



**UNIVERSIDAD MICHOACANA
DE SAN NICOLÁS DE HIDALGO**



FACULTAD DE INGENIERÍA QUÍMICA

**APLICACIONES DE CONTROL ÓPTIMO EN LA DINÁMICA
DE TRANSMISIÓN DEL DENGUE CLÁSICO CON
CONTROL MECÁNICO Y PROFILAXIS**

TESIS QUE PARA OBTENER EL TÍTULO DE:

INGENIERO QUÍMICO

PRESENTA:

P.I.Q. ANGEL ALFARO BERNARDINO

ASESOR:

Dr. FABRICIO NÁPOLES RIVERA

MORELIA, MICHOACÁN, FEBRERO DEL 2020

DEDICATORIA

A la memoria de Artemia Franco Rosales por haber tenido un gran carácter y visión para la vida.

AGRADECIMIENTOS

Agradecer a la Facultad de Ingeniería Química a la división de posgrado por darme un espacio en este, para realizar el programa en sus dispositivos (computadoras).

Agradezco a mi asesor de tesis Dr. Fabricio Nápoles Rivera, por todo este apoyo, que, aunque yo fallaba varios días, él nunca perdía la confianza en mí. Eso vale mucho para mí, que confíen en uno, porque eso motiva para escalar la montaña más alta del mundo. Valoro todo ese apoyo moral que me dio en el transcurso de la realización de la tesis, porque con la motivación uno puede hacer cualquier cosa, mi asesor Fabricio Nápoles Rivera logro hacerlo. También hizo que viera que más podría hacer y ampliar mi visión de la vida.

A mi tío, Julio Cesar Magadan Franco, que siempre me iba a nueva Italia me invitaba a cenar y salíamos a diferentes lugares turísticos con su familia. Dejándome diferentes experiencias en todos esos viajes que hicimos. Y que buena sabia la cena invitado por usted tío Julio Cesar Magadan Franco.

A mi tío, Alfonso Franco Rosales, que siempre me preguntaba que necesitaba para estar en la universidad, yo por pena, decía, que no necesitaba nada que todo estaba bien, agradezco por siempre estar pendiente en lo que necesitara en este periodo de tiempo.

A mi tío, Francisco Javier Magadan Franco, que siempre me apoyo a mi familia cuando no tenían un sustento económico, él siempre vio por mí cuando menos tenia, siempre me decía aquí tienes, para que pases unos días en morelia, no es mucho, pero te servirá para que sigas adelante. Él fue una pieza fundamental en la mi formación como persona, tanto en lo moral, que me decía como era antes de difícil para salir adelante y motivarme para poder hacer todo lo que me proponga. Y seguir la tradición en la familia de ayudar. Porque la familia esta para ayudar, para seguir adelante y tener otra visión de la misma.

A mis padres, María Guadalupe Martina Bernardino Lomelí y Ángel Alfaro Franco, que, en toda mi carrera, cinco años con tres mil pesos pudieron darme estudio, no sé cómo lo hicieron, todos se preguntan cómo lo hicieron, solo ellos saben lo que hicieron para salir adelante, aunque fuera imposible de hacer y tuvieran todo en contra, ellos tuvieron la inteligencia de lograrlo. Agradezco el hambre de mis padres para que me vieran salir adelante. Tal vez por esa inteligencia que ellos tienen para hacer todo posible, tenga la facilidad para resolver grandes problemas, pero nunca podré imaginar cómo lo hicieron para tener este sustento estos cinco años para mantener mis estudios.

También por todo el esfuerzo de mis padres para darme todo y nunca me faltara nada, porque siempre teniendo un motivo en común entre ellos, se hace real, esto comenzó con un sueño, que se fue trabajando varios años y se hizo realidad, con esfuerzo y sacrificio de mis padres y mi parte para salir adelante. Ahora agradezco por darme una carrera, más que una carrera, una visión diferente de la vida, un cambio de mentalidad y eso vale más que todo el oro del mundo. Porque ni con todo el oro del mundo se puede comprar un segundo y mis padres dieron un parte de su vida para hacer de mí, una mejor persona.

ÍNDICE

Dedicatoria.....	I
Agradecimientos	II
Lista de figuras.....	VII
Lista de tablas.....	VIII
Resumen	1
Abstract.....	3
Introducción	5
Antecedentes.....	9
Justificación	10
Hipótesis	11
Objetivo Principal.....	12
Objetivos particulares.....	12
Descripción del trabajo.....	13
Capítulo 1 Generalidades.....	14
1.1. Fundamento Teórico.....	14
1.1.1. Planteamiento de problemas de optimización	15
1.1.2. Definición de modelo matemático	15
1.1.3. Políticas de control.....	16
1.1.4. Restricciones adicionales.....	16
1.1.5. Función objetivo.....	17
1.1.6. Problemas de control óptimo.....	19
1.1.7. Condiciones necesarias y suficientes de optimalidad.....	19
1.1.8. Principio del máximo	23
1.1.9. Método del gradiente reducido.....	24
1.2. Modelo de la dinámica de transmisión del dengue clásico.....	25
1.2.1. Modelo del dengue clásico.....	26
1.2.2. Aplicación de escenarios del modelo	29

1.2.2.1. Caso sin control	29
1.2.2.2. Caso profilaxis	30
1.2.2.3. Caso control mecánico.....	33
1.2.2.4. Caso combinación de controles.....	36
Capítulo 2 Desarrollo Experimental Resolución del modelo.....	38
2.1. Función objetivo	39
2.2. Solución del modelo	39
2.3. Diagrama de flujo del algoritmo del método del gradiente reducido	45
Capítulo 3 Análisis y Discusión de Resultados	46
3.1. Especificaciones generales.....	47
3.2. Método de Euler.....	48
3.2.1. Caso 1	48
3.2.2. Caso 2	49
3.2.3. Caso 3	50
3.3. Método de Rugen-Kutta de 4to (cuarto orden).....	51
3.3.1. Caso 1	52
3.3.2. Caso 2	53
3.3.3. Caso 3	54
3.4. Discusión de resultados	56
Capítulo 4	60
4.1. Conclusiones	60
4.2. Recomendaciones	62
Bibliografía.....	64
Apéndice 1 Código del método de Euler controles combinados	66
Apéndice 2 Código del método de Rugen-Kutta 4 ^{to} orden controles combinados.....	79

LISTA DE FIGURAS

Figura 1 .- Comparación gráfica de las ecuaciones (1.1.10 - 1.1.12). a) Función tipo Mayer, b) Función Lagrangiana y c) función tipo Bolza	18
Figura 2 .- Gráficas de las variables de estado en el caso sin control.....	30
Figura 3 .- Gráficas de las variables de estado en el caso profilaxis con $u_1 = 0.3$	31
Figura 4 .- Gráficas de las variables de estado en el caso profilaxis con $u_1 = 0.6$	32
Figura 5 .- Gráficas de las variables de estado en el caso profilaxis con $u_1 = 0.9$	33
Figura 6 .- Gráficas de las variables de estado en el caso control mecánico con $u_2 = 0.2$	34
Figura 7 .- Gráficas de las variables de estado en el caso control mecánico con $u_2 = 0.4$	35
Figura 8 .- Gráficas de las variables de estado en el caso control mecánico con $u_2 = 0.65$	35
Figura 9 .- Gráficas de las variables de estado en el caso combinación de controles con $u_1 = 0.6$ y $u_2 = 0.65$	37
Figura 10 .- Diagrama del algoritmo del método del gradiente reducido	45
Figura 11 .- Gráficas de las variables de estado para el caso 1 en el método de Euler.....	48
Figura 12 .- Gráficas de los controles u_1 y u_2 para el caso 1 en el método de Euler.....	49
Figura 13 .- Gráficas de las variables de estado para el caso 2 en el método de Euler.....	50
Figura 14 .- Gráfica del control u_1 para el caso 2 en el método de Euler.....	50
Figura 15 .- Gráficas de las variables de estado para el caso 3 en el método de Euler.....	51
Figura 16 .- Gráfica del control u_2 para el caso 3 en el método de Euler.....	51
Figura 17 .- Gráficas de las variables de estado para el caso 1 en el método de Rugen-Kutta 4 ^{to} (cuarto orden).....	52
Figura 18 .- Gráficas de los controles u_1 y u_2 para el caso 1 en el método de Rugen-Kutta 4 ^{to} (cuarto orden)	53
Figura 19 .- Gráficas de las variables de estado para el caso 2 en el método de Rugen-Kutta 4 ^{to} (cuarto orden).....	54
Figura 20 .- Gráfica del control u_1 para el caso 2 en el método de Rugen-Kutta 4 ^{to} (cuarto orden)	54
Figura 21 .- Gráficas de las variables de estado para el caso 3 en el método de Rugen-Kutta 4 ^{to} (cuarto orden).....	55
Figura 22 .- Gráfica del control u_2 para el caso 3 en el método de Rugen-Kutta 4to (cuarto orden)	55

LISTA DE TABLAS

Tabla 1 .- Significado y valores asignados a las variables	26
Tabla 2 .- Significado y valores asignados a las variables	26
Tabla 3 .- Significado y valores asignados a los parámetros.....	28
Tabla 4 .- Valores de las funciones peso en la función objetivo (función costo)	47
Tabla 5 .- Resultados de la función objetivo (función costo) de las simulaciones.....	56

RESUMEN

El dengue es una enfermedad vírica, transmitida entre humanos por mosquitos hembras principalmente de la especie *Aedes aegypti* y, en menor grado, de *A. albopictus*. Los últimos estudios que se han realizado sobre el dengue en todo el mundo han arrojado un aumento en la incidencia de la enfermedad. Como lo muestra un estudio, donde se producen 390 millones de infecciones por dengue cada año, de los cuales 96 millones se manifiestan clínicamente. Un estudio más reciente de la OPS (organización panamericana de la salud) reveló que, en la región de las Américas, notificaron 1.191.815 casos de dengue (118,5 casos por 100.000 habitantes), de los cuales 546.589 casos (46%) fueron confirmados por criterios de laboratorio. Del total de casos notificados, 5.599 (0,47%) fueron clasificados como dengue grave y la letalidad reportada fue de 0,02%. Los esfuerzos para poder predecir el comportamiento de la enfermedad vírica del dengue han dado como resultado modelos matemáticos con una orientación en población humana constante y variable, transmisión vertical del mosquito, espacio temporal, de los diferentes serotipos del virus, la influencia del clima en la ocurrencia del dengue, estructura poblacional y el control del mosquito. Todos estos modelos ya mencionados mantienen un control constante durante el periodo del estudio. En este trabajo se manejará un control óptimo (optimización dinámica) buscando minimizar el número promedio de personas infectadas a través de la determinación del control mecánico y profilaxis óptimo. También el objetivo incluye la minimización del número promedio total de mosquitos portadores del virus, población total de mosquitos y número de promedio de criaderos, siendo estas las variables que más influyen en la dinámica del proceso, siendo seis ecuaciones diferenciales ordinarias. En este trabajo se propone el uso de los métodos numéricos de Euler y Runge-Kutta 4^{to} orden para la resolución del sistema dinámico. Como todo problema de control óptimo es necesario de variables adjuntas, en las cuales, siempre es necesario manejar valores de frontera, estos serán obtenidos por la aplicación del principio del máximo de Pontryagin. Las

ecuaciones adjuntas se resolverán con los métodos numéricos que se ha utilizado por las ecuaciones diferenciales ordinarias.

Los perfiles óptimos obtenidos en este trabajo tratan de representar predicciones teóricas de la dinámica de transmisión del dengue clásico con control mecánico y profilaxis para ayudar a la estrategia del control de la enfermedad.

Palabras clave: ***Modelo Matemático, Función Objetivo, Principio del máximo, Gradiente reducido, Variables adjuntas.***

ABSTRACT

Dengue is a viral disease, transmitted to humans by female mosquitoes mainly of the *Aedes aegypti* species and, in a lesser extent, of *A. albopictus*. Latest studies on dengue have shown an increase in the incidence worldwide of the disease. Some studies indicate that 390 million dengue infections occur each year, from which 96 million are clinically manifested. A more recent study by PAHO (Pan American Health Organization) revealed that, in the region of the Americas, there have been reported 1,191,815 cases of dengue (118.5 cases per 100,000 inhabitants), of which 546,589 cases (46%) were confirmed by laboratory criteria. Of the total reported cases, 5,599 (0.47%) were classified as severe dengue and the lethality reported was 0.02%. Efforts to predict the behavior of dengue have resulted in mathematical models that consider different aspects of the disease transmission such as a constant and variable human population, vertical transmission of the mosquito, account of the different serotypes of the virus, the influence of climate on the Dengue occurrence, population structure and mosquito control. All these models already mentioned maintain a constant control during the study period. In this work, an optimal control (dynamic optimization) will be used to minimize the average number of infected people through the determination of mechanical control and optimal prophylaxis. The objective also includes the minimization of the total average number of mosquitoes carrying the virus, total population of mosquitoes and average number of hatching sites, these being the variables that most influence the dynamics of the process, being six ordinary differential equations. This paper proposes the use of the numerical methods of Euler and Ruge-Kutta 4th order for the resolution of the dynamic system. Like any optimal control problem, it is necessary to have adjoint variables, in which, it is always necessary to manage frontier values, these will be obtained by applying the principle of the maximum of Pontryagin. The adjoint equations will be solved with the numerical methods that have been used with the ordinary differential equations. The optimal profiles obtained in this work try to represent theoretical

predictions of the transmission dynamics of classic dengue with mechanical control and prophylaxis to help the disease control strategy.

INTRODUCCIÓN

Todos en nuestra mente tenemos una idea de lo que es **OPTIMIZACIÓN**, pero la gran mayoría no sabe en realidad lo que es concretamente. Diwekar en su trabajo Introduction to Applied Optimization, da la siguiente definición: “La optimización es un proceso sistemático de toma de decisiones”, mientras que, Edgar-Himmelblau-Lasdon, en Optimization of chemical processes, menciona lo siguiente: “La optimización es el uso de métodos específicos para determinar la solución más eficaz y rentable a un problema o diseño de un proceso”. Ambos coinciden en que la optimización está relacionada con la toma de decisiones para encontrar solución a problemas respecto a su área de estudio. Una definición más general sería, “La optimización es la selección de los mejores valores para las variables de decisión que maximizan o minimizan una función objetivo dada”. Teniendo una idea más clara de lo que es optimización, son necesarios tres factores para llevarla a cabo:

1. El sistema a optimizar (nuestro caso el modelo matemático)
2. Una o más variables para medir la eficiencia del sistema
3. Procedimiento para dar solución al sistema

También es muy importante mencionar que aparte del sistema, hay otros parámetros que afectan la optimización, como es el caso de las restricciones, variables de decisión y función objetivo. Como son el caso de La programación lineal (LP) y La Programación no lineal (PNL) que solo se diferencian en las restricciones, mientras que, en una son lineales en otra es no lo son. En la programación entera (IP) pueden contener variables continuas y discretas. En el caso de la programación mixta entera lineal (MILP) combina los casos anteriores con restricciones lineales, mientras que las variables de decisión pueden ser escalares, enteras o continuas. Mientras que su contraparte La programación mixta entera no lineal (MINLP) involucra decisiones enteras y variables continuas. En el control óptimo las variables de decisión son vectores, para el caso de la

programación estocástica las variables son inciertas (aleatorias) y por último la optimización de múltiple objetivo que no tiene un solo objetivo. La optimización de este trabajo con base en el control óptimo en un sistema dinámico no lineal.

El **CONTROL ÓPTIMO** también conocido como optimización dinámica, al igual que en los diferentes tipos de optimización debe de contar con una función objetivo (siendo esta una integral), restricciones y variables de decisión, como se había mencionado en el apartado de optimización. En el control óptimo siempre es usado un sistema de ecuaciones algebraicas y/o diferenciales, valores iniciales de las variables, los límites en los que va operar los controles, variables y restricciones. Estos parámetros utilizados por el control óptimo hacen que sean parte de un subconjunto de problemas llamados problemas de optimización algebraica diferencial (POAD en sus siglas en español). Estos parámetros solo son el planteamiento del problema debido a que para resolver estos sistemas fácilmente debe de mantenerse un sistema de ecuaciones diferenciales de primer orden. Un método es transformar la integral de la función objetivo en una ecuación no lineal por medio de un Hamiltoniano, esto nos daría que, para cada paso corresponde su propia función objetivo, por consiguiente, tendremos n optimizaciones como n pasos habidos. Esto nos llevaría a un problema de programación no lineal (NLP) debido a que se trabajaría con ecuaciones no lineales. El cambio de una integral a una ecuación no lineal hace que el principio del máximo necesite de variables y ecuaciones diferenciales de primer orden adicionales basadas en variables y ecuaciones adjuntas. Siendo el objetivo de los problemas de control óptimo seleccionar los valores de las variables de decisión que llevan las variables de respuesta a través de una trayectoria óptima en el periodo de integración.

El aumento en los casos del dengue ha dado como resultado miles de casos en todo el mundo, el hombre se ha dado a la tarea de encontrar alguna forma de controlar esta enfermedad, una de las formas de dar solución a ello, es mediante

controles mecánicos (físicos), químicos y por la prevención (profilaxis). Esto le ha dado la idea al hombre de hacer **MODELOS MATEMÁTICOS DE LA DINÁMICA DE LA TRANSMISIÓN DEL DENGUE** con los diferentes parámetros que afectan a esta. Uno de ellos es la variación de la población del vector de transmisión del mosquito, dando cierto criterio promedio anual, donde se tomaba menor a uno, la enfermedad no afectaba a la población humana y en caso contrario lo hacía. Esto resultaba que, si no se invadía a la población humana en cierta temporada, en la siguiente el número de infectados disminuía considerablemente como menciona (Coutinho y col, 2006). Otro de los parámetros tomado para la modelación de un sistema dinámico de la transmisión del mosquito es la propagación de las larvas y de los mosquitos adultos, siendo que las larvas se propagan en un medio acuoso, este se encuentra estático, mientras que los mosquitos se propagan por el viento y la auto difusión (Takahashi y col., 2003). Se han considerado los diferentes serotipos del dengue para la modelación de la dinámica del dengue tomando como interés la invasión y la persistencia de cada serotipo. Dando como resultado el modelo; el estado libre de la enfermedad, el estado donde se presenta un serotipo y donde existen ambos serotipos como puntos de equilibrio en la región biológica de interés (Esteva y col., 2003). La incertidumbre demográfica y la heterogeneidad de transmisión de la enfermedad son otros de los parámetros que se han estudiado en la dinámica de la transmisión del dengue, debido a su gran relación para afectar el sistema de huésped-vector de manera muy significativa, por consiguiente, el aumento o disminución de la enfermedad del dengue (Lloyd y col., 2007). Un caso de aplicación de controles químicos es (Buratani y col., 2007) donde se aplicó insecticida y larvicida en el inicio del brote para minimizar los efectos de la enfermedad, siendo muy efectivas en su aplicación dando resultados en las primeras semanas y tomando en cuenta el mantenimiento del larvicida después de la disminución de la enfermedad. Como se ha visto se han estudiado muchos parámetros para controlar la enfermedad del dengue que han sido efectivos, pero aun así la enfermedad sigue manteniéndose en un nivel de casos considerable en todo el mundo. El modelo propuesto por (Toro-Zapata y col., 2010) escogido en este trabajo combina dos de los controles (mecánico y

profilaxis) más utilizados y dando buenos resultados en la aplicación del modelo teórico. Tomando en cuenta la dinámica de la enfermedad en humanos, de mosquitos y de criaderos.

ANTECEDENTES

En los últimos años ha aumentado considerablemente los casos del dengue en todo el mundo. La gran mayoría de los casos no están notificados y mucho menos clasificados en los centros de salud. Antes de 1970, solo nueve países habían notificado de epidemias de dengue. Ahora en nuestros días más de 100 países han presentado esta enfermedad. Estando países de las diferentes regiones de África, América, el Mediterráneo Oriental, Asia Sudoriental y el Pacífico Occidental. El dengue es una enfermedad que no tienen un tratamiento específico, sino es basado en la experiencia de médicos y enfermeras para su pronta recuperación. El único método para combatir la transmisión del virus del dengue es combatir el vector del mosquito por medio de controles mecánicos, preventivos (profilaxis) y químicos. Por este motivo se han hecho redes de colaboración de laboratorios, capacitación del tratamiento clínico, el diagnóstico, la lucha antivectorial y modelos matemáticos para el control de la transmisión del dengue. El diseñar modelos matemáticos que sean capaces de reducir considerablemente los casos de la enfermedad del dengue se ha vuelto cada vez más necesarios, considerando todas las herramientas posibles para ser más eficientes como es caso del control óptimo.

JUSTIFICACIÓN

Existen diversos modelos matemáticos que predicen la dinámica de transmisión y mecanismos de control de enfermedades transmitidas por vectores, sin embargo, generalmente se plantea el uso de controles constantes lo cual no garantiza la obtención de trayectorias óptimas para las variables de estado y control.

HIPÓTESIS

Aplicar estrategias de control óptimo a problemas dinámicos nos permitirá obtener las trayectorias óptimas para las variables de estado y control.

OBJETIVO PRINCIPAL

El objetivo general de este trabajo es desarrollar estrategias de solución para problemas de control óptimo con aplicación en la dinámica de la transmisión del dengue. Asimismo, determinar el control necesario para minimizar el número promedio de personas infectadas y variables afines a esta.

OBJETIVOS PARTICULARES

1. Deducir las variables y ecuaciones adjuntas
2. Encontrar el método de resolución adecuado para el modelo
3. Probar que el modelo matemático se le pueda aplicar control óptimo
4. Probar si es necesario aplicar el control óptimo al modelo
5. Probar si solo un control es más eficaz con el control óptimo que combinados

DESCRIPCIÓN DEL TRABAJO

En el capítulo 1 se dará la teoría fundamental y el planteamiento a problemas de control óptimo. La necesidad de la aplicación del principio del máximo en estos problemas, así como el método de resolución del gradiente reducido cuando los problemas de valor de frontera (PVF) se hacen presente cuando se aplica el principio del máximo. También en este capítulo se describe el modelo matemático reportado por (Toro-Zapata y col, 2010), se ponen a prueba los controles en diferentes circunstancias, desde no estar presentes hasta actuar conjuntamente. En el capítulo 2 se presenta todas las ecuaciones necesarias para la resolución del problema de control óptimo como la función objetivo (función costo), ecuaciones y variables adjuntas, condiciones finales y ecuaciones de los controles. También se presenta el algoritmo del método del gradiente reducido para este trabajo. En el capítulo 3 se dan especificaciones generales de la función objetivo (función costo) y restricciones de los controles. A continuación, se presentan los resultados gráficos y numéricos de la simulación. Al final, se da una discusión de los resultados. En el capítulo 4 se muestran las conclusiones generales y particulares del trabajo, así como las conclusiones del uso de cada método numérico en el modelo matemático y recomendaciones de la simulación y trabajo.

CAPÍTULO 1 GENERALIDADES

1.1. FUNDAMENTO TEÓRICO

En este subcapítulo se dará la teoría fundamental del control óptimo, así como el principio del máximo en la aplicación del control óptimo y el método de gradiente reducido para la resolución de problemas de frontera (PVF) dados por las condiciones de optimalidad.

1.1.1. PLANTEAMIENTO DE PROBLEMAS DE OPTIMIZACIÓN

En el planteamiento de problemas de control óptimo se deben especificar varios apartados del problema como son:

1. Se debe especificar el modelo dinámico del sistema de estudio
2. Se debe especificar el conjunto de estrategias admisibles (políticas de control)
3. Se debe especificar las restricciones adicionales tanto para las variables de estado y control si es necesario
4. Se debe especificar una función objetivo (función costo) para medir la respuesta del sistema a distintas estrategias admisibles.

Mencionado los cuatro puntos anteriores, se puede entender que, los problemas de control óptimo es optimizar (minimizar o maximizar) una función objetivo por medio de nuestras variables de interés sujetas a las restricciones de nuestros puntos 1, 2 y 3.

