ELSEVIER

Journal of Process Control 11 (2001) 149–177

JOURNAL OF PROCESS CONTROL

www.elsevier.com/locate/jprocont

Robust cross-directional control of large scale sheet and film processes

Jeremy G. VanAntwerp, Andrew P. Featherstone, Richard D. Braatz *

Large Scale Systems Research Laboratory, Department of Chemical Engineering, University of Illinois at Urbana-Champaign, 600 South Mathews Avenue, Box C-3, Urbana, IL 61801-3792, USA

Abstract

Sheet and film processes, which include papermaking, polymer film extrusion, and adhesive coating, are of substantial industrial importance. The processes are poorly conditioned and truly large scale, with up to hundreds of manipulated variables and thousands of sensor locations. The uncertainties in sheet and film process models require that they be explicitly taken into account during the control design procedure. Numerically efficient algorithms are developed that provide robust optimal controllers for a wide variety of uncertainty descriptions. The robust optimality of the controllers can be relaxed to provide low order controllers suitable for real time implementation. Robust controllers are designed for a simulated paper machine, based on a realistic description of the interactions across the machine, and the level of model inaccuracies.

Keywords: Cross-directional control; Sheet and film processes; Papermaking; Robust control; Large scale systems; Polymer film extrusion

1. Introduction

Sheet and film processes have two main control objectives (see Fig. 1). One is the maintenance of the *average* sheet property profile, which is referred to as the *machine-direction* (MD) control problem. The other is the maintenance of flat profiles across the machine web, referred to as the *cross-directional* (CD) control problem. Since the MD problem [1–8] has been extensively studied and is much less difficult than the CD problem [9], only the CD problem is considered here.

Profile properties are controlled by actuators which are almost always located at evenly spaced points along the cross-direction [10]. The number of actuators can be 200 or more. Profile properties that have been controlled include basis weight (weight per unit area), moisture content, caliper, and opacity. Sensor measurements are taken after processing (e.g. pressing, drying, stretching) and are located some distance down the machine. In the past, due to their high cost, a small number of scanning sensors were used. Each sensor measured a zigzag portion of the sheet/film, as illustrated in Fig. 1. From this limited number of measurements, the profile properties were estimated at each sampling time for use by the control algorithm (for instance, by using a time-varying Kalman filter, as has been described by numerous authors [11–13]. Recently sensors have become available which simultaneously measure the CD profile as finely as every millimeter at rates of up to 120,000 times per minute [14]. This could result in as many as 10,000 sensors across the machine. The control problem is to calculate the 100 + control moves based on the measured or estimated profile of 500–10,000 sensor positions at each sampling time. The large dimensionality and the poor conditioning of sheet and film plant matrices makes these processes challenging to control.

Further, it is impossible to generate a highly accurate sheet/film process model, either phenomenologically or via input-output identification, because of unknown disturbances, inaccurate values for the physical parameters, cross-directional movement of the web [15,16], lack of complete understanding of the underlying physical phenomena (for example, during drying) [10], static friction, and equipment wear [16–19]. The large scale nature makes accounting for model uncertainty more

* Corresponding author. Tel.: +1-217-333-5073; fax: +1-217-333-5052.

E-mail address: braatz@uiuc.edu (R.D. Braatz).

0959-1524/01/$ - see front matter
PII: S0959-1524(00)00045-7

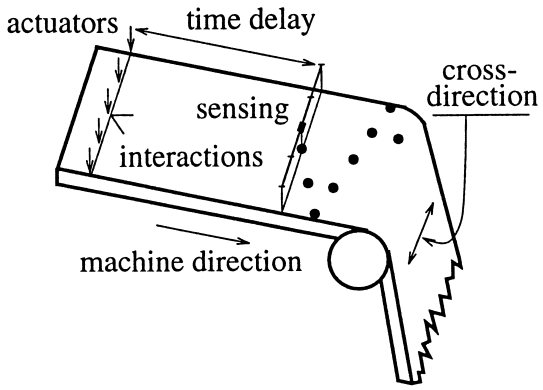

Fig. 1. Generic sheet and film process with scanning gauge (not drawn to scale).

important and more challenging [20–22]. The popular robust controller synthesis and model reduction software packages have demonstrated numerical inaccuracies for processes with large numbers of inputs and outputs [23].

This paper presents computationally efficient algorithms for designing robust optimal CD controllers for sheet and film processes. Further, it is shown how the optimality of the controllers can be relaxed to give low order controllers that are easier to implement.

1.1. Relationship to previous work

There is an extensive literature on non-robust optimal control strategies for sheet and film processes. This literature includes papers on controlling processes with symmetries, one of the early papers being by Roger Brockett and Jan Willems [24], and several recent papers which have proposed model predictive control (MPC) strategies [25–30]. The fast MPC algorithm of VanAntwerp and Braatz [31–33] was designed to avoid exciting uncontrollable plant directions, but does not actually guarantee robustness to all common types of model uncertainties. The reader is referred to [10,34] for a detailed review of optimal control design algorithms for sheet and film processes that do not explicitly address model uncertainty.

