



UNIVERSIDADE FEDERAL DE PERNAMBUCO  
CENTRO DE CIÊNCIAS EXATAS E DA NATUREZA  
PROGRAMA DE PÓS-GRADUAÇÃO EM MATEMÁTICA

NELSON LEAL DOS SANTOS JUNIOR

**CONTROLE ÓTIMO APLICADO A ESTRATÉGIAS DE CONTENÇÃO DE  
DOENÇAS INFECCIOSAS: UM ESTUDO COM DADOS DO BRASIL**

Recife

2024

NELSON LEAL DOS SANTOS JUNIOR

**CONTROLE ÓTIMO APLICADO A ESTRATÉGIAS DE CONTENÇÃO DE  
DOENÇAS INFECCIOSAS: UM ESTUDO COM DADOS DO BRASIL**

Tese apresentada ao Programa de Pós-Graduação em Matemática da Universidade Federal de Pernambuco, como requisito parcial para obtenção do título de doutor em matemática. Área de concentração: Análise.

**Orientador (a): JOÃO ANTÔNIO MIRANDA GONDIM**

Recife

2024

.Catalogação de Publicação na Fonte. UFPE - Biblioteca Central

Santos Júnior, Nelson Leal dos.

Controle ótimo aplicado a estratégias de contenção de doenças infecciosas: um estudo com dados do Brasil / Nelson Leal Dos Santos Júnior. - Recife, 2024.

94f.: il.

Tese (Doutorado) - Universidade Federal de Pernambuco, Centro de Ciências Exatas e da Natureza, Programa de Pós-Graduação em Matemática, 2024.

Orientação: João Antônio Miranda Gondim.

Inclui referências e anexos.

1. Modelo SEIRQ; 2. Modelo SEIRV; 3. Controle Ótimo; 4. Estratégias de Contenção. I. Gondim, João Antônio Miranda. II. Título.

UFPE-Biblioteca Central

**NELSON LEAL DOS SANTOS JÚNIOR**

*Controle ótimo aplicado a estratégias de contenção de doenças infecciosas:  
um estudo com dados do Brasil*

Tese apresentada ao Programa de Pós-graduação do Departamento de Matemática da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutorado em Matemática.

Aprovado em: 21/10/2024

**BANCA EXAMINADORA**

---

Prof. Dr. João Antônio Miranda Gondim (Orientador)  
Universidade Federal de Pernambuco

---

Prof. Dr. César Augusto Rodrigues Castilho (Examinador Interno)  
Universidade Federal de Pernambuco

---

Prof. Dr. Wellington Pinheiro dos Santos (Examinador Externo)  
Universidade Federal de Pernambuco

---

Prof. Dr. Viviane Moraes de Oliveira (Examinador Externo)  
Universidade Federal Rural de Pernambuco

---

Prof. Dr. Pablo Martin Rodriguez (Examinador Externo)  
Universidade Federal de Pernambuco

*Dedico este trabalho à Solange da Fonseca Rutz (In Memoriam).*

## **AGRADECIMENTOS**

A Deus por ter me guiado e dado força para superar todas as dificuldades.

Aos meus pais Nelson Leal e Maria do Patrocínio e a minha irmã Márcia Paloma, pelo incentivo e amor incondicional que sempre me deram.

Com a escolha de não citar nomes, para não cometer o pecado do esquecimento, agradeço a todos os meus companheiros do DMAT, por estarem ao meu lado durante esta caminhada, pelas alegrias, tristezas e desafios compartilhados, sendo estes, parte importante na busca em conjunto pelo conhecimento.

Aos meus companheiros do Apartamento 203, vulgo Os Meninos do 11: Leonardo, Jandeilson, Jefferson, Júlio e Willikat. Assim como, aos meus companheiros da sala 216 do DMAT: Matheus, Mikael, Neurandir e Rafael. Com vocês, as discussões, as pausas entre um parágrafo, uma demonstração ou uma solução, é sempre melhor.

A todos os professores da Pós-Graduação em Matemática da Universidade Federal de Pernambuco, que contribuíram nesta minha etapa acadêmica.

À meu orientador Prof. Dr. João Antônio Miranda Gondim, que não mediu esforços para dar apoio e contribuição na construção deste trabalho. Além de acreditar na minha pessoa, em um momento de bastante dificuldade que vivi em meio ao período do doutorado.

Aos membros da banca examinadora pela disponibilidade de participar da avaliação deste trabalho de tese.

A Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelo apoio financeiro, certamente, sem a bolsa isto não seria possível.

E a todos que direta ou indiretamente fizeram parte da minha formação acadêmica.

Muito obrigado!

## RESUMO

Há cerca de 250 anos, matemáticos tem se destinado a investigar modelos epidemiológicos. Após a pandemia da COVID-19, que promete ser um marco do século XXI, a demanda por tais modelos aumentaram consideravelmente. Além de uma questão de saúde pública, um mecanismo de redução da propagação de uma doença, envolve também uma questão econômica. Neste trabalho, iremos apresentar como a teoria de controle ótimo pode ser utilizada para otimizar estratégias farmacológicas e não farmacológicas de contenção de doenças infecciosas, minimizando não apenas o número de infectados, como também os custos de implementação associados. O controle ótimo é calculado numericamente de acordo com o princípio do máximo de Pontryagin e usando a implementação numérica do método de varredura para frente e para trás.

**Palavras-chave:** Modelo SEIRQ; Modelo SEIRV; Controle Ótimo; Estratégias de Contenção;

## **ABSTRACT**

For about 250 years, mathematics has been dedicated to investigating epidemiological models. After the COVID-19 pandemic, which promises to be a landmark of the 21st century, the demand for such models has increased considerably. In addition to being a public health issue, mechanisms for reducing the spread of a disease also involve economic considerations. In this paper, we will present how optimal control theory can be used to modernize pharmacological and non-pharmacological strategies for containing infectious diseases, minimizing not only the number of infected people but also the associated implementation costs. Optimal control is calculated numerically according to Pontryagin's maximum principle and using the numerical algorithm of forward and backward sweep method.

**Keywords:** SEIRQ Model; SEIRV Model; Optimal Control; Containment Strategies;

## LISTA DE FIGURAS

Figura 1 – Uma função com salto que não é contínua por partes e uma função degrau contínua por partes. . . . .	20
Figura 2 – Plot de controle ótimo encontrado usando o método de varredura. . . . .	37
Figura 3 – Plot da variável de estado encontrada usando o método de varredura. . . . .	38
Figura 4 – Plot de controle ótimo com condição de contorno encontrado usando o método de varredura. . . . .	39
Figura 5 – Diagrama compartimental do modelo. . . . .	41
Figura 6 – Os controles ótimos para diferentes durações de quarentena. . . . .	45
Figura 7 – Curvas de infectados com controle ótimo para diferentes durações de quarentena. . . . .	46
Figura 8 – Curva de infectados sem controle ótimo para 120 dias. . . . .	47
Figura 9 – Controle ótimo para atraso de 10 dias no início da quarentena. . . . .	47
Figura 10 – Controle ótimo para atraso de 20 dias no início da quarentena. . . . .	48
Figura 11 – Gráficos de $D(t)/D(B, T)$ para diferentes durações de quarentena. . . . .	49
Figura 12 – Diagrama compartimental do modelo. . . . .	51
Figura 13 – Os controles ótimos para uso de duas vacinas com eficácia de 91% e 51% em diferentes durações de campanha de vacinação. . . . .	55
Figura 14 – Os controles ótimos para uso de duas vacinas com eficácia de 74% e 51% em diferentes durações de campanha de vacinação. . . . .	56
Figura 15 – Os controles ótimos para uso de duas vacinas com eficácia de 67% e 51% em diferentes durações de campanha de vacinação. . . . .	56
Figura 16 – Comportamento da taxa de transmissão para pessoas vacinadas de acordo com a eficácia da vacina. . . . .	56
Figura 17 – Curvas de vacinados para uso de duas vacinas com eficácia de 91% e 51% em diferentes durações de campanha de vacinação. . . . .	57
Figura 18 – Curvas de vacinados para uso de duas vacinas com eficácia de 74% e 51% em diferentes durações de campanha de vacinação. . . . .	57
Figura 19 – Curvas de vacinados para uso de duas vacinas com eficácia de 67% e 51% em diferentes durações de campanha de vacinação. . . . .	57

Figura 20 – Os controles ótimos para uso de duas vacinas com eficácia de 91% e 74% em diferentes durações de campanha de vacinação. . . . .	58
Figura 21 – Os controles ótimos para uso de duas vacinas com eficácia de 91% e 67% em diferentes durações de campanha de vacinação. . . . .	58
Figura 22 – Curvas de vacinados para uso de duas vacinas com eficácia de 91% e 74% em diferentes durações de campanha de vacinação. . . . .	59
Figura 23 – Curvas de vacinados para uso de duas vacinas com eficácia de 91% e 67% em diferentes durações de campanha de vacinação. . . . .	59
Figura 24 – Os controles ótimos para uso de duas vacinas com eficácia de 74% e 67% em diferentes durações de campanha de vacinação. . . . .	60
Figura 25 – Curvas de vacinados para uso de duas vacinas com eficácia de 74% e 67% em diferentes durações de campanha de vacinação. . . . .	60
Figura 26 – Controles ótimos para $\alpha_1 = 0.08, \alpha_2 = 0.064, \varepsilon_1 = 0.432$ e $\varepsilon_2 = 0.54$ . . . . .	62
Figura 27 – Curvas de vacinados para $\alpha_1 = 0.08, \alpha_2 = 0.064, \varepsilon_1 = 0.432$ e $\varepsilon_2 = 0.54$ . . . . .	62
Figura 28 – Controle ótimo $u_2(t)$ para $\theta_2$ fixo. . . . .	63
Figura 29 – Controle ótimo $u_2(t)$ para $\theta_1$ fixo. . . . .	64
Figura 30 – Curvas de infectados para diferentes contextos de uso das vacinas. . . . .	64

## LISTA DE TABELAS

Tabela 1 – Descrição dos grupos . . . . .	40
Tabela 2 – Valores dos parâmetros. . . . .	41
Tabela 3 – Condições iniciais. . . . .	43
Tabela 4 – Número de casos, recuperações e mortes para cada grupo, de acordo com (ESPAÑA, 2020). . . . .	43
Tabela 5 – Calendário ótimo de relaxamento. . . . .	46
Tabela 6 – Calendário ótimo de relaxamento para atraso de 10 dias no início da quarentena. . . . .	48
Tabela 7 – Calendário ótimo de relaxamento para atraso de 20 dias no início da quarentena. . . . .	48
Tabela 8 – Valores dos parâmetros. . . . .	52
Tabela 9 – Descrição das condições iniciais. . . . .	53
Tabela 10 – A distribuição para compra de vacinas de acordo com a Figura 22. . . . .	59
Tabela 11 – A distribuição para compra de vacinas de acordo com a Figura 23. . . . .	60
Tabela 12 – A distribuição para compra de vacinas de acordo com a Figura 25. . . . .	61
Tabela 13 – Valores de $\alpha_1, \alpha_2, \varepsilon_1$ e $\varepsilon_2$ para análise de sensibilidade. . . . .	61
Tabela 14 – A distribuição para compra de vacinas de acordo com a Figura 26. . . . .	62

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> . . . . .	<b>12</b>
<b>2</b>	<b>REFERENCIAL TEÓRICO</b> . . . . .	<b>16</b>
2.1	MODELOS EPIDEMIOLÓGICOS . . . . .	16
2.2	PRINCÍPIO DO MÁXIMO DE PONTRYAGIN . . . . .	20
2.3	EXISTÊNCIA E UNICIDADE DO CONTROLE ÓTIMO . . . . .	27
2.4	MÉTODO DE VARREDURA FRENTE -TRÁS . . . . .	31
<b>3</b>	<b>PROGRAMA DE QUARENTENA</b> . . . . .	<b>40</b>
3.1	O MODELO SEIRQ ESTRUTURADO POR IDADE . . . . .	40
3.2	O PROBLEMA DE OTIMIZAÇÃO . . . . .	43
3.3	RESULTADOS E DISCUSSÕES . . . . .	45
<b>4</b>	<b>CAMPANHA DE VACINAÇÃO</b> . . . . .	<b>50</b>
4.1	O MODELO SEIRV . . . . .	50
4.2	O PROBLEMA DE OTIMIZAÇÃO . . . . .	53
4.3	RESULTADOS E DISCUSSÕES . . . . .	55
<b>5</b>	<b>CONSIDERAÇÕES FINAIS</b> . . . . .	<b>66</b>
	<b>REFERÊNCIAS</b> . . . . .	<b>68</b>
	<b>ANEXO A – CÓDIGO MATLAB PARA MÉTODO DE VARREDURA DO PROBLEMA DE QUARENTENA ÓTIMA</b> . . . . .	<b>76</b>
	<b>ANEXO B – CÓDIGO MATLAB PARA MÉTODO DE VARREDURA DO PROBLEMA DE VACINAÇÃO ÓTIMA</b> . . . . .	<b>87</b>

## 1 INTRODUÇÃO

De acordo com a definição de (MARTCHEVA, 2015, p.01), a epidemiologia estuda os padrões de saúde e doença a nível populacional e seus fatores associados. Segundo (MARTCHEVA, 2015, p.04), o trabalho considerado o primeiro modelo matemático epidemiológico data de 1766, e dessa forma, podemos dizer que há cerca de 250 anos os estudiosos tem se destinado a investigação matemática das doenças e de suas propagações.

Há décadas, muitos relevantes trabalhos vem sendo desenvolvidos, contribuindo com o estudo da propagação das doenças e de medidas de controle das mesmas.

“Hoje, a epidemiologia matemática tem uma presença constante na literatura de pesquisa, e a modelagem matemática está fazendo contribuições significativas para a matemática e a saúde pública” (MARTCHEVA, 2015, p.05).

Os modelos matemáticos se tornaram uma ferramenta importante para investigações e previsões da dinâmica de doenças infecciosas nas últimas décadas, esses trabalhos ajudam a entender como diferentes estratégias de mitigação funcionam, ao lidarmos com uma epidemia, cita (CHOWELL et al., 2016).

Sabemos que muitos modelos epidemiológicos foram importantes nos últimos anos e que a demanda pelo desenvolvimento desses trabalhos foi alta durante a pandemia da COVID-19, evento que promete ser um marco histórico do século XXI (HORTON, 2020). Tal doença é causada pelo vírus SARS-Cov-2 e seus primeiros casos foram relatados em Wuhan, província de Hubei, na China, em 31 de dezembro de 2019, de acordo com o relatório da Organização Mundial da Saúde (OMS) (WHO, 2022f). No início de 2020, os cientistas começaram a modelar a doença para conhecer e entender o vírus, e surgiram trabalhos como (WU; CHEN; CHAN, 2020). Com o objetivo de prever o impacto da propagação do surto surgiram trabalhos como (WU; LEUNG; LEUNG, 2020; FERGUSON et al., 2020; HELLEWELL et al., 2020). Tais estudos poderiam então influenciar as decisões tomadas por governos e organizações. Esse vírus teve uma transmissão rápida e se espalhou rapidamente pelo mundo, e a Organização Mundial da Saúde (OMS) declarou a COVID-19 uma pandemia global em 11 de março de 2020, o que é mostrado no relatório (WHO, 2022a), levando muitos países a implementar medidas para conter a doença.

Uma estratégia inicial eficaz para desacelerar a propagação da COVID-19 consistiu em lockdowns, conforme indicado por muitos estudos como: (BARKUR; KAMATH et al., 2020; DOMENICO et al., 2020; PEPE et al., 2020; LAU et al., 2020). O Brasil começou a adotar tais medidas de quarentena, e em 24 de março, o lockdown parcial foi ordenado pelo governo do estado de São

---

Paulo, conforme o relatório (SãO... , 2023). No entanto, os lockdowns podem ser vistos como muito rigorosos e podem causar muitos efeitos colaterais econômicos e sociais, e estudos que investigam tais efeitos são: (ASAHI et al., 2021; ARNON; RICCO; SMETTERS, 2020; PRICKETT et al., 2020). Estas, foram medidas interessantes no início do surto, quando não havia outras ações disponíveis a fim de evitar o máximo de mortes possível, mas à medida que a doença progride e se torna endêmica ou menos agressiva, outras políticas menos dispendiosas devem ser implementadas. Isolar apenas indivíduos infectados, ao invés de toda a população, poderia ser uma das estratégias menos custosas.

A vacinação é também uma estratégia de contenção, que segundo (ORENSTEIN; AHMED, 2017), reduz significativamente o risco de infecção, casos graves, hospitalização e morte, e que não só fornece proteção individual às pessoas vacinadas, mas também proteção comunitária. De acordo com (WATSON et al., 2022), estima-se que para o caso da COVID-19, uma redução global de 63% no total de mortes foi alcançada durante o primeiro ano de vacinação, e em países de baixa renda, estima-se que o número de mortes evitadas aumentaria em uma porcentagem de 111% se a meta de cobertura vacinal de 40% estabelecida pela Organização Mundial da Saúde (OMS) fosse alcançada por cada país até o final de 2021.

As estratégias de contenção das doenças geram um custo financeiro para o governo: a compra, armazenamento, transporte e distribuição de vacinas, a instalação de postos de vacinação e equipes médicas, a compra, armazenamento, transporte e distribuição de testes, o pagamento de auxílios emergenciais, além do investimento em campanhas de divulgação. Portanto, além de ser um problema de saúde pública, uma campanha de vacinação é uma estratégia de contenção que tem forte dependência econômica e há muitos estudos com o objetivo de investigar esse impacto: (RODRIGUES; PLOTKIN, 2020; RYAN et al., 2006; EHRETH, 2003). O mesmo vale para estratégias de lockdown ou quarentena e os estudos (ASAHI et al., 2021; ARNON; RICCO; SMETTERS, 2020; PRICKETT et al., 2020) comprovam tal questão. Nesse contexto, há a necessidade de desenvolver modelos matemáticos que investiguem as estratégias de contenção, mas que, ao mesmo tempo, consigam regular os custos econômicos associados, para que os resultados possam refletir nas decisões que os governos devem tomar ao implementar essas políticas.

É de extrema importância descobrir qual a melhor maneira de implementar essas estratégias, o que traz a teoria do controle ótimo aos holofotes. Em modelos epidemiológicos, um problema de controle ótimo pode ser usado para definir o valor ideal de um parâmetro importante do modelo, de modo que ele maximize ou minimize uma determinada outra variável presente. Podemos então utilizar o controle ótimo para minimizar um custo financeiro asso-

ciado a uma campanha de contenção de doenças. A teoria de controle ótimo foi aplicada a modelos de doenças como: HIV (JOSHI, 2002; FISTER; LENHART; MCNALLY, 1998; KIRSCHNER; LENHART; SERBIN, 1997), tuberculose (SILVA; TORRES, 2013; JUNG; LENHART; FENG, 2002), influenza (LEE; CHOWELL; CASTILLO-CHÁVEZ, 2010) e epidemias gerais (BEHNCKE, 2000; MATEUS et al., 2017). Desde 2020, alguns trabalhos usam o controle ótimo aplicado à COVID-19: (DJIDJOU-DEMASSE et al., 2020; SHEN et al., 2021; OBSU; BALCHA, 2020; ZAMIR et al., 2021). Gondim e Machado (GONDIM; MACHADO, 2020) usaram controle ótimo para analisar um modelo SEIR com quarentena de pessoas suscetíveis que representou um sistema de lockdown durante a epidemia de COVID-19.

Historicamente, existem muitos trabalhos que estudam modelos matemáticos de vacinação no contexto de diferentes doenças como: hepatite e sarampo (LIU; TAKEUCHI; IWAMI, 2008), gripe (ALEXANDER et al., 2004), tuberculose (NKAMBA THOMAS TIMOTHEE MANGA; MANYOMBE, 2019) e malária (MOHAMMED-AWEL et al., 2021). Recentemente, muitos modelos foram desenvolvidos para investigar diferentes aspectos da vacinação contra SARS-Cov-2 durante a pandemia de COVID-19, como: a distribuição de vacinas (BALCIK et al., 2022; BUBAR et al., 2021), o impacto no número de mortes (WATSON et al., 2022; MOORE et al., 2021), análise de sensibilidade de parâmetros dos modelos (OLIVARES; STAFFETTI, 2021), o comportamento do número reprodutivo básico (PAUL et al., 2023) e outros contextos de investigação.

O objetivo deste trabalho é investigar o controle ótimo aplicado a estratégias de contenção de doenças infecciosas. Nós usamos um modelo SEIR adaptado para isolamento apenas de pessoas infectadas, que está mais próximo da definição adequada de quarentena, e que pode ser alcançado pela triagem de indivíduos sintomáticos, de acordo com (PADULA, 2020; GRASSLY et al., 2020). Tal estratégia é menos dispendiosa, mas o controle ótimo continua sendo importante, isso porque, embora os testes sejam muito mais baratos do que os bloqueios, ainda é necessário ter uma quantidade grande o suficiente disponível todos os dias, e espalhá-la por todo o país. Neste caso, dividimos a população em faixas etárias como em Gondim e Machado (GONDIM; MACHADO, 2020), isso é importante, pois o impacto da doença pode ser diferente para cada grupo. Usamos ainda um modelo SEIR adaptado para campanha de vacinação de pessoas suscetíveis com dois tipos de vacinas que se diferenciam de acordo com a eficácia. Calculamos numericamente o controle ótimo que minimiza não apenas o número de infectados, como também os custos associados às estratégias consideradas.

No primeiro capítulo, veremos na primeira seção como se comportam e como podemos interpretar os modelos matemáticos epidemiológicos compartimentais, enquanto nas demais

seções, nos concentramos em revisar todas as ferramentas matemáticas necessárias para tratar-mos do controle ótimo, da sua existência, unicidade e limitações, assim como os algoritmos usados durante seus cálculos.

No segundo capítulo, nos concentramos na estratégia de quarentena de infectados. O controle descreve o tempo ótimo de entrada em quarentena de um indivíduo infectado após testar positivo. Obtemos um calendário que indica quando podemos relaxar o tempo de entrada em quarentena, tornando possível aliviar a frequência de testes em cada uma das faixas etárias. Finalmente, observamos como os controles reduzem o número de infectados e de mortes em comparação ao mesmo período sem implementação de quarentena.

No terceiro capítulo, trabalhamos com a estratégia de campanha de vacinação. O controle descreve o tempo ótimo da vacinação de um indivíduo suscetível por um determinado tipo de vacina. Os resultados indicam quando o governo deve usar uma vacina ou outra dependendo de sua eficácia, e nestes casos, indicamos a porcentagem de compra para cada vacina. Quando as duas vacinas têm valores de eficácia muito próximos, apresentamos um estudo de sensibilidade para os demais parâmetros que dependem das características das vacinas. Finalmente, analisamos o número de infectados em três contextos: quando usamos apenas a primeira vacina, quando usamos ambas as vacinas de acordo com o controle ótimo e quando usamos apenas a segunda vacina.

## 2 REFERENCIAL TEÓRICO

Neste capítulo, iremos discutir um pouco sobre todas as ferramentas que darão o suporte necessário para desenvolver os estudos a serem apresentados nos capítulos posteriores.

### 2.1 MODELOS EPIDEMIOLÓGICOS

De acordo com (MARTCHEVA, 2015, p.04), o estudo matemático das doenças e de suas propagações tem uma idade de aproximadamente 250 anos, em 1766, Bernoulli<sup>1</sup> publicou em 1766 o estudo que hoje é considerado o primeiro modelo epidemiológico, a modelagem das doenças infecciosas teve significativos avanços no início do século XX, em 1902, Sir Ronald Ross<sup>2</sup> recebeu o Prêmio Nobel por um trabalho pioneiro sobre a malária. Segundo (BRAUER, 2005, p.01), um dos principais trabalhos no estudo matemático da epidemiologia (KERMACK; MCKENDRICK, 1927) foi desenvolvido por Kermack<sup>3</sup> e McKendrick<sup>4</sup> em 1927, de acordo com (MARTCHEVA, 2015, p.05), neste momento, a epidemiologia matemática foi elevada a outro nível, com um modelo determinístico que classificava a população de acordo com as particularidades de cada membro, os modelos compartimentais.

Os modelos matemáticos epidemiológicos compartimentais baseiam-se na ideia de dividir a população total em classes que representam o estado em que os indivíduos se encontram com relação à doença investigada. Em geral, as classes epidemiológicas podem ser caracterizadas da seguinte maneira:

- **suscetíveis:** formada por indivíduos saudáveis, que são vulneráveis a contrair a doença;

<sup>1</sup> Daniel Bernoulli, nascido em 8 de fevereiro de 1700 em Groningen na Holanda e falecido em 17 de março de 1782 em Basel na Suíça, o segundo filho e talvez o mais ilustre da segunda geração da família Bernoulli de matemáticos suíços, um polímata cujos trabalhos sobre matemática, probabilidade e modelagem de doenças infecciosas foram reconhecidos por sua originalidade, profundidade e aplicabilidade prática (GANI, 2001b).

<sup>2</sup> Sir Ronald Ross, nascido em 13 de maio de 1857 em Almora na Índia e falecido em 16 de setembro de 1932 em Londres na Inglaterra, descobriu como o mosquito transmitia a malária e foi o primeiro britânico a receber o Prêmio Nobel (NYE; GIBSON, 1997).

<sup>3</sup> Willian Ogilvy Kermack, nascido em 26 de abril de 1898 em Kirriemuir na Escócia e falecido em 20 de julho de 1970 em Aberdeen também na Escócia, aos 26 anos ficou completamente cego e com muita força de vontade conseguiu continuar sua carreira científica, entre 1927 e 1940 trabalhou com a teoria matemática de propagação das epidemias (ISRAEL; GASCA, 2002).

<sup>4</sup> Anderson Gray McKendrick nascido em 8 Setembro de 1876 e falecido em 30 de Maio de 1943, médico britânico e matemático competente, grande parte do desenvolvimento subsequente da modelagem epidêmica decorre de seu artigo pioneiro, que juntamente com o trabalho inspirador de Ross, pode ser considerado como fundamento da epidemiologia matemática moderna (GANI, 2001a).

- **expostos:** composta por pessoas que contrairam a doença mas ainda não são infecciosos, ou seja, ainda não tem poder de transmissão;
- **infectados:** formada por indivíduos que contrairam a doença e são infecciosos, ou seja, são agentes transmissores;
- **recuperados:** composta por indivíduos que foram removidos da dinâmica de transmissão, seja por terem adquirido imunidade vacinal ou por exposição à doença, seja por motivo de morte;
- **quarentenados:** formada por indivíduos que foram colocados em quarentena, ou seja, foram destinados a uma isolamento, com a finalidade de diminuir a disseminação da doença;
- **vacinados:** composta por pessoas que receberam uma vacina contra a doença.

Neste sentido, os modelos dependem da quantidade de classes que for definida, tal quantidade pode variar de acordo com as características da doença, mas também pode mudar dependendo da finalidade do estudo.

Mais alguns termos técnicos são essenciais para compreender a dinâmica de um modelo epidemiológico, faremos a descrição de alguns destes termos:

- **taxa de transmissão:** é definida segundo (MARTCHEVA, 2015, p.10) como  $\beta = Pc$ , onde  $c$  é a taxa de contato que um infectado tem com outros indivíduos e  $P$  é a probabilidade de que o contato de um infectado com um suscetível resulte em transmissão;
- **período de latência:** é o tempo compreendido desde o momento da infecção até quando o hospedeiro é capaz de transmitir a doença a outro indivíduo;
- **taxa de latência:** a taxa de latência  $\sigma$  é a fração de indivíduos que são expostos por unidade de tempo, geralmente dada por  $\sigma = \frac{1}{T_e}$ , onde  $T_e$  é o período de latência;
- **período de incubação:** é o período entre o momento no qual o indivíduo é exposto ao um agente transmissor e o momento em que o indivíduo começa a apresentar os sintomas da doença, não necessariamente ele coincide com o período de latência (MARTCHEVA, 2015, p.04), em alguns casos, o período de incubação é maior que o período de latência, no caso da gripe, por exemplo, os indivíduos podem transmitir o vírus antes de apresentar sintomas;

- **período de remoção:** é o período no qual o indivíduo fica dentro da classe dos infectados, ou seja, o tempo que o indivíduo gasta para se recuperar após infecção;
- **taxa de remoção:** a taxa de remoção  $\gamma$  é a fração de indivíduos que se recuperam por unidade de tempo, geralmente dada por  $\gamma = \frac{1}{T_r}$ , onde  $T_r$  é o período de remoção;
- **número reprodutivo básico:** de acordo com (MARTCHEVA, 2015, p.21), é o número de infecções secundárias que surgem de um caso primário numa população completamente susceptível.

O número reprodutivo básico, geralmente denotado por  $R_0$ , é um elemento bastante importante no estudo de uma doença, ele representa o número de pessoas que são contaminadas por um único indivíduo no período de tempo que compreende o momento em que ele se torna infeccioso até o momento no qual tal indivíduo se torna um removido. O valor de  $R_0$  indica a magnitude do risco oferecido a população, neste sentido, uma ótima estimativa para este valor é de extrema importância para o controle da propagação da doença, de acordo com (WALLINGA; LIPSITCH, 2007, p.599). Reconhecemos a importância do número reprodutivo básico para o estudo dos modelos matemáticos epidemiológicos, porém, a sua investigação não será interesse deste trabalho, uma vez que, para os modelos abordados aqui, estudos nesse sentido já são bastante difundidos na literatura, são referências: (DIETZ, 1993; ARINO; DRIESSCHE, 2003; BACAËR; OUIFKI, 2007; KATUL et al., 2020; GONDIM, 2021).

Os modelos são geralmente denotados de acordo com as iniciais das classes abordadas no mesmo. A fim de exemplificar, o modelo do tipo **SIR**, vide trabalhos (WEISS, 2013; JI; JIANG, 2014; CHEN et al., 2020; COOPER; MONDAL; ANTONOPOULOS, 2020), é um dos mais simples e pode ser utilizado quando a doença possui um **período de latência** muito pequeno, e daí, a classe de expostos não é considerada, fazendo com que todos os indivíduos que contraem a doença sejam considerados infecciosos imediatamente. Segundo (MARTCHEVA, 2015, p.10), o primeiro estudo com este tipo de modelo foi o trabalho desenvolvido por Kermack e McKendrick citado anteriormente. O modelo SIR clássico é

$$\begin{cases} \frac{dS}{dt} = -\frac{\beta SI}{N} \\ \frac{dI}{dt} = \frac{\beta SI}{N} - \gamma I \\ \frac{dR}{dt} = \gamma I \end{cases}, \quad (2.1)$$

onde  $\beta$  é dito a taxa de transmissão da doença e  $\gamma$  é dito o taxa de remoção.

O modelo **SEIR**, vide trabalhos (LI et al., 1999; ANNAS et al., 2020; CARCIONE et al., 2020), pode ser utilizado quando estamos interessados em considerar a classe dos expostos, mas ainda não estamos interessados em investigar um possível sistema de quarentena ou de vacinação. Inclusive, os modelos abordados a seguir no nosso trabalho serão adaptações do modelo SEIR. O modelo SEIR clássico é

$$\left\{ \begin{array}{l} \frac{dS}{dt} = -\frac{\beta SI}{N} \\ \frac{dE}{dt} = \frac{\beta SI}{N} - \sigma E \\ \frac{dI}{dt} = \sigma E - \gamma I \\ \frac{dR}{dt} = \gamma I \end{array} \right. , \quad (2.2)$$

onde  $\sigma$  é dito a taxa de latência.

Quando estamos diante de uma situação na qual vamos considerar uma reinfecção pela doença, podemos usar, por exemplo, um modelo **SIRS**, vide (JIN; WANG; XIAO, 2007; LI; TSAI; YANG, 2014; CAI; KANG; WANG, 2017), ou **SEIRS** vide (COOKE; DRIESSCHE, 1996; LIU; ZHANG, 2011; BJØRNSTAD et al., 2020). Nos próximos capítulos, vamos abordar sistemas **SEIRQ** e **SVEIR**.

Segundo (LLOYD, 2009, p.123), uma importante tarefa dentro da teoria dos modelos epidemiológicos é encontrar os valores dos parâmetros a serem utilizados, como por exemplo a taxa de transmissão ou o número reprodutivo básico, a este processo damos o nome de **estimativa dos parâmetros**. Tais valores devem ser calculados de modo que, a curva em função do tempo referente a população de cada compartimento, esteja cada vez mais de acordo com os dados reais da doença, para que o modelo se encaixe da melhor forma com a situação real. A partir deste momento, os modelos matemáticos tem capacidade de prever os próximos passos a serem enfrentados durante a propagação de uma doença.

A estimativa de parâmetros é uma tarefa que não será contemplada pelo objetivo do nosso trabalho, nós usaremos parâmetros disponíveis na literatura e que foram estimados utilizando métodos difundidos, como por exemplo, o método dos mínimos quadrados e o método da cadeia de Markov Monte Carlo. O método dos mínimos quadrados se resume em encontrar os parâmetros que minimizam a soma das distâncias quadradas entre os dados previstos pelo modelo e os dados reais, mais detalhes podem ser vistos em (CHAPRA, 2018, p.359). O método da cadeia de Markov Monte Carlo é um método estatístico que se resume em estabelecer um intervalo para os possíveis valores dos parâmetros, gerar valores aleatórios dentro deste intervalo usando cadeias

de Markov e identificar estimativas ideais a partir das amostras aleatórias usando a integração de Monte Carlo, mais detalhes estão disponíveis em (HAMRA; MACLEHOSE; RICHARDSON, 2013, p.629).

Muitos modelos epidemiológicos partem de hipóteses distintas dependendo do objetivo principal do estudo. Algumas dessas hipóteses podem ser: desconsiderar nascimentos e/ou mortes, considerar quarentenado todo aquele indivíduo que testa positivo para a doença, considerar o número de suscetíveis igual a população total, etc.

## 2.2 PRINCÍPIO DO MÁXIMO DE PONTRYAGIN

**Definição 2.1** *Seja  $I \subseteq \mathbb{R}$  um intervalo. Uma função de valor finito  $u : I \rightarrow \mathbb{R}$  é **contínua por partes** se for contínua em  $I$  com possível exceção de um número finito de pontos de  $I$  e com a condição de que o valor de  $u(t)$  seja igual ao seu limite à esquerda ou à direita em cada  $t \in I$ .*

Para entender melhor a Definição 2.1, uma função que é contínua por partes pode ter descontinuidades com saltos entre um segmento contínuo e outro, mas não pode ter descontinuidade onde o valor da função salta para um ponto isolado.

