

POLITECNICO DI TORINO

Corso di Laurea Magistrale in Mechatronic Engineering

Tesi di Laurea Magistrale

Analysis, Modelling And Control of a Sanforized Compactor Unit



Relatore

Prof. Mazza Luigi

Correlatore:

Prof. Rizzo Alessandro

Candidato

Ethelbert EZEMOBI N

ID number: s229017

Supervisore Aziendale

Mr. Alberto Priero

April 2018

Lock Flag

This master thesis contains confidential information which is property of Bianco S.P.A, Strada Tagliata 18 - 12051 Alba (CN) - Italy. This information is only accessible to authorised persons. Disclosure, publishing or duplication of the thesis or any parts of the thesis is not permitted without expressly authorization by the company, Bianco S.P.A, Strada Tagliata 18 - 12051 Alba (CN) - Italy.

It is not permitted to circulate the content of this thesis to unauthorised third parties or to make the content of this thesis accessible to unauthorised third parties.

Only the following persons are authorised to read this thesis without explicit consent:

1. First Corrector: Prof. Mazza Luigi
2. Second Corrector: Prof. Alessandro Rizzo

This lock flag ends five years after completion of this master thesis.

Date, place and signature

Summary

The general goal of the thesis is to optimize the automation of a unit section of Bianco SPA textile production line. Sanforizing compactor, being the unit section under consideration is responsible for a controlled compression and tension of fabrics to achieve desired dimensional stable compaction on a given fabric material.

Currently, the plant is being controlled with Proportional-Integral (PI) controller and it is subject to undesired tuning for any adjustment in the parameters. Due to non-linear system parameter variations, hugely contributed by wide variation in property of fabric material on which compaction is desired, there is a need to design a control system that can adapt to these variations.

Exploiting standard procedures, a detailed description of a strategy for modelling the Sanforized system using Set membership System Identification approach has been outlined. Also, using Bond-graph modelling procedure, an approximate state space equation model of the system was obtained.

For ease of control, the plant system was split in Stages I and II, and Model Predictive Controller objects were designed to control the modelled plant subsystems whose states are initialized with steady state operating point variables. The MPC controls the plant using manipulated variables obtained through optimisation and prediction of possible violation on the imposed constraints over a prediction horizon.

For adaptability, adaptive MPCs were designed on the basis of the MPC objects. The plant systems is estimated on-line using ARX algorithm. The estimated plant is converted into a state space model and used to updates the internal state of the MPC object at runtime.

The control systems were tested with both internal linearised plant and the external subsystems, and through review on QP Hessian Matrix Validity, Closed-Loop Internal Stability, Closed-Loop Nominal Stability, Closed-Loop Steady-State Gains, Hard MV Constraints, Other Hard Constraints, Soft Constraints, the system was ascertained to be consistent and satisfactory.

Acknowledgements

To my relentless professors and supervisors, Prof. Luigi Mazza and Prof. Alessandro Rizzo, the journey of this piece of accomplishment started with you. Staying close with the zeal to go extra miles is a gift I can't reciprocate. It wouldn't have been easier without your guidance and anyone would be privileged having passed through you.

To the entire family of Bianco Spa, I feel so honoured, having met you. You didn't just tell me that I am part of you, you showed me. The CEO, Dr. Stefano Boromeo, your modest personality and hard-work holds the family centred. The technical manager, Mr. Alberto Priero, the R&D managers, Engr. Giovanni Riccardi, and the Engineers, especially Pierluigi Passavanti, your wealth of experience and ideas were stimulating and transforming. I appreciate all the confidence you reposed on me. All my colleagues, your kind gestures are undying.

It is my privilege also to recognise Prof. Diego Regruto. You willingly gave me a seat in your class and emptied yourself to hand me with resourceful tools for optimum solutions. I am humbled by your kind gestures.

And finally to my loving family and all the amazing friends I have come to know within these study years, I have become this me because I have come to know you.

I am overwhelmed with an inexpressible nostalgia and right words have become too hard to find, but one thing cannot elude me. The joy of meeting you all has created a river of gratitude in me and I can't thank you enough.

Contents

List of Tables	VIII
List of Figures	IX
1 Thesis Opening	1
1.1 Motivation	1
1.2 State of the Art	2
1.2.1 Bianco Spa Textile Finishing Line	2
1.2.2 Proportional-Integrator (PI) Controller	2
1.2.3 Some Variable Parameter Considerations	3
1.3 Thesis outline	4
2 Description and Theory	5
2.1 Compaction	5
2.1.1 Compaction Theory	5
2.1.2 Basis Techniques of Compaction	6
2.1.3 Compaction Process	7
2.2 System Modelling	8
2.3 Robust System Identification	9
2.3.1 Important Definitions:	10
2.4 Parametric Identification of LTI System	10
2.5 System Identification with Error Added:	12
2.6 Error Structure:	12
2.6.1 Equation Error Structure:	12
2.6.2 Error-In-Variables:	14
2.6.3 Output Error Structure:	15
2.7 Parameter Estimators	16
2.7.1 Least Square (e_2 norm) Estimator:	17
2.7.2 L_∞ -norm Estimator:	18
2.8 Set-Membership (Unkonwn But Bounded error) Identification of LTI System	19
2.8.1 Set Membership Identification of Discrete Time LTI Model from Input Out Data Corrupted by Equation Error	19
2.8.2 Set Membership Identification of Discrete Time LTI Model from Input Out Data Corrupted by Output Error	22

2.8.3	Set Membership Identification of Discrete Time LTI Model from Input Out Data Corrupted by Error-In-Variable:	24
2.8.4	Set Membership Identification of Stable Discrete Time LTI System:	25
2.9	System Modelling By Bond-Graph Approach:	26
2.9.1	Introduction	26
2.9.2	Input - Output Relation:	26
2.9.3	Power Relation:	28
2.9.4	Basic 1-Port Elements	29
2.9.5	Basic 2-Port Elements:	36
2.9.6	Multi-Port Junction Elements:	37
2.10	System Control	40
2.10.1	Model Predictive Control(MPC)	40
2.10.2	Optimization Parameters	44
2.10.3	Future Set-point Trajectory:	46
2.10.4	Constraint MPC Formulation:	48
2.10.5	Steady State Target Computation	49
2.10.6	Moving Horizon Implementation	49
2.10.7	Quadratic Programming (QP)	49
2.10.8	Tuning of MPC	51
3	METHODOLOGY: System Modelling and Control of Sanforized Compactor	53
3.1	Specification Requirement:	53
3.2	System Modelling:	54
3.3	Stage I Compaction:	55
3.3.1	Subsystem Description	55
3.3.2	Model Parameters	56
3.3.3	Input / Output Variables	57
3.3.4	Subsystem Modelling with Bond-graph	57
3.3.5	Derivation of State Equation	59
3.4	Stage II Compaction	59
3.4.1	Subsystem Description	60
3.4.2	Stage II Plant Modelling By Set-Membership System Identification Approach	61
3.4.3	Stage II Plant Modelling With Bond-Graph	68
3.4.4	Derivation of State Equation:	69
3.5	Control System Design	69
3.6	Adaptive MPC Controller Design:	71
3.6.1	Creating Operating Point	71
3.6.2	MPC Internal Dynamic Plant Model (Prediction Model) Creation	74
3.6.3	MPC Object Design	75
3.6.4	Implemented Constraints on Manipulated Inputs and Outputs	75
3.6.5	Future Set-point Trajectory Determination	76
3.6.6	Constrained MPC Formulation	76

3.7	Controller Performance Tuning	78
3.7.1	Hard MV Constraints	79
3.7.2	Soft Constraints	79
3.7.3	Adaptive Model Predictive Controller	79
4	Results and Findings	83
4.1	Evaluate MPC performance with linear plant	83
4.1.1	Stage I	83
4.1.2	Stage II	84
4.2	Evaluate MPC performance with non-linear plant:	86
4.2.1	MPC Adaptivity Check	88
4.2.2	Optimum Cost and Optimization Status:	90
4.2.3	Design Review by MPC Review Functionality:	91
4.2.4	Closed-Loop Nominal Stability	92
4.2.5	Closed-Loop Steady-State Gains	93
4.2.6	QP Hessian Matrix Validity	93
4.2.7	Hard MV Constraints	93
4.2.8	Other Hard Constraints	93
4.2.9	Soft Constraints	94
4.2.10	Memory Size for MPC Data	94
4.3	Practical Recommendations	95
5	Future Work, Final Remarks and Conclusion	97
5.1	Future Work	97
5.1.1	Precise System Modelling	97
5.1.2	Simplified and Cost Effective Control	98
5.2	Conclusion	98
A	Modelling of Compression Behaviour of the fabric	101
A.1	Kinematic Modelling of Weaving and Consolidation Process	102
A.2	Mechanical Modelling and Consolidation Process	104
B	MATLAB codes and Simulink Representations	107
B.1	Stage I MPC object Matlab Code	107
B.2	Stage II MPC object Matlab Code	108
B.3	Simulink Representation of Stage II MPC	110
B.4	Simulink Representation of Stage II Adaptive MPC	110
	Bibliography	111

List of Tables

2.1	Power Variable Table	28
3.1	Subsystem Parameters and Corresponding Values	57
3.2	Subsystem Parameters and Corresponding Values	60
3.3	Input and Output Variable and values of Stage I prior to Steady State Operating Point Search	72
3.4	Input and Output variables and values of Stage II prior to Steady State Operating Point Search	73
3.5	Design values of Weights.MVrate for Stages I and II	77
3.6	Design values of Weights.OV for Stages I and II	77
3.7	Design Scale Factors for Stages I and II	79
3.8	Polynomial Coefficients of Numerators (B) and Denominators (A)	80
4.1	Model Predictive Controller Test parameters	83
4.2	Model Predictive Controller Design Variables	85
4.3	Stage I simulation result for controller tested with non-linear plant	91
4.4	Simulation result for controller tested with non-linear plant	91
4.5	Evaluation of Constraints ECR parameters	94
4.6	Memory Estimation for MPC Deployment	95

List of Figures

1.1	Textile Process Line	2
1.2	Current PI Controller Scheme	3
2.1	Knit Fabric Configuration	5
2.2	Woven Fabric Configuration	6
2.3	Compaction Based on Heated Roll and Shoe Principle	6
2.4	Alternative Compaction Based on Heated Roll and Shoe Principle	7
2.5	Compaction based on Rubber Belt	7
2.6	continuous time transfer function	9
2.7	Equation Error Scheme	13
2.8	Equation Error Scheme	14
2.9	Equation Error Scheme	15
2.10	discrete time transfer function	16
2.11	<i>PUI</i> around Polytope	20
2.12	<i>PUI</i> representation for <i>POP</i>	23
2.13	Convex Relaxation of <i>POP</i>	24
2.14	Block Diagram Notation	27
2.15	Causal Stroke Notation	27
2.16	Simple Mechanical System	27
2.17	Simple Mechanical System	28
2.18	Simple Mechanical System	28
2.19	1-Port Resistor with Bond-graph representation	29
2.20	1-Port Resistor Input-Output Plot	30
2.21	1-Port Compliance with Bond-graph Representation	30
2.22	Area interpretation of stored energy for a 1-port capacitor: (a) nonlinear case; (b) linear case.	31
2.23	Compliance Derivative Causality	31
2.24	Compliance Integrative Causality	32
2.25	1-Port Inertial with Bond-graph Representation	33
2.26	Area interpretation of stored energy for 1-port inertia. (a) Nonlinear case; (b) linear case	34
2.27	Inertial Derivative Causality	34
2.28	Inertial Integrative Causality	35
2.29	Integrative and Derivative Path Summary for Inertial and Compliant Components	35

2.30	1-Port Supply with Bond-graph Representation	36
2.31	Transformers: (a) bond graph; (b) ideal rigid lever; (c) gear pair	37
2.32	Common Effort Junction Representation	38
2.33	Common Effort Junction Representation for Mechanical System	38
2.34	Causal Rule and Block Representation for Common Effort Junction	39
2.35	Common Flow Junction Representation	39
2.36	Common Effort Junction Representation for Mechanical System	40
2.37	Causal Rule and Block Representation for Common Flow Junction	40
2.38	System's Schematic Representation	41
2.39	Basic Feedback Structure of MPC [13]	42
2.40	Schematic Representation of Control Horizon and Input Bounds	46
2.41	Future Setpoint Trajectory	47
2.42	Tuning of MPC	51
3.1	Stage I Compaction (by Compression)	55
3.2	Stage I Compaction Scheme	58
3.3	Stage I Compaction Bond-Graph Realization	58
3.4	Stage II Compaction Zone	60
3.5	MIMO Transfer Function Representation	61
3.6	MISO Transfer Function Representation	63
3.7	First Support Matrix	66
3.8	Second Support Matrix	66
3.9	Third Support Matrix	67
3.10	Fourth Support Matrix	67
3.11	Stage II Compaction Scheme, $\omega_c =$ command speed, $\omega_l =$ Line Speed	68
3.12	Stage II Compaction Bond-Graph Realization	68
3.13	Feedback Control System Layout	70
3.14	Representation of Stage I Compaction	71
3.15	Steady State Operating Point Search Result	72
3.16	Representation of Stage II Compaction	73
3.17	Steady State Operating Point Search result for Stage II	73
4.1	Testing of Stage I Model Predictive Controller With Linear Plant compaction with nominal values	84
4.2	Testing of Stage I Model Predictive Controller on Linear Plant with a step size of 0.1 on the reference	84
4.3	Testing of Stage II Model Predictive Controller With Linear Plant compaction with nominal values	85
4.4	Testing of Stage II Model Predictive Controller on linear Plant with steps on disturbance variables	86
4.5	Testing of Stage I Model Predictive Controller with Non-Linear Plant	86
4.6	Cost Function and Quadratic Programming for stage I.	87
4.7	Testing of Stage II Model Predictive Controller With Non-Linear Plant	87
4.8	Cost Function and Quadratic Programming for stage II.	88
4.9	Adaptivity check result of Stage I	88
4.10	Cost function and Quadratic Programming status of Stage I	89

4.11	Adaptivity check result of Stage II	89
4.12	Cost function and Quadratic Programming status of Stage II	90
4.13	Summary on Review of Model Predictive Control for Stages I and II Com- paction	92
4.14	Demonstration of Stage II Plant Stability	92
A.1	Single Layer Result Analysis	103
A.2	Multiple Layer Offset Compensation	104
A.3	(a) Experimental results of a single yarn compaction test (b) derived rela- tionship of the intra yarn volume fraction/compaction load	105
A.4	Comparison between predicted and experimental results for a single layer 2/2 twill weave compaction	106
A.5	Comparison of a single layer compaction with a 6 layer compaction with and with-out nesting	106
B.1	Simulink Representation of Stage II MPC	110
B.2	Simulink Representation of Stage II Adaptive MPC	110

Chapter 1

Thesis Opening

This first chapter is written to give a first insight into the thesis work. Motivations and reasons of the work are explained, together with the actual state of the art. The structure of the thesis is also reported.

1.1 Motivation

The paradigm shift in the production systems over 20 years today, from manual implementation to automation has become the competitive advantage for manufacturing and process industries. The dream of modern industries is to reach a point where every unit of production is automated - at that point, as Tom Preston-Werner puts it, when you are either the one that creates the automation or you're getting automated.

Automation of production systems apparently has become the hope for many companies to remain competitive under recent harsh economic atmosphere and based on statistics, a carefully planned investment in automation can make good financial sense. A two or three-year payback on reduced labour costs is the equivalent of an excellent 30% to 50% return on investment. There is the possibility of increasing productivity by 20% to 30% while handling most of the repetitive and complex tasks. A fast, stable and accurate production flow contributes to the reduction of production times and the cost of the manufactured products. The use of automated production lines significantly reduces production costs and labour costs, and minimizes human errors, ensuring output consistency, improves repeatability, safety and quality.

As an industrial practise, it is customary to automate sub-systems that contributes to an integrated automated production line. To obtain desirable automation within these subsystems, control systems are often required. However, based on the method of implementation of the control system, full or partial automation can be obtained. Partial controlled systems can be designed using PID controller. PID is attractive for its simplicity. However, it may require tuning at certain intervals should parameters of the plant which it controls change. This is often a challenge for the support team of OEM companies, since quality time is bound to be dedicated in clients' support.

To bridge this gap, there is need to design a control system that can adapt to these parameter changes while still maintaining or improving the attributes mentioned above. A

suitable method to achieve this is with the use of an Adaptive MPC with plant parameters obtain through on-line estimation and this shall be the major discussion of this thesis.

1.2 State of the Art

This section discusses the current method of implementing automation on Sanfore. The benefits and limitations of this implementation are also presented here with some references that provide more details.

Sanforization is a treatment process, mainly applied to cotton fabrics and textiles made from natural or chemical fibres, patented by an American, Sanford Lockwood Cluett (1874–1968) in 1930.[33].

1.2.1 Bianco Spa Textile Finishing Line

Bianco Textile finishing line consists of the Linea Rosa (Pink Line), Gruppo Centratore (Centering/fabric entry), Alzatrice (Widthwise Stenter), Compattatore Sanforizzo (Sanforizer Compactor), Compattatore a feltro (Felt Compactor/Calender) and Faldatore (Plaiter) as seen in the figure below.

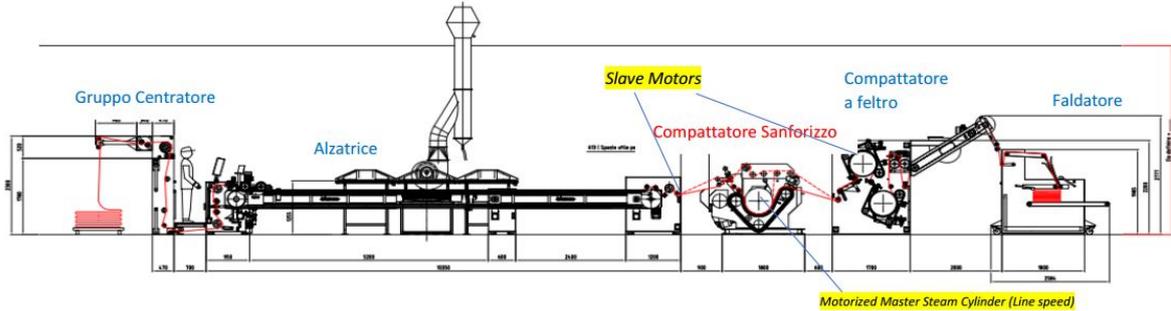


Figure 1.1. Textile Process Line

1.2.2 Proportional-Integrator (PI) Controller

A desired dimension stable fabric is achievable with Sanforized compaction system and this requires a minimum constant tension at the exit of this system unit. The tension is measured using a load cell which works with the principle of Strain gage. This tension is regulated using a Proportional - Integrative (PI) Controller to the desired reference tension. The output of the plant is the controlled speed of the slave motor at the exit. This measured tension, with some gains and offset compensations is the feedback of the PI (Proportional-Integrative) controller. The target is to obtain an error of zero (0) in

comparison with the desired (pre-set) tension. If there is a deviation, the PI controller is tuned based on experts' experience, in such a way that the difference is used to obtain a variable for adjusting the speed of the corresponding motor and to maintain a constant tension. The diagram of the PI controller is shown in figure below.

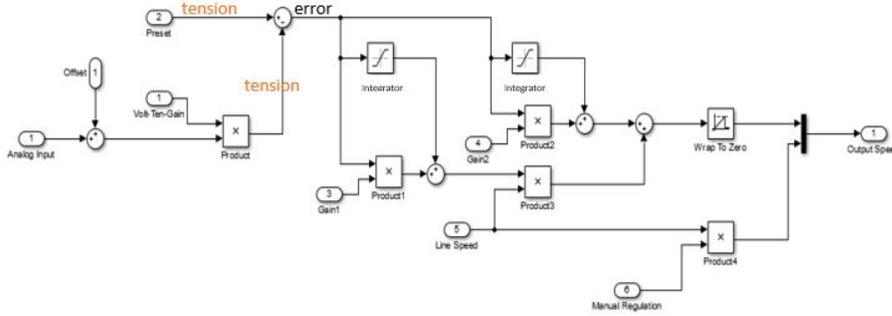


Figure 1.2. Current PI Controller Scheme

1.2.3 Some Variable Parameter Considerations

- The Load cell functions with the principle of Wheatstone bridge. The measured **tension** corresponds to the output voltage of the Load cell. The Load cell produces 0mV that corresponds to 0N load tension and 20mV for 200N tension. The Volt-to-Tension gain represents the scaling factor corresponding to tension obtained from Voltage. For a given class of fabric materials, a corresponding tension is necessary to obtain the required quality and this tension varies for different materials.
- The Offset variable makes up for Load Cell roller weight.
- The **Line Speed** in the speed of the master motor while the Output speed is the speed of the slave motor. Manual Regulation input corresponds to the percentage of slave speed required for manual regulation.
- The belt is subject to ageing and wear and tear on the external surface. As a result, after a while, the belt is ground the **belt thickness**. Since the belt is tensioned from the internal surface, the set tension of the belt is not affected. However, the compaction due to the rubber belt is affected. A sensor is used to detect the thickness of the belt.
- **Fabric Angle of Entry** For a given tension applied on a given fabric material, the quality of the output fabric material can be varied by changing the entry angle of the fabric.

The plant's complexity, coupled with the variation in these parameters raises the need for a robustly estimated it to correspondingly robustly control it.

1.3 Thesis outline

The structure of the thesis work is summarized below.

Chapter 1 : It is the current chapter which introduces the thesis, giving insight on the motivation, the need-gap and the current state of art.

Chapter 2 : A brief description of Sanforized compaction techniques with theoretical background of the system modelling and control strategies.

Chapter 3 : Describes the subsystem divisions, the various subsystem model strategies and the adaptive and non adaptive control design strategies adopted.

Chapter 4 : Dedicated to discussion of the findings based on the results obtained from chapter 3 as well as recommendations.

Chapter 5 : This chapters gives an insight on possibilities for further research work in this field while concluding the thesis.

Appendix A : This give alternative approach to predict by simulation the compaction of a fabric without the physical specimen.

Appendix B : It contains the MPC object MATLAB codes and the representation of the Simulink implementation.

Chapter 2

Description and Theory

This chapter describes the basic techniques and processes of compaction, the theories behind Set Membership Robust System Identification and Bond-graph modelling approaches, and the control of the Sanford compactor.

2.1 Compaction

2.1.1 Compaction Theory

The structure of a fabric contributes to its shrinkage tendency. More open structures have greater propensity to shrink. A mechanical process, known as *Compaction* can be used to reduced fabric length shrinkage by mechanically forcing the fabric structure to compress itself. Yarns are flexible, and during fabrication, are bent into non-linear configurations. In the case of Knit, fabrication and processing tension tends to elongate loops [4]. If the shape of the individual loops can be made more round, the fabric is made more compact and length can be reduced. This process limits washing (residual) shrinkage. Examples of Knit material include: felt, jersey, etc.

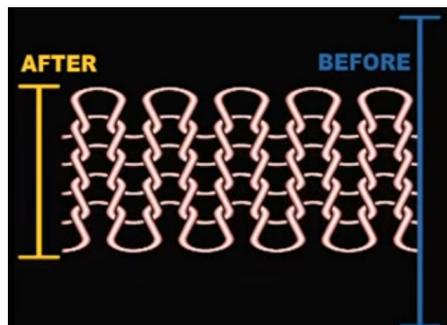


Figure 2.1. Knit Fabric Configuration

In the case of woven fabric, more cramp is created along the warp line and this reduces

its length. The more the cramp, the lower the residual shrinkage. Jean is an example of woven fabric.

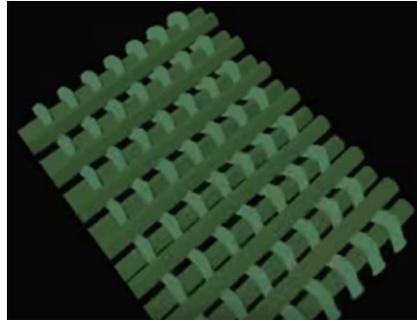


Figure 2.2. Woven Fabric Configuration

When knits and woven fabrics are compacted, the fabrics become heavier and thicker and yardage is reduced. With softeners and steam applied to the fabric in the process, the yarns can easily sleep by each other and readjust themselves.

2.1.2 Basis Techniques of Compaction

Based on Heated Roll and Shoe principle: Steam is applied as fabric raps around a heated feed roll made of steel. Just enough moisture is applied so that fabric is moves freely. The fabrics moves between two heated rolls which have special surfaces to grab the fabrics in a controlled way. The gap between the two shoes is zone for compaction. The Delivery Roll is made of rubber and turns at a slower surface speed than the Feed Roll and along with the shoes, slows the fabric down to allow for compaction. When the fabrics hits the surface of the heated shoes, the fabric slows, shortens and compacts based on the surface friction on the compaction zone. Over compaction will lead to corrugated appearance. The fabric that exits the machine is noted to be thicker than that is going in.

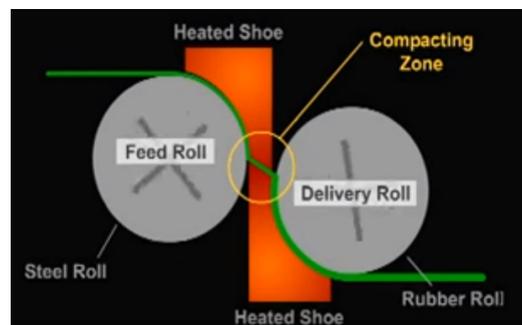


Figure 2.3. Compaction Based on Heated Roll and Shoe Principle

An equivalent design of this technique is seen where a single heated shoe is placed in-between the Feed Roll and Delivery Roll.

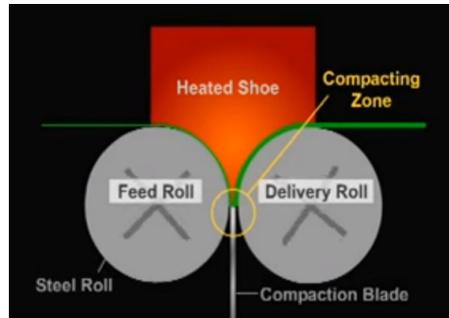


Figure 2.4. Alternative Compaction Based on Heated Roll and Shoe Principle

Based on Belt compaction:

It utilizes either a rubber or a felt belt that is stretched by flexing over a pressure roll. The fabric is laid on the belt while the belt is still stretched. Steam is applied as fabric enters the compressed path between the belt and the heated Palmer unit. Compressive shrinkage of the fabric is obtained with the recoil of the belt when it is flexed around the Palmer unit. This is the method implemented in Bianco Sanforizer Compactor design. In the Sanforizer, the fabric material is pre-moistened with steam at the Chain machine (Alzatrice) that precedes the compactor.

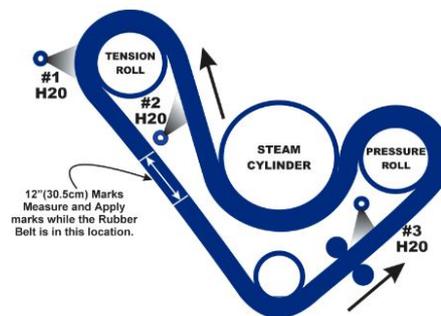


Figure 2.5. Compaction based on Rubber Belt

2.1.3 Compaction Process

The shrinkage of woven and knitted fabrics is partly attributable to tensions imposed during the manufacturing cycle. These tensions are released in finishing. The compressive shrinkage (compaction) process is intended for reducing this fabric residual shrinkage using mechanical treatment. This process improves dimensional stability and reduces the residual shrinkage by removing tensions in warp and compacting the

fabric structure. Before the compaction process, the fabric is moistened to lubricate the fibres and yarns to enhance compaction [3]. A steam heated cylinder is then used to achieve a uniform and stable compaction on the fabric. To determine fabric shrinkage level, Water test is carried out on the compacted fabric. At this point the shrinkage level is compared with the desired. Nevertheless, based on design, the fabric is given a slight tension to obtain a more precise compaction result. If a considerable difference is found, the compaction variables are changed to improve the compaction.

The Sanfore compactor is designed to optimize the lengthwise compaction ratio and this vary based of the following parameters which include;

- compression due to rubber belt
- fabric tension
- motor speed
- angle of entry

Fabrics have varying potential for compaction. For fabrics with high potential for compaction, one of the acceptable strategies applied in Bianco is to first slightly over-compact with the compression on the rubber belt. Afterwards, the over-compacted fabric is slightly tensioned to achieve the desired compaction. The procedure is very desirable for fast and stable response since the response by tension is almost immediate while there is a considerable amount of delay on the compaction due to the rubber belt. For fabrics with less potential for compaction due to structure and preparation, it's often a practice to reduce the line speed of the system until the desire compaction is reached. The speed reduction enables the compaction zone sufficient time for more compaction. The rate of compaction is dependent on three major factors; the temperature of the heated roller (less than $120^{\circ}C$ for most generic fabrics and less than $140^{\circ}C$ for heavy Denim fabric), maximum level of compression on the rubber belt, the amount of moisture on fabric. Using a smooth and soft rubber belt surface gives the best shrinkage result with no defects on fabric. However, the normal use of the belt causes a slow wear on the surface of the belt. The belt surface is often ground after every 500000 – 600000 m of fabrics treated.