Laughlin, Morari and Braatz (LMB) [35] used circulant matrix theory to develop methods for designing conservative robust multivariable controllers based on the design of only one single loop controller. The LMB results applied to sheet and film processes with very highly structured interactions. Circulant symmetric, Toeplitz symmetric, and centrosymmetric symmetric models were all covered by the theory. The controllers were restricted to be either decentralized or decentralized controller in series with a constant decoupler matrix. Forcing the controllers to have these particular structures restricts the performance that can be achieved with these algorithms.

There are substantial differences between the results of LMB and the results presented here. First, LMB treated only restrictive types of interaction matrices, while our approach handles arbitrary interaction matrices. Second, LMB considered only parametric uncertainties in the interaction matrices, whereas here we treat nonparametric uncertainties. Third, the robust controller synthesis and analysis theorems presented here are much less conservative. Fourth, application of the LMB approach to a process with a different number of sensors than actuators would require squaring-up to give a square transfer function matrix. Although squaring-up procedures have been applied industrially for at least the last 15 years [36], they introduce an unnecessary approximation and can result in a loss of performance [22].

Duncan [19] developed a robust controller design algorithm for sheet and film processes with arbitrary interactions across the machine. Sufficient conditions for robust performance with multiplicative input and output uncertainties were derived in terms of satisfying robust performance for single-input single-output (SISO) subsystems similar to those treated here. The robust controller synthesis and analysis theorems presented here are potentially less conservative, and treat much broader types of uncertainty.

Hovd, Braatz and Skogestad (HBS) [23,37,38] presented several robust control results that are applicable to CD processes. Stewart et al. [39] proposed a variation on the robust CD control algorithms of HBS, but with more stringent assumptions on the model uncertainties, performance objectives, and interactions across the machine. A variation of the control algorithms has been implemented by Honeywell-Measurex on industrial CD control hardware working with a hardware-in-the-loop paper machine simulator [39].

The algorithms presented here are extensions and refinements of results by HBS. There are six significant new contributions. First, the HBS results are specialized for application to sheet and film processes, leading to substantially simplified statements of both the theory and the resulting algorithms. Second, the theorems here provide explicit expressions for lower dimension robust control problems whose solutions can be used to construct the robust controller for the original large scale control problem, whereas HBS provided only *conditions for the existence of* the lower dimension problems. Third, for many uncertainty types, we provide a much more complete model reduction. For example, where HBS may reduce the multivariable robust control problem to a large number of single-input single-output (SISO) robust control problems, in many cases our results can reduce the multivariable problem to a *single* SISO robust control problem. Fourth, algorithms for the design of low order robust controllers are investigated in detail. Fifth, nonlinear as well as linear perturbations

are addressed. Sixth, the algorithms are applied to a simulated paper machine, based on a realistic description of the interactions across the machine, and the level of model inaccuracies. This simulated example is of substantially higher dimensionality than that of any robust control problem ever considered.

2. Background on the robust control formulation for sheet and film processes

Here we describe sheet and film process models, the performance objective, appropriate uncertainty descriptions, and provide some background on robustness analysis and synthesis.

2.1. Sheet and film process models

The process model relates the manipulated variable moves to the profile properties measured downstream. All reported sheet and film process models have the form

$$y(s) = P(s)u(s), \qquad P(s) = p(s)P_{\mathrm{CD}}, \tag{1}$$

where y is a vector of measurements, u is a vector of actuator positions, $p(s)$ represents scalar dynamics, and P_{CD} is a constant matrix representing interactions between inputs and outputs. Taking the singular value decomposition (SVD) [40] of the interactions matrix P_{CD} allows the process transfer function to be decomposed into the *pseudo-SVD* form

$$\begin{aligned} P(s) &= p(s)P_{\mathrm{CD}} \\ &= p(s)U\Sigma V^T \\ &= U(p(s)\Sigma)V^T \\ &= U\Sigma_{\mathrm{P}}(s)V^T. \end{aligned} \tag{2}$$

where U and V are real orthogonal matrices. The elements of the diagonal matrix $\Sigma_{\mathrm{P}}(s)$ are transfer functions and are not ordered in any particular manner. These diagonal elements $\Sigma_{\mathrm{P},ii}(s)$ are referred to as *pseudo-singular values* [41]. The pseudo-SVD form is sufficiently general to allow for non-square P_{CD} with arbitrary interactions. For non-square P_{CD}, first augment P with rows or columns of zeros to make a square matrix. Then compute the SVD of the square matrix to result in square U and V. The pseudo-singular values corresponding to the additional rows or columns will be equal to zero. Although there are more compact ways to define the pseudo-SVD for a non-square interactions matrix, this definition leads to the simplest notation throughout the manuscript.

