Model Reduction of Large Scale Gas Distribution Networks

By

Muhammad Junaid Nasir

NUST201463262MRCMS64014F

Supervisor

Dr. Mian Ilyas Ahmad

Department of Computational Engineering

A thesis submitted in partial fulfillment of the requirements for the degree of Masters of Science in Computational Science and Engineering (MS CS&E)

In

Research Center for Modeling and Simulation,

National University of Sciences and Technology (NUST),
Islamabad, Pakistan.
(July 2018)

Dedicated To

My Parents,
Sisters
&
Hadiqa, Bisma

Certificate of Originality

I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any degree or diploma at RCMS, NUST or at any other educational institute, except where due acknowledgement has been made in the thesis. Any contribution made to the research by others, with whom I have worked at RCMS, NUST or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except for the assistance from others in the project's design and conception or in style, presentation and linguistics which has been acknowledged.

Author Name: **Muhammad Junaid Nasir**

Signature: ______________________________

Acknowledgement

My thanks goes first to Almighty Allah, Who gives power to the weak and helps the helpless, The most Gracious and Merciful.

I would like to thank my supervisor, Assistant Professor at Research Center For Modeling and Simulation(RCMS), Dr. Mian Ilyas Ahmad, for his many valuable suggestions, his support, guidance and encouragement throughout my thesis and research work.

I also wish to express my appreciation to the principal of RCMS, National University of Sciences and Technology, Islamabad, Dr. Rizwan Riaz for his valuable support and efforts for the students of his department.

I am also thankful to all the faculty members at RCMS, especially my GEC members Engr Sikandar Hayat Mirza, Dr. Ammar Mushtaq and Dr. Salma Sherbaz for their kind attention and encouragement.

My thanks also go to all of my friends. I cannot name all of them but some, Dr. Nadeem Umar, Usman Sadiq, Farooq Sanawan, Bilal Yasin , Bilal Asad, Zia Ullah, Hafiz Suliman Munawar, Mustafa Kamal, Syed Khizar Abbas, Hamza Ahmad, Hafiz Umar, Asad Hayat, Junaid Akram, Sanwel Zeb, Syed Fasih Gardezi and all others, for making my time memorable.

Last but not the least, my gratitude goes to my family: my parents and to my sisters for continuous encouragement, prayers and unshakable believe. Their spiritual and moral support throughout the thesis phase will truly be remembered in my life.

Muhammad Junaid Nasir

Contents

1 Introduction **1**

1.1 Mathematical Formulation 3

1.2 Problem Statement 3

1.3 Motivation 4

1.4 Objectives 4

1.5 Outline 5

2 Reduction Technique **6**

2.1 Model order reduction 6

2.2 Proper orthogonal decomposition 7

2.3 Discrete empirical interpolation method 8

2.4 Discretization of flow problems 10

3 Model Reduction of DAEs. **12**

3.1 Modeling of gas distribution networks 12

3.1.1 Graph Theory in Gas Distribution Networks 12

3.1.2 The Isothermal Euler Equations 13

3.2 Quasi-Static Model for Gas Network 14

3.3 Dynamic Model for Gas Network 15

3.4 Example Gas Network Model 17

3.5 Indirect POD Method for DAEs 21

3.5.1 Index-1 Linear DAE's 21
3.5.2 Index-1 Bilinear DAE's 23

4 Numerical Results and Discussions 25

4.1 Nonlinear RC-Circuit (Nonlinear ODEs) 25
4.1.1 Implementation of POD 27
4.1.2 Implementation of POD-DEIM 27
4.2 Burger's Equation (Quadratic Bilinear ODEs) 28
4.3 Random Example 32
4.3.1 Linear DAE 32
4.3.2 Bilinear DAE 34

5 Conclusion and Future Work 37

Bibliography 37

Appendix A – MATLAB Codes for Model Order Reduction 42

List of Abbreviations

ODE	Ordinary differential equation
DAE	Differential algebraic equation
PDE	Partial differential equation
POD	Proper orthogonal decomposition
DEIM	Discrete empirical interpolation method
MOR	Model order reduction
SVD	Singlar value decomposition
RC	Resistor capacitor
LTI	Linear time invariant
LTV	Linear time variant
SISO	Single input, single output

List of Tables

4.1 Simualtion Time for Original System, POD and POD-DEIM Reduced System 28

List of Figures

1.1 Block Diagram . . . 2

3.1 Single Pipe . . . 16

3.2 Gas Network Model [17] . . . 18

3.3 Singular Values . . . 20

4.1 Large Scale RC Circuit [7] . . . 25

4.2 Comparison of simulations of the original and POD reduced-order systems for an input $u(t) = e^{-10t}$ . . . 27

4.3 Comparison of simulations of the original and POD-DEIM reduced-order systems for an input $u(t) = e^{-10t}$ . . . 28

4.4 QBDAES . . . 29

4.5 Original and Reduced Solution of burger equation for Input $(u(t) = \cos(\pi t))$ . . . 31

4.6 Original and Reduced Solution of burger equation for Input$(u(t) = 2\sin(\pi t))$ . . . 32

4.7 Direct reduction approach of linear system for r=10 . . . 33

4.8 Indirect reduction approach of linear system for r=10 . . . 33

4.9 Direct reduction approach of linear system for r=20 . . . 34

4.10 Indirect reduction approach of linear system for r=20 . . . 34

4.11 Direct reduction approach of bilinear system for r=10 . . . 35

4.12 Indirect reduction approach of bilinear system for r=10 35
4.13 Direct reduction approach of bilinear system for r=20 36
4.14 Indirect reduction approach of bilinear system for r=20 36

Listings

5.1 POD Code 42

5.2 POD-DEIM Code 42

Abstract

We consider the problem of modeling and simulation of large-scale gas distribution network. In general, a gas distribution network is described by the pressure at the nodes and flow through branches of the network. There are different elements in the gas network that include pipes, valves, resistors, compressors, preheaters and coolers. The flow through these elements can be mathematically modeled by differential as well as algebraic equations. The complete model of the network becomes a system of nonlinear Differential Algebraic Equations (DAEs) also called descriptor systems. Often these systems are represented by large scale models in order to get more and accurate details of the system. Simulation of such large scale models is computationally expensive and prohibitive. An alternate option is to reduce the model mathematically such that the response of reduced and actual model is almost comparable. The reduced model is then used for simulation or control instead of the original large-scale model. In this thesis our focus is on model reduction of nonlinear DAEs. Existing model reduction techniques are not directly applicable to nonlinear DAEs as they are unable to retain the structure of DAEs. This may result in unbounded approximation error. We proposed a new model reduction framework for some special linear and nonlinear DAEs that ensure the structure of original system. For our numerical results we used Proper Orthogonal Decomposition (POD) in the existing framework and in the proposed settings to compare the results. It is observed that the proposed method gives 10% to 15% better relative approximation error as compared to the direct use of standard reduction method and also retain the original structure of the model representing the gas distribution network.

Chapter 1

Introduction

Modeling and simulation is a large and diverse discipline used for providing solutions to complex problems encountered in almost every field of science and technology. Modeling is the mathematical representation of the system and simulation is the solution obtained through the model. Often large scale models are used for system representation in order to get more and accurate details of the system. Simulation of such large scale models is computationally expensive and prohibitive. An alternate option is to reduce the model mathematically such that the response of the reduced and actual model is almost comparable. The reduced model is then used for simulation to predict the behavior of the actual system. The complete scenario in the form of block diagram is shown for the application of gas distribution network in figure 1.1.

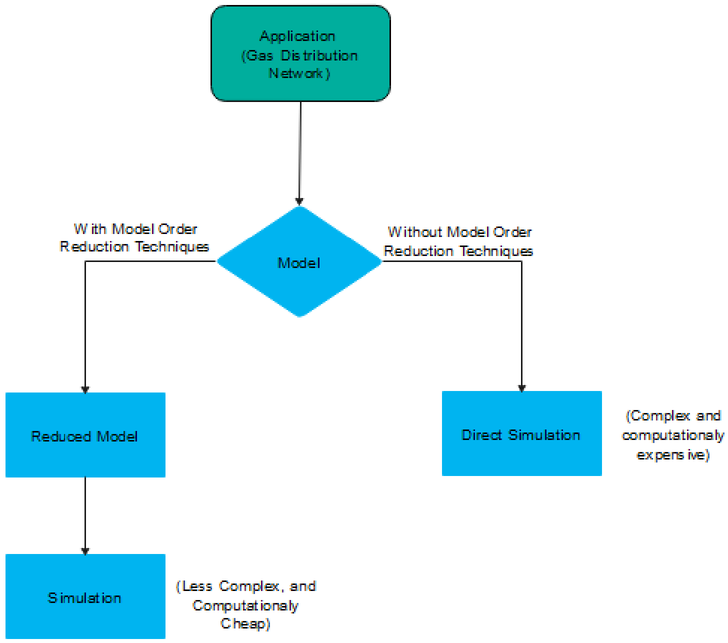


Figure 1.1: Block Diagram

In this thesis our focus is to observe the modeling and simulation of gas distribution network. Gas network simulation becomes challenging as the size of network increases. An important part of gas network is the gas pipe used for transportaion. The dynamics of flow through the pipe can be modeled by using Isothermal euler equations as discussed in [4], [1], [6]. Flow through pipe is nonlinear and its representation involves differential equations that are obtained after discretization. Another component is gas valve that can be modeled by a simple switch with two states, on and off, as expressed in [2]. Its representation is an algebraic equation. Similarly there are other components and the overall gas distribution network is in the form of differential algebraic equations (DAEs).

1.1 Mathematical Formulation

The complete model of the gas distribution network involves nonlinear differential as well as algebraic equations. In this section, we will discuss state space modeling for nonlinear DAEs and the problem of model reduction for such systems.

$$\text{(Nonlinear State Space Model)} : \begin{cases} E\dot{x}(t) = f(x(t)) + bu(t) \\ y(t) = c^T x(t) \end{cases} \tag{1.1}$$

Where $f : \mathbb{R}^n \to \mathbb{R}^n$ is nonlinear state evolution function and $b, c \in \mathbb{R}^n$ denote the input and output vector, respectively. $x(t), u(t), y(t) \in \mathbb{R}^n$ are called the state, input and output of the system, respectively. In case of linear system, $f(x(t))$ can be replaced with $Ax(t)$, where $A \in \mathbb{R}^{n\times n}$. While in case of quadratic system, $f(x(t))$ can be replaced with $Ax(t) + H(x \otimes x)$, where $H \in \mathbb{R}^{n\times n^2}$ and $(x \otimes x) \in \mathbb{R}^{n^2\times 1}$. If n is of large size, it can be reduced to form a reduced nonlinear state space model as given below

$$\text{(Reduced Nonlinear State Space Model)} : \begin{cases} \dot{\tilde{x}}(t) = \tilde{f}(\tilde{x}(t)) + \tilde{b}u(t) \\ \tilde{y} = \tilde{c}^T \tilde{x}(t) \end{cases} \tag{1.2}$$

With $\tilde{f} : \mathbb{R}^r \to \mathbb{R}^r, \tilde{b}, \tilde{c} \in \mathbb{R}^r$ and $r \ll n$.

