



*J. Math. and Its Appl.*

ISSN: 1829-605X

Vol. 1, No. 2, Nov. 2004, 1-7

---

---

# Terapan Aljabar Max-Plus Pada Sistem Produksi Sederhana Serta Simulasinya Dengan Menggunakan Matlab

**Subiono**

Jurusan Matematika, FMIPA - I T S, Surabaya

email: subiono2003@telkom.net

## **Abstrak**

Dalam paper ini dibahas pengertian dari aljabar max-plus. Selanjutnya diberikan terapannya lewat contoh pada sistem proses sederhana. Kemudian dikaji perilaku dinamik sistem melalui suatu simulasi yang berkaitan dengan keadaan sistem dengan menggunakan Matlab.

Kata kunci: Aljabar max-plus, max-linier, keadaan sistem, SED.

## **1. Pendahuluan**

Banyak fenomena dari sistem manufaktur, jaringan telekomunikasi dan sistem transportasi bisa dijelaskan oleh Sistem Even Diskrit (SED). Hal ini tentunya sesuai dengan keadaan saat ini baik bagi dunia industri ataupun dunia akademik yang tertarik pada teknik untuk memodel, menganalisa dan mengontrol sistem-sistem yang kompleks seperti sistem manufaktur, jaringan telekomunikasi, sistem proses parallel, sistem kontrol trafik, sistem logistik dsb. Macam-macam sistem yang telah di sebutkan tadi adalah contoh-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 gambaran karakteristik dari SED adalah 'kedinamikannya' yaitu "event-driven" hal ini bertolak belakang dengan "time-driven". Di sini perilaku suatu SED lebih ditentukan oleh event dari pada waktu. Suatu event berkaitan dengan awal atau akhir dari suatu aktifitas. Bila kita tinjau 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 stokastik.

Suatu pengantar yang berkaitan dengan SED bisa didapat di [2]. Teknik yang paling luas digunakan untuk menganalisa SED adalah simulasi komputer. Kekurangan utama dari simulasi adalah sering tidak memberikan suatu pengertian yang nyata bagaimana parameter mengubah akibat penting sifat-sifat sistem, seperti kestabilan, kerobasan dan keoptimalan dari perilaku sistem. Oleh karena itu, metoda formal dibutuhkan sebagai alat untuk pemodelan, analisa dan kontrol bagi SED.

Pendekatan aljabar max-plus memungkinkan kita untuk 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 yang dapat digunakan untuk menganalisa perilaku suatu sistem yang ada. Pembahasan yang mendalam tentang aljabar max-plus bisa dilihat di [1].

Beberapa gambaran konkrit dari pemakaian aljabar max-plus adalah pada suatu jaringan sistem transportasi [7], pemakaian pada proses produksi perakitan [6]. Dalam konteks aljabar max-plus sistem model yang terjadi adalah linier dan non-linier pada aljabar biasa [5], kelinieran ini tentunya akan memudahkan dalam penganalisaan sistem yang dikaji.

Pada bagian berikutnya dalam paper ini akan dibahas suatu pemakaian dari aljabar max-plus, yaitu pada sistem produksi sederhana. Selanjutnya diberikan suatu asumsi yang realistik untuk mendapatkan suatu jadual yang reguler (teratur) dari sistem yang dikaji kemudian dilakukan simulasi dari sistem yang diberikan lewat keadaannya dengan menggunakan Matlab. Perangkat lunak Matlab yang dipakai menggunakan Max-Plus Algebra Toolbox for Matlab [4]. Pada bagian akhir juga diberikan suatu diskusi berupa beberapa pertanyaan dari hasil-hasil sistem yang dibahas. Namun sebelumnya diperkenalkan secara ringkas mengenai aljabar max-plus dan beberapa notasi yang digunakan.

## **2. Aljabar max-plus dan beberapa notasi yang terkait.**

Dalam bagian ini diberikan contoh klas tertentu dari SED yang bisa dimodelkan dengan menggunakan aljabar max-plus. Contoh ini adalah suatu sistem produksi sederhana yang juga bisa dijumpai di [6] dan [4]. Selanjutnya sistem ini dikaji perilaku dinamikanya melalui simulasi beberapa keadaan dari sistem. Selain dari pada itu tujuan utama dalam bagian ini adalah memberikan suatu ide SED macam apa yang bisa diuraikan dengan model waktu invarian max-linier.

