
 P_{10} : Proses masuknya solar
 P_{11} : Perjalanan kembali dari TBBM Manggis
 P_{12} : Perjalanan kembali dari TBBM Tanjung Wangi
 P_{13} : Kapal tanker baru siap digunakan karena tidak ada yang kembali dari TBBM Manggis (ditunjukkan dengan adanya 1 token)
 P_{14} : Kapal tanker baru siap digunakan karena tidak ada yang kembali dari TBBM Tanjung Wangi (ditunjukkan dengan adanya 1 token).

Transisi:

- T_0 : Mulai persiapan Kapal tanker
 T_1 : Mulai proses *loading*
 T_2 : Berangkat ke TBBM Manggis
 T_3 : Berangkat ke TBBM Tanjung Wangi
 T_4 : Mulai *unloading* di TBBM Manggis
 T_5 : Mulai *unloading* di TBBM Tanjung Wangi
 T_6 : Selesai *unloading* di TBBM Manggis
 T_7 : Selesai *unloading* di TBBM Tanjung Wangi
 T_8 : Produk di TBBM Manggis siap dipasarkan
 T_9 : Produk di TBBM Tanjung Wangi siap dipasarkan
 T_{10} : Solar siap dikirim
 T_{11} : Kapal tanker dari TBBM Manggis masuk TBBM Tuban
 T_{12} : Kapal tanker dari TBBM Tanjung Wangi masuk TBBM Tuban.

Gambar 4.17 menunjukkan keadaan awal Petri Net sebelum *didifire*. Transisi T_{10} berwarna merah menunjukkan bahwa *enabled*. Transisi T_{10} *enabled* karena *place* P_9 yang menjadi input dari transisi T_{10} mempunyai *token* lebih dari atau sama dengan jumlah *token* minimum yang dibutuhkan oleh *transisi* T_{10} . Hal ini menunjukkan solar yang akan dikirim untuk memenuhi 26 permintaan telah siap. Sebagaimana telah dibahas dalam Bagian 4.2 akibat pem-*fire*-an suatu transisi yang *enable*, maka terjadi perubahan *token* yang ada di dalam masing-masing *place*. Perubahan *token* yang terjadi atau kedinamikan dari Petri net dapat disajikan oleh persamaan berikut

$$\mathbf{x} = \mathbf{x}_0 + A\mathbf{e}_j,$$

dimana $\mathbf{x}_0$ adalah keadaan awal dari *place* yang bertanda (ada *token*) sebelum suatu transisi *enable* di-*fire*. Dalam Petri net Gambar 4.17, $\mathbf{x}_0$ mempunyai 15 komponen yang masing-masing komponen menyatakan *place*. Vektor keadaan awal $\mathbf{x}_0$ dan urutan kom-

ponen *placanya* diberikan oleh

$$\mathbf{x}_0 = \begin{bmatrix} 5 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 26 \end{bmatrix}^T.$$

$$\begin{array}{ccccccccccccccc} \swarrow & & \dots & \dots & \dots & \dots & & \downarrow & & \dots & \dots & \dots & \dots & & \searrow \\ P_0 & P_1 & P_{10} & P_{11} & P_{12} & P_{13} & P_{14} & P_2 & P_3 & P_4 & P_5 & P_6 & P_7 & P_8 & P_9 \end{array}$$

Dari $\mathbf{x}_0$ dan urutan komponen *place* dari vektor $\mathbf{x}_0$ terlihat bahwa dalam *place* P_0 ada 5 *token*, P_1 ada 0 *token*, P_{10} ada 0 *token*, P_{11} ada 0 *token*, P_{12} ada 0 *token*, P_{13} ada 1 *token*, P_{14} ada 1 *token*, P_2 ada 0 *token*, P_3 ada 0 *token*, P_4 ada 0 *token*, P_5 ada 0 *token*, P_6 ada 0 *token*, P_7 ada 0 *token*, P_8 ada 0 *token* dan P_9 ada 16 *token*. Urutan dari 13 transisi Petri net tersebut diberikan oleh

$$T_0 \quad T_1 \quad T_{10} \quad T_{11} \quad T_{12} \quad T_2 \quad T_3 \quad T_4 \quad T_5 \quad T_6 \quad T_7 \quad T_8 \quad T_9$$

Urutan dari *place* sebanyak $m = 15$ dan transisi sebanyak $n = 13$ penting untuk menentukan elemen matriks *forward incidence* dan *backward incidence* dari Petri net yang dibahas. Sedangkan $A = A_f - A_b$, dimana A_f adalah *forward incidence* dan A_b adalah *backward incidence*, dan $\mathbf{e}_j$ adalah vektor kolom satuan dengan 13 komponen dan komponen ke- j adalah 1 menandakan bahwa transisi ke- j *enable* di-*fire*. Masing-masing matriks A_f dan A_b berukuran 15×13 . Hal ini menunjukkan bahwa banyaknya *place* dari Petri net adalah 15 sedangkan banyaknya *transisi* adalah 13. Matriks A_f dan A_b adalah:

$$A_f = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

dan

$$A_b = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

dengan Matriks *incidence* dari Petri net tersebut diberikan oleh

$$A = A_f - A_b = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Selanjutnya bila transisi T_{10} (urutan ke-3 dari transisi yang telah disusun) *difire* didapat

$$\mathbf{e}_3 = [0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$$

dan akibat pem-*fire*-an tersebut didapat

$$\begin{aligned} \mathbf{x} &= \mathbf{x}_0 + A\mathbf{e}_3 \\ &= \begin{bmatrix} 5 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 26 \end{bmatrix} + \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \end{aligned}$$

setelah dihitung vektor $\mathbf{x}$ adalah

$$\mathbf{x} = \begin{bmatrix} 5 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 25 \end{bmatrix} \rightarrow \begin{matrix} P_0 \\ P_1 \\ P_{10} \\ P_{11} \\ P_{12} \\ P_{13} \\ P_{14} \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \\ P_7 \\ P_8 \\ P_9 \end{matrix}$$

Akibat pem-*fire*-an transisi *enable* T_{10} dengan keadaan awal $\mathbf{x}_0$ didapat vektor keadaan $\mathbf{x}$ yang berarti bahwa Petri net semula berubah menjadi Petri net dengan keadaan awal $\mathbf{x}$. Dengan demikian dalam *place* P_0 tetap ada 5 *token*, P_{10} ada 1 *token*, P_{13} dan P_{14} tetap ada 1 *token* sedangkan P_9 berubah ada 25 *token* yang semula 26 *token*. Selain dari pada itu, akibat pem-*fire*-an T_{10} transisi T_0 menjadi *enable* dan dalam *place* P_{10} ada 1 *token*. Hal ini menunjukkan bahwa setelah ada solar yang siap untuk mulai proses *loading* maka kapal tanker juga bisa dipersiapkan. Selanjutnya untuk keadaan awal $\mathbf{x}_0 = \mathbf{x}$ dan bila transisi

enable T_0 di-*fire* didapat

$$\begin{aligned}
 \mathbf{x} &= \mathbf{x}_0 + A\mathbf{e}_1 \\
 &= \begin{bmatrix} 5 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 26 \end{bmatrix} + \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \\
 &= \begin{bmatrix} 4 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 25 \end{bmatrix} \begin{matrix} \rightarrow P_0 \\ \rightarrow P_1 \\ \rightarrow P_{10} \\ \rightarrow P_{11} \\ \rightarrow P_{12} \\ \rightarrow P_{13} \\ \rightarrow P_{14} \\ \rightarrow P_2 \\ \rightarrow P_3 \\ \rightarrow P_4 \\ \rightarrow P_5 \\ \rightarrow P_6 \\ \rightarrow P_7 \\ \rightarrow P_8 \\ \rightarrow P_9 \end{matrix}
 \end{aligned}$$