1.1.2. DEFINICIÓN DE MODELO MATEMÁTICO

En este trabajo se considera un sistema dinámico a tiempo continuo que está dado por la siguiente ecuación diferencial ordinaria:

$$\frac{dx}{dt} = f[x(t), u(t), t] \quad 1.1.1$$

Sujeta a condiciones a condiciones iniciales

$$x(t_0) = x_0 \quad 1.1.2$$

Para un tiempo dado en

$$0 \leq t \leq T \quad 1.1.3$$

Donde

$$T \leq \infty \quad 1.1.4$$

1.1.3.POLÍTICAS DE CONTROL

Las trayectorias de los controles $\mathbf{u}(t)$ que es continua en $0 \leq t \leq T$ se define como control admisible cuando se satisface:

$$\mathbf{u}(t) \in \Omega(t) \quad 1.1.5$$

Donde $\Omega(t)$ es el conjunto de todos los valores que puede tomar la variable de control en el tiempo t y generalmente está dado por restricciones físicas del problema.

1.1.4.RESTRICCIONES ADICIONALES

Estas restricciones son únicas de cada problema de control óptimo, nos dan los límites superiores e inferiores de nuestras variables del sistema, así como también dan relaciones y diferencias entre las mismas. Estas restricciones pueden ser de igualdad y/o desigualdad como se muestran:

$$g[\mathbf{x}(t), \mathbf{u}(t), t] = 0 \quad 1.1.6$$

$$h[\mathbf{x}(t), \mathbf{u}(t), t] \leq 0 \quad 1.1.7$$

$$k[\mathbf{x}(t), \mathbf{u}(t), t] \geq 0 \quad 1.1.8$$

$$l[\mathbf{x}(t), \mathbf{u}(t), t] \neq 0 \quad 1.1.9$$

Uno ejemplo de este tipo de restricciones es cuando una variable no afecte al sistema esto hace que sea una igualdad a cero, una de las más usadas es la de no negatividad en diferentes variables (masa, tiempo, volumen, etc.) debido a que no es físicamente posible tener estas variables negativas.

1.1.5.FUNCIÓN OBJETIVO

La función objetivo (función costo) en el control óptimo cuenta con dos partes, la primera parte (termino izquierdo de la igualdad) con una función escalar que es evaluada en el estado y tiempo final del sistema, mientras que la segunda parte (termino derecho de la igualdad) es una función escalar integral dependiendo de las variables de estado y control, y está siendo evaluada desde el estado y tiempo inicial hasta el estado y tiempo final sin llegar al último punto.

$$J = \varphi(\mathbf{x}(T), T) + \int_{t_0}^T L(\mathbf{x}(t), \mathbf{u}(t), t) dt \quad 1.1.10$$

Donde $\varphi(\mathbf{x}(T), T)$ es el termino conocido como función costo terminal, mientras que $L(\mathbf{x}(t), \mathbf{u}(t), t)$ es la ecuación Lagrangiana. La función objetivo (función costo) no está limitada a una forma de determinada, sino que, está basada en las variables de interés que afectan nuestro sistema, estas variables darán la forma a nuestra función objetivo (función costo). Cabe mencionar que nuestra función objetivo no está restringida de contener los dos términos en la ecuación. Cuando contiene únicamente el termino $\varphi(\mathbf{x}(T), T)$ la ecuación es conocida como función tipo Mayer:

$$J = \varphi(\mathbf{x}(T), T) \quad 1.1.11$$

Mientras que si cuenta con el término de la integral solamente es conocida como función Lagrangiana:

$$J = \int_{t_0}^T L(x(t), u(t), t) dt \quad 1.1.12$$

Cuando la ecuación contiene ambos términos se le conoce como función tipo Bolza (1.1.10). Una comparación gráfica de todas las ecuaciones se muestra en la figura (1), donde se observa que en el caso de solo contener la función costo termina (Tipo Mayer) el resultado obtenido será el último estado (estado final), en el caso de solo contener la función Lagrangiana se tomó el resultado como el área bajo la curva, mientras que en el caso de Bolza es una combinación de ambos con la diferencia menor de área bajo la curva de la función Lagrangiana.

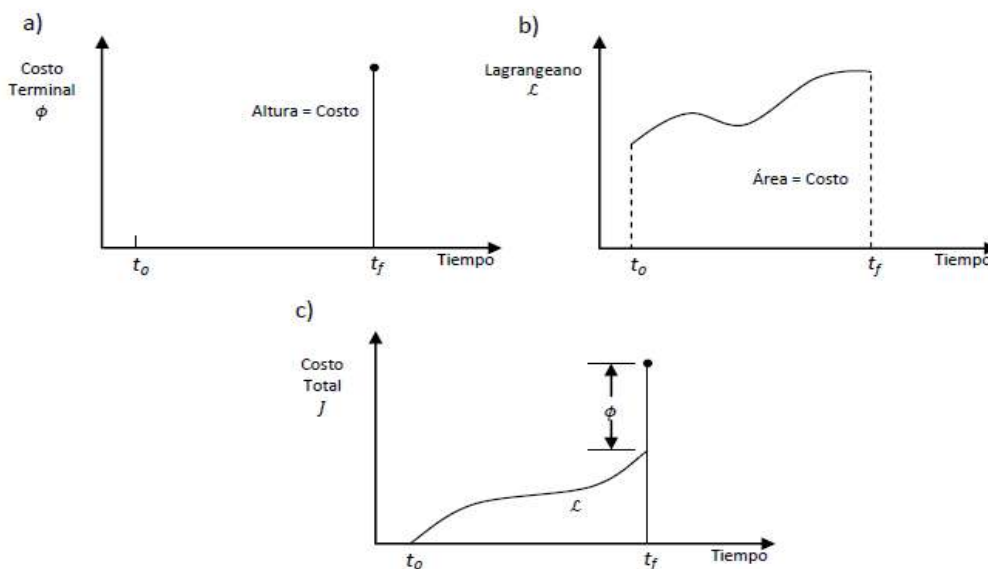


Figura 1 .- Comparación gráfica de las ecuaciones (1.1.10 - 1.1.12). a) Función tipo Mayer, b) Función Lagrangiana y c) función tipo Bolza

Como se observa la función objetivo (función costo) J es dinámica durante el periodo de $0 \leq t \leq T$ cuando se usa la ecuación tipo Bolza, esto nos da como restricciones nuestros controles y variables de interés, ya que la función objetivo (función costo) depende de estas. Esto quiere decir que, cuando se optimiza

controles y variables se presenta una restricción dinámica que se debe satisfacer al optimizar la función objetivo (función costo).

1.1.6.PROBLEMAS DE CONTROL ÓPTIMO

Dadas las definiciones anteriores de optimización se puede formular el problema de optimización de la siguiente forma:

$$\left\{ \begin{array}{l} OPT_{u(t) \in \Omega(t)} \left\{ J = \varphi(\mathbf{x}(T), T) + \int_{t_0}^T L(\mathbf{x}(t), \mathbf{u}(t), t) dt \right\} \\ \text{Sujeto a} \\ \frac{d\mathbf{x}}{dt} = f[\mathbf{x}(t), \mathbf{u}(t), t] \quad \mathbf{x}(t_0) = \mathbf{x}_0 \\ g[\mathbf{x}(t), \mathbf{u}(t), t] = 0 \\ h[\mathbf{x}(t), \mathbf{u}(t), t] \leq 0 \\ k[\mathbf{x}(t), \mathbf{u}(t), t] \geq 0 \\ l[\mathbf{x}(t), \mathbf{u}(t), t] \neq 0 \end{array} \right.$$

Para realizar estos problemas de control óptimo es necesario conocer las condiciones necesarias y suficientes de optimalidad, por este motivo, en los próximos capítulos se hablará de estos parámetros más detalladamente, y sobre todo como se afrontarán las condiciones de frontera por el principio del máximo.

1.1.7.CONDICIONES NECESARIAS Y SUFICIENTES DE OPTIMALIDAD

La teoría de optimización tuvo sus inicios en el cálculo de variaciones, siendo este, una rama de las matemáticas que estudia los valores extremos de las funciones variables continuas. Derivado a esto las condiciones necesarias de optimalidad se obtiene del cálculo de variación. Para que la restricción dinámica de la ecuación (1.1.1) sea considerada en la optimización (maximización y minimización) es necesaria agregarla en la función objetivo (función costo), para no modificar el

valor numérico de la restricción dinámica en el punto óptimo. La restricción dinámica puede ser escrita como:

$$f[\mathbf{x}(t), \mathbf{u}(t), t] - \dot{\mathbf{x}}(t) = 0 \quad \mathbf{x}(t_0) = \mathbf{x}_0$$

Dado que dicha restricción debe ser satisfecha en el intervalo $0 \leq t \leq T$, debe ser considerada en la parte integral de la función objetivo (función costo):

$$J = \varphi(\mathbf{x}(T), T) + \int_{t_0}^T [L(\mathbf{x}(t), \mathbf{u}(t), t) + \boldsymbol{\lambda}^T(t)[f[\mathbf{x}(t), \mathbf{u}(t), t] - \dot{\mathbf{x}}(t)]] dt \quad 1.1.13$$

Si ahora definimos el Hamiltoniano (\mathcal{H}) como:

$$\mathcal{H}[\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\lambda}(t), t] = L(\mathbf{x}(t), \mathbf{u}(t), t) + \boldsymbol{\lambda}^T(t)[f[\mathbf{x}(t), \mathbf{u}(t), t]] \quad 1.1.14$$

Entonces la función objetivo (función costo) puede ser escrita como:

$$J = \varphi(\mathbf{x}(T), T) + \int_{t_0}^T [\mathcal{H}[\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\lambda}(t), t] - \boldsymbol{\lambda}^T(t)[\dot{\mathbf{x}}(t)]] dt \quad 1.1.15$$

La segunda integral puede ser resuelta por partes dando:

$$\int_{t_0}^T [\boldsymbol{\lambda}^T(t)[\dot{\mathbf{x}}(t)]] dt = \boldsymbol{\lambda}^T(t)[\mathbf{x}(t)]|_{t_0}^T - \int_{t_0}^T [\dot{\boldsymbol{\lambda}}^T(t)[\mathbf{x}(t)]] dt$$

Con lo que la función objetivo se convierte en:

$$J = \varphi(\mathbf{x}(T), T) + \boldsymbol{\lambda}^T(t_0)[\mathbf{x}(t_0)] - \boldsymbol{\lambda}^T(T)[\mathbf{x}(T)] + \int_{t_0}^T [\mathcal{H}[\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\lambda}(t), t] - \dot{\boldsymbol{\lambda}}^T(t)[\mathbf{x}(t)]] dt \quad 1.1.16$$

Como se había mencionado anteriormente es necesario que la función objetivo (función costo) sea de primer orden para poder encontrar un punto estacionario cuando el control sea cero en el intervalo de tiempo. Debido a que el control tiene una gran influencia en las variables de estado, cualquier alteración en él afectará durante o después de la perturbación a estas. Para las primeras variaciones de los componentes de la función objetivo (función costo) puede expresarse como:

$$\Delta(\cdot) = \frac{\partial(\cdot)}{\partial \mathbf{u}} \Delta \mathbf{u} + \frac{\partial(\cdot)}{\partial \mathbf{x}} \Delta \mathbf{x}(\Delta \mathbf{u}) \quad 1.1.17$$

Donde $\Delta \mathbf{x}(\Delta \mathbf{u})$ representa las perturbaciones en las variables de estado generadas por un cambio en las variables de control durante o antes del tiempo de interés. Considerando que se sabe el tiempo final, la primera variación en J debido al cambio en el control es:

$$\begin{aligned} \Delta J &= \left\{ \left[\frac{\partial \varphi}{\partial \mathbf{x}} - \boldsymbol{\lambda}^T \right] \Delta \mathbf{x}(\Delta \mathbf{u}) \right\} \Big|_{t=T} + [\boldsymbol{\lambda}^T \Delta \mathbf{x}(\Delta \mathbf{u})] \Big|_{t=t_0} \\ &+ \int_{t_0}^T \left\{ \frac{\partial \mathcal{H}}{\partial \mathbf{u}} \Delta \mathbf{u} + \left[\frac{\partial \mathcal{H}}{\partial \mathbf{x}} + \dot{\boldsymbol{\lambda}}^T \right] \Delta \mathbf{x}(\Delta \mathbf{u}) \right\} dt \\ &\triangleq \Delta J(T) + \Delta J(t_0) + \Delta J(t_0, T) \end{aligned} \quad 1.1.18$$

Donde $\Delta \mathbf{u}(t)$ en el intervalo $0 \leq t \leq T$ es una función diferente de cero, en el caso de la trayectoria óptima de ΔJ debe ser cero esto quiere decir, que las tres trayectorias deben ser cero. Esto nos lleva que para que la primera parte de $\Delta J(t_0)$ al ser igual a cero se debe considerar que el control no debe afectar al estado inicial. Para otras dos partes de ΔJ cumplan la condición de igualdad a cero, $\boldsymbol{\lambda}(t)$ debe satisfacer las siguientes ecuaciones:

$$\dot{\boldsymbol{\lambda}}^T(t) = \frac{-\partial \mathcal{H}[\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\lambda}(t), t]}{\partial \mathbf{x}} \quad 1.1.19$$

Sujetas a las condiciones finales

$$\lambda^T(T) = \left. \frac{\varphi(\mathbf{x}(t), t)}{\partial \mathbf{x}} \right|_T \quad 1.1.20$$

Y

$$\frac{\partial \mathcal{H}[\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\lambda}(t), t]}{\partial \mathbf{u}} = 0 \quad 1.1.21$$

Las ecuaciones (1.1.19 – 1.1.21) son las tres ecuaciones de optimalidad cuando se conoce el tiempo final. También son conocidas como las ecuaciones de Euler-Lagrange. Estas condiciones deben ser aplicadas a la trayectoria de nuestra función objetivo siendo estas, criterios de optimalidad locales. Cuando se aplicas estos criterios de optimalidad se obtiene un problema con valor en la frontera (PVF), ya que en la ecuación (1.1.20) es necesario saber el estado final en el tiempo (T), por ello, es necesario aplicar un método que dé solución a estas condiciones de frontera, en este trabajo se aplicará el método del gradiente reducido, mientras que para las condiciones iniciales se tienen en el tiempo (t_0). Para terminar este apartado es necesario saber que la condición suficiente de optimalidad está dada por la siguiente ecuación:

$$\frac{\partial^2 \mathcal{H}}{\partial \mathbf{u}^2} [\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t] = \mathcal{H}_{uu} \leq 0 \quad 1.1.22$$

Conocida como la condición de Legendre-Clebsch (o condición de convexidad) esta nos dará solución a las ecuaciones de Euler-Lagrange.

1.1.8. PRINCIPIO DEL MÁXIMO

En términos generales el principio del máximo, también llamado principio de Pontryagin consiste en usar el Hamiltoniano para optimizar la función objetivo, con base a las restricciones dinámicas y encontrar los controles óptimos que satisfagan las restricciones de optimización en el tiempo $[t_0, T]$. Como se mencionó las ecuaciones (1.1.19 – 1.1.21) constituyen un criterio local de optimización dependiente de una trayectoria particular para $\mathbf{x}^*(t)$ y $\mathbf{u}^*(t)$ en el intervalo $[t_0, T]$, ahora si se condiremos $\mathbf{x}^*(t)$ y $\mathbf{u}^*(t)$ como valores fijos en cada tiempo, se cumplirá la siguiente condición del Hamiltoniano:

$$\mathcal{H}^* = \mathcal{H}[\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t] \geq \mathcal{H}[\mathbf{x}^*(t), \mathbf{u}(t), \boldsymbol{\lambda}^*(t), t] \quad 1.1.23$$

Donde $\mathbf{u}(t)$ es cualquier valor admisible del control, pero no es el valor óptimo. Mientras que las ecuaciones (1.1.19, 1.1.21, 1.1.22 y 1.1.23) son las condiciones necesarias y suficientes de carácter global y constituyen el principio del máximo siempre y cuando el Hamiltoniano sea convexo. Las ecuaciones (1.1.22 y 1.1.23) son usadas para el caso de que la optimización sea una maximización, en el caso contrario es necesario multiplicar por (-1) para obtener las ecuaciones de minimización como se muestras a continuación:

$$\frac{\partial^2 \mathcal{H}}{\partial \mathbf{u}^2} [\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t] = \mathcal{H}_{uu} \geq 0 \quad 1.1.24$$

$$\mathcal{H}^* = \mathcal{H}[\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t] \leq \mathcal{H}[\mathbf{x}^*(t), \mathbf{u}(t), \boldsymbol{\lambda}^*(t), t] \quad 1.1.25$$

A esta variación del principio del máximo se le conoce como el principio del mínimo, debió a que inicialmente el principio del máximo se usó para una maximización.

1.1.9.MÉTODO DEL GRADIENTE REDUCIDO

Como se mencionó anteriormente en el apartado de condiciones necesarias y suficientes de optimalidad es necesario aplicar el método del gradiente reducido debido a que al aplicar las condiciones de optimalidad nos da un problema de valor en la frontera, para esto se dará los pasos del gradiente reducido (Napoles-Rivera F. 2008):

1. Especificar el estado inicial $x(t_0)$ y suponer un perfil inicial para la variable de control u_0
2. Resolver la ecuación (1.1.1) de $[t_0, T]$
3. Calcular las condiciones finales $\lambda(T)$ con la ecuación (1.1.20)
4. Resolver la ecuación (1.1.19) de $[T, t_0]$
5. Calcular $u_{k+1} = u_k - \varepsilon_k \left[\frac{\partial \mathcal{H}}{\partial u} \right]_k$
6. Si $u_{k+1} - u_k < \text{tolerancia}$ regresar a 2. Si no, finalizar

Donde ε_k es una función de peso de gran importancia, en valores muy grandes de ε_k puede provocar que nuestra diferencia de $(u_{k+1} - u_k < \text{tolerancia})$ no converja y/o marque errores en las iteraciones. En caso contrario, llevaría a gran cantidad de cálculos y más aún si nuestro tamaño de paso es muy pequeño, esto se traduciría, a un tiempo muy grande de convergencia. Como se observa la condición necesaria de optimalidad es revisada cada vez que se actualiza el control y cuando se cumpla el criterio de convergencia, esta condición también es revisada con el mismo margen de error, y por último la condición suficiente se revisa por separado.

1.2. MODELO DE LA DINÁMICA DE TRANSMISIÓN DEL DENGUE CLÁSICO

En este apartado se describe el modelo matemático reportado por Hernán D. Toro-Zapata, Leonardo D. Restrepo, Juan G. Vergaño-Salazar y Aníbal Muñoz-Loaiza (2010), dando a conocer las condiciones iniciales presentadas por el modelo, así como las soluciones dadas en las diferentes condiciones presentadas en el mismo. Con el fin de considerar el uso de medidas preventivas y de control mecánico en la reducción de la transmisión de la enfermedad del dengue clásico.

1.2.1. MODELO DEL DENGUE CLÁSICO

El sistema dinámico siguiente fue presentado por (Toro-Zapata y col., 2010), se modela la dinámica en una región endémica considerando medidas preventivas y de control mecánico. Se plantea un sistema de seis ecuaciones diferenciales ordinarias y mediante simulación numérica se determina su evolución en el tiempo. Teniendo como variables de estado, lo siguiente:

Tabla 1 .- Significado y valores asignados a las variables

Variables		Valor
X_1	Número promedio de personas susceptibles en un tiempo t	999
X_2	Número promedio de personas infectadas en un tiempo t	1
X_3	Número promedio de mosquitos portadores del virus en un tiempo t	1
X_4	Población total de mosquitos en un tiempo t	9999
X_5	Número promedio de mosquitos inmaduros en un tiempo t	1
X_6	Número promedio de criaderos en un tiempo t	1

Y como medio de control del sistema dinámico, lo siguiente:

Tabla 2 .- Significado y valores asignados a las variables

Controles		Valor
u_1	Profilaxis	0 a 1.0
u_2	Control Mecánico	0 a 0.7

Las restricciones dinámicas del sistema consisten en seis ecuaciones diferenciales ordinarias:

$$\frac{dX_1}{dt} = \mu N - \beta(1 - u_1)X_1 \frac{X_3}{X_4} - \mu X_1 \quad 1.2.1$$

$$\frac{dX_2}{dt} = \beta(1 - u_1)X_1 \frac{X_3}{X_4} - (\mu + \theta)X_2 \quad 1.2.2$$

$$\frac{dX_3}{dt} = \frac{\lambda}{N} X_2 (X_4 - X_3) - \delta X_3 \quad 1.2.3$$

$$\frac{dX_4}{dt} = \omega X_5 - \delta X_4 \quad 1.2.4$$

$$\frac{dX_5}{dt} = \varphi X_4 \left(1 - \frac{X_5}{\gamma X_6}\right) - (\omega + \varepsilon) X_5 \quad 1.2.5$$

$$\frac{dX_6}{dt} = \nu X_6 \left(1 - \frac{X_6}{K}\right) - u_2 X_6 \quad 1.2.6$$

Las primeras dos ecuaciones (1.2.1 a 1.2.2) son responsables de la dinámica de la enfermedad en los humanos, mientras que las ecuaciones (1.2.3 a 1.2.5) describen la dinámica del mosquito y la ecuación (1.2.6) describe la dinámica de los criaderos.

Tanto las variables de estado y los controles están sujetos a las restricciones de desigualdad

$$X_i \geq 0 \quad 1.2.7$$

$$u_i \geq 0 \quad 1.2.8$$

Debido a que una minoría de las personas no hace uso de medidas preventivas para evitar la transmisión de la enfermedad, ello implica que, el control u_1 tenga la restricción:

$$u_1 < 1 \quad 1.2.9$$

Al utilizar un control mecánico superior a 0.7, implica la extinción de la población de mosquitos, lo cual no es alcanzable, por ello, el control u_2 tiene la restricción:

$$u_2 < 0.7 \quad 1.2.10$$

Mientras que el sistema dinámico presenta varios parámetros, estos a su vez, tienen restricciones como son las siguientes:

$$\theta, \gamma, \delta, \varepsilon, \omega, \mu, \varphi, k > 0 \quad 1.2.11$$

$$\beta, \lambda \in (0,1) \quad 1.2.12$$

$$v \geq 0 \quad 1.2.13$$

Los valores de los parámetros son los mostrados en la tabla (3) siguiente:

Tabla 3 .- Significado y valores asignados a los parámetros

Parámetros		Valor
N	Población total humana	1000
ω	Tasa de maduración del mosquito al estado adulto	0.9
δ	Tasa de mortalidad de los mosquitos maduros	0.25
φ	Tasa de ovoposición de los mosquitos	0.9
γ	Proporción de mosquitos por criaderos	5
k	Número máximo de criaderos que soporta el medio	10000
ε	Tasa de mortalidad de los estados inmaduros	0.1
μ	Tasa de muerte/natalidad humana	0.002
β	Probabilidad de transmisión persona portador/persona susceptible	0.3
θ	Tasa de recuperación de las personas infectadas	0.14
λ	Probabilidad de transmisión persona infectada/vector no portador	0.5
v	Tasa de creación de criaderos	0.7

Se considera que N es constante, se verificó en las ecuaciones diferenciales del sistema que corresponden al modelo de dinámica de transmisión del dengue clásico con control mecánico y profilaxis.