1.2 Problem Statement

Mathematical models for a gas distribution network are in general large and in the form of nonlinear DAEs. Simulation of large scale non-linear DAEs is complex. To resolve this issue, model order reduction is often used. Model order reduction of nonlinear Ordinary Differential Equations (ODEs) is well known, and different techniques have been proposed in the literature [15]. However, for reduction of

DAEs, there are still many open problems and it requires further research. Since gas distribution networks are represented by nonlinear DAEs, the problem is how to reduce the nonlinear DAEs for modeling of gas distribution network, such that the behavior of original system and reduced system is approximately same.

1.3 Motivation

Model order reduction is an important computational tool for efficient simulation as well as control of different large scale dynamical systems. Efficient simulation of large scale systems improves computational cost and can identify real-time response of the system.

1.4 Objectives

- Model reduction technique (POD) will be used for reduction of nonlinear DAEs.
- Nonlinear DAEs can been transformed to an equivalent system that has a nonlinear ODE part and nonlinear polynomial part before using the reduction technique.
- The reduction technique will be used only on the ODE part to obtain a reduced nonlinear ODE and the polynomial part will be retained in the reduced system.
- This new approach of reduction for nonlinear DAEs will ensure the structure of DAEs in the reduced system.
- Comparison of indirect approach which involves split of DAEs with the direct approach.

1.5 Outline

The remainder of the thesis is organized as follows: In Chapter 2 we discuss the discretization of Partial Differential Equations (PDEs) and POD techniques for model reduction of some standard systems. In Chapter 3, first we have discussed our gas distribution network application along with its mathematical modeling, secondly we have discussed the model reduction of nonlinear DAEs by indirect method. Chapter 4 is about numerical results, where we implemented POD and POD-DEIM on non-linear RC-circuit, POD on burger's equation and POD on structured bilinear descriptor syetem. Chapter 5 draws conclusion and tells about future works.

Chapter 2

Reduction Technique

This chapter is organized in the following manners:We have discussed about the nonlinear systems and model reduction techniques (POD and DEIM) for reducing those nonlinear systems. Secondly we have discussed about discretization of flow problems from PDEs to ODEs.

2.1 Model order reduction

There are different model reduction techniques. Here we will discuss a few of them. Main purpose behind the model order reduction techniques is to obtain computationally efficient simulation of large-scale system.

Many methods for model reduction of linear system are available in literature such as moment-matching method [28], balanced truncation [27] and iterative rational Krylov method [29]. However many system applications like gas distribution network, electrical circuits and water networks have nonlinearities and are described by nonlinear differential algebraic equations (DAEs). There are not many methods for model reduction of such nonlinear systems. However for nonlinear ODEs proper orthogonal decomposition (POD) [8] is well used and is extensively discussed in the next section.

2.2 Proper orthogonal decomposition

POD is a well known method of model reduction. This technique uses empirical data and is used for computing the optimal approximating subspace for the data. In this technique reduced model can be obtained by projecting the dynamics of the original model to that subspace . This is computationally efficient model reduction technique and is used to reduce nonlinear systems [30, 31].

An algorithm for POD along with explanation of each step is given below .

Algorithm 1: Proper Orthogonal Decomposition

1 **INPUT:** Original nonlinear system

2 **OUTPUT:** Reduced System

1: Solve the original non-linear system at some time samples to get the snapshots.

$$X = (x(t_1), x(t_2), ...x(t_m))$$

Where $x(t) \in \mathbb{R}^{n\times 1}$ which is a state space vector computed for m time intervals resulting in a matrix of snapshots $X \in \mathbb{R}^{n\times m}$

2: Get the POD vectors of rank r from Singular value decomposition (SVD) of X.

$$X = U_r S_r V_r^T$$

SVD will be done for X, that will generate $U \in \mathbb{R}^{n\times n}$, $S \in \mathbb{R}^{n\times m}$ and $V \in \mathbb{R}^{m\times m}$. Since we require POD vectors of rank r from the SVD of X, so U, S and V will be transformed to $U_r \in \mathbb{R}^{n\times r}$, $S_r \in \mathbb{R}^{r\times r}$ and $V_r \in \mathbb{R}^{m\times r}$

3: Use projection basis $U_r = (\tilde{u_1}, \tilde{u_2}, ...\tilde{u_r})$ to get ROM.

$$U_r^T E \frac{dz(t)}{dt} = U_r^T f(U_r z(t)) + U_r^T Bu(t)$$

State space vector $x(t)$ is transformed to low dimension state space vector $z(t) \in \mathbb{R}^{r\times 1}$, by projecting vector $z(t)$ onto the plane $U_r \in \mathbb{R}^{m\times r}$ gives us $x(t) = U_r z(t)$.

MATLAB code for this algorithm is given in Appendix A.

2.3 Discrete empirical interpolation method

Considering the algoritm for POD given in section 2.2, where the output is reduced but nonlinear function $f(U_q z(t))$ is not reduced. It still has to be computed on the original state space $\mathbb{R}^n$ and as a result simulation becomes time taking. To catter this issue, there exists different methods like missing point estimation (MPE)[23], emperical interpolation method (EIM) [24], best point interpolation method (BPIM) [26] and DEIM [10].

Here we will discuss DEIM, where nonlinear function $f(U_r z(t))$ is projected onto a subspace with dimensions $\mathbb{R}^r$ and $r \ll n$, this approximates the subspace spanned by the snapshots of the nonlinear function [10].

$U_d = [u_1, \cdots, u_m] \in \mathbb{R}^{n \times m}$ and $P = [e_{\varphi_1}, e_{\varphi_2}, ...e_{\varphi_l}]_n \in \mathbb{R}^{n \times m}$ with m interpolation indices $\{\varphi_1, \varphi_2, ...\varphi_m\}$, being the output of DEIM algorithm will be used for interpolation approximation of original function as follow.

$$P_{DEIM} = U_d(P^T U_d)^{-1} P^T$$

The problem is that how to compute the U_d and how to specify the indices φ_i. The solution to this problem requires the computation of a basis matrix U_f that is :

1. Collect the snapshots of $f(x(t))$ into a matrix $F = (f(x(t_1)), ..., f(x(t_m)))$
   Solving $f(x(t))$ for $[t_1, ...t_m]$ will give us a matrix of snapshots of $f(x((t))$.

2. Apply SVD to F : $F = U_f S (V_f)^T$

3. $U_f = (u_1^F, ...u_l^F)$ are projection basis of rank l.

DEIM approximation is uniquely determined by these projection basis U_f. Algorithm for DEIM is given below.

Algorithm 2: Discrete Empirical Interpolation Method

```
INPUT: POD Basis U_f = (u_1^F, u_2^F, ....u_l^F)
OUTPUT: φ = [φ_1, φ_2, ··· , φ_m] ∈ ℝ^m
[|ρ|, φ_1] = max[|u_1^F|]
U_d = [u_1^F], P = [e_{φ_1}] , φ⃗ = [φ_1]
for l = 2 to m do
    solve (P^T U_d)c = P^T u_l for c
    r = u_l − U_d c
    [|ρ|, φ_l] = max[|r|]
    U_d ← [U_d u_l^F], P ← [P e_{φ_l}], φ⃗ ← [φ⃗ ; φ_l]
end for
```

MATLAB code for DEIM algorithm is given in Appendix A.

For more details see [10].

2.4 Discretization of flow problems

Flow problems can be expressed in terms of PDEs. By applying discretization, we can change the PDEs to ODEs. There are many methods for discretization like forward difference method, backward difference method and central difference method. We will briefly discuss the forward difference method for discretization of 1st and 2nd partial derivative in space.

This means that if we have a PDE with 1st and 2nd derivative in terms of x and t, and we discretize x in $i = 1000$ points, we will have 1000 variables and two boundary values resulting in large scale ODE if we keep the derivative with respect to t. First partial derivative can be discretized using backward difference method as

$$(\frac{\partial u}{\partial x}) \approx \frac{u_{i+1} - u_i}{(\Delta x)} \tag{2.1}$$

Second partial derivative can be be discretized using backward difference as

$$(\frac{\partial^2 u}{\partial x^2}) \approx \frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta x)^2} \tag{2.2}$$

Here $u = u(x,t)$ and $u_i = u(i,t)$ & $u_{i-1} = u(i-1,t)$.

Now we shall convert heat equation from PDE to ODE by discretization.

$$\frac{\partial u}{\partial t} = c^2 \frac{\partial^2 u}{\partial x^2}$$

Initial boundary conditions are $u_0 = u(0,t) = f(x)$, where f(x) is input to the system.

Final boundary conditions are $u_n = u_{n+1}$. So heat equation can be discretized as follow.

$$\frac{du_i}{dt} = c^2 \frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta x)^2}$$

After applying boundary conditions the equation can be written as

$$\dot{u} = Au + Bf(x)$$

Here state vector $u \in \mathbb{R}^{n \times 1}$, with $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$ and input vector $f(x) \in \mathbb{R}^{m \times 1}$. Now this is an ODE obtained by discretization of heat equation which was in the form of PDE.

Chapter 3

Model Reduction of DAEs.

In this chapter we discussed the methodology for representation of the complete model of the gas distribution network as a nonlinear DAE and discussed an indirect POD method for its reduction.

3.1 Modeling of gas distribution networks

The main components of modeling and simulation of gas distribution network are given in following subsections.

3.1.1 Graph Theory in Gas Distribution Networks

A graph is formed by vertices and edges connecting the vertices [22]. So to represent the structure or topology of the gas distribution network, graph theory is commonly used where edges and vertices are connected. Edge of graph can be a pipe, regulator, compressor or any other component of gas network and vertices are nodes. Topology or structure can be modeled as a directed graph $G = (E, N)$ with nodes set N and edges set E. For M number of edges and N number of nodes, the corresponding incidence matrix $A \in \mathbb{R}^{N\times M}$ is defined as:

$$A_{ij} = \begin{cases} 1, & \text{If edge j leaves node i;} \\ -1, & \text{If edge j enter node i;} \\ 0, & \text{else.} \end{cases}$$

Here edge is denoted by ij, where the flow is directed from i to j. Nodes which are assigned to gas sources are supply nodes, while those which are assigned to gas users are demand nodes. Supply node is denoted by N_+ , demand node by N_- and interior nodes by N_o. So overall node N includes

$$N = N_+ + N_- + N_o$$

The valves and the hydraulic resistances can be described by algebraic equations, for details see [2].

3.1.2 The Isothermal Euler Equations

The dynamic behavior of the gas flow through the gas pipelines can be represented by the Isothermal Euler equations as discussed in [20] that is,

$$\partial_t \rho + \partial_x q = 0 \tag{3.1}$$

$$\partial_t q + \partial_x p + \partial_x(\rho v^2) + g\rho\partial_x h = \frac{-\lambda(q)}{2D}\rho v|v| \tag{3.2}$$

$$p = \gamma(T)z(p,T)\rho \tag{3.3}$$

Theses equations tells the transient behavior of the gas. (3.1) is continuity equation obtained by Conservation of mass, (3.2) is pressure loss equation yield by conservation of momentum. While the 3rd one (3.3) is the state of real gas.