Keadaan $\mathbf{x}$ menunjukkan perubahan *token* karena transisi T_0 di-*fire*. Hal ini menyebabkan transisi T_1 menjadi *enable*. Hal ini menunjukkan bahwa pasokan solar untuk 1 permintaan dan sebuah kapal tanker telah dipersiapkan sehingga proses *loading* siap dimulai. Dengan

keadaan awal $\mathbf{x}_0 = \mathbf{x}$ dan transisi *enable* T_1 di-*fire* didapat

$$\begin{aligned}
 \mathbf{x} &= \mathbf{x}_0 + A\mathbf{e}_2 \\
 &= \begin{bmatrix} 4 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 25 \end{bmatrix} + \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \\
 &= \begin{bmatrix} 4 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 25 \end{bmatrix} \begin{matrix} \rightarrow P_0 \\ \rightarrow P_1 \\ \rightarrow P_{10} \\ \rightarrow P_{11} \\ \rightarrow P_{12} \\ \rightarrow P_{13} \\ \rightarrow P_{14} \\ \rightarrow P_2 \\ \rightarrow P_3 \\ \rightarrow P_4 \\ \rightarrow P_5 \\ \rightarrow P_6 \\ \rightarrow P_7 \\ \rightarrow P_8 \\ \rightarrow P_9 \end{matrix}
 \end{aligned}$$

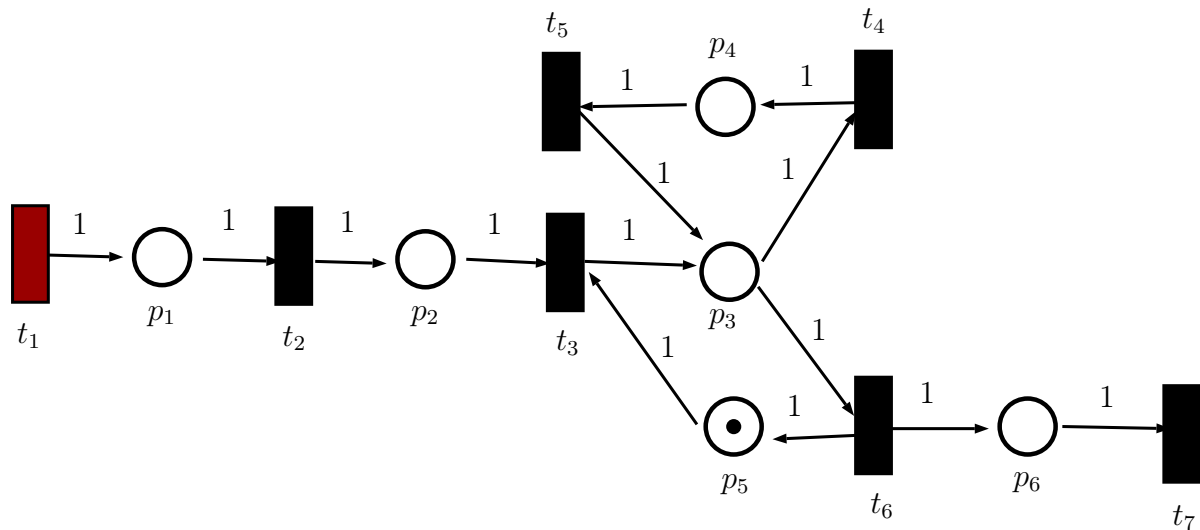
4.7 Penggunaan *Eigenmode* Dalam Sistem Antrian

Pada bagian ini diberikan aplikasi *Eiegenmode* pada masalah antrian. Masalah antrian yang dibahas adalah masalah sistem antrian pelayanan proses pergantian jenis tabungan di suatu *bank* pada satu petugas *customer service*. Pembahasan dimulai dengan membuat model Petri net dari sistem antrian. Selanjutnya memberikan penjelasan masing-masing *transisi* dan masing-masing *place*. Berikutnya didefinisikan peubah-peubah yang digunakan dalam proses pemodelan. Peubah yang digunakan terdiri dari dua jenis, yaitu

peubah yang menunjukkan waktu dan peubah yang menunjukkan lama waktu. Peubah-peubah tersebut sangat penting untuk membuat model dalam bentuk aljabar max-plus. Dari bentuk model aljabar max-plus diperoleh suatu matriks tereduksi yang selanjutnya dikaji perilaku *eigenmode* untuk menentukan waktu proses pelayanan *customer*.

Fenomena antrian pelayanan proses pergantian jenis tabungan biasanya terjadi di kantor cabang *bank* yang terletak di dalam suatu perguruan tinggi saat menjelang wisuda.

Misal dalam suatu *Bank* akan dilakukan analisa sistem antrian pelayanan pergantian jenis tabungan *customer* pada satu petugas *customer service*. Adapun proses pelayanan *customer* mulai datang ke *bank*, dilayani petugas *customer service*, sampai meninggalkan *bank* diberikan dalam bentuk Petri net pada Gambar 4.18.



Gambar 4.18: Petri Net Antrian Pelayanan Pergantian Jenis Tabungan pada Satu Petugas *Customer Service*

Berdasarkan Gambar 4.18, Petri net antrian pelayanan pergantian jenis tabungan *bank* pada satu petugas *customer service* terdiri dari tujuh transisi, yaitu:

t_1 : *customer* datang ke bank,

t_2 : *customer* mengambil nomor antrian,

t_3 : *costumer* dilayani oleh *customer service*,

t_4 : *customer service* membawa berkas *customer* pada (*teller*),

t_5 : berkas *customer* selesai diproses *teller*,

t_6 : *customer* selesai dilayani oleh *customer service*,

t_7 : *customer* meninggalkan bank,

dan enam *place*, yaitu:

p_1 : *customer* yang sedang menunggu giliran mengambil nomor antrian,

p_2 : *customer* yang sedang menunggu giliran dilayani *customer service*,

p_3 : *customer* yang sedang dilayani *customer service*,

p_4 : *customer* yang sedang menunggu pemrosesan berkas oleh *teller*,

p_5 : *Idle* atau *customer service* sedang tidak sibuk,

p_6 : *customer* yang sudah selesai dilayani oleh *customer service*.