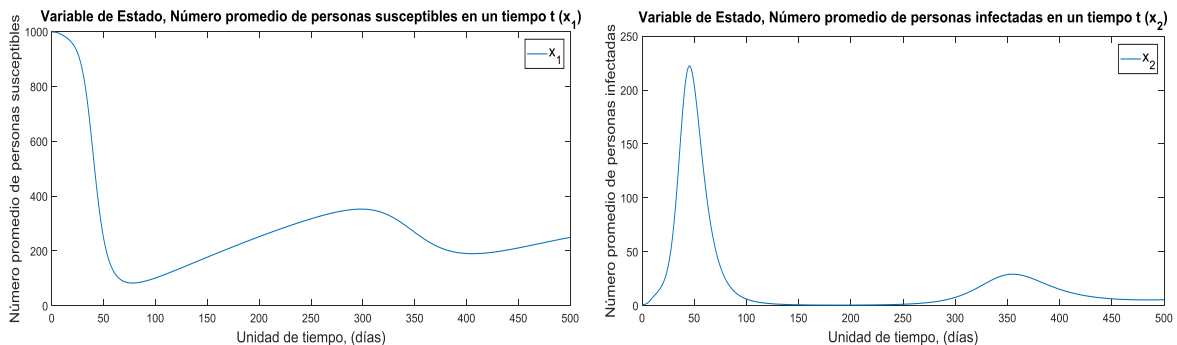
1.2.2. APLICACIÓN DE ESCENARIOS DEL MODELO

Para probar el modelo matemático se proponen cuatro escenarios:

- Caso sin control: No se tomarán medidas de profilaxis, ni control mecánico ($u_1, u_2 = 0$)
- Caso profilaxis: Solo se tomarán medidas de profilaxis, descartando toda medida de control mecánico ($u_1 = 0.3, 0.6, 0.9$)
- Caso control mecánico: Solo se tomarán medidas de control mecánico, descartando toda medida de profilaxis ($u_2 = 0.2, 0.4, 0.65$)
- Caso combinación de controles: Se hará uso de los 2 (dos) medidas, control mecánico y profilaxis ($u_1 = 0.6$ y $u_2 = 0.65$)

1.2.2.1. CASO SIN CONTROL

En el caso del escenario sin control se muestra en la figura (2) se observa que en el caso de las variables población total de mosquitos, número promedio de mosquitos inmaduros y número promedio de criaderos tienden a tener un crecimiento exponencial hasta llegar a un límite y estabilizarse. Mientras que para las variables número promedio de personas infectadas y número promedio de mosquitos portadores del virus tienen un punto máximo y empiezan a descender a un punto donde oscilación levemente hasta el fin del periodo. Por último, la variable de número promedio de personas susceptibles baja hasta un mínimo y comienza a oscilar entre valores casi diez veces menores al valor inicial.



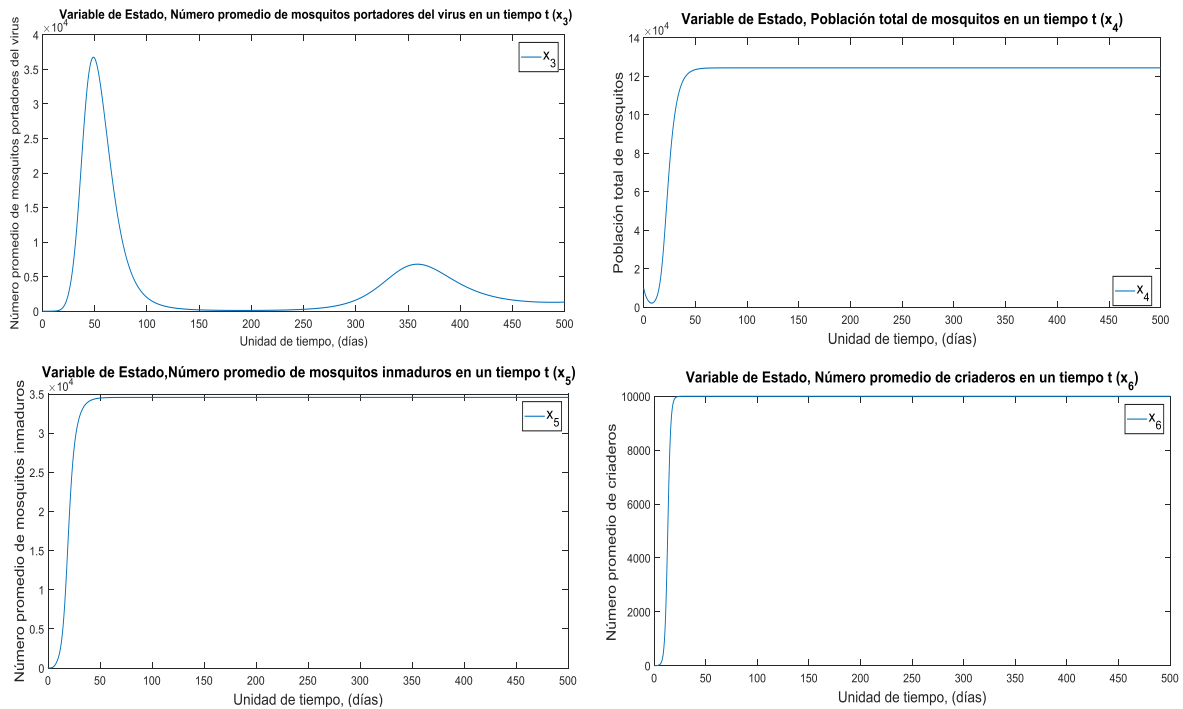


Figura 2 .- Gráficas de las variables de estado en el caso sin control

1.2.2.2. CASO PROFILAXIS

Observe en caso profilaxis en las figuras (3 – 5) en las variables de población total de mosquitos, número promedio de mosquitos inmaduros y número promedio de criaderos, la profilaxis no tiene ningún efecto sobre ellas, pero en el caso de las variables número promedio de personas infectadas y número promedio de mosquitos portadores del virus tiene un efecto positivo debido a la reducción de las mismas con forme el control es más grande se llega a un nivel muy bajo de estas variables y para el caso de la variable número promedio de personas susceptibles empieza a aumentar considerablemente hasta llegar a un punto en que todas las personas son susceptibles.

Donde $u_1 = 0.3$

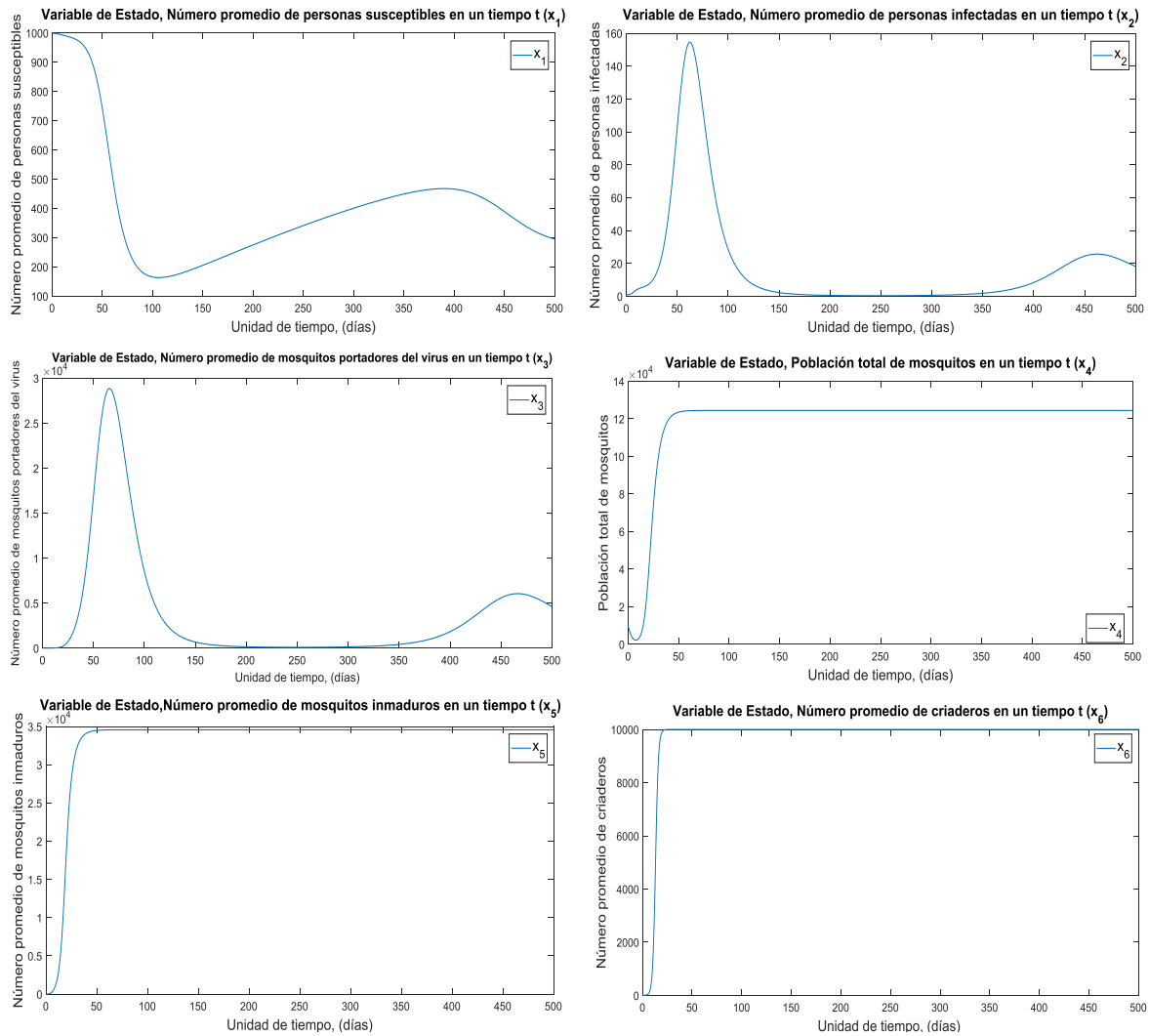
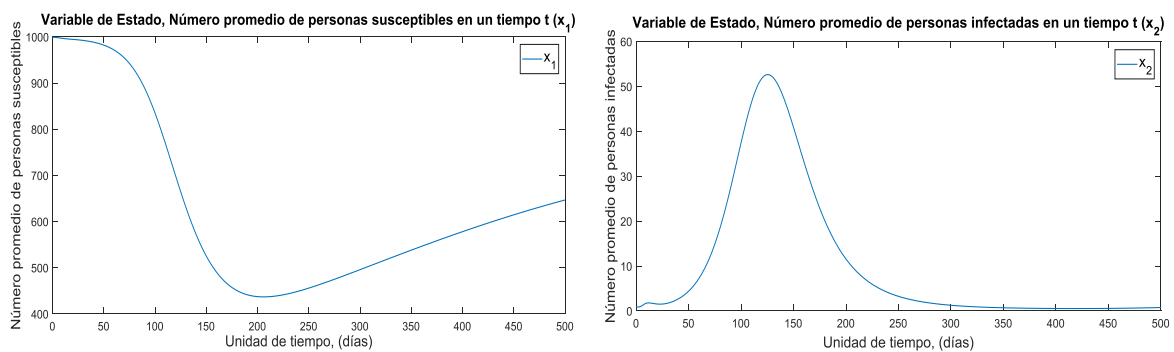


Figura 3 .- Gráficas de las variables de estado en el caso profilaxis con $u_1 = 0.3$

Donde $u_1 = 0.6$



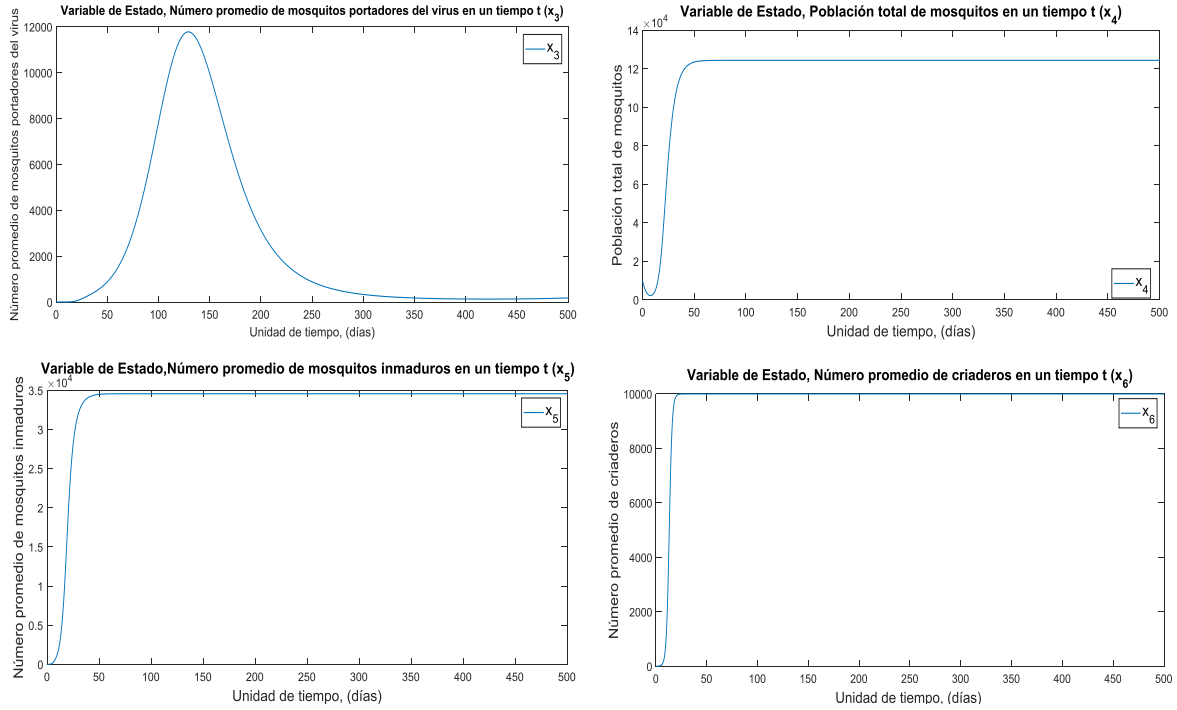
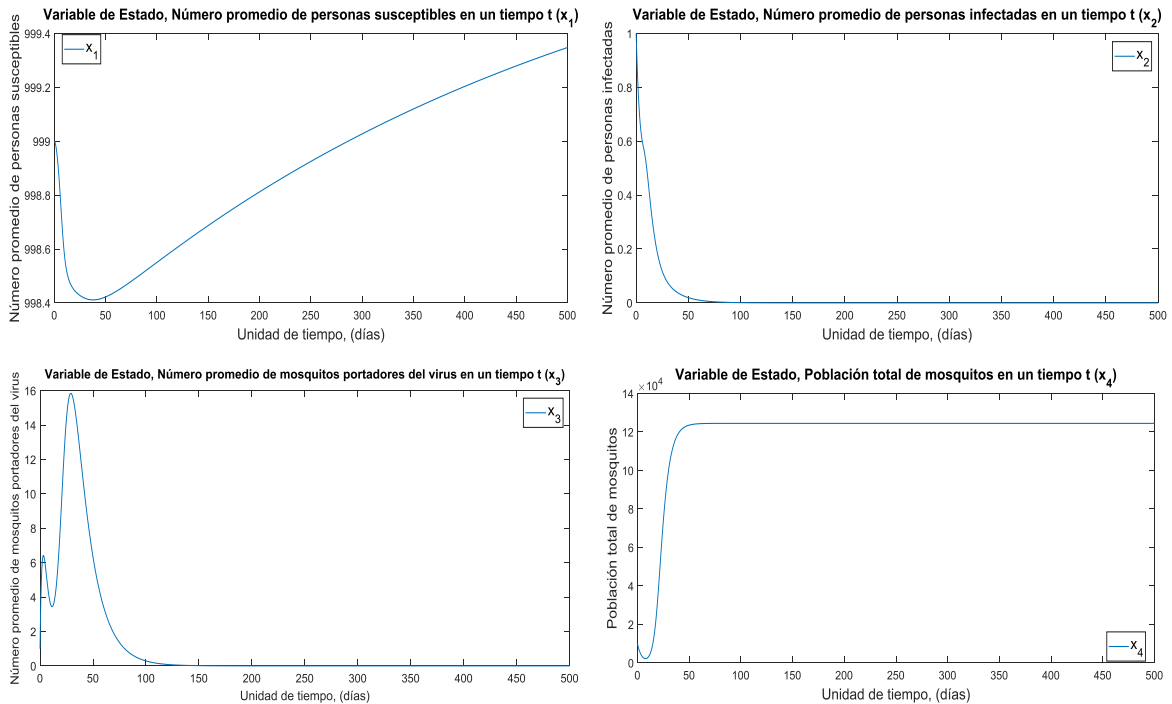


Figura 4 .- Gráficas de las variables de estado en el caso profilaxis con $u_1 = 0.6$

Donde $u_1 = 0.9$



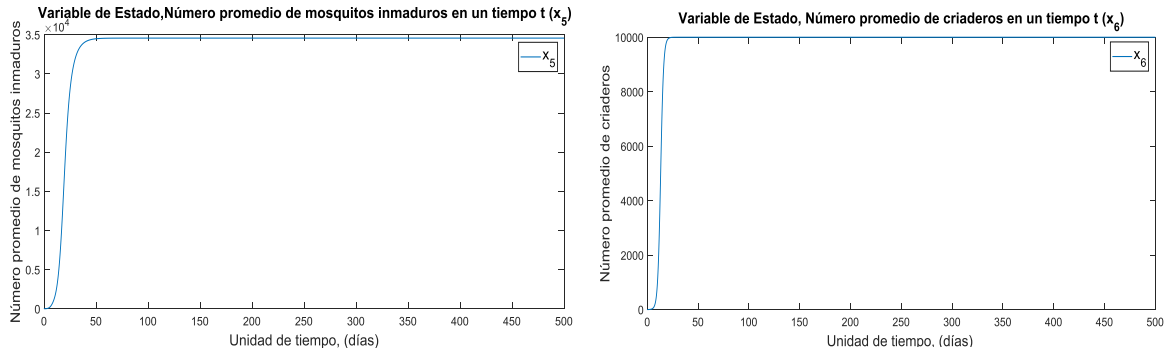
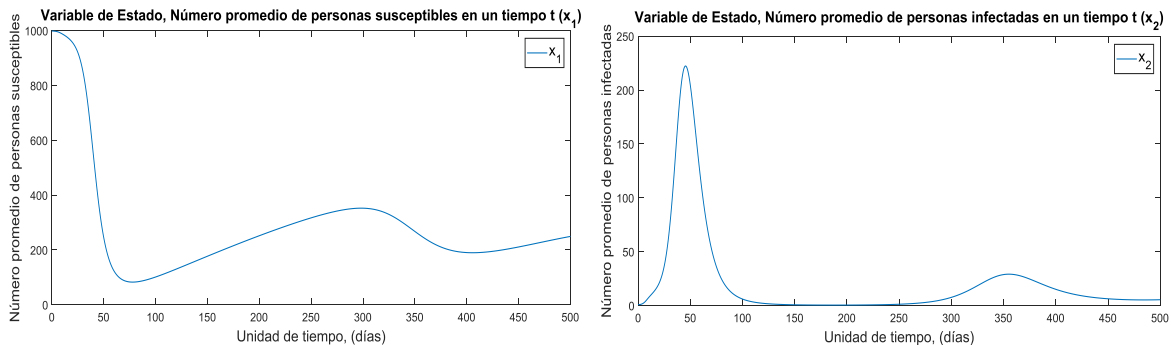


Figura 5 .- Gráficas de las variables de estado en el caso profilaxis con $u_1 = 0.9$

1.2.2.3. CASO CONTROL MECÁNICO

Para el caso de control mecánico se ven en las figuras (6 – 8) las variables número promedio de personas susceptibles y número promedio de personas infectadas no son afectadas por el control mecánico, por el caso contrario con las variables población total de mosquitos, número promedio de mosquitos inmaduros y número promedio de criaderos que tiene un gran impacto de estar en cifras de hasta cinco a pasar a tres dígitos. Mientras que la variable restante de tener un máximo va disminuyéndolo y aumentando otro al final del periodo de estudio, pero no es tan significativo como el máximo sin control.

Donde $u_2 = 0.2$



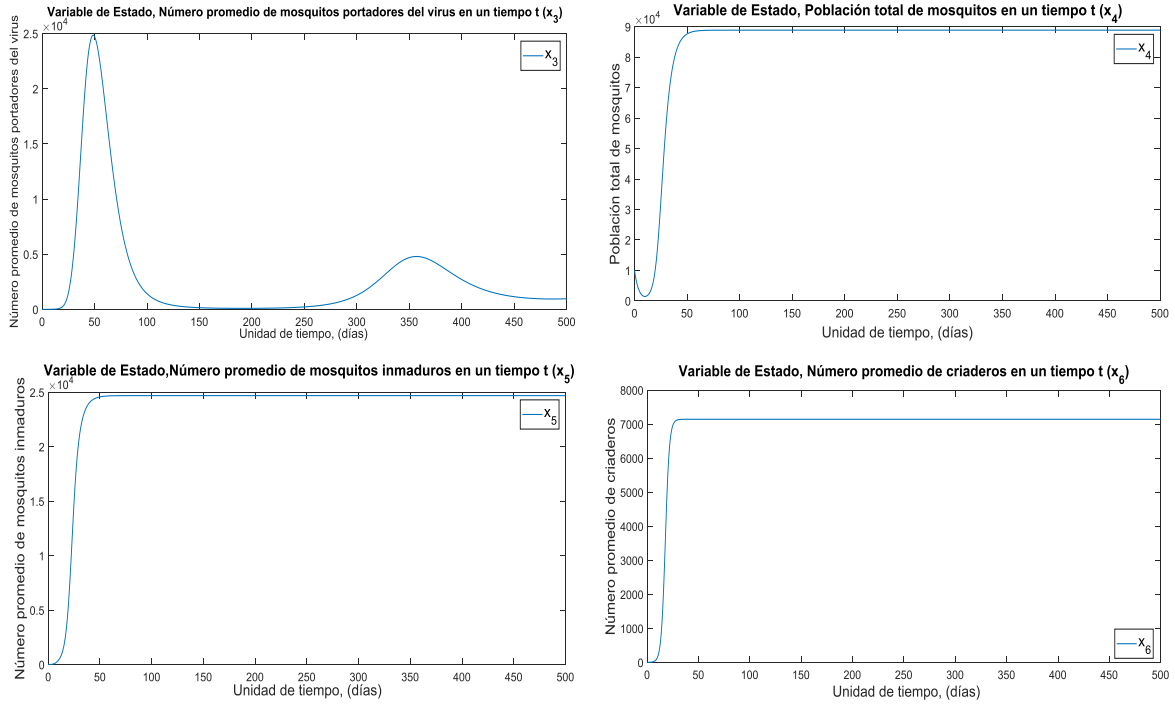
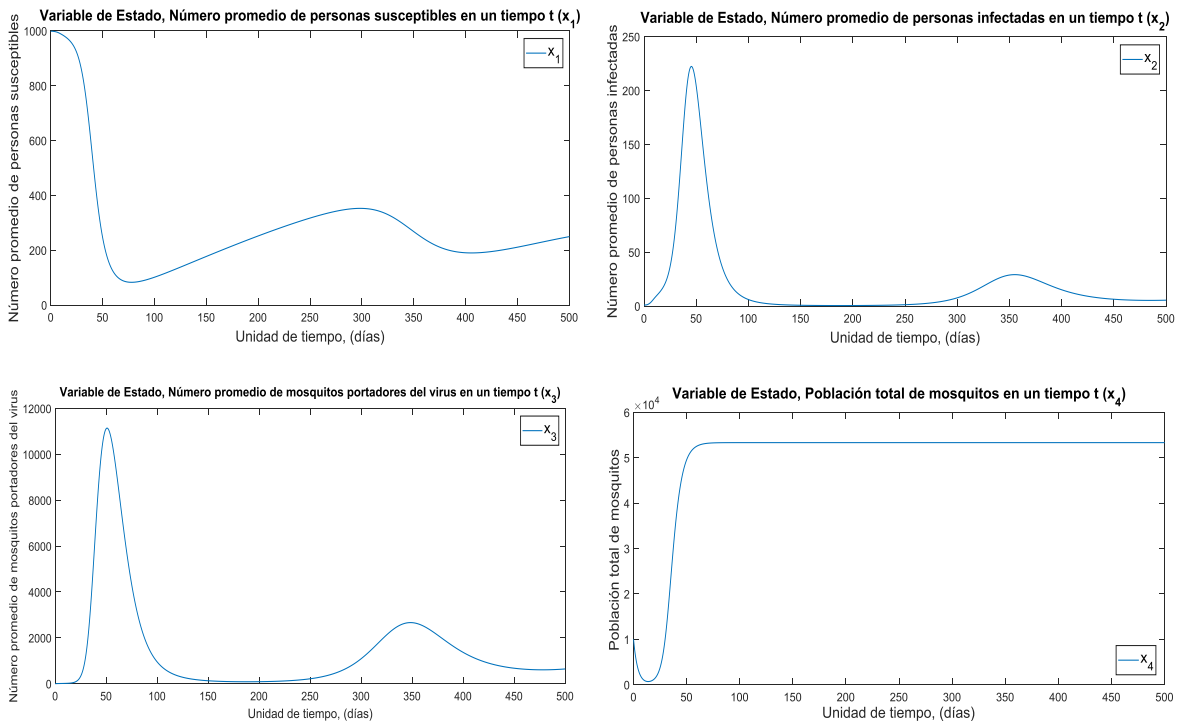