Pertama, diberikan pengenalan singkat konsep dasar dari aljabar max-plus yang akan digunakan dalam bagian ini. Elemen dari aljabar max-plus adalah bilangan real dan  $\varepsilon = -\infty$

Operasi dasar dari aljabar max-plus adalah *maximum* (dinotasikan dengan simbol  $\oplus$ ) dan *tambah* (dinotasikan dengan simbol  $\otimes$ ). Dengan dua operasi tersebut diperoleh:

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

untuk setiap  $x, y \in \mathbb{R}_\varepsilon$ , dimana  $\mathbb{R}_\varepsilon = \mathbb{R} \cup \{\varepsilon\}$ . Catatan:  $x \oplus \varepsilon = x = \varepsilon \oplus x$  untuk semua  $x \in \mathbb{R}_\varepsilon$ . Operasi  $\oplus$  dan  $\otimes$  diperluas ke matriks sebagai berikut:

$$(A \oplus B)_{ij} = a_{ij} \oplus b_{ij} = \max\{a_{ij}, b_{ij}\}$$

dan

$$(A \otimes B)_{ij} = \bigoplus_k a_{ik} \otimes b_{kj} = \max_k\{a_{ik}, b_{kj}\}$$

untuk semua  $i, j$ .

Dipilihnya simbol  $\oplus$  dan  $\otimes$  dalam aljabar max-plus adalah ada analogi diantara  $\oplus$  dengan  $+$  dan  $\otimes$  dengan  $\times$ . Disini terlihat perkalian dua matriks mirip dengan perkalian dua matriks dalam aljabar biasa yaitu, perkalian dua matriks dalam aljabar biasa menggunakan operasi “kali” dan “tambah” sedangkan perkalian dua matriks dalam aljabar max-plus menggunakan operasi “tambah” dan “maksimum” masing-masing sebagai ganti dari operasi “kali” dan “tambah” yang digunakan pada perkalian dua matriks dalam aljabar biasa. Dengan menggunakan simbol-simbol tsb. diskripsi ruang keadaan sistem dapat ditulis dalam bentuk:

$$x(k+1) = A \otimes x(k) \oplus B \otimes u(k) \quad (1)$$

$$y(k) = C \otimes x(k). \quad (2)$$

Pengertian nilai-karakteristik dan vektor-karakteristik yang bersesuaian dari suatu matriks bujur sangkar sebagaimana dijumpai dalam aljabar linier biasa juga dijumpai dalam aljabar max-plus, yaitu bila mempunyai persamaan:

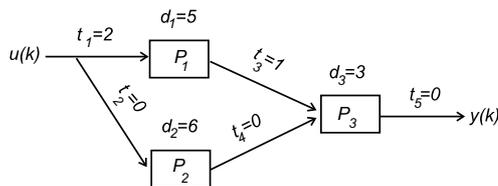
$$A \otimes x = \lambda \otimes x$$

dalam hal ini masing-masing vektor  $x$  dan  $\lambda$  dinamakan vektor-karakteristik dan nilai-karakteristik dari matriks  $A$ . Suatu algoritma untuk memperoleh vektor-karakteristik dan nilai karakteristik dari matriks bujur sangkar  $A$  bisa ditemui di [3].

Selanjutnya ditunjukkan bagaimana perilaku dinamik suatu contoh SED dari suatu sistem proses produksi sederhana sebagaimana yang dijumpai di [6].

