

## PETRI NET DAN MODEL ALJABAR MAX PLUS PADA SISTEM PELAYANAN PASIEN POLI UMUM DI PUSKESMAS XYZ

Mualifatul Munawaroh<sup>1\*</sup>, Subiono<sup>2</sup>

<sup>1,2</sup> Departemen Matematika, Fakultas Sains dan Analitika Data, Institut Teknologi Sepuluh Nopember  
\*Email: <sup>1</sup>mualifatul.19061@mhs.its.ac.id, <sup>2</sup>subiono2008@matematika.its.ac.id

### ABSTRAK

Kesehatan merupakan komponen penting bagi kehidupan manusia, yang dimana harus diperhatikan. Dalam hal ini, pemerintah telah menyediakan berbagai fasilitas untuk menunjang pelayanan kesehatan. Fasilitas tersebut seperti rumah sakit, puskesmas, klinik kesehatan dan sebagainya. Diharapkan setiap fasilitas tersebut dapat membantu melayani masyarakat dengan baik dan efektif. Salah satu contoh pelayanan kesehatan yaitu di Puskesmas, dimana disana terdapat antrian pelayanan yang sangat panjang dan terlalu lama dimana dapat merugikan banyak pihak yang membutuhkan pelayanan kesehatan. Sehingga banyak waktu terbuang selama proses menunggu antrian. Sehingga disini akan dibuat model aljabar Max Plus pada pelayanan pasien terutama pada poli Umum. Dalam sistem pelayanan pasien ada beberapa tahapan untuk membuat model aljabar Max Plus, dimulai dengan pengamatan dan pengambilan data tentang alur pelayanan di Puskesmas XYZ. Tahapan selanjutnya, membentuk model Petri Net alur pelayanan menggunakan *software* PIPE v4.3.0. Pasien akan dibatasi oleh dua kategori, yaitu pasien Umum dan pasien ASKES/BPJS. Tahapan terakhir membuat model aljabar Max Plus dari model Petri Net. Setelah diperoleh model aljabar Max Plus untuk pelayanan pasien Poli Umum di Puskesmas XYZ, kemudian dianalisis. Berdasarkan hasil analisis pada model aljabar Max Plus lamanya waktu dengan pelayanan selesai untuk pasien umum selain diperiksa di Lab membutuhkan waktu 30 menit, dan untuk pasien yang perlu diperiksa di Lab membutuhkan waktu 45 menit. Sedangkan untuk pasien ASKES/BPJS selain diperiksa di Lab membutuhkan waktu 25 menit, dan yang perlu diperiksa di Lab membutuhkan waktu 40 menit.

**Kata Kunci:** Petri Net; max plus; pelayanan Puskesmas.

### PENDAHULUAN

Kesehatan merupakan komponen penting bagi kehidupan manusia, yang dimana harus diperhatikan. Dalam hal ini, pemerintah telah menyediakan berbagai fasilitas untuk menunjang pelayanan kesehatan. Fasilitas tersebut seperti rumah sakit, puskesmas, klinik kesehatan dan sebagainya. Diharapkan setiap fasilitas tersebut dapat membantu melayani masyarakat dengan baik dan efektif.

Salah satu contoh pelayanan kesehatan yaitu di Puskesmas, dimana disana terdapat antrian pelayanan yang sangat panjang dan terlalu lama dimana dapat merugikan banyak pihak yang membutuhkan pelayanan kesehatan. Sehingga banyak waktu terbuang selama proses menunggu antrian. Sehingga disini akan dibuat model aljabar Max Plus pada pelayanan pasien terutama pada poli Umum.

Dalam pelayanan di Puskesmas, ada beberapa prosedur dan sistem perawatan yang harus dijalani oleh pasien sehingga pasien dapat diputuskan untuk mendapatkan resep, dirujuk ke Rumah Sakit atau pasien disarankan untuk di cek ke Laboratorium. Namun, terkadang tidak

dapat dipungkiri jika jumlah pasien datang lebih banyak daripada dengan kapasitas fasilitas pelayanan yang ada, sehingga terjadi yang namanya antrian. Untuk pelayanan pasien di poli Umum, biasanya dikategorikan sebagai pasien yang datang ke Puskesmas yang tidak darurat sehingga terbagi menjadi dua jenis, yaitu pasien ASKES/BPJS dan pasien Umum.

Dari latar belakang di atas, penulisan paper ini ada beberapa tahapan untuk dapat membuat model Aljabar Max Plus pelayanan pasien Poli Umum di Puskesmas XYZ. Tahapan pertama, dimulai dengan pengamatan dan pengambilan data tentang alur pelayanan di Puskesmas XYZ. Tahapan kedua, membentuk Petri Net alur pelayanan pasien di poli Umum dengan menggunakan *software* PIPE v4.3.0. Pasien dibatasi oleh dua kategori, yaitu pasien Umum dan ASKES/BPJS. Tahapan terakhir, yaitu dilakukan membentuk model aljabar Max Plus dari model Petri Net yang dibuat.