Figura 6 .- Gráficas de las variables de estado en el caso control mecánico con $u_2 = 0.2$

Donde $u_2 = 0.4$



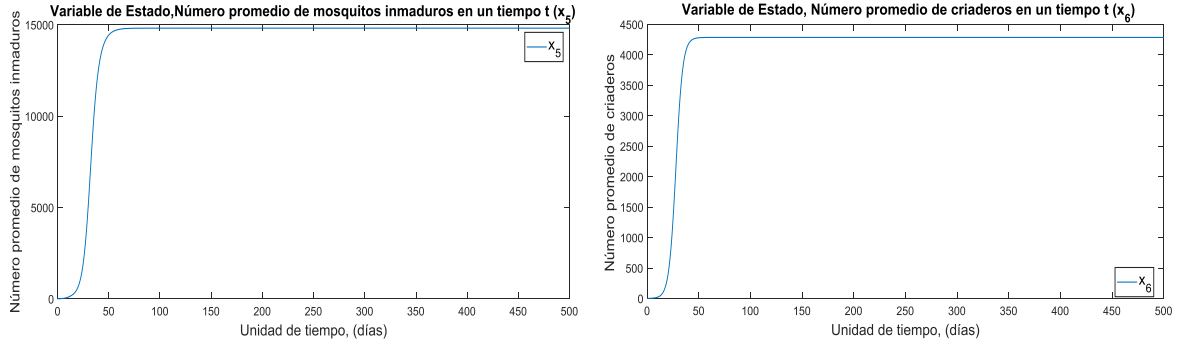


Figura 7 .- Gráficas de las variables de estado en el caso control mecánico con $u_2 = 0.4$

Donde $u_2 = 0.65$

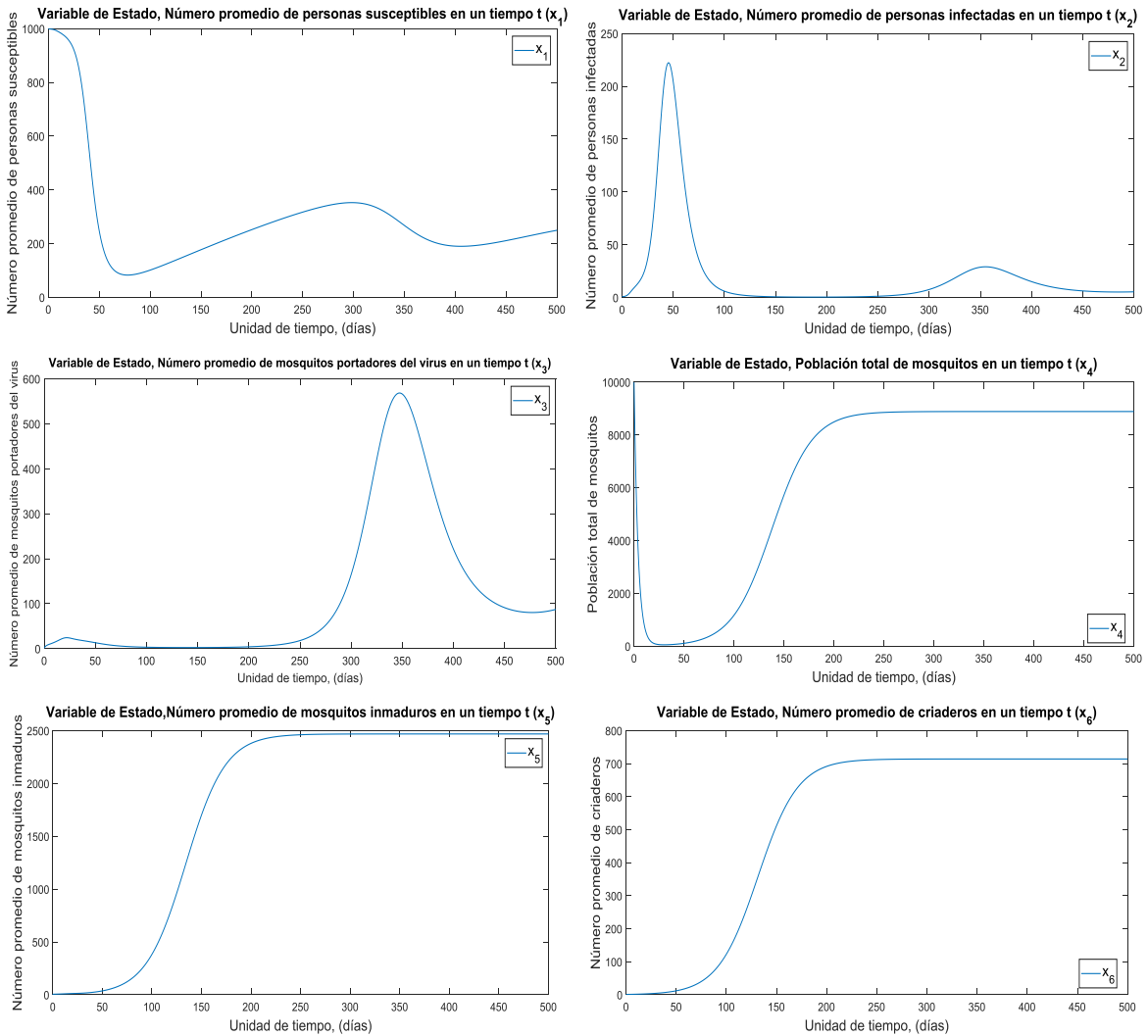
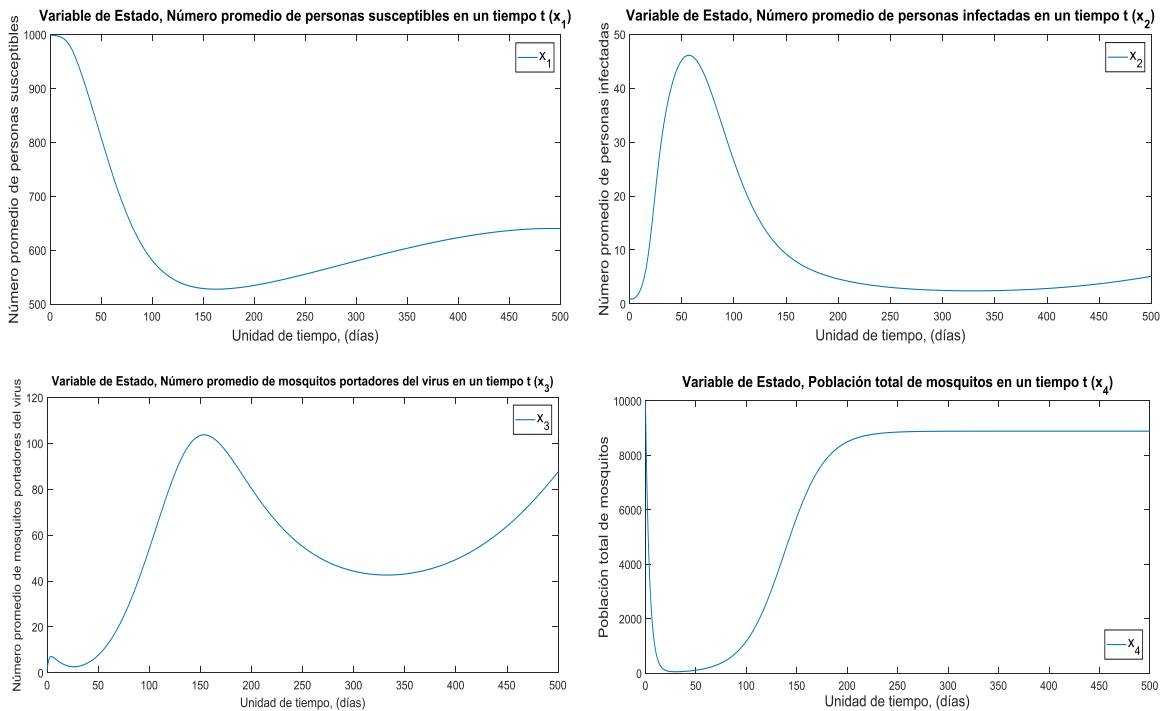


Figura 8 .- Gráficas de las variables de estado en el caso control mecánico con $u_2 = 0.65$

1.2.2.4. CASO COMBINACIÓN DE CONTROLES

Para el caso de combinación de controles observe la figura (9) con la variable número promedio de personas susceptibles los cambios con el control u_1 al comparación del control combinado son nulos, para la variable número promedio de personas infectadas reduce en 5 personas el máximo de personas infectadas pero aumenta el número de personas infectadas al final del tiempo de estudio, se observa que en la primera mitad domina el control u_1 mientras que en la ultima mitad el u_2 , en el caso de número promedio de mosquitos portadores del virus reduce el control combinado entre 400 mosquitos portadores respecto al u_2 en el punto máximo pero al final del tiempo terminan con los mismos mosquitos portadores, en población total de mosquitos, número promedio de mosquitos inmaduros y número promedio de criaderos se mantienen igual que en el control combinado en comparación con el control de u_2 sin aplicación de la profilaxis.



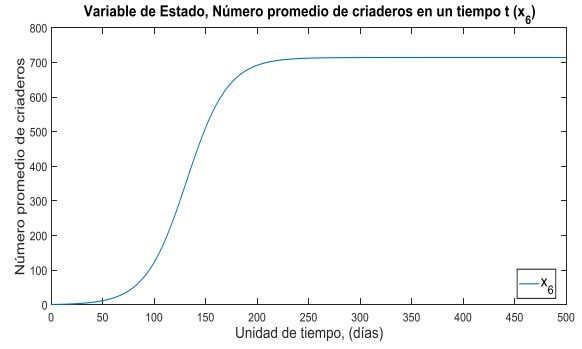
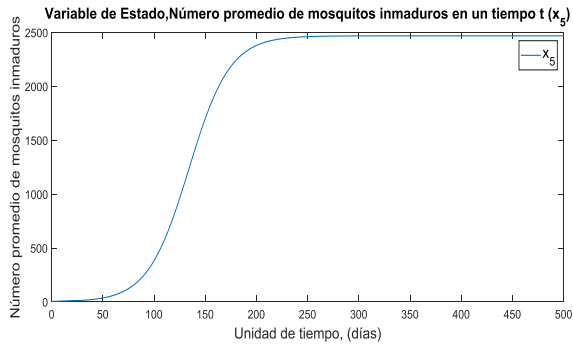


Figura 9 .- Gráficas de las variables de estado en el caso combinación de controles con $u_1 = 0.6$ y $u_2 = 0.65$

CAPÍTULO 2 DESARROLLO EXPERIMENTAL

RESOLUCIÓN DEL MODELO

En este capítulo se dará a conocer la función objetivo (función costo) tipo Bolza del modelo de este trabajo, así como, las ecuaciones adjuntas, condiciones finales, ecuaciones de controles y algoritmo de solución.

2.1. FUNCIÓN OBJETIVO

Se plantea minimizar la función objetivo (función costo) J , con las variables de interés, siendo estas, número promedio de personas infectadas (x_2), número promedio de mosquitos portadores (x_3), población total de mosquitos (x_4) y número promedio de criaderos (x_6), en el caso de los controles se tiene, la profilaxis (u_1) y el control mecánico (u_2). Planteando la función objetivo (función costo) J tipo Bolza:

$$J = (p_{22}X_{2,f}^2 + p_{33}X_{3,f}^2 + p_{44}X_{4,f}^2 + p_{66}X_{6,f}^2) + \int_{t_0}^T [q_{22}X_2^2 + q_{33}X_3^2 + q_{44}X_4^2 + q_{66}X_6^2 + r_{11}u_1^2 + r_{22}u_2^2]dt \quad 2.1$$

Donde los coeficientes p_{ii} , q_{ii} y r_{ii} representan el peso de cada variable en la función. Para el caso en que los dos controles actúan individualmente se tiene

$$J = (p_{22}X_{2,f}^2 + p_{33}X_{3,f}^2 + p_{44}X_{4,f}^2 + p_{66}X_{6,f}^2) + \int_{t_0}^T [q_{22}X_2^2 + q_{33}X_3^2 + q_{44}X_4^2 + q_{66}X_6^2 + r_{11}u_1^2]dt \quad 2.2$$

$$J = (p_{22}X_{2,f}^2 + p_{33}X_{3,f}^2 + p_{44}X_{4,f}^2 + p_{66}X_{6,f}^2) + \int_{t_0}^T [q_{22}X_2^2 + q_{33}X_3^2 + q_{44}X_4^2 + q_{66}X_6^2 + r_{22}u_2^2]dt \quad 2.3$$

2.2. SOLUCIÓN DEL MODELO

Se resolverá el modelo con tres casos, el primer caso, ambos controles actúan mutuamente, el segundo caso, el control u_1 actuará individualmente, y por último el tercer caso, donde solo estará activo el control u_2 . Para la solución de estos tres

casos es necesario de las variables adjuntas en la función objetivo (función costo), de las ecuaciones (2.1), (2.2) y (2.3) se tiene:

$$\begin{aligned}
J = & (p_{22}X_{2,f}^2 + p_{33}X_{3,f}^2 + p_{44}X_{4,f}^2 + p_{66}X_{6,f}^2) \\
& + \int_{t_0}^T \left[q_{22}X_2^2 + q_{33}X_3^2 + q_{44}X_4^2 + q_{66}X_6^2 + r_{11}u_1^2 + r_{22}u_2^2 \right. \\
& + \lambda_1 \left[\mu N - \beta(1 - u_1)X_1 \frac{X_3}{X_4} - \mu X_1 \right] \\
& + \lambda_2 \left[\beta(1 - u_1)X_1 \frac{X_3}{X_4} - (\mu + \theta)X_2 \right] \\
& + \lambda_3 \left[\frac{\lambda}{N} X_2(X_4 - X_3) - \delta X_3 \right] + \lambda_4 [\omega X_5 - \delta X_4] \\
& + \lambda_5 \left[\varphi X_4 \left(1 - \frac{X_5}{\gamma X_6} \right) - (\omega + \varepsilon)X_5 \right] \\
& \left. + \lambda_6 \left[v X_6 \left(1 - \frac{X_6}{K} \right) - u_2 X_6 \right] \right] dt
\end{aligned} \tag{2.4}$$

$$\begin{aligned}
J = & (p_{22}X_{2,f}^2 + p_{33}X_{3,f}^2 + p_{44}X_{4,f}^2 + p_{66}X_{6,f}^2) \\
& + \int_{t_0}^T \left[q_{22}X_2^2 + q_{33}X_3^2 + q_{44}X_4^2 + q_{66}X_6^2 + r_{11}u_1^2 \right. \\
& + \lambda_1 \left[\mu N - \beta(1 - u_1)X_1 \frac{X_3}{X_4} - \mu X_1 \right] \\
& + \lambda_2 \left[\beta(1 - u_1)X_1 \frac{X_3}{X_4} - (\mu + \theta)X_2 \right] \\
& + \lambda_3 \left[\frac{\lambda}{N} X_2(X_4 - X_3) - \delta X_3 \right] + \lambda_4 [\omega X_5 - \delta X_4] \\
& + \lambda_5 \left[\varphi X_4 \left(1 - \frac{X_5}{\gamma X_6} \right) - (\omega + \varepsilon)X_5 \right] \\
& \left. + \lambda_6 \left[v X_6 \left(1 - \frac{X_6}{K} \right) - u_2 X_6 \right] \right] dt
\end{aligned} \tag{2.5}$$

$$\begin{aligned}
J = & (p_{22}X_{2,f}^2 + p_{33}X_{3,f}^2 + p_{44}X_{4,f}^2 + p_{66}X_{6,f}^2) \\
& + \int_{t_0}^T \left[q_{22}X_2^2 + q_{33}X_3^2 + q_{44}X_4^2 + q_{66}X_6^2 + r_{22}u_2^2 \right. \\
& + \lambda_1 \left[\mu N - \beta(1 - u_1)X_1 \frac{X_3}{X_4} - \mu X_1 \right] \\
& + \lambda_2 \left[\beta(1 - u_1)X_1 \frac{X_3}{X_4} - (\mu + \theta)X_2 \right] \\
& + \lambda_3 \left[\frac{\lambda}{N} X_2(X_4 - X_3) - \delta X_3 \right] + \lambda_4 [\omega X_5 - \delta X_4] \\
& + \lambda_5 \left[\varphi X_4 \left(1 - \frac{X_5}{\gamma X_6} \right) - (\omega + \varepsilon)X_5 \right] \\
& \left. + \lambda_6 \left[v X_6 \left(1 - \frac{X_6}{K} \right) - u_2 X_6 \right] \right] dt
\end{aligned} \tag{2.6}$$

Siendo la ecuación (2.4) en donde ambos controles actúan, la ecuación (2.5) donde solo u_1 esta activa y por último en la ecuación (2.6) presente solo u_2 . Mientras que nuestro Hamiltoniano para ambos controles activos se definen como:

$$\begin{aligned}
\mathcal{H} = & q_{22}X_2^2 + q_{33}X_3^2 + q_{44}X_4^2 + q_{66}X_6^2 + r_{11}u_1^2 + r_{22}u_2^2 \\
& + \lambda_1 \left[\mu N - \beta(1 - u_1)X_1 \frac{X_3}{X_4} - \mu X_1 \right] \\
& + \lambda_2 \left[\beta(1 - u_1)X_1 \frac{X_3}{X_4} - (\mu + \theta)X_2 \right] \\
& + \lambda_3 \left[\frac{\lambda}{N} X_2(X_4 - X_3) - \delta X_3 \right] + \lambda_4 [\omega X_5 - \delta X_4] \\
& + \lambda_5 \left[\varphi X_4 \left(1 - \frac{X_5}{\gamma X_6} \right) - (\omega + \varepsilon)X_5 \right] \\
& + \lambda_6 \left[v X_6 \left(1 - \frac{X_6}{K} \right) - u_2 X_6 \right]
\end{aligned} \tag{2.7}$$

Para u_1

$$\begin{aligned}
\mathcal{H} = & q_{22}X_2^2 + q_{33}X_3^2 + q_{44}X_4^2 + q_{66}X_6^2 + r_{11}u_1^2 \\
& + \lambda_1 \left[\mu N - \beta(1 - u_1)X_1 \frac{X_3}{X_4} - \mu X_1 \right] \\
& + \lambda_2 \left[\beta(1 - u_1)X_1 \frac{X_3}{X_4} - (\mu + \theta)X_2 \right] \\
& + \lambda_3 \left[\frac{\lambda}{N} X_2(X_4 - X_3) - \delta X_3 \right] + \lambda_4 [\omega X_5 - \delta X_4] \\
& + \lambda_5 \left[\varphi X_4 \left(1 - \frac{X_5}{\gamma X_6} \right) - (\omega + \varepsilon)X_5 \right] + \lambda_6 \left[v X_6 \left(1 - \frac{X_6}{K} \right) \right]
\end{aligned} \tag{2.8}$$

Mientras que para u_2

$$\begin{aligned}
\mathcal{H} = & q_{22}X_2^2 + q_{33}X_3^2 + q_{44}X_4^2 + q_{66}X_6^2 + r_{22}u_2^2 \\
& + \lambda_1 \left[\mu N - \beta X_1 \frac{X_3}{X_4} - \mu X_1 \right] + \lambda_2 \left[\beta X_1 \frac{X_3}{X_4} - (\mu + \theta)X_2 \right] \\
& + \lambda_3 \left[\frac{\lambda}{N} X_2(X_4 - X_3) - \delta X_3 \right] + \lambda_4 [\omega X_5 - \delta X_4] \\
& + \lambda_5 \left[\varphi X_4 \left(1 - \frac{X_5}{\gamma X_6} \right) - (\omega + \varepsilon)X_5 \right] \\
& + \lambda_6 \left[v X_6 \left(1 - \frac{X_6}{K} \right) - u_2 X_6 \right]
\end{aligned} \tag{2.9}$$

Para obtener las ecuaciones adjuntas se utiliza la ecuación (1.1.19), entonces se tiene que el sistema de ecuaciones adjuntas para el caso 1 es:

$$-\frac{\partial H}{\partial X_1} = \frac{d\lambda_1}{dt} = - \left[\frac{\beta(1 - u_1)X_3}{X_4} (\lambda_2 - \lambda_1) - \mu \lambda_1 \right] \tag{2.10}$$

$$-\frac{\partial H}{\partial X_2} = \frac{d\lambda_2}{dt} = - \left[2q_{22}X_2 - (\mu + \theta)\lambda_2 + \left(\frac{\lambda(X_4 - X_3)}{N} \right) \lambda_3 \right] \tag{2.11}$$

$$-\frac{\partial H}{\partial X_3} = \frac{d\lambda_3}{dt} = - \left[2q_{33}X_3 + \frac{\beta(1 - u_1)X_1}{X_4} (\lambda_2 - \lambda_1) - \left(\frac{\lambda X_2}{N} + \delta \right) \lambda_3 \right] \tag{2.12}$$

$$-\frac{\partial H}{\partial X_4} = \frac{d\lambda_4}{dt} = -\left[2q_{44}X_4 + \frac{\beta(1-u_1)X_1X_3}{X_4^2}(\lambda_1 - \lambda_2) + \left(\frac{\lambda X_2}{N}\right)\lambda_3 - \delta\lambda_4 + \varphi\left(1 - \frac{X_5}{\gamma X_6}\right)\lambda_5\right] \quad 2.13$$

$$-\frac{\partial H}{\partial X_5} = \frac{d\lambda_5}{dt} = -\left[\omega\lambda_4 - \left(\frac{\varphi X_4}{\gamma X_6}\right)\lambda_5 - (\omega + \varepsilon)\lambda_5\right] \quad 2.14$$

$$-\frac{\partial H}{\partial X_6} = \frac{d\lambda_6}{dt} = -\left[2q_{66}X_6 + \left(\frac{\varphi X_4 X_5}{\gamma X_6^2}\right)\lambda_5 + v\lambda_6 - \left(\frac{2vX_6}{K}\right)\lambda_6 - u_2\lambda_6\right] \quad 2.15$$

Para el caso 2

$$-\frac{\partial H}{\partial X_1} = \frac{d\lambda_1}{dt} = -\left[\frac{\beta(1-u_1)X_3}{X_4}(\lambda_2 - \lambda_1) - \mu\lambda_1\right] \quad 2.16$$

$$-\frac{\partial H}{\partial X_2} = \frac{d\lambda_2}{dt} = -\left[2q_{22}X_2 - (\mu + \theta)\lambda_2 + \left(\frac{\lambda(X_4 - X_3)}{N}\right)\lambda_3\right] \quad 2.17$$

$$-\frac{\partial H}{\partial X_3} = \frac{d\lambda_3}{dt} = -\left[2q_{33}X_3 + \frac{\beta(1-u_1)X_1}{X_4}(\lambda_2 - \lambda_1) - \left(\frac{\lambda X_2}{N} + \delta\right)\lambda_3\right] \quad 2.18$$

$$-\frac{\partial H}{\partial X_4} = \frac{d\lambda_4}{dt} = -\left[2q_{44}X_4 + \frac{\beta(1-u_1)X_1X_3}{X_4^2}(\lambda_1 - \lambda_2) + \left(\frac{\lambda X_2}{N}\right)\lambda_3 - \delta\lambda_4 + \varphi\left(1 - \frac{X_5}{\gamma X_6}\right)\lambda_5\right] \quad 2.19$$

$$-\frac{\partial H}{\partial X_5} = \frac{d\lambda_5}{dt} = -\left[\omega\lambda_4 - \left(\frac{\varphi X_4}{\gamma X_6}\right)\lambda_5 - (\omega + \varepsilon)\lambda_5\right] \quad 2.20$$

$$-\frac{\partial H}{\partial X_6} = \frac{d\lambda_6}{dt} = -\left[2q_{66}X_6 + \left(\frac{\varphi X_4 X_5}{\gamma X_6^2}\right)\lambda_5 + v\lambda_6 - \left(\frac{2vX_6}{K}\right)\lambda_6\right] \quad 2.21$$