Selanjutnya, diberikan pula definisi peubah-peubah yang digunakan dalam proses pemodelan. Peubah yang digunakan terdiri dari dua jenis, yaitu peubah yang menunjukkan waktu dan peubah yang menunjukkan lama waktu. Peubah-peubah tersebut sangat penting untuk membuat model dalam bentuk aljabar max-plus. Ada tujuh peubah waktu yang dibutuhkan dalam pemodelan yaitu:

$t_1(k)$: waktu kedatangan *customer* saat ke- k ,

$t_2(k)$: waktu *customer* mengambil nomor antrian saat ke- k ,

$t_3(k)$: waktu *customer* mulai dilayani *customer service* saat ke- k ,

$t_4(k)$: waktu *customer service* membawa berkas *customer* pada (*teller*) saat ke- k ,

$t_5(k)$: waktu berkas *customer* selesai diproses *teller* saat ke- k ,

$t_6(k)$: waktu *customer* selesai dilayani *customer service* saat ke- k ,

$t_7(k)$: waktu *customer* meninggalkan bank saat ke- k .

Sedangkan peubah-peubah yang menunjukkan lama waktu ada lima peubah. Peubah-peubah tersebut adalah:

$v_{t_1,k}$: lama kedatangan *customer* saat ke- k ,

$v_{t_2,k}$: lama *customer* mengambil nomor antrian saat ke- k ,

$v_{t_5,k}$: lama pemrosesan berkas oleh *teller* saat ke- k ,

$v_{t_6,k}$: lama *customer* dilayani oleh *customer* saat ke- k ,

$v_{t_7,k}$: lama *customer* meninggalkan bank saat ke- k .

Dari peubah-peubah yang telah diperoleh yaitu waktu dan peubah yang menunjukkan lama waktu, berikutnya dibuat proses penyusunan model sistem antrian pelayanan pergantian jenis tabungan *bank* pada satu petugas *customer service* menggunakan aljabar max-plus berdasarkan Petri net pada Gambar 4.18 sebagai berikut:

$$\begin{aligned}
t_1(k) &= v_{t_1,k} \otimes t_1(k-1), \\
t_2(k) &= v_{t_2,k} \otimes t_1(k) \\
&= v_{t_2,k} \otimes v_{t_1,k} \otimes t_1(k-1), \\
t_3(k) &= t_2(k) \oplus t_6(k-1) \\
&= v_{t_2,k} \otimes v_{t_1,k} \otimes t_1(k-1) \oplus t_6(k-1), \\
t_4(k) &= v_{t_3,k} \oplus t_5(k-1) \\
&= v_{t_2,k} \otimes v_{t_1,k} \otimes t_1(k-1) \oplus t_6(k-1) \oplus t_5(k-1), \\
t_5(k) &= v_{t_5,k} \otimes t_4(k) \\
&= v_{t_5,k} \otimes (v_{t_2,k} \otimes v_{t_1,k} \otimes t_1(k-1) \oplus t_6(k-1) \oplus t_5(k-1)) \\
&= v_{t_5,k} \otimes v_{t_2,k} \otimes v_{t_1,k} \otimes t_1(k-1) \oplus v_{t_5,k} \otimes t_6(k-1) \oplus v_{t_5,k} \otimes t_5(k-1), \\
t_6(k) &= v_{t_6,k} \otimes (t_3(k) \oplus t_5(k)) \\
&= v_{t_6,k} \otimes t_3(k) \oplus v_{t_6,k} \otimes t_5(k) \\
&= v_{t_6,k} \otimes (v_{t_2,k} \otimes v_{t_1,k} \otimes t_1(k-1) \oplus t_6(k-1)) \oplus \\
&\quad v_{t_6,k} \otimes (v_{t_5,k} \otimes v_{t_2,k} \otimes v_{t_1,k} \otimes t_1(k-1) \oplus v_{t_5,k} \otimes t_6(k-1) \oplus v_{t_5,k} \otimes t_5(k-1)) \\
&= v_{t_6,k} \otimes v_{t_2,k} \otimes v_{t_1,k} \otimes t_1(k-1) \oplus v_{t_6,k} \otimes t_6(k-1) \oplus \\
&\quad v_{t_6,k} \otimes v_{t_5,k} \otimes v_{t_2,k} \otimes v_{t_1,k} \otimes t_1(k-1) \oplus \\
&\quad v_{t_6,k} \otimes v_{t_5,k} \otimes t_6(k-1) \oplus v_{t_6,k} \otimes v_{t_5,k} \otimes t_5(k-1) \\
&= v_{t_6,k} \otimes v_{t_5,k} \otimes v_{t_2,k} \otimes v_{t_1,k} \otimes t_1(k-1) \oplus \\
&\quad v_{t_6,k} \otimes v_{t_5,k} \otimes t_6(k-1) \oplus v_{t_6,k} \otimes v_{t_5,k} \otimes t_5(k-1), \\
t_7(k) &= v_{t_7,k} \otimes t_6(k) \\
&= v_{t_7,k} \otimes (v_{t_6,k} \otimes v_{t_5,k} \otimes v_{t_2,k} \otimes v_{t_1,k} \otimes t_1(k-1) \oplus \\
&\quad v_{t_6,k} \otimes v_{t_5,k} \otimes t_6(k-1) \oplus v_{t_6,k} \otimes v_{t_5,k} \otimes t_5(k-1)) \\
&= v_{t_7,k} \otimes v_{t_6,k} \otimes v_{t_5,k} \otimes v_{t_2,k} \otimes v_{t_1,k} \otimes t_1(k-1) \oplus \\
&\quad v_{t_7,k} \otimes v_{t_6,k} \otimes v_{t_5,k} \otimes t_6(k-1) \oplus \\
&\quad v_{t_7,k} \otimes v_{t_6,k} \otimes v_{t_5,k} \otimes t_5(k-1).
\end{aligned}$$

dengan $k = 1, 2, \dots$

Didapatkan model antrian pelayanan pergantian jenis tabungan *bank* pada satu petugas *customer service* sebagai berikut:

$$\begin{bmatrix} t_1(k) \\ t_5(k) \\ t_6(k) \end{bmatrix} = \begin{bmatrix} v_{t_1,k} & c_1 & c_2 \\ v_{t_5,k} \otimes v_{t_2,k} \otimes v_{t_1,k} & v_{t_5,k} & v_{t_5,k} \\ v_{t_6,k} \otimes v_{t_5,k} \otimes v_{t_2,k} \otimes v_{t_1,k} & v_{t_6,k} \otimes v_{t_5,k} & v_{t_6,k} \otimes v_{t_5,k} \end{bmatrix} \otimes \begin{bmatrix} t_1(k-1) \\ t_5(k-1) \\ t_6(k-1) \end{bmatrix},$$

dengan notasi c_1 dan c_2 dipilih supaya

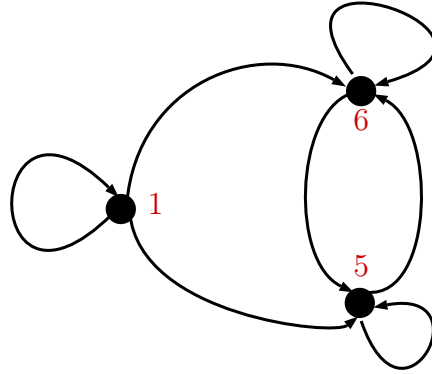
$$(v_{t_1,k} \otimes t_1(k-1)) \oplus (c_1 \otimes t_5(k-1)) \oplus (c_2 \otimes t_6(k-1)) = v_{t_1,k} \otimes t_1(k-1).$$