The above system of PDEs have state variables gas velocity $v = v(x,t)$, gas density $\rho = \rho(x,t)$, gas pressure $p = p(x,t)$, pipe elevation $h = h(s)$ and gas temperature

$T = T(x,t)$. Here gas flow can be calculated from the density and velocity of the gas i.e,

$$q(x,t) = \rho(x,t)v(x,t).$$

Remaining components are: pipe diameter D, gravity constant g, gas state $\gamma(T)$ which is determined by temperature and gas constant R as $\gamma = RT$, friction factor $\lambda(q)$ and compressibility factor $z(p,T)$. Term $\gamma(T)z(p,T)$ in (3.3) is often approximated by square of sound velocity $a \approx 300m/s$. So (3.3) can be written as $p = a^2\rho$. Considering two cases for behavior of gas, in 1st case p and q are time variant and in 2nd case the p and q are time invariant. Former one is dynamic form and later one is quasi-static form which are discussed in next sections.

3.2 Quasi-Static Model for Gas Network

This is the case where flows q and pressure p are time invariant so there will be no continuity Equation (3.1) and pressure loss Equation (3.2) can be written as:

$$\partial_x \rho = -\frac{\lambda}{2Da^2}\frac{q|q|}{\rho} \qquad q = constant$$

$$\partial_x(\rho)\rho = -\frac{\lambda}{2Da^2}q|q|$$
$$\frac{1}{2}\partial_x(\rho^2) = -\frac{\lambda}{2Da^2}q|q|$$

Considering $p = a^2\rho$, the above equation becomes

$$\partial_x(p^2) = -\frac{a^2\lambda}{D}q|q|$$

$${p_j}^2 - {p_i}^2 = -\frac{a^2\lambda}{D_{ij}}q_{ij}|q_{ij}|L_{ij} \qquad q_{ij} = constant$$

After applying Kirchhoff's law at the nodes, the full algebraic system will become:

$${p_j}^2 - {p_i}^2 = -\frac{a^2\lambda}{D_{ij}}q_{ij}|q_{ij}|L_{ij} \tag{3.4}$$

$$\sum_j q_{ij} - \sum_k q_{ij} = 0 \qquad \text{for all} \quad i \in N_o \tag{3.5}$$

$$\sum_j q_{ij} - \sum_k q_{ij} - D_i(t) = 0 \qquad \text{for all} \quad i \in N_{in} \tag{3.6}$$

$$p - \hat{p}_i = 0 \qquad \text{for all} \quad i \in N_{out} \tag{3.7}$$

Parameters $\lambda, \frac{D_{ij}}{L_{ij}}, a^2$ will be collected in p. Pressure supply $\hat{p}_i$ and demand flow D_i will be considered as input and will be collected in vector u. All the states q_{ij}, p_i will be collected in vector x. So, the overall system will be written as.

$$A(p) + H(p)(x \otimes g(x)) + Bu = 0 \tag{3.8}$$

3.3 Dynamic Model for Gas Network

Dropping kinetic energy term, replacing v with $\frac{q}{\rho}$ and putting $p = a^2\rho$ in (3.2), now the Isothermal equations will be of the form

$$\text{(Dynamic Case)} : \begin{cases} \partial_t \rho + \partial_x q = 0 \\ \partial_t q + a^2 \partial_x \rho = -\frac{\lambda}{2D}\frac{q|q|}{\rho}. \end{cases} \tag{3.9}$$

In order to solve the set of PDEs in (3.9), first we have to convert the PDEs into ODEs by discretiztion as discussed in Chapter 2. To do this, pipe length should be $(x_i - x_{i-1}) \in [1000m, 5000m]$). For a given pipe segment ij having length L_{ij}, we took discrete points for pressure at nodes i, j and flow q at beginning and end of that pipe as shown in figure below.

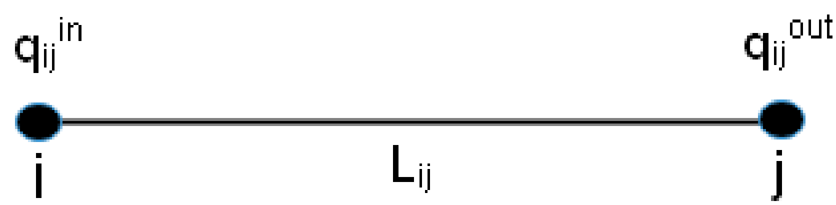


Figure 3.1: Single Pipe

Here q_{ij}^{in} is flow at beginning of the pipe into the section ij, while q_{ij}^{out} is the flow at end of the pipe out of the pipe ij, also there is pressure at node i and j denoted by $a^2\rho_i$ and $a^2\rho_j$ respectively. For a dynamic pipe ij, (3.9) can be written as

$$\begin{cases} \partial_t \frac{\rho_i+\rho_j}{2} + \frac{{q_{ij}}^{out}-{q_{ij}}^{in}}{L_{ij}} = 0 \\ \partial_t \frac{{q_{ij}}^{out}-{q_{ij}}^{in}}{2} + a^2 \frac{\rho_i-\rho_j}{L_{ij}} = -\frac{\lambda}{4D_{ij}} \frac{({q_{ij}}^{out}+{q_{ij}}^{in})|{q_{ij}}^{out}+{q_{ij}}^{in}|}{\rho_i+\rho_j} \end{cases} \tag{3.10}$$

To reduce complexity introducing a new variable $\frac{{q_{ij}}^{out}+{q_{ij}}^{in}}{\rho_i+\rho_j}$. Above system will be transformed to (3.11) & (3.12).

$$\partial_t \frac{\rho_i+\rho_j}{2} + \frac{{q_{ij}}^{out}-{q_{ij}}^{in}}{L_{ij}} = 0 \tag{3.11}$$

$$\partial_t \frac{{q_{ij}}^{out}-{q_{ij}}^{in}}{2} + a^2 \frac{\rho_i-\rho_j}{L_{ij}} + \frac{\lambda}{4D_{ij}} |{q_{ij}}^{out}+{q_{ij}}^{in}| y_{ij} = 0 \tag{3.12}$$

We will also have equations for internal node, demand nodes,supply nodes. Applying Kirchhoff law for interior nodes (sum of gas flow into the node is equal to sum of gas flow outside of that node) and we get (3.13).

$$\sum_{ji\in A} {q_{ij}}^{out} - \sum_{ik\in A} {q_{ij}}^{in} = 0 \qquad \forall i \in N_o \tag{3.13}$$

The sum of flows should be equal to demand at the demand nodes as given in (3.14).

$$\sum_{ji\in A} {q_{ij}}^{out} - \sum_{ik\in A} {q_{ij}}^{in} - D_i(t) = 0 \qquad \forall i \in N_- \tag{3.14}$$

For the supply node we have input pressure $\hat{p}_i(t)$, and for supply nodes we get (3.15).

$$a^2\rho_i(t) - \hat{p}_i(t) = 0 \qquad \forall i \in N_+ \tag{3.15}$$

$$(\rho_i + \rho_j)y_{ij} - ({q_{ij}}^{out} + {q_{ij}}^{in}) = 0 \qquad \forall e = ij \in A_{pipe} \tag{3.16}$$

As an extra variable (y_{ij}) is introduced so it is necessary to take into account (3.16). Parameters $\lambda, \frac{1}{D_{ij}}, \frac{1}{L_{ij}}, a^2$ will be collected in p.Pressure supply $\hat{p}_i(t)$ and demand flow $D_i(t)$ will be considered as input and will be collected in vector $u(t)$. All the states $\rho_i, {q_{ij}}^{out}, {q_{ij}}^{in}, y_{ij}$ will be collected in vector x. So, the overall system of equation $(3.11) - (3.16)$ will be written as.

$$E\dot{x} = A(p)x + H(p)(x \otimes g(x)) + Bu \tag{3.17}$$

Here E is singular, so this system is Differential Algebraic Equation (DAE). $(g(x))_i = x_i$ or $(g(x))_i = |x_i|$. If we assume g(x)=x, then the dynamical system will become quadratic.

3.4 Example Gas Network Model

Considering the gas network model: The network in Figure 3.2 has 16 pipes and 17 nodes(1 supply node and 8 demand nodes) [17]. Considering the quasi static case where pressures at nodes and flows through pipes are time invariant and are considered states of the system. They are collected in column vector x: Since there are 16 pipes and 17 nodes, so there will be 17 different pressures p_i nodes and 16 different flows q_{ij} for 16 pipes of the network. So state vector $x \in \mathbb{R}^{33\times 1}$ will be of the form:

$$x = (\underline{p_1, p_2 \cdots p_{16}, p_{17}} \quad , \quad \underline{q_{1,2}, q_{2,3} \cdots q_{15,16}, q_{15,17}})$$

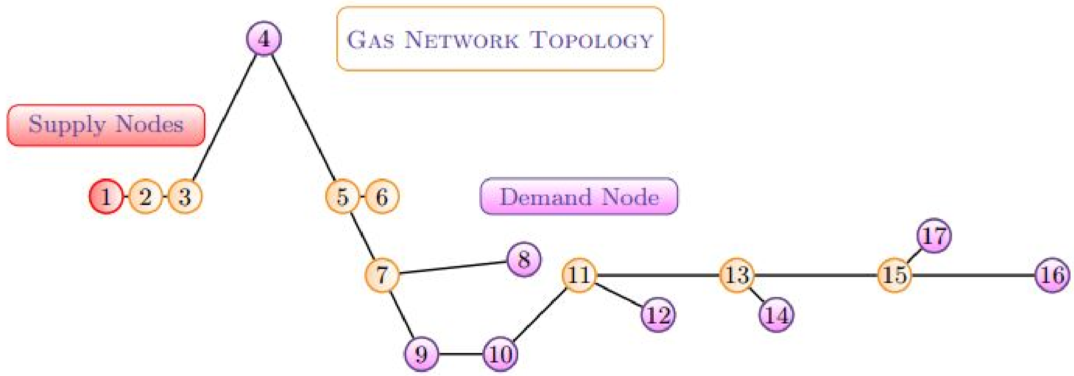


Figure 3.2: Gas Network Model [17]

1 supply node and 8 demand nodes will be considered as inputs and will be collected in input vector $u \in \mathbb{R}^{9\times 1}$ as follow.

$$u = (D_4, D_8, D_9, D_{10}, D_{12}, D_{14}, D_{16}, D_{17}, \hat{p_1})$$

Now come to the modeling of the gas network in quasi-static form:

Since there are 16 pipes, so we will have 16 equations like (3.4).

8 equations related to (3.5) for 8 internal nodes. Internal node equations are denoted using N_I.

8 equations related to (3.6) for 8 demand nodes. Demand node equations are denoted using N_D.

1 equation related to (3.7) for 1 supply node. Supply node equation is denoted using N_S.

