

**PEMODELAN SIR UNTUK PENYEBARAN PENYAKIT PERTUSIS
DENGAN VAKSINASI PADA POPULASI MANUSIA KONSTAN**



Disusun oleh :
Kelompok 8

Nuansa Cahaya Muhammad
Fajar Wahyu Pratama

5002211015
5002211042

Dosen Pengampu :
Prof. Dr. Dra. Mardlijah, MT
Dr. Didik Khusnul Arif, S.Si, M.Si

**DEPARTEMEN MATEMATIKA
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2023**

1 Pendahuluan

Pertusis (batuk rejan) disebut juga whooping cough, tussis quinta, violent cough, dan di Cina disebut batuk seratus hari. Pertusis adalah penyakit yang sangat menular menginfeksi kekebalan pada manusia dan disebabkan oleh bakteri **Bordetella pertussis** [1]. Penyakit pertusis ini di tandai dengan batuk yang berlangsung **28 hari sampai dengan 100 hari**, individu yang sangat rentan adalah bayi dan anak-anak muda [2]. Batuk pertusis memiliki ciri-ciri khas yang dapat dikenali, yaitu:

1. Rentetan batuk keras yang terjadi secara terus-menerus, dan sering diawali dengan bunyi tarikan napas panjang melengking khas yang terdengar mirip “whoop”
2. Batuk kering terus-menerus yang berbunyi keras disertai suara mengi (napas bunyi ngik-ngik atau “whoop”)
3. Wajah tampak memerah atau kebiruan saat batuk
4. Kelelahan ekstrim
5. Dada terasa sakit saat batuk sehingga sulit untuk bernapas

Gejala batuk pertusis dapat terbagi menjadi tiga tahapan terutama pada bayi dan anak-anak, yaitu **tahap catarrhal, tahap paroksismal, dan tahap konvalesen**. **Tahap catarrhal** mirip dengan pilek biasa, dengan gejala seperti hidung tersumbat, bersin, batuk ringan, dan demam rendah. Tahap ini berlangsung selama satu hingga dua minggu. **Tahap paroksismal** adalah tahap di mana batuk menjadi lebih parah dan berlangsung selama satu hingga enam minggu. Pada tahap ini, bayi akan mengalami serangan batuk kering yang parah dan berulang. **Tahap konvalesen** merupakan tahap pemulihan. Gejala batuk perlahan-lahan mulai membaik, meskipun batuk ringan masih mungkin terjadi selama beberapa minggu.

Salah satu cara untuk mengurangi jumlah kasus pertusis ini adalah dengan pemberian vaksin. Vaksin yang digunakan adalah DPT (Difteri, Pertussis, Tetanus), vaksin ditujukan untuk menghasilkan sistem pertahanan tubuh terhadap penyakit ini [3].

Penelitian yang berhubungan dengan pemodelan matematika tentang penyakit pertusis sudah dilakukan oleh beberapa orang. Menurut penelitian Hu dkk. (2012) yaitu pada penelitiannya mempertimbangkan kontinu time model jenis SIRS epidemi dijelaskan oleh persamaan diferensial. Pemodelan matematika ini diharapkan mampu menginterpretasi dan memprediksi penyebaran penyakit pertusis dan langkah penanganannya berupa vaksinasi [4].

2 Pembuatan Model

2.1 Asumsi

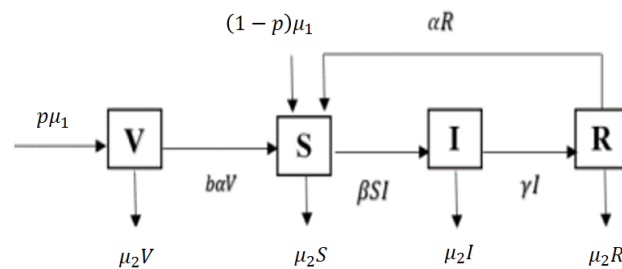
Dalam membangun sebuah model matematika dibutuhkan suatu batasan dalam mengamati suatu phenomena. Sehingga pada konstruksi model matematika untuk penyebaran penyakit pertusis dengan vaksinasi didapatkan asumsi awal. Adapun asumsi awal yang digunakan dalam mengkonstruksi model adalah sebagai berikut :

1. Populasi manusia dianggap konstan

2. Adanya tingkat kelahiran alami individu yang menjadi masukkan data
3. Adanya tingkat penularan dari individu terinfeksi ke individu rentan
4. Adanya tingkat imunitas individu, dimana apabila individu kehilangan imunitas maka individu tersebut mudah terinfeksi
5. Adanya laju kelahiran dan laju kematian alami

2.2 Diagram Kompartemen

Berdasarkan dengan asumsi awal yang telah diberikan, didapatkan suatu model matematika. Proses penyebaran penyakit pertusis dengan vaksinasi dalam suatu populasi dapat disajikan dengan diagram kompartemen berikut



Gambar 1 Diagram Kompartemen Penyebaran Penyakit Pertusis dengan Vaksinasi

2.3 Variabel dan Parameter

Variabel yang digunakan pada model matematika ini tertera pada tabel dibawah ini:

Tabel 1. Variabel dari Model Penyebaran Penyakit Pertusis dengan Vaksinasi

Variabel	Syarat	Keterangan
$V(t)$	$0 \leq V(t) \leq 1$	Proporsi banyak individu yang mendapat vaksinasi pada waktu t
$S(t)$	$0 \leq S(t) \leq 1$	Proporsi banyak individu yang rentan terinfeksi penyakit pada waktu t
$I(t)$	$0 \leq I(t) \leq 1$	Proporsi banyak individu yang terinfeksi penyakit pada waktu t
$R(t)$	$0 \leq R(t) \leq 1$	Proporsi banyak individu yang sembuh pada waktu t

Sedangkan parameter yang digunakan pada model matematika ini tertera pada tabel dibawah ini:

Tabel 2. Parameter dari Model Penyebaran Penyakit Pertusis dengan Vaksinasi

Parameter	Syarat	Keterangan
μ_1	$0 \leq \mu_1 \leq 1$	Laju kelahiran
μ_2	$0 \leq \mu_2 \leq 1$	Laju kematian
α	$0 \leq \alpha \leq 1$	Laju kehilangan imunitas
β	$0 \leq \beta \leq 1$	Peluang individu terinfeksi
γ	$0 \leq \gamma \leq 1$	Laju kesembuhan penyakit
p	$0 \leq p \leq 1$	Proporsi bayi baru lahir yang diberi vaksin
b	$0 \leq b \leq 1$	Rasio kehilangan imunitas pada individu yang divaksin dibandingkan dengan kehilangan imunitas alami pada individu sembuh dari infeksi

2.4 Model Matematika

Pada suatu populasi manusia konstan, proses penyebaran penyakit pertusis dimulai dengan langkah awal dengan pemberian vaksinasi sebagai pencegahan diberikan untuk bayi baru lahir. Lalu terdapat suatu kemungkinan kehilangan imunitas pada individu yang divaksin maupun kehilangan imunitas secara alami yang akan rentan terhadap penyakit pertusis. Kemudian adanya laju kematian pada individu yang telah diberi vaksin akan berpengaruh terhadap proporsi populasi. Sehingga proporsi banyaknya individu yang mendapatkan vaksinasi pada waktu t adalah selisih antara proporsi bayi baru lahir pada tiap laju kelahiran dengan proporsi kehilangan imunitas pada individu yang divaksin dan proporsi individu yang telah divaksin pada tiap laju kematian. Sehingga didapat suatu formulasi sebagai berikut:

$$\frac{dV}{dt} = p\mu_1 - b\alpha V - \mu_2 V \quad (1)$$

Pada individu yang rentan terinfeksi penyakit (*Susceptible*) diperoleh dari kehilangan imunitas pada individu yang divaksin maupun kehilangan imunitas secara alami, bayi baru lahir yang tidak mendapat vaksin pada tiap laju kelahiran dan individu yang sembuh dari penyakit kehilangan imunitasnya. Kemudian adanya laju kematian pada individu yang rentan terinfeksi penyakit terhadap proporsi populasi dan adanya peluang individu terinfeksi pada individu rentan terhadap penyakit dan dipengaruhi oleh individu yang terinfeksi (*Infectives*). Maka diperoleh suatu formulasi banyaknya individu yang rentan pada waktu t sebagai berikut:

$$\frac{dS}{dt} = b\alpha V + (1 - p)\mu_1 + \alpha R - \mu_2 S - \beta SI \quad (2)$$

Pada individu yang terinfeksi penyakit (*Infectives*) diperoleh adanya peluang individu terinfeksi pada individu rentan terhadap penyakit serta dipengaruhi oleh individu yang terinfeksi. Kemudian terdapat kemungkinan laju kematian pada individu yang terinfeksi penyakit dan individu yang berhasil sembuh dari penyakit yang akan masuk fase sembuh (*Recover*). Maka diperoleh suatu formulasi banyaknya individu yang terinfeksi pada waktu t

sebagai berikut:

$$\frac{dI}{dt} = \beta SI - (\mu_2 + \gamma)I \quad (3)$$

Pada fase terakhir, individu yang sembuh dari penyakit diperoleh dari individu yang terinfeksi penyakit berhasil sembuh. Kemudian terdapat kemungkinan individu yang sembuh dari penyakit kehilangan imunitasnya dan laju kematian pada individu yang sembuh. Maka diperoleh suatu formulasi banyaknya individu yang sembuh pada waktu t sebagai berikut:

$$\frac{dR}{dt} = \gamma I - \alpha R - \mu_2 R \quad (4)$$

Dalam model ini jumlah dari proporsi banyaknya individu yang mendapat vaksin, individu yang rentan, individu yang terinfeksi penyakit dan individu yang sembuh dari penyakit merupakan total populasi atau dapat dituliskan sebagai berikut :

$$V + S + I + R = 1 \quad (5)$$

3 Titik Keseimbangan

Keadaan setimbang adalah suatu keadaan ketika perubahan jumlah subpopulasi tertentu sepanjang waktu adalah nol. Berdasarkan definisi dari titik setimbang, model matematika penyebaran penyakit pertusis dengan vaksinasi pada manusia konstan memenuhi keadaan setimbang pada saat $\frac{dV}{dt} = \frac{dS}{dt} = \frac{dI}{dt} = \frac{dR}{dt} = 0$, sehingga persamaan dapat ditulis sebagai berikut

$$p\mu_1 - b\alpha V - \mu_2 V = 0 \quad (6)$$

$$b\alpha V + (1 - p)\mu_1 + \alpha R - \mu_2 S - \beta SI = 0 \quad (7)$$

$$\beta SI - (\mu_2 + \gamma)I = 0 \quad (8)$$

$$\gamma I - \alpha R - \mu_2 R = 0 \quad (9)$$

Titik setimbang dibedakan menjadi dua, yaitu titik setimbang bebas penyakit (non endemik) dan titik setimbang endemik. Dari persamaan-persamaan diatas diperoleh suatu titik setimbang non endemik dan suatu titik setimbang endemik pertusis.