Sehingga dapat diambil $c_1 = c_2 = \varepsilon$, dan model antrian pelayanan pergantian jenis tabungan *bank* pada satu petugas *customer service* menjadi:

$$\begin{bmatrix} t_1(k) \\ t_5(k) \\ t_6(k) \end{bmatrix} = \begin{bmatrix} v_{t_1,k} & \varepsilon & \varepsilon \\ v_{t_5,k} \otimes v_{t_2,k} \otimes v_{t_1,k} & v_{t_5,k} & v_{t_5,k} \\ v_{t_6,k} \otimes v_{t_5,k} \otimes v_{t_2,k} \otimes v_{t_1,k} & v_{t_6,k} \otimes v_{t_5,k} & v_{t_6,k} \otimes v_{t_5,k} \end{bmatrix} \otimes \begin{bmatrix} t_1(k-1) \\ t_5(k-1) \\ t_6(k-1) \end{bmatrix}.$$

Tabel 4.6: Daftar Proses Pelayanan Pergantian Jenis Tabungan *Bank*

Kode	Proses	Lama Waktu (menit)
$v_{t_1,k}$	Lama kedatangan <i>customer</i> saat ke- k	8
$v_{t_2,k}$	Lama <i>customer</i> mengambil nomor antrian saat ke- k	0,5
$v_{t_5,k}$	Lama pemrosesan berkas oleh <i>teller</i> saat ke- k	5
$v_{t_6,k}$	Lama <i>customer</i> dilayani oleh <i>customer service</i> saat ke- k	20



Gambar 4.19: Graf Komunikasi $\mathcal{G}(B)$

Jika lama masing-masing proses diberikan pada Tabel 4.6, maka didapatkan persamaan:

$$\begin{bmatrix} t_1(k) \\ t_5(k) \\ t_6(k) \end{bmatrix} = \begin{bmatrix} 8 & \varepsilon & \varepsilon \\ 13,5 & 5 & 5 \\ 33,5 & 25 & 25 \end{bmatrix} \otimes \begin{bmatrix} t_1(k-1) \\ t_5(k-1) \\ t_6(k-1) \end{bmatrix}.$$

Matriks dari model yang diperoleh merupakan matriks tereduksi representasi dari graf tidak *strongly connected* pada Gambar 4.19.

Selanjutnya, akan dianalisa nilai eigen, vektor eigen, dan *eigenmode* dari matriks

$$B = \begin{bmatrix} 8 & \varepsilon & \varepsilon \\ 13,5 & 5 & 5 \\ 33,5 & 25 & 25 \end{bmatrix}.$$

Berdasarkan hasil karakterisasi nilai eigen dan vektor eigen dari matriks tereduksi, diketahui matriks tereduksi belum tentu memiliki nilai eigen. Oleh karena itu, berikut akan dicari nilai eigen dari matriks tereduksi B dengan menggunakan Algoritma *Power*. Misal dengan keadaan awal $\mathbf{x}(0) = (0 \ 0 \ 0)^T$, diperoleh evolusi keadaan

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} 8 \\ 13,5 \\ 33,5 \end{bmatrix}, \quad \begin{bmatrix} 16 \\ 38,5 \\ 58,5 \end{bmatrix}, \quad \dots$$

Berdasarkan evolusi keadaan, tidak dapat ditemukan bilangan bulat $p > q \geq 0$ dan bilangan real c yang memenuhi $\mathbf{x}(p) = c \otimes \mathbf{x}(q)$. Jadi B tidak memiliki nilai eigen. Meskipun demikian, karena B adalah matriks tereduksi reguler maka dapat dicari *eigenmode* berupa pasangan vektor dengan semua elemen vektor berhingga.

Untuk mendapatkan *eigenmode* dari matriks tereduksi reguler B , terlebih dahulu ditentukan bentuk matriks blok segitiga atas dari B , yaitu:

$$A = \begin{bmatrix} 5 & 5 & 13,5 \\ 25 & 25 & 33,5 \\ \varepsilon & \varepsilon & 8 \end{bmatrix}.$$

Berikutnya, dihitung nilai eigen dari matriks $A_{2,2}$, yaitu $\lambda_2 = 8$, sehingga dapat diambil $\xi_2 = \lambda_2 = 8$ dan misal diambil $\mathbf{v}_2 = 0$. Langkah selanjutnya, dihitung nilai eigen dari matriks $A_{1,1} = \begin{bmatrix} 5 & 5 \\ 25 & 25 \end{bmatrix}$ menggunakan Algoritma *Power*. Dengan keadaan awal $\mathbf{x}(0) = (0 \ 0)^T$, diperoleh evolusi keadaan

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} 5 \\ 25 \end{bmatrix}, \quad \begin{bmatrix} 30 \\ 50 \end{bmatrix}, \quad \dots$$

Sehingga $\lambda_1 = \frac{25}{2-1} = 25$. Karena $\lambda_1 > \xi_2$, maka $\xi_1 = \lambda_1 = 25$ dan dihitung vektor $\mathbf{v}_1$:

$$\begin{aligned} \xi_1 \otimes \mathbf{v}_1 &= (A_{1,1} \otimes \mathbf{v}_1) \oplus (A_{1,2} \otimes \mathbf{v}_2) \\ 25 \otimes \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} &= \left(\begin{bmatrix} 5 & 5 \\ 25 & 25 \end{bmatrix} \otimes \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \right) \oplus \left(\begin{bmatrix} 13,5 \\ 33,5 \end{bmatrix} \otimes 0 \right) \end{aligned} \quad (4.24)$$

Dari Persamaan (4.24) didapatkan

$$\begin{aligned} 25 + v_1 &= \max\{5 + v_1, 5 + v_2, 13, 5\} \\ 25 + v_1 &= 13, 5 \\ v_1 &= -11, 5, \end{aligned}$$

dan

$$\begin{aligned} 25 + v_2 &= \max\{25 + v_1, 25 + v_2, 33, 5\} \\ 25 + v_2 &= \max\{13, 5, 25 + v_2, 33, 5\} \\ 25 + v_2 &= 33, 5 \\ v_2 &= 8, 5. \end{aligned}$$

Jadi, didapatkan $\mathbf{v}_1 = \begin{bmatrix} -11, 5 \\ 8, 5 \end{bmatrix}$. Oleh karena itu, pasangan vektor $(\boldsymbol{\eta}, \mathbf{v})$ dengan $\boldsymbol{\eta} = \begin{pmatrix} 25 & 25 & 8 \end{pmatrix}^T$ dan $\mathbf{v} = \begin{pmatrix} -11, 5 & 8, 5 & 0 \end{pmatrix}^T$ adalah *eigenmode* dari matriks A sebab untuk $k = 0$, memenuhi:

$$A \otimes (0 \times \boldsymbol{\eta} + \mathbf{v}) = \begin{pmatrix} 13, 5 & 33, 5 & 8 \end{pmatrix}^T = 1 \times \boldsymbol{\eta} + \mathbf{v},$$

untuk $k = 1$, memenuhi:

$$A \otimes (1 \times \boldsymbol{\eta} + \mathbf{v}) = \begin{pmatrix} 38, 5 & 58, 5 & 16 \end{pmatrix}^T = 2 \times \boldsymbol{\eta} + \mathbf{v},$$

dan seterusnya, vektor $\boldsymbol{\eta}$ dan $\mathbf{v}$ untuk $k = 0, 1, 2, \dots$ memenuhi

$$A \otimes (k \times \boldsymbol{\eta} + \mathbf{v}) = (k + 1) \times \boldsymbol{\eta} + \mathbf{v}.$$

Dari hasil *eigenmode*, dapat diketahui waktu berakhirnya tiap proses pelayanan *customer* saat ke- k . Misal waktu paling awal terjadi pada pukul 08.00, maka untuk k sama dengan 0 dan 1 didapatkan hasil seperti pada 4.7.