So overall there will be 33 equations for the example gas network that is given in Figure 3.2. System of equation will be of the form (3.8) and is given below.

$$\underset{(33\times 33)}{A(p)}\underset{(33\times 1)}{x} + \underset{(33\times 33^2)}{\mathrm{H(p)}}\underset{(33^2\times 1)}{(x \otimes g(x))} + \underset{(33\times 9)}{B}\underset{(9\times 1)}{u} = 0 \tag{3.18}$$

Where

$$H(p)(x \otimes g(x)) = \begin{pmatrix} {p_2}^2 - {p_1}^2 + \frac{a^2\lambda}{D_{12}} q_{12}|q_{12}|L_{12} \\ {p_3}^2 - {p_2}^2 + \frac{a^2\lambda}{D_{32}} q_{32}|q_{32}|L_{32} \\ \vdots \\ {p_{17}}^2 - {p_{15}}^2 + \frac{a^2\lambda}{D_{15,17}} q_{15,17}|q_{15,17}|L_{15,17} \\ {p_{16}}^2 - {p_{15}}^2 + \frac{a^2\lambda}{D_{15,16}} q_{15,16}|q_{15,16}|L_{15,16} \\ O(17:33,1) \end{pmatrix} \tag{3.19}$$

$$A(p)x = \begin{pmatrix} O(1:16,1) \\ N_I(17:24,1) \\ N_D(25:32,1) \\ N_S(33,1) \end{pmatrix} \tag{3.20}$$

$$Bu = \begin{pmatrix} O(1:24,1) \\ u(25:33,1) \end{pmatrix} \tag{3.21}$$

After Assuming $\lambda = 0.0003328$, diameter of the pipes $D = 0.26$, $a = 430.5m/s$, supply pressure $44.5bar$ and Demands as given in [17]. We computed the solutions $x_0, x_1, \cdots, x_N$ for different inputs $u_0, u_1, \cdots, u_N$ against the (3.18) with

$$u_i = u_0 + \frac{i}{N}(u_N - u_0) \tag{3.22}$$

Here u_0 is trivial solution with supply pressure $p_0 \approx 45bar$. The singular values of snapshot matrix $Y = [x_0, \cdots, x_n]$ are found and are given in Figure 3.3.

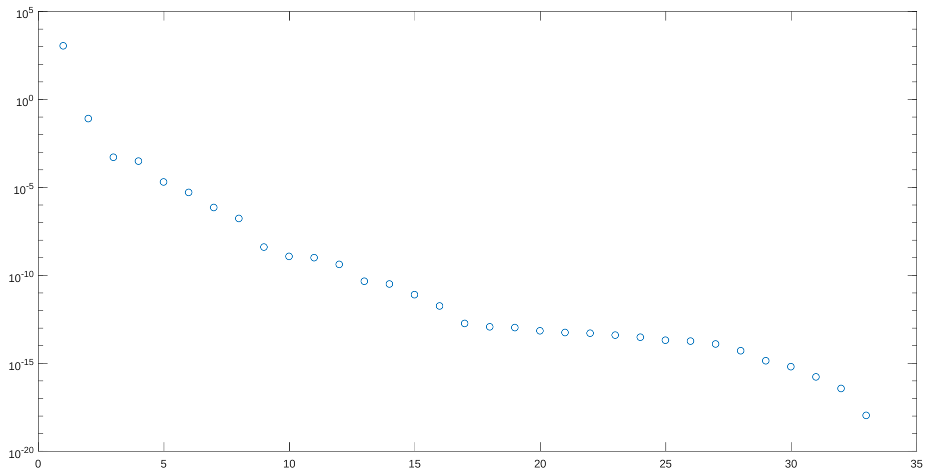


Figure 3.3: Singular Values

The Figure 3.3 is showing that there is a low dimensional linear space in which solution lies. Here 1st 5 singular values are $> 10^{-5}$, other singular values are so small $< 10^{-10}$, it means that the given model of 16 pipes and 17 nodes could be reduced to 5 pipes. POD can be very useful for this quadratic linear function, however direct implementation is expensive.

Now coming to dynamic form, where pressure and flows are time variant. 17 Pressures p at nodes ,16 flows q_{ij}^{out}, 16 flows q_{ij}^{in} and 16 extra variables for every pipe γ_{ij} will be considered states and will be collected in state vector $x \in \mathbb{R}^{65\times 1}$. Dynamic case is having system of equations $(3.11) - (3.16)$ which will be in the form of DAE as follow.

$$\underset{(65\times 65)}{E}\underset{(65\times 1)}{\dot{x}} = \underset{(65\times 65)}{A(p)}\underset{(65\times 1)}{x} + \underset{(65\times 65^2)}{H(p)}\underset{(65^2\times 1)}{(x \otimes g(x))} + \underset{(65\times 9)}{B}\underset{(9\times 1)}{u} \tag{3.23}$$

Where vector x will be of the form:

$$x = (p_1, \cdots, p_{17}, q_{1,2}^{in}, \cdots, q_{15,17}^{in}, q_{1,2}^{out}, \cdots, q_{15,17}^{out}, \gamma_{1,2}, \cdots, \gamma_{15,17})$$

Input vector u is having constant demands and varying supply pressure

$$\hat{p}_1(t) = 44.5 + 2.5 \times (1 - \cos(\frac{\pi t}{1h})) \quad t \in (0, 1.5h)$$

The above system in (3.23) is quadratic form of DAEs, or special structure of DAEs. These systems are known as descriptor systems. When we reduce a descriptor system, it is possible that structure would lost and $\mathcal{H}_2$ and $\mathcal{H}_\infty$ errors would grow. In order to retain the structure, split the DAE into strictly proper and polynomial part and apply model reduction technique on strictly proper part, which is discussed in next section.

3.5 Indirect POD Method for DAEs

In indirect method, DAEs are splitted into strictly proper and polynomial part and then the model reduction technique is applied to strictly proper part only. Details are given in subsections below.

3.5.1 Index-1 Linear DAE's

In this subsection, we shall discuss the splitting of linear descriptor system. Considering a linear descriptor system "LDS",

$$LDS : \begin{cases} E\dot{x}(t) & = Ax(t) + Bu(t) \\ y(t) & = Cx(t) + Du(t) \end{cases} \tag{3.24}$$

Where $E, A \in \mathbb{R}^{n\times n}$ and E is a singular matrix. $x(t) \in \mathbb{R}^n$,$u(t) \in \mathbb{R}^m$ and $Y(t) \in \mathbb{R}^p$ are the states , inputs and outputs respectively. $B \in \mathbb{R}^{n\times m}$,$C \in \mathbb{R}^{p\times n}$ and $D \in \mathbb{R}^{p\times m}$. Applying Laplace transformation on above system 3.24 gives us:

$$\mathcal{L}(E\dot{X}(t)) = \mathcal{L}(AX(t) + Bu(t))$$

$$E(sX(s) + X(0)) = AX(s) + Bu(s)$$

here for zero initial conditions $X(0) = 0$,so

$$X(s) = (Es - A)^{-1}Bu(s)$$

Transfer function of DAEs will be

$$G(s) = \frac{Y(s)}{u(s)}$$

$$G(s) = \frac{CX(s) + Du(s)}{u(s)}$$

Simplifying it will give us $G(s) = C(sE - A)^{-1}B + D$. Similarly for reduced DAEs, it will be $\tilde{G}(s) = \tilde{C}(s\tilde{E} - \tilde{A})^{-1}\tilde{B} + \tilde{D}$.

Using the concept of [13] and splitting $G(s)$ and $\tilde{G}(s)$ into strictly proper and polynomial part.

$$G(s) = G_{sp}(s) + P(s)$$

$$\tilde{G}(s) = \tilde{G}_{sp}(s) + \tilde{P}(s)$$

In order to have bounded $\mathcal{H}_2$ and $\mathcal{H}_\infty$ errors, polynomial part of $G(s)$ should match the polynomial part of $\tilde{G}(s)$, Hence ($P(s) = \tilde{P}(s)$). So model reduction technique will be applied to strictly proper part $G_{sp}(s)$ only and polynomial part will be retained as it is, resulting in error transfer function as given below.

$$G_{err}(s) = G(s) - \tilde{G}(s) = G_{sp}(s) - \tilde{G_{sp}}(s)$$

The above equation shows that the error transfer function doesn't contain a polynomial part and it only contains strictly proper part i.e., $\lim_{s\to\infty} G_{err}(s) = 0$. It means that interpolating $\tilde{G_{sp}}(s)$ on $G_{sp}(s)$ is just like interpolating $\tilde{G}(s)$ on $G(s)$.

Now considering semiexplicit descriptor system (SDS):

$$SDS = \begin{cases} E_{11}\dot{x}_1(t) + E_{12}\dot{x}_2(t) & = A_{11}(x_1(t)) + A_{12}(x_2(t)) + B_1u(t) \\ 0 & = A_{21}(x_1(t)) + A_{22}(x_2(t)) + B_2u(t) \\ y(t) & = C_1(x_1(t)) + C_2(x_1(t)) + Du(t) \end{cases} \tag{3.25}$$

where the state is $x(t) = [x_1^T(t), x_2^T(t)]^T \in \mathbb{R}^n$ with $x_1(t) \in \mathbb{R}^{n_1}, x_2(t) \in \mathbb{R}^{n_2}$, and $n_1 + n_2 = n$, the input is $u(t) \in \mathbb{R}^m$, the output is $y(t) \in \mathbb{R}^p$, and $E_{11}, A_{11} \in \mathbb{R}^{n_1 \times n_1}, E_{12}, A_{12} \in \mathbb{R}^{n_1 \times n_2}, A_{21} \in \mathbb{R}^{n_2 \times n_1}, A_{22} \in \mathbb{R}^{n_2 \times n_2}, B_1 \in \mathbb{R}^{n_1 \times m}, B_2 \in \mathbb{R}^{n_2 \times m}, C_1 \in \mathbb{R}^{p \times n_1}, C_2 \in \mathbb{R}^{p \times n_2}, D \in \mathbb{R}^{p \times m}$. We consider it an index 1 descriptor system after assuming that A_{22} and $E_{11} - E_{12}A_{22}^{-1}A_{21}$ are non-singular, also polynomial part $P(s)$ of $G(s)$ is constant matrix

$$P(s) = CMB + D \tag{3.26}$$

$$M = \lim_{s \to \infty} (sE - A)^{-1} = \begin{pmatrix} 0 & E_A^{-1}E_{12}{A_{22}}^{-1} \\ 0 & -{A_{22}}^{-1}(I + A_{21}E_A^{-1}E_{12}{A_{22}}^{-1}) \end{pmatrix} \tag{3.27}$$

Here $E_A = E_{11} - E_{12}A_{22}^{-1}A_{21}$

3.5.2 Index-1 Bilinear DAE's

Consider a bilinear SISO descriptor system of the form,

$$\begin{aligned} E_{11}\dot{x}_1(t) + E_{12}\dot{x}_2(t) &= A_{11}x_1(t) + A_{12}x_2(t) + N_{11}x_1(t)u(t) + N_{12}x_2(t)u(t) + B_1u(t) \\ 0 &= A_{21}x_1(t) + A_{22}x_2(t) + N_{21}x_1(t)u(t) + N_{22}x_2(t)u(t) + B_2u(t) \\ y(t) &= C_1x_1(t) + C_2x_2(t) \end{aligned} \tag{3.28}$$

