DECENTRALIZED CONTROL OF LARGE-SCALE INTERCONNECTED SYSTEMS WITH APPLICATION TO WEB PROCESSING MACHINES

By

SIRASKAR NILESH BABANRAO

BACHELOR OF ENGINEERING

GOVERNMENT COLLEGE OF ENGINEERING, PUNE

MAHARASHTRA, INDIA

2002

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
DECEMBER, 2004

DECENTRALIZED CONTROL OF LARGE-SCALE INTERCONNECTED SYSTEMS WITH APPLICATION TO WEB PROCESSING MACHINES

Thesis Approved:

Dr. P. R. Pagilla

Thesis Adviser

Dr. L. L. Hoberock

Committee Member

Dr. E. A. Misawa

Committee Member

Dr. A. Gordon Emslie

Dean of the Graduate College

ACKNOWLEDGMENTS

I wish to express my sincerest appreciation to my major advisor, Dr. P. R. Pagilla for his intelligent supervision, constructive guidance, inspiration, and friendship.

I would like to extend my warmest thanks to my masters committee members: Dr. E. A. Misawa and Dr. L. L. Hoberock for their support and suggestions in completion of this research. Their guidance and understanding made the development of this thesis a positive learning experience.

I would also like to thank my research colleagues at Oklahoma State University Ramamurthy V. Dwivedula, Yongliang Zhu, Seshadri Kuppuswamy, and A. Dubey.

TABLE OF CONTENTS

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

NOMENCLATURE

$\mathbb{R}$: Set of real numbers.

$\lambda_{min}(M)$: Minimum eigenvalue of matrix M.

$\lambda_{max}(M)$: Maximum eigenvalue of matrix M.

$(\Delta A)_{max}$: Maximum possible deviation in A.

M^T : Transpose of matrix M.

E : Modulus of elasticity.

A : Cross-sectional area of web.

J_i : Polar moment of inertia of downstream roller of i-th web span.

n_i : Torque ratio from motor shaft to downstream driven roller of i-th web span.

L_i : Length of i-th web span.

R_i : Radius of roller downstream to i-th web span.

b_{fi} : Bearing friction in i-the roller.

t_i : Web tension (force) in i-th web span.

t_{ri} : Reference or steady state operating tension.

T_i : Change in tension from steady state.

τ_i : Motor torque applied on downstream roller of i-th span.

U_i : Change in input torque from steady state.

ω_i : Angular velocity of the i-th roller.

v_i : Web velocity in immediate vicinity of downstream roller of i-th span.

v_{ri} : Reference or steady state operating web velocity.

V_i : Change in web velocity from operating value.

t : Time.

CHAPTER 1

INTRODUCTION

The last century saw revolutionary progress in technology, which has transformed yesterday's inventions into today's inseparable needs of human life. For example, invention of 'controlled' flight by Wright Brothers was followed by establishment of huge aerospace industry and this led to the problem of handling the network of large number of aircrafts flying in the sky. Similar situation arose in the automobile industry, which has produced innumerable vehicles. Regulation of a large number of vehicles is becoming a tough challenge for the transportation industry. An exponential rise in the number of such smaller systems is clearly visible in many areas such as manufacturing systems, information systems and telecommunication systems. In such situations, it is possible to define subsystems which interact with each other to form large-scale systems. Such large-scale systems are emerging with great importance in both physical as well as social sciences. Power networks, Heating, Ventilation and Air Conditioning (HVAC) systems, multiple aircraft formation systems, wireless telecommunication systems, Intelligent Vehicle and Highway Systems (IVHS) are some of the examples of physical systems, whereas, ecological population systems and global economic systems are examples in the social sciences. With rapid changes in the information systems and computers, internet is becoming one of the biggest large-scale system. Scientific analysis of large-scale systems is critical to making progress in many fields because of their wide range of applications. One aspect of such analysis is connective stability. A large-scale system is said to be connectively stable if it is stable for all possible interconnections among subsystems [1].

A large-scale system consists of a number of interconnected subsystems and it is typ-

ically characterized by a large number of state and input variables. Complexities in large-scale systems can make their analysis tedious and numerically intractable. Hence, the study of large-scale systems is usually simplified by decomposing them into a number of smaller interconnected subsystems. Decomposition of a large-scale system is a difficult task and needs scientific insight into the system behavior. This is because the same large-scale system can be decomposed into subsystems in more than one way. A unified decomposition scheme to yield optimum configuration is not available. One of the criteria behind such decomposition process is that each subsystem must be easy to analyze and the collective behavior of all subsystems, through certain interactions, must represent the overall system behavior. Some large-scale systems can be decomposed into subsystems which have a physical meaning attached to them. Such decomposition is known as physical decomposition. As noted in [1], physical decomposition of some of the large-scale systems may not always result in subsystems which are simple to analyze. In such cases, mathematical decomposition can be achieved through transformations of the original physical variables to obtain new variables of the subsystems that are simple to analyze. However, this new transformed large-scale system may lose much of the physical meaning.

Interconnections and coordination among constituting subsystems plays an important role in the study of any large-scale system. For example, consider a flight formation system as shown in a Fig. 1.1 in which each individual aircraft can be considered as a subsystem.

The overall shape of the formation is decided by the trajectory of each aircraft relative to the trajectories of the other aircrafts. If any one of the aircrafts fails to achieve a required position relative to the position of the other aircrafts, formation in the sky will go out of the desired shape. This example addresses a very important issue of coordination among subsystems, which affects the total behavior of entire large-scale system. Hence, along with the analysis of each individual subsystem, it is also important to study the nature of interactions among subsystems to understand overall behavior of the large-scale system.

Control is an important aspect in the operation of all large-scale systems. Extensive

Figure 1.1: Flight formation of aircrafts.

and well-directed research is trying to extend existing control theory to develop systematic analysis and control methodologies to handle complexities in large-scale systems. Towards design of efficient controllers for large-scale systems, one should consider some key challenges such as

(i) flexibility in control design approach,

(ii) uncertainty in information exchange,

(iii) reliability and robustness,

(iv) cost of implementation.

The structure of the large-scale system changes due to addition or removal of subsystems. It is desirable that the control algorithm be able to handle such changes without substantial redesign. Inclusion of such flexibility in control design demands development of a systematic methodology for control of large-scale systems.

The overall behavior of a large-scale system depends upon the information exchange among constituent subsystems. This assumes utmost importance in cases where subsystems are located widely apart or remote to controller, thus, giving rise to unwanted signal delays, attenuation of the signal and lower signal to noise ratio. Such corrupted knowledge

of exchanged information may not be useful to a controller and may cause instability of the closed loop system.

Failure of one subsystem may adversely affect the performance of other subsystems; in extreme situations, overall large-scale system may become unstable. It is a challenging task to design a controller, which will ensure robust stability and reliable performance of a large-scale system against failure of individual subsystem(s). This problem is compounded with the massive nature of large-scale systems and possibility of erroneous information exchange.

The speed and number of computations involved in a control algorithm are crucial factors affecting real-time implementation. Revolutionary inventions in computing speeds of microprocessors and the memory capacity of data storage elements have eased the job of control of large-scale systems. But use of such sophisticated equipments alone may not form an efficient solution in all situations mainly because of economic reasons. Moreover, a control designer has to develop control schemes that lead to minimal cost of computer interfacing and wiring which otherwise may increase exponentially with increase in the complexity of large-scale systems. Thus, the main challenge encountered by a control designer is to develop efficient controller schemes for large-scale systems that can be implemented with minimal cost and resources. The need for a systematic mathematical formulation including computational and computer interfacing cost with respect to a cost minimization design, was first addressed in [2].

Motivated by these challenges, many control strategies have been developed in the past few years. In a broad sense, control methods for large-scale systems can be categorized into two basic approaches: 1) Centralized control schemes and 2) Decentralized control schemes.

In centralized control schemes, the large-scale system is considered with its aggregate model as a Multi-Input Multi-Output (MIMO) system. Centralization of all the feedback information is the basic assumption in this approach. H_∞ Robust controller design, LQG/LTR

design, Lyapunov design are some of the attractive techniques used to design reliable and robust controllers for linear and nonlinear large-scale systems. Using centralized design approach, computational efforts increase tremendously with increase in complexity and extent of large-scale systems. Even for a small modification in the large-scale system, entire control algorithm may have to be redesigned from scratch. The centralized control schemes, in their traditional approach, do not provide affirmative answers to the challenges encountered in the control of large-scale systems.

In decentralized control schemes, decomposition of a large-scale model is achieved first and then the controller for each individual subsystem is designed in such a way that it makes use of local available information only. The underlying feature of a decentralized control scheme is that it does not require any information from other subsystems and thus, problem of imperfect information exchange does not arise at all. However, this poses another challenge of maintaining connective stability of overall large-scale system without using information from other subsystems. Essentially, the decentralized scheme makes use of (1) the decomposed structure of the large-scale system and (2) ideas from the classical centralized control theory. It may be noted that each subsystem in itself could be a multi-input multi-output system. Thus, decentralized control schemes provide convenient feature of extending the classical control theory to find affirmative answers to the challenges in control of large-scale systems. A brief review on the literature of decentralized controllers is given below.

1.1 Past Research in Decentralized Controllers

Development of decentralized control theory based on the fundamental knowledge of the centralized control theory is the chief motive behind vast research done in this field. In late 1960's, study on decentralized controllers and decomposition structure started germinating. Early challenges faced in development of decentralized control were addressed in [2], which also showed early direction of research in decentralized control theory. The

basic framework of decentralized controller was systematically laid down in [1] where application of decentralized controllers for a variety of fields such as power networks, spacecraft systems was discussed. Most recent applications of decentralized controllers include wafer temperature for multi-zone rapid thermal processing systems [3], platoons of underwater vehicles [4], and cooperative robotic system [5]. Research on decentralized controllers is not only concentrated on special applications but the general classes of large-scale systems were also studied.

Generalized results in decentralized control theory started evolving with classical results on linear time invariant large-scale systems. Necessary and sufficient conditions for existence of stabilizing decentralized controllers for linear time invariant systems (LTI) were proposed in [6] with the introduction of the concept of *fixed modes*. Fixed modes are defined in association with a decentralized control system in which linear, constant gain, state feedback controllers are used. Modes of the closed loop large-scale systems, which cannot be influenced by a decentralized control scheme are known as "fixed modes". By definition, these are analogous to centralized fixed modes and further it was shown in [6] that, like the centralized case, decentralized fixed modes are unmovable. It was shown in [7] that these fixed modes are unmovable if constant state-feedback gains are used. However, with time-varying gains, the fixed modes associated with the decentralized control system can be eliminated. Elimination of fixed modes using time varying gains was also achieved independently in [8]. Later, many types of fixed modes were explored during extensive work on large-scale systems. In these works, existence of stabilizing decentralized controllers for linear time invariant large-scale systems was investigated extensively. Two aspects of decentralized control were addressed: 1)Which kind of fixed modes can be eliminated? 2) And what kind of controllers should be used? It is shown in [9] that all of the fixed modes except those associated with unstable zeros of complementary subsystems can be be stabilized by periodically time varying decentralized state feedback controllers. Later it was shown in [10] that if fixed modes associated with decentralized large-scale systems

cannot be eliminated by periodically time varying state feedback law, then no controller can eliminate them. It is important to note that the classical concept of fixed modes and relevant literature is developed for linear, time invariant large-scale systems.

Motivated by the success of decentralized control schemes for LTI systems, efforts were made to develop decentralized control schemes for nonlinear large-scale systems. The first result, which systematically extends classical centralized adaptive control theory to decentralized adaptive control, was developed in [11]. But the result in [11] was obtained for the class of large-scale systems in which isolated subsystems have relative degree less than or equal to two.

Later, results on decentralized adaptive control were developed for more generalized class of large-scale systems in which Lyapunov analysis played an important role. Condition on relative degree of isolated subsystems was relaxed in [12] to obtain a decentralized adaptive controller. A class of large-scale systems in which matched interconnections and uncertainties are assumed to be bounded by the higher order polynomial in the norms of states. The matching condition is said to be satisfied if interconnections and uncertainties enter into the subsystem at the same point where decentralized control input enters into the subsystem. Decentralized control of large-scale systems with matched condition was investigated quite rigourously in the past [13–16] subjected to certain class of large-scale systems. Global decentralized adaptive control was obtained in [13] where a class of nonlinear systems was considered which can be transformed using a global diffeomorphism to the output feedback canonical form, where interconnections are the functions of a subsystem output only. Decentralized Model Reference Adaptive Control (MRAC) was considered in [14–16], which developed decentralized adaptive control schemes for a class of systems in which the matching condition is satisfied. Lower order control law was developed in [17] to show semi-global stability for the case of large-scale systems with higher order interconnections. The sliding mode technique was used in [18] to develop a low order controller.

1.2 Application: Web Processing Lines

A web is any material which is manufactured and processed in continuous, flexible strip form. Examples include paper, plastics, textiles, strip metals, and composites. A typical web material used in packaging industry is shown in Fig. 1.2. Web handling refers

Figure 1.2: Web material.

to the physical mechanics related to the transport and control of web materials through processing machinery. Web processing pervades almost every industry today. It allows us to mass produce a rich variety of products from a continuous strip of material. Products that include web processing somewhere in their manufacturing include aircraft, appliances, automobiles, bags, books, diapers, boxes, newspapers, and many more. Web tension and velocity are two key variables that influence the quality of the finished web and hence the products manufactured from it.

Early development of mathematical models for longitudinal dynamics of a web can be found in [19–22]. In [19], a mathematical model for longitudinal dynamics of a web span between two pairs of pinch rolls, which are driven by two motors, was developed. This model did not predict tension transfer and did not consider tension in the entering span. A modified model that considers tension in the entering span was developed in [21]. In [22], the moving web was considered as a moving continuum and general methods of

continuum mechanics were used in the development of a mathematical model. The study in [22] included the steady state and transient behavior of tensile force, stress, and strain in a web as functions of variables such as wrap angle, position and speed of the driven rollers, density, cross-sectional area, modulus of elasticity and temperature. In [23], equations describing web tension dynamics are derived based on the fundamentals of web behavior and the dynamics of the drives used for web transport; an example system was considered to compare torque control versus velocity control of a roll for regulation of tension in a web. Non-ideal effects such as temperature and moisture change on web tension were studied in [24]; based on the models developed, methods for distributed control of tension in multi-span web transport systems were studied. An overview of lateral and longitudinal dynamic behavior and control of moving webs was presented in [25]. A review of the problems in tension control of webs can be found in [26]. A robust centralized H_∞ controller for a web winding system consisting of an intermediate driven roller and unwind/rewind rolls was proposed in [27].

1.3 Contributions

Literature review of the developments in decentralized control strategies reveals the fact that decentralized control problem with unmatched interconnections is not studied comprehensively. Lack of concrete results in this area inhibited entry of rigorous control theory into many large-scale system applications, one of which is web processing lines. The basic problem is that the web handling dynamics do not satisfy matching conditions, hence most of the decentralized control theory remains unapplicable. This gives a strong motivation to obtain reliable strategies for decentralized control of large-scale systems where matching conditions are not satisfied. In this respect, the contributions of the research work involved in this thesis are summarized next.

1. An adaptive decentralized state-feedback regulator is developed for a class of large-scale systems, with unmatched and linear interconnections. Global asymptotic sta-

bility is shown using Lyapunov analysis.

2. Model reference adaptive controller is developed for a class of large-scale systems, with unmatched and linear interconnections. A new "modified reference model" is proposed to solve the exact tracking problem in the presence of unmatched interconnections.

3. An adaptive decentralized state-feedback regulator is developed for a class of large-scale systems with unmatched and nonlinear interconnections. Depending upon prior knowledge of interconnecting parameters, four different conditions, under which system is semi-globally stable, are obtained.

4. A model for the unwind (rewind) roll is developed by explicitly considering the variation of radius and inertia resulting from release (accumulation) of material to (from) the process.

5. Systematic decentralized control algorithms are developed for web processing lines. A strategy for computing the equilibrium inputs and reference velocities based on the reference of the master speed roller, which sets the desired web transport velocity for the line, is given.

6. The proposed decentralized controllers are implemented on an actual web handling platform (Fig. 1.3), which mimics most of the features of an industrial web processing line.

7. Extensive experiments were conducted to validate the decentralized controllers proposed in this thesis. The results show substantial improvements in the tension error regulation than existing two-loop, industrial PI controllers.

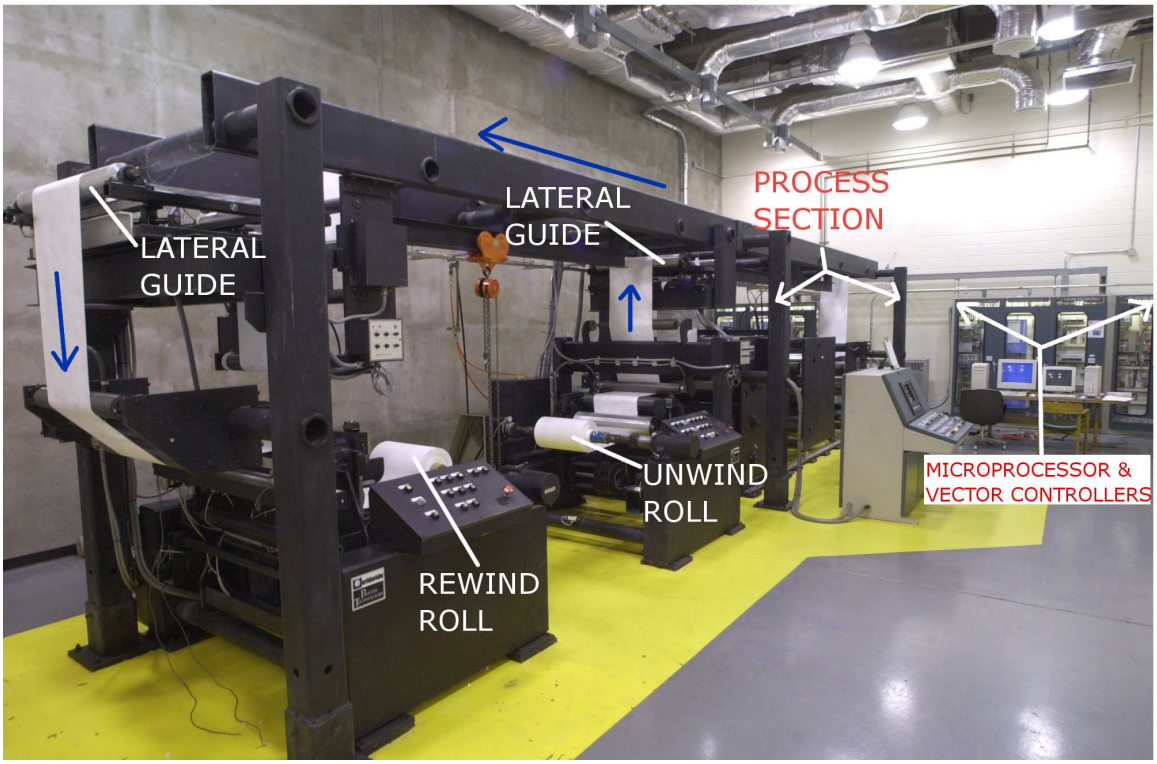


Figure 1.3: A picture of the experimental web line

CHAPTER 2

DESIGN OF DECENTRALIZED CONTROLLERS

In this chapter, stable decentralized controllers for a class of large-scale interconnected systems are developed. Both linear and nonlinear interconnections are considered in the development of the controllers. In the case of linear interconnections, control designs for both adaptive state regulation and tracking are developed. For nonlinear interconnections, a decentralized adaptive scheme that achieves semi-global stability under a set of necessary and sufficient conditions is developed.

The following class of large-scale systems is considered:

$$S_i : \qquad \dot{x}_i(t) = A_i x_i(t) + b_i u_i(t) + g_i(t,x) \tag{2.1}$$

where $x_i(t) \in \mathbb{R}^{n_i}$ and $u_i(t) \in \mathbb{R}$ are the state vector and input vector, respectively, of the i-th subsystem , $i \in I = \{0,1,...,N\}$ and $g_i(t,x)$ is the *unmatched* interconnection function. The constant vectors $b_i \in \mathbb{R}^{n_i}$ are assumed to be known. The total state vector of the entire large-scale system is denoted by $x \in \mathbb{R}^n$ and given in terms of the local states of the subsystems as $x^T = [x_0^T, x_1^T, ..., x_N^T]$. Thus, the dynamics of each subsystem consists of two parts. The first part is a linear system of local states, x_i. The second part consists of interconnections between i-th subsystem and other subsystems, given by $g_i(t,x)$.

Depending upon the type of interconnections, i.e., whether $g_i(t,x)$ is linear or nonlinear, the following decentralized control schemes are developed. A decentralized adaptive controller that achieves state regulation is developed for large-scale systems with unmatched linear interconnections. A new model reference decentralized adaptive controller (MRAC) is also developed. The decentralized MRAC scheme is unique in the sense that the reference model contains information exchange between subsystems, i.e., each subsystem in

the reference model exchanges its reference information with other subsystems. A semi-globally stable adaptive state feedback controller with *unmatched nonlinear interconnections*, each interconnection function $g_i(t,x)$ is assumed to be a higher order polynomial in state x, is developed in section 2.3.

2.1 Preliminaries

The following two lemmas are important in establishing necessary and sufficient conditions that ensure stability of the overall large-scale system for the proposed controllers, which will be given in subsequent sections.

Lemma 2.1.1 *Consider the Algebraic Ricatti Equation*

$$A^\top P + PA + PRP + Q = 0. \tag{2.2}$$

If $R = R^\top \geq 0$, $Q = Q^\top > 0$, *A is Hurwitz, and the associated Hamiltonian matrix* $\mathscr{H} = \begin{bmatrix} A & R \\ -Q & -A^\top \end{bmatrix}$ *is hyperbolic, i.e.,* $\mathscr{H}$ *has no eigenvalues on the imaginary axis, then there exists a unique* $P = P^\top > 0$, *which is the solution of the ARE* (2.2).

If $A = A_{mi}$, $Q = \xi_i^2 I$ and $R = NI$, then the following lemma gives a computable condition under which the Hamiltonian matrix $\mathscr{H}_i = \begin{bmatrix} A_{mi} & NI \\ -\xi_i^2 I & -A_{mi}^\top \end{bmatrix}$ is hyperbolic.

Lemma 2.1.2 *$\mathscr{H}_i$ is hyperbolic if and only if*

$$\min_{\omega \in R} \sigma_{min}(A_{mi} - j\omega I) > \xi_i \sqrt{N} > 0 \tag{2.3}$$

Lemma 2.1.2 is a special case of Theorem 2 in [28] and is obtained by setting $C = 0$ in that Theorem.

Remark 2.1.1 *An efficient numerical algorithm for computation of* $\min_{\omega \in R} \sigma_{min}(A_{mi} - j\omega I)$ *using the bisection method can be found in [29].*

Lemma 2.1.3 *[17] The following inequality is true for any $z_i \in \mathbb{R}^{n_i}$, $z_j \in \mathbb{R}^{n_j}$ and any positive integers N and k:*

$$\sum_{i=0}^{N} ||z_i|| \sum_{j=0}^{N} ||z_j||^k \leq (N+1) \sum_{i=1}^{N} ||z_i||^{k+1} \tag{2.4}$$

Claim 2.1.1 *The following inequality is true for any positive integers i, j and p, $z_i \in \mathbb{R}^{n_i}$, $z_j \in \mathbb{R}^{n_j}$, and $\alpha_i \in \Gamma = \{x \in \mathbb{R} : x \geq 1\}$:*

$$\alpha_i ||z_i|| ||z_j||^p + \alpha_j ||z_i||^p ||z_j|| \leq \alpha_i^2 ||z_i||^{p+1} + \alpha_j^2 ||z_j||^{p+1} \tag{2.5}$$

The following steps show that the claim is true:

$$\begin{aligned} &\alpha_i^2 ||z_i||^{p+1} + \alpha_j^2 ||z_j||^{p+1} - \alpha_i ||z_i|| ||z_j||^p - \alpha_j ||z_i||^p ||z_j|| \\ &\qquad \geq \alpha_i^2 ||z_i||^{p+1} + \alpha_j^2 ||z_j||^{p+1} - \alpha_i \alpha_j ||z_i|| ||z_j||^p - \alpha_j \alpha_i ||z_i||^p ||z_j|| \\ &\qquad = (\alpha_i ||z_i||^p - \alpha_j ||z_j||^p)(\alpha_i ||z_i|| - \alpha_j ||z_j||) \geq 0 \end{aligned} \tag{2.6}$$

Claim 2.1.2 *The following inequality is true for any $z_i \in \mathbb{R}^{n_i}$, $z_j \in \mathbb{R}^{n_j}$ and any positive integers N and k:*

$$\sum_{i=0}^{N} \alpha_i ||z_i|| \sum_{j=0}^{N} ||z_j||^k \leq (N+1) \sum_{i=1}^{N} \alpha_i^2 ||z_i||^{k+1} \tag{2.7}$$

Claim 2.1.2 can be shown using induction procedure on N by repeated use of (2.5).

2.2 Decentralized Controllers for Large-Scale Systems with Linear Interconnections

In this section decentralized adaptive controllers are developed for large-scale systems represented by equation (2.1), in which each nonlinear interconnection function $g_i(t,x)$ is assumed to be linear, that is, $g_i(t,x) = \sum_{j=0, j\neq i}^{N} A_{ij} x_j(t)$. Hence, each subsystem of the large-scale system can be written as

$$S_i : \qquad \dot{x}_i(t) = A_i x_i(t) + b_i u_i(t) + \sum_{j=0, j\neq i}^{N} A_{ij} x_j(t) \tag{2.8}$$

The system matrices A_i contain some parameters, which are not known but it is assumed that the structures of matrices A_i are such that pairs (A_i, b_i) are controllable for all i. With this assumption, a decentralized adaptive controller for state regulation and a model reference decentralized adaptive controller for state tracking are given in the following sections.

2.2.1 Decentralized adaptive state regulation

In this case, it is assumed that A_{ij} are unknown but bounds on them are known, that is, the existence and knowledge of positive numbers, η_{ij}, are known such that

$$\eta_{ij}^2 \geq \lambda_{max}(A_{ij}^T A_{ij}) \tag{2.9}$$

Controllability of each pair (A_i, b_i) implies that one can assign the eigenvalues of A_i arbitrarily using state feedback gain k_i as

$$\bar{A_i} = A_i - b_i k_i^T \tag{2.10}$$

where $\bar{A_i}$ is an asymptotically stable subsystem matrix. The gain vector k_i is not known exactly because A_i is uncertain.

Choose the control inputs (2.11) where the adaptation law (2.12) is used to obtain an estimate, $\hat{k}_i(t)$, of k_i and P_i is a positive definite gain matrix.

$$u_i(t) = -\hat{k}_i^T x_i(t) \tag{2.11}$$

$$\dot{\hat{k}}_i(t) = -(x_i^T(t) P_i b_i) x_i(t) \tag{2.12}$$

Using (2.10) and defining the gain estimation error as $\tilde{k}_i(t) = k_i - \hat{k}_i(t)$, the state dynamics for each subsystem upon simplification becomes

$$\dot{x}_i(t) = \bar{A_i} x_i + b_i \tilde{k}_i^T x_i + \sum_{j=0, j\neq i}^{N} A_{ij} x_j(t) \tag{2.13}$$

To prove the stability of the state dynamics (2.13) for $i \in I$, consider the following Lyapunov function candidate

$$V(x, \tilde{k}) = \sum_{i=0}^{N} x_i^T P_i x_i + \tilde{k}_i^T \tilde{k}_i \tag{2.14}$$

where P_i is positive definite matrix. The derivative of the Lyapunov function candidate along the trajectories of (2.13) and adaptation law (2.12) can be obtained as

$$\dot{V}(x,\tilde{k}) = \sum_{i=0}^{N} x_i^T (\bar{A}_i^T P_i + P_i \bar{A}_i) x_i + x_i^T P_i \sum_{j=0, j\neq i}^{N} A_{ij} x_j(t) + \left(\sum_{j=0, j\neq i}^{N} A_{ij} x_j(t) \right)^T P_i x_i \quad (2.15)$$

Using following inequality for matrices X and Y

$$X^T Y + Y^T X \leq X^T X + Y^T Y \quad (2.16)$$

one can obtain the following bounds for the last term of $\dot{V}$:

$$\sum_{i=0}^{N} \left[x_i^T P_i \sum_{j=0, j\neq i}^{N} A_{ij} x_j(t) + \left(\sum_{j=0, j\neq i}^{N} A_{ij} x_j(t) \right)^T P_i x_i \right] \leq \sum_{i=0}^{N} x_i^T (N P_i P_i) x_i + \sum_{i=0}^{N} \sum_{j=0, j\neq i}^{N} x_j^T A_{ij}^T A_{ij} x_j \quad (2.17)$$

Last term in the above equation can be bounded as

$$\sum_i \sum_{j\neq i} x_j^T A_{ij}^T A_{ij} x_j \leq \sum_i \sum_{j\neq i} \eta_{ij}^2 x_j^T x_j = \sum_i \big(\sum_{j\neq i} \eta_{ji}^2 \big) x_i^T x_i$$

As a result, $\dot{V}$ satisfies

$$\dot{V}(x,\tilde{k}) \leq \sum_{i=0}^{N} x_i^T \left(\bar{A}_i^T P_i + P_i \bar{A}_i + N P_i P_i + \xi_i^2 I \right) x_i \quad (2.18)$$

where $\xi_i^2 = \sum_{j=0, j\neq i}^{N} \eta_{ji}^2$.