Adapun tujuan dari penulisan paper ini sendiri adalah untuk mendapatkan model Petri Net dan model aljabar Max Plus untuk pelayanan pasien Poli Umum di Puskesmas

XYZ, serta untuk mengetahui berapa lama waktu yang dibutuhkan untuk pelayanan pasien umum dan pasien ASKES/BPJS.

### METODE PENELITIAN

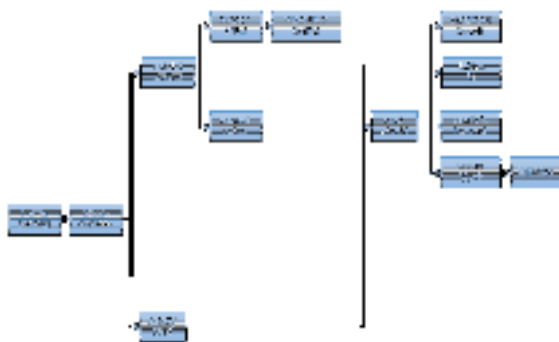
Metode yang digunakan adalah yang berkaitan dengan pemodelan menggunakan Petri Net, yaitu metode pustaka dan pengambilan data primer. Berikut langkah-langkah yang digunakan dalam penelitian ini:

1. Pengamatan dan pengambilan data tentang alur pelayanan di Puskesmas XYZ.
2. Membentuk Petri Net alur pelayanan pasien poli Umum di Puskesmas XYZ dengan menggunakan *software* PIPE v4.3.0. Pasien dibatasi oleh dua kategori, yaitu pasien Umum dan pasien ASKES/BPJS.
3. Membuat model aljabar Max Plus dari Petri Net yang telah dibuat.
4. Analisis dari model aljabar Max Plus untuk pelayanan pasien Poli Umum di Puskesmas XYZ.

### HASIL DAN PEMBAHASAN

#### 1. Pelayanan Pasien Poli umum di Puskesmas XYZ

Ketika berkunjung ke Puskesmas terdekat, tentu akan dijumpai dengan perbedaan alur pelayanan yang diikuti. Perbedaan pelayanan tergantung pada kasus yang bersifat darurat seperti, kecelakaan, serangan jantung, sesak napas, dll. Bila keadaan pasien tidak darurat, maka pasien harus mengikuti prosedur alur pelayanan standar Puskesmas. Untuk alur pelayanan pasien Poli Umum di Puskesmas XYZ dapat dilihat pada Gambar 1.

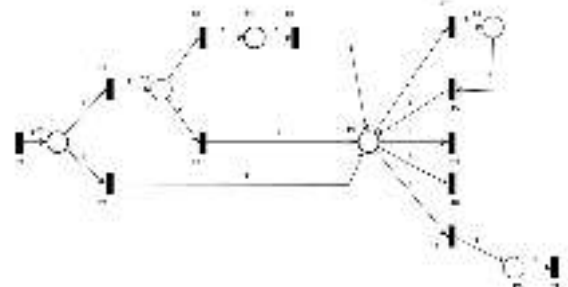


Gambar 1. Alur Pelayanan pasien Poli Umum di Puskesmas XYZ

#### 2. Model Petri Net dan Coverability Tree Pelayanan Pasien Poli Umum di Puskesmas XYZ

*Petri Net* merupakan Graf yang bipartisi. Jika terdapat himpunan  $V$  yang dipartisi menjadi dua himpunan bagian yaitu,  $P$  dan  $T$  masing-masing menyatakan *place* dan *transisi*. *Place* disini dapat berfungsi sebagai input atau output untuk suatu *transisi*. *Place* sebagai input menyatakan keadaan yang harus dipenuhi agar *transisi* dapat terjadi. Setelah *transisi* terjadi maka keadaan tersebut akan berubah. *Place* yang menyatakan keadaan tersebut adalah output dari *transisi*. [1]

Setiap *event* atau kejadian yang terjadi adalah berkaitan dengan *transisi*, sedangkan *place* merupakan kondisi yang harus dipenuhi agar *transisi* dapat terjadi. Model *Petri Net* dari sistem pelayanan pasien Poli Umum di Puskesmas XYZ dapat dilihat pada gambar 2.



Gambar 2. *Petri Net* Sistem Pelayanan Pasien Poli Umum di Puskesmas XYZ

Keterangan:

- $T_0$  = Pasien tiba di puskesmas untuk mengambil nomor antrian
- $T_1$  = Pemanggilan pasien umum
- $T_2$  = Pemanggilan pasien ASKES/BPJS menuju pelayanan tim medis
- $T_3$  = Pendataan pasien umum lama menuju pelayanan tim medis
- $T_4$  = Pemanggilan pasien umum baru menuju pendaftaran pasien
- $T_5$  = Pemanggilan pasien umum baru menuju pelayanan tim medis
- $T_6$  = Pasien dinyatakan sehat dan pulang
- $T_7$  = Pasien diperiksa laboratorium
- $T_8$  = Pasien diberikan resep
- $T_9$  = Pasien kembali ke pelayanan dokter
- $T_{10}$  = Pasien dirujuk ke Rumah Sakit
- $T_{11}$  = Pasien pulang
- $P_0$  = Pasien mendapat nomor antrian
- $P_1$  = Loket pasien umum
- $P_2$  = Pasien umum baru mendapat kartu keanggotaan