Lemma 3.5.1. *Let* $H_k(s_1, \ldots, s_k) = c(s_kE - A)^{-1}N \ldots (s_2E - A)^{-1}N(s_1E - A)^{-1}b$ *be a k-th transfer function associated with a descriptor system of the form (3.28), where* A_{22} *and* $E_{11} - E_{12}A_{22}^{-1}A_{21}$ *are both non-singular. Then the polynomial part of* $H_k(s_1, \ldots, s_k)$ *is a constant matrix given by*

$$P_k = C(MN)^{k-1}MB, \tag{3.29}$$

where $M = \begin{bmatrix} 0 & M_1 \\ 0 & M_2 \end{bmatrix}$, *in which*

$$\begin{aligned} M_1 &= (E_{11} - E_{12}A_{22}^{-1}A_{21})^{-1}E_{12}A_{22}^{-1} && (3.30) \\ M_2 &= -A_{22}^{-1}A_{21}(E_{11} - E_{12}A_{22}^{-1}A_{21})^{-1}E_{12}A_{22}^{-1} - A_{22}^{-1}. && (3.31) \end{aligned}$$

Proof. Let

$$F_k(s_1, \ldots, s_k) = (s_kE - A)^{-1}N \ldots (s_2E - A)^{-1}N(s_1E - A)^{-1}b, \qquad (3.32)$$

then the polynomial part of $H_k(s_1, \ldots, s_k)$ is given by

$$P_k(s_1, \ldots, s_k) = C \lim_{s_1,\ldots,s_k \to \infty} F_k(s_1, \ldots, s_k) \qquad (3.33)$$

Note that for $k = 1$, (3.32) becomes,

$$F_1(s_1) = (s_1E - A)^{-1}B = \begin{bmatrix} s_1E_{11} - A_{11} & s_1E_{12} - A_{12} \\ -A_{21} & -A_{22} \end{bmatrix}^{-1} \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} = \begin{bmatrix} F_{1a}(s_1) \\ F_{1b}(s_1) \end{bmatrix}$$

This leads to

$$\begin{aligned} F_{1a}(s_1) &= \left((s_1E_{11} - A_{11}) - (s_1E_{12} - A_{12})A_{22}^{-1}A_{21}\right)^{-1}(B_1 + (sE_{12} - A_{12})A_{22}^{-1}B_2) \\ F_{1b}(s_1) &= -A_{22}^{-1}\left(B_2 + A_{21}F_{1a}(s_1)\right) \end{aligned}$$

Taking the limit $s_1 \to \infty$, we have

$$\lim_{s_1 \to \infty} F_1(s_1) = \begin{bmatrix} 0 & M_1 \\ 0 & M_2 \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}$$

Using (3.33) for $k = 1$, it is clear that (3.29) holds for $H_1(s_1)$. Now for $k = j \geq 1$, we assume that

$$\lim_{s_1,\ldots,s_j \to \infty} F_j(s_1, \ldots, s_j) = (MN)^{j-1}MB. \qquad (3.34)$$

We need to show that (3.29) holds for $k = j + 1$. Note that

$$F_{j+1}(s_1, \ldots, s_{j+1}) = (s_{j+1}E - A)^{-1}NF_j(s_1, \ldots, s_j)$$

Taking the limit $s_1, \ldots, s_j \to \infty$, we have

$$\lim_{s_1,\ldots,s_j \to \infty} F_{j+1}(s_1, \ldots, s_{j+1}) = (s_{j+1}E - A)^{-1}N(MN)^{j-1}MB.$$

Now using similar formulation as done for $F_1(s_1)$, we have

$$\lim_{s_1,\ldots,s_{j+1} \to \infty} F_{j+1}(s_1, \ldots, s_{j+1}) = MN(MN)^{j-1}MB.$$

Thus by induction (3.29) holds. □

Chapter 4

Numerical Results and Discussions

We have applied the model reduction technique POD with DEIM, and POD without DEIM on the example large scale nonlinear RC circuit. We have also discretized the nonlinear flow problems and applied POD method on it. Since gas distribution networks are represented by descriptor systems. Model reduction of structured SISO linear and bilinear descriptor systems is also performed for a random example.

4.1 Nonlinear RC-Circuit (Nonlinear ODEs)

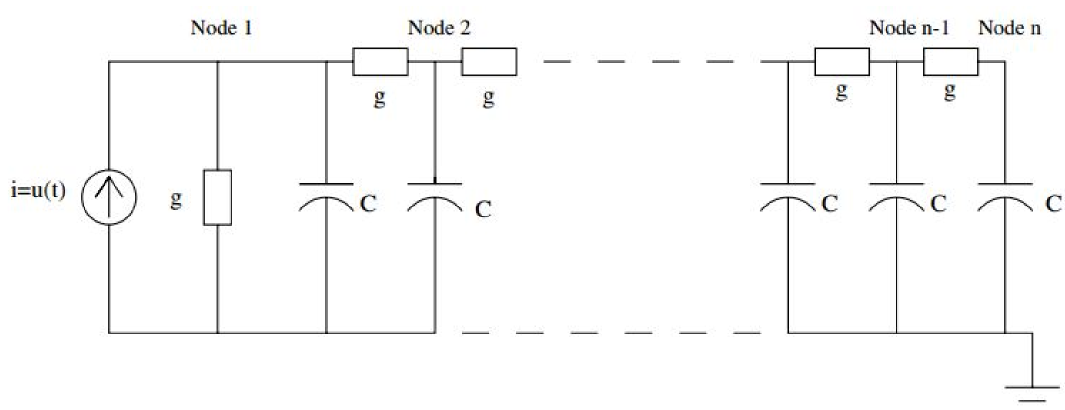


Figure 4.1: Large Scale RC Circuit [7]

Considering a large scale RC circuit as given in Figure 4.1. Here all resistors are nonlinear and are assumed to be the same. When the potential difference from

Node-1 to Node-2 through resistor is v, then the current flowing from Node-1 to Node-2 will be $I = g(v)$ and

$$g(v) = e^{40v} + v - 1 \tag{4.1}$$

Input current source at Node-1 is $i = u(t)$. Output is the potential at node-1 with the state variables being the potential at Nodes $1, \cdots, N$.

Following set of equations can be used for its model

$$\begin{aligned} u(t) &= C\frac{dv_1}{dt} + g(v_1 - v_2) + g(v_1) \\ g(v_1 - v_2) &= C\frac{dv_2}{dt} + g(v_2 - v_3) \\ &\vdots \\ g(v_{n-2} - v_{n-1}) &= C\frac{dv_{n-1}}{dt} + g(v_{n-1} - v_n) \\ g(v_{n-1} - v_n) &= C\frac{dv_n}{dt} \end{aligned}$$

Above set of equations can be arranged as

$$\dot{v}(t) = \begin{pmatrix} -g(v_1) - g(v_1 - v_2) \\ g(v_1 - v_2) - g(v_2 - v_3) \\ \vdots \\ g(v_{n-1} - v_{n-2}) - g(v_{n-1} - v_n) \\ g(v_{n-1} - v_n) \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix} u(t) \tag{4.2}$$

where it is assumed that $C = 1$. (4.2) is a nonlinear system of ODEs and can be written as:

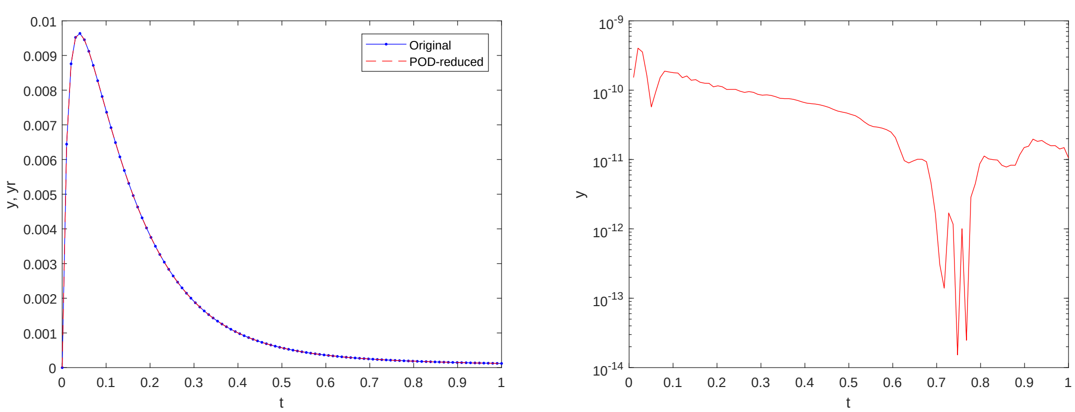
$$\dot{v}(t) = RC\_f(v) + Bu(t) \tag{4.3}$$

4.1.1 Implementation of POD

By applying POD technique we reduced large scale RC-circuit of n nodes to r nodes ($r << n$). State space vector $v(t)$ was transformed to low dimension state space vector $z(t) \in \mathbb{R}^{r\times 1}$, by projecting vector $z(t)$ onto the plane $U_r \in \mathbb{R}^{n\times r}$ which yields $v(t) = U_r z(t)$. We followed the steps of POD algorithm and used the projection basis U_r, so (4.3) can be written as

$$\dot{z}(t) = {U_r}^T RC\_f(U_r * z) + {U_r}^T Bu(t) \tag{4.4}$$

We have solved original large scale RC circuit having 500 nodes by solving (4.3) and reduced the circuit to 20 nodes by solving (4.4). Comparison of simulations of the original and POD reduced-order systems for an input $u(t) = e^{-10t}$ are shown in Figure 4.2.



(a) Similarity (b) Relative Error

Figure 4.2: Comparison of simulations of the original and POD reduced-order systems for an input $u(t) = e^{-10t}$

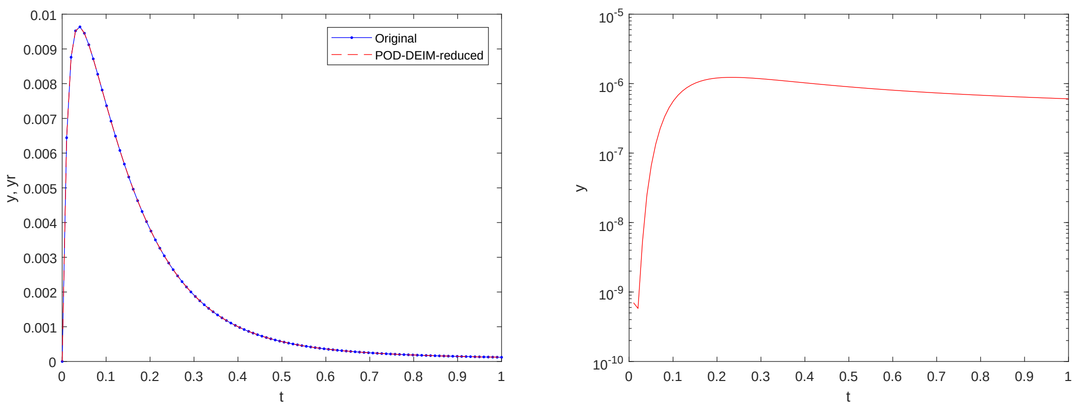
4.1.2 Implementation of POD-DEIM

We applied POD with DEIM to that large-scale RC circuit 4.1,now (4.3) can be written as

$$\dot{z}(t) = ({U_q}^T)(P_{DEIM})[RC\_f(U_q * z)] + Bu(t) \tag{4.5}$$

We solved (4.5) and results are given below, we can see in Figure 4.3a, the results are similar to the original system.