Para el caso 3

$$-\frac{\partial H}{\partial X_1} = \frac{d\lambda_1}{dt} = -\left[\frac{\beta X_3}{X_4}(\lambda_2 - \lambda_1) - \mu\lambda_1\right] \quad 2.22$$

$$-\frac{\partial H}{\partial X_2} = \frac{d\lambda_2}{dt} = -\left[2q_{22}X_2 - (\mu + \theta)\lambda_2 + \left(\frac{\lambda(X_4 - X_3)}{N}\right)\lambda_3\right] \quad 2.23$$

$$-\frac{\partial H}{\partial X_3} = \frac{d\lambda_3}{dt} = -\left[2q_{33}X_3 + \frac{\beta X_1}{X_4}(\lambda_2 - \lambda_1) - \left(\frac{\lambda X_2}{N} + \delta\right)\lambda_3\right] \quad 2.24$$

$$-\frac{\partial H}{\partial X_4} = \frac{d\lambda_4}{dt} = -\left[2q_{44}X_4 + \frac{\beta X_1 X_3}{X_4^2}(\lambda_1 - \lambda_2) + \left(\frac{\lambda X_2}{N}\right)\lambda_3 - \delta\lambda_4 + \varphi\left(1 - \frac{X_5}{\gamma X_6}\right)\lambda_5\right] \quad 2.25$$

$$-\frac{\partial H}{\partial X_5} = \frac{d\lambda_5}{dt} = -\left[\omega\lambda_4 - \left(\frac{\varphi X_4}{\gamma X_6}\right)\lambda_5 - (\omega + \varepsilon)\lambda_5\right] \quad 2.26$$

$$-\frac{\partial H}{\partial X_6} = \frac{d\lambda_6}{dt} = -\left[2q_{66}X_6 + \left(\frac{\varphi X_4 X_5}{\gamma X_6^2}\right)\lambda_5 + v\lambda_6 - \left(\frac{2vX_6}{K}\right)\lambda_6 - u_2\lambda_6\right] \quad 2.27$$

Para obtener las condiciones finales de la integración se utiliza la ecuación (1.1.20), dando como resultado lo siguiente:

$$\lambda_1(T) = 0 \quad 2.28$$

$$\lambda_2(T) = 2p_{22}X_2(T) \quad 2.29$$

$$\lambda_3(T) = 2p_{33}X_3(T) \quad 2.30$$

$$\lambda_4(T) = 2p_{44}X_4(T) \quad 2.31$$

$$\lambda_5(T) = 0 \quad 2.32$$

$$\lambda_6(T) = 2p_{66}X_6(T) \quad 2.33$$

Para encontrar la última ecuación de Euler-Lagrange se usa la ecuación (1.1.21), dando las siguientes ecuaciones

$$\frac{\partial H}{\partial u_1} = 2r_{11}u_1 + \frac{\beta X_1 X_3}{X_4}(\lambda_1 - \lambda_2) = 0 \quad 2.34$$

$$\frac{\partial H}{\partial u_2} = 2r_{22}u_2 - X_6\lambda_6 = 0 \quad 2.35$$

Cabe mencionar que las ecuaciones (2.34 a 2.35) se utilizan para el caso en que ambos controles están activos, la ecuación (2.34) se utiliza para el caso 2 y la ecuación (2.35) se utiliza cuando el control u_2 se encuentra presente únicamente.

2.3. DIAGRAMA DE FLUJO DEL ALGORITMO DEL MÉTODO DEL GRADIENTE REDUCIDO

Como se mencionó en capítulos anteriores es necesario resolver problemas de valor en la frontera (PVF) por el método del gradiente reducido, a continuación, se muestran el diagrama del algoritmo empleado¹:

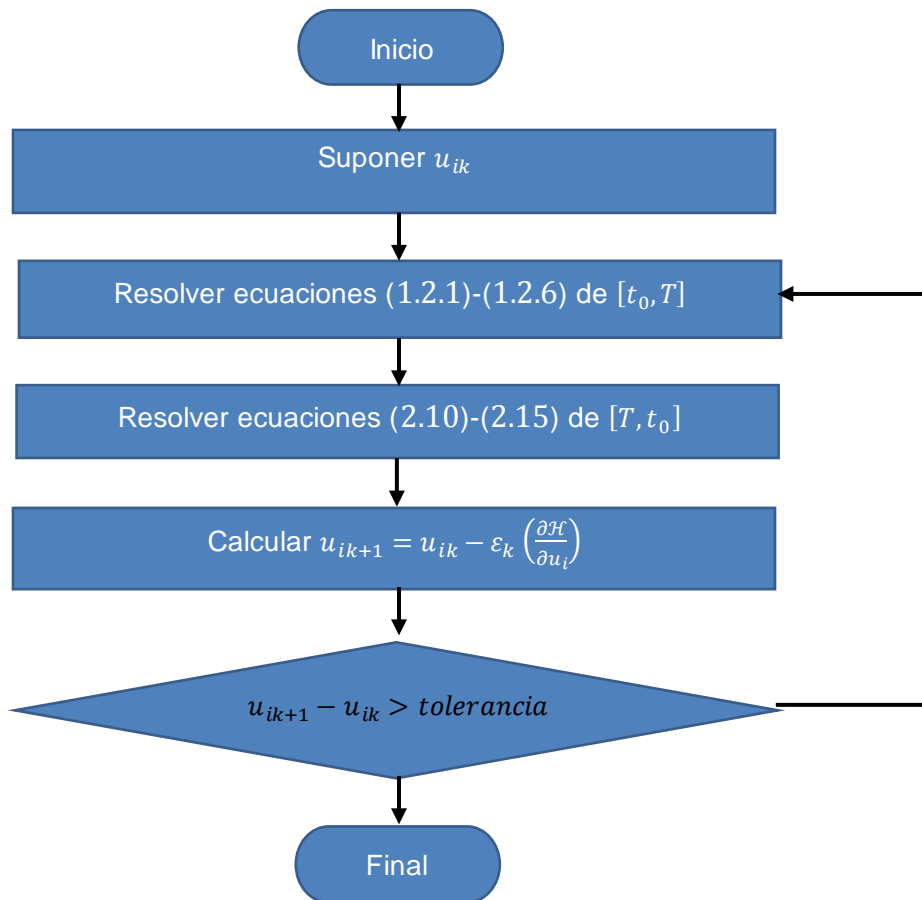


Figura 10 .- Diagrama del algoritmo del método del gradiente reducido

Nota 1: Este diagrama de flujo es para el caso 1, para el caso 2 se sustituyen las ecuaciones (2.16)-(2.21) en lugar de las ecuaciones (2.10)-(2.15) que se evalúan en $[T, t_0]$ y para el caso 3 se sustituyen las ecuaciones (2.21)-(2.27).

CAPÍTULO 3

ANÁLISIS Y DISCUSIÓN DE RESULTADOS

En este capítulo se resuelve el modelo aplicando los métodos de Euler y Runge-Kutta de 4^{to} (cuarto orden), en cada método se dará tres casos de estudio, el primero será que ambos controles actúan, el segundo caso será que el control u_1 estará presente sin u_2 , y el tercer caso mostrará los resultados del caso contrario al segundo caso.

3.1. ESPECIFICACIONES GENERALES

Como se mencionó en capítulos anteriores en la función objetivo (función costo) hay funciones de peso muy importantes que determinan el impacto que tienen las variables en esta. En este trabajo se les dio a todas las funciones de peso el mismo valor como se muestra en la siguiente tabla (4):

Tabla 4 .- Valores de las funciones peso en la función objetivo (función costo)

Función peso	p_{22}	p_{33}	p_{44}	p_{66}	q_{22}	q_{33}	q_{44}	q_{66}	r_{11}	r_{22}
Valor	1	1	1	1	1	1	1	1	1	1

Como se mencionó en el capítulo 1, los controles tienen restricciones, en el caso de las ecuaciones (1.2.8 y 1.2.9) se observa que el control u_1 no puede ser mayor a uno, ni menor a cero, por consiguiente, en el programa se agregó una restricción que si el control supera esos límites se establezca con un valor² de 0.6, mientras que el control u_2 de la ecuación (1.2.8 y 1.2.10), tiene un límite superior de 0.7 y uno inferior de cero, pero el control no puede llegar a 0.7 porque esto implicaría la extinción de todos los mosquitos, esto es casi imposible, por ello, se dio un límite superior a 0.695, todos aquellos valores que superen el límite de $[0,0.695]$ se estabilizarán con un valor de 0.65. Teniendo esto en cuenta se presentarán los resultados de los dos métodos con sus casos respectivos. Se obtuvieron los resultados en un ambiente de Matlab, sin utilizar subprogramas de Matlab, la programación se realizó desde cero, con un procesador icore 7 de tercera generación, Windows 7 y la simulación tardó 24 horas para el método de Euler y 50 horas para el método de Rugen-Kutta 4^{to} orden.

Nota 2: El valor de 0.6 es solamente para el caso en que ambos controles actúen conjuntamente. para el caso que solo esté presente el control u_1 se dará un valor de 0.8, debido a que no converge individualmente con el valor de 0.6.

3.2. MÉTODO DE EULER

A continuación, se muestran los perfiles de cada una de las variables y controles aplicando el método de Euler con control óptimo

3.2.1. CASO 1

3.2.1.1. Variables de estado

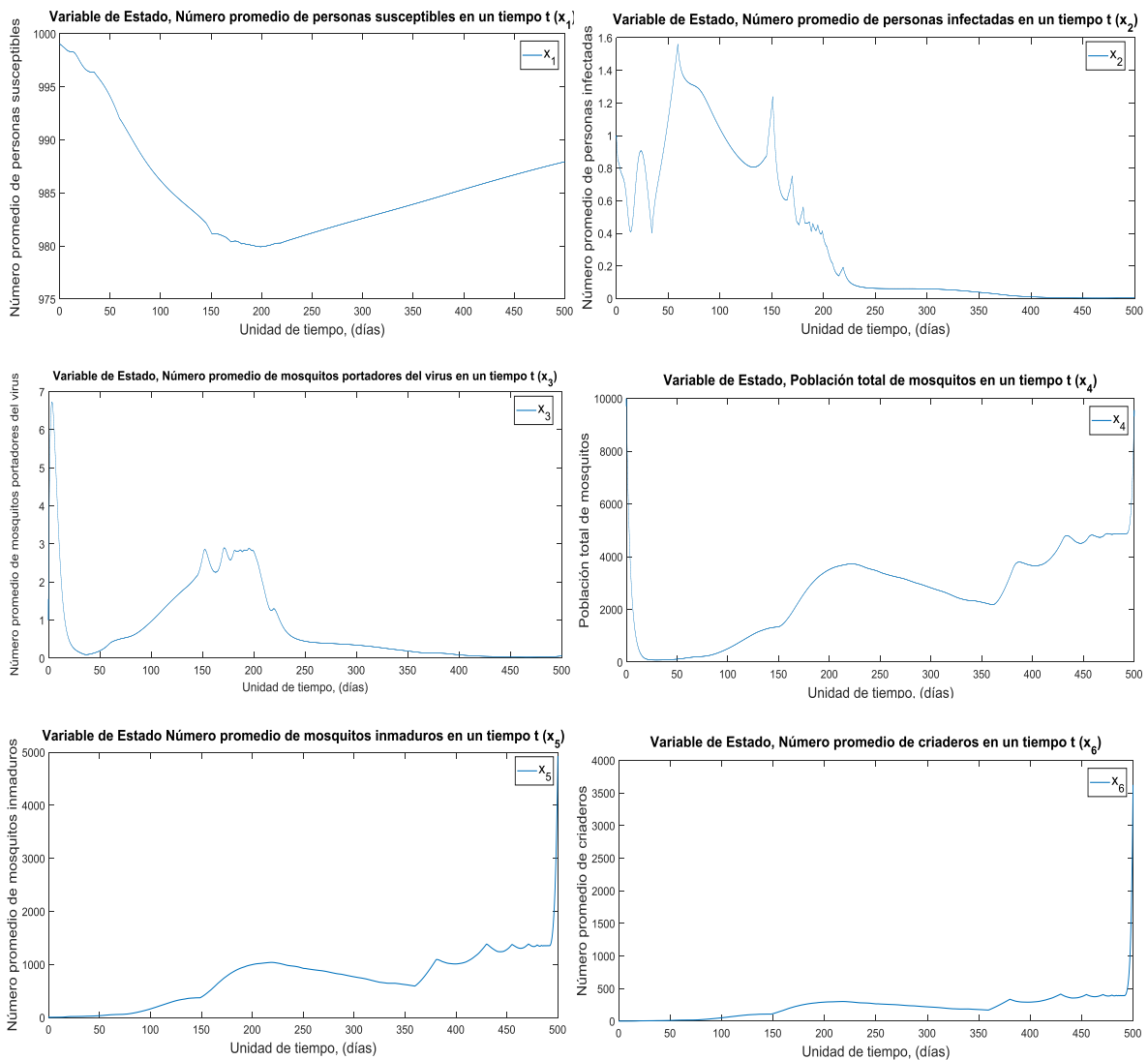


Figura 11 .- Gráficas de las variables de estado para el caso 1 en el método de Euler

3.2.1.2. Controles u_1 y u_2

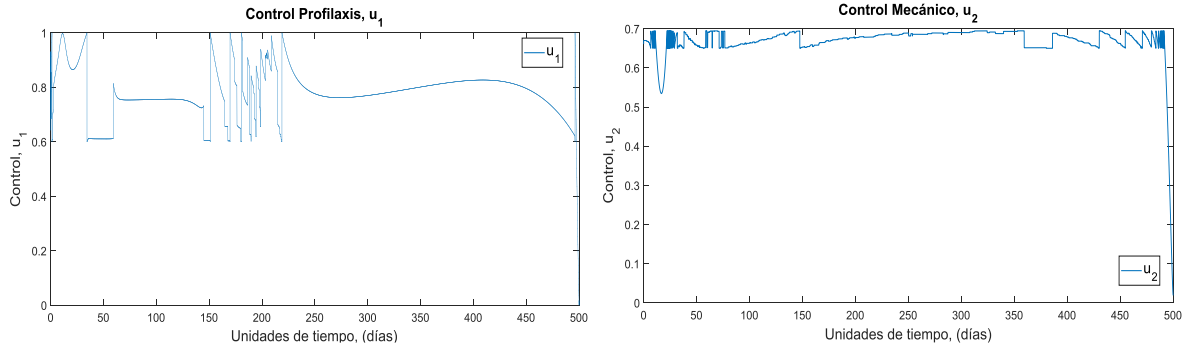
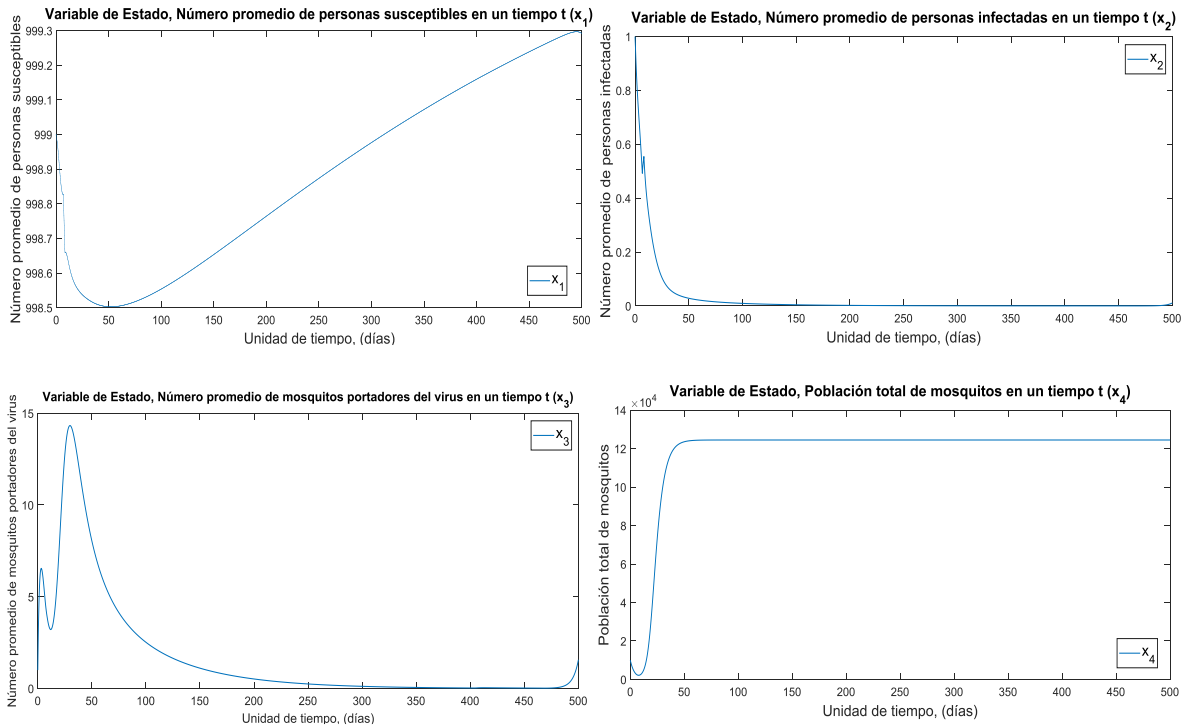


Figura 12 .- Gráficas de los controles u_1 y u_2 para el caso 1 en el método de Euler

3.2.2. CASO 2

3.2.2.1. Variables de estado



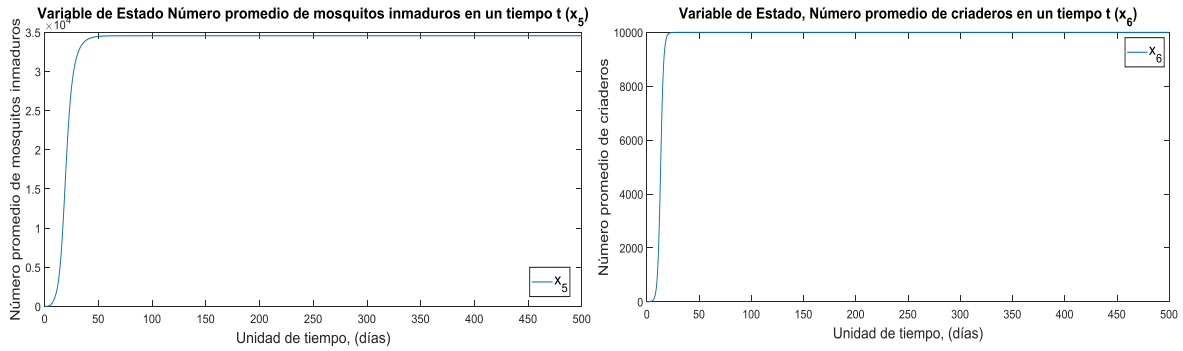


Figura 13 .- Gráficas de las variables de estado para el caso 2 en el método de Euler

3.2.2.2. Control u_1

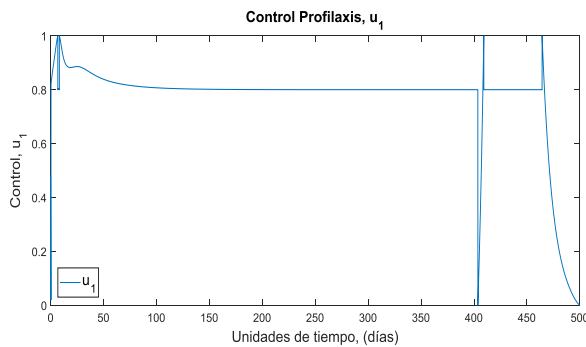
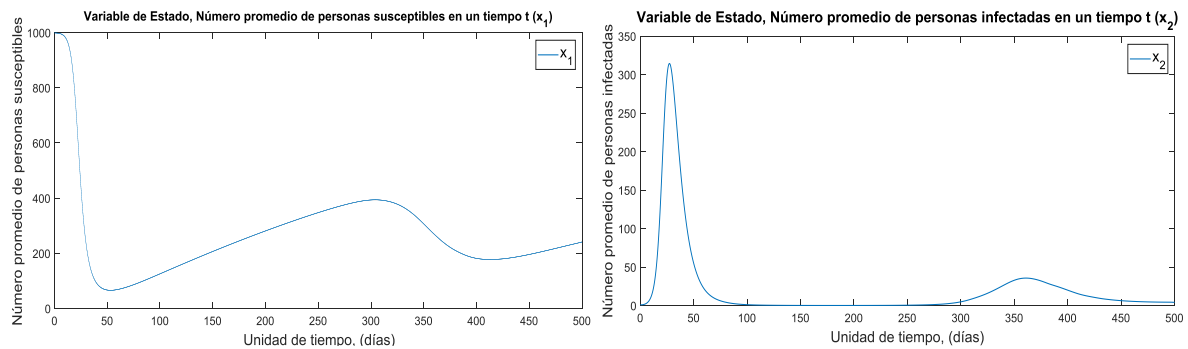


Figura 14 .- Gráfica del control u_1 para el caso 2 en el método de Euler

3.2.3. CASO 3

3.2.3.1. Variables de control



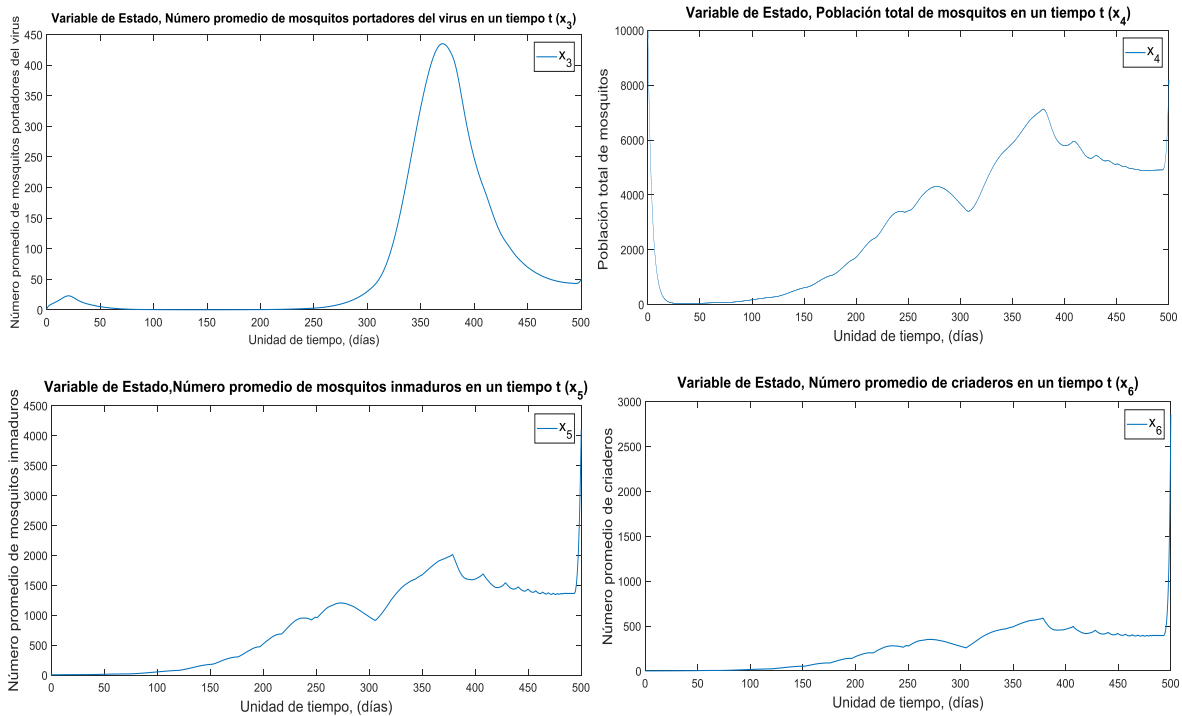


Figura 15 .- Gráficas de las variables de estado para el caso 3 en el método de Euler