**Contoh:** Sistem produksi sederhana.

Berikut ini dibahas sistem produksi yang diberikan oleh Gambar 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  $d_3 = 3$



Gambar 1: Sistem produksi sederhana

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. Kita 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:

$$x_1(k + 1) = \max\{x_1(k) + 5, u(k) + 2\} \quad (3)$$

untuk semua  $k \in \mathbb{N}_0$ , dimana  $\mathbb{N}_0$  menyatakan himpunan bilangan bulat tak negatif. Kondisi awal bahwa pemroses  $P_1$  kosong dan “idle” ini menunjukkan bahwa  $x_1(0) = \varepsilon$ . Oleh karena itu dari (3) 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:

$$x_2(k+1) = \max\{x_2(k) + 6, u(k)\} \quad (4)$$

$$x_3(k+1) = \max\{x_1(k) + 6, x_2(k) + 6, x_3(k) + 3\} \quad (5)$$

$$y(k) = x_3(k) + 3 \quad (6)$$

untuk semua  $k \in \mathbb{N}_0$  Kondisi bahwa semua buffer pada saat awal adalah kosong berke-  
naan 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) dan  
(4)-(6) 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

$$\begin{aligned} x(k+1) &= \begin{bmatrix} 5 & \varepsilon & \varepsilon \\ \varepsilon & 6 & \varepsilon \\ 6 & 6 & 3 \end{bmatrix} \otimes x(k) \oplus \begin{bmatrix} 2 \\ 0 \\ \varepsilon \end{bmatrix} \otimes u(k) \\ y(k) &= [\varepsilon \ \varepsilon \ 3] \otimes x(k) \end{aligned}$$

dimana  $x(k) = [x_1(k) \ x_2(k) \ x_3(k)]'$  dan notasi  $'$  menyatakan transpos. Catatan model  
didasar adalah model dari persamaan (1)-(2), 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 kekon-  
sumen ( $u(k) = y(k)$ ), maka evolusi dari keadaan sistem diberikan oleh:

$$\begin{aligned} x(k+1) &= A \otimes x(k) \oplus B \otimes u(k) \\ &= A \otimes x(k) \oplus B \otimes y(k) \\ &= A \otimes x(k) \oplus B \otimes C \otimes x(k) \\ &= \bar{A} \otimes x(k) \end{aligned}$$

dimana

$$\bar{A} = A \oplus B \otimes C.$$

Untuk menghitung  $\bar{A}$  bisa digunakan Aljabar Max-Plus Toolbox dalam [4] sebagai berikut: dalam Matlab ketik perintah

$\bar{A} = mp\_sum(A, mp\_multi(B, C))$ , 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 Matlab.

Selanjutnya dikaji perilaku dinamik dari sistem dengan mensimulasikan beberapa keadaan awal. Untuk keadaan awal  $x = [0 \ 0 \ 0]'$ , diperoleh evolusi sistem untuk  $k = 0, 1, 2, \dots, 10$ :

$$x = \begin{array}{cccccccccccc} & 0 & 5 & 11 & 17 & 23 & 29 & 35 & 41 & 47 & 53 & 59 \\ 0 & 0 & 6 & 12 & 18 & 24 & 30 & 36 & 42 & 48 & 54 & 60 \\ 0 & 6 & 12 & 18 & 24 & 30 & 36 & 42 & 48 & 54 & 60 & \end{array}$$

$$y = 3 \quad 9 \quad 15 \quad 21 \quad 27 \quad 33 \quad 39 \quad 45 \quad 51 \quad 57 \quad 63$$

Terlihat keadaan sistem mencapai periodik pada saat  $k = 1$  dengan periode sama dengan 6. Berikutnya dicoba keadaan awal  $x = [1 \ 2 \ 2]'$ , diperoleh evolusi sistemnya:

$$x = \begin{array}{cccccccccccc} & 1 & 7 & 13 & 19 & 25 & 31 & 37 & 43 & 49 & 55 & 61 \\ 2 & 2 & 8 & 14 & 20 & 26 & 32 & 38 & 44 & 50 & 56 & 62 \\ 2 & 8 & 14 & 20 & 26 & 32 & 38 & 44 & 50 & 56 & 62 & \end{array}$$

$$y = 5 \quad 11 \quad 17 \quad 23 \quad 29 \quad 35 \quad 41 \quad 47 \quad 53 \quad 59 \quad 65$$