- $P_3$  = Pasien mendapat pelayanan tim medis
- $P_4$  = Pasien mendapat hasil pemeriksaan laboratorium
- $P_5$  = Pasien mendapat obat

Selanjutnya akan dilakukan analisa *Petri Net* pelayanan pasien di Puskesmas XYZ sebagai berikut:

- $Place = 6$   
Himpunan  $P = \{P_0, P_1, P_2, P_3, P_4, P_5\}$
- Transisi = 12  
Himpunan  $T = \{T_0, T_1, T_2, T_3, T_4, T_5, T_6, T_7, T_8, T_9, T_{10}, T_{11}\}$
- Maka bentuk matrik *incidence* dengan ordo  $6 \times 12$
- Bobot untuk masing-masing *Arc* adalah sebesar satu
- Dengan keadaan awal

$$m_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, m_1 = [0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$$

- Matrik *forward*, dari transisi ke *place* diperoleh:

$$A_f = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

- Matrik *backward*, dari *place* ke transisi diperoleh:

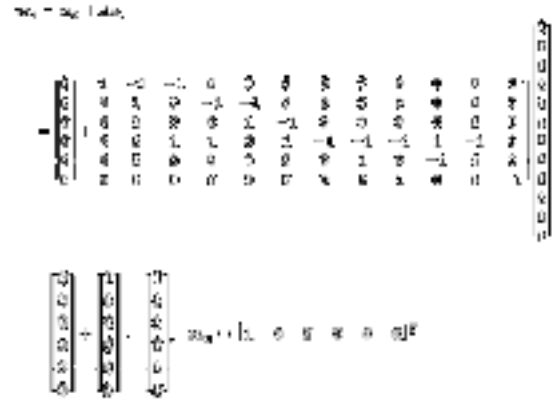
$$A_b = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

- Matrik keadaan  $A$  diperoleh dari *matrik forward* – *matrik backward*, matrik keadaan  $A$  merupakan matrik yang dapat mengubah keadaan awal dari alur *Petri Net* ketika dilakukan *fire* pada salah satu transisi yang *enable*. Sehingga diperoleh matrik  $A$  sebagai berikut:

$$A = \begin{bmatrix} 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & -1 & -1 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \end{bmatrix}$$

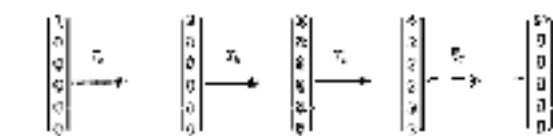
Selanjutnya dengan menggunakan persamaan  $m(p + 1) = m(p) + Ae$  dapat menentukan keadaan berikutnya: misal transisi  $T_0$  di *fire* yang berarti pasien ke  $k - 1$  ( $p = 0$ )

datang mengantri untuk mendapat nomor antrian, dengan kondisi awal  $m_0 = [0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$ , sehingga akan dihasilkan keadaan setelah transisi  $T_0$  di *fire*:



Hal tersebut merupakan keadaan yang baru transisi *enabled* adalah  $T_0, T_1$  dan  $T_2$ . Untuk menentukan keadaan yang baru, maka  $m_1$  menjadi keadaan awal untuk  $m_2$ , demikian seterusnya.

Untuk membangun *Coverability Tree* *Petri Net* pada Gambar 2, dapat dijelaskan bahwa keadaan awal yaitu  $m_0 = [0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$  dapat didefinisikan dengan *node root*. Keadaan tersebut menandakan bahwa *place*  $P_0$  belum terisi token sampai pada transisi  $T_0$  di *fire*. Sehingga keadaan transisi yang *enabled* yaitu  $T_0$ . Pemfirean transisi tersebut menyebabkan keadaan *Petri Net* berubah. Jika  $T_0$  di *fire* maka keadaan *Petri Net* akan berubah menjadi  $m_1 = [1 \ 0 \ 0 \ 0 \ 0 \ 0]^T$ . Setelah transisi  $T_0$  di *fire* maka transisi yang *enabled* adalah  $T_0, T_1$ , dan  $T_2$ , sehingga ada lima pilihan untuk pemfirean yaitu : (1)  $T_0$  di *fire*, (2)  $T_0$  tidak di *fire* dan  $T_1$  di *fire*, (3)  $T_0$  tidak di *fire* dan  $T_2$  di *fire*, (4)  $T_0$  di *fire* dan  $T_1$  di *fire*, (5)  $T_0$  di *fire* dan  $T_2$  di *fire*. Misal  $T_0$  di *fire*. Setelah pemfirean yang pertama maka jumlah token di  $P_0$  adalah satu. Apabila transisi  $T_0$  di *fire* berkali-kali maka jumlah token di  $P_0$  selalu bertambah. Sehingga *Coverability Treenya* adalah sebagai berikut:



Gambar 3. *Coverability Tree* *Petri Net* Alur Pelayanan Pasien Poli Umum Puskesmas XYZ ketika  $T_0$  di *fire* Berulang Kali

Karena disini akan dibahas proses membangun *Coverability Tree* *Petri Net* pada

gambar 2, maka hal yang perlu dilakukan adalah melakukan pemfirean terhadap masing-masing transisi dengan hubungan kondisi awal yang sesuai dengan gambar 3. Selanjutnya dengan menggunakan persamaan  $m(p + 1) = m(p) + Ae$  dapat menentukan keadaan berikutnya.

