SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA
FACULTY OF CHEMICAL AND FOOD TECHNOLOGY
INSTITUTE OF INFORMATION ENGINEERING, AUTOMATION AND MATHEMATICS

DYNAMIC OPTIMIZATION OF PROCESSES

Diploma thesis

FCHPT-5414-17457

Bratislava 2012 **Bc. Martin Jelemenský**

SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA
FACULTY OF CHEMICAL AND FOOD TECHNOLOGY
INSTITUTE OF INFORMATION ENGINEERING, AUTOMATION AND MATHEMATICS

DYNAMIC OPTIMIZATION OF PROCESSES

Diploma thesis

FCHPT-5414-17457

Study program: Automation and information engineering in chemical and food technologies.
Study field: 5.2.14 Automation
Workplace: Faculty of Chemical and Food Technology
Supervisor: Prof. Ing. Miroslav Fikar, DrSc.
Consultant: Ing. Radoslav Paulen

Bratislava 2012 **Bc. Martin Jelemenský**

Slovak University of Technology in Bratislava
Institute of Information Engineering, Automation and Mathematics

Faculty of Chemical and Food Technology
Academic year: 2011/2012
Reg. No.: FCHPT-5414-17457

S T U
F C H P T

DIPLOMA THESIS TOPIC

Student:	**Bc. Martin Jelemenský**
Student's ID:	17457
Study programme:	Automation and Informatization in Chemistry and Food Industry
Study field:	5.2.14 Automation
Thesis supervisor:	prof. Ing. Miroslav Fikar, DrSc.
Consultant:	Ing. Radoslav Paulen
Workplace:	ÚIAM FCHPT STU v Bratislave

Topic: **Dynamic optimization of processes**

Specification of Assignment:

This work deals with dynamic systems optimal control design. In first stage various dynamic optimization methods are studied. Second stage is devoted to application of selected methods for computing several optimal control examples. One of the examples includes a study of copolymerization reaction of styrene and alphamethylstyrene. Problem of dynamic optimization is to compute dynamic temperature profile of emulsion polymerization reaction to ensure time-optimal production of required amount of polymer and desired quality of this product.

Length of thesis: 60

Selected bibliography:

1. FIKAR, M. *Dynamická optimalizácia procesov.* 2007. 161 s. ISBN 978-80-89316-08-3.
2. MIKLEŠ, J. – FIKAR, M. *Process Modelling, Identification, and Control.* Berlin Heidelberg: Springer Berlin Heidelberg New York, 2007. 480 s. ISBN 978-3-540-71969-4.

Assignment procedure from: 13. 02. 2012

Date of thesis submission: 19. 05. 2012

L. S.

Bc. Martin Jelemenský
Student

prof. Ing. Miroslav Fikar, DrSc.
Head of office

prof. Ing. Miroslav Fikar, DrSc.
Study programme supervisor

I would like to thank my supervisor professor Miroslav Fikar for help, guidance and for giving me the opportunity to participate in the Erasmus program. I would also like to thank professor M. Abderazzak Latifi from ENSIC-INPL for support and advice which helped me and inspired me during writing of this thesis. My other thanks goes to Ing. Radoslav Paulen for patience, advice, guidance, and comments which inspired me and helped me in many ways. My last thanks goes to my family who supported me during my study.

Martin Jelemenský
Bratislava, 2012

Abstract

This work is aimed to solve dynamic optimization problems. Secondly, the aim of the work is to study methods which helps us to solve dynamic optimization problems. In this work we are dealing mainly with two numerical methods. The first method is control vector parametrization (CVP) and the second is orthogonal collocation (OC). We also discuss several approaches for computing gradients, which are very important in the computing algorithm. The work is divided into two parts. The first part discusses the theoretical basis for solving optimization problems. In the second part we show the application of the methods and procedures in several optimization problems.

Abstrakt

Táto práca je zameraná na riešenie problémov dynamickej optimalizácie. Cieľom práce je štúdium metód, ktoré nám pomáhajú pri riešení optimalizačných problémov. Práca sa zaoberá predovšetkým dvoma numerickými metódami. Prvá metóda je parametizácia vektora riadenia (CVP) a druhá metóda je ortogonálna kolokácia (OC). Tiež sa zaoberáme postupom výpočtu gradientov, ktoré sú veľmi dôležité pri výpočtovom algoritme. Práca je rozdelená do dvoch častí. Prvá časť sa zaoberá teoretickými základmi pre riešenie optimalizačných úloch. V druhej časti ukazujeme aplikáciu metód a postupov pri riešení optimalizačných problémov.

Contents

Nomenclature

A	initiator concentration [mol$\,l^{-1}$]/cooling surface [m^2]
a_s	surface area occupied by an emulsifier molecule [dm^2]
C_{pj}	cooling fluid heat capacity [$Jkg^{-1}K^{-1}$]
f	initiator efficiency
F_j	cooling fluid flowrate [ls^{-1}]
f_{MS}	α–methylstyrene molar fraction in the initial load
i_c	inhibition coefficient
J	objective function
K_d	death coefficient [h^{-1}]
k_d	rate constant for initiator decomposition [s^{-1}]
k_p	rate constant for propagation [$dm^3mol^{-1}s^{-1}$]
k_{cm}	rate constant for initiator radical entry into micelles [$dm^3micelle^{-1}s^{-1}$]
k_{cp}	rate constant for initiator radical entry into particles [$dm^3part^{-1}s^{-1}$]
K_{IP_c}	product inhibition kinetic coefficient [gL^{-1}]
K_{IS_c}	substrate inhibition kinetic coefficient [gL^{-1}]
K_{S_c}	substrate saturation kinetic coefficient [gL^{-1}]
k_{trM}	rate constant for transfer to monomer [$dm^3mol^{-1}s^{-1}$]

L kinetic chain length [$\mathrm{g\,mol^{-1}}$]

M global monomer concentration [$\mathrm{mol\,dm^{-3}}$]

m number of micelles per unit volume [$\mathrm{micelle\,dm^{-3}}$]

M_{M} monomer molecular weight [$\mathrm{g\,mol^{-1}}$]

M_{p} monomer concentration in particles [$\mathrm{mol\,dm^{-3}}$]

m_c maintenance coefficient [$\mathrm{h^{-1}}$]

$m_r C_p$ reactor total heat capacity [$\mathrm{J\,K^{-1}}$]

M_{pc} critical monomer concentration in particles [$\mathrm{mol\,dm^{-3}}$]

$\bar{M}_n$ number–average molecular weight [$\mathrm{g\,mol^{-1}}$]

$\bar{n}$ average number of radicals per particle

N number of inactive particles per unit volume [$\mathrm{particle\,dm^{-3}}$]

$N^{\bullet}$ number of active particles per unit volume [$\mathrm{particle\,dm^{-3}}$]

N_{p} total number of particles per unit volume [$\mathrm{particle\,dm^{-3}}$]

N_A Avogadro's number [$\mathrm{mol^{-1}}$]

n_s aggregation number of micelles

P dead polymer concentration [$\mathrm{mol\,dm^{-3}}$]

P_c product concentration [$\mathrm{gL^{-1}}$]

P_j number of polymers with chain length j

Q_i ith moment of the molecular weight distribution

r reaction rate [$\mathrm{gL^{-1}h^{-1}}$]

$R^{\bullet}$ initiator radical concentration [$\mathrm{mol\,dm^{-3}}$]

R_a initiator decomposition rate [$\mathrm{mol\,dm^{-3}s^{-1}}$]

R_i initiation rate [$\mathrm{mol\,dm^{-3}s^{-1}}$]

R_n particle formation rate [$\mathrm{mol\,dm^{-3}s^{-1}}$]

R_p polymerization rate [$\mathrm{mol\,dm^{-3}s^{-1}}$]

R_t termination rate [$\mathrm{mol\,dm^{-3}s^{-1}}$]

R_{trM} transfer to monomer rate [$\mathrm{mol\,dm^{-3}s^{-1}}$]

S emulsifier concentration [$\mathrm{mol\,dm^{-3}}$]

S_c substrate concentration [$\mathrm{gL^{-1}}$]

S_i ith mode of the process

T reactor temperature [K]

t time [s]

T_j jacket temperature [K]

$T_{j,in}$ cooling fluid inlet temperature [K]

U heat transfer coefficient [$\mathrm{JK^{-1}s^{-1}m^{-2}}$]

u control variable

V reactor contents volume [L]

V reactor jacket volume [L]

x state vector

$\dot{x}$ vector of state derivatives

N_I number of intervals

X monomer conversion

X_c critical monomer conversion

X_d dead biomass concentration [$\mathrm{gL^{-1}}$]

X_{ab} active biomass concentration [$\mathrm{gL^{-1}}$]

$Y_{P_{sc}S_c}$ stoichiometric yield coefficient

$Y_{X_{sc}S_c}$ stoichiometric yield coefficient

DoF degree of freedom

Greek Symbols

ε constant describing the efficiency of the particles relative to the micelles in collecting an initiator radical

ΔH polymerization reaction enthalpy [$\mathrm{J\,mol^{-1}}$]

ρ_{M} monomer density [$\mathrm{g\,dm^{-3}}$]

ρ_{p} polymer particle density [$\mathrm{g\,dm^{-3}}$]

ρ_{j} cooling fluid density [$\mathrm{g\,dm^{-3}}$]

ρ_{P} polymer density [$\mathrm{g\,dm^{-3}}$]

ω_{P} polymer weight fraction in the particles

μ specific growth rate of biomass [$\mathrm{h^{-1}}$]

μ_{max} maximal specific growth rate of biomass [$\mathrm{h^{-1}}$]

Subscripts

0 initial

f final

Superscripts

L lower bound

U upper bound

Chapter 1

Introduction

The search of optimal solution for problem is a every day struggle for everyone. Every day we solve optimization problem when we are trying to find the optimal solution for our problem. By searching for optimal solution we can use many optimization methods. For solving optimization problem we require well formulated mathematical description of the problem. We also have to consider that every process has also constraints for example maximum speed of a car, maximal flow in the pipes. Therefore search for optimal solution must obey such constraints. When we talk about optimality (optimal solution) we talk about minimization or maximization of objective function for example minimization of time, costs and energy and maximization of production and profit. In this work we show the application of the methods and procedures for solving dynamic optimization problems in chemical technological processes. These methods are divided into two groups. First methods we talk about analytical ones, and the second are numerical methods. For solving optimization problems we use mainly numerical methods, specifically control vector parametrization (CVP) and orthogonal collocation (OC). The work is divided into two parts. The first part discusses the general formulation of dynamic optimization problems. We discuss analytical and numerical methods. In the last section of the first part, we examine procedures for calculating gradients using finite differences, sensitivity equations, and adjoint variables methods. In the second part of the work we show the application of these methods on several dynamic optimization problems. In the first example we discuss the application of sensitivity equations for parameter estimation problem. In the second example, optimal control of tubular reactor, we compare numerical methods with different ways of calculating of gradients. In the next example, time-optimal control of car, we show the procedure for calculating gradients using adjoint variables and the procedure for solving optimization problem using orthogonal collocation.

In the last example, the emulsion polymerization process, we compare the results obtained using control vector parametrization method with finite differences or adjoint variables.

Part I

Theoretical basis

Chapter 2

Dynamic Optimization

Dynamic optimization is usually referred to open-loop optimal control. In this chapter we discuss the general formulation of dynamic optimization problems. We also discuss constraints on the state and control variables. (Fikar 2007)

2.1 Problem Formulation

The main requirement for solving optimization problems is well formulated and defined problem. Formulation of optimal control problem requires (Kirk 1970):

1. Mathematical description of system which has to be controlled
2. Definition of constraints
3. Specification of minimization criterion

We consider system described by a set of ordinary differential equations

$$\dot{x}(t) = f(x(t), u(t), p, t) \qquad x(t_0) = x_0 \tag{2.1}$$

where $x(t)$ is a vector of state variables, $u(t)$ is vector of control variables, p is a vector of time independent optimized variables, t is independent time variable and $x(t_0)$ is a vector of initial conditions. Next we define the objective functional which can be written in three basic forms

- Bolza form

$$J(u(t), p) = G(x(t_f), p) + \int_{t_0}^{t_f} F(x(t), u(t), p, t)\mathrm{d}t \tag{2.2}$$

- Lagrange form

$$J(u(t),p) = \int_{t_0}^{t_f} F(x(t),u(t),p,t)\mathrm{d}t \tag{2.3}$$

- Mayer form

$$J(u(t),p) = G(x(t_f),p) \tag{2.4}$$

where J represents the optimization criterion, G and F are differentiable scalar functions. Our objective is to find such control which will minimize the objective functional.

2.2 Constraints

Certain restrictions exist for all processes of chemical technology. These are the criteria which we must respect while solving the optimization problems. Wide range of constraints can occur in optimization problems. These can be divided into several cases.

- Interior equality constraints

$$h(x,u,p,t_i) = 0 \quad t_i \in [t_0,t_f] \tag{2.5}$$

- Interior inequality constraints

$$g(x,u,p,t_i) \leq 0 \quad t_i \in [t_0,t_f] \tag{2.6}$$

- terminal equality constraints

$$h(x,u,p,t_f) = 0 \tag{2.7}$$

- terminal inequality constraints

$$g(x,u,p,t_f) \leq 0 \tag{2.8}$$

Next we consider the upper and lower bounds on states and controls.