Titik setimbang non endemik atau titik setimbang bebas penyakit merupakan suatu kondisi saat tidak terjadi penyebaran penyakit pertusis pada suatu wilayah. Keadaan ini terjadi ketika tidak ada manusia yang terinfeksi pertusis. Oleh karena itu dapat dinyatakan bahwa proporsi populasi manusia yang terinfeksi pertusis sama dengan nol ($I = 0$). Karena tidak ada populasi manusia yang terinfeksi, maka tidak ada proporsi populasi manusia yang sembuh dari penyakit pertusis ($R = 0$). Dengan mensubstitusi nilai $I = 0$ dan $R = 0$ pada persamaan, dapat diperoleh sebagai berikut.

Persamaan (6) dapat ditulis menjadi,

$$\begin{aligned} p\mu_1 &= b\alpha V + \mu_2 V \\ p\mu_1 &= (b\alpha + \mu_2)V \\ V &= \frac{p\mu_1}{(b\alpha + \mu_2)} \end{aligned}$$

Persamaan (7) dapat ditulis menjadi,

$$\begin{aligned} \mu_2 S &= b\alpha V + (1-p)\mu_1 \\ \mu_2 S &= \frac{b\alpha p\mu_1}{b\alpha + \mu_2} + (1-p)\mu_1 \\ \mu_2 S &= \frac{b\alpha p\mu_1 + b\alpha\mu_1 + \mu_1\mu_2 - b\alpha p\mu_1 - p\mu_1\mu_2}{b\alpha + \mu_2} \\ S &= \frac{1}{\mu_2} \left(\frac{b\alpha\mu_1 + (1-p)\mu_1\mu_2}{b\alpha + \mu_2} \right) \\ S &= \frac{b\alpha\mu_1 + (1-p)\mu_1\mu_2}{\mu_2(b\alpha + \mu_2)} \end{aligned}$$

Misalkan titik setimbang bebas penyakit (non endemik) dari model matematika penyebaran penyakit pertusis dengan manusia konstan dinyatakan dalam $P_0 = (V, S, 0, 0)$. Maka

$$P_0 = (V, S, I, R) = \left(\frac{p\mu_1}{(b\alpha + \mu_2)}, \frac{b\alpha\mu_1 + (1-p)\mu_1\mu_2}{\mu_2(b\alpha + \mu_2)}, 0, 0 \right)$$

Selanjutnya, titik setimbang endemik yaitu suatu kondisi ketika terjadi penyebaran penyakit pertusis dengan manusia konstan. Keadaan ini terjadi ketika terdapat manusia yang terinfeksi pertusis. Oleh karena itu dapat dinyatakan bahwa proporsi populasi manusia yang terinfeksi pertusis tidak sama dengan nol $I \neq 0$. Misalkan titik setimbang endemik dari model matematika penyebaran penyakit pertusis dengan manusia konstan dinyatakan dalam $P_1 = (V^*, S^*, I^*, R^*)$ sehingga sistem persamaan sebelumnya dapat ditulis menjadi:

$$p\mu_1 - b\alpha V^* - \mu_2 V^* = 0 \quad (10)$$

$$b\alpha V^* + (1-p)\mu_1 + \alpha R^* - \mu_2 S^* - \beta S^* I^* = 0 \quad (11)$$

$$\beta S^* I^* - (\mu_2 + \gamma) I^* = 0 \quad (12)$$

$$\gamma I^* - \alpha R^* - \mu_2 R^* = 0 \quad (13)$$

Selanjutnya, persamaan (10) dapat ditulis menjadi,

$$\begin{aligned} p\mu_1 &= b\alpha V^* + \mu_2 V^* \\ p\mu_1 &= (b\alpha + \mu_2)V^* \\ V^* &= \frac{p\mu_1}{(b\alpha + \mu_2)} \end{aligned}$$

Persamaan (12) dapat ditulis menjadi,

$$(\beta S^* - \mu_2 + \gamma)I^* = 0$$

karena $I^* \neq 0$ maka

$$S^* = \frac{\mu_2 + \gamma}{\beta}$$

Persamaan (13) dapat ditulis menjadi,

$$\begin{aligned} \gamma I^* - (\alpha + \mu_2)R^* &= 0 \\ R^* &= \frac{\gamma I^*}{\alpha + \mu_2} \end{aligned}$$

Persamaan (12) dapat ditulis menjadi,

$$\begin{aligned} b\alpha V^* + (1-p)\mu_1 + \alpha R^* - \mu_2 S^* - \beta S^* I^* &= 0 \\ \Leftrightarrow \frac{b\alpha p\mu_1}{b\alpha + \mu_2} + (1-p)\mu_1 + \frac{\alpha\gamma I^*}{\alpha + \mu_2} - \frac{\mu_2(\mu_2 + \gamma)}{\beta} - (\mu_2 + \gamma)I^* &= 0 \\ \Leftrightarrow I^* \left[(\mu_2 + \gamma) - \frac{\alpha\gamma}{\alpha + \mu_2} \right] &= \frac{\beta b\alpha p\mu_1 + (1-p)\mu_1\beta(b\alpha + \mu_2) - \mu_2(\mu_2 + \gamma)(b\alpha + \mu_2)}{\beta(b\alpha + \mu_2)} \\ \Leftrightarrow I^* \left[\frac{(\mu_2 + \gamma)(\alpha + \mu_2) - \alpha\gamma}{\alpha + \mu_2} \right] &= \frac{\beta b\alpha\mu_1 + (1-p)\mu_1\beta\mu_2 - \mu_2(\mu_2 + \gamma)(b\alpha + \mu_2)}{\beta(b\alpha + \mu_2)} \\ \Leftrightarrow I^* \left[\frac{\alpha\mu_2 + \mu_2^2 + \gamma\mu_2}{\alpha + \mu_2} \right] &= \frac{\beta b\alpha\mu_1 + (1-p)\mu_1\beta\mu_2 - \mu_2(\mu_2 + \gamma)(b\alpha + \mu_2)}{\beta(b\alpha + \mu_2)} \end{aligned}$$

Persamaan tersebut dapat disederhanakan lagi menjadi,

$$\begin{aligned} I^* &= \frac{1}{\left[\frac{\alpha\mu_2 + \mu_2^2 + \gamma\mu_2}{\alpha + \mu_2} \right]} \cdot \frac{\beta b\alpha\mu_1 + (1-p)\mu_1\beta\mu_2 - \mu_2(\mu_2 + \gamma)(b\alpha + \mu_2)}{\beta(b\alpha + \mu_2)} \\ I^* &= \frac{\alpha + \mu_2}{\alpha\mu_2 + \mu_2^2 + \gamma\mu_2} \cdot \frac{\beta b\alpha\mu_1 + (1-p)\mu_1\beta\mu_2 - \mu_2(\mu_2 + \gamma)(b\alpha + \mu_2)}{\beta(b\alpha + \mu_2)} \\ I^* &= \frac{(\alpha + \mu_2) [\beta b\alpha\mu_1 + (1-p)\mu_1\beta\mu_2 - \mu_2(\mu_2 + \gamma)(b\alpha + \mu_2)]}{\beta(b\alpha + \mu_2)(\alpha\mu_2 + \mu_2^2 + \gamma\mu_2)} \end{aligned}$$

Mengingat syarat dari titik kesetimbangan yang menyatakan bahwa setiap titik harus positif maka nilai $I^* > 0$, sehingga diperoleh pertidaksamaan sebagai berikut,

$$\begin{aligned} I^* &= \frac{(\alpha + \mu_2) [\beta b\alpha\mu_1 + (1-p)\mu_1\beta\mu_2 - \mu_2(\mu_2 + \gamma)(b\alpha + \mu_2)]}{\beta(b\alpha + \mu_2)(\alpha\mu_2 + \mu_2^2 + \gamma\mu_2)} > 0 \\ \frac{(\alpha + \mu_2)\mu_2(\mu_2 + \gamma)(b\alpha + \mu_2)}{\beta(b\alpha + \mu_2)(\alpha\mu_2 + \mu_2^2 + \gamma\mu_2)} \cdot \left[\frac{\beta b\alpha\mu_1 + (1-p)\mu_1\beta\mu_2}{\mu_2(\mu_2 + \gamma)(b\alpha + \mu_2)} - 1 \right] &> 0 \\ \frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{\beta(\alpha + \mu_2 + \gamma)} \cdot \left[\frac{\beta b\alpha\mu_1 + (1-p)\mu_1\beta\mu_2}{\mu_2(\mu_2 + \gamma)(b\alpha + \mu_2)} - 1 \right] &> 0 \\ \left[\frac{\beta b\alpha\mu_1 + (1-p)\mu_1\beta\mu_2}{\mu_2(\mu_2 + \gamma)(b\alpha + \mu_2)} - 1 \right] &> 0 \\ \left[\frac{\beta b\alpha\mu_1 + (1-p)\mu_1\beta\mu_2}{\mu_2(\mu_2 + \gamma)(b\alpha + \mu_2)} \right] &> 1 \end{aligned}$$

Sehingga I^* bernilai positif apabila $\left[\frac{\beta b \alpha \mu_1 + (1-p) \mu_1 \beta \mu_2}{\mu_2 (\mu_2 + \gamma) (b \alpha + \mu_2)} \right] > 1$. Jadi diperoleh titik setimbang endemik dari model matematika penyebaran penyakit pertusis dengan manusia konstan $P_1 = (V^*, S^*, I^*, R^*)$ adalah sebagai berikut.

$$\begin{aligned} V^* &= \frac{p \mu_1}{(b \alpha + \mu_2)} \\ S^* &= \frac{\mu_2 + \gamma}{\beta} \\ I^* &= \frac{(\alpha + \mu_2) [\beta b \alpha \mu_1 + (1-p) \mu_1 \beta \mu_2 - \mu_2 (\mu_2 + \gamma) (b \alpha + \mu_2)]}{\beta (b \alpha + \mu_2) (\alpha \mu_2 + \mu_2^2 + \gamma \mu_2)} \\ R^* &= \frac{\gamma I^*}{\alpha + \mu_2} \end{aligned}$$

Untuk menganalisis P_1 diperlukan bilangan reproduksi dasar (R_0), angka rasio reproduksi dasar ditentukan dengan mencari kondisi yang menjamin $I^* > 0$. Dari proses di atas telah ditunjukkan $I^* > 0$ apabila $\left[\frac{\beta b \alpha \mu_1 + (1-p) \mu_1 \beta \mu_2}{\mu_2 (\mu_2 + \gamma) (b \alpha + \mu_2)} \right] > 1$ sehingga dapat didefinisikan R_0 yaitu sebagai berikut,

$$R_0 = \left[\frac{\beta b \alpha \mu_1 + (1-p) \mu_1 \beta \mu_2}{\mu_2 (\mu_2 + \gamma) (b \alpha + \mu_2)} \right]$$

Jadi diperoleh titik setimbang endemik dari model matematika penyebaran penyakit pertusis manusia konstan $P_1 = (V^*, S^*, I^*, R^*)$ dengan

$$\begin{aligned} V^* &= \frac{p \mu_1}{(b \alpha + \mu_2)} \\ S^* &= \frac{\mu_2 + \gamma}{\beta} \\ I^* &= \frac{(\alpha + \mu_2) (\mu_2 + \gamma)}{\beta (\alpha + \mu_2 + \gamma)} (R_0 - 1) \\ R^* &= \frac{\gamma (\mu_2 + \gamma)}{\beta (\alpha + \mu_2 + \gamma)} (R_0 - 1) \end{aligned}$$

Berdasarkan perhitungan titik setimbang yang telah dilakukan diatas, diperoleh teorema 1 sebagai berikut.