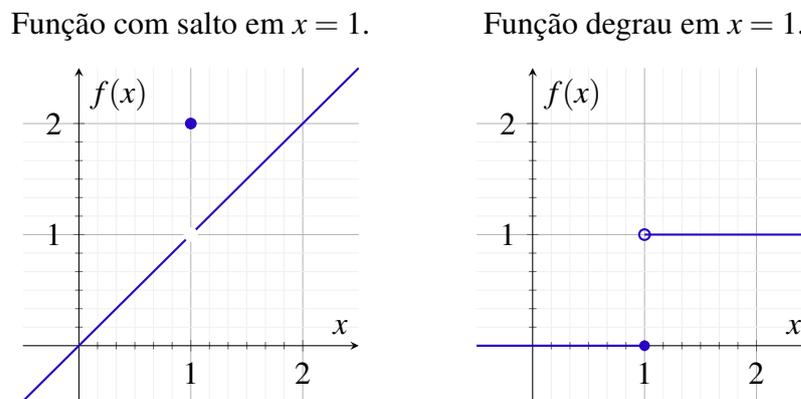


Figura 1 – Uma função com salto que não é contínua por partes e uma função degrau contínua por partes.

**Definição 2.2** *Uma função  $x : I \rightarrow \mathbb{R}$  que é diferenciável em  $I$  tal que sua derivada  $x'$  é contínua, é dita **continuamente diferenciável**.*

**Definição 2.3** *Uma função  $x : I \rightarrow \mathbb{R}$  é **continuamente diferenciável por partes** se for continuamente diferenciável em  $I$  com exceção de um número finito de pontos de  $I$ .*

Vamos iniciar a apresentação do problema de controle ótimo que abordaremos. Considere então  $g : \mathbb{R}^3 \rightarrow \mathbb{R}$  uma função continuamente diferenciável em suas três variáveis, e a equação

diferencial

$$x'(t) = g(t, x(t), u(t)). \quad (2.3)$$

Chamaremos  $x(t)$  de **variável de estado** e a função  $u(t)$  será dita a **variável de controle**. Perceba que à medida que o controle muda, a variável de estado muda, pois sua derivada depende do controle. Ademais, nestas condições, segundo (LOGEMANN; RYAN, 2014, p.203),  $x(t)$  será continuamente diferenciável por partes.

Nosso objetivo é encontrar um controle  $u(t)$  e a variável de estado associada  $x(t)$  de forma a maximizar ou minimizar o funcional

$$J[u(t)] = \int_{t_0}^{t_1} f(t, x(t), u(t)) dt, \quad (2.4)$$

que está sujeito a (2.3) com  $x(t_0) = x_0$  e  $x(t_1)$  sem valor determinado, onde  $f : \mathbb{R}^3 \rightarrow \mathbb{R}$  é uma função continuamente diferenciável em suas três variáveis. O controle  $u(t)$  que contempla o nosso objetivo é chamado de **controle ótimo**.

O problema de controle ótimo está posto. A seguir, vamos explorar um pouco à cerca de condições necessárias e suficientes que o controle ótimo deve satisfazer, condições de limitação impostas ao controle ótimo, além de discutir o uso de métodos computacionais utilizados para calculá-lo.

Uma condição necessária para a existência de um controle ótimo  $u(t)$  do problema (2.3)-(2.4) foi desenvolvida em (PONTRYAGIN et al., 1986, p.17) e ficou conhecida como o **princípio do máximo de Pontryagin**<sup>5</sup> (PMP). Pontryagin introduziu a noção de função adjunta, a qual tem um propósito semelhante aos multiplicadores de Lagrange do cálculo. Resumidamente, o princípio garante a existência de uma nova equação diferencial que quando resolvida juntamente com a equação diferencial inicial do problema, nos oferece o comportamento do controle ótimo.

**Teorema 2.1 (Princípio do Máximo de Pontryagin)** *Seja  $u^*(t)$  um controle ótimo para o problema (2.3)-(2.4). Então existe uma variável adjunta  $\lambda : [t_0, t_1] \rightarrow \mathbb{R}$  diferenciável por partes, de forma que vale a condição ótima*

$$\left. \frac{\partial H}{\partial u} \right|_{u=u^*} = 0, \quad (2.5)$$

<sup>5</sup> Lev Semyonovich Pontryagin, nascido em 3 de setembro de 1908 em Moscou e falecido em 3 de maio de 1988 também em Moscou, pode-se dizer que Pontryagin inicialmente estudou vários problemas relacionados à homologia e à homotopia, e posteriormente se dedicou a estudos sobre equações diferenciais e problemas relacionados, o próprio Pontryagin explicou a mudança de assunto pela necessidade de fazer algo que fosse mais “útil para a sociedade”, por recomendações de alguns dirigentes do Instituto Steklov de Matemática, desejo de alguns de seus colaboradores e, por fim, pela sua comunicação com engenheiros (ANOSOV, 2010).

onde  $H$  é o Hamiltoniano

$$H = f(t, x(t), u(t)) + \lambda(t)g(t, x(t), u(t)), \quad (2.6)$$

que atende a equação adjunta

$$\lambda'(t) = -\frac{\partial H(t, x^*(t), u^*(t), \lambda(t))}{\partial x}, \quad (2.7)$$

com condição de transversalidade

$$\lambda(t_1) = 0. \quad (2.8)$$

**Demonstração.** Assuma que existe um controle ótimo e denote-o por  $u^*(t)$  com  $x^*(t)$  sendo o estado correspondente. Vamos supor sem perda de generalidade que este controle ótimo maximiza o funcional (2.4). Neste caso, segue que  $J[u(t)] \leq J[u^*(t)] < \infty$  para todas as demais funções controle  $u(t)$ . Vamos agora estabelecer uma variação para o controle ótimo, seja  $\varepsilon \in [-\varepsilon_0, \varepsilon_0]$  para  $\varepsilon_0 > 0$  e  $h : [t_0, t_1] \rightarrow \mathbb{R}$  uma função contínua por partes, considere

$$u_\varepsilon(t) := u^*(t) + \varepsilon h(t), \quad (2.9)$$

e denote por  $x_\varepsilon(t)$  o estado associado que satisfaz  $x'_\varepsilon(t) = g(t, x_\varepsilon(t), u_\varepsilon(t))$  com  $x_\varepsilon(t) = x_0$  para que todas as soluções de estado partam do mesmo ponto  $x_0$ .

Podemos ver ainda que a partir de (2.9),

$$\left. \frac{\partial u_\varepsilon(t)}{\partial \varepsilon} \right|_{\varepsilon=0} = \lim_{\varepsilon \rightarrow 0} \frac{u^*(t) + \varepsilon h(t) - u_0(t)}{\varepsilon} = h(t). \quad (2.10)$$

Considere agora  $\lambda : [t_0, t_1] \rightarrow \mathbb{R}$  uma função diferenciável a ser determinada, e note que

$$\int_{t_0}^{t_1} f(t, x(t), u(t)) dt = \int_{t_0}^{t_1} \left[ f(t, x(t), u(t)) + \lambda(t)g(t, x(t), u(t)) - \lambda(t)x'(t) \right] dt, \quad (2.11)$$

onde  $u(t)$  é qualquer função controle.

Integrando por partes a última parcela do lado direito de (2.11), temos

$$-\int_{t_0}^{t_1} \lambda(t)x'(t) dt = \lambda(t_0)x_0 - \lambda(t_1)x(t_1) + \int_{t_0}^{t_1} \lambda'(t)x(t) dt \quad (2.12)$$

Substituindo (2.12) em (2.11), temos

$$\begin{aligned} \int_{t_0}^{t_1} f(t, x(t), u(t)) dt &= \int_{t_0}^{t_1} \left[ f(t, x(t), u(t)) + \lambda(t)g(t, x(t), u(t)) + \lambda'(t)x(t) \right] dt \\ &\quad + \lambda(t_0)x_0 - \lambda(t_1)x(t_1). \end{aligned} \quad (2.13)$$

Como (2.13) é válida para qualquer função controle, podemos tomar  $u(t) = u_\varepsilon(t)$ , teremos então

$$J[u_\varepsilon(t)] = \int_{t_0}^{t_1} \left[ f(t, x_\varepsilon(t), u_\varepsilon(t)) + \lambda(t)g(t, x_\varepsilon(t), u_\varepsilon(t)) + \lambda'(t)x_\varepsilon(t) \right] dt + \lambda(t_0)x_0 - \lambda(t_1)x_\varepsilon(t_1). \quad (2.14)$$

Agora, considere a aplicação

$$\begin{aligned} T : [-\varepsilon_0, \varepsilon_0] &\longrightarrow \mathbb{R} \\ \varepsilon &\longmapsto J(u_\varepsilon) \end{aligned},$$

e note que  $T(\varepsilon) = J[u_\varepsilon(t)] \leq J[u^*(t)] = T(0)$ , pois o controle ótimo maximiza o problema. Portanto,  $\varepsilon = 0$  é ponto de máximo de  $T$ , logo é ponto crítico, e assim, temos

$$0 = T'(0) = \left. \frac{d}{d\varepsilon} J[u_\varepsilon(t)] \right|_{\varepsilon=0}. \quad (2.15)$$

Podemos usar (2.14) e (2.15) para concluir que

$$\begin{aligned} 0 = \int_{t_0}^{t_1} \left. \frac{\partial}{\partial \varepsilon} \left[ f(t, x_\varepsilon(t), u_\varepsilon(t)) + \lambda(t)g(t, x_\varepsilon(t), u_\varepsilon(t)) + \lambda'(t)x_\varepsilon(t) \right] \right|_{\varepsilon=0} dt \\ - \lambda(t_1) \left. \frac{\partial}{\partial \varepsilon} x_\varepsilon(t_1) \right|_{\varepsilon=0}. \end{aligned}$$

Aplicando a regra da cadeia para  $f$  e  $g$ , segue que

$$\begin{aligned} 0 = \int_{t_0}^{t_1} \left[ f_x(t, x^*(t), u^*(t)) \frac{\partial x_\varepsilon}{\partial \varepsilon} + f_u(t, x^*(t), u^*(t)) \frac{\partial u_\varepsilon}{\partial \varepsilon} + \lambda'(t) \frac{\partial x_\varepsilon}{\partial \varepsilon}(t) \right] dt \\ + \int_{t_0}^{t_1} \lambda(t) \left[ g_x(t, x^*(t), u^*(t)) \frac{\partial x_\varepsilon}{\partial \varepsilon} + g_u(t, x^*(t), u^*(t)) \frac{\partial u_\varepsilon}{\partial \varepsilon} \right] dt - \lambda(t_1) \left. \frac{\partial x_\varepsilon}{\partial \varepsilon}(t_1) \right|_{\varepsilon=0}, \end{aligned}$$

pois, como diferenciamos no ponto  $\varepsilon = 0$ , as derivas parciais  $f_x, f_u, g_x, g_u$  são calculadas ao longo de  $(t, x^*, u^*)$ .

Rearranjando os termos e usando (2.10), temos

$$\begin{aligned} 0 = \int_{t_0}^{t_1} \left[ f_x(t, x^*(t), u^*(t)) + \lambda(t)g_x(t, x^*(t), u^*(t)) + \lambda'(t) \right] \left. \frac{\partial x_\varepsilon}{\partial \varepsilon} \right|_{\varepsilon=0} dt \\ + \int_{t_0}^{t_1} h(t) \left[ f_u(t, x^*(t), u^*(t)) + \lambda(t)g_u(t, x^*(t), u^*(t)) \right] dt - \lambda(t_1) \left. \frac{\partial x_\varepsilon}{\partial \varepsilon}(t_1) \right|_{\varepsilon=0}. \end{aligned} \quad (2.16)$$

Agora, vamos escolher a função  $\lambda(t)$  de forma a zerar a primeira e a última parcela do lado direito de (2.16). Tomemos então

$$\lambda'(t) = - \left[ f_x(t, x^*(t), u^*(t)) + \lambda(t)g_x(t, x^*(t), u^*(t)) \right], \quad (2.17)$$

além de

$$\lambda(t_1) = 0, \quad (2.18)$$

estas serão a equação adjunta e a condição de transversalidade.

Dessa forma, (2.16) se reduz a

$$\int_{t_0}^{t_1} h(t) \left[ f_u(t, x^*(t), u^*(t)) + \lambda(t) g_u(t, x^*(t), u^*(t)) \right] dt = 0. \quad (2.19)$$

Sendo  $f$  e  $g$  continuamente diferenciáveis, podemos escolher

$$h(t) = f_u(t, x^*(t), u^*(t)) + \lambda(t) g_u(t, x^*(t), u^*(t)).$$

Logo, teremos

$$\int_{t_0}^{t_1} \left[ f_u(t, x^*(t), u^*(t)) + \lambda(t) g_u(t, x^*(t), u^*(t)) \right]^2 dt = 0,$$

o que implica em

$$f_u(t, x^*(t), u^*(t)) + \lambda(t) g_u(t, x^*(t), u^*(t)) = 0. \quad (2.20)$$

Sendo  $H = f + \lambda g$ , a condição ótima (2.5) segue de (2.20). A equação adjunta e a condição de transversalidade seguem de (2.17) e (2.18), respectivamente. ■

Note que no PMP, a função  $H$  é realmente uma função Hamiltoniana, uma vez que,

$$\lambda'(t) = -\frac{\partial H}{\partial x} \quad \text{e} \quad x'(t) = \frac{\partial H}{\partial \lambda}.$$

Ademais, esta prova para o PMP foi feita considerando que o controle ótimo maximiza o problema, todavia poderíamos considerar que o controle é minimizante e os passos para a demonstração seguiriam de forma análoga.

Finalmente, como descobrir o comportamento do controle ótimo a partir do PMP? Bem,  $(t, x^*(t), u^*(t))$  é ponto crítico do Hamiltoniano  $H(t, x(t), u(t))$ , daí, usando a condição ótima (2.5), podemos escrever  $u^*(t)$  em função do estado  $x^*(t)$  e da variável adjunta  $\lambda(t)$ . Dessa forma, a equação de estado (2.3) e a equação adjunta (2.7), nos oferecem um sistema de equações diferenciais ordinárias nas variáveis  $x^*(t)$  e  $\lambda(t)$ . Basta então resolver tal sistema e calcular o controle ótimo  $u^*(t)$ .

Em alguns casos, o controle ótimo pode ser calculado analiticamente, vejamos abaixo um exemplo.

**Exemplo 2.1** *Encontre a variável adjunta, o controle ótimo  $u^*(t)$  e o estado ótimo associado  $x^*(t)$ , que minimizam o funcional*

$$J[u(t)] = \int_1^2 t u^2(t) + t^2 x(t) dt,$$

que está sujeito a equação de estado

$$x'(t) = -u(t), \quad x(1) = 1.$$

**Solução:** Montando a função Hamiltoniana, temos

$$H(t, x(t), u(t)) = tu^2(t) + t^2x(t) - \lambda(t)u(t),$$

onde  $\lambda(t)$  é a variável adjunta que satisfaz

$$\lambda'(t) = -t^2, \quad \lambda(2) = 0.$$

A solução da equação adjunta é  $\lambda(t) = -\frac{t^3}{3} + K$ , com  $K$  constante de integração. Usando a condição de transversalidade, temos  $K = \frac{8}{3}$ . Dessa forma, a variável adjunta será

$$\lambda(t) = -\frac{t^3}{3} + \frac{8}{3}.$$

A condição ótima nos oferece  $0 = 2tu^*(t) - \lambda(t)$ , ou seja,  $u^*(t) = \frac{\lambda(t)}{2t}$ . Substituindo o valor de  $\lambda(t)$ , temos o controle ótimo

$$u^*(t) = \frac{-t^3 + 8}{6t}.$$

A restrição de estado nos dá  $x'(t) = \frac{t^3 - 8}{6t}$ , que tem solução  $x^*(t) = \frac{1}{6} \left( \frac{t^3}{3} - 8 \ln(t) \right) + C$ , onde  $C$  é também uma constante de integração. Usando o estado inicial  $x(1) = 1$ , temos então  $C = \frac{17}{18}$ . Dessa forma, o estado ótimo será

$$x^*(t) = \frac{1}{6} \left( \frac{t^3}{3} - 8 \ln(t) + \frac{17}{3} \right).$$

□

Em problemas de maior complexidade, o uso do PMP nos oferece sistemas de equações diferenciais muito complicados de resolver analiticamente, neste caso, recorreremos ao uso de métodos computacionais para a resolução. Isso será visto na Seção 2.4.

Em determinados casos, seria importante impor uma limitação para o controle ótimo, como por exemplo, quando o controle representa alguma quantidade química ou algum percentual. Considere novamente o problema (2.3)-(2.4) e adicione ao mesmo, a condição de contorno para o controle

$$a \leq u(t) \leq b, \quad a, b \in \mathbb{R}. \quad (2.21)$$

Uma versão do PMP como condição necessária para a existência do controle ótimo continua sendo válida para os controles limitados, a prova é essencialmente a mesma, com a diferença de que todos os controles  $u_\varepsilon$  envolvidos, são limitados.

Considerando novamente o Hamiltoniano  $H(t, x, u, \lambda)$ , dado um problema com condição de contorno (2.21), seja  $\bar{u}(t)$  o valor de  $u(t)$  no qual  $\frac{\partial H}{\partial u} = 0$ , então seguindo (LENHART; WORKMAN, 2007) vale a caracterização

$$u^*(t) = \min(b, \max(a, \bar{u}(t))). \quad (2.22)$$

Em outras palavras, o que acontece em (2.22) é um truncamento do valor de  $\bar{u}(t)$ , ou seja, no intervalo de tempo no qual este valor excede  $a$  por baixo, faça  $u^*(t) = a$  e no intervalo de tempo no qual este valor excede  $b$  por cima, faça  $u^*(t) = b$ .

A teoria até aqui desenvolvida visando o problema de controle ótimo (2.3)-(2.4) pode ser estendida para múltiplas variáveis de controle e de estado.

Considere funções  $g_i : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$  continuamente diferenciáveis em todas as suas variáveis e o sistema de equações diferenciais

$$\begin{aligned} x'_i(t) &= g_i(t, x_1(t), \dots, x_n(t), u_1(t), \dots, u_m(t)), \\ x_i(t_0) &= x_{i0}, \quad i = 1, \dots, n. \end{aligned} \quad (2.23)$$

Nosso objetivo será o de encontrar controles ótimos  $u_j(t)$  com  $j = 1, \dots, m$ , e as variáveis de estado associadas  $x_i(t)$  com  $i = 1, \dots, n$ , de forma a maximizar ou minimizar o funcional

$$J[u_1(t), \dots, u_m(t)] = \int_{t_0}^{t_1} f(t, x_1(t), \dots, x_n(t), u_1(t), \dots, u_m(t)) dt, \quad (2.24)$$

que está sujeito a (2.23) com  $x_i(t_1)$  sem valor determinado para  $i = 1, \dots, n$ , onde a função  $f : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$  é continuamente diferenciável em todas as suas variáveis.

Uma versão do PMP também é válida para este caso:

**Teorema 2.2 (Princípio do Máximo de Pontryagin para múltiplas variáveis)** *Seja  $u_j^*(t)$  um controle ótimo para o problema (2.23)-(2.24). Existem  $n$  variáveis adjuntas  $\lambda_i : [t_0, t_1] \rightarrow \mathbb{R}$  diferenciáveis por partes, de forma que valem as condições ótimas*

$$\left. \frac{\partial H}{\partial u_j} \right|_{u=u_j^*} = 0, \quad j = 1, \dots, m, \quad (2.25)$$

onde  $H$  é o Hamiltoniano

$$H = f(t, x_1(t), \dots, x_n(t), u_1(t), \dots, u_m(t)) + \sum_{i=1}^n \lambda_i(t) g_i(t, x_1(t), \dots, x_n(t), u_1(t), \dots, u_m(t)), \quad (2.26)$$

que atende as *equações adjuntas*

$$\lambda_i'(t) = -\frac{\partial H(t, x_i^*(t), u_j^*(t), \lambda(t))}{\partial x_i}, \quad i = 1, \dots, n, \quad (2.27)$$

com *condições de transversalidade*

$$\lambda_i(t_1) = 0, \quad i = 1, \dots, n. \quad (2.28)$$

**Demonstração.** A demonstração é garantida usando essencialmente os mesmos argumentos apresentados na demonstração do Teorema 2.1. ■

Se adicionamos condições de contorno para os controles em múltiplas variáveis

$$a_j \leq u_j(t) \leq b_j, \quad a_j, b_j \in \mathbb{R}, \quad j = 1, \dots, m, \quad (2.29)$$

a caracterização (2.22) é válida para cada uma das funções controle, ou seja,

$$u_j^*(t) = \min(b_j, \max(a_j, \bar{u}_j(t))), \quad j = 1, \dots, m. \quad (2.30)$$

O problema de controle ótimo para múltiplas variáveis está posto. Por simplicidade de notação, a partir de agora, adotaremos uma notação vetorial:

$$u(t) := \vec{u}(t) = [u_1(t), \dots, u_m(t)],$$

$$x(t) := \vec{x}(t) = [x_1(t), \dots, x_n(t)],$$

$$x_0 := \vec{x}_0 = [x_{10}(t), \dots, x_{n0}]$$

$$g(t, x(t), u(t)) := \vec{g}(t, x(t), u(t)) = [g_1(t, x(t), u(t)), \dots, g_n(t, x(t), u(t))].$$

### 2.3 EXISTÊNCIA E UNICIDADE DO CONTROLE ÓTIMO

Na Seção 2.2, assumimos a existência do controle ótimo e abordamos condições necessárias acerca do problema (2.3)-(2.4), porém, podemos encontrar casos em que as condições necessárias podem ser desenvolvidas mesmo quando o problema não tem solução. Além disso, pode haver casos em que o controle ótimo maximiza ou minimiza o funcional (2.4) com valores  $-\infty$  ou  $+\infty$ , tais casos não nos interessam, vejamos um exemplo.

**Exemplo 2.2** Considere o problema de maximização com

$$J[u(t)] = \int_0^1 x(t) + u(t) dt,$$

sujeito a equação de estado

$$x'(t) = 1 - u^2(t), \quad x(0) = 1.$$

Usando o PMP para tal problema, encontramos a equação adjunta  $\lambda'(t) = -1$  com  $\lambda(1) = 0$ , e daí,  $\lambda(t) = 1 - t$ . Portanto, o estado e o controle ótimo serão

$$x^*(t) = t - \frac{1}{4(1-t)} + \frac{5}{4}, \quad u^*(t) = \frac{1}{2(1-t)}.$$

Todavia, temos para  $u = u^*$

$$J[u^*(t)] = \int_0^1 t - \frac{1}{2(1-t)} + \frac{5}{4} dt = \infty.$$

Nesta seção, vamos apresentar condições para a existência e também para a unicidade do controle ótimo.

**Definição 2.4** Seja  $X$  um espaço topológico, uma função  $f : X \rightarrow \mathbb{R}$  é uma função semicontínua superiormente (respectivamente, semicontínua inferiormente) em  $a \in X$  se para cada número real  $h > f(a)$  (respectivamente  $h < f(a)$ ) se existe uma vizinhança  $V$  de  $a$  tal que para qualquer  $x \in V$  temos  $h > f(x)$  (respectivamente  $h < f(x)$ ).

Em outras palavras,  $f$  é semicontínua superiormente (respectivamente, semicontínua inferiormente) se quando  $x$  está próximo de  $a$ ,  $f(x)$  está próximo à direita de  $f(a)$  (respectivamente,  $f(x)$  está próximo à esquerda de  $f(a)$ ).

**Definição 2.5** Seja  $X$  um espaço vetorial normado, um subconjunto  $S \subset X$  é convexo se para todos  $x, y \in S$  e todo  $t \in [0, 1]$ , temos  $(1-t)x + ty \in S$ .

Em outras palavras,  $S$  é convexo se todo segmento de reta de extremos  $x, y \in S$ , está contido totalmente em  $S$ .

**Definição 2.6** Uma  $\sigma$ -álgebra em um conjunto  $X$  é uma coleção  $\Lambda$  de subconjuntos de  $X$  que contém o conjunto vazio e que é fechada por união enumerável e complemento. O par  $(M, \Lambda)$  é dito um **espaço mensurável**. Uma função  $f : (X_1, \Lambda_1) \rightarrow (X_2, \Lambda_2)$  é dita  $\Lambda_1 \Lambda_2$ -mensurável se  $f^{-1}(A) \in \Lambda_1$  para todo  $A \in \Lambda_2$ . Em particular, uma função  $f : (X, \Lambda) \rightarrow \mathbb{R}$  é dita  $\Lambda$ -mensurável se para todo  $\alpha \in \mathbb{R}$ , o conjunto  $f^{-1}((\alpha, \infty))$  é  $\Lambda$ -mensurável. Ademais,  $f$  é **Lebesgue mensurável** quando  $\Lambda$  é a  $\sigma$ -álgebra de Lebesgue.

Em poucas palavras, função Lebesgue mensurável é aquela que tem propriedades suficientes tal que seja possível tomar sua integral. Mais detalhes sobre mensurabilidade, como por exemplo, a caracterização da  $\sigma$ -álgebra de Lebesgue, podem ser vistos em (BARTLE, 2014, p.144).

**Teorema 2.3** *Considere o conjunto*

$$U = \{u(t) / a \leq u(t) \leq b, 0 \leq t \leq T, u(t) \text{ é Lebesgue mensurável}\}$$

e assumamos que

(i) Existe  $M > 0$  tal que  $\|x(t)\| \leq M$  para toda  $u \in U$  e  $0 \leq t \leq T$ ;

(ii) A função  $f(t, x(t), u(t))$  é semicontínua inferiormente;

(iii) O conjunto

$$D = \{(y^0(t), y(t)) / \exists v(t) \in U, y(t) = g(t, x(t), v(t)), y^0(t) \geq f(t, x(t), v(t))\}$$

é convexo para  $(t, x(t)) \in [0, T] \times \{\|x(t)\| \leq M\}$ ;

Então existe um controle ótimo  $u^*(t) \in U$  para o problema (2.23)-(2.24).

**Demonstração.** A demonstração deste teorema não será o foco do nosso trabalho, mas está disponível em (MACKI; STRAUSS, 1982, p.91). ■

Na Seção 2.2, elaboramos nosso problema de controle ótimo e usamos o PMP para controles que sejam contínuos por partes e note que o Teorema 2.4 garante a existência de um controle que é Lebesgue mensurável, no entanto, não há problemas quanto a esta pequena discordância, caso isto ocorra, podemos remodelar o PMP e contornar esta situação, de acordo com (LENHART; WORKMAN, 2007, p.25).

A unicidade do controle ótimo depende da unicidade do sistema formado pela equação de estado e pela equação adjunta. Mais especificamente, depende da unicidade do sistema

$$\begin{cases} x'(t) = g(t, x^*(t), u^*(t)) \\ \lambda'(t) = -f_x(t, x^*(t), u^*(t)) - \lambda(t)g_x(t, x^*(t), u^*(t)) \\ x(t_0) = x_0, \quad \lambda(t_1) = 0. \end{cases}$$

Neste caso, a unicidade é garantida pelo seguinte resultado.

**Teorema 2.4** *Considere o problema de valor de contorno*

$$\begin{cases} x'(t) = F(t, x(t), y(t)) \\ y'(t) = G(t, x(t), y(t)) \\ x(0) = x_0, \quad y(T) = y_T. \end{cases}, \quad (2.31)$$

onde  $x, y \in \mathbb{R}^n$  e  $F, G: \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^n$  são funções contínuas. Assuma que  $F, G$  são limitadas e Lipschitz com relação as variáveis  $x, y$  e constante  $C > 0$ . Então, as soluções do sistema (2.31) são únicas para  $T$  suficientemente pequeno.

**Demonstração.** Suponha que o problema (2.31) tem duas soluções  $(x_1(t), y_1(t))$  e  $(x_2(t), y_2(t))$ .

Note que podemos escrever

$$x_1(t) = x_0 + \int_0^t F(t, x_1(r), y_1(r)) dr, \quad x_2(t) = x_0 + \int_0^t F(t, x_2(r), y_2(r)) dr$$

Daí, como  $F(t, x(t), y(t))$  é Lipschitz, temos

$$\begin{aligned} \|x_1(t) - x_2(t)\| &\leq \int_0^t |F(t, x_1(r), y_1(r)) - F(t, x_2(r), y_2(r))| dr \\ &\leq C \int_0^t (\|x_1(r) - x_2(r)\| + \|y_1(r) - y_2(r)\|) dr. \end{aligned} \quad (2.32)$$

Como  $G(t, x(t), y(t))$  é também uma função Lipschitz, de maneira análoga, o mesmo vale para  $y_1(t)$  e  $y_2(t)$ , ou seja,

$$\begin{aligned} \|y_1(t) - y_2(t)\| &\leq \int_t^T |G(t, x_1(r), y_1(r)) - G(t, x_2(r), y_2(r))| dr \\ &\leq C \int_t^T (\|x_1(r) - x_2(r)\| + \|y_1(r) - y_2(r)\|) dr. \end{aligned} \quad (2.33)$$

Somando (2.32) e (2.33), segue que

$$\|x_1(t) - x_2(t)\| + \|y_1(t) - y_2(t)\| \leq C \int_0^T (\|x_1(r) - x_2(r)\| + \|y_1(r) - y_2(r)\|) dr.$$

Como a norma é uma função contínua, usando o teorema do valor médio para integrais, concluímos que existe  $\xi \in (0, T)$  tal que

$$\|x_1(t) - x_2(t)\| + \|y_1(t) - y_2(t)\| < TC (\|x_1(\xi) - x_2(\xi)\| + \|y_1(\xi) - y_2(\xi)\|).$$

Se tomarmos  $T$  suficientemente pequeno de forma que  $TC < 1$ , teremos

$$\|x_1(t) - x_2(t)\| + \|y_1(t) - y_2(t)\| \leq \|x_1(\xi) - x_2(\xi)\| + \|y_1(\xi) - y_2(\xi)\|,$$

para todo  $t \in [0, T]$ . O que seria uma contradição, portanto, só pode haver uma única solução do problema (2.31). ■

## 2.4 MÉTODO DE VARREDURA FRENTE -TRÁS

Citamos anteriormente que o PMP pode nos oferecer condições nas quais seria complicado calcular o controle analiticamente. Por exemplo, considere o problema:

**Exemplo 2.3** *Encontre a variável adjunta, o controle ótimo  $u^*(t)$  e o estado ótimo associado  $x^*(t)$ , que maximizam o funcional*

$$J[u(t)] = \int_0^1 x(t) - u^2(t) dt,$$

que está sujeito a equação de estado

$$x'(t) = -\frac{x^2(t)}{2} + u(t), \quad x(0) > -2.$$

Montando a função Hamiltoniana para o exemplo, temos

$$H(t, x(t), u(t)) = x(t) - u^2(t) - \lambda(t) \frac{x^2(t)}{2} + \lambda(t)u(t)$$

onde  $\lambda(t)$  é a variável adjunta que satisfaz

$$\lambda'(t) = \lambda(t)x(t) - 1, \quad \lambda(1) = 0.$$

A partir deste momento, se torna um pouco complicado resolver a equação adjunta e encontrar o comportamento do controle ótimo. Nestes casos, podemos recorrer ao uso de métodos numéricos computacionais para desviar de tal empecilho. O objetivo principal desta seção será de apresentar o **método de varredura frente-trás**, uma ferramenta que nos oferece uma aproximação numérica para o controle ótimo, a partir das consequências do PMP.

Antes de concentrarmos na descrição do método de varredura, vamos primeiramente falar um pouco sobre um algoritmo que nos permite resolver uma equação diferencial ordinária com determinada condição inicial, este é o **método de Runge-Kutta**, vejamos como este método funciona.

Suponha então que tenhamos uma equação diferencial

$$\frac{dy}{dt} = f(t, y), \quad y(0) = y_0. \quad (2.34)$$

O método de Runge-Kutta consiste em considerar

$$y_{i+1} = y_i + h\phi, \quad (2.35)$$

onde  $\phi$  é dita a função de incremento e  $h$  é o tamanho do passo, ou seja,  $h = t_{i+1} - t_i$  com  $i = 1, \dots, n-1$ .

A função de incremento pode ser escrita, em geral, da forma

$$\phi = a_1 k_1 + \dots + a_n k_n,$$

onde  $a_i, i = 1, \dots, n$  são constantes e

$$k_1 = f(t_i, y_i)$$

$$k_2 = f(t_i + p_1 h, y_i + q_{11} k_1 h)$$

$$k_3 = f(t_i + p_2 h, y_i + q_{21} k_1 h + q_{22} k_2 h),$$

$\vdots$

$$k_n = f(t_i + p_{n-1} h, y_i + q_{n-1,1} k_1 h + q_{n-1,2} k_2 h + \dots + q_{n-1,n-1} k_{n-1} h)$$

para  $p_i, q_{ij}, i, j = 1, \dots, n-1$  também constantes.

Várias versões de Runge-Kutta podem ser derivadas dependendo do valor de  $n$ , na verdade, tal valor determina a ordem do método. Para  $n = 1$ , temos por exemplo, o Runge-Kutta de 1º ordem, que a saber, coincide com o Método de Euler. Para  $n = 4$  temos o Runge-Kutta de 4º ordem, que seria talvez o mais popular, sua forma pode se modificar dependendo da escolhas das constantes envolvidas  $a_i, p_i, q_{ij}$ , mas a versão mais comumente usada é a que considera

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

e finalmente,

$$y_{i+1} = y_i + \frac{h}{6} (k_1 + 2k_2 + 2k_3 + k_4). \quad (2.36)$$

Ademais, para encontrar a aproximação numérica da solução  $y(t)$ , basta aplicar (2.36) passo a passo. Mais informações sobre os variados métodos de Runge-Kutta podem ser encontrados em (CHAPRA, 2018, p.593).

Passaremos agora a essencialmente descrever o método de varredura, que a saber, é bastante intuitivo, dadas as condições que nos serão apresentadas.

Nós iremos desenvolver o método em detalhes para o problema de uma variável, todavia em problemas de múltiplas variáveis o método segue de forma análoga, basta repetir o algoritmo

para cada uma das funções controle do problema. Nos próximos capítulos, iremos tratar de problemas de múltiplas variáveis e disponibilizaremos em anexo os códigos computacionais do método de varredura em MATLAB aplicados a tais problemas.