Misalkan transisi  $T_0$  di *fire* yang berarti pasien ke  $k - 1$  ( $p = 0$ ) datang mengantri untuk mendapat nomor antrian, dengan kondisi awal  $m_0 = [0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$ , sehingga akan dihasilkan keadaan setelah transisi  $T_0$  di *fire*:

$$m_1 = m_0 + Ae_1$$

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad m_1 = [1 \ 0 \ 0 \ 0 \ 0 \ 0]^T$$

Setelah transisi  $T_0$  di *fire* terdapat transisi yang *enabled* yaitu  $T_1$  dan  $T_2$ . Misalkan transisi  $T_1$  di *fire* yang berarti pemanggilan pasien umum, didapatkan keadaan:

$$m_2 = m_1 + Ae_2$$

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad m_2 = [2 \ 0 \ 0 \ 0 \ 0 \ 0]^T$$

Misalkan transisi  $T_2$  di *fire* yang berarti pemanggilan pasien ASKES/BPJS, didapatkan keadaan:

$$m_3 = m_1 + Ae_3$$

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad m_3 = [1 \ 1 \ 0 \ 0 \ 0 \ 0]^T$$

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad m_4 = [1 \ 0 \ 0 \ 0 \ 0 \ 0]^T$$

Misalkan transisi  $T_3$  di *fire* yang berarti pendataan pasien umum lama menuju pelayanan tim medis, didapatkan keadaan:

$$m_4 = m_2 + Ae_4$$

$$\begin{bmatrix} 2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad m_4 = [2 \ 1 \ 0 \ 0 \ 0 \ 0]^T$$

Misalkan transisi  $T_4$  di *fire* yang berarti pemanggilan pasien umum baru menuju pendaftaran pasien, didapatkan keadaan:

$$m_5 = m_2 + Ae_5$$

$$\begin{bmatrix} 2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad m_5 = [2 \ 0 \ 1 \ 0 \ 0 \ 0]^T$$

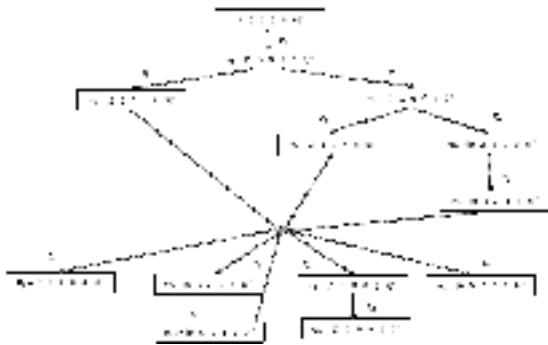
Misalkan transisi  $T_5$  di *fire* yang berarti pemanggilan pasien umum baru menuju pelayanan tim medis, didapatkan keadaan:

$$m_6 = m_5 + Ae_6$$

$$\begin{bmatrix} 2 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 3 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad m_6 = [3 \ 0 \ 1 \ 0 \ 0 \ 0]^T$$

Misalkan transisi  $T_6$  di *fire* yang berarti pasien dinyatakan sehat dan diperbolehkan pulang, dengan mempertimbangkan bahwa nilai  $m_4$  dan  $m_6$  nilainya adalah sama dengan  $m_3$ ,

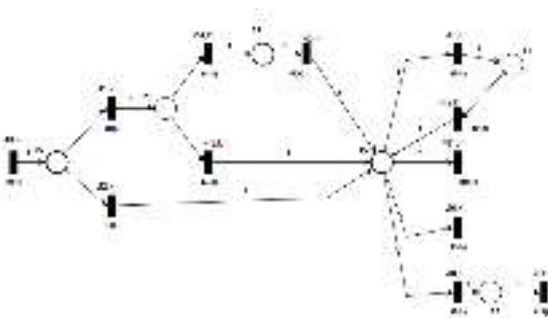




Gambar 4. *Coverability Tree Model Jaringan Petri Net Pelayanan Pasien Poli Umum Puskesmas XYZ*

### 3. Model Aljabar *Max Plus* Pelayan Pasien Poli umum di Puskesmas XYZ yang Dikaitkan dengan Waktu

*Petri Net* dengan waktu terdapat dua peubah yang digunakan dalam proses pemodelan aljabar *Max Plus*, yaitu peubah waktu dan peubah yang menunjukkan lamanya waktu [3]. Sehingga peubah waktu yang dibutuhkan dalam pemodelan aljabar *Max Plus* pada pelayanan pasien Poli Umum di Puskesmas XYZ yang dikaitkan dengan waktu dapat disajikan dalam Gambar 5.