3.2.3.2. Control u_2

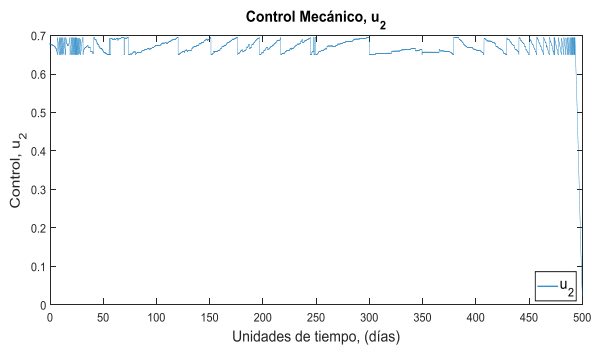


Figura 16 .- Gráfica del control u_2 para el caso 3 en el método de Euler

3.3. MÉTODO DE RUGEN-KUTTA DE 4TO (CUARTO ORDEN)

A continuación, se muestran los perfiles de cada una de las variables y controles aplicando el método de Rugen-Kutta de 4^{to} (cuarto orden) con control óptimo

3.3.1. CASO 1

3.3.1.1. Variables de estado

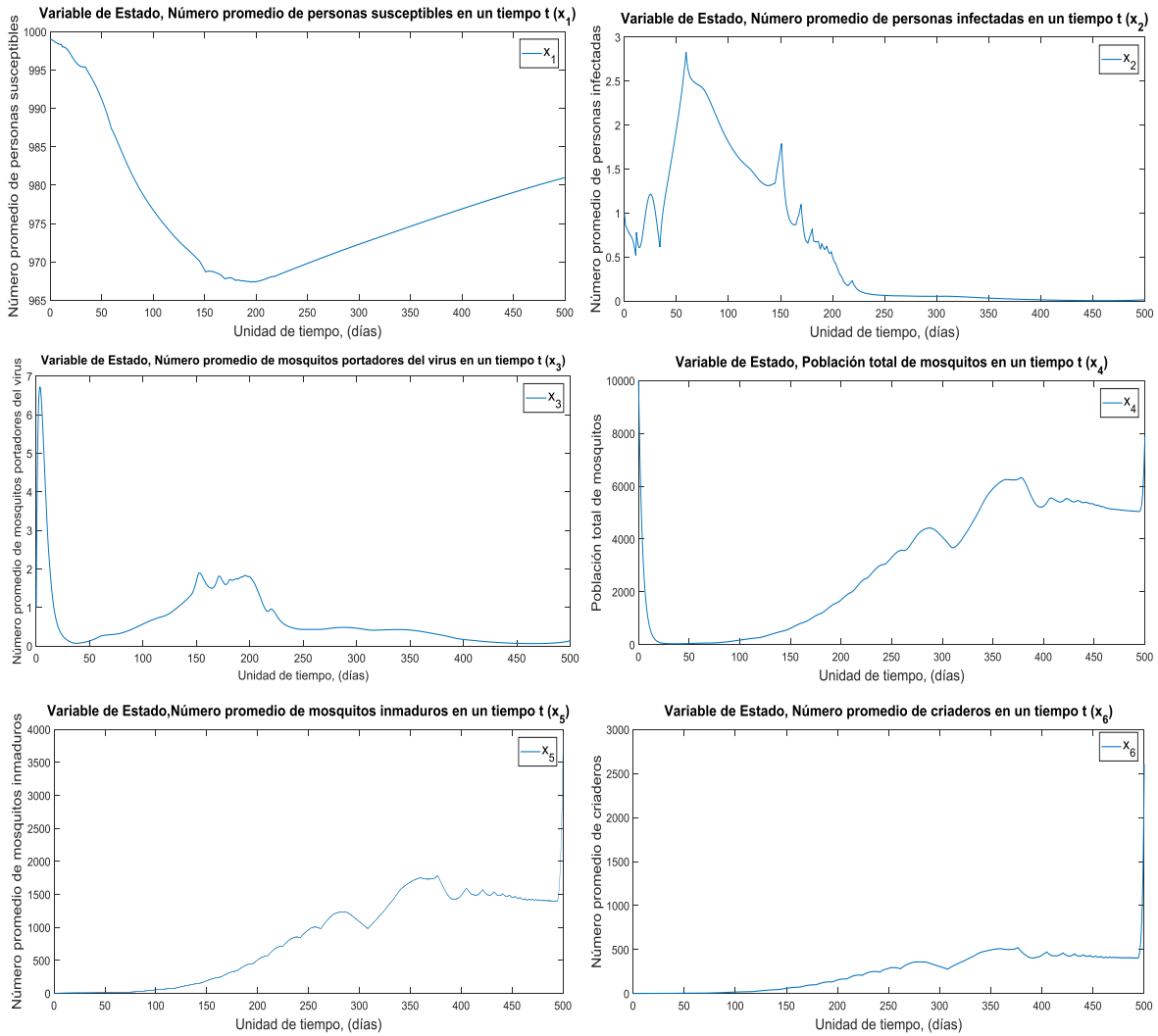


Figura 17 .- Gráficas de las variables de estado para el caso 1 en el método de Rugen-Kutta 4^{to} (cuarto orden)

3.3.1.2. Controles u_1 y u_2

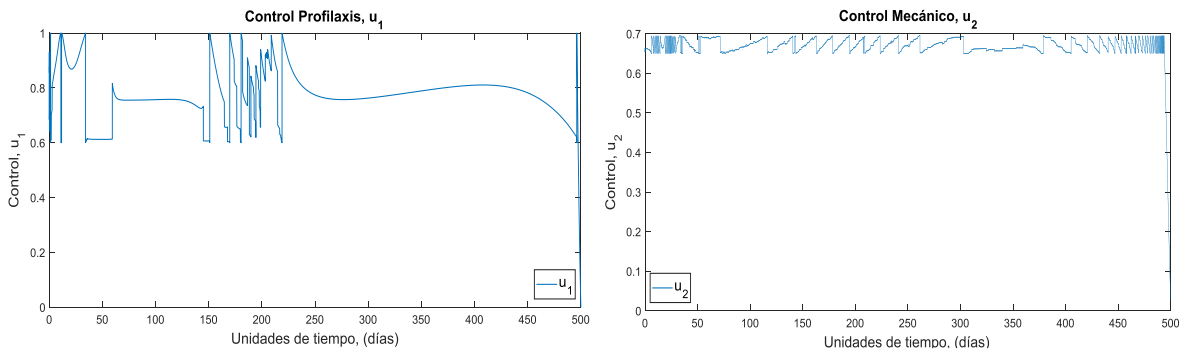
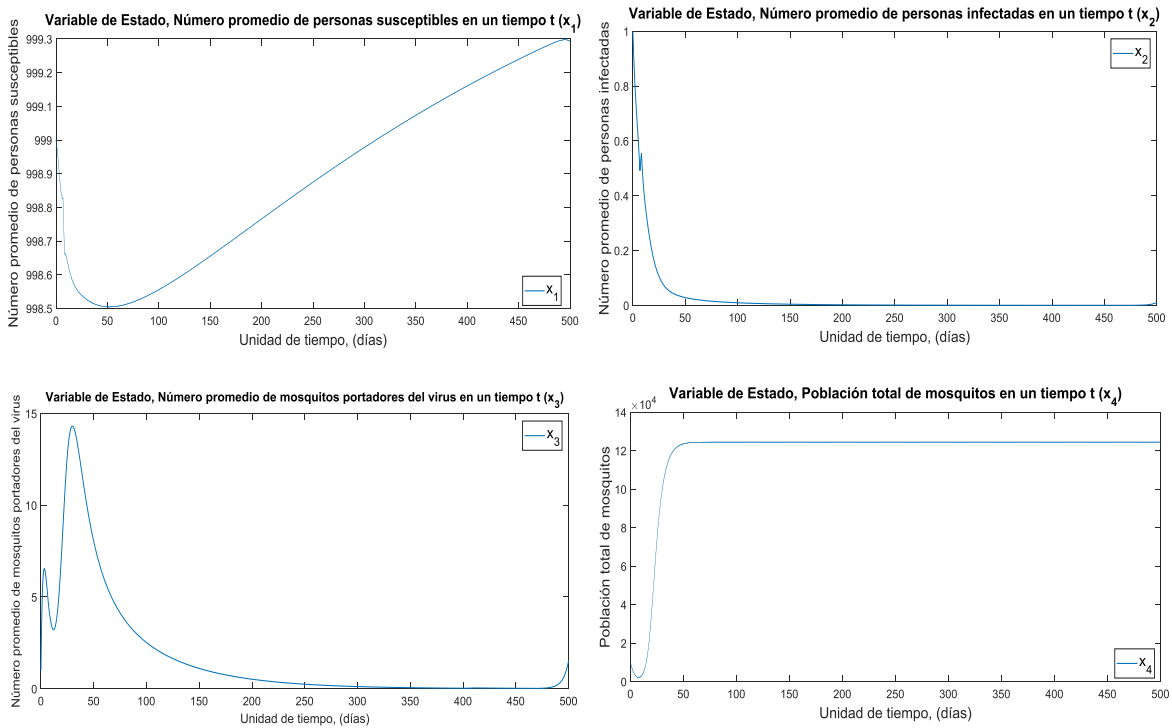


Figura 18 .- Gráficas de los controles u_1 y u_2 para el caso 1 en el método de Rugen-Kutta 4^{to} (cuarto orden)

3.3.2. CASO 2

3.3.2.1. Variables de estado



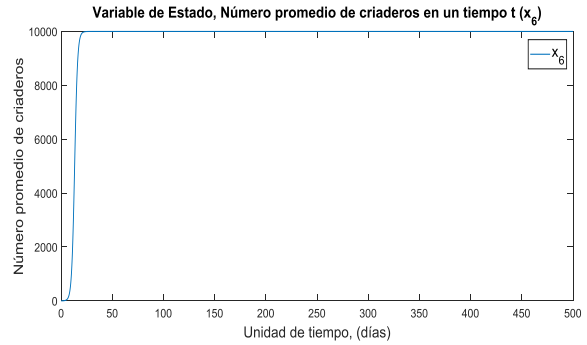
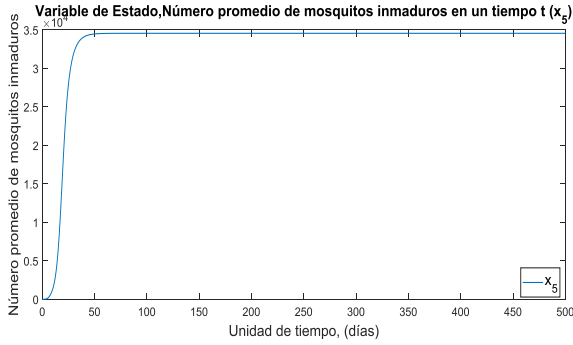


Figura 19 .- Gráficas de las variables de estado para el caso 2 en el método de Rugen-Kutta 4^{to} (cuarto orden)

3.3.2.2. Control u_1

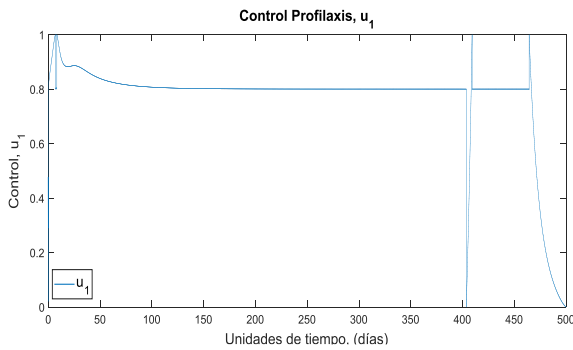
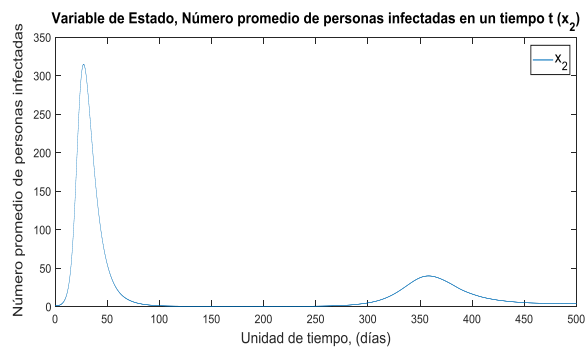
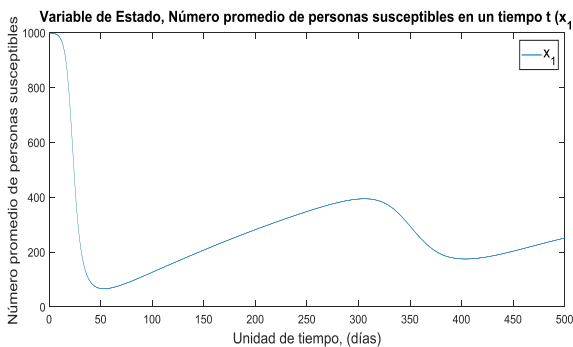


Figura 20 .- Gráfica del control u_1 para el caso 2 en el método de Rugen-Kutta 4^{to} (cuarto orden)

3.3.3. CASO 3

3.3.3.1. Variables de control



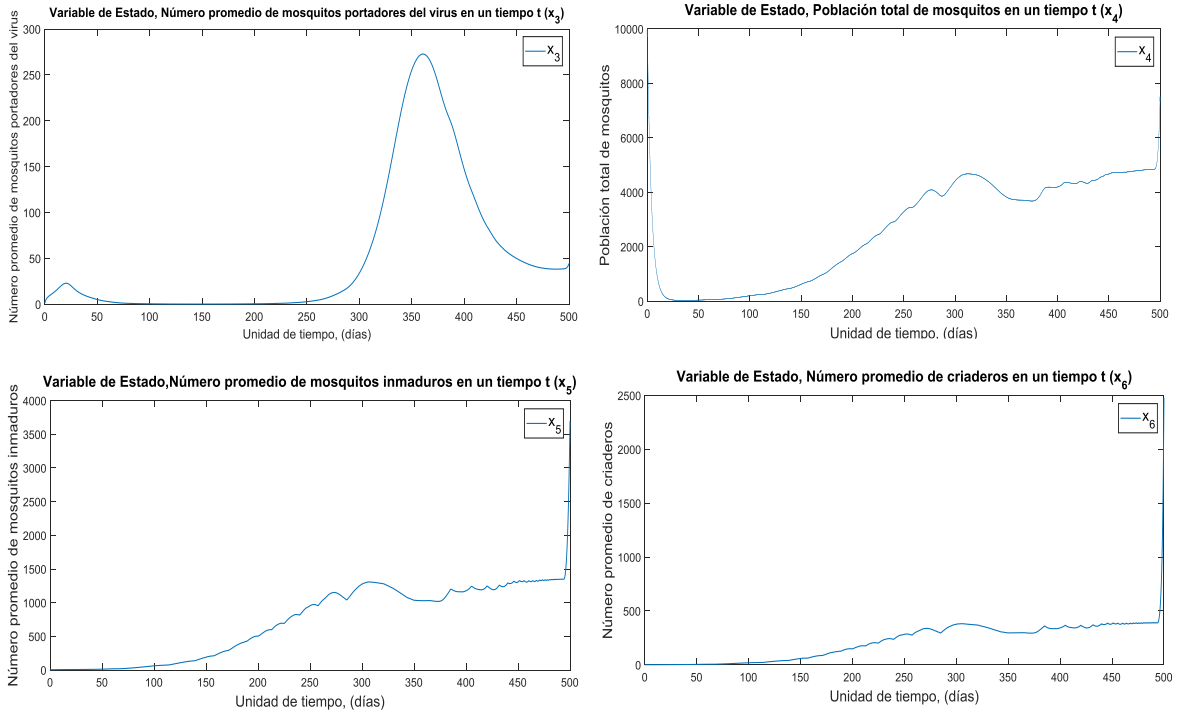


Figura 21 .- Gráficas de las variables de estado para el caso 3 en el método de Rugen-Kutta 4^{to} (cuarto orden)

3.3.3.2. Control u_2

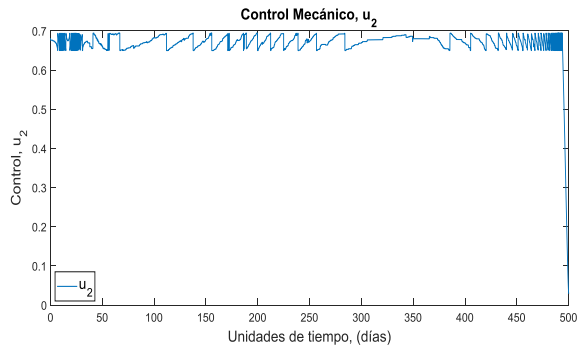


Figura 22 .- Gráfica del control u_2 para el caso 3 en el método de Rugen-Kutta 4^{to} (cuarto orden)

A continuación, se muestra en tabla (5) la comparativa entre los métodos:

Tabla 5.- Resultados de la función objetivo (función costo) de las simulaciones

Control	Euler	RK4	Modelo
	Función objetivo (J)	Función objetivo (J)	Función objetivo (J)
u_1 y u_2	4.8481×10^9	7.1836×10^9	2.7466×10^{10}
u_1	7.3647×10^{12}	7.3648×10^{12}	7.3736×10^{12}
u_2	7.5699×10^9	5.3343×10^9	2.7484×10^{10}

3.4. DISCUSIÓN DE RESULTADOS

Para el caso de la aplicación del método de Euler las variables de estado que fueron optimizadas tuvieron una mejora considerable a comparación del caso de los controles constantes. Las variables Número promedio de personas infectadas (X_2) y Número promedio de mosquitos portadores del virus (X_3) fueron las que tuvieron una mejor optimización a comparación de las variables (X_4) y Número promedio de criaderos (X_6), esto se debe a que el control u_1 no afecta a las variables Población total de mosquitos (X_4), Número promedio de mosquitos inmaduros (X_5) y Número promedio de criaderos (X_6), en caso contrario el control u_2 si las afecta estando activo, pero a diferencia del control u_1 , este afecta a la variable Número promedio de mosquitos portadores del virus (X_3) y Población total de mosquitos (X_4) que están muy presente en las ecuaciones de las variables Número promedio de personas infectadas (X_1) y Número promedio de personas infectadas (X_2), por lo tanto, las variables Número promedio de personas infectadas (X_2) y Número promedio de mosquitos portadores del virus (X_3) tuvieron una gran optimización a comparación de las variables Población total de mosquitos (X_4) y Número promedio de criaderos (X_6) donde sus valores eran casi muy cercanos a los del caso de los controles constantes.

Se observa que el control u_1 tuvo muchas oscilaciones entre el día 150 a 250, esto se puede deber a que las variables Número promedio de personas infectadas (X_2) y Número promedio de mosquitos portadores del virus (X_3) tuvieron un máximo

simultáneamente provocando que el control u_1 entrara en acción para bajar estas variables. Esto también pudiera ser provocado por la variable Población total de mosquitos (X_4) debido al aumento en el mismo intervalo de tiempo y estando muy relacionada a las variables inferiores esto debió ser una causa más de la oscilación del control u_1 .

Después de la oscilación el control u_1 empieza a bajar, esto no provoca un aumento en las variables que son afectadas por el mismo, sino que, empiezan a llegar a un mínimo. La razón es la relación con el control u_2 que provoca tal disminución. Esto se debe que entre los días 250 a 350 el control u_2 empieza a mantener unos valores muy cercanos al 0.695 afectando a sus variables en las que tiene más peso con ellas, Número promedio de mosquitos portadores del virus (X_3) y Población total de mosquitos (X_4).

Cuando inició a haber la oscilación del control u_1 esto debido a las variables Número promedio de mosquitos portadores del virus (X_3) y Población total de mosquitos (X_4), el aumento de las variables correspondientes al vector transmisor (X_3 , X_4 , X_5 y X_6) donde tiene más peso el control u_2 no se hizo esperar debido a que aumentaron gradualmente, esto hizo que el control u_2 aumentara su presencia, a comparación del control u_1 , este no actuó con tanta rapidez al control de las variables, sino que fue más progresiva su presencia.

En los días 350 a 450 el control u_2 empieza a oscilar con ondas de gran longitud provocando que las variables Población total de mosquitos (X_4), Número promedio de mosquitos inmaduros (X_5) y Número promedio de criaderos (X_6) aumentaran levemente, después de ese tiempo el control u_2 empieza oscilar con gran frecuencia dando como resultado que las variables ya mencionadas se estabilizaran en cierto valor. Casi al terminar el periodo de estudio el control u_2 empieza a disminuir hasta llegar a un valor de 0 (cero), haciendo aumentar sus variables donde tiene más peso, en el caso del control u_1 también empieza a llegar a un valor de 0 (cero) pero este no afecta a sus variables. Esto se provocó por el límite de aplicación de controles los cuales no se agregaron a la función Mayer de la función objetivo.

En el caso en el que solo actúa el control u_1 las variables más afectadas son Número promedio de personas infectadas (X_1) y Número promedio de personas infectadas (X_2) llegando rápidamente a sus límites máximos de optimización, para el caso de Número promedio de mosquitos portadores del virus (X_3) fue muy eficiente después del día 200, donde se mantuvo a casi nula su presencia. Ahora si hablamos de la trayectoria del control u_1 fue muy constante a comparación de la aplicación de ambos controles.

Cuando actúa únicamente el control u_2 tiene los mismos efectos finales al igual que cuando actúa conjuntamente con el control u_1 , pero a diferencia al actuar ambos controles, las variables presentan máximos más pronunciados en su trayectoria. Esto conlleva a que, el control tenga más oscilaciones en sus valores de aplicación. En el caso de la variable Número promedio de mosquitos portadores del virus (X_3) tiene una buena aplicación hasta el día 250, después de ahí tiempo la variable aumenta considerablemente, se puede ver una relación cuando se aplica el control u_1 que controla al final de Número promedio de mosquitos portadores del virus (X_3) y el control u_2 control al inicio.

Para el caso de la aplicación del método de Rugen-Kutta 4^{to} (cuarto orden) cuando se aplican ambos controles los resultados son muy semejantes, pero en las mayorías de las variables de estado que son Número promedio de personas infectadas (X_2), Población total de mosquitos (X_4), Número promedio de mosquitos inmaduros (X_5) y Número promedio de criaderos (X_6) tienen máximos muy prolongados esto hace que entre su trayectoria tenga valores muy altos, pero al final llegan al mismo objetivo del método de Euler. Para las variables Número promedio de personas infectadas (X_1) y Número promedio de mosquitos portadores del virus (X_3) fue donde destaco a comparación del otro método teniendo valores más pequeños en su trayectoria.

En el caso de las gráficas de los controles cuando actúan conjuntamente, varia solo en el control u_2 donde en el método de Rugen-Kutta 4^{to} (cuarto orden) oscila mucho más entre los valores de 0.65 y 0.695 que en el método de Euler. Esto provocaría los máximos de todas las gráficas en sus trayectorias.

Para el caso del uso solamente del control u_1 los resultados con el método de Euler son iguales, no existe diferencia entre todas las gráficas de ambos métodos, caso contrario cuando actúa solamente el control u_2 los valores de la trayectoria de todas las variables de estado son más pequeños a comparación del método de Euler usando solo el control u_2 , esto puede deber a que el control u_2 en el caso de Ruge-Kutta 4^{to} (cuarto orden) están un poco más cercanos al 0.695.