Comparison of simulations of the original and POD-DEIM reduced-order systems for an input $u(t) = e^{-10t}$ is shown in the Figure 4.3.



(a) Transient response (b) Relative error

Figure 4.3: Comparison of simulations of the original and POD-DEIM reduced-order systems for an input $u(t) = e^{-10t}$

Simulation time for POD and POD-DEIM are given in table below.

State Space Size (n)	Simulation Time)		
	Original	POD	POD-DEIM
n=500, r=20	11.63s	5.97s	6.17s

Table 4.1: Simualtion Time for Original System, POD and POD-DEIM Reduced System

4.2 Burger's Equation (Quadratic Bilinear ODEs)

A burger equation is fundamental PDE and for a given field $v(x, t)$ and a given diffusion coefficient d, it is of the form

$$v_t + vv_x = dv_{xx} \tag{4.6}$$

Initial boundary conditions are.

- For $x = 0, v(0, t)t$ is considered as an input $u(t)$ to the system.

- For $x = n$, $v_{n+1} = v_n$

This equation can be observed as standard numerical test example for nonlinear model reduction.

As discussed in Chapter 2: Any physical system represented by nonlinear PDEs can be converted to nonlinear ODEs and then it can be reduced using any model reduction technique. Complete scenario is shown in Figure 4.4 below.

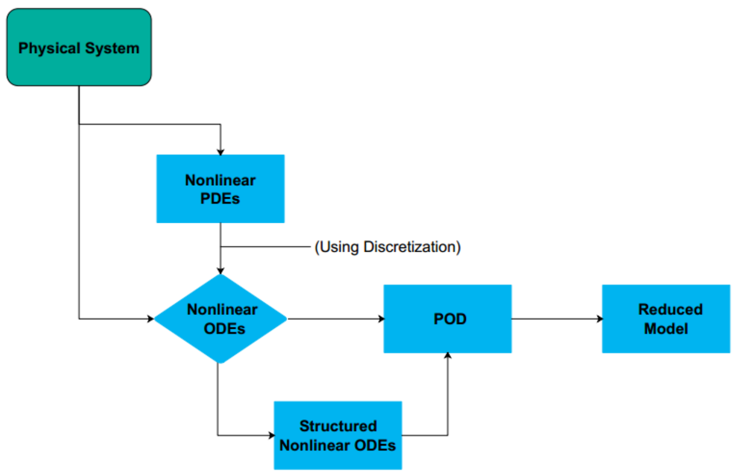


Figure 4.4: QBDAES

We discretized the burger equation using central difference method with respect to the dependent variable x, and after discretization (4.6) that is PDE, will become ODE of the form given below.

$$\dot{v_i} + v_i(\frac{v_i - v_{i-1}}{h}) = d(\frac{v_{i+1} - 2v_i + v_{i-1}}{h^2}) \tag{4.7}$$

Here h is the difference of v between the two-time intervals.

For $i = 1$ (4.7) becomes

$$\dot{v_1} + v_1(\frac{v_1 - v_\circ}{h}) = d(\frac{v_2 - 2v_1 + v_\circ}{h^2})$$

By following initial boundary condition ($x = 0$), "$v_\circ$" will be considered as input

$u(t)$ to the system. So above equation will become

$$\dot{v_1} + v_1(\frac{v_1 - u(t)}{h}) = d(\frac{v_2 - 2v_1 + u(t)}{h^2})$$

For $i = 2$, (4.7) will become

$$\dot{v_2} + v_2(\frac{v_2 - v_1}{h}) = d(\frac{v_3 - 2v_2 + v_1}{h^2})$$

Similarly for $i = n.$,

$$\dot{v_n} + v_n(\frac{v_n - v_{n-1}}{h}) = d(\frac{v_{n+1} - 2v_n + v_{n-1}}{h^2})$$

. By following final boundary condition, for time $t = n$, $v_{n+1} = v_n$. Defining

$$v(t) = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \tag{4.8}$$

This will define other matrices E, A, H, N & B. Matrix E will be generated against $\dot{v(t)}$, matrix A will be generated against $v(t)$, matrix H will be generated against square terms of $v(t)$, matrix N will be generated against bilinear terms $v(t)u(t)$ and finally matrix B will be generated for input $u(t)$. Whole system will be of the form of a descriptor system.

$$E\dot{v}(t) = Av(t) + H(v \otimes v) + Nv(t)u(t) + Bu(t) \tag{4.9}$$

The main idea of descriptor system formed in (4.9) was introduced in [16]. It's a quadratic bi-linear descriptor system, where $E \in \mathbb{R}^{n \times n}$, $A \in \mathbb{R}^{n \times n}$, $N \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times 1}$ are state space matrices and their dimension is fixed by the state. $u(t) \in \mathbb{R}^n$ is input to the system. $H \in \mathbb{R}^{n \times n^2}$ is matricization of Hesse tensor of the right hand side of the system of DAEs in (4.9).

We solved the burger equation for given boundary conditions and considering diffusion coefficient ($d = 0.02$) and input ($u(t) = \cos(\pi t)$ & $u(t) = 2\sin(\pi t)$) as given in [12].

We then applied POD on discretized burger equation (4.9). Vector $v(t)$ will be projected on a lower dimension i.e., $r << n$ subspace $z(t)$.

$$E\dot{z}(t) = A_r z(t) + H_r(z \otimes z) + N_r z(t)u(t) + B_r u(t) \tag{4.10}$$

Here $E_r \in \mathbb{R}^{r\times r}$, $A_r \in \mathbb{R}^{r\times r}$, $N_r \in \mathbb{R}^{r\times r}$, $B_r \in \mathbb{R}^{r\times 1}$ and $H_r \in \mathbb{R}^{r\times r^2}$. Results of burger equation for original and reduced system are given below.

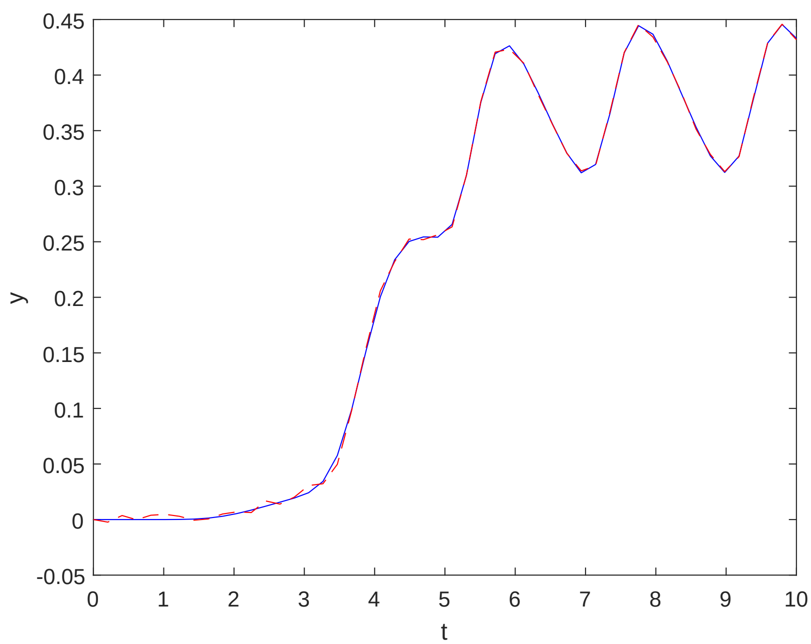


Figure 4.5: Original and Reduced Solution of burger equation for Input ($u(t) = \cos(\pi t)$)

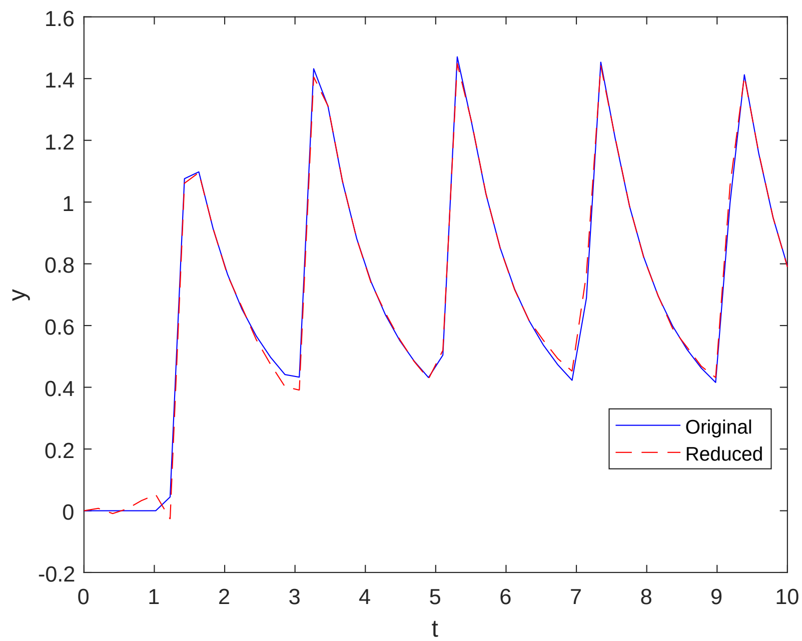


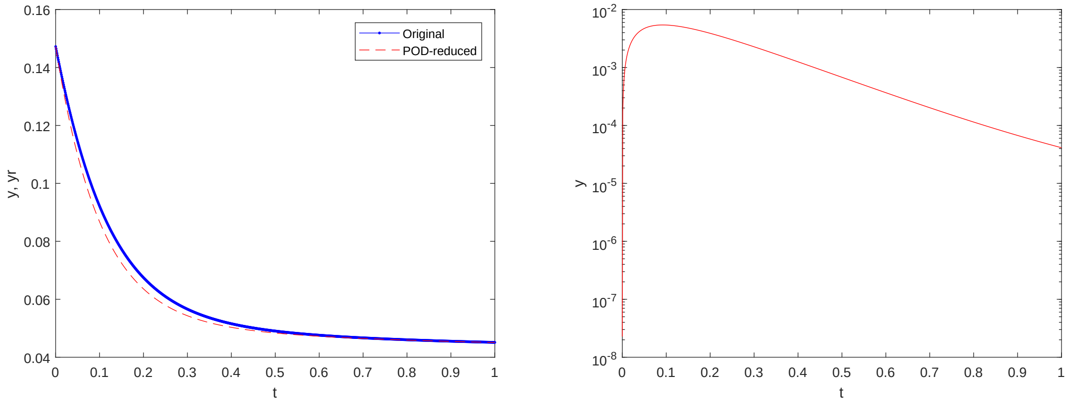
Figure 4.6: Original and Reduced Solution of burger equation for Input($u(t) = 2\sin(\pi t)$)

Results in Figures 4.5 and 4.6 are same for both original and reduced system.