Teorema 1

Dipunyai $R_0 = \left[\frac{\beta b \alpha \mu_1 + (1-p) \mu_1 \beta \mu_2}{\mu_2 (\mu_2 + \gamma) (b \alpha + \mu_2)} \right]$. Dari model matematika pada subbab (2.4) berdasarkan nilai R_0 tersebut diperoleh

1. Apabila $R_0 \leq 1$ maka model matematika pada subbab (2.4) hanya mempunyai 1 titik kesetimbangan yaitu titik kesetimbangan bebas penyakit (P_0) dengan

$$(V, S, I, R) = \left(\frac{p \mu_1}{(b \alpha + \mu_2)}, \frac{b \alpha \mu_1 + (1-p) \mu_1 \mu_2}{\mu_2 (b \alpha + \mu_2)}, 0, 0 \right)$$

2. Apabila $R_0 > 1$ maka model matematika pada subbab (2.4) mempunyai 2 titik kesetimbangan yaitu titik kesetimbangan bebas penyakit (P_0) dengan

$$(V, S, I, R) = \left(\frac{p \mu_1}{(b \alpha + \mu_2)}, \frac{b \alpha \mu_1 + (1-p) \mu_1 \mu_2}{\mu_2 (b \alpha + \mu_2)}, 0, 0 \right)$$

dan titik kesetimbangan endemik (P_1) dengan

$$(V^*, S^*, I^*, R^*) \\ = \left(\frac{p\mu_1}{(b\alpha + \mu_2)}, \frac{\mu_2 + \gamma}{\beta}, \frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{\beta(\alpha + \mu_2 + \gamma)}(R_0 - 1), \frac{\gamma(\mu_2 + \gamma)}{\beta(\alpha + \mu_2 + \gamma)}(R_0 - 1) \right)$$

4 Linierisasi

Untuk mendapatkan hasil pelinieran dari sistem persamaan diferensial digunakan matriks jacobian sebagai berikut

$$J(f(x)) = \begin{bmatrix} \frac{\partial \dot{V}}{\partial V} & \frac{\partial \dot{V}}{\partial S} & \frac{\partial \dot{V}}{\partial I} & \frac{\partial \dot{V}}{\partial R} \\ \frac{\partial \dot{S}}{\partial V} & \frac{\partial \dot{S}}{\partial S} & \frac{\partial \dot{S}}{\partial I} & \frac{\partial \dot{S}}{\partial R} \\ \frac{\partial \dot{I}}{\partial V} & \frac{\partial \dot{I}}{\partial S} & \frac{\partial \dot{I}}{\partial I} & \frac{\partial \dot{I}}{\partial R} \\ \frac{\partial \dot{R}}{\partial V} & \frac{\partial \dot{R}}{\partial S} & \frac{\partial \dot{R}}{\partial I} & \frac{\partial \dot{R}}{\partial R} \end{bmatrix}_{(V^*, S^*, I^*, R^*)}$$

Akan dimisalkan terlebih dahulu persamaan modelnya,

$$\begin{aligned} \frac{dV}{dt} &= \dot{V} \\ \frac{dS}{dt} &= \dot{S} \\ \frac{dI}{dt} &= \dot{I} \\ \frac{dR}{dt} &= \dot{R} \end{aligned}$$

sehingga diperoleh matriks jacobianya,

$$J(f(x)) = \begin{bmatrix} -b\alpha - \mu_2 & 0 & 0 & 0 \\ b\alpha & -\mu_2 - \beta I & -\beta S & \alpha \\ 0 & \beta I & \beta S - (\mu_2 + \gamma) & 0 \\ 0 & 0 & \gamma & -\alpha - \mu_2 \end{bmatrix}_{(V, S, I, R)}$$

Dengan demikian diperoleh pelinieran dari persamaan model, yaitu

$$\begin{bmatrix} \dot{V} \\ \dot{S} \\ \dot{I} \\ \dot{R} \end{bmatrix} = \begin{bmatrix} -b\alpha - \mu_2 & 0 & 0 & 0 \\ b\alpha & -\mu_2 - \beta I & -\beta S & \alpha \\ 0 & \beta I & \beta S - (\mu_2 + \gamma) & 0 \\ 0 & 0 & \gamma & -\alpha - \mu_2 \end{bmatrix}_{(V, S, I, R)} \begin{bmatrix} V \\ S \\ I \\ R \end{bmatrix}$$

5 Analisis Kestabilan

Kestabilan kedua titik kesetimbangan yang diperoleh dari **teorema 1** dianalisis dengan menggunakan nilai eigen dari matriks Jacobian yang telah diperoleh yaitu,

$$J(f(x)) = \begin{bmatrix} -b\alpha - \mu_2 & 0 & 0 & 0 \\ b\alpha & -\mu_2 - \beta I & -\beta S & \alpha \\ 0 & \beta I & \beta S - (\mu_2 + \gamma) & 0 \\ 0 & 0 & \gamma & -\alpha - \mu_2 \end{bmatrix}_{(V,S,I,R)}$$

Kemudian berdasarkan matriks Jacobian tersebut, akan dianalisis pada kedua titik ekuilibrium. Sehingga diperoleh matriks Jacobian pada **titik kesetimbangan nonendemik** $P_0 = (V, S, I, R) = \left(\frac{p\mu_1}{(b\alpha + \mu_2)}, \frac{b\alpha\mu_1 + (1-p)\mu_1\mu_2}{\mu_2(b\alpha + \mu_2)}, 0, 0 \right)$ sebagai berikut.

$$J(P_0) = \begin{bmatrix} -b\alpha - \mu_2 & 0 & 0 & 0 \\ b\alpha & -\mu_2 & -\frac{\beta b\alpha\mu_1 + \beta(1-p)\mu_1\mu_2}{\mu_2(b\alpha + \mu_2)} & \alpha \\ 0 & 0 & \frac{\beta b\alpha\mu_1 + \beta(1-p)\mu_1\mu_2}{\mu_2(b\alpha + \mu_2)} - (\mu_2 + \gamma) & 0 \\ 0 & 0 & \gamma & -\alpha - \mu_2 \end{bmatrix}$$

Dari matriks jacobian titik kesetimbangan P_0 diperoleh nilai eigen sebagai berikut,

$$\begin{aligned} \lambda_1 &= -b\alpha - \mu_2 \\ \lambda_2 &= -\mu_2 \\ \lambda_3 &= \frac{\beta b\alpha\mu_1 + \beta(1-p)\mu_1\mu_2}{\mu_2(b\alpha + \mu_2)} - (\mu_2 + \gamma) \\ &= (\mu_2 + \gamma) \left[\frac{\beta b\alpha\mu_1 + (1-p)\mu_1\beta\mu_2}{\mu_2(\mu_2 + \gamma)(b\alpha + \mu_2)} - 1 \right] \\ &= (\mu_2 + \gamma)(R_0 - 1) \\ \lambda_4 &= -\alpha - \mu_2 \end{aligned}$$

Jelas $\lambda_1, \lambda_2, \lambda_4$ bernilai negatif, $\lambda_3 < 0$ apabila $R_0 < 1$, $\lambda_3 = 0$ apabila $R_0 = 1$, dan $\lambda_3 > 0$ apabila $R_0 > 1$. Saat $R_0 = 1$ terdapat nilai eigen yang bernilai nol sehingga titik ekuilibrium P_0 tidak hiperbolik. Hal ini berakibat kestabilan P_0 hanya dilihat saat $R_0 \neq 1$. Jadi diperoleh titik ekuilibrium P_0 stabil asimtotik lokal apabila $R_0 < 1$ dan tidak stabil apabila $R_0 > 1$.