2.2 System Modelling

The model of a plant can be derived from mathematical and physical laws. Models derived with this method are often simplified and hence are only approximations of the true system. This approximation results in the uncertainty of the plant model. Some parameter are not exactly known a-priori, and this results in *parametric uncertainty* of the plant. Other parameters are not modelled at all, and this results in *dynamic uncertainty*.

The basic approach is to describe the plant under study as a member of a set of systems [18]. This is set then known as *model set*. Considering an (Linear Time Invariant) LTI system, a parametric uncertain plant is modelled with a finite number of parameters and this defined by a *structured model set*. On the other hand, a total ignorance of the order

and phase is assumed for a dynamic uncertain plant and this modelled as *unstructured model set*.

The method applied to derive the plant model is very important. Generally speaking, models can be obtained by First principle model or by experimental method.

- First Principle Method plus A set of A-priori Information on the Physical Parameters: This is known as the *White-Box Modelling*. The mapping of the inputs and the outputs is obtained through computation application of physical and mathematical laws.
- Based on Experimental Data plus A set of A-priori Information on the Model Structure: This method is essentially known as system Identification. It's devoid of errors associated with physical and mathematical laws. However, since input and output data is required, the result of this method is dependent on how accurately these data are collected and the consideration of possible data noise. The structure (eg: linear, non-linear, time variant etc) of the system is conveniently selected in such a way that optimization can be simplified.

When significant structural information is available based on some physical insight, the model is known as *Gray-Box Modelling*. However, if only mild assumptions are made on the system class, the system is known as *Black-Box Modelling*.

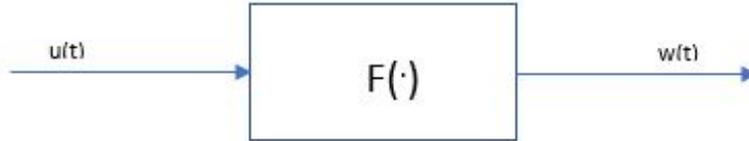


Figure 2.6. continuous time transfer function

$$w(t) = f[w(t - 1), w(t - 2), \dots, w(t - m), u(t), u(t - 1), \dots, u(t - m)] \quad (2.1)$$

$$t = 1, 2, 3, 4, \dots, 5$$

The above equation considers the past and present input and output variables as the state variables of a dynamic system.

2.3 Robust System Identification

Certain sources of data used for system Identification (such as the measuring instruments) are subject to uncertainty. This section emphasize on deriving the mathematical model of

a system considering the influence of these uncertainties.

$$G_p = \frac{\alpha_1 s + \alpha_0}{\beta_2 s^2 + \beta_1 s + \beta_0} \quad (2.2)$$

where $\alpha_i, \beta_j \in [-\infty + \infty]$ are the parameters of the plant G_p computed considering uncertainties. The data used for this computation is affected by noise which is represented by the uncertainty range. Also, sometimes, due to approximation of physical systems, some flexural poles and zeros might be neglected. These poles and zeros are accommodated within the uncertainty range [19].

2.3.1 Important Definitions:

The a-priori information for system identification is related to functional knowledge of the system.

$$f \in F \quad (2.3)$$

- **Parametric and Non-Parametric Functions:** From above, f is a:
Parametric Function if the class F can be described by finite number of Parameters, or
Non-parametric Function if the class F cannot be describe by finite number of parameters.
 Considering a second order system in equation 2.2, the current output is defined by the finite number of parameters, $\alpha_1, \alpha_0, \beta_2, \beta_1, \beta_0$.
- **Time-Variant and Time-Invariant Models:** A model is time-invariant if the function f does not depend on time. Otherwise, it is time-variant.
 A time in-variant model has the parameters α_i, β_i as constant variable while in other case, the the parameters are non-constant variables.
- **Linear and Non-linear Functions:** A function is said to be linear if;
 $f \in F = \{\text{linear combination of } w(t-1) w(t-2) \dots w(t-m) u(t) u(t-1) \dots u(t-m)\}$
 parametrized by
 $\theta = [\alpha_0, \alpha_1, \dots, \beta_1, \beta_0]$

Hence, System Identification problem is the problem of estimating the values of θ

2.4 Parametric Identification of LTI System

Given a set of input and output data,

$$\{w(t), u(t)\} \quad t = 1, \dots, N$$

$$w(t) = w(t-1), w(t-2), \dots, w(t-m), u(t), u(t-1), \dots, u(t-m)$$

For a second order System,

$$w(3) = -\alpha_1 w(t-1) - \alpha_2 + \beta_0 u(t) + \beta_1 u(t-1) + \beta_2 u(t-2) \quad (2.4)$$

$$\begin{bmatrix} w(3) \\ w(4) \\ w(5) \\ w(6) \\ w(7) \end{bmatrix} = \begin{bmatrix} -w(2) & -w(1) & u(3) & u(2) & u(1) \\ -w(3) & -w(2) & u(4) & u(3) & u(2) \\ -w(4) & -w(3) & u(5) & u(4) & u(3) \\ -w(5) & -w(4) & u(6) & u(5) & u(4) \\ -w(6) & -w(5) & u(7) & u(6) & u(5) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} \quad (2.5)$$

Which can be represented as;

$$b = A.\theta \quad (2.6)$$

Since A is a square matrix from equation 2.5, A is invertible, hence;

$$\theta = A^{-1}.b$$

The variable b corresponds with the number of measurements, N taken. Often, with a very few number of measurements, a problem of *over-fitting* is often encountered. This is overcome by taking infinite number of measurements. This however, leads to a non-invertible matrix A and hence to an inconsistent system. To solve this, a pseudo-inverse of A is derived. Thus,

$$b = A.\theta$$

$$\begin{aligned} b &\in R^{N+1} \\ \theta &\in R^{(2m+1) \times 1} \\ A &\in R^{N \times (2m+1)} \\ N &\gg 2m+1 \end{aligned}$$

$$\begin{aligned} (A^T A)^{-1} A^T b &= (A^T A)^{-1} (A^T A) \theta \\ A^T b &= A^T A \theta \end{aligned}$$

$$\theta = (A^T A)^{-1} A^T b \quad (2.7)$$

where $(A^T A)^{-1} A^T$ is the pseudo-inverse of A .

Pseudo-inverse of A apparently converts the *inconsistent* system to a *consistent* one.

Note:

It is not a good idea to take measurements at steady state since this often results to a matrix whose columns are linearly dependent and hence to under-determined system with no unique solution.

2.5 System Identification with Error Added:

Introducing error into equation 2.6, the basic equation becomes;

$$b = A\theta + e \quad (2.8)$$

and hence the *error equation* becomes;

$$e = b - A\theta \quad (2.9)$$

This introduces us the ***system identification problem statement*** which goes as follows;

find the value of θ such that error is minimized

$$\theta = \operatorname{argmin} \|e\| = \operatorname{argmin} \|b - A\theta\| \quad (2.10)$$

θ affected by error is hence;

$$\theta_N = (A^T A)^{-1} A^T b - (A^T A)^{-1} A^T e \quad (2.11)$$

If $e(t)$ is Gaussian (ie, normally distributed), and with zero (0) mean,

$$\lim_{t \rightarrow \infty} \theta_N = \theta_{true}$$

Error introduced in the system is considered in terms of structure and the type of estimator in used.

2.6 Error Structure:

The error structure defines the way in which error enters a given system of equation. Three main error structures shall be discussed here and they include:

- Equation Error Structure
- Error in Variable Structure
- Output Error

2.6.1 Equation Error Structure:

Given,

$$G_d(q^{-1}) = \frac{N(q^{-1})}{D(q^{-1})} \quad (2.12)$$

Where q^{-1} is the backwards shift operator ($q = z$)

$$w(t) = \frac{N(q^{-1})}{D(q^{-1})} u(t) \tag{2.13}$$

where $u(t)$ and $w(t)$ are true input and output of the system.

If there is no noise (uncertainty),

$$y \cong w$$

For a second order system,

$$w(t) = \begin{bmatrix} -w(t-1) & -w(t-2) & -u(t) & -u(t-1) - u(t-2) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix}$$

for $t = 3, \dots, N$

$$b = A.\theta$$

A and b are absolutely known.

Error is said to enter a problem in an equation error structure form if the system equation structure appears as follows:

$$D(q^{-1}).y(t) = D(q^{-1}) + e(t) \tag{2.14}$$

$$w(t) = \frac{N(q^{-1})}{D(q^{-1})}.u(t) + \frac{1}{D(q^{-1})}e(t) \tag{2.15}$$

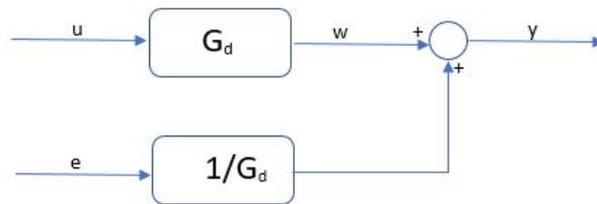


Figure 2.7. Equation Error Scheme

This demonstrates that error enters the output of the system filtered by the denominator of the transfer function of the system to be Identified. This infers that the device that is used for input and output data collection, and which is affected by noise is itself related to the system which is to be identified. This presents the limitation for error entering the output in this structural manner.

With error introduced, we have

$$b = A.\theta + e \quad (2.16)$$

Although the error appear to be dissociated with the system to be identified, looking closely, it shall be discovered that the previous values of y which are components of A are affected by noise. This means that A which should be known is affected by noise. For more information on equation error, refer to [25]

2.6.2 Error-In-Variables:

For equation 2.13

$$w(t) = \frac{N_d(q^{-1})}{D_d(q^{-1})} u(t)$$

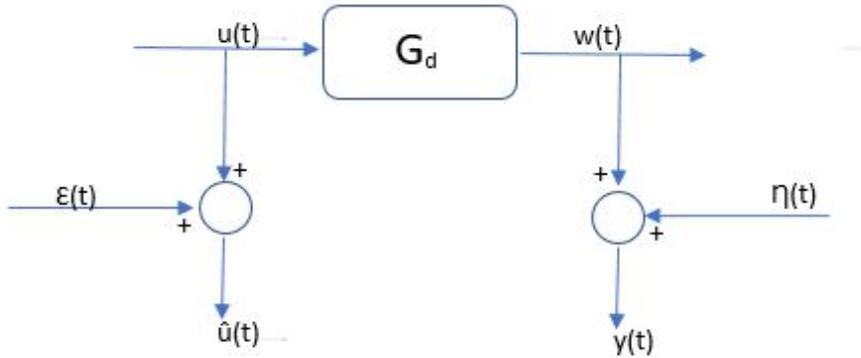


Figure 2.8. Equation Error Scheme

$$w(t) = y(t) - \eta(t) \quad (2.17)$$

$$u(t) = \hat{u}(t) - \epsilon(t) \quad (2.18)$$

$$y(t) - \eta(t) = \frac{N_d(q^{-1})}{D_d(q^{-1})} (\hat{u}(t) - \epsilon(t)) \quad (2.19)$$

For a second order system,

$$m = 2$$

$$y(t) - \eta(t) = -\alpha_1 y(t-1) + \alpha_1 \eta(t-1) - \alpha_2 y(t-2) + \alpha_2 \eta(t-2) + \beta_0 \hat{u}(t) - \beta_0 \epsilon(t) + \beta_1 \hat{u}(t-1) - \beta_1 \epsilon(t-1) + \beta_2 \hat{u}(t-2) - \beta_2 \epsilon(t-2).$$

This demonstrates that the parameter to be estimated is affected by error non-linearly.

The *EIV* is often useful when both the inputs and the outputs of the system must be measured. The application is seen when the input of the system to be identified is an output of a super-system. [19] [20] [21] [23]

Sometimes, the input of the system is known exactly. In such case, the Error-In-Variable Structure results in *Output Error Structure*.

2.6.3 Output Error Structure:

This is a particular case of *EIV* with the input known exactly and hence unaffected by measurement noise [24]. The scheme is as shown below:

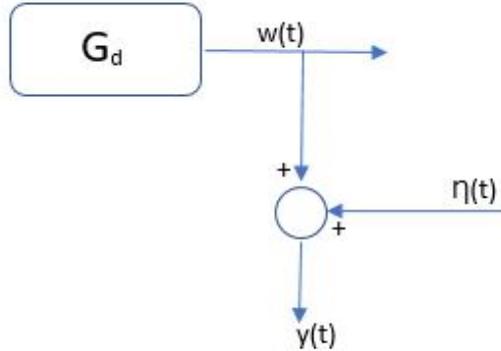


Figure 2.9. Equation Error Scheme

To solve, *EIV*, a-priori information about the ϵ and η is required. Such a-priori information on ϵ and η include:

- **Structure:** The structures of ϵ and η may be either,
 - Equation Error (EE)
 - Error-In-Variable (EIV) or
 - Output Error (OE)

- **Other Information:**

- ϵ and η are assumed to be random variables statistically distributed according to probability density function (PDF) that is at least partially known. The assumption is only possible if variables are random and usually, large amount of data required to obtain statistical consistency.
- ϵ and η are bounded. That is, they are only known to belong to a given bounded set. With this approach, small amount of data can be useful.

$$\begin{aligned}
 |\eta| &\leq \Delta\eta(t) \quad \forall t \\
 |\epsilon| &\leq \Delta\epsilon(t) \quad \forall t \\
 &\text{where} \\
 \Delta\eta(t) &\in R \text{ is known, and} \\
 \Delta\epsilon(t) &\in R \text{ is known}
 \end{aligned}$$

2.7 Parameter Estimators

In a broad sense, two kinds of estimators exist;

- **Point-wise Estimator (PWE):**

PWE functions with an algorithm that provides solution to estimation by finding single point values of the system parameters.

Given a transfer function,

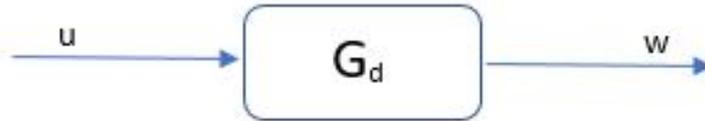


Figure 2.10. discrete time transfer function

From equation 2.13

$$w(t) = \frac{N(q^{-1})}{D(q^{-1})} u(t)$$

where $u(t)$ and $w(t)$ are true input and output of the system.

Now, let $y(t)$ be the collected measurement of $w(t)$ that is affected by noise. If the error or uncertainty is assumed to enter the problem in an equation error structured form as in the equation 2.6.1,

$$D(q^{-1}).y(t) = D(q^{-1}).w(t) + e(t)$$

Estimators of this kind include:

- **Least Square (e_2 norm) Estimator:**
- **L_∞ -norm Estimator:**

2.7.1 Least Square (e_2 norm) Estimator:

The least square estimators aims at reducing the energy of the error. Refer to [25] [26] for more details

$$\theta_{ls} = \theta_N = \underset{\theta}{\operatorname{argmin}} \|y - \phi^T \theta\|_2 \quad (2.20)$$

θ_{ls} is the value that minimizes the 2-norm of the error.
 ϕ^T is known as the matrix of regressors.

$$\theta_{ls} = \underset{\theta}{\operatorname{argmin}} \|D(q^{-1}).y - N(q^{-1}).u\|_2 \quad (2.21)$$

$$h = D(q^{-1}).y - N(q^{-1}).u$$

$$h = [h(1), h(2) h(3), \dots, h(N)]$$

$$\theta_{ls} = \underset{\theta}{\operatorname{argmin}} \|h\|_2$$

$$D(q^{-1}) = 1 + \alpha_1 q^{-1} + \alpha_2 q^{-2} + \dots + \alpha_m q^{-m}$$

$$N(q^{-1}) = \beta_0 + \beta_1 q^{-1} + \dots + \beta_m q^{-m}$$

$$h(t) = y(t) + \alpha_1 y(t-1) + \dots + \alpha_m y(t-m) - \beta_0 u(t-1) - \beta_1 u(t-2) - \dots - \beta_m u(t-m)$$

$$t = 1, \dots, N$$

$$h = y(t) - \phi^T(t)\theta$$

where

$$\phi^T(t) = \begin{bmatrix} y(t) & y(t-1) & \dots & y(t-m) & -u(t-1) & -u(t-2) & \dots & -u(t-m) \end{bmatrix}$$

$$\theta = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \dots \\ \alpha_m \\ \beta_0 \\ \beta_1 \\ \beta_2 \\ \dots \\ \beta_m \end{bmatrix} \quad b = \begin{bmatrix} y(m+1) \\ \cdot \\ \cdot \\ \cdot \\ y(N) \end{bmatrix} \quad A = \begin{bmatrix} \phi^T(m+1) \\ \cdot \\ \cdot \\ \cdot \\ \phi^T(N) \end{bmatrix}$$

2.7.2 L_∞ -norm Estimator:

The L_∞ -norm Estimator aims at reducing the value of the maximum error. Refer to [27]

$$\theta_{l_\infty} = \underset{\theta}{\operatorname{argmin}} \|D(q^{-1}) \cdot y - N(q^{-1}) \cdot u\|_\infty \quad (2.22)$$

$$\theta_{l_\infty} = \underset{\theta}{\operatorname{argmin}} \|b - A\theta\|_\infty \quad (2.23)$$

$$h = b - A\theta$$

$$\theta_{l_\infty} = \underset{\theta}{\operatorname{argmin}} (|h(1)|, |h(2)|, |h(3)|, \dots, |h(N)|) \quad (2.24)$$

The optimization problem here is associated with minimizing the maximum value of $h(i)$. This is achieved by inducing a slack variable, γ that represents the norm of the maximum value of $h(i)$.

$$\theta_{l_\infty} = \underset{\theta, \gamma}{\operatorname{argmin}} \gamma \quad (2.25)$$

$$\begin{aligned} h(1) &= |\gamma| \rightarrow -\gamma \leq h(1) \leq \gamma \\ h(2) &= |\gamma| \rightarrow -\gamma \leq h(2) \leq \gamma \\ &\dots \\ &\dots \\ &\dots \\ h(N) &= |\gamma| \rightarrow -\gamma \leq h(N) \leq \gamma \end{aligned}$$

We are minimizing a linear function of γ and θ subject to a number of constraints on γ and θ . The optimization problem to be solved is a *Linear Programming Problem (LPP)* and hence a *Convex Problem* with *unique global minimum* value of the error obtained.

A linear programming tool of matlab, *linprog*, can be used for this solution.

$$x = \text{linprog}(f, \hat{A}, \hat{b}) \quad (2.26)$$

that solves the equation,

$$x = \underset{x}{\text{argmin}} f^T .x \quad (2.27)$$

subject to,

$$\hat{A}.x \leq \text{hatb} \quad (2.28)$$

where f^T is a row vector of zeros (of dimension $2m + 1$) and a 1.

$$f^T = [0 \ 0 \ 0 \ \dots \ 0 \ 1]$$

x is a column vector of dimension $2(m + 1)$

$$x = \begin{bmatrix} \theta \\ \gamma \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ \dots \\ \alpha_m \\ \beta_0 \\ \dots \\ \beta_m \\ \gamma \end{bmatrix}$$

- **Set-Wise Estimator (SWE)** This approach of system Identification in presence of error assumes that the error is **Unknown But Bounded**. It shall be covered in the next section.

2.8 Set-Membership (Unkonwn But Bounded error) Identification of LTI System

2.8.1 Set Membership Identification of Discrete Time LTI Model from Input Out Data Corrupted by Equation Error

- A-priori assumption on the plant

$$f \in F$$

- A-priori assumption on noise due to measurement

– *Structure*: Considering Equation Error (EE) Structure,

$$y(t) = -\alpha_1 y(t-1) - \dots - \alpha_m y(t-m) + \beta_0 u(t-1) + \beta_1 u(t-2) + \dots + \beta_m u(t-m) + \epsilon(t)$$

- *Other Information*: Bounded Error

$$(\epsilon(t), \eta(t)) \in S$$

$$S = \{\epsilon(t), \eta(t) : h_i(\epsilon(t), \eta(t)) \geq 0, i = 1, \dots, r, t = 1, \dots, N\}$$

Error Set S : $S = \{e(t), t = 1, \dots, N, |e(t)| < \Delta e\}$

Feasible Parameter Set (FPS), D_θ : This is a set of all the values of θ which are consistent with the a-priori information on the plant and the noise for all the collected data.

$$D_\theta = \{\theta \in R^{2m+1} : y(t) = -\theta_1 y(t-1) \dots - \theta_m y(t-m) + \theta_{m+1} u(t) + \dots + \theta_{2m+1} u(t-m) + e(t),$$

$$|e(t)| \leq \Delta e(t)$$

$$\forall t = m+1, \dots, N\}$$

(2.29)

It can be shown that since the FPS consists of the equality linear dynamic system and inequality error equation, the optimization problem is a *convex solution of a polytope* and hence a Linear Programming Problem.

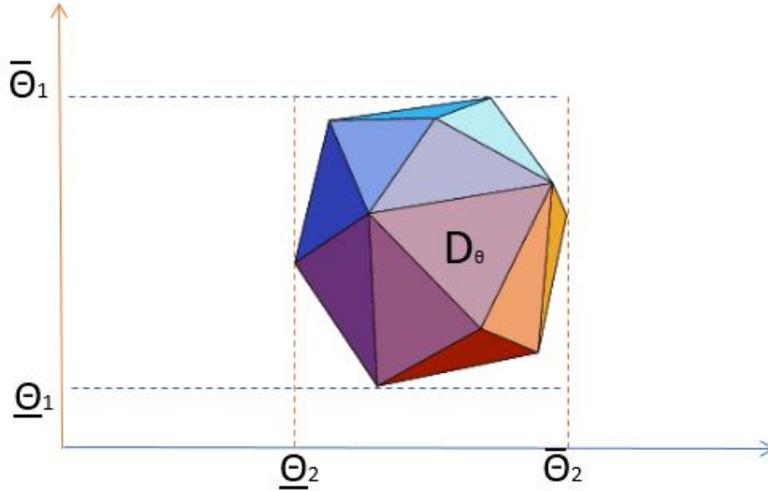


Figure 2.11. PUI around Polytope

Parameter Uncertainty Interval (PUI):

This solution which corresponds to the global minimum and maximum intervals is computed for each of the parameters.[18] [25]

Computing *PUI* introduces conservativeness as seen from figure 2.13

$$\begin{aligned}
 PUI_1 &= [\theta_1, \bar{\theta}_1] \\
 PUI_2 &= [\theta_2, \bar{\theta}_2] \\
 &\dots \\
 &\dots \\
 &\dots \\
 PUI_{2m+1} &= [\theta_{-2m+1}, \bar{\theta}_{2m+1}]
 \end{aligned}$$

The linear programming problem is solved using *linprog* of matlab as follows:

$$[x, f_{val}] = \text{linprog}(f, A, b) \tag{2.30}$$

Where x, known as minimizer, is the vector of the parameters to be estimated

$$\begin{aligned}
 x &= \underset{x}{\text{argmin}} f^T \cdot x \\
 &\text{subject to} \\
 &A \cdot x \leq b
 \end{aligned}$$

f_{val} is the minimum bound of the Parameter Uncertainty Interval (PUI),

$$\theta_i = \min f^T x \tag{2.31}$$

The maximum bound of the Parameter Uncertainty Interval (PUI) is given as;

$$\bar{\theta}_i = \max \theta_i = \min(-\theta_i) \tag{2.32}$$

f^T is as indicated in equation 2.26

Nominal Parameter

The nominal parameters of a plant is the defined as the center of the set known as *Chebyshev* center of *FBS* in L_∞ form.

$$\theta_c = \underset{\theta' \in R^{2m+1}}{\text{arg}} \min \max \|\theta' - \theta\|_\infty \tag{2.33}$$

$$\theta_c = [\theta_1^c, \theta_2^c, \dots, \theta_{2m+1}^c] \tag{2.34}$$

$$\theta_i^c = \frac{\theta_i + \bar{\theta}_i}{2} \quad (2.35)$$

$$\theta_i = \min_{\theta \in D_\theta} \theta_i ; \bar{\theta}_i = \max_{\theta \in D_\theta} \theta_i$$

2.8.2 Set Membership Identification of Discrete Time LTI Model from Input Out Data Corrupted by Output Error

Here, the a-priori assumptions on plant and noise are the similar to that specified for Equation Error with the exception that the noise is of Output Error structure. [24]

$$w(t) = -\alpha_1 w(t-1) - \dots - \alpha_m w(t-m) + \beta_0 u(t-1) + \beta_1 u(t-2) + \dots + \beta_m u(t-m) + \epsilon(t)$$

Feasible Parameter Set (FPS):

$$\begin{aligned} D_\theta = \{ \theta \in R^{2m+1} : & y(t) - \eta(t) = -\theta_1(y(t-1) - \eta(t-1)) \dots \\ & -\theta_m(y(t-m) - \eta(t-m)) + \theta_{m+1}u(t) + \dots + \theta_{2m+1}u(t-m), \\ & y(t) = w(t) + \eta(t) \\ & |\eta(t)| \leq \Delta\eta(t) \\ & \forall t = m+1, \dots, N \} \end{aligned} \quad (2.36)$$

Extended Feasible Parameter Set (EFPS):

$$\begin{aligned} D_{\theta,\eta} = \{ \theta \in R^{2m+1}, \eta \in R^N : & y(t) - \eta(t) = -\theta_1(y(t-1) - \eta(t-1)) \dots \\ & -\theta_m(y(t-m) - \eta(t-m)) + \theta_{m+1}u(t) + \dots + \theta_{2m+1}u(t-m), \\ & y(t) = w(t) + \eta(t) \\ & |\eta(t)| \leq \Delta\eta(t) \\ & \forall t = m+1, \dots, N \} \end{aligned} \quad (2.37)$$

The *EFPS* is described by bi-linear equality plant and inequality error. This results in a Polynomial Optimization Problem which is non-linear and non-convex.

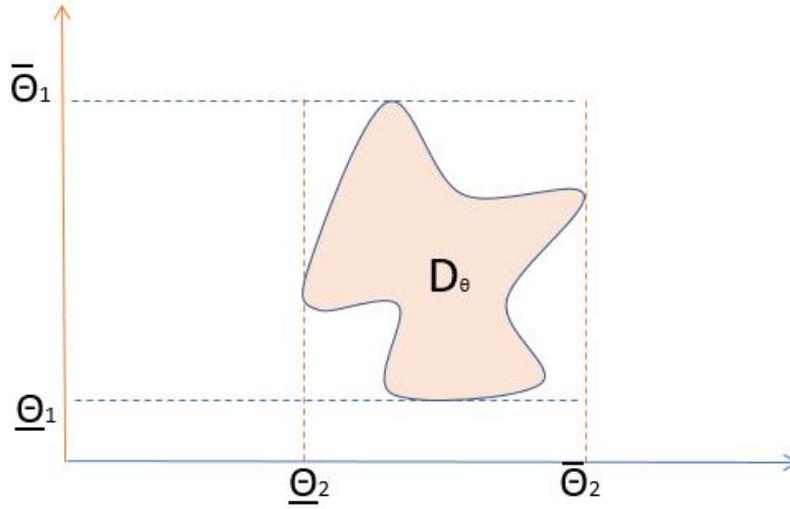


Figure 2.12. *PUI* representation for *POP*

Hence, the optimization problem for the Parameter Uncertainty Interval of

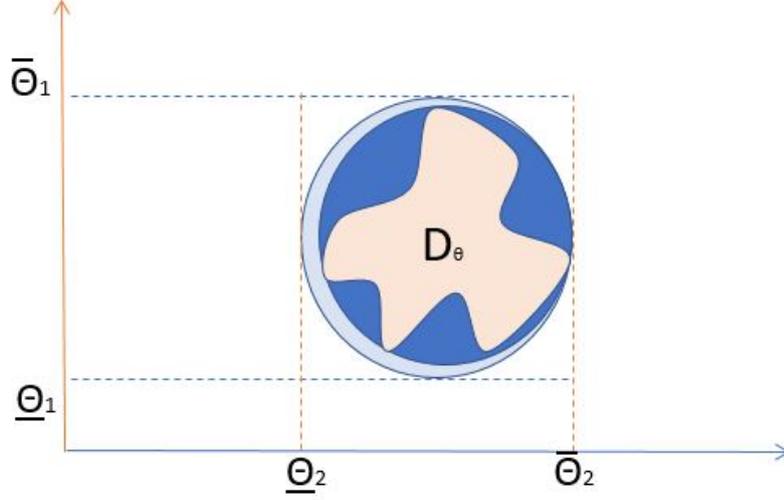
$$PUI_{\theta_i} = [\underline{\theta}_i, \bar{\theta}_i]$$

is

$$\underline{\theta}_i = \min_{\theta, \eta \in D_{\theta, \eta}} \theta_i \quad ; \quad \bar{\theta}_i = \max_{\theta, \eta \in D_{\theta, \eta}} \theta_i$$

Convex Relaxation:

This is the process of replacing non-convex Polynomial Optimization Problem sets with with convex ones which are solved to obtain global minimum and maximum. It involves the approximation of the *POP* from the outside. This approach, however introduces some degrees of conservativeness. [23]


 Figure 2.13. Convex Relaxation of *POP*

The measure of convex relaxation of the convex set is given by the ***order of relaxation***.