- boundaries for the states

$$x(t)^L \leq x(t) \leq x(t)^U \tag{2.9}$$

- boundaries for the controls

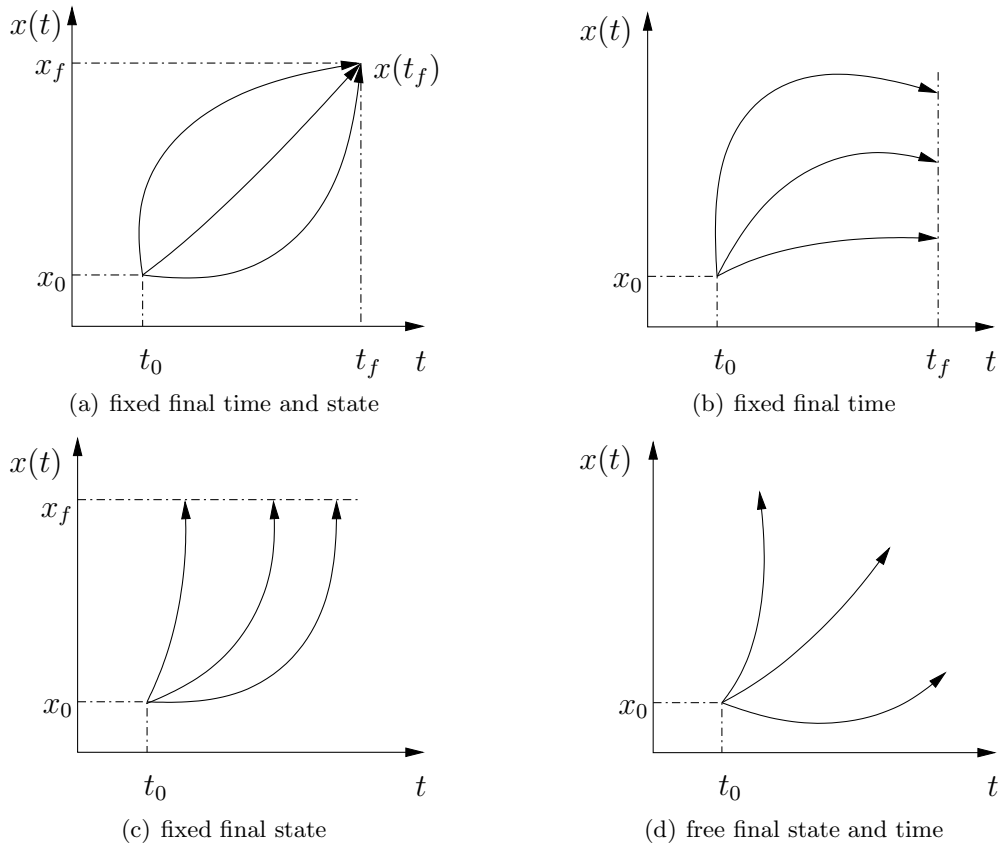
$$u(t)^L \leq u(t) \leq u(t)^U \tag{2.10}$$

Where superscripts L and U denote lower and upper boundaries respectively.

Chapter 3

Optimal Control

In this chapter we discuss the general problems with optimal control of processes. We will deal with the basic approach to solve the optimal control problems. By solving the problems of optimal control we consider the minimization or maximization of the objective functional.



(a) fixed final time and state

(b) fixed final time

(c) fixed final state

(d) free final state and time

Figure 3.1: General problems of optimal control

3.1 Problems of Optimal Control

For solving optimal control problems we can consider the maximization or minimization of objective functional. By maximization of objective function we mean the maximization of profit or conversion of reactants. As a minimization problem, we consider the minimization of costs, outlet, energy and time. In the following four cases Fig.(3.1) we show some problems of optimal control. Our objective is to find the optimal state trajectory from all admissible state trajectories $x(t)$ with the corresponding vector of control variables $u(t)$ (Hirmajer 2007). The first case of optimal control Fig.(3.1(a)) is when we have specified the final value of time and state variables. This problem may seem very easy to solve, but the specified final values of state and time variables can be very restrictive. In the next two cases Fig.(3.1(b)) and Fig.(3.1(c))we have specified the final conditions for time or state variables. By fixed final state value we can consider for example the minimization of final time by which we reach the desired state (e.g. required conversion of reactants). The least restrictive case is when the value of final state variables and time are unknown Fig.(3.1(d)).

3.2 Necessary Conditions of Optimality

This section discusses the necessary conditions of optimality which can give, beside other things, information about the gradients of objective function. First we have to mentioned that every constraint can be adjoined to the functional J by using Lagrange multipliers (Hull 2003). Then we can consider the function in following form

$$\bar{J} = J + \sum_{j=1}^{n_c} \nu_j J_j \tag{3.1}$$

where $\bar{J}$ is the augmented functional, n_c is the number of constraints and ν is a vector of Lagrange multipliers. We join the functional with the process described in equation (2.1).

$$\bar{J} = G + \int_{t_0}^{t_f} [F + \lambda^T (f - \dot{x})] \mathrm{d}t + \sum_{j=1}^{n_c} \nu_j \left\{ G_j + \int_{t_0}^{t_f} [F_j + \lambda_j^T (f - \dot{x})] \mathrm{d}t \right\} \tag{3.2}$$

We define the Hamilton function

$$H(x(t), \lambda(t), u(t), p, t) = F(x(t), u(t), p, t) + \lambda^T(t) f(x(t), u(t), p, t) \tag{3.3}$$

Next we define functions as augmented functional

$$\bar{G} = G + \sum_{j=1}^{n_c} \nu_j G_j \tag{3.4a}$$

$$\bar{F} = F + \sum_{j=1}^{n_c} \nu_j F_j \tag{3.4b}$$

$$\bar{\lambda} = \lambda + \sum_{j=1}^{n_j} \nu_j \lambda_j \tag{3.4c}$$

$$\bar{H} = H + \sum_{j=1}^{n_j} \nu_j H_j \tag{3.4d}$$

New form of augmented functional can be written as follows

$$\bar{J}(u(t), p) = \bar{G}(x(t_i), p, t_i) + \int_{t_0}^{t_f} (\bar{H}(x, \bar{\lambda}, u, p, t) - \bar{\lambda}^T \dot{x}) \mathrm{d}t \tag{3.5}$$

where $i \in \{1, \ldots . n_i + 1\}$ represents interior points. For the derivation of the augmented functional we will consider typical problems of optimization.

- fixed initial time ($t_0 = 0$),
- free initial conditions ($x(t_0, p) = x_0(p)$)
- free final conditions ($x(t_f) = x_f$)
- free final time (t_f)

Differential of functional (3.5) can be written as follows

$$\mathrm{d}\bar{J} = \mathrm{d}\bar{G} + \int_{t_0}^{t_f} \delta\bar{H}\mathrm{d}t - \int_{t_0}^{t_f} \delta(\bar{\lambda}^T \dot{x})\mathrm{d}t + (\bar{H} - \bar{\lambda}^T \dot{x})|_{t_f} \mathrm{d}t_f + \sum_{i=1}^{n_i} [\bar{H} - \bar{\lambda}^T \dot{x}]_{t_i^-}^{t_i^+} \mathrm{d}t_i \tag{3.6}$$

Using integration method *by parts* we transform $\int_{t_0}^{t_f} \delta(\bar{\lambda}^T \dot{x})\mathrm{d}t$ into

$$\begin{aligned} -\int_{t_0}^{t_f} \delta(\bar{\lambda}^T \dot{x})\mathrm{d}t &= -\int_{t_0}^{t_f} (\delta\bar{\lambda}^T \dot{x} + \bar{\lambda}^T \delta\dot{x})\mathrm{d}t \\ &= \int_{t_0}^{t_f} (\dot{\bar{\lambda}}^T \delta x - \delta\bar{\lambda}^T \dot{x})\mathrm{d}t - [\bar{\lambda}^T \delta x]_{t_0}^{t_f} - \sum_{i=1}^{n_i} [\bar{\lambda}^T \delta x]_{t_i^-}^{t_i^+} \end{aligned} \tag{3.7}$$

Now we can express all the differentials and variations in (3.6) and considering (3.7) we obtain

$$\begin{aligned} \mathrm{d}\bar{J} =& \frac{\partial \bar{G}}{\partial x^T}\bigg|_{t=t_f} \mathrm{d}x_{t_f} + \sum_{i=1}^{n_i} \frac{\partial \bar{G}}{\partial x^T}\bigg|_{t=t_i} \mathrm{d}x_{t_i} + \frac{\partial \bar{G}}{\partial p^T}\mathrm{d}p + \frac{\partial \bar{G}}{\partial t_f}\mathrm{d}t_f + \sum_{i=1}^{n_i} \frac{\partial \bar{G}}{\partial t_i}\mathrm{d}t_i \\ &+ \int_{t_0}^{t_f} \left(\frac{\partial \bar{H}}{\partial x^T}\delta x + \frac{\partial \bar{H}}{\partial \bar{\lambda}^T}\delta\bar{\lambda} + \frac{\partial \bar{H}}{\partial u^T}\delta u + \frac{\partial \bar{H}}{\partial p^T}\delta p + \dot{\bar{\lambda}}^T \delta x - \delta\bar{\lambda}^T \dot{x} \right) \mathrm{d}t \\ &- \bar{\lambda}_{t_f}^T \delta x_{t_f} + \bar{\lambda}_{t_0}^T \delta x_{t_0} + \sum_{i=1}^{n_i} (\bar{\lambda}_{t_i^-}^T \delta x_{t_i^-} - \bar{\lambda}_{t_i^+}^T \delta x_{t_i^+}) + \bar{H}_{t_f}\mathrm{d}t_f - \bar{\lambda}_{t_f}^T \dot{x}_{t_f}\mathrm{d}t_f \\ &+ \sum_{i=1}^{n_i} (\bar{H}_{t_i^-} - \bar{H}_{t_i^+})\mathrm{d}t_i + \sum_{i=1}^{n_i} (\bar{\lambda}_{t_i^-}^T \dot{x}_{t_i^-} - \bar{\lambda}_{t_i^+}^T \dot{x}_{t_i^+})\mathrm{d}t_i \end{aligned} \tag{3.8}$$

The next step is to regroup the corresponding terms. We notice that $\mathrm{d}x_{t_i} = \delta x_{t_i^{\pm}} + \dot{x}_{t_i^{\pm}} \mathrm{d}t_i$ and by equating $\delta x_{t_0} = \frac{\partial x_{t_0}}{\partial p^T} \mathrm{d}p$ we obtain

$$\begin{aligned}
\mathrm{d}\bar{J} = &\left(\left.\frac{\partial \bar{G}}{\partial x^T}\right|_{t_f} - \bar{\lambda}_{t_f}^T \right) \mathrm{d}x_{t_f} + \left(\frac{\partial \bar{G}}{\partial t_f} + \bar{H}_{t_f} \right) \mathrm{d}t_f + \left(\frac{\partial \bar{G}}{\partial p^T} + \int_{t_0}^{t_f} \frac{\partial \bar{H}}{\partial p^T} \mathrm{d}t + \frac{\partial x_0}{\partial p^T} \right) \mathrm{d}p \\
&+ \sum_{i=1}^{n_i} \left(\left.\frac{\partial \bar{G}}{\partial x^T}\right|_{t=t_i} + \bar{\lambda}_{t_i^-}^T - \bar{\lambda}_{t_i^+}^T \right) \mathrm{d}x_{t_i} + \sum_{i=1}^{n_i} \left(\frac{\partial \bar{G}}{\partial t_i} + \bar{H}_{t_i^-} - \bar{H}_{t_i^+} \right) \mathrm{d}t_i \\
&+ \int_{t_0}^{t_f} \left[\left(\frac{\partial \bar{H}}{\partial x^T} + \dot{\lambda}^T \right) \delta x + \left(\frac{\partial \bar{H}}{\partial \lambda^T} - \dot{x} \right) \delta \lambda + \frac{\partial \bar{H}}{\partial u^T} \delta u \right] \mathrm{d}t
\end{aligned} \tag{3.9}$$

The differential of function $\bar{J}$ must equal zero. Then the necessary conditions of optimality we obtain when all bracketed terms are equal to zero.(Paulen et al. 2010)

- optimality condition for

  control variables
  $$\frac{\partial \bar{H}}{\partial u^T} = 0 \tag{3.10a}$$

  parameters
  $$\frac{\partial \bar{G}}{\partial p^T} - \int_{t_0}^{t_f} \frac{\partial \bar{H}}{\partial p^T} \mathrm{d}t + \bar{\lambda}^T(t_0) \frac{\partial x_0}{\partial p^T} = 0 \tag{3.10b}$$

  final time
  $$\frac{\partial \bar{G}}{\partial t_f} + \bar{H}_{t_f} = 0 \tag{3.10c}$$

  state variables
  $$\dot{x} = \frac{\partial \bar{H}}{\partial \bar{\lambda}} \quad \forall t \in [t_0, t_f] \tag{3.10d}$$

- optimal switching conditions for

  times
  $$\frac{\partial \bar{G}}{\partial t_i} + \bar{H}_{t_i^-} - \bar{H}_{t_i^+} = 0 \quad \forall i \in \{1, \ldots, n_i\} \tag{3.10e}$$

  adjoint variables
  $$\left.\frac{\partial \bar{G}}{\partial x^T}\right|_{t=t_i} + \bar{\lambda}_{t_i^-}^T - \bar{\lambda}_{t_i^+}^T = 0 \quad \forall i \in \{1, \ldots, n_i\} \tag{3.10f}$$

- adjoint variables

  definition
  $$\dot{\bar{\lambda}} = -\frac{\partial \bar{H}}{\partial x} \quad \forall t \in [t_0, t_f] \tag{3.10g}$$

  boundary conditions
  $$\bar{\lambda}_{t_f} = \left.\frac{\partial \bar{G}}{\partial x}\right|_{t=t_f} \tag{3.10h}$$

Chapter 4

Dynamic Optimization Methods

For solving optimization problems we can use several optimization methods which can be divided into two main groups

- Analytical methods
- Numerical methods

4.1 Analytical Methods

Analytical methods encompass

- Dynamic Programming
- Pontrygin's principle of minimum
- Variational Calculus

4.1.1 Dynamic Programming

Dynamic programming is based on Bellman's principle of optimality (Bellman 1957, Fikar 2007). This principle is based on the fact that optimal control of process depends only on the initial and final state. We can use dynamic programming for continuous or discontinuous systems. When we consider that trajectory form point A to point C is optimal (see Fig 4.1), then the trajectory form point B to point C is also optimal. It shows what we have already stated previously. The optimal control depends on the initial and final state but not on the history of control actions.