Bem, note que a condição ótima (2.5) pode ser manipulada a fim de encontrarmos uma representação do controle ótimo em função de  $t$ ,  $x(t)$  e da variável adjunta  $\lambda(t)$ , tal valor de  $u^*(t)$  pode ser substituído na equação de estado (2.3) e na equação adjunta (2.7) e teremos então dois problemas de valor inicial, um com condição inicial  $x(t_0) = x_0$  e outro com condição final  $\lambda(t_1) = 0$ .

Agora, considere uma partição  $t_0 = b_1, b_2, \dots, b_N, b_{N+1} = t_1$  do intervalo de tempo  $[t_0, t_1]$ , as aproximações para o controle, o estado e a variável adjunta serão denotadas por:

$$\begin{aligned}\vec{u} &= (u_1, u_2, \dots, u_{N+1}), \\ \vec{x} &= (x_1, x_2, \dots, x_{N+1}), \\ \vec{\lambda} &= (\lambda_1, \lambda_2, \dots, \lambda_{N+1}),\end{aligned}$$

onde os valores  $u_i, x_i, \lambda_i$  para  $i = 1, \dots, N+1$  são os valores do controle, do estado e da variável adjunta calculadas a cada passo ao longo desta partição escolhida.

As etapas serão então descritas a seguir:

**Etapa 1:** Considere uma estimativa inicial para o controle  $\vec{u}$ ;

**Etapa 2:** Usando a condição inicial  $x(t_0) = x_0 = x_1$  e os valores escolhidas para  $\vec{u}$  na Etapa 1, resolva a equação de estado adiante no tempo usando Runge-Kutta e encontre  $\vec{x}$ ;

**Etapa 3:** Usando a condição de transversalidade  $\lambda(t_1) = 0 = \lambda_{N+1}$  e os valores encontrados para  $\vec{u}$  e  $\vec{x}$  nas etapas anteriores, resolva a equação adjunta retroativamente no tempo usando novamente Runge-Kutta e encontre  $\vec{\lambda}$ ;

**Etapa 4:** Respeitando a condição ótima, atualize  $\vec{u}$ , usando os valores para  $\vec{x}$  e  $\vec{\lambda}$  encontrados nas Etapas 2 e 3, respectivamente;

**Etapa 5:** Verifique a convergência do processo. Se os valores para  $u(t)$  nesta iteração e na última, forem suficientemente próximos, considere-os como o valor do controle. Caso contrário, retorne para a Etapa 2.

Neste momento, duas observações são importantes de serem feitas:

- a estimativa inicial  $\vec{u} = 0$  é quase sempre suficiente, a menos de casos onde encontramos divisão por zero;

- se o método apresentar problemas de convergência, uma mudança na estimativa inicial de  $\vec{u}$ , pode ser uma solução.

Veamos agora um pouco sobre como funciona a convergência citada na Etapa 5. Denote  $\overrightarrow{oldu}$  o valor encontrado para o controle em uma iteração anterior, sendo  $\vec{u}$  o valor do controle na iteração atual, para obter uma aproximação apropriada, devemos requerer que a média relativa entre estes valores obedeça

$$\frac{\|\vec{u} - \overrightarrow{oldu}\|}{\|\vec{u}\|} \leq \delta,$$

ou ainda,

$$\delta \|\vec{u}\| - \|\vec{u} - \overrightarrow{oldu}\| \geq 0,$$

onde as normas dos vetores são

$$\|\vec{u}\| = \sum_1^{N+1} |u_i|, \quad \|\vec{u} - \overrightarrow{oldu}\| = \sum_1^{N+1} |u_i - oldu_i|, \quad (2.37)$$

e  $\delta$  é o erro. É aceitável, por exemplo, tomarmos  $N = 1000$  e  $\delta = 0.001$ .

Vamos usar o Exemplo 2.3, para entender com detalhes como realizar a implementação do código do método de varredura de forma detalhada utilizando o software MATLAB:

```

1 function y = codigoMATLAB(x0)
3 tic
5 test = -1;
7 delta = 0.001;
  N = 1000;
9 t=linspace(0,T,N+1);
  h=T/N;
11 h2 = h/2;
13 u = zeros(1,N+1);
  x = zeros(1,N+1);
15 lambda = zeros(1,N+1);
17 x(1) = x0;
19 while(test < 0)
21 oldu = u;
  oldx = x;
23 oldlambda = lambda;
```

Na linha 1 nomeamos `codigoMATLAB` a função do MATLAB que será criada e as variáveis  $x_0$  é o dado de entrada que depende do problema de otimização. Esta função nos oferece a variável  $y$  que será o dado de saída. Na linha 3 inserimos o comando `tic` para iniciar uma contagem do tempo gasto na compilação do nosso código (esta escolha é opcional). A variável `test` criada na linha 5 é uma variável para o teste de convergência que é realizado na Etapa 5 do método de varredura.

As linhas 7 e 8 determinam os valores de  $N$  e  $\delta$  mencionados anteriormente. Na linha 9, criamos um vetor  $t$  que representa a variável temporal. O `linspace` é uma função de MATLAB que cria  $N + 1 = 1001$  pontos igualmente espaçados entre os limites 0 e 1, determinando a partição do intervalo  $[0, 1]$ . O espaçamento da partição será o tamanho do passo e está definido na linha 10 denotado por  $h$ . Na linha 11 determinamos uma notação a ser utilizada nos métodos de Runge-Kutta.

Nas linhas 13, 14 e 15, elegemos estimativas iniciais para as variáveis  $u, x$  e  $\lambda$ , essencialmente, isto significa tomar  $u_i = x_i = \lambda_i = 0$  para  $i = 1, \dots, N + 1$ , estes valores serão substituídos durante as etapas do processo de varredura. Na linha 17, recordamos a condição inicial para a variável de estado. Na linha 19, o comando `while` inicia o loop que caracteriza a varredura para frente e para trás. Assim que obtivermos o valor `test` não negativo, de acordo com (2.4), teremos a convergência do método e a função `while` encerra o loop. As linhas 21, 22 e 23 são as primeiras dentro da função `while`, as mesmas convertem os valores de  $u, x$  e  $\lambda$  em valores atuais  $oldu, oldx$  e  $old\lambda$ , para que sejam utilizados no teste de convergência.

```

24
    for i = 1:N
26 k1 = -0.5*x(i)^2 + u(i);
    k2 = -0.5*(x(i) + h2*k1)^2 + 0.5*(u(i) + u(i+1));
28 k3 = -0.5*(x(i) + h2*k2)^2 + 0.5*(u(i) + u(i+1));
    k4 = -0.5*(x(i) + h*k3)^2 + u(i+1);
30 x(i+1) = x(i) + (h/6)*(k1 + 2*k2 + 2*k3 + k4);
    end
32
    for i = 1:N
34 j = N + 2 - i;
    k1 = lambda(j)*x(j) - 1;
36 k2 = (lambda(j) - h2*k1)*0.5*(x(j)+x(j-1)) - 1;
    k3 = (lambda(j) - h2*k2)*0.5*(x(j)+x(j-1)) - 1;
38 k4 = (lambda(j) - h*k3)*x(j-1) - 1;
    lambda(j-1) = lambda(j) - (h/6)*(k1 + 2*k2 + 2*k3 + k4);
40 end

```

Nas linhas 25-31, temos o método de Runge-Kutta de 4º ordem utilizado para encontrar a solução  $x(t)$  adiante no tempo seguindo a maneira usual que já foi comentada anteriormente. Note que a função  $-\frac{1}{2}x^2 + u$  não depende explicitamente de  $t$ , então para encontrar os valores  $k_{i+1}$ , vamos substituir  $x$  por  $x + \frac{h}{2}k_i$ . Além disso, o nosso controle  $u$  é uma função apenas de  $t$ , então devemos substituir  $u(i)$  por  $u\left(i + \frac{h}{2}\right)$ . Para isto, perceba que no nosso método, os valores de  $t$  variam espaçadamente entre 0 e 1001 como foi definido na linha 7, logo, vamos aproximar  $u\left(i + \frac{h}{2}\right)$  pela média  $\frac{u(i) + u(i+1)}{2}$ , substituindo também  $u(i+h)$  por  $u(i+1)$ . Ademais, note que a cada passo,  $x(i-1)$  é usado para encontrar  $x(i)$ , conseqüentemente,  $x(1)$  e  $x(N+1)$  são calculados usando respectivamente  $x_0$  e  $x(N)$ , ou seja, no comando for da Linha 21, basta variar  $i$  de 1 a  $N$ .

Nas linhas 33-40, temos novamente um Runge-Kutta de 4º ordem, utilizado agora para encontrar o valor da variável adjunta  $\lambda(t)$  retroativamente no tempo. Para isso, adicionamos uma nova variável de tempo  $j$  que depende de  $i$ . Enquanto  $i$  varia de 1 a  $N$ ,  $j$  varia de  $N+1$  a 2 retrocendo no tempo, através da expressão  $j = N + 2 - i$ . Note que  $\lambda(j+1)$  é usado para calcular  $\lambda(j)$ , o que justifica o fato de termos  $j$  variando apenas até 2. Ademais, perceba que desta forma, o nosso incremento de tempo se torna  $-\frac{1}{N}$ , isso justifica os sinais de menos na troca de  $\lambda$  por  $\lambda - \frac{h}{2}k_i$  e no cálculo de  $\lambda(j-1)$  da linha 39. O que aconteceu para  $u(t)$  no Runge-Kutta anterior, agora acontece para  $x(t)$  com a média  $\frac{x(j) + x(j-1)}{2}$ .

```

42 u1 = lambda./2;
   u = 0.5*(u1 + oldu);
44
   temp1 = delta*sum(abs(u)) - sum(abs(oldu - u));
46 temp2 = delta*sum(abs(x)) - sum(abs(oldx - x));
   temp3 = delta*sum(abs(lambda)) - sum(abs(oldlambda - lambda));
48
   test = min(temp1, min(temp2, temp3));
50 end

52 y(1,:) = t;
   y(2,:) = x;
54 y(3,:) = lambda;
   y(4,:) = u;
56
   toc

```

Na linha 42, iniciamos a Etapa 4 do método, através da condição ótima e dos valores de  $\lambda(t)$  encontrados no Runge-Kutta anterior, atualizamos o valor de  $u(t)$  denotando-o por  $u1$ , pois esse

será um valor temporário, usado para teste de convergência. Na linha 43, definimos o controle como a diferença relativa do valor *oldu* definido na linha 21 e o valor *u1* definido na linha 42.

Nas linhas 45-47, definimos os parâmetros para o teste de convergência das variáveis de estado, controle e adjunta, de acordo com (2.37). Na linha 49, chegamos finalmente na Etapa 5 do método e vamos realizar o teste de convergência. Se o valor *test* for negativo, o comando *while* mantém o loop da varredura, por outro lado, se o valor for não negativo, como tomamos o mínimo, teremos os valores de *temp1*, *temp2* e *temp3* todos não negativos, alcançando a convergência do método, e o comando *while* encerra o loop de varredura. Vale lembrar que a função *min* do MATLAB é binária, por isso temos que testar o mínimo para cada par de valores dos parâmetros do teste de convergência.

Alcançada a convergência, o método se encerra na linha 50 e os valores finais de *x*,  $\lambda$  e *u* são armazenados como dados de saída nas linhas 53, 54 e 55, como a variável *t* de tempo sendo definida na linha 52. Na linha 57, inserimos o comando *toc* para finalizar a contagem do tempo gasto na compilação.

Portanto, para explicitar o controle ótimo  $u^*(t)$ , com condição inicial  $x_0 = 0$ , por exemplo, basta digitar na janela de comando do MATLAB, o comando `y=codigoMATLAB(0)` para armazenar os valores das variáveis do problema e o comando `plot(y(1,:),y(4,:))` para realizar a plotagem. Para o Exemplo 2.3, este plot nos retorna a Figura 2.

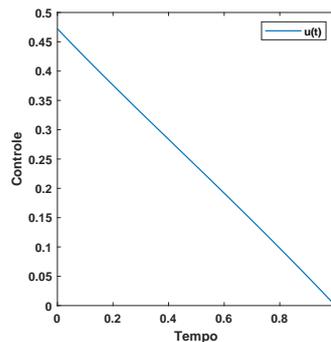


Figura 2 – Plot de controle ótimo encontrado usando o método de varredura.

Ademais, podemos ainda saber o comportamento da variável de estado para a condição inicial  $x_0 = 0$ , para isto, basta digitar `plot(y(1,:),y(2,:))` na janela de comando. Este plot nos retornará a Figura 3.

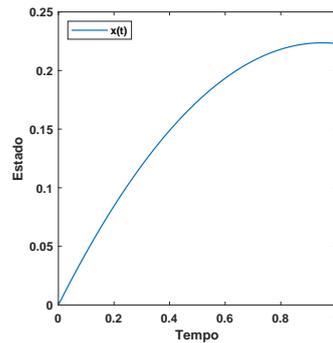


Figura 3 – Plot da variável de estado encontrada usando o método de varredura.

Assim como os problemas sem limitação, existem problemas com limitação que podem ser bastante complicados de resolver analiticamente. Neste sentido, a caracterização (2.30) pode ser usada para resolver o problema numericamente com o método de varredura. Isto é feito com uma leve alteração nas linhas do código.

Por exemplo, no código da função `codigoMATLAB` que descrevemos anteriormente, onde teríamos

```
42 u1 = lambda./2;
   u = 0.5*(u1 + oldu);
```

substituímos por

```
42 u1 = min(b, max(a, lambda./2));
   u = 0.5*(u1 + oldu);
```

e seguimos sem alteração nas demais linhas do código.

Para exemplificar, se adicionamos a condição de contorno  $\frac{1}{5} \leq u(t) \leq \frac{4}{5}$  ao problema de otimização do Exemplo 2.3 com condição de estado inicial  $x(0) = -1$ . Usando a caracterização acima, o método de varredura nos retorna o controle ótimo da Figura 4. Neste momento, fica visível o truncamento do controle para os valores de  $t$  nos quais o controle da Figura 3 excede as condições de contorno.

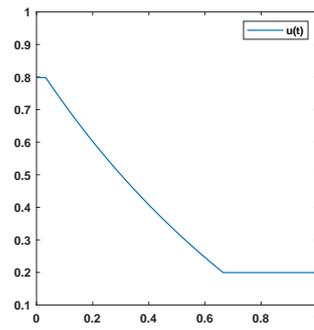


Figura 4 – Plot de controle ótimo com condição de contorno encontrado usando o método de varredura.

### 3 PROGRAMA DE QUARENTENA

#### 3.1 O MODELO SEIRQ ESTRUTURADO POR IDADE

Neste capítulo, investigamos um modelo SEIR clássico adaptado para uma quarentena de indivíduos infectados que testam positivo, uma estratégia não farmacêutica para contenção de doenças infecciosas. A população será dividida em três faixas etárias de acordo com a Tabela 1.

Tabela 1 – Descrição dos grupos

Grupo	Classificação por idade
1	Jovens, de 0 a 19 anos
2	Adultos, de 20 a 59 anos
3	Idosos, a partir de 60 anos

Sejam respectivamente  $S_i(t)$ ,  $E_i(t)$ ,  $I_i(t)$ ,  $R_i(t)$  e  $Q_i(t)$  o número de indivíduos suscetíveis, expostos, infectados, recuperados e em quarentena, em cada faixa etária, no tempo  $t \geq 0$ . Assumimos que a população total

$$N(t) = \sum_{i=1}^3 (S_i(t) + E_i(t) + I_i(t) + R_i(t) + Q_i(t))$$

permanece constante. Essa é uma suposição razoável, pois estamos lidando apenas com um curto período de tempo em comparação com a escala de tempo demográfica.

As equações do modelo, para  $i = 1, 2, 3$ , são descritas em (3.1).

$$\begin{cases} \frac{dS_i}{dt} = -\frac{S_i}{N} \left( \sum_{j=1}^3 \beta_{ij} I_j \right) \\ \frac{dE_i}{dt} = \frac{S_i}{N} \left( \sum_{j=1}^3 \beta_{ij} I_j \right) - \sigma_i E_i \\ \frac{dI_i}{dt} = \sigma_i E_i - \gamma_i I_i - u_i(t) I_i \\ \frac{dR_i}{dt} = \gamma_i I_i + \tau Q_i \\ \frac{dQ_i}{dt} = u_i(t) I_i - \tau Q_i \end{cases} \quad (3.1)$$

A Figura 5 representa a dinâmica entre os compartimentos do modelo.

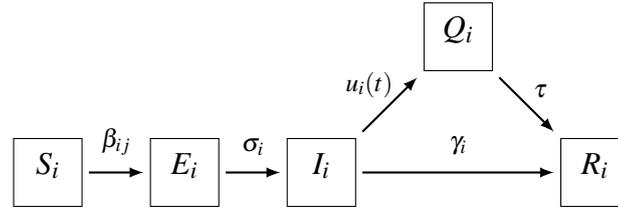


Figura 5 – Diagrama compartimental do modelo.

Todos os parâmetros são não negativos. O coeficiente  $\beta_{ij}$  é a taxa de transmissão da faixa etária  $i$  para a faixa etária  $j$  como em (CASTILHO et al., 2020), será assumido que  $\beta_{ij} = \beta_{ji}$  para todos os  $i$  e  $j$ .  $\sigma_i$  e  $\gamma_i$  são as taxas de latência e remoção, respectivamente, para a faixa etária  $i$ .  $\tau$  é a taxa de saída da quarentena. Os valores dos parâmetros são retirados de (CASTILHO et al., 2020), onde o ajuste dos dados foi feito por meio de uma adaptação de um algoritmo de mínimos quadrados, tais valores estão listados na Tabela 2.

Tabela 2 – Valores dos parâmetros.

Parâmetro	Valor	Parâmetro	Valor
$\beta_{11}$	1.76168	$\sigma_1$	0.27300
$\beta_{12}$	0.36475	$\sigma_2$	0.58232
$\beta_{13}$	1.32468	$\sigma_3$	0.69339
$\beta_{22}$	0.63802	$\gamma_1$	0.06862
$\beta_{23}$	0.35958	$\gamma_2$	0.03317
$\beta_{33}$	0.57347	$\gamma_3$	0.35577

Na recomendação da OMS de 27 de maio de 2020 (WHO, 2020), o critério para alta de um paciente com COVID-19 é 10 dias de isolamento após o início dos sintomas, adicionamos de pelo menos 3 dias sem sintomas para pacientes sintomáticos e 10 dias de isolamento após um teste positivo para SARS-CoV-2 para casos assintomáticos. Consideramos que um indivíduo infectado fica em quarentena, então, por 13 dias. Portanto, como o parâmetro de saída da quarentena, assim como os demais parâmetros, é tomado como fração do tempo gasto no status de quarentenado, tomamos  $\tau = \frac{1}{13}$ .

Os parâmetros  $u_i(t)$  denotam os controles, eles representam a fração de indivíduos infectados em cada faixa etária que são colocados em quarentena por unidade de tempo, no tempo  $t$ . Os controles satisfazem, portanto,

$$0 \leq u_i(t) \leq 1, \quad i = 1, 2, 3, \quad (3.2)$$

uma vez que, vamos assumir que não existirá tempo máximo para entrada em quarentena e o tempo mínimo é de 1 dia.

A população total do grupo  $i$  é definida como

$$N_i(t) = S_i(t) + E_i(t) + I_i(t) + R_i(t) + Q_i(t).$$

Somando as equações do sistema (3.1), fica claro que  $N_i(t)$  é constante para  $i = 1, 2, 3$ , ou seja,  $N_i'(t) = 0$ . Além disso,  $R_i(t)$  só aparece nas demais equações como parte de  $N_i(t)$ , então substituímos as equações de  $R_i'(t)$  por  $N_i'(t)$ . Portanto, podemos de reescrever o sistema (3.1) da forma (3.3).

$$\begin{cases} \frac{dS_i}{dt} = -\frac{S_i}{N} \left( \sum_{j=1}^3 \beta_{ij} I_j \right) \\ \frac{dE_i}{dt} = \frac{S_i}{N} \left( \sum_{j=1}^3 \beta_{ij} I_j \right) - \sigma_i E_i \\ \frac{dI_i}{dt} = \sigma_i E_i - \gamma_i I_i - u_i(t) I_i \\ \frac{dQ_i}{dt} = u_i(t) I_i - \tau Q_i \\ \frac{dN_i}{dt} = 0 \end{cases} \quad (3.3)$$

Consideramos que o número de recuperações é dado pela diferença entre o número total de casos e o número de mortes. Embora isso possa não refletir a realidade, já que pode haver alguns pacientes que consideramos recuperados, mas ainda carregam a doença, procedemos dessa forma devido à escassez de melhores informações disponíveis sobre recuperações.

Os dados iniciais para as nossas simulações serão definidos a partir de dados do Brasil em 8 de maio de 2020. De acordo com (WORLDMETERS, 2024; GONDIM; MACHADO, 2020), haviam 65.124 recuperados e 76.603 casos ativos no país, nesta data. O período médio de incubação da doença foi estimado em cerca de 5 dias (LAUER et al., 2020). Ainda de acordo com as mesmas referências, haviam 97.575 casos ativos de COVID-19 no Brasil até 13 de maio de 2020, então definimos o número inicial de indivíduos expostos como o aumento de 20.972 casos ativos desses 5 dias. Assumimos que não há indivíduos em quarentena no início das simulações, além disso, como os números de expostos, infecções e recuperações são bastante pequenos em relação à população total, assumimos que o número de indivíduos suscetíveis para cada faixa etária é o mesmo que a população total dessa respectiva faixa etária. Finalmente, fixamos a população total em 200 milhões, dos quais 40% são jovens, 50% são adultos e 10% são idosos. Um resumo das condições iniciais, arredondadas para os inteiros mais próximos, é mostrado na Tabela 3.

Tabela 3 – Condições iniciais.

<b>Classe</b>	$i = 1$	$i = 2$	$i = 3$
Suscetíveis	80000000	10000000	20000000
Expostos	216	9987	10769
Infectados	789	36478	39335
Recuperados	729	33415	30979
Quarentenados	0	0	0

Usando os dados disponíveis em um relatório de 2020 do Ministério da Saúde da Espanha (ESPAÑA, 2020), calculamos como as infecções e mortes são distribuídas entre as faixas etárias, e as porcentagens estão disponíveis na Tabela 4.

Tabela 4 – Número de casos, recuperações e mortes para cada grupo, de acordo com (ESPAÑA, 2020).

<b>Grupo</b>	<b>Casos</b>	<b>% de casos</b>	<b>Recuperados</b>	<b>% de recuperados</b>	<b>Mortes</b>
1	2448	1.03%	2441	1.12%	7
2	113059	47.62%	112168	51.31%	981
3	121928	51.35%	103980	47.57%	17948
Total	237435	100%	218589	100%	18846

### 3.2 O PROBLEMA DE OTIMIZAÇÃO

Restritos ao sistema (3.3), vamos minimizar o funcional

$$J = \int_0^T \sum_{i=1}^3 (I_i(t) + B_i u_i^2(t)) dt, \quad (3.4)$$

onde  $T$  é a duração da quarentena, e os parâmetros  $B_i$  são os custos do controle. Será assumido que  $B_i > 0$  para  $i = 1, 2, 3$ , bem como

$$B = B_1 + B_2 + B_3, \quad (3.5)$$

onde  $B \in \mathbb{R}$  é o custo total do controle.

Pela forma como definimos (3.4), a partir da positividade das funções, será possível minimizar o número de infectados e os custos associados ao controle, este é o nosso objetivo.

O Teorema 2.3 pode ser usado para garantir a existência do controle ótimo. As funções do integrando em (3.4) e as equações de (3.3) são bem comportadas, a convexidade em  $u_i$  de nosso funcional é essencial para satisfazer as hipóteses de existência.

O PMP 2.2 afirma que os controles ótimos são soluções do sistema Hamiltoniano

$$H = \sum_{i=1}^3 \left[ B_i u_i^2 + u_i I_i (\lambda_Q^i - \lambda_I^i) + \frac{S_i}{N} \left( \sum_{j=1}^3 \beta_{ij} I_j \right) (\lambda_E^i - \lambda_S^i) \right] + \sum_{i=1}^3 \left[ \sigma_i E_i (\lambda_I^i - \lambda_E^i) + I_i (1 - \gamma_i \lambda_I^i) - \tau Q_i \lambda_Q^i \right], \quad (3.6)$$

envolvendo as variáveis de estado  $S_i, E_i, I_i, R_i, Q_i$  e as variáveis adjuntas  $\lambda_S^i, \lambda_E^i, \lambda_I^i, \lambda_Q^i, \lambda_N^i$ , que satisfazem o sistema adjunto

$$\begin{cases} \frac{d\lambda_S^i}{dt} = \frac{1}{N} \left( \sum_{j=1}^3 \beta_{ij} I_j \right) (\lambda_S^i - \lambda_E^i) \\ \frac{d\lambda_E^i}{dt} = \sigma_i (\lambda_E^i - \lambda_I^i) \\ \frac{d\lambda_I^i}{dt} = u_i (\lambda_I^i - \lambda_Q^i) + \gamma_i \lambda_I^i - 1 + \frac{1}{N} \left( \sum_{j=1}^3 \beta_{ij} S_j (\lambda_S^j - \lambda_E^j) \right) \\ \frac{d\lambda_Q^i}{dt} = \tau \lambda_Q^i \\ \frac{d\lambda_N^i}{dt} = \frac{1}{N^2} \left( \sum_{j,k=1}^3 \beta_{kj} S_k I_j (\lambda_E^k - \lambda_S^k) \right) \end{cases} \quad (3.7)$$

As condições de transversalidade para as variáveis adjuntas são

$$\lambda_S^i(T) = \lambda_E^i(T) = \lambda_I^i(T) = \lambda_Q^i(T) = \lambda_N^i(T) = 0, \quad i = 1, 2, 3. \quad (3.8)$$

Finalmente, os controles ótimos obedecem as condições ótimas

$$\left. \frac{\partial H}{\partial u_i} \right|_{u_i = \bar{u}_i^*} = 0,$$

resultando em

$$\bar{u}_i^* = \frac{I_i (\lambda_I^i - \lambda_Q^i)}{2B_i}. \quad (3.9)$$

Como consideramos os limites do controle (3.2), a partir da caracterização (2.30), os valores de  $u_i^*$  são calculados usando

$$u_i^* = \min \left\{ 1, \max \left\{ 0, \frac{I_i (\lambda_I^i - \lambda_Q^i)}{2B_i} \right\} \right\}. \quad (3.10)$$

Analisando as expressões envolvidas nos sistemas (3.3) e (3.7), a unicidade do controle ótimo, ao menos para um tempo  $T$  suficientemente pequeno, é garantida pelo Teorema 2.4.

As soluções numéricas para o sistema (3.3) e (3.7) são encontradas usando o método de varredura da Seção (2.4) com condição inicial em  $t = 0$  para (3.3) e condição final em  $t = T$  para o sistema (3.7).

O código MATLAB está disponível no Anexo A

### 3.3 RESULTADOS E DISCUSSÕES

No nosso modelo, os indivíduos infectados são identificados através de triagem realizada por meio de testes. A compra, o armazenamento e a distribuição destes testes, além da organização das equipes médicas, tem importância significativa nessa estratégia, e portanto a quarentena envolve também fatores econômicos. É por isso, que cada grupo  $i$  estará associado ao custo  $B_i$ .

Note que uma maior testagem é seguida de uma maior identificação dos infectados, quando os mesmos forem isolados teremos uma menor exposição destes na sociedade, e portanto, um menor poder de transmissão da doença, e claramente isto pode interferir no número reprodutivo básico. O  $R_0$  apresenta um decaimento maior para o grupo 1 a partir do aumento da taxa de recuperação  $\gamma_i$ , de acordo com (CASTILHO et al., 2020, p.10). Isso implica que o grupo 1 necessita de maior investimento em testes, mas levando a um maior custo. Ainda de acordo com (CASTILHO et al., 2020, p.10), à medida que  $\gamma_i$  aumenta,  $R_0$  apresenta comportamento semelhante para os grupos 2 e 3. Embora o número de testes seja maior para o grupo 2 devido à sua população maior, o grupo 3 requer um investimento maior em uma campanha publicitária porque tem acesso limitado às informações, daí consideramos que esses grupos têm o mesmo custo de controle. Assumimos então que o custo total  $B = 7500$ , será distribuído em 40%, 30% e 30% para os grupos 1, 2 e 3, respectivamente.

A Figura 6 apresenta os gráficos dos controles ótimos  $u_1(t)$ ,  $u_2(t)$  e  $u_3(t)$  para a distribuição de custo escolhida. A simulação é feita para 60, 90 e 120 dias.

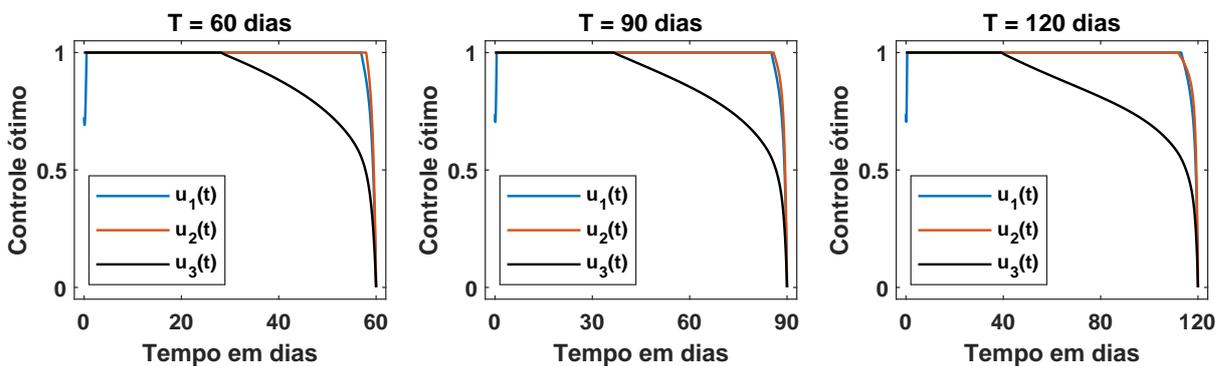


Figura 6 – Os controles ótimos para diferentes durações de quarentena.

Uma característica interessante dos gráficos de controle ótimo é que eles fornecem um “calendário ótimo” de quando a quarentena deve começar a ser relaxada. Estes momentos são definidos quando deixamos de ter  $u_i = 1$ . O calendário é mostrado na Tabela 5.

Tabela 5 – Calendário ótimo de relaxamento.

Grupo	$T = 60$	$T = 90$	$T = 120$
1	57 dias	85 dias	113 dias
2	58 dias	86 dias	112 dias
3	28 dias	37 dias	39 dias

Segundo (GONDIM; MACHADO, 2020), para um programa de lockdown, o relaxamento do isolamento é inversamente proporcional ao custo de controle para cada grupo: adultos que tem custos mais altos são liberados primeiro, e os idosos que têm os custos mais baixos são liberados por último. Em nosso modelo de quarentena de infectados, conforme mostrado na Tabela 5, a ordem de relaxamento segue a distribuição populacional dos grupos: os idosos que tem a menor população são liberados primeiro, e os adultos com a maior população são liberados por último.

A Figura 7 mostra os gráficos das curvas de infectados para uma quarentena com controle ótimo. Em todos os casos de duração de quarentena, o número de infectados atinge um mínimo pouco antes do final da simulação, e quando a quarentena termina, o número de casos está aumentando, todavia permanece muito menor do que na condição inicial. Isto sugere que o tempo de quarentena poderia ser estendido de forma a minimizar ainda mais os números de infectados.

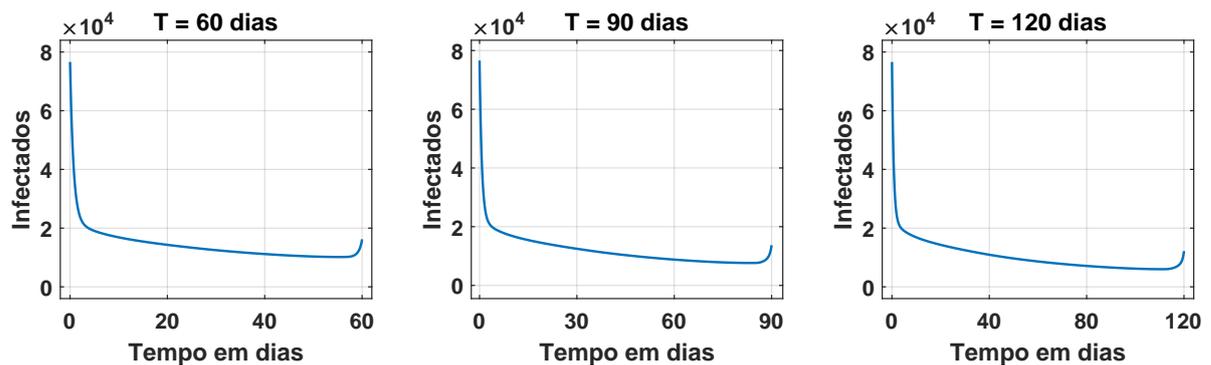


Figura 7 – Curvas de infectados com controle ótimo para diferentes durações de quarentena.

A importância da quarentena é refletida no número de infectados, e isso é mostrado na Figura 8, onde não consideramos um controle de quarentena e temos um grande número cumulativo de infectados. As curvas de infectados sem controle de quarentena para 60 e 90 dias não são apresentadas porque se comportam exatamente como restrições para o caso de 120 dias.

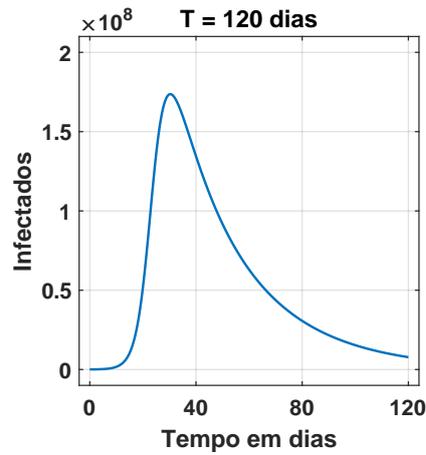


Figura 8 – Curva de infectados sem controle ótimo para 120 dias.