It is desirable to shrink the convex set to the barest minimum which is known as ***Convex Hull for D_θ*** . This results in a minimum conservativeness.

As $\delta \rightarrow \infty$, *PUI* that results in a convex hull is obtained. However, increasing δ increases computation time and memory storage demand.

2.8.3 Set Membership Identification of Discrete Time LTI Model from Input Out Data Corrupted by Error-In-Variable:

Here, there are both input and output noise, though unknown, they are bounded. [20] [21]

$$w(t) = -\alpha_1 w(t-1) - \dots - \alpha_m w(t-m) + \beta_0 u(t-1) + \beta_1 u(t-2) + \dots + \beta_m u(t-m) + \epsilon(t)$$

Feasible Parameter Set (FPS):

$$\begin{aligned} D_\theta = \{ \theta \in R^{2m+1} : & y(t) - \eta(t) = -\theta_1(y(t-1) - \eta(t-1)) \dots \\ & -\theta_m(y(t-m) - \eta(t-m)) + \theta_{m+1}u(t) + \dots + \theta_{2m+1}(u(t-m) - \epsilon(t-m)), \\ & y(t) = w(t) + \eta(t) \\ & |\eta(t)| \leq \Delta\eta(t) \\ & |\epsilon(t)| \leq \Delta\epsilon(t) \\ & \forall t = m+1, \dots, N \} \end{aligned} \quad (2.38)$$

Extended Feasible Parameter Set (EFPS):

$$\begin{aligned}
 D_{\theta,\eta,\epsilon} = \{ & \theta \in R^{2m+1}, \eta \in R^N, \epsilon \in R^N : y(t) - \eta(t) = -\theta_1(y(t-1) - \eta(t-1)) \dots \\
 & -\theta_m(y(t-m) - \eta(t-m)) + \theta_{m+1}u(t) + \dots + \beta_{2m+1}(u(t-m) - \epsilon(t-m)), \\
 & y(t) = w(t) + \eta(t) \\
 & |\eta(t)| \leq \Delta\eta(t) \\
 & |\epsilon(t)| \leq \Delta\epsilon(t) \\
 & \forall t = m+1, \dots, N \}
 \end{aligned} \tag{2.39}$$

This also results in a Polynomial Optimization Problem, and the Parameter Uncertain Interval is computed thus;

$$PUI_{\theta_i} = [\underline{\theta}_i, \bar{\theta}_i]$$

$$\underline{\theta}_i = \min_{\theta,\eta,\epsilon \in D_{\theta,\eta,\epsilon}} \theta_i \quad ; \quad \bar{\theta}_i = \max_{\theta,\eta \in D_{\theta,\eta,\epsilon}} \theta_i$$

2.8.4 Set Membership Identification of Stable Discrete Time LTI System:

If BIBO stability is an a-priori information about the plant, then, stability criteria are imposed on the Identification process to actualize this. [28] [29]

$G(q^{-1})$ is BIBO stable if and only if poles of $G(q^{-1})$, that is, the roots of the polynomials, $A(q^{-1})$ are all inside the unit circle in the complex plane. The Jury criteria are used to check if the requirements are fulfilled.

Jury Criteria states that the roots of polynomial $A(q^{-1})$ belongs to the open unit circle (strictly inside) if and only if the following conditions are met;

- $A(1) > 0$
- $(-1)^{n_a} > 0$
- $|a_{n_a}| < 1$
- Entries of the Jury table satisfies the following conditions

$$\begin{array}{cccccc}
 a_{n_a} & a_{n_a-1} & a_{n_a-2} & \dots & a_1 & 1 \\
 1 & a_1 & a_2 & \dots & a_{n_a-1} & a_{n_a} \\
 c_{n_a-1} & c_{n_a-2} & \dots & c_1 & c_0 & \\
 c_0 & c_1 & \dots & c_{n_a-2} & c_{n_a-1} & \\
 d_{n_a-2} & d_{n_a-1} & \dots & d_0 & & \\
 d_0 & d_1 & \dots & d_{n_a-2} & & \\
 \cdot & \cdot & & & & \\
 \cdot & \cdot & & & & \\
 \cdot & \cdot & & & & \\
 q_2 & q_1 & q_0 & & &
 \end{array}$$

$$|c_{n_a-1}| < |c_0|, |d_{n_a-2}| < |d_0|, \dots, |q_2| < |q_0|$$

$$c_{n_a-jc} = \begin{bmatrix} a_{n_a} & a_{n_a-jc} \\ 1 & a_{jc} \end{bmatrix}$$

$$d_{n_a-jd} = \begin{bmatrix} c_{n_a-1} & c_{n_a-jd} \\ c_0 & c_{jd} \end{bmatrix}$$

where n_a is the System's order
 $q_0 = 1$

2.9 System Modelling By Bond-Graph Approach:

2.9.1 Introduction

Bond graphs are a domain-independent graphical description of dynamic behaviour of physical systems. This means that systems from different domains (cf. electrical, mechanical, hydraulic, acoustical, thermodynamic, material) are described in the same way. The basis is that bond graphs are based on energy and energy exchange. Analogies between domains are more than just equations being analogous: the used physical concepts are analogous. Using bond graphs and the classification of power and energy variables, only a few basic types of multiport elements are required in order to represent models in a wide variety of energy domains [30]. In order to maintain relevance, we shall be restricted to only mechanical domain. ¹

2.9.2 Input - Output Relation:

It can be shown that if **Effort**, \mathbf{e} is input to a subsystem, then, **Flow**, \mathbf{f} necessarily has to be the output.

$$\mathbf{e} = R\mathbf{f} \tag{2.40}$$

¹Some of the formulas and diagrams in this section are with reference to [30]

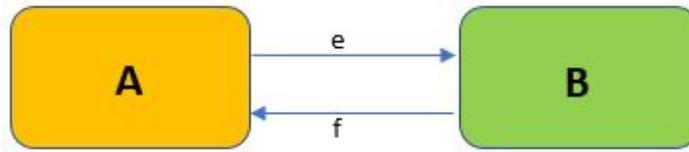


Figure 2.14. Block Diagram Notation

Causal Stroke: A causal stroke is equivalent to a subsystem receiving receiving **Effort, e** as an input.

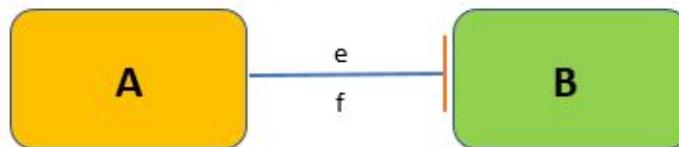


Figure 2.15. Causal Stroke Notation

Considering a mechanical subsystem of mass, m represented below,



Figure 2.16. Simple Mechanical System

$$F = m\dot{v} \tag{2.41}$$

$$v = \int F/m dt = 1/m \int F dt \tag{2.42}$$

Considering effort, F as the input, we have the following representation,

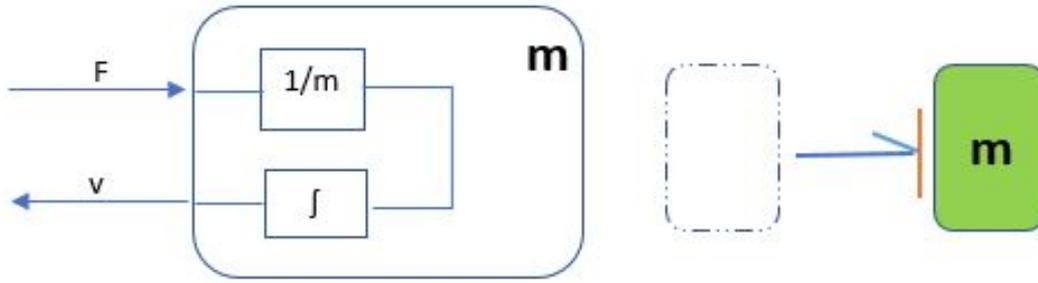


Figure 2.17. Simple Mechanical System

Considering flow, v as input, we have the following representation,

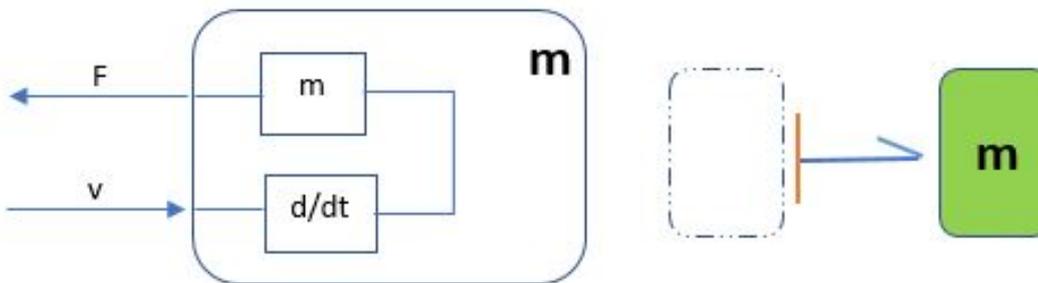


Figure 2.18. Simple Mechanical System

2.9.3 Power Relation:

Power indicates the common relationship that exists between interacting sub-systems belonging to different domains. Table 2.1 shows different power variables for mechanical domain with their corresponding units.

Domain	Power	Unit
General	ef	
Mechanical	Fv	$F [N], v[m/s]$
Mechanical	$T\omega$	$T[Nm], \omega [rad/s]$

Table 2.1. Power Variable Table

Other variable obtained from Power Variable:

Linear Momentum, P :

$$F = \frac{dP}{dt} \tag{2.43}$$

Angular Momentum P_θ

$$F = \frac{dP_\theta}{dt} \tag{2.44}$$

Hence, generally, effort, e is

$$e = \frac{dP}{dt} \tag{2.45}$$

$p = [P, P_\theta]$ While flow, f is;

$$f = \frac{dq}{dt} \tag{2.46}$$

where $q =$ displacement $q = [q, \theta]$

2.9.4 Basic 1-Port Elements

A 1-port element is addressed through a single power port, and at the port a single pair of effort and flow variables exists.

Here we first deal with the most primitive 1-ports. We consider, in order, 1-port elements that dissipate power, store energy, and supply power.

Non-conservative Components (1-port resistor):

The 1-port resistor is an element in which the effort and flow variables at the single port are related by a static function. These are mechanical components that dissipate energy. In physical terms, a resistor is an idealization of such devices as electric resistor, damper, torsional damper, and hydraulic damper. The mechanical dashpot is a 1-port resistor. If an ideal dashpot is characterized by a force–velocity relationship such as;

$$F = bV \tag{2.47}$$

where $b = [b, b_\theta]$ is the dashpot constant (damping coefficient, $b[Ns/m]$ and torsional damping coefficients $[Nms/rad]$), then it is represented as a 1-port R element. Since F is an effort and is V a flow, the constitutive relationship also fits our definition of a 1-port resistor.

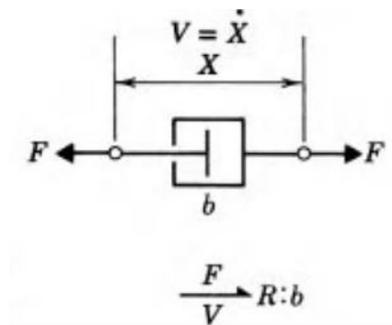


Figure 2.19. 1-Port Resistor with Bond-graph representation

1-port resistor Energy:

$$E = \int P dt = \int e f dt \tag{2.48}$$

where P = Power

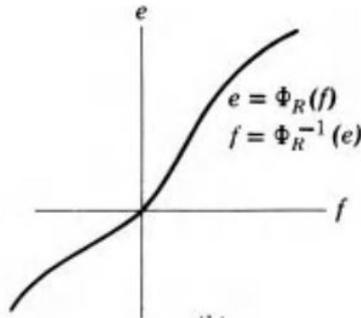


Figure 2.20. 1-Port Resistor Input-Output Plot

It can be seen from equation 2.48 that energy is a function of time and since we cannot go back in time, resistor is hence a dissipative component.

1-Port Conservative Components (Compliance):

These are 1-port devices in which a static constitutive relation exists between an effort and a displacement. Such a device stores and gives up energy without loss. In bond graph terminology, an element that relates e to q is called a 1-port capacitor or compliance. In physical terms, a capacitor is an idealization of such devices as springs, torsion bars, electrical capacitors, gravity tanks, and hydraulic accumulators.

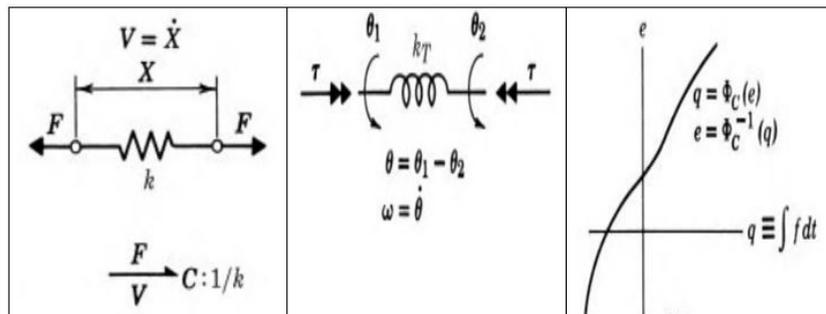


Figure 2.21. 1-Port Compliance with Bond-graph Representation

$$e = \phi_c^{-1} q \tag{2.49}$$

where q = displacement $[x, \theta]$ for spring and torsional spring respectively.

$$q = \int f dt \rightarrow f = dq/dt \tag{2.50}$$

1-port Compliance Energy:

$$E = \int_0^t P dt = \int_0^t e f dt = \int_0^q e dq \tag{2.51}$$

$$\int_0^q e dq = - \int_q^0 e dq \tag{2.52}$$

$$E_{0q} = -E_{q0} \tag{2.53}$$

Since E is a function of q from equation 2.51, energy is therefore conservative.

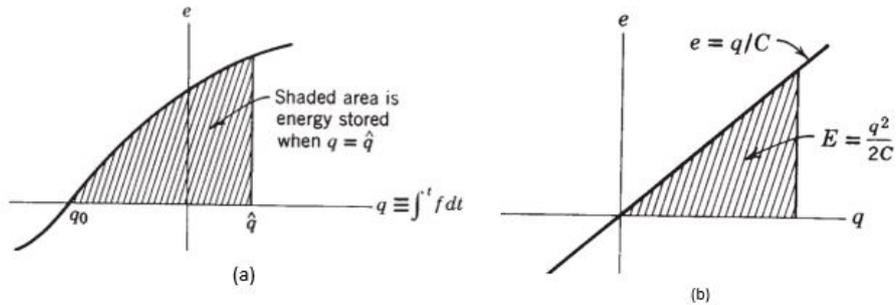


Figure 2.22. Area interpretation of stored energy for a 1-port capacitor: (a) nonlinear case; (b) linear case.

Compliance Derivative Causality:

It can be shown that if effort, e is the input to a compliant subsystem, a derivative causality is obtained as shown in figure 2.23.

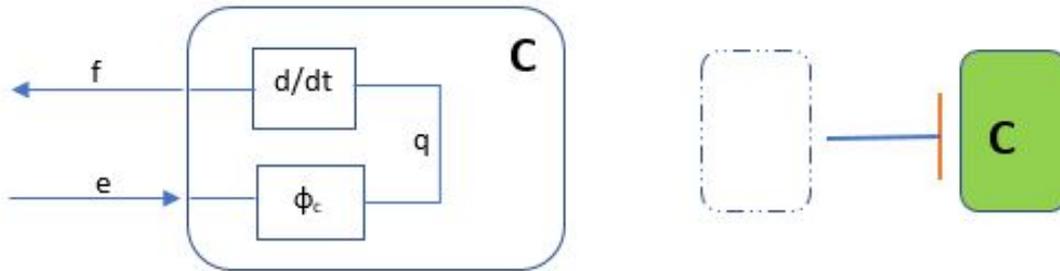


Figure 2.23. Compliance Derivative Causality

With derivative causality, there is no initial condition and hence no state variable.

Compliance Integrative Causality:

It can also be shown that if flow is the input to a compliance subsystem, an integrative causality is obtained as shown in figure 2.24.

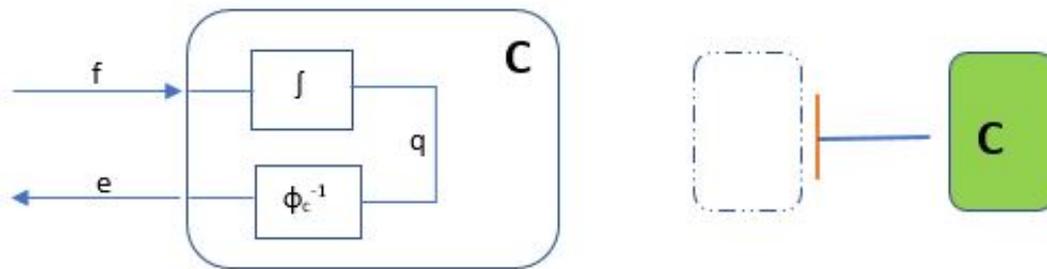


Figure 2.24. Compliance Integrative Causality

With integrative causality, an initial condition is obtained and hence a state variable.

$$\dot{q} = f \rightarrow \text{State equation, 1 initial condition}$$

$$e = \phi_c(q) \rightarrow \text{output}$$

1-Port Conservative Components (Inertial):

A second energy-storing 1-port arises if the momentum p is related by a static constitutive law to the flow f . Such an element is called an inertia in bond graph terminology. The bond graph symbol for an inertia, the constitutive relation, and several physical examples are shown in Figure 2.25. The inertia is used to model inductance effects in electrical systems, and mass or inertia effects in mechanical or fluid systems.

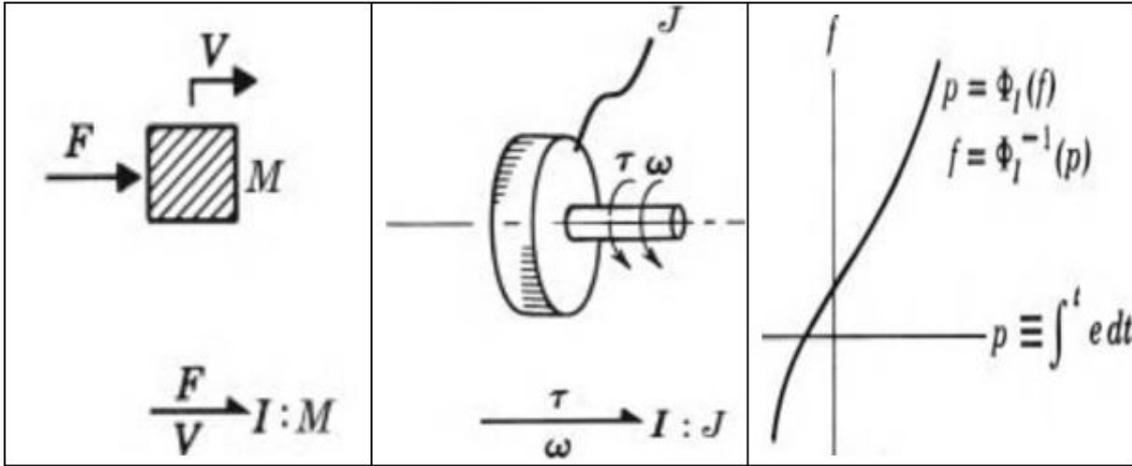


Figure 2.25. 1-Port Inertial with Bond-graph Representation

$$f = \phi_I^{-1}p \quad (2.54)$$

where $P = \text{momentum } [P, P_\theta]$ for linear and angular momentum respectively.

$$p = \int e dt \rightarrow e = dp/dt \quad (2.55)$$

1-port Inertial Energy:

$$E = \int_0^t P dt = \int_0^t e f dt = \int_0^p f dp \quad (2.56)$$

$$\int_0^p f dp = - \int_p^0 f dp \quad (2.57)$$

$$E_{0p} = -E_{p0} \quad (2.58)$$

Since E is a function of p from equation 2.56, energy is therefore conservative.

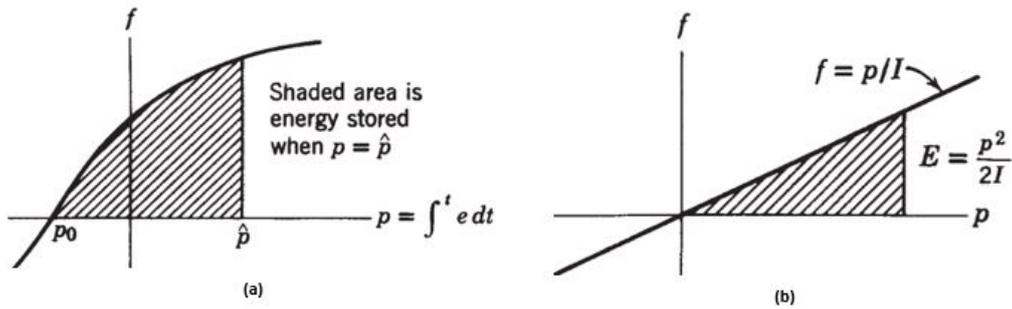


Figure 2.26. Area interpretation of stored energy for 1-port inertia. (a) Nonlinear case; (b) linear case

Inertial Derivative Causality:

It can be shown that if flow, f is the input to a inertial subsystem, a derivative causality is obtained as shown in figure 2.27.

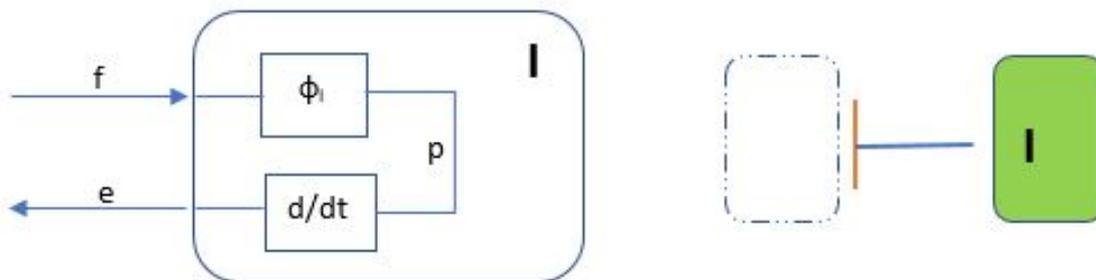


Figure 2.27. Inertial Derivative Causality

With derivative causality, there is no initial condition and hence no state variable.

Inertial Integrative Causality:

It can also be shown that if effort, e is the input to a compliance subsystem, an integrative causality is obtained as shown in figure 2.28.

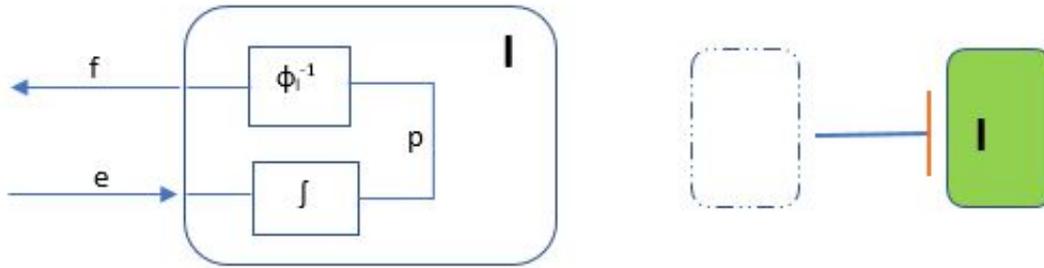


Figure 2.28. Inertial Integrative Causality

With integrative causality, an initial condition is obtained and hence a state variable.

$\dot{p} = e \rightarrow$ State equation, 1 initial condition
 $f = \phi_I(p) \rightarrow$ output;
 p is the state variable for e input

Integrative and Derivative Path Summary for Inertial and Compliant Components is shown in figure 2.29

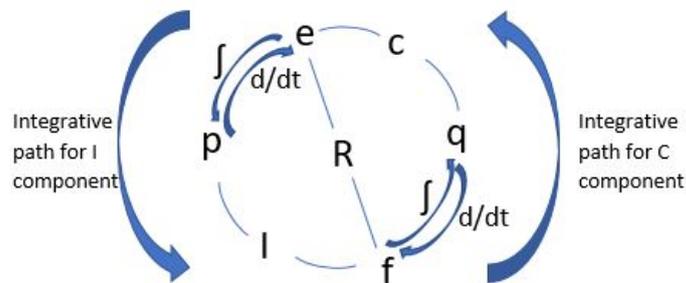


Figure 2.29. Integrative and Derivative Path Summary for Inertial and Compliant Components

1-Port Supply (Effort and Flow Source):

the 1-port sources are idealized versions of voltage supplies, pressure sources, vibration shakers, constant-flow systems, and the like. In each case, an effort or flow is either maintained reasonably constant, independent of the power supplied or absorbed by the source, or constrained to be some particular function of time.

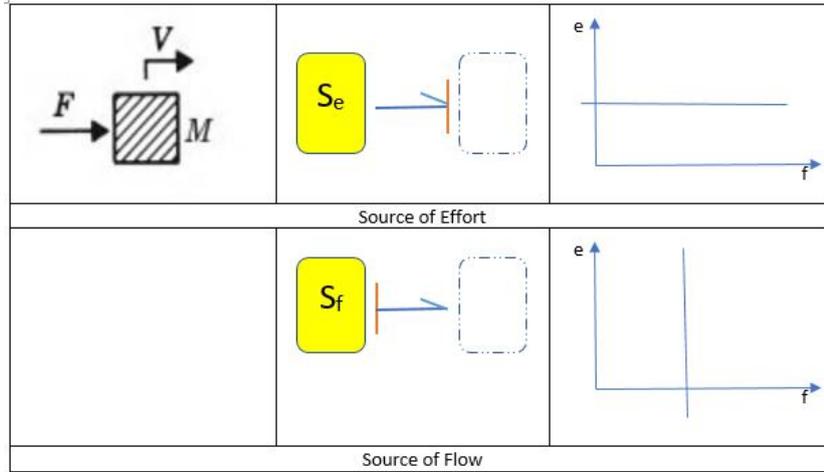


Figure 2.30. 1-Port Supply with Bond-graph Representation

1-port Source Energy Consideration:

For Effort source, $e = \text{constant}$

$$E = \int P dt = \int e f dt = e \int f dt \tag{2.59}$$

The longer the supply is used, the more the energy obtained from it.

2.9.5 Basic 2-Port Elements:

The 2-ports to be discussed here are ideal in the specific sense that power is conserved. If any 2-port, —TP—, has the sign convention

$$\begin{matrix} e_1 & TP & e_2 \\ f_1 & & f_2 \end{matrix},$$

then power conservation means that at every instant of time

$$e_1(t)f_1(t) = e_2(t)f_2(t) \tag{2.60}$$

where the bonds are numbered so that all variables can be simply described as a type, e, f, p, or q with a bond number subscript.

Equation 2.60 states that whatever power is flowing into one side of the 2-port is simultaneously flowing out of the other side.

Transformers are found to satisfy equation 2.60 and the constitutive laws for an ideal 2-port transformer is

$$e_1 = m e_2, m f_1 = f_2, \tag{2.61}$$

parameter m is called the transformer modulus and the subscripts 1 and 2 correspond to the two ports, as shown in Figure 2.31 for ideal rigid lever and gear pair.

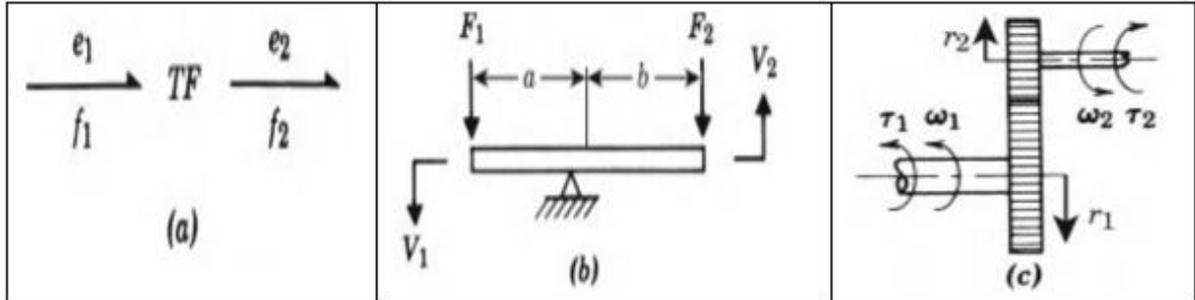


Figure 2.31. Transformers: (a) bond graph; (b) ideal rigid lever; (c) gear pair

The lever is an ideal transformer because kinematics dictates that $(b/a)V_1 = V_2$ and moment equilibrium requires $F_1 = (b/a)F_2$. This is exactly the definition for the ideal transformer from Equation 2.61 with the ratio (b/a) playing the role of the modulus m .

