Swansea University
Prifysgol Abertawe

SCHOOL OF ENGINEERING

MSc in Computational Mechanics

NONLINEAR OPTIMAL CONTROL: A COMPUTATIONAL STRATEGY AND APPLICATION TO A MICRO-SWIMMER

Submitted to the University of Wales in fulfilment of the requirements for the Degree of Master of Science

Rodolfo Miguel Nogueira Fleury - 628055

Supervisors:

- Dr. Wulf G. Dettmer
- Prof. Djordje Peric

Swansea, 13^{th} of June 2011.

DECLARATION

This work has not previously been accepted in substance for any degree and is not being currently submitted in candidature for any degree.

Signed.....(candidate)
Date.....

STATEMENT 1

This thesis is the result of my own investigation, except where otherwise stated. Other sources are acknowledged by footnotes giving explicit references. A bibliography is appended.

Signed.....(candidate)
Date.....

STATEMENT 2

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organizations.

Signed.....(candidate)
Date.....

Abstract

This work is a preliminary work for the development of a numerical strategy for the optimal control of Fluid-Structure Interactions systems. A simple model is chosen to address such problems. This work focuses only on the solution of highly nonlinear systems, ignoring the coupling aspect of Fluid-Structure Interaction problems. A simple three-spheres micro-swimmer is chosen as a model problem. The motion equations of the micro-swimmer are presented and later expressed as an optimal control problem. The Variational approach is applied to the micro-swimmer problem and necessary equations are derived.

A monolithic computational strategy is suggested for the solution of a nonlinear two-point boundary value problem. The validation of the numerical strategy for linear and nonlinear problems are presented. Finally, the optimal stroke of the swimmer is obtained for different initial guesses. Furthermore, the analysis of different numerical integration methods is conducted. A comparative study is also done to monitor the energy waste for different number of strokes necessary to reach a prescribed displacement. It is shown that the formulation and numerical strategy used in this work are suitable to the micro-swimmer problem and can be extended to more complex systems.

Acknowledgements

I would like to express my profound gratitude to my supervisors Dr. Wulf Dettmer and Prof. Djordje Peric for their guidance during this work. They have encouraged me to overcome all the problems faced during the development of this thesis. They have also provided me helpful advices and suggestions, without which, the results obtained in this work would not have been possible. I would like to give a special thanks to Dr. Wulf Dettmer for his time and willingness to help every time I needed.

I also would like to thank the European Commission for sponsoring me in this Master program and all the staff at the Center for Numerical Methods in Engineering (CIMNE) in Barcelona and at Swansea University for creating this successful consortium. I also thank Dr. Antonio J. Gil, course coordinator at Swansea University, for his attention and help throughout the last two years.

I would like to thank all my colleagues of this course, specially Ganesh Chithambalam and Vekatesh Gopinath that have been with me throughout the last two years. And also, Dhrubajyoti Mukherjee, Caner Kara, Regina Khakimova and all my friends in Swansea and in Brazil.

I also thank my girlfriend Katharina Ain for her companionship and for helping me during the difficult moments of the thesis. Also I thank her sister Anne Ain for her friendship and for helping me with the report. Finally, I thank my mother, my father, my brothers and all my family in Brazil for their unconditional support and for encouraging me to do this course even if that meant that I would be away during this period.

Rodolfo Miguel Nogueira Fleury

Contents

List of Figures

List of Tables

List of Symbols

x	Relative distance between spheres 1 and 2
y	Relative distance between spheres 2 and 3
c	Center of mass of the micro-swimmer
$\mathbf{v_f}$	Vector of fluid velocities
p	Pressure field in the fluid
ρ	Density of the fluid
μ	Dynamic viscosity of the fluid
$\mathbf{f}_i$	Force vector acting on each sphere i
$\mathbf{v}_i$	Velocity vector acting on each sphere i
$\mathbf{S}(x, y)$	Oseen tensor
$\mathbf{r}$	Vectorial distance between the spheres
$\mathbf{E}(x, y)$	Error component of the Oseen tensor
$\mathbf{S}_\infty(x, y)$	Oseen tensor for large distances between the spheres
φ	Dimensionless ratio between radius and distance of the spheres
V_x(x,y)	Function that relates $\dot{x}$ with $\dot{c}$
V_y(x,y)	Function that relates $\dot{y}$ with $\dot{c}$
D	Initial distance between the spheres
w	Relative velocity between the spheres
a	Radius of the spheres of the micro-swimmer
ϵ	Relative change in the distance of the spheres
Δ	Net displacement of the swimmer
t_f	Final time
t	Time instant
t_k	Discrete time instant
θ	Integration coefficient of the generalized midpoint rule
Δt	Time increment
$\mathbf{x}$	State vector
$\mathbf{u}$	Control vector
$\boldsymbol{\lambda}$	Costate vector

E	Energy waste
J	Performance index
$\mathbf{P}(x, y)$	Nonlinear weighting matrix of the control variables
$\mathcal{H}$	Hamiltonian functional
$\mathcal{L}$	Lagrangian functional
n	Number of state variables
m	Number of control variables
N	Number of time steps
n_{eq}	Number of optimal control equations
$\mathbf{C}$	Matrix of coefficients in the linear monolithic strategy
$\mathbf{x_0}$	Vector of initial conditions
$\mathbf{x_f}$	Vector of final conditions
$\mathbf{R}$	Residual vector
$\mathbf{J}$	Jacobian matrix
$\mathbf{X}$	Vector with all discrete state, costate and control variables

1 INTRODUCTION

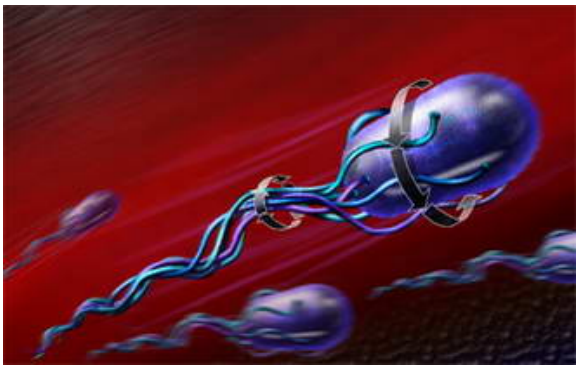
1.1 MOTIVATION AND AIM OF THE THESIS

This work is a preliminary study for solving the optimal control problem in Fluid-Structure Interaction (FSI) systems. In recent years, the study of problems involving both fluid and structural domains has increased significantly and many numerical techniques have been developed to solve such systems. After great advances in the coupling of the problem, perhaps the next step in this field of research is to optimize those systems by adapting the different optimization techniques the problem. Among the different ways of optimizing a system, this work will focus on the optimal control theory with the aim of later applying it to systems with fluid-structural interaction.

Many FSI systems are controlled by a small number of parameters, which can be varied during process. Therefore, it can be often desirable to optimize the choice of those parameters and the process itself, with the objective of reducing the energy waste, the total process time or any other feature of the system. Applications range from the control of wind turbine generators and aeroplane control surfaces to the swimming motion of micro organisms, see Fig.1.1. In the power generation context, the optimization of the process is directly related to the efficiency of the system and, hence, the amount of energy generated. For example, in a wind turbine the energy produced is controlled by the transmission of the rotation of the blades to the power generator. This control is done by a gearbox and the optimal choice of the gear may vary depending on the intensity of the wind at a given instant.

This work addresses such problems on a conceptual level. Initially, a small model problem will be formulated and appropriate mathematical techniques will be chosen for the optimal parameter control. A computational strategy will be developed to solve highly nonlinear problems, as it is often the case of FSI systems. The computational strategy will be applied to a micro-swimmer problem, which is one of the simplest one dimensional FSI problems. However, the problem solved in this work will not deal with the coupling of the systems, but with the direct solution of the motion equations that govern the micro-swimmer problem.

(a)



(b)

Figure 1.1: Examples of FSI systems with variable control parameters. (a) Wind turbines; (b) Micro swimmers.

1.2 STATE OF THE ART

1.2.1 Optimal control theory

The idea of optimal control began a long time ago with the calculus of variations, the branch of mathematics that deals with finding the maximum or minimum of a functional. It is argued that the first optimal control problem was the *brachystochrone problem* proposed by Johann Bernoulli in 1696 [Sussman, 1996], which is to find the path of quickest descent between two points with different height. Bernoulli's problem was a true minimum-time problem and it explicitly asked for the optimal path that minimizes the time, which can be viewed as the performance index in an analogy with the optimal control problems.

The modern theory of optimal control had its main developments during the 1950s with the formulation of two main optimization techniques: Dynamic Programming, introduced by R. E. Bellman [Bellman, 1952], and the Pontryagin's Minimum Principle [Pontryagin et al., 1956]. The approaches are significantly different but both of them still have applications nowadays. The Dynamic Programming makes use of the principle of optimality and it is suitable for solving discrete problems, allowing a significant reduction in the computation of the optimal controls, see [Kirk, 1998]. It is also possible to obtain a continuous approach to the principle of optimality that leads to the solution of a partial differential equation called the Hamilton-Jacobi-bellman equation. An alternative approach to the optimal control problem is to use the calculus of variation

and the Pontryagin's Minimum Principle that results in the Hamiltonian approach, which is generalization of the Euler-Lagrange equations. The variational approach will lead to the solution of a nonlinear two-point boundary value problem and it will require appropriate techniques for its solution. The variational approach presents a more elegant solution and it will be the one used in the solution of the micro-swimmer optimal control problem.

1.2.2 The micro-swimmer problem

The micro-swimmer problem has strong relevance in different fields of science, such as, biology, nano-mechanics and medicine. For this reason, many authors have worked to find mechanisms capable of swimming in systems with high viscosity and small dimensional scales. Those circumstances require special swimming mechanisms to allow the propagation of the swimmer in the viscous media. One of the pioneering works in this field is given in [Purcel, 1977] and it is shown that animals like scallops can not swim in a highly viscous flow by the opening and closing movement of a single hinge. Purcel proposed that a non-reciprocal motion is required to produce a net displacement because it breaks the time-reversal property of the Stokes equations. Following that, a number of micro-swimmers have been proposed, such as, swimmers made of joined rods, spheres with prescribed tangential velocities and liked spheres undergoing shape changes [Putz and Yeomans, 1999]. One example of a swimmer capable of self propulsion in a viscous driven motion, which is also one of the simplest swimmers proposed, is the three-spheres swimmer introduced by Najafi [Najafi and Golestanian, 2004].

Many authors have used the three-linked-spheres swimmer as a model problem to their analysis. In 2008, Alouges and collaborators introduced the idea of finding the optimal strokes of a micro-swimmer, using the three-spheres swimmer as their model in [Alouges et al., 2008]. Later, they expanded their work to axisymmetric micro-swimmers in [Alouges et al., 2010]. The synchronization of micro-swimmers and the hydrodynamic interaction in low Reynolds number were studied with the three-spheres swimmer in [Putz and Yeomans, 1999] and [Alexander and Yeomans, 2008]. This swimmer will be used in this work with the objective of finding the optimal relative displacement of the spheres that results in a certain net displacement with the minimum energy waste.

1.3 LAYOUT OF THE THESIS

Chapter 2 introduces the optimal control theory. The main aspects of an optimal control problem are presented, such as, the state variable representation of systems, the choice of the performance index and the restrictions experienced by the control and state variables. The chapter also brings the formulation of the two main approaches to the problem, the Dynamic Programming and the Variational approach.

In chapter 3, the formulation of the micro-swimmer problem is given. The motion equations of the micro-swimmer are presented, as well, as the Oseen-tensor formulation of the problem. Next, the kinematics of the problem are discussed, resulting in the ordinary differential equation that governs the motion of the swimmer. A simple four-stage stroke is then presented and the numerical solution of the motion of the swimmer will be verified with the data found in the literature.

In chapter 4, the optimal control theory is applied to the micro-swimmer problem. The performance index of the problem is defined as the minimal energy waste and the system is rewritten using the state variable representation. The Hamiltonian approach is applied and the optimal control equations necessary for the solution are obtained.

Chapter 5 presents the numerical procedure adopted in the solution of the problem. The time discretization of the problem is applied by using the generalized midpoint rule. Next, a monolithic approach for solving linear optimal control equations is introduced and the verification of the methodology is done by solving linear problems with exact analytical solution. Finally, the Newton-Raphson method is applied to the monolithic approach and example problems are solved to verify the nonlinear procedure.

In chapter 6,s the numerical procedure is applied to the micro-swimmer problem. The optimal stroke of the swimmer is obtained with an energy waste reduction. Other local minimum of the solution are also presented and compared with the initial strokes studied. Next, a convergence analysis of the different types of numerical integration is done. At last, the comparison of the energy waste for achieving a certain point in space with one single stroke and with several small strokes are presented.

Chapter 7 brings the conclusions of this thesis and some directions for future works.

2 OPTIMAL CONTROL THEORY

The main objective of the optimal control theory is to determine the control inputs that will cause the system to satisfy its physical constraints while extremizing, that is, maximizing or minimizing a chosen performance index or criterion. The standard approach to the problem is to quantify the value of each possibility, so that the best controls for a certain problem may be found. Examples of optimal control problems are found in different areas of engineering and applied math, such as, minimizing the fuel consumption of an aircraft, maximizing the range of a rocket, minimizing the amount of material used in the manufacturing of an object or even in maximizing the profits of an economic enterprise.

The two main approaches to the optimal control problem are the Dynamic Programming and the Variational approach. The dynamic programming approach is based in the principle of optimality and has introduced a significant reduction in the computational time, if compared with the global evaluation of all possibilities of a system. Furthermore, a continuous approach of the principle of optimality may be presented, which results in the solution of the partial differential Hamilton-Jcaobi-Bellman equation. The approach based on the principle of optimality gives a closed-loop solution, Fig.2.1a, resulting in a global search of the optimal controls. More efficient and elegant, the variational approach uses the Pontryagin's minimum principle, which is a generalization of the Euler-Lagrange approach . However, the variational, or Hamiltonian, approach is an open-loop optimal control, Fig.2.1b, and gives the optimal values for specific initial conditions.

In this chapter, the general aspects of the optimal control problem are discussed, such as, the description of the problem, the different possibilities of objective functions commonly used and their formulation. The two main procedures used in the optimal control solution, dynamic programming and variational approach, are presented and derived. More details of the optimal control theory may be found in [Kirk, 1998], [Naidu, 2003] and [Subchan and Zbikowski, 2009].

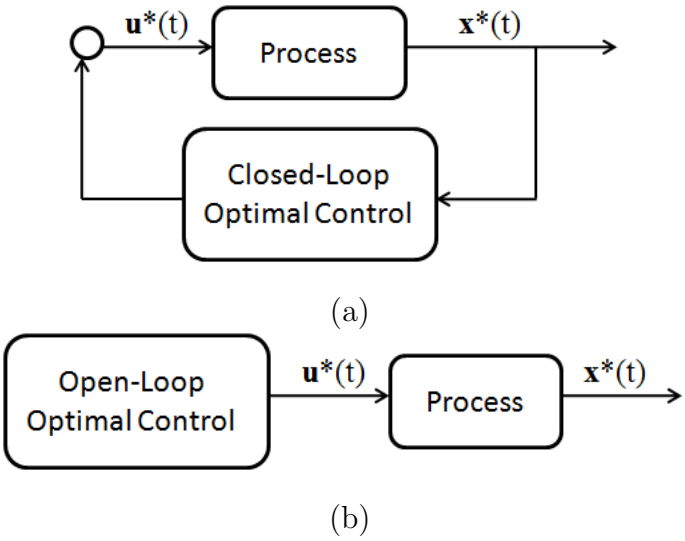


Figure 2.1: (a) Closed-loop optimal control system; (b) Open-loop optimal control system.

2.1 PROBLEM DESCRIPTION

The main steps of the formulation of the optimal control problem are the modelling of the controlled system, the specification of a performance criterion and the physical constraints to be satisfied, that is, path, initial and terminal constraints. Each one of those steps is discussed next.

2.1.1 Optimal Control Formulation

Many model problems in engineering are governed by first or higher order ordinary differential equations (ODEs). However, it is possible to formulate higher order differential equations as a system of first order ODEs depending on the *state* and *control* variables only. State variables are the quantities $x_1(t), x_2(t), ..., x_n(t)$ that, if known at time $t = t_0$, are known for a time $t \geq t_0$, provided that the inputs of the system are given. The control variables $u_1(t), u_2(t), ..., u_n(t)$ are the inputs of the system at each instant t and can be chosen according to the controller's interests. The state variable representation of systems is convenient for the solution of problems, numerically or analytically, and has strong physical motivation [Kirk, 1998].

A general nonlinear optimal control problem is to find the optimal control $\mathbf{u}^*(t)$ (* indicates the optimal value throughout this work) that causes a certain system of differential equations

$$\dot{\mathbf{x}} = \mathbf{f}\left(\mathbf{x}(t), \mathbf{u}(t), t\right) \tag{2.1}$$

to give the optimal trajectory $\mathbf{x}^*(t)$ that optimizes a chosen performance index

$$J = S\left(\mathbf{x}(t_f), t_f\right) + \int_{t_0}^{t_f} V\left(\mathbf{x}(t),\, \mathbf{u}(t),\; t\right)\; dt. \tag{2.2}$$

In equations (2.1) and (2.2), $\mathbf{x}(t)$ is the vector containing the state variables and $\mathbf{u}(t)$ the vector of control variables. The aspects and the choice of the performance index are discussed in the following section.

2.1.2 Performance Index

The objective of the optimal control theory is to choose the control input, such that, it will result in the "best" possible output. The characterization of best output depends on the physical property or function (performance index) that one wants to minimize or maximize. Examples of performance index are time, energy, fuel consumption, terminal cost, among others. Therefore, the choice of the performance index will depend on the properties of each problem and on the interests of the operator or designer. A description of some of the common performance index used are presented next.

2.1.2.1 Minimum-Time Problems

If the objective of the problem is to reduce the total time of a certain process, the problem can be classified as a minimum-time problem (or time-optimal control system). This type of problem consists of transferring the system from an initial state $\mathbf{x}(t_0)$ to a specified final state $\mathbf{x}(t_f)$ in minimal time. The performance measure to be minimized is

$$J = \int_{t_0}^{t_f} dt = t_f - t_0 = t^*. \tag{2.3}$$

Operations in which the time consumed is more important than any other factor such as energy or fuel consumption are suitable for this type of problem formulation. Some typical examples where the minimum-time performance index is used are the

interception of attacking aircraft and missiles, and the slewing mode operation of a radar or gun system.

2.1.2.2 Terminal Control Problems

In a terminal control problem, one is interested in minimizing the error (or deviation) between the desired final state $\mathbf{x}_d(t_f)$ and the actual final state $\mathbf{x}_a(t_f)$. The target error may be defined as the vector $\mathbf{x}(t_f) = \mathbf{x}_a(t_f) - \mathbf{x}_d(t_f)$. Since both the positive and negative deviations of the final state are undesirable, it is convenient to define the performance criterion as the squared of the error vector. The terminal cost problem written in matrix notation is defined as

$$J = \mathbf{x}^T(t)\mathbf{F}\,\mathbf{x}(t), \tag{2.4}$$

where $\mathbf{F}$ is a positive semi-definite weighting matrix. As it will be shown later, the performance index might involve more than one term to be minimized. Hence, the weighting matrix $\mathbf{F}$ can evaluate the importance of the terminal cost term to the overall performance criterion, as well as the different contributions of each term inside the vector $\mathbf{x}(t)$ to the system.

2.1.2.3 Minimum-Energy Control Problems

In most control problems, the selection of the control inputs acts directly on the energy consumption of the system. This energy consumption will later determine, for example, the amount of fuel or electrical current that the system will require. Therefore, one might be interested in minimizing the energy consumption for economical and/or environmental reasons, or simply to increase the system's efficiency. The minimization of the total power or total energy rate may be defined as

$$J = \int_{t_0}^{t_f} \mathbf{u}^T(t)\,\mathbf{P}\,\mathbf{u}(t)\;dt. \tag{2.5}$$

Here, $\mathbf{P}$ is a positive definite matrix that weights the control variables. In the example of an electric current network minimization, the matrix $\mathbf{P}$ is correspondent to the electrical resistance associated to each control. One can also think of problems where the

state variables are determinant to the system's energy consumption. Such problems may be stated as

$$J = \int_{t_0}^{t_f} \mathbf{x}^T(t)\,\mathbf{Q}\,\mathbf{x}(t)\ dt \tag{2.6}$$

where $\mathbf{Q}$, in this case, is the respective weighting semi-positive matrix of the state variables. The weighting matrix defines, among other things, the importance of the energy consumption to the total value of the performance index.

2.1.2.4 General Optimal Control Problems

A general problem may involve the minimization of one or more of the conditions described above. Combining all cases presented in only one criterion function, the following performance index is obtained,

$$J = \mathbf{x}^T(t)\mathbf{F}\,\mathbf{x}(t) \int_{t_0}^{t_f} \left[\mathbf{x}^T(t)\,\mathbf{Q}\,\mathbf{x}(t)\ + \mathbf{u}^T(t)\,\mathbf{P}\,\mathbf{u}(t)\right]\ dt. \tag{2.7}$$

The above equation is also called the performance index for the linear regulator optimal control problem. Notice that the matrices $\mathbf{P}$ and $\mathbf{Q}$ might be time dependent or even dependent on the state variables, resulting in a nonlinear problem. Therefore, one may also define the general nonlinear form of the performance index as

$$J = S\left(\mathbf{x}(t_f), t_f\right) + \int_{t_0}^{t_f} V\left(\mathbf{x}(t),\, \mathbf{u}(t),\ t\right)\ dt. \tag{2.8}$$

The problems of optimal control may also be classified according to their performance index J. A problem where the performance index is determined only by the terminal cost is called the *Mayer* problem, while the *Lagrange* problem has only the integral cost term and the *Bolza* type problem presents both integral and final state components of (2.8).

2.1.3 Constraints

Physical constraints appear in optimal control problems in two main forms, boundary conditions and control (or state) restrictions. The first is necessary for any optimal problem solution. As it will be shown later in this chapter, the initial state of the problem is essential in the variational approach. However, it can be omitted in the dynamic programming, resulting in a global search other than a local minimization. That is, the search will find not only the optimal path, but the optimal path for all possible initial conditions. With respect to the final condition, it might be defined in different manners, for example, as an imposed final state $\mathbf{x}(t_f) = \mathbf{x}_f$ and free final time t_f or as a free final state $\mathbf{x}(t_f)$ and imposed final time $t_f = T$.