Podemos analisar o que acontece se o governo demorar muito para implementar a quarentena. Em 13 de maio de 2020, o número de casos ativos no Brasil dobra após 10 dias e quadruplica após 20 dias, segundo (GONDIM; MACHADO, 2020). Avaliamos esta situação, considerando que as condições iniciais de indivíduos expostos, infectados e recuperados são duas e quatro vezes maiores que seus valores originais.

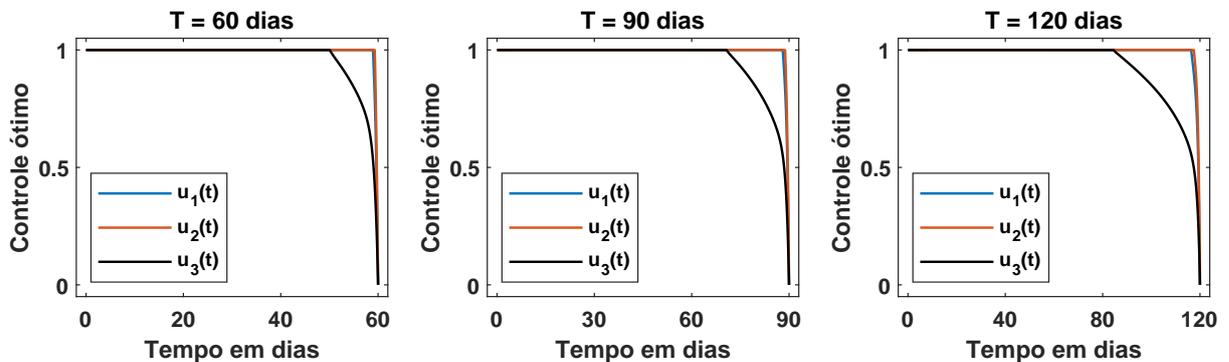


Figura 9 – Controle ótimo para atraso de 10 dias no início da quarentena.

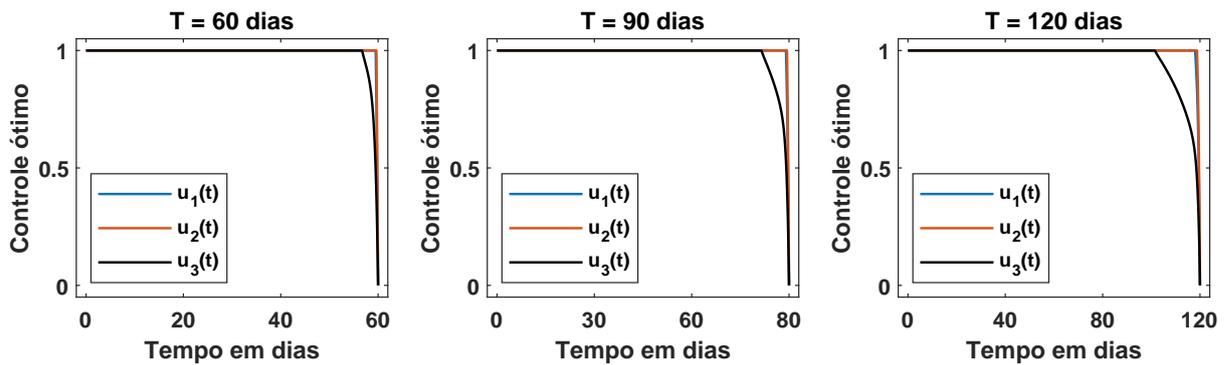


Figura 10 – Controle ótimo para atraso de 20 dias no início da quarentena.

Os controles mostrados nas Figuras 9 e 10 fornecem diferentes calendários de relaxamento, que são descritos nas Tabelas 6 e 7.

Tabela 6 – Calendário ótimo de relaxamento para atraso de 10 dias no início da quarentena.

<b>Grupo</b>	$T = 60$	$T = 90$	$T = 120$
1	59 dias	88 dias	116 dias
2	59 dias	89 dias	117 dias
3	50 dias	71 dias	84 dias

Tabela 7 – Calendário ótimo de relaxamento para atraso de 20 dias no início da quarentena.

<b>Grupo</b>	$T = 60$	$T = 90$	$T = 120$
1	59 dias	89 dias	118 dias
2	60 dias	89 dias	119 dias
3	57 dias	81 dias	101 dias

Comparando os calendários das Tabelas 5, 6 e 7, o atraso de 10 e 20 dias no início do isolamento dos infectados resulta em uma extensão do tempo de isolamento rígido. Essa extensão é suficiente para que praticamente não haja relaxamento para os grupos 1 e 2. Para o grupo 3, essa extensão é bem maior se comparada com os outros grupos e o tempo de quarentena aumenta em 0.78 e 1.03 vezes para uma campanha de 60 dias, em 0.91 e 1.18 vezes para uma campanha de 90 dias e em 1.15 e 1.58 vezes para uma campanha de 120 dias.

Analisaremos como as quarentenas com controle ótimo reduzem o número de mortes. Em nosso modelo, as mortes são calculadas como fração dos indivíduos recuperados. De acordo

com a Tabela 4, obtemos as taxas de mortalidade  $\mu_1$ ,  $\mu_2$  e  $\mu_3$  para os grupos 1, 2 e 3, respectivamente. Os valores das taxas são:

$$\mu_1 = 0,003 \quad \mu_2 = 0,008 \quad \mu_3 = 0,147.$$

Vamos denotar por  $D(B, T)$  o número cumulativo de mortes em uma campanha de controle ótimo com custo total  $B$  e tempo final  $T$ , e por  $D(t)$  o número de mortes para o modelo sem quarentena no tempo  $t$ . Neste momento, temos a intenção de analisar o quanto  $D(t)$  é maior que  $D(B, T)$ . Na Figura 11, plotamos então os gráficos de  $D(t)$  dividido por  $D(B, T)$  para valores de  $T$  iguais a 60, 90 e 120 dias.

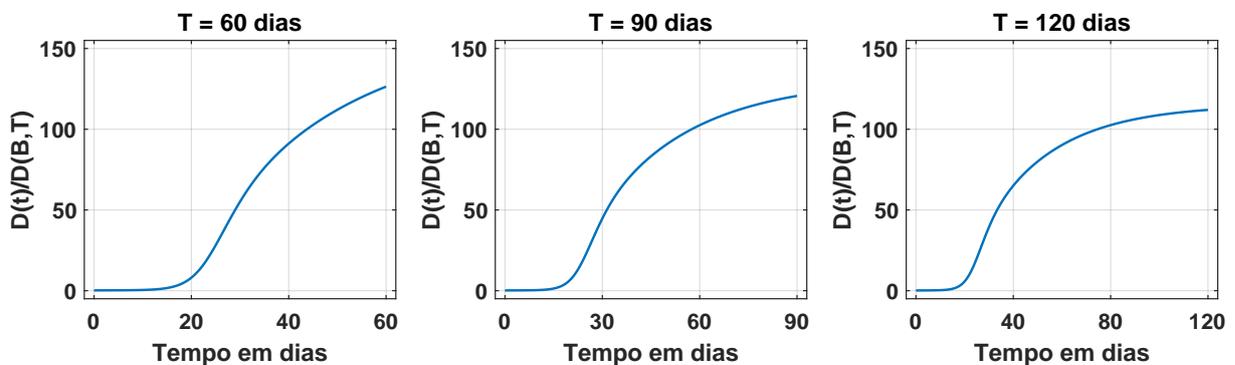


Figura 11 – Gráficos de  $D(t)/D(B, T)$  para diferentes durações de quarentena.

Devido à natureza incerta dos parâmetros e ao alto número de casos não relatados, não calculamos os números brutos de  $D(B, T)$  e  $D(t)$ . No entanto, ao final da quarentena, os gráficos de  $D(t)/D(B, T)$  na Figura 11, indicam que os controles ótimos reduzem o número de mortes em 126.4, 120.5 e 111.9 vezes para 60, 90 e 120 dias, respectivamente.

## 4 CAMPANHA DE VACINAÇÃO

### 4.1 O MODELO SEIRV

Neste capítulo, investigamos um modelo SEIR modificado para uma estratégia de vacinação de dose única em indivíduos suscetíveis, uma estratégia farmacêutica para contenção de doenças infecciosas. No caso da COVID-19, por exemplo, estudos como (BARTSCH et al., 2020; MAKHOUL et al., 2020) garantiam antecipadamente que uma vacina com proteção parcial era suficiente para resolver a pandemia de SARS-CoV-2. Trabalharemos então com vacinas que não são 100% eficazes. Ademais, consideraremos que duas vacinas  $V_1$  e  $V_2$  têm eficácia  $\theta_1$  e  $\theta_2$ , respectivamente, com  $\theta_2 < \theta_1 < 1$ .

De acordo com (TENTORI et al., 2021), a eficácia  $\theta_i$ , para  $i = 1, 2$ , é dada por  $\theta_i = (P - P_i)/P$ , onde  $P$  e  $P_i$  indicam as taxas de ataque entre não vacinados e vacinados, ou seja, as probabilidades de que um contato com um não vacinado ou com um vacinado resulte em transmissão. Então, temos  $P_i = P(1 - \theta_i)$  para  $i = 1, 2$ . Sabemos que a taxa de transmissão para um modelo SEIR é  $\beta = cP$ , onde  $c$  é a taxa de contato. Portanto, para indivíduos vacinados, a taxa de transmissão é dada por  $\beta_i = cP_i = cP(1 - \theta_i) = \beta(1 - \theta_i)$ , ou seja, a eficácia da vacina tem um efeito direto na redução da taxa de transmissão da doença.

Os valores que serão escolhidos para serem usados nas simulações como valores de eficácia da vacina são motivados por valores reais das vacinas para COVID-19, por exemplo: a vacina Pfizer é 91% eficaz (WHO, 2022e), a vacina AstraZeneca é 74% eficaz (WHO, 2022c), a vacina Janssen é 67% eficaz (WHO, 2022b) e a vacina Coronavac é 51% eficaz (WHO, 2022d). Essas referências são de ensaios de fase 3 em diferentes países com diferentes grupos de pessoas, tais eficácias são garantidas para duas doses da vacina, mas a fim de simplificação, consideramos que essa eficácia é válida para uma dose única da vacina.

A vacinação foi associada à alta proteção do SARS-CoV-2 por um curto período, mas diminuiu consideravelmente após alguns meses, segundo (HALL et al., 2022). Essa estratégia contém a propagação de uma doença, mas não elimina completamente a possibilidade de infecção, reinfeções após a vacinação foram relatadas, de acordo com (RAHMAN et al., 2022; FAKHROO et al., 2021). Portanto, nosso modelo admitirá um efeito de reinfeção.

Sejam  $S(t)$ ,  $V_1(t)$ ,  $V_2(t)$ ,  $E(t)$ ,  $I(t)$  e  $R(t)$  o número de indivíduos suscetíveis, vacinados com a vacina  $V_1$ , vacinados com a vacina  $V_2$ , expostos, infectados e recuperados, respectivamente.

Assumimos que a população total

$$N(t) = S(t) + V_1(t) + V_2(t) + E(t) + I(t) + R(t)$$

é constante, pois estamos lidando apenas com um curto período de tempo em comparação à escala de tempo demográfica.

A Figura 12 representa a dinâmica entre os compartimentos do modelo.

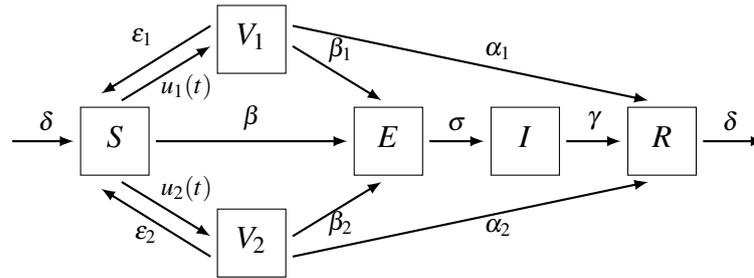


Figura 12 – Diagrama compartimental do modelo.

A população suscetível diminuirá quando: passam para a classe de expostos a uma taxa de transmissão  $\beta$ , são vacinados com a vacina  $V_1$  para uma taxa de controle  $u_1(t)$  ou são vacinados com a vacina  $V_2$  para um controle  $u_2(t)$ . Os controles denotam a fração de indivíduos que são vacinados por unidade de tempo  $t$ , logo, nossa condição de limitação para o controle é

$$0 \leq u(t) \leq 1, \quad (4.1)$$

já que não existirá tempo máximo para ser vacinado e o tempo mínimo é 1 dia.

No modelo, há três possibilidades para um indivíduo vacinado:

1. Sair da classe dos vacinados com imunidade vacinal e passar a ser recuperado a uma taxa  $\alpha_i$ , para  $i = 1, 2$ ;
2. Manter-se vulnerável mesmo sendo pós-vacinado com a possibilidade de contrair a doença, passando para a classe suscetível a uma taxa  $\epsilon_i$ , para  $i = 1, 2$ ;
3. Contrair a doença tornando-se um indivíduo exposto transmitindo-a a uma taxa reduzida  $\beta_i = 1 - \theta_i$ , para  $i = 1, 2$ ;

A população exposta diminui quando passa para a classe de infectados com uma taxa de latência  $\sigma$ . A população recuperada aumenta a taxa de recuperação  $\gamma$  e diminui a uma taxa de reinfeção  $\delta$ . Note que neste modelo, a população se recupera adquirindo imunidade vacinal à doença sem necessariamente sofrer as consequências da infecção.

O sistema de equações é

$$\left\{ \begin{array}{l} \frac{dS}{dt} = -\frac{\beta SI}{N} - u_1(t)S - u_2(t)S + \varepsilon_1 V_1 + \varepsilon_2 V_2 + \delta R \\ \frac{dV_1}{dt} = u_1(t)S - \frac{\beta_1 V_1 I}{N} - \varepsilon_1 V_1 - \alpha_1 V_1 \\ \frac{dV_2}{dt} = u_2(t)S - \frac{\beta_2 V_2 I}{N} - \varepsilon_2 V_2 - \alpha_2 V_2 \\ \frac{dE}{dt} = \frac{\beta SI}{N} + \frac{\beta_1 V_1 I}{N} + \frac{\beta_2 V_2 I}{N} - \sigma E \\ \frac{dI}{dt} = \sigma E - \gamma I \\ \frac{dR}{dt} = \gamma I - \delta R + \alpha_1 V_1 + \alpha_2 V_2 \end{array} \right. \quad (4.2)$$

Todos os nossos parâmetros são não negativos e foram retirados de (PAUL et al., 2023), onde são estimados por mínimos quadrados e o método estatístico da cadeia de Markov Monte Carlo mostra a convergência para tais valores. Nesta referência em questão, é considerada apenas uma vacina, então somente um valor para a taxa de imunidade da vacina e taxa de retorno à classe suscetível é disposto. Em nosso estudo, temos duas vacinas e precisamos de dois valores para cada parâmetro, então para simplificar consideraremos inicialmente  $\alpha_1 = \alpha_2$  e  $\varepsilon_1 = \varepsilon_2$ . Os parâmetros estão descritos na Tabela 8.

Tabela 8 – Valores dos parâmetros.

Parâmetro	Descrição	Valor
$\beta$	taxa de transmissão	0.45
$\sigma$	taxa de latência	0.25
$\gamma$	taxa de recuperação	0.07
$\delta$	taxa de reinfecção	0.65
$\alpha_1$	taxa de imunidade da vacina $V_1$	0.08
$\alpha_2$	taxa de imunidade da vacina $V_2$	0.08
$\varepsilon_1$	taxa de retorno da vacina $V_1$	0.54
$\varepsilon_2$	taxa de retorno da vacina $V_2$	0.54

O número de suscetíveis é considerado igual à população total, pois a proporção de expostos, infectados e recuperados é muito pequena em comparação à população total, que será fixada em 200 milhões. Consideramos que não há indivíduos vacinados quando a simulação começa. Nosso tempo inicial consistirá em dados atualizados do Brasil em 08 de Maio de 2020 de acordo com (GONDIM; MACHADO, 2020; WORLDMETERS, 2024). O número de recuperados é a diferença entre o número de casos ativos e o número de mortes. Acredita-se que o período

médico de incubação da doença seja em torno de 5 dias, segundo (LAUER et al., 2020). Portanto, o número de expostos é então determinado subtraindo o número de casos ativos do dia 13 de Maio pelo número de casos ativos de 08 de Maio. Ademais, assumimos que não há indivíduos vacinados no início das simulações. As condições iniciais são descritas na Tabela 9.

Tabela 9 – Descrição das condições iniciais.

Grupo	Condição Inicial
Suscetíveis	200000000
Vacinados com a vacina $V_1$	0
Vacinados com a vacina $V_2$	0
Expostos	65124
Infectados	76603
Recuperados	65124

## 4.2 O PROBLEMA DE OTIMIZAÇÃO

Restritos ao sistema (4.2), vamos minimizar o funcional

$$J = \int_0^T (I(t) + B_1 u_1^2(t) + B_2 u_2^2(t)) dt. \quad (4.3)$$

onde  $T$  é a duração da campanha de vacinação e os valores  $B_1$  e  $B_2$  são os custos de implementação das vacinas  $V_1$  e  $V_2$ , respectivamente. Assumimos que  $B_1, B_2 > 0$ .

Novamente, pela forma como definimos 2.24, considerando a positividade das funções, será possível minimizar o número de infectados e os custos associados ao controle, este é o nosso objetivo.

O Teorema 2.3 pode ser usado para garantir a existência do controle ótimo. As funções do integrando em 4.3 e as equações de 4.2 são bem comportadas, a convexidade em  $u_i$  no nosso funcional é essencial para satisfazer as hipóteses de existência.

Usando o PMP, para encontrar o controle ótimo, basta encontrar as soluções do sistema Hamiltoniano

$$\begin{aligned} H = & I + B_1 u_1^2 + B_2 u_2^2 + u_1 S(\lambda_{V_1} - \lambda_S) + u_2 S(\lambda_{V_2} - \lambda_S) \\ & + \frac{\beta S I}{N} (\lambda_E - \lambda_S) + \frac{\beta_1 V_1 I}{N} (\lambda_E - \lambda_{V_1}) + \frac{\beta_2 V_2 I}{N} (\lambda_E - \lambda_{V_2}) \\ & + \varepsilon_1 V_1 (\lambda_S - \lambda_{V_1}) + \varepsilon_2 V_2 (\lambda_S - \lambda_{V_2}) + \alpha_1 V_1 (\lambda_R - \lambda_{V_1}) \\ & + \alpha_2 V_2 (\lambda_R - \lambda_{V_2}) + \sigma E (\lambda_I - \lambda_E) + \delta R (\lambda_S - \lambda_R) + \gamma I (\lambda_R - \lambda_I). \end{aligned} \quad (4.4)$$

envolvendo as variáveis de estado  $S, V_1, V_2, E, I, R$  e as variáveis adjuntas  $\lambda_S, \lambda_{V_1}, \lambda_{V_2}, \lambda_E, \lambda_I, \lambda_R$ , que satisfazem o sistema adjunto

$$\left\{ \begin{array}{l} \frac{d\lambda_S}{dt} = \frac{I}{N^2} \left[ \beta_1 V_1 (\lambda_E - \lambda_{V_1}) + \beta_2 V_2 (\lambda_E - \lambda_{V_2}) + \beta (N - S) (\lambda_S - \lambda_E) \right] \\ \quad + u_1 (\lambda_S - \lambda_{V_1}) + u_2 (\lambda_S - \lambda_{V_2}) \\ \frac{d\lambda_{V_1}}{dt} = \frac{I}{N^2} \left[ \beta S (\lambda_E - \lambda_S) + \beta_2 V_2 (\lambda_E - \lambda_{V_2}) + \beta_1 (N - V_1) (\lambda_{V_1} - \lambda_E) \right] \\ \quad + \varepsilon_1 (\lambda_{V_1} - \lambda_S) + \alpha_1 (\lambda_{V_1} - \lambda_R) \\ \frac{d\lambda_{V_2}}{dt} = \frac{I}{N^2} \left[ \beta S (\lambda_E - \lambda_S) + \beta_1 V_1 (\lambda_E - \lambda_{V_1}) + \beta_2 (N - V_2) (\lambda_{V_2} - \lambda_E) \right] \\ \quad + \varepsilon_2 (\lambda_{V_2} - \lambda_S) + \alpha_2 (\lambda_{V_2} - \lambda_R) \\ \frac{d\lambda_E}{dt} = \frac{I}{N^2} \left[ \beta S (\lambda_E - \lambda_S) + \beta_1 V_1 (\lambda_E - \lambda_{V_1}) + \beta_2 V_2 (\lambda_E - \lambda_{V_2}) \right] \\ \quad + \sigma (\lambda_E - \lambda_I) \\ \frac{d\lambda_I}{dt} = \frac{N - I}{N^2} \left[ \beta S (\lambda_S - \lambda_E) + \beta_1 V_1 (\lambda_{V_1} - \lambda_E) + \beta_2 V_2 (\lambda_{V_2} - \lambda_E) \right] \\ \quad + \gamma (\lambda_I - \lambda_R) - 1 \\ \frac{d\lambda_R}{dt} = \frac{I}{N^2} \left[ \beta S (\lambda_E - \lambda_S) + \beta_1 V_1 (\lambda_E - \lambda_{V_1}) + \beta_2 V_2 (\lambda_E - \lambda_{V_2}) \right] \\ \quad + \delta (\lambda_R - \lambda_S) \end{array} \right. \quad (4.5)$$

As variáveis adjuntas satisfazem as condições de transversalidade

$$\lambda_S(T) = \lambda_{V_1}(T) = \lambda_{V_2}(T) = \lambda_E(T) = \lambda_I(T) = \lambda_R(T) = 0. \quad (4.6)$$

Sabendo que os controles ótimos obedecem as condições ótimas

$$\left. \frac{\partial H}{\partial u} \right|_{u=\bar{u}^*} = 0,$$

isso resulta em

$$\bar{u}_i^* = \frac{S(\lambda_S - \lambda_{V_i})}{2B_i}, \quad i = 1, 2. \quad (4.7)$$

Como estamos considerando controles limitados seguindo (3.2), a partir da caracterização (2.30), o controle ótimo  $u_i^*$  é calculado usando

$$u_i^* = \min \left\{ 1, \max \left\{ 0, \frac{S(\lambda_S - \lambda_{V_i})}{2B_i} \right\} \right\}, \quad i = 1, 2. \quad (4.8)$$

Analisando as expressões envolvidas nos sistemas (4.2) e (4.5), a unicidade do controle ótimo, ao menos para um tempo  $T$  suficientemente pequeno, é garantida pelo Teorema 2.4.

As soluções numéricas para os sistemas (4.2) e (4.5) podem ser encontradas usando o método de varredura discutido na Subseção 2.4, com condição inicial em  $t = 0$  para o sistema (4.2) e uma condição final em  $t = T$  para o sistema (4.5).

O código MATLAB está disponível no Anexo B.

### 4.3 RESULTADOS E DISCUSSÕES

Assim como no caso da quarentena, a campanha de vacinação envolve fatores econômicos. A compra, o armazenamento e a distribuição das vacinas, além da organização das equipes médicas, tem importância significativa. Nosso problema de controle ótimo depende dos custos de implementação  $B_1, B_2$  das duas vacinas  $V_1, V_2$ , respectivamente. Aqui, os custos das vacinas serão distribuídos como  $B_i = \theta_i \cdot 10^4$  para  $i = 1, 2$ , ou seja, o custo de implementação da vacina será proporcional ao seu poder de eficácia, essa decisão foi tomada devido a dificuldade de encontrar referências sobre gastos governamentais na compra de vacinas.

A seguir, veja os gráficos dos controles ótimos  $u_1(t)$  e  $u_2(t)$  para vacinação com duração de 60, 120 e 180 dias. Em todos os casos,  $u_1$  representa a implementação da vacina que tem maior eficácia, e conseqüentemente maior custo.

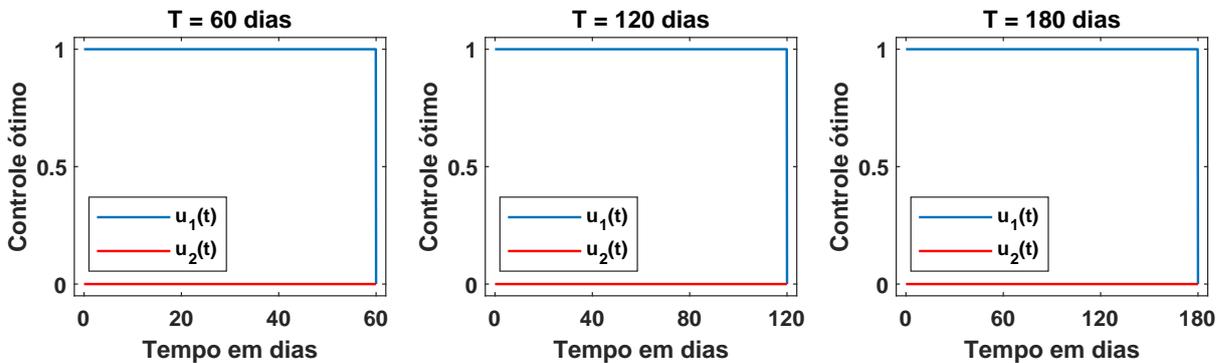


Figura 13 – Os controles ótimos para uso de duas vacinas com eficácia de 91% e 51% em diferentes durações de campanha de vacinação.

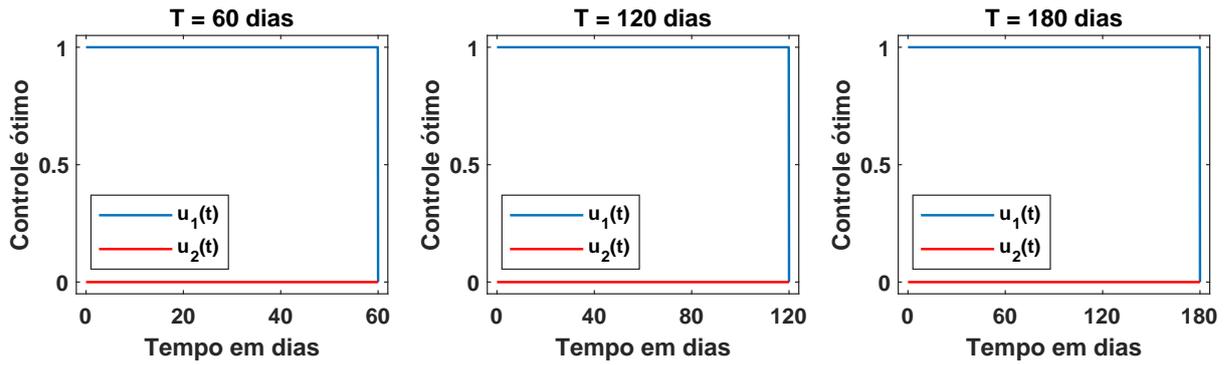


Figura 14 – Os controles ótimos para uso de duas vacinas com eficácia de 74% e 51% em diferentes durações de campanha de vacinação.

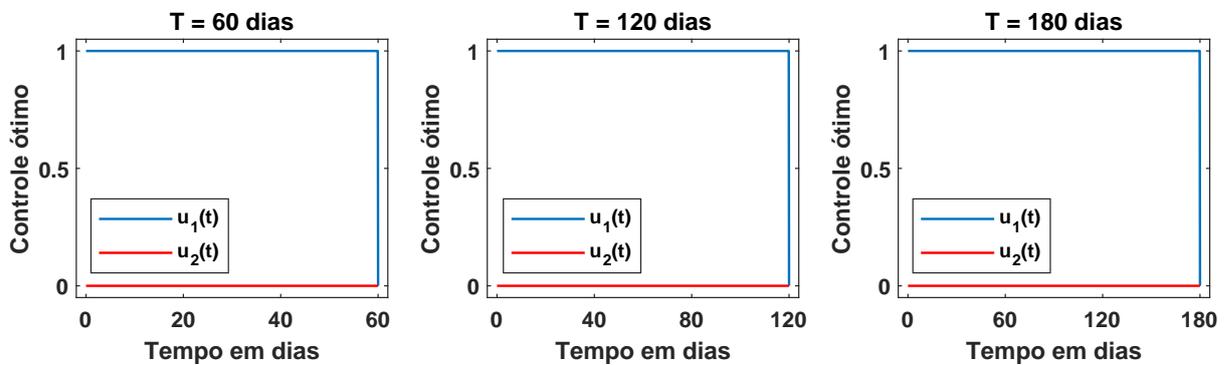


Figura 15 – Os controles ótimos para uso de duas vacinas com eficácia de 67% e 51% em diferentes durações de campanha de vacinação.

Nas Figuras 13, 14 e 15, independentemente do maior custo econômico, o controle ótimo indica o uso da vacina  $V_1$  com eficácia de 91%, 74% ou 67% em comparação com a vacina  $V_2$  com eficácia de 51%. Entendemos que isso é consequência do efeito da eficácia da vacina na taxa de transmissão da doença. De fato, para pessoas vacinadas, a taxa de transmissão diminui linearmente conforme segue na Figura 16. Neste contexto, veja que  $\beta_2 = 0,2378$  é uma taxa pequena comparada à taxa usual para não vacinados  $\beta = 0,45$ , todavia  $\beta_1 = 0,1601$ ,  $\beta_1 = 0,1262$  ou  $\beta_1 = 0,0436$  é muito menor quando comparado a  $\beta_2$ .

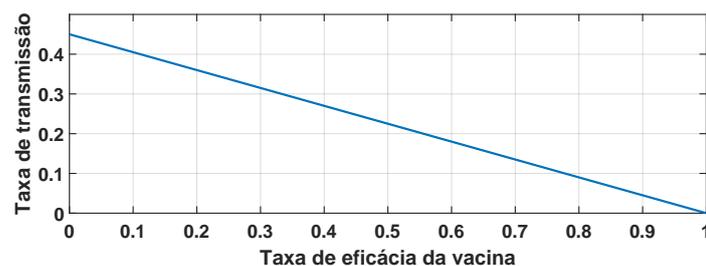


Figura 16 – Comportamento da taxa de transmissão para pessoas vacinadas de acordo com a eficácia da vacina.

Nas Figuras 17, 18 e 19 veja a curva de vacinados para os controles das Figuras 13, 14 e 15, respectivamente. O número de pessoas vacinadas com a vacina  $V_2$  é praticamente zero quando comparado ao tamanho da população suscetível, este é um resultado esperado considerando o que vimos nos gráficos dos controles ótimos. Após esta discussão, recomenda-se que nestes cenários, o governo compre apenas a vacina mais eficaz, a vacina com 51% de eficácia só será usada se for a única disponível.

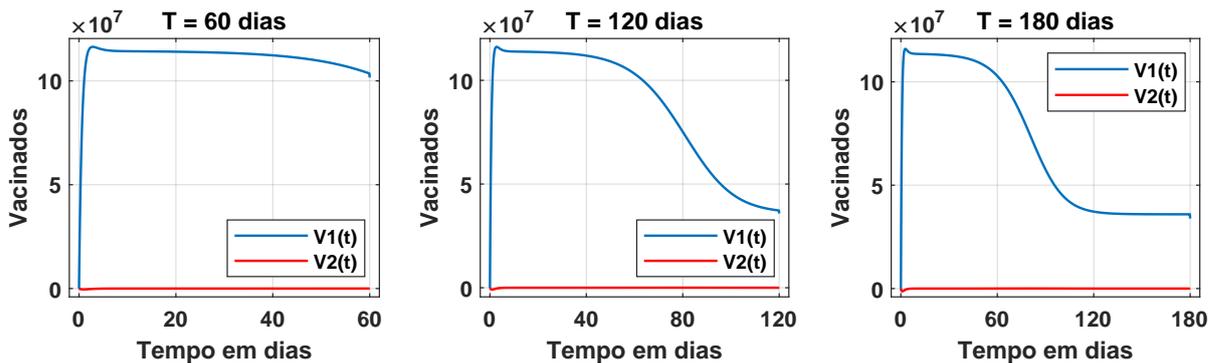


Figura 17 – Curvas de vacinados para uso de duas vacinas com eficácia de 91% e 51% em diferentes durações de campanha de vacinação.

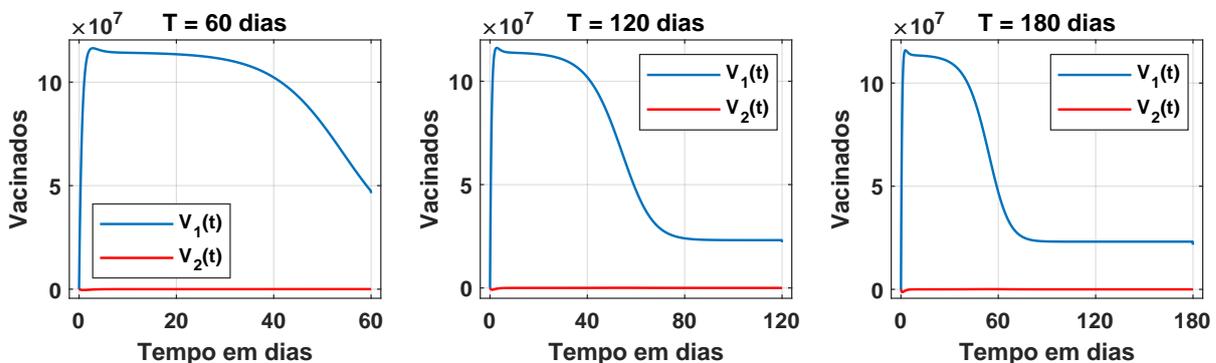


Figura 18 – Curvas de vacinados para uso de duas vacinas com eficácia de 74% e 51% em diferentes durações de campanha de vacinação.