The same is also applicable to the ideal gear set,

$$\begin{aligned} (r_1/r_2)\omega_1 &= \omega_2 \\ \tau_1 &= (r_1/r_2)\tau_2 \end{aligned}$$

Other form of 2-port elements such as Gytrators (GY), Modulated Transformers (MTF), Capacitive Capacitive (CC), Capacitive Inductive (CI), etc, are not covered within this scope.

2.9.6 Multi-Port Junction Elements:

The idea of Junction formulation is to represent in multiport form the two types of connections, which, in electrical terms, are called the *series* and *parallel* connections. Such connections are extendible to all types of systems, including mechanical systems as shall be considered here.

Junction elements are ideal in that power is neither dissipated nor stored.

Common Effort Junction:

This is also known as flow junction or 0-junction and it is symbolized by zero(0) with bonds emanating from it. It represents the parallel connection of the electrical system.

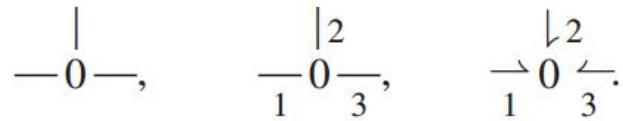


Figure 2.32. Common Effort Junction Representation

Using the inward power sign convention (+ve for inward bond) shown in the last version of the junction of figure 2.32 , this implies

$$e_1 f_1 + e_2 f_2 + e_3 f_3 = 0. \quad (2.62)$$

The common effort junction is defined such that all efforts are the same, thus,

$$e_1(t) = e_2(t) = e_3(t). \quad (2.63)$$

hence,

$$f_1(t) + f_2(t) + f_3(t) = 0. \quad (2.64)$$

The efforts on all bonds of a common effort junction are always identical, and the algebraic sum of the flows always vanishes. In other words, if power is flowing in on two ports of the three, then it must be flowing out of the third port.

A good example of the mechanical representation of the common effort junction is shown in figure 2.33

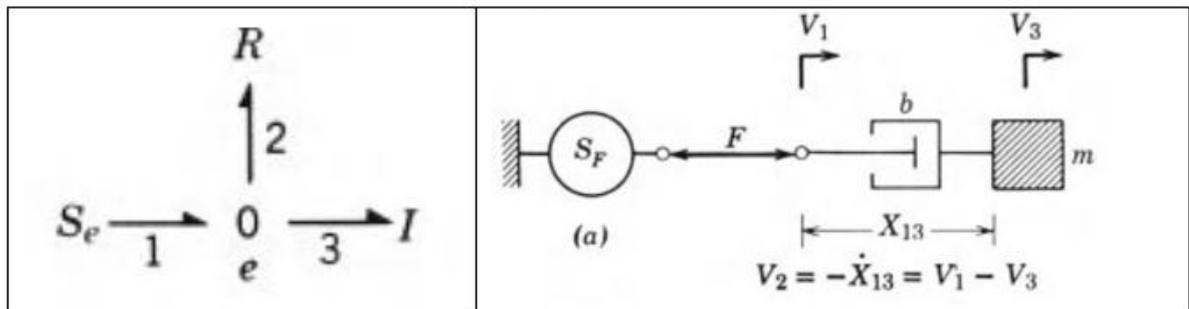


Figure 2.33. Common Effort Junction Representation for Mechanical System

The causal rule for common effort junction stipulates that there must be one effort imposed on a common effort junction. All other efforts are imposed by the junction to the components or subsystems.

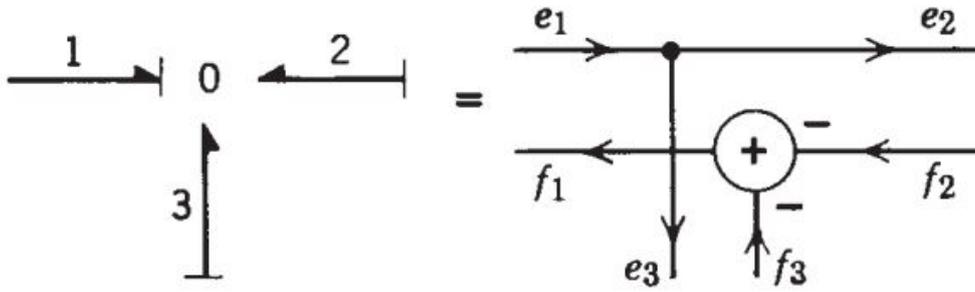


Figure 2.34. Causal Rule and Block Representation for Common Effort Junction

Common Flow Junction:

This is also known as effort junction or 1-junction and it is symbolized by one(1) with bonds emanating from it. It represents the series connection of the electrical system.

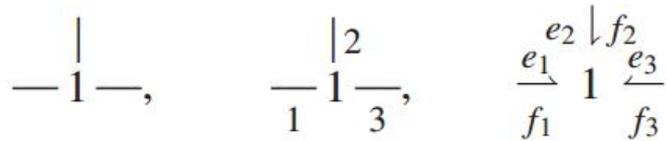


Figure 2.35. Common Flow Junction Representation

Using the inward power sign convention (+ve for inward bond) shown in the last version of the junction of figure 2.35 , this implies

$$e_1 f_1 + e_2 f_2 + e_3 f_3 = 0. \tag{2.65}$$

The common flow junction is defined such that all flows are identical, thus,

$$f_1(t) = f_2(t) = f_3(t). \tag{2.66}$$

hence,

$$e_1(t) + e_2(t) + e_3(t) = 0. \tag{2.67}$$

The efforts on all bonds of a common flow junction are always identical, and the algebraic sum of the efforts always vanishes.

A good example of the mechanical representation of the common flow junction is shown in figure 2.36

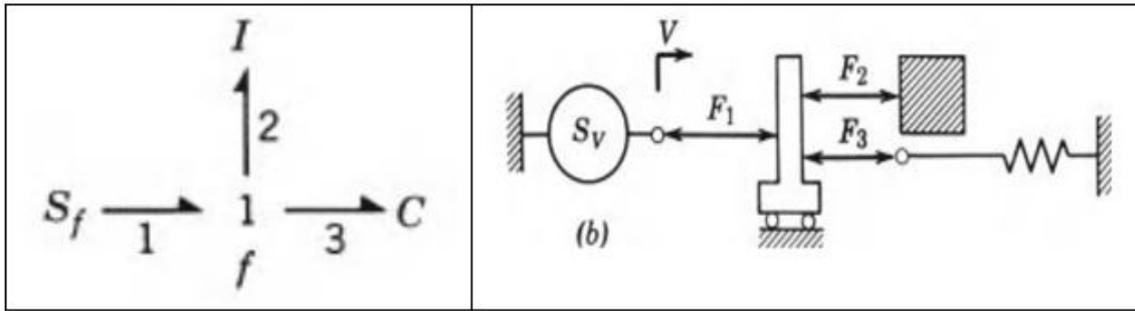


Figure 2.36. Common Effort Junction Representation for Mechanical System

The causal rule for common flow junction stipulates that there must be one flow input on the common flow junction. All other flows are outputs from the common flow junction to the components or subsystems.

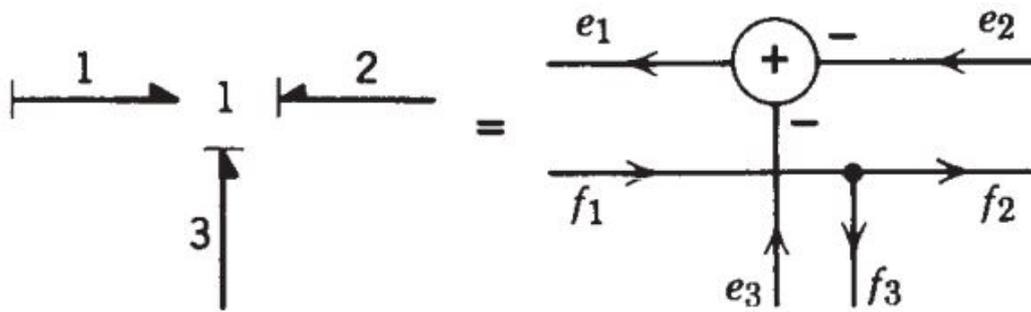


Figure 2.37. Causal Rule and Block Representation for Common Flow Junction

2.10 System Control

In this section, we shall discuss the approach adopted for controlling the modelled plant, their benefits and limitations will also be highlighted. The controller design is based on Model Predictive Controller (MPC) as will be demonstrated below according to ([31]: Sachin C. Patwardhan, 2106).

2.10.1 Model Predictive Control(MPC)

Considering that more often than not, the parameters of the plant system change, it is desirable to approach the problem using a controller that is capable of handling online estimation. An adaptive MPC is suitable for this and more advantages which include: It's capability to systematically and optimally

- handle multi-variable interactions

- impose operating input and output constraints
- process non-linearity

Theory Behind MPC: MPC is based on iterative, finite-horizon optimization. At time, t the current plant state is sampled and a cost minimizing control strategy is computed (via a numerical minimization algorithm) for a relatively short time horizon in the [12]. Given a model of plant dynamics, possible consequences of the current input- moves on future plant behaviour can be forecast online, and used while deciding the input-moves that will not violate the future constraints.

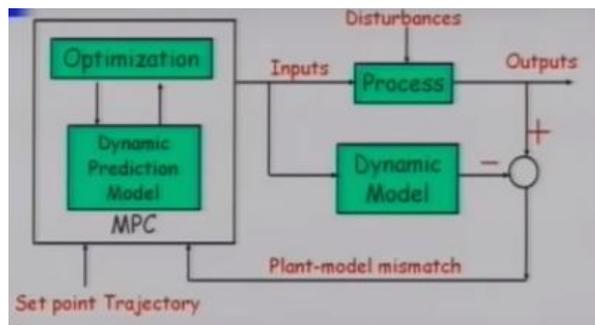


Figure 2.38. System's Schematic Representation

The Dynamic Model is useful for online forecast over a moving time horizon (or window). At each sampling time, an optimization problem is formulated over the finite window.

Description of Approach

Finite Horizon Formulation: Optimization problem is formulated over a finite window of time starting from the current instant.

At each sampling instant, a constrained optimization problem is formulated over the window and solved.

Moving Horizon / Window: the time window keeps moving or receding from

$$[k, k + p] \rightarrow [k + 1, k + p + 1] \quad (2.68)$$

... and so on

p is known as the prediction horizon

Proactive Constraint Management: Given a good dynamic model, an online forecast can be performed any possible constraint violation can be predicted within the window.

Components of MPC

- Internal Model and State Estimator:

- Discrete Line State Space Model developed from mechanistic approach / time series modelling (FIR or Finite State Model were used before)
- State Estimator: Open-loop observer / Kalman Predictor /Kalman Filters /Leunberg Observer / Innovative form of State Observer developed from ARX/AR-MAX/BJ model
- Prediction of Future Plant Behaviour This involves handling drifting disturbances and plant model mismatch
- On-line Constraint Optimization Strategy: The two major strategies are Linear and Quadratic Programming approaches

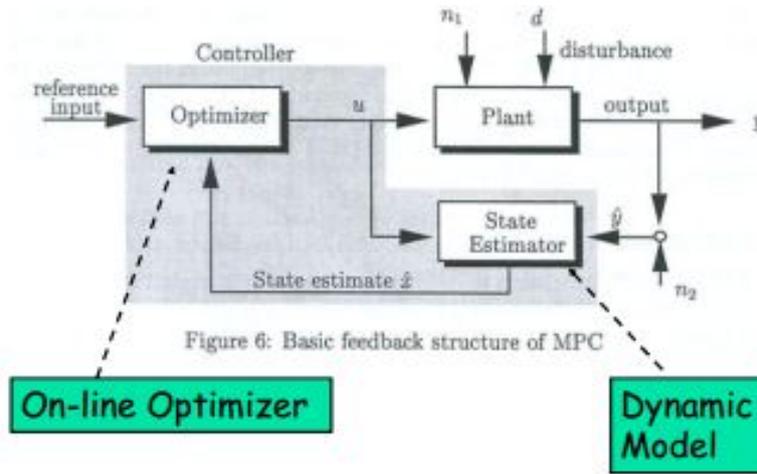


Figure 2.39. Basic Feedback Structure of MPC [13]

• **State Estimator and Predictor:**

We shall adopt state estimation and prediction using prediction form of observer;

$$\begin{aligned} \hat{x}(k+1|k) &= \Phi\hat{x}(k|k-1) + \Gamma u(k) + L_p e(k) \\ e(k) &= y(k) - C\hat{x}(k|k-1) \end{aligned} \tag{2.69}$$

where $L_p = L_{inf}$

Such an observer can be developed using the outlined state estimators.

Estimation of Current State And Innovation:

$$\begin{aligned} \hat{x}(k|k-1) &= \Phi\hat{x}(k-1|k-2) + \Gamma u(k-1) + L_p e(k-1) \\ e(k-1) &= y(k-1) - C\hat{x}(k|k-2) \end{aligned} \tag{2.70}$$

Future Prediction

Innovative Bias Approach:

When the model is perfect, the innovation sequence $e(k)$ is a *zero mean white noise* signal. However with the plant affected by drifting coloured unmeasured drifting disturbances which has not been accounted for by the plant, the innovation sequence $e(k)$ is no longer a white noise and hence, the expectation of $e(k)$,

$$E\{e(k)\} \neq 0 \quad (2.71)$$

The effect of model plan mismatch and/or unmeasured disturbance signal is extracted by filtering the innovation through a unity gain low-pass filter to obtain $e_f(k)$. This obtains the non-zero mean of the signal.

$$e_f(k) = \Phi e_f(k-1) + [1 - \Phi_e] e(k) \quad (2.72)$$

$$\Phi_e = \text{diag}[\alpha_1 \ \alpha_2 \ \dots \ \alpha_r]$$

where

$0 \leq \alpha_i < 1$ for $i = 1, 2, 3, \dots, r$ are tuning parameters

Given a guess of future manipulated inputs,

$$\{u(k+j|k) : j = 0, 1, 2, \dots, p-1\}$$

Model prediction over future time window $[k+1 : k+p]$ are generated using the discrete dynamic model as follow, assuming that $e_f(k)$ is constant:

future instance (k+1):

$$\begin{aligned} \hat{z}(k+1) &= \Phi \hat{z}(k) + \Gamma u(k|k) + L e_f(k) \\ \hat{y}(k+1|k) &= C \hat{z}(k+1|k) + e_f(k) \end{aligned}$$

$$\hat{z}(k) = \hat{x}(k|k-1) \quad (2.73)$$

where \hat{z} is the prediction variable and \hat{x} is the observer variable.

future instance (k+2):

$$\begin{aligned} \hat{z}(k+2) &= \Phi \hat{z}(k+1) + \Gamma u(k+1|k) + L e_k \\ &= \Phi^2 \hat{z} + \Phi \Gamma u(k|k) + \Gamma u(k+1|k) + (\Phi + I) L e_f(k) \\ \hat{y}(k+2|k) &= C \hat{z}(k+2) + e_f(k) \end{aligned}$$

...
future instance (k+p):

$$\begin{aligned}\hat{z}(k+p) &= \Phi\hat{z}(k+p-1) + \Gamma u(k+p-1|k) + Le_k \\ &= \Phi^p\hat{z} + \Phi^{p-1}\Gamma u(k|k) + \Phi^{p-2}\Gamma u(k+1|k) + \Gamma u(k+p-1|k) + (\Phi^{p-1} + \Phi^{p-2} + I)Le_f(k) \\ \hat{y}(k+p|k) &= C\hat{z}(k+p) + e_f(k)\end{aligned}\tag{2.74}$$

Hence, the future trajectory prediction for a given prediction horizon p is ;

$$\begin{aligned}\hat{y}(k+p|k) &= [C\Phi^p]\hat{x}(k|k-1) + (C\Phi^{p-1}\Gamma)u(k|k) + \dots + \\ &(C\Gamma)u(k+p-1|k) + [\Phi^{p-1}L + \Phi^{p-2}L + \dots + L + I]e_f(k)\end{aligned}\tag{2.75}$$

This shows that

$$\begin{aligned}\text{Future Output Prediction} &= \\ &\text{Effect of the past States on the future outputs} + \\ &\text{Effect of the future outputs on the future outputs} + \\ &\text{Effect of Plant Model mismatch and the unmeasured disturbance on the future} \\ &\text{outputs}\end{aligned}$$

p is the prediction horizon. It is the period (or window) over which the future behaviour is being predicted.

2.10.2 Optimization Parameters

- **Model Prediction Equation**

The prediction generated using innovation bias approach is equivalent to carrying out prediction using observer augmented with an artificially introduced integrated white noise model, that is, the prediction generated using the following dynamic system.

$$\begin{aligned}\hat{z}(k+j+1) &= \Phi\hat{z}(k+j) + \Gamma u(k+j|k) + L\hat{\epsilon}(k+j) \\ \hat{\epsilon}(k+j+1) &= \hat{\epsilon}(k+j) \\ \hat{y}(k+j|k) &= C\hat{z}(k+j|k) + \hat{\epsilon}_f(k+j)\end{aligned}\tag{2.76}$$

Initial conditions:

$$\hat{z}(k) = \hat{x}(k|k-1) \text{ and } \hat{\epsilon}(k) = \hat{\epsilon}_f(k) \text{ for } j = 0,1,2,3,\dots,p-1$$

The introduction of white noise in prediction helps to achieve an offset-free closed loop behaviour.

- **Constraints in Manipulated Inputs**

To reduce the dimension of the online optimization problem, degrees of freedom available for shaping the future trajectory are often restricted to the first q moves.

$$\{u(k|k), u(k+1|k), \dots, u(k+q-1|k)\} \quad (2.77)$$

By imposing input constraints of the form

$$u(k+q|k) = u(k+q+1|k) = \dots = u(k+p-1|k) = u(k+q-1|k) \quad (2.78)$$

q is known as the *control horizon*, which is the horizon over which the inputs are allowed to change and after which it is kept constant

- **Bounds on Manipulated Inputs and Outputs**

$$u^l \leq u(k+j|k) \leq u^H \text{ for } j=0,1,2,\dots,q-1$$

$$y^l \leq \hat{y}_c(k+j|k) \leq y^H \text{ for } j=0,1,2,\dots,q-1$$

The output bound is useful for zone control where superscripts l and H indicates lower and upper bounds respectively This constraint gives rise to zone control variable

- **Bounds on the rate of change of Manipulated Inputs**

$$\Delta u^l \leq \Delta u(k+j|k) \leq \Delta u^H \text{ for } j=0,1,2,\dots,q-1 \quad (2.79)$$

$$\Delta u(k+j|k) = u(k+j|k) - u(k+j-1|k) \text{ for } j=0,1,2,\dots,q-1 \quad (2.80)$$

$$\Delta u(k|k) = u(k|k) - u(k-1) \quad (2.81)$$

Alternatively, degree of freedom for shaping the future trajectory can be used through input blocking constraints

Since predictions are carried out online, we can choose output moves that respect the above constraints.

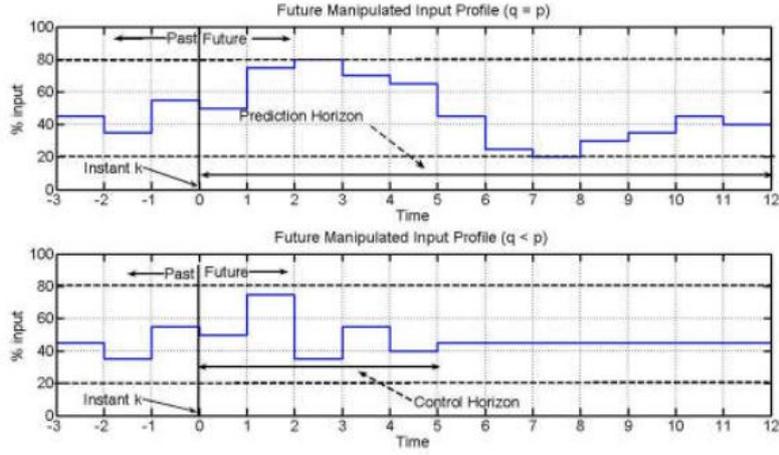


Figure 2.40. Schematic Representation of Control Horizon and Input Bounds

2.10.3 Future Set-point Trajectory:

In addition to predicting the future output trajectory, at each instance, *filtered future set-point* trajectory is generated using a reference difference equation system of the form:

$$\begin{aligned}
 x_r(k+j+1|k) &= \Phi_r x_r(k+j|k) + [I - \Phi][r(k) - y(k)] \\
 y_r(k+j+1|k) &= y(k) + x_r(k+j+1|k) \text{ for } j = 0, 1, 2, 3, \dots, p-1 \\
 \Phi_r &= \text{diag}[\gamma_1 \ \gamma_2 \ \dots \ \gamma_r] \\
 0 \leq \phi_i &\leq 1 \text{ for } i = 1, 2, 3, \dots, r
 \end{aligned} \tag{2.82}$$

with initial condition $x_r = \bar{0}$

Here $r(k) \in R^r$ represent the set-point vector. The tuning parameter, γ_i is chosen once to generate a smooth trajectory from the current set-point.

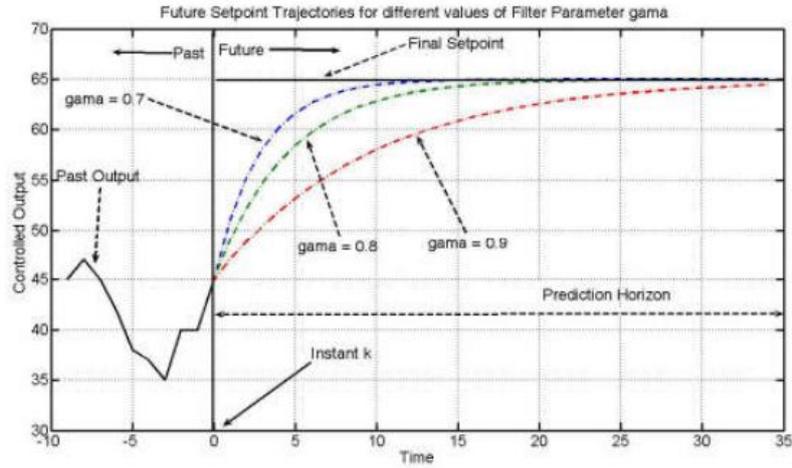


Figure 2.41. Future Setpoint Trajectory

In some practical cases, simple reference signal previewing is used instead of applying filters. [17]

When the reference vector is a 1-by- n_y signal, where n_y is the number of outputs, there is no reference signal previewing. The controller applies the current reference values across the prediction horizon.

To apply signal previewing, reference signal matrix of dimension N-by- n_y has to be specified. N is the number of time steps for which the reference values are specified.

Here, $1 < N \leq p$, and p is the prediction horizon.

Previewing usually improves performance, since the controller can anticipate future reference signal changes. The first row of reference signal matrix specifies the n_y references for the first step in the prediction horizon (at the next control interval $k = 1$), and so on for N steps.

If $N < p$, the last row designates constant reference values for the remaining $p - N$ steps.

For more insight, suppose $n_y = 2$ and $p = 15$. At a given control instant, the signal connected to the ref inport is:

$$\begin{bmatrix} 2 & 5 \\ 2 & 6 \\ 2 & 7 \\ 2 & 8 \end{bmatrix} \rightarrow \begin{matrix} k = 1 \\ k = 2 \\ k = 3 \\ k = 4 \end{matrix}$$

The signal informs the controller that:

- Reference values for the first prediction horizon step $k = 1$ are 2 and 5.
- The first reference value remains at 2, but the second increases gradually.
- The second reference value becomes 8 at the beginning of the fourth step $k = 4$ in the prediction horizon.
- Both values remain constant at 2 and 8 respectively for steps $5 \rightarrow 15$ of the prediction horizon.

This demonstrates set-point filtering in a more visualizable way. However, when precision and smoothness of trajectory is of utmost importance, using of filter is inevitable.

It is worthy of nothing that the reference vector $r(k)$ dimension must not change from one control instant to the next each of its element must be a real number.

2.10.4 Constraint MPC Formulation:

Given the prediction model, input and output constraints, and desired set-point trajectory, the MPC optimization problem at the sampling instant is formulated as follows:

$$\left\{ \begin{array}{l} \min_{u_f(k)} E_s(k+p|k)^T W_\infty E_s(k+p|k) \\ \quad + \sum_{j=1}^{p-1} E(k+j|k) W_E E(k+j|k) \\ \quad + \sum_{j=0}^{q-1} \Delta u(k+m_j|k) W_{\Delta u} u(k+m_j|k) \\ u_f(k) = [u(k|k)^T \quad u(k+m_1|k)^T \quad \dots \quad u(k+m_{q-1}|k)^T]^T \\ E(k+j|k) = y_r(k-j|k) - \hat{y}(k+j|k) \\ E_s(k+p|k) = \hat{x}(k+j|k) - \bar{x}_s(k) \end{array} \right. \quad (2.83)$$

This optimization is a minimization problem subject to :

- Model prediction Equation
- bounds on future inputs and predicted outputs

W_E = Symmetric positive definite error weighting matrix
 $W_{\Delta u}$ = Symmetric positive semi-definite input weighting matrix

These matrix are treated as tuning parameters which are used to shape the closed loop behaviour of the input and output.

The terminal state weighting function W_∞ can be solved by solving discrete Lyapanov equation. When poles of Φ are inside the unit circle, W_∞ can be solved thus,

$$W_\infty = C^T W_E C + \Phi^T W_\infty \Phi \quad (2.84)$$

When some poles are outside the unit circle, the procedure for computing the terminal weighting matrix is given in Muske and Rawlings (1993)^[16]

2.10.5 Steady State Target Computation

$$\begin{aligned} \min_{u_s(k)} & \left[r(k) - \bar{y}(k) \right]^T W_E \left[r(k) - \bar{y}(k) \right] \\ & \text{subject to} \end{aligned} \tag{2.85}$$

$$\begin{aligned} [I - \Phi] \bar{x}_s(k) &= \Gamma u_s(k) + L e_f(k) \\ \bar{y}(k) &= C \bar{x}_s(k) + e_f(k) \\ u^L &\leq u_s(k) \leq u^H \end{aligned}$$

Case 1: If number of manipulated inputs = Number of controller outputs, then, an unconstrained solution exists.

$$\bar{u}_s(k) = [C(\Phi - I)^{-1}\Gamma]^{-1} [r(k) - (C(I - \Phi)^{-1}L + I)e_f(k)] \tag{2.86}$$

$$\bar{x}_s(k) = [I - \Phi]^{-1} [\Gamma u_s(k) + L e_f(k)] \tag{2.87}$$

Case 2: When the number of outputs is greater than the number of inputs, zone variable are used.

2.10.6 Moving Horizon Implementation

The resulting constrained optimization problem is solved online for each sampling instant using any standard optimization method.

The controller is implemented in a moving horizon framework, thus solving optimization problem over the window $[k, k + p]$. Only the first optimal move is implemented on the plant, that is,

$$u(k) = u_{opt}(k|k) \tag{2.88}$$

The optimization problem is formulated again on the next sampling instance over the time window $[k + 1, k + p + 1]$ based on the updated information from the plant and is re-resolved.

Note: Reducing q reduces manipulability but enhances optimization efficiency.

2.10.7 Quadratic Programming (QP)

A constrained optimization problem is known as Quadratic Programming (QP) formulation if it has the following standard form;

$$\min_U \left\{ \frac{1}{2} U^T H U + F^T U \right\} \tag{2.89}$$

subject to
 $AU \leq b$

where H is a complex matrix of $[\Phi, \Gamma, C]$

A large dimensional QP formulation can be solved very quickly using an efficient search method. Through a series of algebraic manipulations, the Constrained MPC formulation can be transformed to a Quadratic Programming (QP) Problem. To understand how the MPC optimization problem can be transformed to a quadratic programming problem, consider MPC formulation without terminal state weighting just for simplicity.