Gambar 5. *Petri Net dengan Waktu Pelayanan Pasien Poli Umum di Puskesmas XYZ*

Untuk mengetahui lama waktu pelayanan dan waktu berakhirnya pelayanan pasien poli Umum, maka untuk *Petri Net* dari sistem pelayanan pasien poli Umum dapat diselesaikan dengan menggunakan model aljabar *Max Plus*. Namun, terlebih dahulu akan diberikan definisi variabel-variabel yang digunakan untuk memodelkan sistem pelayanan tersebut. Berikut variabel yang digunakan untuk menunjukkan waktu:

$t_0(k)$  = Waktu pasien tiba di Puskesmas untuk mengambil nomor antrian pada saat ke-  $k$

$t_1(k)$  = Waktu pemanggilan pasien umum pada saat ke-  $k$

$t_2(k)$  = Waktu pemanggilan pasien ASKES/ BPJS menuju pelayanan tim medis pada saat ke-  $k$

$t_3(k)$  = Waktu pendataan pasien umum lama menuju pelayanan tim medis pada saat ke-  $k$

$t_4(k)$  = Waktu pemanggilan pasien umum baru menuju pendaftaran pasien pada saat ke-  $k$

$t_5(k)$  = Waktu pemanggilan pasien umum baru menuju pelayanan tim medis pada saat ke-  $k$

$t_6(k)$  = Waktu pasien dinyatakan sehat dan pulang pada saat ke-  $k$

$t_7(k)$  = Waktu pasien diperiksa laboratorium pada saat ke-  $k$

$t_8(k)$  = Waktu pasien diberikan resep pada saat ke-  $k$

$t_9(k)$  = Waktu pasien kembali ke pelayanan dokter pada saat ke-  $k$

$t_{10}(k)$  = Waktu pasien dirujuk ke Rumah Sakit pada saat ke-  $k$

$t_{11}(k)$  = Waktu pasien pulang pada saat ke-  $k$ .

Adapun variabel-variabel yang menunjukkan lama waktu dalam menjalankan setiap proses, yaitu:

$v_{t_0,k}$  = Lamanya proses pasien datang untuk mengambil nomor antrian

$v_{t_1,k}$  = Lamanya proses pemanggilan pasien umum

$v_{t_2,k}$  = Lamanya proses pemanggilan pasien ASKES/BPJS menuju pelayanan tim medis

$v_{t_3,k}$  = Lamanya proses pendataan pasien umum lama menuju pelayanan tim medis

$v_{t_4,k}$  = Lamanya proses pemanggilan pasien umum baru menuju pendaftaran pasien

$v_{t_5,k}$  = Lamanya proses pemanggilan pasien umum baru menuju pelayanan tim medis

$v_{t_6,k}$  = Lamanya proses pasien dinyatakan sehat dan pulang

$v_{t_7,k}$  = Lamanya proses pasien diperiksa laboratorium

$v_{t_8,k}$  = Lamanya proses pasien diberikan resep

$v_{t_9,k}$  = Lamanya proses pasien kembali ke pelayanan dokter

$v_{t_{10},k}$  = Lamanya proses pasien dirujuk ke Rumah Sakit

$v_{t_{11},k}$  = Lamanya proses pasien pulang/keluar dari Puskesmas

a. Proses pelayanan pasien Umum

$$t_0(k) = v_{t_0,k} \otimes t_0(k-1)$$

$$t_1(k) = v_{t_1,k} \otimes t_0(k) \\ = v_{t_1,k} \otimes v_{t_0,k} \otimes t_0(k-1)$$

$$t_3(k) = v_{t_3,k} \otimes t_1(k) \\ = v_{t_3,k} \otimes v_{t_1,k} \otimes v_{t_0,k} \otimes t_0(k-1)$$

$$t_4(k) = v_{t_4,k} \otimes t_1(k) \\ = v_{t_4,k} \otimes v_{t_1,k} \otimes v_{t_0,k} \otimes t_0(k-1)$$

$t_5(k)$  memiliki model aljabar *Max Plus* yang sama dengan  $t_3(k)$  karena memiliki *coverability tree* yang sama.

$$t_6(k) = v_{t_6,k} \otimes t_3(k) \oplus t_9(k-1) \\ = v_{t_6,k} \otimes (v_{t_3,k} \otimes v_{t_1,k} \otimes v_{t_0,k} \otimes t_0(k-1)) \\ \oplus t_9(k-1)$$

$$= v_{t_6,k} \otimes v_{t_3,k} \otimes v_{t_1,k} \otimes v_{t_0,k} \otimes t_0(k-1) \\ \oplus v_{t_6,k} \otimes t_9(k-1)$$

$$t_7(k) = v_{t_7,k} \otimes t_3(k) \oplus t_9(k-1) \\ = v_{t_7,k} \otimes (v_{t_3,k} \otimes v_{t_1,k} \otimes v_{t_0,k} \otimes t_0(k-1)) \\ \oplus t_9(k-1)$$