Selanjutnya dianalisis matriks Jacobian untuk mengetahui kestabilan **titik ekuilibrium endemik**

$$P_1 = (V^*, S^*, I^*, R^*) = \left(\frac{p\mu_1}{(b\alpha + \mu_2)}, \frac{\mu_2 + \gamma}{\beta}, \frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{\beta(\alpha + \mu_2 + \gamma)}(R_0 - 1), \frac{\gamma(\mu_2 + \gamma)}{\beta(\alpha + \mu_2 + \gamma)}(R_0 - 1) \right)$$

$$J(P_1) = \begin{bmatrix} -b\alpha - \mu_2 & 0 & 0 & 0 \\ b\alpha & -\mu_2 - \beta I^* & -\beta S^* & \alpha \\ 0 & \beta I^* & \beta S^* - (\mu_2 + \gamma) & 0 \\ 0 & 0 & \gamma & -\alpha - \mu_2 \end{bmatrix}_{(V^*, S^*, I^*, R^*)}$$

Kemudian akan dicari persamaan karakteristiknya dari matriks $J(P_1)$, yaitu:

$$\begin{aligned} |J(P_1) - \lambda I| &= 0 \\ \left| \begin{bmatrix} -b\alpha - \mu_2 & 0 & 0 & 0 \\ b\alpha & -\mu_2 - \beta I^* & -\beta S^* & \alpha \\ 0 & \beta I^* & \beta S^* - (\mu_2 + \gamma) & 0 \\ 0 & 0 & \gamma & -\alpha - \mu_2 \end{bmatrix} - \begin{bmatrix} \lambda & 0 & 0 & 0 \\ 0 & \lambda & 0 & 0 \\ 0 & 0 & \lambda & 0 \\ 0 & 0 & 0 & \lambda \end{bmatrix} \right| &= 0 \\ \left| \begin{bmatrix} -b\alpha - \mu_2 - \lambda & 0 & 0 & 0 \\ b\alpha & -\mu_2 - \beta I^* - \lambda & -\beta S^* & \alpha \\ 0 & \beta I^* & \beta S^* - (\mu_2 + \gamma) - \lambda & 0 \\ 0 & 0 & \gamma & -\alpha - \mu_2 - \lambda \end{bmatrix} \right| &= 0 \end{aligned}$$

Diperoleh persamaan karakteristiknya yaitu,

$$\begin{aligned} (\mu_2 + \lambda)(b\alpha + \mu_2 + \lambda)(\lambda^2 + \lambda(\gamma - \beta S^* + \beta I^* + 2\mu_2 + \alpha) + \beta I^* \gamma + \mu_2 \gamma - \mu_2 \beta S^* \\ + \mu_2 \beta I^* + \mu_2^2 + \alpha \gamma - \alpha \beta S^* + \alpha \beta S^* + \alpha \beta I^* + \alpha \mu_2) = 0 \end{aligned}$$

Diperoleh λ_1 dan λ_2 sebagai berikut

$$\begin{aligned} \lambda_1 &= -\mu_2 \\ \lambda_2 &= -(b\alpha + \mu_2) \end{aligned}$$

Perhatikan bahwa $\lambda_1, \lambda_2 < 0$. Kemudian akan diselidiki λ_3 dan λ_4 , perhatikan persamaan berikut

$$\begin{aligned}
& \lambda^2 + \lambda(\gamma - \beta S^* + \beta I^* + 2\mu_2 + \alpha) + \beta I^* \gamma + \mu_2 \gamma - \mu_2 \beta S^* + \mu_2 \beta I^* + \mu_2^2 + \alpha \gamma \\
& - \alpha \beta S^* + \alpha \beta S^* + \alpha \beta I^* + \alpha \mu_2) = 0 \\
\Leftrightarrow & \lambda^2 + \lambda \left(\gamma - \beta \frac{\mu_2 + \gamma}{\beta} + \beta \frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{\beta(\alpha + \mu_2 + \gamma)} (R_0 - 1) + 2\mu_2 + \alpha \right) + \beta \frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{\beta(\alpha + \mu_2 + \gamma)} (R_0 - 1) \gamma \\
& + \mu_2 \gamma - \mu_2 \beta \frac{\mu_2 + \gamma}{\beta} + \mu_2 \beta \frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{\beta(\alpha + \mu_2 + \gamma)} (R_0 - 1) + \mu_2^2 + \alpha \gamma - \alpha \beta \frac{\mu_2 + \gamma}{\beta} + \alpha \beta \frac{\mu_2 + \gamma}{\beta} + \\
& \alpha \beta \frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{\beta(\alpha + \mu_2 + \gamma)} (R_0 - 1) + \alpha \mu_2) = 0 \\
\Leftrightarrow & \lambda^2 + \lambda \left(\mu_2 + \alpha + \left(\frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{(\alpha + \mu_2 + \gamma)} (R_0 - 1) \right) \right) + (\alpha + \mu_2 + \gamma) \left(\frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{(\alpha + \mu_2 + \gamma)} (R_0 - 1) \right) \\
& + \mu_2 \gamma - \mu_2 (\mu_2 + \gamma) + \mu_2^2 + \alpha \gamma - \alpha (\mu_2 + \gamma) + \alpha \mu_2 = 0 \\
\Leftrightarrow & \lambda^2 + \lambda \left(\mu_2 + \alpha + \left(\frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{(\alpha + \mu_2 + \gamma)} (R_0 - 1) \right) \right) + (\alpha + \mu_2)(\mu_2 + \gamma)(R_0 - 1) \\
& + \mu_2 \gamma - \mu_2^2 - \mu_2 \gamma + \mu_2^2 + \alpha \gamma - \alpha \mu_2 - \alpha \gamma + \alpha \mu_2 = 0 \\
\Leftrightarrow & \lambda^2 + \lambda \left(\mu_2 + \alpha + \left(\frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{(\alpha + \mu_2 + \gamma)} (R_0 - 1) \right) \right) + (\alpha + \mu_2)(\mu_2 + \gamma)(R_0 - 1) = 0
\end{aligned}$$

Untuk memeriksa apakah akar-akar persamaan tersebut mempunyai bagian real negatif digunakan kriteria Routh-Hurwitz

s^2	1	$(\alpha + \mu_2)(\mu_2 + \gamma)(R_0 - 1)$
s^1	$\left(\mu_2 + \alpha + \left(\frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{(\alpha + \mu_2 + \gamma)} (R_0 - 1) \right) \right)$	0
s^0	b_1	0

diperoleh nilai b_1 , yaitu

$$\begin{aligned}
b_1 &= \frac{\left(\mu_2 + \alpha + \left(\frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{(\alpha + \mu_2 + \gamma)} (R_0 - 1) \right) \right) ((\alpha + \mu_2)(\mu_2 + \gamma)(R_0 - 1)) - 0}{\left(\mu_2 + \alpha + \left(\frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{(\alpha + \mu_2 + \gamma)} (R_0 - 1) \right) \right)} \\
&= (\alpha + \mu_2)(\mu_2 + \gamma)(R_0 - 1)
\end{aligned}$$

Jelas bahwa semua nilai parameter bernilai positif, dan jelas bahwa titik kesetimbangan endemik ada saat $R_0 > 1$. Jadi $R_0 - 1 > 0$ sehingga nilai dari

$$\begin{aligned}
& \left(\mu_2 + \alpha + \left(\frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{(\alpha + \mu_2 + \gamma)} (R_0 - 1) \right) \right) > 0, \text{ dan} \\
& (\alpha + \mu_2)(\mu_2 + \gamma)(R_0 - 1) > 0.
\end{aligned}$$

Sehingga diperoleh nilai kolom pertama tidak ada perubahan tanda. Jadi, nilai eigen λ_1, λ_2 bernilai negatif dan bagian real nilai eigen λ_3, λ_4 bernilai negatif. Dengan demikian, titik kesetimbangan P_1 stabil asimtot lokal apabila $R_0 > 1$.

Berdasarkan analisa yang telah dilakukan diperoleh Teorema 2 sebagai berikut. **Teorema 2**

Dipunyai $R_0 = \left\lceil \frac{\beta b \alpha \mu_1 + (1-p) \mu_1 \beta \mu_2}{\mu_2 (\mu_2 + \gamma) (b \alpha + \mu_2)} \right\rceil$.

1. Titik kesetimbangan nonendemik P_0 dengan

$$(V, S, I, R) = \left(\frac{p\mu_1}{(b\alpha + \mu_2)}, \frac{b\alpha\mu_1 + (1-p)\mu_1\mu_2}{\mu_2(b\alpha + \mu_2)}, 0, 0 \right)$$

stabil asimtot lokal apabila $R_0 < 1$ dan tidak stabil apabila $R_0 > 1$.

2. Titik kesetimbangan endemik P_1 dengan

$$(V^*, S^*, I^*, R^*) = \left(\frac{p\mu_1}{(b\alpha + \mu_2)}, \frac{\mu_2 + \gamma}{\beta}, \frac{(\alpha + \mu_2)(\mu_2 + \gamma)}{\beta(\alpha + \mu_2 + \gamma)}(R_0 - 1), \frac{\gamma(\mu_2 + \gamma)}{\beta(\alpha + \mu_2 + \gamma)}(R_0 - 1) \right)$$

stabil asimtotik lokal apabila $R_0 > 1$.

6 Simulasi Numerik

Simulasi numerik model matematika penyebaran penyakit pertusis dilakukan dengan menggunakan metode Runge-Kutta Orde Empat dan bantuan *software* MATLAB R2022a.

6.1 Metode Runge-Kutta Orde Empat

Metode Runge-Kutta merupakan metode yang memberikan ketelitian hasil yang lebih besar dan tidak memerlukan turunan dari fungsi [5]. Metode Runge-Kutta orde empat adalah sebagai berikut

$$y_{i+1} = y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \quad (14)$$

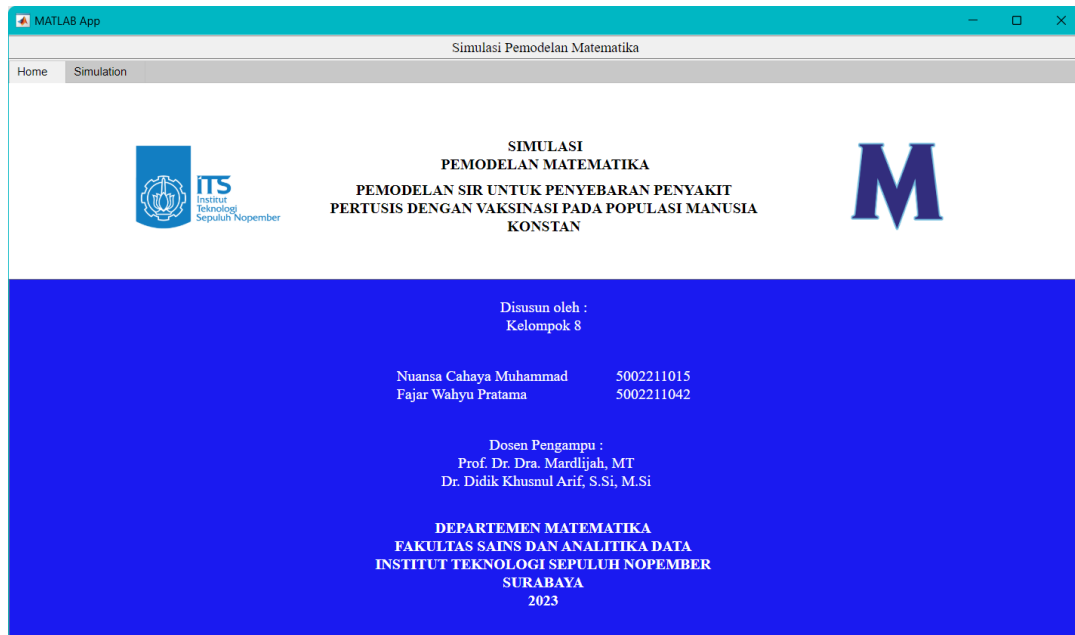
dengan

$$\begin{aligned} k_1 &= hf(t_i, y_i) \\ k_2 &= hf\left(t_i + \frac{1}{2}h, y_i + \frac{1}{2}k_1\right) \\ k_3 &= hf\left(t_i + \frac{1}{2}h, y_i + \frac{1}{2}k_2\right) \\ k_4 &= hf(t_i + h, y_i + k_3) \\ h &= \frac{t_f - t_0}{n} \end{aligned}$$

dimana h = ukuran langkah, n = banyak langkah, t_0 = waktu awal, dan t_f = waktu akhir.