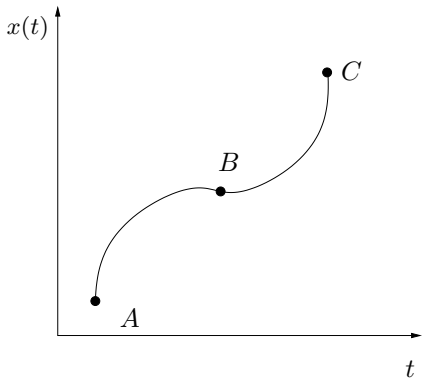


Figure 4.1: Optimal trajectory

Consider functional eq.(2.2) and system eq.(2.1). We also consider that our problem has a solution. We define function which is also called the Bellman function

$$\nu(t, x(t)) = \min_{u(t)} \left[G(t_f, x(t_f)) + \int_t^{t_f} F(x(t), u(t), t)dt \right] \tag{4.1}$$

differentiating eq.(4.1) we obtain the partial derivation of Bellman function

$$-\frac{\partial \nu}{\partial t} = \min_{u(t)} \left[F(x(t), u(t), t) + \left(\frac{\partial \nu}{\partial x(t)} \right)^T f(x(t), u(u), t) \right] \tag{4.2}$$

by satisfying the boundary condition

$$\nu(t_f, x(t_f)) = G(t_f, x(t_f)) \tag{4.3}$$

Bellman's equation (4.2) and the boundary conditions (4.3) representing necessary conditions for finding minimum.(Hirmajer 2007, Čižniar 2005)

4.1.2 Pontryagin's Principle of Minimum

Pontryagin's principle of minimum (maximum) stands for a very efficient approach by solving optimization control problems. This method is very useful when we are trying to get the system form one steady state to another.(Hirmajer 2007, Pontryagin 1964, Čižniar 2005) We consider the control problem (2.1) and Hamiltonian function (3.3). Next we define the adjoint variable

$$\lambda(t) = \frac{\partial \nu}{\partial x} \tag{4.4}$$

next we differentiate separately left and right side of adjoint system with the respect to x

$$-\frac{\partial^2 \nu}{\partial x \partial t} = \frac{\partial H}{\partial x} + \frac{\partial^2 \nu \partial H}{\partial x^2 \partial \lambda} \tag{4.5}$$

$$\dot{\lambda} = \frac{\partial^2 \nu}{\partial x^2}\dot{x} + \frac{\partial^2 \nu}{\partial x \partial t} \tag{4.6}$$

We obtain canonical differential equations which represent the principle of minimum

$$\dot{\lambda}(t) = -\frac{\partial H}{\partial x} \tag{4.7}$$

$$\dot{x} = \frac{\partial H}{\partial \lambda} \tag{4.8}$$

The necessary conditions for control problem (2.1), using the Pontryagin principle of minimum are the following

- optimality condition for control variable (3.10a)
- adjoint variables definition (3.10g)
- adjoint variables boundary conditions (3.10h)

4.1.3 Variational Calculus

Basic relations for variational calculus are obtained from Bellman partial differential equations. Variational calculus is applied only for specific optimal control problems.(Hirmajer 2007)

We consider Euler-Lagrange differential equation

$$\frac{\partial \tau}{\partial x} - \frac{d}{dt}\left(\frac{\partial \tau}{\partial \dot{x}}\right) = 0 \tag{4.9}$$

where τ is the Lagrange function define as

$$\tau(x, \dot{x}, u, \lambda, t) = F(x, u, t) + \lambda^T[f(x, u, t) - \dot{x}] \tag{4.10}$$

we consider objective function define eq.(2.2) then we can formulate the necessary conditions of optimality

- optimality conditions for control variables

$$\frac{\partial \tau}{\partial u} = 0 \quad t \in [t_0, t_f] \tag{4.11}$$

- definition of adjoint variables

$$\dot{\lambda}(t) = -\frac{\partial \tau}{\partial x} \quad t \in [t_0, t_f] \tag{4.12}$$

- terminal conditions for adjoint variables

$$\lambda(t_f) = \frac{\partial G}{\partial x(t_f)} \tag{4.13}$$

4.2 Numerical Methods

Numerical methods can be divide them into two main groups

- **indirect methods**

  Boundary Condition Iteration (BCI)

  Control Vector Iteration (CVI)

- **direct methods**

  Control Vector Parametrization (CVP)

  Orthogonal Collocation (OC)

4.2.1 Control Vector Parametrization

Control vector parametrization (CVP) falls into the class of numerical methods for solving dynamic optimization problems. This method is based on approximation (discretization) of original continuous control trajectory with finite number of control intervals of polynomial nature. In this work we consider constant and linear control over these intervals. (Paulen et al. 2010, Teo et al. 1991)

We consider system described by equation (2.1). The main issue with continuous control trajectory is that we need to find optimal value of control variable at each time. As it is shown at Fig. 4.2 there is infinite number of degrees of freedom. Hence that by discretization, the

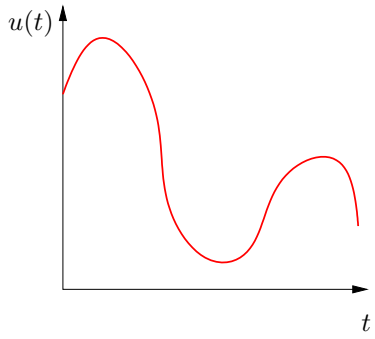


Figure 4.2: Continuous control trajectory.

original control profile, we replace the original infinite dimensional decision trajectory with one possessing finite number of degrees of freedom (Fig. 4.3). In this case we consider 6 optimization parameters (3 constant control values, 3 corresponding time intervals). This

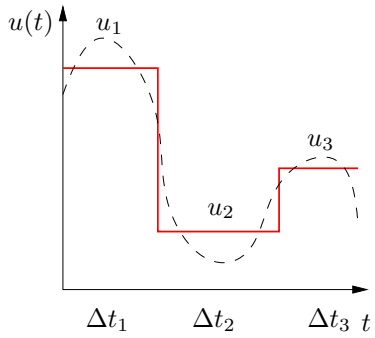


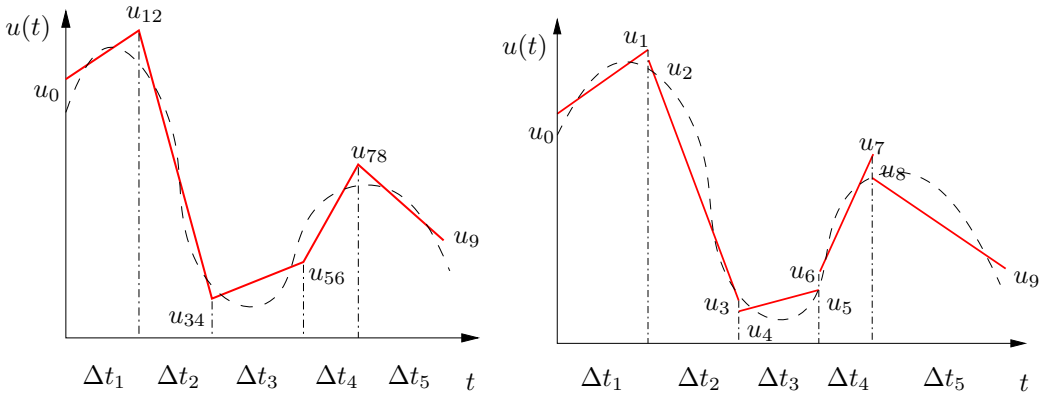
Figure 4.3: Discretized control trajectory.

yields 6 degrees of freedom to our optimization problem. Discretized control can be expressed as

$$u(t) = u_i \qquad t_{i-1} \leq t < t_i \tag{4.14}$$

where u_i represent constant control value as defined in Fig.(4.3). We also define the length of time intervals as $\Delta t_i = t_i - t_{i-1}$. So we transformed the original infinite dimensional optimization problem into finite dimensional problem of non-linear programming (Paulen et al. 2010). Resulting problem involves finite number of decision parameters (degrees of freedom).

We can also consider replacing the continuous control trajectory with piece-wise linear control (see Fig. 4.4). Unlike in case of piece-wise constant trajectory (see Fig. 4.3), we may



(a) continuous case (b) discontinuous case

Figure 4.4: Continuous and discontinuous piece-wise linear control.

consider continuous as well as discontinuous control profile. Then the continuous piece-wise linear control trajectory (Fig. 4.4) on each time interval can be described with following

equation.

$$u(t) = u_{i-1} + \frac{u_i - u_{i-1}}{t_i - t_{i-1}}(t - t_{i-1}) \qquad \forall t \in [t_{i-1}, t_i] \tag{4.15}$$

Control trajectories for discontinuous profiles can be described analogically.

Here we show how continuity of considered discretization influence the number of degrees of freedom:

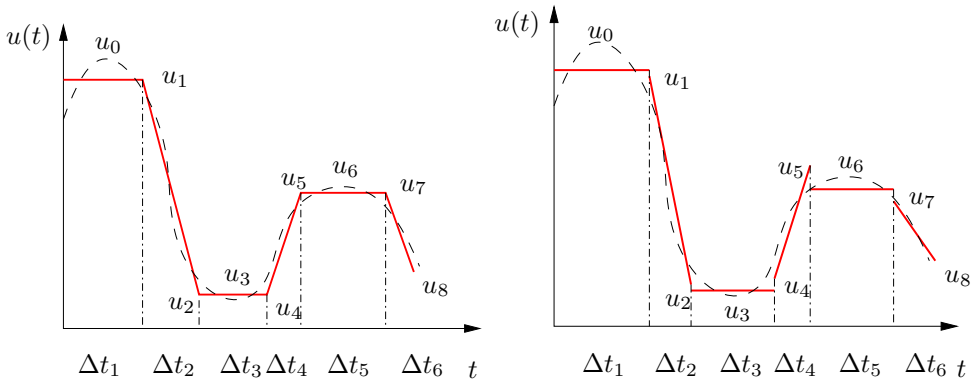
- continuous control Fig.(4.4(a)) (DoF = 11)

  Since $u_{12} : u_1 = u_2$, $u_{34} : u_3 = u_4$, $u_{56} : u_5 = u_6$, $u_{78} : u_7 = u_8$, the final number of decision variables is given by 6 control variables and 5 time intervals.

- discontinuous control (DoF = 15)

  In this case we optimize 10 control variables and 5 time intervals.

Next we consider special case. We replace the continuous control trajectory with piecewise constant and linear control Fig.(4.5). We consider continuous and discontinuous control



(a) continuous (b) discontinuous

Figure 4.5: Continuous and discontinuous constant-linear control

profile with 3 constant and 2 linear segments. Continuous or discontinuous control trajectory on each time interval can be again described using either equation (4.15) for linear segments or equation (4.14) for constant segments.

Fig.(4.5) shows number of degrees of freedom for each considered strategy:

- continuous control (DOF = 10)

  In this case, $u_0 = u_1$, $u_2 = u_3 = u_4$, $u_5 = u_6 = u_7$ and optimization is to decide about values of 4 control variables and lengths of 6 time intervals.

- discontinuous control (DOF = 15)

  We have to optimize 9 control variables and 6 time intervals.