Therefore, we have the following. If there exist positive definite solutions, P_i, to the Algebraic Ricatti Equations (AREs)

$$\bar{A}_i^T P_i + P_i \bar{A}_i + N P_i P_i + (\xi_i^2 + \varepsilon_i) I = 0 \quad (2.19)$$

then

$$\dot{V}(x,\tilde{k}) \leq -\sum_{i=0}^{N} \varepsilon_i x_i^T x_i \quad (2.20)$$

Hence, if there exist positive definite solutions, P_i, to the AREs (2.19), $V(x,\tilde{k})$ is a Lyapunov function. As a result $x_i(t) \in \mathscr{L}_2 \cap \mathscr{L}_\infty$ for all $i \in I$. Further from the closed-loop dynamics (2.13), $\dot{x}_i(t) \in \mathscr{L}_\infty$ for all $i \in I$. This implies that $\lim_{t\to\infty} x_i(t) = 0$ for all $i \in I$.

Using lemma 2.1.1 and lemma 2.1.2, it can be seen that there exist positive definite solution to ARE (2.19) if

$$\min_{\omega \in R} \sigma_{min}(\bar{A_i} - j\omega I) > \xi_i \sqrt{N} > 0 \tag{2.21}$$

holds. Finally, the existence of $\varepsilon_i, i \in I$, in the AREs (2.19) follows from the continuity of the functions $f_i(\xi_i^2) := \min_{\omega \in R} \sigma_{min}(A_{mi} - j\omega I) - \sqrt{N\xi_i^2}$, that is, if there exists a ξ_i such that $f_i(\xi_i^2) > 0$, then there exists an $\varepsilon_i > 0$ such that $f_i(\xi_i^2 + \varepsilon_i) > 0$.

Remark 2.2.1 *Non-adaptive state regulation is a special case of the adaptive state regulation. If the system matrices, A_i are known perfectly then one can directly compute the gains k_i exactly and no adaptation is required. Hence, control inputs can be chosen as*

$$u_i(t) = -k_i^T x_i(t) \tag{2.22}$$

Stability proof is similar to that of the adaptive state regulation case, hence omitted. As expected, the sufficient conditions are given by equation (2.21)*. Further, in this case one can relax the condition that input u_i is a scalar because the stability proof holds even for $u_i(t) \in \mathbb{R}^{m_i}$ where m_i is number of inputs.*

2.2.2 Model reference adaptive control: Modified reference model

For the development of MRAC scheme, interconnecting parameters, A_{ij} in equation (2.8) are assumed to be known exactly. Further, each subsystem matrix $A_i \in \mathbb{R}^{n_i \times n_i}$ are unknown but it is assumed that constant vectors $k_i \in \mathbb{R}^{n_i}$ exist such that for a chosen asymptotically stable matrix A_{mi},

$$(A_i - A_{mi}) = b_i k_i^T \tag{2.23}$$

The entire large-scale system can be expressed as

$$S : \dot{x}(t) = Ax(t) + BU(t) \tag{2.24}$$

where $x^T(t) = [x_0^T(t), x_1^T(t), \ldots, x_N^T(t)]$, $U^T(t) = [u_0(t), u_1(t), \ldots, u_N(t)]$, A is a matrix composed of block diagonal matrix elements A_i and off-diagonal matrix elements A_{ij}, and B is a block diagonal matrix composed of B_i. The pair (A, B) is assumed to be controllable.

Existing research [11, 14, 15] has considered the decentralized MRAC problem for large-scale systems with a reference model given by

$$S_{mi}: \qquad \dot{x}_{mi}(t) = A_{mi}x_{mi}(t) + b_i r_i(t) \tag{2.25}$$

where $x_{mi}(t)$ are the reference state vectors and $r_i(t)$ are bounded reference inputs. This reference model is successfully used to show model reference adaptive control scheme for large-scale systems with matched interconnections. In this thesis, to solve the unmatched problem, the structure of the reference models is modified by making use of the known interconnection matrices, A_{ij}, in the reference model. The modified reference model for each individual subsystem, S_{mi}, is described by the equations

$$S_{mi}: \qquad \dot{x}_{mi}(t) = A_{mi}x_{mi}(t) + b_i r_i(t) - b_i k_{mi}^T x_m + \sum_{j=0, j\neq i}^{N} A_{ij}x_{mj}(t) \tag{2.26}$$

where $k_{mi} \in \mathbb{R}^n, n = \sum_{i=0}^{N} n_i$ and $x_m^T(t) = [x_{m0}^T(t), x_{m1}^T(t), \ldots, x_{mN}^T(t)]$. The reason for including the term $B_i k_{mi}^T x_m$ in the reference model of each individual subsystem is to stabilize the system matrix of the large-scale reference model. The reference model for the entire large-scale system is given by

$$S_m : \dot{x}_m(t) = A_m x_m(t) + Br(t) - BK_m^T x_m \tag{2.27}$$

where $r^T(t) = [r_0(t), r_1(t), \ldots, r_N(t)]$, $K_m = [k_{m0}, k_{m1}, \ldots, k_{mN}]$, and

$$A_m = \begin{bmatrix} A_{m0} & A_{01} & A_{02} & . & . & . & A_{0N} \\ A_{10} & A_{m1} & A_{12} & . & . & . & A_{1N} \\ . & . & & & & . & \\ A_{N0} & A_{N1} & . & . & . & . & A_{mN} \end{bmatrix}.$$

Notice that if A_m is not stable for given A_{mi}, then one can place the eigenvalues of $A_m - BK_m^T$ by choosing K_m; the controllability of (A,B) implies that (A_m,B) is controllable. If A_m is asymptotically stable for given A_{mi}, then one can simply choose K_m to be the null matrix.

Defining subsystem error as $e_i(t) = x_i(t) - x_{mi}(t)$, the error dynamics can be obtained as

$$\dot{e}_i(t) = A_i x_i(t) + b_i u_i(t) - A_{mi} x_{mi}(t) - b_i r_i(t) + b_i k_{mi}^T x_m(t) + \sum_{j=0, j\neq i}^{N} A_{ij} e_j(t) \tag{2.28}$$

The goal is to design bounded decentralized control inputs $u_i(t)$ such that $x_i(t)$ are bounded and the error $e_i(t)$ converges to zero, that is, $\lim_{t\to\infty} e_i(t) = 0$ for all $i \in I = \{0,1,\ldots,N\}$.

2.2.3 Adaptation law and proof of convergence

Choose the control inputs (2.29) where the adaptation law (2.30) is used to obtain an estimate, $\hat{k}_i(t)$, of k_i and P_i is a positive definite gain matrix.

$$u_i(t) = r_i(t) - k_{mi}^T x_m(t) - \hat{k}_i^T x_i(t) \tag{2.29}$$

$$\dot{\hat{k}}_i(t) = -(e_i^T(t) P_i b_i) x_i(t) \tag{2.30}$$

Using (2.23) and defining the gain estimation error as $\tilde{k}_i(t) = k_i - \hat{k}_i(t)$, the error dynamics upon simplification becomes

$$\dot{e}_i(t) = A_{mi} e_i + (A_i - A_{mi}) x_i - b_i \hat{k}_i^T x_i + \sum_{j=0, j\neq i}^{N} A_{ij} e_j(t) \tag{2.31}$$

$$= A_{mi} e_i + b_i \tilde{k}_i^T x_i + \sum_{j=0, j\neq i}^{N} A_{ij} e_j(t) \tag{2.32}$$

To prove stability of the error dynamics (2.32) for $i \in I$ together with adaptation laws (2.30), consider the following Lyapunov function candidate

$$V(e,\tilde{k}) = \sum_{i=0}^{N} (e_i^T P_i e_i + \tilde{k}_i^T \tilde{k}_i). \tag{2.33}$$

where P_i is positive definite matrix. The derivative of the Lyapunov function candidate along the trajectories of (2.30) and (2.32) is given by

$$\dot{V}(e,\tilde{k}) = \sum_{i=0}^{N} e_i^T (A_{mi}^T P_i + P_i A_{mi}) e_i + e_i^T P_i \sum_{j=0, j\neq i}^{N} A_{ij} e_j(t) + \left(\sum_{j=0, j\neq i}^{N} A_{ij} e_j(t) \right)^T P_i e_i \tag{2.34}$$

Now consider a case with cross terms involving product of e_i with e_j as

$$(\sum_{j\neq i}^{N} A_{ij} e_j)^T P_i e_i + e_i^T P_i (\sum_{j\neq i}^{N} A_{ij} e_j) = (A_{i1} e_1)^T P_i e_i + e_i^T P_i (A_{i1} e_1) + (A_{i2} e_2)^T P_i e_i + e_i^T P_i (A_{i2} e_2) + ... + (A_{iN} e_N)^T P_i e_i + e_i^T P_i (A_{iN} e_N) \tag{2.35}$$

To achieve strict decentralized controllers we need to decouple cross terms in state errors, which can be done using the following inequality for two matrices M and N:

$$M^T N + N^T M \leq M^T M + N^T N \tag{2.36}$$

One can obtain the following:

$$(A_{ij} e_j)^T P_i e_i + e_i^T P_i (A_{ij} e_j) \leq (A_{ij} e_j)^T (A_{ij} e_j) + (P_i e_i)^T (P_i e_i) = e_j^T (A_{ij}^T A_{ij}) e_j + e_i^T (P_i P_i) e_i \tag{2.37}$$

Using similar decoupling procedure for all the terms of equation (2.35) we obtain:

$$(\sum_{j\neq i}^{N} A_{ij} e_j)^T P_i e_i + e_i^T P_i (\sum_{j\neq i}^{N} A_{ij} e_i) \leq N e_i^T (P_i P_i) e_i + \sum_{j\neq i}^{N} e_j^T (A_{ij}^T A_{ij}) e_j \tag{2.38}$$

Now we can rewrite equation (2.34) as

$$\dot{V}(e,\tilde{k}) \leq \sum_{i=0}^{N} e_i^T (A_{mi}^T P_i + P_i A_{mi}) e_i + N e_i^T P_i P_i e_i + \sum_{i=0}^{N} \sum_{j=0, j\neq i}^{N} e_j^T A_{ij}^T A_{ij} e_j \tag{2.39}$$

$$= \sum_{i=0}^{N} e_i^T (A_{mi}^T P_i + P_i A_{mi} + N P_i P_i) e_i + \sum_{i=0}^{N} e_i^T \left(\sum_{j=0, j\neq i}^{N} A_{ji}^T A_{ji} \right) e_i \tag{2.40}$$

Defining ξ_i^2 as the maximum eigenvalue of the matrix $\sum_{j=0, j\neq i}^{N} A_{ji}^T A_{ji}$, we obtain

$$\dot{V}(e,\tilde{k}) \leq \sum_{i=0}^{N} e_i^T (A_{mi}^T P_i + P_i A_{mi} + N P_i P_i + \xi_i^2 I) e_i \tag{2.41}$$

Therefore, if

$$A_{mi}^T P_i + P_i A_{mi} + N P_i P_i + (\xi_i^2 + \varepsilon_i) I = 0 \tag{2.42}$$

then

$$\dot{V}(e, \tilde{k}) \leq -\sum_{i=0}^{N} \varepsilon_i e_i^T e_i \tag{2.43}$$

Hence, if there exist symmetric, positive definite solutions, P_i, to the AREs (2.42), $V(e, \tilde{k})$ is a Lyapunov function. As a result $e_i \in \mathscr{L}_2 \cap \mathscr{L}_\infty$ for all $i \in I$. Further, from the error dynamics (2.32), $\dot{e}_i$ is bounded for all $i \in I$. This implies that $\lim_{t\to\infty} e_i(t) = 0$ for all $i \in I$. Using lemma 2.1.1 and lemma 2.1.2, it can be seen that there exist positive definite solutions to ARE (2.42) if

$$\min_{\omega \in R} \sigma_{min}(A_{mi} - j\omega I) > \xi_i \sqrt{N} > 0 \tag{2.44}$$

Note that the existence of $\varepsilon_i, i \in I$, in the AREs (2.42) follows from the continuity of the functions $f_i(\xi_i^2) := \min_{\omega \in R} \sigma_{min}(A_{mi} - j\omega I) - \sqrt{N\xi_i^2}$, that is, if there exists a ξ_i such that $f_i(\xi_i^2) > 0$, then there exists an $\varepsilon_i > 0$ such that $f_i(\xi_i^2 + \varepsilon_i) > 0$.

Remark 2.2.2 *Constant gain state tracking is a simple special case of the model reference adaptive controller. If the system matrices, A_i, are known then one can directly compute the gains, k_i, accurately and no adaptation is required. Hence, control inputs are given by*

$$u_i(t) = r_i(t) - k_{mi}^T x_m(t) - k_i^T x_i(t) \tag{2.45}$$

Reference model is again given by equation (2.26) *and the stability proof is similar to that of the MRAC case. Chosen A_{mi} has to satisfy the same conditions as that in MRAC case. The sufficient conditions are given by equation* (2.44)*. Once ARE* (2.42) *is solved for A_{mi}, then the constant gain vectors, k_i, can be found out using known A_{mi} and the following equation*

$$A_i - A_{mi} = b_i k_i^T \tag{2.46}$$

2.3 Decentralized Adaptive Controller for Large Scale Systems with Nonlinear Interconnections

The development of decentralized controllers in previous section assumed linear interconnections among subsystems. In this section, $g_i(t,x)$ in (2.1) is assumed to involve terms with the higher orders of states. More general representation for $g_i(t,x)$ is assumed such that it consists of uncertain parameters in local subsystem as well as interconnections. In addition to parameter uncertainties, the system also contains model uncertainties. Though $g_i(t,x)$ has parameter as well as structural uncertainties, it is assumed that $g_i(t,x)$ can be bounded polynomially. Different types of polynomial bounds are considered, which will result in different conditions on the controller gains as explained below. Choose the control inputs (2.47) where the adaptation law (2.48) is used to obtain an estimate, $\hat{k}_i(t)$, of k_i and P_i is a positive definite gain matrix.

$$u_i(t) = \hat{k}_i^T x_i(t) \tag{2.47}$$

$$\dot{\hat{k}}_i(t) = -(x_i^T(t) P_i b_i) x_i(t) \tag{2.48}$$

Defining the gain estimation error as $\tilde{k}_i(t) = k_i - \hat{k}_i(t)$, the state dynamics upon simplification become

$$\dot{x}_i(t) = \bar{A}_i x_i + b_i \tilde{k}_i^T x_i + g_i(t,x) \tag{2.49}$$

where $\bar{A}_i = A_i - b_i k_i^T$ is a stable matrix with eigen values in the left half plane.

2.3.1 Stability proof

Consider the Lyapunov function candidate as

$$V(x,\tilde{k}) = \sum_{i=0}^{N} x_i^T P_i x_i + \tilde{k}_i^T \tilde{k}_i \tag{2.50}$$

where P_i is positive definite symmetric matrix. Differentiating V along the trajectories of (2.49) and using the adaptation law (2.48) we obtain

$$\dot{V} = \sum_{i=0}^{N} (\bar{A_i} x_i + g_i(t,x))^T P_i x_i + x_i^T P_i (\bar{A_i} x_i + g_i(t,x)) \tag{2.51}$$

Two different cases are considered depending upon the bounding structure for $g_i(t,x)$.

Case (A)

In this case $g_i(t,x)$ is assumed to be bounded as

$$||g_i(t,x)|| \leq \sum_{p=1}^{P_i} \sum_{k=0}^{N} \beta_{i,pk} ||x_k||^p \tag{2.52}$$

where $\beta_{i,pk}$ are known non-negative (that is positive or zero) real constants. The controller gains k_i are obtained by solving the algebraic Lyapunov equation

$$(A_i - b_i k_i^T)^T P_i + P_i (A_i - b_i k_i^T) = -Q_i \tag{2.53}$$

Because A_i and b_i form a controllable pair, for any positive definite Q_i, a positive definite solution, P_i, to equation (2.53) always exists. With the use of worst case bounds on $g_i(t,x)$ as given by equation (2.52), $\dot{V}$ can be obtained as

$$\dot{V} \leq \sum_{i=0}^{N} -x_i^T Q_i x_i + 2||P_i x_i|| \sum_{p=1}^{P_i} \sum_{k=0}^{N} \beta_{i,pk} ||x_k||^p \tag{2.54}$$

$$\dot{V} \leq \sum_{i=0}^{N} -\lambda_{min}(Q_i) ||x_i||^2 + \sum_{i=0}^{N} 2||P_i|| ||x_i|| \sum_{p=1}^{P_i} \sum_{k=0}^{N} \beta_{i,pk} ||x_k||^p \tag{2.55}$$

Three different cases are considered depending upon how the second term in the right-side of inequality (2.55) is simplified.

Method A1:

Define new positive constants as

$$\sigma_{ip} = max_k(\beta_{i,pk}) \tag{2.56}$$

$$\alpha_{ip} = \begin{cases} ||P_i|| \sigma_{ip} & \text{if } ||P_i|| \sigma_{ip} \geq 1; \\ 1 & \text{if } 0 < ||P_i|| \sigma_{ip} < 1. \end{cases} \tag{2.57}$$

The second term in the right-side of inequality (2.55) can be simplified as

$$\sum_{i=0}^{N} 2||P_i||||x_i|| \sum_{p=1}^{P_i} \sum_{k=0}^{N} \beta_{i,pk}||x_k||^p \leq 2 \sum_{p=1}^{P_i} \sum_{i=0}^{N} \alpha_{ip}||x_i|| \sum_{k=0}^{N} ||x_k||^p \tag{2.58}$$

$$\leq 2 \sum_{p=1}^{P_i} \sum_{i=0}^{N} \alpha_{ip}||x_i|| \sum_{k=0}^{N} ||x_k||^p \tag{2.59}$$

$$\leq 2(N+1) \sum_{p=1}^{P_i} \sum_{i=0}^{N} (\alpha_{ip})^2 ||x_i||^{p+1} \tag{2.60}$$

To obtain the last equation, Claim 2.1.2 is used. Now, (2.55) can be simplified as

$$\dot{V} \leq \sum_{i=0}^{N} -\lambda_{min}(Q_i)||x_i||^2 + 2(N+1) \sum_{p=1}^{P_i} \sum_{i=0}^{N} (\alpha_{ip})^2 ||x_i||^{p+1} \tag{2.61}$$

$$\leq \sum_{i=0}^{N} -||x_i||^2 \left(\lambda_{min}(Q_i) - 2(N+1) \sum_{p=1}^{P_i} (\alpha_{ip})^2 ||x_i||^{p-1} \right) \tag{2.62}$$

Assume that there exists known $R_i \in \mathbb{R}$ such that

$$||x_i(t_0)|| \leq R_i \tag{2.63}$$

Use gains k_i such that minimum eigenvalue of positive definite matrix Q_i in (2.53) satisfies

$$\lambda_{min}(Q_i) = 2(N+1) \sum_{p=1}^{P_i} (\alpha_{ip})^2 (R_i)^{p-1} + \gamma_i \tag{2.64}$$

where γ_i is any positive real constant.

Method A2:

Define a new constant as

$$\sigma_p = max_{i,k}(||P_i||\beta_{i,pk}) \tag{2.65}$$

The second term in the right-side of inequality (2.55) can be simplified as

$$\sum_{i=0}^{N} 2||P_i||||x_i|| \sum_{p=1}^{P_i} \sum_{k=0}^{N} \beta_{i,pk}||x_k||^p \leq 2 \sum_{p=1}^{P_i} \sigma_p \sum_{i=0}^{N} ||x_i|| \sum_{k=0}^{N} ||x_k||^p$$

$$\leq 2(N+1) \sum_{p=1}^{P_i} \sigma_p \sum_{i=0}^{N} ||x_i||^{p+1} \tag{2.66}$$

To obtain the last inequality, lemma 2.1.3 is used. Hence, inequality (2.55) can be simplified as

$$\dot{V} \leq \sum_{i=0}^{N} -\lambda_{min}(Q_i)||x_i||^2 + 2(N+1)\sum_{p=1}^{P_i}\sigma_p \sum_{i=0}^{N}||x_i||^{p+1} \tag{2.67}$$

$$\leq \sum_{i=0}^{N} -||x_i||^2 \left(\lambda_{min}(Q_i) - 2(N+1)\sum_{p=1}^{P_i}\sigma_p||x_i||^{p-1}\right) \tag{2.68}$$

Assume that there exists known $R_i \in \mathbb{R}$ such that

$$||x_i(t_0)|| \leq R_i \tag{2.69}$$

Use gains k_i such that minimum eigenvalue of positive definite matrix Q_i in (2.53) satisfies

$$\lambda_{min}(Q_i) = 2(N+1)\sum_{p=1}^{P_i}\sigma_p(R_i)^{p-1} + \gamma_i \tag{2.70}$$

Method A3:

Define a new constant as

$$\sigma = max_{i,p,k}(||P_i||\beta_{i,pk}) \tag{2.71}$$

$$\sum_{i=0}^{N} 2||P_i||||x_i||\sum_{p=1}^{P_i}\sum_{k=0}^{N}\beta_{i,pk}||x_k||^p \leq 2\sigma\sum_{i=0}^{N}||x_i||\sum_{p=1}^{P_i}\sum_{k=0}^{N}||x_k||^p$$

$$\leq 2(N+1)\sigma\sum_{i=0}^{N}\sum_{p=1}^{P_i}||x_i||^{p+1} \tag{2.72}$$

To obtain the last inequality, lemma 2.1.3 is used. Hence, inequality (2.55) can be simplified as

$$\dot{V} \leq \sum_{i=0}^{N} -\lambda_{min}(Q_i)||x_i||^2 + 2(N+1)\sigma\sum_{i=0}^{N}\sum_{p=1}^{P_i}||x_i||^{p+1}$$

$$\leq \sum_{i=0}^{N} -||x_i||^2 \left(\lambda_{min}(Q_i) - 2(N+1)\sigma\sum_{p=1}^{P_i}||x_i||^{p-1}\right) \tag{2.73}$$

Again, with positive real γ_i, condition on Q_i can be given as

$$\lambda_{min}(Q_i) = 2(N+1)\sigma\sum_{p=1}^{P_i}||R_i||^{p-1} + \gamma_i \tag{2.74}$$

Case B: ARE approach

In this case the nonlinear interconnections $g_i(t,x)$ are assumed to be bounded as

$$g_i(t,x)^T g_i(t,x) \leq \sum_{p=2}^{P_i} \sum_{k=0}^{N} \delta_{i,pk} ||x_k||^p \tag{2.75}$$

where $\delta_{i,pk}$ are real numbers. Derivative of the Lyapunov function candidate in equation (2.51) can be written as

$$\dot{V} = \sum_{i=0}^{N} x_i^T [\bar{A}_i^T P_i + P_i \bar{A}_i] x_i + g_i^T(t,x) P_i x_i + x_i^T P_i g_i(t,x) \tag{2.76}$$

Using the inequality

$$M^T N + N^T M \leq M^T M + N^T N \tag{2.77}$$

the derivative of the Lyapunov function candidate satsfies

$$\dot{V} \leq \sum_{i=0}^{N} x_i^T [\bar{A}_i^T P_i + P_i \bar{A}_i] x_i + x_i^T (P_i P_i) x_i + g_i^T(t,x) g_i(t,x) \tag{2.78}$$

Now, using worst case bounds on the $g_i(t,x)$ from equation (2.75),

$$\dot{V} \leq \sum_{i=0}^{N} x_i^T [\bar{A}_i^T P_i + P_i \bar{A}_i + P_i P_i] x_i + \sum_{p=2}^{P_i} \sum_{k=0}^{N} \delta_{i,pk} ||x_k||^p \tag{2.79}$$

Thus, for positive definite P_i and Q_i, it is required to solve the ARE given as

$$\bar{A}_i^T P_i + P_i \bar{A}_i + P_i P_i + Q_i = 0 \tag{2.80}$$

If the positive definite solution P_i to the ARE exists, then equation (2.79) can be written as

$$\dot{V} \leq \sum_{i=0}^{N} -\lambda_{min}(Q_i) ||x_i||^2 + \sum_{p=2}^{P_i} \sum_{k=0}^{N} \delta_{i,pk} ||x_k||^p \tag{2.81}$$

$$\leq \sum_{i=0}^{N} -||x_i||^2 (\lambda_{min}(Q_i) - \sum_{p=2}^{P_i} \sum_{k=0}^{N} \delta_{k,pi} ||x_i||^{p-2}) \tag{2.82}$$

Let us assume that there exists known $R_i \in \mathbb{R}$ such that

$$||x_i(t_0)|| \leq R_i \tag{2.83}$$

Thus, the condition on positive definite matrix Q_i is

$$\lambda_{min}(Q_i) = \sum_{p=2}^{P_i} \sum_{k=0}^{N} \delta_{k,pi} (R_i)^{p-2} + \gamma_i \tag{2.84}$$

where, γ_i is any positive real constant.

Using lemma 2.1.1 and lemma 2.1.2, condition for existence of positive definite solutions to the ARE (2.80) can be given as

$$\min_{\omega \in R} \sigma_{min}(\bar{A_i} - j\omega I) > \sqrt{\lambda_{min}(Q_i)} > 0 \tag{2.85}$$

For case A (each of the methods A1, A2 and A3) and case B, if condition on Q_i, established for each one of them, is achieved, then $\dot{V}$ becomes

$$\dot{V} \leq -\sum_{i=0}^{N} \gamma_i ||x_i||^2 \tag{2.86}$$

Thus, we conclude that any trajectory on or inside the region defined for all i by $\Omega = \{x_i(t) : ||x_i(t)|| \leq R_i\}$ can be *exponentially* stabilized to zero.

With methods A2 and A3, design of controller for the i-th subsystem requires knowledge of $||P_i||$ from all other subsystems. Methods A1 and case B do not require the knowledge of $||P_i||$ from other subsystems, but they involve square of the parameter α_{ip}. The choice of method A1, method A2 or method A3 should be done according to the availability of information about interconnections. Method A2 results in less conservative bounds on $g_i(t,x)$ than the third method, hence superior in this respect. But, in some applications, interconnection parameters may involve uncertainties such that the maximum of $||P_i||\beta_{i,pk}$ is known over i, p and k, where method A3 may be useful. Method B involves solving of an ARE. A control engineer, who has physical insight into the particular problem, can creatively use one of the above methods in such a way that the resulting control gains are within physical saturation limits.

Remark 2.3.1 *Nonadaptive state regulation is just a special case of the adaptive state regulation. If the system matrices, A_i are known then one can directly compute the gains k_i accurately and no adaptation is required. Moreover, in case of nonadaptive state regulation, one can relax the condition that input u_i be a scalar, the stability proof holds even for $u_i(t) \in \mathbb{R}^{m_i}$ where m_i is number of inputs.*

The above stability analysis reveals that if the initial conditions of all the subsystems are known, then the first order local state feedback controllers are sufficient to achieve stability of entire large-scale system whose subsystems are linked with each other through interconnections with any order greater than one. The region of attraction for each subsystem can be increased by the choice of feedback gain vector for that subsystem alone. The proposed decentralized controller can guarantee robust stability in the presence of parametric as well as structural uncertainties and interconnection perturbations by considering worst case of these uncertainties in the bounds of g_i.