Terlihat mulai awal keadaan sistem sudah periodik dengan periode sama dengan 6. Akhirnya, dicoba lagi dengan keadaan awal  $x = [20 \ 1 \ 23]'$ , didapat evolusi sistemnya:

$$x = \begin{array}{cccccccccccc} & 20 & 28 & 33 & 38 & 44 & 50 & 55 & 61 & 67 & 73 & 79 \\ 1 & 1 & 26 & 32 & 39 & 44 & 50 & 56 & 62 & 68 & 74 & 80 \\ 23 & 26 & 34 & 38 & 45 & 50 & 56 & 62 & 68 & 74 & 80 & \end{array}$$

$$y = 26 \quad 29 \quad 37 \quad 41 \quad 48 \quad 53 \quad 59 \quad 65 \quad 71 \quad 77 \quad 83$$

Terlihat keadaan sistem mencapai periodik pada saat  $k = 6$  dengan periode sama dengan 6. Cara memperoleh evolusi dari sistem dalam uraian diatas bisa dilakukan dalam Matlab dengan mengetik perintah sebagai berikut:  $x = mp\_system(\bar{A}, x0, 10, 0)$ , dimana  $x0$  adalah keadaan awal.

Dari beberapa keadaan awal yang diberikan, bisa disimpulkan bahwa keadaan awal  $x = [1 \ 2 \ 2]'$  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, kita akan memperoleh 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 bujur sangkar.

Perhatikan, bisa dicek bahwa vektor keadaan awal  $x = [1 \ 2 \ 2]'$  adalah suatu vektor-karakteristik dari matriks  $\bar{A}$ , sedangkan nilai-karakteristik dari  $\bar{A}$  sama dengan 6. Dalam hal ini, juga bisa digunakan perintah dalam Matlab dengan mengetik:  $[vkar, nkar] = mp\_egv1(\bar{A}, x0)$ , dimana  $x0$  sebarang vektor dengan ukuran yang sesuai sedangkan masing-masing  $vkar$  dan  $nkar$  adalah hasil perhitungan yang merupakan vektor-karakteristik dan nilai-karakteristik dari matriks  $\bar{A}$ .

Dari uraian yang telah disajikan, didapat suatu gambaran bagaimana menurunkan suatu model dari suatu sistem produksi sederhana melalui pengertian aljabar max-plus. Dari model ini diberikan beberapa simulasi dari keadaan sistem dan dengan menggunakan pengertian vektor-karakteristik dan nilai karakteristik diperoleh suatu evolusi keadaan sistem yang periodik dengan periode sama dengan nilai karakteristik tsb. Dari sini didapat suatu jadwal yang teratur dari proses produksi yang dikaji. Ada suatu hal yang menarik, yaitu 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 yang berkaitan dengan kestabilan dan mengetahui apa syarat-syaratnya sistem yang dikaji stabil.

## Daftar Pustaka

- [1] Baccelli, F., Cohen G., Olsder, G.J, and Quadrat, J.-P, *Synchronization and linearity: an algebra for discrete event systems*, Wiley. 1992.
- [2] Cassandras, C. and Lafortune, S., *Introduction to Discrete Event Systems*, Kluwer Academic Publisher, 1999.
- [3] Subiono and van der Woude, J., "Power Algorithms for (max,+)-and bipartite (min,max,+)-systems", *Discrete Event Dynamic Systems: Theory and Applications*, 10(4), 369-389, 2000.
- [4] Stańczyk, J., *Maxplus Algebra Toolbox for Matlab and GNU Octave*, Internet version: <http://ifatwww.et.uni-magdeburg.de/~stanczyk/mpa/>, ver.1.02, 2004.
- [5] Subiono, "Operator linier dalam aljabar max plus dan terapannya", *Proceeding SEMINAR NASIONAL MATEMATIKA: Peran Matematika Memasuki Milenium III*, ITS-Surabaya, 2000.
- [6] Subiono, "Terapan aljabar max-plus pada proses produksi-perakitan", *Seminar Nasional Matematika*, UGM-Jogya, 2001.
- [7] Subiono, "Pemodelan suatu sistem jaringan kereta dengan menggunakan jadwal keberangkatan kereta", *Natural* 6, 172-177, 2002.