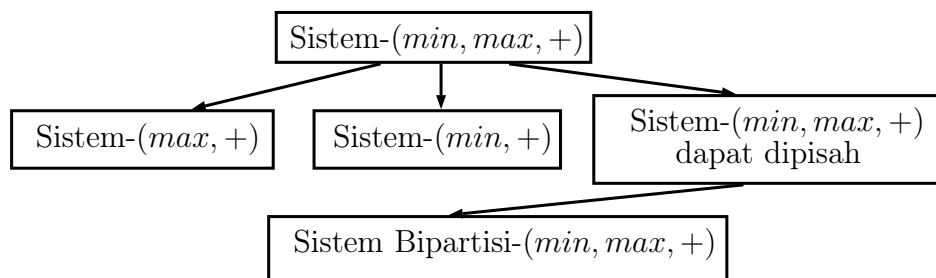
Tabel 4.7: Waktu Proses Pelayanan *Customer* Pertama dan Kedua

Keterangan	Waktu untuk <i>Customer</i>	
	Pertama	Kedua
Berkas <i>customer</i> selesai diproses <i>teller</i>	08:05:30	08:30:30
<i>Customer</i> selesai dilayani <i>customer service</i>	08:25:30	08:50:30
Kedatangan <i>customer</i>	08:00:00	08:08:00

Bab 5

Pengenalan Sistem- $(\min, \max, +)$

Dalam bab dikenalkan sistem Min Max Plus khususnya sistem bipartisi min max plus. Sistem yang dibahas adalah bagian dari apa yang dinamakan Sistem Dinamis Event Diskrit. Kejadian diskrit sistem dapat digunakan untuk mempelajari proses yang didorong oleh terjadinya peristiwa. Peubah yang relevan merupakan waktu di mana kejadian berlangsung. Diasumsikan bahwa evolusi sistem dapat digambarkan dengan cara persamaan menggunakan tiga jenis operasi, yaitu minimum, maksimum dan tambah. Sistem yang didasarkan pada ketiga operasi ini disebut sistem- $(\min, \max, +)$, dan sistem yang hanya didasarkan pada dua operasi minimum dan tambah disebut sistem- $(\min, +)$. Sedangkan sistem yang menggunakan dua operasi yaitu maksimum dan tambah disebut sistem- $(\max, +)$. Berbagai kelas sistem- $(\min, \max, +)$ sistem dapat didefinisikan. Hubungan antara kelas-kelas tersebut diberikan oleh gambar diagram berikut.



Suatu perilaku dinamis tertentu sistem- $(\min, \max, +)$ telah dikenalkan di [4], juga keberadaan nilai eigen dari sistem telah dipelajari. Lebih lanjut tentang sistem- $(\min, \max, +)$ dan dasar-dasar teori event sistem diskrit dengan operasi minimum, maksimum dan tambah dapat ditemukan dalam [13]. Dalam [5] disebutkan bahwa keberadaan struktur eigen untuk sistem bipartisi- $(\min, \max, +)$ telah dipelajari untuk pertama kalinya. Dalam bagian ini, dibahas kondisi keberadaan ini didasarkan pada hasil yang muncul di [9]. Kondisi ini bentuk sistem matriks yang menggambarkan suatu sistem bipartisi- $(\min, \max, +)$. Sistem matriks diwakili oleh sepasang matriks dalam bentuk (A, B) yang mana untuk keberadaan struktural eigenvalue didasarkan pada pengertian matriks taktereduksi. Penggunaan gagasan ketaktereduksian dalam konteks sistem- $(\min, \max, +)$ adalah baru, meskipun

gagasan itu sendiri telah digunakan sebelumnya dalam konteks lain dan dalam bentuk alternatif dalam [30]. Hal ini dapat dibandingkan dengan gagasan yang telah dikenal matriks persegi taktereduksi (lihat misalnya [2]). Istilah " struktural " yang berkaitan dengan sifat keberadaan (struktural) eigenvalue hanya tergantung pada " struktur " dari matriks yang bersangkutan dan bukan pada nilai-nilai numerik tertentu yang digunakan. Dalam konteks ini, nilai eigen dari pasangan matriks taktereduksi disebut struktural.

Berbeda dengan hasil di atas, dua sistem bipartisi- $(\min, \max, +)$ dapat dikombinasikan untuk membentuk sistem yang lebih besar. Jika dua sistem bipartisi- $(\min, \max, +)$ yaitu S_1 dan S_2 terhubung, masing-masing dengan eigenvalue struktural, maka sistem terhubung meskipun masih sistem bipartisi- $(\min, \max, +)$, belum tentu memiliki nilai eigen struktural. Suatu eigenvalue "biasa" mungkin masih bisa ada. Dalam pembahasan diberikan kondisi untuk jenis eksistensi yang dapat dilihat sebagai perluasan kondisi bagi keberadaan nilai eigen dari interkoneksi yang terkecil (dalam hal dimensi) sistem bipartisi- $(\min, \max, +)$, disebut atom (lihat [31]). Dalam [4] algoritma yang dinamakan *algorithm power* dikenalkan untuk menghitung nilai eigen dan vektor eigen yang sesuai untuk sistem- $(\min, \max, +)$ yang dapat dipisah. Hal ini membuktikan bahwa dalam kondisi tertentu algoritma power dapat digunakan untuk menghitung eigen value dan eigen vektor yang sesuai untuk sistem- $(\max, +)$ dan suatu perluasan algoritma yang dinamakan *extended power algorithm* digunakan bila kondisi tidak dipenuhi (lihat [11]).

Dalam pembahasan berikutnya diberikan algoritma power yang lebih sederhana (lihat juga [7]) dengan beberapa kondisi ringan dapat digunakan untuk menghitung nilai eigen dan vektor eigen yang sesuai. Algoritma ini juga dapat ditemukan dalam [6]. Selanjutnya, diberikan algoritma power baru untuk menghitung eigenvalue dan vektor eigen yang sesuai dari sistem bipartisi- $(\min, \max, +)$.

