

**Review Article**

# Different optimization strategies for the optimal control of tumor growth

Sweilam NH<sup>1</sup>, Tharwat AA<sup>2</sup> and Abd El Moniem NK<sup>3\*</sup>

<sup>1</sup>Department of Mathematics, Cairo University, Faculty of Science, Giza, Egypt

<sup>2</sup>Department of Management and Marketing, Colleague of BA, American University at UAE

<sup>3</sup>Department of Information Technology, Cairo University, National Cancer Institute, Egypt

## Abstract

In this article different numerical techniques for solving optimal control problems is introduced, the aim of this paper is to achieve the best accuracy for the Optimal Control Problem (OCP) which has the objective of minimizing the size of tumor cells by the end of the treatment. An important aspect is considered, which is the optimal concentrations of drugs that not affect the patient's health significantly. To study the behavior of tumor growth, a mathematical model is used to simulate the dynamic behavior of tumors since it is difficult to prototype dynamic behavior of the tumor. A tumor-immune model with four components, namely, tumor cells, active cytotoxic T-cells (CTLs), helper T-cells, and a chemotherapeutic drug is used. Two general categories of optimal control methods which are indirect methods and direct ones based on nonlinear programming solvers and interior point algorithms are compared. Within the direct optimal control techniques, we review three different solutions techniques namely (i) multiple shooting methods, (ii) trapezoidal direct collocation method, (iii) Hermit- Simpson's collocation method and within the indirect methods we review the Pontryagin's Maximum principle with both collocation method and the backward forward sweep method. Results show that the direct methods achieved better control than indirect methods.

## Introduction

Mutations is the major consequence of abnormal behavior of the genetic material DNA. Mutations affect normal growth and division i.e., mutations cause uncontrollable growth of cells, these mutations are caused by two reasons which are the signals telling cells to begin dividing are left on continuously or growth suppressing signals telling cells not divide are turned off [19]. The process starts from an evolutionary process which may give rise to abnormal DNA when a cell duplicates its genome due to defects in tumor suppressor or DNA mismatch repair genes. Tumors releases hormones that alter body function, they can grow and interfere with the digestive, nervous and circulatory systems, they can also invade nearby tissues and successfully spread to other parts of the body and grow, these tumors are of malignant type, whereas benign tumors are not cancerous and not spread to other parts of the body.

The immune system is composed of a wide variety of cells with different functions, most important types of them are innate, or nonspecific and specific, or adaptive, immune response [1], the similarity between innate and adaptive responses is that both must contact with target tumor cells in order to be able to kill them, whereas the difference between them is that the innate cells are always on patrol, and

kill tumor cells not recognized them as self, so it is an early defense against pathogens. While adaptive, immune response must be primed to recognize antigen specific to the tumor cells. Cell-mediated immunity, also called cellular immunity, is mediated by T lymphocytes (also called T cells) are part of the adaptive immune response. as well as the Killer T cells are referred to as CTL "Cytotoxic T Lymphocyte" cells, or CD8+ T cells, which directly attack and eliminate infected cells, while, the CD4 helper T cells, assist other cells of the immune system during an infection.

The immune system alone usually fails to effectively fight the tumor for the following reasons:

1. The first is an insufficient immune response due to the tumor being poorly antigenic. This is described by Curilas [13] "too little of a good thing" Cytotoxic T cells are not sufficiently activated by the tumor, and therefore the response is minimal.
2. The second cause for failure is "too much of a bad thing" in which immunosuppressive factors damage otherwise capable immune system. Over the last few decades, the importance of the second paradigm has become clear; immune suppression is likely to be a significant factor when cancer is present in the host.

## More Information

**\*Address for Correspondence:** Abd El Moniem NK, Department of Information Technology, Cairo University, National Cancer Institute, Egypt, Email: nermeen2000@hotmail.com; nermeen.elzaweli@nci.cu.edu.eg

**Submitted:** 22 October 2019

**Approved:** 09 December 2019

**Published:** 10 December 2019

**How to cite this article:** Sweilam NH, Tharwat AA, Abd El Moniem NK. Different optimization strategies for the optimal control of tumor growth. Arch Cancer Sci Ther. 2019; 3: 052-062.

**DOI:** dx.doi.org/10.29328/journal.acst.1001010

**Copyright:** © 2019 Sweilam NH, et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

**Keywords:** Optimal control for differential equations; Tumor-immune model; Optimal control direct methods; Opti-mal control indirect methods; Nonlinear programming





Both mathematical modeling and experimental “clinical” results are used to explore the significance of tumor-immune interactions. Previous work has included models of T cells with interleukin-2 (IL-2) [24], transforming growth factor beta (TGF- $\beta$ ) [25], regulatory T cells (Tregs) [28], and natural killer cells [14].

Several answers to the question how to choose best possible dose to reduce harm to healthy cells while beat cancer are introduced. These answers was introduced before based on empirical methods, empirical methods depend basically on drug holidays, which are rest periods give the health cells the opportunity to recover from the toxic attack of the drug these periods are specified by try and error methods. Apart from the empirical methods the mathematical modeling and optimal control can answer this question, for more details see [35].

In this article a base line model [37] is used based on tumor-immune model. A comparative study between the results of the Computation method used in our baseline article [37] which uses indirect method resulted in two point boundary value problem (TPBVP) and solved by collocation method is compared with another method also under the category of indirect method which also resulted in TPBVP but solved by other method named forward backward sweep method (FBWM), results of the base line article [37] also compared with other three different computational methods under the category of direct methods.

Previous comparative studies between direct and indirect control given in [38], where this paper two different approaches (direct and indirect) for the numerical solution of fractional optimal control problems (FOCPs) based on a spectral method using Chebyshev polynomial are presented. Moreover, in [18] a comparative study between singular arc method (without considering inequality constraints) as indirect method and three different direct methods namely Hermite-Simpson’s collocation method, 5<sup>th</sup> degree Gauss-Lobatto collocation method, Radau Psuedospectral collocation method using GOPP [34]. In [9] a comparative study between dynamic programming, indirect methods and direct methods is presented.

This article is organized as follows: in section 2 we consider a tumor growth model exhibiting the effect of tumor-immune interaction with chemotherapeutic. The model construction and assumptions is introduced in “Mathematical Model” section. In section 3, the optimization stratigetis is presented. In section 4, the solution of OC problem is discussed. In section 5 a comparative study for solving OC problem is presented. The conclusion is given is section 6.

**Mathematical modeling and optimal control problem for a tumor-immune system**

In this section, a mathematical model problem is introduced, for more details see [37] as a description of the

phenomena of tumor-immune interaction and the prediction of the outcome of the application of chemotherapy regimen is examined by using optimal control [5,33]. We can consider optimal control problem as a type of optimization problem where the objective is to determine the inputs (equivalently, the trajectory, state or path), the control inputs (equivalently, the trajectory, state or path), the control input  $u \in (t) R^m$ , the initial time  $t_0 \in R$  and the terminal time,  $t_f \in R$  (where  $t \in [t_0, t_f]$  the independent variable) of the dynamical system [34] optimize (i.e., minimize or maximize) a specified objective function (performance index) while satisfying any constraints on the motion of the system.

the Objective function (performance index) is represented by:

$$J(u^*) = \min \{ J(u(t)) : u \in U \},$$

Where

$$J(u(t)) = \int_0^{t_f} [B_1 T + \frac{1}{2} B_2 u^2] dt \tag{2.1}$$

where  $B_1, B_2$  are positive constants representing the weights of the terms. The first term represents the tumor cell populations and the second term represents the harmful effects of drug on body. The square of the control variable ( $u^2(t)$ ) reflects the severity of the side effects of the drug imposed, for more details see [22,42] and the references cited therein. When chemotherapeutic drugs are administered in high dose, they are toxic to the human body, which justifies the quadratic terms in the functional. So the functional given in Eq. (2.1) should be minimized.