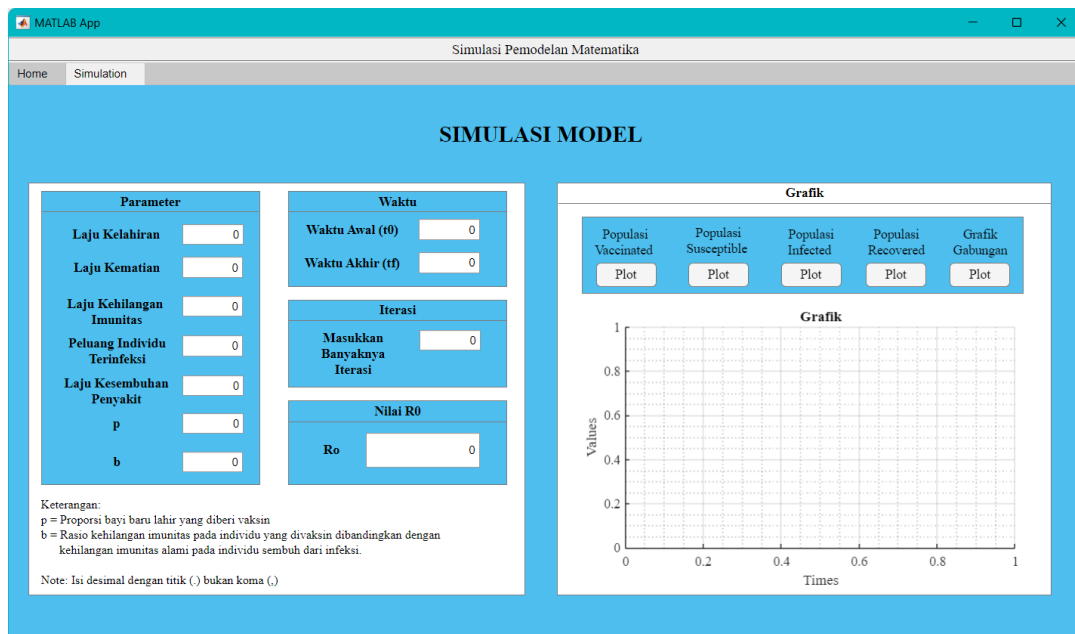
6.2 Implementasi Antarmuka

Implementasi antarmuka digunakan untuk mempermudah penggunaan program, implementasi antarmuka terdiri dari 2 menu yaitu menu “Home” dan “Simulasi”.



Gambar 2 Home Matlab App Simulasi

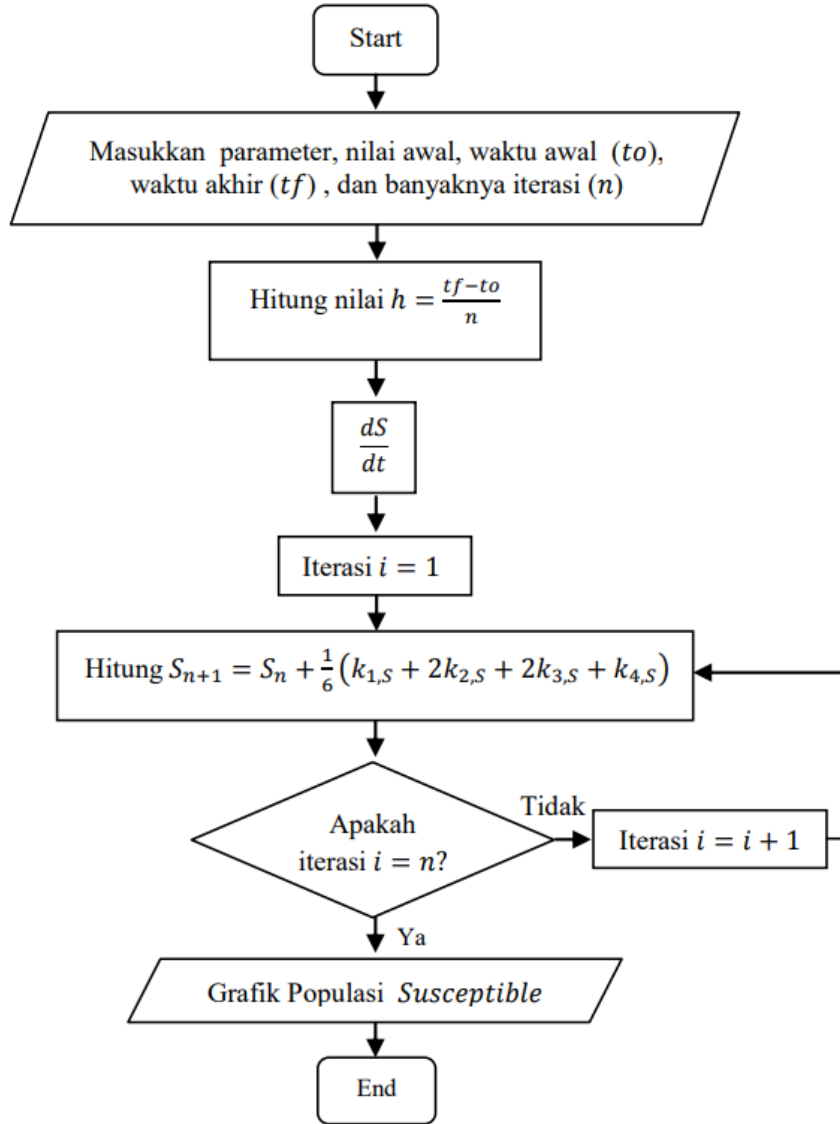
Simulasi numerik dilakukan pada menu “Simulasi”. Menu “Simulasi” ini terdiri dari 5 pushbutton “Plot” untuk menampilkan grafik populasi *vaccinated*, *susceptible*, *infected*, dan *recovered*, serta grafik gabungannya.



Gambar 3 Matlab App Simulasi

6.3 Flow Chart

Flow Chart merupakan diagram yang digunakan untuk menjelaskan proses berjalannya program dari sistem yang disimulasikan. Simulasi ini dilakukan dengan memberi nilai parameter $\mu_1, \mu_2, \alpha, \beta, \gamma, p, b$ dan nilai awal untuk masing-masing subpopulasi V, S, I, R serta waktu awal, waktu akhir dan banyaknya iterasi yang diinginkan. Kemudian nilai tersebut dihitung menggunakan metode Runge-Kutta sesuai persamaan (14). Flow Chart simulasi numerik pada model matematika penyebaran penyakit pertusis ditunjukkan oleh gambar berikut.



Gambar 4 Flow Chart Simulasi

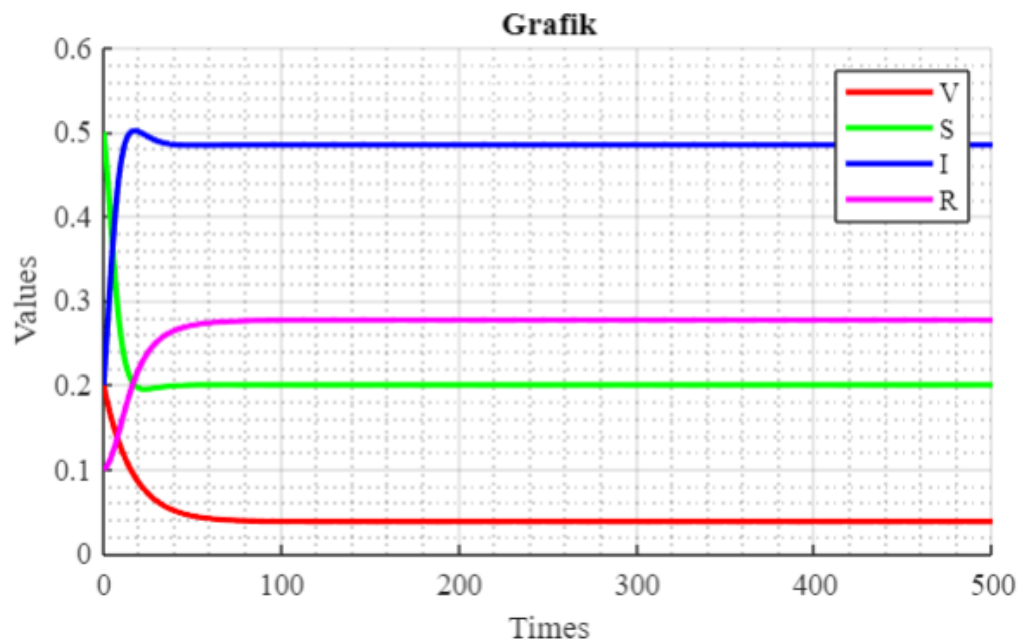
6.4 Hasil Simulasi

Simulasi diawali dengan memberikan nilai awal pada $V(0) = 0.2$, $S(0) = 0.5$, $I(0) = 0.2$, dan $R(0) = 0.1$ serta memberikan nilai awal pada parameter-parameter. Simulasi ini diberikan untuk memberikan gambaran geometris dari teorema eksistensi dan kestabilan dari titik-titik kesetimbangan model.