5.1 Sistem Bipartisi- $(\min, \max, +)$

Sistem Bipartisi- $(\min, \max, +)$ merupakan gabungan dari sistem $(\max, +)$ dan sistem $(\min, +)$, selain itu dapat dikatakan juga sebagai gabungan dari aljabar max-plus dan aljabar min-plus. Misalkan terdapat sistem sebagai berikut

$$\begin{aligned} x_i(k+1) &= \max [a_{i,1} + y_1(k), a_{i,2} + y_2(k), \dots, a_{i,n} + y_n(k)] \\ y_j(k+1) &= \min [b_{j,1} + x_1(k), b_{j,2} + x_2(k), \dots, b_{j,m} + x_m(k)] \end{aligned} \quad (5.1)$$

dengan $k = 1, 2, \dots$, dimana $a_{i,j} \in \mathbb{R}_\varepsilon$ dan $b_{j,i} \in \mathbb{R}_{\varepsilon'} \forall i \in \underline{m}, j \in \underline{n}$. Dengan menggunakan operasi $\oplus, \oplus'$, dan $\otimes$ sistem diatas dapat dirubah menjadi

$$\begin{aligned} x_i(k+1) &= (a_{i,1} \otimes y_1(k)) \oplus \dots \oplus (a_{i,n} \otimes y_n(k)) \quad i \in \underline{m} \\ y_j(k+1) &= (b_{j,1} \otimes x_1(k)) \oplus' \dots \oplus' (b_{j,m} \otimes x_m(k)), \quad j \in \underline{n} \end{aligned} \quad (5.2)$$

kemudian dapat disederhanakan menjadi

$$\begin{aligned} x_i(k+1) &= \bigoplus_{j=1}^n (a_{i,j} \otimes y_j(k)), \quad i \in \underline{m} \\ y_j(k+1) &= \bigoplus_{i=1}^m (b_{j,i} \otimes x_i(k)), \quad j \in \underline{n} \end{aligned} \quad (5.3)$$

Selanjutnya dapat diubah menjadi

$$\left. \begin{aligned} \mathbf{x}(k+1) &= A \otimes \mathbf{y}(k) \\ \mathbf{y}(k+1) &= B \otimes' \mathbf{x}(k) \end{aligned} \right\} \quad (5.4)$$

dengan

$$\mathbf{x}(k) = \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_m(k) \end{bmatrix} \in \mathbb{R}_{\varepsilon'}^m, \quad \mathbf{y}(k) = \begin{bmatrix} y_1(k) \\ y_2(k) \\ \vdots \\ y_n(k) \end{bmatrix} \in \mathbb{R}_{\varepsilon}^n$$

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} \end{bmatrix} \in \mathbb{R}_{\varepsilon}^{m \times n}, \quad B = \begin{bmatrix} b_{1,1} & b_{1,2} & \dots & b_{1,m} \\ b_{2,1} & b_{2,2} & \dots & b_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n,1} & b_{n,2} & \dots & b_{n,m} \end{bmatrix} \in \mathbb{R}_{\varepsilon'}^{n \times m}$$

Pada [3], bipartisie (min,max,+)-system juga dapat dinotasikan sebagai berikut

$$\mathbf{z}(k+1) = \mathcal{M}(\mathbf{z}(k)), \quad k = 1, 2, \dots \quad (5.5)$$

dengan

$$\mathbf{z}(k) = \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{y}(k) \end{bmatrix}, \quad \mathcal{M}(\mathbf{z}(k)) = \mathcal{M} \left(\begin{bmatrix} \mathbf{x}(k) \\ \mathbf{y}(k) \end{bmatrix} \right) = \begin{bmatrix} A \otimes \mathbf{y}(k) \\ B \otimes' \mathbf{x}(k) \end{bmatrix}$$

Sistem dari (5.4) disebut sistem bipartisi (*bipartite system*) karena sistem tersebut dapat dinyatakan dalam graf bipartisi berarah (*bipartite directed graph*). Representasi graf dari sistem (5.4) memiliki $n + m$ titik yakni $x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_m$. Terdapat suatu garis dari verteks y_j ke x_i jika $a_{i,j} \neq \varepsilon$ dan nilai $a_{i,j}$ dijadikan sebagai bobot dari garis tersebut. Terdapat garis dari verteks x_i ke y_j jika $b_{j,i} \neq \top$ dan nilai $b_{j,i}$ dijadikan sebagai bobot dari garis tersebut. Titik $x_1, x_2, \dots, x_n$ disebut *titik maksimalisasi* sedangkan titik $y_1, y_2, \dots, y_m$ disebut *titik minimalisasi*. Bipartisi-(min, max, +) disebut regular sistem bipartite-(min, max, +) jika pada setiap baris matriks A dan B terdapat paling tidak satu elemen berhingga [3].

5.2 Sifat Tereduksi dalam Sistem Bipartisi- $(\min, \max, +)$

Sama halnya pada aljabar max-plus, dalam sistem bipartisi- $(\min, \max, +)$ juga terdapat sifat tereduksi. Namun sifat tereduksi dalam sistem bipartisi- $(\min, \max, +)$ melibatkan permutasi baris dan kolom dari matriks $A \in \mathbb{R}_\varepsilon^{n \times m}$ dan $B \in \mathbb{R}_{\varepsilon'}^{m \times n}$.

Misalkan W merupakan matriks $n \times m$, α merupakan permutasi dari $\underline{n}$ dan β merupakan permutasi dari $\underline{m}$. Matriks $W(\alpha, \beta)$ merupakan matriks yang diperoleh dengan mempermutasikan baris matriks W berdasarkan α dan mempermutasikan kolom matriks W berdasarkan β . Dengan demikian elemen baris ke- i dan kolom ke- j dari matriks $W(\alpha, \beta)$ adalah $W_{\alpha(i), \beta(j)}$.

Definisi 5.2.1 Misalkan terdapat sistem bipartisi- $(\min, \max, +)$ seperti pada sistem (5.4).

- Jika sistem (5.4) memiliki nilai karakteristik dan vektor karakteristik dalam sistem bipartisi- $(\min, \max, +)$ serta keberadaan nilai karakteristik dan vektor karakteristik tidak tergantung pada elemen-elemen hingga di matriks A dan B maka sistem (5.4) dikatakan secara struktural memiliki nilai karakteristik dan vektor karakteristik.
- Sistem (5.4) atau sepasang matriks (A, B) dikatakan tereduksi jika terdapat permutasi α dari $\underline{n}$ dan permutasi β dari $\underline{m}$ sedemikian hingga

$$A(\alpha, \beta) = \begin{bmatrix} A_{11} & A_{12} \\ \varepsilon & A_{22} \end{bmatrix}, B(\beta, \alpha) = \begin{bmatrix} B_{11} & \top \\ B_{21} & B_{22} \end{bmatrix}$$

dengan

- A_{ij} memiliki ukuran yang sama dengan B_{ji}^T
- Pada tiap baris A_{ij} terdapat paling tidak satu elemen terhingga.
- Pada tiap baris B_{ji} terdapat paling tidak satu elemen terhingga.
- ε merupakan matriks yang semua elemennya $-\infty$
- $\top$ merupakan matriks yang semua elemennya $+\infty$

Jika tidak terdapat permutasi α, β yang memenuhi maka sistem (5.4) atau pasangan matriks (A, B) dikatakan tidak tereduksi

Sifat tereduksi erat kaitannya terhadap keberadaan nilai karakteristik dan vektor karakteristik.