Defining the future input vector U_f and the predicted output vector $\hat{Y}_f(k)$

$$U_f(k) = [u(k|k)^T \ u(k+m_1|k)^T \ \dots \ u(k+m_{q-1}|k)^T]^T$$

$$\hat{Y}_f(k) = [\hat{Y}(k|k)^T \ \hat{Y}(k+m_1|k)^T \ \dots \ \hat{Y}(k+m_{q-1}|k)^T]^T$$

Then the prediction model will be

$$\hat{Y}_f(k) = S_x \hat{x}(k|k-1) + S_u U_f(k) + S_e \in_f(k) \quad (2.90)$$

$$S_x = \begin{bmatrix} C\Phi \\ C\Phi^2 \\ \dots \\ C\Phi^p \end{bmatrix} \quad S_e = \begin{bmatrix} CL + I_r \\ C\Phi L + CL + I_r \\ \dots \\ C\Phi^{p-1}L + \dots + CL + I_r \end{bmatrix}$$

$$S_u = \begin{bmatrix} C\Gamma_u & 0 & 0 & \dots & 0 \\ C\Phi\Gamma & C\Gamma_u & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ C\Phi^{q-1}\Gamma_u & C\Phi^{q-2}\Gamma_u & 0 & \dots & C(\Phi+1)\Gamma_u \\ \dots & \dots & \dots & \dots & \dots \\ C\Phi^{p-1}\Gamma_u & C\Phi^{p-2}\Gamma_u & 0 & \dots & C(\Phi^{p-q} + \dots + 1)\Gamma_u \end{bmatrix}$$

S_x is the matrix relating to the effect of the past state to the future predictions
 S_e = Matrix relating the effect of the past unmeasured disturbances and the model plant mismatch on the future predictions

S_u (Number of inputs \times Number of prediction horizon)

It is the matrix relating the future manipulated inputs on future predictions. It consists of impulse response coefficients of the model, and it is referred to as Dynamic Matrix.

2.10.8 Tuning of MPC

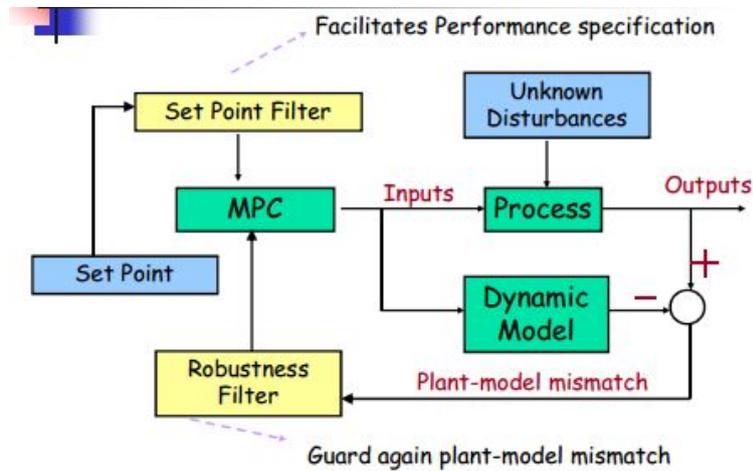


Figure 2.42. Tuning of MPC

- Prediction Horizon: Typically chosen close to open loop settling time.
- Control Horizon: Typically chosen small to avoid model inversion problems
- Input rate constraints
- Zone / Range Control: Not necessary to specify set points on each output. Instead, high and low limits can be defined within which output should be maintained

Chapter 3

METHODOLOGY: System Modelling and Control of Sanforized Compactor

The previous chapter has been dedicated to the theoretical analysis of the methodology used in this work. Here, we will demonstrate how they are applied in the development of this project, while pointing out the necessary adjustments and the limitations. All the assumptions made in order to simplify the system are also pointed out here. The result analysis of this design shall be discussion in the next chapter.

3.1 Specification Requirement:

The main objectives of this modelling and control include:

- to optimize automation of Sanforizing machine by controlling the compaction ratio of the fabric
- to obtain a fast regulation with the least possible rise time, settling time and overshoot, that guarantees a stable output with acceptable 20% variation in amplitude.
- to find a good compromise between productive and maintenance cost, considering compressive rubber belt usage.

The above requirements are subject to the following constraints;

- bounds on the input of electric motor = $[5 \ 50]$ m/min
- bounds on the compaction ratio = $[1 \ 0]$

The last constraint has been imposed to prevent negative compaction and to limit compaction to 100% maximum value.

3.2 System Modelling:

Considering the complexity of the system, it was necessary to split the compaction process into two stages for ease of modelling and control, and in order to achieve the above goals. The splitting will be based on the variables that influence compaction, and will be in stages I and II. It will be seen that the subsystems will be inter-dependent on each other for variables.

As was mentioned in chapter 2, fabric compaction with Sanforizing machine is dependent mainly on some variables such as;

- steam cylinder compression on the rubber belt
- tension on fabric
- motor speed

Other variables that influence compaction include;

- fabric angle of entry
- temperature of steam drum
- amount of humidity on fabric
- fabric material etc. For simplicity

For simplicity, only the main variables will be modelled as input variables. The theoretical description of the principle behind system modelling by means of Set membership and Bond graph approaches have been demonstrated. This chapter will extract these principles with just a few modifications to achieve the desired models of the subsystems of Stages I and II compaction.

Only a SISO¹ system modelling has been considered so far with Set membership approach of system Identification. The stage II compaction subsystem is however a MISO² system. For this reason, it is necessary to modify the approach for MISO Identification as will be demonstrated in the Stage II compaction subsection.

Bond graph approach will be used here as an alternative modelling approach to approximate the subsystems and in principle, obtain an approximate result.

For further insight on how fabric volume compaction can be predicted with a good degree of accuracy and without access to the physical specimen. through FEA³, please refer to Appendix A.

¹MISO - Single Input Single Output

²MISO - Multiple Input Single Output

³FEA - Finite Element Analysis

3.3 Stage I Compaction:

This first level of compaction is obtained at the compaction zone of the steam cylinder. The amount of compaction obtained here is a function of the force of compression. The force of compression is turn a function of the position of the free compression roller. This level of compaction is characterised by slow dynamics. As a result, it is desired to over-compact the fabric at this stage and then use a faster dynamic system at stage II to obtain a more regular and precise compaction.

3.3.1 Subsystem Description

This stage of compaction is obtained through the compressive effect of pressure roll and the steam cylinder on the compressive rubber belt as demonstrated in figure(3.1).

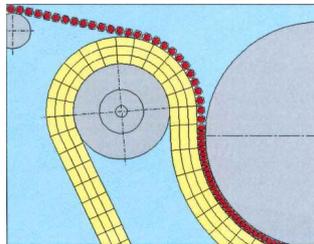


Figure 3.1. Stage I Compaction (by Compression)

The compressive rubber belt of thickness, 50.8mm to 70 mm [2] is worn in such a way that it forms a convex shape on the free rolling pressure roll and the tension roll, and then a concave shape on the motorized steam cylinder as shown. This configuration produces the maximum rubber belt surface extension on the pressure and the tension rolls. A piece of fabric is now laid on the rubber surface on the pressure roll end, with entire system rotating to sandwich the fabric between the rubber surface and steam cylinder [4][3][5].

As the rubber is allowed to straighten, the length on the fabric is observed to exceed the length of the rubber, forming a loop shape. This surplus fabric is available for compressive compaction.[5] Since the fabric can slide smoothly on the the hot, polished and chromium plated steam cylinder, it therefore follows the pattern of the rubber at the compression zone[3]. The fabric exits this unit by-passing the rubber belt extended mode at the opposite end free roll cylinder. The higher the contact pressure of the rubber belt, the greater the compaction. At the entry point, the rubber belt can be compressed at most down to 25% of its normal thickness.[4]

The major factor that influence compaction at this stage is the force of compression. The force The command speed affects the rate of compaction but not the actual compaction level since the rate of compaction at steady state equals zero (0). The force of compression therefore will be considered as the only major input to the stage I compaction. Hence,

we have a SISO stage I subsystem. As mentioned earlier, other variable such as humidity, temperature and fabric angle of entry shall be considered fixed.

Since the motion experienced by every unit of this subsystem is constant, the compaction will be modelled considering a common flow junction (1-junction) modelling strategy. Since the focus is only on the mechanical behaviour of the fabric, only the parameters and variables that affect this behaviour is modelled. The fabric will be modelled as a spring with internal damper. The spring parameter models the elastic nature of fabric while the damping feature models the acceptable permanent fabric deformation.

3.3.2 Model Parameters

Elastic Constant (Young’s Modulus), E: This defines the tensile elasticity, or the tendency of an object to deform along a given axis when opposing forces are applied along that axis. It is a physical property of a material and it is independent of the material dimension. It can be demonstrated that based on the direction of tension imposed on a fabric material, the Elastic constant of cotton fibre can range from 0.821 MPa for 40° to 32.559 MPa for 0° [15]. Also, treating fabric with steam changes slightly the elastic constant of the material.

$$E = \frac{\sigma}{\epsilon} \quad (3.1)$$

where σ = Stress and *epsilon* = Strain

Force Constant, k This is also a physical property of a material that depends on the Elastic Constant the dimension of the material. Like the elastic constant or Young’s modulus, it describes the elastic behaviour of a material for a given tensile or compressive force, but in this case, influenced by the material dimension as seen in equation 3.2.

$$k = \frac{AE}{l} \quad (3.2)$$

where A = Area of the material and l = initial length of the material. ⁴

Damping Coefficient, c This is the internal property of the fabric material that act as friction against the vibration of the fabric spring property. Since it is not desired that the fabric material return to it’s original position after compaction, the damping ratio which is a function of the damping coefficient is greater than one (1).

$$\zeta = \frac{c}{2\sqrt{mk}} > 1 \quad (3.3)$$

where k = mass.

⁴You may reference for further details on how to determine the spring constant of a fabric material [15]

Others Parameters Including other parameters such as; fabric dimension, rubber belt dimension, motorized and free-roll cylinders’ dimension, cylinder mass inertial and so on.

Table 3.1 summarises the values of these parameters.

Parameter	Unit	Value
Young’s Modulus, (Y_m)	MPa	0.738900
Force Constant, k	N/m	2.2903e+03
Fabric internal Damping Coefficient (c)	Ns/m	1600
Fabric angle of Entry, (θ_e)	deg	45°
Fabric Thickness, (t_f)	m	2.7e-3
Initial length of Fabric, (l_{0f})	m	1
Fabric width, (w_f)	m	1.43
Steam Cylinder radius, (r)	m	0.3050
Compression rubber width, (w_r)	m	2.54
Partial Mass Inertial of Rubber Belt (J_b)	kg	401.7297
Moment of Inertia of Steam Cylinder (J_c)	kgm ²	367.866168

Table 3.1. Subsystem Parameters and Corresponding Values

3.3.3 Input / Output Variables

The input and output variables of the stage I subsystem include:

- *Compression Force (F_c):* This is the only input variable to the stage on subsystem. It is hence automatically the manipulated variable of the closed loop control system.
- *First Compaction:* This is the output of the closed loop control system. It shall be modelled here as the spring displacement. It is desired that it tracks the reference given the control system.

3.3.4 Subsystem Modelling with Bond-graph

In order to design a controller for this subsystem, a mathematical representation that mimics the behaviour of the system must be extracted. It must therefore be modelled as plant using any standard modelling approach. This subsystem will be modelled using Bond-graph approach as described in section 2.9. A pictorial representation of the subsystem is shown in figure 3.2

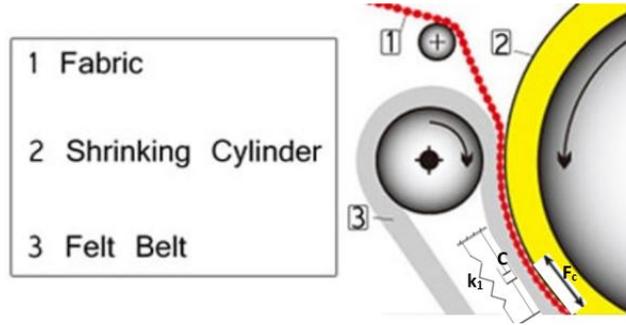


Figure 3.2. Stage I Compaction Scheme

The compaction obtained at this stage is as a result of the difference in pressure level along the circumference of the rubber belt courtesy of its convolution. This pressure difference along the rubber belt and the fabric material is modelled with Bond-graph as a common flow junction (1-junction). The Bond-graph representation of stage I compaction is shown in figure 3.3.

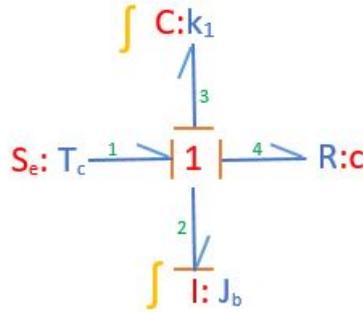


Figure 3.3. Stage I Compaction Bond-Graph Realization

Based on the mass spring damper system representation, the the common flow junction (1-junction) indicates a common speed (flow) of all the parameter components in the compaction zone. This speed is imposed by the inertia of the rubber belt. The force T_c corresponds to the linear compression force on the fabric by the belt [2]. J_b represents the partial mass inertia of the rubber belt on the fabric. k_1 models the spring behaviour of the fabric material as given in equation 3.3.2. c models the damping property of the fabric.

It is note worthy that we have assumed that only a few percentage of the force of compression imposed on the rubber belt is transmitted to the fabric due to the property of high force absorbing capacity of the rubber belt.

3.3.5 Derivation of State Equation

From the Bond-graph representation of figure 3.3, it can be seen there are two states as indicated with f . One state variable is associated with the spring displacement, q_3 and the second is associated with rubber belt momentum p_2 . This results in two state equations.

$$\begin{aligned} \dot{p}_2 &= e_1 - e_3 - e_4 \\ \dot{p}_2 &= T_c - k_1 q_3 - f_4 c_1 \\ \dot{p}_2 &= T_c - k_1 q_3 - \frac{p_2}{J_b} c_1 \end{aligned} \quad (3.4)$$

$$\dot{q}_3 = f_3 = \frac{p_2}{J_b} \quad (3.5)$$

$$\begin{bmatrix} \dot{p}_2 \\ \dot{q}_3 \end{bmatrix} = \begin{bmatrix} \frac{c}{J_b} & -k_1 \\ \frac{1}{J_b} & 0 \end{bmatrix} \begin{bmatrix} p_2 \\ q_3 \end{bmatrix} + \begin{bmatrix} T_c \\ 0 \end{bmatrix} \quad (3.6)$$

$$y = f_3 = \dot{q}_3 = \frac{p_2}{J_b} \quad (3.7)$$

Equation 3.7 shows that the output of the subsystem corresponds to the rate of compaction q_3 .

At steady state,

$$q_3 = \lim_{s \rightarrow 0} \frac{T_c}{k_1 + s J_b (s + c_1 / J_b)} \quad (3.8)$$

3.4 Stage II Compaction

In this section, it will be demonstrated how the second stage of compaction is modelled using both Set-membership System Identification and Bond-graph modelling approaches. Using set-membership system identification approach generates a highly accuracy estimation and considering also the noise in the system. However, in absence of data, using Bond-graph modelling approach can generate a reasonable approximation of the system. In our case, the control system will be designed with the plant model developed from Bond-graph modelling approach.

The second level of compaction is characterized by a fast dynamics. It is therefore useful to take advantage of this to obtain a more accurate compaction ratio. In fact, since, there is over-compaction in the first stage of compaction, we use a negative compaction at this stage to obtain a more accurate result.

When a fabric under process leaves the Sanforized compaction zone, about to be introduced to the Felt compaction zone, the fabric is subjected to a slight tension which is measured by *Load Cell*. This tension results from a difference in speed between the line (master) speed motor of the Felt Compactor and the command (slave) speed motor of the

Sanforized compactor. This slight tension contributes a negative compaction which has been referred here as the Stage II Compaction.

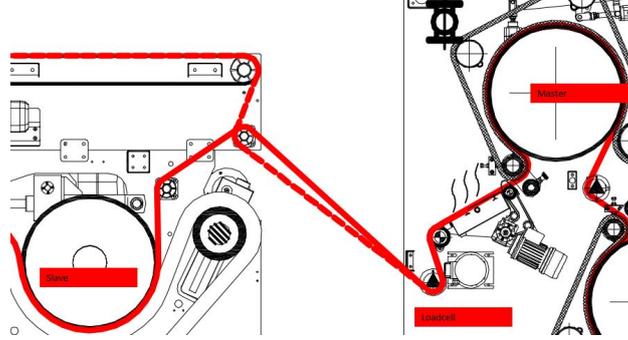


Figure 3.4. Stage II Compaction Zone

3.4.1 Subsystem Description

This subsystem is defined mainly by two (2) motorized cylinders (master and slave), the load cell and the fabric material as demonstrated in the figure 3.4. It is desired to obtain a constant tension output computed from the difference in the command speed and the line speed inputs. It will be seen that this requirement will be satisfied by the *common effort junction* modelling strategy.

The electric motor parameters of the subsystem will not be considered. Instead, only the parameters of the fabric and the cylinders will be highlighted. As discussed earlier, based on the mechanical properties of fabric, it can be modelled as a spring with internal damper.

All the parameters are the same for both stages I and II compaction with only a variation in the force constant parameter as expected.

Parameter	Unit	Value
Force Constant, (k)	N/m	1.5849e+03

Table 3.2. Subsystem Parameters and Corresponding Values

The Input / Output Variables of stage II subsystem include;

- *Line Speed, (ω_l)*: This is an input to the master speed motor and it is the basis for obtaining a command speed. It has been modelled here as an input disturbance to the Stage II compaction subsystem since it's change is desired to be attenuated.
- *Command Speed, (ω_c)*: This is the manipulated variable input to the plant and the adaptive output of the controller given to the plant in order to track the reference variable.

- *Tension, (F_1):* This is a plant output that is obtained on the basis of the difference in the line speed and the command speed. It is modelled to be maintained constant along the fabric using 0-junction bond-graph approach.
- *Compaction:* This is the controlled output of the feedback system. It shall be modelled as the displacement in the fabric length. It is desired that the reference signal track this value.

3.4.2 Stage II Plant Modelling By Set-Membership System Identification Approach

Set Member Identification of Multiple Inputs - Single Output (MISO) Discrete Time LTI System: The theoretical analysis of Set Membership approach of performed in chapter 2.8 has only covered SISO systems. Here, the approach will be extended for MISO system considerations.

Since there are three inputs for the Stage II compaction, it is necessary to model it as a MISO system. It will be seen that MISO systems are subsets of MIMO systems.

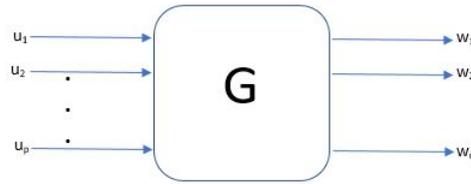


Figure 3.5. MIMO Transfer Function Representation

$G(q^{-1})^5$ is a matrix whose entries are transfer functions. Hence, considering a given output i ,

$$w_i(t) = G_{i1}(q^{-1})u_1(t) + G_{i2}(q^{-1})u_2(t) + \dots + G_{ip}(q^{-1})u_p(t) \quad (3.9)$$

This means that MIMO can be addressed as series of MISO systems.

For a single output system,

$$w(t) = [G_1(q^{-1}) + G_{i2}(q^{-1}) + \dots + G_{ip}(q^{-1})]u(t) \quad (3.10)$$

where,

$$u(t) = [u_1(t), u_2(t), \dots, u_p(t)]'$$

⁵ q^{-1} is known as the backward shift

A-Priori Information:

- G_1 relates u_1 to output
- G_2 relates u_2 to output and so on
- An idea of the order of the system is necessary

Considering a MISO system, the optimization problem is a polynomial problem of constraints $p+1$ for all the inputs summed to p considered separately.

Slack Variable Approach to Identification of MISO LTI Discrete Time Model

With this approach, the problem of solving $p+1$ optimization constraints is eliminated.

$$w(t) = G_1(q^{-1})u_1(t) + G_2(q^{-1})u_2(t) + \dots + G_p(q^{-1})u_p(t) \quad (3.11)$$

$$w(t) = z_1(t) + z_2(t) + \dots + z_p(t) \quad (3.12)$$

where z_i is the partial output (slack variable) for each input

$$z_1(t) = G_1(q^{-1})u_1(t)$$

$$z_2(t) = G_2(q^{-1})u_2(t)$$

$$z_3(t) = G_3(q^{-1})u_3(t)$$

Hence,

Model Function, F

$$w(t) = z_1(t) + z_2(t) + \dots + z_p(t)$$

$$z_i(t) = G_i(q^{-1})u_i(t)$$

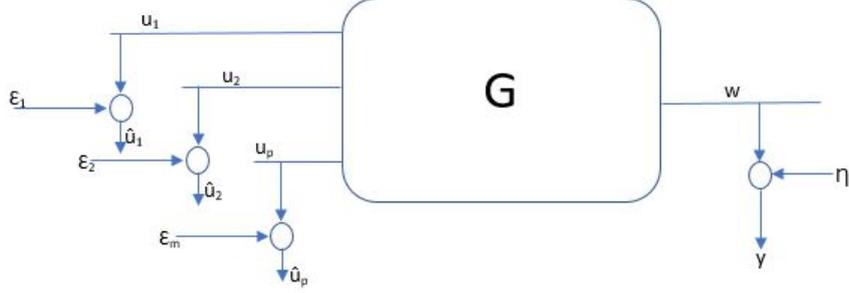
Noise


Figure 3.6. MISO Transfer Function Representation

Error-In-Variable Structure

Bound Error assumption Both input and output errors are unknown but bounded

$$|\eta(t)| \leq \Delta\eta \forall t, |\epsilon(t)| \leq \Delta\epsilon \forall t \quad (3.13)$$

The set of all the feasible parameters is given by the Feasible Parameter Set (FPS) and the correspond solutions are given in Feasible Solution Set (FSS)

Feasibility Parameter Set (FPS)

For a given partial output,

$$z_i(t) = \frac{\beta_0^i + \beta_1^i q^{-1} + \dots + \beta_m^i q^{-m}}{1 + \alpha_1^i q^{-1} + \dots + \alpha_m^i q^{-m}} \quad (3.14)$$

$$\forall i = 1, 2, 3, \dots, p^6$$

$$\begin{aligned} D_\theta &= \{\theta \in R^{(2m+1)p} : w(t) = z_1(t) + z_2(t) + \dots + z_p(t), \\ z_i(t) &= -\alpha_1^i z_i(t-1) \dots - \alpha_m^i z_i(t-m) + \beta_0^i [\hat{u}_i(t) - \hat{\epsilon}(t)] + \dots + \beta_m^i [\hat{u}_i(t-m) - \hat{\epsilon}(t-m)], \\ &\quad \forall i = 1, 2, \dots, p \\ &\quad \forall t = m+1, m+2, \dots, N \\ &\quad u_i(t) = \hat{u}_i(t) - \epsilon_i(t) \\ &\quad w(t) = y(t) - \eta(t) \\ &\quad \forall t = 1, 2, \dots, N\} \end{aligned}$$

⁶\$p\$ represents the number of inputs and \$m\$ is the order of the system

Now, since our model considers three (3) inputs, the MISO reduces to Three Inputs Single Output system,
Assuming a first (1st) order system,

Also, since equation defining the FPS depends also on unknown but bounded samples of noise sequences, we need to define the Extended Feasibility Parameter Set (EFPS)

Extended Feasibility Parameter Set (EFPS)

$$D_{\theta,\eta,\epsilon} = \{\theta \in R^{(2*1+1)3}, \eta \in R^N, z \in R^N : \quad (3.15)$$

$$w(t) = z_1(t) + z_2(t) + z_3(t) \quad (3.16)$$

$$y(t) = w(t) + \eta(t) \quad (3.17)$$

$$z_1(t) = -\alpha_1^1 z_1(t-1) + \beta_0^1 [u_1(t) - \epsilon_1(t)] + \beta_1^1 [u_1(t-1) - \epsilon_1(t-1)] \quad (3.18)$$

$$z_2(t) = -\alpha_1^2 z_2(t-1) + \beta_0^2 [u_2(t) - \epsilon_2(t)] + \beta_1^2 [u_2(t-1) - \epsilon_2(t-1)] \quad (3.19)$$

$$z_3(t) = -\alpha_1^3 z_3(t-1) + \beta_0^3 [u_3(t) - \epsilon_3(t)] + \beta_1^3 [u_3(t-1) - \epsilon_3(t-1)] \quad (3.20)$$

where $\theta \in [\alpha_i^j, \beta_i^j]$

Parameter Uncertainty Interval (PUI)

Then we define the Parameter Uncertainty Interval (PUI) as follows:

$$PUI_{\theta_i} = \left[\theta_i, \bar{\theta}_i \right] \quad (3.21)$$

$$\theta_i = \min_{\theta,\eta,\epsilon \in D_{\theta,\eta,\epsilon}} \theta_i ; \quad \bar{\theta}_i = \max_{\theta,\eta,\epsilon \in D_{\theta,\eta,\epsilon}} \theta_i$$

Nominal Parameter:

$$\theta_i^c = \frac{\theta_i + \bar{\theta}_i}{2} \quad (3.22)$$

Objective function

$$\theta_i = \min \alpha_i^j \quad (3.23)$$

$$\bar{\theta}_i = \max \alpha_i^j = -\min \alpha_i^j \quad (3.24)$$

subject to some constraints:

Equality Constraints

Assume $\eta_1 = \eta_2 = \eta_3$

Equality Constraint 1

$$z_1(t) + \alpha_1^1 z_1(t-1) - \beta_0^1 [u_1(t) - \epsilon_1(t)] - \beta_1^1 [u_1(t-1) - \epsilon_1(t-1)] = 0 \quad (3.25)$$

Equality Constraint 2

$$z_2(t) + \alpha_1^2 z_2(t-1) - \beta_0^2 [u_2(t) - \epsilon_2(t)] - \beta_1^2 [u_2(t-1) - \epsilon_2(t-1)] = 0 \quad (3.26)$$

Equality Constraint 3

$$z_3(t) + \alpha_1^3 z_3(t-1) - \beta_0^3 [u_3(t) - \epsilon_3(t)] - \beta_1^3 [u_3(t-1) - \epsilon_3(t-1)] = 0 \quad (3.27)$$

Equality Constraint 4

$$y(t) = \eta(t) - z_1(t) - z_2(t) - z_3(t) = 0 \quad (3.28)$$

Minimum and maximum bounds on error

$$\begin{aligned} \eta(t) &\leq \Delta\eta; \quad \eta(t) \geq -\Delta\eta; \\ \epsilon &\leq \Delta\epsilon; \quad \epsilon \geq \Delta - \epsilon; \\ &\forall t = m+1, m+2, \dots, N \end{aligned}$$

This leads to a Bilinear (Polynomial) optimization problem. Such a problem is non-convex and the global minimum for such as a non-linear problem can be computed by convex relaxation. This can be implemented using SparsePOP software.[14]

Solving the problem with SparsePOP involves building support matrices from the equality and non-equality constraints, obtaining an array of coefficients of the variables being optimized, and choosing the relaxation order.