The dynamical system is defined by a set of ordinary differential equations (ODE’s):

$$\begin{aligned} \frac{dT}{dt} &= r_1 T(1 - p_1 T) - \alpha T I_H - q_1 D T, \\ \frac{dI_H}{dt} &= \beta I_H I_R - \alpha_2 T I_H - d I_H - q_2 D I_H, \\ \frac{dI_R}{dt} &= r_2 I_R(1 - p_2 I_R) - \beta I_H I_R - q_3 D I_R, \\ \frac{dD}{dt} &= u(t) - \gamma D, \end{aligned}$$

$$\text{satisfying } T(0) \geq T_0, I_H(0) \geq 0, I_R(0) \geq 0, D(0) \geq 0$$

Where

- $T(t)$  is the numbers of tumor cell.
- $I_H(t)$  is the active CTL cells (hunting CTL cells).
- $I_R(t)$  is the helper T-cells (resting T-cells).

$D(t)$  is the density of chemotherapeutic drug at time  $t$ .

The tumor-immune model of the base line paper is originally developed by de Pillis and Radunskaya [31], an optimal control for this model is introduced in [15]. This

model considers interactions between tumor cell population and two types of immune cell populations which are helper (resting) T-cells and active (hunting) CTL cells. The damage done to each cell due to chemotherapy is subtracted from each cell population. According to [37] which is based on [14,15] the model assumptions are as follows:

- Hunting CTL cells are capable of killing tumor cells, the effect of CTL on tumor cells is represented by both the terms:

1.  $\alpha T I_H$  which represents the loss of tumor cells it is proportional to the product of the densities of tumor cells and active CTL cell (hunting) which is subtracted in the equation represents rate of change of tumor population  $\frac{dT}{dt}$
2.  $\alpha_2 T I_H$  is the loss in the active CTL cells due to encounters of tumor cells which is assumed to be proportional to the product of the densities of tumor cells and active CTL cells. Which is subtracted in the equation represents rate of change active CTL population  $\frac{dI_H}{dt}$

- In the absence of active (hunting) CTL cells and chemotherapeutic drug both the tumor cell population and helper T-cell population are assumed to grow logistically.
- Mass-action kill rate assumes that all immune cells are similarly prone to communicate with any tumor cell: it assumes spatial homogeneity.
- $I_H$  helper (resting) T-cells are not able to attack and destroy tumor cells directly but it convert CTLs into active (hunting) CTL helper either by releasing cytokines (interleukin-2) or by direct contact with them.
- $I_R$  active (hunting) CTL cells have the role of attacking, destroying, or ingesting the tumor cells.
- Chemotherapeutic drug destroys tumor cells as well as helper T-cells and active CTL cells; that is, chemotherapeutic drug has a negative effect on both tumor cells and immune cells and this negative effect is expressed by subtracting this destroy from each cell population equation.

The model parameters are described as follows, for more details see [37] (Table 1).

### Optimization strategies for OCP

The problem formulation which is given in (2.1),(2.2), based on open loop control (also referred to it in literatures as dynamic optimization ), i.e., feedback is not utilized (control input  $u(t)$  is independent of state). There are three main approaches to numerically solve continuous time OCP:

**Table 1:** The model parameters.

Parameter	Description	Estimated value
$r_1$	per capita growth rates of tumor cells	0.44/day
$r_2$	per capita growth rates of helper (resting) T-cells,	0.0246/day
$a_1$	Rate of loss of tumor cells due to encounter with the active (hunting) CTL cells.	$1.101 \times 10^{-7}$ /cells/day
$a_2$	Rate of loss of active (hunting) CTL cells due to encounter with the tumor cells.	$3.422 \times 10^{-10}$ /cells/day
$\beta$	rate of conversion of helper (resting) T-cells to active (hunting) CTL cells.	$3.422 \times 10^{-10}$ /cells/day
$\gamma$	per capita decay rate of the chemotherapeutic drug;	0.01/day
$p_1$	Reciprocal carrying capacities for tumor cells.	$5 \times 10^{-9}$ /cells
$p_2$	Reciprocal carrying capacities for helper (resting) T-cells.	$1 \times 10^{-10}$ /cells/day
$q_1$	Response coefficients to the chemotherapy drug for tumor cells.	0.08/day
$q_2$	Response coefficients to the chemotherapy drug for active (hunting) CTL cells.	$2 \times 10^{-11}$ /cells/day
$q_3$	Response coefficients to the chemotherapy drug for helper (resting) T-cells.	$1 \times 10^{-5}$ /day
$d$	Per capita decay rate of active (hunting) CTL cells.	0.0412/day

1. **Dynamic programming methods:** The optimal criterion in continuous time is based on the Hamilton-Jacobi-Bellman partial differential equation, for more details see [4], which is not in the scope of this paper.
2. **Indirect methods:** It take an approach optimize first then discretize, also relies on Pontryagin’s Maximum Principle (PMP), for more details see [32]. Typically, the optimal control problem is turned into TPBVP containing the same mathematical information as the original one by means of necessary conditions of optimality, for more details see [3], [38] and [39].
3. **Direct methods:** It take an approach discretize first then optimize, it can be applied without deriving the necessary condition of optimality. Direct methods are based on a finite dimensional parameterization of the infinite dimensional problem. The finite dimensional problem is typically solved using an optimization method, such as nonlinear programming (NLP) techniques. NLP problems can be solved to local optimality relying on the so called Karush-Kuhn-Tucker conditions (KKT), for more details see [3], if we are using KKT we can claim the first-order conditions of optimality. These conditions were first derived by Karush in 1939 [23], and later, in 1951, independently by Kuhn and Tucker [26].

### Indirect methods

In indirect methods the necessary optimality conditions is derived by using Pontryagin’s maximum principle [32], by considering a simple optimal control problem.

$$\min J(t, x(t), u(t)) = \Phi(t_f, x(t_f)) + \int_{t_0}^{t_f} L(t, x(t), u(t)) dt,$$

subject to

$$\dot{x}(t) = f(t, x(t), u(t)), \quad (\text{dynamic constraints})$$

$$\Psi(x(t_f)) = 0, \quad (\text{final boundary condition})$$



where  $J$  is the objective function in Bolza form, where Bolza form is the sum of the Mayer term  $\Phi(t_f, x(t_f))$ , and the Lagrange term,

$$\int_{t_i}^{t_f} L(t, x(t), u(t)) dt,$$

the basic principle of Pontryagin's maximum principle is defining the Hamiltonian, where Hamiltonian is a scalar function  $\mathcal{H}: [t_i, t_f] \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_\lambda} \rightarrow \mathbb{R}$ , defined

$$\mathcal{H}(t, x(t), u(t), \lambda(t)) = L(t, x(t), u(t)) + \lambda^T(t) f(t, x(t), u(t)),$$

by defining the auxiliary function  $\phi: [t_i, t_f] \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_\lambda} \rightarrow \mathbb{R}$ , by:

$$\phi(t, x) = \Phi(t, x(t)) + \mu^T \Psi(x(t)),$$

by setting first variation of the Lagrangian to zero  $\delta \mathcal{L} = 0$ , where  $\mathcal{L}: [t_i, t_f] \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_\lambda} \times \mathbb{R}^{n_\mu} \rightarrow \mathbb{R}$  is defined as

$$\begin{aligned} \mathcal{L}(t, x(t), \mu, \lambda(t)) &= (\Phi(t_f, x(t_f)) + \mu^T \Psi(x(t_f))) + \\ &\int_{t_i}^{t_f} L(t, x(t), u(t)) + \lambda^T(t) (f(t, x(t), u(t)) - \dot{x}) dt, \end{aligned} \quad (3.2)$$

i.e., Integrating by parts the last term on the right side in Eq.(3.2), it yields:

$$\begin{aligned} \mathcal{L}(t, x(t), \mu, \lambda(t)) &= [\phi(t, x(t))]_{t=t_f} - \lambda^T(t_f) x(t_f) + \\ &\lambda^T(t_i) x(t_i) + \int_{t_i}^{t_f} [(\mathcal{H}(t, x(t), u(t), \lambda(t)) + \dot{\lambda}^T(t) x(t))] dt. \end{aligned}$$