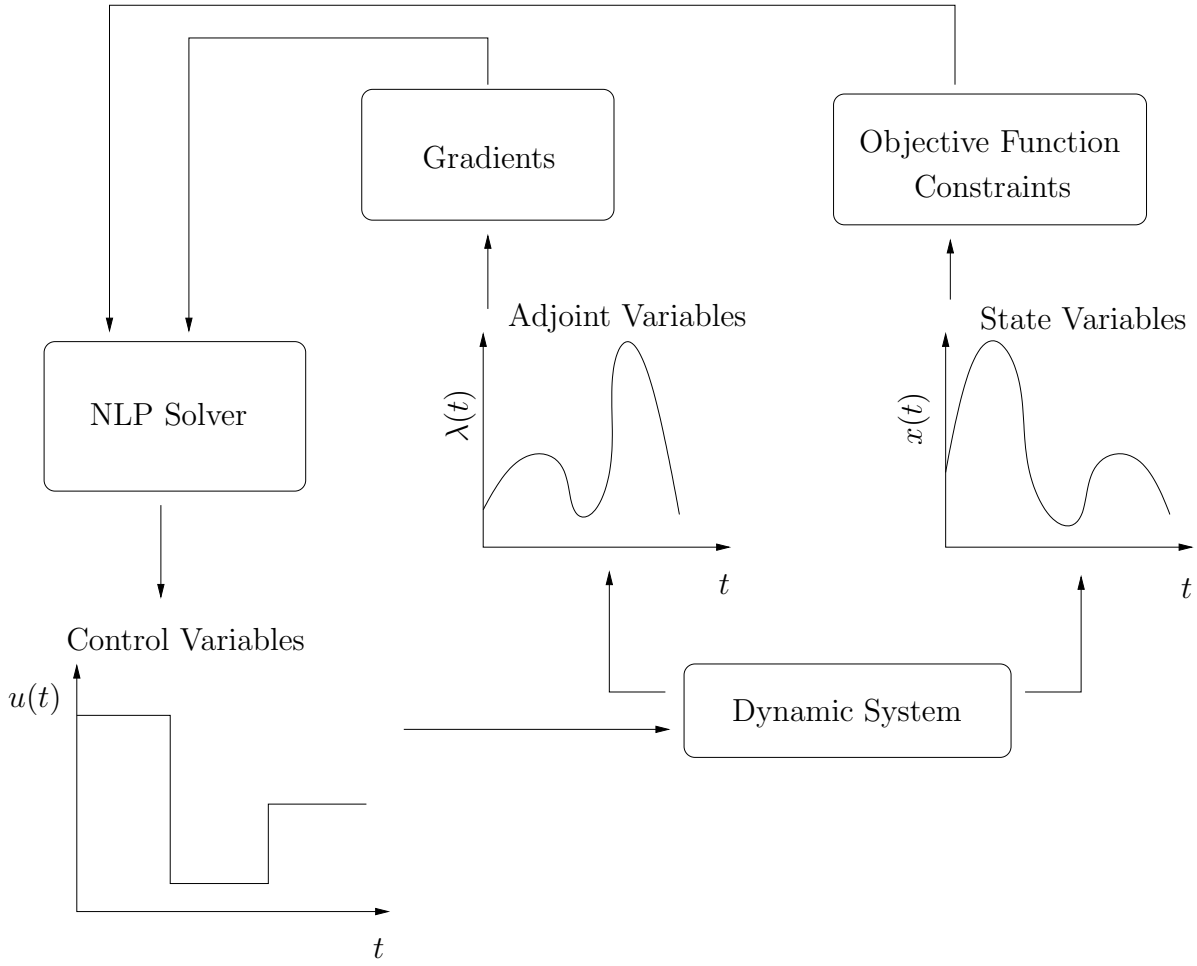


Figure 4.6: Algorithm of the CVP method

In Fig.(4.6), we show general algorithm for solving optimization problems using CVP method with adjoint variables which are used for computing gradients. The first step is to discretized the continuous control trajectory with finite numbers of intervals with constant control. Next we integrate the process model eq.(2.1) forward in time and adjoint system eq.(3.10g) backward in time. From integration of process model we calculate the value of constraints and from integration of adjoint system we calculate gradients on objective function and constraints. The final step is to evaluate the value of objective function with respect to constraints with NLP solver. The whole process is repeated until the value of objective function stops to change.

4.2.2 Orthogonal Collocation

Orthogonal collocation (OC) transforms the original dynamic optimization problem to static through the polynomial approximation of state and control profiles. In order to provide the orthogonal behavior we use Lagrange polynomials for the approximation. The roots of

Legendre polynomials determines the distribution of collocation points.(Cuthrell and Biegler 1987, Lauw-Bieng and Biegler 1991, Čižniar 2005).

We consider the system of ordinary differential equations (2.1) with finite numbers of elements i in time $t \in [\xi_i, \xi_{i+1}]$, we also consider constraints described in equations (2.5) and (2.6). The next step is to choose random time interval i shown on picture Fig.(4.7) with time $t \in [\xi_i, \xi_{i+1}]$, state and control variables are approximated through Lagrange polynomials which are written as follow (Fikar 2007)

$$x^n_{K+1}(t) = \sum_{j=0}^{K} x^n_{ij}\phi_j(t) \qquad \phi_j(t) = \prod_{k=0,j}^{K} \frac{t - t_{ik}}{t_{ij} - t_{ik}} \tag{4.16}$$

$$u^m_K(t) = \sum_{j=1}^{K} u_{ij}\theta_j(t) \qquad \theta_j(t) = \prod_{k=1,j}^{K} \frac{t - t_{ik}}{t_{ij} - t_{ik}} \tag{4.17}$$

$$\text{for } i = 1, \ldots, N_I$$

where N_I represents finite numbers of elements, $k = 0, j$ means that by $k = 0$ we begin, but $k \neq j$, K is number of collocation points, n is index of state variable and m is index of control variable. When we consider that number of collocation points for states and control equals,

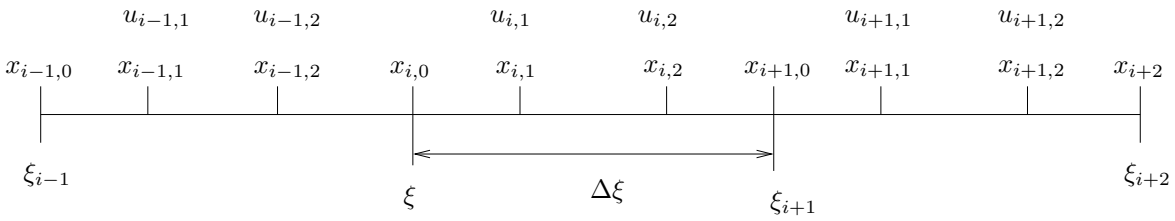


Figure 4.7: Distribution of time intervals and collocation points for state and control variables

then we can consider the distribution described on Fig.(4.7). Lagrange polynomials have the following properties

$$x_{K+1}(t_{ij}) = x_{ij} \tag{4.18}$$

By mentioned polynomials we can directly define the constraints on state and control variables. Then the equations (2.1) which described the system, can be written in collocation points as follows, while we have to consider the normalized interval $\Delta\xi_i(\tau)$ where $\tau \in [0, 1]$

$$\Delta\xi \boldsymbol{r}(t_{ik}) = M \sum_{j=0}^{K} x^n_{ij}\dot{\phi}_j(\tau_k) - \Delta\xi f(t_{ik}, x_{ik}, u_{ik}) \tag{4.19}$$

$$i = 1, \ldots, N_I, \quad j = 0, \ldots, K, \quad k = 1, \ldots, K$$

where $\dot{\phi}_j(\tau_k)$ is independent from interval length, but only from the distribution of collocation points on the normalized interval. In order to obtain the desired NLP problem we transform objective function and constraints the same way as we transform the system of ordinary differential equations.

Chapter 5

Methods for Computing Gradients

In this section we discuss the methods for computing gradients. In dynamic optimization gradients are the basis in the computing algorithm. For obtaining gradients we can use one of the following methods.

5.1 Finite Differences

The system is integrated n-times and in each of them is one of the variable y_i changed. Then we can express the gradients as follow

$$\nabla_{y_j} J_i = \frac{J_i(y_1, \ldots, y_j + \Delta y_j, \ldots, y_n) - J_i(y)}{\Delta y_j} \qquad i = 0, \ldots, n_e + n_{ne} \tag{5.1}$$

where n_e is the number of equality constraints, n_{ne} is the number of inequality constraints and y_i is the vector of optimized variables define as follow

$$y^T = (\Delta t_1, \ldots, \Delta t_f, u_1^T, \ldots, u_f^T, p_1 \ldots, p_p) \tag{5.2}$$

where p is number of paramters. The main advantage of this method is the simplicity of implementation. The disadvantage is the amount of integration that is necessary to repeat for each optimized parameter (Fikar 2007). These gradients are calculated only with some precision. Finite difference method is the least accurate method for computing gradients. The main issue of finite differences method, is how to slightly change the variable that the change would not be too big or too small. By too small (too big) we mean for example when we evaluate the objective function or constraints we must integrate the system forward in time. This integration is accurate only to some point, so it can happen that the gradients calculated with finite differences method are inaccurate considering the integration precision. General procedure for calculating gradients with finite differences method is following:

1. Initial guess, y, for values of optimized variables eq.(5.2).
2. Forward integration of system eq.(2.1).
3. Calculating the value of objective function $J_i(y)$.
4. Choosing the value of Δy (small positive number) and initializing $i = 1$.
5. Changing the value of i-th parameter in vector of optimized variables by Δy.
6. Forward integration of system with new vector of optimized variables.
7. Calculating new values for objective function $J_i(y_1, \ldots, y_j + \Delta y_j, \ldots, y_n)$.
8. Calculating gradients according the equation (5.1)
9. If i equals the number of optimized variables then quit. Else increment i and go to step 5.

5.2 Adjoint Variables

Form the necessary conditions of optimality we define the adjoint system and adjoint variables eq.(3.10g) and eq.(3.10h). In the next step we express the gradients to objective function as

$$\frac{\partial \bar{J}}{\partial t_f} = \bar{H}(t_f) + \frac{\partial \bar{G}}{\partial t_f} \tag{5.3}$$

$$\frac{\partial \bar{J}}{\partial t_j} = \bar{H}(t_j^-) - \bar{H}(t_j^+) \qquad j = 1, \ldots, N_I - 1 \tag{5.4}$$

$$\frac{\partial \bar{J}}{\partial p^T} = \frac{\partial \bar{G}}{\partial p^T} + \bar{J}_p(t_0) + \bar{\lambda}_{t_0}^T \frac{\partial x_0}{\partial p^T} \tag{5.5}$$

$$\frac{\partial \bar{J}}{\partial u_j} = \bar{J}_u(t_{j-1}) - \bar{J}_u(t_j) \qquad j = 1, \ldots, N_I - 1 \tag{5.6}$$

where

$$\dot{\bar{J}}_u = \frac{\partial \bar{H}}{\partial u^T} \tag{5.7}$$

$$\dot{\bar{J}}_p = \frac{\partial \bar{H}}{\partial p^T} \tag{5.8}$$

By expressing gradients on equations (5.3) to (5.6) we have to realize that the optimization variables are time increments

$$t_f = \sum_{i=1}^{N_I} \Delta t_i \tag{5.9}$$

where N_I represents finite numbers of elements. For expressing gradients on time increments the final gradients are expressed as

$$\frac{\partial \bar{J}_i}{\partial \Delta t_j} = \sum_{k=1}^{N_I} \frac{\partial \bar{J}_i}{\partial t_k} \tag{5.10}$$

The general procedure for expressing gradients on objective functional and constraints can be written in the following steps(Paulen 2010)

1. Forward integration of system eq.(2.1) with current guess of optimized variables.
2. Initializing of adjoint variables according to eq.(3.10h)
3. Backward integration of adjoint system eq.(3.10g), eq.(5.7) and eq.(5.8)
4. Calculating gradients according the equations (5.3) to (5.6).

By the backward integration of adjoint system we require the knowledge of state variables $x(t)$. One of the possibilities is to integrate the state equations together with the adjoint equations backward in time. By this approach there may occur numerical problems because the backward integration of state equations can be unstable (Fikar 2007). The recommended approach is to store the values of the state variables using some (possibly dense) grid of time points. Then, if we integrate backward in time we use these grid points and interpolate between respective state values.

5.3 Sensitivity Equations

By integration of the sensitivity system to state variables we obtain the sensitivities of individual states and parameters that we optimize (Caracotsios and Stewart 1985). The sensitivity equations for state variables and parameters are define as follow

$$\dot{s}_{u_{ij}}(t) = \frac{d}{dt}\left(\frac{\partial x}{\partial u_{ij}^T}\right) = \frac{\partial f_i}{\partial x^T} s_{u_{ij}} + \frac{\partial f_i}{\partial u^T} \tag{5.11}$$

$$\dot{s}_p(t) = \frac{d}{dt}\left(\frac{\partial x}{\partial p^T}\right) = \frac{\partial f_i}{\partial x^T} s_p + \frac{\partial f_i}{\partial p^T} \tag{5.12}$$

where $s_{u_{ij}}$ and s_p are sensitivity coefficients. The initial conditions for sensitivities are written as follows

$$s_k(0) = \frac{\partial x(0)}{\partial u} = \frac{\partial x_0}{\partial u} \tag{5.13}$$

if the initial conditions depends on certain parameter, following equation is applied

$$x(t_0) = x_0(p) \quad \Rightarrow \quad s_k(0) \neq 0$$

When we defined the sensitivities then we can calculate the gradients according to the optimization criterion

$$\nabla_{u_{ij}} J = \sum_{i=1}^{N_I} \frac{\partial G}{\partial x^T(t_i^-)} s_{u_{ij}} + \sum_{i=1}^{N_I} \frac{\partial G}{\partial u_i^T} + \int_{t_0}^{t_f} \left(\frac{\partial f_i}{\partial x^T} s_{u_{ij}} + \frac{\partial f_i}{\partial u^T} \right) dt \tag{5.14}$$

$$\nabla_p J = \sum_{i=1}^{N_I} \frac{\partial G}{\partial x^T(t_i^-)} s_p + \frac{\partial G}{\partial p^T} + \int_{t_0}^{t_f} \left(\frac{\partial f_i}{\partial x^T} s_p + \frac{\partial f_i}{\partial p^T} \right) dt \tag{5.15}$$

Calculation of gradients using the sensitivity equations are mainly used when we have few optimized variables but many constraints (Fikar 2007).

The procedure for calculating gradients using sensitivity equations is following (Hirmajer 2007)

1. Guess for initial values of optimized variables.
2. Forward integration of system eq.(2.1) and sensitivity equations (5.11) and (5.12).
3. Calculating the value of objective function and constraints.
4. Calculating the gradients according the equations (5.14) and (5.15).

Part II

Examples

Chapter 6

Lactic Acid Fermentation

Biological engineering problems can be described by a set of ordinary differential. These dynamic models are semi-empirical and they rely on empirical data. Mathematical description of bioprocesses leads to models with parameters which we have to determined from experimental data. The aim was to calculate the optimal value of prescribed parameters in bioprocess applications. (Kovács 2010).