CAPÍTULO 4

4.1. CONCLUSIONES

- Se desarrollaron buenas estrategias para la solución del problema de control óptimo para el modelo de la dinámica de transmisión del dengue. Dando un resultado positivo en la resolución de este con diferentes métodos.
- La implementación de las variables y ecuaciones adjuntas obtenidas por medio de las ecuaciones de Euler-Lagrange han sido correctas para el problema de control óptimo de este trabajo.
- Se comprobó que cada método de resolución afecta más a un control en particular que el otro método, esto debido al método de resolución utilizado que es muy sensible, por ello, las gráficas de las variables de estado y controles varían conforme se selecciona el método de resolución de las ecuaciones.
- El control óptimo se aplicó correctamente al modelo matemático de la dinámica de transmisión del dengue dando resultados lógicos y coherentes presentados en las gráficas de las variables de estado.
- Hubo gran diferencia entre el modelo matemático de la dinámica de la transmisión del dengue con controles constantes y cuando los controles son dinámicos en el caso de la aplicación del control óptimo. Dando resultados mucho más favorables y mejores cuando se aplica el control óptimo.
- La aplicación de un solo control en el modelo matemático aplicando el control óptimo no supera la combinación de los controles hablando de la función objetivo (función costo), pero si supera en algunas variables de estado el control individual al combinado dando mejores resultados en la trayectoria.

- El método de Euler es mucho más rápido dando los resultados su contraparte, dando las iteraciones del problema de control óptimo en un tiempo más corto.
- En el caso en que actuaron los controles conjuntamente usando el método de Euler, el control u_1 tuvo mucha más presencia al final del tiempo de estudio que el control u_2 . Cuando el control u_2 estuvo en el último trayecto del tiempo de estudio sus variables donde tiene más peso aumentaron gradualmente, pero al tener más efecto u_1 al final del tiempo esto resulto como una mejora en la trayectoria de las variables de estado.
- Siendo mucho más a fin el método de Euler al control u_1 debido al método de solución, al actuar individualmente el control u_2 provoco que en las variables donde tiene más peso el control u_1 , al iniciar el periodo de estudio no tuvieron una trayectoria favorable con valores muy elevados, al ir finalizando el tiempo de estudio fueron siendo mucho mejores estos valores en las variables de estado.
- En el caso donde solo se usó el control u_2 a comparación del uso de controles conjuntamente, en sus variables donde tiene más peso, al final del tiempo de estudio, tuvieron una trayectoria elevada esto provoco que varias variables de estado donde no tenía el peso completo en el caso de x_3 hubo un aumento considerable.
- En el método de Euler cuando el control u_2 tiene valores casi similares y/o tiene una oscilación con longitud muy pronunciada provoca que sus variables donde tiene más peso se descontrolen más fácilmente. En caso contrario con el control u_1 donde no paso esto.
- Al tener más ecuaciones al resolver este método tardo casi el doble del tiempo que el método de Euler dando iteraciones mucho más lentas y mayor capacidad de memoria para su procesamiento.
- Como se menciona anteriormente cada método tiene afinidad con un control en el caso del uso del método de Rugen-Kutta 4^{to} orden fue con el control u_2 debido al método de solución, esto dio como resultado que al inicio del periodo no contara tanto con el efecto del control u_1 teniendo

máximos al inicio de los primeros días cuando actuaban los controles conjuntamente a comparación del método del Euler.

- Teniendo más influencia el método de Rugen-Kutta 4^{to} orden en el control u_2 debido a la sensibilidad del método de solución, hizo que a mediados y final del periodo de estudio las variables de estado donde tiene más peso no fueran afectas tanto con máximos como es el caso del método de Euler.
- El aplicar solo el control u_1 en este método dio los mismos resultados en la función objetivo (función costo) y en las gráficas que, en el método de Euler, se puede concluir que cuando se usan conjuntamente los dos controles en el método de Rugen-Kutta 4^{to} orden no tiene la misma sensibilidad el método de solución al control u_1 .
- En el caso del control u_2 se sigue manteniendo la sensibilidad del método de solución a este control, con ello al final del del tiempo de estudio mantuvo los máximos que hubo con los controles combinados en niveles mucho más bajos, también mencionar que obtuvo un mejor valor en la función objetivo (función costo) que el método de Euler.
- En el caso de usar los controles conjuntamente este método no da buena optimización comprándolo con el método de Euler, pero usando los controles individuales da excelente optimización.

4.2. RECOMENDACIONES

- Este trabajo intenta comprobar la aplicación del control óptimo en este modelo dinámico.
- Se recomienda que u_1 cuando supere el valor de la unidad se le dé un valor de 0.8 en lugar de 0.6 debido a que con este valor no converge cuando el control u_1 actúe individualmente, en cambio cuando se actúan

conjuntamente los controles no es necesario cambiar el valor del control u_1 a 0.8 sino mantenerlo a 0.6.

- El modelo presentado es completamente teórico, no se podría afirmar que represente completamente el desarrollo de la enfermedad.
- El valor del método gradiente reducido es muy sensible a los valores de ε_k debido a que si se da valores de 1×10^{-8} y 1×10^{-12} de controles u_1 y u_2 respectivamente, el método no converge, en caso contrario si se da valores menores a estos, el método empieza a ser más lento en la convergencia.

BIBLIOGRAFÍA

- 1.- Organización Mundial de la Salud (OMS). Dengue y dengue grave. Centro de prensa. Notas descriptivas. 15 de abril de 2019.
- 2.- Bhatt S, Gething PW, Brady OJ, Messina JP, Farlow AW, Moyes CL et.al. The global distribution and burden of dengue. *Nature*;496:504-507.)
- 3.- Brady OJ, Gething PW, Bhatt S, Messina JP, Brownstein JS, Hoen AG et al. Refining the global spatial limits of dengue virus transmission by evidence-based consensus. *PLoS Negl Trop Dis.* 2012;6: e1760. doi:10.1371/journal.pntd.0001760.
- 4.- Organización Panamericana de la salud (OPS). Actualización Epidemiológica Dengue. *Epidemiological Alerts.* 25 de junio de 2019.
- 5.- Hernán D. Toro-Zapata, Leonardo D. Restrepo, Juan G. Vergaño-Salazar y Aníbal Muñoz-Loaiza. dinámica de transmisión del dengue clásico con control mecánico y profilaxis. *Revista de Salud Pública*, 2010, Volumen 12, Número 6. 27 de diciembre del 2010.
- 6.-Diwekar, U.M. 2003. *Introduction to Applied Optimization.* Kluwer Academic Publishers.
- 7.-Napoles-Rivera F. 2008. Tesis de maestria. Aplicaciones de control optimo estocástico en el tratamiento de enfermedades. instituto tecnologico de celaya.
- 8.-Thomas F. Edgar, David M. Himmelblau y Leon Lasdon. 2001. *Optimization of chemical processes, second edition.* McGraw-Hill Hill Higher Education.
- 9.-Buratani MN, Chen M, Chow A, Coutinho FAB, Goh KT, Lopez LF, et al. Modelling the control strategies against dengue in Singapore. *Epidemiol. Infect.* 2007.
- 10.-Coutinho FAB, Burattini MN, Lopez LF, Massad E. Threshold conditions for a nonautonomous epidemic system describing the population dynamics of dengue. *Bulletin of Mathematical Biology.* 2006.

- 11.-Takahashi LT, Maidana NA, Ferreira Jr. WC. O Aedes e sua onda. Biomatemática XIII 2003.
- 12.-Lloyd AL, Zhang J, Root M. Stochasticity and heterogeneity in host-vector models. Journal of Royal Society Interface 2007.
- 13.-Esteva L, Vargas C. Coexistence of different serotypes of dengue virus. J. Math. Biol.2003.
- 14.- Hernández-Lerma, O. 2005. Control Óptimo Y Juegos Estocásticos. Notas Para el Curso: "Juegos Dinámicos y Control Óptimo" Impartido en CIMAT, Guanajuato.
- 15.- Lewis Frank L., VrabieDraguna y Vassilis Syrmos L. 2012. Optimal Control. 3^{ra} edición. JOHN WILEY & SONS, INC.

APÉNDICE 1

CÓDIGO DEL MÉTODO DE EULER CONTROLES COMBINADOS

```
function Angel_x2_x3_x4_x6_cost_function_Euler_Rest_C
clc
clear
format long

%=====
%Definición de parámetros
%=====

global v0 v1 v2 v3 v4 v5 v6 v7 v8 v9 v10 v11 q22 q33 q44 q66 r11 r22 p22
p33 p44 p66

v0 = 1000;           %Población total humana
v1 = 0.9;            %Tasa de maduración del mosquito al estado adulto
v2 = 0.25;          %Tasa de mortalidad de los mosquitos maduros
v3 = 0.9;            %Tasa de ovoposición de los de los mosquitos
v4 = 5;              %Proporción de mosquitos por criadero
v5 = 10000;         %Número máximo de criaderos que soporta el medio
v6 = 0.1;            %Tasa de mortalidad de los estados inmaduros
v7 = 0.002;         %Tasa de muerte/natalidad humana
v8 = 0.3;            %Probabilidad de transmisión mosquito
portador/persona susceptible
v9 = 0.14;           %Tasa de recuperación de las personas infectadas
v10 = 0.5;           %Probabilidad de transmisión persona infectada/vector
no portador
v11 = 0.7;           %Tasa de creación de criaderos
q22 = 1;
q33 = 1;
q44 = 1;
q66 = 1;
r11 = 1;
r22 = 1;
p22 = 1;
p33 = 1;
p44 = 1;
p66 = 1;

%=====
%Condiciones para la resolución del problema
%=====

ti=0;                %Tiempo inicial
tf=500;              %Tiempo final

nstep=500000;        %Número de pasos
h=(tf-ti)/nstep;     %Tamaño de paso
```

```

ne=6; %Número de ecuaciones diferenciales (son
seis para las variables de estado y seis para las variables adjuntas)

%=====
%Perfiles supuestos para las variables de control
%=====

u1=zeros(nstep+1,1);
u1(1)=0.6; %Inicializa el control u1

u2=zeros(nstep+1,1);
u2(1)=0.65; %Inicializa el control u2

Ts=zeros(nstep+1,1);
Ts(1)=ti; %para integrar hacia adelante

Tb=zeros(nstep,1);
Tb(nstep+1)=tf; %para integrar hacia atras

%=====
%Inicialización de variables
%=====

k1=zeros(ne); %Inicializa valores de k utilizados en
la integración Euler de las variables de estado
ka=zeros(ne); %Inicializa valores de k utilizados en
la integración Euler de las variables adjuntas

%=====
%Inicializa las variables en las que se almacenan los resultados de la
integración Euler para las variables de estado
%=====

xu=zeros(nstep+1,ne);

%=====
%Inicializa las variables en las que se almacenan los resultados de la
integración Euler para las variables adjuntas
%=====

lu=zeros(nstep+1,ne);

%=====
%Condiciones iniciales: son las mismas sin importar que control sea usado
%=====

```

```

x0(1)=999; %X1 = 999 Número promedio de
personas susceptibles en un tiempo t
x0(2)=1; %X2 = 1 Número promedio de
personas infectadas en un tiempo t
x0(3)=1; %X3 = 1 Número promedio de
mosquitos portadores del virus en un tiempo t
x0(4)=9999; %X4 = 9999 Población total de
mosquitos en un tiempo t
x0(5)=1; %X5 = 1 Número promedio de
mosquitos inmaduros en un tiempo t
x0(6)=1; %X6 = 1 Número promedio de
criaderos en un tiempo t

%=====
%Asignación de las condiciones iniciales en el vector de las variables de
estado
%=====

for i=1:ne
    xu(1,i)=x0(i);
end

%=====
%Asignación del error superior al del ciclo y numero de iteraciones
%=====

errorT = 2;
nit = 0;

%=====
%Iniciación del calculo
%=====

while errorT >= 1

    %=====
    %Integración Euler para la variable de estado
    %=====

    for j=2:nstep+1

        dx=ec_es(Ts(j-1),xu(j-1,:),u1(j-1),u2(j-1));
        for i=1:ne
            k1(i)= h*dx(i);
%Evalua k1 = xu*h
        end

        %=====

```

```

%Perfil del tiempo
%=====
Ts(j)=Ts(j-1)+ h;

%=====
%Perfiles de las variables de estado
%=====

xu(j,1) = xu(j-1,1) + k1(1);
xu(j,2) = xu(j-1,2) + k1(2);
xu(j,3) = xu(j-1,3) + k1(3);
xu(j,4) = xu(j-1,4) + k1(4);
xu(j,5) = xu(j-1,5) + k1(5);
xu(j,6) = xu(j-1,6) + k1(6);

end

%=====
%=====
%Condiciones finales se tendrá que definir una para cada caso (se
obtienen de la integración de las variables de estado)

%=====
%=====

lf(1)=0;
lf(2)=2*p22*xu(end,2);
lf(3)=2*p33*xu(end,3);
lf(4)=2*p44*xu(end,4);
lf(5)=0;
lf(6)=2*p66*xu(end,6);

%=====
%=====
%Asignación de las condiciones finales en el vector de las variables
adjuntas

%=====
%=====

for i=1:ne
    lu(end,i)=lf(i);
end

%=====
%Integración Euler para las variables adjuntas
%=====

```

```

for j=nstep+1:-1:2

dlu=ec_adj(Tb(j),lu(j,:),xu(j,1),xu(j,2),xu(j,3),xu(j,4),xu(j,5),xu(j,6),
u1(j),u2(j)); %Evalua la funcion en f(tn,xn)
    for i=1:ne
        ka(i)= h*dlu(i);
%Evalua ka = lu * h
    end

%=====
%Perfil del tiempo
%=====

Tb(j-1)=Tb(j)- h;

%=====
%Perfiles de las variables de adjuntas
%=====

lu (j-1,1) = lu(j,1) - ka(1);
lu (j-1,2) = lu(j,2) - ka(2);
lu (j-1,3) = lu(j,3) - ka(3);
lu (j-1,4) = lu(j,4) - ka(4);
lu (j-1,5) = lu(j,5) - ka(5);
lu (j-1,6) = lu(j,6) - ka(6);

end

%=====
%Actualización del control u(k+1)=u(k)-eps(dH/du)
%=====

disp
('=====');

%=====
%Actualización del control u1
%=====

u1n = u1 - 1e-8.*(2*r11.*u1 +
((v8.*xu(:,1)).*xu(:,3))./xu(:,4)).*(lu(:,1)-lu(:,2)));

ulk = zeros (nstep+1,1);

for i=1:nstep+1
    if u1n(i)<0
        u1n(i)=0.6;
        ulk(i) = u1n(i);
    else

```

```

        if u1n(i)>1
            u1n(i) = 0.6;
            u1k(i) = u1n(i);
        else
            u1k(i) = u1n (i);
        end
    end
end

su1 = sum (u1);
sul1 = sum(u1k);

error1=abs(su1-sul1);

u1= u1k;

disp ('error1');
disp (error1)

disp ('suma de u1')
disp (su1)

disp ('suma de u1n');
disp (sul1)

%=====
%Actualización del control u2
%=====

u2n = u2 - 1e-12.*(2*r22.*u2 - xu(:,6).*lu(:,6));

u2k = zeros (nstep+1,1);

for i=1:nstep+1
    if u2n(i)<0
        u2n(i)=0.65;
        u2k(i) = u1n (i);
    else
        if u2n(i)>0.695
            u2n(i) = 0.65;
            u2k(i) = u1n (i);
        else
            u2k (i) = u2n (i);
        end
    end
end

su2 = sum (u2);
su2k = sum(u2k);

error2=abs (su2-su2k);

```



```

u2=u2k;

disp ('error2');
disp (error2)

disp ('suma de u2');
disp (su2)

disp ('suma de u2n');
disp (su2k)

%=====
%Calculo del error total
%=====

errorT = error2 + error1;

disp ('errorT');
disp (errorT)

%=====
%Actualización de la iteración
%=====

nit = nit + 1;
disp ('nit');
disp (nit)

disp
('=====');
end

disp('  - - - - - ')
disp(' |                               RESULTADOS')
disp('  - - - - - ')

%=====
%Evaluación de las funciones objetivo
%La parte integral se evalua con un cumulative Simpson
%=====

%=====Cost Function=====
%                               Arguments of the Cost Function

```

```

yx2=q22*xu(:,2).^2;
yx3=q33*xu(:,3).^2;
yx4=q44*xu(:,4).^2;
yx6=q66*xu(:,6).^2;

y1=r11*u1.^2;
y2=r22*u2.^2;

ar0=cumsimpson(Ts,yx2);
ar1=cumsimpson(Ts,yx3);
ar2=cumsimpson(Ts,yx4);
ar3=cumsimpson(Ts,yx6);

ar32=cumsimpson(Ts,y1);
ar33=cumsimpson(Ts,y2);

q22x2sqr=ar0(end);
q33x3sqr=ar1(end);
q44x4sqr=ar2(end);
q66x6sqr=ar3(end);

r11u1sqr=ar32(end);
r22u2sqr2=ar33(end);

%Cost Function
J=(p22*xu(end,2)^2+p33*xu(end,3)^2+p44*xu(end,4)^2+p66*xu(end,6)^2)+(q22x
2sqr+q33x3sqr+q44x4sqr+q66x6sqr+r11u1sqr+r22u2sqr2);

disp('J');
disp(J)

disp(' ----- ')
disp(' | GRÁFICOS | ')
disp(' ----- ')

%=====
%
% Gráficas de las variables de estado
%=====

figure(1)
plot(Ts,xu(:,1))
xlabel('Unidad de tiempo, (días)')
ylabel('Número promedio de personas susceptibles')
legend('x_1');
title('Variable de Estado, Número promedio de personas susceptibles en un
tiempo t (x_1)')

```

```

figure(2)
plot(Ts,xu(:,2))
xlabel('Unidad de tiempo, (días)')
ylabel('Número promedio de personas infectadas')
legend('x_2');
title('Variable de Estado, Número promedio de personas infectadas en un tiempo t (x_2)')

```

```

figure(3)
plot(Ts,xu(:,3))
xlabel('Unidad de tiempo, (días)')
ylabel('Número promedio de mosquitos portadores del virus')
legend('x_3');
title('Variable de Estado, Número promedio de mosquitos portadores del virus en un tiempo t (x_3)')

```

```

figure(4)
plot(Ts,xu(:,4))
xlabel('Unidad de tiempo, (días)')
ylabel('Población total de mosquitos')
legend('x_4');
title('Variable de Estado, Población total de mosquitos en un tiempo t (x_4)')

```

```

figure(5)
plot(Ts,xu(:,5))
xlabel('Unidad de tiempo, (días)')
ylabel('Número promedio de mosquitos inmaduros')
legend('x_5');
title('Variable de Estado, Número promedio de mosquitos inmaduros en un tiempo t (x_5)')

```

```

figure(6)
plot(Ts,xu(:,6))
xlabel('Unidad de tiempo, (días)')
ylabel('Número promedio de criaderos')
legend('x_6');
title('Variable de Estado, Número promedio de criaderos en un tiempo t (x_6)')

```

```

%=====
==
%                 Gráficas de las variables Adjuntas
%=====
==

```

```

figure(7)
plot(Tb,lu(:,1))
xlabel('Unidades de tiempo, (días)')
ylabel('Variable Adjunta, lambda_1')
legend('lambda_1');
title('Variables adjuntas, lambda_1')

```

```

figure(8)

```

```

plot(Tb,lu(:,2))
xlabel('Unidades de tiempo, (días)')
ylabel('Variable Adjunta, lambda_2')
legend('lambda_2');
title('Variables adjuntas, lambda_2')

```

```

figure(9)
plot(Tb,lu(:,3))
xlabel('Unidades de tiempo, (días)')
ylabel('Variable Adjunta, lambda_3')
legend('lambda_3');
title('Variables adjuntas, lambda_3')

```

```

figure(10)
plot(Tb,lu(:,4))
xlabel('Unidades de tiempo, (días)')
ylabel('Variable Adjunta, lambda_4')
legend('lambda_4');
title('Variables adjuntas, lambda_4')

```

```

figure(11)
plot(Tb,lu(:,5))
xlabel('Unidades de tiempo, (días)')
ylabel('Variable Adjunta, lambda_5')
legend('lambda_5');
title('Variables adjuntas, lambda_5')

```

```

figure(12)
plot(Tb,lu(:,6))
xlabel('Unidades de tiempo, (días)')
ylabel('Variable Adjunta, lambda_6')
legend('lambda_6');
title('Variables adjuntas, lambda_6')

```

```

%=====
==
%               Gráficas de los Controles
%=====
==

```

```

figure(13)
plot(Ts,u1)
xlabel('Unidades de tiempo, (días)')
ylabel('Control, u_1')
legend('u_1');
title('Control Profilaxis, u_1')

```

```

figure(14)
plot(Ts,u2)
xlabel('Unidades de tiempo, (días)')
ylabel('Control, u_2')
legend('u_2');
title('Control Mecánico, u_2')

```

end

```
%=====
==
%           Ecuaciones diferenciales para las variables de estado
%=====
==
```

```
function dx=ec_es(t,x,u1,u2)
```

```
global v0 v1 v2 v3 v4 v5 v6 v7 v8 v9 v10 v11
```

```
%Sistema de ecuaciones diferenciales de variables de estado
```

```
dx=zeros(6,1);
```

```
dx(1) = v7*v0 - v8*(1-u1)*x(1)*(x(3)/x(4)) - v7*x(1);
dx(2) = v8*(1-u1)*x(1)*(x(3)/x(4)) - (v7 + v9)*x(2);
dx(3) = (v10/v0)*x(2)*(x(4)-x(3)) - v2*x(3);
dx(4) = v1*x(5) - v2*x(4);
dx(5) = v3*x(4)-(x(5)*v3*x(4))/(v4*x(6)) - (v1+v6)*x(5);
dx(6) = v11*x(6)*(1-(x(6))/(v5)) - u2*x(6);
```

end

```
%=====
==
%           Ecuaciones para las variables adjuntas
%=====
==
```

```
function dl=ec_adj(t,l,x1,x2,x3,x4,x5,x6,u1,u2)
```

```
global v0 v1 v2 v3 v4 v5 v6 v7 v8 v9 v10 v11 q22 q33 q44 q66
```

```
%%Sistema de ecuaciones diferenciales de variables adjuntas
```

```
dl=zeros(6,1);
```

```
dl(1) = -( ((v8*(1-u1)*x3)/(x4))*l(2) - l(1)) - v7*l(1) );
dl(2) = -( 2*q22*x2 - (v7+v9)*l(2) + ((v10*(x4-x3))/(v0))*l(3) );
dl(3) = -( 2*q33*x3+((v8*(1-u1)*x1)/(x4))*l(2) - l(1)) - ((v10*x2)/(v0)
+ v2)*l(3) );
dl(4) = -( 2*q44*x4 + ((v8*(1-u1)*x1*x3)/(x4^2))*l(1) - l(2)) +
(v10*l(3)*x2)/(v0) - v2*l(4) + v3*(1-(x5)/(v4*x6))*l(5) );
dl(5) = -( v1*l(4) - (v3*x4*l(5))/(v4*x6) - (v1+v6)*l(5) );
dl(6) = -( 2*q66*x6 + (v3*x4*x5*l(5))/(v4*x6^2) + v11*l(6) -
(2*v11*x6*l(6))/(v5) - u2*l(6) );
```

end

```

function z=cumsimpson(x,y)
%{
  CUMSIMPSON Cumulative Simpson Numerical Integration.
  Z=CUMSIMPSON(Y) computes an approximation of the cumulative integral of
  Y
  using a Simpson rule with unit spacing between the data points in Y. To
  compute the integral for spacing different from one, multiply Z by the
  spacing increment. When Y is a matrix, the cumulative integral is
  computed over each column of Y.

  Z=CUMSIMPSON(X,Y) computes the integral with respect to the data in
  vector X. X need NOT be EQUALLY spaced but must have the same number of
  elements as Y. When Y is a matrix, Y must have as many rows as X has
  elements.

  Z has the same dimensions as Y.

  See also CUMSUM, CUMTRAPZ, QUAD, QUADV.
  D.C. Hanselman, University of Maine, Orono, ME 04469
  MasteringMatlab@yahoo.com
  Mastering MATLAB 7
  2005-09-30, 2006-02-21
%}

if nargin<2
  y=x;
  [ry,cy]=size(y);
  if ry==1
    x=1:cy;
  else
    x=1:ry;
  end
elseif nargin==2
  [ry,cy]=size(y);
else
  error('One or Two Inputs Are Required.')
end
if ndims(y)~=2
  error('N-dimensional Data is Not Supported.')
end
if min(size(x))>1
  error('X Must be a Vector.')
end
x=x(:); % make x a column
if ry==1 % y is a row vector, make it a column
  yisrow=true;
  y=y.';
  ry=cy;
  cy=1;
else
  yisrow=false;
end
if length(x)~=ry