Support Matrix and Coefficient Array for Objective Function

Considering parameter, α_1^1 ,

Objective Function:

$$\alpha_1 = \min_{\theta, \eta, \epsilon \in D_{\theta, \eta, \epsilon}} \alpha_1$$

Support Matrix: [1, 0, 0, ..., 0] (1x (N+(number of parameters to be estimated=9)))

Array of Coefficients:

$$[1] \text{ for min } \alpha_1 \text{ and} \quad (3.29)$$

$$[-1] \text{ for max } \alpha_1 \quad (3.30)$$

Support Matrix and Coefficient Array for Bilinear Equality Constraints

Using the equality constraint equations, the equality matrices are developed thus;

Support Matrices:

Equality Constraint 1:

$$z_1(t) + \alpha_1^1 z_1(t - 1) - \beta_0^1 [u_1(t) - \epsilon_1(t)] - \beta_1^1 [u_1(t - 1) - \epsilon_1(t - 1)] = 0$$

Array of Equality Coefficient 1:

$$[1, 1, -u_1(t), -u_1(t - 1), 1, 1]$$

For t =2;

	α_1^1	β_0^1	β_1^1	α_1^2	β_0^2	β_1^2	α_1^3	β_0^3	β_1^3	$z_1(t)$	$z_1(t-1)$...	$z_1(N)$	$z_2(t)$	$z_2(t-1)$...	$z_2(N)$	$z_3(t)$	$z_3(t-1)$...	$z_3(N)$	$\epsilon(t)$	$\epsilon(t-1)$...	$\epsilon(N)$	$\eta(t)$...	$\eta(N)$
1	0	0	0	0	0	0	0	0	0	1	0	...	0	0	0	...	0	0	0	...	0	0	0	...	0	0	...	0
1	1	0	0	0	0	0	0	0	0	0	1	...	0	0	0	...	0	0	0	...	0	0	0	...	0	0	...	0
-u ₁ (t)	0	1	0	0	0	0	0	0	0	0	0	...	0	0	0	...	0	0	0	...	0	0	0	...	0	0	...	0
-u ₁ (t-1)	0	0	1	0	0	0	0	0	0	0	0	...	0	0	0	...	0	0	0	...	0	0	0	...	0	0	...	0
1	0	1	0	0	0	0	0	0	0	0	0	...	0	0	0	...	0	0	0	...	0	1	0	...	0	0	...	0
1	0	0	1	0	0	0	0	0	0	0	0	...	0	0	0	...	0	0	0	...	0	0	1	...	0	0	...	0

Figure 3.7. First Support Matrix

Equality Constraint 2:

$$z_2(t) + \alpha_1^2 z_2(t - 1) - \beta_0^2 [u_2(t) - \epsilon_2(t)] - \beta_1^2 [u_2(t - 1) - \epsilon_2(t - 1)] = 0$$

Array of Equality Coefficient 2:

$$[1, 1, -u_2(t), -u_2(t - 1), 1, 1]$$

	α_1^1	β_0^1	β_1^1	α_1^2	β_0^2	β_1^2	α_1^3	β_0^3	β_1^3	$z_1(t)$	$z_1(t-1)$...	$z_1(N)$	$z_2(t)$	$z_2(t-1)$...	$z_2(N)$	$z_3(t)$	$z_3(t-1)$...	$z_3(N)$	$\epsilon(t)$	$\epsilon(t-1)$...	$\epsilon(N)$	$\eta(t)$...	$\eta(N)$
1	0	0	0	0	0	0	0	0	0	0	0	...	0	1	0	...	0	0	0	...	0	0	0	...	0	0	...	0
1	0	0	0	1	0	0	0	0	0	0	0	...	0	0	1	...	0	0	0	...	0	0	0	...	0	0	...	0
-u ₂ (t)	0	0	0	0	1	0	0	0	0	0	0	...	0	0	0	...	0	0	0	...	0	0	0	...	0	0	...	0
-u ₂ (t-1)	0	0	0	0	0	1	0	0	0	0	0	...	0	0	0	...	0	0	0	...	0	0	0	...	0	0	...	0
1	0	0	0	0	1	0	0	0	0	0	0	...	0	0	0	...	0	0	0	...	0	1	0	...	0	0	...	0
1	0	0	0	0	0	1	0	0	0	0	0	...	0	0	0	...	0	0	0	...	0	0	1	...	0	0	...	0

Figure 3.8. Second Support Matrix

Equality Constraint 3:

$$z_3(t) + \alpha_1^3 z_3(t-1) - \beta_0^3 [u_3(t) - \epsilon_3(t)] - \beta_1^3 [u_3(t-1) - \epsilon_3(t-1)] = 0$$

Array of Equality Coefficient 3:

$$[1, 1, -u_3(t), -u_3(t-1), 1, 1]$$

	α_1^1	β_0^1	β_1^1	α_1^2	β_0^2	β_1^2	α_1^3	β_0^3	β_1^3	$z_1(t)$	$z_1(t-1)$...	$z_1(N)$	$z_2(t)$	$z_2(t-1)$...	$z_2(N)$	$z_3(t)$	$z_3(t-1)$...	$z_3(N)$	$\epsilon(t)$	$\epsilon(t-1)$...	$\epsilon(N)$	$\eta(t)$...	$\eta(t)$
1	0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	...	0	1	0	...	0	0	0	...	0	0	...	0
1	0	0	0	0	0	0	1	0	0	0	0	...	0	0	0	...	0	0	1	...	0	0	0	...	0	0	...	0
$-u_2(t)$	0	0	0	0	0	0	0	1	0	0	0	...	0	0	0	...	0	0	0	...	0	0	0	...	0	0	...	0
$-u_2(t-1)$	0	0	0	0	0	0	0	0	1	0	0	...	0	0	0	...	0	0	0	...	0	0	0	...	0	0	...	0
1	0	0	0	0	0	0	0	1	0	0	0	...	0	0	0	...	0	0	0	...	0	1	0	...	0	0	...	0
1	0	0	0	0	0	0	0	0	1	0	0	...	0	0	0	...	0	0	0	...	0	0	1	...	0	0	...	0

Figure 3.9. Third Support Matrix

Equality Constraint 4:

$$y(t) = \eta(t) - z_1(t) - z_2(t) - z_3(t) = 0$$

Array of Equality Coefficient 3:

$$[y(t), -1, -1, -1 - 1]$$

	α_1^1	β_0^1	β_1^1	α_1^2	β_0^2	β_1^2	α_1^3	β_0^3	β_1^3	$z_1(t)$	$z_1(t-1)$...	$z_1(N)$	$z_2(t)$	$z_2(t-1)$...	$z_2(N)$	$z_3(t)$	$z_3(t-1)$...	$z_3(N)$	$\epsilon(t)$	$\epsilon(t-1)$...	$\epsilon(N)$	$\eta(t)$...	$\eta(t)$
$y(t)$	0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	...	0	0	0	...	0	0	0	...	0	0	...	0
-1	0	0	0	0	0	0	0	0	0	1	0	...	0	0	0	...	0	0	0	...	0	0	0	...	0	0	...	0
-1	0	0	0	0	0	0	0	0	0	0	0	...	0	1	0	...	0	0	0	...	0	0	0	...	0	0	...	0
-1	0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	...	0	1	0	...	0	0	0	...	0	0	...	0
-1	0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	...	0	0	0	...	0	0	0	...	0	1	...	0

Figure 3.10. Fourth Support Matrix

relaxation Order

Since the highest order of the Polynomial constraints is two(2), the minimum relaxation order will be;

$$\delta_{min} = m/2 \tag{3.31}$$

$$2/2 = 1$$

3.4.3 Stage II Plant Modelling With Bond-Graph

In the design of controllers for a specific plants, mathematical models which represent the plants are used. Any model can at best be the approximation of the true plant. In our case, the state equations of the plant will be extracted using bond-graph approach.

The model scheme of stage II compaction is shown in figure 3.11

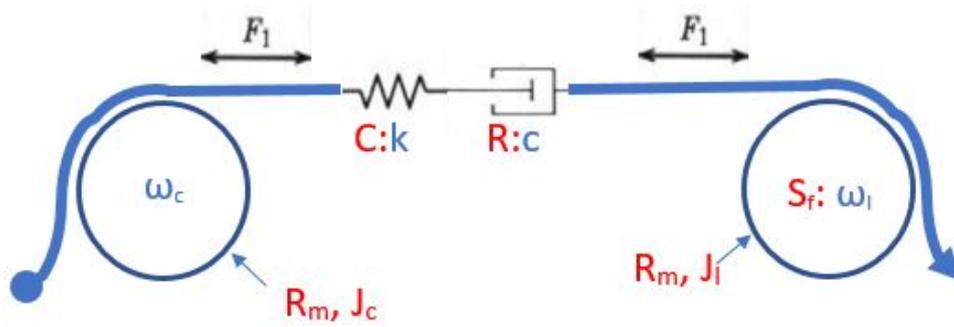


Figure 3.11. Stage II Compaction Scheme, ω_c = command speed, ω_l = Line Speed

Since the tension on the fabric results from the change in speed between the line speed and the command speed, a common effort junction (0-junction) is used to model this change in speed as well as the constant tension as desired.

Based on the analysis done in section 2.9, the following bond-graph is obtained,

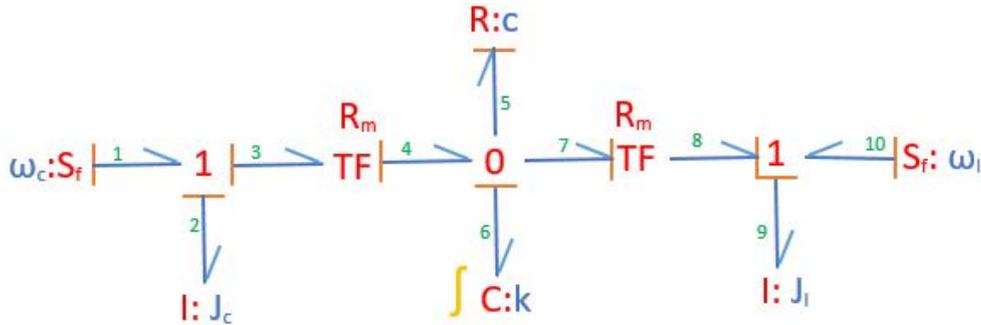


Figure 3.12. Stage II Compaction Bond-Graph Realization

The first common flow junction (1-Junction) from the left indicates the common speed that exists between the slave speed motor the mass inertia of the steam cylinder. This speed is imposed by the slave speed motor. The angular variables (Torque and angular speed) of this motor are transformed into linear variables (Force and linear speed) using the transfer function (TF) and the linear variable with the damper and the spring are subjected to a common effort indicated by common effort junction (0-Junction). In other

words, the 0-junction is used for summing the difference in speed imposed by slave and the master motors. The linear variables are converted again into angular variables and are subjected to a common flow with the mass inertia of the Felt cylinder. This is common flow is imposed by the master speed motor. As in stage I, k and c model spring and damping properties of the fabric respectively.

3.4.4 Derivation of State Equation:

From the above analysis, it can be seen that only one state variable is possible and hence, a single state equation. The state variable is associate to the spring and it is the spring displacement, q_6 .

$$\begin{aligned} \dot{q}_6 &= f_4 - f_5 - f_7 \\ \dot{q}_6 &= f_3 R_m - f_5 - f_8 R_m \\ \dot{q}_6 &= \omega_c R_m - e_5/c - \omega_l R_m \\ \dot{q}_6 &= \omega_c R_m - kq_6/c - \omega_l R_m \end{aligned} \tag{3.32}$$

$$\dot{q}_6 = -kq_6/c + R_m(\omega_c - \omega_l) \tag{3.33}$$

$$y = e_1 = P_2/J_c \tag{3.34}$$

Equation 3.33 is the state equation while equation 3.34 is the output. From the state equation, the steady state value of q_6 can be computed.

$$q_6(s + k/c) = R_m(\omega_c - \omega_l) \tag{3.35}$$

$$q_6 = \lim_{s \rightarrow 0} R_m(\omega_c - \omega_l)c/k \tag{3.36}$$

3.5 Control System Design

A feedback control system is necessary to take into account the uncertainty of the plant and to ensure that performance specifications are met within guaranteed limit. Feedback systems also have the capacity of reducing the effect of disturbances and compensate for plant parameters uncertainty and model mismatch.

The representation of a feedback control system is shown in figure 3.13

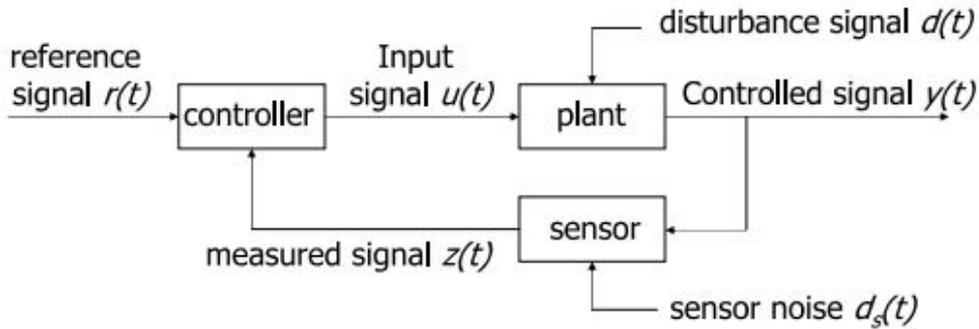


Figure 3.13. Feedback Control System Layout

Based on the layout of figure 3.13, our feedback control system has the following components:

- **The Plant** represents the subsystem to be controlled.
- **The Sensor** gives information about the plant. Here, the pick-counter is used as the output sensor for compaction value. Also, values read from PLC gives information of other plant variable such as command speed, line speed and so on.
- **The Controller** Compares measured values to their desired values and provides input variable to the plant.

The main signals of interest include:

- **Reference Signal, $r(t)$** gives prescribed values of the controlled signal.
- **Input Signal, $u(t)$** influences the plant and it is a manipulatable variable. It is provided by the controller.
- **Controlled Signal, $y(t)$** is the signal under control
- **Disturbance Signal, $d_p(t)$** influences the plant but cannot be manipulated.
- **Sensor noise, $d_s(t)$** affects the measured signal.
- **Measured signal, $z(t)$** is measured by means of sensor and contains information about the state of the plant.

The closed-loop controller provides signal to the plant on the basis of past and present values of both the reference signal and the measured signal.

3.6 Adaptive MPC Controller Design:

Since the subsystem are subject to parameter variation during operation, it is necessary to adapt to this variations online and this motivates the use of Adaptive Model Predictive Control (MPC) for the control design.

The Adaptive MPC is initialized with an MPC object which is a controller that has been designed at a given operating point for a linearised plant model. In this case, the plant has been linearised at steady state operating point. This linearised dynamic plant model is internal to MPC object and it is needed for on-line forecast within a given moving time horizon and it runs in parallel with the given non-linear plant model. In adaptive mode and at runtime, the state of the internal plant is updated on-line with the state variable estimated from the inputs and output variable of the non-linear plant model.

The given plant state space equation is as given in equations 3.33 and 3.34.

3.6.1 Creating Operating Point

The MPC object will be controlling a plant that has been linearised at given operating point. Using *trim output constraint* of Simulink, an output signal variable is defined for steady state operating point computation.

The MPC tool is incorporated with functionalities that determines the steady state operating point of a system whose known outputs variables at steady state are defined with the trim function. Trim function uses constraint optimisation technique and attempts to find values for X, U and Y that set the state derivatives, DX, of the S-function 'SYS' to zero (0).

It is desired that these known output variables track the reference variables of the feedback control system. Every other input or output variable is assumed to to be unknown with this strategy.

Stage I Subsystem Steady State Operating Point The Simulink representation of Stage I are shown in figures 3.14

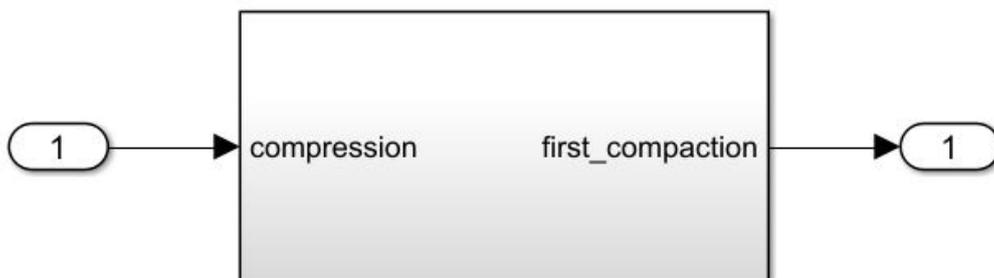


Figure 3.14. Representation of Stage I Compaction

The input and output variable values for this subsystem is described in table 3.3

Signal Variable	Variable Type	Value
Manipulated Variable (Compression)	input	unknown
Controlled Variable (first_compaction)	output	0.22
State 1	state	0
State 2	state	0

Table 3.3. Input and Output Variable and values of Stage I prior to Steady State Operating Point Search

With the manipulated variable signal connected to the output of the controller and the controlled output variable signal connected to the measured output port of the MPC controller, an operating point search is performed and the report for *Stage I Plant* is shown in 3.15 It can be seen from figure, 3.15 that as desired, the change in the state (DX) variable

State	Desired Value	Actual Value	Desired dx	Actual dx
Stage_I_Plant/CompressionPlant/Integrator1				
State - 1	[-Inf , Inf]	-4.3493e-17	0	2.2737e-13
State - 2	[-Inf , Inf]	0.22	0	-1.0827e-19
Stage_I_Plant/MPC Controller/MPC/last_mv				
State - 1	[-Inf , Inf]	200.2291	0	0

Figure 3.15. Steady State Operating Point Search Result

at steady state tends to zero (0).

Stage II Subsystem Steady State Operating Point: The Simulink representation of Stage II subsystem is shown in figures 3.16.

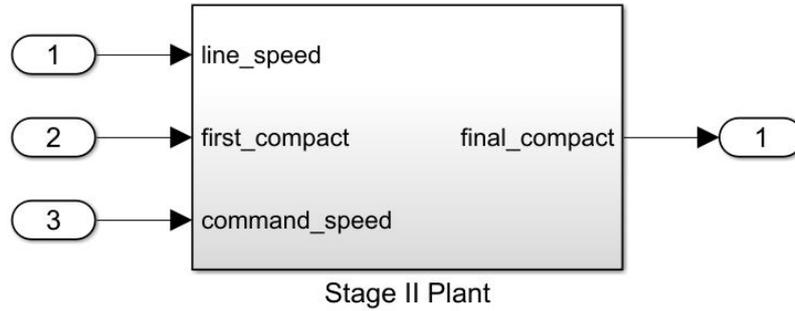


Figure 3.16. Representation of Stage II Compaction

The input and output variable values for this subsystem is described in table 3.4

Signal Variable	Variable Type	Value
Manipulated Variable (command_speed)	input	unknown
Unmeasured Disturbance Variable 1 (line_speed)	input	10
Unmeasured Disturbance Variable 2 (first_compact)	input	0.22
Controlled Variable (final_compact)	output	0.2
State 1	state	0

Table 3.4. Input and Output variables and values of Stage II prior to Steady State Operating Point Search

With the manipulated variable signal connected to the output of the controller and the controlled output variable signal connected to the measured output port of the MPC controller, an operating point search is performed as in stage I and the report for *Stage II Plant* is shown in 3.17.

Optimizer Output : Details				
State	Input	Output		
State	Desired Value	Actual Value	Desired dx	Actual dx
Stage_II_Plant/SystemPlant/MPC Controller/MPC/last_mv				
State - 1	[-Inf , Inf]	10.5944	0	0
Stage_II_Plant/SystemPlant/Stage II Plant/Integrator				
State - 1	[-Inf , Inf]	0.01	0	-9.2148e-09
Output		Desired Value	Actual Value	
Stage_II_Plant/SystemPlant/Stage II Plant				
Output - 1		0.2		0.2

Figure 3.17. Steady State Operating Point Search result for Stage II

Also, it can be seen from figure, 3.17 that as desired, the change in the state variable at steady state tends to zero (0).

3.6.2 MPC Internal Dynamic Plant Model (Prediction Model) Creation

This operating point object is then used to initialise the non-linear plant model and the plant model state is linearised at this initial condition.

The linearised plant is a continuous time plant system. Since the MPC can only work with discrete time plant, the linear plant is converted to discrete time plant system with a *sampling time of 0.1*. The linearised model for stages I and II shown in the listing below 3.6.2 and 3.6.2 respectively.

```

1 % The resulting state space of stage I linearized plant
2
3 A =
4 Integrator1(  Integrator1(
5 Integrator1(      0.6496      -187.1
6 Integrator1(      0.0002034      0.975
7
8 B =
9 compression
10 Integrator1(      0.2056
11 Integrator1(      2.742e-05
12
13 C =
14 Integrator1(  Integrator1(
15 first compac      0      1
16
17 D =
18 compression
19 first compac      0
20
21 Input groups:
22 Name      Channels
23 Manipulated      1
24
25 Output groups:
26 Name      Channels
27 Measured      1
28
29 Sample time: 0.1 seconds
30 Discrete-time state-space model.

```

```

1 % The resulting state space of stage II linearized plant
2
3 A =
4 Integrator
5 Integrator      0.9057
6
7 B =
8 mv      Mux2(1)      Mux2(2)
9 Integrator      0.001587      -0.001587      0
10

```

```

11 C =
12 Integrator
13 ov          -1
14
15 D =
16 mv  Mux2(1)  Mux2(2)
17 ov          0          0          1
18
19 Input groups:
20 Name          Channels
21 Manipulated    1
22 Unmeasured     2,3
23
24 Output groups:
25 Name          Channels
26 Measured       1
27
28 Sample time: 0.1 seconds
29 Discrete-time state-space model.

```

3.6.3 MPC Object Design

Given the discrete time dynamic plant model, future reference vector, and with the input and output constraints defined, an on-line forecast is performed to foresee any possible constraint violation over the finite time window. At each sampling time, a constrained optimization problem is formulated over a finite window of time and solved starting from the current instance through a moving horizon that has been defined in equation 2.68.

3.6.4 Implemented Constraints on Manipulated Inputs and Outputs

The time series optimized input variables is as follows;

$$\{u(k|k), u(k+1|k), \dots, u(k+q-1|k)\}$$

The dimension of on-line optimization is reduced by restricting the *control horizon*, q to it's first set of moves.

$$u(k+q|k) = u(k+q+1|k) = \dots = u(k+p-1|k) = u(k+q-1|k)$$

- **Control Horizon**, $q = 3$ (chosen)
- **Prediction Horizon**, $p = 15$ (chosen)
- **Bounds on Manipulated Inputs:**
 - Stage I
 - Lower limit, u_{cp}^L of the compression = 0 [kg]
 - Upper limit, u_{cp}^H of the compression = 500 [kg]

Stage II

Lower limit, u_{cs}^L of the command speed = 5 [meter/min]

Upper limit, u_{cs}^H of the command speed = 50 [meter/min]

Lower limit, u_{ls}^L of the line speed = 5 [meter/min]

Upper limit, u_{ls}^H of the line speed = 50 [meter/min]

Lower limit, u_{fc}^L of the first compaction = 0

Upper limit, u_{fc}^H of the first compaction = 1

- **Bounds on Outputs:**

Lower limit, y^L of the compaction variable = 0

Upper limit, y^H of the compaction variable = 1

Output constraints are useful for zone variable prediction when the number of outputs is greater than the number of inputs.

3.6.5 Future Set-point Trajectory Determination

At every sampling time, the optimization problem solved is used to predicting the future trajectory with the prediction horizon. The trajectory targets the set-points at steady states. An intuitive simplification of the method used in subsection 2.10.3 for computation of the future set-point trajectory is obtainable through reference signal previewing.

For the reference signal vector of dimension (N-by- n_y), our $n_y = 1$ since we have only a single reference signal in both stages I and II compaction. Also, we have maintained the reference matrix to a single row (N=1) vector just for simplicity. This means that, since $N < p$, this only row designates constant reference values for the remaining future $p - 1$ steps.

3.6.6 Constrained MPC Formulation

Given the discrete time prediction model, input and output constraints, and desired set-point trajectory, the MPC optimization problem is formulated according to section 2.10.4.

As pointed out, the optimization is a minimization problem, subject to

- Prediction model
- bounds on future inputs and predicted outputs

error weighting matrix, W_E must be Symmetric positive definite

input weighting matrix, $W_{\Delta u}$ must be Symmetric positive semi-definite

The Constrained MPC Formulation problem is transformed into **Quadratic Programming (QP)** problem through series of algebraic manipulations and solved very quickly using an efficient search method.

The MPC quadratic programming (QP) problem must have a unique solution. To achieve this, cost function parameters (penalty weights) and horizons were chosen such that the QP's Hessian matrix is positive-definite.

- **Penalty Weights On Manipulated Variable Rates** A positive-definite Hessian was guaranteed by increasing penalties on manipulated variables' rate of change (MVrate). Table 3.5 shows the design values of **Weights.MVrate** for both stages I and II

<i>Manipulated Variable</i>	Compaction Stage	Value
compression	I	0.0960789
command_speed	II	0.4392945

Table 3.5. Design values of **Weights.MVrate** for Stages I and II

The value of **Weights.MVrate** also influences the response speed of the control system.

- **Penalty Weights On Output Variables:**

The penalty weighting function on the output variables **Weights.OV** also affect the Hessian. The values have been chosen according to table 3.6 to emphasize the importance of the output variable tracking and to increase the chances of obtaining a unique QP solution.

<i>Output Variable</i>	Compaction Stage	Value
first compaction	I	0.22764
final compaction	II	1.04081

Table 3.6. Design values of **Weights.OV** for Stages I and II

- **Penalty Weights On Input Variables:**

The weighting function on manipulated variables **Weights.MV** have been retained at zero (0) for the both stages.

- **Weight on overall constraint softening**

If there are soft constraints, increasing the overall constraint softening weighting function, **Weights.ECR**⁷ value makes these constraints relatively harder. The controller then places a higher priority on minimizing the magnitude of the predicted worst-case constraint violation.

Soft constraints have been implemented on the output variable's lower and upper bounds.

The assigned weight on the overall constraint softener **Weights.ECR** = 100000

⁷ECR means Equal Concern for Relaxation

Tuning Weights.ECR may be helpful to achieve the proper balance in relation to the other control objectives.

3.7 Controller Performance Tuning

Response Speed: Though Increasing the weighting function on the rate of change of manipulated variable (MVrate) may guarantee positive definite QP's Hessian matrix, it can consequently lead to a more sluggish controller response. Maintaining MVrate at the design values of table 3.5 returns the response speed while retaining a positive definite QP's Hessian matrix.

Minimum error tracking: Non-zero of Weights.OV values emphasize the importance of OV target tracking. This is helpful in maintaining the output error very minimum.

Horizon Parameters For Plants With Delay: If a plant includes delay T , the prediction and control horizons (P and M) must be set in such a way that,

$$P - M > T. \tag{3.37}$$

This ensures that the last MV movement in the optimal control sequence which occurs at step M , affect the OVs within the prediction horizon. Otherwise, the last MV movement is irrelevant to OV control and the QP might not have a unique solution.

In our case however, the maximum total delay in the prediction model is $T = 0$ sampling periods, and $P(+1) - M = 13$.

It is recommended to use prediction horizon of close to the settling time. However, for computational efficiency, we have chosen a lower prediction horizon of 15 sample steps.

Scale Factors Scaling converts the relationship between output variables and manipulated variables to dimensionless form. Scale factor specifications can improve QP numerical accuracy. They also make it easier to specify tuning weight magnitudes.

In order for the outputs to be controllable, each must respond to at least one manipulated variable within the prediction horizon. If the plant is well scaled, the maximum absolute value of such responses should be of order unity.

The scaling factor on the manipulated variables and output variables have been set as described in table 3.7

<i>Variable</i>	Compaction Stage	Scale Factor
compression	I	500
command speed	II	10
first compaction	I	1
final compaction	II	0.2

Table 3.7. Design Scale Factors for Stages I and II

3.7.1 Hard MV Constraints

The controller should always satisfy hard bounds on a manipulated variable OR its rate-of-change. Specifying both constraint types simultaneously, however, may result in a conflict during real-time use. For this reason, only hard bounds on the manipulated variable (command speed) has been used the design.

$$\text{Lower limit, } u^L \text{ of the command speed} = 5[\text{meter}/\text{min}]$$

$$\text{Upper limit, } u^H \text{ of the command speed} = 50[\text{meter}/\text{min}]$$

3.7.2 Soft Constraints

If a constraint is too soft, an unacceptable violation may occur. If it is too hard, the controller might pay it too much attention. Moreover, making a constraint harder cannot prevent a violation if the constraint is fundamentally infeasible.

Two soft constraints have been implemented for each of the stages and they include;

- Soft constraint on the lower limit of the output variable,
OV.MinECR = 1;
- Soft constraint on the upper limit of the output variable,
OV.MaxECR = 1;

3.7.3 Adaptive Model Predictive Controller

Based on the operation modes on the Sanforized Compactor, it is anticipated that the system is subject to continuous change in its parameters. For this reason, it is necessary to adapt to these changes on-line. This is achieved by adapting the Model Predictive Controller that has been designed in section 3.6.3 using the MPC Tool box of Matlab with the changes in these parameter.

The adaptive MPC is designed on the basis of MPC object structure. Using an estimator (external to the MPC object) that has been initialized with the state of the MPC object internal plant, new state of the plant is obtained at runtime and used to update the state of the MPC internal plant. Then, future prediction is optimized on the basis of the new state.

External Model Estimator

Many options exist for external on-line estimation of the plant model. In our case, a Recursive Polynomial Model Estimator block of Matlab has been adopted to estimate an ARX model structure.

$$ARX \rightarrow A(q)y(t) = B(q)u(t - nk) + e(t) \tag{3.38}$$

q is the time-shift operator and nk is the delay.

$u(t)$ is the input, $y(t)$ is the output, and $e(t)$ is the error. For MISO models as applicable to in this case, there are as many $B(q)$ polynomials as the number of inputs. The orders of this model is as follows:

$$na \rightarrow 1 + a_1q^{-1} \quad nb \rightarrow b_1 + b_2q^{-1}$$

The orders na , nb and delay, nk , are known a priori and fixed. They corresponds to the number of states of the MPC internal plant. They are provided as the initial estimate and structure for the estimation process

It is a good practice to use the MPC internal plant state parameters as the initial estimate. This is readily available through *tfdata()* of Matlab.

Using the input $u(t)$ and output $y(t)$ as input to the estimator, the parameters $A(q)$, $B(q)$ are estimated and output as bus signal with the following elements:

```

1  %Obtaining the Polynomial Coefficients of Numerators (B) and Denominators (A) of
2
3  % Stage I transfer function data
4  [num_I, den_I] = tfdata(mpc_I_obj.Model.Plant);
5  Aq_I = den_I{1};
6  Bq_I = num_I{1};
7
8  % Stage II transfer function data
9  [num_II, den_II] = tfdata(mpc_II_obj.model.Plant);
10 Aq_II = den_II{1};
11 Bq_II = num_II{1};

```

Stage I	
$A_{q_I} \rightarrow$	[1.0000 -1.6247 0.6715]
$B_{q_I} \rightarrow$	[0 0.2742 0.2401]
Stage II	
$A_{q_II} \rightarrow$	[1.0000 -0.9057]
$B_{q_II} \rightarrow$	[0 -0.0016]

Table 3.8. Polynomial Coefficients of Numerators (B) and Denominators (A)

The number of values in each polynomial coefficient vector indicates the number of states (+ 1) of the internal plant.