$$= v_{t_7,k} \otimes v_{t_3,k} \otimes v_{t_1,k} \otimes v_{t_0,k} \otimes t_0(k-1) \\ \oplus v_{t_7,k} \otimes t_9(k-1)$$

$$t_8(k) = v_{t_8,k} \otimes t_3(k) \oplus t_9(k-1) \\ = v_{t_8,k} \otimes (v_{t_3,k} \otimes v_{t_1,k} \otimes v_{t_0,k} \\ \otimes t_0(k-1) \oplus t_9(k-1))$$

$$= v_{t_8,k} \otimes v_{t_3,k} \otimes v_{t_1,k} \\ \otimes v_{t_0,k} \otimes t_0(k-1) \oplus v_{t_8,k} \otimes t_9(k-1)$$

$$t_9(k) = v_{t_9,k} \oplus t_7(k-1) \\ = v_{t_9,k} \otimes v_{t_7,k} \otimes v_{t_3,k} \otimes (v_{t_1,k} \otimes v_{t_0,k} \\ \otimes t_0(k-1)) \oplus v_{t_7,k} \otimes t_9(k-1)$$

$$= v_{t_9,k} \otimes v_{t_7,k} \otimes v_{t_3,k} \otimes v_{t_1,k} \otimes v_{t_0,k} \\ \otimes t_0(k-1) \oplus v_{t_9,k} \otimes v_{t_7,k} \otimes t_9(k-1) \\ = v_{t_9,k} \oplus v_{t_9,k} \otimes t_7(k-1)$$

$t_{10}(k)$  dan  $t_{11}(k)$  memiliki model aljabar *Max Plus* yang sama dengan  $t_6(k)$  karena memiliki *coverability tree* yang sama.

Sehingga diperoleh jadwal lamanya pelayanan pasien umum:

$$\begin{bmatrix} t_0(k) \\ t_6(k) \\ t_9(k) \end{bmatrix} = \begin{bmatrix} v_{t_0,k} & a & a \\ b & a & v_{t_6,k} \\ c & a & d \end{bmatrix} \otimes \begin{bmatrix} t_0(k-1) \\ t_6(k-1) \\ t_9(k-1) \end{bmatrix}$$

Dengan keterangan

Nilai  $a = \varepsilon$ , supaya:

$$v_{t_0,k} \otimes t_0(k-1) \oplus a \otimes t_6(k-1) \\ \oplus a \otimes t_9(k-1) \\ = v_{t_0,k} \otimes t_0(k-1)$$

Nilai  $b$  didefinisikan sebagai:

$$b = v_{t_6,k} \otimes v_{t_3,k} \otimes v_{t_1,k} \otimes v_{t_0,k}$$

Nilai  $c$  didefinisikan sebagai:

$$c = v_{t_9,k} \otimes v_{t_7,k} \otimes v_{t_3,k} \otimes v_{t_1,k} \otimes v_{t_0,k}$$

Nilai  $d$  didefinisikan sebagai:

$$d = v_{t_9,k} \otimes v_{t_7,k}$$

Untuk selanjutnya, misalkan diberi lama waktu proses (dalam menit) tiap tahap sebagai berikut:

$$\begin{array}{ll} v_{t_0,k} = 3 & v_{t_6,k} = 20 \\ v_{t_1,k} = 5 & v_{t_7,k} = 20 \\ v_{t_2,k} = 5 & v_{t_8,k} = 20 \\ v_{t_3,k} = 5 & v_{t_9,k} = 15 \\ v_{t_4,k} = 10 & v_{t_{10},k} = 20 \\ v_{t_5,k} = 5 & v_{t_{11},k} = 5 \end{array}$$

Maka

$$b = v_{t_6,k} \otimes v_{t_3,k} \otimes v_{t_1,k} \otimes v_{t_0,k} \\ = 20 \otimes 5 \otimes 5 \otimes 3 = 33 \text{ menit}$$

$$c = v_{t_9,k} \otimes v_{t_7,k} \otimes v_{t_3,k} \otimes v_{t_1,k} \otimes v_{t_0,k} \\ = 15 \otimes 20 \otimes 5 \otimes 5 \otimes 3 = 48 \text{ menit}$$

$$d = v_{t_9,k} \otimes v_{t_7,k} \\ = 15 \otimes 20 = 35 \text{ menit}$$

Untuk keadaan awal

$$\begin{bmatrix} t_0(0) \\ t_6(0) \\ t_9(0) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

didapatkan

$$\begin{bmatrix} t_0(1) \\ t_6(1) \\ t_9(1) \end{bmatrix} = \begin{bmatrix} 3 & \varepsilon & \varepsilon \\ 33 & \varepsilon & 20 \\ 48 & \varepsilon & 35 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 3 \\ 33 \\ 48 \end{bmatrix}$$

$$\begin{bmatrix} t_0(1) \\ t_6(1) \\ t_9(1) \end{bmatrix} = \begin{bmatrix} 3 \\ 33 \\ 48 \end{bmatrix}$$

dan hasil yang diperoleh sama dengan menggunakan program *software* Scilab, sehingga dapat ditulis

$$\begin{bmatrix} 3 & \varepsilon & \varepsilon \\ 33 & \varepsilon & \varepsilon \\ 48 & \varepsilon & \varepsilon \end{bmatrix}$$

Kemudian dengan menggunakan program *software* Scilab diperoleh nilai eigen  $\lambda = 3$  dan vektor eigen  $v = [3, 33, 48]^T$ .