4.3 Random Example

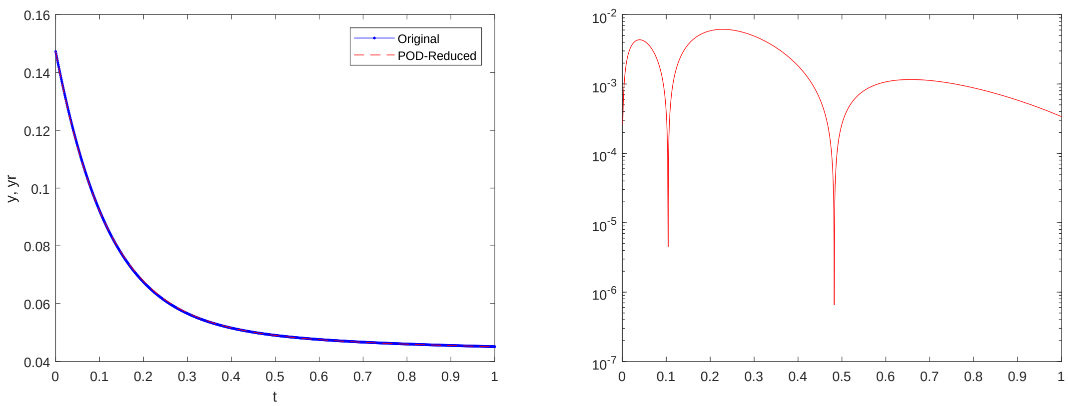
4.3.1 Linear DAE

As explained in Chapter 3, Section 3.5, we have reduced the Linear descriptor system (3.25) by direct method (without splitting) and indirect method (with splitting). Considering input function $u = e^{-10t}$, a linear descriptor system having state dimension $n = 100$ is reduced to $r = 10$ and $r = 20$. The time-domain responses of the actual and the reduced linear systems, obtained by using the POD method (direct & indirect), are shown in Fig. 4.7-4.10.



(a) Transient response (b) Relative error

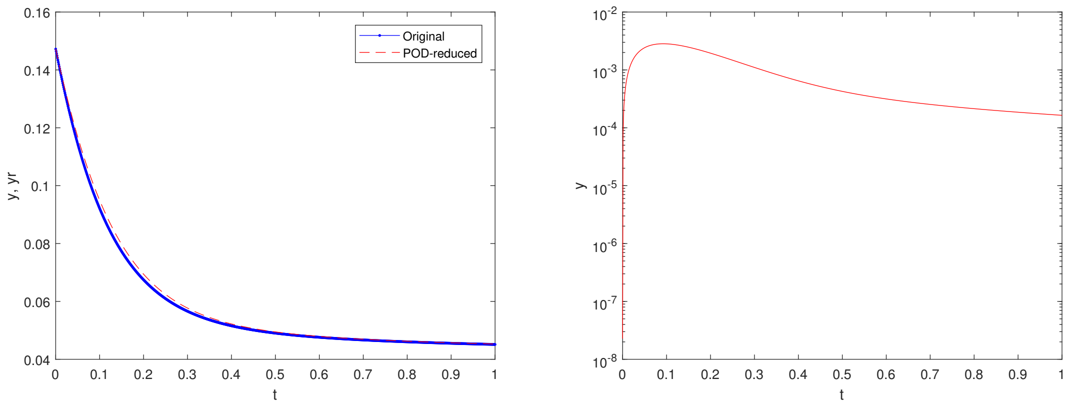
Figure 4.7: Direct reduction approach of linear system for r=10



(a) Transient response (b) Relative error

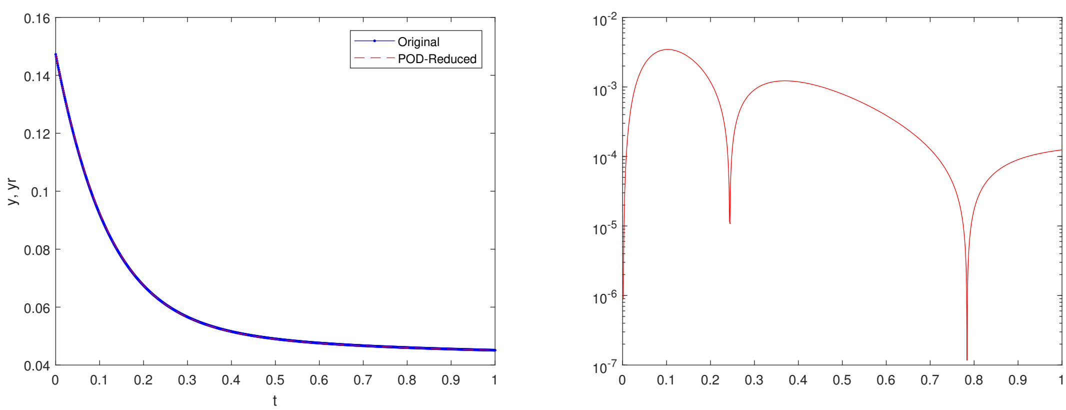
Figure 4.8: Indirect reduction approach of linear system for r=10

By comparing results in Figures 4.7 and 4.8, we can see that the results of original system and reduced system are improved for indirect approach as compared to direct approach.



(a) Transient response (b) Relative error

Figure 4.9: Direct reduction approach of linear system for r=20



(a) Transient response (b) Relative error

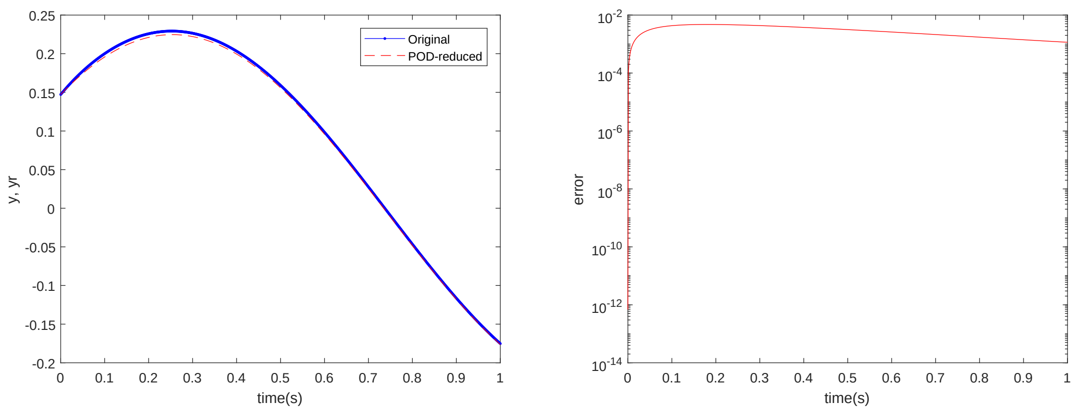
Figure 4.10: Indirect reduction approach of linear system for r=20

Similarly for $r = 20$, comparing results in Figures 4.9 and 4.10, we can see that the results of original system and reduced system are improved for our indirect approach as compared to direct approach and as a result relative error also decreased.

4.3.2 Bilinear DAE

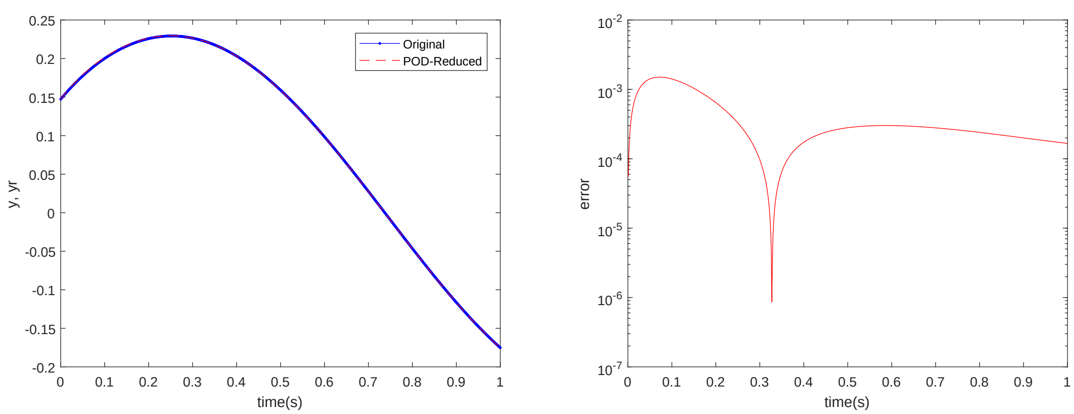
Considering bilinear descriptor system (3.28) that is given in Chapter 3, Section 3.5, we have reduced it by direct method (without splitting) and indirect method (with splitting). Considering input function $u = \cos(\pi t)$, a structured [1] bilinear descriptor system having state dimension $n = 100$ is reduced to $r = 10$ and $r = 20$.

The time-domain responses of the actual and the reduced bilinear systems, obtained by using the POD method (direct & indirect), are shown in Figures 4.11-4.14.



(a) Transient response (b) Relative error

Figure 4.11: Direct reduction approach of bilinear system for r=10



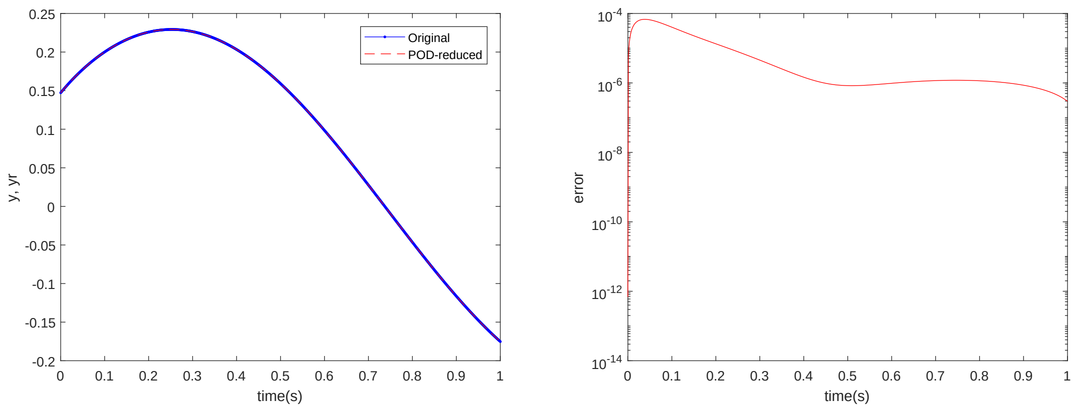
(a) Transient response (b) Relative error

Figure 4.12: Indirect reduction approach of bilinear system for r=10

By comparing results in Figures 4.11 and 4.12, we can see that the relative error