Restrictions in the control and state variables are also often found in real problems. For example, an automobile has a maximum limitation in the acceleration depending on the engine power, and its deceleration is restricted by the braking system parameters. Maximum and minimum constraints in the state variables can also be found, such that, the path must be within a certain admissible area of the space. Those restrictions may be mathematically represented by

$$\mathbf{U}_- \leq \mathbf{u}(t) \leq \mathbf{U}_+ \qquad \mathbf{X}_- \leq \mathbf{x}(t) \leq \mathbf{X}_+ \tag{2.9}$$

In this work, the constraints in the control and state variables will be left aside. The unconstrained problem has less complications and leads to more elegant results. As an initial approach to the optimal control problem, it will serve as a good evaluation parameter of the methodology implemented.

2.2 DYNAMIC PROGRAMMING APPROACH

The Dynamic Programming approach for the optimal control problem was first introduced by R. E. Bellman in 1952 [Bellman, 1952]. The idea of the dynamic programming is that it minimizes a discrete multistage optimization problem such that for each stage all options are evaluated and the best discrete value is stored. For that, the dynamic programming make use of the *principle of optimality* as it will defined further in this section. Notice that the discretization of the admissible controls and state variables is generally a discrete approximation of a continuous space and, therefore,

errors associated to the discretization are normally found in the dynamic programming solution. This approach is suitable to digital computers but an alternative approach might also be presented, which leads to the solution of the Hamilton-Jacobi-Bellman partial differential equation. The derivation of the partial differential equation is presented in this section and more details on the dynamic programming approach and the principle of optimality might be found in [Kirk, 1998], [Subchan and Zbikowski, 2009], [Knowels, 1981] and [Dyer and McReynolds, 1970].

2.2.1 The Principle of Optimality

The principle of optimality states that if an optimal control is broken into two pieces, then the last piece is itself optimal. The basic assumption of the principle of optimality is that the system can be characterized by its state $\mathbf{x}(t)$ at time t, which completely summarizes the effect of all inputs $\mathbf{u}(t)$ prior to time t. This allows for a local characterization of optimality as given in the following formal statement of the principle [Primbs, 1999]:

Definition 2.1: ***Principle of Optimality:*** *If $\boldsymbol{u}^*(\tau)$ is optimal over the interval $[t, t_f]$, starting at state $\boldsymbol{x}(t)$, then $\boldsymbol{u}^*(\tau)$ is necessarily optimal over the subinterval $[t + \Delta t, t_f]$, for any Δt such that $t_f - t \geq \Delta t > 0$.*

Computationally speaking, the principle of optimality allows a significant reduction of the memory and time used in the solution, with respect to a global search of all admissible values of the control variables. A global search would require the storage of all possible control variables at each time step and the evaluation of all allowable paths to obtain the combination with the lowest performance index (an exhaustive search). Using the principle of optimality it is possible to work backward from the final state $\mathbf{x}(t_f)$ and search within each time interval the optimal control input for that specific step. Then, only the optimal value of each time step needs to be stored in the code, reducing significantly the number of combinations evaluated and the computational cost in the optimal control analysis.

2.2.2 The Hamilton-Jacobi-Bellman Equation

Alternatively to the discrete multistage approach of the dynamic programming, it is possible to use a continuous approach to the principle of optimality that will lead to the

solution of a partial differential equation called the Hamilton-Jacobi-Bellman equation. Using equations (2.1) and (2.2) as our optimal control problem and considering only the integral term of the performance index in (2.2), the minimal performance index for the optimal state $\mathbf{x}^*(t)$ with initial condition $\mathbf{x}(0) = \mathbf{x}_0$ is represented by $J^*(\mathbf{x}(t_0), t)$, such that

$$J^*(\mathbf{x}(t_0), t) = \int_{t_0}^{t_f} V\left(\mathbf{x}^*(t),\ \mathbf{u}^*(t),\ t\right)\ dt. \tag{2.10}$$

Notice that $J^*(\mathbf{x}(t_0), t)$ is independent of the control $\mathbf{u}(t)$, since the knowledge of the minimal performance index already determines the necessary controls $\mathbf{u}^*(t)$ for achieving its minimum. From the principle of optimality, one may obtain the minimum of the performance index function $J^*(\mathbf{x}^*(t), t)$ for a an arbitrary initial state along the optimal path $\mathbf{x}^*(t)$ at time t,

$$J^*(\mathbf{x}^*(t), t) = \int_{t}^{t_f} V\left(\mathbf{x}^*(\tau),\ \mathbf{u}^*(\tau),\ t\right)\ d\tau. \tag{2.11}$$

The attempt of finding the controls that will minimize (2.11) can start by using the additive property of integrals and the principle of optimality, such that

$$J^*(\mathbf{x}(t), t) = \min_{\substack{\mathbf{u}(\tau) \\ t \le \tau \le t_f}} \left\{ \int_{t}^{t+\Delta t} V(\mathbf{x}(\tau), \mathbf{u}(\tau),\ \tau)\ d\tau + J^*(\mathbf{x}(t+\Delta t),\ t+\Delta t) \right\}. \tag{2.12}$$

That is, the optimal cost at an arbitrary state is the minimum cost to move to $\mathbf{x}(t+\Delta t)$ plus the optimal cost $J^*(\mathbf{x}(t+\Delta t), t+\Delta t)$ from the state $\mathbf{x}(t+\Delta t)$, as stated in the principle of optimality. Therefore, the problem of finding the optimal control over the total time interval is reduced to finding the optimal control in the subinterval $[t, t+\Delta t]$.

If the interval Δt is small enough, the integral term in (2.12) can be approximated by $V(\mathbf{x}(t), \mathbf{u}(t), t)\Delta t$. Also, the optimal term $J^*(\mathbf{x}(t+\Delta t), t+\Delta t)$ at $\mathbf{x}(t+\Delta t)$ can be expanded in a Taylor series with respect to the point $(\mathbf{x}(t), t)$, resulting in

$$J^*(\mathbf{x}(t),t) = \min_{\mathbf{u}} \left\{ V(\mathbf{x}(t),\mathbf{u}(t),\ t)\Delta t + J^*(\mathbf{x}(t),\ t) + \frac{\partial J^*}{\partial t}\Delta t + \left(\frac{\partial J^*(\mathbf{x}(t),t)}{\partial \mathbf{x}}\right)^T [\mathbf{x}(t+\Delta t) - \mathbf{x}(t)] \right\}. \tag{2.13}$$

Dividing all terms by Δt, considering Δt to be small enough and knowing that the terms of $J^*(\mathbf{x}(t),t)$ does not depend on the minimization of $\mathbf{u}(t)$, (2.13) may be rewritten as

$$\frac{\partial J^*(\mathbf{x}(t),t)}{\partial t} + \min_{\mathbf{u}} \left\{ V(\mathbf{x}(t),\mathbf{u}(t),\ t) + \left(\frac{\partial J^*(\mathbf{x}(t),t)}{\partial \mathbf{x}}\right)^T [\mathbf{f}(\mathbf{x}(t),\mathbf{u}(t),t)] \right\} = 0. \tag{2.14}$$

Furthermore, introducing the Hamiltonian function $\mathcal{H}$ as

$$\mathcal{H} = V(\mathbf{x}(t),\mathbf{u}(t),t) + \left(\frac{\partial J^*(\mathbf{x}(t),t)}{\partial \mathbf{x}}\right)^T \mathbf{f}(\mathbf{x}(t),\mathbf{u}(t),t) \tag{2.15}$$

and knowing that the minimized control $\mathbf{u}^*(t)$ will result in the optimal value of the variables involved in the Hamiltonian, the Hamilton-Jacobi-Bellman equation might be obtained:

$$J^*_{\mathbf{t}}(\mathbf{x}(t),t) + \mathcal{H}\left(\mathbf{x}^*(t),\mathbf{u}^*(t),J^*_{\mathbf{x}},t\right) = 0, \tag{2.16}$$

where the notation $J^*_{\mathbf{t}}$ and $J^*_{\mathbf{x}}$ denotes to the partial derivatives of $J^*(\mathbf{x}(t),t)$ with respect to the time t and the state $\mathbf{x}(t)$ respectively.

The Hamilton-Jacobi-Bellman equation is, for most real problems, a nonlinear partial differential equation in J^* and it presents many difficulties in its solution. The boundary condition for this equation is the terminal cost given by the term $S(\mathbf{x}(t_f),t_f)$ in (2.8) and can be applied for problems with fixed final time or free final time t_f equally. Simple problems might be solved by guessing the form of the cost function $V(\mathbf{x}(t),\mathbf{u}(t),t)$, however, for most problems it is not possible to find a solution so easily [Kirk, 1998]. In the dynamic programming solution it was shown that the continuous process is approximated by a discrete system and, using recurrence equations, one obtains an exact solution to the approximated problem. In the Hamilton-Jacobi-Bellman

approach the continuous problem is not discretized, but the numerical solution for the nonlinear partial equation often requires a discretization. Therefore, the second approach leads to an approximate solution of the exact problem. In both cases the solution can be obtained by a closed-loop optimal control and its accuracy will be dependent on the quality of the discretization of the system (dynamic programming) or of the solution (continuous approach).

2.3 VARIATIONAL APPROACH TO OPTIMAL CONTROL PROBLEMS

Alternatively to the dynamic programming solution of the optimal control problem presented in section 2.2, the variational approach to the optimal control problem is presented, which is based on a generalization of the calculus of variations. The variational approach is also called the indirect approach to the solution of the optimal control problem. Necessary conditions for an extremum are derived by considering the first variation of the performance index J with constraints adjoined in the manner of Lagrange method. The approach is called indirect, because the optimal control is found by solving the auxiliary two-point boundary value problem, rather than by a direct focus on the original problem.

2.3.1 Variational Formulation

The general nonlinear problem given by the system of differential equations in (2.1) is considered, minimizing a performance index with final and integral cost functions as shown in (2.2). This problem is called the *Bolza* problem. It is also assumed that the problem has initial conditions and free-final time and state. The problem is to solve the system of differential equations

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t) \tag{2.17}$$

such that the performance index

$$J(\mathbf{u}(t)) = S(\mathbf{x}(t_f), t_f) + \int_{t_0}^{t_f} V(\mathbf{x}(t), \mathbf{u}(t), t)\ dt \tag{2.18}$$

is minimized, whilst the following boundary conditions are satisfied,

$$\begin{cases} \mathbf{x}(t_0) = \mathbf{x_0} \\ \mathbf{x}(t_f) \text{ is free and } t_f \text{ is free.} \end{cases} \tag{2.19}$$

Notice that, if S is considered to be a differentiable function, the final cost term of the performance index equation may be rewritten as

$$S(\mathbf{x}(t_f), t_f) = \int_{t_0}^{t_f} \frac{d}{dt} \left[S(\mathbf{x}(t), t) \right] \, dt + S(\mathbf{x}(t_0), t_0) \tag{2.20}$$

so that the performance index becomes

$$J(\mathbf{u}(t)) = \int_{t_0}^{t_f} \left\{ V(\mathbf{x}(t), \mathbf{u}(t), t) + \frac{d}{dt} \left[S(\mathbf{x}(t), t) \right] \right\} dt + S(\mathbf{x}(t_0), t_0). \tag{2.21}$$

However, since the initial state $\mathbf{x}(t_0)$ and the initial time t_0 are fixed, the initial cost term does not affect the minimization of J. Therefore, only the integral term of the functional needs to be considered,

$$J(\mathbf{u}(t)) = \int_{t_0}^{t_f} \left\{ V(\mathbf{x}(t), \mathbf{u}(t), t) + \frac{d}{dt} \left[S(\mathbf{x}(t), t) \right] \right\} dt. \tag{2.22}$$

Using the chain rule of differentiation, it is possible to rewrite the term containing the time derivative of S in (2.22) as

$$\frac{d}{dt} \left[S(\mathbf{x}(t), t) \right] = \left[\frac{\partial S(\mathbf{x}(t), t)}{\partial \mathbf{x}} \right]^T \dot{\mathbf{x}}(t) + \frac{\partial S(\mathbf{x}(t), t)}{\partial t} \tag{2.23}$$

Then, making use of (2.23), the performance index functional (2.22) can be rewritten and it becomes

$$J(\mathbf{u}(t)) = \int_{t_0}^{t_f} \left\{ V(\mathbf{x}(t), \mathbf{u}(t), t) + \left[\frac{\partial S(\mathbf{x}(t), t)}{\partial \mathbf{x}} \right]^T \dot{\mathbf{x}}(t) + \frac{\partial S(\mathbf{x}(t), t)}{\partial t} \right\} dt. \tag{2.24}$$

Now, the functional J is considered to be on its minimum and, therefore, the state and control variables to be optimum, $\mathbf{x}^*(t)$ and $\mathbf{u}^*(t)$. Also, the differential equation

constraints are introduced to form the augmented functional

$$J_a(\mathbf{u}^*(t)) = \int_{t_0}^{t_f} \left\{ V(\mathbf{x}^*(t), \mathbf{u}^*(t), t) + \left[\frac{\partial S(\mathbf{x}^*(t), t)}{\partial \mathbf{x}}\right]^T \dot{\mathbf{x}}^*(t) + \frac{\partial S(\mathbf{x}^*(t), t)}{\partial t} + \boldsymbol{\lambda}^{*T}(t) \left[\mathbf{f}(\mathbf{x}^*(t), \mathbf{u}^*(t), t) - \dot{\mathbf{x}}^*(t)\right] \right\} dt \tag{2.25}$$

by using the optimal Lagrange multiplier vector $\boldsymbol{\lambda}^*(t)$ and the dynamic equations in (2.17). For convenience, the Hamiltonian $\mathcal{H}$ is also introduced as function of the Lagrange multiplier, the state and control variables, such that,

$$\mathcal{H}(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\lambda}(t), t) = V(\mathbf{x}(t), \mathbf{u}(t), t) + \boldsymbol{\lambda}^T \left[\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t)\right] \tag{2.26}$$

and equation (2.25) can be rewritten as

$$\begin{aligned} J_a(\mathbf{u}^*(t)) &= \int_{t_0}^{t_f} \left\{ \mathcal{H}(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t) + \frac{\partial S(\mathbf{x}^*(t), t)}{\partial \mathbf{x}}\, \dot{\mathbf{x}}^*(t) \right. \\ &\qquad \left. + \frac{\partial S(\mathbf{x}^*(t), t)}{\partial t} - \boldsymbol{\lambda}^{*T}(t) \dot{\mathbf{x}}^*(t) \right\} dt \\ &= \int_{t_0}^{t_f} \mathcal{L}(\mathbf{x}^*(t), \dot{\mathbf{x}}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t)\, dt. \end{aligned} \tag{2.27}$$

As defined in the boundary conditions of the problem (2.19), it is assumed that both final state $\mathbf{x}(t_f)$ and final time t_f are free. Later it will also be shown that this boundary condition is the most general form and other cases of final boundary condition may be derived from this one. Furthermore, the variation of J_a is obtained by introducing the variations $\delta\mathbf{x}, \delta\dot{\mathbf{x}}, \delta\mathbf{u}, \delta\boldsymbol{\lambda}$ and δt_f in (2.27), resulting in

$$\begin{aligned}\delta J_a(\mathbf{u}^*(t)) = \int_{t_0}^{t_f} \Bigg\{ & \left[\frac{\partial \mathcal{H}}{\partial \mathbf{x}}(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t) + \frac{\partial^2 S}{\partial \mathbf{x} \partial \mathbf{x}}(\mathbf{x}^*(t), t) + \frac{\partial^2 S}{\partial \mathbf{x} \partial t}(\mathbf{x}^*(t), t)\right] \delta \mathbf{x}(t) \\ & + \left[\frac{\partial \mathcal{H}}{\partial \mathbf{u}}(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t)\right] \delta \mathbf{u} \\ & + \left[\frac{\partial \mathcal{H}}{\partial \boldsymbol{\lambda}}(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t) - \dot{\mathbf{x}}(t)\right] \delta \boldsymbol{\lambda}(t) \\ & + \left[\frac{\partial S}{\partial \mathbf{x}}(\mathbf{x}^*(t), t) - \boldsymbol{\lambda}(t)\right] \delta \dot{\mathbf{x}}(t) \Bigg\} \, dt \\ & + \left[\mathcal{L}(\mathbf{x}^*(t_f), \dot{\mathbf{x}}^*(t_f), \mathbf{u}(t), \boldsymbol{\lambda}(t_f), t_f)\right] \delta t_f = 0 \ .\end{aligned} \tag{2.28}$$

The term outside the integral in (2.28) appears by using the first variation and expanding about each variable in a Taylor series as shown in [Kirk, 1998]. Also, notice that the variation δJ_a is zero, since it is evaluated with optimal values and, therefore, on an extremum. Next, the integration by parts on the coefficient of $\delta \dot{\mathbf{x}}(t)$, the last term inside the integral, is used. Since the state is prescribed at t_0 and free only at t_f, the variation $\delta \mathbf{x}(t_0)$ is zero and (2.28) may be rewritten as

$$\begin{aligned}\delta J_a(\mathbf{u}^*(t)) = \int_{t_0}^{t_f} \Bigg\{ & \left[\frac{\partial \mathcal{H}}{\partial \mathbf{x}}(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t) + \frac{\partial^2 S}{\partial \mathbf{x} \partial \mathbf{x}}(\mathbf{x}^*(t), t) + \frac{\partial^2 S}{\partial \mathbf{x} \partial t}(\mathbf{x}^*(t), t)\right. \\ & \left. - \frac{d}{dt}\frac{\partial S}{\partial \mathbf{x}}(\mathbf{x}^*(t), t) + \dot{\boldsymbol{\lambda}}(t)\right] \delta \mathbf{x}(t) + \left[\frac{\partial \mathcal{H}}{\partial \mathbf{u}}(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t)\right] \delta \mathbf{u}(t) \\ & + \left[\frac{\partial \mathcal{H}}{\partial \boldsymbol{\lambda}}(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t) - \dot{\mathbf{x}}(t)\right] \delta \boldsymbol{\lambda}(t) \Bigg\} \, dt \\ & + \left[\mathcal{L}(\mathbf{x}^*(t_f), \dot{\mathbf{x}}^*(t_f), \mathbf{u}(t), \boldsymbol{\lambda}(t_f), t_f)\right] \delta t_f \\ & + \left[\frac{\partial S}{\partial \mathbf{x}}(\mathbf{x}^*(t_f), t_f) - \lambda(t_f)\right] \delta \mathbf{x}(t_f).\end{aligned} \tag{2.29}$$

The relation (2.23) is used for the time derivative of the S term in (2.29). Moreover, it is assumed that the second partial derivatives of S are continuous and that the order of differentiation can be interchanged, then, the terms of S inside the integral add to zero. On an extremum, maximum or minimum, the integral of (2.29) must vanish regardless of the boundary condition. Then, the integral term becomes

$$\int_{t_0}^{t_f} \Bigg\{ \left[\frac{\partial \mathcal{H}}{\partial \mathbf{x}}(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t) + \dot{\boldsymbol{\lambda}}(t) \right] \delta\mathbf{x}(t) + \left[\frac{\partial \mathcal{H}}{\partial \mathbf{u}}(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t) \right] \delta\mathbf{u} + \left[\frac{\partial \mathcal{H}}{\partial \boldsymbol{\lambda}}(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t) - \dot{\mathbf{x}}(t) \right] \delta\boldsymbol{\lambda}(t) \Bigg\} dt. \tag{2.30}$$

Using the differential equation (2.17) and the definition of the Hamiltonian (2.26), it is shown that the coefficient of $\delta\boldsymbol{\lambda}(t)$ is zero on an extremum, resulting in the *state equations* that are written as

$$\dot{\mathbf{x}}^*(t) = \frac{\partial \mathcal{H}}{\partial \boldsymbol{\lambda}}(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t). \tag{2.31}$$

In addition, the Lagrange multipliers are arbitrary, so they can be selected in such a way that the coefficient of $\delta\mathbf{x}(t)$ becomes equal to zero, that is,

$$\dot{\boldsymbol{\lambda}}^*(t) = -\frac{\partial \mathcal{H}}{\partial \mathbf{x}}(\mathbf{x}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t). \tag{2.32}$$

The equations (2.32) are called the *costate equations* and the Lagrange multiplier $\boldsymbol{\lambda}(t)$ will be henceforth called the *costate*. Finally, the term containing the variation $\delta\mathbf{u}(t)$ is independent and its coefficient must be zero. Thus,

$$0 = \frac{\partial \mathcal{H}}{\partial \mathbf{u}}(\mathbf{x}^*(t), \mathbf{u}^*(t), \lambda^*(t), t). \tag{2.33}$$

Moreover, it is shown in Fig.2.2 that the variation between the final state $\delta\mathbf{x}_f$ and the state at the final time $\delta\mathbf{x}(t_f)$ are different, because of the free final time t_f. The variation $\delta\mathbf{x}_f$ can be defined as

$$\delta\mathbf{x}_f = \delta\mathbf{x}(t_f) + \dot{\mathbf{x}}(t_f)\delta t_f. \tag{2.34}$$

Substituting equation (2.34) in the term outside the integral of (2.29), knowing that, on an extremal, the variation δJ_a and the integral of (2.29) for any $t \in [t_0, t_f]$ are zero, the following equations are obtained,

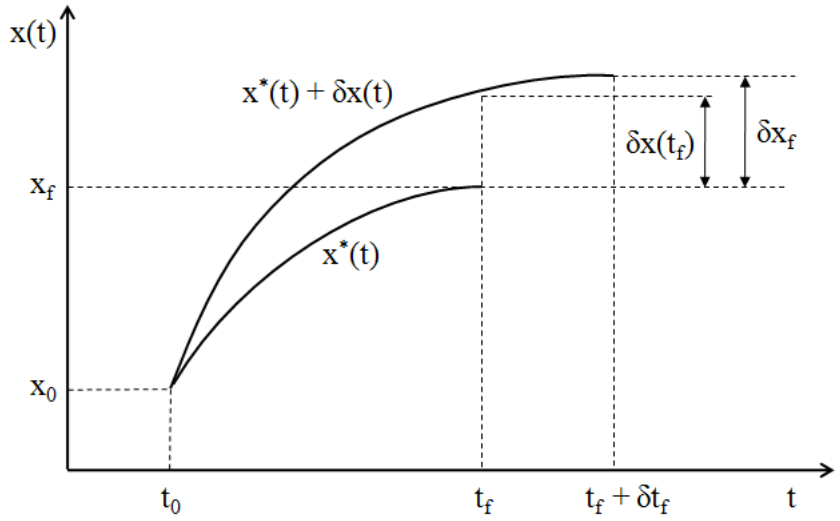


Figure 2.2: Variations for free-final time and free-final state system [Naidu, 2003].

$$0 = \left[\mathcal{L}(\mathbf{x}^*(t), \dot{\mathbf{x}}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t) - \left[\frac{\partial S}{\partial \mathbf{x}}(\mathbf{x}^*(t_f), t_f)\dot{\mathbf{x}}^*(t_f) - \boldsymbol{\lambda}^{*T}\dot{\mathbf{x}}(t_f)\right]\right]\delta t_f + \left[\frac{\partial S}{\partial \mathbf{x}}(\mathbf{x}^*(t_f), t_f) - \boldsymbol{\lambda}^*(t_f)\right]\delta \mathbf{x}_f. \tag{2.35}$$

Next, using the definition of the functional $\mathcal{L}(\mathbf{x}^*(t), \dot{\mathbf{x}}^*(t), \mathbf{u}^*(t), \boldsymbol{\lambda}^*(t), t)$ in (2.27), the resulting equation is

$$0 = \left[\mathcal{H}(\mathbf{x}^*(t_f), \mathbf{u}^*(t_f), \boldsymbol{\lambda}^*(t_f), t_f) + \frac{\partial S}{\partial t}(\mathbf{x}^*(t_f), t_f)\right]\delta t_f + \left[\frac{\partial S}{\partial \mathbf{x}}(\mathbf{x}^*(t_f), t_f) - \boldsymbol{\lambda}^*(t_f)\right]\delta \mathbf{x}_f. \tag{2.36}$$

Equation (2.36) is the fundamental equation for applying the boundary condition at the free-final point and it admits a variety of situations, which will be discussed later.