5.3 Nilai karakteristik dan Vektor Karakteristik dalam Sistem Bipartisi- $(\min, \max, +)$

Sistem bipartisi- $(\min, \max, +)$ pada sistem (5.4) dikatakan mempunyai nilai karakteristik dan vektor karakteristik jika terdapat $\lambda \in \mathbb{R}$ dan vektor $\mathbf{x} \in \mathbb{R}^m$, $\mathbf{y} \in \mathbb{R}^n$ yang memenuhi

$$\begin{bmatrix} A \otimes \mathbf{y} \\ B \otimes' \mathbf{x} \end{bmatrix} = \lambda \otimes \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \quad (5.6)$$

atau dapat dituliskan dengan

$$\begin{aligned} A \otimes \mathbf{y} &= \lambda \mathbf{1}_m + \mathbf{x} \\ B \otimes' \mathbf{x} &= \lambda \mathbf{1}_n + \mathbf{y} \end{aligned} \quad (5.7)$$

dengan $\mathbf{1}_k$ merupakan vektor $k \times 1$ yang semua elemennya bernilai 1.

Pada [3, 8] juga dijelaskan *Power Algorithm* untuk mendapatkan nilai karakteristik dan vektor karakteristik dalam sistem bipartisi- $(min, max, +)$.

1. Ambil sebarang vektor awal $\mathbf{z}(0)$.
2. Iterasi $\mathbf{z}(k) = \mathcal{M}(\mathbf{z}(k-1))$ untuk $k = 1, 2, \dots$ sampai terdapat bilangan bulat positif p, q dan bilangan riil c sehingga $\mathbf{z}(p) = c \otimes \mathbf{z}(q)$.
3. Definisikan sebagai nilai karakteristik $\lambda = \frac{c}{p-q}$.
4. Definisikan sebagai kandidat vektor karakteristik $\mathbf{v} = \bigoplus_{j=1}^{p-q} (\lambda^{\otimes(p-q-j)} \otimes \mathbf{z}(q+j-1))$
5. Jika $\mathcal{M}(\mathbf{v}) = \lambda \otimes \mathbf{v}$ maka sistem bipartisi- $(min, max, +)$ memiliki $\mathbf{v}$ merupakan vektor karakteristik dan algoritma berhenti. Jika $\mathcal{M}(\mathbf{v}) \neq \lambda \otimes \mathbf{v}$ maka algoritma dilanjutkan dengan langkah selanjutnya.
6. Ambil $\mathbf{z}(0) = \mathbf{v}$ sebagai vektor awal yang baru. Ulangi langkah ke-2 sampai terdapat bilangan bulat positif r yang memenuhi $\mathbf{z}(r+1) = \lambda \otimes \mathbf{z}(r)$. Maka $\mathbf{z}(r)$ merupakan vektor karakteristik dengan nilai karakteristik λ .

Terdapat suatu teorema yang mengaitkan sifat tereduksi dengan keberadaan nilai karakteristik dan vektor karakteristik.

Teorema 5.3.1 *Sistem bipartit- $(min, max, +)$ dengan pasangan matriks (A, B) secara struktural memiliki nilai karakteristik dan vektor karakteristik jika dan hanya jika pasangan matriks (A, B) tidak tereduksi*

Bukti dari Teorema 5.3.1 cukup panjang sehingga tidak dituliskan pada buku ini namun dapat dilihat pada [3].

Terdapat kasus khusus untuk pasangan matriks $(\overline{A}, \overline{B})$ tereduksi. Karena merupakan pasangan matriks tereduksi maka matriks $\overline{A}, \overline{B}$ dapat dinyatakan dengan

$$\overline{A} = \begin{bmatrix} \overline{A}_{11} & \overline{A}_{12} \\ \varepsilon & \overline{A}_{22} \end{bmatrix}, \overline{B} = \begin{bmatrix} \overline{B}_{11} & \top \\ \overline{B}_{21} & \overline{B}_{22} \end{bmatrix}$$

Kemudian vektor $\mathbf{x}(k), \mathbf{y}(k)$ diasumsikan dapat dipartisi menjadi

$$\mathbf{x}(k) = \begin{bmatrix} \mathbf{x}_1(k) \\ \mathbf{x}_2(k) \end{bmatrix}, \mathbf{y}(k) = \begin{bmatrix} \mathbf{y}_1(k) \\ \mathbf{y}_2(k) \end{bmatrix}$$

Sehingga sistem pada (5.4) dapat dinyatakan menjadi

$$\left. \begin{aligned} \begin{bmatrix} \mathbf{x}_1(k+1) \\ \mathbf{x}_2(k+1) \end{bmatrix} &= \begin{bmatrix} \overline{A}_{11} \otimes \mathbf{y}_1(k) & \oplus & \overline{A}_{12} \otimes \mathbf{y}_2(k) \\ & & \overline{A}_{22} \otimes \mathbf{y}_2(k) \end{bmatrix} \\ \begin{bmatrix} \mathbf{y}_1(k+1) \\ \mathbf{y}_2(k+1) \end{bmatrix} &= \begin{bmatrix} \overline{B}_{11} \otimes' \mathbf{x}_1(k) \\ \overline{B}_{21} \otimes' \mathbf{x}_1(k) & \oplus' & \overline{B}_{22} \otimes' \mathbf{x}_2(k) \end{bmatrix} \end{aligned} \right\} \quad (5.8)$$

Kemudian didefinisikan subsistem bipartisi- $(\min, \max, +)$ S_1 dan S_2 berturut-turut sebagai berikut

$$\begin{bmatrix} \mathbf{x}_1(k+1) \\ \mathbf{y}_1(k+1) \end{bmatrix} = \begin{bmatrix} \overline{A}_{11} \otimes \mathbf{y}_1(k) \\ \overline{B}_{11} \otimes' \mathbf{x}_1(k) \end{bmatrix} \quad (5.9)$$

dan

$$\begin{bmatrix} \mathbf{x}_2(k+1) \\ \mathbf{y}_2(k+1) \end{bmatrix} = \begin{bmatrix} \overline{A}_{22} \otimes \mathbf{y}_2(k) \\ \overline{B}_{22} \otimes' \mathbf{x}_2(k) \end{bmatrix} \quad (5.10)$$

Diandaikan kedua subsistem S_1 dan S_2 mempunyai nilai karakteristik berturut-turut λ_1 dan λ_2 dengan vektor karakteristik berhingga.

Teorema 5.3.2 *Misalkan S merupakan sistem pada (5.8) yang memiliki subsistem S_1 dan S_2 dengan nilai karakteristik berturut-turut adalah λ_1 dan λ_2 serta vektor karakteristiknya berhingga. Jika sistem S memiliki nilai karakteristik λ_0 serta vektor karakteristiknya berhingga, maka $\lambda_1 \leq \lambda_0 \leq \lambda_2$.*

Bukti dari Teorema 5.3.2 juga tidak dituliskan namun dapat dilihat di [3].