6.1 Problem Formulation

We consider fermentation process for the production of lactic acid using saccharose as substrate. The saccharose is converted into both biomass and lactic acid during the fermentation. The microbial reaction is given by

$$S_c \xrightarrow{X_{ab}} P_c$$

In the reaction we use catalyst X_{ab} which concentration changes in time. The rate of production growth of the micro-organism X_{ab} is given by

$$r_{X_{ab}} = \mu X_{ab} \tag{6.1}$$

where μ is the specific growth of the biomass and given by

$$\mu = \frac{\mu_{max} S_c}{K_{S_c} + S_c + \left(\frac{S_c^2}{K_{IS_c}}\right)} \frac{1}{1 + \left(\frac{P_c}{K_{IP_c}}\right)^{n_c}} - K_d \tag{6.2}$$

The rate of consumption r_{S_c} of substrate is given by

$$r_{S_c} = -\frac{1}{Y_{X_{ab}S_c}} r_{X_{ab}} + m_c X_{ab} \tag{6.3}$$

where the negative sign signifies that the substrate concentration decreases due to its consumption by the micro-organisms. Note that r_{X_c} is positive because the micro-organisms grow during the reaction. The rate of production r_{P_c} of lactic acid is give by

$$r_{P_c} = Y_{P_cS_c} r_{S_c} \tag{6.4}$$

Cell death rate r_{X_d} follows first-order decay

$$r_{X_d} = -K_d X_{ab} \tag{6.5}$$

The dynamic model is described by four differential equations

$$\dot{X}_{ab} = r_{X_{ab}} \tag{6.6}$$

$$\dot{S}_c = r_{S_c} \tag{6.7}$$

$$\dot{P}_c = r_{P_c} \tag{6.8}$$

$$\dot{X}_d = r_{X_d} \tag{6.9}$$

The total biomass X_t is a sum of the active biomass X_{ab} and the dead cells X_d such as

$$X_t = X_{ab} + X_d \tag{6.10}$$

Our objective is to find optimal value for nine parameters

$$p = [\mu_{max}, K_{S_c}, K_d, K_{IS_c}, K_{IP_c}, Y_{X_{ab}S_c}, Y_{P_cS_c}, i_c, m_c]$$

Based on the technological considerations, it can be assumed that the parameters range between the below defined lower and upper bounds

$$lb = [0.1, 0.01, 0.01, 1e1, 1e\text{-}1, 0.01, 1e\text{-}3, 1, 1e\text{-}4]$$

$$ub = [0.9, 1, 0.5, 1e4, 1e3, 0.6, 1, 10, 1]$$

Further we were given the experimental values for S_c, P_c and X_t for three measurements at different times.

6.2 Results

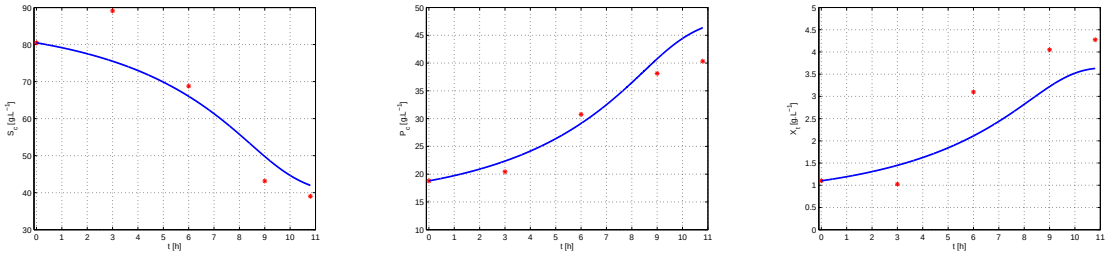
Along with the example were provided experimental data for three measurements which were obtained in different time. According to specified experimental data we calculated the optimal trajectory for the calculated optimal parameters. Using sensitivity equations for calculating gradients we calculated the optimal values for nine parameters

$$\begin{array}{lll} \mu_{max} = 0.3328 & K_{IS_c} = 1000 & Y_{P_cS_c} = 0.7159 \\ K_{S_c} = 1.000 & K_{IP_c} = 50.5444 & i_c = 10 \\ K_d = 0.1098 & Y_{X_{ab}S_c} = 0.0858 & m_c = 1.000 \end{array} \tag{6.11}$$

Value of objective function

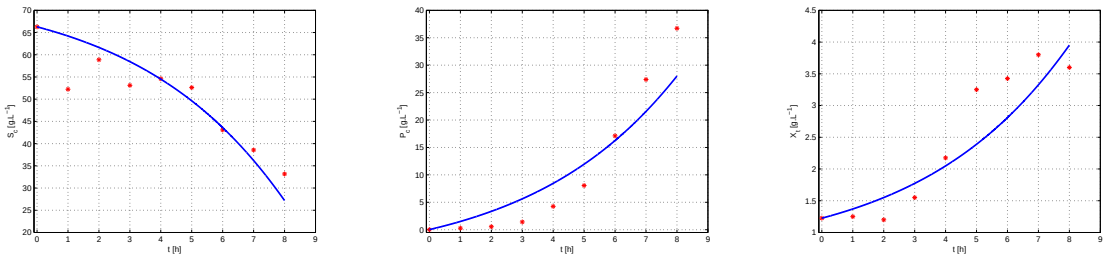
$$J = 838.2669$$

For calculating optimal values of the parameters we used Matlab integrated NLP-solver $fmincon$. Integration was performed using $ode45$ integrator where 4^{th} order Runge-Kutta numerical integration method is implemented. We can notice that most of the estimated parameters (6.11) are on the lower and upper bounds. Better results can be obtained by considering larger permitted space for optimization. In pictures Fig.(6.1), Fig.(6.2) and Fig.(6.3) we compare measured data (*) and the state model (-) obtained by numerical integration of the estimated parameters for the concentration of substrate, product and total biomass for all three measurements.



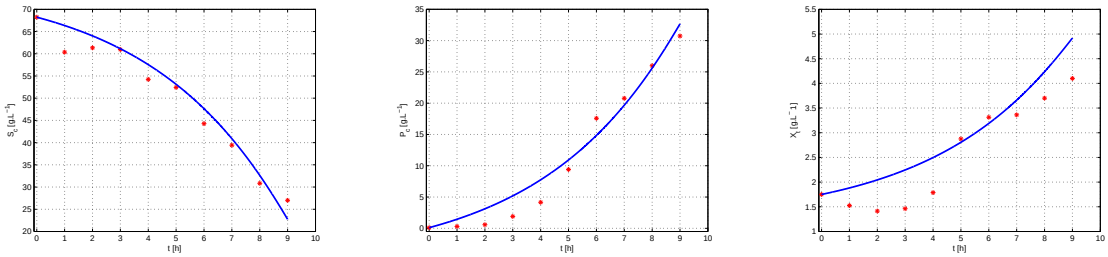
(a) Substrate concentration (b) Product concentration (c) Total biomass

Figure 6.1: Comparison of measured data (*) and the state model (-) for the first measurement



(a) Substrate concentration (b) Product concentration (c) Total biomass

Figure 6.2: Comparison of measured data (*) and the state model (-) for the second measurement



(a) Substrate concentration (b) Product concentration (c) Total biomass

Figure 6.3: Comparison of measured data (*) and the state model (-) for the third measurement

Chapter 7

Control of Chemical Reactor

In this section we deal with the optimal control problem of tubular reactor. For solving the problem we used two numerical methods. The first numerical method is control vector parametrization(CVP). And the second method is orthogonal collocation(OC). For computing gradients with CVP we used finite differences method(FD) and adjoint variables(AV). (Čižniar et al. 2005)

7.1 Problem Formulation

The aim of the example was to calculate such control of tubular reactor, which will maximize the concentration of the desired component B at the end of the reaction.
We consider two parallel reaction

$$A \rightarrow B$$
$$B \rightarrow C$$

The system is described by two ordinary differential equations

$$\dot{x}_1 = -(u + 0.5u^2)x_1 \qquad x_1(0) = 1 \tag{7.1}$$
$$\dot{x}_2 = ux_1 \qquad x_2(0) = 0 \tag{7.2}$$
$$u \in [0, 5] \qquad t_f = 1 \tag{7.3}$$

where, x_1 is the concentration of the component A, x_2 is the concentration of component B and u is the control variable. The objective function can be written as follow

$$\max_{u(t)} x_2(t_f) \tag{7.4}$$

7.2 Results

The example is divided into two parts. In the first part (Case 1) the length of time intervals were optimized variables and the objective was to optimal control of the tubular reactor. In the second part (Case 2) we fixed the length of the time intervals and our objective was to calculate the optimal control of tubular reactor. For the method of orthogonal collocation we used 5 collocation points for state variables and 2 collocation points for control variables. By both numerical methods we used 4 time intervals for state variables and control variables. For calculating the optimal control trajectory and value of the time intervals we used Matlab integrated NLP-solver *fmincon*. Obtained results are shown in Table(7.1). In the first part (Case 1) we compare the results where we optimized control variables and time intervals. In the second part (Case 2) we fixed the time intervals and the optimized variable was control variable. Further for both parts we compare the value of objective function ($x_2(t_f)$), number of NLP iteration (#it) and computing time (CPU). For obtaining results we used two numerical methods. The first method was control vector parametrization (CVP) by which we used two ways for computing gradients - adjoint variables (AV) and finite differences (FD). The second numerical method was orthogonal collocation (OC). By comparing the results

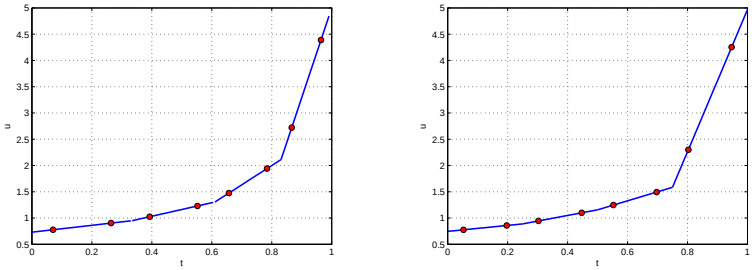
Table 7.1: Comparison of different numerical methods by fixed and optimized time intervals

	Case 1			Case 2		
Method	CPU [s]	#it	$x_2(t_f)$	CPU [s]	#it	$x_2(t_f)$
CVP FD	47.99	50	0.5706	27.5076	28	0.5664
CVP AV	6.7441	5	0.5591	4.9982	29	0.5664
OC	63.0022	871	0.5730	4.6768	83	0.5726

for the first part (Case 1) we notice that the method of orthogonal collocation yield the best value of objective function of all three methods. But compare to other method it needed more NLP iterations and computing time. We notice that CVP FD yields better results for objective function than CVP AV. This behaviour can occur when the optimization indetified only local minimum. By comparing the results of the second part (Case 2) we notice that by using OC we obtained the maximum value of objective function by minimum computing time but NLP solver needed more iterations to converge. If we compare the results using CVP method with AV and FD we can conclude that by using CVP FD we calculated the maximal value of objective function but the computing time and NLP iterations increased compare to CVP AV. We can also conclude that by optimizing not only control variables but also time

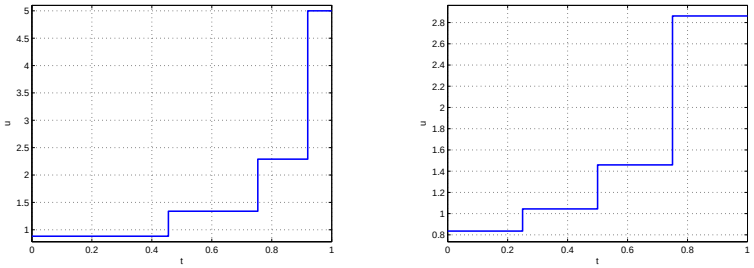
intervals we obtain better results compare to results obtained when we optimized only control variables. This behavior was expected because by increase of degrees of freedom and also by providing accurate gradients, the optimization algorithm was able to approximate better the objective function value.

In the following pictures Figs.(7.1),(7.2) and (7.3) we compare the optimal control trajectories for all three methods.



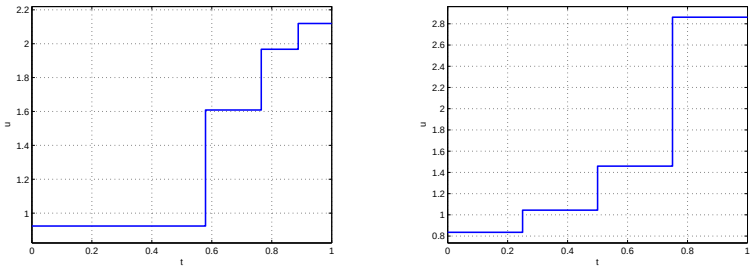
(a) Optimized u,Δt (b) Optimized u

Figure 7.1: Comparison of control trajectory by optimized and fixed time intervals for OC



(a) Optimized u,Δt (b) Optimized u

Figure 7.2: Comparison of control trajectory by optimized and fixed time intervals for CVP FD



(a) Optimized u,Δt (b) Optimized u

Figure 7.3: Comparison of control trajectory by optimized and fixed time intervals for CVP AV

Chapter 8

Time-Optimal Control of Car

In this chapter we discuss the application of two dynamic optimization methods: control vector parametrization and orthogonal collocation. In this example we compare two mentioned methods. The first method was CVP with two ways of calculating gradients - adjoint variables (AV) and finite differences (FD). The second method was OC with finite difference gradients. In both cases we used Matlab NLP solver *fmincon* with the same initial conditions. Our goal was to minimize the final time for which the car gets from point 0 to point p.