CHAPTER 3

EXPERIMENTAL PLATFORM AND DYNAMIC MODEL

Figure. 3.1 shows the web handling setup used for the experimentation. Figure. 3.2 shows a sketch of the experimental platform and the web path for conducting experiments with the proposed controllers. The line mimics most of the features of an industrial web process line, and is developed with the aim of open-architecture design that allows for modifying the line to conform to test specific research experimentation. The line contains a number of different stations and a number of driven rollers, as pictured in Fig. 3.1 and illustrated schematically in Fig. 3.2. For the experimentation, the web is threaded through four driven rollers M0 to M3 as shown in Fig. 3.2, and through many other idle rollers throughout the line to facilitate transport of the web from the unwind to rewind.

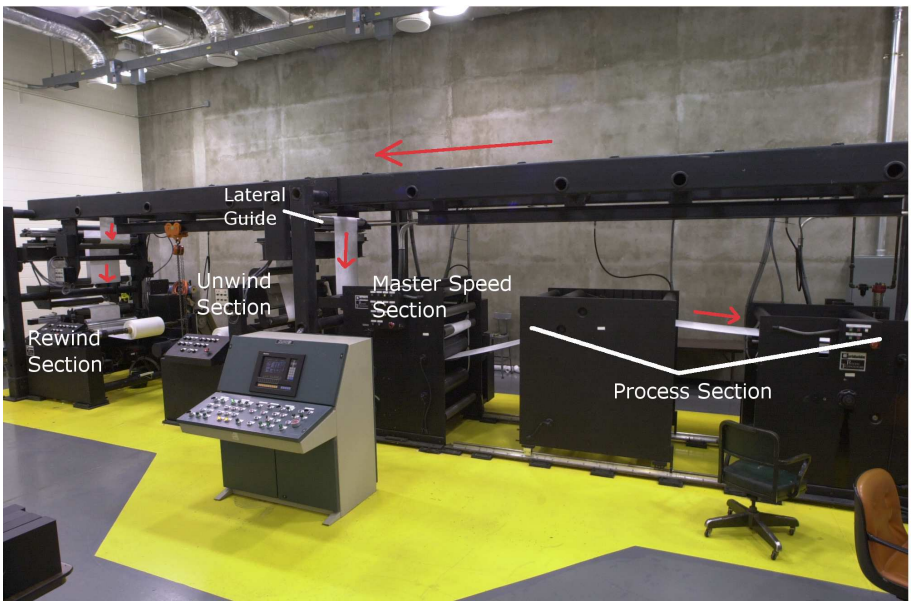


Figure 3.1: Experimental Platform

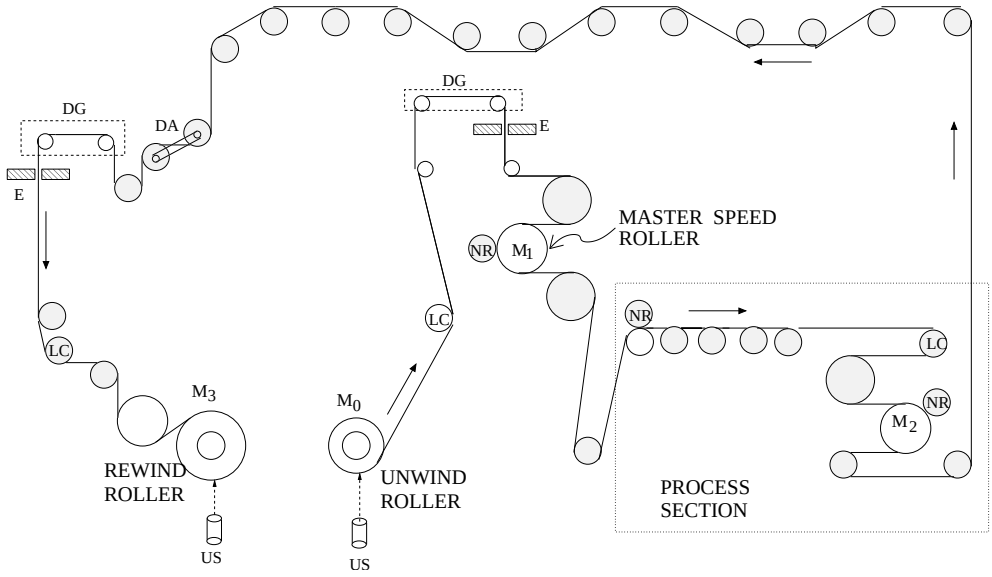


Figure 3.2: Experimental Platform

The nip rollers (denoted by NR), which are pneumatically driven, are used to maintain contact of the web with the driven rollers. Two controlled lateral guides (guides are denoted by DG and the web edge sensors by E), near unwind and rewind sections, respectively, are used to maintain the lateral position of the web on the rollers during web transport. Three-phase induction motors, with 30 *hp* capacity, from Rockwell Automation are used to drive the unwind and rewind rolls, whereas master speed and process section rollers are driven by 15 *hp* induction motors. The motor drive system, the real-time architecture which includes micro-processors, I/O cards, and real-time software (AUTOMAX) is from Rockwell Automation. In the experimental platform, each motor is driven by a dedicated vector controller. The feedback control loops in the driver or vector controller are very fast and hence they have little effect on the transient response of entire plant, and hence dynamics of the vector controller is taken as unity. Reference torque and flux signals for each of the vector controllers are generated by corresponding microprocessors, which are part of the AUTOMAX distributed control system. To implement the desired control algorithms, programs in AUTOMAX can be modified using an off line personal computer, and then uploaded to the dedicated microprocessors. Similar to a typical industrial web line

control system, microprocessors used in the experimental platform are located in two racks: A00 and A01. Rack A00 has microprocessors and vector control drives for the rewind roll (M3) and process section roller (M2). Rack A01 has microprocessors and vector control drives for the unwind roll (M0) and master speed roller (M1). Depending upon the number of process sections, an actual industrial setup may have a large number of such racks. Decentralized controllers are often preferred, and mostly used, by the web handling industry due to the ease of tuning individual stations without considering the cumulative effect of the entire process line and provide reliable operation of the process line in the event of occasional actuator and sensor malfunctions. A general goal is to design control algorithms to minimize data communication between microprocessors and to reduce network complexity of the distributed control system.

It is common in the web handling industry to divide the process line into several tension zones by calling the span between two successive driven rollers as a tension zone, thus ignoring the effect of the free rollers that lie between two driven rollers. Since the free roller dynamics have an effect on the web tension during the transients due to acceleration/deceleration of the web line and negligible effect during steady state operation, the assumption that the free rollers do not contribute to web dynamics during steady state operation is reasonable, which is explained next.

From Fig. 3.2, notice that the web is threaded through driven as well as idle rollers. Idle rollers act as energy consuming elements in the transport of web from an unwind roll to a rewind roll through various web spans. Idle rollers consume energy during an acceleration/deceleration phase due to the inertia of each idle roller. Bearing friction in the rollers is another constant source of energy dissipation. The power required to rotate each idle roller is the torque acting on it multiplied by its angular velocity, which is provided by the web. Assume that an angular velocity (RPM) of an idle roller is such that the linear velocity on its surface is same as that of a web moving over it, which is possible if there is no slip. Now the necessary torque will be given by the rise in tension when the web passes

over that idle roller. This is explained in Fig. 3.3. The upstream and downstream tensions

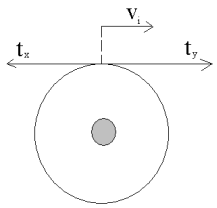


Figure 3.3: Rise in Tension over idle roller

are represented by t_x and t_y respectively, respectively, b_i is the coefficient of friction, υ_i is the linear velocity on the surface of the idle roller. Assuming the idle roller shown in Fig. 3.3 is accelerating, its velocity dynamics is given as

$$\frac{J_i}{R_i}\dot{\upsilon}_i = -b_i\upsilon_i + (t_y - t_x)R_i \tag{3.1}$$

Equation (3.1) clearly shows that the increase in tension from t_x to t_y is because of inertia as well as bearing friction of the roller. If it is assumed that bearing friction and inertia are too small then the force required to drive the roller $(t_x - t_y)$ will also be very small. This means that t_x is approximately equal to t_y. This assumption is valid in steady state operation and leads to considerable simplification by neglecting idle rollers and keeping only driven rollers. This assumption will be used in developing the dynamic model in this section. Further, following assumptions are also made

- The cross-sectional area of the web is uniform through out the individual process section.
- The web is perfectly elastic, i.e., stress is linearly proportional to strain.
- The web is homogeneous in un-stretched condition so that all physical properties like density, modulus of elasticity remain constant.

- The web does not slip on the driver rollers. In actual web processing machines nip rollers are used on the driven rollers to avoid web-slipping.

- The rollers do not show whirling effect. That is the center of mass of each rotating element lies exactly on axis of rotation.

Because of all these un-modeled processes, developed model does not reflect exact web handling dynamics but has structural uncertainties involved in it. Additionally, parameters like elasticity constant (E), coefficient of friction (b_f), web cross-sectional area (A) are not known accurately but only nominal values are known. Hence it is required to design a controller that is not only strictly decentralized but also robust against structural as well as parametric uncertainties. Figure 3.4 shows a web line with three tension zones; the

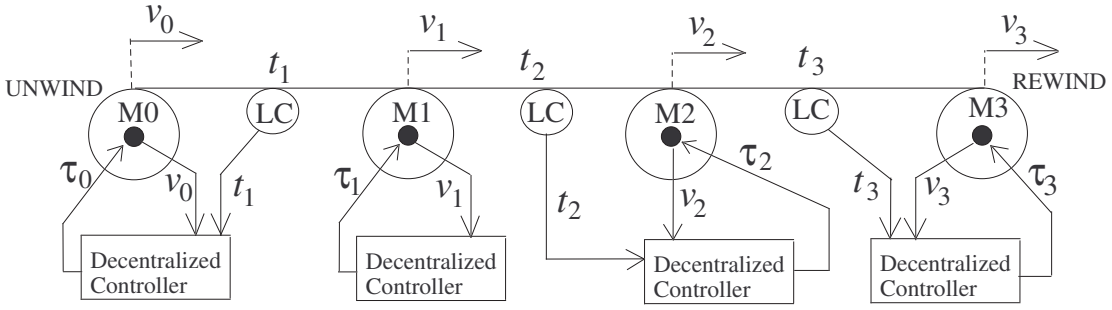


Figure 3.4: Simplified high speed web line with decentralized control scheme

line consists of the unwind/rewind rolls and two intermediate driven rollers. In the figure, LC denotes the load cell roller, which is mounted on a pair of load-cells on either side for measuring web tension. The driving motors are represented by M_i for $i = 0,1,2,3$, τ_i represents input torque from the i-th motor, v_i represents the transport velocity of the web on the i-th roller, and t_i represents web tension in the span between $(i-1)$-th and i-th driven rollers. There are four sections in the web line shown in Fig. 3.4, which are the unwind section, master speed roller, process section, and rewind section. The name master speed roller is given to a driven roller which sets the reference web transport speed for the entire web line, and is generally the first driven roller upstream of the unwind roll in

almost all web process lines; the purpose of the master speed roller is to regulate web line speed and is not used to regulate tension in the spans adjacent to it. The unwind/rewind rolls release/accumulate material to/from the processing section of the web line. Thus, their radii and inertia are time-varying. The dynamics of each of the four sections is presented in the following.

3.1 Dynamic Model

Unwind section: A cross-sectional view of the unwind roll is shown in Fig. 3.5. The associated local state variables for the unwind section are web velocity v_0 and tension t_1. At any instant of time t, the effective inertia $J_0(t)$ of the unwind section is given by

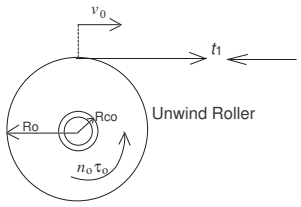


Figure 3.5: Cross-sectional view of the unwind roll

$$J_0(t) = n_0^2 J_{m0} + J_{c0} + J_{w0}(t) \tag{3.2}$$

where n_0 is the gearing ratio between the motor shaft and unwind roll shaft, J_{m0} is the inertia of all the rotating elements on the motor side, which includes inertia of motor armature, driving pulley (or gear), driving shaft, etc., J_{c0} is the inertia of the driven shaft and the core mounted on it, and $J_{wo}(t)$ is the inertia of the cylindrically wound web material on the core. Both J_{m0} and J_{c0} are constants, but the inertia due to cylindrically wound web material, $J_{w0}(t)$, is not constant because the web is continuously released into the process. The inertia, $J_{w0}(t)$, is given by

$$J_{w0}(t) = \frac{\pi}{2} b_w \rho_w (R_0^4(t) - R_{c0}^4) \tag{3.3}$$

where b_w is the web width, ρ_w is the density of the web material, R_{c0} is the radius of the empty core mounted on the unwind roll-shaft, and $R_0(t)$ is the radius of the material roll.

From Fig. 3.5, the velocity dynamics of the unwind roll can be written as

$$\frac{d}{dt}(J_0\omega_0) = t_1R_0 - n_0\tau_0 - b_{f0}\omega_0$$

$$\dot{J}_0\omega_0 + \dot{\omega}_0J_0 = t_1R_0 - n_0\tau_0 - b_{f0}\omega_0 \tag{3.4}$$

where ω_0 is the angular velocity of the unwind roll and b_{f0} is the coefficient of friction in the unwind roll shaft. The rate of change in $J_0(t)$ is only because of the change in $J_{w0}(t)$, and from equation (3.3), the rate of change of $J_0(t)$ is given by

$$\dot{J}_0(t) = \dot{J}_{w0}(t) = 2\pi b_w\rho_w R_0^3\dot{R}_0 \tag{3.5}$$

The transport velocity of the web coming off the unwind roll is related to the angular velocity of the unwind roll by $v_0 = R_0\omega_0$, and hence one can obtain $\dot{\omega}_0$ in terms of v_0 as

$$\dot{\omega}_0 = \frac{\dot{v}_0}{R_0} - \frac{\dot{R}_0v_0}{R_0^2} \tag{3.6}$$

Substitution of (3.5) and (3.6) into the velocity dynamics given by equation (3.4) and simplifying results in

$$\frac{J_0}{R_0}\dot{v}_0 = t_1R_0 - n_0\tau_0 - \frac{b_{f0}}{R_0}v_0 + \frac{\dot{R}_0v_0}{R_0^2}J_0 - 2\pi\rho_w b_w R_0^2\dot{R}_0v_0 \tag{3.7}$$

But the rate of change of radius, $\dot{R}_0$, is a function of the transport velocity v_0 and the web thickness, t_w, and is approximately given by

$$\dot{R}_0 \approx -\frac{t_w}{2\pi}\frac{v_0(t)}{R_0(t)} \tag{3.8}$$

Notice that (3.8) is approximate because the thickness affects the rate of change of radius of the roll after each revolution of the roll; the continuous approximation is valid since the thickness is generally very small. Also, notice that the last term in the velocity dynamics (3.7) is often ignored in the literature under the assumption that the roll radius is slowly time-varying. But in practice, since the web transport velocity is kept constant, the last two

terms in (3.7) are significant as the roll radius becomes smaller. Hence, equation (3.7) can be simplified to

$$\frac{J_0}{R_0}\dot{v}_0 = t_1 R_0 - n_0\tau_0 - \frac{b_{f0}}{R_0}v_0 - \frac{t_w}{2\pi R_0}\left(\frac{J_0}{R_0^2} - 2\pi\rho_w b_w R_0^2\right)v_0^2 \tag{3.9}$$

Dynamic behavior of the web tension, t_1, in the span immediately downstream of the unwind roll is given by

$$L_1\dot{t_1} = AE[v_1 - v_0] + t_0 v_0 - t_1 v_1 \tag{3.10}$$

where L_1 is the length of the web span between unwind roller (M0) and master speed roller (M1), A is the area of cross-section of the web, E is the modulus of elasticity of the web material, and t_0 represents the wound-in tension of the web in the unwind roll.

Master speed roller: The dynamics of the master speed roller is given by

$$\frac{J_1}{R_1}\dot{v_1} = (t_2 - t_1)R_1 + n_1\tau_1 - \frac{b_{f1}}{R_1}v_1 \tag{3.11}$$

Process section: The web tension and web velocity dynamics in the process section are given by

$$L_2\dot{t_2} = AE[v_2 - v_1] + t_1 v_1 - t_2 v_2 \tag{3.12}$$

$$\frac{J_2}{R_2}\dot{v_2} = (t_3 - t_2)R_2 + n_2\tau_2 - \frac{b_{f2}}{R_2}v_2 \tag{3.13}$$

Rewind section: The web velocity dynamics entering the rewind roll can be determined along similar lines as those presented for the unwind roll. The web tension and velocity dynamics in the rewind section are

$$L_3\dot{t_3} = AE[v_3 - v_2] + t_2 v_2 - t_3 v_3 \tag{3.14}$$

$$\frac{J_3}{R_3}\dot{v_3} = -t_3 R_3 + n_3\tau_3 - \frac{b_{f3}}{R_3}v_3 + \frac{t_w}{2\pi R_3}\left(\frac{J_3}{R_3^2} - 2\pi\rho_w b_w R_3^2\right)v_3^2 \tag{3.15}$$

Equations (3.9) through (3.15) represent the dynamics of the web and rollers for the web line configuration shown in Fig. 3.4. Extension to other web lines can be easily made based on this model. For web process lines that have a series of process sections between

the master speed roller and the rewind roll, then equations (3.12) and (3.13) can be written down for each process section.

The dynamic model given by equations (3.9) through (3.15) is nonlinear and time-varying. For many web process lines, the goal is to transport the web at a specific speed while maintaining a specified tension in each zone.To achieve this goal one has to systematically design the control inputs such that the non-zero set point problem is converted to a regulation problem.

3.2 Equilibrium Inputs and Linearized Dynamics

The control goal is to regulate web tension in each of the tension zones while maintaining the prescribed web transport velocity. To achieve this, first, one systematically has to calculate the constant (or equilibrium) control input required to keep the web line at the forced equilibrium of the reference web tension and web velocity in each of the zones. Then, some additional compensation must be included to provide error convergence in the presence of time varying radius and inertia of the roll. A simple procedure for the calculation of equilibrium control inputs is given, which is easy to understand and implement by practising engineers.

Define the following variables: $T_i = t_i - t_{ri}$ and $V_i = v_i - v_{ri}$, where t_{ri} and v_{ri} are tension and velocity references, respectively, T_i and V_i are the variations in tension and velocity, respectively, around their reference values, τ_{ieq} as the control input that maintains the forced equilibrium at the reference values, and $U_i = \tau_i - \tau_{ieq}$ is the variation of the control input. Define the state vector for the unwind section as $x_0^T = [T_1, V_0]$ and the state for the master speed roller as $x_1 = V_1$. After master speed section, define the state vector for the j-th subsystem as $x_j^T = [T_j, V_j]$ for $j = 2, 3$. In the following, equilibrium control inputs and reference velocities are determined for each driven roll/roller based on the reference velocity of the master speed roller and reference tension in each tension zone.

3.2.1 Unwind section: Subsystem 0

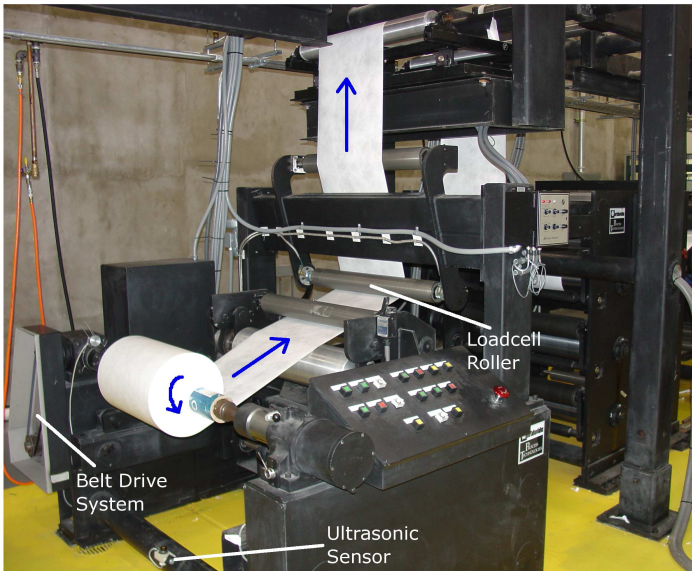


Figure 3.6: Unwind section

The velocity dynamics in the unwind section can be written as

$$\frac{J_0}{R_0}(\dot{V}_0+\dot{v}_{r0}) = (T_1+t_{r1})R_0 - n_0(U_0+\tau_{0eq}) - \frac{b_{f0}}{R_0}(V_0+v_{r0}) - \frac{t_w}{2\pi R_0}\left(\frac{J_0}{R_0^2} - 2\pi\rho_w b_w R_0^2\right)(V_0+v_{r0})^2 \tag{3.16}$$

At the forced equilibrium, assuming the variations, T_1 and V_0, and their derivatives as zero, the input that maintains this equilibrium is given by

$$\tau_{0eq} = -\frac{b_{f0}}{n_0 R_0}v_{r0} + \frac{R_0}{n_0}t_{r1} - \frac{t_w}{2\pi n_0 R_0}\left(\frac{J_0}{R_0^2} - 2\pi\rho_w b_w R_0^2\right)v_{r0}^2 - \frac{J_0}{n_0 R_0}\dot{v}_{r0} \tag{3.17}$$

The web tension dynamics in the unwind section can be written as

$$L_1(\dot{T}_1+\dot{t}_{r1}) = AE[(V_1+v_{r1})-(V_0+v_{r0})] + t_0(V_0+v_{r0}) - (T_1+t_{r1})(V_1+v_{r1}) \tag{3.18}$$

From (3.18), assuming $\dot{T}_1$, T_1 and V_i as zero at the forced equilibrium, the relationship between the reference velocities v_{r0} and v_{r1} is given by

$$v_{r0} = \frac{AE-t_{r1}}{AE-t_0}v_{r1} - \frac{L_1}{AE-t_0}\dot{t}_{r1} \tag{3.19}$$

Note that t_0 is the tension in the web, which is already wound on the unwind roller. Tension t_0 is not a controlled variable and may vary in different layers of the unwind roll. By choosing the reference velocity of the unwind roll as a function of the master speed roller as given by (3.19), the variational dynamics of the unwind section can be written as

$$L_1\dot{T}_1 = AE[(V_1 - V_0)] + t_0V_0 - T_1v_{r1} - V_1t_{r1} - T_1V_1 \tag{3.20}$$

$$\frac{J_0}{R_0}\dot{V}_0 = T_1R_0 - n_0U_0 - \frac{b_{f0}}{R_0}V_0 - \frac{t_w}{2\pi R_0}\left(\frac{J_0}{R_0^2} - 2\pi\rho_w b_w R_0^2\right)(V_0^2 + 2v_{r0}V_0) \tag{3.21}$$

which can be arranged in the desired form as

$$\dot{x}_0 = \begin{bmatrix} \dot{T}_1 \\ \dot{V}_0 \end{bmatrix} = A_0x_0 - B_0U_0 - B_0f_0(V_0) + g_0(x) \tag{3.22}$$

where

$$f_0(V_0) = \frac{t_w}{2\pi n_0R_0}\left(\frac{J_0}{R_0^2} - 2\pi\rho_w b_w R_0^2\right)(V_0^2 + 2v_{r0}V_0)$$

$$A_0 = \begin{bmatrix} -v_{r1}/L_1 & (t_0 - AE)/L_1 \\ R_0^2/J_0 & -b_{f0}/J_0 \end{bmatrix}, B_0 = \begin{bmatrix} 0 \\ \frac{n_0R_0}{J_0} \end{bmatrix}, g_0(x) = \begin{bmatrix} \frac{(AE - t_{r1})V_1 - T_1V_1}{L_1} \\ 0 \end{bmatrix} + \Delta A_0x_0$$

where $g_0(x)$ is given as:

$$g_0(x) = \begin{bmatrix} \frac{AE - t_{r1}}{L_1}V_1 - \frac{T_1V_1}{L_1} \\ 0 \end{bmatrix} + \Delta A_0x_0 \tag{3.23}$$

And the interconnection function $g_0(x)$ can be bounded as:

$$\begin{aligned} ||g_0(x)|| &\leq \left(\frac{AE_{max} - t_r}{L_{1min}}\right)|V_1| + \frac{T_1V_1}{L_{1min}} + ||(\Delta A_0)_{max}x_0|| \\ &\leq ||(\Delta A_0)_{max}x_0|| + \left(\frac{AE_{max} - t_r}{L_{1min}}\right)|V_1| + \frac{T_1^2}{2L_{1min}} + \frac{V_1^2}{2L_{1min}} \\ &= \beta_{0,10}||x_0|| + \beta_{0,20}||x_0||^2 + \beta_{0,11}||x_1|| + \beta_{0,21}||x_1||^2 \end{aligned} \tag{3.24}$$

where

$$\beta_{0,10} = ||(\Delta A_0)||$$
$$\beta_{0,20} = \frac{1}{2L_{1min}}$$
$$\beta_{0,11} = \left(\frac{AE_{max} - t_r}{L_{1min}}\right)$$
$$\beta_{0,21} = \frac{1}{2L_{1min}}$$

Note that the matrix A_0 is a function of time, because R_0 and J_0 are functions of time. Moreover, considering the physical nature, wound in tension t_0 cannot be equal to AE and $R_0 \neq 0$, hence for all time (A_0, B_0) is controllable. It may be noted that the term $g_0(x)$ involving interconnecting and nonlinear terms does not satisfy matching condition. Assuming that the product of variations T_1V_1 is negligible in $g_0(x)$, the linearized dynamics can be written as

$$\dot{x}_0 = \begin{bmatrix} \dot{T}_1 \\ \dot{V}_0 \end{bmatrix} = A_0 x_0 - B_0 U_0 - B_0 f_0(V_0) + \sum_{j\neq 0, j=1}^{N} A_{0j} x_j \tag{3.25}$$

where

$$A_{01} = \left[\frac{AE - t_{r1}}{L_1}, \quad 0\right]^T$$

A_{02} and A_{03} are null matrices.