Equations (2.31) to (2.33) are the necessary conditions to determine the optimal control and, together with the boundary condition equation at the final point (2.36), are sufficient to solve an optimal control problem. If the problem is considered to have n state variables and m control variables, the variational approach will lead to the solution of $2n$ equations referent to (2.31) and (2.32), plus m equations of (2.33). Also, a set of n equations $\mathbf{x}^*(t_0) = \mathbf{x}_0$ and an additional set of n or $n+1$ equations, whether or not t_f is specified, from (2.36). The problem of optimal control results, therefore, in a two-point boundary value problem and, very often, nonlinearity also occurs. The

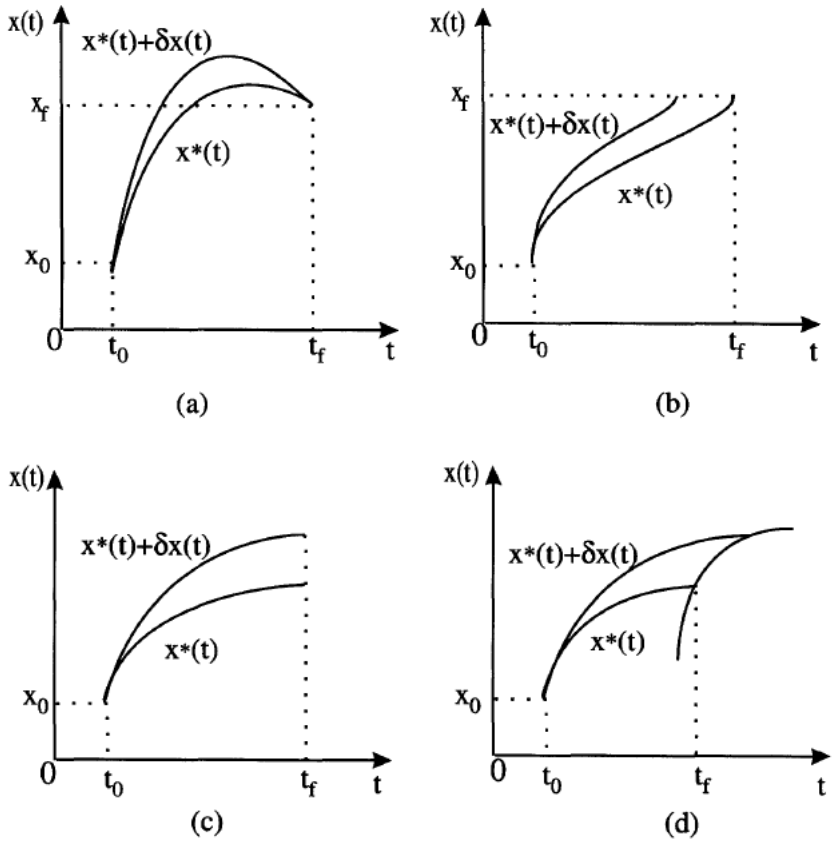


Figure 2.3: Different types of final boundary condition: (a) Fixed final time and state; (b) Free final time and fixed final state; (c) Fixed final time and free final state; (d) Free final time and state [Naidu, 2003].

appearance of nonlinearity will be later seen in chapter 4 and often requires numerical techniques for its solution, which are discussed in section 6.

2.3.2 Boundary Conditions

As mentioned in section 2.3.1, it is assumed that the initial conditions of the problem are completely defined, that is, time and state are known *a priori* at the initial stage. However, questions are still posed to the treatment of the final boundary conditions. The optimal control problem involves the solution of the state (2.31), costate (2.32) and control (2.33) equations, the n equations of the initial state, $\mathbf{x}(t_0) = \mathbf{x}_0$, and the solution of the final boundary condition (2.36). It is now presented the different cases of final condition that may arise in optimal control problems, as illustrated in Fig.2.3.

The first type of problem studied is when both final time t_f and final state $\mathbf{x}(t_f)$ are given. In this case, the variations $\delta\mathbf{x}_f$ and δt_f are zero. Then, the additional n equations necessary for the solution of the problem come from the given values at the final point,

$$\mathbf{x}^*(t_f) = \mathbf{x}_f. \tag{2.37}$$

Still assuming the final state to be fixed, it is considered that the final time is now free, as illustrated in Fig.2.3b. Then, the variation δt_f is zero and may be substituted in (2.36). Moreover, since the variation $\delta\mathbf{x}_f$ is arbitrary, the set of n equations required are given by

$$\left(\mathcal{H} + \frac{\partial S}{\partial t}\right)_{*t_f} = 0. \tag{2.38}$$

Here, the notations of the equation were simplified and the terms ${}^*t_f$ means that the functionals are evaluated for the optimal values and at the final time.

Now, it is considered that the final time is specified and the state is free at the final point, as shown in Fig.2.3c. In this case, it is the variation $\delta\mathbf{x}_f$ that is zero and substituted into (2.36). Then, the equations that needs to be satisfied are

$$\boldsymbol{\lambda}^*(t_f) = \left(\frac{\partial S}{\partial \mathbf{x}}\right)_{*t_f}. \tag{2.39}$$

At last, the condition of free final time and free final state may also be found in optimal control problems. This condition was the motivation for the derivation of (2.36) and, in this case, both variations δt_f and $\delta\mathbf{x}_f$ are arbitrary. Therefore, their coefficients must be zero and it results that,

$$\left(\mathcal{H} + \frac{\partial S}{\partial t}\right)_{*t_f} = \left(\frac{\partial S}{\partial \mathbf{x}} - \boldsymbol{\lambda}\right)_{*t_f} = 0. \tag{2.40}$$

Also, notice that if $S = 0$, that is, if the performance index contains only the integral term, (2.40) becomes

$$\mathcal{H}(\mathbf{x}^*(t_f), \mathbf{u}^*(t_f), \boldsymbol{\lambda}^*(t_f), t_f) = \boldsymbol{\lambda}^*(t_f) = 0. \tag{2.41}$$

The conditions for the free final time and free final state shown above consider the case where the variables t_f and $\mathbf{x}(t_f)$ are independent. Variations of this final boundary conditions may be found, when the final state lies on a curve or a surface that changes with the final time. Further discussion on the treatment of the boundary condition necessary for the solution of the optimal control problem may be found in [Kirk, 1998] and [Naidu, 2003].

3 THE MICRO-SWIMMER PROBLEM

In an usual swimming mechanism, the swimmer obtains a forward movement by performing cyclic movements and, because of inertia, the motion of the second half of the period does not cancel the motion of the first half. This is the mechanism seen, for example, when a human being swims in a swimming pool or when fish swim in the ocean. However, in the microscopic world, such as, when observing the propagation of a bacteria in a certain organic media or when dealing with micro-robots for nano-medicine applications, the usual swimming mechanism is not valid anymore. In a micro-swimmer motion, the inertia term is generally negligible if compared to the viscous term. Therefore, it is a case of low Reynolds number and the motion is governed by the Stokes equations. It was shown by Purcel that, at low Reynolds number, a scallop with a single hinge cannot advance through a reciprocal opening and closing motion, which is also known as the "scallop theorem" [Purcel, 1977]. Then, for a micro-swimmer to propagate in a viscous media, it needs to perform a non-reciprocal movement to break the time-reversal symmetry of the Stokes equations. Otherwise, all the motion that the scallop obtains when opening the hinge will be recovered when the scallop closes it at the end of each cycle, returning the scallop to its initial position.

The swimmer introduced by Najafi and Golestanian [Najafi and Golestanian, 2004] is used in this work as a model problem. The simplest one-dimensional form of the swimmer is presented in Fig.3.1, where all three spheres have equal radius a. The swimmer motion is governed by the relative displacement x and y between spheres $1-2$ and $2-3$ and, consequently, the motion of its center of mass c.

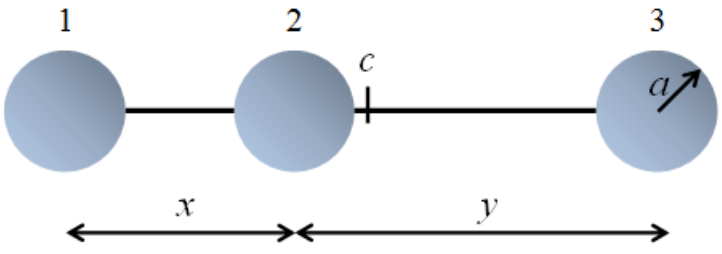


Figure 3.1: Geometry of the thee-spheres micro-swimmer.

In this chapter the motion equations of the three-spheres micro-swimmer is presented and the kinematics involved in the problem. It is also discussed the Oseen-tensor approach of the motion that can be used when dealing with a low Reynolds flow [Batchelor, 1975], due to the linearity of the Stokes equations. Furthermore, a description of a non-reciprocal stroke for the three-spheres swimmer is presented. Finally, a motion analysis is performed to verify the methodology used and to check the existence of a net displacement of the swimmer after one completed cycle.

3.1 EQUATIONS OF MOTION

3.1.1 Governing equations

To derive the equations of motion of the micro-swimmer problem, one should first start discussing the general equations involved in the problem. From the fluid dynamic theory, it is known that the governing equations of an arbitrary fluid motion are the Navier-Stokes equations. For a viscous incompressible fluid, with density ρ, dynamic viscosity μ and without the presence of body forces, the Navier-Stokes equations are given by

$$\rho\left(\frac{\partial \mathbf{v}_f}{\partial t} + \mathbf{v}_f \cdot \nabla \mathbf{v}_f\right) = -\nabla p + \mu \nabla^2 \mathbf{v}_f \tag{3.1}$$

$$\nabla \cdot \mathbf{v}_f = 0. \tag{3.2}$$

Here, $\mathbf{v}_f$ is the vector of the fluid velocities in the directions of the coordinate system (in this work the fluid velocity vector is represented by $\mathbf{v}_f$ to avoid confusion with the control variables vector $\mathbf{u}$ used in chapter 2). Equation (3.1) is called the momentum equation and expresses the balance between the forces and the rate of changes of the linear momentum. The second Navier-Stokes equation, equation (3.2), is known as the continuity equation and assures the flow to be volume preserving, in the case of an incompressible flow.

The Navier-Stokes equations present a nonlinearity due to the inertial term on the right hand side of (3.1). However, at a sufficiently low Reynolds number, dimensionless number that measures the ratio between the inertial and viscous terms, the

inertial term of the Navier-Stokes equations can be neglected. This defines the regime of Stokes-flow, as it is the case of the micro-swimmer problem. Then, the fluid velocity field $\mathbf{v}_f$ is governed by the Stokes equations, which can be written as

$$\mu\nabla^2\mathbf{v}_f - \nabla p = 0 \tag{3.3}$$

$$\nabla \cdot \mathbf{v}_f = 0, \tag{3.4}$$

where μ is the dynamic viscosity of the fluid and p is the pressure field in the medium. Without the inertial terms, the above governing equations have a linear behavior and this allows an alternative approach to the micro-swimmer problem as it is discussed next.

3.1.2 The Oseen-tensor formulation

It is assumed that each sphere moves in the fluid with a velocity vector $\mathbf{v}_i$, with i being the index of each sphere. By solving the Stokes equations (3.3) and (3.4), the fluid velocity is obtained and, hence, the corresponding stress tensor. Moreover, one can retrieve the force vector $\mathbf{f}_i$ acting on each sphere from the stress tensor. Since the Stokes equations are linear and, therefore, the stress tensor and velocity field are linearly related, the velocity vector can be written as a linear combination of the forces acting on each sphere, such that,

$$\mathbf{v}_i = \sum_{j=1}^{3} \mathbf{S}_{ij}\,\mathbf{f}_j \qquad \text{with} \quad i = 1, 2, 3\ . \tag{3.5}$$

The tensor $\mathbf{S}_{ij}$ is symmetric and it is known as the Oseen tensor. It depends on the viscosity, the geometry of the immersed bodies and their relative orientation. From the hydrodynamic interaction of particles in a viscous media, the Oseen tensor, in its general form, is defined by [Batchelor, 1975],

$$\mathbf{S}_{ij} = \frac{1}{6\pi\mu R}\left[A_{ij}(\varphi)\frac{\mathbf{r}\otimes\mathbf{r}}{r^2} + B_{ij}(\varphi)\left(\mathbf{I} - \frac{\mathbf{r}\otimes\mathbf{r}}{r^2}\right)\right]. \tag{3.6}$$

Assuming that the distances between the spheres are sufficiently large, the components $A_{ij}(\varphi)$ and $B_{ij}(\varphi)$ were presented in [Najafi and Golestanian, 2004] and may be

given as

$$A_{ij} = \begin{cases} 1 + O(\varphi^4), & i = j \\ \dfrac{3}{2}\varphi_{ij} + O(\varphi^3), & i \neq j \end{cases} \tag{3.7}$$

$$B_{ij} = \begin{cases} 1 + O(\varphi^4), & i = j \\ \dfrac{3}{4}\varphi_{ij} + O(\varphi^3), & i \neq j \end{cases} \tag{3.8}$$

where $\varphi_{ij} = a/r_{ij}$ is a dimensionless quantity, with a being the radius of the spheres and r_{ij} the norm of the vector $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$, which is the distance between the ith and jth spheres. For simplicity, only the translational motion is being considered because the problem is one dimensional, although, the rotational motion can be taken into account in a similar way.

Equations (3.6) to (3.8) define the relation between the velocities and forces of the spheres for a general two or three dimensional problem. However, for the one dimensional model problem studied, the terms $\mathbf{S}_{ij}$ in (3.5) become scalar values. Gathering the components S_{ij} in the matrix $\mathbf{S}(x, y)$, also called the Oseen matrix, it is possible to rewrite the linear system of equations of the the spheres as

$$\begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \mathbf{S}(x, y) \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix} . \tag{3.9}$$

The Oseen matrix in (3.9) may be simplified as $\mathbf{S}(x, y) = \mathbf{S}_\infty(x, y) + \mathbf{E}(x, y)$, where $\mathbf{E}(x, y)$ is the error associated to the power of φ. For the one dimensional case, it can be shown that the only nonzero term in (3.6) is A_{ij}. This occurs because the term A_{ij} relates the velocity to the forces parallel to the motion, while the term B_{ij} relates the velocity to perpendicular forces. Since the motion is one dimensional, there are only forces normal to the motion, that is, acting on the horizontal axis of Fig.3.1. Hence, the $\mathbf{S}_\infty(x, y)$ matrix with only the A_{ij} term can be written as follows,

$$\mathbf{S}_\infty = \frac{1}{\pi\mu}\begin{pmatrix} \frac{1}{6a} & \frac{1}{4x} & \frac{1}{4(x+y)} \\ \frac{1}{4x} & \frac{1}{6a} & \frac{1}{4y} \\ \frac{1}{4(x+y)} & \frac{1}{4y} & \frac{1}{6a} \end{pmatrix}. \tag{3.10}$$

Notice that for $\mathbf{S}_\infty(x, y)$ to be considered, it is assumed that the values φ_{ij} are sufficiently small, that is, the magnitude of the distances between the spheres are sufficiently larger than their radius. If that is the case, the error associated to the power of φ in (3.7) can be neglected. Alternative approximation for the Oseen tensor may be found in [Rotne and Prager, 1969]. The Oseen matrix (3.10) together with (3.9) define the hydrodynamic interaction between the spheres in a viscous media, as it can be seen in [Batchelor, 1975] and [Najafi and Golestanian, 2004], and it will be used throughout this work as the constitutive equation of the three-spheres micro-swimmer problem.

3.1.3 Kinematics of the swimmer

By definition, a swimming mechanism must be capable of self-propulsion, that is, it is capable of propagating in the fluid without the aid of external force and, hence, the system is force-free. Therefore, the energy expended by the swimmer for the relative motion of the spheres is generated internally and, assuming that there are no body forces such as gravity, the sum of forces acting on the spheres must be zero [Alouges et al., 2008]:

$$f_1 + f_2 + f_3 = 0\ . \tag{3.11}$$

Moreover, since all three spheres are identical, same radius and same mass, the center of mass is the average of the positions x_1, x_2, x_3 of spheres 1, 2 and 3, respectively. Then, the velocity of the center of mass c of the swimmer is the average of the velocities of the spheres and may be given as

$$\dot{c} = \frac{dc}{dt} = (v_1 + v_2 + v_3)/3. \tag{3.12}$$

By performing a simple geometric analysis of the three-spheres swimmer presented in Fig.3.1, it is possible to establish that $x = x_2 - x_1$ and $y = x_3 - x_2$. Then, one may write the velocities of the spheres 1 and 3 in terms of the velocity of sphere 2 and the rate of change of the relative displacements $\dot{x}$ and $\dot{y}$ between spheres $1-2$ and $2-3$,

$$\begin{cases} v_1 = v_2 - \dot{x} \\ v_3 = v_2 + \dot{y} \end{cases} \tag{3.13}$$

The relation in (3.13) is important to reduce the number of state variables in the system. Also, as it will be seen later, it gives the two remaining equations to find the functions that relate the center of mass c with the rate of change of the relative displacements $\dot{x}$ and $\dot{y}$.

Due to the symmetry of the the Oseen matrix $\mathbf{S}(x, y)$ and because the system satisfies (3.11), it can be shown that the center of mass c satisfies the following ODE [Alouges et al., 2008]:

$$\frac{dc}{dt} = V_x(x, y)\frac{dx}{dt} + V_y(x, y)\frac{dy}{dt}, \tag{3.14}$$

where the functions $V_x(x, y)$ and $V_y(x, y)$ can be computed from $\mathbf{S}(x, y)$ by solving (3.11) to (3.13), together with the system of equations presented in (3.9). The ODE in (3.14) is the governing equation of the micro-swimmer problem and by introducing a certain history of the pair (x, y) in the equation, it is possible to obtain the net displacement of the swimmer. The main difficulty comes from the functions $V_x(x, y)$ and $V_y(x, y)$ that must be evaluated for each instant (x, y) and are extremely nonlinear.

For the one dimensional three-sphere swimmer, where the hydrodynamic interaction of the spheres is driven by (3.10), it is possible to obtain the functions $V_x(x, y)$ and $V_y(x, y)$ of the micro-swimmer by solving the following system of equations

$$\begin{cases} v_1 = v_2 - \dot{x} \\ v_3 = v_2 + \dot{y} \\ \dot{c} = (v_1 + v_2 + v_3)/3 \\ 0 = f_1 + f_2 + f_3 \\ \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \frac{1}{\pi\mu} \begin{pmatrix} \frac{1}{6a} & \frac{1}{x} & \frac{1}{4(x+y)} \\ \frac{1}{x} & \frac{1}{6a} & \frac{1}{y} \\ \frac{1}{4(x+y)} & \frac{1}{y} & \frac{1}{6a} \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix} \end{cases}$$

With the computational aid of the software *Wolfram Mathematica* and its symbolic tool, the system above was solved in terms of the variables $\dot{c}$, v_1, v_2, v_3, f_1, f_2 and f_3. The functions $V_x(x,y)$ and $V_y(x,y)$ are defined as the coefficients multiplying $\dot{x}$ and $\dot{y}$, respectively, form the resulting function of $\dot{c}(x,y)$. Using the *simplify* function of the program, the following equations for $V_x(x,y)$ and $V_y(x,y)$ were obtained

$$V_x(x,y) = \frac{a\left(-2xy\,(x+y)\left(2x^2-y^2\right)+a\left(6x^4+6x^3y-3x^2y^2-3xy^3-3y^4\right)\right)}{-12x^2y^2\,(x+y)^2+12axy\,(x+y)\,(x^2+3xy+y^2)+9a^2\,(x^4-2x^3y-5x^2y^2-2xy^3+y^4)} \tag{3.15}$$

and

$$V_y(x,y) = \frac{a\left(-2xy\,(x+y)\left(x^2-2y^2\right)+3a\left(x^4+x^3y+x^2y^2-2xy^3-2y^4\right)\right)}{-12x^2y^2\,(x+y)^2+12axy\,(x+y)\,(x^2+3xy+y^2)+9a^2\,(x^4-2x^3y-5x^2y^2-2xy^3+y^4)} \tag{3.16}$$

As it is seen, equations (A.1) and (3.16) are extremely nonlinear and it is important to notice that the fluid viscosity μ disappears, making the functions dependent on the geometry of the swimmer only, radius a and distances x and y. Using (A.1) and (3.16) in (3.14), and introducing a certain history of the pair (x,y) it is possible to obtain the velocity of the center of mass $\dot{c}$ for each instant (x,y). The velocities of each sphere may also be obtained, by using the following relation

$$\mathbf{v} = \begin{Bmatrix} v_1 \\ v_2 \\ v_3 \end{Bmatrix} = \frac{1}{3} \begin{Bmatrix} 3\dot{c} - (2\dot{x} + \dot{y}) \\ 3\dot{c} + (\dot{x} - \dot{y}) \\ 3\dot{c} + (\dot{x} + 2\dot{y}) \end{Bmatrix}. \tag{3.17}$$

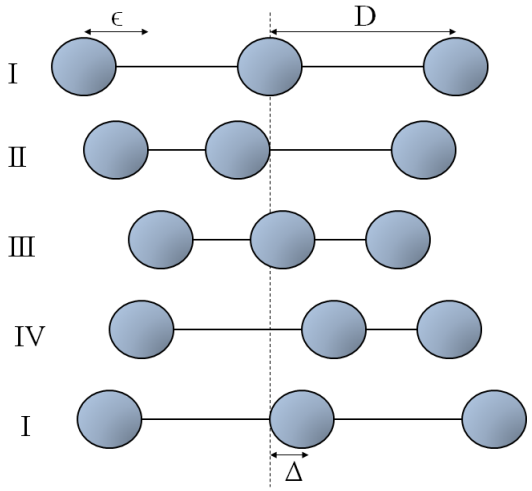


Figure 3.2: The four-stage non-reciprocal stroke of the three-sphere swimmer.