Figure 8.1: Optimal control of car

8.1 Problem Formulation

The system is described with two differential equations

$$\dot{x}_1 = x_2 \tag{8.1a}$$

$$\dot{x}_2 = u \tag{8.1b}$$

where x_1 is the track, x_2 is the speed of the car and u is the acceleration of car. In both cases we consider the same initial and terminate conditions

$$x_1(0) = 0 \qquad x_1(t_f) = 300 \tag{8.2a}$$

$$x_2(0) = 0 \qquad x_2(t_f) = 0 \tag{8.2b}$$

the constraints on control can be written as follow

$$-2 \leq u \leq 2 \qquad u^L \leq u \leq u^U \tag{8.3}$$

the objective function is follow

$$\min_u t_f \tag{8.4}$$

8.2 Procedure for CVP

We consider piece-wise constant control over the time intervals. With piece-wise constant control over the interval we can convert the dynamic optimization problem into nonlinear programing problem.(Hirmajer 2007) We consider the functional (8.4) and constraints (8.2). For computing gradients on objective function and constraints we procedure described in section 5.2. Further we used two time intervals Fig.(8.2) with piece-wise constant control. In the following procedure we show the expression of gradients on objective function. The

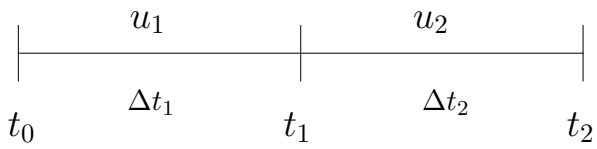


Figure 8.2: CVP example for two time intervals with piece-wise constant control

objective function can be written as follow

$$J_0 = \min_u tf \tag{8.5}$$

the second step is to express the Hamilton function

$$H_0 = \lambda^{0,T} f(x,u) = \lambda_1^0 x_2 + \lambda_2^0 u \tag{8.6}$$

where f is the right side of differential equations. Further we express the boundary conditions for adjoint variables. Adjoint variables are define only in the final time

$$\lambda(t_f) = \frac{\partial \bar{G}}{\partial x^T} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{8.7}$$

according to equation (3.10g) we obtain differential equations for adjoint system

$$\dot{\lambda}_1 = 0 \tag{8.8}$$

$$\dot{\lambda}_2 = -\lambda_1 \tag{8.9}$$

Gradients on objective function are express from equations (5.3) and (5.4)

$$\frac{\partial J_0}{\partial t_f} = \frac{\partial J_0}{\partial t_2} = 1 \tag{8.10}$$

$$\frac{\partial J_0}{\partial t_i} = \left[\lambda_1^0(t_i^-)x_2(t_i^-) + \lambda_2^0(t_i^-)u(t_i^-)\right] - \left[\lambda_1^0(t_i^+)x_2(t_i^+) + \lambda_2^0(t_i^+)u(t_i^+)\right] \tag{8.11}$$

by simplifying the equation (8.11) we obtain

$$\frac{\partial J_0}{\partial t_i} = \lambda_2^0(t_i)(u(t_i^-) - u(t_i^+)) \tag{8.12}$$

where $i = 0, 1$. In Tab.(8.1) we summarized the procedure how to express gradients for each time interval. We can also notice that we expressed the constraints as a new minimization criterion. Then the procedure for expressing gradients on constraints is the same as for objective function.

Table 8.1: Expression on gradients on objective function and constraints

	$J_0 = \min tf$	$J_1 = \min x_1(t_f) - 300$	$J_2 = \min x_2(t_f)$
u_1	$J_u(t_0) - J_u(t_1)$	$J_u(t_0) - J_u(t_1)$	$J_u(t_0) - J_u(t_1)$
u_2	$J_u(t_1)$	$J_u(t_1)$	$J_u(t_1)$
Δt_1	$\lambda_2(t_1)(u_1 - u_2) + \frac{\partial J_0}{\partial t_2}$	$\lambda_2(t_1)(u_1 - u_2) + \frac{\partial J_1}{\partial t_2}$	$\lambda_2(t_1)(u_1 - u_2) + \frac{\partial J_2}{\partial t_2}$
Δt_2	$\frac{\partial J_0}{\partial t_2} = \lambda_2(t_f) = 1$	$\frac{\partial J_1}{\partial t_2} = \lambda_2(t_f) = x_2(t_f)$	$\frac{\partial J_2}{\partial t_2} = \lambda_2(t_f) = u(t_f)$
	$\frac{\partial J_0}{\partial \Delta t_1} = \frac{\partial J_0}{\partial t_1} + \frac{\partial J_0}{\partial t_2}$ $\frac{\partial J_0}{\partial \Delta t_2} = \frac{\partial J_0}{\partial t_2}$	$\frac{\partial J_1}{\partial \Delta t_1} = \frac{\partial J_1}{\partial t_1} + \frac{\partial J_1}{\partial t_2}$ $\frac{\partial J_1}{\partial \Delta t_2} = \frac{\partial J_1}{\partial t_2}$	$\frac{\partial J_2}{\partial \Delta t_1} = \frac{\partial J_2}{\partial t_1} + \frac{\partial J_2}{\partial t_2}$ $\frac{\partial J_2}{\partial \Delta t_2} = \frac{\partial J_2}{\partial t_2}$

8.3 Procedure for OC

In this section we explain on example the application of orthogonal collocation for solving optimization problems. We consider $K = 2$ and $NE = 2$. The following equations we obtain

using equations (4.16) and (4.17) for the intervals.

Interval 1:

$$\phi_0(t) = \frac{t - t_{11}}{t_{10} - t_{11}} \frac{t - t_{12}}{t_{10} - t_{12}} \tag{8.13}$$

$$\phi_1(t) = \frac{t - t_{10}}{t_{11} - t_{10}} \frac{t - t_{12}}{t_{11} - t_{12}} \tag{8.14}$$

$$\phi_2(t) = \frac{t - t_{10}}{t_{12} - t_{10}} \frac{t - t_{11}}{t_{12} - t_{11}} \tag{8.15}$$

$$\theta_1(t) = \frac{t - t_{12}}{t_{11} - t_{12}} \tag{8.16}$$

Interval 2:

$$\phi_0(t) = \frac{t - t_{21}}{t_{20} - t_{21}} \frac{t - t_{22}}{t_{20} - t_{22}} \tag{8.17}$$

$$\phi_1(t) = \frac{t - t_{20}}{t_{21} - t_{20}} \frac{t - t_{22}}{t_{21} - t_{22}} \tag{8.18}$$

$$\phi_2(t) = \frac{t - t_{20}}{t_{22} - t_{20}} \frac{t - t_{21}}{t_{22} - t_{21}} \tag{8.19}$$

$$\theta_1(t) = \frac{t - t_{22}}{t_{21} - t_{22}} \tag{8.20}$$

The distribution of collocation points Fig.(8.3) we obtain as the roots of Legendre polynomials. According to equations (4.16) and (4.17) we obtained equations.

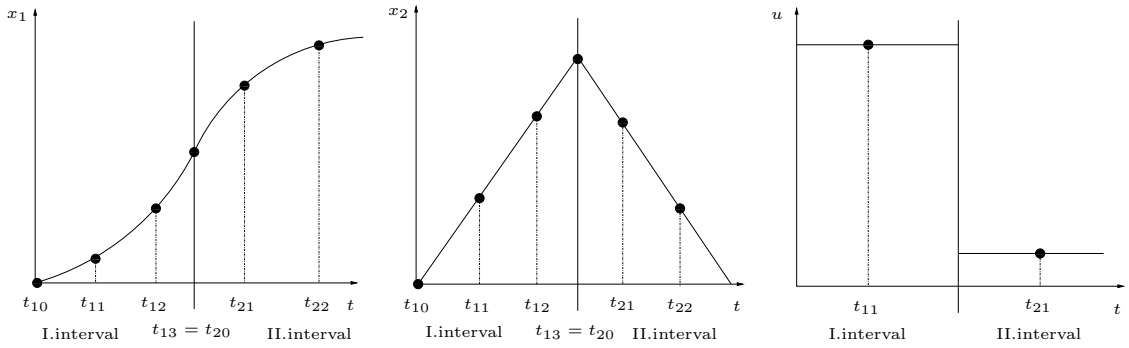


Figure 8.3: Distribution of colocation points

Example 1. interval.

$$[x_{10}^1\dot{\phi}_0(t_{10}) + x_{11}^1\dot{\phi}_1(t_{10}) + x_{12}^1\dot{\phi}_2(t_{10})] - [x_{10}^2\phi_0(t_{10}) + x_{11}^2\phi_1(t_{10}) + x_{12}^2\phi_2(t_{10})] = 0 \tag{8.21}$$

$$[x_{10}^1\dot{\phi}_0(t_{11}) + x_{11}^1\dot{\phi}_1(t_{11}) + x_{12}^1\dot{\phi}_2(t_{11})] - [x_{10}^2\phi_0(t_{11}) + x_{11}^2\phi_1(t_{11}) + x_{12}^2\phi_2(t_{11})] = 0 \tag{8.22}$$

$$[x_{10}^1\dot{\phi}_0(t_{12}) + x_{11}^1\dot{\phi}_1(t_{12}) + x_{12}^1\dot{\phi}_2(t_{12})] - [x_{10}^2\phi_0(t_{12}) + x_{11}^2\phi_1(t_{12}) + x_{12}^2\phi_2(t_{12})] = 0 \tag{8.23}$$

Further we rewrite the equations (8.1b) to the following form

$$[x_{10}^2\dot{\phi}_0(t_{10}) + x_{11}^2\dot{\phi}_1(t_{10}) + x_{12}^2\dot{\phi}_2(t_{10})] - [u_1^1\theta_1(t_{11})] = 0 \tag{8.24}$$

$$[x_{10}^2\dot{\phi}_0(t_{11}) + x_{11}^2\dot{\phi}_1(t_{11}) + x_{12}^2\dot{\phi}_2(t_{11})] - [u_1^1\theta_1(t_{11})] = 0 \tag{8.25}$$

$$[x_{10}^2\dot{\phi}_0(t_{12}) + x_{11}^2\dot{\phi}_1(t_{12}) + x_{12}^2\dot{\phi}_2(t_{12})] - [u_1^1\theta_1(t_{11})] = 0 \tag{8.26}$$

8.4 Results and Discussion

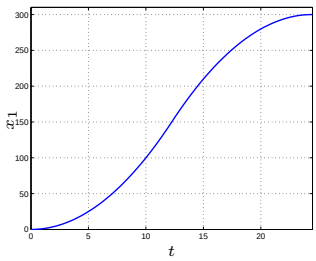
In this chapter we compare two numerical methods: CVP and OC. For calculating the optimal control trajectory and value of time intervals we used Matlab integrated NLP-solver $fmincon$. In table Tab.(8.2) are obtained results. For computing gradients in CVP method we used finite differences method (FD) and adjoint variables (AV). It can be seen Tab.(8.2) that by using adjoint variables for computing gradients the number of iteration increased and the computational time decreased. We can notice by comparing CVP and OC with gradients computed by finite differences method, that CVP method required more computational time but less iteration to reach optimum than OC. Also we notice, that by comparing CVP method using gradients computed by finite differences and adjoint variables, that the results for control variables and time intervals are the same. The CVP method with gradients computed by finite differences method required less iterations but more computational time then CVP with gradients computed with adjoint variables. The values of objective function, time intervals and control variables are the same in all three cases. Where #it represents the number of

Table 8.2: Comparison of two numerical methods

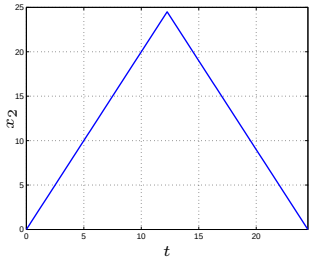
	CVP FD	CVP AV	OC
u^U	2	2	2
u^L	-2	-2	-2
Δt_1	12.2474	12.2474	12.2474
Δt_2	12.2474	12.2474	12.2474
t_f[s]	24.4949	24.4949	24.4949
#it	5	8	8
CPU [s]	5.2595	3.9550	0.5183

NLP iterations, CPU represents the computational time, t_f is the value of objective function, Δt_1 and Δt_2 are the values of delta intervals and u^U,u^L are the maximum and minimum values of control variables.

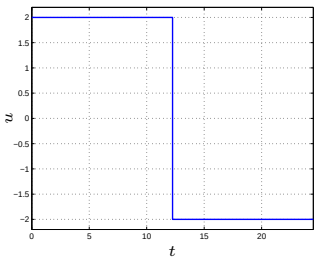
In pictures Fig.(8.4) we can notice that we used two time intervals. On the first we accelerate and on the second we decelerate. We can notice in Fig.(8.4(a)) and Fig.(8.4(d)) that we meet the constraints which were characterized by the final conditions eq.(8.2). We can also notice the switching time Fig.(8.4(b)) and Fig.(8.4(c)). Switching time is when the car stops to accelerate and starts to slowing down. We also notice that the switching time is the same in both numerical methods.