Maka dapat diketahui proses pelayanan pasien umum di poli umum mulai dari kedatangan sampai dengan pelayanan selesai untuk pasien selain diperiksa di Lab membutuhkan waktu  $t_6(1) - t_0(1) = 33 - 3 = 30$  menit. Sedangkan untuk pasien yang perlu diperiksa di Lab membutuhkan waktu  $t_9(1) - t_0(1) = 48 - 3 = 45$  menit.

b. Proses pelayanan pasien ASKES/BPJS

$$t_0(k) = v_{t_0,k} \otimes t_0(k-1)$$

$$t_2(k) = v_{t_2,k} \otimes t_0(k)$$

$$\begin{aligned}
 &= v_{t_2,k} \otimes v_{t_0,k} \otimes t_0(k-1) \\
 t_6(k) &= v_{t_6,k} \otimes t_2(k) \oplus t_9(k-1) \\
 &= v_{t_6,k} \otimes (v_{t_2,k} \otimes v_{t_0,k} \otimes t_0(k-1) \oplus t_9(k-1)) \\
 &= v_{t_6,k} \otimes v_{t_2,k} \otimes v_{t_0,k} \otimes t_0(k-1) \oplus v_{t_6,k} \otimes t_9(k-1) \\
 t_7(k) &= v_{t_7,k} \otimes t_2(k) \oplus t_9(k-1) \\
 &= v_{t_7,k} \otimes (v_{t_2,k} \otimes v_{t_0,k} \otimes t_0(k-1) \oplus t_9(k-1)) \\
 &= v_{t_7,k} \otimes v_{t_2,k} \otimes v_{t_0,k} \otimes t_0(k-1) \oplus v_{t_7,k} \otimes t_9(k-1) \\
 t_8(k) &= v_{t_8,k} \otimes t_2(k) \oplus t_9(k-1) \\
 &= v_{t_8,k} \otimes (v_{t_2,k} \otimes v_{t_0,k} \otimes t_0(k-1) \oplus t_9(k-1)) \\
 &= v_{t_8,k} \otimes v_{t_2,k} \otimes v_{t_0,k} \otimes t_0(k-1) \oplus v_{t_8,k} \otimes t_9(k-1) \\
 t_9(k) &= v_{t_9,k} \otimes t_7(k) \\
 &= v_{t_9,k} \otimes (v_{t_7,k} \otimes v_{t_2,k} \otimes v_{t_0,k} \otimes t_0(k-1) \oplus v_{t_7,k} \otimes t_9(k-1)) \\
 &= v_{t_9,k} \otimes v_{t_7,k} \otimes v_{t_2,k} \otimes v_{t_0,k} \otimes t_0(k-1) \oplus v_{t_9,k} \otimes v_{t_7,k} \otimes t_9(k-1)
 \end{aligned}$$

$t_{10}(k)$  dan  $t_{11}(k)$  memiliki model aljabar *Max Plus* yang sama dengan  $t_6(k)$  karena memiliki *coverability tree* yang sama.

Sehingga diperoleh jadwal lamanya pelayanan pasien ASKES/BPJS:

$$\begin{bmatrix} t_0(k) \\ t_6(k) \\ t_9(k) \end{bmatrix} = \begin{bmatrix} v_{t_0,k} & a & a \\ b & a & v_{t_6,k} \\ c & a & d \end{bmatrix} \otimes \begin{bmatrix} t_0(k-1) \\ t_6(k-1) \\ t_9(k-1) \end{bmatrix}$$

Dengan keterangan

Nilai  $a = \varepsilon$ , supaya:

$$v_{t_0,k} \otimes t_0(k-1) \oplus a \otimes t_6(k-1) \oplus a \otimes t_9(k-1) = v_{t_0,k} \otimes t_0(k-1)$$

Nilai  $b$  didefinisikan sebagai:

$$b = v_{t_6,k} \otimes v_{t_2,k} \otimes v_{t_0,k}$$

Nilai  $c$  didefinisikan sebagai:

$$c = v_{t_9,k} \otimes v_{t_7,k} \otimes v_{t_2,k} \otimes v_{t_0,k}$$

Nilai  $d$  didefinisikan sebagai:

$$d = v_{t_9,k} \otimes v_{t_7,k}$$

Selanjutnya, misalkan diberikan lama waktu proses (dalam menit) tiap tahap sebagai berikut:

$$\begin{array}{ll}
 v_{t_0,k} = 3 & v_{t_6,k} = 20 \\
 v_{t_1,k} = 5 & v_{t_7,k} = 20 \\
 v_{t_2,k} = 5 & v_{t_8,k} = 20 \\
 v_{t_3,k} = 5 & v_{t_9,k} = 15 \\
 v_{t_4,k} = 10 & v_{t_{10},k} = 20 \\
 v_{t_5,k} = 5 & v_{t_{11},k} = 5
 \end{array}$$