It is also possible to recover the displacement history of each sphere as well as that of the center of mass by using numerical integration. It is expected that at the end of each complete cycle the final position of sphere 2 coincides with the center of mass. Hence, the net displacement obtained after each cycle may be measured by the difference of final and initial positions of sphere 2 after the completion of the cycle.

3.2 VERIFICATION OF THE SWIMMER'S MOTION

To check the validity of the equations presented in section 3.1, the four-stage stroke for the three-spheres swimmer is presented. A simple analysis is performed by introducing the four-stage motion and the results will be checked with the available literature.

3.2.1 The four-stage stroke

With the objective of breaking the time reversal property of the Stokes equations (3.3) and (3.4), as mentioned previously, Najafi and Golestanian introduced a non-reciprocal motion for the three-sphere swimmer [Najafi and Golestanian, 2004]. The stroke is subdivided in four stages and allows the swimmer to obtain a net displacement Δ at the end of each cycle. An illustration of the four-stage stroke is shown in Fig.3.2.

In the four-stage stroke, the rods of the swimmer have initial length D, that is,

$x_0 = y_0 = D$. Moreover, it will be considered that the rate of change of the distance between the spheres will be made with constant velocity w. The steps of the four-stage motion are described below:

$(I \to II)$ In the first stage, the swimmer contracts the left arm of a length ϵ and, due to the hydrodynamic interaction of the spheres, the swimmer moves a short distance to the left.

$(II \to III)$ Then, the right arm is contracted by the same amount while the left arm remains fixed with length $D - \epsilon$. This stage will result in a movement of the swimmer to the right.

$(III \to IV)$ Next, the right arm is kept with fixed length $D-\epsilon$ and the left arm recover its initial length D, which causes the swimmer to move to the right again.

$(IV \to I)$ Finally, the right arm also returns to its initial length and the swimmer move a short distance back to the left.

Since the stages that produce a movement to the right direction are done with the opposite arm contacted, the hydrodynamic interaction are higher and a final displacement Δ to the right is obtained at the end of the cycle. This stroke successfully breaks the time reversal symmetry and it was used by different authors to analyze the micro-swimmer problem. The four-stage stroke will serve as a reference for the following analysis and its efficiency will later be discussed. The four-stage stroke will also be referred as the NG-stroke, in reference to its authors.

3.2.2 Numerical results of the motion analysis.

To solve the motion of the swimmer a simple code was written in *Matlab*, implementing the ODE in (3.14). The trapezoidal rule was used as the numerical integration to obtain the displacement of the spheres, once calculated their velocities. It was chosen a radius $a = 0.05$ mm and an initial distance $D = 0.5$ mm between the spheres, resulting in a ratio $a/D = 0.1$. A history of the relative displacements (x, y), obeying the four-stages stroke, was introduced with 32 integration points, 8 points in each stage. The total time of the stroke was imposed as $T = 1$ s and the velocity w of the

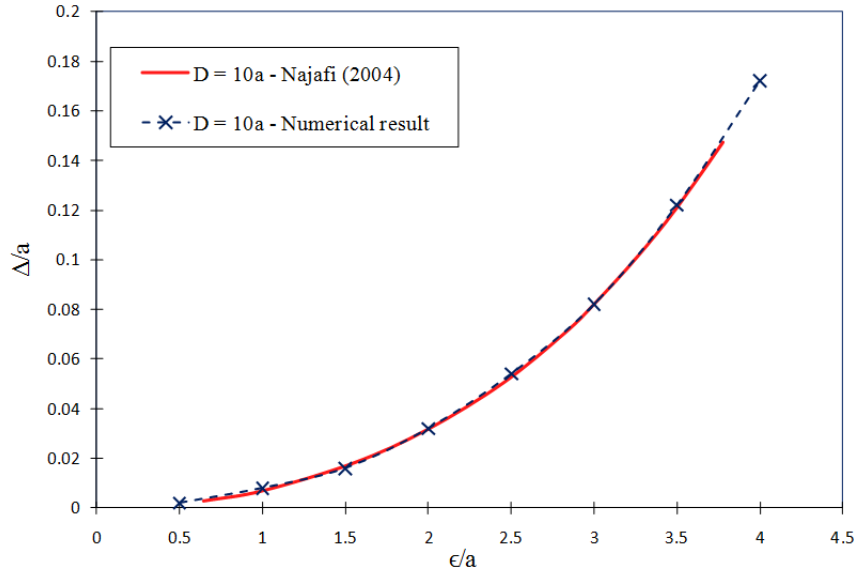
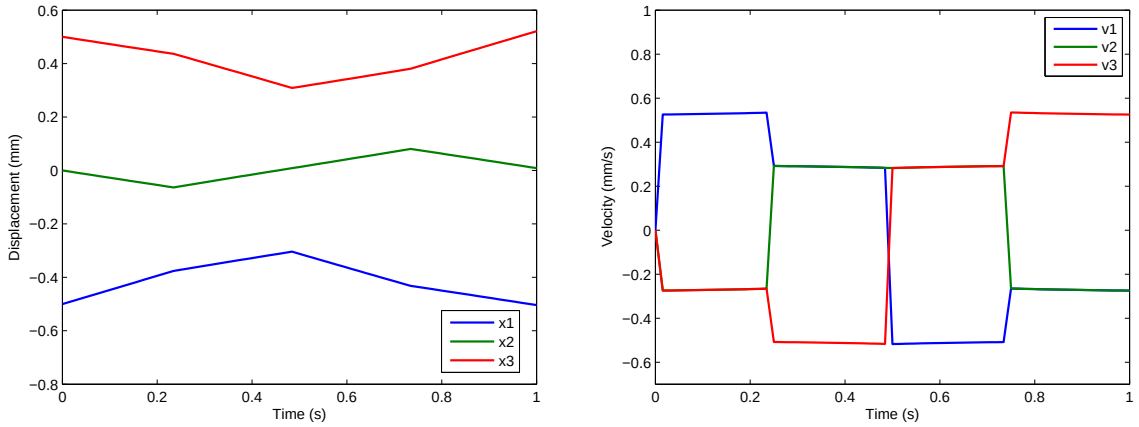


Figure 3.3: Dimensionless displacement of the swimmer obtained in a complete cycle as a function of the dimensionless relative displacement of the spheres.



(a) (b)

Figure 3.4: (a) Displacement and (b) Velocity of the spheres during one complete four-stage stroke, using 32 time steps.

relative motion can be obtained by $w = 4\epsilon/T$, being the relative displacement ϵ the entry value of the numerical solution.

To evaluate the methodology and equations presented in this chapter, a mapping of the net displacement Δ with respect to different relative displacements ϵ was performed. The evolution of the dimensionless net displacement Δ/a with respect to the dimensionless relative displacement ϵ/a was compared with the values presented in

[Najafi and Golestanian, 2004]. The results can be found in Fig.3.3.

The velocity and displacement history of all three spheres during the complete four-stage stroke, for $\epsilon = 0.2$ mm, are presented in Fig.3.4a and 3.4b, respectively.

The results presented in this section demonstrate that the methodology and motion equations used in the analysis of the swimmer is coherent with the model proposed by Najafi and Golestanian, 2004. Furthermore, the error associated with the numerical integration seems not to play an important role in this analysis, since an accurate result was obtained with small number of time steps. Once the validation of the equation used in the analysis tool of the swimmer's motion is done, it is possible to implement the optimal control theory in the problem, as it is shown in the following chapters.

4 SWIMMER AS AN OPTIMAL CONTROL PROBLEM

As presented in chapter 2, the optimal control problem is to define the control variables that solve a system of equations, while minimizing a certain functional called the performance index. In this chapter, the equations of motion of the micro-swimmer problem are rewritten using the optimal control formalism, that is, they are written in terms of the state and control variables. It is also chosen a performance index for the problem that minimizes the energy consumed during one cycle. Special care is given to define the performance index, writing it in terms of the control variables and in agreement with the formalism adopted.

4.1 OPTIMAL CONTROL FORMULATION

4.1.1 State variable representation of the micro-swimmer problem

It was established in section 3.1.3 that the motion of the center of mass c of the swimmer after each stroke is given by (3.14). To relate the micro-swimmer problem with the optimal control problem of chapter 2, it will be defined the velocities $\dot{x}$ and $\dot{y}$ of the relative motion of the spheres as the system's controls u_1 and u_2, respectively. Furthermore, using the same technique used to transform a higher order differential equation into a system of first order ODEs, it can be done a variable substitution to drop out the time derivatives from (3.14). Then, the governing system of equations in the optimal control formalism may be defined as

$$\begin{cases} \dfrac{dx}{dt} = u_1, \\ \dfrac{dy}{dt} = u_2, \\ \dfrac{dc}{dt} = u_1 V_x(x,y) + u_2 V_y(x,y). \end{cases} \tag{4.1}$$

By using this methodology, the state variables of the problem are the distances x and y between the spheres and the center of mass c of the swimmer. For a more elegant formulation, it is also convenient to use a matricial formulation of the system, which can be written as

$$\dot{\mathbf{x}} = \mathbf{F}(x, y)\ \mathbf{u} \tag{4.2}$$

where

$$\dot{\mathbf{x}} = \begin{Bmatrix} \dot{x} \\ \dot{y} \\ \dot{c} \end{Bmatrix} \quad , \quad \mathbf{u} = \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} \tag{4.3}$$

and

$$\mathbf{F}(x, y) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ V_x & V_y \end{bmatrix} . \tag{4.4}$$

The vector $\dot{\mathbf{x}}$ is the time derivative of the position, or state, vector $\mathbf{x} = [x, y, c]^T$ and $\mathbf{u}$ is the control vector that contains the relative velocities of the spheres. The matrix $\mathbf{F}(x, y)$ measures the response of the system's velocity with respect to its controls. In (4.4), the variables V_x and V_y are understood to be $V_x(x, y)$ and $V_y(x, y)$, respectively, but the short notation will be used henceforth for simplicity. Moreover, notice that (4.2) is equivalent to the nonlinear form of (2.1) and, hence, a similar procedure for the optimal control solution, based on the variational approach, will be applied.

4.1.2 Performance index for minimal energy

The energy spent by the swimmer to complete a certain movement can be obtained by the integral of the forces acting on each sphere and their respective velocity fields. However, using (3.9), the energy can be rewritten in terms of the velocity vector $\mathbf{v}$ and the Oseen matrix $\mathbf{S}(x, y)$,

$$E = \int_{t_0}^{t_f} \mathbf{f} \cdot \mathbf{v}\ dt = \int_{t_0}^{t_f} \mathbf{v}^T \cdot \mathbf{S}^{-1}(x, y)\, \mathbf{v}\ dt. \tag{4.5}$$

Notice that the velocity vector $\mathbf{v}$ is the vector containing the velocities of each sphere, and it is not equivalent to the time derivative of the state variables vector $\dot{\mathbf{x}}$ of (4.3).

It is convenient to rewrite the energy equation (4.5) using the optimal control formalism. For that, one must first write the vector of velocities $\mathbf{v}$ defined in (3.17) in terms of the time derivatives of the state variables $\dot{\mathbf{x}}$,

$$\mathbf{v} = \frac{1}{3}\begin{Bmatrix} 3\dot{c} - (2\dot{x} + \dot{y}) \\ 3\dot{c} + (\dot{x} - \dot{y}) \\ 3\dot{c} + (\dot{x} + 2\dot{y}) \end{Bmatrix} = \underbrace{\frac{1}{3}\begin{bmatrix} -2 & -1 & 3 \\ 1 & -1 & 3 \\ 1 & 2 & 3 \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{Bmatrix} \dot{x} \\ \dot{y} \\ \dot{c} \end{Bmatrix}}_{\dot{\mathbf{x}}} . \tag{4.6}$$

Furthermore, substituting (4.2) into (4.6), the velocities of the spheres $\mathbf{v}$ in terms of the control vector $\mathbf{u}$ are obtained, such that,

$$\mathbf{v} = \mathbf{A}\,\dot{\mathbf{x}} \;=\; \mathbf{A}\left(\mathbf{F}(x,y)\;\mathbf{u}\right) \;=\; \mathbf{B}(x,y)\;\mathbf{u}\;, \tag{4.7}$$

where the tensor $\mathbf{B}(x,y)$ is defined as,

$$\mathbf{B}(x,y) = \frac{1}{3}\begin{bmatrix} (3V_x - 2) & (3V_y - 1) \\ (3V_x + 1) & (3V_y - 1) \\ (3V_x + 1) & (3V_y + 2) \end{bmatrix} . \tag{4.8}$$

Finally, substituting (4.7) and (4.8) into (4.5), the equation of the energy consumed in the system becomes

$$\begin{aligned} E &= \int_{t_0}^{t_f} \mathbf{v}^T \cdot \mathbf{S}^{-1}(x,y)\,\mathbf{v}\;dt \\ &= \int_{t_0}^{t_f} \left(\mathbf{B}(x,y)\mathbf{u}\right)^T \mathbf{S}^{-1}(x,y)\;\mathbf{B}(x,y)\mathbf{u}\;\;dt \\ &= \int_{t_0}^{t_f} \mathbf{u}^T \underbrace{\mathbf{B}^T(x,y)\;\mathbf{S}^{-1}(x,y)\;\mathbf{B}(x,y)}_{\mathbf{P}(\mathbf{x},\mathbf{y})}\mathbf{u}\;\;dt\;. \end{aligned}$$

If the energy waste functional is chosen to be minimized in the optimal control problem, the performance index of the problem may be defined as

$$J = \int_{t_0}^{t_f} \mathbf{u}^T \mathbf{P}(x,y)\mathbf{u} \; dt. \tag{4.9}$$

Notice that the performance index (4.9) is written in terms of the control variables and it is equivalent to the functional presented in (2.5). Therefore, the micro-swimmer problem can be defined as a minimum-energy control problem with a *Lagrange* type performance index. Now that the swimmer is formulated as an optimal control problem, the variational approach discussed in section 2.3 can be used to solve the problem, as it is presented next.

4.2 VARIATIONAL APPROACH TO THE MICRO-SWIMMER OPTIMAL CONTROL PROBLEM

4.2.1 Hamiltonian formulation: Matrix form

It is convenient to introduce a general approach to the optimal control micro-swimmer problem. In this section the tensorial form of the optimal control equations are presented. This results in a more elegant form of the equations and a more general representation as well, that is, that can be applied to different variations of the micro-swimmer problem. First, the Hamiltonian function is defined by using (4.2) and (4.9), such that,

$$\mathcal{H}(\mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}) = \mathbf{u}^T \, \mathbf{P}(x,y)\mathbf{u} \; + \; \boldsymbol{\lambda}^T \left[\mathbf{F}(x,y) \, \mathbf{u}\right] \; , \tag{4.10}$$

where $\boldsymbol{\lambda}$ is the vector containing the Lagrange multipliers, or costate variables. The necessary conditions for the optimal control problem were defined in chapter 2 and are given by the derivatives of the Hamiltonian with respect to the control, state and costate variables. Assuming the existence of an optimal control vector $\mathbf{u}^*$ and optimal state vector $\mathbf{x}^*$ that minimize the performance index (4.9), the state and costate equations of the optimal control micro-swimmer problem are given by

$$\frac{\partial \mathcal{H}}{\partial \boldsymbol{\lambda}} = \dot{\mathbf{x}} \;\; \Rightarrow \;\; \dot{\mathbf{x}}^* = \mathbf{F}(x,y) \; \mathbf{u}^* \tag{4.11}$$

and

$$\frac{\partial \mathcal{H}}{\partial \mathbf{x}} = -\dot{\boldsymbol{\lambda}} \;\Rightarrow\; \dot{\boldsymbol{\lambda}} + \mathbf{u}^{*T}\,\frac{\partial \mathbf{P}(x,y)}{\partial \mathbf{x}}\,\mathbf{u}^{*} + \boldsymbol{\lambda}^{T}\frac{\partial \mathbf{F}(x,y)}{\partial \mathbf{x}}\,\mathbf{u}^{*}\;, \tag{4.12}$$

respectively. Each of the equations above consists of a set of n equations, where n is the number of state variables in the system and, hence, the number of costate variables as well. In the costate equations (4.12) the derivatives of the tensors $\mathbf{P}(x,y)$ and $\mathbf{F}(x,y)$ are also known as the gradient of $\mathbf{P}(x,y)$ and $\mathbf{F}(x,y)$. The gradients of the second order tensors result in third order tensors and are written in indicial notation as

$$\frac{\partial \mathbf{P}(x,y)}{\partial \mathbf{x}} = \text{grad}\;\mathbf{P}(x,y) = P_{ij,k} \qquad i,j = 1,2 \;\;;\;\; k = 1,2,3\;,$$
$$\frac{\partial \mathbf{F}(x,y)}{\partial \mathbf{x}} = \text{grad}\;\mathbf{F}(x,y) = F_{mn,k} \qquad m,k = 1,2,3\;;\; n = 1,2\;.$$

The last necessary equations for the solution of the optimal control problem are the control equations, which give the minimization of the Hamiltonian. The control equations are given by the derivatives of the Hamiltonian with respect to the control variables, such that,

$$\frac{\partial \mathcal{H}}{\partial \mathbf{u}} = 0 \;\Rightarrow\; \mathbf{P}(x,y)\,\mathbf{u} * + \boldsymbol{\lambda}^{T}\mathbf{F}(x,y) = 0\;, \tag{4.13}$$

and result in a set of m equations, with m being the number of control variables in the system. Equations (4.11) to (4.13) satisfy the necessary conditions for finding the optimal solution of the problem and result in a system of $2n + m$ equations.

4.2.2 Hamiltonian formulation: Direct solution

The performance index presented in (4.9) presents a multiplication between the tensor $\mathbf{P}(x,y)$ and the control vector $\mathbf{u}$. Notice that $\mathbf{u}$ is a vector of dimensions 2×1 and $\mathbf{P}(x,y)$ is a 2×2 tensor. The result of the inner product inside the integral gives a scalar function dependent on x, y and the controls u_1 and u_2. Expanding the term inside the integral and renaming it as a function $j(x,y,u_1,u_2)$, the performance index may be rewritten as

$$J = \int_{t_0}^{t_f} j(x, y, u_1, u_2)\, dt. \tag{4.14}$$

To proceed with the analysis, it is important to look inside the function $j(x, y, u_1, u_2)$, since later it will be needed to derive $j(x, y, u_1, u_2)$ in terms of the control and state variables. The term $j(x, y, u_1, u_2)$ can be defined as

$$j(x, y, u_1, u_2) = \frac{1}{9\pi\mu}\left\{u_1^2 j_1(x, y) + u_2^2 j_2(x, y) + u_1 u_2 j_3(x, y)\right\}, \tag{4.15}$$

with $j_1(x, y)$, $j_2(x, y)$ and $j_3(x, y)$ being the terms of the tensor $\mathbf{P}(x, y)$, such that, $j_1 = \mathbf{P}(1, 1)$, $j_2 = \mathbf{P}(2, 2)$ and $j_3 = \mathbf{P}(1, 2) + \mathbf{P}(2, 1)$. The functions of j_1, j_1 and j_3 are defined in the appendix A, as well as their required derivatives.

Next, with the function $j(x, y, u_1, u2)$ being defined, it is possible to obtain the Hamiltonian functional of the problem. The Hamiltonian is obtained by applying the Lagrange multipliers λ_1, λ_2 and λ_3 to each term of the right hand side of the system (4.1) and adding them to the term inside the integral of the performance index (A.2), such that,

$$\mathcal{H} = j(x, y, u_1, u_2) + \lambda_1 u_1 + \lambda_2 u_2 + \lambda_3 \left(u_1 V_x + u_2 V_y\right) \ . \tag{4.16}$$

Next, the state, costate and control equations defined in (2.31), (2.32) and (2.33), respectively, are applied. Then, taking the derivatives of the Hamiltonian with respect to the variables $x, y, c, \lambda_1, \lambda_2, \lambda_3, u_1$ and u_2, it is possible to obtain the eight necessary equations for the solution of the micro-swimmer optimal control problem, which are given as follows,

$$\begin{aligned}
1: \quad & \frac{\partial \mathcal{H}}{\partial x} = -\dot{\lambda}_1 \quad \Rightarrow \quad \dot{\lambda}_1 + \frac{\partial j}{\partial x} + \lambda_3 \left(u_1 \frac{\partial V_x}{\partial x} + u_2 \frac{\partial V_y}{\partial x}\right) = 0 \\
2: \quad & \frac{\partial \mathcal{H}}{\partial y} = -\dot{\lambda}_2 \quad \Rightarrow \quad \dot{\lambda}_2 + \frac{\partial j}{\partial y} + \lambda_3 \left(u_1 \frac{\partial V_x}{\partial y} + u_2 \frac{\partial V_y}{\partial y}\right) = 0 \\
3: \quad & \frac{\partial \mathcal{H}}{\partial c} = -\dot{\lambda}_3 \quad \Rightarrow \quad \dot{\lambda}_3 = 0
\end{aligned}$$

$$4: \quad \frac{\partial \mathcal{H}}{\partial \lambda_1} = \dot{x} \quad \Rightarrow \quad \dot{x} - u_1 = 0$$

$$5: \quad \frac{\partial \mathcal{H}}{\partial \lambda_2} = \dot{y} \quad \Rightarrow \quad \dot{y} - u_2 = 0$$

$$6: \quad \frac{\partial \mathcal{H}}{\partial \lambda_3} = \dot{c} \quad \Rightarrow \quad \dot{c} - (u_1 V_x + u_2 V_y) = 0$$

$$7: \quad \frac{\partial \mathcal{H}}{\partial u_1} = \frac{1}{9\pi\mu}(2u_1 j_1 + u_2 j_3) + \lambda_1 + \lambda_3 V_x = 0$$

$$8: \quad \frac{\partial \mathcal{H}}{\partial u_2} = \frac{1}{9\pi\mu}(2u_2 j_2 + u_1 j_3) + \lambda_2 + \lambda_3 V_y = 0$$

Since there are also only eight unknown variables, the equations above are sufficient to obtain the solution of the optimal controls u_1 and u_2. Consequently, the optimal path, that is, the values of x, y and c, can also be obtained. The eight equations presented are the ones that will be use in the numerical analysis of the problem.

Both approaches presented for the formulation of the optimal control equations are equivalent and may be used next to solve the micro-swimmer problem. As it is seen, the problem results in solving a set of differential equations and it will require the use of numerical to obtain the solution. Also, because of the high nonlinearity of some of the equations, it is necessary to use the Newton-Raphson method in the analysis. The numerical techniques and procedure used in this work to solve the optimal control problem are presented in the following chapter, as well as the results obtained in the numerical analysis.