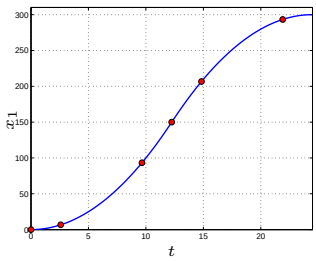
(a) CVP with adjoint variables



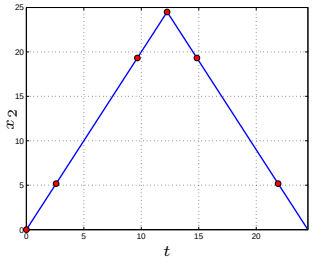
(b) CVP with adjoint variables



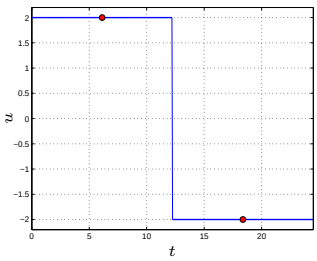
(c) CVP with adjoint variables



(d) OC with finite-differences



(e) OC with finite-differences



(f) OC with finite-differences

Figure 8.4: Comparison of CVP and OC numerical method

Chapter 9

Emulsion Polymerization Process

Emulsion polymerization process is used to produce polymers. For example latex is often product of emulsion polymerization. The final properties of polymers are highly related to the reaction time, molecular characteristics and reactor operating conditions. Reaction time is influenced by operating conditions. Advantages of emulsion polymerization include high molecular weight polymers that can be made by fast polymerization rate. But there are also

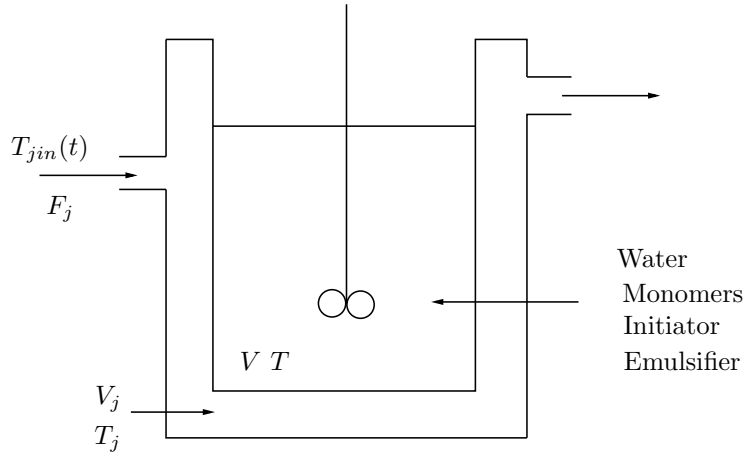


Figure 9.1: Schematic representation of the batch polymerization reactor

disadvantages e.g. the removal of water to dry the polymers is energy intensive process.The reaction of styrene and α–methylstyrene takes place in batch emulsion copolymerization reactor Fig.(9.1). We assume that the contents of the reactor and the jacket are perfectly mixed. The control variable represents the cooling fluid inlet temperature (T_{jin}).

9.1 Process Model

The process can be divided into several parts. Kinetic model, molecular weight distribution and reactor temperature dynamic model. This model was originally described by Castellanos (Castellanos 1996).

9.1.1 Kinetic Mechanism

For emulsion polymerization process we need 4 components. Dispersion medium, monomer, initiator and emulsifier. For dispersion medium we can use water. Monomer is slightly soluble in dispersion medium. Initiator has to be water soluble. Process of emulsion polymerization can be divided into three stages. Experimental validation was carried out by Castellanos (Castellanos 1996) and Gentric (Gentric 1997).

Three stages of the model:

I: The first phase is characterized by the production of free radicals by initiator decomposition. Free radicals are captured by the micelles. The termination of the first stage accured with disapearing of all micelles.

II: This stage is characterized by particles growth. Their amount is constant during the second and the third stage. The particles are saturated with monomer. This stage terminate when all the monomer droplets have disappeared.

III: The monomer concentration in the particles decreases and the rate of polymerization is decreasing.

The kinetic mechanism can be written as follows

Initiator decomposition:	$\mathrm{A} \longrightarrow 2\mathrm{R}^{\bullet}$	$R_a = 2fk_dA$
Particle formation:	$\mathrm{R}^{\bullet} + \mathrm{m} \longrightarrow \mathrm{N}^{\bullet}$	$R_n = k_1 m R^{\bullet}$
Initiation:	$\mathrm{N} + \mathrm{R}^{\bullet} \longrightarrow \mathrm{N}^{\bullet}$	$R_i = k_2 N R^{\bullet}$
Termination:	$\mathrm{N}^{\bullet} + \mathrm{R}^{\bullet} \longrightarrow \mathrm{N}$	$R_t = k_2 N^{\bullet} R^{\bullet}$
Propagation:	$\mathrm{P}_j^{\bullet} + \mathrm{M} \longrightarrow \mathrm{P}_{j+1}^{\bullet}$	$R_p = k_p M_{\mathrm{p}} N^{\bullet}$
Transfer to monomer:	$\mathrm{P}_j^{\bullet} + \mathrm{M} \longrightarrow \mathrm{M}^{\bullet} + \mathrm{P}_j^{\bullet}$	$R_{trM} = k_{trM} M_{\mathrm{p}} N^{\bullet}$

9.1.2 Kinetic Model

This model is based on the experiment carried out at three different temperatures and four initial monomer compositions. In this work we consider only 10% in mass of α–methylstyrene.

In the work Gentric (Gentric 1997) they described that there was no composition change during the polymerization. So we can write the global propagation constant as follows

$$k_p = k'_p \exp(-a.f_{\mathrm{MS}}) \tag{9.1}$$

where a is a constant, f_{MS} is molar fraction of α–methylstyrene at the beginning of a reaction. The rate of particles can be written

$$\dot{N}_{\mathrm{p}} = k_{cm} m \frac{R_a N_A}{k_{cm} m + k_{cp} N_{\mathrm{p}}} \tag{9.2}$$

next we introduce a capturing efficiency of the particles with respect to the micelles

$$\varepsilon = \frac{k_{cp} S N_A}{k_{cm} m} \tag{9.3}$$

$$\dot{N}_{\mathrm{p}} = \frac{R_a N_A}{1 + (\varepsilon N_{\mathrm{p}} / S N_A)} \tag{9.4}$$

emulsifier molecules will be absorbed in monomolecular layers at the polymer particles surface

$$S = S_o - k_v (X M_o)^{2/3} N_{\mathrm{p}}^{1/3} \tag{9.5}$$

where

$$k_v = \left[\frac{36\pi M_{\mathrm{M}}^2}{\omega_{\mathrm{P}}^2 (a_s N_A)^3 \rho_{\mathrm{P}}^2} \right]^{1/3} \tag{9.6}$$

where X is the conversion rate and M_0 is the initial monomer concentration. Then we can write the rate of monomer consumption as follow

$$\dot{M} = -R_p = -k_p M_{\mathrm{p}} \frac{N_{\mathrm{p}}}{N_A} \bar{n} \tag{9.7}$$

The monomer concentration in the particles can be written

$$M_{\mathrm{p}} = M_{\mathrm{pc}} = \frac{(1 - X_c)\rho_{\mathrm{M}}}{[(1 - X_c) + X_c \rho_{\mathrm{M}} / \rho_{\mathrm{P}}]\, M_{\mathrm{M}}} \qquad X \leq X_c \tag{9.8a}$$

$$M_{\mathrm{p}} = \frac{(1 - X)\rho_{\mathrm{M}}}{[(1 - X) + X \rho_{\mathrm{M}} / \rho_{\mathrm{P}}]\, M_{\mathrm{M}}} \qquad X > X_c \tag{9.8b}$$

where M_{pc} critical monomer concentration and X_c is critical conversion rate.

9.1.3 Molecular Weight Distribution Model

We also have to obtain the necessary polymer properties. These properties are linked to polymer structure.(Paulen et al. 2010). The considered properties are for example global

macromolecule concentration and the moments of the polymerization degrees. The degrees of production of the moments of the molecular weight distribution can be written as follow:

$$\dot{Q}_0 = R_t + R_{trM} \tag{9.9a}$$

$$\dot{Q}_1 = L(R_t + R_{trM}) \tag{9.9b}$$

$$\dot{Q}_2 = 2L^2(R_t + R_{trM}) \tag{9.9c}$$

where

$$R_t = \frac{R_a \bar{n} N_\mathrm{p}}{N_\mathrm{p} + \frac{S}{\varepsilon}} \tag{9.10}$$

$$R_{trM} = k_{trM} M_\mathrm{p} \frac{N_\mathrm{p}}{N_A} \bar{n} \tag{9.11}$$

$$L = \frac{R_p}{R_t + R_{trM}} \tag{9.12}$$

Here variable L denotes kinetic chain length. Once these moments are known, the number-average molecular weight ($\bar{M_n}$) can be calculated according to:

$$\bar{M_n} = M_\mathrm{M} \frac{Q_1}{Q_0} \tag{9.13}$$

9.1.4 Heat Balance Equation

For the control of temperature inside the reactor we have to describe the temperature dynamic of the reactor. The heat inside the reactor is controlled by cooling fluid. The temperature dynamic of the reactor and the cooling jacket is described by the following equations (Salhi et al. 2004)

$$\dot{T} = -\frac{V \Delta H}{m_r C_p} R_p + \frac{UA}{m_r C_p}(T_j - T) \tag{9.14}$$

$$\dot{T_j} = \frac{F_j}{V_j}(T_{jin} - T_j) - \frac{UA}{\rho_j V_j C_{pj}}(T_j - T) \tag{9.15}$$

where V and Vj are the reactors contents and jacket volume, U is the heat-transfer coefficient, F_j is the flow rate of the cooling fluid, $m_r C_p$ is the reactors total heat capacity, ρ_j represents the density of the cooling fluid, C_{pj} is the heat capacity of the cooling fluid.

9.2 Problem Formulation

The objective of this work was to calculate the optimal control by which we reach the desire state at minimum time. The desire state is characterized by final conversion written as follow

$$X_f = 1 - \frac{M(t_f)}{M_0} \tag{9.16}$$

and final number average molecular weight is define as

$$M_{nf} = M_M \frac{M_0 - M(t_f)}{Q_0(t_f)} \tag{9.17}$$

The objective function can be written as

$$\min_{u(t)} \; t_f \tag{9.18}$$

where initial state vector for the first stage is defined as

$$(x_0)^T = [M_0, 0, 0, 343.15, 343.15] \tag{9.19}$$

State vector is represented by global monomer concentration, number of particles,three moments of molecular weight distribution, initial temperature inside the reactor and cooling jacket: $x^T = (M, N_p, Q_0, T_0, T_{j,0})$. The optimized control variable is the temperature of the inlet cooling fluid (T_{jin}).

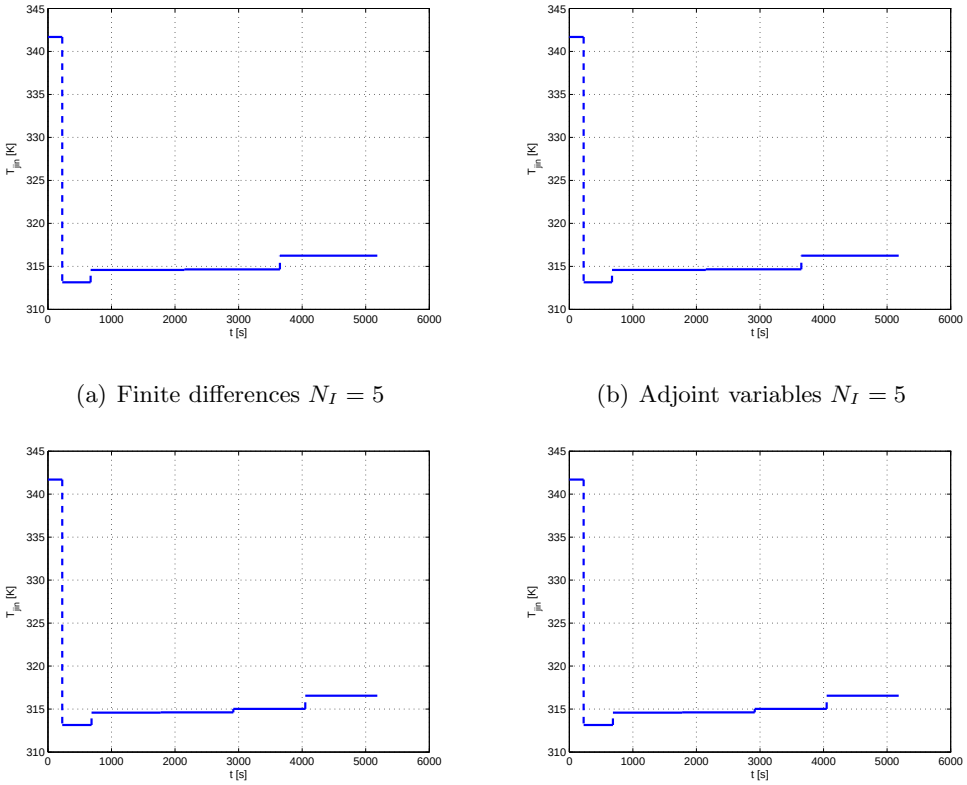
9.3 Results and Discussion

For solving the optimization problem we used NLP solver SNOPT (Sparse Nonlinear Optimizer). For integration we used the *ode*45 integrator which uses the 4th order Runge-Kutta method. Results are shown in Table 9.1 where we compare the computational time (CPU), number of NLP iterations (#it), absolute and relative integration error tolerance (Abs/RelTol) and the value of objective function (t_f) for two methods of computing gradients. The first method for computing gradients was the method of finite differences (FD) and the second method was the method of adjoint variables (AV). N_I stands for the number

Table 9.1: Comparison of computational aspects for different numbers of intervals of piecewise constant control and different methods for computing gradients.