3.2.2 Master speed roller: Subsystem 1

This subsystem has one state $x_1 = V_1$ and the velocity error dynamics can be obtained as

$$J_1\dot{v}_1 = (t_2 - t_1){R_1}^2 + n_1\tau_1 R_1 - b_{f1}v_1 \tag{3.26}$$

$$J_1(\dot{V}_1 + \dot{v}_{r1}) = -(T_1 + t_{r1}){R_1}^2 + n_1(U_1 + \tau_{1eq})R_1 - b_{f1}(V_1 + v_{r1}) \tag{3.27}$$

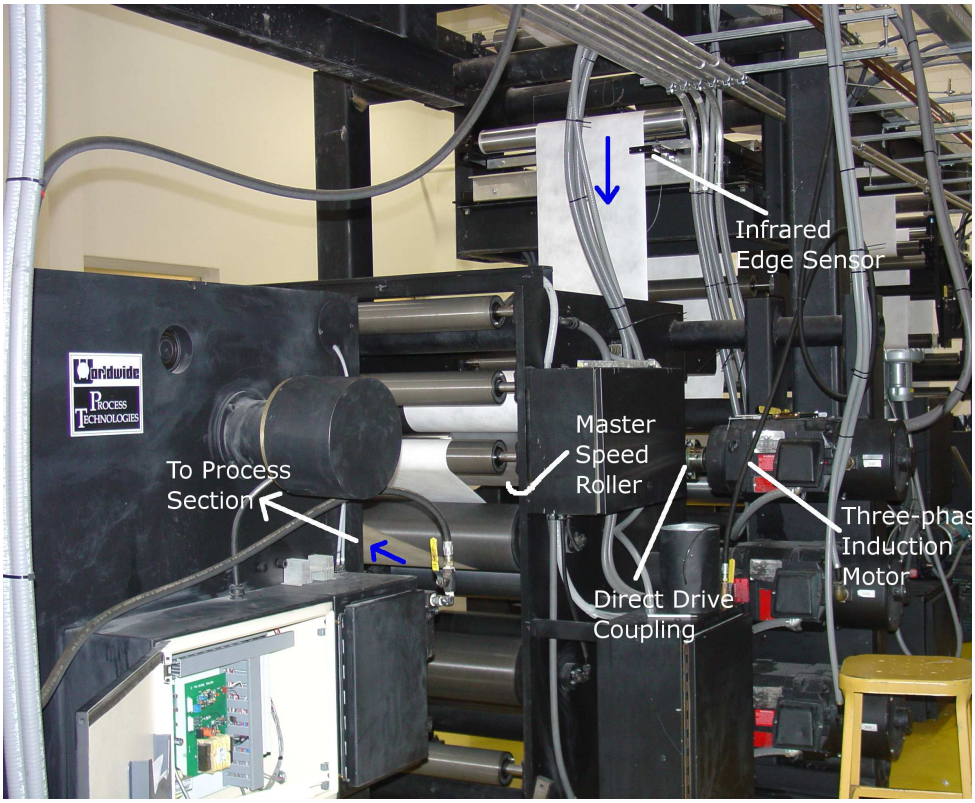


Figure 3.7: Master speed section

Equilibrium input, u_{1eq}, can be found out using stationary equilibrium condition as:

$$0 = -t_{r1}{R_1}^2 + n_1\tau_{1eq}R_1 - b_{f1}v_{r1} - J_1\dot{v}_{r1} \tag{3.28}$$

$$\tau_{1eq} = \frac{b_{f1}}{n_1R_1}v_{r1} + \frac{R_1}{n_1}t_{r1} + \frac{J_1}{n_1R_1}\dot{v}_{r1} \tag{3.29}$$

Using equilibrium solution in equation (3.27), we get the velocity error dynamics as

$$J_1\dot{V}_1 = -T_1R_1^2 + n_1U_1R_1 - b_{f1}V_1 \tag{3.30}$$

which can be arranged in desired form as

$$\dot{x}_1 = \dot{V}_1 = A_1x_1 + B_1U_1 + g_1(x) \tag{3.31}$$

where

$$A_1 = -\frac{b_{f1}}{J_1}, \quad B_1 = \frac{n_1R_1}{J_1}, \quad g_1(x) = (T_2 - T_1)\frac{R_1^2}{J_1} + \Delta A_1V_1 \tag{3.32}$$

The interconnection function $g_1(x)$ is bounded as

$$||g_1(x)|| \leq \beta_{1,10}||x_0|| + \beta_{1,11}||x_1|| \tag{3.33}$$

where

$$\beta_{1,10} = \left(R_1^2/J_1\right)_{max}, \;\; \beta_{1,11} = ||\Delta A_1||_{max} = \Delta_{max}(b_{f1}/J_1)$$

Note that the velocity dynamics for master speed roller does not involve any nonlinear interconnection terms. Hence, in terms of the linear interconnection matrix, the same velocity dynamics for master speed roller can be arranged as

$$\dot{x}_1 = \dot{v}_1 = A_1 x_1 + B_1 U_1 + \sum_{j=0, j\neq 1}^{3} A_{1j} x_j \tag{3.34}$$

where

$$A_{10} = \left[\frac{-R_1^2}{J_1}, 0\right], \; A_{12} = \left[\frac{R_1^2}{J_1}, 0\right], \; A_{13} = [0, 0]$$

3.2.3 Process section: Subsystem 2

The state associated with this system is

$$x_2 = \begin{bmatrix} T_2 \\ V_2 \end{bmatrix} = \begin{bmatrix} t_2 - t_{r2} \\ v_2 - v_{r2} \end{bmatrix} \tag{3.35}$$

where v_{r2} is reference velocity. Let control input be $\tau_2 = U_2 + \tau_{2eq}$ and at equilibrium point:

$$v_{r2} = \left(\frac{AE - t_{r1}}{AE - t_{r2}}\right) v_{r1} - \frac{L_2}{AE - t_{r2}} \dot{t}_{r2} \tag{3.36}$$

$$\tau_{2eq} = \frac{b_{f2}}{n_2 R_2} v_{r2} - \frac{R_2}{n_2}(t_{r3} - t_{r2}) + \frac{J_2}{n_2 R_2} \dot{v}_{r2} \tag{3.37}$$

With this equilibrium input τ_{2eq} and reference velocity v_{r2} , error dynamics for third subsystem can be obtained as:

$$\dot{x}_2 = \begin{bmatrix} \dot{T}_2 \\ \dot{V}_1 \end{bmatrix} = A_2 x_1 + B_2 U_1 + g_2(x) \tag{3.38}$$

where

$$A_2 = \begin{bmatrix} -v_{r2}/L_2 & (AE - t_{r2})/L_2 \\ -R_2^2/J_2 & -b_{f2}/J_2 \end{bmatrix} B_2 = \begin{bmatrix} 0 \\ \frac{n_2 R_2}{J_2} \end{bmatrix} \tag{3.39}$$

$$g_2(x) = \begin{bmatrix} \frac{t_{r1}-AE}{L_2} V_1 + \frac{v_{r1}}{L_2} T_1 + \frac{T_1 V_1}{L_2} - \frac{T_2 V_2}{L_2} \\ \frac{R_2^2}{J_2} T_3 \end{bmatrix} \tag{3.40}$$

Hence, interconnection function $g_2(x)$ can be bounded as:

$$||g_2(x)|| \leq \sum_{p=1}^{2} \sum_{k=1}^{N} \beta_{2,pk} ||x_k||^p \tag{3.41}$$

where

$$\begin{aligned} \beta_{2,10} &= (v_{r1}/L_2)_{max}, & \beta_{2,20} &= 0.5/L_{2min} \\ \beta_{2,11} &= (AE/L_2)_{max}, & \beta_{2,21} &= 0.5/L_{2min} \\ \beta_{2,12} &= ||\Delta A_2||_{max}, & \beta_{2,22} &= 0.5/L_{2min} \\ \beta_{2,13} &= \left(R_2^2/J_2\right)_{max} \end{aligned} \tag{3.42}$$

Assuming that the product of variations $T_i V_i$ is negligible in $g_2(x)$, the linearized dynamics can be written as

$$\dot{x}_2 = \begin{bmatrix} \dot{T}_2 \\ \dot{V}_2 \end{bmatrix} = A_2 x_2 + B_2 U_2 + \sum_{j \neq 2, j=0}^{N} A_{2j} x_j \tag{3.43}$$

where

$$A_{20} = \begin{bmatrix} \frac{v_{r1}}{L_2} & 0 \\ 0 & 0 \end{bmatrix}, \; A_{21} = \begin{bmatrix} \frac{t_{r1}-AE}{L_2}, 0 \end{bmatrix}^T, \; A_{23} = \begin{bmatrix} 0 & 0 \\ \frac{R_2^2}{J_2} & 0 \end{bmatrix}$$

3.2.4 Rewind section: Subsystem 3

The state associated with the rewind section is

$$x_3 = \begin{bmatrix} T_3 \\ V_3 \end{bmatrix} = \begin{bmatrix} t_3 - t_{r3} \\ v_3 - v_{r3} \end{bmatrix} \tag{3.44}$$

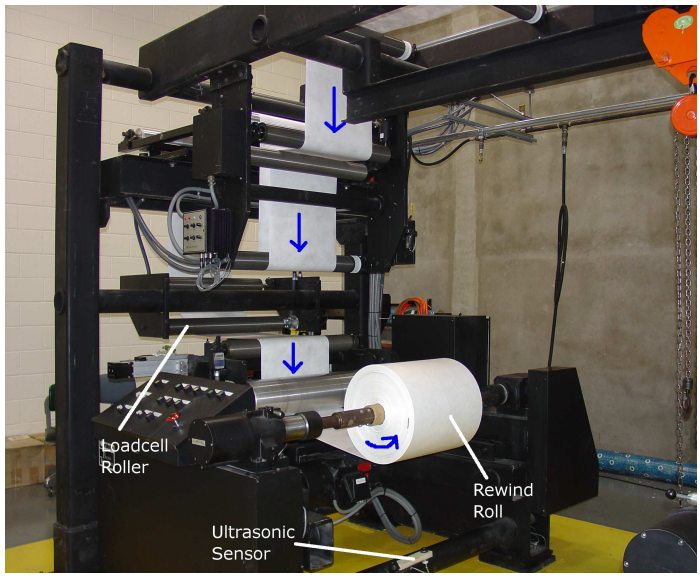


Figure 3.8: Rewind section

where v_{r3} is reference velocity. Let control input be $\tau_3 = U_3 + \tau_{3eq}$ where

$$v_{r3} = \left(\frac{AE - t_{r1}}{AE - t_{r3}}\right) v_{r1} - \frac{L_3}{AE - t_{r3}} \dot{t}_{r3} \tag{3.45}$$

$$\tau_{3eq} = \frac{b_{f3}}{n_3 R_3} v_{r3} + \frac{R_3}{n_3} t_{r3} - \frac{t_w}{2\pi n_3 R_3} \left(\frac{J_3}{R_3^2} - 2\pi\rho_w b_w R_3^2\right) v_{r3}^2 + \frac{J_3}{n_3 R_3} \dot{v}_{r3} \tag{3.46}$$

With this equilibrium input τ_{3eq} and reference velocity v_{r3} , error dynamics for fourth subsystem can be obtained as

$$\dot{x}_3 = \begin{bmatrix} \dot{T}_3 \\ \dot{V}_3 \end{bmatrix} = A_3 x_3 + B_3 U_3 + B_3 f_3(V_3) + g_3(x) \tag{3.47}$$

where

$$A_3 = \begin{bmatrix} -v_{r3}/L_3 & (AE - t_{r3})/L_3 \\ -R_3^2/J_3 & -b_{f3}/J_3 \end{bmatrix} \quad B_3 = \begin{bmatrix} 0 \\ \frac{n_3 R_3}{J_3} \end{bmatrix} \tag{3.48}$$

$$f_3(V_3) = \frac{t_w}{2\pi n_3 R_3} \left(\frac{J_3}{R_3^2} - 2\pi\rho_w b_w R_3^2\right) (V_3^2 + 2 v_{r3} V_3)$$

$$g_3(x) = \begin{bmatrix} \frac{t_{r2} - AE}{L_3} V_2 + \frac{T_2 V_2}{L_3} - \frac{T_3 V_3}{L_3} + \frac{v_{r2}}{L_3} T_2 \\ 0 \end{bmatrix} \tag{3.49}$$

Now, $g_3(x)$ can be bounded as

$$||g_3(x)|| \leq \sum_{p=1}^{2} \sum_{k=1}^{N} \beta_{3,pk} ||x_k||^p \tag{3.50}$$

where

$$\begin{aligned} &\beta_{3,12} = ||c_{3,12}|| \quad \text{where,} \quad c_{3,12} = [(v_{r2}/L_3)_{max}, (AE/L_3)_{max}] \\ &\beta_{3,13} = ||\Delta A_3||_{max}, \qquad \beta_{3,22} = 0.5/L_{3min} \qquad \beta_{3,23} = 0.5/L_{3min} \end{aligned} \tag{3.51}$$

Assuming that the product of variations $T_i V_i$ is negligible in $g_3(x)$, the linearized dynamics can be written as

$$\dot{x}_3 = \begin{bmatrix} \dot{T}_3 \\ \dot{V}_3 \end{bmatrix} = A_3 x_3 + B_3 U_3 + B_3 f_3(V_3) + \sum_{j \neq 3, j=0}^{N} A_{3j} x_j \tag{3.52}$$

where

$$A_{32} = \begin{bmatrix} \frac{v_{r2}}{L_3} & \frac{t_{r2} - AE}{L_3} \\ 0 & 0 \end{bmatrix}$$

and A_{30}, A_{31} are null matrices with proper dimensions.

The dynamic model for each section of the web line is obtained via a systematic development of equilibrium conditions. This procedure indicates that velocity references and tension references cannot be chosen arbitrarily. One can choose reference tensions t_{ri} and reference speed for the master speed roller independently. Based on these reference values, velocity references in other subsystems are computed using equations (3.19), (3.36) and (3.45). Now the control objective is to obtain $U_i = U_i(x_i)$ such that the variations, T_i and V_i, converge to zero, which will imply that $t_i \rightarrow t_{ri}$ and $v_i \rightarrow v_{ri}$.

Remark 3.2.1 *The motor shaft and unwind/rewind are connected through a belt-pulley and gear transmission system. Transmission dynamics, which reflects the compliance effect of belt as well as backlash in meshing gears, is ignored in this study. Effective inertia of rotating elements in a motor at load shaft is combined with the roll/roller inertia to obtain J_i in the mathematical model and thus transmission dynamics is taken as unity.*

CHAPTER 4

COMPARATIVE EXPERIMENTAL RESULTS

This chapter presents the application of the decentralized controllers, proposed in Chapter 2, to web processing lines. Linearized dynamic models developed in Chapter 3 are used for the design. Three decentralized controllers are compared through extensive experimentation:

1. Industrial PI controller.
2. Decentralized non-adaptive state feedback controller with inertia compensation.
3. Decentralized adaptive controller with inertia compensation.

Experimental platform, shown in Fig. 1.3, is used for comparative experimentation and validation. For inertia compensation and equilibrium control of unwind and rewind rolls, values of their radii must be calculated at each instant of time. Angular velocity of each driven roller is calculated by differentiating encoder signal from corresponding motor. The angular velocity of unwind (rewind) roll is then used to calculate radius of unwind roll by integrating the following equation

$$\dot{R}_0 \approx -\frac{t_w}{2\pi}\omega_0 \tag{4.1}$$

For the rewind roll, radius is calculated using

$$\dot{R}_3 \approx \frac{t_w}{2\pi}\omega_3 \tag{4.2}$$

The trapezoidal rule is used in the program to carry out numerical integration. At the start of the controller execution, initial value of the roll-radius is read by an ultrasonic sensor

Parameter	FPS units	SI units
EA	2090 lbf	9300 N
t	7 $mils$	0.178 mm
J_1	2 $lb-ft^2$	0.0257 $Kg-m^2$
J_2	2 $lb-ft^2$	0.0257 $Kg-m^2$
R_1	0.339 ft	0.09144 m
R_2	0.339 ft	0.09144 m
b_{fi}	1 $lbf-ft-sec/rad$	1.3558 $N-m-sec/rad$
L_1	20 ft	6.096 m
L_2	33 ft	10.06 m
L_3	67 ft	20.4 m
Web width	1.708 ft	0.52 m

Table 4.1: Nominal values of the parameters

mounted under unwind (rewind) roll. Using the radius of each roller/roll, R_i, and calculated angular velocity ω_i, linear velocity is computed using $v_i = R_i\omega_i$.

Nominal values of all constant parameters involved in the dynamic model are given in the Table 4.1. Web material used is Tyvek, which is a product made by Dupont. Tensile test was done on the web material to determine the value of EA, which is given in the table.

4.1 Industrial PI Controller

In most industrial web process lines, two decentralized PI control loops, as shown in Fig. 4.1, are used. Notice that the output of the tension loop becomes reference velocity error correction for the velocity loop. With PI speed and tension controllers, it was observed that tension did not converge to the desired reference value t_{ri}=24.5 lbf. A sample of real-time tension response for the unwind section is shown in Fig. 4.2.

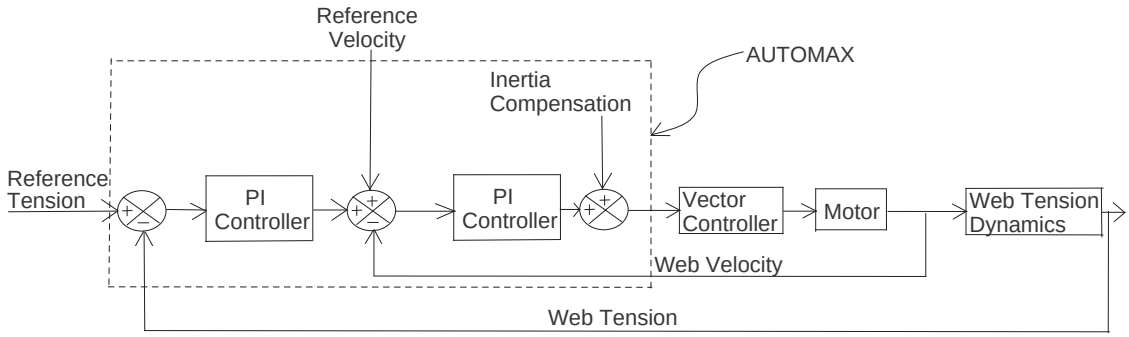


Figure 4.1: Present control block diagram.

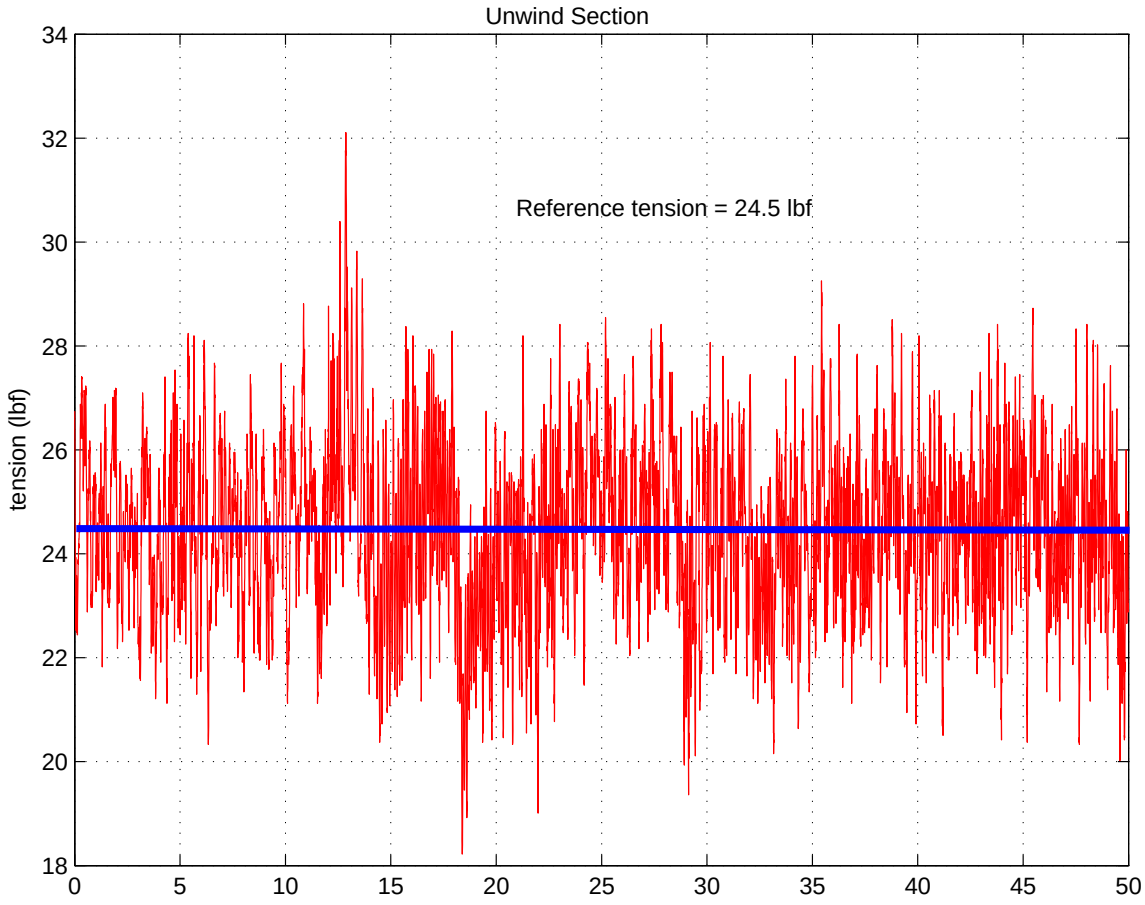


Figure 4.2: Tension response using industrial controller.

In the currently used industrial control strategy, the reference velocities, v_{ri}, for each driven roller are set equal to the master speed roller, which sets the web transport speed in the process line. Setting the reference velocities of all the driven rolls/rollers to the

same value will cause unacceptable steady state tension response as shown in Fig. 4.2. To explain this, assume that the controller is able to bring tension error to zero, i.e., t_1 = t_{r1}. Consequently, output of the tension PI controller also converges to zero. Thus, there is no correction term added to the reference velocity v_{r0} and velocity reference to inner PI controller is now v_{r0}. Let us assume that the inner velocity controller is also working perfectly, hence, velocity v_0 is also brought to v_{r0}. But note that the second motor, M_1, corresponding to master speed roller, is also controlled to run at the reference speed $v_{r1} = v_{r0}$. This means both motors M_0 and M_1 are forced to run at the same speed when tension error, $T_1 = t_1 - t_{r1}$, is zero. So there is no further strain in the web. But unwind roll is continuously passing web material into the zone between M_0 and M_1 (see Fig. 3.4). The released material may not have tension, t_0, same as t_{r1}. This will cause a change in the web tension, t_1. Thus, wound-in web material from unwind roll, when released, continuously disturbs the tension in the unwind section. Consequently, in response to error $T_1 = t_1 - t_{r1}$, tension controller will give the necessary speed correction signal. Velocity controller will track this new velocity which is equal to the correction signal plus v_{r0}. This new velocity will again bring the tension error back to zero and again the newly released material will cause a tension disturbance. This sequence is repeated continuously. Ultimately, there will be a continuous oscillations in the tension response. Because of interconnections between tension zones, this disturbance is propagated forward to all subsequent sections causing oscillatory tension response in each section.

Note that the above discussion given using an intuitive physical explanation can also be confirmed from the tension dynamic equation:

$$L_1\dot{t}_1 = EA(v_1 - v_0) + t_0v_0 - t_1v_1 \tag{4.3}$$

When both tension and velocity are in steady state, $v_0 = v_1 = v_{r0}$ and $t_1 = t_{r1}$, then $L_1\dot{t_1} = -t_{r1}v_{r0} \neq 0$ which means tension, t_1, is not held constant at t_{r1}.

Using the control structure shown in Fig. 4.1 and setting the same reference velocity

for all sections, simulations[1] are conducted. The web dynamic model derived in Chapter 3 is used for simulations. Simulated tension response is shown in Fig. 4.3, which shows oscillations in steady state tension response. A number of experiments were conducted

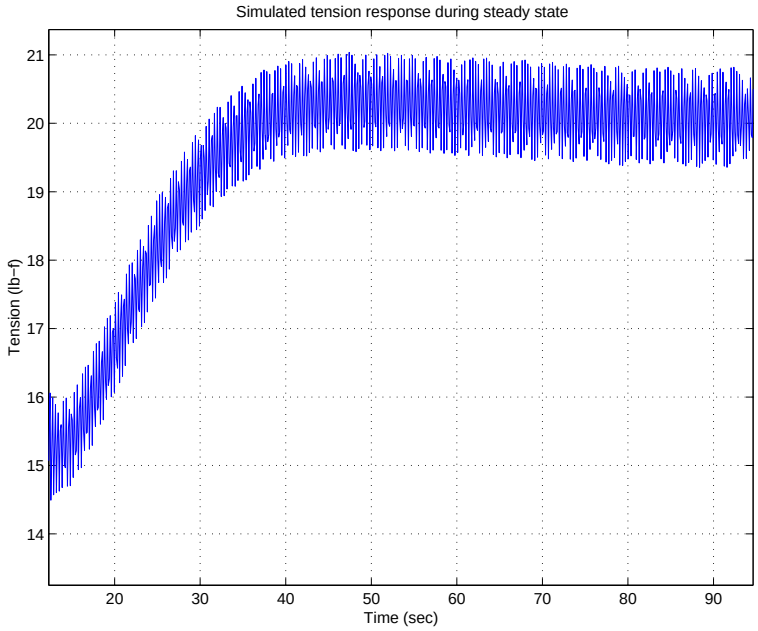


Figure 4.3: Simulated tension (steady state) with present structure and PI controllers.

at different speeds with the industrial PI controller. Corresponding results are shown in the subsequent section and compared with the experimental results obtained with proposed decentralized controllers.

4.2 Decentralized Nonadaptive State Feedback Controller

This section explains the design of the decentralized controller proposed in Chapter 2 for large-scale systems involving linear interconnections. In Chapter 2, the class of large-scale systems is considered whose each subsystem can be written as

$$S_i : \qquad \dot{x}_i(t) = A_i x_i(t) + b_i u_i(t) + \sum_{j=0, j\neq i}^{N} A_{ij} x_j(t) \tag{4.4}$$

[1]Simulink block diagram for running the industrial PI controller is documented in the appendix

Note from equations (3.34) and (3.43) that the dynamic models of the process section and master speed section are arranged in the desired form as given by equation (4.4). But unwind and rewind dynamic models (see equations (3.25),(3.52)) are not exactly in the desired form. Hence, choose the decentralized control input for each motor as follows:

$$U_0 = -f_0(V_0) + K_0^T x_0 \tag{4.5}$$

$$U_1 = -K_1^T x_1 \tag{4.6}$$

$$U_2 = -K_2^T x_2 \tag{4.7}$$

$$U_3 = -f_3(V_3) - K_3^T x_3 \tag{4.8}$$

where $K_i,\ \ i = 0,1,2,3$ are feedback gain vectors. The dynamics of each subsystem under these decentralized control inputs can be simplified to

$$\dot{x}_i = \bar{A}_i x_i + \sum_{j=0, j\neq i}^{N} A_{ij} x_j \tag{4.9}$$

where $\bar{A}_i := A_i - B_i K_i^T$. The error convergence rate can be adjusted with a suitable choice of ε_i in the design, which results in a new condition on $\bar{A}_i$ as stated next.

The equilibrium, $x_i = 0$, of the dynamics given by (4.9) is globally exponentially stable, if the feedback gains K_i are chosen such that

$$\min_{\omega \in R} \sigma_{min}(\bar{A}_i - j\omega I) > \sqrt{N(\xi_i^2 + \varepsilon_i)} > 0 \tag{4.10}$$

where

$$\xi_i^2 = \sum_{j=0, j\neq i}^{N} \eta_{ji}^2,\ \ \eta_{ij} = \sigma_{max}(A_{ij}).$$

Thus, the implementation strategy for the proposed decentralized controller can be summarized as shown in Fig. 4.4.

The control design involves the process in which K_i needs to be chosen iteratively so that resulting $\bar{A}_i$ satisfies the sufficient conditions given by (4.10). The selection of gains K_i can be done using the pole placement technique or the Linear Quadratic Regulator (LQR) algorithm.

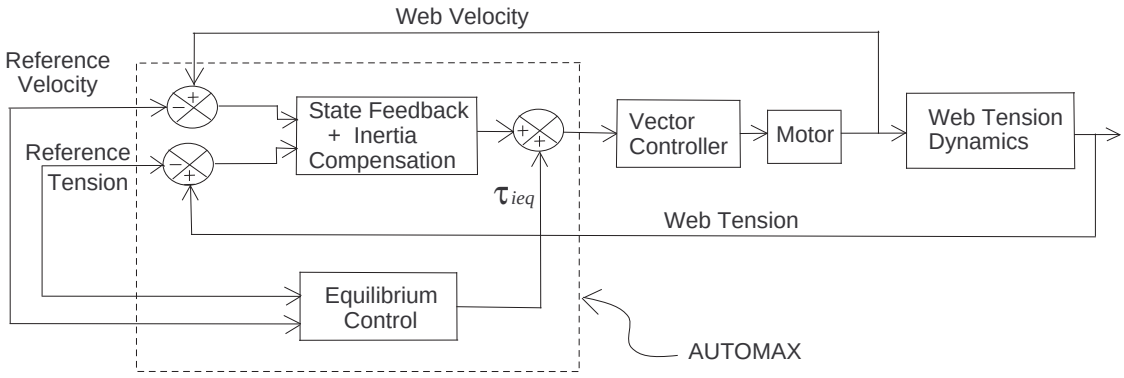


Figure 4.4: Decentralized control strategy with proposed controller.

Note that the stability analysis in chapter 2 is done with reference to the time invariant matrices A_i and b_i. But web velocity dynamics for unwind and rewind section involve time varying parameters such as inertia and radius of the roll. Even then proposed state feedback controller can be used for unwind and rewind sections with careful design and modification. One design approach is to find constant gains K_i using pole placement or LQR algorithm and, then, compute $\bar{A_i} = A_i - B_i K_i^T$. With this $\bar{A_i}$, check whether the conditions given by (4.10) are satisfied or not for all the possible values of the radius of the roll, from full (empty) roll to empty (full) roll in case of the unwind (rewind) roll. This approach may result in high gain values and, hence, may lead to a conservative design. A better approach is to obtain K_i explicitly in terms of the radius of the roll. This design approach is explained next in two systematic steps.