5 NUMERICAL PROCEDURE

The optimal control equations are the control, state and costate equations that results in a system of differential equations, with the number of equations dependent on the number of state and control variables. The first step in the approach adopted for the solution of the problem is to discretize the problem in time. Therefore, the time derivatives that appear in the optimal control equations can be handled by the use of a numerical integration method, such as, the generalized midpoint rule. After the discretization in time, the problem is solved by using a monolithic approach, that is, all the equations will be solved simultaneously. Hence, the optimal controls and the optimal path, given by the state variables, are obtained for the entire time interval at once. This simultaneous solution approach is particulary well suited, since the optimal control problem is a two-point boundary value problem. However, recent works have presented alternative solutions, such as, for example, the Receding Horizon Approach described in [Primbs, 1999] and [Primbs et al., 1999].

In this chapter the numerical procedure adopted in the solution of the optimal control equations is described. The time discretization and the monolithic approach used in the solution of the problem are presented. Examples with a known exact analytical solution are also solved numerically to verify the procedure. Moreover, the Newton-Raphson method is implemented in the procedure to deal with nonlinear problems, such as the micro-swimmer problem, and a nonlinear problem is solved and compared with the results found in the literature. The computational strategy described in this chapter is then used to solve the micro-swimmer problem, as shown in chapter 6.

5.1 DISCRETIZATION OF THE TIME DOMAIN

The generalized midpoint rule is used to discretize the variables of the optimal control problem in time. Notice that the final time might be known or unknown, depending on the boundary conditions of the problem. If the final time is unknown, it will result in an open time domain and the inclusion of the additional boundary condition equations in (2.36). If the final time is known, the problem presents a fixed number of time steps and a fixed time increment Δt. First, the system of equations

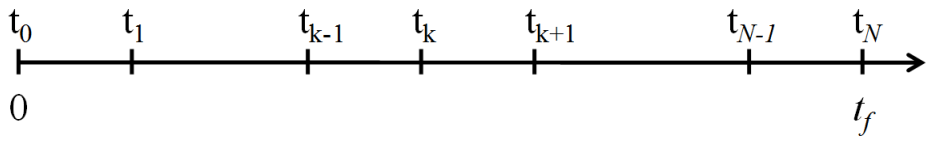


Figure 5.1: Discretization of the time domain.

that results from the variational approach will be rewritten, as seen in chapter 2. The system containing the state, costate and control equations is given by

$$\begin{cases} \dot{\mathbf{x}}^*(t) - \dfrac{\partial \mathcal{H}}{\partial \boldsymbol{\lambda}}(\mathbf{x}^*(t), \mathbf{u}^*(t), \lambda^*(t), t) = 0, \\ \dot{\boldsymbol{\lambda}}^*(t) + \dfrac{\partial \mathcal{H}}{\partial \mathbf{x}}(\mathbf{x}^*(t), \mathbf{u}^*(t), \lambda^*(t), t) = 0, \\ \dfrac{\partial \mathcal{H}}{\partial \mathbf{u}}(\mathbf{x}^*(t), \mathbf{u}^*(t), \lambda^*(t), t) = 0, \end{cases} \tag{5.1}$$

where the Hamiltonian is defined in (2.26). The vector $\mathbf{x} = [x_1, x_2, ..., x_n]^T$ contains the n state variables, $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, ..., \lambda_n]^T$ the costate and $\mathbf{u} = [u_1, u_2, ...u_m]^T$ the m controls of the system. The system (5.1) is a two-point boundary value problem and needs an initial and a final condition to be solved.

The time domain can be discretized in subintervals as represented in Fig.5.1. Using the generalized midpoint rule for the numerical integration, the states, costates and controls are evaluated at a time instant $t_{k+\theta}$, with $t_k \leq t_{k+\theta} \leq t_{k+1}$ and $t_{k+1} - t_k = \Delta t$. Then, the variables may be rewritten as

$$\begin{cases} \mathbf{x}_{k+\theta} = \theta \mathbf{x}_{k+1} + (1-\theta)\mathbf{x}_k, \\ \boldsymbol{\lambda}_{k+\theta} = \theta \boldsymbol{\lambda}_{k+1} + (1-\theta)\boldsymbol{\lambda}_k, \\ \mathbf{u}_{k+\theta} = \theta \mathbf{u}_{k+1} + (1-\theta)\mathbf{u}_k, \end{cases} \tag{5.2}$$

where $\mathbf{x}$ has n state variables and $\mathbf{u}$ m control inputs. Moreover, the value of θ may be chosen, such that, $\theta = 1$ results in the implicit backward Euler scheme, $\theta = 0$ in the explicit forward Euler scheme and $\theta = 1/2$ gives the trapezoidal rule of integration.

The time derivatives of the vectors $\mathbf{x}$ and $\boldsymbol{\lambda}$, which appear in the system of differential equations (5.1), may be replaced by the following general derivative approximation,

$$\begin{cases} \left.\dfrac{d\mathbf{x}}{dt}\right|_{k+\theta} = \dfrac{\mathbf{x}_{k+1} - \mathbf{x}_k}{\Delta t}, \\ \left.\dfrac{d\boldsymbol{\lambda}}{dt}\right|_{k+\theta} = \dfrac{\boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}_k}{\Delta t}. \end{cases} \tag{5.3}$$

Again, the discretized derivatives presented above are evaluated at a time instant $t_{k+\theta}$. The time step increment Δt depends on the number of integration points and on the size of the time interval. The choice of an appropriate time step increment is of great importance for the solution of the problem, especially when dealing with nonlinear problems. The time discretization will increase the number of equations in the system by $N+1$ equations times the size of the original system, $2n+m$ equations, where N is the number of time steps. In the monolithic solution of the problem, the equations of each time step will be assembled in one single matrix and solved simultaneously. Next, the procedure used to solve the system of equations will be discussed, once the system is discretized.

5.2 MONOLITHIC APPROACH TO LINEAR PROBLEMS

Since the variational approach to the optimal control problem results in a two-point boundary value problem, the trivial approach is not to use an iterative solution method, but to solve the equations using a monolithic scheme. When solving all the equations of the problem simultaneously, the main questions arise in the assembly of the equations of the system and the inclusion the boundary conditions in the solution. The assembly and the treatment of the boundary conditions will be made in analogy with the implicit finite differences schemes and are discussed next.

5.2.1 Matrix assembly

Consider a problem with two state variables and one control variable, that is, $n = 2$ and $m = 1$, resulting in a total of 5 optimal control equations ($n_{eq} = 2n + m$). Evaluating the variables at $t_{k+\theta}$ and using (5.2) and (5.3), it is noticed that for one time step, the discretization depends only on the value of the variables in two time instants, t_k and t_{k+1}. The controls u_j, however, are calculated at the time instants $t_{k+\theta}$ and not at t_k and t_{k+1}, resulting in only one control per time step. For a discretization with only two time steps, $N = 2$, the assembly of the matrix $\mathbf{C}$ containing the coefficients

of each variable can be represented as follows,

$$\mathbf{C} = \begin{bmatrix}
c1_{x1}^{(0)} & c1_{x2}^{(0)} & c1_{\lambda 1}^{(0)} & c1_{\lambda 2}^{(0)} & c1_{u1}^{(0+\theta)} & c1_{x1}^{(1)} & c1_{x2}^{(1)} & c1_{\lambda 1}^{(1)} & c1_{\lambda 2}^{(1)} & 0 & 0 & 0 & 0 & 0 \\
c2_{x1}^{(0)} & c2_{x2}^{(0)} & c2_{\lambda 1}^{(0)} & c2_{\lambda 2}^{(0)} & c2_{u1}^{(0+\theta)} & c2_{x1}^{(1)} & c2_{x2}^{(1)} & c2_{\lambda 1}^{(1)} & c2_{\lambda 2}^{(1)} & 0 & 0 & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
c5_{x1}^{(0)} & c5_{x2}^{(0)} & c5_{\lambda 1}^{(0)} & c5_{\lambda 2}^{(0)} & c5_{u1}^{(0+\theta)} & c5_{x1}^{(1)} & c5_{x2}^{(1)} & c5_{\lambda 1}^{(1)} & c5_{\lambda 2}^{(1)} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & c1_{x1}^{(1)} & c1_{x2}^{(1)} & c1_{\lambda 1}^{(1)} & c1_{\lambda 2}^{(1)} & c1_{u1}^{(1+\theta)} & c1_{x1}^{(2)} & c1_{x2}^{(2)} & c1_{\lambda 1}^{(2)} & c1_{\lambda 2}^{(2)} \\
0 & 0 & 0 & 0 & 0 & c2_{x1}^{(1)} & c2_{x2}^{(1)} & c2_{\lambda 1}^{(1)} & c2_{\lambda 2}^{(1)} & c2_{u1}^{(1+\theta)} & c2_{x1}^{(2)} & c2_{x2}^{(2)} & c2_{\lambda 1}^{(2)} & c2_{\lambda 2}^{(2)} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & c5_{x1}^{(1)} & c5_{x2}^{(1)} & c5_{\lambda 1}^{(1)} & c5_{\lambda 2}^{(1)} & c5_{u1}^{(1+\theta)} & c5_{x1}^{(2)} & c5_{x2}^{(2)} & c5_{\lambda 1}^{(2)} & c5_{\lambda 2}^{(2)}
\end{bmatrix} \tag{5.4}$$

where $c1_{x1}^{(0)}$ is the coefficient of the state variable x_1 in the first equation at the time instant t_0, $c2_{x2}^{(1)}$ the coefficient of x_2 in the second equation at the time instant t_1 and so on. Notice that, for most problems, each equation does not depend on all variables, so it is expected many of the coefficients in the matrix $\mathbf{C}$ to be zero. Furthermore, due to the derivative approximation, the coefficient matrix also depends on the time step interval Δt. The blocks of coefficients in (5.4) separate the equations of each time step and it can be seen that the controls only appear at the center of the time step. Moreover, the coefficient matrix multiplies the vector of variables

$$\mathbf{X} = \begin{bmatrix} x_1^{(0)} & x_2^{(0)} & \lambda_1^{(0)} & \lambda_2^{(0)} & u_1^{(0+\theta)} & x_1^{(1)} & x_2^{(1)} & \lambda_1^{(1)} & \lambda_2^{(1)} & u_1^{(1+\theta)} & x_1^{(2)} & x_2^{(2)} & \lambda_1^{(2)} & \lambda_2^{(2)} \end{bmatrix}^T, \tag{5.5}$$

such that, $\mathbf{CX} = \mathbf{0}$. The matrix $\mathbf{C}$ has dimensions 10×14 and the vector $\mathbf{X}$ contains 14 variables. It is clear that the problem requires four boundary conditions, two initial and two final conditions. The nature of the conditions, however, depends on each problem. For example, if the final state is known, $x_1^{(2)}$ and $x_2^{(2)}$ are prescribed, and if the state is unknown at the end of the time interval, $\lambda_1^{(2)}$ and $\lambda_2^{(2)}$ are defined according to (2.39). For a general case, it can be shown that the coefficient matrix has $N \times n_{eq}$ lines and $[(N+1) \times n_{eq} - 1]$ columns. Which implies that $2n$ boundary conditions are required, that is, the states $\mathbf{x}^{(0)}$ are known at the origin and either $\mathbf{x}^{(N+1)}$ or $\boldsymbol{\lambda}^{(N+1)}$ are known at the final time instant.

5.2.2 Boundary conditions

As it is done in most finite differences schemes, the boundary condition results in the elimination of the respective known terms in the coefficient matrix $\mathbf{C}$ and in the vector of variables $\mathbf{X}$. The terms of the known variables are, then, sent to the right hand side of the equilibrium equation, that is, in the residual vector $\mathbf{R}$. However, the treatment of the boundary condition presented here differs from the standard boundary condition procedure seen in the one dimensional finite element method or in implicit finite difference schemes, mainly due to the non-squared shape of the coefficient matrix.

Consider that the initial and final state are known and the boundary conditions are given by $\mathbf{x_0} = [x_1^{(0)}, x_2^{(0)}, ..., x_{n_{eq}}^{(0)}]$ and $\mathbf{x_f} = [x_1^{(N+1)}, x_2^{(N+1)}, ..., x_{n_{eq}}^{(N+1)}]$, respectively. Then, the columns corresponding to the known variables can be suppressed and $\mathbf{C}$ will become a squared matrix. The coefficients of the columns suppressed, together with the values of the boundary conditions vectors, will be sent to the right hand side of the equation. If $\mathbf{C_{x0}}$ and $\mathbf{C_{xf}}$ are the columns of the known initial and final variables that were suppressed, the residual vector may be obtained as

$$\mathbf{R} = -\mathbf{C_{x0}} \cdot \mathbf{x_0} - \mathbf{C_{xf}} \cdot \mathbf{x_f}, \tag{5.6}$$

where $\mathbf{C_{x0}}$ and $\mathbf{C_{xf}}$ are matrices of dimensions $n \times N_L$, where $N_L = (n_{eq} \times numdt)$ is the number of lines in the $\mathbf{C}$ matrix. The inner product of those matrices with the boundary condition vectors $\mathbf{x_0}$ and $\mathbf{x_f}$, result in vectors with N_L variables. In the case with only two time steps, (5.4) is reduced to 10 columns and the optimal control system becomes

$$\begin{bmatrix} c1_{\lambda 1}^{(0)} & c1_{\lambda 2}^{(0)} & c1_{u1}^{(0+\theta)} & c1_{x1}^{(1)} & c1_{x2}^{(1)} & c1_{\lambda 1}^{(1)} & c1_{\lambda 2}^{(1)} & 0 & 0 & 0 \\ c2_{\lambda 1}^{(0)} & c2_{\lambda 2}^{(0)} & c2_{u1}^{(0+\theta)} & c2_{x1}^{(1)} & c2_{x2}^{(1)} & c2_{\lambda 1}^{(1)} & c2_{\lambda 2}^{(1)} & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c5_{\lambda 1}^{(0)} & c5_{\lambda 2}^{(0)} & c5_{u1}^{(0+\theta)} & c5_{x1}^{(1)} & c5_{x2}^{(1)} & c5_{\lambda 1}^{(1)} & c5_{\lambda 2}^{(1)} & 0 & 0 & 0 \\ 0 & 0 & 0 & :cc1_{x1}^{(1)} & c1_{x2}^{(1)} & c1_{\lambda 1}^{(1)} & c1_{\lambda 2}^{(1)} & c1_{u1}^{(1+\theta)} & c1_{\lambda 1}^{(2)} & c1_{\lambda 2}^{(2)} \\ 0 & 0 & 0 & c2_{x1}^{(1)} & c2_{x2}^{(1)} & c2_{\lambda 1}^{(1)} & c2_{\lambda 2}^{(1)} & c2_{u1}^{(1+\theta)} & c2_{\lambda 1}^{(2)} & c2_{\lambda 2}^{(2)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & c5_{x1}^{(1)} & c5_{x2}^{(1)} & c5_{\lambda 1}^{(1)} & c5_{\lambda 2}^{(1)} & c5_{u1}^{(1+\theta)} & c5_{\lambda 1}^{(2)} & c5_{\lambda 2}^{(2)} \end{bmatrix} \begin{bmatrix} \lambda_1^{(0)} \\ \lambda_2^{(0)} \\ u_1^{(0)} \\ x_1^{(1)} \\ x_2^{(1)} \\ \lambda_1^{(1)} \\ \lambda_2^{(1)} \\ u_1^{(1)} \\ \lambda_1^{(2)} \\ \lambda_2^{(2)} \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \\ r_6 \\ r_7 \\ r_8 \\ r_9 \\ r_{10} \end{bmatrix} \tag{5.7}$$

The variables $r_1, r_2, ..., r_{10}$ are the components of $\mathbf{R}$ defined in (5.6). If the vector $\mathbf{R}$

is non zero the system has a nontrivial solution, which gives the optimal values of the state, costate and control variables. In the next section some examples of linear problems that have analytical solutions are solved. The numerical solutions obtained with the approach presented here will be compared with the exact solution of the problems to validate the methodology used.

5.2.3 Verification of the linear monolithic approach

5.2.3.1 Example with fixed initial and final state

Consider the problem of solving the second order differential equation $\ddot{x} = u$. Using the state variable representation, the equation can be rewritten with two state variables and one control, given by the following system of equations,

$$\begin{cases} \dot{x}_1(t) = x_2(t), \\ \dot{x}_2(t) = u(t). \end{cases} \tag{5.8}$$

Moreover, the performance index of the problem is given by

$$J = \frac{1}{2} \int_{t_0}^{t_f} u^2(t)\ dt, \tag{5.9}$$

with the following fixed state boundary conditions,

$$\mathbf{x}(0) = [1 \quad 2]^T\ ; \quad \mathbf{x}(2) = [1 \quad 0]^T. \tag{5.10}$$

Hence, the Hamiltonian of the problem can be written as,

$$\mathcal{H} = \frac{1}{2}u^2 + \lambda_1 x_2 + \lambda_2 u. \tag{5.11}$$

The system has two state variables and one control, so it is expected five optimal control equations. The optimal control equations can be obtained by substituting the Hamiltonian in the state, costate and control equations presented in (5.1). Applying the time discretization (5.2) and (5.3) to the optimal control equations, the following discrete equations are obtained

$$
\begin{aligned}
0 &= \lambda_1^{k+1} - \lambda_1^k \\
0 &= \lambda_2^{k+1} - \lambda_2^k + [\theta\lambda_1^{k+1} + (1-\theta)\lambda_1^k]\Delta t \\
0 &= x_1^{k+1} - x_1^k - [\theta x_2^{k+1} + (1-\theta)x_2^k]\Delta t \\
0 &= x_2^{k+1} - x_2^k - u^{k+\theta}\Delta t \\
0 &= u^{k+\theta} + [\theta\lambda_2^{k+1} + (1-\theta)\lambda_2^k]
\end{aligned}
\tag{5.12}
$$

With the discrete equations above, it is possible to assemble the coefficient matrix and solve the problem by applying the boundary conditions as described previously. This example problem also has an exact analytical solution and its derivation is found in [Naidu, 2003]. The exact solution of the state and control variables are given by,

$$
\begin{aligned}
x_1^*(t) &= 0.5t^3 - 2t^2 + 2t + 1, \\
x_2^*(t) &= 1.5t^2 - 4t + 2, \\
u^*(t) &= 3t - 4.
\end{aligned}
$$

The exact solutions are shown in Fig.5.2, together with numerical results obtained by using the monolithic approach. It is seen that the numerical result matches well with the analytical solution of the problem. Due to the simplicity of this first example, the solution of a more elaborated problem is required to completely validate the monolithic approach for linear problems, as it is shown next.

5.2.3.2 Example with fixed initial and free final state

It will be considered the same second order differential equation solved in the previous section, $\ddot{x} = u$, with the state variable representation as written in (5.8). However, a more complex performance index is used, which involves the integral terms, as well as the terminal cost terms. The performance index of the problem is given by

$$
J = \frac{1}{2}\left[x_1(2) - 4\right]^2 + \frac{1}{2}\left[x_x(2) - 2\right]^2 + \frac{1}{2}\int_{t_0}^{t_f} u^2(t)\ dt, \tag{5.13}
$$

with the following boundary conditions,

$$
\mathbf{x}(0) = [1 \quad 2]^T\ ; \quad \mathbf{x}(2) = free. \tag{5.14}
$$

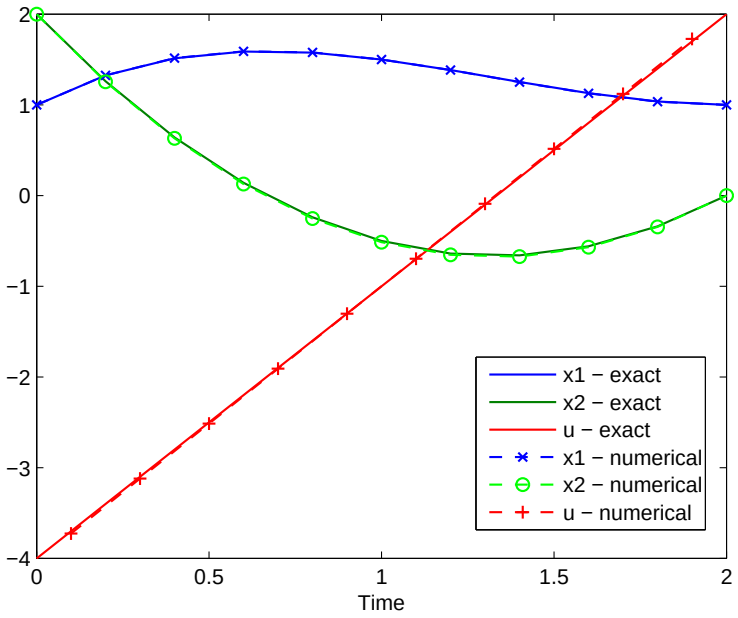


Figure 5.2: Numerical and analytical solution for the fixed initial and final state problem, using backward Euler ($\theta = 1$) and $N = 10$.

Although the performance index is different than the one in the previous example, the integral term is the same and, hence, the Hamiltonian of the problems is given by

$$\mathcal{H} = \frac{1}{2}u^2 + \lambda_1 x_2 + \lambda_2 u. \tag{5.15}$$

Since the Hamiltonian above is the same as in (5.11) the discrete equations of the optimal control system will also be the same equations shown in (5.23). The main difference of this example problem is at the final boundary condition. Looking at the general performance index in (2.8), it can be defined the function S of the performance index as,

$$S = \frac{1}{2}\left[x_1(2) - 4\right]^2 + \frac{1}{2}\left[x_x(2) - 2\right]^2. \tag{5.16}$$

Furthermore, since the final state is free, the costate variables are defined according to (2.39) and result in,

$$\lambda_1^*(2) = \frac{\partial S}{\partial x_1} = x_1(2) - 4,$$
$$\lambda_2^*(2) = \frac{\partial S}{\partial x_2} = x_2(2) - 2.$$

With the costate defined at the final boundary conditions, the system of equation can be assembled in the matrix form in a similar way. At the final time step, the costate and the respective columns of coefficients are suppressed and added to the residual vector. The analytical solution of the problem is also found in [Naidu, 2003] and are given as follows,

$$x_1^*(t) = \frac{1}{14}t^3 - \frac{2}{7}t^2 + 2t + 1,$$
$$x_2^*(t) = \frac{3}{14}t^2 - \frac{4}{7}t + 2,$$
$$u^*(t) = \frac{3}{7}t - \frac{4}{7}.$$

The comparison between the numerical results obtained and the analytical results is shown in Fig.5.3. Once again, the numerical results match with the exact solution, even for a small number of time steps. Therefore, the monolithic approach for linear problems is considered to be appropriate. In the following section it is introduced the monolithic approach to nonlinear problems, which will later be used in the micro-swimmer problem.

5.3 MONOLITHIC APPROACH TO NONLINEAR PROBLEMS

A different solution method will have to be presented for the solution of nonlinear problems. The approach used in section 5.2 would result in a nonlinear coefficient matrix $\mathbf{C}$ and it would not be straight forward to invert the matrix and obtain the solution of the vector of variables $\mathbf{X}$. Therefore, the Newton-Raphson method is used together with the monolithic approach to deal with the nonlinearity. The Newton-Raphson method requires the creation of the Jacobian matrix and a different construction of the residual vector. The description of the Newton-Raphson procedure is found in the appendices and the validation of the nonlinear monolithic approach is presented later in this section.