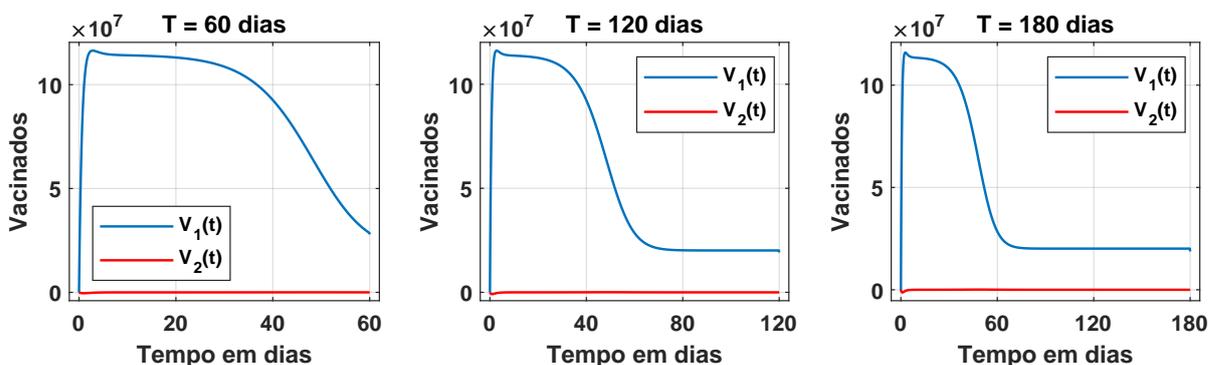


Figura 19 – Curvas de vacinados para uso de duas vacinas com eficácia de 67% e 51% em diferentes durações de campanha de vacinação.

Na Figura 20, durante quase todo o tempo da campanha, o controle ótimo indica o uso da vacina  $V_1$  com 91% de eficácia, em comparação com a vacina  $V_2$  com eficácia de 74%, mas havendo uso simultâneo entre essas duas vacinas a partir do dia 51, 109 e 169, respectivamente, para campanhas de vacinação com duração de 60, 120 e 180 dias. Quando a vacina  $V_2$  é 67% efetiva, veja na Figura 21 que o comportamento do controle ótimo é praticamente o mesmo da Figura 20 mas tendo uso simultâneo entre essas duas vacinas durante um período menor, acontecendo a partir do dia 57, 116 e 176, respectivamente, para campanhas de vacinação com duração de 60, 120 e 180 dias.

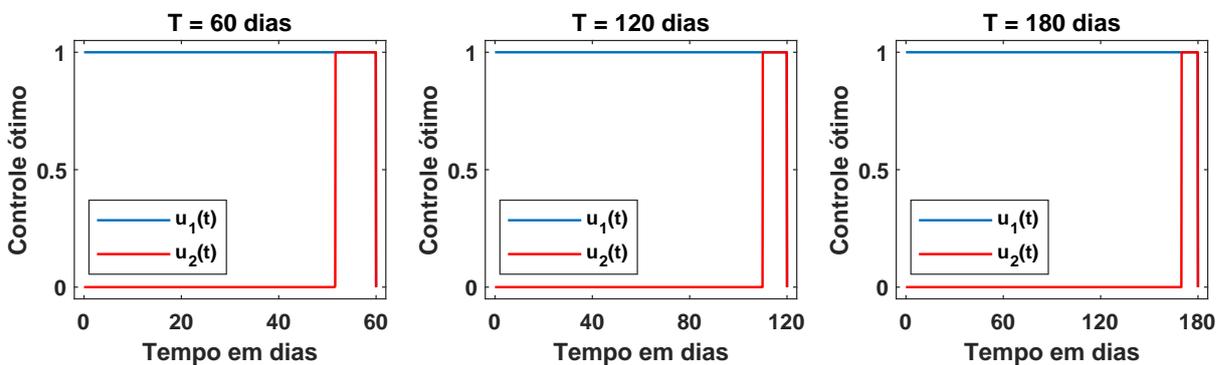


Figura 20 – Os controles ótimos para uso de duas vacinas com eficácia de 91% e 74% em diferentes durações de campanha de vacinação.

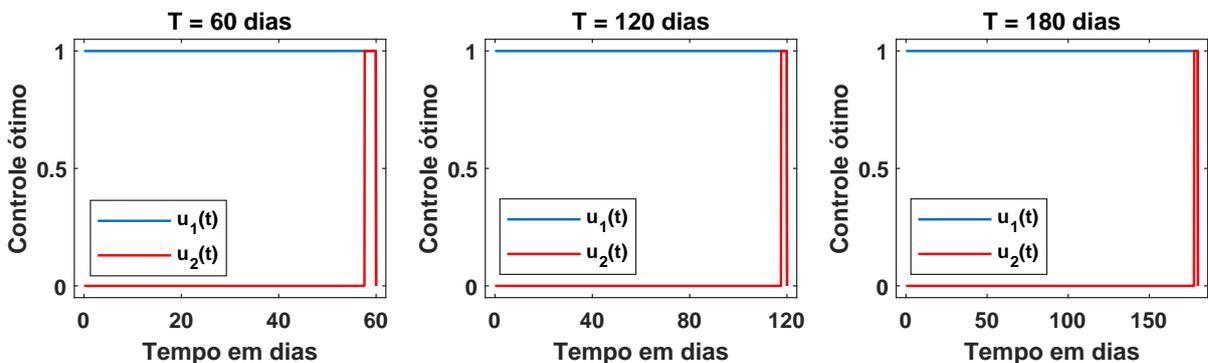


Figura 21 – Os controles ótimos para uso de duas vacinas com eficácia de 91% e 67% em diferentes durações de campanha de vacinação.

Entendemos que os controles ótimos das Figuras 20 e 21 buscam equilibrar os gastos econômicos, considerando que a vacina  $V_2$  tem menor custo de implementação. Além disso, a menor efetividade da vacina  $V_2$  não causará complicações durante o uso simultâneo, considerando que a população possivelmente já possa ter alcançado uma imunidade coletiva com a administração inicial da vacina  $V_1$ , que é mais efetiva.

Nas Figuras 22 e 23 veja a curva de pessoas vacinadas para os controles das Figuras 20 e 21, com o uso da vacina  $V_2$  determinado pelos controles, podemos ver um aumento no número de pessoas vacinadas com esta vacina nos últimos dias de campanha.

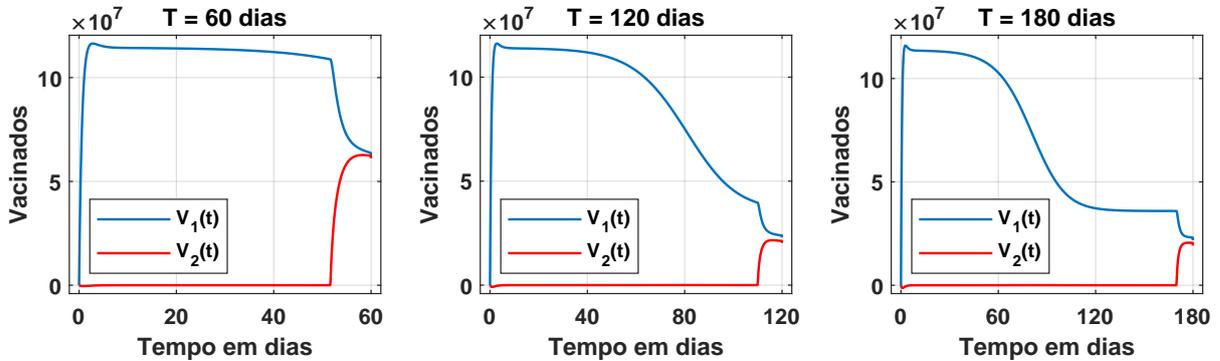


Figura 22 – Curvas de vacinados para uso de duas vacinas com eficácia de 91% e 74% em diferentes durações de campanha de vacinação.

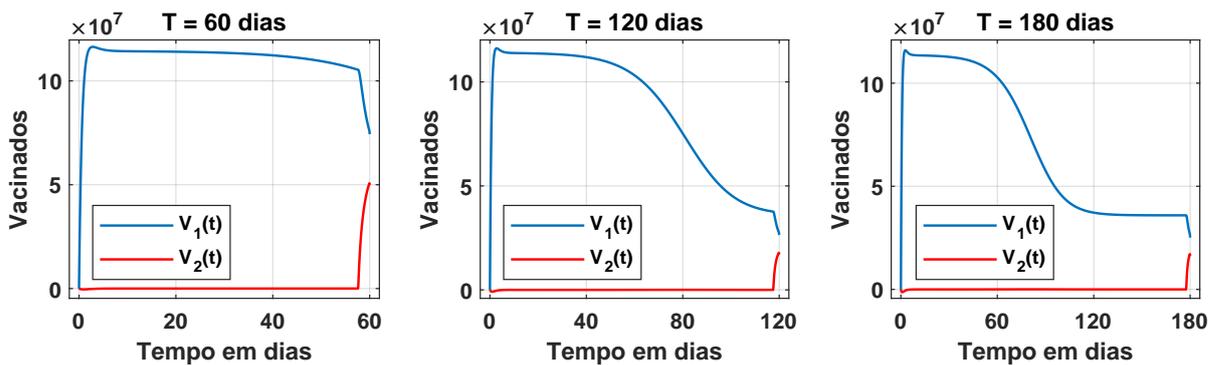


Figura 23 – Curvas de vacinados para uso de duas vacinas com eficácia de 91% e 67% em diferentes durações de campanha de vacinação.

Calculamos a integral das curvas de vacinados  $V_1(t)$  e  $V_2(t)$  para encontrar a quantidade de vacinas utilizadas durante a campanha, e assim recomendar uma distribuição aos órgãos governamentais para a compra de vacinas. Para isso, fizemos uso da regra trapezoidal para integração numérica, cujos detalhes podem ser vistos em (CHAPRA, 2018, p.494).

Tabela 10 – A distribuição para compra de vacinas de acordo com a Figura 22.

Vacina	T = 60 dias	T = 120 dias	T = 180 dias
$V_1$	93,44 %	98,18 %	98,55 %
$V_2$	6,66 %	1,82 %	1,45 %

Tabela 11 – A distribuição para compra de vacinas de acordo com a Figura 23.

Vacina	T = 60 dias	T = 120 dias	T = 180 dias
$V_1$	98,77 %	99,70 %	99,76 %
$V_2$	1,23 %	0,30 %	0,24 %

Para vacinas com eficácia de 74% e 67%, temos uma pequena diferença entre as taxas de eficácia  $\theta_1$  e  $\theta_2$ , e conseqüentemente entre as taxas de transmissão  $\beta_1$  e  $\beta_2$ . Neste caso, veja na Figura 24 que os gráficos de controle ótimo estão sobrepostos, indicando um uso simultâneo na administração das vacinas para todas as durações da campanha de vacinação.

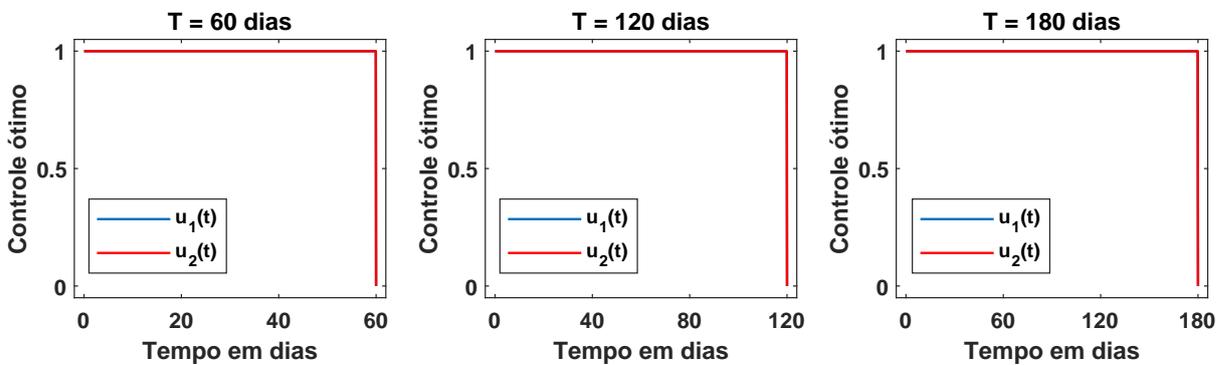


Figura 24 – Os controles ótimos para uso de duas vacinas com eficácia de 74% e 67% em diferentes durações de campanha de vacinação.

Veja na Figura 25, a curva de pessoas vacinadas com base no controle ótimo da Figura 24.

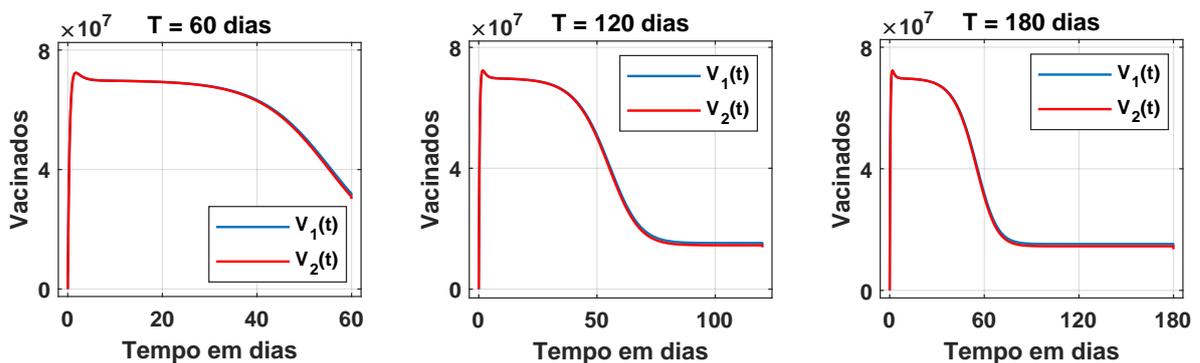


Figura 25 – Curvas de vacinados para uso de duas vacinas com eficácia de 74% e 67% em diferentes durações de campanha de vacinação.

Em todos os casos da Figura 25, o número de pessoas vacinadas com as vacinas  $V_1$  e  $V_2$  é praticamente o mesmo durante todo o período de vacinação. Neste caso, calculamos também a integral das curvas vacinadas  $V_1(t)$  e  $V_2(t)$ , veja a quantidade de vacinas utilizadas durante a campanha na Tabela 12.

Tabela 12 – A distribuição para compra de vacinas de acordo com a Figura 25.

Vacina	T = 60 dias	T = 120 dias	T = 180 dias
$V_1$	50,11 %	50,34 %	50,48 %
$V_2$	49,89 %	49,66 %	49,52 %

Seguindo os valores dos parâmetros adotados na Tabela 2, a Figura 24 mostra que o controle ótimo não indica preferência pelo uso de uma vacina em particular. Neste contexto, nos surge uma necessidade de verificar quais demais fatores podem gerar uma preferência no uso de uma vacina em detrimento a outra. Sendo assim, investigamos a sensibilidade da taxa de imunidade, da taxa de retorno à classe suscetível e das eficácias, parâmetros que no mundo real podem variar de acordo com características individuais das vacinas.

Tabela 13 – Valores de  $\alpha_1, \alpha_2, \varepsilon_1$  e  $\varepsilon_2$  para análise de sensibilidade.

$\alpha_1 > \alpha_2$ $\varepsilon_1 > \varepsilon_2$	$\alpha_1 < \alpha_2$ $\varepsilon_1 < \varepsilon_2$	$\alpha_1 > \alpha_2$ $\varepsilon_1 < \varepsilon_2$	$\alpha_1 < \alpha_2$ $\varepsilon_1 > \varepsilon_2$
$\alpha_1 = 0.088$ $\varepsilon_1 = 0.594$ $\alpha_2 = 0.08$ $\varepsilon_2 = 0.54$	$\alpha_1 = 0.08$ $\varepsilon_1 = 0.54$ $\alpha_2 = 0.008$ $\varepsilon_2 = 0.594$	$\alpha_1 = 0.088$ $\varepsilon_1 = 0.54$ $\alpha_2 = 0.08$ $\varepsilon_2 = 0.594$	$\alpha_1 = 0.08$ $\varepsilon_1 = 0.594$ $\alpha_2 = 0.088$ $\varepsilon_2 = 0.54$
$\alpha_1 = 0.096$ $\varepsilon_1 = 0.648$ $\alpha_2 = 0.08$ $\varepsilon_2 = 0.54$	$\alpha_1 = 0.08$ $\varepsilon_1 = 0.54$ $\alpha_2 = 0.096$ $\varepsilon_2 = 0.648$	$\alpha_1 = 0.096$ $\varepsilon_1 = 0.54$ $\alpha_2 = 0.08$ $\varepsilon_2 = 0.648$	$\alpha_1 = 0.08$ $\varepsilon_1 = 0.648$ $\alpha_2 = 0.096$ $\varepsilon_2 = 0.54$
$\alpha_1 = 0.08$ $\varepsilon_1 = 0.54$ $\alpha_2 = 0.072$ $\varepsilon_2 = 0.486$	$\alpha_1 = 0.072$ $\varepsilon_1 = 0.486$ $\alpha_2 = 0.08$ $\varepsilon_2 = 0.54$	$\alpha_1 = 0.08$ $\varepsilon_1 = 0.486$ $\alpha_2 = 0.072$ $\varepsilon_2 = 0.54$	$\alpha_1 = 0.072$ $\varepsilon_1 = 0.54$ $\alpha_2 = 0.08$ $\varepsilon_2 = 0.486$
$\alpha_1 = 0.08$ $\varepsilon_1 = 0.54$ $\alpha_2 = 0.064$ $\varepsilon_2 = 0.432$	$\alpha_1 = 0.064$ $\varepsilon_1 = 0.432$ $\alpha_2 = 0.08$ $\varepsilon_2 = 0.54$	$\alpha_1 = 0.08$ $\varepsilon_1 = 0.432$ $\alpha_2 = 0.064$ $\varepsilon_2 = 0.54$	$\alpha_1 = 0.064$ $\varepsilon_1 = 0.54$ $\alpha_2 = 0.08$ $\varepsilon_2 = 0.432$

Na Tabela 13, variamos os valores dos parâmetros  $\alpha_1, \alpha_2, \varepsilon_1$  e  $\varepsilon_2$ , para cima e para baixo em uma porcentagem de 10% e 20%, e os controles ótimos foram calculados para todos estes valores. Para todas as variações percentuais para cima, os controles ótimos têm o mesmo comportamento da Figura 24. Em variações para baixo, existe apenas um caso de mudança do controle, que é mostrado na Figura 26.

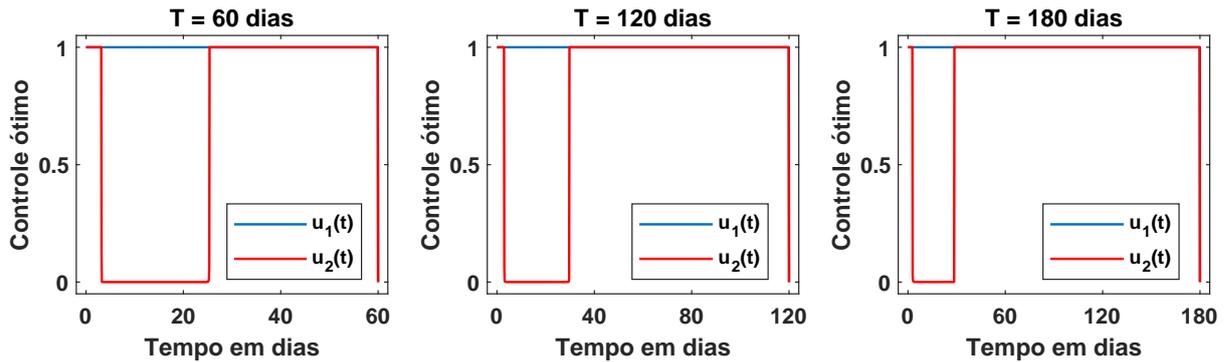


Figura 26 – Controles ótimos para  $\alpha_1 = 0.08, \alpha_2 = 0.064, \varepsilon_1 = 0.432$  e  $\varepsilon_2 = 0.54$ .

Na Figura 26, a redução no uso da vacina  $V_2$  é aceitável. Note que temos uma redução de 20% nos valores dos parâmetros  $\alpha_2$  e  $\varepsilon_1$ , esta mudança significa que a vacina  $V_2$  tem menor poder de recuperação e a vacina  $V_1$  tem menor quantidade de pessoas retornando à classe suscetível. Além disso, a variação para baixo de 20% é um limite, ou seja, uma redução de 19% não apresenta mudança no comportamento dos controles ótimos da Figura 24.

Para o controle da Figura 26, a curva de vacinados é mostrada na Figura 27, calculamos também a integral das curvas de vacinados. Veja a distribuição da quantidade de vacinas utilizadas durante a campanha, na Tabela 14.

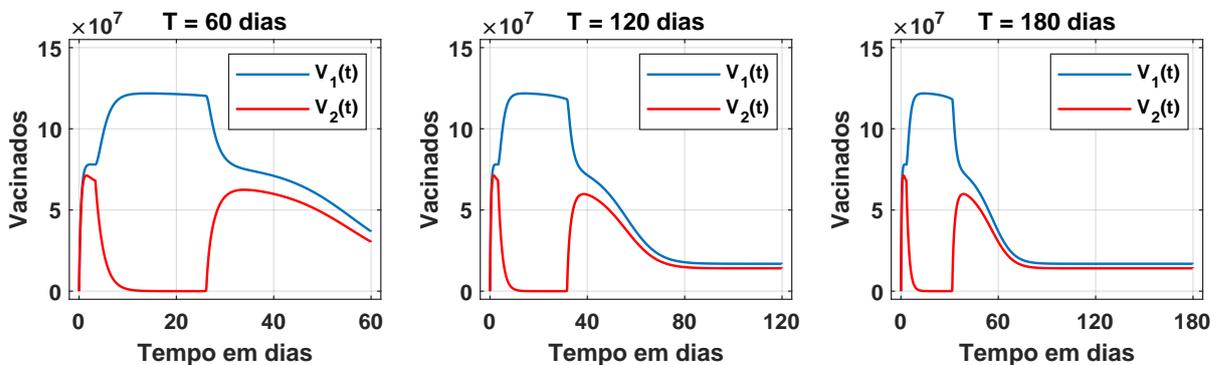


Figura 27 – Curvas de vacinados para  $\alpha_1 = 0.08, \alpha_2 = 0.064, \varepsilon_1 = 0.432$  e  $\varepsilon_2 = 0.54$ .

Tabela 14 – A distribuição para compra de vacinas de acordo com a Figura 26.

Vacina	T = 60 dias	T = 120 dias	T = 180 dias
$V_1$	71,60 %	71,21%	68,37 %
$V_2$	28,40 %	28,79%	31,63 %

Também variamos os parâmetros de eficácia  $\theta_1$  e  $\theta_2$  para fazer uma análise de sensibilidade dos mesmos. Em todos os cenários, o comportamento do controle ótimo  $u_1(t)$  visto na Figura

24 não muda, e o controle  $u_2(t)$  é alterado da seguinte forma:

- Fixado  $\theta_2 = 0.67$ , o controle  $u_2(t)$  muda de comportamento para valores de  $\theta_1$  acima de 0.77, isso é mostrado na Figura 28;
- Fixado  $\theta_1 = 0.74$ , o controle  $u_2(t)$  muda de comportamento para valores de  $\theta_2$  abaixo de 0.65, isso é mostrado na Figura 29.

Resumidamente, à medida que os parâmetros de eficácia ficam mais próximos, o uso simultâneo das vacinas é indicado, quando os valores começam a divergir, mantém-se a indicação do uso da vacina  $V_1$  com maior eficácia e diminui-se o uso da vacina  $V_2$ .

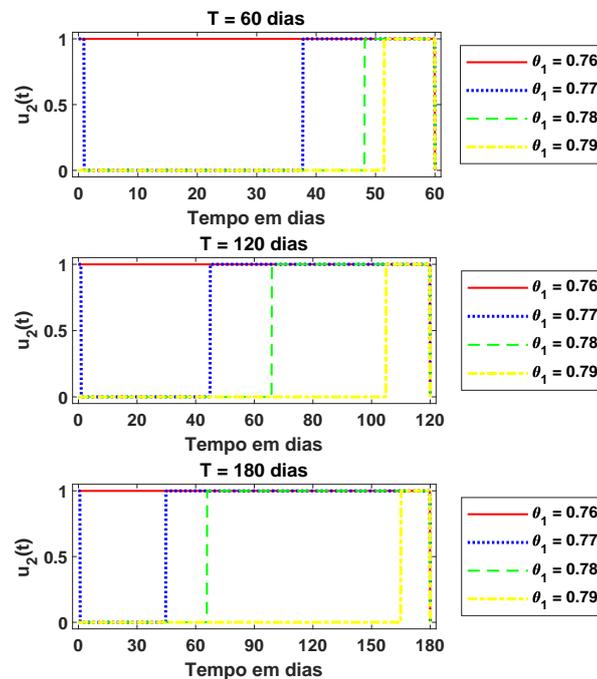


Figura 28 – Controle ótimo  $u_2(t)$  para  $\theta_2$  fixo.

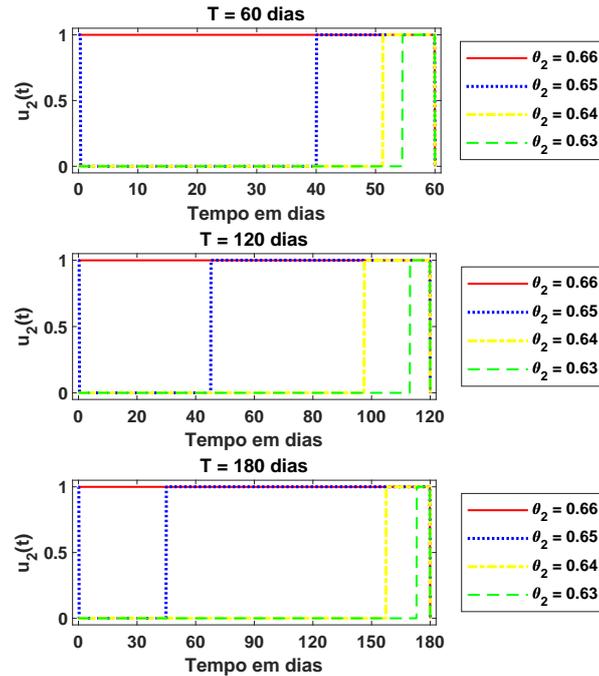


Figura 29 – Controle ótimo  $u_2(t)$  para  $\theta_1$  fixo.

Por fim, vamos investigar o comportamento do número de infectados. Até o momento, o controle ótimo apontou diversos cenários para o uso das vacinas. Nesta última discussão, vamos considerar o cenário com uso simultâneo das vacinas com 74% e 67% de eficácia, e veremos na Figura 30, as curvas de pessoas infectadas em três contextos: quando utilizamos apenas a vacina  $V_1$ , utilizando ambas as vacinas conforme determina o controle ótimo da Figura 24 e quando utilizamos apenas a vacina  $V_2$ .

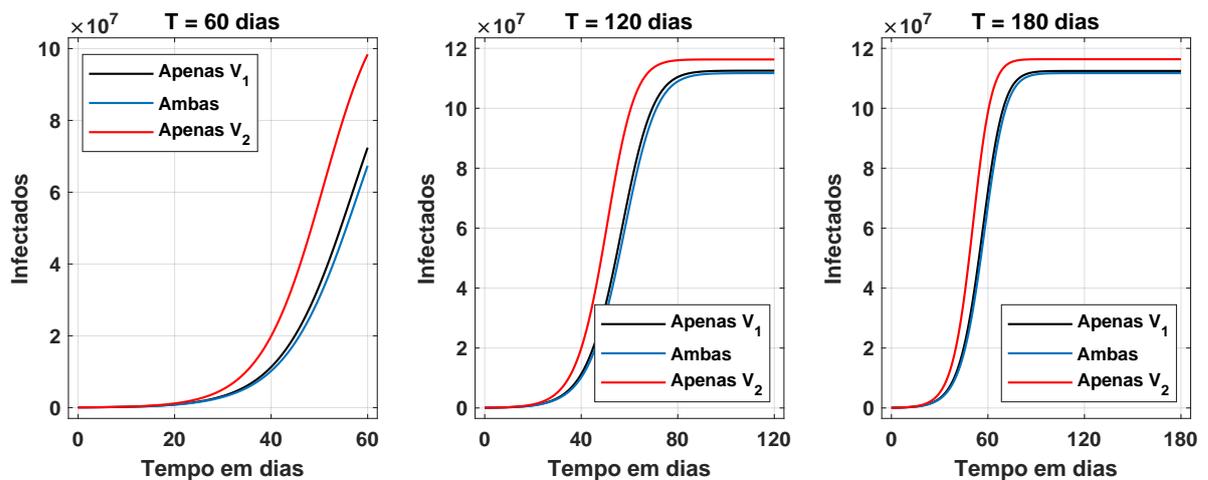


Figura 30 – Curvas de infectados para diferentes contextos de uso das vacinas.

Analisando a Figura 30, destacamos a importância econômica do controle ótimo. Quando usamos apenas a vacina  $V_2$ , temos o menor custo possível, porém um aumento considerável

no número de infectados quando comparado às outras duas estratégias. Note ainda que a curva de infectados tem praticamente o mesmo comportamento quando usamos as duas vacinas ou usando apenas a vacina  $V_1$ , que a saber, é mais eficaz e tem implementação mais cara para o governo. Neste caso, a estratégia de usar as duas vacinas definida pelo controle ótimo apresenta uma leve redução para o número de infectados, todavia com um custo acessível.

## 5 CONSIDERAÇÕES FINAIS

Neste trabalho, investigamos um modelo SEIRQ estruturado por idade, onde os parâmetros de entrada das pessoas infectadas na quarentena são considerados controles do sistema, e um modelo SEIRV para campanha de vacinação com dois tipos de vacinas, onde os controles são parâmetros de uso das vacinas segundo a sua eficácia. Encontramos os controles ótimos por meio do princípio do máximo de Pontryagin e usamos o método de varredura para frente e para trás para calculá-los numericamente. Conseguimos extrair inúmeras informações quanto ao uso do controle ótimo em uma estratégia farmacológica (vacinação) e outra não farmacológica (quarentena), usadas para contenção de doenças.

No programa de quarentena das pessoas infectadas, analisamos como o controle influencia os resultados: no número de infectados, no atraso da implementação do programa, nos calendários de relaxamento, e no número de mortes ao final da quarentena. O controle ótimo indica um bom tempo de relaxamento para os idosos e um tempo mínimo para os adultos e crianças. Uma demora de 10 ou 20 dias no início da quarentena não causa atraso significativo no relaxamento para as crianças ou para os adultos, todavia, para o grupo dos idosos, o tempo de isolamento aumenta significativamente. Em todos os casos investigados, o número de indivíduos infectados atingiu um mínimo pouco antes do fim da simulação, e os casos começaram a subir novamente, embora tenham atingido um número muito menor do que o de início, isso mostra que a quarentena foi efetiva, mas poderia ter tido uma duração maior. Em comparação ao mesmo período de tempo sem quarentena, os controles ótimos produziram uma redução no número de mortes em 126.4, 120.5 e 111.9 vezes, para campanhas de 60, 90 e 120 dias, respectivamente.

Na campanha de vacinação, utilizamos quatro valores para eficácia das vacinas, a saber 91%, 74%, 67%, e 51%, e investigamos os controles ótimos para cada par destes valores. Com base no que foi indicado pelo controle ótimo, mostramos como a compra das vacinas deve ser distribuída e como elas devem ser usadas. Quando os valores de eficácia estavam muito próximos, apresentamos uma análise de sensibilidade para os parâmetros. Avaliamos ainda o comportamento dos números de recuperados e infectados quando usamos as vacinas em três contextos: apenas a primeira vacina, ambas de acordo com o controle ótimo, e apenas a segunda vacina. A menos que tenhamos uma vacina com eficácia muito baixa (51% por exemplo, onde o uso não é indicado) ou duas vacinas com eficácias muito próximas (onde o uso simultâneo é indicado para todo o tempo), os controles ótimos indicam um comportamento padrão: uso

inicial da vacina mais eficaz seguido de uso simultâneo. O contexto em que consideramos o uso das duas vacinas de acordo com o controle ótimo, apresenta menor número de infectados, quando comparado com os contextos em que usamos somente uma das vacinas, e isto ocorre minimizando os custos econômicos.

Em nossos modelos, utilizamos dados da COVID-19 no Brasil para condições iniciais e dados da literatura para valores dos parâmetros. Esse tipo de estudo pode ser feito se forma a investigar uma campanha de vacinação para outras doenças, adaptando o modelo de acordo com suas características, realizando uma estimativa adequada nos parâmetros, e com condições iniciais que se adequem ao local (como o Brasil, por exemplo, é muito grande, pode haver sub notificações e outros fatores que causam ruído nos dados).

Finalmente, esperamos que os nossos resultados possam sugerir decisões governamentais ao implementar campanhas de vacinação ou que possam ajudar nas decisões que os governos devem tomar ao implementar políticas de quarentena.

## REFERÊNCIAS

- ALEXANDER, M. E.; BOWMAN, C.; MOGHADAS, S. M.; SUMMERS, R.; GUMEL, A. B.; SAHAI, B. M. A vaccination model for transmission dynamics of influenza. *SIAM Journal on Applied Dynamical Systems*, v. 3, n. 4, p. 503–524, 2004.
- ANNAS, S.; Isbar Pratama, M.; RIFANDI, M.; SANUSI, W.; SIDE, S. Stability analysis and numerical simulation of seir model for pandemic covid-19 spread in indonesia. *Chaos, Solitons & Fractals*, v. 139, p. 110072, 2020. ISSN 0960-0779. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0960077920304690>>.
- ANOSOV, D. V. On the mathematical work of I.S. pontryagin. *Proceedings of the Steklov Institute of Mathematics*, Pleiades Publishing, Ltd, v. 268, p. 5,16, 2010.
- ARINO, J.; DRIESSCHE, P. van den. The basic reproduction number in a multi-city compartmental epidemic model. In: BENVENUTI, L.; SANTIS, A. D.; FARINA, L. (Ed.). *Positive Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003. p. 135–142.
- ARNON, A.; RICCO, J.; SMETTERS, K. Epidemiological and economic effects of lockdown. *Brookings Papers on Economic Activity*, Johns Hopkins University Press, v. 2020, n. 3, p. 61–108, 2020.
- ASAHI, K.; UNDURRAGA, E. A.; VALDÉS, R.; WAGNER, R. The effect of covid-19 on the economy: Evidence from an early adopter of localized lockdowns. *Journal of global health*, International Society for Global Health, v. 11, 2021.
- BACAËR, N.; OUIFKI, R. Growth rate and basic reproduction number for population models with a simple periodic factor. *Mathematical Biosciences*, v. 210, n. 2, p. 647–658, 2007. ISSN 0025-5564. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0025556407001526>>.
- BALCIK, B.; YUCESOY, E.; AKCA, B.; KARAKAYA, S.; GEVSEK, A. A.; BAHARMAND, H.; SGARBOSSA, F. A mathematical model for equitable in-country covid-19 vaccine allocation. *International Journal of Production Research*, Taylor & Francis, v. 60, n. 24, p. 7502–7526, 2022.
- BARKUR, G.; KAMATH, G. B. et al. Sentiment analysis of nationwide lockdown due to covid 19 outbreak: Evidence from india. *Asian journal of psychiatry*, Elsevier, v. 51, p. 102089, 2020.
- BARTLE, R. *The Elements of Integration and Lebesgue Measure*. Wiley, 2014. (Wiley Classics Library). ISBN 9781118626122. Disponível em: <<https://books.google.com.br/books?id=aE1YBAAAQBAJ>>.
- BARTSCH, S. M.; O'SHEA, K. J.; FERGUSON, M. C.; BOTTAZZI, M. E.; WEDLOCK, P. T.; STRYCH, U.; MCKINNELL, J. A.; SIEGMUND, S. S.; COX, S. N.; HOTEZ, P. J.; LEE, B. Y. Vaccine efficacy needed for a covid-19 coronavirus vaccine to prevent or stop an epidemic as the sole intervention. *American Journal of Preventive Medicine*, v. 59, n. 4, p. 493–503, 2020. ISSN 0749-3797.
- BEHNCKE, H. Optimal control of deterministic epidemics. *Optimal control applications and methods*, Wiley Online Library, v. 21, n. 6, p. 269–285, 2000.