Step 1: Instead of selecting gains K_i and checking condition for corresponding $\bar{A_i}$, directly choose matrix $\bar{A_i}$ for the i-th subsystem such that

1. it is hurwitz,

2. it satisfies the condition given by equation (4.10).

Because (A_i, B_i) pair is controllable, it is possible to adjust eigenvalues of $\bar{A_i}$ arbitrarily. Note that just placing the eigenvalues is not sufficient to satisfy the condition in (4.10), but

the elements chosen within $\bar{A}_i$ need to be manipulated to achieve that condition. Not all the elements of $\bar{A}_i$ can be manipulated because of the fact that $\bar{A}_i$ must satisfy $\bar{A}_i = A_i - B_i K_i^T$. Thus, the input matrix B_i decides which elements can be adjusted freely. Except for master speed dynamics, all other subsystems have $B_i = [0, \ n_i R_i / J_i]^T$. Because of zero entry in the first row of B_i, one cannot adjust elements in the first row of A_i. Hence, $\bar{A}_i$ must be chosen to have first row same as A_i and elements in the second row of A_i can be manipulated arbitrarily. In conclusion, second row of $\bar{A}_i$ needs to be chosen such that $\bar{A}_i$ satisfies both the conditions in Step 1. For the unwind section, $\bar{A}_0$ is chosen as

$$\bar{A}_0 = \begin{bmatrix} -v_{r0}/L_1 & (AE - t_0)/L_1 \\ C_{01} & -C_{02} \end{bmatrix} \tag{4.11}$$

For the master speed section, $\bar{A}_1$ is chosen as

$$\bar{A}_1 = -C_{12} \tag{4.12}$$

In case of process and rewind sections, $\bar{A}_i$ is chosen as

$$\bar{A}_i = \begin{bmatrix} -v_{ri}/L_i & (AE - t_{ri})/L_i \\ -C_{i1} & -C_{i2} \end{bmatrix} \tag{4.13}$$

where C_{i1} and C_{i2} are positive constants, which are chosen such that the conditions in Step 1 are satisfied. The MATLAB program to calculate $\min_{\omega \in R} \sigma_{min}(\bar{A}_i - j\omega I)$ is provided in the appendix. For each subsystem, the convergence rate is chosen as $\varepsilon_i = 10$.

Remark 4.2.1 *Note that $\bar{A}_i$ is not time-varying because it is not a function of time-varying parameters R_i and J_i. Hence, the choice of $\bar{A}_i$ is fixed and need not be changed with the change in the radius and inertia of the unwind or rewind rolls.*

But, $\bar{A}_i$ involves reference values v_{ri} and t_{ri}. Hence, in the design, choice of $\bar{A}_i$ is made such that for all $v_{ri} \in [100, 2000] \ ft/min$ and $t_{ri} \in [3, 30] \ lbf$ both the conditions are satisfied. Since the quantity AE is much larger than t_{ri} for most web handling applications, the sufficient condition as a function of v_{ri} is of value. Master speed reference v_{r1}

is changed online and accordingly reference speed v_{ri} for other rollers is calculated, hence sufficient conditions need to be checked for various v_{r1}. Figure 4.5 gives the quantities $\alpha_i = \sqrt{N(\xi_i^2 + \varepsilon_i)}$ and $\beta_i = \min_{\omega \in R} \sigma_{min}(\bar{A}_i - j\omega I)$ as a function of reference web transport speed v_{r1} with a reference tension $t_{ri} = 14.35\ lbf$ and $\varepsilon_i = 10$ for all i. The following C_{i1} and C_{i2} values are used to obtain these plots: $C_{01} = 120$, $C_{02} = 200$, $C_{12} = 4000$, $C_{21} = 1500$, $C_{22} = 400$, $C_{31} = 15$, $C_{32} = 15$.

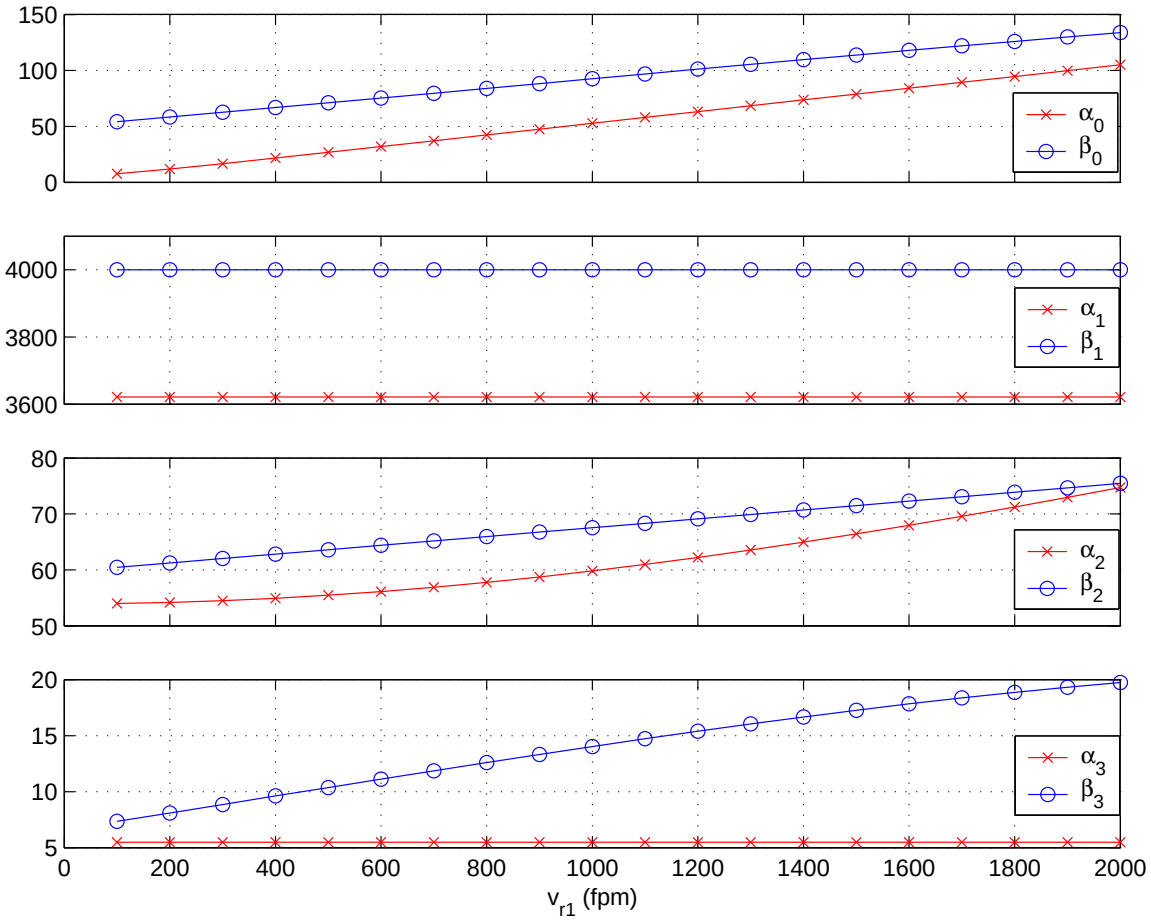


Figure 4.5: Sufficient condition check for different reference velocities ($\alpha_i = \sqrt{N(\xi_i^2 + \varepsilon_i)}$ and $\beta_i = \min_{\omega \in R} \sigma_{min}(\bar{A}_i - j\omega I)$.

Step 2: Once the design of closed loop system matrix $\bar{A}_i$ is done, controller gains can be computed for all sections, except master speed section, using the following equation:

$$K_i^T = \left(\frac{J_i}{n_i R_i}\right)\left[\left(\frac{R_i^2}{J_i} - C_{i1}\right),\ \left(-\frac{b_{fi}}{J_i} + C_{i2}\right)\right] \tag{4.14}$$

For the master speed roller, gain can be computed as

$$K_1 = (\frac{J_1}{n_1 R_1})(-\frac{b_{f1}}{J_1} + C_{12}) \tag{4.15}$$

The above expressions for the gains are obtained using the relation $A_i - B_i K_i^T = \bar{A_i}$ where A_i and B_i are obtained in Chapter 3 and $\bar{A_i}$ is known from Step 1.

4.3 Decentralized Adaptive Controller

In the development of nonadaptive state feedback controller, it was inherently assumed that the coefficient of friction is constant and exactly known. But depending upon running conditions, level of bearing lubrication, and operating motor speed, coefficient of friction, b_{fi} may change. In such a case calculated gains may not result in good performance. This naturally motivates investigation and implementation of a suitable adaptive control algorithm. In Chapter 2 decentralized MRAC controller is proposed, which requires overall large-scale system to be controllable. Forming overall system matrix A and input matrix B from linearized web handling dynamics, it is observed that the large-scale web handling system is controllable. MRAC design is for web handling application is explained next, with systematic design steps.

Step 1: Select matrix A_{mi} for each subsystem such that

1. it ensures the existence of some k_i^T such that the relation $A_i - A_{mi} = B_i k_i^T$ is satisfied.

2. it is hurwitz, and

3. it ensures the existence of positive definite solution to the ARE (2.42).

The above conditions are similar to those addressed in Step 1 of the non-adaptive state feedback controller design. Hence, the design of A_{mi} is similar to the $\bar{A_i}$ as explained in previous section. In fact, the matrix A_{mi} is chosen to be equal to $\bar{A_i}$ as given by equations (4.11) to (4.13).

Step 2: The next step is to stabilize the large-scale reference model to get desired reference state trajectories. For this purpose, a suitable feedback gain k_m is designed to achieve the stability of the overall reference model system matrix A_m. The pair $(A,\ B)$ is controllable, which implies that $(A_m,\ B)$ is controllable because $A_{mi} = A_i - B_i K_i^T$. This ensures existence of a stabilizing gain k_m.

The LQR algorithm is used to obtain the feedback gain k_m, which ensures that the reference states go to their desired values in an optimal sense. The optimal feedback gain k_m is obtained as

$$k_m = \begin{bmatrix} k_{m0} \\ k_{m1} \\ k_{m2} \\ k_{m3} \end{bmatrix} = \begin{bmatrix} -433.2 & 792.2 & -232.5 & -19.8 & -0.3 & -36.8 & -3.1 \\ 70.0 & -63.0 & 37.3 & -4.3 & 0.1 & -3.2 & -0.3 \\ 3.6 & -1.8 & 1.2 & -20.4 & 187.9 & -100.8 & -1.6 \\ 12.3 & -12.5 & -3.7 & 630.5 & -1.2 & 1236.9 & 2984.9 \end{bmatrix} \tag{4.16}$$

Step 3: Now, choose the decentralized control input for each motor as follows:

$$U_0 = -f_0(V_0) + \hat{K}_0^T x_0 \tag{4.17}$$

$$U_1 = -\hat{K}_1^T x_1 \tag{4.18}$$

$$U_2 = -\hat{K}_2^T x_2 \tag{4.19}$$

$$U_3 = -f_3(V_3) - \hat{K}_3^T x_3 \tag{4.20}$$

Solve the ARE (2.42) to get positive definite matrices P_i. The gain matrix at initial time, $\hat{K}_i(0)$ can be computed using equation similar to (4.14) and (4.15), and these gains can be adapted using adaptation law $\dot{\hat{K}}_i = (e_i^T P_i B_i) x_0$, where $e_i = [T_i - T_{mi},\ \ V_i - V_{mi}]^T$. Actual numerical values of P_i, $\hat{K}_i(0)$ and $\dot{\hat{K}}_i$ for each section are given below. The following C_{i1} and C_{i2} values are used: $C_{01} = 120$, $C_{02} = 200$, $C_{12} = 4000$, $C_{21} = 1500$, $C_{22} = 400$, $C_{31} = 15$, $C_{32} = 15$.

Unwind section:

$$P_0 = \begin{bmatrix} 0.5582 & -0.5549 \\ -0.5549 & 5.8131 \end{bmatrix}$$

$$\hat{K}_0^T(0) = (\frac{J_0}{n_0 R_0}) \left[(\frac{R_0^2}{J_0} - C_{01}),\ \ (-\frac{b_{f0}}{J_0} + C_{02}) \right]$$

$$\dot{\hat{K}}_0(t) = \frac{n_0 R_0}{J_0} e_0^T [-0.5549,\ \ 5.8131]^T x_0 \tag{4.21}$$

Master speed section:

$$P_1 = 833.4940$$

$$\hat{K}_1^T(0) = (\frac{J_1}{n_1 R_1}) \left(-\frac{b_{f1}}{J_1} + C_{12} \right) = 23596$$

$$\dot{\hat{K}}_1(t) = \frac{n_1 R_1}{J_1} e_1 (833.4940) x_1 = e_1 (141.2772) x_1$$

Process section:

$$P_2 = \begin{bmatrix} 55.8264 & 3.4204 \\ 3.4204 & 1.8708 \end{bmatrix}$$

$$\hat{K}_2^T(0) = (\frac{J_2}{n_2 R_2}) [(\frac{R_2^2}{J_2} + C_{21}),\ (-\frac{b_{f0}}{J_0} + C_{22})] = [8849.2,\ \ 2356.9]$$

$$\dot{\hat{K}}_2(t) = \frac{n_2 R_2}{J_2} e_2^T [3.4204,\ \ 1.8708]^T x_2 = e_2^T [0.5798,\ \ 0.3171]^T x_2$$

Rewind section:

$$P_3 = \begin{bmatrix} 0.7274 & 0.3712 \\ 0.3712 & 1.2989 \end{bmatrix}$$

$$\hat{K}_3^T(0) = (\frac{J_3}{n_3 R_3}) [(\frac{R_3^2}{J_3} + 1C_{31}),\ \ (-\frac{b_{f3}}{J_3} + C_{32})]$$

$$\dot{\hat{K}}_3(t) = \frac{n_3 R_3}{J_3} e_3^T [0.3712,\ \ 1.2989]^T x_3$$

The proposed adaptation method uses a gradient algorithm for which the estimated gains may increase and saturate the control signal. To avoid this, the gradient projection

algorithm is used, which maintains estimates within prescribed bounds. The minimum and maximum limits on $\hat{K}_i^T$ are obtained using nominal values of all the parameters and their maximum possible deviations.

4.4 Experimental Results

Extensive experiments at different web process line velocities were conducted with the currently used industrial decentralized PI controller and the proposed adaptive and non-adaptive decentralized controllers. Controllers were implemented using AUTOMAX distributed control system, which uses "Basic programming language" to implement discrete algorithms. Programs that implement the control algorithms were scanned by dedicated microprocessors with a sampling rate of 5 ms. Real-time tension and velocity signals were collected through the data acquisition system also at a sampling rate of 5 ms. Appendix B gives the implementation procedure in AUTOMAX. A complete step-by-step algorithm for implementation of the proposed decentralized controllers is given in Appendix C.

Experimental results for three cases and three controllers are shown. In each case, variation of the web line speed at the master speed roller and tension variations in each tension zone are shown. Control input signals, τ_i, for all the four sections are also presented.
Case 1) Reference velocity $v_{r1} = 1000\ ft/min$; $t_{r1} = 24.6\ lbf$, $t_{r2} = 20.5\ lbf$, and $t_{r3} = 16.4\ lbf$; the roll diameter varies from 18 to 13 *inches*. See Figures 4.6 through 4.8
Case 2) Reference velocity $v_{r1} = 1500\ ft/min$; the reference tension is the same in all the three zones and is chosen as $t_{ri} = 14.35\ lbf$; the roll diameter varies from 14 to 5 *inches*. See Figures 4.12 through 4.14.
Case 3) Reference velocity $v_{r1} = 750\ ft/min$; the reference tension is the same in all the three zones and is chosen as $t_{ri} = 14.35\ lbf$. See Figures 4.15 through 4.17.

As compared to the existing decentralized PI controller, results using the proposed decentralized controllers show much improved web tension regulation in each of the zones. For Case 1 control inputs are also presented in Figures 4.9 through 4.11, which show that

-	PI	Non-adaptive	Adaptive
$\|\|V_1\|\|$	179.72	168.51	160.36
$\|\|T_1\|\|$	27.39	11.19	19.20
$\|\|T_2\|\|$	26.01	12.33	18.47
$\|\|T_3\|\|$	47.79	21.11	21.11

Table 4.2: Comparison of controllers: Velocity reference 1000 ft/min

-	PI	Non-adaptive	Adaptive
$\|\|V_1\|\|$	217.41	156.66	184.75
$\|\|T_1\|\|$	74.76	18.3889	19.17
$\|\|T_2\|\|$	49.16	16.3152	20.30
$\|\|T_3\|\|$	97.95	27.29	20.77

Table 4.3: Comparison of controllers: Velocity reference 1500 ft/min

the adaptive and nonadaptive control inputs show very small oscillations compared to PI control inputs. It means control energy injected into the system is much less for new proposed controllers.

Tables 4.2 through 4.4 show the two-norm of the tension and velocity signals for three controllers for three cases. One can observe that the proposed decentralized controllers, both non-adaptive and adaptive, outperform the decentralized PI controller.

-	PI	Non-adaptive	Adaptive
$\|\|V_1\|\|$	552.25	132.48	155.53
$\|\|T_1\|\|$	33.23	11.27	14.41
$\|\|T_2\|\|$	27.65	11.55	14.16
$\|\|T_3\|\|$	43.99	15.35	19.66