We can conclude that the necessary optimality conditions for the unconstrained optimal is derived which stated as:

$$\frac{d\lambda}{dt} = -\frac{\partial \mathcal{H}}{\partial x}, \text{ adjoint equations,}$$

$$\frac{\partial \mathcal{H}}{\partial u} = 0, \text{ control equations,}$$

$$\lambda(t_f) = \left[ \frac{\partial \phi}{\partial x} \right]_{t=t_f}, \text{ transversality conditions,}$$

$$\lambda(t_i) = 0, \text{ transversality conditions.}$$

They are referred to as the Euler-Lagrange equations.

For more details see [11], which will be solved numerically. Many numerical methods which are based on the Euler-Lagrange differential equation (EL-DEQ) are available to solve the TPBVP. One may classify these numerical methods according to the particular approach used in [2].

In this article we introduce two approaches namely:

- Indirect Collocation Method.
- Forward-Backward Sweep Method (FBSM).

The purpose of examining two methods is to choose the method with the higher accuracy and set this method as the base method to compare against it. In the following we explained one of the most important methods for solving resultant system of the differential equations.

**Indirect collocation method [36]:** The Indirect Collocation Method is the merit of the techniques for solving TPBVP, finite difference method with continuous extension as well as collocation method are the mathematical tools for this method. As a difference methods it is based on Implicit Runge-Kutta (IRK) method which is equivalent to collocation method according to the following theorem.

Referring to the theorem of Guillou & Soule'1969, Wright 1970 which states that:

The collocation method defined as:

given  $s$  positive integer and  $c_1, \dots, c_s$  distinct real numbers (typically between 0 and 1), the corresponding collocation polynomial  $u(x)$  of degree  $s$  is defined by

$$\begin{aligned} u(x_0) &= y_0 && \text{(initial value),} \\ u(x_0 + c_i h) &= f(x_0 + c_i h, u(x_0 + c_i h)), i=1, \dots, s. \end{aligned}$$

the numerical solution is given

$$y_1 = u(x_0 + h)$$

is equivalent to the  $s$ -stage IRK-method which is defined as

$$k_i = f(x_0 + c_i h, y_0 + h \sum_{j=1}^s a_{ij} k_j), \quad i=1, \dots, s,$$

$$y_1 = y_0 + h \sum_{i=1}^s b_i k_i.$$

with coefficients

$$a_{ij} = \int_0^{c_i} \ell_j(t) dt, \quad b_j = \int_0^1 \ell_j(t) dt, \quad i, j=1, \dots, s,$$

where the  $\ell_j(t)$  are the Lagrange polynomials

$$\ell_j(t) = \prod_{k \neq j} \frac{(t - c_k)}{(c_j - c_k)}.$$

for theorem proof see [21].

**Forward-backward sweep method [27]:** In FBSM the initial value problem of the state equation is initialized by using an estimate for the control and costate variables and solved forward in time. Then the costate final value problem is solved backwards in time. An early reference to a technique that has the forward-backward flavor is [30]. FBSM method can be summarized by the following algorithm.

Information about convergence and stability of Runge-Kutta 4 ODE's solver can be found in [20].

<b>Algorithm 1:</b> FBSM algorithm.
Notations: let $\bar{x} = (x_0, \dots, x_N)$ and $\bar{\lambda} = (\lambda_1, \dots, \lambda_N)$ are the vector approximations for the state and adjoint.
Make an initial guess for $\bar{u}$ over the interval.
1: while not converged do
2: Using the initial condition $x_0 = x(t_0)$ and the values for $\bar{u}$ , solve $\bar{x}$ forward in time according to its differential equation in the optimality system using Runge-Kutta 4 ODE's solver
3: Using the transversality condition $\lambda_N = \lambda(t_f) = 0$ and the values for $\bar{u}$ and $\bar{x}$ , solve $\bar{\lambda}$ backward in time according to its differential equation in the optimality system using Runge-Kutta 4 ODE's solver.
4: Update $\bar{u}$ by entering the new values $\bar{x}$ and $\bar{\lambda}$ into the calculations of the optimal control.
5: Check convergent
6: end while

### Direct methods

Sequential and simultaneous, are the two main classes of direct methods: for more details see [41]. Sequential methods only parameterize the control while simultaneous methods parameterize both the state and control.

The differences on how we discretize the problem and how the continuity between discretized intervals are defined result in different transcription method. In the next sections different methods depend on different discretization schemes and different continuity conditions are presented namely multiple shooting method, trapezoidal direct collocation and Hermite Simpson's direct collocation.

**Multiple shooting method:** We start with the single shooting method, since single shooting is a special case of multiple shooting [12].

#### Step1: Transcription:

In single shooting system the time interval  $[t_0, t_f]$  is divided into equal sub intervals (segments)  $[t_i, t_{i+1}]$  such that  $0 = t_0 < t_1 < \dots < t_N = t_f$ , where  $N$  is the total number of sub intervals.

Then the control vector is transformed into a parameterized finite dimensional control vector  $u(t, q)$  that depends on the finite dimensional parameter vector  $q \in \mathbf{R}^N$  there are several parametrization schemes for more details see [3,8], we assume piece wise constant control:

$$u(t) := u_k \quad \text{for } t \in [t_k, t_{k+1}), k = 0, \dots, N - 1.$$

#### 3. The initial value problem (IVP)

$$\dot{x}(t) = f(x(t), u(t, q), t), \quad x(t_0) = x_0, \quad t \in [t_0, t_f],$$

which is solved to yield the state vector  $x(t, q)$  in the time interval  $[t_0, t_f]$ .

4. The integral of the objective function is calculated together with the initial value problem solution (by using a quadrature formula). The Lagrangian part of the cost function is evaluated on each interval independently.

### Optimization

1. Due to the numerical simulation the model equations are eliminated. The path constraints are also discretized. Thus the optimal control problem Eq.(3.1) is rewritten as:

$$\min J = \varphi(x(t_0), \mathbf{q}, t_0, x(t_f), \mathbf{q}, t_f) + \int_{t_0}^{t_f} g(x(t, \mathbf{q}), u(t, \mathbf{q})) dt,$$

subject to

$$C(x(t_i, \mathbf{q}), u(t_i, \mathbf{q})) \leq 0,$$

$$E(x(t_0, \mathbf{q}, t_0, x(t_f), \mathbf{q}, t_f)) = 0.$$

Which is an NLP problem that is solved using the Interior Point Method (IP) in this paper.

**Continuity constraints:** Since the simulation is done over the whole time horizon this method does not have continuity constraints.

**Accuracy:** Determine the accuracy of the finite-dimensional approximation and if necessary repeat the transcription and optimization steps (i.e. go to step 1).

single shooting method has the following drawbacks:

- Convergence of the NLP solution is slow, because of the high nonlinear dependence of the objective and constraint functions on the variable  $u$ ,
- NLP solver cannot initialize with an initial guess for  $x_1, \dots, x_N$  even it is available
- Parallel evaluation of the states and objective functions is impossible because of the recursive elimination for both states and objective functions

**Direct multiple shooting method [10]:** This method combines the advantages of simultaneous methods like collocation method with the main advantages of the single shooting method, so that it is sometimes called a hybrid method [17].