For MISO data, A_q and B_q are matrices where the i -th row parameters correspond to the i -th input.

Stage I $\rightarrow 1 \times 1$ cell array

Stage II $\rightarrow 1 \times 3$ cell array

These parameter signals are converted from ARX form into state space form using *Model Type Converter* of Simulink and are collected through a *Bus Selector* into *Signal Specification* blocks. Signal specification is necessary to maintain consistency with the parameter specification of the MPC internal dynamic plant model. The dimension specification is as follows;

Stage I

state space parameters

$\langle A \rangle \rightarrow [2 \times 2]$

$\langle B \rangle \rightarrow [2 \times 1]$

$\langle C \rangle \rightarrow [1 \times 2]$

$\langle D \rangle \rightarrow [1 \times 1]$

```

1 % Norminal values of the Model Predictive Controller Object for Stage I
2
3 mpc_I_obj.Model.Nominal.X = [0; 0]
4 mpc_I_obj.Model.Nominal.Y = 0.2200
5 mpc_I_obj.Model.Nominal.U = 200.2291
6 mpc_I_obj.Model.Nominal.DX = [0; 0]

```

Stage II

state space parameters

$\langle A \rangle \rightarrow [1 \times 1]$

$\langle B \rangle \rightarrow [1 \times 1]$

$\langle C \rangle \rightarrow [1 \times 1]$

$\langle D \rangle \rightarrow [1 \times 1]$

```

1 % Norminal values of the Model Predictive Controller Object for Stage II
2
3 mpc_II_obj.Model.Nominal.X = 0
4 mpc_II_obj.Model.Nominal.Y = 0.2
5 mpc_II_obj.Model.Nominal.U = [10.5944, 0, 0]
6 mpc_II_obj.Model.Nominal.DX = 0

```

The state space parameters together with the nominal variables of the controller, are obtained through a *Bus Creator* and used to update the MPC internal plant model and the nominal variables for optimization at runtime.

Chapter 4

Results and Findings

This chapter is dedicated discussion of the result obtained from the design of chapter 3 for the both stages of compaction. Based on the requirement specifications outlined in section 3.1, the feedback system has been designed to satisfy these requirements. The robust stability and time domain performance will be analysed here on the basis of Simulink simulation and design review function of MPC toolbox.

4.1 Evaluate MPC performance with linear plant

A linearised plant internal to MPC has been obtained at the design stage and a performance verification check was carried out with the MPC design toolbox.

4.1.1 Stage I

The goal is to verify that the designed controller tracks the reference signal when given a step reference signal. Here, there is no measured or unmeasured disturbance. Given the set variables described in 4.1,

Variable	Value	Variation type
Reference Value (first compaction)	0.22	step size=0.1
Prediction Horizon	15	
Control Horizon	3	

Table 4.1. Model Predictive Controller Test parameters

the MPC is first run with its linear plant in a feedback system using the design nominal values showed on *listing 3.7.3*. It can be seen that the system response tracks the nominal reference as desired.

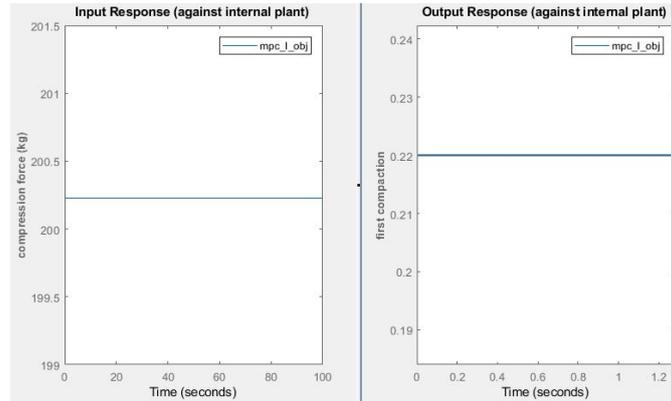


Figure 4.1. Testing of Stage I Model Predictive Controller With Linear Plant compaction with nominal values

With the weighting functions and constraints on the input and output variables imposed, the reference input is given a step size of 0.1. The controller is then tuned to reach the desired speed. The simulation is run for 10 control steps. The result of the simulation is as shown in the figure 4.2

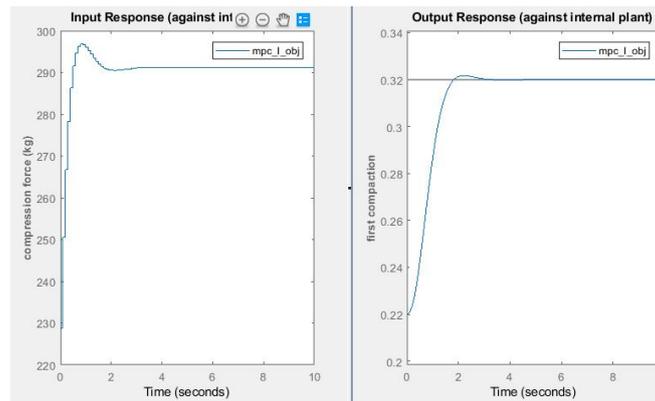


Figure 4.2. Testing of Stage I Model Predictive Controller on Linear Plant with a step size of 0.1 on the reference

The result shows that the control system consisting of the linear plant will reach a steady state at about 3 time steps with the optimum control input and that the step reference value is tracked.

4.1.2 Stage II

The goal is to verify that the designed controller tracks the reference signal in spite of the input disturbances.

Given the set variables outlined in table 4.2,

Variable	Value	Variation type
Reference Value (Compaction)	0.2	constant
Unmeasured Disturbance (line speed)	10	step size = 1
Unmeasured Disturbance (first compaction)	0.22	step size = 0.1
Prediction Horizon	15	
Control Horizon	3	

Table 4.2. Model Predictive Controller Design Variables

the MPC is tested with the linear plant in a feedback system using the design nominal values. Also as desired here, it can be seen from 4.3 the system response tracks the reference.

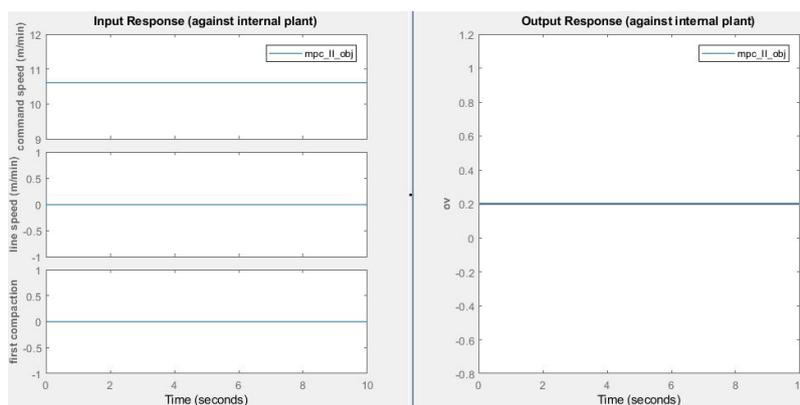


Figure 4.3. Testing of Stage II Model Predictive Controller With Linear Plant compaction with nominal values

With the weighting functions and constraints on the input and output variables imposed, the line speed variable is given a step size of 1, the first compaction variable is given a step size of 0.1 and the reference value is maintained constant to retain its nominal value. The controller is then tuned to reach the desired speed. The simulation is then run for 350 time steps. The result of the simulation is as shown in the figure 4.4

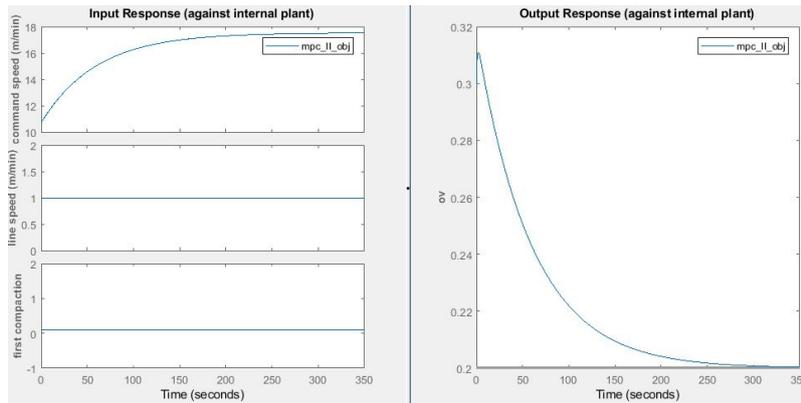


Figure 4.4. Testing of Stage II Model Predictive Controller on linear Plant with steps on disturbance variables

From the result, it can be shown that the control system consisting of the linear plant will reach a steady state at about 300 sec, tracking the reference input in spite of the input disturbances.

4.2 Evaluate MPC performance with non-linear plant:

With this check, it is desired to have a better understanding of how the designed controller will work in real life scenario. Exporting the design controller and updating the MPC block with the designed MPC object, the plants' manipulated input variables are connected to the mv port of the controllers while the output variables of the plant are connected to the measured output port of the controller to form a closed loop. The simulation for stage I is run for 10 sec and result is viewed through a scope as shown in figure 4.5

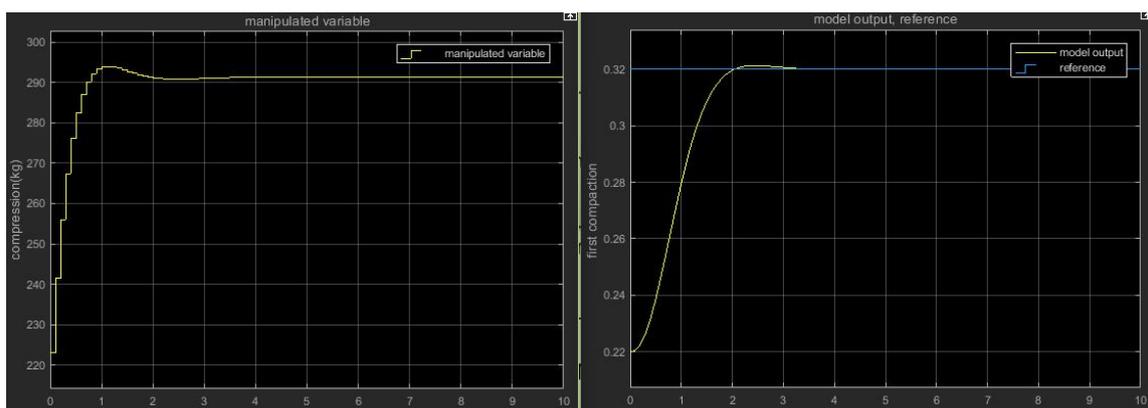


Figure 4.5. Testing of Stage I Model Predictive Controller with Non-Linear Plant

The result shows that the control system response is able to track the step reference value for the non-linear plant just like in the case of the linear plant.

To check the effectiveness of the designed control system, it is necessary to view the plots for the cost function and quadratic programming status as shown in figure 4.6.

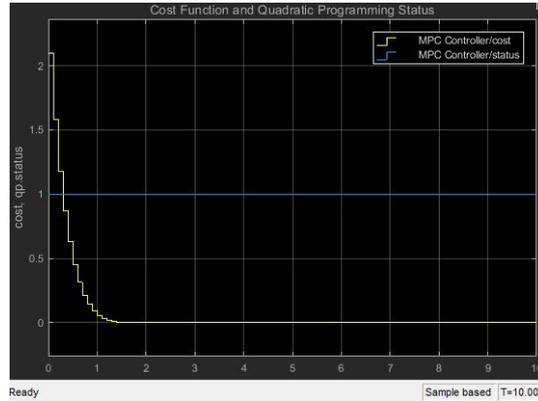


Figure 4.6. Cost Function and Quadratic Programming for stage I.

The very low value cost function indicates that the controller is performing well and that there are no constraint violations. The quadratic programming (qp.status) value of one (1) means that the number of QP solver iteration using in the computation is just 1.

In stage II compaction control subsystem, the Line speed and the first compaction inputs are increased by 1 and 0.1 respectively to reflected the set step size used in the test of section 4.1.2. The simulation is then run for 350 steps. The result of the simulation viewed through a scope is as shown in figure 4.7

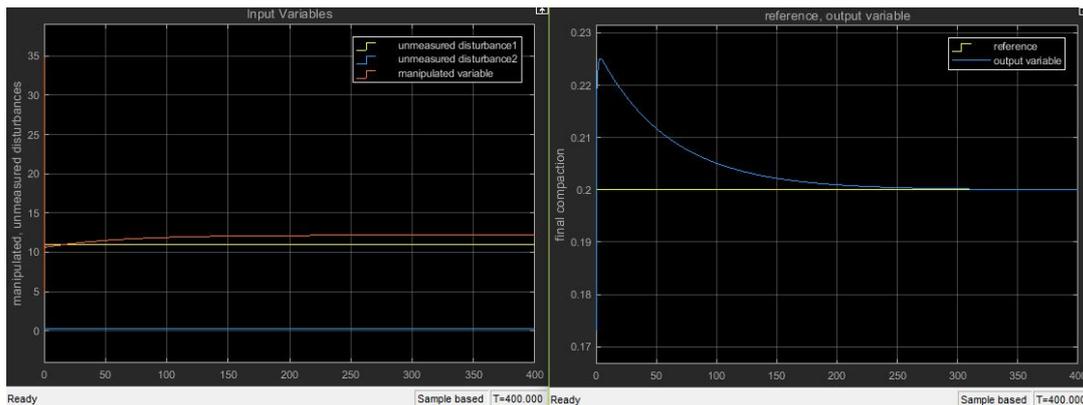


Figure 4.7. Testing of Stage II Model Predictive Controller With Non-Linear Plant

The resulting control system generates optimum inputs for the manipulatable variable

and drives the control system to the steady state at about 270s, tracking the reference step input, and indicating similarity with the linear plant representation. This shows that in spite of the disturbances, the performance of the control system is still retained.

The cost function and quadratic programming status as shown in figure 4.8.

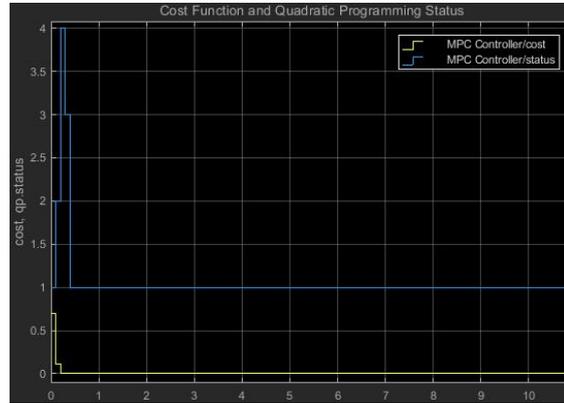


Figure 4.8. Cost Function and Quadratic Programming for stage II.

4.2.1 MPC Adaptivity Check

To check the adaptivity of the control system, the MPC block is replaced with an Adaptive MPC block and the rest of the features described in subsection 3.7.3 are implemented. The stage I adaptive result shown in figure 4.9 indicates the Adaptive MPC displays an acceptable tracking error even with an on-line estimated parameters.

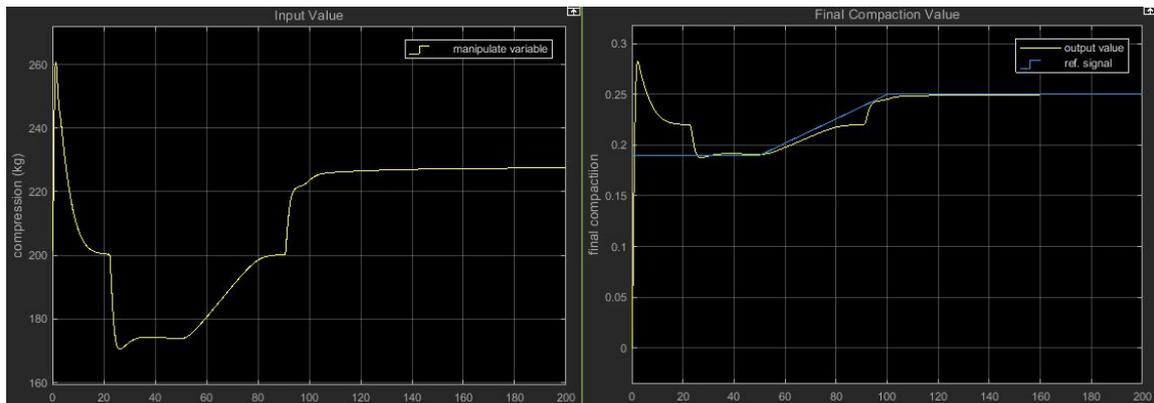


Figure 4.9. Adaptivity check result of Stage I

The cost function and Quadratic Programming status of the Stage I Adaptive MPC is shown in figure 4.10

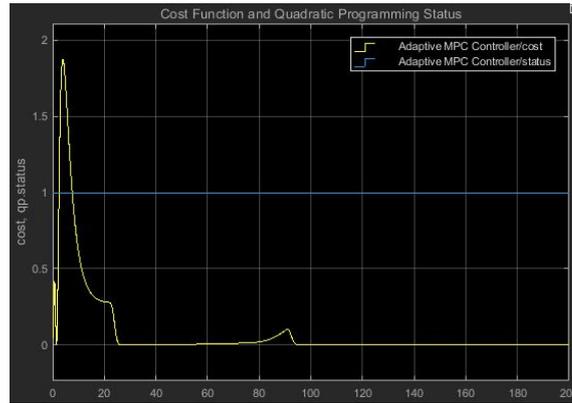


Figure 4.10. Cost function and Quadratic Programming status of Stage I

Also, for stage II, the simulation result is as shown in figure 4.11. This results shows that even though the controller of stage is not fast enough to track the reference value within a short period, reference signal tracking is guaranteed within considerable time.

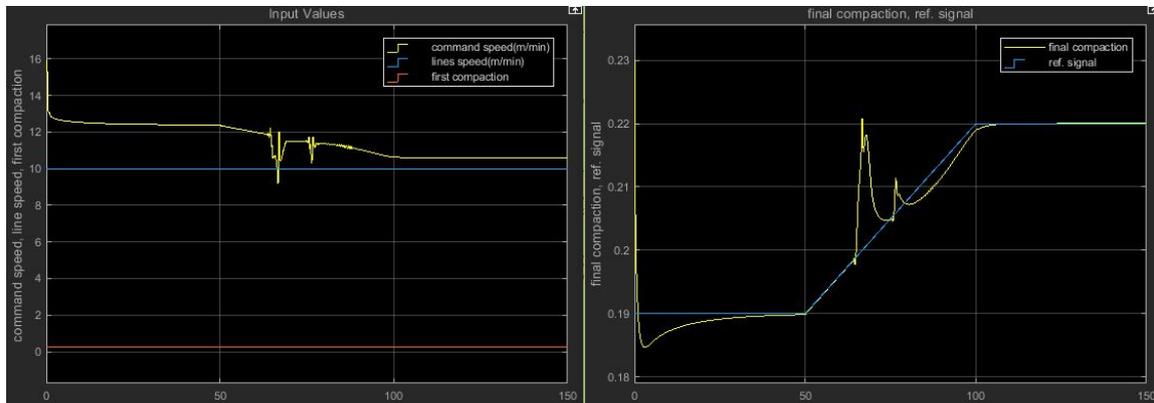


Figure 4.11. Adaptivity check result of Stage II

While the cost function and Quadratic Programming status of the Stage II Adaptive MPC is shown in figure 4.12

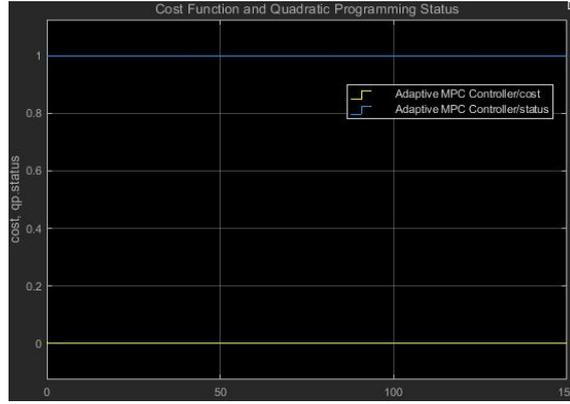


Figure 4.12. Cost function and Quadratic Programming status of Stage II

A Simulink representation of MPC and Adaptive MPC for Stage II is shown in Appendix B

4.2.2 Optimum Cost and Optimization Status:

These are features of MPC that act as indicators to the performance of the controller on the bases of successful optimization as well as constraints violation status. A small value of *Optimum cost* indicates that the controller is performing well and no constraint has been violated. If optimization problem is infeasible, the value will be meaningless.

If a QP problem is solved successfully at a given control interval, the `qp.status` output returns the number of QP solver iterations used in computation. This value is a finite, positive integer and is proportional to the time required for the calculations. Thus, a large value means a relatively slow block execution for this time interval.

The QP solver can fail to find an optimal solution for the following reasons:

$qp.status = 0$ → The QP solver cannot find a solution within the maximum number of iterations specified in the mpc object. In this case, if the Optimizer

$qp.status = -1$ → The QP solver detects an infeasible QP problem.

$qp.status = -2$ — The QP solver has encountered numerical difficulties in solving a severely ill-conditioned QP problem. In this case, the block holds its `mv` output at the most recent successful solution.

The simulation results are shown in a tabular form;

Variable	Value	Remark
Manipulated Variable	200.2	Satisfactory
Optimum Cost	4.189×10^{-19}	Satisfactory
Quadratic Programming Status(qp.status)	1	Satisfactory
Disturbance and Noise Model State	-8.267×10^{-12}	Satisfactory
Measured Output \approx Reference	0.22	Satisfactory

Table 4.3. Stage I simulation result for controller tested with non-linear plant

Stage I simulation result for Non-Linear Plant

Variable	Value	Remark
Manipulated	12.18	Satisfactory
Optimum Cost	4.098×10^{-14}	Satisfactory
Quadratic Programming Status(qp.status)	1	Satisfactory
Disturbance and Noise Model State	2.173×10^{-7}	Satisfactory
Measured Output \approx Reference	0.2	Satisfactory

Table 4.4. Simulation result for controller tested with non-linear plant

Stage I simulation result for Non-Linear Plant

4.2.3 Design Review by MPC Review Functionality:

This review is carried out to reveal the stability status, effectiveness, suitability, and limitations of the control system. Based on the results from the review, adjustments were made to obtain an optimum controller.

Figure 4.13 shows the result summary of the review which the same of stages I and II.

Summary of Performed Tests

Test	Status
MPC Object Creation	Pass
QP Hessian Matrix Validity	Pass
Closed-Loop Internal Stability	Pass
Closed-Loop Nominal Stability	Pass
Closed-Loop Steady-State Gains	Pass
Hard MV Constraints	Pass
Other Hard Constraints	Pass
Soft Constraints	Pass
Memory Size for MPC Data	Pass

Individual Test Result

Figure 4.13. Summary on Review of Model Predictive Control for Stages I and II Compaction

Closed-Loop Internal Stability Check Considering here that the plants of stages I and II are stable as demonstrated by its bounded output for a given bounded input as shown in figure 4.14, the designed feedback system is internally stable if all controller modes are exponentially stable or pure integrators.

This test extracts the A matrix from the unconstrained controller's state space realization, and then calculates its eigenvalues. If the absolute value of each eigenvalue is 1 or less and the plant is stable, the feedback system is internally stable.

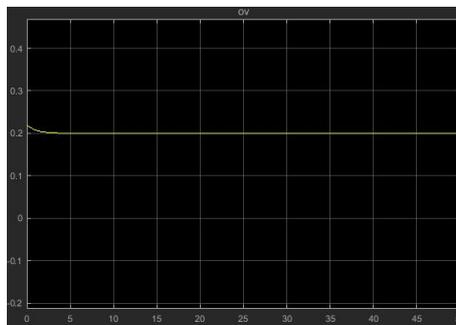


Figure 4.14. Demonstration of Stage II Plant Stability

Finding: the controller is internally stable.

4.2.4 Closed-Loop Nominal Stability

The feedback connection of the plant and controller should be stable for the nominal case: when the controller's prediction model represents the plant perfectly and no constraints are active.

This test obtains the discrete-time state-space realization of the closed-loop system – the plant and controller connected in a feedback configuration. It extracts the A matrix from this and calculates its eigenvalues. If the absolute value of each such eigenvalue is 1 or less, the nominal (unconstrained) system is stable. *Finding:* The Closed-Loop Nominal Stability is satisfied.

4.2.5 Closed-Loop Steady-State Gains

clffset is used to determine whether the controller forces all controlled output variables to their targets at steady state, in the absence of constraints. The controller is verified to forces all Output Variables to their targets at steady state

4.2.6 QP Hessian Matrix Validity

Since QP's Hessian matrix must be positive-definite, positive-definite Hessian can be guaranteed by increasing penalties on manipulated variable change (MV rates). This may however consequently result in a sluggish controller response. Based on the controller's minimum MV rates given in table 3.5, there is satisfaction in the controller behaviour.

Also, using a penalty weights on output variables emphasis the importance of output tracking. The minimum weight given on table 3.6 on the output variable is helpful for an acceptable effect on the Hessian.

Finally, as discussed in section 3.7, the Hessian is also influenced by scale factor and horizon parameter for plants with delay.

Finding: No problem was detected.

4.2.7 Hard MV Constraints

The controller should always satisfy hard bounds on a manipulated variable OR its rate-of-change. Specifying both constraint types simultaneously, however, may result in a conflict during real-time use. For example, if an event pushes an MV outside a specified hard bound and the hard MV rate bounds are too small, the resulting QP will be infeasible.

Finding: No hard MV constraint conflicts were detected on the control system design based on the test review.

4.2.8 Other Hard Constraints

It can be impossible for the designed controller to satisfy all its hard constraints under all conditions. If so, the controller will declare the optimal control (QP) problem to be infeasible, an error condition.

The manipulated variable rates (MVRates) are the QP decision variables, so their bounds are always feasible. Other variables, such as outputs (OVs), depend upon the MVRates. Unusual events might make OV bounds impossible to satisfy. Hard custom constraints can also become infeasible.

Finding: no hard output or custom constraints detected.

4.2.9 Soft Constraints

ECR Parameters

This test evaluates the constraint ECR parameters to help you achieve the proper balance of using hard and soft constraints. Having defined 2 soft constraints, table 4.5 below lists these and shows potential violations based on specified variable bounds and other factors.

Constraint	Assumed Violation	Impact Factor	Sensitivity Ratio
Stage I			
Lower limit: Output Variable	0.1	1	1000
Upper limit: Output Variable	0.1	1	1000
Stage II			
Lower limit: Output Variable	0.1	1	1000
Upper limit: Output Variable	0.1	1	1000

Table 4.5. Evaluation of Constraints ECR parameters

Impact Factor: the increase in the MPC cost function caused by this constraint violation relative to the average such increase. Rows are sorted in order of decreasing impact.

Sensitivity Ratio: the increase in the MPC cost function caused by this constraint violation relative to the typical cost function magnitude when there are no violations.

We consider a possible constraint violation equal to 10% of the nominal OV range. It then estimates the impact of such a violation on the MPC objective function relative to the impact of other violations. A large impact factor indicates a high-priority controller objective, and vice versa. A sensitivity ratio greater than $1e+08$ may degrade QP solution accuracy.

Finding: Sensitivity ratios are acceptable.

Impact of delays

Delays can make it impossible to satisfy output constraints. The presence of unattainable constraints usually degrades performance. Let j be the location (within the prediction horizon) of the first finite constraint value (Min or Max) for $OV(i)$. If all delays for $OV(i)$ exceed j , the constraint is unattainable.

Finding: no problems detected.

4.2.10 Memory Size for MPC Data

A recommendation on the estimate amount of memory required for an on-line implementation of the MPC has been detailed also through this test. It is assumed that a scalar value takes 4 bytes in single precision and 8 bytes in double precision. Table 4.6 below estimates how much physical memory, for example RAM on board, is needed to store the matrices used in online optimization. The value depends on the MPC controller settings such as horizons, plant order, plant size and the number of constraints.

Type	Single Precision (kB)	Double Precision (kB)
Stage I		
MPC	10	20
MPC with Online Tuning	20	40
Stage II		
MPC	10	20
MPC with Online Tuning	20	40

Table 4.6. Memory Estimation for MPC Deployment

The estimation does not include source code memory size (memory required to store the generated code).

4.3 Practical Recommendations

Hard MV Constraint Conflict This conflict is avoided by specifying either hard MV bounds OR hard MV rate bounds, but not both. Or if both must be specified, the lower-priority constraint should be soften by setting its ECR to a value greater than zero.

qp.status In a real-time application, qp.status can be used to set an alarm or take other special action.

Memory Issues If the physical memory size of the hardware is less than the estimated data memory requirements of the controller, the hardware can run out of memory when the controller is deployed. The controller should be redesign to reduce its memory requirements by using shorter horizons, reducing the plant, or reducing the constraints. Alternatively, the available physical memory should be increased.