Table 4.4: Comparison of controllers: Velocity reference 750 ft/min

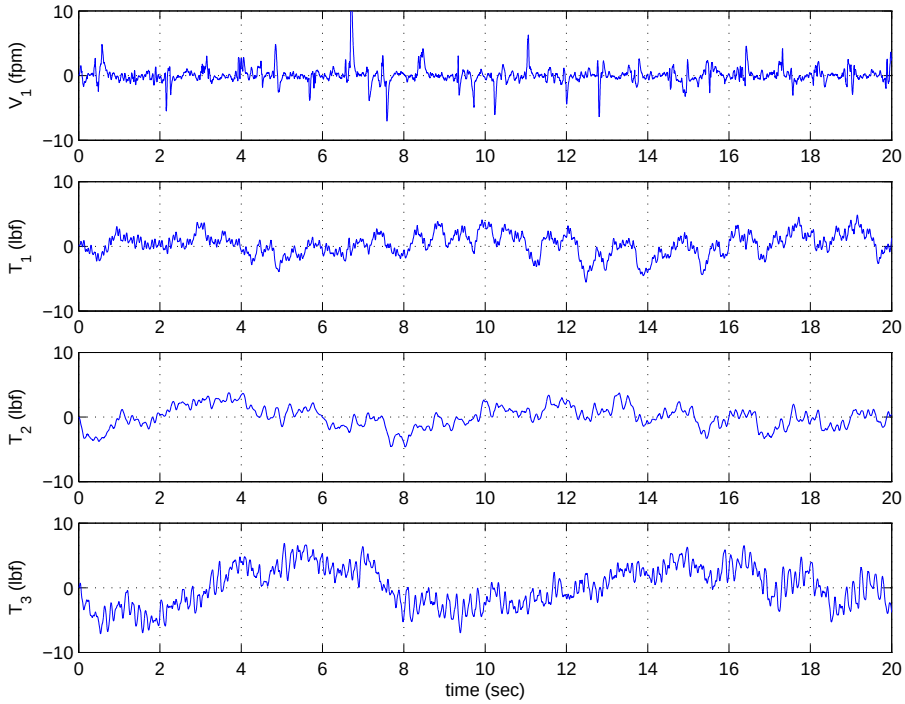


Figure 4.6: Decentralized PI controller: Reference velocity 1000 ft/min

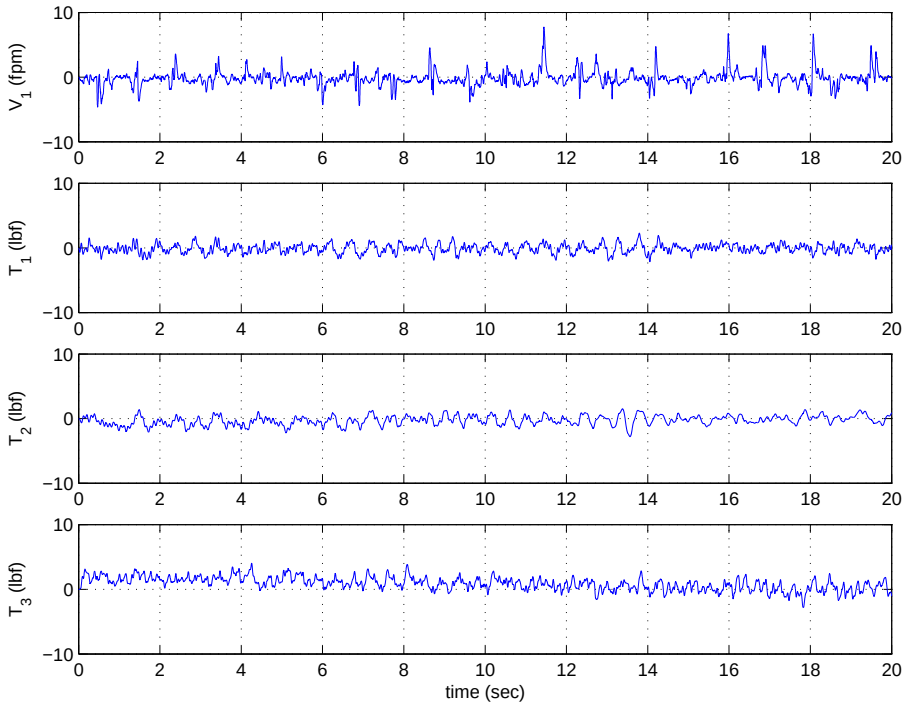


Figure 4.7: Decentralized nonadaptive controller: Reference velocity 1000 ft/min

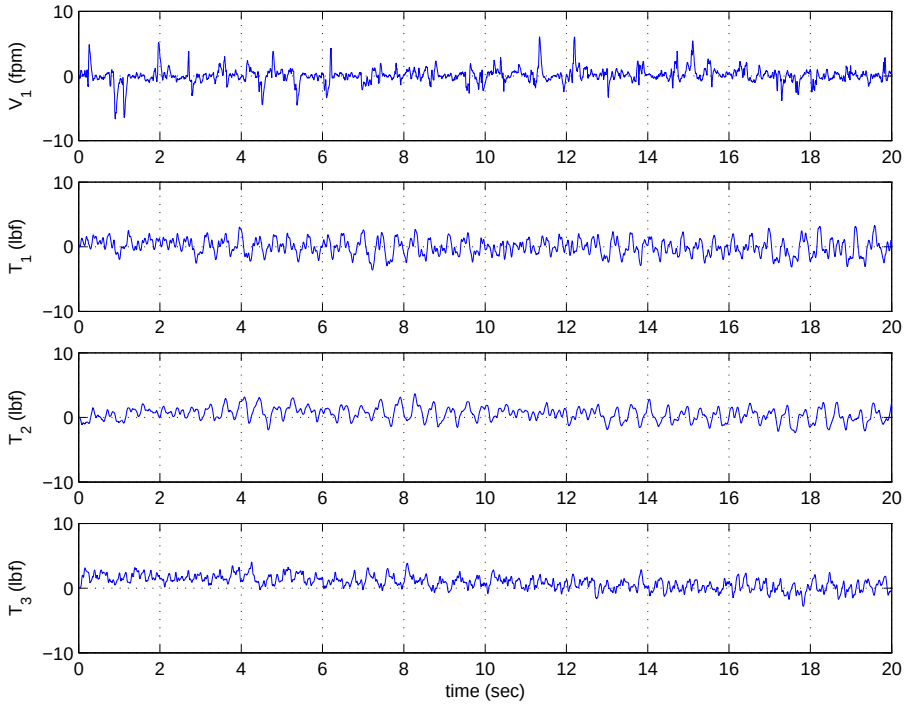


Figure 4.8: Decentralized adaptive controller: Reference velocity 1000 ft/min

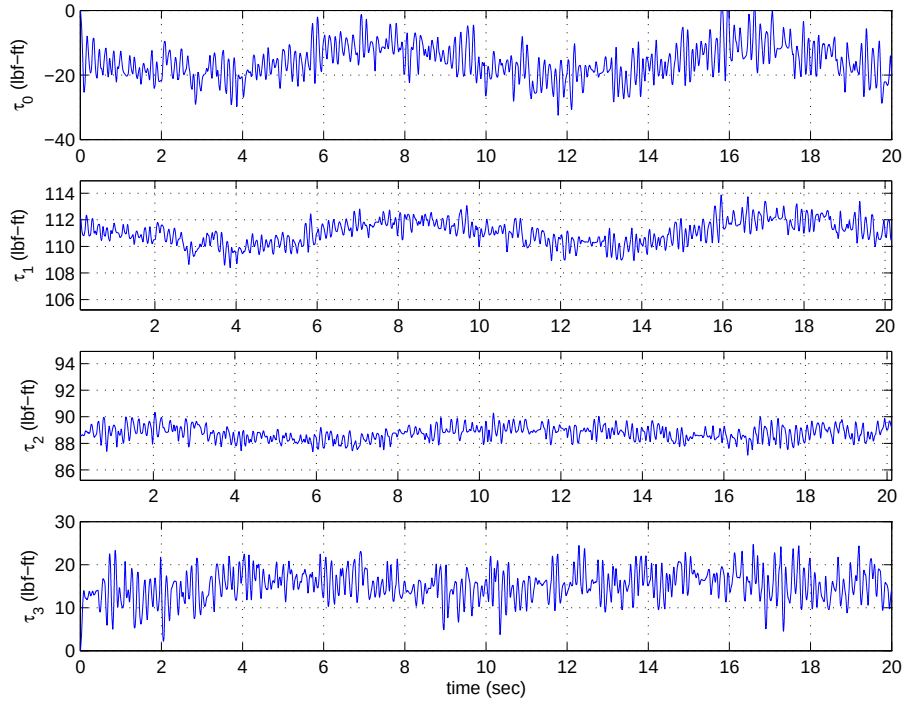


Figure 4.9: Control inputs (PI): Reference velocity 1000 ft/min

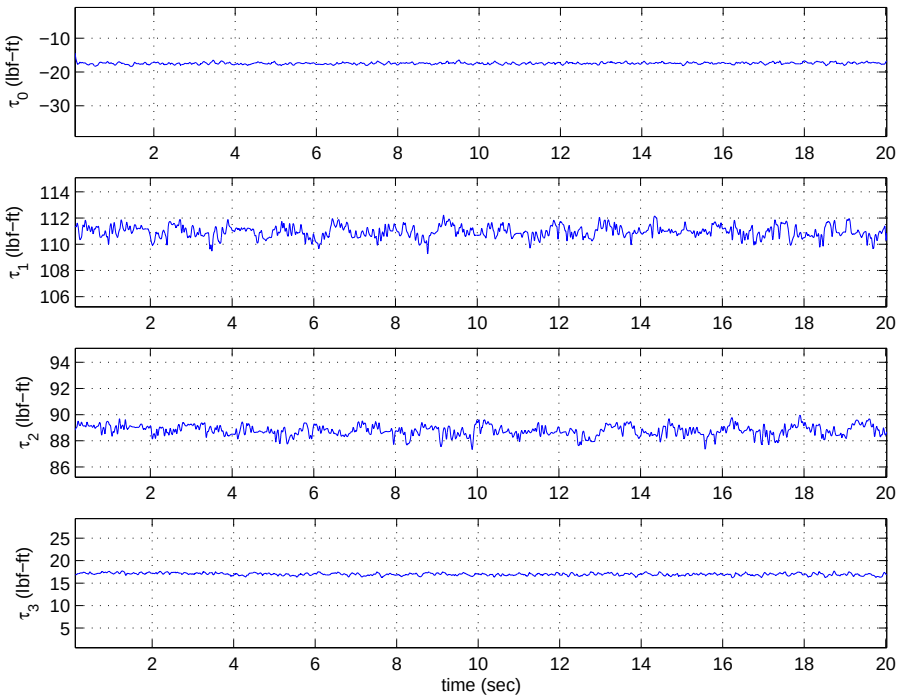


Figure 4.10: Control inputs (non-adaptive): Reference velocity 1000 ft/min

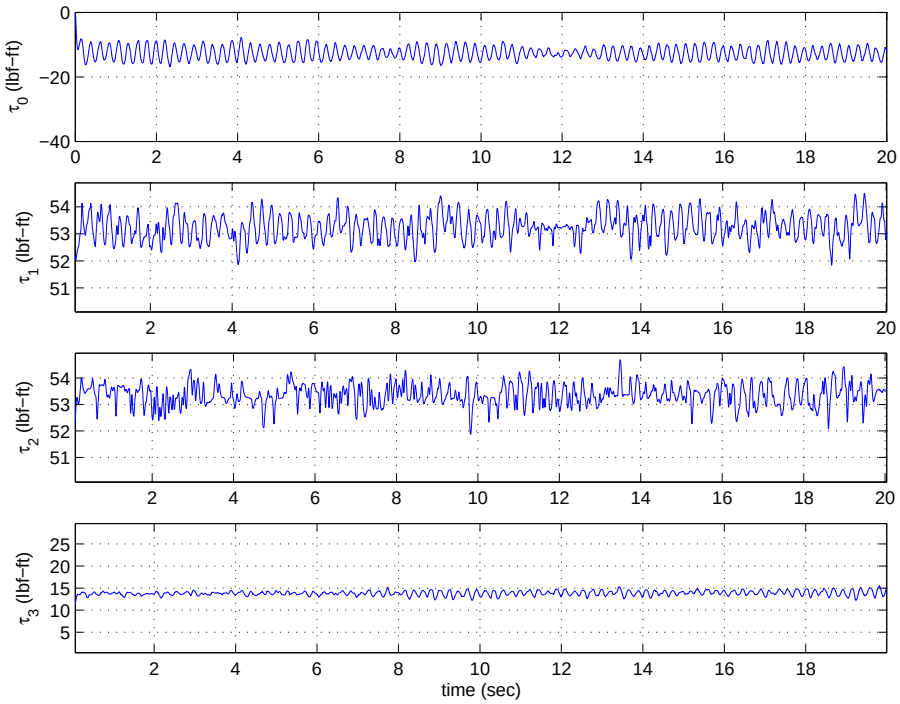


Figure 4.11: Control inputs (adaptive): Reference velocity 1000 ft/min

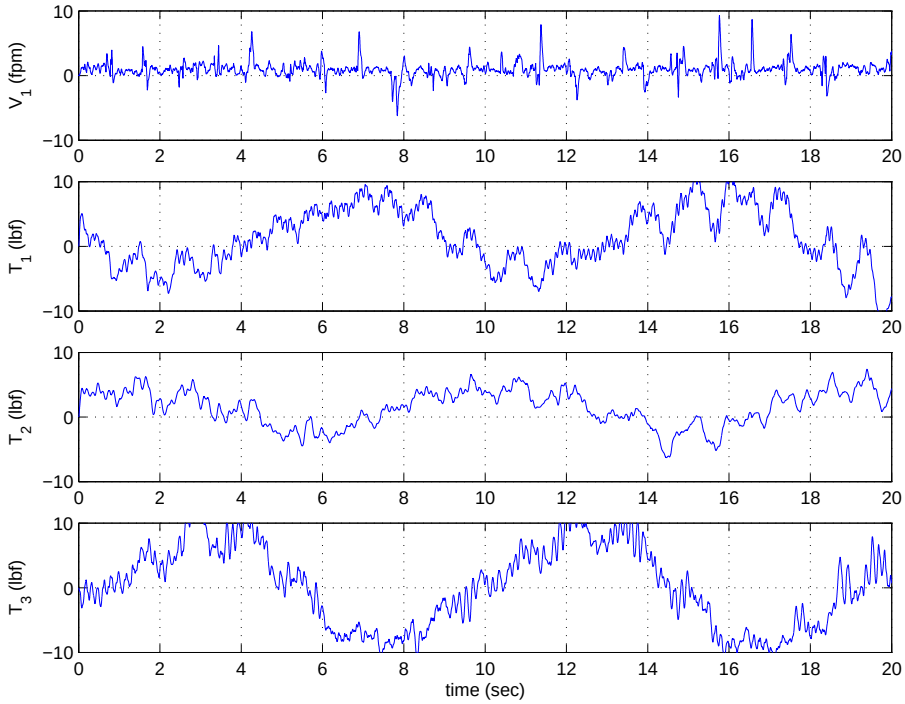


Figure 4.12: Decentralized PI controller: Reference velocity 1500 ft/min

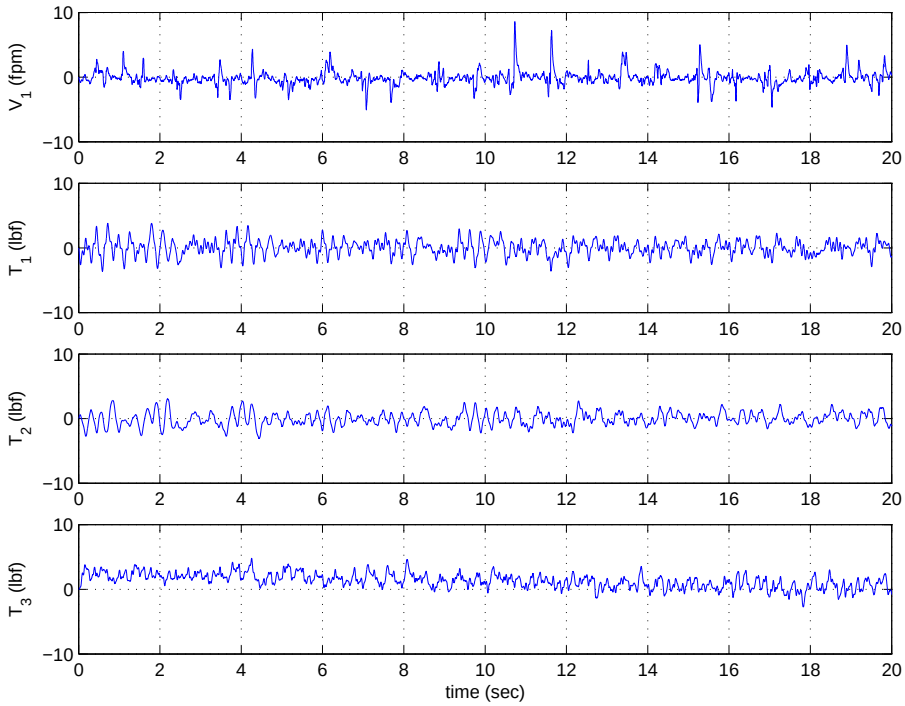


Figure 4.13: Decentralized nonadaptive controller: Reference velocity 1500 ft/min

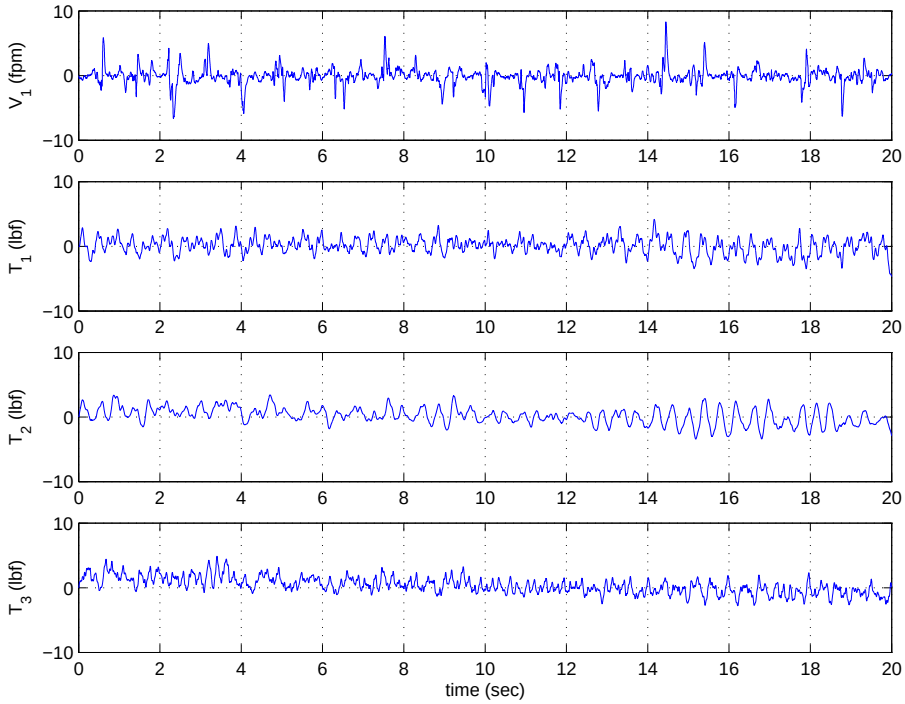


Figure 4.14: Decentralized adaptive controller: Reference velocity 1500 ft/min

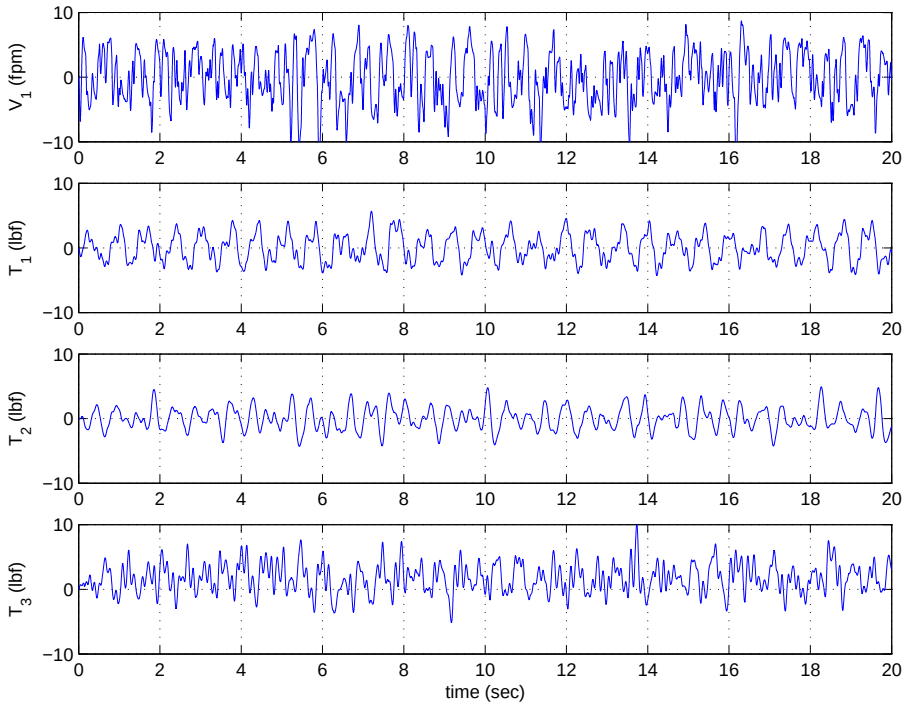


Figure 4.15: Decentralized PI controller: Reference velocity 750 ft/min

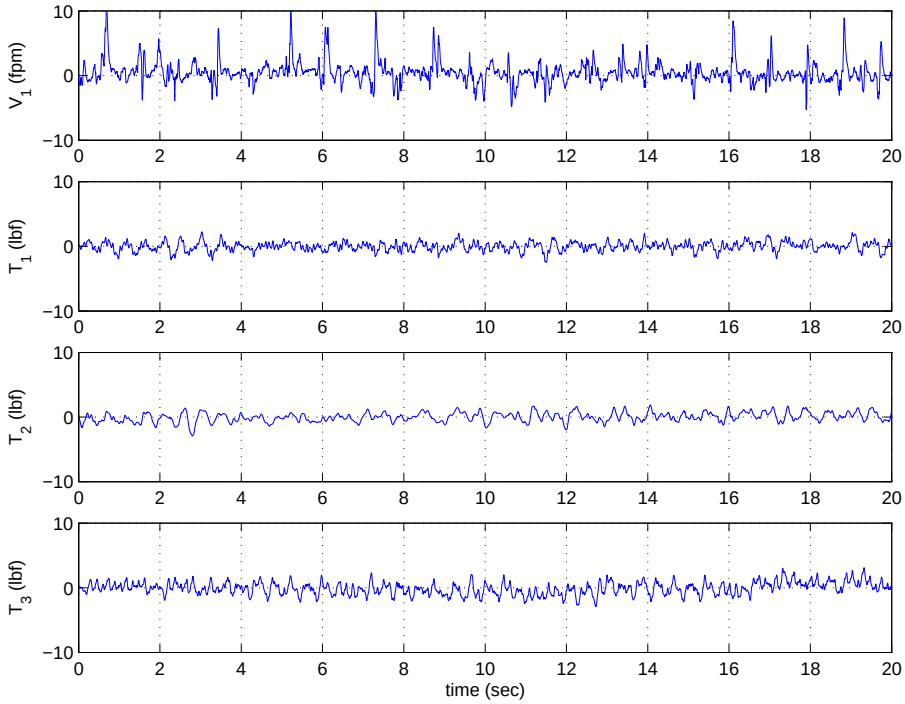


Figure 4.16: Decentralized nonadaptive controller: Reference velocity 750 ft/min

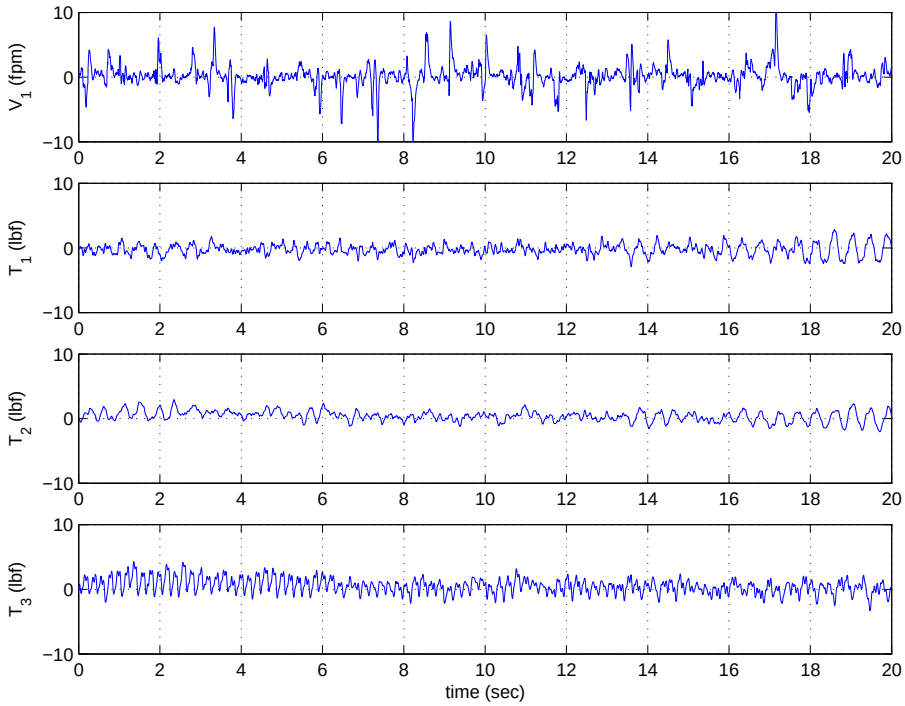


Figure 4.17: Decentralized adaptive controller: Reference velocity 750 ft/min

Note that the performance of the nonadaptive decentralized controller is better than the adaptive decentralized controller. This is because of the use of projection gradient algorithm in the implementation of the adaptive controller, which keeps the gain estimates within prescribed bounds. This causes oscillations in the controller gains thus adding more oscillations in the tension and velocity response as compared to the response due to a decentralized controller with nonadaptive gains. Figure 4.18 shows the estimated gains for Case 2.

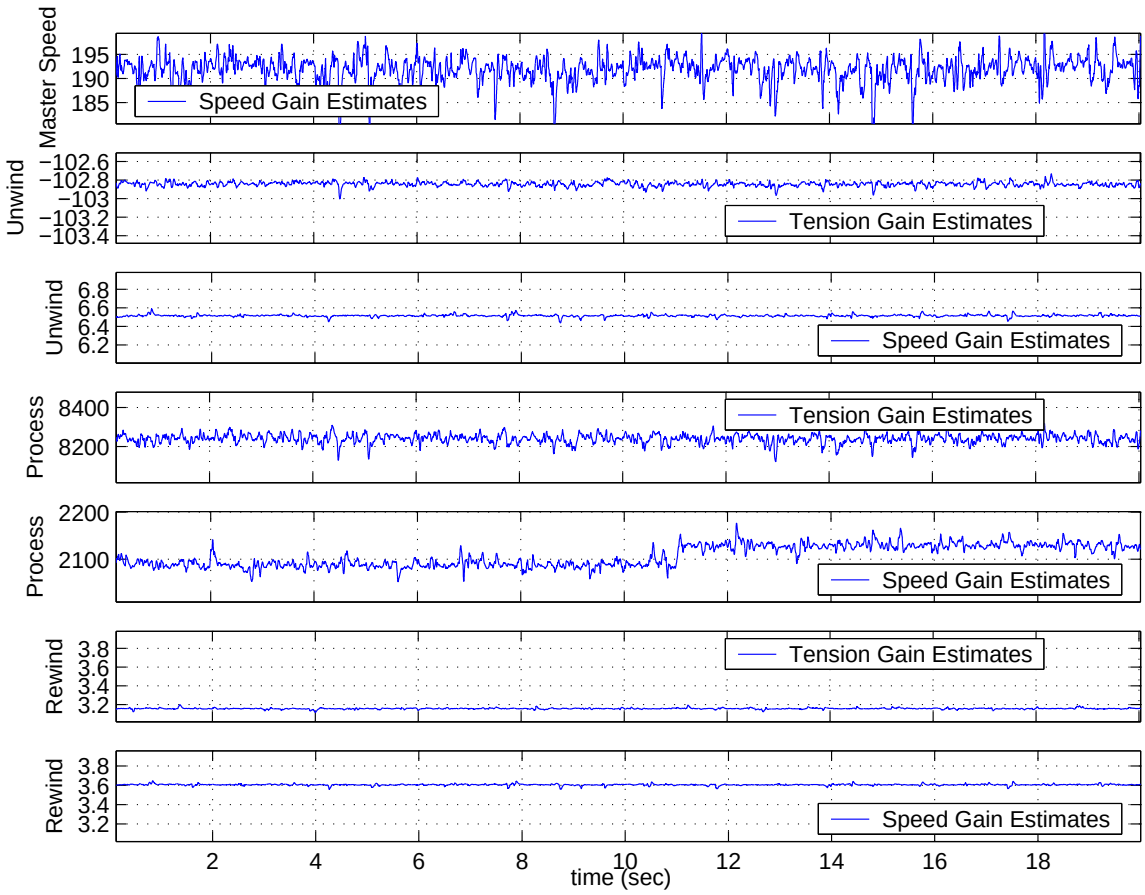


Figure 4.18: Estimated gains: Reference velocity 1500 ft/min

4.4.1 Robustness of the decentralized controller: An experimental evaluation

The nonadaptive decentralized controller is checked for robustness against online change in the reference speed v_{r1} in real-time. Figures 4.19 and 4.20 show the tension response t_1 in unwind section when master speed reference is changed from 500 to 800 fpm and 1000 to 1500 fpm, respectively. Compared to the decentralized PI controller, the nonadaptive decentralized controller does not show much change in the tension when the speed reference is changed.

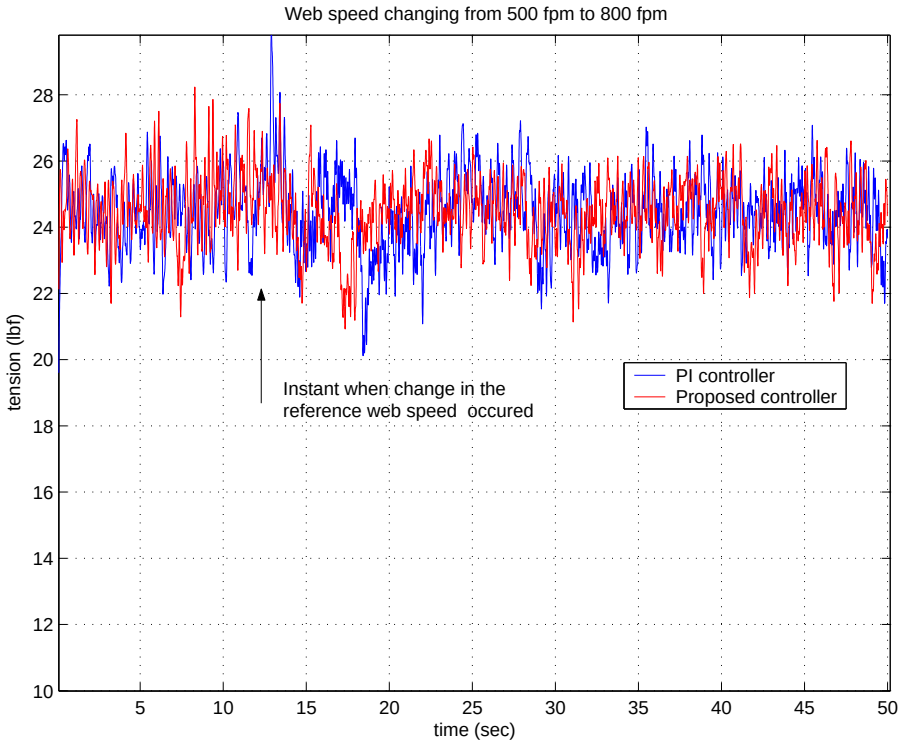


Figure 4.19: Comparison of tension signals in response to change in the reference web speed from 500 to 800 fpm.

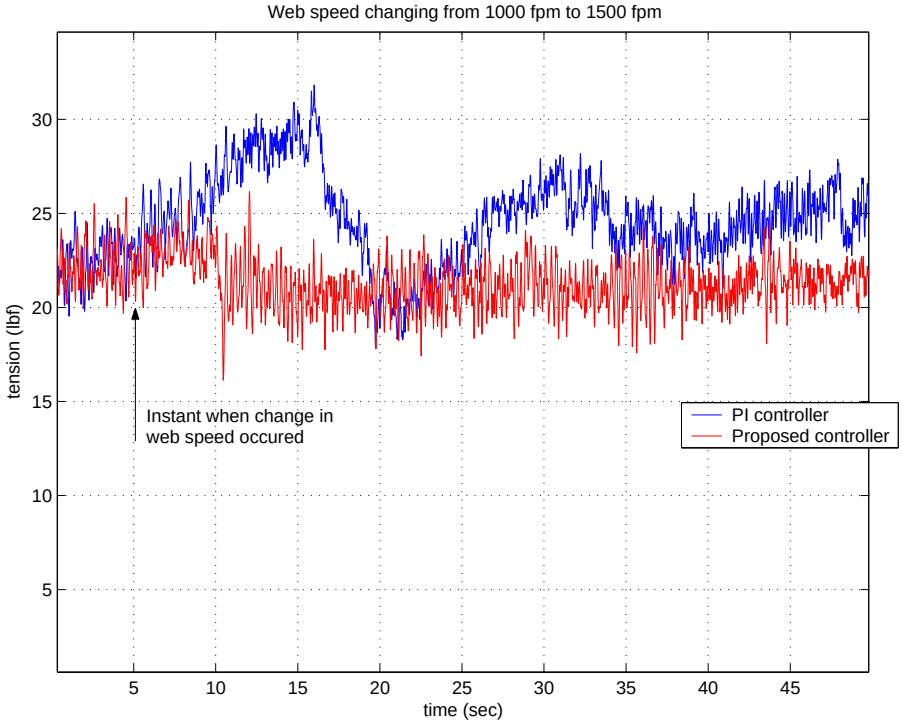


Figure 4.20: Comparison of tension signals in response to change in the reference web speed from 1000 to 1500 fpm.

Industrial application also demands high level of safety and hence the proposed non-adaptive controller was also checked when the feedback signal from the sensor does not reflect exact physical values. To check such robustness of the controller, velocity feedback from the motor encoders is divided by 2 and used as feedback to compare with reference v_{ri}. With the use of same gains as earlier, nonadaptive controller was not able to maintain steady state values of web velocity in different sections to desired reference values. But velocities were maintained at scaled values of desired v_{ri}. It means that the system stability was maintained even with the faulty feedback of velocity v_i. The corresponding tension signals are shown in the Fig. 4.22. Tension references were kept at $t_{ri} = 20\ lbf$ for all i. Note that with nonadaptive controller unwind tension is almost kept at desired reference value but there is a steady state error in process tension t_2. Note that with the PI decentralized controller (see Fig. 4.21), system not only looses the steady state but tends to go

unstable and after some time web would have broken.

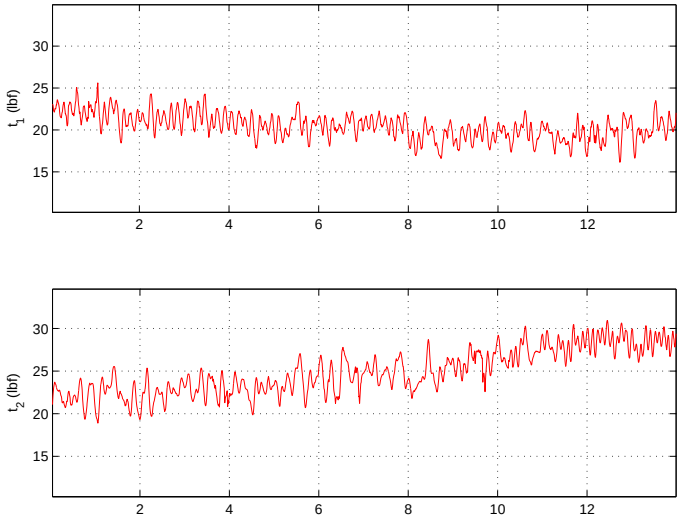


Figure 4.21: Decentralized PI controller with halved velocity feedback.

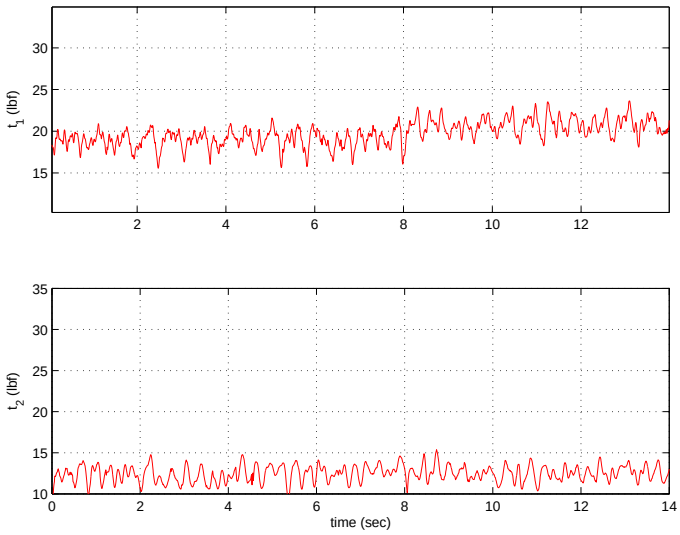


Figure 4.22: Decentralized nonadaptive controller with halved velocity feedback.

During extensive experimentation, resonance was observed at line reference speed of 2000 fpm when the nonadaptive decentralized controller was implemented. The results at resonance are shown in the Fig. 4.23. Figure 4.24 shows errors in the velocity of different driven rollers other than master speed roller. To avoid such resonating conditions, it is necessary to analyze the closed loop system with proposed nonadaptive controller in frequency domain.