We follow the same steps in single shooting method except that we solve ODE's in each interval  $[t_i, t_{i+1}]$  independently, starting with an artificial initial value  $S_i$ :

$$\dot{x}_i(t) = f(x_i(t), q_i), \quad t \in [t_i, t_{i+1}],$$

$$x_i(t_i) = s_i.$$

Solving these initial value problems numerically, we can obtain trajectory pieces  $x_i(t; s_i; q_i)$ , where the extra arguments after the semicolon are introduced to denote the dependence on the interval's initial values and controls. Simultaneously with the decoupled ODE's solution, we also numerically compute the integrals is numerically computed by:

$$l_i(s_i, q_i) := \int_{t_i}^{t_{i+1}} L(x_i(t; s_i; q_i), q_i) dt.$$

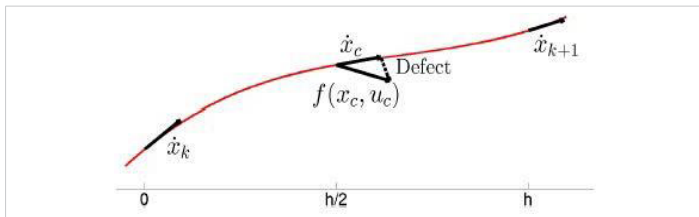


Figure 3.1: Collocation Defect Constraints.

**Continuity constrains:** To enforce continuity between discretized intervals, the following NLP constraints is added at the interface of each sub-interval, this constraint is formed such that the propagated (as an illustration for propagation means consider a cannon be aimed such that the cannonball hit its target i.e., shoot) or integrated value of the state from the previous phase match the value of the state at the current state. The continuity depends on propagation which is approximate because propagation is using algebraic formulas based on numerical integration schemes (or discretization schemes).

So Continuity Constrains is defined as:

$$s_{i+1} = x_i(t_{i+1}; s_i, q_i)$$

We have the following NLP, but contains the extra variables  $s_p$  and has a block sparse structure.

$$\begin{aligned} \min \quad & J = \varphi(x(t_f), \mathbf{q}_0, \mathbf{s}_0, t_0, x(t_f), \mathbf{q}, t_f) + \sum_{i=0}^{N-1} g(x(t_i), \mathbf{q}_i, \mathbf{s}_i), u(t_i, \mathbf{q}_i, \mathbf{s}_i)), \\ \text{subject to} \quad & \\ & s_0 - x_0 = 0, (\text{initial value}), \\ & s_{i+1} - x_i(t_{i+1}; \mathbf{s}_i, \mathbf{q}_i) = 0, (\text{continuity}), \\ & C(x(t_i), \mathbf{q}_i, \mathbf{s}_i), u(t_i, \mathbf{q}_i, \mathbf{s}_i) \leq 0, (\text{discretized path constraints}), \\ & E(x(t_0), \mathbf{q}_0, \mathbf{s}_0, t_0, x(t_f), \mathbf{q}_N, \mathbf{s}_N, t_f) = 0, ((\text{terminal constraints}). \end{aligned}$$

**Direct collocation:** Problem size, non-linearity and sparsity of the NLP resulting from direct transcription methods are the prices of moving from one method to another. In single shooting the NLP is highly nonlinear while its size is small. Multiple shooting is less nonlinear but larger and sparser. The direct collocation goes further in the same direction as it is less nonlinear, sparser but even larger, direct collocation method is introduced by Dickmanns [16] as a direct transcription for solving optimal control problems. It has the fundamental steps of direct transcription method, but differs slightly.

It similarly discretizes the state trajectory  $[t_0, t_f]$  into equal sub intervals (segments)  $[t_i, t_{i+1}]$ , such that  $0 = t_0 < t_1 < \dots < t_N = t_f$  where  $N$  is the total number of sub intervals.

But it differs at

It further divide the interval  $[t_i, t_{i+1}]$  into  $K$  sub intervals  $[\tau_j, \tau_{j+1}]$ , where

$$\tau_j = t_i + h_i \alpha_j, \quad j = 1, \dots, K, \quad h_i = t_{i+1} - t_i, \quad (3.3)$$

It employs an interpolating function to approximate the state of the system. Usually the interpolating function is polynomial.

Polynomials consider the following  $K^{\text{th}}$ -degree piece wise polynomial:

$$X(t) \approx \sum_{k=0}^K a_k (t - t_i)^k, \quad t \in [t_i, t_{i+1}], \quad (3.4)$$

suppose further that the coefficients  $(a_0, \dots, a_K)$  of the piece wise polynomial are chosen to match the value of the function at the beginning of the step, i.e.,

$$X(t_i) = x(t_i).$$

finally, suppose we choose to match the derivative of the state at the points defined by Eq.(3.3), i.e.,

$$\dot{X}(\tau_j) = f(x(\tau_j), \tau_j), \quad j = 1, \dots, K. \quad (3.5)$$

Eq.(3.5) is called collocation condition because the approximation to the derivative is set equal to the right-hand side of the differential equation evaluated at each of the intermediate points  $(\tau_1, \dots, \tau_K)$ .

- By setting  $K = 2$  in Eq.(3.4) we get the quadratic interpolation polynomial which results in trapezoidal method.
- By setting  $K = 3$  in Eq.(3.4) results in Hermite-Simpson's method.

Trapezoidal method and Hermite-Simpson's method are the two direct collocation methods we consider in this article.

**Defects constraints:** The difference between the first derivative of the interpolating polynomial at the midpoint of a segment and the first derivative calculated from the equations of motion at the segment midpoint is used as the defect. To have a good approximation of the actual states the defects must approach zero and thus interpolating polynomial is a good approximation, the defect is written in the following form:

$$\zeta_j = \dot{X}(\tau_j) - f(x(\tau_j), \tau_j), \quad (3.6)$$

the defect constraint for the trapezoidal method (by setting

$$P(x) = x(\tau_j) - \frac{h}{2}(f_j + f_{j+1}) \text{ and } x_{i+1} = x(\tau_{j+1}) \text{ at Eq.(3.10))}$$

is defined as:

$$\zeta_j(\tau_j) = x(\tau_{j+1}) - x(\tau_j) - \frac{h_j}{2}(f_j + f_{j+1}) = 0$$

the defect constraint for the Hermite-Simpson's method (by setting  $P(x) = x(\tau_j) - \frac{h_k}{6}(f_j + 4\bar{f}_j + f_{j+1})$  and  $x_{i+1} = x(\tau_{i+1})$ )

at Eq.(3.6) is defined as:

$$\text{let } \bar{\tau}_j = (\tau_j + \tau_{j+1}) / 2, \text{ and } \bar{u}(\bar{\tau}_j) = (u(\tau_j) + u(\tau_{j+1})) / 2,$$

$$\bar{x}(\bar{\tau}_j) = \frac{1}{2}[x(\tau_j) + x(\tau_{j+1})] + \frac{h_k}{8}(f_j - f_{j+1}),$$

$$\bar{f}_j = f[\bar{x}(\bar{\tau}_j), \bar{u}(\bar{\tau}_j), p, \bar{\tau}_j],$$

$$\zeta(\tau_j) = x(\tau_{j+1}) - x(\tau_j) + \frac{h_k}{6}(f_j + 4\bar{f}_j + f_{j+1}) = 0.$$