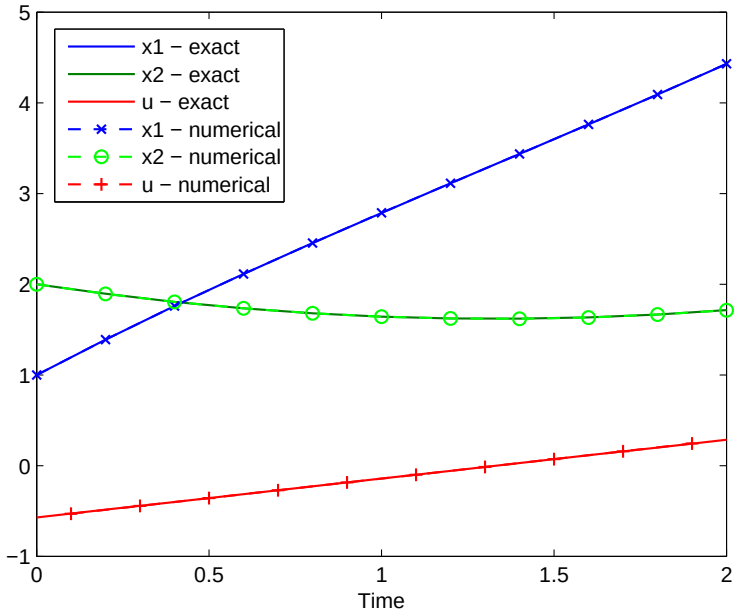


Figure 5.3: Numerical and analytical solution for the fixed initial and free final state problem, using trapezoidal rule ($\theta = 1/2$) and $N = 10$.

5.3.1 Newton-Raphson procedure and the monolithic approach

The Newton-Raphson method is the most famous iterative method for obtaining the root of a function or the solution of a system of nonlinear equations. Geometrically speaking, the method approximates the function $f(x)$ by a linear function tangent to f at the iteration point (x_i) and, through an iterative process, the method obtains a quadratic convergence towards the root of the function, or the solution in the case of a nonlinear system of equations, see appendix B.

In the solution of a system of equations, the Newton-Raphson method approximates the equations by the first-order Taylor's series expansion. The solution is obtained by solving the following equations at each iteration,

$$\begin{aligned} &\mathbf{J}(\mathbf{x}^k)\Delta\mathbf{x}^{k+1} = -\mathbf{R}, \\ &\mathbf{x}_{i+1} = \mathbf{x}_i + \Delta\mathbf{x}^{k+1}, \end{aligned} \tag{5.17}$$

where $\mathbf{R}$ is the residual vector and $\mathbf{J}$ is the Jacobian matrix. The residual vector in the nonlinear case can be given by the discrete optimal control equations, that is, combining (5.1) with (5.2) and (5.3). The assembly of the residual vector is done by

simply allocating the n_{eq} discrete optimal control equations in the residual vector at each time step, resulting in a vector $\mathbf{R}$ with N_L lines, where $N_L = n_{eq} \times N$, that is, number of optimal control equations equations times number of time steps.

The Jacobian $\mathbf{J}$ is the derivative of the discrete equations with respect to each one of the variables of the system. Being $\mathbf{X}$ the vector that contains all variables, as it is given in (5.5) for a problem with $n = 2$, $m = 1$ and $N = 2$, the Jacobian matrix may be represented by

$$\mathbf{J} = \frac{\partial \mathbf{R}}{\partial \mathbf{X}} = \begin{bmatrix} \frac{\partial R_1}{\partial X_1} & \frac{\partial R_1}{\partial X_2} & \cdots & \frac{\partial R_1}{\partial X_M} \\ \frac{\partial R_2}{\partial X_1} & \frac{\partial R_2}{\partial X_2} & \cdots & \frac{\partial R_2}{\partial X_M} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial R_N}{\partial X_1} & \frac{\partial R_N}{\partial X_2} & \cdots & \frac{\partial R_N}{\partial X_M} \end{bmatrix} \tag{5.18}$$

The number of discrete variables in the vector $\mathbf{X}$ is given by $N_C = [(N+1) \times n_{eq} - 1]$ and, therefore, the Jacobian matrix $\mathbf{J}$ has dimensions $N_L \times N_C$. To obtain the increment $\Delta \mathbf{x}^{k+1}$ in (5.17) the Jacobian needs to be invertible and, hence, a squared matrix. To transform the Jacobian in a squared matrix, it can be noticed that the number of columns of the matrix needs to be reduced by $2n$. This reduction derives from the boundary conditions.

5.3.2 Boundary conditions in the nonlinear approach

In the nonlinear monolithic methodology the inclusion of the boundary conditions is straightforward. The first step is to suppress in the Jacobian the columns corresponding to the known variables. Differently from the linear approach, it will not be necessary to send the deleted columns to the right-hand side of the system of equations. The known variable are already been taken into account in the residual and the only caution that needs to be taken is not to update the boundary values in (5.17) during the iterative process. The algorithm of the nonlinear monolithic approach is given in table 5.1.

5.3.3 Verification of the nonlinear monolithic approach

The validation of the nonlinear problem will be made by solving two example problems. The first is a linear problem similar to the ones solved in section 5.2.3, but with

Define the initial $(x_0, y_0, 0)$ and final conditions $(x_0, y_0, \Delta c)$

Define the number of time steps N

Start with an initial guess: $\mathbf{X} = \mathbf{X_0}$

While $|\mathbf{R}(\mathbf{X}^k)| > tol$ and $k < N$

  Loop over the time steps

    - Compute and assemble the Residual vector $\mathbf{R}(\mathbf{X}^k)$

    - Compute and assemble the Jacobian matrix $\mathbf{J}(\mathbf{X}^k)$

  End

  - Apply boundary conditions according to section 5.3.2

  - Obtain $\Delta\mathbf{X}^{k+1}$ from $\mathbf{J}(\mathbf{X}^k)\Delta\mathbf{X}^{k+1} = -\mathbf{R}(\mathbf{X}^k)$

  - Update the variables: $\mathbf{X}_{i+1} = \mathbf{X}_i + \Delta\mathbf{X}^{k+1}$

End

Table 5.1: Algorithm of the monolithic approach with the Newton-Raphson method.

different boundary condition. The linear problem is expected to have convergence in one time step and it will be solved to be compared with its exact analytical solution. The second problem is a nonlinear problem presented in [Primbs et al., 1999] and it will be compared with the results obtained by the authors.

5.3.3.1 Linear example problem with one fixed and one free final state

As the previous examples, the problem consists of solving the equation $\ddot{x} = u$ which can be written in a state variable representation as

$$\begin{cases} \dot{x}_1(t) = x_2(t), \\ \dot{x}_2(t) = u(t). \end{cases} \tag{5.19}$$

The performance index of the problem is also by

$$J = \frac{1}{2}\int_{t_0}^{t_f} u^2(t)\ dt, \tag{5.20}$$

but with boundary conditions defined as,

$$\mathbf{x}(0) = [1 \quad 2]^T\ ; \qquad x_1(2) = 0; \qquad x_2(2) = free. \tag{5.21}$$

Then, the Hamiltonian is also given by

$$\mathcal{H} = \frac{1}{2}u^2 + \lambda_1 x_2 + \lambda_2 u. \tag{5.22}$$

Hence, using the midpoint rule integration, it is possible to obtain the discrete optimal control equations. The assembly of the residual vector may be written as

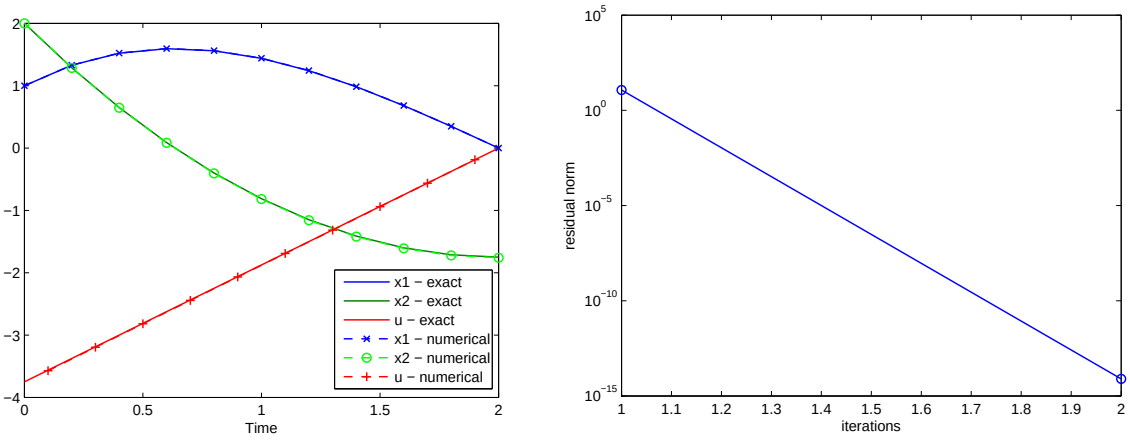
$$\begin{aligned}
&\mathbf{R}(k+1) = 0 = \lambda_1^{k+1} - \lambda_1^k, \\
&\mathbf{R}(k+2) = 0 = \lambda_2^{k+1} - \lambda_2^k + [\theta\lambda_1^{k+1} + (1-\theta)\lambda_1^k]\Delta t, \\
&\mathbf{R}(k+3) = 0 = x_1^{k+1} - x_1^k - [\theta x_2^{k+1} + (1-\theta)x_2^k]\Delta t, \\
&\mathbf{R}(k+4) = 0 = x_2^{k+1} - x_2^k - u^{k+\theta}\Delta t, \\
&\mathbf{R}(k+5) = 0 = u^{k+\theta} + [\theta\lambda_2^{k+1} + (1-\theta)\lambda_2^k],
\end{aligned} \tag{5.23}$$

with k varying from 0 to N, number of time steps. Moreover, the Jacobian can be obtained by taking the derivatives of the above discrete equations with respect to each of the variables. For one time step, the Jacobian $\mathbf{J}^{k+\theta}$ is given as

$$\mathbf{J}^{k+\theta} = \begin{bmatrix}
0 & 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & (1-\theta)\Delta t & -1 & 0 & 0 & 0 & \theta\Delta t & 1 \\
-1 & (1-\theta)\Delta t & 0 & -1 & 0 & 1 & \theta\Delta t & 0 & 0 \\
0 & -1 & 0 & 0 & -\Delta t & 0 & 1 & 0 & 0 \\
0 & (1-\theta) & 0 & 0 & 1 & 0 & \theta & 0 & 0
\end{bmatrix}. \tag{5.24}$$

After the appropriate assembly of the Jacobian matrix and according to the boundary condition of the problem, the columns corresponding to the derivatives with respect to $x_1^{(0)}$, $x_1^{(0)}$, $x_1^{(N+1)}$ and $\lambda_1^{(N+1)}$ should be suppressed, since those variables are known at the boundaries.

The analytical solution of the problem is presented in [Naidu, 2003] and it is shown in Fig.5.4a, together with the numerical solution obtained. It is seen that both solutions are in agreement. Moreover, the convergence of the norm of the residual vector is presented in Fig.5.4b and, as expected for a linear problem, it converges in one single time step.



(a) (b)

Figure 5.4: (a) Numerical and analytical solution and (b) residual convergence for the one fixed and one free final state problem, using trapezoidal rule ($\theta = 1/2$) and $N = 10$.

5.3.3.2 Nonlinear oscillator example problem

The two-dimensional nonlinear oscillator problem is presented in [Primbs et al., 1999] and it is written in a state variable representation as,

$$\begin{cases} \dot{x}_1 = x_2 \\ \\ \dot{x}_2 = -x_1\left(\dfrac{\pi}{2} + \arctan(5x_1)\right) - \dfrac{5x_1^2}{2(1+25x_1^2)} + 4x_2 + 3u \end{cases} \tag{5.25}$$

with the following performance index,

$$J = \int_0^\infty \left(x_2^2 + u^2\right)\, dt. \tag{5.26}$$

The Hamiltonian and derivatives of the oscillator problem were obtained by using the symbolic tool of *Matlab*. Then, using the nonlinear monolithic methodology discussed, the solution is presented in Fig.5.5, obtained for a time increment $\Delta t = 0.2$ and 50 time steps. The cost of taking the system from the initial to the final position is measured by the integral of the performance index (5.26) and the map of the variables x_1 and x_2 is shown in Fig.5.6a. The cost obtained with the numerical approach is presented in table 5.2. The error with respect to the cost obtained in [Primbs et al., 1999] decreases with the number of time steps and for 500 time steps, time increment $\Delta t = 0.02$ s, the

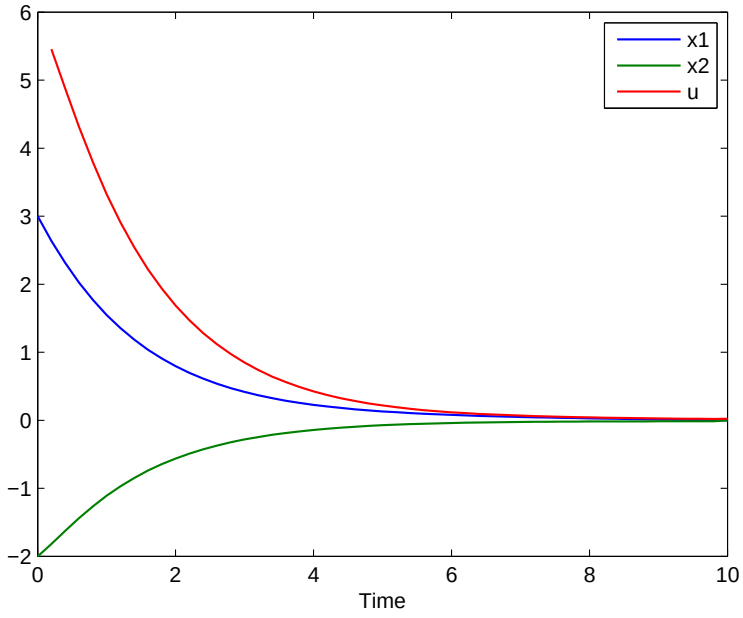
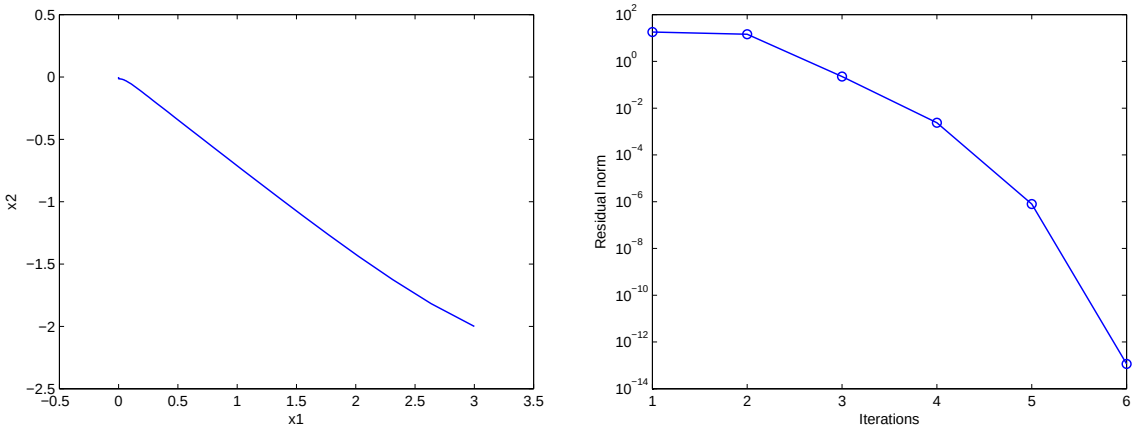


Figure 5.5: Numerical and analytical solution of the nonlinear oscillator problem, using backward Euler ($\theta = 1/2$) and $\Delta t = 0.2$ s.



(a) (b)

Figure 5.6: (a) Values of the states x_1 and x_2 and (b) residual convergence of the nonlinear oscillator problem, using backward Euler ($\theta = 1/2$) and $\Delta t = 0.2$ s.

relative error is 0.03%. Furthermore, The convergence of the norm of the residual is shown in Fig.5.6b and it has a quadratic behavior, which is expected when using the Newton-Raphson procedure.

The numerical results obtained are satisfactory and complete the verification of the monolithic approach to the optimal control problem. It is known that the monolithic

Table 5.2: Solution of the cost of the nonlinear oscillator problem.

Optimal Solution	Cost
Numerical result - $\Delta t = 0.4$s	33.1
Numerical result - $\Delta t = 0.2$s	32.8
Numerical result - $\Delta t = 0.1$s	32.4
Numerical result - $\Delta t = 0.02$s	31.8
Result - Primbs (1999)	**31.7**

approach presented is not efficient in terms of computation and might be computationally too expensive when dealing with large system of equations. However, it is adequate for one dimensional problems and it will be implemented in the micro-swimmer problem in the following chapter.

6 NUMERICAL RESULTS

In this chapter the numerical results obtained for the optimal control of the micro-swimmer problem are presented. Some examples were solved to show the relevance of the proposed problem, especially when comparing the efficiency of the optimal path obtained with some of the other strokes that have been proposed. Also, the importance of an appropriate initial guess in the solution and the convergence of different numerical integration methods are demonstrated. Finally, the comparison of the energy consumption of a displacement with one single stroke and several strokes is studied.

The results presented in this chapter are obtained by applying the methodology presented throughout this work. The micro-swimmer as an optimal control problem was formulated according to chapter 4, resulting in the eight equations presented in section 4.2.2. Those equations were discretized in time by applying the midpoint rule of numerical integration which, depending on the value of the parameter θ, can result in the forward Euler, backward Euler or trapezoidal rule methods.

6.1 SOLUTION OF THE MICRO-SWIMMER PROBLEM

In the numerical analysis, it was considered a three-sphere swimmer with radius $a = 0.05\ mm$ swimming in a water medium with viscosity $\mu = 10^3\ kg \cdot mm^{-1}s^{-1}$. The proposed problem is to make the swimmer obtain a displacement of $\Delta = 0.01\ mm$ at the end of one cycle, during a total time $t_f = 1\ s$ and while minimizing the energy waste of the stroke. The initial shape of the swimmer used in the analysis was defined as $x_0 = y_0 = 0.3\ mm$ and, hence, the final boundary condition of the stroke was also imposed to be $x_f = y_f = 0.3\ mm$, but with a net displacement $\Delta = 0.01\ mm$. Other parameters used in the analysis, such as, the initial guess and the number of time steps are described next.

6.1.1 Optimal stroke

To obtain the optimal stroke of the problem the NG-stroke, described in section 3.2.1, is used as the initial guess in the numerical analysis. The parameter ϵ was chosen to give a net displacement $\Delta = 0.01\ mm$ in the end of one cycle with $t_f = 1$s. Recalling

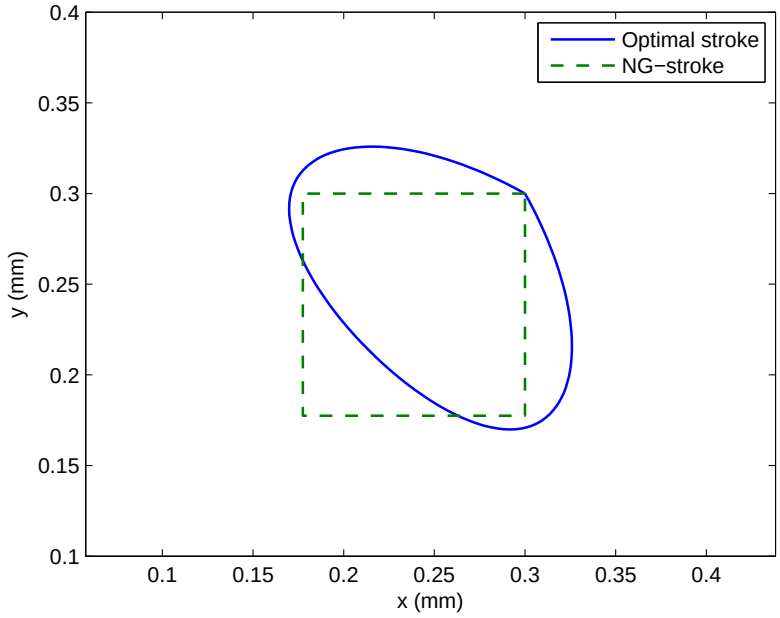


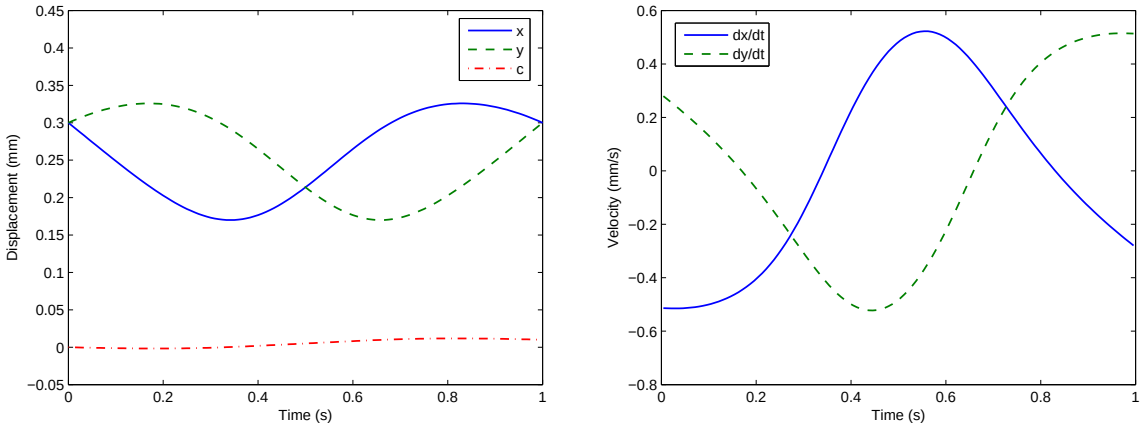
Figure 6.1: Optimal stroke for a net displacement $\Delta = 0.01\ mm$ and $t_f = 1\ s$, using the NG-stroke as initial guess.

the definitions given in section 3.2.1, the NG-stroke is defined by the parameter ϵ, which is the amount of relative displacement of the spheres in each stage of the cycle. To obtain the necessary displacement and time during the NG-stroke, it was found $\epsilon = 0.1225\ mm$. Taking the NG-stroke as the initial guess of the nonlinear monolithic approach, the result obtained for the optimal path is presented in Fig.6.1. The results were computed for $\theta = 1/2$, trapezoidal rule, and with a total number of integration points $N = 64$. The relative displacements and velocities of the spheres are shown in Fig.6.2a and Fig.6.2b, respectively. The convergence of the residual in the numerical solution of the optimal stroke is given in Fig.6.3 and shows the quadratic convergence obtained in the solution.