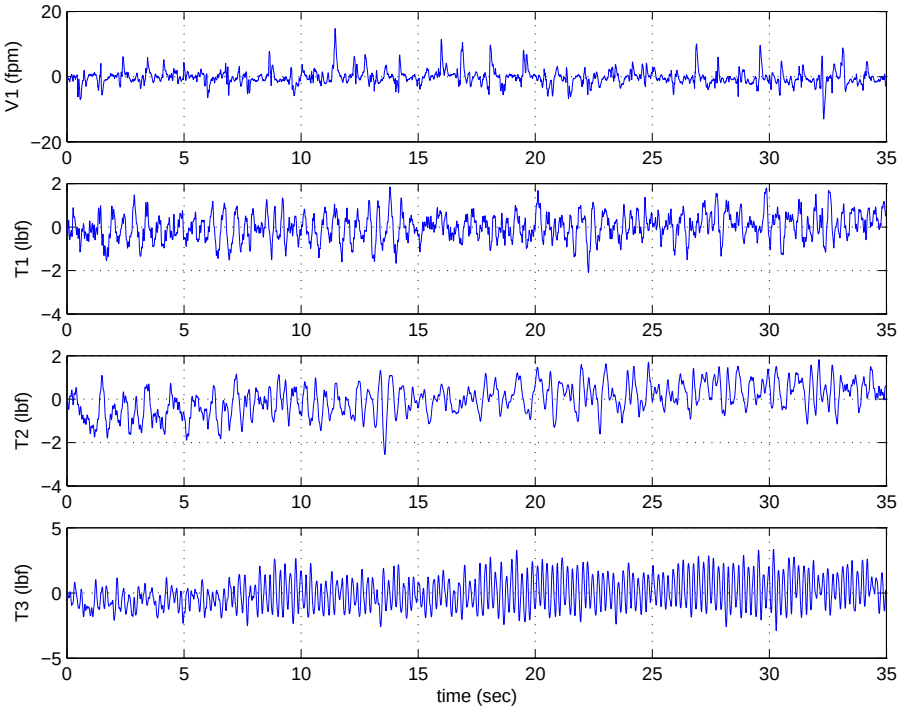


Figure 4.23: With decentralized nonadaptive controller at resonating condition.

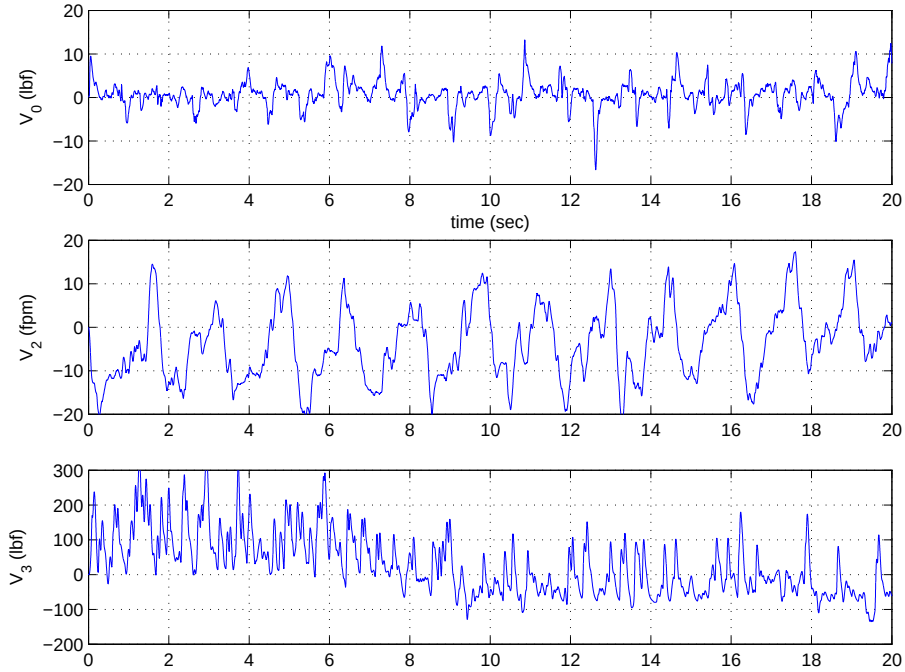


Figure 4.24: Velocity errors for different rollers at resonating condition.

CHAPTER 5

SUMMARY AND FUTURE WORK

5.1 Summary

Large-scale systems are emerging with great importance in many fields. Analysis and control of large-scale systems is a challenging task due to the complex nature of interconnections among constituent subsystems. This thesis involved development and implementation of decentralized control schemes for certain class of large-scale systems. The application of the proposed controllers is shown on control of web processing lines. Following paragraphs give a chapter-by-chapter summary of this report.

In Chapter 2, the class of large-scale systems with unmatched linear interconnections is considered to develop decentralized adaptive controllers. Both, adaptive state regulation and Model Reference Adaptive Controllers (MRAC) are developed. Adaptive state regulation case does not assume exact knowledge of interconnecting parameters. Whereas in the MRAC scheme, a modified reference model, which makes use of known interconnecting parameters, is developed to achieve exact tracking.

The conditions, under which these controllers result in global asymptotic stability, are obtained in terms of solution to the Algebraic Riccati Equation (ARE). Sufficient conditions are given for the existence of positive definite solutions to the ARE.

Although applicable to web handling systems, assumption that interconnections are linear is restrictive . Hence, a decentralized adaptive controller for the class of large-scale systems involving unmatched nonlinear interconnections is obtained. The design of the controller requires knowledge of initial conditions or upper bound on the initial conditions of the states. Four different approaches are considered depending upon the prior knowledge

of interconnecting parameters. Correspondingly four different conditions are obtained to ensure asymptotic stability of the large-scale system.

In Chapter 3, a model for the unwind (rewind) roll is developed by explicitly considering the variation of radius and inertia resulting from release (accumulation) of material to (from) the process. Based on the new model developed, a decentralized control scheme is designed, which involves equilibrium control inputs and feedback control inputs. A strategy for computing the equilibrium control inputs and reference velocities for each driven roll/roller is given. This strategy is based on dividing the web processing line into tension zones. Each tension zone uses reference web tension and the reference velocity of the master speed roller. With the use of equilibrium control inputs and reference velocities, the state dynamic model is transformed into a model in terms of state errors.

In Chapter 4, as an application, a web handling system is considered for implementation. Web handling systems inherently form a class of large-scale systems with unmatched interconnections. Based on the dynamic model in terms of state errors, feedback control laws are obtained using decentralized controllers proposed in Chapter 2. Two types of controllers are designed:

- Decentralized nonadaptive state regulator.
- Decentralized adaptive controller.

A number of experiments were conducted to check the robustness of the proposed controllers. The performance of the proposed controllers is compared with that of an often used industrial PI controllers. Substantial improvement in web tension error regulation is observed with the proposed controllers.

5.2 Future Work

In the case of large-scale systems with linear interconnections global asymptotic stability is achieved. But with nonlinear interconnections semi-globally stable decentralized

controllers are developed, which require knowledge of initial conditions of the states or upper bounds on them. To make the discussion complete, globally stable decentralized controllers, which do not require initial conditions of the states, need to be considered. In the case of linear interconnections, global asymptotic stability is achieved with relative ease because part of the control input energy is appropriately used to overpower maximum interconnection energy to achieve stability. But in the case of nonlinear interconnections a situation may arise in which energy of a control input may become zero but still energy associated with the unknown interconnections does not become zero. This condition particularly arises because of unmatched condition, that is, because the control input does not enter the subsystem at the same point where interconnections enter. Switched/ Hybrid control scheme involving analysis with multiple Lyapunov functions may find application in this problem.

Proposed decentralized scheme relies on an iterative process to arrive at numerical values of controller parameters. For large systems this iterative process may prove to be tedious. Future research should focus on obtaining closed form solutions for the controller gains.

In Chapter 3, development of decentralized controllers for the web handling system assumes that the product of elasticity constant and area of cross-section, EA, and thickness, t, are perfectly known. In some applications these may not be known and hence new adaptive controller is required to adapt for parameters EA and t. Design of controllers also assume full knowledge of state vector $[T_i, \ V_i]^T$. However, tension signal may not be available for some subsystems in web processing lines. For example it may not be possible to place load cell sensor in a hot chamber. In such cases, it is desirable to design decentralized observer based control scheme, which will use velocity signal and estimate tension signal to generate stabilizing control signal.

BIBLIOGRAPHY

[1] D. D. Siljak, *Large-scale dynamic systems*. Elsevier North-Holland, Inc., 1978.

[2] J. N. R. Sandell, P. Varaiya, M. Athans, and M. G. Safonov, "Survey of decentralized control methods for large scale systems," *IEEE Transactions on Automatic Control*, vol. 23, pp. 108–128, April 1978.

[3] C. D. Schaper, T. Kailath, and Y. J. Lee, "Decentralized control of wafer temperature for multizone rapid thermal processing systems," *IEEE Transactions on Semiconductor Manufacturing*, vol. 12, pp. 193–199, May 1999.

[4] D. J. Stilwell and B. E. Bishop, "Platoons of underwater vehicles," *IEEE Control Systems Magazine*, pp. 45–52, December 2000.

[5] J. T. Feddema, C. Lewis, and D. A. Schoenwald, "Decentralized control of cooperative robotic vehicles: theory and application," *IEEE Transactions on Robotics and Automation*, vol. 18, pp. 852–864, October 2002.

[6] S. H. Wang and E. J. Davison, "On the stabilization of decentralized control systems," *IEEE Transactions on Automatic Control*, vol. 18, pp. 473–478, 1973.

[7] B. D. O. Anderson and J. B. Moore, "Time-varying feedback laws for decentralized control," *IEEE Transactions on Automatic Control*, vol. 26, pp. 1133–1138, 1981.

[8] S. H. Wang, "An example in decentralized control systems," *IEEE Transactions on Automatic Control*, vol. 23, pp. 938–938, 1989.

[9] P. P. Khargonekar and A. B. Ozguler, "Decentralized control and periodic feedback," *IEEE Transactions on Automatic Control*, vol. 39, pp. 877–882, April 1994.

[10] Z. Gong and M. Aldeen, “Stabilization of decentralized control systems,” *Journal of Mathematical Systems, Estimation and Control*, vol. 7, no. 1, 1997.

[11] P. A. Ioannou, “Decentralized adaptive control of interconnected systems,” *IEEE Transactions on Automatic Control*, vol. 31, no. 4, pp. 291–298, 1986.

[12] L. Shi and S. K. Singh, “Decentralized adaptive controller design for large-scale systems with higher order interconnections,” *IEEE Transactions on Automatic Control*, vol. 37, no. 8, pp. 1106–1118, 1992.

[13] S. Jain and F. Khorrami, “Decentralized adaptive control of a class of large-scale interconnected nonlinear systems,” *IEEE Transactions on Automatic Control*, vol. 42, no. 2, 1997.

[14] D. T. Gavel and D. D. Siljak, “Decentralized adaptive control: Structural conditions for stability,” *IEEE Transactions on Automatic Control*, vol. 34, pp. 413–426, April 1989.

[15] K. S. Narendra and N. O. Oleng’, “Exact output tracking in decentralized adaptive control systems,” *IEEE Transactions on Automatic Control*, vol. 47, pp. 390–395, February 2002.

[16] B. M. Mirkin and P.-O. Gutman, “Decentralized output-feedback mrac of linear state delay systems,” *IEEE Transactions on Automatic Control*, vol. 48, pp. 1613–1619, September 2003.

[17] P. R. Pagilla and H. Zhong, “Semi-globally stable decentralized control of a class of large-scale interconnected nonlinear systems,” *Proceedings of the Americal Control Conference*, 2003.

[18] P. R. Pagilla, “Robust decentralized control of large-scale interconnected systems: General interconnections,” *Proceedings of the Americal Control conference*, vol. 6, pp. 4527–4531, 1999.

[19] D. P. Campbell, *Dynamic Behavior of the Production Process, Process Dynamics*. John Wiley and Sons Inc., New York, first ed., 1958.

[20] K. P. Grenfell, “Tension control on paper-making and converting machinery,” *IEEE 9-th Annual Conference on Electrical Engineering in the Pulp and Paper Industry*, pp. 20–21, June 1963.

[21] D. King, “The mathematical model of a newspaper press,” *Newspaper Techniques*, pp. 3–7, December 1969.

[22] G. Brandenburg, “New mathematical models for web tension and register error,” *International IFAC Conference on Instrumentation and Automation in the Paper Rubber and Plastics Industry*, vol. 1, pp. 411–438, 1977.

[23] J. J. Shelton, “Dynamics of web tension control with velocity or torque control,” *Proceedings of the American Control Conference*, pp. 1423–1427, June 1986.

[24] K. H. Shin, “Distributed control of tension in multi-span web transport systems,” *Ph.D. thesis, Oklahoma State University, Stillwater, Oklahoma*, May 1991.

[25] G. Young and K. Reid, “Lateral and longitudinal dynamic behavior and control of moving webs,” *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 115, pp. 309–317, June 1993.

[26] W. Wolfermann, “Tension control of webs, a review of the problems and solutions in the present and future,” *Proceedings of the Third International Conference on Web Handling*, pp. 198–229, June 1995.

[27] H. Koc, D. Knittel, M. de Mathelin, and G. Abba, "Modeling and robust control of winding systems for elastic webs," *IEEE Transactions on Control Systems Technology*, vol. 10, pp. 197–208, March 2002.

[28] C. Aboky, G. Sallet, and J. C. Vivalda, "Observers for lipschitz nonlinear systems," *International Journal of Control*, vol. 75, no. 3, pp. 204–212, 2002.

[29] R. Byers, "A bisection method for measuring the distance of a stable matrix to the unstable matrices," *SIAM Journal of Scientific and Statistical Computing*, vol. 9, no. 3, pp. 875–881, 1988.

[30] P. Vas, *Sensorless vector and direct torque control*. Oxford University Press, 1998.

[31] A. M. Trzynadlowski, *Control of Induction motors*. Academic Press, 2001.

APPENDIX A

MATLAB/SIMULINK Programs

A.1 M-files for computations in Chapter 4

MATLAB programs used in the design of $\bar{A}_i$ and A_{mi} for the state feedback controller and MRAC, respectively, are provided. To find $\min_{\omega \in R} \sigma_{min}(\bar{A}_i - j\omega I)$, the function $dist_sing(A, C, prec)$ is also provided.

1. To design $\bar{A}_i$:

```
clc
close all
clear all
%
%System parameters and reference values%
N=3 % Total number of subsystems-1
L=[20,33,67];%Span lengths in inches
Ea=2090;%Product E*A in lbf
bf=1;%Coefficient of friction
J=[8 2 2 4];%Roller inertias in lbf/in^2
R=[0.75 0.339 0.339 0.5];%Roller radii in ft
tr=[0 20 20 20];%Reference tensions in different spans in lbf
vr(2)=20;%Master-speed reference in ft/sec
vr(1)=(tr(2)-Ea)*vr(2)/(tr(1)-Ea);
vr(3)=(Ea-tr(2))*vr(2)/(Ea-tr(3));
```

```
vr(4)=(Ea-tr(3))*vr(3)/(Ea-tr(4));
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%System Matrices:Open loop and desired closed loop%
%Actual open loop system matrix (Unwind)
A0=[-vr(2)/L(1) (tr(1)-Ea)/L(1);R(1)^2/J(1) -bf/J(1)];
B0=[0;R(1)/J(1)];%Input vector (Unwind)
%desired closed loop system matrix (Unwind)
Abar0=[-vr(2)/L(1) (tr(1)-Ea)/L(1);12 -20];
%
A1=-bf/J(2);
Abar1=-4000;%desired closed loop system matrix (Master-speed)
B1=R(2)/J(2);%Input vector (Master-speed)
%
%Actual open loop system matrix (Process)
A2=[-vr(3)/L(2) (Ea-tr(3))/L(2);-R(3)^2/J(3) -bf/J(3)];
B2=[0;R(3)/J(3)];%Input vector (Process)
%Desired closed loop system matrix (Process)
Abar2=[-vr(3)/L(2) (Ea-tr(3))/L(2);-1500 -400];
%
%Actual open loop system matrix (Rewind)
A3=[-vr(4)/L(3) (Ea-tr(4))/L(3);-R(4)^2/J(4) -bf/J(4)];
B3=[0;R(4)/J(4)];%Input vector (Rewind)
%Desired closed loop system matrix (Rewind)
Abar3=[-vr(4)/L(3) (Ea-tr(4))/L(3);-15 -15];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Interconnecting matrices%
A01=[Ea-tr(1)/L(1);0];
```

```
A02=[0 0;0 0];
A03=[0 0;0 0];
A10=[-R(2)*R(2)/J(2) 0];
A12=[R(2)*R(2)/J(2) 0];
A13=[0 0];
A20=[vr(2)/L(2) 0;0 0];
A21=[(tr(1)-Ea)/L(2);0];
A23=[0 0;R(3)*R(3)/J(3) 0];
A30=[0 0;0 0]; A31=[0;0];
A32=[vr(3)/L(3) (tr(2)-Ea)/L(3);0 0];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Calculations for the condition checks%
%Unwind Section
epsilon0=10;
eta10=max(svd(A10));
eta20=max(svd(A20));
xi0_square=eta10^2+eta20^2;
check0=sqrt(N*(xi0_square+epsilon0))
%Call for the function, which calculates
%minimum of singular value.
sigma_min_unwind=dist_sing(Abar0',0,1e-8)

%Master-speed section
epsilon1=10;
eta01=max(svd(A01));
eta21=max(svd(A21));
xi1_square=eta01^2+eta21^2;
```

```
check1=sqrt(N*(xi1_square+epsilon1))
%Call for the function, which calculates
%minimum of singular value.
sigma_min_master=dist_sing(Abar1',0,1e-8)



%Process section
epsilon2=10;
eta12=max(svd(A12));
eta32=max(svd(A32));
xi2_square=(eta12)^2+(eta32)^2;
check2=sqrt(N*(xi2_square+epsilon2))
%Call for the function, which calculates
%minimum of singular value.
sigma_min_process=dist_sing(Abar2',0,1e-8)



%Process section
epsilon3=10; eta23=max(svd(A23)); xi3_square=(eta23)^2;
check3=sqrt(N*(xi3_square+epsilon3))
%Call for the function, which calculates
%minimum of singular value.
sigma_min_rewind=dist_sing(Abar3',0,1e-8)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

2. To compute $\min_{\omega \in R} \sigma_{min}(\bar{A}_i - j\omega I)$:

```
%    prec: precision (e.g. 1e-10)
%   This algorithm makes use of bisection algorithm first given in
%   R. Byers, A bisection method for measuring the distance of a
%   stable matrix to the unstable matrices,
%   SIAM Journal of Scientific and Statistical
%   Computing, vol. 9, no. 3, pp. 875881, 1988.
%%%%%                                          %%%%%%%%%%%%%%%%%%%
%   Following MATLAB program is provided in
%   Observers for Lipschitz non-linear systems
%   C. Aboky, G. Sallet and J.C. Vivalda
%   Int. J. Control, 2002, vol75, no.3, 204--212
%
function out=dist_sing(A,C, prec) a=0; b=norm(A,2);
N=ceil(log2(b/prec))*2; n=length(A); In=eye(n);

for j=1:N,
   gamma=(a+b)/2;
   H_gamma=[A In;C'*C-gammaˆ2*In -A'];
   if min(abs(real(eig(H_gamma))))<=prec  %~=0
      b=gamma;
   else
      a=gamma;
   end
end

out=(a+b)/2;
```

3. To design A_{mi} in MRAC scheme:

```
clc
close all
clear all
%
%System parameters and reference values%
N=3 % Total number of subsystems-1
L=[20,33,67];%Span lengths in inches
Ea=2090;%Product E*A in lbf
bf=1;%Coefficient of friction
J=[8 2 2 4];%Roller inertias in lbf/in^2
R=[0.75 0.339 0.339 0.5];%Roller radii in ft
%Reference tensions in different spans in lbf
tr=[0 20 20 20];
vr(2)=40;%Master-speed reference in ft/sec
vr(1)=(tr(2)-Ea)*vr(2)/(tr(1)-Ea);
vr(3)=(Ea-tr(2))*vr(2)/(Ea-tr(3));
vr(4)=(Ea-tr(3))*vr(3)/(Ea-tr(4));
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%System Matrices:Open loop and desired closed loop%
%Actual open loop system matrix (Unwind)
A0=[-vr(2)/L(1) (tr(1)-Ea)/L(1);R(1)^2/J(1) -bf/J(1)];
B0=[0;R(1)/J(1)];%Input vector (Unwind)
%desired closed loop system matrix (Unwind)
Am0=[-vr(2)/L(1) (tr(1)-Ea)/L(1);10 -20];
%
```

```
A1=-bf/J(2);
Am1=-4000;%desired closed loop system matrix (Master-speed)
B1=R(2)/J(2);%Input vector (Master-speed)
%
%Actual open loop system matrix (Process)
A2=[-vr(3)/L(2) (Ea-tr(3))/L(2);-R(3)^2/J(3) -bf/J(3)];
B2=[0;R(3)/J(3)];%Input vector (Process)
%Desired closed loop system matrix (Process)
Am2=[-vr(3)/L(2) (Ea-tr(3))/L(2);-1500 -400];
%
%Actual open loop system matrix (Rewind)
A3=[-vr(4)/L(3) (Ea-tr(4))/L(3);-R(4)^2/J(4) -bf/J(4)];
B3=[0;R(4)/J(4)];%Input vector (Rewind)
%Desired closed loop system matrix (Rewind)
Am3=[-vr(4)/L(3) (Ea-tr(4))/L(3);-15 -15];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Interconnecting matrices%
A01=[Ea-tr(1)/L(1);0];
A02=[0 0;0 0];
A03=[0 0;0 0];
A10=[-R(2)*R(2)/J(2) 0];
A12=[R(2)*R(2)/J(2) 0];
A13=[0 0];
A20=[vr(2)/L(2) 0;0 0];
A21=[(tr(1)-Ea)/L(2);0];
A23=[0 0;R(3)*R(3)/J(3) 0];
A30=[0 0;0 0];
```

```
A31=[0;0];
A32=[vr(3)/L(3) (tr(2)-Ea)/L(3);0 0];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Reference trajectory generation and
%the design of overall gain matrix km
Am=[Am0 A01 A02 A03;
    A10 Am1 A12 A13;
    A20 A21 Am2 A23;
    A30 A31 A32 Am3];

B=[B0 [0 0 0;0 0 0];
    0 B1 0 0;
   [0 0;0 0] B2 [0;0];
    [0 0 0;0 0 0] B3];
Q=[1000 0 0 0 0 0 0;0 10 0 0 0 0 0;
   0 0 1 0 0 0 0;0 0 0 100 0 0 0;
   0 0 0 0 900 0 0;0 0 0 0 0 4000
   0;0 0 0 0 0 0 9000];

R=[0.003 0 0 0;0 0.02 0 0;0 0 0.001 0;0 0 0 0.001];

[km,P,E]=lqr(Am,B,Q,R); Eg=-10*[2;1;1;5;5;1;2]; C=eye(7);
D=zeros(7,4); sys=ss((Am-B*km),B,C,D);

G=expm((Am-B*km)*5e-3);%System matrix after discritization
x(:,1)=[5;2;2;5;2;5;2];
```

```
t(1)=0;

for k=1:50
x(:,k+1)=G*x(:,k);
    t(k+1)=k*5e-3;
end

subplot(7,1,1) plot(t,x(1,:),'b-'); grid; ylabel('T_{r1}')
subplot(7,1,2) plot(t,x(2,:),'b-'); grid; ylabel('V_{r0}')
subplot(7,1,3) plot(t,x(3,:),'b-'); grid; ylabel('V_{r1}')
subplot(7,1,4) plot(t,x(4,:),'b-'); grid; ylabel('T_{r2}')
subplot(7,1,5) plot(t,x(5,:),'b-'); grid; ylabel('V_{r2}')
subplot(7,1,6) plot(t,x(6,:),'b-'); grid; ylabel('T_{r3}')
subplot(7,1,7) plot(t,x(7,:),'b-'); grid; ylabel('V_{r3}')
xlabel('time (sec)')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Calculations for the condition checks%
%Unwind Section
epsilon0=10; eta10=max(svd(A10));
eta20=max(svd(A20));
xi0_square=eta10^2+eta20^2;
check0=sqrt(N*(xi0_square+epsilon0))
%Call for the function, which calculates
%minimum of singular value.
sigma_min_unwind=dist_sing(Am0',0,1e-8)

%Master-speed section
```

```
epsilon1=10; eta01=max(svd(A01));
eta21=max(svd(A21));
xi1_square=eta01^2+eta21^2;
check1=sqrt(N*(xi1_square+epsilon1))
%Call for the function, which calculates
%minimum of singular value.
sigma_min_master=dist_sing(Am1’,0,1e-8)

%Process section
epsilon2=10; eta12=max(svd(A12)); eta32=max(svd(A32));
xi2_square=(eta12)^2+(eta32)^2;
check2=sqrt(N*(xi2_square+epsilon2))
%Call for the function, which calculates
%minimum of singular value.
sigma_min_process=dist_sing(Am2’,0,1e-8)

%Rewind section
epsilon3=10; eta23=max(svd(A23)); xi3_square=(eta23)^2;
check3=sqrt(N*(xi3_square+epsilon3))
%Call for the function, which calculates minimum
%of singular value.
sigma_min_rewind=dist_sing(Am3’,0,1e-8)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Before computing the gains, note the output
%in Matlab window to
%check whether the conditions are satisfied or not.
%If conditions are not satisfied for i-th subsystem
```

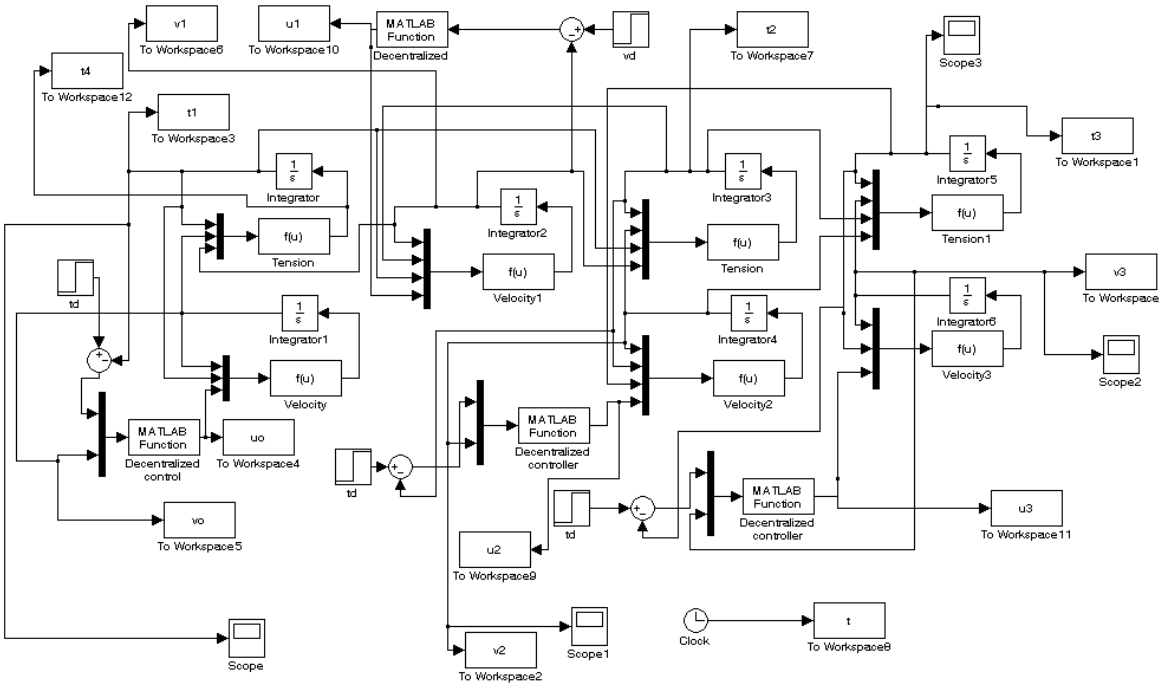
```
%then change the second row of ‘‘Ami’’.
%After iterative process if the conditions are
%satisfied then solve ARE as below.