- The quadrature rule used for integration is consistent with the numerical method used for solving the differential equation. If one is using a Runge-Kutta method for solving the differential equation, the cost would also be approximated using Runge-Kutta integration. In the case of an Hermite-Simpson's collocation method, the integration rule is Hermite-Simpson collocated quadrature rule [33].
- Thus the optimal control problem Eq.(3.1) is rewritten as: let  $s_i$  be values of state vector at grid point,  $s_i^*$  which represent the states at the collocation points in each subinterval.

$$\min J = \varphi(x(t_0), \mathbf{q}_0, \mathbf{s}_0, t_0, x(t_f), \mathbf{q}, t_f) + \sum_{i=0}^{N-1} g(x(t, \mathbf{q}_i, \mathbf{s}_i, \hat{s}_i), u(t, \mathbf{q}_i, \mathbf{s}_i, \hat{s}_i)),$$

subject to

$$s_0 - x_0 = 0, \text{ (initial value),}$$

$$s_{i+1} - x_i(t_{i+1}; s_i, \hat{s}_i, \mathbf{q}_i) = 0, \text{ (continuity),}$$

$$C(x(t_i, \mathbf{q}_i, \mathbf{s}_i, \hat{s}_i), u(t_i, \mathbf{q}_i, \mathbf{s}_i, \hat{s}_i)) \leq 0, \text{ (discretized path constraints),}$$

$$E(x(t_0), \mathbf{q}_0, \mathbf{s}_0, t_0, x(t_f), \mathbf{q}_N, \mathbf{s}_N, t_f) = 0. \text{ (terminal constraints).}$$

### Solution of the NLP problem

Numerical methods for solving NLPs fall into categories: gradient based (local) methods and heuristic (global) methods, gradient based (local) approach is now described in the following algorithm (the heuristic method is out of the interest of this article, for more details see [40]).

The convergence may depend on a given tolerance or depends on no further change in the objective function after several iterations. There are many ways to modify

Algorithm 2: Gradient Based Algorithm.
Notations: Let $z$ is the unknown decision vector, $k$ is the iteration counter, $p$ is the search direction along which to change the current value $z_k$ , $\alpha_k$ is the magnitude of the change in $z_k$
1: while not converged do.
2: Steps are taken in a certain direction i.e., the $k^{\text{th}}$ iteration, a search direction is $p_k$ and a step length, $\alpha_k$ are determined. The update from $z_k$ to $z_{k+1}$ has the form: $z_{k+1} = z_k + \alpha p_k$
3: The objective function is evaluated, in case of minimization, the search direction is chosen to sufficiently decrease the objective function in the form. $f(z_{k+1}) \leq f(z_k) + K\alpha_k \nabla J^T(z_k) p_k$ ,
4: If the objective function improves take another step at the same direction else change the direction, step size or both.
5: end while.

direction selection and step size, and this leads to many different algorithms. Some of common ones included Steepest Descent Methods, Conjugate Direction Method, Simplex Method, Interior Point Method (IP) and Sequential Quadratic Programming, more details on the IP methods will be given here.

**IP Method:** it is typically eliminate constraints of the form  $\underline{h} \leq h(x) \leq \bar{h}$  by introducing slack variables additional equality constraints. After locating an interior point, i.e., a point where  $\underline{x} < x < \bar{x}$  holds with strict inequality, which may require reformulating some decision variables as parameters, they formulate a barrier problem of the form:

$$\min f(x) - \mu \sum_{i=1}^{n_x} \log(\bar{x}_i - x_i) + \log(x_i - \underline{x}_i)$$

Subject to

$$g(x) = 0$$

Sequences of barrier problems for increasing values of the barrier parameter  $\mu$  are then solved with Newton-type methods, for more details see [7,40].

### Accuracy

Since the solution obtained from the NLP problem is a discrete set of numbers we don't know what happen between the discrete points, a continuous approximation to the discrete solution resulted from the NLP is needed, i.e., representing the solution (interpolation) [6].

**Spline representation:** To get the solution as continuous approximation  $[y^*(t), u^*(t)]$  from the NLP solution we use spline representation (where spline is a sequence i.e., whole collection of polynomial) defined as follows:

$$\begin{bmatrix} y(t) \\ u(t) \end{bmatrix} = \begin{bmatrix} \tilde{y}(t) \\ \tilde{u}(t) \end{bmatrix} = \sum_{i=1}^{n_1} \alpha_i \beta_i(t), \tag{3.7}$$

where  $n_1 = 2M$ ,  $M$  is the number of mesh points,  $y(t)$ ,  $u(t)$  are the true state and control respectively,  $y^*(t)$ ,  $u^*(t)$  are the approximate state and control respectively, the functions  $\beta_i(t)$  form a basis for  $C^0$  or  $C^1$  cubic B-splines with  $n_1 = 2M$ , where  $M$  is the number of mesh points. The coefficients  $\alpha_i$  in the state or control variable representation which is defined by different Interpolation of discrete solution depending on the discretization method, the spline approximation Eq. (3.7) must match the state at the grid points i.e.,

$$\tilde{y}(t) = y_k, \quad \text{for } k = 1, \dots, M,$$

the derivative of the spline approximation must match the right-hand side of the differential equations,

$$\frac{d}{dt} \tilde{y}(t) = f_k, \quad \text{for } k = 1, \dots, M,$$

we require the spline approximation in Eq.(3.7) to match the control at the grid points, i.e.,

$$\tilde{u}(t_k) = u_k, \quad \text{for } k = 1, \dots, M,$$

the state variable is  $C^1$  cubic function whereas the control variable is  $C^0$  linear or quadratic functions depending on the discretization method. Since both  $\tilde{y}(t), \tilde{u}(t)$  are approximate to the true solution the degree to which this approximation approximate the true solution need to be determined which is known as discretization error and relative local error.

**Discretization error:** by assuming that the computed control is correct and assume the spline solutions  $\tilde{y}(t), \tilde{u}(t)$  produced from the NLP, and the single interval  $t_k \leq t \leq t_k + h_k$ , from the state equation we can define

$$y(t_k + h_k) = y(t_k) + \int_{t_k}^{t_k+h_k} \dot{y} dt = y(t_k) + \int_{t_k}^{t_k+h_k} f(y, u, t) dt,$$

where  $y$  and  $u$  are the true state and control values, we can consider the approximation:

$$\hat{y}(t_k + h_k) = y(t_k) + \int_{t_k}^{t_k+h_k} f[\tilde{y}(t), \tilde{u}(t), t] dt, \quad (3.8)$$

from Eq.(3.8) we can define the discretization error on the  $k^{th}$  mesh iteration as:

$$\eta_k = \max_i \{a_i |\tilde{y}_i(t_k + h_k) - \hat{y}_i(t_k + h_k)|\},$$

For  $i = 1, 2, \dots, n$ , where the weights  $a_i$  are chosen to appropriately normalize the error.

**Relative local error:** is the maximum relative error over all components  $i$  in the state equations  $y' = f$  evaluated in the interval  $k$ , and is defined as

$$\epsilon_k \approx \max_i \frac{\eta_{i,k}}{(w_i + 1)},$$

where the scale weight  $w_i = \max_{k=1}^M [|\tilde{y}_{i,k}|, |\dot{\tilde{y}}_{i,k}|]$ , defines the maximum value for the  $i^{th}$  state variable or its derivative over the  $M$  grid points in the phase. An equivalent form can be defined as follows the absolute local error on a particular step by

$$\eta_{i,k} = \int_{t_k}^{t_{k+1}} |\varepsilon_i(s)| ds, \quad (3.9)$$

Where  $\varepsilon(t) = \dot{\tilde{y}}(t) - f[\tilde{y}(t), \tilde{u}(t), t]$  defines the error in the differential equation as a function of  $t$ . An accurate estimate for the integral in Eq.(3.9) is evaluated by using a standard quadrature method, (i.e., Romberg quadrature algorithm).