Table 6.1: Energy consumption of the optimal and NG strokes.

	NG-stroke (J)	Optimal stroke (J)	Energy saving
Numerical solution	0.195	0.16658	16.76 %
Alouges (2008)	0.278	0.229	21.40 %

The value of energy consumption obtained for both strokes are given in table 6.1.



(a) (b)

Figure 6.2: (a) Variation of x and y and c in one cycle; (b) Velocity of the relative motion of the spheres for optimal stroke.

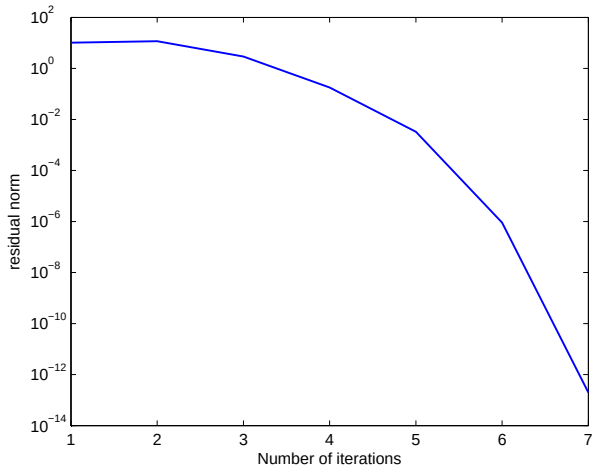


Figure 6.3: Convergence of the Newton-Raphson procedure for the optimal stroke solution.

The values of the energies obtained by Alouges in [Alouges et al., 2008] are also presented in the table. It is noticed the existence of a discrepancy between the energy values obtained numerically and the ones presented by Alouges. Not even the energy waste of NG-stroke obtained matches the one given in the reference. The value given by Alouges for the relative displacement of the NG-stroke that gives a net displacement of 1 mm is $\epsilon = 0.144\ mm$. However, according to the motion equations presented in chapter 3, a net displacement of $\Delta = 0.0159\ mm$ was found in this work for the same relative change of $\epsilon = 0.144\ mm$. This difference in the results can be explained in

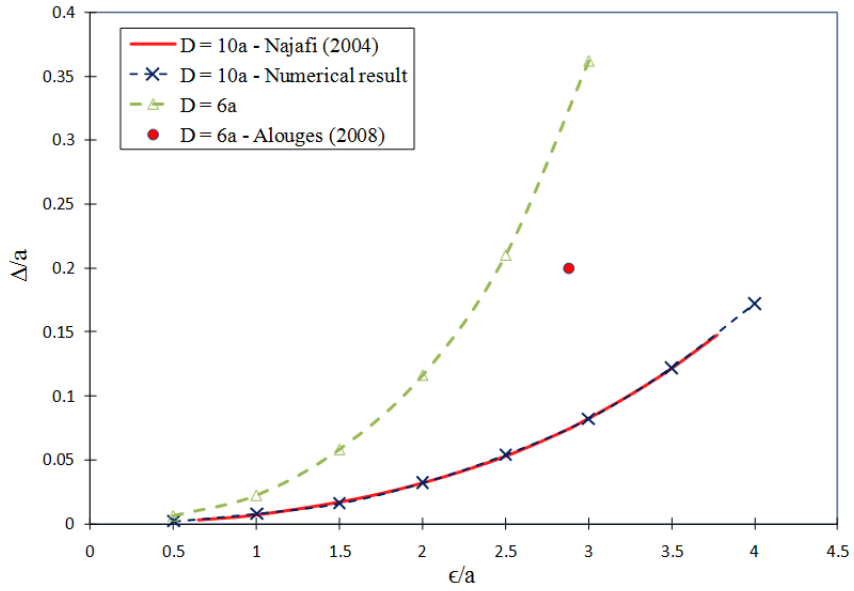


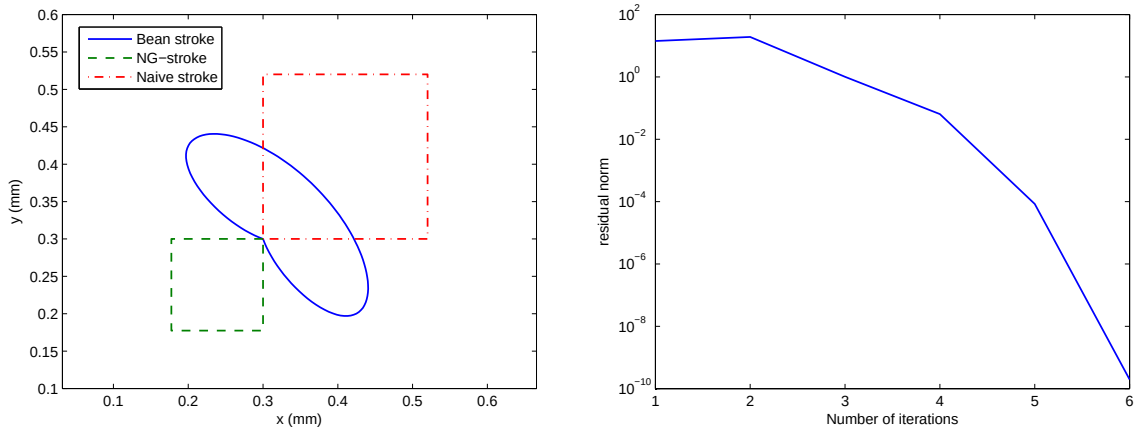
Figure 6.4: Dimensionless displacement for initial distance ratio of $D = 10a$ and $D = 6a$.

Fig.6.4. Following the same methodology used to validate the swimmer's motion equations with the motion given by [Najafi and Golestanian, 2004], the curve for an initial distance of $D = 6a$ was obtained. The point given by Alouges in [Alouges et al., 2008] is not situated along the curve. Hence, if the motion equations differs from one another, the results obtained in the energy consumption analysis are not comparable.

6.1.2 Local minima for various initial guesses

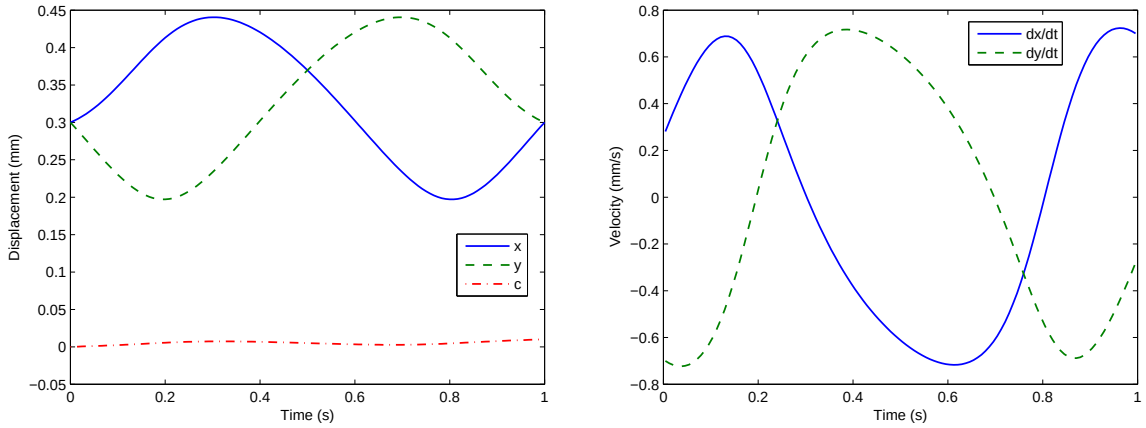
It has been observed during the numerical analysis that the solution of the micro-swimmer problem has other local minima. Depending on the initial guess, different stroke paths were obtained, but they do not correspond to the optimal stroke of the problem. A net displacement with positive variation of the relative distances between the spheres can be obtained, that is, with the expansion of the swimmer's arms. This stroke is called the Naive stroke and a net displacement of $\Delta = 0.01\ mm$ is obtained with a relative expansion of the arms of $\epsilon = -0.22\ mm$.

If the Naive stroke is used as an initial guess, the optimal solution obtained was named the *Bean* stroke due to its shape, Fig.6.5a. The solution was obtained by using the trapezoidal rule with $N = 64$ and the convergence of the residual is shown in Fig.6.5b. The displacements and velocities are shown in Fig.6.6a and Fig.6.6b.



(a) (b)

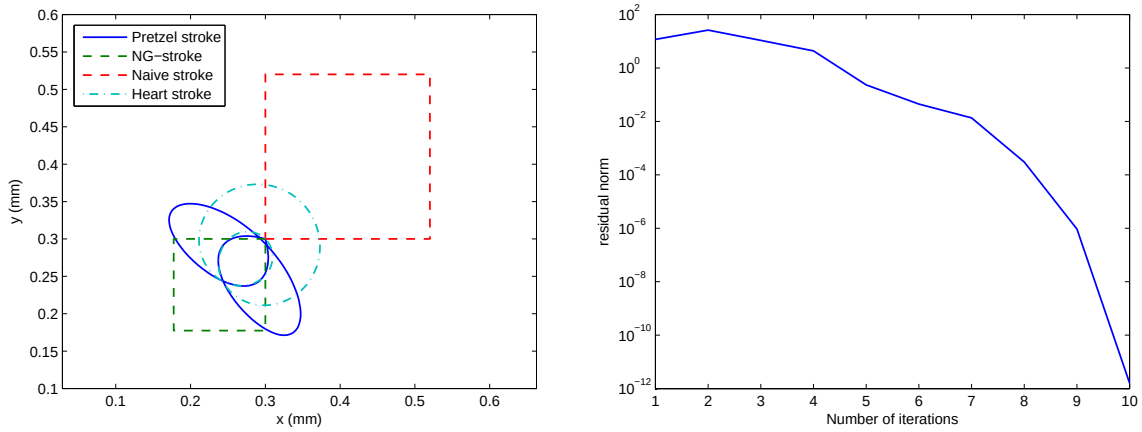
Figure 6.5: (a) Bean stroke for a net displacement $\Delta = 0.01\ mm$ and $t_f = 1\ s$, using the Naive stroke as initial guess; (b) Convergence of the Newton-Raphson procedure for the Bean stroke solution.



(a) (b)

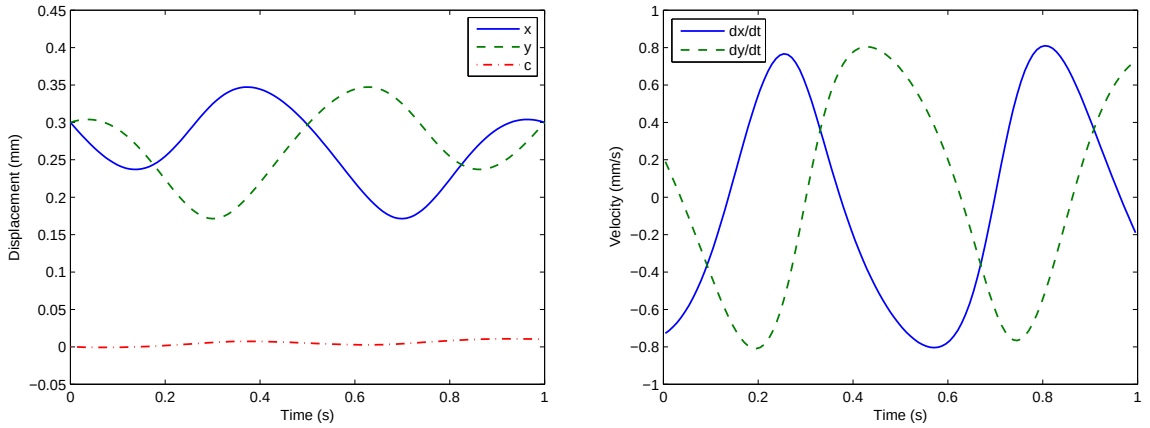
Figure 6.6: (a) Variation of x and y and c in one cycle; (b) Velocity of the relative motion of the spheres for Bean stroke.

Another stroke that can give a final net displacement is the "Heart" shape stroke. This stroke involves a more complex path in which one stroke has two relative minima. The Heart function is given by $r = \epsilon(x_0 + cos(\pi + 2t\pi))$, with the points being defined by $x = r\cos(\pi + 2t\pi)$ and $y = r\sin(\pi + 2t\pi)$. For obtaining the net displacement of 0.01 mm, it was found the value of $\epsilon = 0.1145$. If the Heart stroke is used as initial guess, and using the same parameters used in the Bean solution, the solution of the optimal problem converges to another local minimum called the Pretzel solution. The stroke



(a) (b)

Figure 6.7: (a) Pretzel stroke for a net displacement $\Delta = 0.01\ mm$ and $t_f = 1\ s$, using the Heart stroke as initial guess; (b) Convergence of the Newton-Raphson procedure for the Pretzel stroke solution, using $N = 120$.

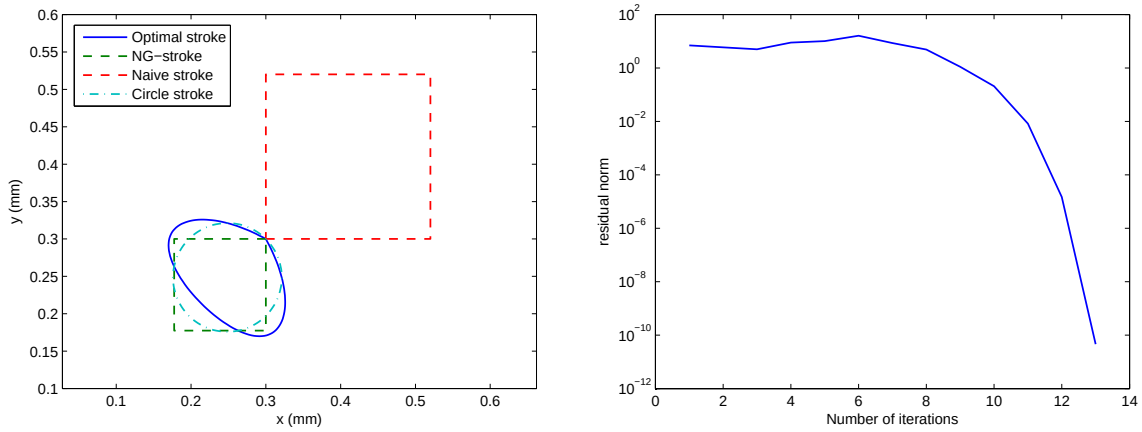


(a) (b)

Figure 6.8: (a) Variation of x and y and c in one cycle; (b) Velocity of the relative motion of the spheres for pretzel stroke.

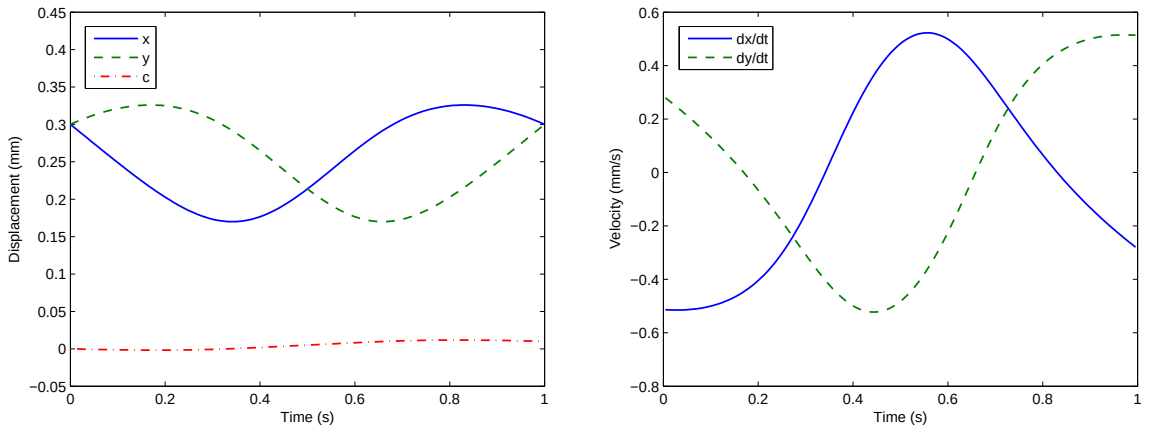
is presented in Fig.6.7a and the convergence of the residual is presented in Fig.6.5b. The displacements and velocities of the relative distance of the spheres for the Pretzel solution are shown in Fig.6.8a and Fig.6.8b.

Using a circular stroke, given by sinusoidal functions, with a radius of 0.0727 mm and a center on the point $(0.3, 0.3)$ mm, the swimmer also obtains a net displacement of $\Delta = 0.01\ mm$. If the circle stroke is used as initial guess, the solution converges to



(a) (b)

Figure 6.9: (a) Circle stroke for a net displacement $\Delta = 0.01\ mm$ and $t_f = 1\ s$ and using the Circle stroke as initial guess; (b) Convergence of the Newton-Raphson procedure for the optimal stroke solution, using $N = 120$.



(a) (b)

Figure 6.10: (a) Variation of x and y and c in one cycle; (b) Velocity of the relative motion of the spheres for circle stroke.

the optimal stroke, Fig.6.9a, but with a slow convergence, Fig.6.9b. The difficulty of the solution to converge to the optimal path is due to the small difference in the energy consumption of both strokes. Hence, since the solution starts from a point with low energy waste, it is harder for the numerical procedure to find the global minimum of the problem. The energy waste of the strokes used as initial guess are shown in table 6.2 and the energy of the local minimum strokes are shown in table 6.3. The shape changes x and y and their velocities are shown in Fig.6.10a and Fig.6.10b.

Table 6.2: Energy consumption (*Joule*) of the initial guess strokes with $\Delta = 0.01$ mm.

	NG-stroke	Naive stroke	Heart stroke	Circle stroke
Numerical solution	0.1949	0.5571	0.5571	0.1672

Table 6.3: Energy consumption (*Joule*) of the optimal strokes with $\Delta = 0.01$ mm.

	Optimal stroke	Bean stroke	Pretzel stroke
Numerical solution	0.1665	0.3060	0.3792

The results shown in the tables above demonstrate that the optimal stroke found is indeed the stroke with the smallest energy waste. Although the value of the optimal stroke energy is still smaller, the difference between the expended energy in the circle stroke and the optimal stroke is very small. This small difference explains why the convergence shown in Fig.6.9b is slow. Since the initial guess is already very close to the optimal solution, the search for the optimal path becomes more difficult. All the results presented in this section were made with the use of the trapezoidal rule. Next, the influence of the numerical integration method on the results will be shown.

6.2 OTHER NUMERICAL ANALYSES

6.2.1 Analysis of the numerical integration method

Using the NG-stroke as the initial guess and, therefore, having the optimal stroke as the solution of the problem, three numerical integration methods will be analyzed. For $\theta = 0$ the problem is solved with the forward Euler method, for $\theta = 1/2$ with trapezoidal rule and backward Euler for $\theta = 1$. The approach used is to start with a low number of integration points and check if the methods converge to one single value when increased the number of points.

The solution for $N = 60$ is presented in Fig.6.11. It is observed that the solution with forward Euler is distorted to the right, while the solution with backward Euler is distorted to the left. In other words, due to the characteristics of the discretization, the forward Euler scheme prioritizes the initial condition over the final condition, whereas the backward Euler gives more importance to the final conditions of the system. Increasing the number of time steps, Fig.6.12 shows the solution for $10\times$ more integration

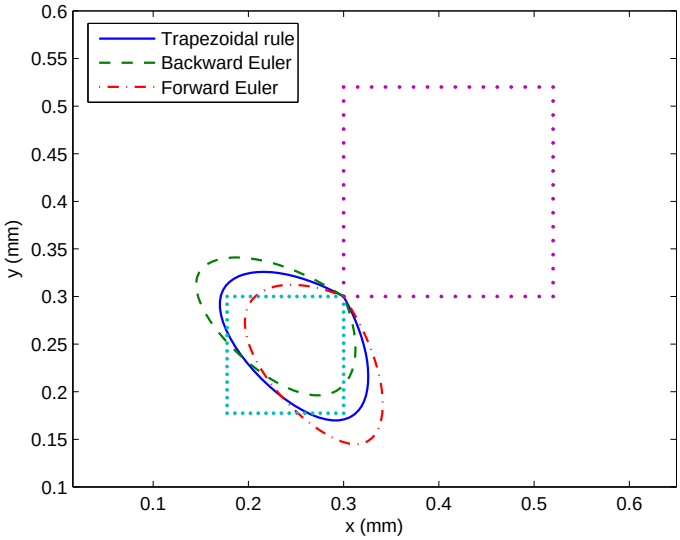


Figure 6.11: Different numerical integration methods with $N = 60$.

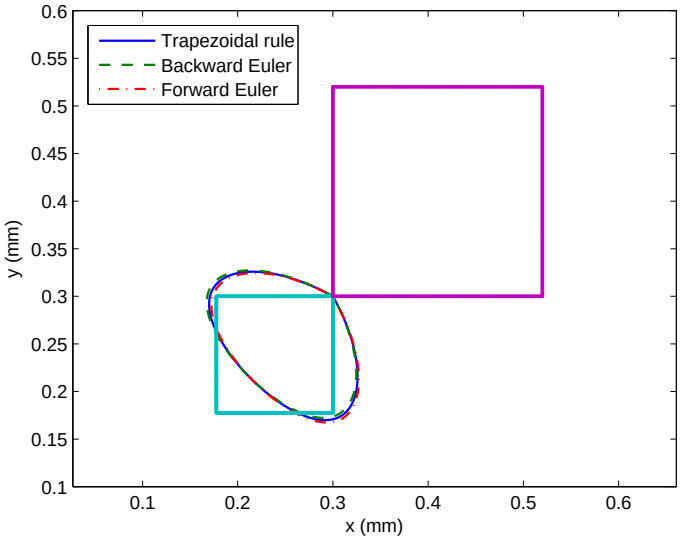


Figure 6.12: Different numerical integration methods with $N = 600$.

points. It is seen that both backward Euler and forward Euler schemes converge to the solution of the trapezoidal rule. The trapezoidal rule solution, on the other hand, is stable for both types of discretization. The reason of the stability of the trapezoidal rule is the balanced nature of the scheme, that uses the previous and next time instants in the computation of the variables in each time step. Hence, the scheme gives equal importance to the initial and final conditions and it seems to be appropriate for the solution of boundary value problems.