- BJØRNSTAD, O. N.; SHEA, K.; KRZYWINSKI, M.; ALTMAN, N. The seirs model for infectious disease dynamics. *Nature Methods*, v. 17, n. 6, p. 557–558, 2020. Disponível em: <<https://doi.org/10.1038/s41592-020-0856-2>>.
- BRAUER, F. The kermack–mckendrick epidemic model revisited. *Mathematical Biosciences*, v. 198, n. 2, p. 119–131, 2005. ISSN 0025-5564. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0025556405001331>>.
- BUBAR, K. M.; REINHOLT, K.; KISSLER, S. M.; LIPSITCH, M.; COBEY, S.; GRAD, Y. H.; LARREMORE, D. B. Model-informed covid-19 vaccine prioritization strategies by age and serostatus. *Science*, v. 371, n. 6532, p. 916–921, 2021.
- CAI, Y.; KANG, Y.; WANG, W. A stochastic sirs epidemic model with nonlinear incidence rate. *Applied Mathematics and Computation*, v. 305, p. 221–240, 2017. ISSN 0096-3003. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0096300317300991>>.
- CARCIONE, J. M.; SANTOS, J. E.; BAGAINI, C.; BA, J. A simulation of a covid-19 epidemic based on a deterministic seir model. *Frontiers in Public Health*, v. 8, 2020. ISSN 2296-2565. Disponível em: <<https://www.frontiersin.org/articles/10.3389/fpubh.2020.00230>>.
- CASTILHO, C.; GONDIM, J.; MARCHESIN, M.; SABETI, M. Assessing the efficiency of different control strategies for the covid-19 epidemic. *Texas State University, Department of Mathematics*, 2020.
- CHAPRA, S. *Applied Numerical Methods with MATLAB for Engineers and Scientists*. McGraw-Hill Education, 2018. ISBN 9781260083255. Disponível em: <<https://books.google.com.br/books?id=4JldtAEACAAJ>>.
- CHEN, Y.-C.; LU, P.-E.; CHANG, C.-S.; LIU, T.-H. A time-dependent sir model for covid-19 with undetectable infected persons. *IEEE Transactions on Network Science and Engineering*, v. 7, n. 4, p. 3279–3294, 2020.
- CHOWELL, G.; SATTENSPIEL, L.; BANSAL, S.; VIBOUD, C. Mathematical models to characterize early epidemic growth: A review. *Physics of Life Reviews*, v. 18, p. 66–97, 2016. ISSN 1571-0645.
- COOKE, K.; DRIESSCHE, P. van den. Analysis of an seirs epidemic model with two delays. *Journal of mathematical biology*, v. 35, n. 2, p. 240–260, December 1996. ISSN 0303-6812. Disponível em: <<https://doi.org/10.1007/s002850050051>>.
- COOPER, I.; MONDAL, A.; ANTONOPOULOS, C. G. A sir model assumption for the spread of covid-19 in different communities. *Chaos, Solitons & Fractals*, v. 139, p. 110057, 2020. ISSN 0960-0779. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0960077920304549>>.
- DIETZ, K. The estimation of the basic reproduction number for infectious diseases. *Statistical methods in medical research*, Sage Publications Sage CA: Thousand Oaks, CA, v. 2, n. 1, p. 23–41, 1993.
- DJIDJOU-DEMASSE, R.; MICHALAKIS, Y.; CHOISY, M.; SOFONEA, M. T.; ALIZON, S. Optimal covid-19 epidemic control until vaccine deployment. *MedRxiv*, Cold Spring Harbor Laboratory Press, p. 2020–04, 2020.

- DOMENICO, L. D.; PULLANO, G.; SABBATINI, C. E.; BOËLLE, P.-Y.; COLIZZA, V. Impact of lockdown on covid-19 epidemic in île-de-france and possible exit strategies. *BMC medicine*, Springer, v. 18, p. 1–13, 2020.
- EHRETH, J. The global value of vaccination. *Vaccine*, Elsevier, v. 21, n. 7-8, p. 596–600, 2003.
- ESPAÑA. Actualización nº 104. enfermedad por el coronavirus (covid-19). 13.05.2020. *Gobierno de España, Ministerio de Sanidad, Centro de coordinación de alertas y emergencias sanitarias*, 2020.
- FAKHROO, A.; ALKHATIB, H. A.; THANI, A. A. A.; YASSINE, H. M. Reinfections in covid-19 patients: Impact of virus genetic variability and host immunity. *Vaccines*, v. 9, n. 10, 2021. ISSN 2076-393X.
- FERGUSON, N. M.; LAYDON, D.; NEDJATI-GILANI, G.; IMAI, N.; AINSLIE, K.; BAGUELIN, M.; BHATIA, S.; BOONYASIRI, A.; CUCUNUBÁ, Z.; CUOMO-DANNENBURG, G. et al. *Report 9: Impact of non-pharmaceutical interventions (NPIs) to reduce COVID19 mortality and healthcare demand*. [S.l.]: Imperial College London London, 2020. v. 16.
- FISTER, K. R.; LENHART, S.; MCNALLY, J. S. Optimizing chemotherapy in an hiv model. *Electronic Journal of Differential Equations*, Southwest Texas State University, Department of Mathematics, 1998.
- GANI, J. Anderson gray mckendrick. In: \_\_\_\_\_. *Statisticians of the Centuries*. New York, NY: Springer New York, 2001. p. 323–327. ISBN 978-1-4613-0179-0. Disponible em: <[https://doi.org/10.1007/978-1-4613-0179-0\\_69](https://doi.org/10.1007/978-1-4613-0179-0_69)>.
- GANI, J. Daniel bernoulli. In: \_\_\_\_\_. *Statisticians of the Centuries*. New York, NY: Springer New York, 2001. p. 64–67. ISBN 978-1-4613-0179-0. Disponible em: <[https://doi.org/10.1007/978-1-4613-0179-0\\_12](https://doi.org/10.1007/978-1-4613-0179-0_12)>.
- GONDIM, J. A. Preventing epidemics by wearing masks: An application to covid-19. *Chaos, Solitons & Fractals*, v. 143, p. 110599, 2021. ISSN 0960-0779. Disponible em: <<https://www.sciencedirect.com/science/article/pii/S0960077920309905>>.
- GONDIM, J. A.; MACHADO, L. Optimal quarantine strategies for the covid-19 pandemic in a population with a discrete age structure. *Chaos, Solitons & Fractals*, v. 140, p. 110166, 2020. ISSN 0960-0779.
- GRASSLY, N. C.; PONS-SALORT, M.; PARKER, E. P.; WHITE, P. J.; FERGUSON, N. M.; AINSLIE, K.; BAGUELIN, M.; BHATT, S.; BOONYASIRI, A.; BRAZEAU, N. et al. Comparison of molecular testing strategies for covid-19 control: a mathematical modelling study. *The Lancet Infectious Diseases*, Elsevier, v. 20, n. 12, p. 1381–1389, 2020.
- HALL, V.; FOULKES, S.; INSALATA, F.; KIRWAN, P.; SAEI, A.; ATTI, A.; WELLINGTON, E.; KHAWAM, J.; MUNRO, K.; COLE, M. et al. Protection against sars-cov-2 after covid-19 vaccination and previous infection. *New England Journal of Medicine*, Mass Medical Soc, v. 386, n. 13, p. 1207–1220, 2022.
- HAMRA, G.; MACLEHOSE, R.; RICHARDSON, D. Markov chain monte carlo: an introduction for epidemiologists. *International journal of epidemiology*, Oxford University Press, v. 42, n. 2, p. 627–634, 2013.

HELLEWELL, J.; ABBOTT, S.; GIMMA, A.; BOSSE, N.; JARVIS, C.; RUSSELL, T.; MUNDAY, J.; KUCHARSKI, A.; EDMUNDS, W.; SUN, F. et al. *Feasibility of controlling COVID-19 outbreaks by isolation of cases and contacts. Lancet Glob Health.* 2020; 8 (4): e488–96. 2020.

HORTON, R. *The COVID-19 Catastrophe: What's Gone Wrong and How to Stop it Happening Again.* Cambridge: Polity Press, 2020. ISBN 9781509546466.

ISRAEL, G.; GASCA, A. M. Letters between william o. kermack and vito volterra. In: \_\_\_\_\_. *The Biology of Numbers: The Correspondence of Vito Volterra on Mathematical Biology.* Basel: Birkhäuser Basel, 2002. p. 221–222. ISBN 978-3-0348-8123-4. Disponível em: <[https://doi.org/10.1007/978-3-0348-8123-4\\_11](https://doi.org/10.1007/978-3-0348-8123-4_11)>.

JI, C.; JIANG, D. Threshold behaviour of a stochastic sir model. *Applied Mathematical Modelling*, v. 38, n. 21, p. 5067–5079, 2014. ISSN 0307-904X. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0307904X14001401>>.

JIN, Y.; WANG, W.; XIAO, S. An sirs model with a nonlinear incidence rate. *Chaos, Solitons & Fractals*, v. 34, n. 5, p. 1482–1497, 2007. ISSN 0960-0779. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0960077906003882>>.

JOSHI, H. R. Optimal control of an hiv immunology model. *Optimal control applications and methods*, Wiley Online Library, v. 23, n. 4, p. 199–213, 2002.

JUNG, E.; LENHART, S.; FENG, Z. Optimal control of treatments in a two-strain tuberculosis model. *Discrete and Continuous Dynamical Systems Series B*, AIMS PRESS, v. 2, n. 4, p. 473–482, 2002.

KATUL, G. G.; MRAD, A.; BONETTI, S.; MANOLI, G.; PAROLARI, A. J. Global convergence of covid-19 basic reproduction number and estimation from early-time sir dynamics. *PLOS ONE*, Public Library of Science, v. 15, n. 9, p. 1–22, 09 2020. Disponível em: <<https://doi.org/10.1371/journal.pone.0239800>>.

KERMACK, W. O.; MCKENDRICK, A. G. A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, The Royal Society, v. 115, n. 772, p. 700–721, 1927. ISSN 09501207. Disponível em: <<http://www.jstor.org/stable/94815>>.

KIRSCHNER, D.; LENHART, S.; SERBIN, S. Optimal control of the chemotherapy of hiv. *Journal of mathematical biology*, Springer, v. 35, p. 775–792, 1997.

LAU, H.; KHOSRAWIPOUR, V.; KOCBACH, P.; MIKOLAJCZYK, A.; SCHUBERT, J.; BANIA, J.; KHOSRAWIPOUR, T. The positive impact of lockdown in wuhan on containing the covid-19 outbreak in china. *Journal of travel medicine*, Oxford University Press, v. 27, n. 3, p. taaa037, 2020.

LAUER, S. A.; GRANTZ, K. H.; BI, Q.; AL et. The incubation period of coronavirus disease 2019 (covid-19) from publicly reported confirmed cases: Estimation and application. *Annals of Internal Medicine*, v. 172, n. 9, p. 577–582, 2020.

LEE, S.; CHOWELL, G.; CASTILLO-CHÁVEZ, C. Optimal control for pandemic influenza: the role of limited antiviral treatment and isolation. *Journal of Theoretical Biology*, Elsevier, v. 265, n. 2, p. 136–150, 2010.

LENHART, S.; WORKMAN, J. *Optimal Control Applied to Biological Models*. Chapman and Hall/CRC, 2007. Disponível em: <<https://doi.org/10.1201/9781420011418>>.

LI, C.-H.; TSAI, C.-C.; YANG, S.-Y. Analysis of epidemic spreading of an sirs model in complex heterogeneous networks. *Communications in Nonlinear Science and Numerical Simulation*, v. 19, n. 4, p. 1042–1054, 2014. ISSN 1007-5704. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1007570413003833>>.

LI, M. Y.; GRAEF, J. R.; WANG, L.; KARSAL, J. Global dynamics of a seir model with varying total population size. *Mathematical Biosciences*, v. 160, n. 2, p. 191–213, 1999. ISSN 0025-5564. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0025556499000309>>.

LIU, J.; ZHANG, T. Epidemic spreading of an seirs model in scale-free networks. *Communications in Nonlinear Science and Numerical Simulation*, v. 16, n. 8, p. 3375–3384, 2011. ISSN 1007-5704. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1007570410006088>>.

LIU, X.; TAKEUCHI, Y.; IWAMI, S. Svir epidemic models with vaccination strategies. *Journal of Theoretical Biology*, v. 253, n. 1, p. 1–11, 2008. ISSN 0022-5193.

LLOYD, A. L. Sensitivity of model-based epidemiological parameter estimation to model assumptions. *Mathematical and statistical estimation approaches in epidemiology*, Springer, p. 123–141, 2009.

LOGEMANN, H.; RYAN, E. *Ordinary Differential Equations: Analysis, Qualitative Theory and Control*. Springer London, 2014. (Springer Undergraduate Mathematics Series). ISBN 9781447163978. Disponível em: <<https://books.google.com.br/books?id=4thHngEACAAJ>>.

MACKI, J.; STRAUSS, A. *Introduction to Optimal Control Theory*. Springer-Verlag, 1982. (Undergraduate texts in mathematics). ISBN 9783540906247. Disponível em: <<https://books.google.com.br/books?id=5QSmQgAACAAJ>>.

MAKHOUL, M.; AYOUB, H. H.; CHEMAITELLY, H.; SEEDAT, S.; MUMTAZ, G. R.; AL-OMARI, S.; ABU-RADDAD, L. J. Epidemiological impact of sars-cov-2 vaccination: Mathematical modeling analyses. *Vaccines*, v. 8, n. 4, 2020. ISSN 2076-393X.

MARTCHEVA, M. *An Introduction to Mathematical Epidemiology*. Springer US, 2015. (Texts in Applied Mathematics). ISBN 9781489976123. Disponível em: <<https://books.google.com.br/books?id=tt7HCgAAQBAJ>>.

MATEUS, J. P.; REBELO, P.; ROSA, S.; SILVA, C. M.; TORRES, D. F. Optimal control of non-autonomous seirs models with vaccination and treatment. *arXiv preprint arXiv:1706.06843*, 2017.

MOHAMMED-AWEL, J.; NUMFOR, E.; ZHAO, R.; LENHART, S. A new mathematical model studying imperfect vaccination: Optimal control analysis. *Journal of Mathematical Analysis and Applications*, v. 500, n. 2, p. 125132, 2021. ISSN 0022-247X.

MOORE, S.; HILL, E. M.; DYSON, L.; TILDESLEY, M. J.; KEELING, M. J. Vaccination and non-pharmaceutical interventions for covid-19: a mathematical modelling study. *The Lancet Infectious Diseases*, v. 21, p. 793–802, 2021.

- NKAMBA THOMAS TIMOTHEE MANGA, F. A. L. N.; MANYOMBE, M. L. M. Mathematical model to assess vaccination and effective contact rate impact in the spread of tuberculosis. *Journal of Biological Dynamics*, Taylor & Francis, v. 13, n. 1, p. 26–42, 2019.
- NYE, E.; GIBSON, M. *Ronald Ross: Malariologist and Polymath : a Biography*. St. Martin's Press, 1997. ISBN 9780333625514. Disponível em: <<https://books.google.com.br/books?id=BISBQgAACAAJ>>.
- OBSU, L. L.; BALCHA, S. F. Optimal control strategies for the transmission risk of covid-19. *Journal of biological dynamics*, Taylor & Francis, v. 14, n. 1, p. 590–607, 2020.
- OLIVARES, A.; STAFFETTI, E. Uncertainty quantification of a mathematical model of covid-19 transmission dynamics with mass vaccination strategy. *Chaos, Solitons & Fractals*, v. 146, p. 110895, 2021. ISSN 0960-0779.
- ORENSTEIN, W. A.; AHMED, R. Simply put: Vaccination saves lives. *Proceedings of the National Academy of Sciences*, v. 114, n. 16, p. 4031–4033, 2017.
- PADULA, W. V. *Why only test symptomatic patients? Consider random screening for COVID-19*. [S.l.]: Springer, 2020. 333–334 p.
- PAUL, J. N.; MBALAWATA, I. S.; MIRAU, S. S.; MASANDAWA, L. Mathematical modeling of vaccination as a control measure of stress to fight covid-19 infections. *Chaos, Solitons & Fractals*, v. 166, p. 112920, 2023. ISSN 0960-0779.
- PEPE, E.; BAJARDI, P.; GAUVIN, L.; PRIVITERA, F.; LAKE, B.; CATTUTO, C.; TIZZONI, M. Covid-19 outbreak response, a dataset to assess mobility changes in italy following national lockdown. *Scientific data*, Nature Publishing Group UK London, v. 7, n. 1, p. 230, 2020.
- PONTRYAGIN, L.; BOLTYANSKII, V.; GAMKRELIDZE, R.; MISHCHENKO, E. *Mathematical Theory of Optimal Processes*. [S.l.]: Gorgon and Breach Science Publishers, 1986. v. 4.
- PRICKETT, K. C.; FLETCHER, M.; CHAPPLE, S.; DOAN, N.; SMITH, C. Life in lockdown: The economic and social effect of lockdown during alert level 4 in new zealand. *Te Herenga Waka—Victoria University of Wellington*, 2020.
- RAHMAN, S.; RAHMAN, M. M.; MIAH, M.; BEGUM, M. N.; SARMIN, M.; MAHFUZ, M.; HOSSAIN, M. E.; RAHMAN, M. Z.; CHISTI, M. J.; AHMED, T. et al. Covid-19 reinfections among naturally infected and vaccinated individuals. *Scientific reports*, Nature Publishing Group UK London, v. 12, n. 1, p. 1438, 2022.
- RODRIGUES, C. M.; PLOTKIN, S. A. Impact of vaccines; health, economic and social perspectives. *Frontiers in microbiology*, Frontiers Media SA, v. 11, p. 1526, 2020.
- RYAN, J.; ZOELLNER, Y.; GRADL, B.; PALACHE, B.; MEDEMA, J. Establishing the health and economic impact of influenza vaccination within the european union 25 countries. *Vaccine*, Elsevier, v. 24, n. 47-48, p. 6812–6822, 2006.
- SHEN, Z.-H.; CHU, Y.-M.; KHAN, M. A.; MUHAMMAD, S.; AL-HARTOMY, O. A.; HIGAZY, M. Mathematical modeling and optimal control of the covid-19 dynamics. *Results in Physics*, Elsevier, v. 31, p. 105028, 2021.

- SILVA, C. J.; TORRES, D. F. Optimal control for a tuberculosis model with reinfection and post-exposure interventions. *Mathematical Biosciences*, Elsevier, v. 244, n. 2, p. 154–164, 2013.
- São Paulo State - Decreto n. 64.881, de 22 de Março de 2020. 2023. Accessed on November 14, 2023. Disponível em: <<https://www.al.sp.gov.br/norma/193361>>.
- TENTORI, K.; PASSERINI, A.; TIMBERLAKE, B.; PIGHIN, S. The misunderstanding of vaccine efficacy. *Social Science & Medicine*, v. 289, p. 114273, 2021. ISSN 0277-9536.
- WALLINGA, J.; LIPSITCH, M. How generation intervals shape the relationship between growth rates and reproductive numbers. *Proceedings of the Royal Society B: Biological Sciences*, v. 274, p. 599 – 604, 2007. Disponível em: <<https://api.semanticscholar.org/CorpusID:903823>>.
- WATSON, O. J.; BARNESLEY, G.; TOOR, J.; HOGAN, A. B.; WINSKILL, P.; GHANI, A. C. Global impact of the first year of covid-19 vaccination: a mathematical modelling study. *The Lancet Infectious Diseases*, v. 22, p. 1293–1302, 2022.
- WEISS, H. H. The sir model and the foundations of public health. *Materials matematics*, p. 01–17, 2013.
- WHO. Criteria for releasing covid-19 patients from isolation. *World Health Organization*, 2020.
- WHO. Coronavirus disease 2019 (covid-19) situation report–51. *World Health Organization*, 2022.
- WHO. Interim recommendations for the use of the janssen ad26.cov2.s (covid-19) vaccine. *World Health Organization*, 2022.
- WHO. Interim recommendations for use of the chadox1-s [recombinant] vaccine against covid-19 (astrazeneca covid-19 vaccine azd1222 vaxzevria™, sii covishield™). *World Health Organization*, 2022.
- WHO. Interim recommendations for use of the inactivated covid-19 vaccine, coronavac, developed by sinovac. *World Health Organization*, 2022.
- WHO. Interim recommendations for use of the pfizer biontech covid 19 vaccine bnt162b2 , under emergency use listing. *World Health Organization*, 2022.
- WHO. Novel coronavirus disease (2019-ncov) situation report–1. *World Health Organization*, 2022.
- WORLDMETERS. 2024. Accessed on November 10, 2023. Disponível em: <<https://www.worldometers.info/coronavirus/>>.
- WU, J. T.; LEUNG, K.; LEUNG, G. M. Nowcasting and forecasting the potential domestic and international spread of the 2019-ncov outbreak originating in wuhan, china: a modelling study. *The lancet*, Elsevier, v. 395, n. 10225, p. 689–697, 2020.
- WU, Y.-C.; CHEN, C.-S.; CHAN, Y.-J. The outbreak of covid-19: An overview. *Journal of the Chinese medical association*, LWW, v. 83, n. 3, p. 217–220, 2020.

ZAMIR, M.; ABDELJAWAD, T.; NADEEM, F.; WAHID, A.; YOUSEF, A. An optimal control analysis of a covid-19 model. *Alexandria Engineering Journal*, Elsevier, v. 60, n. 3, p. 2875–2884, 2021.

## ANEXO A – CÓDIGO MATLAB PARA MÉTODO DE VARREDURA DO PROBLEMA DE QUARENTENA ÓTIMA

```

function y = control_u(T,l)
2  tic
   test = -1;
4  delta = 0.001;
   M = 100*T;
6  t = linspace(0,T,M +1) ;
   h = T/M;
8  h2 = h /2;
   B1 = 3000;
10 B3 = 2250;
   B2 = 2250;
12 S10 = 80e06 ; S20 = 100e06 ; S30 = 20e06 ;
   Q10 = 0; Q20 = 0; Q30 = 0;
14 E10 = 216; E20 = 9987; E30 = 10769;
   I10 = 789; I20 = 36478; I30 = 39335;
16 R10 = 729; R20 = 33415; R30 = 30979;
   %E10 = 2*267; E20 = 2*12349; E30 = 2*13318;
18 %I10 = 2*869; I20 = 2*40191; I30 = 2*43339;
   %R10 = 2*1638; R20 = 2*75055; R30 = 2*69583;
20 %E10 = 4*267; E20 = 4*12349; E30 = 4*13318;
   %I10 = 4*869; I20 = 4*40191; I30 = 4*43339;
22 %R10 = 4*1638; R20 = 4*75055; R30 = 4*69583;
   beta11 = 1.761679788383897; beta12 = 0.364751290201338;
24 beta13 = 1.324679121381364; beta22 = 0.638017428324060;
   beta23 = 0.359585327158739; beta33 = 0.573473735873102;
26 sigma1 = 0.273002904657466; sigma2 = 0.582321553388525;
   sigma3 = 0.693390800078658; gamma1 = 0.068624674761811;
28 gamma2 = 0.033170255758324; gamma3 = 0.355770925567391;
   beta21 = beta12; beta31 = beta13; beta32 = beta23;
30 tau = 1/13;
   S1 = zeros(1,M +1); S2 = zeros(1,M +1); S3 = zeros(1,M+1);
32 E1 = zeros(1,M +1); E2 = zeros(1,M +1); E3 = zeros(1,M+1);
   I1 = zeros(1,M +1); I2 = zeros(1,M +1); I3 = zeros(1,M+1);
34 R1 = zeros(1,M +1); R2 = zeros(1,M +1); R3 = zeros(1,M+1);
   Q1 = zeros(1,M +1); Q2 = zeros(1,M +1); Q3 = zeros(1,M+1);
36 N1 = zeros(1,M +1); N2 = zeros(1,M +1); N3 = zeros(1,M+1);
   S1(1) = S10; S2(1) = S20; S3(1) = S30;
38 E1(1) = E10; E2(1) = E20; E3(1) = E30;
   I1(1) = I10; I2(1) = I20; I3(1) = I30;
40 R1(1) = R10; R2(1) = R20; R3(1) = R30;
   Q1(1) = Q10; Q2(1) = Q20; Q3(1) = Q30;
42 N1(1) = S10 + E10 + I10 + R10 + Q10;
   N2(1) = S20 + E20 + I20 + R20 + Q20;

```

```

44 N3(1) = S30 + E30 + I30 + R30 + Q30 ;
   lambdaS1 = zeros(1,M+1);
46 lambdaS2 = zeros(1,M+1);
   lambdaS3 = zeros(1,M+1);
48 lambdaE1 = zeros(1,M+1);
   lambdaE2 = zeros(1,M+1);
50 lambdaE3 = zeros(1,M+1);
   lambdaI1 = zeros(1,M+1);
52 lambdaI2 = zeros(1,M+1);
   lambdaI3 = zeros(1,M+1);
54 lambdaQ1 = zeros(1,M+1);
   lambdaQ2 = zeros(1,M+1);
56 lambdaQ3 = zeros(1,M+1);
   lambdaN1 = zeros(1,M+1);
58 lambdaN2 = zeros(1,M+1);
   lambdaN3 = zeros(1,M+1);
60 u1 = zeros(1,M+1); u2 = zeros(1,M+1); u3 = zeros(1,M+1);
   while(test < 0)
62     oldu1 = u1; oldu2 = u2; oldu3 = u3;
       oldS1 = S1; oldS2 = S2; oldS3 = S3;
64     oldE1 = E1; oldE2 = E2; oldE3 = E3;
       oldI1 = I1; oldI2 = I2; oldI3 = I3;
66     oldQ1 = Q1; oldQ2 = Q2; oldQ3 = Q3;
       oldN1 = N1; oldN2 = N2; oldN3 = N3;
68     oldlambdaS1 = lambdaS1;
       oldlambdaS2 = lambdaS2;
70     oldlambdaS3 = lambdaS3;
       oldlambdaE1 = lambdaE1;
72     oldlambdaE2 = lambdaE2;
       oldlambdaE3 = lambdaE3;
74     oldlambdaI1 = lambdaI1;
       oldlambdaI2 = lambdaI2;
76     oldlambdaI3 = lambdaI3;
       oldlambdaQ1 = lambdaQ1;
78     oldlambdaQ2 = lambdaQ2;
       oldlambdaQ3 = lambdaQ3;
80     oldlambdaN1 = lambdaN1;
       oldlambdaN2 = lambdaN2;
82     oldlambdaN3 = lambdaN3;
       %Runge-Kutta para variaveis de estado
84     for i = 1:M
           m1S1 = - (S1(i)/(N1(i)+N2(i)+N3(i)))*(beta11*I1(i)+ beta12*I2(i)+ beta13*
               I3(i));
86           m1S2 = - (S2(i)/(N1(i)+N2(i)+N3(i)))*(beta21*I1(i)+ beta22*I2(i)+ beta23*
               I3(i));
           m1S3 = - (S3(i)/(N1(i)+N2(i)+N3(i)))*(beta31*I1(i)+ beta32*I2(i)+ beta33*
               I3(i));

```

```

88     m1E1 = (S1(i)/(N1(i)+N2(i)+N3(i)))*(beta11 *I1(i)+ beta12*I2(i)+ beta13*
        I3(i)) - sigma1*E1(i);
        m1E2 = (S2(i)/(N1(i)+N2(i)+N3(i)))*(beta21 *I1(i)+ beta22*I2(i)+ beta23*
        I3(i)) - sigma2*E2(i);
90     m1E3 = (S3(i)/(N1(i)+N2(i)+N3(i)))*(beta31 *I1(i)+ beta32*I2(i)+ beta33*
        I3(i)) - sigma3*E3(i);
        m1I1 = sigma1*E1(i) - (u1(i)+gamma1)*I1(i);
92     m1I2 = sigma2*E2(i) - (u2(i)+gamma2)*I2(i);
        m1I3 = sigma3*E3(i) - (u3(i)+gamma3)*I3(i);
94     m1Q1 = u1(i)*I1(i) - tau*Q1(i);
        m1Q2 = u2(i)*I2(i) - tau*Q2(i);
96     m1Q3 = u3(i)*I3(i) - tau*Q3(i);
        m1N1 = 0;
98     m1N2 = 0;
        m1N3 = 0;
100    %=====
        m2S1 = - ((S1(i)+h2*m1S1)/((N1(i)+h2*m1N1)+(N2(i)+h2*m1N2)+(N3(i)+h2*m1N3
        ))*(beta11*(I1(i)+h2*m1I1)+beta12*(I2(i)+h2*m1I2)+beta13*(I3(i)+h2*
        m1I3)));
102    m2S2 = - ((S2(i)+h2*m1S2)/((N1(i)+h2*m1N1)+(N2(i)+h2*m1N2)+(N3(i)+h2*m1N3
        ))*(beta21*(I1(i)+h2*m1I1)+beta22*(I2(i)+h2*m1I2)+beta23*(I3(i)+h2*
        m1I3)));
        m2S3 = - ((S3(i)+h2*m1S3)/((N1(i)+h2*m1N1)+(N2(i)+h2*m1N2)+(N3(i)+h2*m1N3
        ))*(beta31*(I1(i)+h2*m1I1)+beta32*(I2(i)+h2*m1I2)+beta33*(I3(i)+h2*
        m1I3)));
104    m2E1 = ((S1(i)+h2*m1S1)/((N1(i)+h2*m1N1)+(N2(i)+h2*m1N2)+(N3(i)+h2*m1N3))
        )*(beta11*(I1(i)+h2*m1I1)+beta12*(I2(i)+h2*m1I2)+beta13*(I3(i)+h2*
        m1I3)) - sigma1*(E1(i)+h2*m1E1);
        m2E2 = ((S2(i)+h2*m1S2)/((N1(i)+h2*m1N1)+(N2(i)+h2*m1N2)+(N3(i)+h2*m1N3))
        )*(beta21*(I1(i)+h2*m1I1)+beta22*(I2(i)+h2*m1I2)+beta23*(I3(i)+h2*
        m1I3)) - sigma1*(E2(i)+h2*m1E2);
106    m2E3 = ((S3(i)+h2*m1S3)/((N1(i)+h2*m1N1)+(N2(i)+h2*m1N2)+(N3(i)+h2*m1N3))
        )*(beta31*(I1(i)+h2*m1I1)+beta32*(I2(i)+h2*m1I2)+beta33*(I3(i)+h2*
        m1I3)) - sigma1*(E3(i)+h2*m1E3);
        m2I1 = sigma1*(E1(i)+h2*m1E1) - (0.5*(u1(i)+u1(i+1))+gamma1)*(I1(i)+h2*
        m1I1);
108    m2I2 = sigma2*(E2(i)+h2*m1E2) - (0.5*(u2(i)+u2(i+1))+gamma2)*(I2(i)+h2*
        m1I2);
        m2I3 = sigma3*(E3(i)+h2*m1E3) - (0.5*(u3(i)+u3(i+1))+gamma3)*(I3(i)+h2*
        m1I3);
110    m2Q1 = 0.5*(u1(i)+u1(i+1))*(I1(i)+h2*m1I1) - tau*(Q1(i)+h2*m1Q1);
        m2Q2 = 0.5*(u2(i)+u2(i+1))*(I2(i)+h2*m1I2) - tau*(Q2(i)+h2*m1Q2);
112    m2Q3 = 0.5*(u3(i)+u3(i+1))*(I3(i)+h2*m1I3) - tau*(Q3(i)+h2*m1Q3);
        m2N1 = 0;
114    m2N2 = 0;
        m2N3 = 0;
116    %=====