## Comparative Study and Results

In this part we compare the results from the our base line article [37], with another indirect method solved by sweep method [27] and another three direct method namely direct

multiple shooting, trapezoidal direct collocation and Hermite-Simpson's direct collocation.

In all the previous figures the direct methods achieve better results than indirect methods. The values of controls and number of tumor cells made the big differences in the objective function of the theses methods. The number of tumor

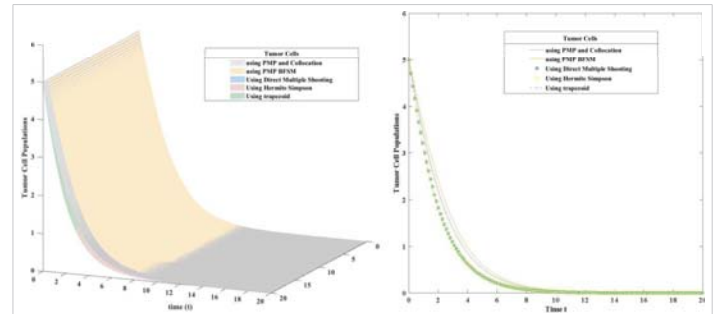


Figure 4.1: Tumor Cell population using 3D plot (left figure) and 2D plot (right figure).

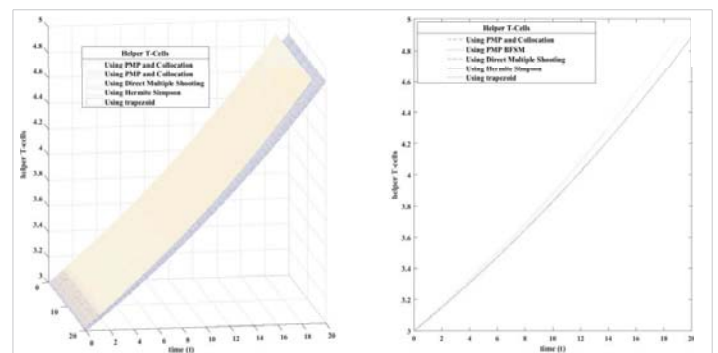


Figure 4.2: Helper T cells using 3D plot (left figure) and 2D plot (right figure).

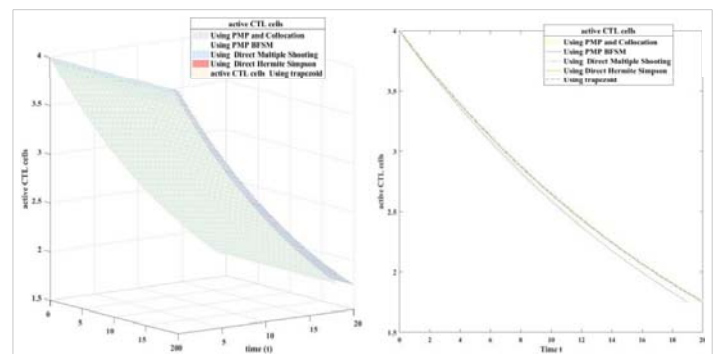


Figure 4.3: Active CTL using 3D plot (left figure) and 2D plot (right figure).

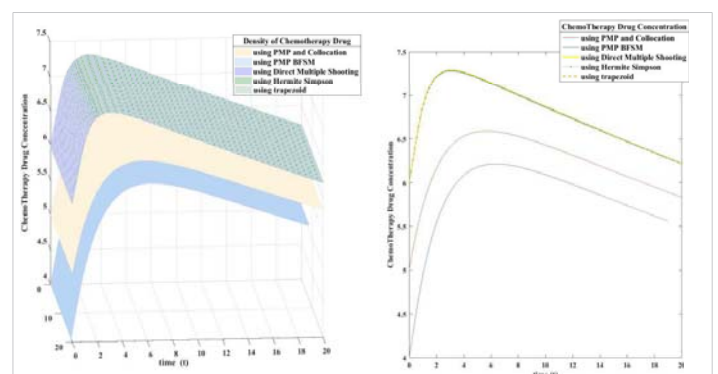


Figure 4.4: Density of Chemo Therapy Drug (3D plot) left side, (2D plot) right side.



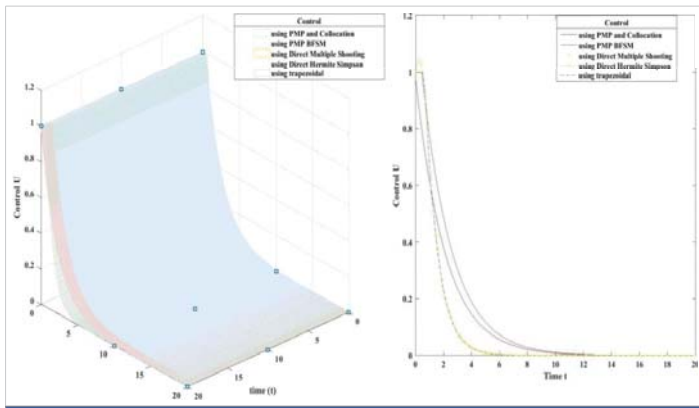


Figure 4.5: Control Plot using different methods.

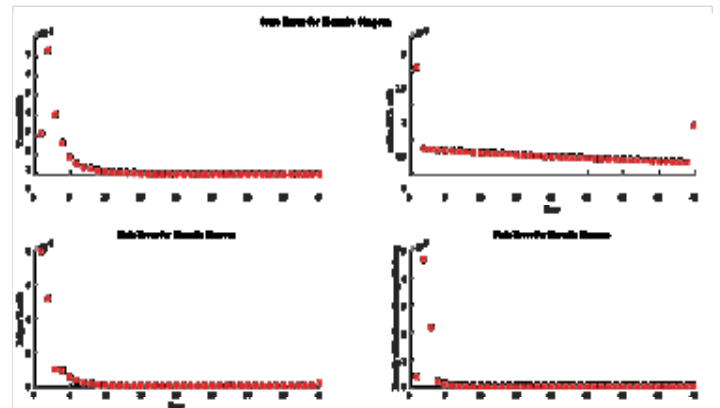


Figure 4.8: State Error for Hermite Simpson Method.

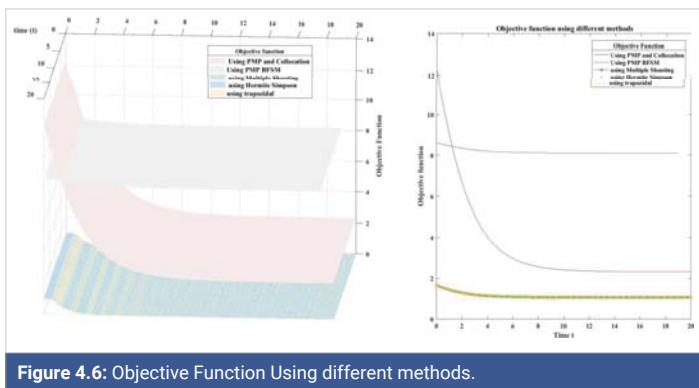


Figure 4.6: Objective Function Using different methods.