P0=are(Am0,-N*eye(2),(xi0_square+epsilon0)*eye(2))
Q0=Am0’*P0+P0*Am0+N*P0*P0+(xi0_square+epsilon0)*eye(2);
E0=eig(Q0)

P1=are(Am1,-N,(xi1_square+epsilon1))
Q1=Am1’*P1+P1*Am1+N*P1*P1+(xi1_square+epsilon1);
E1=eig(Q1)

P2=are(Am2,-N*eye(2),(xi2_square+epsilon2)*eye(2))
Q2=Am2’*P2+P2*Am2+N*P2*P2+(xi2_square+epsilon2)*eye(2);
E2=eig(Q2)

P3=are(Am3,-N*eye(2),(xi3_square+epsilon3)*eye(2))
Q3=Am3’*P3+P3*Am3+N*P3*P3+(xi3_square+epsilon3)*eye(2);
E3=eig(Q3)
```



Simulink block diagram for decentralized control of HSWL

A.2 Simulink block diagram

APPENDIX B

Dynamic Model Parameters and Calibration

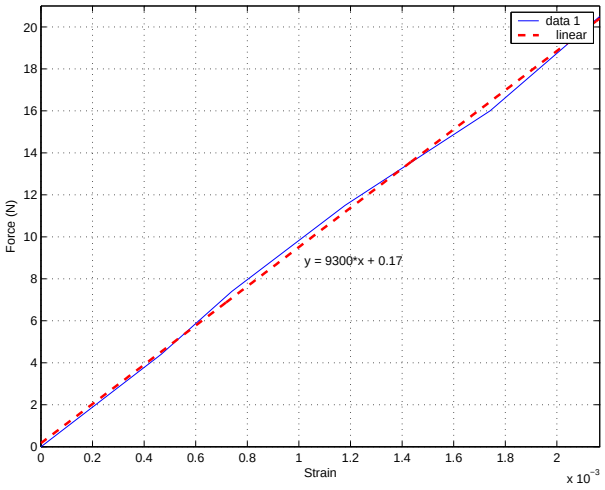
B.1 Model Parameters

Tensile testing experiments were carried out to find the modulus of elasticity of the web material. The web material used is Tyvec, which is made by Dupont. In tensile testing, predetermined load was applied on a web with the un-stretched length equal to 36.4744 m and resulting change in length, δL, was recorded. Table B.1 shows the experimental observations from the tensile test.

The graph of Force (N) against strain is obtained as shown in Fig. B.1. The linear curve fit shows the approximate value of EA to be 9300 N or 2090 lbf

OBS No.	Force (N)	δL (mm)	Strain ($\times\ 10^{-4}$)	EA (N)
0	0	0	0	-
1	4.4	17	4.66	9442.06
2	7.4	27	7.4	10000
3	11.5	43	11.789	9754.856
4	16.0	63.5	17.409	9190.649
5	20.5	79	21.659	9464.888

Table B.1: Tensile test on web material



Note that the mathematical model requires the product of elasticity E and area of cross-section A. Taking advantage of this, EA is obtained directly, which eliminates the possible error in the evaluation of A from web thickness t.

B.2 Calibration

Actual physical variables have different values than corresponding variables used in the AUTOMAX program. Hence, the same gains and equilibrium controllers as obtained above cannot be used directly but need to be scaled properly. This scaling depends upon the relation between actual physical variables and corresponding program variables. The relation between actual physical variables and their corresponding program variables in AUTOMAX are given below.

Tension: $UN\_LOAD\_CELL\%$, $NP2\_LOAD\_CELL\%$, $WN\_LOAD\_CELL\%$ are the variables used in the AUTOMAX program to represent tension in the unwind, process and rewind sections, respectively. To get the actual tension, t_i in lbf, from the program variables, a scaling factor of 0.022 is used. For example, tension in the unwind zone is given by $t_1 = 0.022 * UN\_LOAD\_CELL\%$.

Velocity: $UN\_SPD\_FDBK\%$, $NP1\_SPD\_FDBK\%$, $NP2\_SPD\_FDBK\%$, and $WN\_SPD\_FDBK\%$ represent the program variables for web velocity in ft/min. It may be

noted that the velocity in the controller design is used in ft/sec and hence each program variable needs to be multiplied by $1/60$.

Diameter: The diameter of the unwind and rewind roll are sensed by the ultrasonic sensors and they are represented by $UN\_DIA\_SC\%$ and $WN\_DIA\_SC\%$ respectively. These program variables need to be multiplied by 0.01 to get the actual diameter in inches.

Torque: The input torques to the motors are represented by the program variables $UN\_REFERENCE3\%$, $NP1\_REFERENCE3\%$, $NP2\_REFERENCE3\%$, and $WNC\_REFERENCE3\%$ for unwind, master speed, process and rewind rollers respectively. But the actual torques are found out to be 20 times more than the values represented by these variables. Hence, the control inputs calculated in terms of the program variables are reduced by the factor $1/20$.

APPENDIX C

Step-by-Step Algorithm for Decentralized Controller Development

(1) Decompose the given web handling plant into subsystems such as unwind section, master speed section, process section(s) and rewind section. The decomposition procedure is explained in Chapter 3. A typical web processing line is shown in Fig. C.1, which has many process sections in between the master speed section and the rewind section. Total $(N+1)$ rollers are shown in which the 0-th roll is assigned to unwind roll and the N-th roll is the rewind roll.

(2) Obtain the dynamic model for each section in matrix form as demonstrated in Chapter 3.

(a) The dynamic model for the unwind section can be written as

$$\dot{x}_0 = \begin{bmatrix} \dot{T}_1 \\ \dot{V}_0 \end{bmatrix} = A_0 x_0 - B_0 U_0 - B_0 f_0(V_0) + \sum_{j\neq 0, j=1}^{N} A_{0j} x_j \tag{C.1}$$

where

$$A_0 = \begin{bmatrix} -v_{r1}/L_1 & (t_0 - AE)/L_1 \\ R_0^2/J_0 & -b_{f0}/J_0 \end{bmatrix},\ B_0 = \begin{bmatrix} 0 \\ \dfrac{n_0 R_0}{J_0} \end{bmatrix} \tag{C.2}$$

$$A_{01} = \begin{bmatrix} \dfrac{AE - t_{r1}}{L_1}, & 0 \end{bmatrix}^T,\ f_0(V_0) = \frac{t_w}{2\pi n_0 R_0}\left(\frac{J_0}{R_0^2} - 2\pi\rho_w b_w R_0^2\right)(V_0^2 + 2v_{r0}V_0)$$

Remaining A_{ij} matrices are null matrices with proper dimensions.

(b) The dynamic model of the master speed section can be written as

$$\dot{x}_1 = \dot{v}_1 = A_1 x_1 + B_1 U_1 + \sum_{j=0, j\neq 1}^{3} A_{1j} x_j \tag{C.3}$$

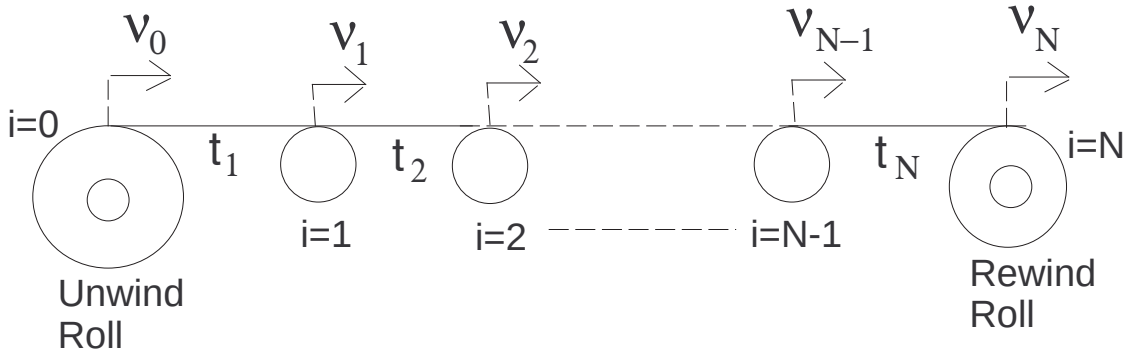


Figure C.1: A typical web processing line

where

$$A_1 = -\frac{b_{f1}}{J_1}, \quad B_1 = \frac{n_1 R_1}{J_1},$$
$$A_{10} = \left[\frac{-R_1^2}{J_1}, 0\right], \; A_{12} = \left[\frac{R_1^2}{J_1}, 0\right]$$

Remaining A_{ij} matrices are null matrices with proper dimensions.

(c) The dynamic model for each process section can be written as

$$\dot{x}_i = \begin{bmatrix} \dot{T}_i \\ \dot{V}_i \end{bmatrix} = A_i x_i + B_i U_i + \sum_{j \neq i, j=0}^{N} A_{ij} x_j \tag{C.4}$$

where $i = 2, 3, \ldots, N-1$,

$$A_i = \begin{bmatrix} -v_{ri}/L_i & (AE - t_{ri})/L_i \\ -R_i^2/J_i & -b_{fi}/J_i \end{bmatrix} B_i = \begin{bmatrix} 0 \\ \frac{n_i R_i}{J_i} \end{bmatrix} \tag{C.5}$$
$$A_{ii-2} = \begin{bmatrix} \frac{v_{ri-1}}{L_i} & 0 \\ 0 & 0 \end{bmatrix}, \; A_{ii-1} = \left[\frac{t_{ri-1} - AE}{L_i}, 0\right]^T, \; A_{ii+1} = \begin{bmatrix} 0 & 0 \\ \frac{R_i^2}{J_i} & 0 \end{bmatrix}$$

The remaining A_{ij} matrices are null matrices with proper dimensions.

(d) The dynamic model for the rewind section is given by

$$\dot{x}_N = \begin{bmatrix} \dot{T}_N \\ \dot{V}_N \end{bmatrix} = A_N x_N + B_N U_N + B_N f_N(V_N) + \sum_{j \neq N, j=0}^{N} A_{Nj} x_j \tag{C.6}$$

where

$$A_{NN-1} = \begin{bmatrix} \frac{v_{rN-1}}{L_N} & \frac{t_{rN-1}-AE}{L_N} \\ 0 & 0 \end{bmatrix}$$

$$f_N(V_N) = \frac{t_w}{2\pi n_N R_N}\left(\frac{J_N}{R_N^2} - 2\pi\rho_w b_w R_N^2\right)(V_N^2 + 2v_{rN}V_N)$$

The remaining A_{ij} matrices are null matrices with proper dimensions.

(3) Once the dynamic model is obtained for each subsystem, design $\bar{A}_i$ and gain vectors K_i, following the two step procedure, which is described in section 4.2. Note that for the rewind and unwind rolls these gains need to be computed on-line using instantaneous values of their radii.

(4) Compute the angular velocity of each driven roller on-line, by differentiating encoder signal from corresponding motor.

(5) Using computed angular velocity of unwind (rewind) roll calculate the radius of unwind roll by integrating the following equation

$$\dot{R}_0 \approx -\frac{t_w}{2\pi}\omega_0 \tag{C.7}$$

For the rewind roll, radius is calculated using

$$\dot{R}_N \approx \frac{t_w}{2\pi}\omega_N \tag{C.8}$$

The trapezoidal rule may be used in the program to carry out numerical integration. At the start of the controller execution initial value of the roll-radius must be provided to initiate numerical integration. This can be done with the help of the ultrasonic sensor mounted on the unwind (rewind) stands.

(6) Using the radius of each roller/roll, R_i, and calculated angular velocity ω_i, compute linear velocity using $v_i = R_i\omega_i$.

(7) Errors in tension, T_i, and velocity, V_i, are required to be calculated on-line using $T_i = t_i - t_{ri}$ and $V_i = v_i - v_{ri}$. Calculate velocity references for each section depending upon master speed reference velocity and tension references. The tension reference in each zone and velocity reference of master speed roller is set by the operator. For the unwind roll, the reference velocity is given by

$$v_{r0} = \frac{AE - t_{r1}}{AE - t_0} v_{r1} \tag{C.9}$$

For the process section driven rollers and the rewind roll, calculate reference velocity using the following equation:

$$v_{ri} = \left(\frac{AE - t_{r1}}{AE - t_{ri}} \right) v_{r1} \tag{C.10}$$

(8) Then use the decentralized control input for each section as $\tau_i = \tau_{ieq} + U_i$ where τ_{ieq} are equilibrium inputs.

(a) For the unwind subsystem,

$$\tau_{0eq} = -\frac{b_{f0}}{n_0 R_0} v_{r0} + \frac{R_0}{n_0} t_{r1} - \frac{t_w}{2\pi n_0 R_0} \left(\frac{J_0}{R_0^2} - 2\pi \rho_w b_w R_0^2 \right) v_{r0}^2 - \frac{J_0}{n_0 R_0} \dot{v}_{r0} \tag{C.11}$$

$$U_0 = -f_0(V_0) + K_0 [T_0,\ V_0]^T \tag{C.12}$$

(b) For the master speed roller,

$$\tau_{1eq} = \frac{b_{f1}}{n_1 R_1} v_{r1} + \frac{R_1}{n_1} t_{r1} \tag{C.13}$$

$$U_1 = -K_1 V_1 \tag{C.14}$$

(c) For all other driven rollers in the process section,

$$\tau_{ieq} = \frac{b_{fi}}{n_i R_i} v_{ri} - \frac{R_i}{n_i} (t_{ri+1} - t_{ri}) \tag{C.15}$$

$$U_i = -K_i [T_i,\ V_i]^T \tag{C.16}$$

(d) For the rewind subsystem,

$$\tau_{Neq} = \frac{b_{fN}}{n_N R_N} v_{rN} + \frac{R_N}{n_N} t_{rN} - \frac{t_w}{2\pi n_N R_N} \left(\frac{J_N}{R_N^2} - 2\pi \rho_w b_w R_N^2 \right) v_{rN}^2 + \frac{J_N}{n_N R_N} \dot{v}_{rN} \tag{C.17}$$

$$U_N = -f_N(V_N) - K_N x_N \tag{C.18}$$

(9) The design of model reference decentralized adaptive controllers:

Follow same steps as explained above with exception that in Step 3, instead of $\bar{A}_i$, reference system matrices A_{mi} need to be chosen and in Step 8, estimated gains $\hat{K}_i$ are required to be used in the feedback control law U_i. Follow the three steps given in section 4.3 to design A_{mi}, choose initial value $\hat{K}_i(0)$ and adaptation law $\dot{\hat{K}}_i$.

(10) Computations of control inputs, τ_i, must be done keeping appropriate scaling between programmed variables and actual physical variables. This aspect is discussed in Appendix B.2

APPENDIX D

CONTROLLER HARDWARE IN AUTOMAX SYSTEM

The objective of web speed and tension control boils down to a control of speed of all motors based on the the tension and velocity feedback. For different motors different strategies are used. For motors M_0 and M_3 the control block diagram is as shown in Fig. 4.1. As seen there are two loops namely inner loop for velocity control and outer loop for tension control. For an outer loop, there are two options for tension feedback viz. feedback depending upon the dancer position or feedback from load-cell. Outer loop controller generates correction term, which is then added to the reference speed. This new required velocity is compared with feedback velocity. If there is no change in the tension from reference value then there will be zero correction term and hence speed will also be maintained at reference value by inner loop. Inner and outer loop controllers are implemented using Automax control system. In nip station number 1, motor M_1 is master speed motor as it is not controlled with the tension feedback but only with inner velocity loop.

D.0.1 Adjustable Speed Drives

All motors are three phase induction motors. Velocity Control of induction motors is much more complex than DC motors. But still they are popular as industrial drives because of rugged construction and low cost for applications demanding any power range. Vector control method is implemented for each motor. Details about vector control as well as dynamic model of the induction motor are discussed in [30] and [31]. Invertron (VCI) is used as driver for each motor as a controller except motors M2 and M3 for which HR2000 is used. The vector controller provides signal (current) for corresponding motor depending

upon the control signal from Automax. Rack A01 has the following drive units in it: Regen Unit, Nip station ♯1 lead drive, Unwind drive, Unwind carriage drive, Nip station ♯1 drive ♯2, Nip station ♯1 drive ♯3. And rack A00 has the following drive units in it: Winder carriage drive, Winder surface drive, Winder center drive, Nip station ♯2 drive. The feedback control loops in the driver or vector controller are very fast and hence have negligible effect on the transient response of entire plant.

D.0.2 Role of Auotomax in Control System

Variety of tasks are performed in Automax, which is a programmable, micro-processor based control system capable of performing real time control with millie-second response time. Automax system is modular so one can customize the system to meet specific requirements of the application. The design of a system allows maximum of 43 racks, each containing at least one processor module, to be connected together as a part of control network using Network Communication modules. In the HSWL application two racks, A00 and A01, are used with four and two processor modules respectively.

Feedback options as well as different constant values of various parameters like reference velocity, tension reference, material and geometrical specifications of web are entered in to a control panel. All this information is transformed to a processor in a rack A01. Each of these values as well as other common variables used in the control algorithm are accessible to all processors simply by referencing the appropriate variable name in a task. The information will be shared with all two processors in rack A01 as well as all the four processors in rack A00. Application programs or tasks are created off-line in MS-DOS or MS-windows environment using an 80386 compatible personal computer. These tasks are compiled and then transferred to rack A01 via direct link. Different tasks are performed in different processor modules. The tasks to be performed by processors in rack A00 are transferred via network communication link from rack A01. To write the task algorithms Automax supports three different programming languages:Ladder logic language, control

block language or Enhanced BASIC language. Each of these languages is suited to different type of tasks commonly found in industrial and process control environment.

Sequential tasks like checking On/Off switch position, emergency stop after fault detection, etc. are written in ladder logic language.

The controller algorithms are written in Control Block language (.blk format). Depending upon the references entered in control panel, AUTO-Max algorithm generates suitable reference values for compatible comparison with feedback signal from the sensor. The filtering and proper scaling (if required) of feedback signal is also done using algorithm written in .blk format. The error signal after comparison is used by controller algorithm written for each motor. Special function calls like integrators, function generators, PID controller, etc. are used to develop these control algorithms.

The enhanced BASIC language is used for keyboard and CRT (Cathode Ray Tube)-based operator interfaces and numeric processing.

For a better clarification of control strategy, control of motor M0 is discussed in detail in next section.

D.1 Control of motor M0

Detailed control block diagram for motor M0 is shown in figure D.1. As mentioned earlier Invertron is used as drive for this motor. Ladder of circuits is implemented in this drive which is responsible for all sequential operations in it. As an example, thermal protection ladder uses thermostat of the motor which stops the current to motor if motor is overheated. Also there are different circuit ladders to flash indicator light indicating the correct functioning of various tasks.

As shown in the figure, drive also has to perform very important task of vector control of motor M0 which has current loop and torque loop in it. The vector control method is discussed in next subsection. The reference torque value for torque loop in vector control is generated by control algorithm in Automax. Algorithm uses same structure of inner veloc-

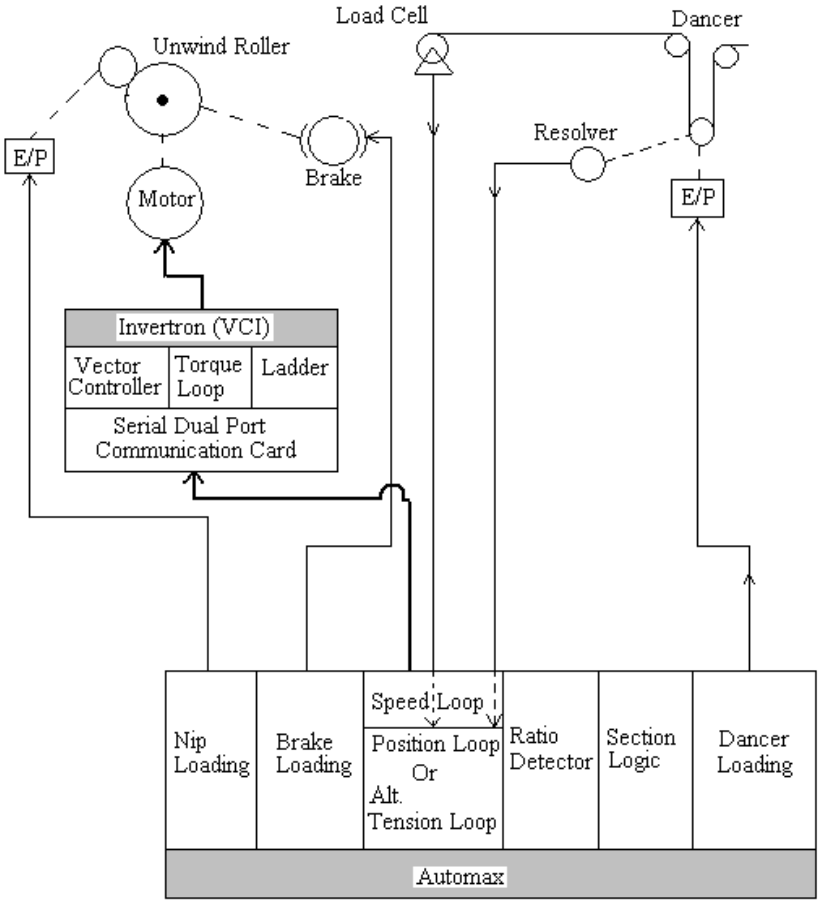


Figure D.1: Feedback Control Block diagram

ity loop and outer tension loop as shown in figure 4.1. There are two separate algorithms for tension control corresponding to two choices of tension feedback methods namely dancer position feedback and direct feedback from load cell. This option is required to be selected on a control panel. Control panel sends the binary variable (i.e. 1 or 0) showing which option is chosen by the user. Control algorithm makes use of this variable so that at a time one tension loop is active while other is inactive.

Depending upon reference tension, Tref, entered in control panel, Automax has to set force acting on dancer roller to a correct value. The algorithm written in Automax uses the equilibrium condition for generating loading signal. For S-wrap dancer loading has to balance torque due to tensions on both rollers. Similarly, there are separate Automax

algorithms for giving nip loading, brake loading signals.

As web unwinds from the unwind roller to rewind roller, a time may arrive when there will be no web on unwind roller. In this situation web will be broken. To avoid this, ratio detector tracks radii of both unwind and rewind rolls. Automax algorithm for ratio detector is written which calculates diameter of unwind and rewind coils. Ratio detector makes use of actual speed of the web, initial diameter of the roller with coil on it and thickness of the web.

Also ultrasonic sensors are provided under each web coil roller. This is an extra option for checking actual radii of each roller with web coil. Sensor transmits ultrasonic wave which is made incident on coil and then repelled ultrasonic wave is collected. The time lapsed in between transmission and reception of wave decides the distance between coil diametrical surface and fixed sensor. Diameter of the coil and distance detected are proportionally related and thus coil diameter can be tracked.

'Section-logic' implements ladder programs, which perform all the sequential tasks (different than those performed by Invertron). These are basically on/off type of tasks.

D.1.1 Vector Control

General block diagram of a vector control is shown in figure D.2. Vector control or Adjustable speed drive (ASD) needs variable frequency source corresponding to different speeds of the motor. For this 'invertor' is used which is a dc to ac convertor and dc power for invertor is supplied by rectifier which is fed from the ac line through a capacitor filter. Pulse Width Modulation technique is used to control the output voltages of the invertor.

Three phase induction motor has a three components of a current. This three phase quantity (denoted as 'abc' format) is expressed in a space vector form (denoted as 'dq' format). The vectorial representation uses direct(d)-quadrature(q) frame of reference. The three phase quantity is expressed in these d and q components. Vector controller has current loop which compares 'required' current components (in d-q form) with 'actual' current

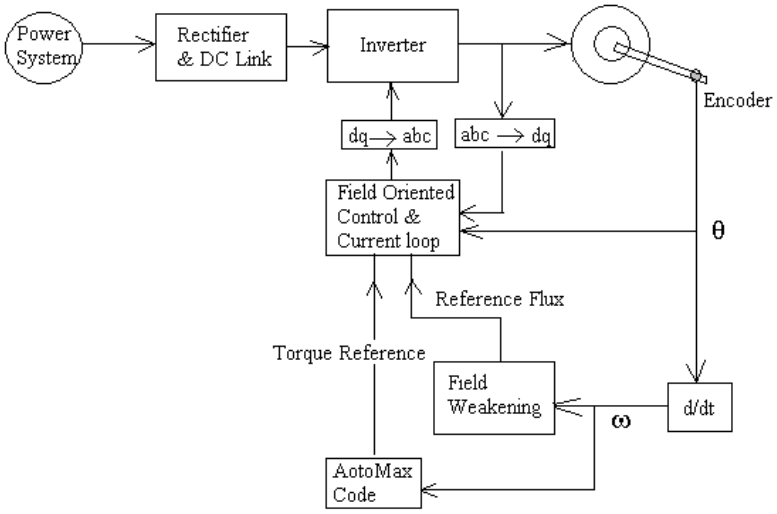


Figure D.2: Vector Control Block diagram

components (also in d-q form). Current sensors are used to get actual current values in 'abc' format which are then converted into required 'dq' vector format. The current feedback circuit also ensures that circuit current is not exceeding the specified maximum value.

The required or reference current components are provided by field oriented controller [30]. Torque produced by the motor is given [31] as,

$$T_M = \frac{2}{3} P_p \frac{L_m}{L_r} Im\{i_s \lambda_r^*\} \tag{D.1}$$

Where, T_M is developed torque, P_p is the number of pole pairs, i_s is a stator current vector, λ_r^* is the rotor flux space vector, L_m and L_r are magnetizing inductances of stator and roller respectively. Also

$$Im\{i_s \lambda_r^*\} = i_s \lambda_r sin[\angle(i_s, \lambda_r)]$$

Thus dynamics of torque developed is dependent on the dynamics of rotor (or stator)flux

vector as well as stator current vector. In case of induction motor, field is revolving and not stationary like dc motor. That is why only magnitude control is not enough but field orientation i.e. angle also need to be controlled and that is why the control is known as vector control. The main idea of the field oriented control is to align direct(d) axis of the revolving reference frame with rotor flux vector i.e.λ_r so that induction motor emulates the DC motor. Given a reference torque (from Automax algorithm) and a reference flux-vector to be developed, field oriented control will generate required current components to be used by current loop. Here rotor speed is used to obtain the reference flux value. Reference flux is generated such that it ensures that the stator voltage under the field weakening conditions will not exceed the rated value. The torque reference is produced by Automax control algorithm. Feedback from the encoder is the angular position of the rotor shaft which is then differentiated to get angular speed. The speed is also needed to the controller algorithm written in Auto-Max. Invertron not only drives motor but also gives necessary speed feedback to Automax controller.

VITA

Nilesh B. Siraskar

Candidate for the Degree of

Master of Science

Thesis: DECENTRALIZED CONTROL OF LARGE-SCALE INTERCONNECTED SYSTEMS WITH APPLICATION TO WEB PROCESSING MACHINES

Major Field: Mechanical Engineering

Biographical:

Personal Data: Born in Maharashtra, India, on May 6, 1981, the son of Babanrao M. Siraskar and Suman B. Siraskar.

Education: Received the B.E. degree from Government College of Engineering, Pune, Maharashtra, India, in 2002, in Mechanical Engineering; Completed the requirements for the Master of Science degree with a major in Mechanical Engineering at Oklahoma State University in December, 2004.

Experience: Research Assistant at Oklahoma State University from January 2003-Present; Teaching Assistant at Oklahoma State University from January 2003-Present; Project Engineer at Tata Engineering and Locomotive Co., Ltd., Maharashtra, India from Jan 2001 to May 2002.

Professional Memberships: American Society of Mechanical Engineers; Member of Mensa.

Name: Nilesh B. Siraskar Date of Degree: DECEMBER, 2004

Institution: Oklahoma State University Location: Stillwater, Oklahoma

Title of Study: DECENTRALIZED CONTROL OF LARGE-SCALE INTERCONNECTED SYSTEMS WITH APPLICATION TO WEB PROCESSING MACHINES

Pages in Study: 105 Candidate for the Degree of Master of Science

Major Field: Mechanical Engineering

Scope and Method of Study: The thesis investigates decentralized control of a class of large-scale nonlinear systems, which contain unmatched, linear and nonlinear interconnections between constituent subsystems. Adaptive methods and Lyapunov analysis are used to develop stable decentralized control algorithms. Improvements to existing dynamic models of fundamental web handling elements are suggested and discussed. A large experimental platform, which mimics most of the features of an industrial web process line, is used for experimentation with the proposed decentralized controller.

Findings and Conclusions: It is shown that decentralized control schemes can be developed for a class of unmatched large-scale systems. Sufficient conditions are developed that will result in closed-loop stability. For linear interconnections, these conditions ensure global stability and for nonlinear interconnections, they ensure semi-global stability. A model reference decentralized adaptive controller is developed for systems with unmatched, linear interconnections. To circumvent the unmatched problem, a new reference model that exchanges reference information between subsystems is proposed. Based on the improved dynamic models for unwind/rewind rolls, accurate inertia compensation terms are computed for on-line compensation. A strategy for obtaining the equilibrium control inputs and reference velocities is given based on the reference tension in each web tension zone and master-speed roller reference velocity. Extensive comparative experiments with the proposed decentralized controllers and an often used industrial decentralized PI controller show substantial reduction in web tension error with the proposed decentralized controllers.

ADVISOR'S APPROVAL: _____