Daftar Pustaka

- [1] Bernd Heidergott, Geert Jan Olsder, and Jacob van der Woude. "Max Plus at Work Modelling and Analysis of Synchronized System: A Course on Max-Plus Algebra and Its Application", Princeton University Press, Princeton and Oxford, 2006.
- [2] Baccelli, F., G. Cohen, G.J. Olsder, and J.-P. Quadrat. "Synchronization and Linearity", John Wiley and Sons, New York, 1992. Buku ini dapat juga di download dari Web site <http://www-rocq.inria.fr/metalau/cohen/SED/bookonline.html>
- [3] Subiono. "On Classes of Min-Max-Plus Systems and Their Applications", PhD thesis, Delft University of Technology, The Netherlands, 2000. Buku ini dapat juga di download dari Web site [/http://www.its.ac.id/personal/material.php?id=Subiono](http://www.its.ac.id/personal/material.php?id=Subiono)
- [4] G.J. Olsder. "Eigenvalues of dynamic min-max systems". Journal of Discrete Event Dynamic Systems, 1:177-207, (1991).
- [5] G.J. Olsder, "On structural properties of min-max systems", Report of The Faculty of Technical Mathematics and Informatics no. 93-95, TU Delft, (1993).
- [6] L. Elsner, P. van den Driessche, "On the power method in max algebra", Linear Algebra and its Applications, 302-303:17-32, (1999).
- [7] Subiono, J.W. van der Woude, "Power algorithms for (max,+)-and bipartite (min,max,+)-systems", Report of the faculty of Technical Mathematics and Informatics no. 98-29, TU Delft, (1998), also accepted for publication in Discrete Event Dynamic System.
- [8] Subiono, and J.W. van der Woude. "Power Algorithms for (max,+)-and bipartite (min,max,+)-systems". Discrete Event Dynamic Systems, 10:369-389, 2000.
- [9] Woude J.W. van der, and Subiono. "Condition for the structural existence of an eigenvalue of a bipartite (min,max,+) system". Theoretical Computer Science, 293:13-24, 2003.

- [10] Soto y Koelemeijer. "On the Behaviour of Classes of Min-Max-Plus System". PhD thesis, Delft University of Technology, The Netherlands, 2003.
- [11] J.G. Braker, G.J. Olsder, "The power algorithm in max algebra", *Linear Algebra and its Applications*, 182:67-89, (1993).
- [12] J.G. Braker, "Algorithms and Applications in Timed Discrete Event Systems". PhD thesis, Delft University of Technology, The Netherlands, 1993.
- [13] J. Gunawardena, "Min-max function", *Journal of Discrete Event Dynamic Systems*, 4:377-407, (1994).
- [14] Ratna Novitasari. "Analisis Masalah Generator dari Possible dan Universal Eigenvector pada Matriks Interval dalam Aljabar Max-Plus". Thesis S2, Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Teknologi Sepuluh Nopember, Surabaya, 2009.
- [15] Nur Shofiana. "Analisis Kedinamikan Sistem pada Penjadwalan Flow Shop Menggunakan Aljabar Max-Plus". Thesis S2, Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Teknologi Sepuluh Nopember, Surabaya, 2009.
- [16] Winarni. "Penjadwalan Jalur Bus Dalam Kota dengan Aljabar Max-Plus". Tesis S2, Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Teknologi Sepuluh Nopember, Surabaya, 2009.
- [17] Himmatul Mursyidah. "Karakterisasi Nilai Eigen, Vektor Eogen, dan Eigenmode dari Matriks Tak-terduksi dan Terduksi dalam Aljabar Max-Plus". Tesis S2, Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Teknologi Sepuluh Nopember, Surabaya, 2014.
- [18] Nahlia Rakhmawati, Subiono and Subchan, Busway Schedule Planning In Surabaya Using Max-Plus Algebra. *Proceedings of National Graduate Seminar XII - ITS, Surabaya*, July 12, 2012.
- [19] Dyah Arum Anggraeni, Subchan and Subiono, Analysis of Aircraft Transit Timetable in Airport Using Max-Plus Algebra, *Journal POMITS Vol. 1, No. 1*, (2013), page: 1-5
- [20] Fatma Ayu Nuning F.A., Modeling and Scheduling Integrated System of Monorail and Train in Surabaya City Using Max-Plus Algebra, *Final Project, Mathematics Department Faculty of Mathematics and Science, Sepuluh Nopember Institute of Technology*, 2013.
- [21] Ema Enggar Wati, Subiono, and Subchan, Analysis Scheduling of Supply Chain Using Max-Plus Algebra (Case Study Terminal Bahan Bakar Minyak (TBBM) Manggis, Bali), *Journal POMITS Vol. 1, No.1*, (2014), page: 1-6

- [22] Widdya P. Sierliawati and Subiono, Rancangan dan analisis penjadwalan distribusi pada rantai pasok bahan bakar minyak menggunakan Petri Net dan Aljabar Max-Plus, *Prosiding Seminar Nasional Pascasarjana XIV-ITS*, ISBN 978-602-96565-7-2, Volume I, hal.73-80, 2014.
- [23] Shofiyatun Mufidah. "Model Rantai Pasok Menggunakan Petri Net dan Aljabar Max-Plus dengan Mempertimbangkan Prioritas Kapal Tanker". Tesis Jurusan Matematika FMIPA-ITS, 2015.
- [24] B. De Schutter. "Max-Algebraic system Theory For Discrete Event Systems" . PhD. Thesis, Katholieke Universiteit Leuven, Departement Elektrotechniek, 1996.
- [25] G.J. Olsder, Subiono. "On Large Scale Max-Plus Algebra Models in Railway Systems". Proceeding of IFAC conference on System Structure and Control, Nantes, France, pp.681-685, 1998.
- [26] Subiono, Dieky Adzkiya dan Kistosil Fahim. " Max-Plus Algebra Toolbox, ver. 2.0.0". Jurusan Matematika ITS, Surabaya, 2014. File toolbox bisa didownload di: http://atoms.scilab.org/toolboxes/maxplus_petri
- [27] Jörg Raisch. "Course Notes Discrete Event and Hybrid Systems". Technische Universität Berlin, 2009.
- [28] Maria H.Andersen, "Max-Plus Algebra: Properties and Applications". Master of Science in Mathematic Thesis Department of Mathematics, Laramie, WY, May 2002.
- [29] Cassandras, C.G. and Stephane Lafortune "Introduction to Discrete Event Systems", Second Edition Spriger, 2008.
- [30] M.V. Menon, "Some spectral properties of an operator associated with a pair of non-negative matrices", Trans.Amer.Math.Soc., 132:369-376, (1968).
- [31] Subiono, G.J. Olsder, "On bipartite min-max-plus systems", CD-ROM of proceeding European Control Conference (ECC), Brussels, Belgium, paper in session Th-E-K4, (1997).
- [32] Cochet-Terrasson, J., Cohen, G., Gaubert, S., Mc Gettrick, M., Quadrat, J.P., "Numerical Computation of Spectral Elements in $(\max, +)$ Algebra", IFAC Conference on System Structure and Control, Nantes, France, 1998.
- [33] Howard, R.A., "Dynamic Programming and Markov Processes", MIT Press/Wiley, New York, 1960.