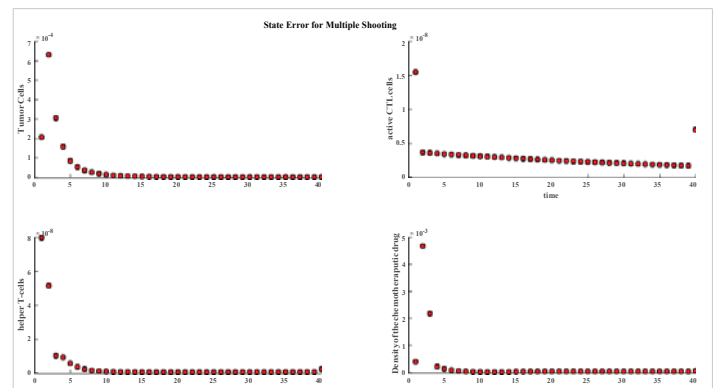


Figure 4.9: State Error for Direct Multiple Shooting Method.

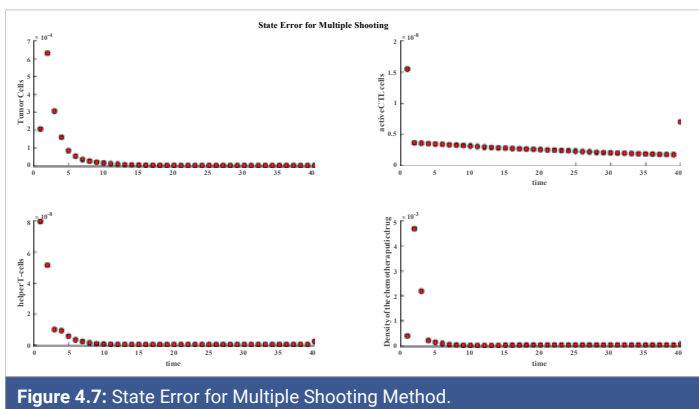


Figure 4.7: State Error for Multiple Shooting Method.

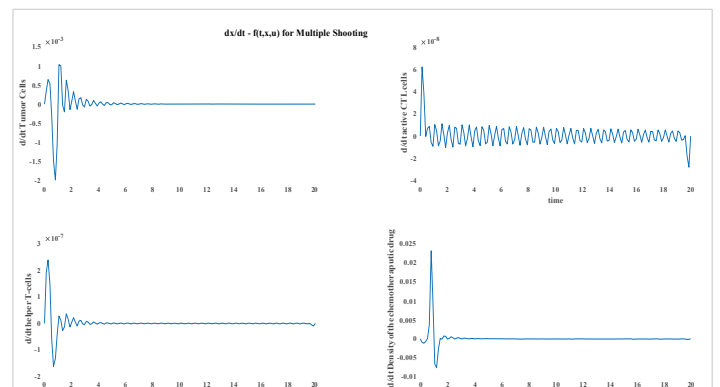


Figure 4.10:  $dx/dt - f(t,x,u)$  for Multiple Shooting.

**Table 2:** Objective function values for different methods.

Method	Objective Function J at $t_f$ value
Direct Multiple Shooting	1.067535615031329
Hermite Simpson DT	1.065628747428853
Trapezoidal DT	1.532368800622538
Pontryagin Maximum Principle and 3 <sup>rd</sup> Lobatto collocation	2.324027433971033
Pontryagin Maximum principle and BFSM	8.118443453241154

**Table 3:** Different Comparison merits for the direct method.

Method Name	Absolute local error	NLP iteration number	NLP time
Direct Multiple Shooting	0.004675646596448	60	9.305698207820010
Hermite Simpson DT	7.125004407381137e-04	35	2.210647539360693
Trapezoidal DT	0.039534796829716	80	1.645410588354529

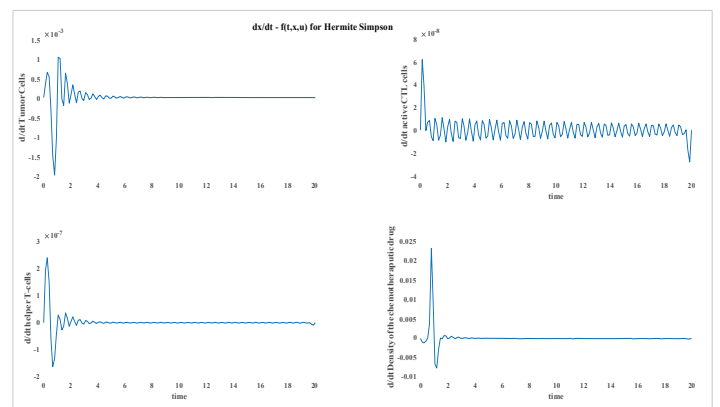


Figure 4.11:  $dx/dt - f(t,x,u)$  for Hermite Simpson.

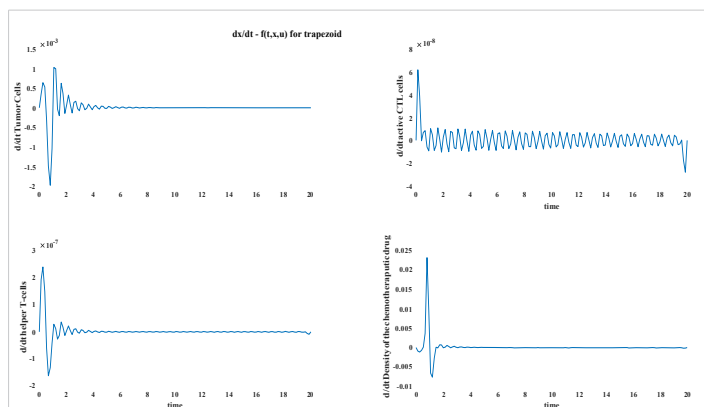


Figure 4.12:  $dx/dt - f(t,x,u)$  for trapezoid.

cells is lower in direct method than indirect methods also direct methods achieve better control than indirect method the following table shows the value of objective function for five different methods.

Since the direct methods achieve better results than indirect method this motivate us to give more insight into direct methods through new merits such as discretization error, absolute local error, number of NLP iterations, NLP time the following table gives the values of the previous merits for the methods of direct methods only.

Within the direct method apart from indirect methods the results are better in both Hermite Simpson DT and direct multiple shooting is better than the trapezoidal since both two methods represented by polynomials, the Hermite Simpson's DT is better than the Trapezoidal DT since Hermite Simpson's DT uses a higher order polynomial than trapezoidal DT which is an agreement with the concept of (p-method), for more details see [29], the hermite Simpson is the most accurate method on the price of time.

The following plots shows the different discretization errors and state errors for the three method.

## Conclusion

In this article five numerical techniques to study the tumor immune dynamic optimization model are discussed, these methods are indirect method in which the resulting TPBVP is studied by collocation method, and by the backward forward sweep method and direct methods which are multiple shooting method, trapezoid direct collocation method, Hermite-Simpson's direct collocation. Many studies claim that indirect methods have drawbacks. The two major drawbacks are that the differential equations obtained are often difficult to solve due to strong nonlinearity and instability and also define a guess initial solution for the Lagrangian multiplier. As a matter of fact, these variables do not have a physical meaning, this leads to problems in the definition of an initial guess solution to start the algorithm. Another problem that it is necessary to take into account in the resolution of a BVP is that the existence and uniqueness of solution is not guaranteed as

in an initial value problem. Each problem may have a unique solution, several solutions or no solution at all. By comparing different methods both in direct and indirect methods we can claim that direct method can be used to get better results than indirect method and it is easy to implement. A question that is needed to be answered is how to choose the discretization method among different methods related to direct method to get the best results this question can be a future work.