Chapter 5

Future Work, Final Remarks and Conclusion

5.1 Future Work

Fabric quality actualization objective of Sanforized compactor makes it indispensable in textile industry. The major attention on this system is to optimize its compaction precision with a robust control system that accommodated parameter variations. For this reason, researchers are being drawn from different engineering domains in a bid to find solutions to the bothering challenges, and to invariably contribute to the sustainable supply of the huge demands of the textile market.

5.1.1 Precise System Modelling

The subsystems of Sanforized compactor has been modelled in this project using an approximated model approach using Bond-graph. Better results can be obtained with Bond-graph modelling if the parameters of the systems can be determined experimentally with improved precision. This however may be way expensive. Robust system Identification still remains the most viable option for the system modelling based on time and cost. With the output sensors provided at strategic positions on the subsystems, the input and outputs values are automatically collected at runtime and desired results are correspondingly obtainable.

ARX model which has been exploited here for the on-line estimation returns a good approximation of the on-line model. Set-Membership System Identification approach results in a more precise and robust system estimation. However due to it's high computation demand, it is currently not on-line implementable. Nevertheless, with a few optimization, the precision of ARX model can be improved too. A lot of researches are on-going in the industry in improving on the computational demands and in finding alternative means for fast on-line robust estimation.

A couple of other approaches that provide very accurate and robust system estimation have been lay down on literatures. Recently, error-domain model falsification (EDMF) has been demonstrate to be useful for situations where little information is available to

define the probability density function (PDF) of modelling errors. Also a fast and robust on-line system identification based on multi-layer recurrent neural networks strategy has also been proposed by ([32]:Won-Kuk Son et al, 1997).

5.1.2 Simplified and Cost Effective Control

In comparison with the current PI controller being implemented in Sanforized compactor, Model Predictive Controller has a larger structure complexity. The quadratic optimization is not trivial for a layman's understanding. For proper implementation and maintenance, there should always be need for MPC experts. For this reason, it is desired that MPC be structured for user friendliness.

The system has been split into stages I and II here for simplicity and for demonstrative reasons, during implementation, the two (2) subsystems will be controlled by a single controller. With this, the memory demand is reduced to half of the proposed value.

The overall process of Bianco textile production line is managed by Schneider Programmable Logic Controller (PLC) using any of the IEC 61131 standard programming language and the process visualisation is obtained with Human Machine Interface (HMI) programme. Bianco SPA Sanforized compactor runs on Unity Pro PLC and Vijeo HMI. The challenge here is that, until now, there is no cross-compiler for converting Matlab Simulink code into Unity Pro Standard PLC code. Any attempt to manually implement this is very unpleasant and time consuming. One of the two proposed possibilities is to use an advanced HMI panel. The advanced HMI is incorporated with it extra high speed memory for application extension. This memory is way enough to implement the MPC. The drawback here is on the cost of the advanced HMI panel. Alternative, the MPC can be deployed into an embedded micro-controller and the controller input and output values captured through the ports and read across the PLC. This is a more viable option for cost effectiveness. However, a more suitable and integrated control system will be guaranteed if the need for Unity Pro cross-compiler is met.

5.2 Conclusion

Based on the needs under consideration, the analyses, designs, and results obtained so far, it can be concluded that:

- an approximate Sanforized system model can be obtained using Bond-graph system modelling approach. However, a more precise system model can be robustly estimated using Set-membership system identification approach.
- Sanforized compactor requires a robust controller for the optimization of compaction results and for obtaining a corresponding optimized production system automation.
- Model Predictive Controller may be enough to control the system, while respecting the constraints, if the parameters of the system are not expected to change at run-time.

- for an anticipated change in the system parameters at run-time, it is necessary to use an Adaptive MPC in order to adapt to this changes at run-time.

Appendix A

Modelling of Compression Behaviour of the fabric

This analysis is mainly based on the work of (A. J. Thompson et al, 2017). [6] The mechanistic analysis of the compressibility of fibrous structures has been the focus of a substantial amount of research which can be traced back to [7, Van Wyk C.(1946)] and his pioneering work on the compressibility of randomly orientated fibres. Van Wyk attempted to simulate the mechanics of fibre interaction by modelling them as a series of bending units. Assuming that fibre bending was the only relevant deformation mechanism, he was able to derive a relation between pressure and volume fraction of stochastically placed fibres. Later on, [8, Gutowski (1992)] developed a similar method with specific application to composite reinforcements.

Many works which followed focused on modifying these relationships to account for other important mechanisms to gain higher accuracy ([9, Komori and Itoh (1997)]; [10, Komori and Makishima (1977)]; [11, Pan (1993)]).

According to [6, A. J. Thompson et al, 2017], one of the main difficulties for predicting the compressive response of woven textiles is their multi-scale nature. Typically, their micro-scale behaviour is determined by the interactions of thousands of aligned fibres, bundled together to form yarns. These yarns are woven together to form complex, ordered, meso-scale structures, the behaviour of which is driven by the inter-yarn interactions as well as the fibre interactions within the yarns. As the meso-scale structure of woven fabrics has shown to affect both the permeability and the mechanical behaviour of the final component, it is necessary to predict the deformations of the structure as well as its compressive response.

The multi-chain group of methods has been one of the major breakthroughs in the discrete modelling of fibres within woven structures (Green et al., 2014; Mahadik and Hallett, 2010; Zhou et al., 2004). These methods are based on the discretisation of yarns into multiple 1D element chains, with each element chain representing a bundle of fibres. The interactions between the element chains during weaving and compaction processes is simulated through the use of contact models.

Recent studies (El Said et al., 2014; Green et al., 2014) have shown the ability of these methods to capture both yarn cross-section deformation and yarn path evolution for 3D woven structures during the compaction process, achieving good correlation with x-ray micro-computed tomography (CT) scans. However, the kinematic nature of these methods limits their ability to determine the mechanical response of the fabric, as the properties of the physical constituents are not fully preserved within the model.[6]

A couple of other widely used methods have been published in the recent time. Nonetheless, considering the mechanical response limitation of this method, a two-step method approach will be used to integrate two modelling techniques that capture the multi-scale nature of textile reinforcement compaction.

This framework allows multiple layers of fabrics and their interactions to be simulated using basic geometric data of the fabric architecture and a single yarn compaction curve. The first step utilises a kinematic, multi-chain method for generation of the initial aswoven geometry of the fabric. The second step uses this geometry to generate a **3D finite element model**, able to capture both the kinematic and mechanical response of fabrics during compressive loading. *The key benefit of this method is that the dependency on detailed geometric and mechanical examination of the physical specimen is removed. Hence, the possibility of making accurate predictions without access to the physical specimen is attained.* [6]

A multi-chain method for 2D fabrics is developed, based on (Green et al., 2014) to predict as-woven and compacted single and multi-layer fabric geometries. The fidelity of the method in capturing the kinematic behaviour of fabrics during compressive loading is examined and compared to X-ray micro-CT scans. The geometric features of the as-woven fabric, predicted by the kinematic method, are then used as an input for the 3D finite element analysis of fabric compaction. A constitutive relation linking the pressure applied on the yarn to its volume fraction is formulated within a hyper-elastic framework and implemented within a user material for the finite element package Abaqus.

The model's ability to capture both the mechanical and kinematic behaviour of fabric during compaction processes is examined for both single and multi-layered preforms.

A.1 Kinematic Modelling of Weaving and Consolidation Process

To evaluate with multi-chain method the evolution of the internal level architecture for a single layer compaction, we consider analysing with;

- a 2/2 twill weave fabric
- Number of warp and weft yarns fibres - 12,000 fibres
- Diameter of fibre - 7 μm

these are then woven into a fabric with; - a unit cell size of 9.5 mm²

- areal density of 660 g/m²

With the analysis, the geometry was extracted was extracted for further for further

analysis at three stages: as-woven, compaction to 55% volume fraction (Vf) and compaction to 60% Vf.

To analyse the results, the centrelines and surface geometries of each yarn were extracted from the multi-chain model following the methodology outlined by Green et al. (2013). The result is as shown graphically in figure A.1;

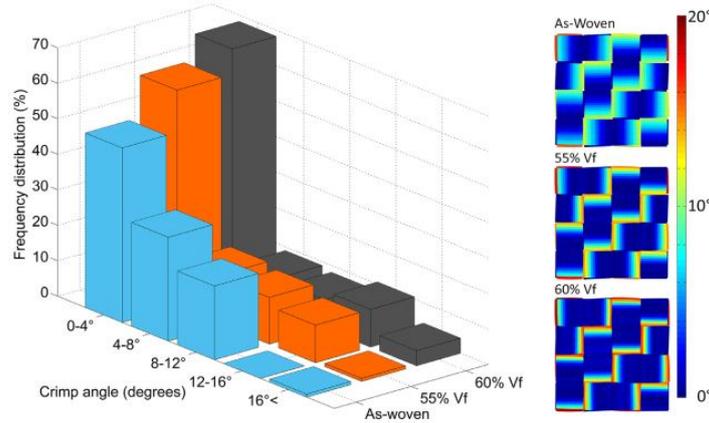


Fig. 2. (a) Histogram showing the frequency distribution of crimp angle at 3 stages of compaction (b) Gradient plots showing the evolution of crimp angle (within the fabric) at 3 stages of compaction.

Figure A.1. Single Layer Result Analysis

For multi-layer woven fabric preforms one of the most important contributions to compaction is the nesting and packing of layers. Nesting, which occurs when successive layers are offset, alters the contact surface between layers, changing the compressive response of the fabric and the resulting deformations.

One of the challenges in modelling offsets between layers is that the unit-cell of the fabric no longer remains constant from layer to layer, creating regions where yarns fall outside the unit cell, thus compromising the periodicity of the model. In order to combat this issue, slave yarns are introduced. Slave yarns are placed outside the unit cell and mimic the behaviour of a yarn which lies within the unit cell at its opposing side. This is achieved by copying the yarns which lie at the edges of the unit cell (master yarns) and translating them by one unit cell length to their opposing side as shown in Figure A.2

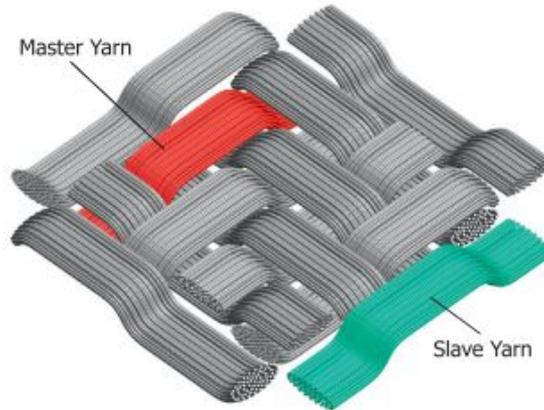


Figure A.2. Multiple Layer Offset Compensation

A.2 Mechanical Modelling and Consolidation Process

A method is therefore proposed here, whereby, the detailed as-woven geometry generated using the multi-chain element method is extracted and meshed with solid hexahedral elements. An experimentally determined compaction curve of a single yarn is used to deduce a relationship between intra-yarn volume fraction and pressure. Combining this relationship with the as-woven geometry enables the initial transverse stiffness for each yarn to be determined and updated incrementally based on the volumetric strain. By implementing this relationship into a hyper-elastic framework (applied within a user defined material subroutine in the FE package Abaqus) the compressibility of the yarns is handled by the constitutive relationship and the overall compressive response of the fabric, resulting from the interaction of interlacing yarns, is handled through the application of contact models.

Based on this method, a single layer compaction test result was plotted. The graph shows the typical behaviour of a fibrous network where the compressibility of the yarn is reduced with increased pressure until the yarn reaches maximum compaction as shown in figure A.3a.

The compaction pressure-fibre volume fraction relationship is presented in A.3b, which shows the yarn to stiffen as the volume fraction increases.

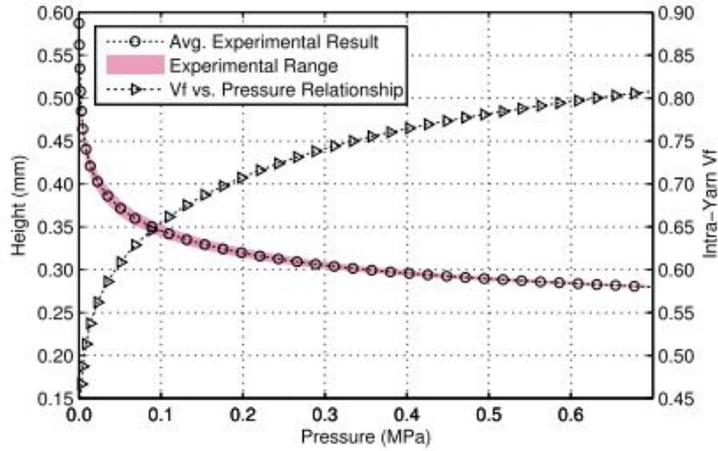


Figure A.3. (a) Experimental results of a single yarn compaction test (b) derived relationship of the intra yarn volume fraction/compaction load

With the above procedure, the elastic constants are also determined.

To determine the accuracy of the proposed method, compaction experiments were performed on 5 samples of the same fabric as modelled here for comparison. Each sample was a 100mm², single layer of the 2/2 twill weave. The samples were placed on the bottom plate of a 10 kN load cell universal testing machine and compacted at a rate of 0.5 mm/min. The machine compliance was measured prior to the test and the nonlinear machine displacement was subtracted from the measured displacements in the post-processing of the results.

It can be shown that there is a close correspondence between the experimental and predicted results in figure A.4

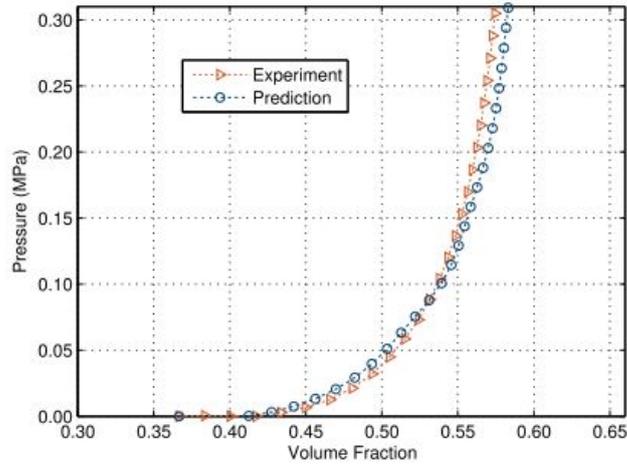


Figure A.4. Comparison between predicted and experimental results for a single layer 2/2 twill weave compaction

The developed model can be further used to consider the compaction of a stack of 6 layers, comparing compressibility of the aligned and shifted configuration as shown in figure A.5.

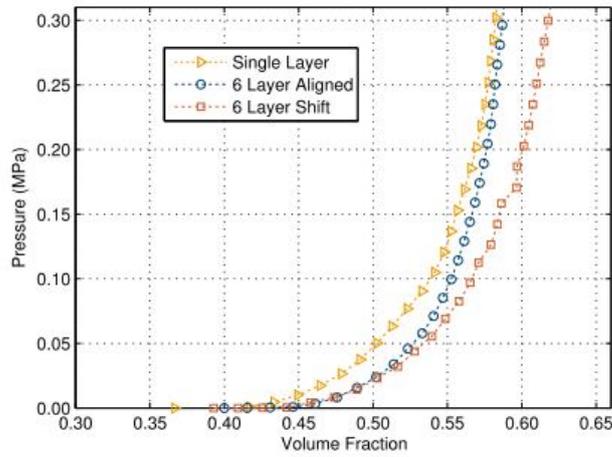


Figure A.5. Comparison of a single layer compaction with a 6 layer compaction with and without nesting

Based on the results of this analysis, the first stage compaction plant was obtained through estimation by set membership approach using the input and output data available from the plot.

Appendix B

MATLAB codes and Simulink Representations

B.1 Stage I MPC object Matlab Code

```
1 %% Manual MPC Design for Stage I Compaction
2
3 plant_md1 = 'Stage_I_MPC_Plant';
4 op = operspec(plant_md1);
5
6 %First Compaction Input Variable
7 op.Outputs(1).y = 0.22;
8 op.Outputs(1).Known = true;
9
10 %Computation of Initial Conditions
11 [op_point, op_report] = findop(plant_md1,op);
12
13 %Obtaining nominal values of x, y and u.
14 x0 = [op_report.States(1).x];
15 y0 = [op_report.Outputs(1).y];
16 u0 = [op_report.Inputs(1).u];
17
18 %Obtain linear plant model at the initial condition.
19 sys = linearize(plant_md1, op_point);
20
21 %Dropping the unused inputs and outputs
22 sys = sys(1,1);
23
24 %Discretize the plant model because Adaptive MPC controller only accepts a discret
25 Ts = 0.1;
26 plant = c2d(sys,Ts);
27
28 %Designing an MPC at the initial operating condition. When running in the adaptiv
29 %Specify signal types used in MPC.
30 plant.InputGroup.ManipulatedVariables = 1;
```

```
31 plant.OutputGroup.MeasuredOutput = 1;
32 plant.InputName = {'compression(kg)'};
33 plant.OutputName = {'first compaction'};
34
35 %Create MPC controller with default prediction and control horizons
36 mpcobj = mpc(plant);
37
38 %Set nominal values in the controller
39 mpcobj.Model.Nominal = struct('X', x0, 'U', u0, 'Y', y0, 'DX', [0; 0]);
40
41 %Set scale factors because plant input and output signals have different orders o
42
43 mpcobj.MV.ScaleFactor = u0;
44 mpcobj.OV.ScaleFactor = y0;
45
46 %Constaints
47 mpcobj.ManipulatedVariables.Min = 0;
48 mpcobj.ManipulatedVariables.Max = 500;
```

B.2 Stage II MPC object Matlab Code

```
1
2 % Manual MPC Design for Stage II Compaction
3
4 plant_md1 = 'Stage_II_MPC_Plant';
5 op = operspec(plant_md1);
6
7 %Final Compaction Output Variable
8 op.Outputs(1).y = 0.2;
9 op.Outputs(1).Known = true;
10
11 %Line Speed Input Variable
12 op.Inputs(1).u = 10;
13 op.Inputs(1).Known = true;
14
15 %Initial Compaction Input Variable
16 op.Inputs(2).u = 0.22;
17 op.Inputs(2).Known = true;
18
19
20 %Computation of Initial Conditions
21 [op_point, op_report] = findop(plant_md1,op);
22
23 %Obtaining nominal values of x, y and u.
24 x0 = [op_report.States(1).x];
25 y0 = [op_report.Outputs(1).y];
26 u0 = [op_report.Inputs(1).u; op_report.Inputs(2).u; op_report.Inputs(3).u];
27
```

```
28 %Obtain linear plant model at the initial condition.
29 sys = linearize(plant_mdl, op_point);
30
31 %Dropping the unused inputs and outputs
32 sys = sys(1,3);
33
34 %Discretize the plant model because Adaptive MPC controller only accepts a discret
35 Ts = 0.1;
36 plant = c2d(sys,Ts);
37
38 %Designing an MPC at the initial operating condition. When running in the adaptiv
39 %Specify signal types used in MPC.
40
41 % plant.InputGroup.MeasuredDisturbances = 1;
42 plant.InputGroup.ManipulatedVariables = 1;
43 plant.OutputGroup.Measured = 1;
44 plant.InputName = {'command speed (m/min)'};
45 plant.OutputName = {'final compaction compaction'};
46
47 %Create MPC controller with default prediction and control horizons
48 mpc_II_obj = mpc(plant);
49
50 %Set nominal values in the controller
51 mpc_II_obj.Model.Nominal = struct('X', x0, 'U', u0(3), 'Y', y0, 'DX', 0);
52
53 %Set scale factors because plant input and output signals have different orders o
54
55 mpcobj.MV.ScaleFactor = u0(3);
56 mpcobj.OV.ScaleFactor = y0;
57
58 %Constaints
59 mpcobj.ManipulatedVariables.Min = 5;
60 mpcobj.ManipulatedVariables.Max = 50;
```

B.3 Simulink Representation of Stage II MPC

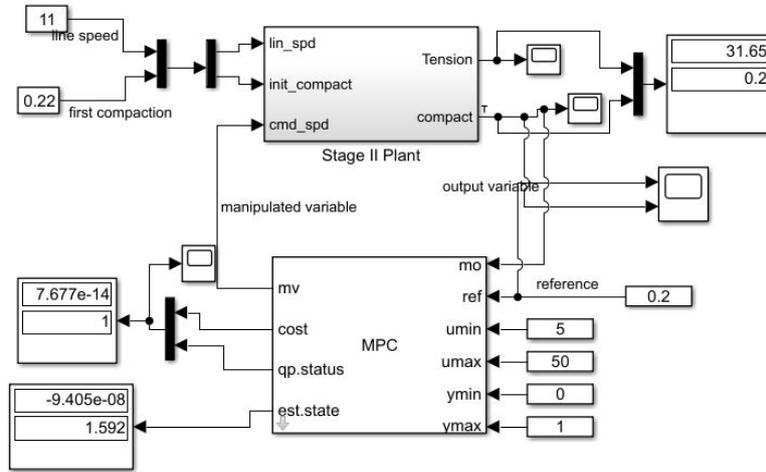


Figure B.1. Simulink Representation of Stage II MPC

B.4 Simulink Representation of Stage II Adaptive MPC

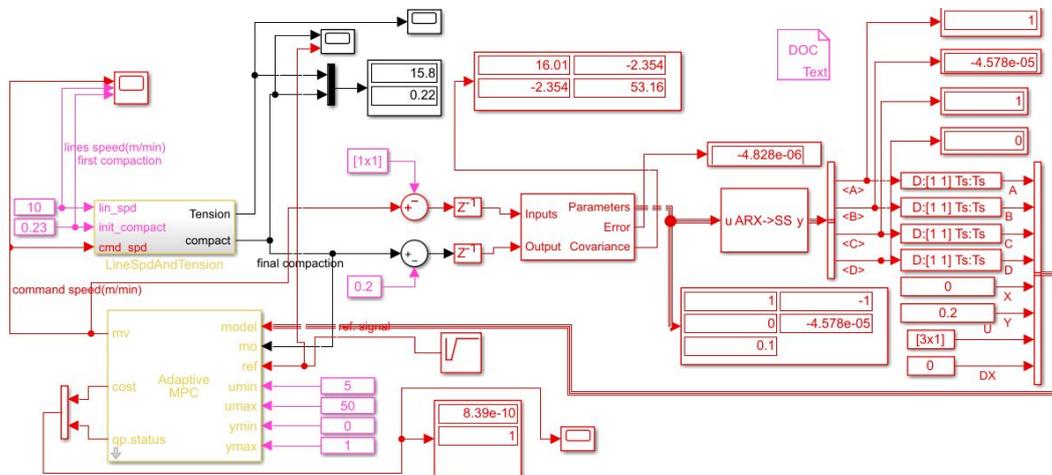


Figure B.2. Simulink Representation of Stage II Adaptive MPC

Bibliography

- [1] Mario M. and Gustavo B, 1982, "*Estimation Theory and Uncertainty Interval evaluation in presence of Unknown But Bounded Errors*", IEEE transactions on automatic controls, vol. AC-27,no.2, April 1982.
- [2] Daytex Shrinkage Belt, Saurer Daytex manual, www.saurer.com, last check. March, 2018
- [3] Cibitex Enhancing Finishing, 2010, "*Shrinkage Unit*", Manual Handbook, ed.10/2010, rev.3
- [4] Hans-Karl R., 2000, "*Compressive Shrinkage*", Encyclopedia of Textile Finishing, Autumn 2000, pp.174-180
- [5] Hawa Textile Zero Sanforize Belt, <http://hawasanforizebelt.com/products.html>, last check, January 2018
- [6] A. J. Thompson et al, 2017, "*High fidelity modelling of the compression behaviour of 2D woven fabrics*", International Journal of Solids and Structures (2017), <http://dx.doi.org/10.1016/j.ijsolstr.2017.06.027>
- [7] Van Wyk, C., 1946, "*20—Note on the compressibility of wool*", J. Text. Inst. Trans. 37, T285–T292.
- [8] Gutowski, T., Dillon, G., 1992, "*The elastic deformation of lubricated carbon fiber bundles: comparison of theory and experiments*", J. Compos. Mater. 26, 2330–2347.
- [9] Komori, T., Itoh, M., 1997, "*Analyzing the compressibility of a random fiber mass based on the modified theory of fibre contact*", Text. Res. J. 67, 204–210.
- [10] Komori, T., Makishima, K., 1977, "Numbers of fiber-to-fiber contacts in general fibre assemblies", Text. Res. J. 47, 13–17.
- [11] Pan, N., 1993, "*A modified analysis of the microstructural characteristics of general fiber assemblies*", Text. Res. J. 63, 336–345
- [12] Michael Nikolaou, 2001 "*Model predictive controllers: A critical synthesis of theory and industrial needs*", Advances in Chemical Engineering, Academic Press, 2001, Volume 26, Pages 131-204
- [13] Kothare et al, 2000, IEEE Control Systems Technology
- [14] Hayato Waki et al, August 2009 "*a Sparse Semidefinite Programming Relaxation of Polynomial Optimization Problems*", User Manual for SparsePOP, March 2005, B–414, Revised August 2009, ISSN 1342-2804
- [15] Željko P. et al, 2014 "*Determination of the Elastic Constants of Plain Woven Fabrics by a Tensile Test in Various Directions*" FIBRES and TEXTILES in Eastern Europe 2014, Vol. 22, No. 2(104)

-
- [16] Kenneth R. Muske and James B. Rawlings (1993) "*Model Predictive Control With Linear Models*", AIChE Journal, February 1993, Vol. 39, No.2
- [17] Matlab Documentation (2017) "*MPC Controller*" 2017b version
- [18] M.Milanese, A.vicino (1996) "*Optimum Estimation Theory for Dynamic Systems with Set Membership Uncertainty*", Bound approaches to System Identification, ISBN. 0-30b-4502L-b, plenum press, New York, 1996
- [19] V.Cerone, et al (2016) "*MIMO linear systems identification in the presence of bounded noise*" American Control Conference (ACC) Boston Marriott Copley Place July 6-8, 2016. Boston, MA, USA
- [20] V. Cerone, "Feasible parameter set for linear models with bounded errors in all variable," *Automatica*, vol. 29, no. 6, pp. 1551–1555, 1993.
- [21] V. Cerone, D. Piga, and D. Regruto, "*Improved parameters bounds for set-membership EIV problems*" *International Journal of Adaptive Control and Signal Processing*, vol. 57, no. 2, pp. 208–227, 2011.
- [22] —, "*Enforcing stability constraints in set-membership identification of linear dynamic systems*" *Automatica*, vol. 47, no. 11, pp. 2488–2494, 2011.
- [23] —(2012), "*Set-Membership Error-in-Variables Identification Through Convex Relaxation Techniques*" *IEEE Transactions on Automatic Control*, vol. 57, no. 2, pp. 517–522, 2012.
- [24] M. Pouliquen, E. Pigeon, and O. Gehan, "*Output error identification for multi-input multi-output systems with bounded disturbances*" in Proc. of joint 50rd IEEE Conference on Decision and Control and 2011 European control conference, 2011, pp. 7200–7205.
- [25] M. Milanese and C. Belforte (1982) "*Estimation Theory and Uncertainty Intervals Evaluation in Presence of Unknown But Bounded Errors: Linear Families of Models and Estimators*", *IEEE Transaction on Automatic Control*. VOL. AC-27.NO. 2. APRIL 1982.
- [26] T. Soderstrom and P. Stoica (2001) "*System Identification*", Prentice Hall international series in systems and control engineering, ISBN 0-13-i/81236-5, first pub.1989.
- [27] M. Milanese and M. Taragna, "H-infinity set membership identification: a survey", *Automatica*, vol. 41, no. 12, pp. 2019–2032.
- [28] — (2011), "Enforcing stability constraints in set-membership identification of linear dynamic systems," *Automatica*, vol. 47, no. 11, pp. 2488–2494, 2011.
- [29] V.Cerone, D.Piga, D.Regruto (2011) ("*Enforcing stability constraints in set-membership identification of linear dynamic systems*"), *A journal of EFAC*, rev.24 Jan. 2011,
- [30] Dean C. Karnopp, Donald L. Margolis and Ronald C. Rosenberg (2012)"*System Dynamics: Modeling, Simulation, and Control of Mechatronic Systems*", ed.5, Copyright © 2012 John Wiley and Sons, Inc.
- [31] "An Introduction to Model Predictive Control", https://www.iitk.ac.in/tkic/workshop/control%20techniques/ppt/patvardhan/Introduction_To_MPC_H last check. March 2018.
- [32] Won-Kuk Son et al, 1997"Fast and robust on-line system identification based on multi-layer recurrent neural network"*IEEE 1997 Canadian conference*

- [33] *Simpson J. R., Weiner E. S. C. (1989) "The Oxford English dictionary" Oxford: Clarendon Press. ISBN 0-19-861186-2.*