Tabel 3. Nilai Parameter

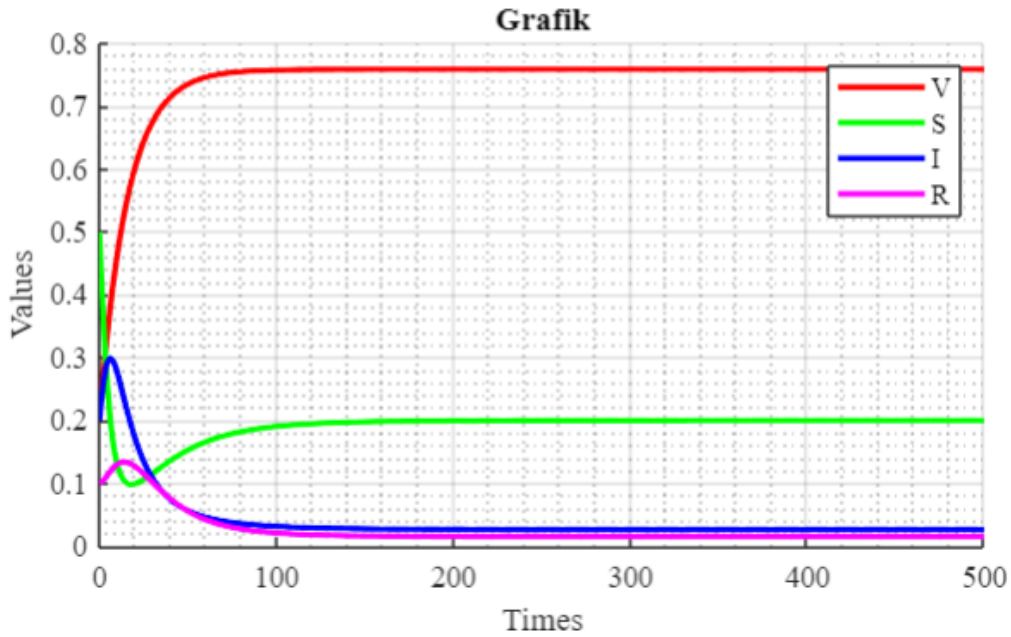
Parameter	Nilai Parameter	Sumber
μ_1	0.06	(Asyabah et al, 2018)
μ_2	0.06	(Asyabah et al, 2018)
α	0.01	Asumsi
β	0.5	Asumsi
γ	0.04	(Asyabah et al, 2018)
p	0.04	(Asyabah et al, 2018)
b	0.33	(Asyabah et al, 2018)

Dengan menggunakan nilai parameter pada **Tabel 3** dan nilai $t_0 = 0$ serta $t_f = 500$ hari, diperoleh nilai $R_0 = 4.8104 > 1$, dan berikut adalah grafik dari hasil simulasi:



Gambar 5 Simulasi Endemik

Kemudian akan dilakukan percobaan simulasi jika nilai parameter p dinaikkan menjadi 0.8 maka diperoleh nilai $R_0 = 1.2085 > 1$, dan berikut adalah hasil simulasinya



Gambar 6 Simulasi Endemik dengan nilai $p=0.8$

Dari gambar 5 diperoleh bahwa kelas $V(t)$ mengalami penurunan grafik, sedangkan dari gambar 6 dengan p yang lebih besar diperoleh bahwa kelas $V(t)$ mengalami kenaikan grafik, hal tersebut terjadi karena populasi yang masuk ke kelas vaksin dipengaruhi besarnya parameter p . Sehingga diperoleh hasil bahwa semakin kecil nilai p , maka populasi yang berada di kelas $V(t)$ sedikit dan sebaliknya semakin besar nilai p maka populasi yang berada di kelas $V(t)$ banyak. Hal ini bertolak belakang dengan kelas $I(t)$ dimana semakin kecil nilai p , maka individu yang terinfeksi semakin besar. Sebaliknya, semakin besar nilai p maka individu yang terinfeksi semakin kecil.

Sedangkan untuk grafik kelas $S(t)$, dari gambar 5 dan 6 dapat dilihat bahwa grafik mengalami turun naik. Terjadinya penurunan terlebih dahulu karena pada saat sistem dimulai individu yang awalnya rentan hanya dari kehilangan imunitas alami, dan grafik akan naik karena ketika ada individu yang sudah divaksin akan muncul individu rentan yang baru (individu kehilangan imunitas karena vaksin) dan juga populasi $S(t)$ bisa bertambah dari populasi individu yang sembuh tetapi kehilangan imunitas dari vaksin.

Dari gambar 5 dan 6 juga dapat dilihat bahwa grafik kelas $I(t)$ dan $R(t)$ linier, karena jika kelas individu yang terinfeksi sedikit, maka individu yang akan sembuh dari kelas terinfeksi juga akan sedikit, begitu juga sebaliknya. Hal ini sesuai dengan definisi, dimana batuk pertusis akan memiliki tahap pemulihan (tahapan konvalesen).

7 Kesimpulan

Dari pembahasan diperoleh model matematika $VSIR$ untuk penyebaran penyakit pertusis pada populasi manusia konstan. Analisis kedua model menghasilkan Teorema 1, Teorema 2 mengenai R_0 , titik kesetimbangan endemik dan nonendemik dan analisis kestabilan titik kesetimbangan endemik dan nonendemik. Selanjutnya pada simulasi model menghasilkan beberapa fakta bahwa semakin besar proporsi bayi yang divaksin (p) akan memperkecil jumlah penderita.

References

- [1] J. Mäkinen, J. Mertsola, F. R. Mooi, S. Van Amersfoort, H. Arvilommi, M. K. Viljanen, and Q. He, “Bordetella pertussis isolates, finland,” *Emerging Infectious Diseases*, vol. 11, no. 1, p. 183, 2005.
- [2] A. Elomaa, A. Advani, D. Donnelly, M. Antila, J. Mertsola, Q. He, and H. Hallander, “Population dynamics of bordetella pertussis in finland and sweden, neighbouring countries with different vaccination histories,” *Vaccine*, vol. 25, no. 5, pp. 918–926, 2007.
- [3] Z. Asyabah, S. B. Waluya, and M. Kharis, “Pemodelan sir untuk penyebaran penyakit pertusis dengan vaksinasi pada populasi manusia konstan,” *UNNES Journal of Mathematics*, vol. 7, no. 1, pp. 96–107, 2018.
- [4] Z. Hu, Z. Teng, and H. Jiang, “Stability analysis in a class of discrete sirs epidemic models,” *Nonlinear Analysis: Real World Applications*, vol. 13, no. 5, pp. 2017–2033, 2012.
- [5] R. Munir, “Metode numerik,” *Informatika, Bandung*, 2003.

Lampiran MATLAB Code

Source Code pada Push Button Plot Populasi Vaccinated

```
1  %----- inisialisasi parameter -----
2  t0 = app.t0.Value;
3  tf = app.tf.Value;
4  n = app.n.Value;
5
6  m1 = app.m1.Value;
7  m2 = app.m2.Value;
8  alpha = app.alpha.Value;
9  beta = app.beta.Value;
10 gamma = app.gamma.Value;
11 p = app.p.Value;
12 b = app.b.Value;
13
14 h=(tf-t0)/n;
15 t=0:h:n*h;
16 hh=1/2;
17
18 %----- variable state -----
19 V=zeros(n+1,1);
20 S=zeros(n+1,1);
21 I=zeros(n+1,1);
22 R=zeros(n+1,1);
23
24 %----- inisialisasi variabel state -----
25 V(1)=0.2;
26 S(1)=0.5;
27 I(1)=0.2;
28 R(1)=0.1;
29 N(1)=V(1)+S(1)+I(1)+R(1);
30 %----- Basic Reproduction Number -----
31 Ro=(beta*b*alpha*m1+(1-p)*m1*beta*m2)/(m2*(m2+gamma)*(b*alpha+m2));
32 %Menampilkan hasil perhitungan Ro
33 app.R0.Value=Ro;
34 %----- Representasi Metode Runge-Kutta -----
35 for i=1:n
36 %Step 1
37 K1_V(i)= h*(p*m1-b*alpha*V(i)-m2*V(i));
38 K1_S(i)=h*(b*alpha*V(i)+(1-p)*m1+alpha*R(i)-m2*S(i)-beta*S(i)*I
    (i));
39 K1_I(i)=h*(beta*S(i)*I(i)-(m2+gamma)*I(i));
40 K1_R(i)=h*(gamma*I(i)-alpha*R(i)-m2*R(i));
```

```

41 %Step2
42 K2_V(i)= h*(p*m1-b*alpha*(V(i)+hh*K1_V(i))-m2*(V(i)+hh*K1_V(i))
    );
43 K2_S(i)=h*(b*alpha*(V(i)+hh*K1_V(i))+(1-p)*m1+alpha*(R(i)+hh*
    K1_R(i))-m2*(S(i)+hh*K1_S(i))-beta*(S(i)+hh*K1_S(i))*(I(i)+hh
    *K1_I(i)));
44 K2_I(i)=h*(beta*(S(i)+hh*K1_S(i))*(I(i)+hh*K1_I(i))-(m2+gamma)
    *(I(i)+hh*K1_I(i)));
45 K2_R(i)=h*(gamma*(I(i)+hh*K1_I(i))-alpha*(R(i)+hh*K1_R(i))-m2*(
    R(i)+hh*K1_R(i)));
46 %Step3
47 K3_V(i)= h*(p*m1-b*alpha*(V(i)+hh*K2_V(i))-m2*(V(i)+hh*K2_V(i))
    );
48 K3_S(i)=h*(b*alpha*(V(i)+hh*K2_V(i))+(1-p)*m1+alpha*(R(i)+hh*
    K2_R(i))-m2*(S(i)+hh*K2_S(i))-beta*(S(i)+hh*K2_S(i))*(I(i)+hh
    *K2_I(i)));
49 K3_I(i)=h*(beta*(S(i)+hh*K2_S(i))*(I(i)+hh*K2_I(i))-(m2+gamma)
    *(I(i)+hh*K2_I(i)));
50 K3_R(i)=h*(gamma*(I(i)+hh*K2_I(i))-alpha*(R(i)+hh*K2_R(i))-m2*(
    R(i)+hh*K2_R(i)));
51 %Step4
52 K4_V(i)= h*(p*m1-b*alpha*(V(i)+K3_V(i))-m2*(V(i)+K3_V(i)));
53 K4_S(i)=h*(b*alpha*(V(i)+K3_V(i))+(1-p)*m1+alpha*(R(i)+K3_R(i))
    -m2*(S(i)+K3_S(i))-beta*(S(i)+K3_S(i))*(I(i)+K3_I(i)));
54 K4_I(i)=h*(beta*(S(i)+K3_S(i))*(I(i)+K3_I(i))-(m2+gamma)*(I(i)+
    K3_I(i)));
55 K4_R(i)=h*(gamma*(I(i)+K3_I(i))-alpha*(R(i)+K3_R(i))-m2*(R(i)+
    K3_R(i)));
56 %Total
57 V(i+1) = V(i) + (1/6)*(K1_V(i) + (2*K2_V(i)) + (2*K3_V(i)) +
    K4_V(i));
58 S(i+1) = S(i) + (1/6)*(K1_S(i) + (2*K2_S(i)) + (2*K3_S(i)) +
    K4_S(i));
59 I(i+1) = I(i) + (1/6)*(K1_I(i) + (2*K2_I(i)) + (2*K3_I(i)) +
    K4_I(i));
60 R(i+1) = R(i) + (1/6)*(K1_R(i) + (2*K2_R(i)) + (2*K3_R(i)) +
    K4_R(i));
61 end
62
63 plot(app.UIAxes,t,V,'r','LineWidth',2)
64 legend(app.UIAxes, 'V')
65 hold(app.UIAxes, 'off')