```

```

m3S1 = - ((S1(i)+h2*m2S1)/((N1(i)+h2*m2N1)+(N2(i)+h2*m2N2)+(N3(i)+h2*m2N3
)))*(beta11*(I1(i)+h2*m2I1)+beta12*(I2(i)+h2*m2I2)+beta13*(I3(i)+h2*
m2I3));
118 m3S2 = - ((S2(i)+h2*m2S2)/((N1(i)+h2*m2N1)+(N2(i)+h2*m2N2)+(N3(i)+h2*m2N3
)))*(beta21*(I1(i)+h2*m2I1)+beta22*(I2(i)+h2*m2I2)+beta23*(I3(i)+h2*
m2I3));
m3S3 = - ((S3(i)+h2*m2S3)/((N1(i)+h2*m2N1)+(N2(i)+h2*m2N2)+(N3(i)+h2*m2N3
)))*(beta31*(I1(i)+h2*m2I1)+beta32*(I2(i)+h2*m2I2)+beta33*(I3(i)+h2*
m2I3));
120 m3E1 = ((S1(i)+h2*m2S1)/((N1(i)+h2*m2N1)+(N2(i)+h2*m2N2)+(N3(i)+h2*m2N3))
)*(beta11*(I1(i)+h2*m2I1)+beta12*(I2(i)+h2*m2I2)+beta13*(I3(i)+h2*
m2I3)) - sigma1*(E1(i)+h2*m2E1);
m3E2 = ((S2(i)+h2*m2S2)/((N1(i)+h2*m2N1)+(N2(i)+h2*m2N2)+(N3(i)+h2*m2N3))
)*(beta21*(I1(i)+h2*m2I1)+beta22*(I2(i)+h2*m2I2)+beta23*(I3(i)+h2*
m2I3)) - sigma1*(E2(i)+h2*m2E2);
122 m3E3 = ((S3(i)+h2*m2S3)/((N1(i)+h2*m2N1)+(N2(i)+h2*m2N2)+(N3(i)+h2*m2N3))
)*(beta31*(I1(i)+h2*m2I1)+beta32*(I2(i)+h2*m2I2)+beta33*(I3(i)+h2*
m2I3)) - sigma1*(E3(i)+h2*m2E3);
m3I1 = sigma1*(E1(i)+h2*m2E1) - (0.5*(u1(i)+u1(i+1))+gamma1)*(I1(i)+h2*
m2I1);
124 m3I2 = sigma2*(E2(i)+h2*m2E2) - (0.5*(u2(i)+u2(i+1))+gamma2)*(I2(i)+h2*
m2I2);
m3I3 = sigma3*(E3(i)+h2*m2E3) - (0.5*(u3(i)+u3(i+1))+gamma3)*(I3(i)+h2*
m2I3);
126 m3Q1 = 0.5*(u1(i)+u1(i+1))*(I1(i)+h2*m2I1) - tau*(Q1(i)+h2*m2Q1);
m3Q2 = 0.5*(u2(i)+u2(i+1))*(I2(i)+h2*m2I2) - tau*(Q2(i)+h2*m2Q2);
128 m3Q3 = 0.5*(u3(i)+u3(i+1))*(I3(i)+h2*m2I3) - tau*(Q3(i)+h2*m2Q3);
m3N1 = 0;
130 m3N2 = 0;
m3N3 = 0;
132 %=====
m4S1 = - ((S1(i)+h*m3S1)/((N1(i)+h*m3N1)+(N2(i)+h*m3N2)+(N3(i)+h*m3N3)))
*(beta11*(I1(i)+h*m3I1)+beta12*(I2(i)+h*m3I2)+beta13*(I3(i)+h*m3I3));
134 m4S2 = - ((S2(i)+h*m3S2)/((N1(i)+h*m3N1)+(N2(i)+h*m3N2)+(N3(i)+h*m3N3)))
*(beta21*(I1(i)+h*m3I1)+beta22*(I2(i)+h*m3I2)+beta23*(I3(i)+h*m3I3));
m4S3 = - ((S3(i)+h*m3S3)/((N1(i)+h*m3N1)+(N2(i)+h*m3N2)+(N3(i)+h*m3N3)))
*(beta31*(I1(i)+h*m3I1)+beta32*(I2(i)+h*m3I2)+beta33*(I3(i)+h*m3I3));
136 m4E1 = ((S1(i)+h*m3S1)/((N1(i)+h*m3N1)+(N2(i)+h*m3N2)+(N3(i)+h*m3N3)))*
(beta11*(I1(i)+h*m3I1)+beta12*(I2(i)+h*m3I2)+beta13*(I3(i)+h*m3I3)) -
sigma1*(E1(i)+h*m3E1);
m4E2 = ((S2(i)+h*m3S2)/((N1(i)+h*m3N1)+(N2(i)+h*m3N2)+(N3(i)+h*m3N3)))*
(beta21*(I1(i)+h*m3I1)+beta22*(I2(i)+h*m3I2)+beta23*(I3(i)+h*m3I3)) -
sigma1*(E2(i)+h*m3E2);
138 m4E3 = ((S3(i)+h*m3S3)/((N1(i)+h*m3N1)+(N2(i)+h*m3N2)+(N3(i)+h*m3N3)))*
(beta31*(I1(i)+h*m3I1)+beta32*(I2(i)+h*m3I2)+beta33*(I3(i)+h*m3I3)) -
sigma1*(E3(i)+h*m3E3);
m4I1 = sigma1*(E1(i)+h*m3E1) - (u1(i+1)+gamma1)*(I1(i)+h*m3I1);

```

```

140     m4I2 = sigma2*(E2(i)+h*m3E2) - (u2(i+1)+gamma2)*(I2(i)+h*m3I2);
        m4I3 = sigma3*(E3(i)+h*m3E3) - (u3(i+1)+gamma3)*(I3(i)+h*m3I3);
142     m4Q1 = u1(i+1)*(I1(i)+h*m3I1) - tau*(Q1(i)+h*m3Q1);
        m4Q2 = u2(i+1)*(I2(i)+h*m3I2) - tau*(Q2(i)+h*m3Q2);
144     m4Q3 = u3(i+1)*(I3(i)+h*m3I3) - tau*(Q3(i)+h*m3Q3);
        m4N1 = 0;
146     m4N2 = 0;
        m4N3 = 0;
148     %====
        S1(i+1) = S1(i) + (h/6)*(m1S1 + 2*m2S1 + 2*m3S1 + m4S1);
150     S2(i+1) = S2(i) + (h/6)*(m1S2 + 2*m2S2 + 2*m3S2 + m4S2);
        S3(i+1) = S3(i) + (h/6)*(m1S3 + 2*m2S3 + 2*m3S3 + m4S3);
152     E1(i+1) = E1(i) + (h/6)*(m1E1 + 2*m2E1 + 2*m3E1 + m4E1);
        E2(i+1) = E2(i) + (h/6)*(m1E2 + 2*m2E2 + 2*m3E2 + m4E2);
154     E3(i+1) = E3(i) + (h/6)*(m1E3 + 2*m2E3 + 2*m3E3 + m4E3);
        I1(i+1) = I1(i) + (h/6)*(m1I1 + 2*m2I1 + 2*m3I1 + m4I1);
156     I2(i+1) = I2(i) + (h/6)*(m1I2 + 2*m2I2 + 2*m3I2 + m4I2);
        I3(i+1) = I3(i) + (h/6)*(m1I3 + 2*m2I3 + 2*m3I3 + m4I3);
158     Q1(i+1) = Q1(i) + (h/6)*(m1Q1 + 2*m2Q1 + 2*m3Q1 + m4Q1);
        Q2(i+1) = Q2(i) + (h/6)*(m1Q2 + 2*m2Q2 + 2*m3Q2 + m4Q2);
160     Q3(i+1) = Q3(i) + (h/6)*(m1Q3 + 2*m2Q3 + 2*m3Q3 + m4Q3);
        N1(i+1) = N1(i) + (h/6)*(m1N1 + 2*m2N1 + 2*m3N1 + m4N1);
162     N2(i+1) = N2(i) + (h/6)*(m1N2 + 2*m2N2 + 2*m3N2 + m4N2);
        N3(i+1) = N3(i) + (h/6)*(m1N3 + 2*m2N3 + 2*m3N3 + m4N3);
164     end
        % Runge-Kutta para variaveis adjuntas
166     for i = 1:M
        j = M + 2 - i;
168     m1S1 = (beta11*I1(j)+beta12*I2(j)+beta13*I3(j))*(lambdaS1(j)-lambdaE1(j))
            /(N1(j)+N2(j)+N3(j));
        m1S2 = (beta21*I1(j)+beta22*I2(j)+beta23*I3(j))*(lambdaS2(j)-lambdaE2(j))
            /(N1(j)+N2(j)+N3(j));
170     m1S3 = (beta31*I1(j)+beta32*I2(j)+beta33*I3(j))*(lambdaS3(j)-lambdaE3(j))
            /(N1(j)+N2(j)+N3(j));
        m1E1 = sigma1*(lambdaE1(j)-lambdaI1(j));
172     m1E2 = sigma2*(lambdaE2(j)-lambdaI2(j));
        m1E3 = sigma3*(lambdaE3(j)-lambdaI3(j));
174     m1I1 = (beta11*S1(j)*(lambdaS1(j)-lambdaE1(j))+beta12*S2(j)*(lambdaS2(j)-
            lambdaE2(j))+beta13*S3(j)*(lambdaS3(j)-lambdaE3(j)))/(N1(j)+N2(j)+N3(
            j)) - 1 + gamma1*lambdaI1(j)+ u1(j)*(lambdaI1(j)-lambdaQ1(j));
        m1I2 = (beta21*S1(j)*(lambdaS1(j)-lambdaE1(j))+beta22*S2(j)*(lambdaS2(j)-
            lambdaE2(j))+beta23*S3(j)*(lambdaS3(j)-lambdaE3(j)))/(N1(j)+N2(j)+N3(
            j)) - 1 + gamma2*lambdaI2(j)+ u2(j)*(lambdaI2(j)-lambdaQ2(j));
176     m1I3 = (beta31*S1(j)*(lambdaS1(j)-lambdaE1(j))+beta32*S2(j)*(lambdaS2(j)-
            lambdaE2(j))+beta33*S3(j)*(lambdaS3(j)-lambdaE3(j)))/(N1(j)+N2(j)+N3(
            j)) - 1 + gamma3*lambdaI3(j)+ u3(j)*(lambdaI3(j)-lambdaQ3(j));
        m1Q1 = tau*lambdaQ1(j);

```

```

178     m1Q2 = tau*lambdaQ2(j);
        m1Q3 = tau*lambdaQ3(j);
180     m1N1 = (I1(j)*(beta11*S1(j)*(lambdaE1(j)-lambdaS1(j))+beta21*S2(j)*(
            lambdaE2(j)-lambdaS2(j))+beta31*S3(j)*(lambdaE3(j)-lambdaS3(j)))) + I2
            (j)*(beta12*S1(j)*(lambdaE1(j)-lambdaS1(j))+beta22*S2(j)*(lambdaE2(j)
            -lambdaS2(j))+beta32*S3(j)*(lambdaE3(j)-lambdaS3(j)))+ I3(j)*(beta13*
            S1(j)*(lambdaE1(j)-lambdaS1(j))+beta23*S2(j)*(lambdaE2(j)-lambdaS2(j)
            )+ beta33*S3(j)*(lambdaE3(j)-lambdaS3(j)))))/(N1(j)+N2(j)+N3(j))^2);
        m1N2 = m1N1;
182     m1N3 = m1N1;
        %=====
184     m2S1 = (beta11*0.5*(I1(j)+I1(j-1))+beta12*0.5*(I2(j)+I2(j-1))+beta13
            *0.5*(I3(j)+I3(j-1)))*((lambdaS1(j)-h2*m1S1)-(lambdaE1(j)-h2*m1E1))
            /(0.5*(N1(j)+N1(j-1))+0.5*(N2(j)+N2(j-1))+0.5*(N3(j)+N3(j-1)));
        m2S2 = (beta21*0.5*(I1(j)+I1(j-1))+beta22*0.5*(I2(j)+I2(j-1))+beta23
            *0.5*(I3(j)+I3(j-1)))*((lambdaS2(j)-h2*m1S2)-(lambdaE2(j)-h2*m1E2))
            /(0.5*(N1(j)+N1(j-1))+0.5*(N2(j)+N2(j-1))+0.5*(N3(j)+N3(j-1)));
186     m2S3 = (beta31*0.5*(I1(j)+I1(j-1))+beta32*0.5*(I2(j)+I2(j-1))+beta33
            *0.5*(I3(j)+I3(j-1)))*((lambdaS3(j)-h2*m1S3)-(lambdaE3(j)-h2*m1E3))
            /(0.5*(N1(j)+N1(j-1))+0.5*(N2(j)+N2(j-1))+0.5*(N3(j)+N3(j-1)));
        m2E1 = sigma1*((lambdaE1(j)-h2*m1E1)-(lambdaI1(j)-h2*m1I1));
188     m2E2 = sigma2*((lambdaE2(j)-h2*m1E2)-(lambdaI2(j)-h2*m1I2));
        m2E3 = sigma3*((lambdaE3(j)-h2*m1E3)-(lambdaI3(j)-h2*m1I3));
190     m2I1 = (beta11*0.5*(S1(j)+S1(j-1))*((lambdaS1(j)-h2*m1S1)-(lambdaE1(j)-h2
            *m1E1))+beta12*0.5*(S2(j)+S2(j-1))*((lambdaS2(j)-h2*m1S2)-(lambdaE2(j)
            )-h2*m1E2))+beta13*0.5*(S3(j)+S3(j-1))*((lambdaS3(j)-h2*m1S3)-(
            lambdaE3(j)-h2*m1E3)))/(0.5*(N1(j)+N1(j-1))+0.5*(N2(j)+N2(j-1))+0.5*(
            N3(j)+N3(j-1))) - 1 + gamma1*(lambdaI1(j)-h2*m1I1)+ 0.5*(u1(j)+u1(j
            -1))*((lambdaI1(j)-h2*m1I1)-(lambdaQ1(j)-h2*m1Q1));
        m2I2 = (beta21*0.5*(S1(j)+S1(j-1))*((lambdaS1(j)-h2*m1S1)-(lambdaE1(j)-h2
            *m1E1))+beta22*0.5*(S2(j)+S2(j-1))*((lambdaS2(j)-h2*m1S2)-(lambdaE2(j)
            )-h2*m1E2))+beta23*0.5*(S3(j)+S3(j-1))*((lambdaS3(j)-h2*m1S3)-(
            lambdaE3(j)-h2*m1E3)))/(0.5*(N1(j)+N1(j-1))+0.5*(N2(j)+N2(j-1))+0.5*(
            N3(j)+N3(j-1))) - 1 + gamma2*(lambdaI2(j)-h2*m1I1)+ 0.5*(u2(j)+u2(j
            -1))*((lambdaI2(j)-h2*m1I2)-(lambdaQ2(j)-h2*m1Q2));
192     m2I3 = (beta31*0.5*(S1(j)+S1(j-1))*((lambdaS1(j)-h2*m1S1)-(lambdaE1(j)-h2
            *m1E1))+beta32*0.5*(S2(j)+S2(j-1))*((lambdaS2(j)-h2*m1S2)-(lambdaE2(j)
            )-h2*m1E2))+beta33*0.5*(S3(j)+S3(j-1))*((lambdaS3(j)-h2*m1S3)-(
            lambdaE3(j)-h2*m1E3)))/(0.5*(N1(j)+N1(j-1))+0.5*(N2(j)+N2(j-1))+0.5*(
            N3(j)+N3(j-1))) - 1 + gamma3*(lambdaI3(j)-h2*m1I3)+ 0.5*(u3(j)+u3(j
            -1))*((lambdaI3(j)-h2*m1I3)-(lambdaQ3(j)-h2*m1Q3));
        m2Q1 = tau*(lambdaQ1(j)-h2*m1Q1);
194     m2Q2 = tau*(lambdaQ2(j)-h2*m1Q2);
        m2Q3 = tau*(lambdaQ3(j)-h2*m1Q3);
196     m2N1 = (0.5*(I1(j)+I1(j-1))*(beta11*0.5*(S1(j)+S1(j-1))*((lambdaE1(j)-h2*
            m1E1)-(lambdaS1(j)-h2*m1S1))+beta21*0.5*(S2(j)+S2(j-1))*((lambdaE2(j)
            -h2*m1E2)-(lambdaS2(j)-h2*m1S2))+beta31*0.5*(S3(j)+S3(j-1))*((

```

```

lambdaE3(j)-h2*m1E3)-(lambdaS3(j)-h2*m1S3)))+ 0.5*(I2(j)+I2(j-1))*(
beta12*0.5*(S1(j)+S1(j-1))*((lambdaE1(j)-h2*m1E1)-(lambdaS1(j)-h2*
m1S1))+beta22*0.5*(S2(j)+S2(j-1))*((lambdaE2(j)-h2*m1E2)-(lambdaS2(j)
-h2*m1S2))+beta32*0.5*(S3(j)+S3(j-1))*((lambdaE3(j)-h2*m1E3)-(
lambdaS3(j)-h2*m1S3)))+ 0.5*(I3(j)+I3(j-1))*(beta13*0.5*(S1(j)+S1(j
-1))*((lambdaE1(j)-h2*m1E1)-(lambdaS1(j)-h2*m1S1))+beta23*0.5*(S2(j)+
S2(j-1))*((lambdaE2(j)-h2*m1E2)-(lambdaS2(j)-h2*m1S2))+ beta33*0.5*(
S3(j)+S3(j-1))*((lambdaE3(j)-h2*m1E3)-(lambdaS3(j)-h2*m1S3)))/(0.5*(
N1(j)+N1(j-1))+0.5*(N2(j)+N2(j-1))+0.5*(N3(j)+N3(j-1)))^2);
m2N2 = m2N1;
198 m2N3 = m2N1;
%====
200 m3S1 = (beta11*0.5*(I1(j)+I1(j-1))+beta12*0.5*(I2(j)+I2(j-1))+beta13
*0.5*(I3(j)+I3(j-1)))*((lambdaS1(j)-h2*m2S1)-(lambdaE1(j)-h2*m2E1))
/(0.5*(N1(j)+N1(j-1))+0.5*(N2(j)+N2(j-1))+0.5*(N3(j)+N3(j-1)));
m3S2 = (beta21*0.5*(I1(j)+I1(j-1))+beta22*0.5*(I2(j)+I2(j-1))+beta23
*0.5*(I3(j)+I3(j-1)))*((lambdaS2(j)-h2*m2S2)-(lambdaE2(j)-h2*m2E2))
/(0.5*(N1(j)+N1(j-1))+0.5*(N2(j)+N2(j-1))+0.5*(N3(j)+N3(j-1)));
202 m3S3 = (beta31*0.5*(I1(j)+I1(j-1))+beta32*0.5*(I2(j)+I2(j-1))+beta33
*0.5*(I3(j)+I3(j-1)))*((lambdaS3(j)-h2*m2S3)-(lambdaE3(j)-h2*m2E3))
/(0.5*(N1(j)+N1(j-1))+0.5*(N2(j)+N2(j-1))+0.5*(N3(j)+N3(j-1)));
m3E1 = sigma1*((lambdaE1(j)-h2*m2E1)-(lambdaI1(j)-h2*m2I1));
204 m3E2 = sigma2*((lambdaE2(j)-h2*m2E2)-(lambdaI2(j)-h2*m2I2));
m3E3 = sigma3*((lambdaE3(j)-h2*m2E3)-(lambdaI3(j)-h2*m2I3));
206 m3I1 = (beta11*0.5*(S1(j)+S1(j-1))*((lambdaS1(j)-h2*m2S1)-(lambdaE1(j)-h2
*m2E1))+beta12*0.5*(S2(j)+S2(j-1))*((lambdaS2(j)-h2*m2S2)-(lambdaE2(j)
-h2*m2E2))+beta13*0.5*(S3(j)+S3(j-1))*((lambdaS3(j)-h2*m2S3)-(
lambdaE3(j)-h2*m2E3)))/(0.5*(N1(j)+N1(j-1))+0.5*(N2(j)+N2(j-1))+0.5*(
N3(j)+N3(j-1))) - 1 + gamma1*(lambdaI1(j)-h2*m2I1)+ 0.5*(u1(j)+u1(j
-1))*((lambdaI1(j)-h2*m2I1)-(lambdaQ1(j)-h2*m2Q1));
m3I2 = (beta21*0.5*(S1(j)+S1(j-1))*((lambdaS1(j)-h2*m2S1)-(lambdaE1(j)-h2
*m2E1))+beta22*0.5*(S2(j)+S2(j-1))*((lambdaS2(j)-h2*m2S2)-(lambdaE2(j)
-h2*m2E2))+beta23*0.5*(S3(j)+S3(j-1))*((lambdaS3(j)-h2*m2S3)-(
lambdaE3(j)-h2*m2E3)))/(0.5*(N1(j)+N1(j-1))+0.5*(N2(j)+N2(j-1))+0.5*(
N3(j)+N3(j-1))) - 1 + gamma2*(lambdaI2(j)-h2*m2I1)+ 0.5*(u2(j)+u2(j
-1))*((lambdaI2(j)-h2*m2I2)-(lambdaQ2(j)-h2*m2Q2));
208 m3I3 = (beta31*0.5*(S1(j)+S1(j-1))*((lambdaS1(j)-h2*m2S1)-(lambdaE1(j)-h2
*m2E1))+beta32*0.5*(S2(j)+S2(j-1))*((lambdaS2(j)-h2*m2S2)-(lambdaE2(j)
-h2*m2E2))+beta33*0.5*(S3(j)+S3(j-1))*((lambdaS3(j)-h2*m2S3)-(
lambdaE3(j)-h2*m2E3)))/(0.5*(N1(j)+N1(j-1))+0.5*(N2(j)+N2(j-1))+0.5*(
N3(j)+N3(j-1))) - 1 + gamma3*(lambdaI3(j)-h2*m2I3)+ 0.5*(u3(j)+u3(j
-1))*((lambdaI3(j)-h2*m2I3)-(lambdaQ3(j)-h2*m2Q3));
m3Q1 = tau*(lambdaQ1(j)-h2*m2Q1);
210 m3Q2 = tau*(lambdaQ2(j)-h2*m2Q2);
m3Q3 = tau*(lambdaQ3(j)-h2*m2Q3);
212 m3N1 = (0.5*(I1(j)+I1(j-1))*(beta11*0.5*(S1(j)+S1(j-1))*((lambdaE1(j)-h2*
m2E1)-(lambdaS1(j)-h2*m2S1))+beta21*0.5*(S2(j)+S2(j-1))*((lambdaE2(j)

```

```

-h2*m2E2)-(lambdaS2(j)-h2*m2S2))+beta31*0.5*(S3(j)+S3(j-1))*((
lambdaE3(j)-h2*m2E3)-(lambdaS3(j)-h2*m2S3)))+ 0.5*(I2(j)+I2(j-1))*(
beta12*0.5*(S1(j)+S1(j-1))*((lambdaE1(j)-h2*m2E1)-(lambdaS1(j)-h2*
m2S1))+beta22*0.5*(S2(j)+S2(j-1))*((lambdaE2(j)-h2*m2E2)-(lambdaS2(j)
-h2*m2S2))+beta32*0.5*(S3(j)+S3(j-1))*((lambdaE3(j)-h2*m2E3)-(
lambdaS3(j)-h2*m2S3)))+ 0.5*(I3(j)+I3(j-1))*(beta13*0.5*(S1(j)+S1(j
-1))*((lambdaE1(j)-h2*m2E1)-(lambdaS1(j)-h2*m2S1))+beta23*0.5*(S2(j)+
S2(j-1))*((lambdaE2(j)-h2*m2E2)-(lambdaS2(j)-h2*m2S2))+ beta33*0.5*(
S3(j)+S3(j-1))*((lambdaE3(j)-h2*m2E3)-(lambdaS3(j)-h2*m2S3)))
/((0.5*(N1(j)+N1(j-1))+0.5*(N2(j)+N2(j-1))+0.5*(N3(j)+N3(j-1)))^2);
m3N2 = m3N1;
214 m3N3 = m3N1;
%====
216 m4S1 = (beta11*I1(j-1)+beta12*I2(j-1)+beta13*I3(j-1))*((lambdaS1(j)-h*
m3S1)-(lambdaE1(j)-h*m3E1))/(N1(j-1)+N2(j-1)+N3(j-1));
m4S2 = (beta21*I1(j-1)+beta22*I2(j-1)+beta23*I3(j-1))*((lambdaS2(j)-h*
m3S2)-(lambdaE2(j)-h*m3E2))/(N1(j-1)+N2(j-1)+N3(j-1));
218 m4S3 = (beta31*I1(j-1)+beta32*I2(j-1)+beta33*I3(j-1))*((lambdaS3(j)-h*
m3S3)-(lambdaE3(j)-h*m3E3))/(N1(j-1)+N2(j-1)+N3(j-1));
m4E1 = sigma1*((lambdaE1(j)-h*m3E1)-(lambdaI1(j)-h*m3I1));
220 m4E2 = sigma2*((lambdaE2(j)-h*m3E2)-(lambdaI2(j)-h*m3I2));
m4E3 = sigma3*((lambdaE3(j)-h*m3E3)-(lambdaI3(j)-h*m3I3));
222 m4I1 = (beta11*S1(j-1))*((lambdaS1(j)-h*m3S1)-(lambdaE1(j)-h*m3E1))+beta12
*S2(j-1))*((lambdaS2(j)-h*m3S2)-(lambdaE2(j)-h*m3E2))+beta13*S3(j-1)
*((lambdaS3(j)-h*m3S3)-(lambdaE3(j)-h*m3E3))/(N1(j-1)+N2(j-1)+N3(j
-1)) - 1 + gamma1*(lambdaI1(j)-h*m3I1)+ u1(j-1))*((lambdaI1(j)-h*m3I1)
-(lambdaQ1(j)-h*m3Q1));
m4I2 = (beta21*S1(j-1))*((lambdaS1(j)-h*m3S1)-(lambdaE1(j)-h*m3E1))+beta22
*S2(j-1))*((lambdaS2(j)-h*m3S2)-(lambdaE2(j)-h*m3E2))+beta23*S3(j-1)
*((lambdaS3(j)-h*m3S3)-(lambdaE3(j)-h*m3E3))/(N1(j-1)+N2(j-1)+N3(j
-1)) - 1 + gamma2*(lambdaI2(j)-h*m3I2)+ u2(j-1))*((lambdaI2(j)-h*m3I2)
-(lambdaQ2(j)-h*m3Q2));
224 m4I3 = (beta31*S1(j-1))*((lambdaS1(j)-h*m3S1)-(lambdaE1(j)-h*m3E1))+beta32
*S2(j-1))*((lambdaS2(j)-h*m3S2)-(lambdaE2(j)-h*m3E2))+beta33*S3(j-1)
*((lambdaS3(j)-h*m3S3)-(lambdaE3(j)-h*m3E3))/(N1(j-1)+N2(j-1)+N3(j
-1)) - 1 + gamma3*(lambdaI3(j)-h*m3I3)+ u3(j-1))*((lambdaI3(j)-h*m3I3)
-(lambdaQ3(j)-h*m3Q3));
m4Q1 = tau*(lambdaQ1(j)-h*m3Q1);
226 m4Q2 = tau*(lambdaQ2(j)-h*m3Q2);
m4Q3 = tau*(lambdaQ3(j)-h*m3Q3);
228 m4N1 = (I1(j-1))*(beta11*S1(j-1))*((lambdaE1(j)-h*m3E1)-(lambdaS1(j)-h*m3S1
))+beta21*S2(j-1))*((lambdaE2(j)-h*m3E2)-(lambdaS2(j)-h*m3S2))+beta31*
S3(j-1))*((lambdaE3(j)-h*m3E3)-(lambdaS3(j)-h*m3S3))+ I2(j-1)*(beta12
*S1(j-1))*((lambdaE1(j)-h*m3E1)-(lambdaS1(j)-h*m3S1))+beta22*S2(j-1)
*((lambdaE2(j)-h*m3E2)-(lambdaS2(j)-h*m3S2))+beta32*S3(j-1))*((
lambdaE3(j)-h*m3E3)-(lambdaS3(j)-h*m3S3))+ I3(j-1)*(beta13*S1(j-1)
*((lambdaE1(j)-h*m3E1)-(lambdaS1(j)-h*m3S1))+beta23*S2(j-1))*((

```

```

        lambdaE2(j)-h*m3E2)-(lambdaS2(j)-h*m3S2))+ beta33*S3(j-1)*((lambdaE3(
        j)-h*m3E3)-(lambdaS3(j)-h*m3S3)))/((N1(j-1)+N2(j-1)+N3(j-1))^2);
230     m4N2 = m4N1;
        m4N3 = m4N1;
        %=====
232     lambdaS1(j-1) = lambdaS1(j) - (h/6)*(m1S1 + 2*m2S1 + 2*m3S1 + m4S1);
        lambdaS2(j-1) = lambdaS2(j) - (h/6)*(m1S2 + 2*m2S2 + 2*m3S2 + m4S2);
234     lambdaS3(j-1) = lambdaS3(j) - (h/6)*(m1S3 + 2*m2S3 + 2*m3S3 + m4S3);
        lambdaE1(j-1) = lambdaE1(j) - (h/6)*(m1E1 + 2*m2E1 + 2*m3E1 + m4E1);
236     lambdaE2(j-1) = lambdaE2(j) - (h/6)*(m1E2 + 2*m2E2 + 2*m3E2 + m4E2);
        lambdaE3(j-1) = lambdaE3(j) - (h/6)*(m1E3 + 2*m2E3 + 2*m3E3 + m4E3);
238     lambdaI1(j-1) = lambdaI1(j) - (h/6)*(m1I1 + 2*m2I1 + 2*m3I1 + m4I1);
        lambdaI2(j-1) = lambdaI2(j) - (h/6)*(m1I2 + 2*m2I2 + 2*m3I2 + m4I2);
240     lambdaI3(j-1) = lambdaI3(j) - (h/6)*(m1I3 + 2*m2I3 + 2*m3I3 + m4I3);
        lambdaQ1(j-1) = lambdaQ1(j) - (h/6)*(m1Q1 + 2*m2Q1 + 2*m3Q1 + m4Q1);
242     lambdaQ2(j-1) = lambdaQ2(j) - (h/6)*(m1Q2 + 2*m2Q2 + 2*m3Q2 + m4Q2);
        lambdaQ3(j-1) = lambdaQ3(j) - (h/6)*(m1Q3 + 2*m2Q3 + 2*m3Q3 + m4Q3);
244     lambdaN1(j-1) = lambdaN1(j) - (h/6)*(m1N1 + 2*m2N1 + 2*m3N1 + m4N1);
        lambdaN2(j-1) = lambdaN2(j) - (h/6)*(m1N2 + 2*m2N2 + 2*m3N2 + m4N2);
246     lambdaN3(j-1) = lambdaN3(j) - (h/6)*(m1N3 + 2*m2N3 + 2*m3N3 + m4N3);

    end

248     tempu1 =(I1.*(lambdaI1 - lambdaQ1))./(2*B1);
        tempu2 =(I2.*(lambdaI2 - lambdaQ2))./(2*B2);
250     tempu3 =(I3.*(lambdaI3 - lambdaQ3))./(2*B3);
        u11 = min(1, max(0, tempu1));
252     u21 = min(1, max(0, tempu2));
        u31 = min(1, max(0, tempu3));
254     u1 = 0.5*(u11 + oldu1);
        u2 = 0.5*(u21 + oldu2);
256     u3 = 0.5*(u31 + oldu3);

        % Teste para criterio de parada
258     temp1 = delta*sum(abs(u1)) - sum(abs(oldu1 - u1));
        temp2 = delta*sum(abs(u2)) - sum(abs(oldu2 - u2));
260     temp3 = delta*sum(abs(u3)) - sum(abs(oldu3 - u3));
        temp4 = delta*sum(abs(S1)) - sum(abs(oldS1 - S1));
262     temp5 = delta*sum(abs(S2)) - sum(abs(oldS2 - S2));
        temp6 = delta*sum(abs(S3)) - sum(abs(oldS3 - S3));
264     temp7 = delta*sum(abs(E1)) - sum(abs(oldE1 - E1));
        temp8 = delta*sum(abs(E2)) - sum(abs(oldE2 - E2));
266     temp9 = delta*sum(abs(E3)) - sum(abs(oldE3 - E3));
        temp10 = delta*sum(abs(I1)) - sum(abs(oldI1 - I1));
268     temp11 = delta*sum(abs(I2)) - sum(abs(oldI2 - I2));
        temp12 = delta*sum(abs(I3)) - sum(abs(oldI3 - I3));
270     temp13 = delta*sum(abs(Q1)) - sum(abs(oldQ1 - Q1));
        temp14 = delta*sum(abs(Q2)) - sum(abs(oldQ2 - Q2));
272     temp15 = delta*sum(abs(Q3)) - sum(abs(oldQ3 - Q3));
        temp16 = delta*sum(abs(N1)) - sum(abs(oldN1 - N1));

```

```

274     temp17 = delta*sum(abs(N2)) - sum(abs(oldN2 - N2));
        temp18 = delta*sum(abs(N3)) - sum(abs(oldN3 - N3));
276     temp19 = delta*sum(abs(lambdaS1)) - sum(abs(oldlambdaS1 - lambdaS1));
        temp20 = delta*sum(abs(lambdaS2)) - sum(abs(oldlambdaS2 - lambdaS2));
278     temp21 = delta*sum(abs(lambdaS3)) - sum(abs(oldlambdaS3 - lambdaS3));
        temp22 = delta*sum(abs(lambdaE1)) - sum(abs(oldlambdaE1 - lambdaE1));
280     temp23 = delta*sum(abs(lambdaE2)) - sum(abs(oldlambdaE2 - lambdaE2));
        temp24 = delta*sum(abs(lambdaE3)) - sum(abs(oldlambdaE3 - lambdaE3));
282     temp25 = delta*sum(abs(lambdaI1)) - sum(abs(oldlambdaI1 - lambdaI1));
        temp26 = delta*sum(abs(lambdaI2)) - sum(abs(oldlambdaI2 - lambdaI2));
284     temp27 = delta*sum(abs(lambdaI3)) - sum(abs(oldlambdaI3 - lambdaI3));
        temp28 = delta*sum(abs(lambdaQ1)) - sum(abs(oldlambdaQ1 - lambdaQ1));
286     temp29 = delta*sum(abs(lambdaQ2)) - sum(abs(oldlambdaQ2 - lambdaQ2));
        temp30 = delta*sum(abs(lambdaQ3)) - sum(abs(oldlambdaQ3 - lambdaQ3));
288     temp31 = delta*sum(abs(lambdaN1)) - sum(abs(oldlambdaN1 - lambdaN1));
        temp32 = delta*sum(abs(lambdaN2)) - sum(abs(oldlambdaN2 - lambdaN2));
290     temp33 = delta*sum(abs(lambdaN3)) - sum(abs(oldlambdaN3 - lambdaN3));
        test = min(temp1 ,min(temp2 ,min(temp3 , min ...
292         (temp4 ,min(temp5 ,min(temp6 ,min(temp7 , min ...
            (temp8 ,min(temp9 ,min(temp10 , min(temp11 , min ...
294         (temp12 , min(temp13 , min(temp14 , min(temp15 , min ...
            (temp16 , min(temp17 , min(temp18 , min(temp19 , min ...
296         (temp20 , min(temp21 , min(temp22 , min(temp23 , min ...
            (temp24 , min(temp25 , min(temp26 , min(temp27 , min ...
298         (temp28 , min(temp29 , min(temp30 , min(temp31 , min ...
            (temp32 , temp33 ))))))))))))))))))))))))))))))));
300 end
    for i=1:M
302         m1R1 = gamma1*I1(i) + tau*Q1(i);
            m2R1 = gamma1*0.5*(I1(i)+I1(i+1)) + tau*0.5*(Q1(i)+Q1(i+1));
304         m3R1 = gamma1*0.5*(I1(i)+I1(i+1)) + tau*0.5*(Q1(i)+Q1(i+1));
            m4R1 = gamma1*I1(i+1) + tau*Q1(i+1);
306         R1(i+1) = R1(i) + (h/6) *(m1R1 + 2*m2R1 + 2*m3R1 + m4R1);
            m1R2 = gamma1*I2(i) + tau*Q2(i);
308         m2R2 = gamma1*0.5*(I2(i)+I2(i+1)) + tau*0.5*(Q2(i)+Q2(i+1));
            m3R2 = gamma1*0.5*(I2(i)+I2(i+1)) + tau*0.5*(Q2(i)+Q2(i+1));
310         m4R2 = gamma1*I2(i+1) + tau*Q2(i+1);
            R2(i+1) = R2(i) + (h/6) *(m1R2 + 2*m2R2 + 2*m3R2 + m4R2);
312         m1R3 = gamma1*I3(i) + tau*Q3(i);
            m2R3 = gamma1*0.5*(I3(i)+I3(i+1)) + tau*0.5*(Q3(i)+Q3(i+1));
314         m3R3 = gamma1*0.5*(I3(i)+I3(i+1)) + tau*0.5*(Q3(i)+Q3(i+1));
            m4R3 = gamma1*I3(i+1) + tau*Q3(i+1);
316         R3(i+1) = R3(i) + (h/6) *(m1R3 + 2*m2R3 + 2*m3R3 + m4R3);
    end
318     y(1,:) = t;
        y(2,:) = S1;
320     y(3,:) = S2;