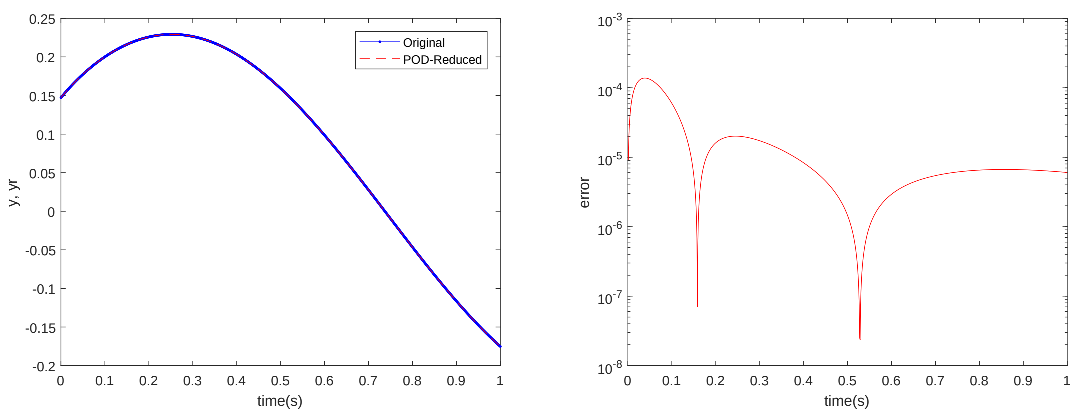
[1]From structured it means that for bilinear terms $(N_{ij}x_i(t)u(t))$, we are considering only N_{11} is nonzero while N_{12}, N_{21}, N_{22} are considered zero matrices

is decreased when we used indirect approach.



(a) Transient response (b) Relative error

Figure 4.13: Direct reduction approach of bilinear system for r=20



(a) Transient response (b) Relative error

Figure 4.14: Indirect reduction approach of bilinear system for r=20

By comparing results in Figures 4.13 and 4.14, we can see that the relative error is decreased when we used indirect approach for bilinear system.

Chapter 5

Conclusion and Future Work

Mathematical models of the gas distribution networks are in the form of nonlinear differential algebraic equations also called descriptor systems. Since there size is often large, model order reduction is used as an efficient tool for numerical simulation. It is observed that existing model reduction techniques are not directly applicable to nonlinear DAEs as they are unable to retain the structure of DAEs. This may results in unbounded approximation error. However for some structured nonlinear descriptor systems, the system can be decomposed into strictly proper and polynomial parts. An indirect POD technique for model reduction of descriptor systems has been proposed, where only the strictly proper part of the original system is reduced and the polynomial part of the original system is retained as it is in the original system. Reduction of DAEs associated with the gas distribution networks is more efficient in the indirect POD method in terms of system error, especially at very high frequencies.

An important future work will be the splitting of more general nonlinear DAEs. Also it would be interesting to work on the implementation of the proposed reduction technique on an actual gas distribution network.

Bibliography

[1] Bales, P., Kolb, O., & Lang, J. (2009). Hierarchical modelling and model adaptivity for gas flow on networks. Computational ScienceICCS 2009, 337-346.

[2] Jansen, L., & Tischendorf, C. (2014). A unified (P) DAE modeling approach for flow networks. In Progress in Differential-Algebraic Equations (pp. 127-151). Springer Berlin Heidelberg.

[3] Ehrhardt, K., & Steinbach, M. C. (2005). Nonlinear optimization in gas networks. In Modeling, simulation and optimization of complex processes (pp. 139-148). Springer Berlin Heidelberg.

[4] Herty, M., Mohring, J., & Sachers, V. (2010). A new model for gas flow in pipe networks. Mathematical Methods in the Applied Sciences, 33(7), 845-855.

[5] Steinbach, M. C. (2007). On PDE solution in transient optimization of gas networks. Journal of computational and applied mathematics, 203(2), 345-361.

[6] Brouwer, J., Gasser, I., & Herty, M. (2011). Gas pipeline models revisited: model hierarchies, nonisothermal models, and simulations of networks. Multiscale Modeling & Simulation, 9(2), 601-623.

[7] Chen, Y. (1999). Model order reduction for nonlinear systems (Doctoral dissertation, Massachusetts Institute of Technology).

[8] Volkwein, S. (2011). Model reduction using proper orthogonal decomposition. Lecture Notes, Institute of Mathematics and Scientific Computing, University of Graz. see http://www. uni-graz. at/imawww/volkwein/POD. pdf.

[9] Rathinam, M., & Petzold, L. R. (2003). A new look at proper orthogonal decomposition. SIAM Journal on Numerical Analysis, 41(5), 1893-1925.

[10] Chaturantabut, S., & Sorensen, D. C. (2010). Nonlinear model reduction via discrete empirical interpolation. SIAM Journal on Scientific Computing, 32(5), 2737-2764.

[11] Feng, P. B. L. Model Reduction for Dynamical Systems-Lecture 9.

[12] Benner, P., & Breiten, T. (2012). Two-sided moment matching methods for nonlinear model reduction.

[13] Gugercin, S., Stykel, T., & Wyatt, S. (2013). Model reduction of descriptor systems by interpolatory projection methods. SIAM Journal on Scientific Computing, 35(5), B1010-B1033.

[14] Antoulas, Athanasios C., Danny C. Sorensen, and Serkan Gugercin. "A survey of model reduction methods for large-scale systems." Contemporary mathematics 280 (2001): 193-220.

[15] Antoulas, Athanasios C. Approximation of large-scale dynamical systems. Society for Industrial and Applied Mathematics, 2005.

[16] C. Gu, "QLMOR: a projection-based nonlinear model order reduction approach using quadratic-linear representation of nonlinear systems," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 30, no. 9, pp. 13071320, 2011.

[17] Grundel, S., Hornung, N., Klaassen, B., Benner, P., & Clees, T. (2013). Computing surrogates for gas network simulation using model order reduction. In Surrogate-Based Modeling and Optimization (pp. 189-212). Springer New York.

[18] Grundel, S., Hornung, N., & Roggendorf, S. (2016). Numerical aspects of model order reduction for gas transportation networks. In Simulation-Driven Modeling and Optimization (pp. 1-28). Springer, Cham.

[19] Grundel, S., Jansen, L., Hornung, N., Clees, T., Tischendorf, C., & Benner, P. (2014). Model order reduction of differential algebraic equations arising from the simulation of gas transport networks. In Progress in differential-algebraic equations (pp. 183-205). Springer, Berlin, Heidelberg.

[20] Osiadacz, A. (1984). Simulation of transient gas flows in networks. International journal for numerical methods in fluids, 4(1), 13-24.

[21] Azevedo-Perdicolis, T. P., & Jank, G. (2007). Modelling aspects of describing a gas network through a DAE system. IFAC Proceedings Volumes, 40(20), 40-45.

[22] Gross, J. L., & Yellen, J. (2005). Graph theory and its applications. CRC press.

[23] P. Astrid, S. Weiland, K. Willcox, and T. Backx. Missing point estimation in models described by proper orthogonal decomposition. IEEE T. Automat. Contr., 53(10):22372251, 2008.

[24] M. Barrault, Y. Maday, N.C. Nguyen, and A.T. Patera. An empirical interpolation method: application to efficient reduced-basis discretization of partial differential equations. C. R. Math., 339(9):667 672, 2004.

[25] M.A. Grepl, Y. Maday, N.C. Nguyen, and A.T. Patera. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. ESAIM: Math. Model. Num., 41(03):575605, 2007.

[26] N.C. Nguyen, A.T. Patera, and J. Peraire. A best points interpolation method for efficient approximation of parametrized functions. Int. J. Numer. Meth. Eng., 73(4):521543, 2008.

[27] Moore, Bruce. "Principal component analysis in linear systems: Controllability, observability, and model reduction." IEEE transactions on automatic control 26.1 (1981): 17-32.

[28] Freund, Roland W. "Model reduction methods based on Krylov subspaces." Acta Numerica 12 (2003): 267-319.

[29] Gugercin, Serkan, Athanasios C. Antoulas, and Cchristopher Beattie. "$\mathcal{H}_2$ model reduction for large-scale linear dynamical systems." SIAM journal on matrix analysis and applications 30.2 (2008): 609-638.

[30] Lall, Sanjay, Jerrold E. Marsden, and Sonja Glavaki. "A subspace approach to balanced truncation for model reduction of nonlinear control systems." International Journal of Robust and Nonlinear Control: IFACAffiliated Journal 12.6 (2002): 519-535.

[31] Rathinam, Muruhan, and Linda R. Petzold. "A new look at proper orthogonal decomposition." SIAM Journal on Numerical Analysis 41.5 (2003): 1893-1925.

Appendix A – MATLAB Codes for Model Order Reduction

NLPOD.m

```
function [ Br, Cr,Up,z,yr] = NLPOD (t,v0,x_s, B, C, r)
[U,~,~] = svd(x_s);
Up = U(:,1:r);
Br = Up'*B;
Cr = C*Up;
v1=Up'*v0;
dzdt = @(t,z) Up'*RC_f(Up*z) + Br*input1(t);
options = odeset('RelTol',1e-8,'AbsTol',1e-10);
[~,z] = ode15s(dzdt,t,v1,options);
yr = Cr*z';
```

Listing 5.1: POD Code

POD-DEIM.m

```
function [Br, Cr,Up, v1,Pdeim,yr] = NLPOD_DEIM (B, C,N, Np)
t = linspace(0,10,N);
tp = linspace(0,10,Np);
v0 = sparse(N,1);
dvdtp = @(tp,v) RC_f(v)+ B*input1(tp);
options = odeset('RelTol',1e-8,'AbsTol',1e-10);
[~,v] = ode15s(dvdtp,tp,v0,options);
v=v';
S=size(RC_f(v))
[U,~,~] = svd(v);
Up = U(:,1:Np);
Br = Up'*B;
Cr = C*Up;
vt=v;
F = [];
for i = 1: Np
    F = [F, RC_f(vt(:,i))];
end
q=size(F)
[Uf,~,~] = svd(F);
Uf = Uf(:,1:Np); %projection basis
[~,index1]=max(abs(Uf(:,1)));
Ud = Uf(:,1);     %U
I = eye(N);
P = I(:,index1);
cP = index1;
for i=2:Np
```

```
        alpha=inv(P'*Ud)* P'*Uf(:,i);
        res = Uf(:,i) - Ud*alpha;
      [~,index_new]=max(abs(res));
     cP = [cP; index_new];
     P = [P I(:,index_new)];
     Ud = [Ud Uf(:,i)];
end
Pdeim = Ud*inv(P'*Ud)*P';
v1 = sparse(Np,1);
dzdt = @(t,z) Up'*Pdeim*RC_f(Up*z) + Br*input1(t);
options = odeset('RelTol',1e-8,'AbsTol',1e-10);
[~,z] = ode15s(dzdt,t,v1,options);
yr = Cr*z';
```

Listing 5.2: POD-DEIM Code