```

```

    error('Length of X Must Match Length of Y or Rows of Matrix Y.')
end
if length(x)<3
    error('At Least 3 Data Points are Required.')
end
dx= repmat(diff(x),1,cy);
dy=diff(y);

dx1=dx(1:end-1,:);
dx2=dx(2:end,:);
dxs=dx1+dx2;
dy1=dy(1:end-1,:);
dy2=dy(2:end,:);

a=(dy2./(dx2.*dxs) - dy1./(dx1.*dxs))/3;
b=(dy2.*dx1./(dx2.*dxs) + dy1.*dx2./(dx1.*dxs))/2;
c=y(2:end-1,:);

i1=(a.*dx1-b).*dx1+c).*dx1; % left half integral
i2=(a.*dx2+b).*dx2+c).*dx2; % right half integral

z=zeros(ry,cy); % pure cumulative Simpson
z(2:2:end-1,:)=i1(1:2:end,:);
z(3:2:end,:)=i2(1:2:end,:);
z(end,:)=i2(end,:);
z=cumsum(z);

% z=[zeros(1,cy);... % original algorithm here has ~2X error
% cumsum([i1(1,:); (i1(2:end,:)+i2(1:end-1,:))/2; i2(end,:)])];

if yisrow
    z=z.';
end

end

```

APÉNDICE 2

CÓDIGO DEL MÉTODO DE RUGEN-KUTTA 4^{TO} ORDEN CONTROLES COMBINADOS

```
function Angel_x2_x3_x4_x6_cost_function_RK4_Rest_C
clc
clear
format long

%=====
%Definición de parámetros
%=====

global v0 v1 v2 v3 v4 v5 v6 v7 v8 v9 v10 v11 q22 q33 q44 q66 r11 r22 p22
p33 p44 p66

v0 = 1000;           %Población total humana
v1 = 0.9;            %Tasa de maduración del mosquito al estado adulto
v2 = 0.25;           %Tasa de mortalidad de los mosquitos maduros
v3 = 0.9;            %Tasa de ovoposición de los de los mosquitos
v4 = 5;              %Proporción de mosquitos por criadero
v5 = 10000;          %Número máximo de criaderos que soporta el medio
v6 = 0.1;            %Tasa de mortalidad de los estados inmaduros
v7 = 0.002;          %Tasa de muerte/natalidad humana
v8 = 0.3;            %Probabilidad de transmisión mosquito
portador/persona susceptible
v9 = 0.14;           %Tasa de recuperación de las personas infectadas
v10 = 0.5;           %Probabilidad de transmisión persona infectada/vector
no portador
v11 = 0.7;           %Tasa de creación de criaderos
q22 = 1;
q33 = 1;
q44 = 1;
q66 = 1;
r11 = 1;
r22 = 1;
p22 = 1;
p33 = 1;
p44 = 1;
p66 = 1;

%=====
%Condiciones para la resolución del problema
%=====

ti=0;                %Tiempo inicial
tf=500;              %Tiempo final

nstep=500000;        %Número de pasos
```



```

h=(tf-ti)/nstep;           %Tamaño de paso

ne=6;                      %Número de ecuaciones diferenciales (son
seis para las variables de estado y seis para las variables adjuntas)

%=====
%Perfiles supuestos para las variables de control
%=====

u1=zeros(nstep+1,1);
u1(1)=0.6;                 %Inicializa el control u1

u2=zeros(nstep+1,1);
u2(1)=0.65;               %Inicializa el control u2

Ts=zeros(nstep+1,1);
Ts(1)=ti;                 %para integrar hacia adelante

Tb=zeros(nstep,1);
Tb(nstep+1)=tf;          %para integrar hacia atras

%=====
%Inicialización de variables
%=====

k1=zeros(ne);             %Inicializa valores de k utilizados en
la integración RK4 de las variables de estado
ka=zeros(ne);             %Inicializa valores de k utilizados en
la integración RK4 de las variables adjuntas

%=====
%Inicializa las variables en las que se almacenan los resultados de la
integración RK4 para las variables de estado
%=====

xu=zeros(nstep+1,ne);

%=====
%Inicializa las variables en las que se almacenan los resultados de la
integración RK4 para las variables adjuntas
%=====

lu=zeros(nstep+1,ne);

```

```

%=====
%Condiciones iniciales: son las mismas sin importar que control sea usado
%=====

x0(1)=999;                %X1 = 999           Número promedio de
personas susceptibles en un tiempo t
x0(2)=1;                 %X2 = 1           Número promedio de
personas infectadas en un tiempo t
x0(3)=1;                 %X3 = 1           Número promedio de
mosquitos portadores del virus en un tiempo t
x0(4)=9999;              %X4 = 9999        Población total de
mosquitos en un tiempo t
x0(5)=1;                 %X5 = 1           Número promedio de
mosquitos inmaduros en un tiempo t
x0(6)=1;                 %X6 = 1           Número promedio de
criaderos en un tiempo t

%=====
=====
%Asignación de las condiciones iniciales en el vector de las variables de
estado
%=====
=====

    for i=1:ne
        xu(1,i)=x0(i);
    end

%=====
%Asignación del error superior al del ciclo y numero de iteraciones
%=====

errorT = 2;
nit = 0;

%=====
%Iniciación del calculo
%=====

while errorT >= 1

    %=====
    %Integración RK4 para la variable de estado
    %=====

    for j=2:nstep+1

        dx=ec_es(Ts(j-1),xu(j-1,:),u1(j-1),u2(j-1));
%Evalua la funcion en f(tn,xn)
        for i=1:ne

```

```

        k1(i,1)= h*dx(i);
%Evalua k1
    end

    dx=ec_es(Ts(j-1)+0.5*h,xu(j-1,:)+0.5.*k1(:,1)',u1(j-1),u2(j-1));
%Evalua la funcion en f(tn+0.5h,xn+0.5k1)
    for i=1:ne
        k1(i,2)= h*dx(i);
%Evalua k2
    end

    dx=ec_es(Ts(j-1)+0.5*h,xu(j-1,:)+0.5.*k1(:,2)',u1(j-1),u2(j-1));
%Evalua la funcion en f(tn+0.5h,xn+0.5k2)
    for i=1:ne
        k1(i,3)= h*dx(i);
%Evalua k3
    end

    dx=ec_es(Ts(j-1)+h,xu(j-1,:)+k1(:,3)',u1(j-1),u2(j-1));
%Evalua la funcion en f(tn+h,xn+k3)
    for i=1:ne
        k1(i,4)= h*dx(i);
%Evalua k4
    end

    %=====
    %Perfil del tiempo
    %=====
    Ts(j)=Ts(j-1)+ h;

    %=====
    %Perfiles de las variables de estado
    %=====

    xu(j,1)=xu(j-
1,1)+(1/6).* (k1(1,1)+k1(1,4))'+(1/3).* (k1(1,2)+k1(1,3))';
    xu(j,2)=xu(j-
1,2)+(1/6).* (k1(2,1)+k1(2,4))'+(1/3).* (k1(2,2)+k1(2,3))';
    xu(j,3)=xu(j-
1,3)+(1/6).* (k1(3,1)+k1(3,4))'+(1/3).* (k1(3,2)+k1(3,3))';
    xu(j,4)=xu(j-
1,4)+(1/6).* (k1(4,1)+k1(4,4))'+(1/3).* (k1(4,2)+k1(4,3))';
    xu(j,5)=xu(j-
1,5)+(1/6).* (k1(5,1)+k1(5,4))'+(1/3).* (k1(5,2)+k1(5,3))';
    xu(j,6)=xu(j-
1,6)+(1/6).* (k1(6,1)+k1(6,4))'+(1/3).* (k1(6,2)+k1(6,3))';

    end

    %=====
    %=====

```

%Condiciones finales se tendrá que definir una para cada caso (se obtienen de la integración de las variables de estado)

```
%=====
=====
```

```
lf(1)=0;
lf(2)=2*p22*xu(end,2);
lf(3)=2*p33*xu(end,3);
lf(4)=2*p44*xu(end,4);
lf(5)=0;
lf(6)=2*p66*xu(end,6);
```

```
%=====
=====
```

%Asignación de las condiciones finales en el vector de las variables adjuntas

```
%=====
=====
```

```
for i=1:ne
    lu(end,i)=lf(i);
end
```

```
%=====
%Integración RK4 para las variables adjuntas
%=====
```

```
for j=nstep+1:-1:2
```

```
    dlu=ec_adj(Tb(j),lu(j,:),xu(j,1),xu(j,2),xu(j,3),xu(j,4),xu(j,5),xu(j,6),
    u1(j),u2(j)); %Evalua la funcion en f(tn,xn)
```

```
        for i=1:ne
            ka(i,1)= h*dlu(i);
%Evalua k1
        end
```

```
        dlu=ec_adj(Tb(j)-0.5*h,lu(j, :)-
0.5.*ka(:,1)',xu(j,1),xu(j,2),xu(j,3),xu(j,4),xu(j,5),xu(j,6),u1(j),u2(j)
); %Evalua la funcion en f(tn+0.5h,xn+0.5k1)
```

```
        for i=1:ne
            ka(i,2)= h*dlu(i);
%Evalua k2
        end
```

```
        dlu=ec_adj(Tb(j)-0.5*h,lu(j, :)-
0.5.*ka(:,2)',xu(j,1),xu(j,2),xu(j,3),xu(j,4),xu(j,5),xu(j,6),u1(j),u2(j)
); %Evalua la funcion en f(tn+0.5h,xn+0.5k2)
```

```
        for i=1:ne
            ka(i,3)= h*dlu(i);
%Evalua k3
        end
```

```

        dlu=ec_adj(Tb(j)-h,lu(j,:)-
ka(:,3)',xu(j,1),xu(j,2),xu(j,3),xu(j,4),xu(j,5),xu(j,6),u1(j),u2(j));
%Evalua la funcion en f(tn+h,xn+k3)
    for i=1:ne
        ka(i,4)= h*dlu(i);
%Evalua k4
    end

%=====
%Perfil del tiempo
%=====

Tb(j-1)=Tb(j)- h;

%=====
%Perfiles de las variables de adjuntas
%=====

    lu(j-1,1)=lu(j,1)-(1/6).*(ka(1,1)+ka(1,4))'-
(1/3).*(ka(1,2)+ka(1,3))';
    lu(j-1,2)=lu(j,2)-(1/6).*(ka(2,1)+ka(2,4))'-
(1/3).*(ka(2,2)+ka(2,3))';
    lu(j-1,3)=lu(j,3)-(1/6).*(ka(3,1)+ka(3,4))'-
(1/3).*(ka(3,2)+ka(3,3))';
    lu(j-1,4)=lu(j,4)-(1/6).*(ka(4,1)+ka(4,4))'-
(1/3).*(ka(4,2)+ka(4,3))';
    lu(j-1,5)=lu(j,5)-(1/6).*(ka(5,1)+ka(5,4))'-
(1/3).*(ka(5,2)+ka(5,3))';
    lu(j-1,6)=lu(j,6)-(1/6).*(ka(6,1)+ka(6,4))'-
(1/3).*(ka(6,2)+ka(6,3))';

    end

%=====
%Actualización del control u(k+1)=u(k)-eps(dH/du)
%=====

    disp
('=====
=====');

%=====
%Actualización del control u1
%=====

    uln = u1 - 1e-8.*(2*r11.*u1 +
((v8.*xu(:,1)).*xu(:,3))./xu(:,4)).*(lu(:,1)-lu(:,2)));

    ulk = zeros (nstep+1,1);

```

```

for i=1:nstep+1
    if u1n(i)<0
        u1n(i)=0.6;
        ulk(i) = u1n(i);
    else
        if u1n(i)>1
            u1n(i) = 0.6;
            ulk(i) = u1n(i);
        else
            ulk(i) = u1n (i);
        end
    end
end

sul = sum (u1);
sul1 = sum(ulk);

error1=abs(sul-sul1);

u1= ulk;

disp ('error1');
disp (error1)

disp ('suma de u1')
disp (sul)

disp ('suma de u1n');
disp (sul1)

%=====
%Actualización del control u2
%=====

u2n = u2 - 1e-12.*(2*r22.*u2 - xu(:,6).*lu(:,6));

u2k = zeros (nstep+1,1);

for i=1:nstep+1
    if u2n(i)<0
        u2n(i)=0.65;
        u2k(i) = u1n (i);
    else
        if u2n(i)>0.695
            u2n(i) = 0.65;
            u2k(i) = u1n (i);
        else
            u2k (i) = u2n (i);
        end
    end
end
end

```

```

su2 = sum (u2);
su2k = sum(u2k);

error2=abs (su2-su2k);

u2=u2k;

disp ('error2');
disp (error2)

disp ('suma de u2');
disp (su2)

disp ('suma de u2n');
disp (su2k)

%=====
%Calculo del error total
%=====

errorT = error2 + error1;

disp ('errorT');
disp (errorT)

%=====
%Actualización de la iteración
%=====

nit = nit + 1;
disp ('nit');
disp (nit)

disp
('=====
=====');
end

disp(' - - - - ')
disp(' | RESULTADOS ')
disp(' - - - - ')

%=====
%Evaluación de las funciones objetivo
%La parte integral se evalua con un cumulative Simpson

```

```

%=====

%=====Cost Function=====
%           Arguments of the Cost Function

yx2=q22*xu(:,2).^2;
yx3=q33*xu(:,3).^2;
yx4=q44*xu(:,4).^2;
yx6=q66*xu(:,6).^2;

y1=r11*u1.^2;
y2=r22*u2.^2;

ar0=cumsimpson(Ts,yx2);
ar1=cumsimpson(Ts,yx3);
ar2=cumsimpson(Ts,yx4);
ar3=cumsimpson(Ts,yx6);

ar32=cumsimpson(Ts,y1);
ar33=cumsimpson(Ts,y2);

q22x2sqr=ar0(end);
q33x3sqr=ar1(end);
q44x4sqr=ar2(end);
q66x6sqr=ar3(end);

r11u1sqr=ar32(end);
r22u2sqr2=ar33(end);

%Cost Function
J=(p22*xu(end,2)^2+p33*xu(end,3)^2+p44*xu(end,4)^2+p66*xu(end,6)^2)+(q22x
2sqr+q33x3sqr+q44x4sqr+q66x6sqr+r11u1sqr+r22u2sqr2);

disp('J');
disp(J)

disp(' ----- ')
disp(' | GRÁFICOS | ')
disp(' ----- ')

%=====
==
%           Gráficas de las variables de estado
%=====
==

figure(1)
plot(Ts,xu(:,1))
xlabel('Unidad de tiempo, (días)')

```



```

ylabel('Número promedio de personas susceptibles')
legend('x_1');
title('Variable de Estado, Número promedio de personas susceptibles en un
tiempo t (x_1)')

```

```

figure(2)
plot(Ts,xu(:,2))
xlabel('Unidad de tiempo, (días)')
ylabel('Número promedio de personas infectadas')
legend('x_2');
title('Variable de Estado, Número promedio de personas infectadas en un
tiempo t (x_2)')

```

```

figure(3)
plot(Ts,xu(:,3))
xlabel('Unidad de tiempo, (días)')
ylabel('Número promedio de mosquitos portadores del virus')
legend('x_3');
title('Variable de Estado, Número promedio de mosquitos portadores del
virus en un tiempo t (x_3)')

```

```

figure(4)
plot(Ts,xu(:,4))
xlabel('Unidad de tiempo, (días)')
ylabel('Población total de mosquitos')
legend('x_4');
title('Variable de Estado, Población total de mosquitos en un tiempo t
(x_4)')

```

```

figure(5)
plot(Ts,xu(:,5))
xlabel('Unidad de tiempo, (días)')
ylabel('Número promedio de mosquitos inmaduros')
legend('x_5');
title('Variable de Estado, Número promedio de mosquitos inmaduros en un
tiempo t (x_5)')

```

```

figure(6)
plot(Ts,xu(:,6))
xlabel('Unidad de tiempo, (días)')
ylabel('Número promedio de criaderos')
legend('x_6');
title('Variable de Estado, Número promedio de criaderos en un tiempo t
(x_6)')

```

```

%=====
==
%                               Gráficas de las variables Adjuntas
%=====
==

```

```

figure(7)
plot(Tb,lu(:,1))
xlabel('Unidades de tiempo, (días)')

```

```
ylabel('Variable Adjunta, lambda_1')
legend('lambda_1');
title('Variables adjuntas, lambda_1')
```

```
figure(8)
plot(Tb,lu(:,2))
xlabel('Unidades de tiempo, (días)')
ylabel('Variable Adjunta, lambda_2')
legend('lambda_2');
title('Variables adjuntas, lambda_2')
```

```
figure(9)
plot(Tb,lu(:,3))
xlabel('Unidades de tiempo, (días)')
ylabel('Variable Adjunta, lambda_3')
legend('lambda_3');
title('Variables adjuntas, lambda_3')
```

```
figure(10)
plot(Tb,lu(:,4))
xlabel('Unidades de tiempo, (días)')
ylabel('Variable Adjunta, lambda_4')
legend('lambda_4');
title('Variables adjuntas, lambda_4')
```

```
figure(11)
plot(Tb,lu(:,5))
xlabel('Unidades de tiempo, (días)')
ylabel('Variable Adjunta, lambda_5')
legend('lambda_5');
title('Variables adjuntas, lambda_5')
```

```
figure(12)
plot(Tb,lu(:,6))
xlabel('Unidades de tiempo, (días)')
ylabel('Variable Adjunta, lambda_6')
legend('lambda_6');
title('Variables adjuntas, lambda_6')
```

```
%=====
==
%                Gráficas de los Controles
%=====
==
```

```
figure(13)
plot(Ts,u1)
xlabel('Unidades de tiempo, (días)')
ylabel('Control, u_1')
legend('u_1');
title('Control Profilaxis, u_1')
```

```
figure(14)
plot(Ts,u2)
```

```

xlabel('Unidades de tiempo, (días)')
ylabel('Control, u_2')
legend('u_2');
title('Control Mecánico, u_2')

```

```
end
```

```

%=====
==
%           Ecuaciones diferenciales para las variables de estado
%=====
==

```

```
function dx=ec_es(t,x,u1,u2)
```

```
global v0 v1 v2 v3 v4 v5 v6 v7 v8 v9 v10 v11
```

```
%Sistema de ecuaciones diferenciales de variables de estado
```

```
dx=zeros(6,1);
```

```

dx(1) = v7*v0 - v8*(1-u1)*x(1)*(x(3)/x(4)) - v7*x(1);
dx(2) = v8*(1-u1)*x(1)*(x(3)/x(4)) - (v7 + v9)*x(2);
dx(3) = (v10/v0)*x(2)*(x(4)-x(3)) - v2*x(3);
dx(4) = v1*x(5) - v2*x(4);
dx(5) = v3*x(4)-(x(5)*v3*x(4))/(v4*x(6)) - (v1+v6)*x(5);
dx(6) = v11*x(6)*(1-(x(6))/(v5)) - u2*x(6);

```

```
end
```

```

%=====
==
%           Ecuaciones para las variables adjuntas
%=====
==

```

```
function dl=ec_adj(t,l,x1,x2,x3,x4,x5,x6,u1,u2)
```

```
global v0 v1 v2 v3 v4 v5 v6 v7 v8 v9 v10 v11 q22 q33 q44 q66
```

```
%Sistema de ecuaciones diferenciales de variables adjuntas
```

```
dl=zeros(6,1);
```

```

dl(1) = -( ((v8*(1-u1)*x3)/(x4))*l(2) - l(1)) - v7*l(1) );
dl(2) = -( 2*q22*x2 - (v7+v9)*l(2) + ((v10*(x4-x3))/(v0))*l(3) );
dl(3) = -( 2*q33*x3+(v8*(1-u1)*x1)/(x4))*l(2) - l(1) - ((v10*x2)/(v0)
+ v2)*l(3) );
dl(4) = -( 2*q44*x4 + ((v8*(1-u1)*x1*x3)/(x4^2))*l(1) - l(2) ) +
(v10*l(3)*x2)/(v0) - v2*l(4) + v3*(1-(x5)/(v4*x6))*l(5) );
dl(5) = -( v1*l(4) - (v3*x4*l(5))/(v4*x6) - (v1+v6)*l(5) );

```

```
dl(6) = -( 2*q66*x6 +(v3*x4*x5*1(5))/(v4*x6^2) + v11*1(6) -
(2*v11*x6*1(6))/(v5) - u2*1(6));
```

```
end
```

```
function z=cumsimpson(x,y)
%{
  CUMSIMPSON Cumulative Simpson Numerical Integration.
  Z=CUMSIMPSON(Y) computes an approximation of the cumulative integral of
  Y
  using a Simpson rule with unit spacing between the data points in Y. To
  compute the integral for spacing different from one, multiply Z by the
  spacing increment. When Y is a matrix, the cumulative integral is
  computed over each column of Y.

  Z=CUMSIMPSON(X,Y) computes the integral with respect to the data in
  vector X. X need NOT be EQUALLY spaced but must have the same number of
  elements as Y. When Y is a matrix, Y must have as many rows as X has
  elements.

  Z has the same dimensions as Y.

  See also CUMSUM, CUMTRAPZ, QUAD, QUADV.
  D.C. Hanselman, University of Maine, Orono, ME 04469
  MasteringMatlab@yahoo.com
  Mastering MATLAB 7
  2005-09-30, 2006-02-21
%}
```

```
if nargin<2
  y=x;
  [ry,cy]=size(y);
  if ry==1
    x=1:cy;
  else
    x=1:ry;
  end
elseif nargin==2
  [ry,cy]=size(y);
else
  error('One or Two Inputs Are Required.')
end
if ndims(y)~=2
  error('N-dimensional Data is Not Supported.')
end
if min(size(x))>1
  error('X Must be a Vector.')
end
x=x(:); % make x a column
if ry==1 % y is a row vector, make it a column
  yisrow=true;
  y=y.';
  ry=cy;
```

```

        cy=1;
else
    yisrow=false;
end
if length(x)~=ry
    error('Length of X Must Match Length of Y or Rows of Matrix Y.')
end
if length(x)<3
    error('At Least 3 Data Points are Required.')
end
dx= repmat(diff(x),1,cy);
dy=diff(y);

dx1=dx(1:end-1,:);
dx2=dx(2:end,:);
dxs=dx1+dx2;
dy1=dy(1:end-1,:);
dy2=dy(2:end,:);

a=(dy2./(dx2.*dxs) - dy1./(dx1.*dxs))/3;
b=(dy2.*dx1./(dx2.*dxs) + dy1.*dx2./(dx1.*dxs))/2;
c=y(2:end-1,:);

i1=(a.*dx1-b).*dx1+c).*dx1; % left half integral
i2=(a.*dx2+b).*dx2+c).*dx2; % right half integral

z=zeros(ry,cy); % pure cumulative Simpson
z(2:2:end-1,:)=i1(1:2:end,:);
z(3:2:end,:)=i2(1:2:end,:);
z(end,:)=i2(end,:);
z=cumsum(z);

% z=[zeros(1,cy);... % original algorithm here has ~2X error
%   cumsum([i1(1,:); (i1(2:end,:)+i2(1:end-1,:))/2; i2(end,:)])];

if yisrow
    z=z.';
end

end

```