```

Source Code pada Push Button Plot Populasi Susceptible

```

1  %----- inisialisasi parameter -----
2  t0 = app.t0.Value;
3  tf = app.tf.Value;
4  n = app.n.Value;
5
6  m1 = app.m1.Value;
7  m2 = app.m2.Value;
8  alpha = app.alpha.Value;
9  beta = app.beta.Value;
10 gamma = app.gamma.Value;
11 p = app.p.Value;
12 b = app.b.Value;
13
14 h=(tf-t0)/n;
15 t=0:h:n*h;
16 hh=1/2;
17
18 %----- variable state -----
19 V=zeros(n+1,1);
20 S=zeros(n+1,1);
21 I=zeros(n+1,1);
22 R=zeros(n+1,1);
23
24 %%----- inisialisasi variabel state -----
25 V(1)=0.2;
26 S(1)=0.5;
27 I(1)=0.2;
28 R(1)=0.1;
29 N(1)=V(1)+S(1)+I(1)+R(1);
30 %----- Basic Reproduction Number -----
31 Ro=(beta*b*alpha*m1+(1-p)*m1*beta*m2)/(m2*(m2+gamma)*(b*alpha+
    m2));
32 %Menampilkan hasil perhitungan Ro
33 app.R0.Value=Ro;
34 %----- Representasi Metode Runge-Kutta -----
35 for i=1:n
36 %Step 1
37 K1_V(i)= h*(p*m1-b*alpha*V(i)-m2*V(i));
38 K1_S(i)=h*(b*alpha*V(i)+(1-p)*m1+alpha*R(i)-m2*S(i)-beta*S(i)*I
    (i));
39 K1_I(i)=h*(beta*S(i)*I(i)-(m2+gamma)*I(i));
40 K1_R(i)=h*(gamma*I(i)-alpha*R(i)-m2*R(i));
41 %Step2

```

```

42 K2_V(i)= h*(p*m1-b*alpha*(V(i)+hh*K1_V(i))-m2*(V(i)+hh*K1_V(i))
    );
43 K2_S(i)=h*(b*alpha*(V(i)+hh*K1_V(i))+(1-p)*m1+alpha*(R(i)+hh*
    K1_R(i))-m2*(S(i)+hh*K1_S(i))-beta*(S(i)+hh*K1_S(i))*(I(i)+hh
    *K1_I(i)));
44 K2_I(i)=h*(beta*(S(i)+hh*K1_S(i))*(I(i)+hh*K1_I(i))-(m2+gamma)
    *(I(i)+hh*K1_I(i)));
45 K2_R(i)=h*(gamma*(I(i)+hh*K1_I(i))-alpha*(R(i)+hh*K1_R(i))-m2*(
    R(i)+hh*K1_R(i)));
46 %Step3
47 K3_V(i)= h*(p*m1-b*alpha*(V(i)+hh*K2_V(i))-m2*(V(i)+hh*K2_V(i))
    );
48 K3_S(i)=h*(b*alpha*(V(i)+hh*K2_V(i))+(1-p)*m1+alpha*(R(i)+hh*
    K2_R(i))-m2*(S(i)+hh*K2_S(i))-beta*(S(i)+hh*K2_S(i))*(I(i)+hh
    *K2_I(i)));
49 K3_I(i)=h*(beta*(S(i)+hh*K2_S(i))*(I(i)+hh*K2_I(i))-(m2+gamma)
    *(I(i)+hh*K2_I(i)));
50 K3_R(i)=h*(gamma*(I(i)+hh*K2_I(i))-alpha*(R(i)+hh*K2_R(i))-m2*(
    R(i)+hh*K2_R(i)));
51 %Step4
52 K4_V(i)= h*(p*m1-b*alpha*(V(i)+K3_V(i))-m2*(V(i)+K3_V(i)));
53 K4_S(i)=h*(b*alpha*(V(i)+K3_V(i))+(1-p)*m1+alpha*(R(i)+K3_R(i))
    -m2*(S(i)+K3_S(i))-beta*(S(i)+K3_S(i))*(I(i)+K3_I(i)));
54 K4_I(i)=h*(beta*(S(i)+K3_S(i))*(I(i)+K3_I(i))-(m2+gamma)*(I(i)+
    K3_I(i)));
55 K4_R(i)=h*(gamma*(I(i)+K3_I(i))-alpha*(R(i)+K3_R(i))-m2*(R(i)+
    K3_R(i)));
56 %Total
57 V(i+1) = V(i) + (1/6)*(K1_V(i) + (2*K2_V(i)) + (2*K3_V(i)) +
    K4_V(i));
58 S(i+1) = S(i) + (1/6)*(K1_S(i) + (2*K2_S(i)) + (2*K3_S(i)) +
    K4_S(i));
59 I(i+1) = I(i) + (1/6)*(K1_I(i) + (2*K2_I(i)) + (2*K3_I(i)) +
    K4_I(i));
60 R(i+1) = R(i) + (1/6)*(K1_R(i) + (2*K2_R(i)) + (2*K3_R(i)) +
    K4_R(i));
61 end
62
63 plot(app.UIAxes,t,S,'g','LineWidth',2)
64 legend(app.UIAxes, 'S')
65 hold(app.UIAxes, 'off')

```

Source Code pada Push Button Plot Populasi Infected

```

1 %----- inisialisasi parameter -----

```

```

2  t0 = app.t0.Value;
3  tf = app.tf.Value;
4  n = app.n.Value;
5
6  m1 = app.m1.Value;
7  m2 = app.m2.Value;
8  alpha = app.alpha.Value;
9  beta = app.beta.Value;
10 gamma = app.gamma.Value;
11 p = app.p.Value;
12 b = app.b.Value;
13
14 h=(tf-t0)/n;
15 t=0:h:n*h;
16 hh=1/2;
17
18 %----- variable state -----
19 V=zeros(n+1,1);
20 S=zeros(n+1,1);
21 I=zeros(n+1,1);
22 R=zeros(n+1,1);
23
24 %%----- inisialisasi variabel state -----
25 V(1)=0.2;
26 S(1)=0.5;
27 I(1)=0.2;
28 R(1)=0.1;
29 N(1)=V(1)+S(1)+I(1)+R(1);
30 %----- Basic Reproduction Number -----
31 Ro=(beta*b*alpha*m1+(1-p)*m1*beta*m2)/(m2*(m2+gamma)*(b*alpha+m2));
32 %Menampilkan hasil perhitungan Ro
33 app.R0.Value=Ro;
34 %----- Representasi Metode Runge-Kutta -----
35 for i=1:n
36 %Step 1
37 K1_V(i)= h*(p*m1-b*alpha*V(i)-m2*V(i));
38 K1_S(i)=h*(b*alpha*V(i)+(1-p)*m1+alpha*R(i)-m2*S(i)-beta*S(i)*I(i));
39 K1_I(i)=h*(beta*S(i)*I(i)-(m2+gamma)*I(i));
40 K1_R(i)=h*(gamma*I(i)-alpha*R(i)-m2*R(i));
41 %Step2
42 K2_V(i)= h*(p*m1-b*alpha*(V(i)+hh*K1_V(i))-m2*(V(i)+hh*K1_V(i)));

```

```

43 K2_S(i)=h*(b*alpha*(V(i)+hh*K1_V(i))+(1-p)*m1+alpha*(R(i)+hh*
    K1_R(i))-m2*(S(i)+hh*K1_S(i))-beta*(S(i)+hh*K1_S(i))*(I(i)+hh
    *K1_I(i)));
44 K2_I(i)=h*(beta*(S(i)+hh*K1_S(i))*(I(i)+hh*K1_I(i))-(m2+gamma)
    *(I(i)+hh*K1_I(i)));
45 K2_R(i)=h*(gamma*(I(i)+hh*K1_I(i))-alpha*(R(i)+hh*K1_R(i))-m2*(
    R(i)+hh*K1_R(i)));
46 %Step3
47 K3_V(i)= h*(p*m1-b*alpha*(V(i)+hh*K2_V(i))-m2*(V(i)+hh*K2_V(i))
    );
48 K3_S(i)=h*(b*alpha*(V(i)+hh*K2_V(i))+(1-p)*m1+alpha*(R(i)+hh*
    K2_R(i))-m2*(S(i)+hh*K2_S(i))-beta*(S(i)+hh*K2_S(i))*(I(i)+hh
    *K2_I(i)));
49 K3_I(i)=h*(beta*(S(i)+hh*K2_S(i))*(I(i)+hh*K2_I(i))-(m2+gamma)
    *(I(i)+hh*K2_I(i)));
50 K3_R(i)=h*(gamma*(I(i)+hh*K2_I(i))-alpha*(R(i)+hh*K2_R(i))-m2*(
    R(i)+hh*K2_R(i)));
51 %Step4
52 K4_V(i)= h*(p*m1-b*alpha*(V(i)+K3_V(i))-m2*(V(i)+K3_V(i)));
53 K4_S(i)=h*(b*alpha*(V(i)+K3_V(i))+(1-p)*m1+alpha*(R(i)+K3_R(i))
    -m2*(S(i)+K3_S(i))-beta*(S(i)+K3_S(i))*(I(i)+K3_I(i)));
54 K4_I(i)=h*(beta*(S(i)+K3_S(i))*(I(i)+K3_I(i))-(m2+gamma)*(I(i)+
    K3_I(i)));
55 K4_R(i)=h*(gamma*(I(i)+K3_I(i))-alpha*(R(i)+K3_R(i))-m2*(R(i)+
    K3_R(i)));
56 %Total
57 V(i+1) = V(i) + (1/6)*(K1_V(i) + (2*K2_V(i)) + (2*K3_V(i)) +
    K4_V(i));
58 S(i+1) = S(i) + (1/6)*(K1_S(i) + (2*K2_S(i)) + (2*K3_S(i)) +
    K4_S(i));
59 I(i+1) = I(i) + (1/6)*(K1_I(i) + (2*K2_I(i)) + (2*K3_I(i)) +
    K4_I(i));
60 R(i+1) = R(i) + (1/6)*(K1_R(i) + (2*K2_R(i)) + (2*K3_R(i)) +
    K4_R(i));
61 end
62
63 plot(app.UIAxes,t,I,'b','LineWidth',2)
64 legend(app.UIAxes, 'I')
65 hold(app.UIAxes, 'off')