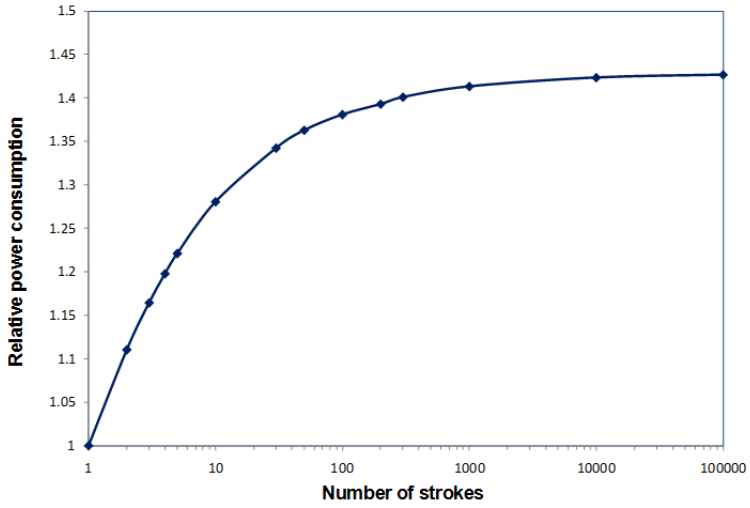


Figure 6.13: Relative energy consumption per number of strokes required to reach $\Delta = 0.01\ mm$.

6.2.2 Multiple strokes vs. one single stroke

One may be interested in checking whether the minimum energy of the swimming mechanism to reach a certain point in space comes from a single stroke with large deformations or from multiple strokes of small relative displacements. To answer this question it was calculated the energy waste by the swimmer to move an amount of $\Delta = 0.01\ mm$ in $t_f = 1\ s$ with one single step, as presented previously. Furthermore, it was obtained the energy waste of the swimmer to move an amount of Δ/n_s, where n_s is the number of strokes of the swimmer. The total energy consumption is then assumed to be the energy obtained times the number of strokes.

The results of the energy waste versus the number of strokes are shown in Fig.6.13. It can be seen that the energy consumption increases with the number of strokes needed to reach the destination point. However, this growth in the energy expended seems to reach an asymptotic value for a large n_s. It can be concluded that it is desirable to try to reach the destination with the minimum number of strokes. Naturally, limitations in the geometry of the micro-swimmer may restrict the choice of number of strokes, but whenever it is possible, the motion should be done with one single stroke to reduce the energy waste.

7 CONCLUSIONS

7.1 SUMMARY AND CONCLUSIONS

Throughout this work, a numerical procedure for solving the optimal control problem of a micro-swimmer was presented. The theory of the optimal control was described in chapter 2. There, the two main approaches for solving optimal control problems were detailed, the Dynamic Programming approach and the Variational approach. The later approach was used in the micro-swimmer problem. Furthermore, the kinematics and governing equations of the three-spheres micro-swimmer problem were presented in chapter 3. To verify the governing equations, the simple four-stage stroke was solved numerically and compared with the results in [Najafi and Golestanian, 2004]. The evolution of the final displacement of the swimmer in one cycle, with respect to the amount of relative displacement imposed, matched perfectly the results presented by Najafi and Golestanian.

The first step in the optimal solution was to represent the three-spheres micro-swimmer as an optimal control problem. In chapter 4, the theories and equations presented in chapters 2 and 3 were combined. The state representation of the problem resulted in having the relative distances x and y and the center of mass c as the state variables of the system. The control inputs were assumed to be the velocities of the relative movement between the spheres, $\dot{x}$ and $\dot{y}$. The energy waste was selected as the performance index, that is, the function to be minimized during the process. The energy waste was also formulated by using the optimal control formalism and it resulted in a *Lagrange* type performance index dependent on the control inputs. The problem was solved by the Hamiltonian approach and resulted in three state equations, three costate equations and two control equations, resulting in the eight necessary optimal control equations.

The micro-swimmer optimal control problem formulated in this work resulted in a nonlinear two-point boundary value problem. To solve the problem a monolithic strategy was suggested. First, the time discretization of the problem was done by applying the generalized midpoint rule. Next, the monolithic procedure for linear problems was

introduced. The treatment of the boundary conditions and the matrix assembly were described in section 5.2. Linear problems with known exact solutions were solved to verify the linear monolithic strategy. The numerical solutions with trapezoidal rule matched well with the corresponding exact solutions even for a small number of time steps and the strategy was considered satisfactory. Then, the Newton-Raphson procedure was implemented to transform the monolithic approach in a nonlinear strategy. The algorithm used in the numerical procedure was described in section 5.3 and example problems were solved to verify the strategy. The solution of the linear example problem displayed excellent numerical results, with the residual converging in one single iteration. A nonlinear problem was also solved. The results presented in table 5.2 showed that the numerical solution converges to the solution presented in [Primbs, 1999] with the increase of the number of time steps. Hence, the verification of both linear and nonlinear monolithic strategies was considered satisfactory and the nonlinear strategy was used in the three-spheres micro-swimmer problem.

Applying the nonlinear monolithic procedure to the micro-swimmer optimal control problem formulated in chapter 4, an optimal stroke was obtained. An initial guess was used by considering the NG-stroke with same net displacement. Comparing the values of the energy saved in the optimal stroke with the ones shown in [Alouges et al., 2008], a discrepancy in the results was found. It was shown in Fig.6.4 that the motion analysis of the referenced literature did not match the one used in this work and, hence, the results are not comparable. However, the motion equations used in this work were verified with the introductory work of the three-spheres swimmer, [Najafi and Golestanian, 2004]. Other local minima were also found for the problem. Using several different strokes that produce similar net displacement at each cycle, but with higher energy cost, different optimal solutions were found. The energy waste for each of these strokes were presented in section 6.1.2 and the optimal stroke showed an energy reduction of 16.76% with respect to the NG-stroke. The analysis of the numerical integration methods have shown that the forward and backward Euler schemes present difficulties in the accuracy of the solution for small number of time steps. However, the solution with those schemes converged to the optimal solution for high discretization of the time period. The trapezoidal rule, on the other hand, has shown satisfactory results even for small number of time steps, which can be explained by the balanced nature

of the scheme, that takes into account the initial and final conditions equally. Finally, the analysis of a net displacement done by one single stroke and n_s small strokes was presented in section 6.2.2. It was shown that the higher the number of strokes needed for the swimmer to reach a certain point, the higher the power consumption. Hence, when possible, the swimmer should reach its destination with one single stroke to reduce the energy waste.

7.2 SCOPE FOR FUTURE WORKS

Although the numerical strategy presented in this work has provided good results, it is not ideal for solving complex systems. When dealing with higher-dimensional problems, the number of variables in the system will increase significantly. Furthermore, in most FSI systems the time interval might be undefined and other techniques might be needed to be applied, such as, the Receding Horizon approach, see [Primbs, 1999]. Hence, it is suggested that future works focus on the numerical strategy to develop a non monolithic approach. One possible idea is to develop a iterative process that computes two or three time steps at the time and makes a loop over the time domain until convergence is achieved.

APPENDICES

A VARIATIONAL APPROACH - CALCULATIONS

Derivatives of $\mathbf{V_x}$ and $\mathbf{V_y}$

The functions V_x and V_y are given as:

$$V_x(x,y) = \frac{a\left(-2xy\left(x+y\right)\left(2x^2-y^2\right)+a\left(6x^4+6x^3y-3x^2y^2-3xy^3-3y^4\right)\right)}{-12x^2y^2\left(x+y\right)^2+12axy\left(x+y\right)\left(x^2+3xy+y^2\right)+9a^2\left(x^4-2x^3y-5x^2y^2-2xy^3+y^4\right)} \tag{A.1}$$

and

$$V_y(x,y) = \frac{a\left(-2xy\left(x+y\right)\left(x^2-2y^2\right)+3a\left(x^4+x^3y+x^2y^2-2xy^3-2y^4\right)\right)}{-12x^2y^2\left(x+y\right)^2+12axy\left(x+y\right)\left(x^2+3xy+y^2\right)+9a^2\left(x^4-2x^3y-5x^2y^2-2xy^3+y^4\right)}$$

Splitting the functions into their numerator and denominator, the following functions are defined:

$$\begin{aligned} f_x = \;& a[-2xy(x+y)(2x^2-y^2)+a(6x^4+6x^3y-3x^2y^2-3xy^3-3y^4)] \\ f_y = \;& a[-2xy(x+y)(x^2-2y^2)+3a(x^4+x^3y+x^2y^2-2xy^3-2y^4)] \\ g = \;& -12x^2y^2(x+y)^2+12axy(x+y)(x^2+3xy+y^2) \\ & +9a^2(x^4-2x^3y-5x^2y^2-2xy^3+y^4) \end{aligned}$$

such that,

$$\begin{aligned} V_x(x,y) &= f_x/g, \\ V_y(x,y) &= f_y/g. \end{aligned}$$

and

$$\begin{aligned} \frac{\partial V_x}{\partial x} &= \frac{\partial f_x}{\partial x}\frac{1}{g} - \frac{f_x}{g^2}\frac{\partial g}{\partial x}, \\ \frac{\partial V_x}{\partial y} &= \frac{\partial f_x}{\partial y}\frac{1}{g} - \frac{f_x}{g^2}\frac{\partial g}{\partial y}, \\ \frac{\partial V_y}{\partial x} &= \frac{\partial f_y}{\partial x}\frac{1}{g} - \frac{f_y}{g^2}\frac{\partial g}{\partial x}, \\ \frac{\partial V_y}{\partial y} &= \frac{\partial f_y}{\partial y}\frac{1}{g} - \frac{f_y}{g^2}\frac{\partial g}{\partial y}. \end{aligned}$$

The derivatives of f_x and f_y are given as follows,

$$\frac{\partial f_x}{\partial x} = a[-2y(x+y)(2x^2-y^2) - 2xy(2x^2-y^2) - 8x^2y(x+y) + a(24x^3+18x^2y-6xy^2-3y^3)],$$

$$\frac{\partial f_x}{\partial y} = a[-2x(x+y)(2x^2-y^2) - 2xy(2x^2-y^2) + 4xy^2(x+y) + a(6x^3-6x^2y-9xy^2-12y^3)]$$

$$\frac{\partial f_y}{\partial x} = a[-2y(x+y)(x^2-2y^2) - 2xy(x^2-2y^2) - 4x^2y(x+y) + 3a(4x^3+3x^2y+2xy^2-2y^3)],$$

$$\frac{\partial f_y}{\partial y} = a[-2x(x+y)(x^2-2y^2) - 2xy(x^2-2y^2) + 8xy^2(x+y) + 3a(x^3+2x^2y-6xy^2-8y^3)],$$

and the derivatives of g are given as,

$$\begin{aligned}\frac{\partial g}{\partial x} = \; & -24xy^2(x+y)^2 - 24x^2y^2(x+y) + 12ay(x+y)(x^2+3xy+y^2) \\ & +12axy(x^2+3xy+y^2) + 12axy(x+y)(2x+3y) + 9a^2(4x^3-6x^2y-10xy^2-2y^3),\end{aligned}$$

$$\begin{aligned}\frac{\partial g}{\partial y} = \; & -24x^2y(x+y)^2 - 24x^2y^2(x+y) + 12ax(x+y)(x^2+3xy+y^2) \\ & +12axy(x^2+3xy+y^2) + 12axy(x+y)(3x+2y) + 9a^2(-2x^3-10x^2y-6xy^2+4y^3).\end{aligned}$$

Derivatives of the performance index

In the computation of the optimal control equations of the micro-swimmer problem, the derivatives of the terms inside the integral of the performance index needs to be derived. If the performance index of 4.9 is rewritten as

$$J = \int_{t_0}^{t_f} j(x, y, u_1, u_2)\, dt. \tag{A.2}$$

with $j(x, y, u_1, u_2)$ being defined as

$$j(x, y, u_1, u_2) = \frac{1}{9\pi\mu}\left\{u_1^2 j_1(x,y) + u_2^2 j_2(x,y) + u_1 u_2 j_3(x,y)\right\}. \tag{A.3}$$

Then the terms j_1, j_2 and j_3 are given by:

$$\begin{aligned} j_1(x,y) &= (3V_x - 2)r_{x1} + (3V_x + 1)r_{x2} + (3V_x + 1)r_{x3} \\ j_2(x,y) &= (3V_y - 1)r_{y1} + (3V_y + 2)r_{y2} + (3V_y - 1)r_{y3} \\ j_3(x,y) &= [(3V_y - 1)r_{x1} + (3V_y + 2)r_{x2} + (3V_y - 1)r_{x3}] \\ &\quad + [(3V_x - 2)r_{y1} + (3V_x + 1)r_{y2} + (3V_x + 1)r_{y3}]. \end{aligned} \tag{A.4}$$

Moreover, the functions r_{xi} and r_{yi} with $i = 1, 2, 3.$ are written as

$$r_{x1} = \left(\frac{3V_x - 2}{6a} + \frac{3V_x + 1}{4x} + \frac{3V_x + 1}{4(x+y)}\right) \qquad r_{y1} = \left(\frac{3V_y - 1}{6a} + \frac{3V_y - 1}{4x} + \frac{3V_y + 2}{4(x+y)}\right)$$
$$r_{x2} = \left(\frac{3V_x - 2}{4(x+y)} + \frac{3V_x + 1}{4y} + \frac{3V_x + 1}{6a}\right) \qquad r_{y2} = \left(\frac{3V_y - 1}{4(x+y)} + \frac{3V_y - 1}{4y} + \frac{3V_y + 2}{6a}\right)$$
$$r_{x3} = \left(\frac{3V_x - 2}{4x} + \frac{3V_x + 1}{6a} + \frac{3V_x + 1}{4y}\right) \qquad r_{y3} = \left(\frac{3V_y - 1}{4x} + \frac{3V_y - 1}{6a} + \frac{3V_y + 2}{4y}\right)$$

The derivatives of the terms j_1, j_2 and j_3 with respect to x are presented as follows,

$$\frac{\partial j_1}{\partial x} = \left(3\frac{\partial V_x}{\partial x}r_{x1} + (3V_x - 2)\frac{\partial r_{x1}}{\partial x}\right) + \left(3\frac{\partial V_x}{\partial x}r_{x2} + (3V_x + 1)\frac{\partial r_{x2}}{\partial x}\right) + \left(3\frac{\partial V_x}{\partial x}r_{x3} + (3V_x + 1)\frac{\partial r_{x3}}{\partial x}\right),$$
$$\frac{\partial j_2}{\partial x} = \left(3\frac{\partial V_y}{\partial x}r_{y1} + (3V_y - 2)\frac{\partial r_{y1}}{\partial x}\right) + \left(3\frac{\partial V_y}{\partial x}r_{y2} + (3V_y + 1)\frac{\partial r_{y2}}{\partial x}\right) + \left(3\frac{\partial V_y}{\partial x}r_{y3} + (3V_y + 1)\frac{\partial r_{y3}}{\partial x}\right),$$
$$\begin{aligned}\frac{\partial j_3}{\partial x} = {} & \left(3\frac{\partial V_y}{\partial x}r_{x1} + (3V_y - 2)\frac{\partial r_{x1}}{\partial x}\right) + \left(3\frac{\partial V_y}{\partial x}r_{x2} + (3V_y + 1)\frac{\partial r_{x2}}{\partial x}\right) + \left(3\frac{\partial V_y}{\partial x}r_{x3} + (3V_y + 1)\frac{\partial r_{x3}}{\partial x}\right) \\ & + \left(3\frac{\partial V_x}{\partial x}r_{y1} + (3V_x - 2)\frac{\partial r_{y1}}{\partial x}\right) + \left(3\frac{\partial V_x}{\partial x}r_{y2} + (3V_x + 1)\frac{\partial r_{y2}}{\partial x}\right) + \left(3\frac{\partial V_x}{\partial x}r_{y3} + (3V_x + 1)\frac{\partial r_{y3}}{\partial x}\right).\end{aligned}$$

The derivatives with respect to y are done in a similar way. Furthermore, in the computation of the Jacobian Matrix in chapter 6 it is necessary to obtain the discrete second derivatives of the optimal control equations. Using chain rule and derivating the above equations term by term the second derivatives were obtaind, which will not be presented due to the length of the equations.

B THE NEWTON-RAPHSON METHOD

Solution of a single nonlinear equation

To find the roots of a nonlinear equation, one of the most popular methods is the Newton-Raphson method. It is one of the most powerful procedures in all numerical analysis and it always converges quadratically if the initial approximations is sufficiently close to the root [Hoffman, 2001]. Its only disadvantage is the evaluation of the derivative $f'(x)$ of the function. Given a nonlinear function $f(x)$, the Newton-Raphson method approximates the function by a linear function $g(x)$ that is tangent to $f(x)$ and find the solution for $g(x) = 0$. Approximating the derivative by the first order terms of the Taylor series,

$$f(x_{i+1}) = f(x_i) + f'(x_i)(x_{i+1} - x_i) + \mathcal{O}(x^2), \tag{B.1}$$

where $\mathcal{O}(x^2)$ is the error of the higher order terms. Ignoring the error and considering the root of the function to be $f(x_{i+1}) = 0$, the solution of x_{i+1} is presented as

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}. \tag{B.2}$$

The iterative process of (B.2) gives a quadratic convergence of the solution and it is graphically represented in Fig.B.1.

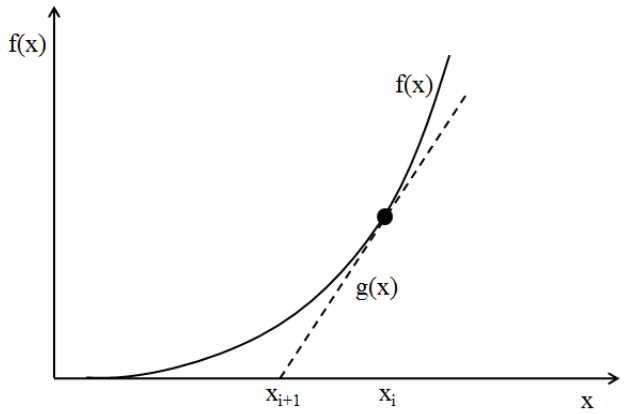


Figure B.1: Newton-Raphson method.

Solution of a system of nonlinear equations

If one is interested of solving a system of nonlinear equations such that $\mathbf{F}(\mathbf{x}) = \mathbf{0}$ and $\mathbf{F}(\mathbf{x}) = \mathbf{A}\mathbf{x} - \mathbf{b}$, the Newton-Raphson iteration is written as,

$$\Delta\mathbf{x}^{i+1} = -[\mathbf{F}(\mathbf{x}^i)]^{-1}\mathbf{F}(\mathbf{x}^i) \tag{B.3}$$

$$\mathbf{x}^{i+1} = \mathbf{x}^i + \Delta\mathbf{x}^{i+1} \tag{B.4}$$

where $[\mathbf{F}(\mathbf{x}^i)]^{-1}$ can also be viewed as the Jacobian matrix, matrix that contains the derivatives of the equations with respect to each of the variables. The convergence of the solution of a nonlinear system of equation can also be proven to be quadratic. The convergence analysis of the Newton-Raphson method and more details of the numerical procedure may be found in [Hoffman, 2001] and [Golub and Ortega, 1992].

Bibliography

[Alexander and Yeomans, 2008] Alexander, G., P., Yeomans. J., M.,"Hydrodynamic Interactions at Low Reynolds Number.", Experimental Mechanics, 2010.

[Alouges et al., 2008] Alouges, F., DeSimone. A., Lefebvre A.,"Optimal Stroke for Low Reynolds Number Swimmers: An Example.", J. Nonlinear Sci vol. 18, pp. 277-302, 2008.

[Alouges et al., 2010] Alouges, F., DeSimone. A., Heltai L.,"Numerical strategies for stroke optimization of axisymmetric microswimmers.", SISSA, 2010.

[Batchelor, 1975] Batchelor, G. K., "Brownian diffusion of particles with hydrodynamic interaction.", J. Fluid Mech., vol. 74, part 1, pp. 1-29, 1976.

[Bellman, 1952] Bellman, R. E., "The theory of dynamic programming.", In Proc. Nat. Acad. Sci., USA, number 38, 1952.

[Dyer and McReynolds, 1970] Dyer, P., McReynolds, S., "The computation and theory of optimal control.", Academic Press, New York, USA, 1970.

[Golub and Ortega, 1992] Hoeffman, J., D.,"Scientific Computing adn Differential Equations: An Introduction to Numerical Methods, 2^{nd} Ed., Academic Press, USA, 1992.

[Hoffman, 2001] Hoeffman, J., D.,,"Numerical Methods for Engineers and Scientists", 2^{nd} Ed., Marcel Decker, USA, 2001.

[Kirk, 1998] Kirk, D. E.,"Optimal Control Theory: An Introduction", 2^{nd} Ed., Dover Publications, USA, 1998.

[Knowels, 1981] Knowels, G., "An introduction to applied optimal control theory.", Academic Press, New York, USA, 1981.

[Najafi and Golestanian, 2004] Najafi, A., Golestanian. R., "Simple swimmer at low Reynolds number: Three linked spheres.", Physical Review E, vol. 69, 062901, 2004.

[Najafi and Golestanian, 2005] Najafi, A., Golestanian. R., "Propulsion at low Reynolds number.", J. Phys. Condens. Matter, vol. 17, pp. 1203-1208, 2005.

[Naidu, 2003] Naidu, D. S.,"Optimal Control Systems", CRC Press, Idaho State University, USA, 2003.

[Pao, 1967] Pao, R. H. F.,"Fluid Dynamics", Merrill Books, USA, 1967.

[Purcel, 1977] Purcel E. M., "Life at low Reynolds numbers", Am. J. Phys., vol. 45, pp. 3-11, 1977.

[Pontryagin et al., 1956] Boltyanskii, V. G., Gamkrelidze, R. V., Pontryagin, L. S., "On the theory of optimal processes", Dokl. Akad. Nauk SSSR, vol. 110, pp. 7-10, 1956. (in Russian).

[Primbs, 1999] Primbs, J. A., "Nonlinear Optimal Control: A Receding Horizon Approach", California Institute of technology, Pasadena, California, USA, 1999.

[Primbs et al., 1999] Primbs, J. A., Nevistić, V., Doyle, J., C., "Nonlinear Optimal Control: A Control Lyapunov Function and Receding Horizon Perspective", Asian Journal of Control, Vol. 1 No. 1, pp. 14-24, March 1999.

[Putz and Yeomans, 1999] Putz, V. B., Yeomans, J., M., "Hydrodynamic Synchronisation of Model Microswimmers", J. Sat. Phys, vol. 137, pp. 1001-1013, 2009.

[Rotne and Prager, 1969] Rotne, J., Prager, S., "Variational treatment of hydrodynamic interaction in polymers", J. Chem. Phys, vol. 50 (11), pp. 4831-4837, 1969.

[Subchan and Zbikowski, 2009] Subchan, S. , Zbikowski, R., "Computational Optimal Control", John Wiley and Sons, United Kingdom, 2009.

[Sussman, 1996] Sussman, H. J., "From the brachystochrone to the maximum principle and back", In Proceedings of the 35^{th} IEEE Conference on Decision and Control, p. 1588-1593, Kobe, Japan, December 1996.