Maka

$$\begin{aligned}
 b &= v_{t_6,k} \otimes v_{t_2,k} \otimes v_{t_0,k} \\
 &= 20 \otimes 5 \otimes 3 = 28 \text{ menit} \\
 c &= v_{t_9,k} \otimes v_{t_7,k} \otimes v_{t_2,k} \otimes v_{t_0,k} \\
 &= 15 \otimes 20 \otimes 5 \otimes 3 = 43 \text{ menit} \\
 d &= v_{t_9,k} \otimes v_{t_7,k} \\
 &= 15 \otimes 20 = 35 \text{ menit}
 \end{aligned}$$

Untuk keadaan awal

$$\begin{bmatrix} t_0(0) \\ t_6(0) \\ t_9(0) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

didapatkan

$$\begin{bmatrix} t_0(1) \\ t_6(1) \\ t_9(1) \end{bmatrix} = \begin{bmatrix} 3 & \varepsilon & \varepsilon \\ 28 & \varepsilon & 20 \\ 43 & \varepsilon & 35 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 3 \\ 28 \\ 43 \end{bmatrix}$$

$$\begin{bmatrix} t_0(1) \\ t_6(1) \\ t_9(1) \end{bmatrix} = \begin{bmatrix} 3 \\ 28 \\ 43 \end{bmatrix}$$

dan hasil yang diperoleh sama dengan menggunakan program *software* Scilab, sehingga dapat ditulis

$$\begin{bmatrix} 3 & \varepsilon & \varepsilon \\ 28 & \varepsilon & \varepsilon \\ 43 & \varepsilon & \varepsilon \end{bmatrix}$$

Kemudian dengan menggunakan program *software* Scilab diperoleh nilai eigen  $\lambda = 3$  dan vector eigen  $v = [3, 28, 43]^T$

Maka dapat diketahui proses pelayanan pasien ASKES/BPJS di poli umum mulai dari kedatangan sampai dengan pelayanan selesai untuk pasien selain diperiksa di Lab membutuhkan waktu  $t_6(1) - t_0(1) = 28 - 3 = 25$  menit. Sedangkan untuk pasien yang perlu diperiksa di Lab membutuhkan waktu  $t_9(1) - t_0(1) = 43 - 3 = 40$  menit.



## KESIMPULAN

Berdasarkan latar belakang dan hasil pembahasan di atas, maka dapat disimpulkan bahwa diperoleh penggunaan Petri Net sangat baik untuk menganalisis sistem pelayanan pasien poli Umum di puskesmas XYZ baik pasien Umum maupun pasien ASKES/BPJS. Diperoleh model matematika dari pelayanan pasien poli Umum dalam bentuk Petri Net dan model aljabar Max Plus dengan waktu. Analisa yang diperoleh dari alur Petri Net pelayanan pasien poli Umum di puskesmas XYZ dengan menggunakan model aljabar Max Plus untuk setiap satu kali proses pelayanan pasien dari datang sampai dengan pelayanan selesai, untuk pasien umum yang tidak diperiksa di Lab membutuhkan waktu 30 menit, dan pasien umum yang perlu untuk diperiksa di Lab membutuhkan waktu 45 menit. Sedangkan untuk pasien ASKES/BPJS yang tidak diperiksa di Lab membutuhkan waktu 25 menit, dan pasien ASKES/BPJS yang perlu untuk diperiksa di Lab membutuhkan waktu 40 menit. Untuk selanjutnya bagi pembaca yang ingin mengembangkan paper ini, dapat mengembangkannya dengan membuat Petri Net secara detail mengenai pelayanan pasien di puskesmas, yang dimana melibatkan sembilan poli yang ada di puskesmas, serta membedakan pasien darurat dan tidak darurat.

## DAFTAR PUSTAKA

- [1] Cassandras, C.G. & Lafortune, S. 2008. *Introduction to Discrete Event System*. New York: Second edition, Springer Science+Business Media.
- [2] Subiono. 2020. *Aljabar Min-Max Plus dan Terapannya*. Surabaya: Jurusan Matematika Institut Teknologi Sepuluh Nopember.
- [3] Mustofani, D. & Afif, A. 2018. Model Antrian Pelayanan Farmasi Menggunakan Petri Net Dan Aljabar Max-Plus. *Jurnal Matematika dan Pendidikan Matematika* Vol. 3 No. 1.
- [4] Cahyani, M. 2015. Penggunaan Aljabar Max Plus dan petri Net untuk Etimasi Lamanya Sistem Pelayanan dan Kerja Karyawan Pemasangan Instalasi di PDAM. Surabaya: *Tesis* Jurusan Matematika Institut Tenologi Sepuluh Nopember.

- [5] Windiarni, R. 2016. *Pembentukan Model Petri Net dan Aljabar Max Plus pada Alur Pelayanan Nasabah Bank*. Yogyakarta: Skripsi Jurusan matematika UIN Sunan Kalijaga.