```

Source Code pada Push Button Plot Populasi Recovered

```

1 %----- inisialisasi parameter -----
2 t0 = app.t0.Value;
3 tf = app.tf.Value;

```



```

4  n = app.n.Value;
5
6  m1 = app.m1.Value;
7  m2 = app.m2.Value;
8  alpha = app.alpha.Value;
9  beta = app.beta.Value;
10 gamma = app.gamma.Value;
11 p = app.p.Value;
12 b = app.b.Value;
13
14 h=(tf-t0)/n;
15 t=0:h:n*h;
16 hh=1/2;
17
18 %----- variable state -----
19 V=zeros(n+1,1);
20 S=zeros(n+1,1);
21 I=zeros(n+1,1);
22 R=zeros(n+1,1);
23
24 %----- inisialisasi variabel state -----
25 V(1)=0.2;
26 S(1)=0.5;
27 I(1)=0.2;
28 R(1)=0.1;
29 N(1)=V(1)+S(1)+I(1)+R(1);
30 %----- Basic Reproduction Number -----
31 Ro=(beta*b*alpha*m1+(1-p)*m1*beta*m2)/(m2*(m2+gamma)*(b*alpha+m2));
32 %Menampilkan hasil perhitungan Ro
33 app.R0.Value=Ro;
34 %----- Representasi Metode Runge-Kutta -----
35 for i=1:n
36 %Step 1
37 K1_V(i)= h*(p*m1-b*alpha*V(i)-m2*V(i));
38 K1_S(i)=h*(b*alpha*V(i)+(1-p)*m1+alpha*R(i)-m2*S(i)-beta*S(i)*I
    (i));
39 K1_I(i)=h*(beta*S(i)*I(i)-(m2+gamma)*I(i));
40 K1_R(i)=h*(gamma*I(i)-alpha*R(i)-m2*R(i));
41 %Step2
42 K2_V(i)= h*(p*m1-b*alpha*(V(i)+hh*K1_V(i))-m2*(V(i)+hh*K1_V(i))
    );
43 K2_S(i)=h*(b*alpha*(V(i)+hh*K1_V(i))+(1-p)*m1+alpha*(R(i)+hh*
    K1_R(i))-m2*(S(i)+hh*K1_S(i))-beta*(S(i)+hh*K1_S(i))*(I(i)+hh
    *K1_I(i)));

```

```

44 K2_I(i)=h*(beta*(S(i)+hh*K1_S(i))*(I(i)+hh*K1_I(i))-(m2+gamma)
    *(I(i)+hh*K1_I(i)));
45 K2_R(i)=h*(gamma*(I(i)+hh*K1_I(i))-alpha*(R(i)+hh*K1_R(i))-m2*(
    R(i)+hh*K1_R(i)));
46 %Step3
47 K3_V(i)= h*(p*m1-b*alpha*(V(i)+hh*K2_V(i))-m2*(V(i)+hh*K2_V(i))
    );
48 K3_S(i)=h*(b*alpha*(V(i)+hh*K2_V(i))+(1-p)*m1+alpha*(R(i)+hh*
    K2_R(i))-m2*(S(i)+hh*K2_S(i))-beta*(S(i)+hh*K2_S(i))*(I(i)+hh
    *K2_I(i)));
49 K3_I(i)=h*(beta*(S(i)+hh*K2_S(i))*(I(i)+hh*K2_I(i))-(m2+gamma)
    *(I(i)+hh*K2_I(i)));
50 K3_R(i)=h*(gamma*(I(i)+hh*K2_I(i))-alpha*(R(i)+hh*K2_R(i))-m2*(
    R(i)+hh*K2_R(i)));
51 %Step4
52 K4_V(i)= h*(p*m1-b*alpha*(V(i)+K3_V(i))-m2*(V(i)+K3_V(i)));
53 K4_S(i)=h*(b*alpha*(V(i)+K3_V(i))+(1-p)*m1+alpha*(R(i)+K3_R(i))
    -m2*(S(i)+K3_S(i))-beta*(S(i)+K3_S(i))*(I(i)+K3_I(i)));
54 K4_I(i)=h*(beta*(S(i)+K3_S(i))*(I(i)+K3_I(i))-(m2+gamma)*(I(i)+
    K3_I(i)));
55 K4_R(i)=h*(gamma*(I(i)+K3_I(i))-alpha*(R(i)+K3_R(i))-m2*(R(i)+
    K3_R(i)));
56 %Total
57 V(i+1) = V(i) + (1/6)*(K1_V(i) + (2*K2_V(i)) + (2*K3_V(i)) +
    K4_V(i));
58 S(i+1) = S(i) + (1/6)*(K1_S(i) + (2*K2_S(i)) + (2*K3_S(i)) +
    K4_S(i));
59 I(i+1) = I(i) + (1/6)*(K1_I(i) + (2*K2_I(i)) + (2*K3_I(i)) +
    K4_I(i));
60 R(i+1) = R(i) + (1/6)*(K1_R(i) + (2*K2_R(i)) + (2*K3_R(i)) +
    K4_R(i));
61 end
62
63 plot(app.UIAxes,t,R,'m','LineWidth',2)
64 legend(app.UIAxes, 'R')
65 hold(app.UIAxes, 'off')

```

Source Code pada Push Button Plot Grafik Gabungan

```

1 %----- inisialisasi parameter -----
2 t0 = app.t0.Value;
3 tf = app.tf.Value;
4 n = app.n.Value;
5
6 m1 = app.m1.Value;

```

```

7 m2 = app.m2.Value;
8 alpha = app.alpha.Value;
9 beta = app.beta.Value;
10 gamma = app.gamma.Value;
11 p = app.p.Value;
12 b = app.b.Value;
13
14 h=(tf-t0)/n;
15 t=0:h:n*h;
16 hh=1/2;
17
18 %----- variable state -----
19 V=zeros(n+1,1);
20 S=zeros(n+1,1);
21 I=zeros(n+1,1);
22 R=zeros(n+1,1);
23
24 %----- inisialisasi variabel state -----
25 V(1)=0.2;
26 S(1)=0.5;
27 I(1)=0.2;
28 R(1)=0.1;
29 N(1)=V(1)+S(1)+I(1)+R(1);
30 %----- Basic Reproduction Number -----
31 Ro=(beta*b*alpha*m1+(1-p)*m1*beta*m2)/(m2*(m2+gamma)*(b*alpha+m2));
32 %Menampilkan hasil perhitungan Ro
33 app.R0.Value=Ro;
34 %----- Representasi Metode Runge-Kutta -----
35 for i=1:n
36 %Step 1
37 K1_V(i)= h*(p*m1-b*alpha*V(i)-m2*V(i));
38 K1_S(i)=h*(b*alpha*V(i)+(1-p)*m1+alpha*R(i)-m2*S(i)-beta*S(i)*I(i));
39 K1_I(i)=h*(beta*S(i)*I(i)-(m2+gamma)*I(i));
40 K1_R(i)=h*(gamma*I(i)-alpha*R(i)-m2*R(i));
41 %Step2
42 K2_V(i)= h*(p*m1-b*alpha*(V(i)+hh*K1_V(i))-m2*(V(i)+hh*K1_V(i)));
43 K2_S(i)=h*(b*alpha*(V(i)+hh*K1_V(i))+(1-p)*m1+alpha*(R(i)+hh*K1_R(i))-m2*(S(i)+hh*K1_S(i))-beta*(S(i)+hh*K1_S(i))*(I(i)+hh*K1_I(i)));
44 K2_I(i)=h*(beta*(S(i)+hh*K1_S(i))*(I(i)+hh*K1_I(i))-(m2+gamma)*(I(i)+hh*K1_I(i)));

```

```

45 K2_R(i)=h*(gamma*(I(i)+hh*K1_I(i))-alpha*(R(i)+hh*K1_R(i))-m2*(
    R(i)+hh*K1_R(i)));
46 %Step3
47 K3_V(i)= h*(p*m1-b*alpha*(V(i)+hh*K2_V(i))-m2*(V(i)+hh*K2_V(i))
    );
48 K3_S(i)=h*(b*alpha*(V(i)+hh*K2_V(i))+(1-p)*m1+alpha*(R(i)+hh*
    K2_R(i))-m2*(S(i)+hh*K2_S(i))-beta*(S(i)+hh*K2_S(i))*(I(i)+hh
    *K2_I(i)));
49 K3_I(i)=h*(beta*(S(i)+hh*K2_S(i))*(I(i)+hh*K2_I(i))-(m2+gamma)
    *(I(i)+hh*K2_I(i)));
50 K3_R(i)=h*(gamma*(I(i)+hh*K2_I(i))-alpha*(R(i)+hh*K2_R(i))-m2*(
    R(i)+hh*K2_R(i)));
51 %Step4
52 K4_V(i)= h*(p*m1-b*alpha*(V(i)+K3_V(i))-m2*(V(i)+K3_V(i)));
53 K4_S(i)=h*(b*alpha*(V(i)+K3_V(i))+(1-p)*m1+alpha*(R(i)+K3_R(i))
    -m2*(S(i)+K3_S(i))-beta*(S(i)+K3_S(i))*(I(i)+K3_I(i)));
54 K4_I(i)=h*(beta*(S(i)+K3_S(i))*(I(i)+K3_I(i))-(m2+gamma)*(I(i)+
    K3_I(i)));
55 K4_R(i)=h*(gamma*(I(i)+K3_I(i))-alpha*(R(i)+K3_R(i))-m2*(R(i)+
    K3_R(i)));
56 %Total
57 V(i+1) = V(i) + (1/6)*(K1_V(i) + (2*K2_V(i)) + (2*K3_V(i)) +
    K4_V(i));
58 S(i+1) = S(i) + (1/6)*(K1_S(i) + (2*K2_S(i)) + (2*K3_S(i)) +
    K4_S(i));
59 I(i+1) = I(i) + (1/6)*(K1_I(i) + (2*K2_I(i)) + (2*K3_I(i)) +
    K4_I(i));
60 R(i+1) = R(i) + (1/6)*(K1_R(i) + (2*K2_R(i)) + (2*K3_R(i)) +
    K4_R(i));
61 end
62
63 plot(app.UIAxes,t,V,'r','LineWidth',2)
64 hold(app.UIAxes, 'on')
65 plot(app.UIAxes,t,S,'g','LineWidth',2)
66 hold(app.UIAxes, 'on')
67 plot(app.UIAxes,t,I,'b','LineWidth',2)
68 hold(app.UIAxes, 'on')
69 plot(app.UIAxes,t,R,'m','LineWidth',2)
70 legend(app.UIAxes, 'V','S','I','R')
71 hold(app.UIAxes, 'off')

```