	FD				AV			
N_I	CPU [s]	#it	Abs/RelTol	t_f [s]	CPU [s]	#it	Abs/RelTol	t_f [s]
1	1025.5	1	$1e^{-10}/1e^{-8}$	6064.9	362.3	9	$1e^{-10}/1e^{-8}$	6064.9
2	866.1	7	$1e^{-10}/1e^{-8}$	5192.6	329.5	7	$1e^{-10}/1e^{-8}$	5192.6
3	2778.5	16	$1e^{-10}/1e^{-8}$	5185.8	757.5	14	$1e^{-10}/1e^{-8}$	5185.8
4	3825.2	11	$1e^{-10}/1e^{-8}$	5183.0	640.8	9	$1e^{-10}/1e^{-8}$	5183.0
5	3220.8	10	$1e^{-10}/1e^{-8}$	5182.9	864.2	9	$1e^{-10}/1e^{-8}$	5182.9
6	4492.8	34	$1e^{-10}/1e^{-8}$	5182.6	1100.9	15	$1e^{-10}/1e^{-8}$	5182.6

of control intervals. For each method we used the same initial conditions and the same constraints X_f= 0.6 and $M_{nf} = 3 \times 10^6$ [g.mol^{-1}]. For finding the optimal control trajectories we fixed the time intervals which we calculated using the finite differences method we can notice that by increase of time intervals the value of objective function is decreasing. Then we compared the number of NLP iterations and computational time. We can notice in Table 9.1 that the computational time using adjoint variables is less than compare to method of finite differences. This behavior might occur because of the inaccurate gradients provided by method of finite differences. We can also notice that by the method of adjoint variables NLP solver required less iterations to converge. The graphical comparison of optimal control tra-



(a) Finite differences $N_I = 5$

(b) Adjoint variables $N_I = 5$

(c) Finite differences $N_I = 6$

(d) Adjoint variables $N_I = 6$

Figure 9.2: Comparison of control trajectories for different number of time intervals

jectories are shown in Fig.(9.2) for 5 and 6 time intervals. We can notice that by using finite differences Fig.(9.2(a)) and Fig.(9.2(c)) and adjoint variables Fig.(9.2(b)) and Fig.(9.2(d)) the control trajectories are very similar. This behavior was expected because of decreasing the degree of freedom by fixing the value of time intervals. The main differences are in the number of NLP iterations and computational time.

Chapter 10

Conclusions

In this work we studied the problem of dynamic optimization of processes. We considered two numerical methods, control vector parametrization and orthogonal collocation, for solving dynamic optimization problems. Both methods transforms the original infinite dimensional problem into finite dimensional problem of non-linear programming. As a part of the work, we studied several methods for computing gradients. In several examples we compare obtained results by using the mentioned numerical methods with three different methods for computing gradients. The first method was the method of finite differences. This method is very easy to implement but the gradients computed by this method can be inaccurate. We used finite differences method with the combination of adjoint variables to verify the correctness of the computed gradients. Adjoint variables are mainly used when we have a large number of optimized variables and only few contraints. Its because every constraint generate a system of equations which we have to integrate. Sensitivity equations are on the other hand very effective by large number of constraints but their disadvantage is when we have large number of optimized variables, because every optimized variable generate system of equations which have to be integrated. The results show that the most effective method was control vector parametrization with gradients computed by adjoint variables. The further work can be devoted to study the sensitivity equations of second order which give more precise Hessian for NLP problem.

Chapter 11

Resumé

Úvod

Hľadanie optimálneho riešenia je každodenným problémom každého z nás. Na riešenie optimalizačných problémov, ktoré riešime v bežnom živote môžeme použiť viacero metód, ktoré nám uľahčujú hľadanie optimálneho riešenia. Na vyriešenie optimalizačného problému musíme mať v prvom rade dobre zadefinovaný problém. Avšak pri riešení optimalizačných úloh nemôžeme zabúdať na obmedzenia, ktoré je nutné rešpektovať. Sú to väčšinou technologické obmedzenia napríklad maximálna rýchlosť auta a maximálny prietok v potrubí. Ak sa hovorí o optimálnosti resp. optimálnom riešení, tak sa jedná o minimalizáciu alebo maximalizáciu účelovej funckie. Napríklad minimalizácia času, energie alebo nákladov a maximalizácia produkcie alebo zisku. Táto práca sa hlavne zaoberá aplikáciou metód na riešenie optimalizačných úloh. Tieto metódy sa rozdeľujú na dve skupiny a to analytické a numerické metódy. Práca sa hlavne zaoberá numerickými metódami ako sú napríklad parametrizácia vektora riadenia a ortogonálna kolokácia.

Dynamická Optimalizácia

Dynamická optimilizácia pojednáva o optimálnom riadení pri otvorenej slučke. Ako sme už spomínali na riešenie optimalizačného problému musíme mať dobre zadefinovaný problém ktorý pozostáva z nasledovných častí

- Matematický opis systému (2.1)
- Definícia obmedzení v tvare rovnosti (2.5) a nerovnosti (2.6)
- Definícia minimalizačného kritéria

Minimalizačné kritérium môžeme zapísať v troch nasledujúcich tvarcoh

- Bolzov tvar (2.2)
- Lagrangerov tvar (2.3)
- Mayerov tvar (2.4)

Našou úlohou je nájsť také riadenie, ktoré minimalizuje účelovú funkciu.

Metódy Dynamickej Optimalizácie

V tejto sekcii rozoberáme dve hlavné metódy: parametrizáciu vektora riadenia a ortogonálnu kolokáciu. Parametrizácia vektora riadenia je založená na aproximácií (diskretizácií) pôvodnej spojitej trajektorií riadenia obr.(4.2) konečným počtom riadiacich úsekov obr.(4.3). V tejto práci uvažujeme konštantné riadenie na jednotlivých časových úsekoch. Pri metóde ortogonálnej kolokácie sa pôvodne dynamický optimalizačný problém prevedie na statický vďaka aproximácií časových trajektorií stavov a riadenia ich polynomickými aproximáciami. Uvažované aproximácie je vhodné vytvárať s použitím Lagrangeových polynómov podľa rovníc (4.16) a (4.17). Aby sa zabezpečili ortogonálne vlastnosti takýchto polynómov je možné ich konštruovať napríklad na základe koreňov Legendrovych polynómov.

Metódy Výpočtu Gradientov

V rámci tejto práce sme sa zaoberali metódami výpočtu gradientov. Rozoberajú sa tri hlavné metódy počítania gradientov:

- metóda konečných rozdielov
- adjungované premenné
- citlivostné rovnice

Metóda konečných rozdielov je spomedzi troch spomínaných metód najmenej presná avšak najjednoduchšie aplikovateľná. Jej hlavnou nevýhodou je množstvo integrácií ktoré je treba vykonať. Metóda konečných rozdielov sa hlavne používa v kombinácií s inou metódou počítania gradientov na overenie správnosti výpočtu gradientov. Druhou metódou počítania gradientov sú adjungované premenné. Táto metóda sa hlavne používa ak vrámci optimalizačného problému máme veľké množstvo optimalizačných premenných s malým množstvom

obmedzení. Je to preto, lebo každé obmedzenie generuje systém rovníc ktoré treba integrovať. Pri citlivostných rovniciach máme opačný prípad ako pri adjungovnaých premenných. Citlivostné rovnice sa používajú pri malom množstve optimalizačných premenných a veľkom množstve obmedzení. Pretože každá optimalizovaná premenná generuje systém rovníc ktoré treba integrovať.

Príklady

V rámci diplomovej práci sme rozoberali viacero optimalizačných príkladov. V rámci jednotlivých príkladov sme porovnávali výsledky, ktoré sme získali použitím viacero metód s rôznym prístupom počítania gradientov.

Prvý príklad bol fermentácia kyseliny mliečnej. Tento príklad bol zameraný na odhad parametrov, kde sme použili citlivostné rovnice na počítanie gradientov. Uvažovali sme fermentačný proces na produkciu kyseliny mliečnej použitím sacharózy ako substrátu. Počas fermentačného procesu sa sacharóza premení na biomasu a kyselinu mliečnu. Väčšina odhadnutých paramterov (6.11) sa nachádza na hornom a dolnom ohraničení. Lepšie výsledky by sa dali získať pri uvažovaní väčšieho dovoleného priestoru.

V druhom príklade sa zaoberáme optimálnym riadením rurkového chemického reaktora. Porovnávali sme výsledky zísakne pomocou parametrizácie vektora riadenia a ortogonálnej kolokácie. Pri parametrizácií vektora riadenia sme použili dva spôsoby počítania gradientov a to metódu konečných rozdielov a adjungované premenné. Príklad sme mali rozdelený na dve časti. V prvej časti sme uvažovali, že dĺžky časových intervalov sú optimalizované premenné a úlohou bolo vypočítať optimálne riadenie chemického reaktora. V druhej časti sme časové intervaly zafixovali a opäť sme mali za úlohu vypočítať optimálne riadenie chemického reaktora. Na základe výsledkov, ktoré sú zhrnuté v tabuľke tab.(7.1) si môžeme všimnúť, že najväčšiu hodnotu účelovej funkcie sme dosiahli použitím metódy ortogonálnej kolokácie.

V tret'om príklade sme mali za úlohu vypočítať optimálne riadenie auta aby prešlo zadanú dráhu za najkratší čas. Opäť sme použili paramterizáciu vektora riadenia a ortogonálnu kolokáciu. V rámci príkladu sme vysvetlili podrobný postup na počítanie gradientov pomocou adjungovaných premenných. Výsledky zhrnuté v tabuľke tab.(8.2) ukazujú, že najkratší výpočtový čas potrebovala metóda ortogonálnej kolokácie.

Posledný príklad bol zameraný na emulzný polymerizačný proces na výrobu polymérov. Latex je jeden z najznámejších produktov emulzného polymerizačného procesu. Výhodou polymerizácie je napríklad veľká molekulová hmotnosť polymérov, ktoré je možné vytvoriť pri rýchlej polymerizácií. Nevýhodou je napríklad energetická náročnosť pri sušení polymérov. Celá reakcia styrénu a α–metylstrénu prebieha v sádzkovom polymerizačnom reaktore. V

príklade sme uvažovali zafixované časové intervaly a úlohou bolo vypočítať optimálne riadenie reaktora pri ktorom dosiahneme požadovaný stav za minimálny čas. Na riešenie príkladu sme použili metódu parametrizácie vektora riadenia s dvoma spôsobmi počítania gradientov. Prvá bola metóda konečných rozdielov a druhá bola metóda adjungovaných premenných. Výsledky zobrazené v tabuľke tab.(9.1) ukazujú, že metóda adjungovaných premenných potrebovala menej výpočtového času ako metóda konečných rozdielov.

Záver

Diplomová práca bola zameraná na riešenie optimalizačných úloh. V rámci práce sme rozobrali viacero metód ako sú napríklad parametrizácia vektora riadenia a ortogonálna kolokácia. Ďalej sme rozoberali viacero spôsobov na počítanie gradientov. Aplikáciu spomínaných metód sme ukázali vo viacero optimalizačných príkladov. Výsledky príkladov sú podrobne spracované v rámci každého príkladu.

Bibliography

R. Bellman. *Dynamic Programming*. Princeton University Press, 1957.

M. Caracotsios and W. E. Stewart. Sensitivity analysis of initial value problems with mixed ODEs and algebraic equations. *Computers and Chemical Engineering*, 9:359–365, 1985.

J. R. Castellanos. *Contribution á la modélisation du procédé do copolymérisation en émulsion de l'α-methylstyréne et du styréne*. PhD thesis, INPL, Nancy, France, 1996.

J. E. Cuthrell and L. T. Biegler. On the optimization of differential-algebraic process systems. *AIChE Journal*, 33(8):1257–1270, 1987.

M. Fikar. *Dynamická optimalizácia procesov*. STU Bratislava, Slovenská e-akadémia, n. o., 2007.

C. Gentric. *Optimisation Dynamique et Commande non Lineaire d'un Reacteur de Polymerisation en Emulsion*. PhD thesis, INPL, Nancy, France, 1997.

T. Hirmajer. *Dynamická optimalizácia procesov s hybridnou dynamikou*. PhD thesis, Oddelenie informatizácie a riadenia procesov, FCHPT STU, Radlinského 9, 812 37 Bratislava, Slovenská republika, 10.1.2007 2007.

D. G. Hull. *Optimal Control Theory for Applications*. Mechanical Engineering Series. Springer-Verlag New York, 2003.

D. E. Kirk. *Optimal Control Theory: An Introduction*. Prentice-Hall, London, 1970.

Z. Kovács. Parameter estimation problems in bioprocess applications, 2010.

T. Lauw-Bieng and L. T. Biegler. Simultaneous solution and optimization strategies for parameter estimation of differential-algebraic equation systems. *Ind. Eng. Chem. Res.*, 30: 376–385, 1991.

R. Paulen. Global dynamic optimization of processes. *Minithesis*, 2010.

R. Paulen, M. Fikar, and M. A. Latifi. Dynamic optimization of a hybrid system: Emulsion polymerization reaction. *Journal of Cybernetics and Informatics*, 9, 2010.

L. S. Pontryagin. *The Mathematical Theory of Optimal Processes*. Pergamon Press, New York, 1964.

D. Salhi, M. Daroux, C. Gentric, J. P. Corriou, F. Pla, and M. A. Latifi. Optimal temperature-time programming in a batch copolymerization reactor. *Ind. Eng. Chem. Res.*, 43:7392–7400, 2004.

K. L. Teo, C. J. Goh, and K. H. Wong. *A Unified Computational Approach to Optimal Control Problems*. John Wiley and Sons, Inc., New York, 1991.

M. Čižniar. Dynamic optimisation of processes. Master's thesis, Faculty of Chemical and Food Technology, Radlinskeho 9, 812 37 Bratislava, Slovak Republic, 2005.

M. Čižniar, M. Fikar, and M. A. Latifi. Matlab dynamic optimisation code dynopt. user's guide. Technical report, KIRP FCHPT STU Bratislava, Slovak Republic, 2005.