## References

1. Abbas AK, Lichtman AH, Pillai S. Cellular and molecular immunology. Saunders Elsevier. 2014.
2. Aktas Z, Stetter HJ. A classification and survey of numerical methods for boundary value problems in ordinary differential equations. Int J Numerical Methods Engineering. 1977; 11: 771-796.
3. Anita S, Arnautu V, Capasso V. An introduction to optimal control problems in life sciences and economics: from mathematical models to numerical simulation with MATLAB®. Modeling and simulation in science, engineering and technology. Birkhäuser. New York. 2011.
4. Bellman RE. Dynamic Programming. Courier Corporation. 2003.
5. Betts JT. A Survey of Numerical Methods for Trajectory Optimization. Control and Dynamics. 1998; 21:193-207.
6. Betts JT. Practical methods for optimal control and estimation using nonlinear programming. Advances in design and control. Society for Industrial and Applied Mathematics. Philadelphia. 2<sup>nd</sup> edition. 2010.
7. Biegler LT. Nonlinear programming: concepts, algorithms, and applications to chemical processes. Number 10 in MOS-SIAM series on optimization. SIAM. Philadelphia. 2010.
8. Binder T, Blank L, Bock HG, Bulirsch R, Dahmen W, et al. Introduction to Model Based Optimization of Chemical Processes on Moving Horizons. In Introduction to Model Based Optimization of Chemical Processes on Moving Horizons. Springer Berlin Heidelberg. 2001; 295-339.
9. Biral F, Bertolazzi E, Bosetti P. Notes on Numerical Methods for Solving Optimal Control Problems. IEEJ J Industry Applications. 2015; 5:154-166.
10. Bock H, Plitt K. A multiple shooting algorithm for direct solution of optimal control problems. In 9<sup>th</sup> IFAC. Pergamon Press. 1984; 242-247.
11. Bryson AE, Ho YC. Applied optimal control. Hemisphere Publication Corporation. 1975.
12. Chachuat B. Nonlinear and Dynamic Optimization: From Theory to Practice - IC-32: Spring Term. EPFL. 2009.
13. Curiel T. Tregs and rethinking cancer immunotherapy. J Clin Investigation. 2007; 117: 1167-1174.  
[PubMed: https://www.ncbi.nlm.nih.gov/pubmed/17476346](https://www.ncbi.nlm.nih.gov/pubmed/17476346)
14. De Pillis LG, Radunskaya AE. Mixed immunotherapy and chemotherapy of tumors: modeling, applications and biological interpretations. J Theoretical Biol. 2006; 238.
15. De Pillis LG, Gu W, Fister KR, Head T, Maples K, et al. Chemotherapy for tumors: An analysis of the dynamics and a study of quadratic and linear optimal controls. Mathematical Biosciences. 2007; 209: 292-315.
16. Dickmanns ED, Well KH. Approximate solution of optimal control problems using third order hermite polynomial functions. In Marchuk GI, editor. Optimization Techniques IFIP Technical Conference Novosibirsk, number 27 in Lecture Notes in Computer Science, pages. Springer Berlin Heidelberg. 1974; 158-166.

17. Diehl M, Findeisen R, Schwarzkopf S, Uslu I, Allgöwer F, et al. An Efficient Algorithm for Nonlinear Model Predictive Control of Large-Scale Systems Part I: Description of the Method. *At-Automatisierungstechnik Methoden und Anwendungen der Steuerungs-, Regelungs- und Informationstechnik*. 2002; 50: 557.
18. García-Heras J, Soler M, Sáez FJ. A Comparison of Optimal Control Methods for Minimum Fuel Cruise at Constant Altitude and Course with Fixed Arrival Time. *Procedia Engineering*. 2014; 80:231-244.
19. Gibbs WW. Untangling the roots of cancer. *Scientific America*. 2003.
20. Hackbusch W. A numerical method for solving parabolic equations with opposite orientations. *Computing*. 1978; 20: 229-240.
21. Hairer E, Norsett SP, Wanner G. *Solving Ordinary Differential Equations I Nonstiff Problems*. Springer-Verlag Berlin Heidelberg, 2008.
22. Joshi HR. Optimal control of an HIV immunology model. *Optimal Control Applications and Methods*. 2002; 23: 199-213.
23. Karush W. *Minima of Functions of Several Variables with Inequalities as Side Constraints*. Ph.D. Department of Mathematics. University of Chicago. Chicago. 1939.
24. Kirschner D, Panetta P. Modeling immuno therapy of the tumor-immune interaction. *Journal of Mathematical Biology*. 1998; 37: 235-252. [PubMed: https://www.ncbi.nlm.nih.gov/pubmed/9785481](https://www.ncbi.nlm.nih.gov/pubmed/9785481)
25. Kirschner DE, Jackson TL, Arciero JC. A mathematical model of tumor-immune evasion and siRNA treatment. *Discrete and continuous dynamical systems series- B*. 2003; 37: 39-58.
26. Kuhn H, Tucker A. *Nonlinear Programming*. 1951; 481-492, California. University of California Press. Berkeley.
27. Lenhart S, Workman JT. *Optimal Control Applied to Biological Models*. Chapman & Hall/CRC Mathematical and Computational Biology. CRC Press. Taylor & Francis Group. 2007.
28. Leon K, Garcia K, Carneiro J, Lage A. How regulatory CD25 (+)CD4(+) T cells impinge on tumor immunobiology? On the existence of two alternative dynamical classes of tumors. *J Theoretical Biol*. 2007; 247: 122-137.
29. Matthew PK. *Transcription Methods for Trajectory Optimization A beginners tutorial*. Technical report. Cornell University. 2015.
30. Mitter SK. The successive approximation method for the solution of optimal control problems. *Automatica*. 1996; 3:135-149.
31. Pillis LG, Radunskaya AE. A mathematical model of immune response to tumor invasion. MIT. 2003; 1661-16668
32. Pontryagin LS. *Mathematical Theory of Optimal Processes*. CRC Press. March 1987.
33. Rao AV. A survey of numerical methods for optimal control. *Advances in the Astronautical Sciences*. 2009; 135: 497-528.
34. Rao AV, Benson DA, Darby C, Patterson MA, Franconin C, et al. Algorithm 902: GPOPS, A MATLAB software for solving multiple-phase optimal control problems using the gauss pseudospectral method. *ACM Transactions on Mathematical Software (TOMS)*, 2010; 37: 1-39.
35. Schattlera H, Urszula L. *Optimal Control for Mathematical Models of Cancer Therapies*. Springer Publishing Co., USA. 2015.
36. Shampine LF, Gladwell I, Thompson S. *Solving ODEs with MATLAB*. Cambridge University Press. 2003.20.
37. Sharma S, Samanta GP. Dynamical Behaviour of a Tumor-Immune System with Chemotherapy and Optimal Control. *Journal of Nonlinear Dynamics*. 2013: 1-13, 2013.
38. Sweilam NH, Al-Ajami TM. Legendre spectral-collocation method for solving some types of fractional optimal control problems. *J Advanced Res*. 2015; 393-403. [PubMed: https://www.ncbi.nlm.nih.gov/pubmed/26257937](https://www.ncbi.nlm.nih.gov/pubmed/26257937)
39. Sweilam NH, AL-Mekhla M. On the Optimal Control for Fractional Multi-Strain TB Model. *Optimal Control, Applications and Methods*. 2016.
40. Törn A, Žilinskas A, Goos G, Hartmanis J, Barstow D, et al. *Global Optimization*, volume 350 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg. Berlin. Heidelberg. 1989.
41. Victor VM. *Practical Direct Collocation Methods for Computational Optimal Control*. In *Modeling and Optimization in Space Engineering*. Volume 73 of *Springer Optimization and Its Applications*. Springer New York. 2013; 33-60.
42. Zaman G, Han Kang Y, Jung IH. Stability analysis and optimal vaccination of an SIR epidemic model. *BioSystems*. 93: 240-249. 2008. [PubMed: https://www.ncbi.nlm.nih.gov/pubmed/18584947](https://www.ncbi.nlm.nih.gov/pubmed/18584947)