```

```
y(4,:) = S3;  
322 y(5,:) = E1;  
y(6,:) = E2;  
324 y(7,:) = E3;  
y(8,:) = I1;  
326 y(9,:) = I2;  
y(10,:) = I3;  
328 y(11,:) = R1;  
y(12,:) = R2;  
330 y(13,:) = R3;  
y(14,:) = Q1;  
332 y(15,:) = Q2;  
y(16,:) = Q3;  
334 y(17,:) = u1;  
y(18,:) = u2;  
336 y(19,:) = u3;  
toc
```

## ANEXO B – CÓDIGO MATLAB PARA MÉTODO DE VARREDURA DO PROBLEMA DE VACINAÇÃO ÓTIMA

```
1 function y = cont8(T)
   tic
3 test = -1;
   erro = 0.001;
5 M = 100*T;
   t=linspace(0,T,M+1);
7 h=T/M;
   h2 = h/2;
9 B1=7400;
   B2=6700;
11 beta = 0.45;
   sigma = 0.25;
13 gamma = 0.07;
   delta = 0.65;
15 theta1 = 0.74;
   theta2 = 0.67;
17 beta1 = 1-theta1;
   beta2 = 1-theta2;
19 alpha1 = 0.08;
   alpha2 = 0.064;
21 epsilon1 = 0.432;
   epsilon2 = 0.54;
23 S0 = 200000000;
   V10 = 0;
25 V20 = 0;
   E0 = 20972;
27 I0 = 76603;
   R0 = 65124;
29 S=zeros(1,M+1);
   V1=zeros(1,M+1);
31 V2=zeros(1,M+1);
   E=zeros(1,M+1);
33 I=zeros(1,M+1);
   R=zeros(1,M+1);
35 S(1)=S0;
   V1(1)=V10;
37 V2(1)=V20;
   E(1)=E0;
39 I(1)=I0;
   R(1)=R0;
41 lambdaS=zeros(1,M+1);
   lambdaV1=zeros(1,M+1);
43 lambdaV2=zeros(1,M+1);
```

```

lambdaE=zeros(1,M+1);
45 lambdaI=zeros(1,M+1);
lambdaR=zeros(1,M+1);
47 u1=zeros(1,M+1);
u2=zeros(1,M+1);
49 while(test < 0)
    oldu1 = u1;
51    oldu2 = u2;
    oldS = S;
53    oldV1 = V1;
    oldV2 = V2;
55    oldE = E;
    oldI = I;
57    oldR = R;
    oldlambdaS = lambdaS;
59    oldlambdaV1 = lambdaV1;
    oldlambdaV2 = lambdaV2;
61    oldlambdaE = lambdaE;
    oldlambdaI = lambdaI;
63    oldlambdaR = lambdaR;
    %Runge-Kutta para variaveis de estado
65    for i = 1:M
        m1S = - (beta*S(i)*I(i))/(S(i)+V1(i)+V2(i)+E(i)+I(i)+R(i)) - u1(i)*S(i) -
            u2(i)*S(i) + epsilon1*V1(i) + epsilon2*V2(i) + delta*R(i);
67        m1V1 = u1(i)*S(i) - (beta1*V1(i)*I(i))/(S(i)+V1(i)+V2(i)+E(i)+I(i)+R(i))
            - (epsilon1 + alpha1)*V1(i);
        m1V2 = u2(i)*S(i) - (beta2*V2(i)*I(i))/(S(i)+V1(i)+V2(i)+E(i)+I(i)+R(i))
            - (epsilon2 + alpha2)*V2(i);
69        m1E = (beta*S(i)*I(i))/(S(i)+V1(i)+V2(i)+E(i)+I(i)+R(i)) - sigma*E(i) + (
            beta1*V1(i)*I(i))/(S(i)+V1(i)+V2(i)+E(i)+I(i)+R(i)) + (beta2*V2(i)*I(
            i))/(S(i)+V1(i)+V2(i)+E(i)+I(i)+R(i));
        m1I = sigma*E(i) - gamma*I(i);
71        m1R = gamma*I(i) - delta*R(i) + alpha1*V1(i) + alpha2*V2(i);
        %====
73        m2S = - (beta*(S(i)+h2*m1S)*(I(i)+h2*m1I))/((S(i)+h2*m1S)+(V1(i)+h2*m1V1)
            +(V2(i)+h2*m1V2) +(E(i)+h2*m1E)+(I(i)+h2*m1I)+(R(i)+h2*m1R)) - 0.5*(
            u1(i)+u1(i+1))*(S(i)+h2*m1S) - 0.5*(u2(i)+u2(i+1))*(S(i)+h2*m1S) +
            epsilon1*(V1(i)+h2*m1V1) + epsilon2*(V2(i)+h2*m1V2) + delta*(R(i)+h2*
            m1R);
        m2V1 = 0.5*(u1(i)+u1(i+1))*(S(i)+h2*m1S) - (beta1*(V1(i)+h2*m1V1)*(I(i)+
            h2*m1I))/((S(i)+h2*m1S)+(V1(i)+h2*m1V1) +(V2(i)+h2*m1V2)+(E(i)+h2*m1E
            )+(I(i)+h2*m1I)+(R(i)+h2*m1R)) - (epsilon1 + alpha1)*(V1(i)+h2*m1V1);
75        m2V2 = 0.5*(u2(i)+u2(i+1))*(S(i)+h2*m1S) - (beta2*(V2(i)+h2*m1V2)*(I(i)+
            h2*m1I))/((S(i)+h2*m1S)+(V1(i)+h2*m1V1) +(V2(i)+h2*m1V2)+(E(i)+h2*m1E
            )+(I(i)+h2*m1I)+(R(i)+h2*m1R)) - (epsilon2 + alpha2)*(V2(i)+h2*m1V2);
        m2E = (beta*(S(i)+h2*m1S)*(I(i)+h2*m1I))/((S(i)+h2*m1S)+(V1(i)+h2*m1V1)+(
            V2(i)+h2*m1V2)+(E(i)+h2*m1E)+(I(i)+h2*m1I) +(R(i)+h2*m1R)) - sigma*(E

```

```

(i)+h2*m1E) + (beta1*(V1(i)+h2*m1V1)*(I(i)+h2*m1I))/((S(i)+h2*m1S)+(
V1(i)+h2*m1V1) +(V2(i)+h2*m1V2) +(E(i)+h2*m1E)+(I(i)+h2*m1I)+(R(i)+h2
*m1R)) +(beta2*(V2(i)+h2*m1V2)*(I(i)+h2*m1I))/((S(i)+h2*m1S)+(V1(i)+
h2*m1V1)+(V2(i)+h2*m1V2)+(E(i)+h2*m1E)+(I(i)+h2*m1I)+(R(i)+h2*m1R));
77 m2I = sigma*(E(i)+h2*m1E) - gamma*(I(i)+h2*m1I);
m2R = gamma*(I(i)+h2*m1I) - delta*(R(i)+h2*m1R) + alpha1*(V1(i)+h2*m1V1)
+ alpha2*(V2(i)+h2*m1V2);
79 %====
m3S = - (beta*(S(i)+h2*m2S)*(I(i)+h2*m2I))/((S(i)+h2*m2S)+(V1(i)+h2*m2V1)
+(V2(i)+h2*m2V2) +(E(i)+h2*m2E)+(I(i)+h2*m2I)+(R(i)+h2*m2R)) - 0.5*(
u1(i)+u1(i+1))*(S(i)+h2*m2S) - 0.5*(u2(i)+u2(i+1))*(S(i)+h2*m2S) +
epsilon1*(V1(i)+h2*m2V1) + epsilon2*(V2(i)+h2*m2V2) + delta*(R(i)+h2*
81 m3V1 = 0.5*(u1(i)+u1(i+1))*(S(i)+h2*m2S) - (beta1*(V1(i)+h2*m2V1)*(I(i)+
h2*m2I))/((S(i)+h2*m2S)+(V1(i)+h2*m2V1) +(V2(i)+h2*m2V2)+(E(i)+h2*m2E
)+(I(i)+h2*m2I)+(R(i)+h2*m2R)) - (epsilon1 + alpha1)*(V1(i)+h2*m2V1);
m3V2 = 0.5*(u2(i)+u2(i+1))*(S(i)+h2*m2S) - (beta2*(V2(i)+h2*m2V2)*(I(i)+
h2*m2I))/((S(i)+h2*m2S)+(V1(i)+h2*m2V1) +(V2(i)+h2*m2V2)+(E(i)+h2*m2E
)+(I(i)+h2*m2I)+(R(i)+h2*m2R)) - (epsilon2 + alpha2)*(V2(i)+h2*m2V2);
83 m3E = (beta*(S(i)+h2*m2S)*(I(i)+h2*m2I))/((S(i)+h2*m2S)+(V1(i)+h2*m2V1)+(
V2(i)+h2*m2V2)+(E(i)+h2*m2E)+(I(i)+h2*m2I) +(R(i)+h2*m2R)) - sigma*(E
(i)+h2*m2E) + (beta1*(V1(i)+h2*m2V1)*(I(i)+h2*m2I))/((S(i)+h2*m2S)+(
V1(i)+h2*m2V1) +(V2(i)+h2*m2V2)+(E(i)+h2*m2E)+(I(i)+h2*m2I)+(R(i)+h2*
m2R)) + (beta2*(V2(i)+h2*m2V2)*(I(i)+h2*m2I))/((S(i)+h2*m2S)+(V1(i)+
h2*m2V1)+(V2(i)+h2*m2V2)+(E(i)+h2*m2E)+(I(i)+h2*m2I)+(R(i)+h2*m2R));
m3I = sigma*(E(i)+h2*m2E) - gamma*(I(i)+h2*m2I);
85 m3R = gamma*(I(i)+h2*m2I) - delta*(R(i)+h2*m2R) + alpha1*(V1(i)+h2*m2V1)
+ alpha2*(V2(i)+h2*m2V2);
%====
87 m4S = - (beta*(S(i)+h*m3S)*(I(i)+h*m3I))/((S(i)+h*m3S)+(V1(i)+h*m3V1)+(V2
(i)+h*m3V2) +(E(i)+h*m3E)+(I(i)+h*m3I)+(R(i)+h*m3R)) - u1(i+1)*(S(i)+
h*m3S) - u2(i+1)*(S(i)+h*m3S) + epsilon1*(V1(i)+h*m3V1) + epsilon2*(
V2(i)+h*m3V2) + delta*(R(i)+h*m3R);
m4V1 = u1(i+1)*(S(i)+h*m3S) - (beta1*(V1(i)+h*m3V1)*(I(i)+h*m3I))/((S(i)+
h*m3S)+(V1(i)+h*m3V1)+(V2(i)+h*m3V2) +(E(i)+h*m3E)+(I(i)+h*m3I)+(R(i)
+h*m3R)) - (epsilon1 + alpha1)*(V1(i)+h*m3V1);
89 m4V2 = u2(i+1)*(S(i)+h*m3S) - (beta2*(V2(i)+h*m3V2)*(I(i)+h*m3I))/((S(i)+
h*m3S)+(V1(i)+h*m3V1)+(V2(i)+h*m3V2) +(E(i)+h*m3E)+(I(i)+h*m3I)+(R(i)
+h*m3R)) - (epsilon2 + alpha2)*(V2(i)+h*m3V1);
m4E = (beta*(S(i)+h*m3S)*(I(i)+h*m3I))/((S(i)+h*m3S)+(V1(i)+h*m3V1)+(V2(i)
)+h*m3V2)+(E(i)+h*m3E)+(I(i)+h*m3I) +(R(i)+h*m3R)) - sigma*(E(i)+h*
m3E) + (beta1*(V1(i)+h*m3V1)*(I(i)+h*m3I))/((S(i)+h*m3S)+(V1(i)+h*
m3V1) +(V2(i)+h*m3V2) +(E(i)+h*m3E)+(I(i)+h*m3I)+(R(i)+h*m3R)) + (
beta2*(V2(i)+h*m3V2)*(I(i)+h*m3I))/((S(i)+h*m3S)+(V1(i)+h*m3V1)+(V2(i)
)+h*m3V2)+(E(i)+h*m3E)+(I(i)+h*m3I)+(R(i)+h*m3R));
91 m4I = sigma*(E(i)+h*m3E) - gamma*(I(i)+h*m3I);

```

```

m4R = gamma*(I(i)+h*m3I) - delta*(R(i)+h*m3R) + alpha1*(V1(i)+h*m3V1) +
      alpha2*(V2(i)+h*m3V2);
93      %====
S(i+1) = S(i) + (h/6)*(m1S + 2*m2S + 2*m3S + m4S);
95      V1(i+1)=V1(i) + (h/6)*(m1V1 + 2*m2V1 + 2*m3V1 + m4V1);
V2(i+1)=V2(i) + (h/6)*(m1V2 + 2*m2V2 + 2*m3V2 + m4V2);
97      E(i+1) = E(i) + (h/6)*(m1E + 2*m2E + 2*m3E + m4E);
I(i+1) = I(i) + (h/6)*(m1I + 2*m2I + 2*m3I + m4I);
99      R(i+1) = R(i) + (h/6)*(m1R + 2*m2R + 2*m3R + m4R);

end
101 %Runge-Kutta para variaveis adjuntas
for i = 1:M
103     j = M + 2 - i;
m1S = (1/((S(j)+V1(j)+V2(j)+E(j)+I(j)+R(j))^2))*(beta1*V1(j)*I(j)*(
      lambdaE(j)-lambdaV1(j)) + beta2*V2(j)*I(j)*(lambdaE(j)-lambdaV2(j)) +
      beta*I(j)*(V1(j)+V2(j)+E(j)+I(j)+R(j))*(lambdaS(j)-lambdaE(j))) + u1
      (j)*(lambdaS(j)-lambdaV1(j)) + u2(j)*(lambdaS(j)-lambdaV2(j));
105     m1V1 = (1/((S(j)+V1(j)+V2(j)+E(j)+I(j)+R(j))^2))*(beta*S(j)*I(j)*(lambdaE
      (j)-lambdaS(j)) + beta2*V2(j)*I(j)*(lambdaE(j)-lambdaV2(j)) + beta1*I
      (j)*(S(j)+V2(j)+E(j)+I(j)+R(j))*(lambdaV1(j)-lambdaE(j))) + epsilon1
      *(lambdaV1(j)-lambdaS(j)) + alpha1*(lambdaV1(j)-lambdaR(j));
m1V2 = (1/((S(j)+V1(j)+V2(j)+E(j)+I(j)+R(j))^2))*(beta*S(j)*I(j)*(lambdaE
      (j)-lambdaS(j)) + beta1*V1(j)*I(j)*(lambdaE(j)-lambdaV1(j)) + beta2*I
      (j)*(S(j)+V1(j)+E(j)+I(j)+R(j))*(lambdaV2(j)-lambdaE(j))) + epsilon2
      *(lambdaV2(j)-lambdaS(j)) + alpha2*(lambdaV2(j)-lambdaR(j));
107     m1E = (1/((S(j)+V1(j)+V2(j)+E(j)+I(j)+R(j))^2))*(beta*S(j)*I(j)*(lambdaE(
      j)-lambdaS(j)) + beta1*V1(j)*I(j)*(lambdaE(j)-lambdaV1(j)) + beta2*V2
      (j)*I(j)*(lambdaE(j)-lambdaV2(j))) + sigma*(lambdaE(j)-lambdaI(j));
m1I = (1/((S(j)+V1(j)+V2(j)+E(j)+I(j)+R(j))^2))*(beta*S(j)*(S(j)+V1(j)+V2
      (j)+E(j)+R(j))*(lambdaS(j)-lambdaE(j)) + beta1*V1(j)*(S(j)+V1(j)+V2(j)
      )+E(j)+R(j))*(lambdaV1(j)-lambdaE(j)) + beta2*V2(j)*(S(j)+V1(j)+V2(j)
      )+E(j)+R(j))*(lambdaV2(j)-lambdaE(j))) + gamma*(lambdaI(j)-lambdaR(j))
      - 1;
109     m1R = (1/((S(j)+V1(j)+V2(j)+E(j)+I(j)+R(j))^2))*(beta*S(j)*I(j)*(lambdaE(
      j)-lambdaS(j)) + beta1*V1(j)*I(j)*(lambdaE(j)-lambdaV1(j)) + beta2*V2
      (j)*I(j)*(lambdaE(j)-lambdaV2(j))) + delta*(lambdaR(j)-lambdaS(j));
      %====
111     m2S = (1/(((0.5*(S(j)+S(j-1)))+(0.5*(V1(j)+V1(j-1)))+(0.5*(V2(j)+V2(j-1))
      )+(0.5*(E(j)+E(j-1)))+(0.5*(I(j)+I(j-1)))+(0.5*(R(j)+R(j-1))))^2))*
      (beta1*(0.5*(V1(j)+V1(j-1)))*(0.5*(I(j)+I(j-1)))*((lambdaE(j)-h2*m1E)
      -(lambdaV1(j)-h2*m1V1)) + beta2*(0.5*(V2(j)+V2(j-1)))*(0.5*(I(j)+I(j
      -1)))*((lambdaE(j)-h2*m1E)-(lambdaV2(j)-h2*m1V2)) + beta*(0.5*(I(j)+I
      (j-1)))*((0.5*(V1(j)+V1(j-1)))+(0.5*(V2(j)+V2(j-1)))+(0.5*(E(j)+E(j
      -1)))+(0.5*(I(j)+I(j-1)))+(0.5*(R(j)+R(j-1))))*(lambdaS(j)-h2*m1S)
      -(lambdaE(j)-h2*m1E))) + 0.5*(u1(j)+u1(j-1))*((lambdaS(j)-h2*m1S)-(
      lambdaV1(j)-h2*m1V1)) + 0.5*(u2(j)+u2(j-1))*((lambdaS(j)-h2*m1S)-(
      lambdaV2(j)-h2*m1V2));

```



```

-1)))*((lambdaE(j)-h2*m2E)-(lambdaV2(j)-h2*m2V2)) + beta*(0.5*(I(j)+I
(j-1)))*((0.5*(V1(j)+V1(j-1)))+(0.5*(V2(j)+V2(j-1)))+(0.5*(E(j)+E(j
-1)))+(0.5*(I(j)+I(j-1)))+(0.5*(R(j)+R(j-1)))))*((lambdaS(j)-h2*m2S)-
(lambdaE(j)-h2*m2E)) + 0.5*(u1(j)+u1(j-1))*((lambdaS(j)-h2*m2S)-
(lambdaV1(j)-h2*m2V1)) + 0.5*(u2(j)+u2(j-1))*((lambdaS(j)-h2*m2S)-
(lambdaV2(j)-h2*m2V2));
%====
119 m3V1 = (1/(((0.5*(S(j)+S(j-1)))+(0.5*(V1(j)+V1(j-1)))+(0.5*(V2(j)+V2(j-1)
)))+(0.5*(E(j)+E(j-1)))+(0.5*(I(j)+I(j-1)))+(0.5*(R(j)+R(j-1))))^2))*
(beta*(0.5*(S(j)+S(j-1)))*0.5*(I(j)+I(j-1)))*((lambdaE(j)-h2*m2E)-
(lambdaS(j)-h2*m2S)) + beta2*(0.5*(V2(j)+V2(j-1)))*0.5*(I(j)+I(j-1))
)*((lambdaE(j)-h2*m2E)-(lambdaV2(j)-h2*m2V2)) + beta1*(0.5*(I(j)+I(j
-1)))*((0.5*(S(j)+S(j-1)))+(0.5*(V2(j)+V2(j-1)))+(0.5*(E(j)+E(j-1))
)+(0.5*(I(j)+I(j-1)))+(0.5*(R(j)+R(j-1)))))*((lambdaV1(j)-h2*m2V1)-
(lambdaE(j)-h2*m2E)) + epsilon1*((lambdaV1(j)-h2*m2V1)-(lambdaS(j)-h2
*m2S)) + alpha1*((lambdaV1(j)-h2*m2V1)-(lambdaR(j)-h2*m2R));
m3V2 = (1/(((0.5*(S(j)+S(j-1)))+(0.5*(V1(j)+V1(j-1)))+(0.5*(V2(j)+V2(j-1)
)))+(0.5*(E(j)+E(j-1)))+(0.5*(I(j)+I(j-1)))+(0.5*(R(j)+R(j-1))))^2))*
(beta*(0.5*(S(j)+S(j-1)))*0.5*(I(j)+I(j-1)))*((lambdaE(j)-h2*m2E)-
(lambdaS(j)-h2*m2S)) + beta1*(0.5*(V1(j)+V1(j-1)))*0.5*(I(j)+I(j-1))
)*((lambdaE(j)-h2*m2E)-(lambdaV1(j)-h2*m2V1)) + beta2*(0.5*(I(j)+I(j
-1)))*((0.5*(S(j)+S(j-1)))+(0.5*(V1(j)+V1(j-1)))+(0.5*(E(j)+E(j-1))
)+(0.5*(I(j)+I(j-1)))+(0.5*(R(j)+R(j-1)))))*((lambdaV2(j)-h2*m2V2)-
(lambdaE(j)-h2*m2E)) + epsilon2*((lambdaV2(j)-h2*m2V2)-(lambdaS(j)-h2
*m2S)) + alpha2*((lambdaV2(j)-h2*m2V2)-(lambdaR(j)-h2*m2R));
121 m3E = (1/(((0.5*(S(j)+S(j-1)))+(0.5*(V1(j)+V1(j-1)))+(0.5*(V2(j)+V2(j-1)
)))+(0.5*(E(j)+E(j-1)))+(0.5*(I(j)+I(j-1)))+(0.5*(R(j)+R(j-1))))^2))*
(beta*(0.5*(S(j)+S(j-1)))*0.5*(I(j)+I(j-1)))*((lambdaE(j)-h2*m2E)-
(lambdaS(j)-h2*m2S)) + beta1*(0.5*(V1(j)+V1(j-1)))*0.5*(I(j)+I(j-1))
)*((lambdaE(j)-h2*m2E)-(lambdaV1(j)-h2*m2V1)) + beta2*(0.5*(V2(j)+V2(j
-1)))*0.5*(I(j)+I(j-1)))*((lambdaE(j)-h2*m2E)-(lambdaV2(j)-h2*m2V2))
) + sigma*((lambdaE(j)-h2*m2E)-(lambdaI(j)-h2*m2I));
m3I = (1/(((0.5*(S(j)+S(j-1)))+(0.5*(V1(j)+V1(j-1)))+(0.5*(V2(j)+V2(j-1)
)))+(0.5*(E(j)+E(j-1)))+(0.5*(I(j)+I(j-1)))+(0.5*(R(j)+R(j-1))))^2))*
(beta*(0.5*(S(j)+S(j-1)))*((0.5*(S(j)+S(j-1)))+(0.5*(V1(j)+V1(j-1))
)+(0.5*(V2(j)+V2(j-1)))+(0.5*(E(j)+E(j-1)))+(0.5*(R(j)+R(j-1)))))*
(lambdaS(j)-h2*m2S)-(lambdaE(j)-h2*m2E)) + beta1*(0.5*(V1(j)+V1(j-1))
)*((0.5*(S(j)+S(j-1)))+(0.5*(V1(j)+V1(j-1)))+(0.5*(V2(j)+V2(j-1))
)+(0.5*(E(j)+E(j-1)))+(0.5*(R(j)+R(j-1)))))*((lambdaV1(j)-h2*m2V1)-
(lambdaE(j)-h2*m2E)) + beta2*(0.5*(V2(j)+V2(j-1)))*((0.5*(S(j)+S(j-1)
)))+(0.5*(V1(j)+V1(j-1)))+(0.5*(V2(j)+V2(j-1)))+(0.5*(I(j)+I(j-1))
)+(0.5*(R(j)+R(j-1)))))*((lambdaV2(j)-h2*m2V2)-(lambdaE(j)-h2*m2E)) +
gamma*((lambdaI(j)-h2*m2I)-(lambdaR(j)-h2*m2R)) - 1;
123 m3R = (1/(((0.5*(S(j)+S(j-1)))+(0.5*(V1(j)+V1(j-1)))+(0.5*(V2(j)+V2(j-1)
)))+(0.5*(E(j)+E(j-1)))+(0.5*(I(j)+I(j-1)))+(0.5*(R(j)+R(j-1))))^2))*
(beta*(0.5*(S(j)+S(j-1)))*0.5*(I(j)+I(j-1)))*((lambdaE(j)-h2*m2E)-
(lambdaS(j)-h2*m2S)) + beta1*(0.5*(V1(j)+V1(j-1)))*0.5*(I(j)+I(j-1))

```

```

        *((lambdaE(j)-h2*m2E)-(lambdaV1(j)-h2*m2V1)) + beta2*(0.5*(V2(j)+V2(j-1)))
        *(0.5*(I(j)+I(j-1)))*((lambdaE(j)-h2*m2E)-(lambdaV2(j)-h2*m2V2))
        ) + delta*((lambdaR(j)-h2*m2R)-(lambdaS(j)-h2*m2S));
%=====
125 m4S = (1/((S(j-1)+V1(j-1)+V2(j-1)+E(j-1)+I(j-1)+R(j-1))^2))*(beta1*V1(j-1)*I(j-1)*((lambdaE(j)-h*m3E)-(lambdaV1(j)-h*m3V1)) + beta2*V2(j-1)*I(j-1)*((lambdaE(j)-h*m3E)-(lambdaV2(j)-h*m3V2)) + beta*I(j-1)*(V1(j-1)+V2(j-1)+E(j-1)+I(j-1)+R(j-1))*((lambdaS(j)-h*m3S)-(lambdaE(j)-h*m3E))) + u1(j-1)*((lambdaS(j)-h*m3S)-(lambdaV1(j)-h*m3V1)) + u2(j-1)*((lambdaS(j)-h*m3S)-(lambdaV2(j)-h*m3V2)));
m4V1 = (1/((S(j-1)+V1(j-1)+V2(j-1)+E(j-1)+I(j-1)+R(j-1))^2))*(beta*S(j-1)*I(j-1)*((lambdaE(j)-h*m3E)-(lambdaS(j)-h*m3S)) + beta2*V2(j-1)*I(j-1)*((lambdaE(j)-h*m3E)-(lambdaV2(j)-h*m3V2)) + beta1*I(j-1)*(S(j-1)+V2(j-1)+E(j-1)+I(j-1)+R(j-1))*((lambdaV1(j)-h*m3V1)-(lambdaE(j)-h*m3E)))+ epsilon1*((lambdaV1(j)-h*m3V1)-(lambdaS(j)-h*m3S)) + alpha1*((lambdaV1(j)-h*m3V1)-(lambdaR(j)-h*m3R)));
127 m4V2 = (1/((S(j-1)+V1(j-1)+V2(j-1)+E(j-1)+I(j-1)+R(j-1))^2))*(beta*S(j-1)*I(j-1)*((lambdaE(j)-h*m3E)-(lambdaS(j)-h*m3S)) + beta1*V1(j-1)*I(j-1)*((lambdaE(j)-h*m3E)-(lambdaV1(j)-h*m3V1)) + beta2*I(j-1)*(S(j-1)+V1(j-1)+E(j-1)+I(j-1)+R(j-1))*((lambdaV2(j)-h*m3V2)-(lambdaE(j)-h*m3E))) + epsilon2*((lambdaV2(j)-h*m3V2)-(lambdaS(j)-h*m3S)) + alpha2*((lambdaV2(j)-h*m3V2)-(lambdaR(j)-h*m3R)));
m4E = (1/((S(j-1)+V1(j-1)+V2(j-1)+E(j-1)+I(j-1)+R(j-1))^2))*(beta*S(j-1)*I(j-1)*((lambdaE(j)-h*m3E)-(lambdaS(j)-h*m3S)) + beta1*V1(j-1)*I(j-1)*((lambdaE(j)-h*m3E)-(lambdaV1(j)-h*m3V1)) + beta2*V2(j-1)*I(j-1)*((lambdaE(j)-h*m3E)-(lambdaV2(j)-h*m3V2))) + sigma*((lambdaE(j)-h2*m2E)-(lambdaI(j)-h2*m2I));
129 m4I = (1/((S(j-1)+V1(j-1)+V2(j-1)+E(j-1)+I(j-1)+R(j-1))^2))*(beta*S(j-1)*(S(j-1)+V1(j-1)+V2(j-1)+E(j-1)+R(j-1))*((lambdaS(j)-h*m3S)-(lambdaE(j)-h*m3E)) + beta1*V1(j-1)*(S(j-1)+V1(j-1)+V2(j-1)+E(j-1)+R(j-1))*((lambdaV1(j)-h*m3V1)-(lambdaE(j)-h*m3E)) + beta2*V2(j-1)*(S(j-1)+V1(j-1)+V2(j-1)+I(j-1)+R(j-1))*((lambdaV2(j)-h*m3V2)-(lambdaE(j)-h*m3E))) + gamma*((lambdaI(j)-h*m3I)-(lambdaR(j)-h*m3R)) - 1;
m4R = (1/((S(j-1)+V1(j-1)+V2(j-1)+E(j-1)+I(j-1)+R(j-1))^2))*(beta*S(j-1)*I(j-1)*((lambdaE(j)-h*m3E)-(lambdaS(j)-h*m3S)) + beta1*V1(j-1)*I(j-1)*((lambdaE(j)-h*m3E)-(lambdaV1(j)-h*m3V1)) + beta2*V2(j-1)*I(j-1)*((lambdaE(j)-h*m3E)-(lambdaV2(j)-h*m3V2))) + delta*((lambdaR(j)-h*m3R)-(lambdaS(j)-h*m3S));
131 %=====
lambdaS(j-1) = lambdaS(j) - (h/6)*(m1S + 2*m2S + 2*m3S + m4S);
133 lambdaV1(j-1)=lambdaV1(j) - (h/6)*(m1V1 + 2*m2V1 + 2*m3V1 + m4V1);
lambdaV2(j-1)=lambdaV2(j) - (h/6)*(m1V2 + 2*m2V2 + 2*m3V2 + m4V2);
135 lambdaE(j-1) = lambdaE(j) - (h/6)*(m1E + 2*m2E + 2*m3E + m4E);
lambdaI(j-1) = lambdaI(j) - (h/6)*(m1I + 2*m2I + 2*m3I + m4I);
137 lambdaR(j-1) = lambdaR(j) - (h/6)*(m1R + 2*m2R + 2*m3R + m4R);
end
139 tempu1=(S.*(lambdaS - lambdaV1))./2*B1;

```

```
tempu2=(S.*(lambdaS - lambdaV2))./2*B2;
141 u11 = min(1,max(0,tempu1));
    u21 = min(1,max(0,tempu2));
143 u1 = 0.5*(u11 + oldu1);
    u2 = 0.5*(u21 + oldu2);
145 % Teste para criterio de parada
    temp1 = erro*sum(abs(u1)) - sum(abs(oldu1 - u1));
147 temp2 = erro*sum(abs(u2)) - sum(abs(oldu2 - u2));
    temp3 = erro*sum(abs(S)) - sum(abs(oldS - S));
149 temp4 = erro*sum(abs(V1)) - sum(abs(oldV1 - V1));
    temp5 = erro*sum(abs(V2)) - sum(abs(oldV2 - V2));
151 temp6 = erro*sum(abs(E)) - sum(abs(oldE - E));
    temp7 = erro*sum(abs(I)) - sum(abs(oldI - I));
153 temp8 = erro*sum(abs(R)) - sum(abs(oldR - R));
    temp9 = erro*sum(abs(lambdaS)) - sum(abs(oldlambdaS - lambdaS));
155 temp10 = erro*sum(abs(lambdaV1)) - sum(abs(oldlambdaV1 - lambdaV1));
    temp11 = erro*sum(abs(lambdaV2)) - sum(abs(oldlambdaV2 - lambdaV2));
157 temp12 = erro*sum(abs(lambdaE)) - sum(abs(oldlambdaE - lambdaE));
    temp13 = erro*sum(abs(lambdaI)) - sum(abs(oldlambdaI - lambdaI));
159 temp14 = erro*sum(abs(lambdaR)) - sum(abs(oldlambdaR - lambdaR));
    test = min(temp1, min(temp2, min(temp3, min(temp4, min(temp5, min(temp6, min(
        temp7, min(temp8, min(temp9, min(temp10, min(temp11, min(temp12, min(
        temp13, temp14))))))))))))));
161 end
    y(1,:) = t;
163 y(2,:) = S;
    y(3,:) = V1;
165 y(4,:) = V2;
    y(5,:) = E;
167 y(6,:) = I;
    y(7,:) = R;
169 y(8,:) = u1;
    y(9,:) = u2;
```