For symmetric P_{CD} $\left(P_{\mathrm{CD}} = P_{\mathrm{CD}}^T\right)$, an orthogonal decomposition of P_{CD} (e.g. Theorem 3 of [42]) allows U to be chosen equal to V. In this case, $U^T = U^{-1}$ and the diagonal elements of $\Sigma_{\mathrm{P}}(s)$ can be interpreted as *pseudo-eigenvalues*.

While many modern polymer film extruders have square interaction matrices, modern paper machines have many more sensors than actuators, resulting in a non-square interactions matrix [32]. Although almost all of the results in this manuscript will apply to the general model (1), somewhat stronger results will be reported for symmetric models.

2.2. Performance objective

A block diagram of the closed loop system is shown in Fig. 2. The objective of the controller $K(s)$ is to minimize the effect of disturbances d on the profile properties y. Since the sensitivity function $(I + PK)^{-1}$ is the transfer function between d and y, this objective can be quantified by

$$\begin{aligned} &\left\| W_{\mathrm{P}}(s)(I + P(s)K(s))^{-1} \right\|_\infty \\ &\quad \equiv \sup_{s=j\omega} \bar{\sigma}\left(W_{\mathrm{P}}(s)(I + P(s)K(s))^{-1}\right), \end{aligned} \tag{3}$$

where $\bar{\sigma}(A)$ refers to the maximum singular value of A, and the weight $W_{\mathrm{P}}(s)$ is selected to define the desired performance (e.g. bandwidth). The weight is also used to normalize the desired performance objective:

$$\left\| W_{\mathrm{P}}(s)(I + P(s)K(s))^{-1} \right\|_\infty \leqslant 1. \tag{4}$$

The goal of the CD control problem is to maintain flat profiles across the entire width of the machine, implying that the performance weight $W_{\mathrm{P}}(s)$ should be selected as a scalar weight $w_{\mathrm{P}}(s)$ multiplied by the identity matrix. The most commonly used weight has the form

$$w_{\mathrm{P}}(s) = b\frac{as + 1}{as}, \tag{5}$$

where a and b are constant real scalars [35]. With this performance weight, the maximum disturbance amplification will be less than $1/b$ at all frequencies, and the closed loop system will have a bandwidth of at least $1/a$.

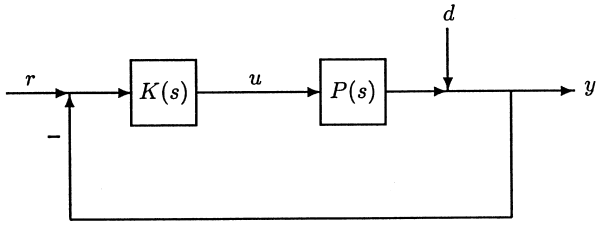

Fig. 2. Standard feedback control system.

A physically meaningful performance weight must satisfy $0 < b < 1$ and $a > 0$.

The performance objective in (3) is selected for several reasons. First, the objective allows the direct specification of the closed loop bandwidth, or equivalently, the closed loop speed of response [this is equal to a in (5)]. Second, the objective (3) is the worst-case gain for sinusoidal inputs at any frequency, which allows the engineer to directly specify a bound on the effect of oscillatory disturbances on the closed loop system. Third, the objective (3) bounds the integral of the squared profile deviations across the machine subject to disturbances bounded by the integral-squared-error norm (this can be interpreted for deterministic or stochastic disturbances, in which case the expected value is used [43]). Minimizing the squared profile deviations is the stated goal of most CD control systems [10,35,44]. Fourth, it is simpler mathematically to develop robust control algorithms based on the performance objective (3) than for most other performance objectives. Detailed discussions on selecting performance objectives and performance weights are available [43,45].

2.3. Uncertainty descriptions

Due to their poor conditioning and the limited input-output data available, a sheet/film process model is only an approximation of the true process. The inaccuracy is represented by describing the process model as a set of plants $\hat{P}$, given by a nominal model P and a set of norm bounded perturbations Δ. The six major types of multivariable uncertainty descriptions are listed in Table 1 [43,46].

Through weights each perturbation is normalized to be of size one

$$\left\|\Delta_i(s)\right\|_\infty \leqslant 1, \tag{6}$$

where $\Delta_i(s)$ is a stable transfer function representing unmodeled dynamics. In the more general case where Δ_i is not treated as being linear time invariant, other norms on Δ_i are used [47–49]. Uncertainties which have been carefully characterized include nonlinear time invariant (NLTI), nonlinear time varying (NLTV), linear time varying (LTV) [49], and arbitrarily-slow time varying (SLTV) [48].

Table 1
Six major types of multivariable uncertainty descriptions (dependence on s suppressed for brevity)

Uncertainty type	Mathematical representation
Additive	$\hat{P} = P + w_A\Delta_A$
Multiplicative input	$\hat{P} = P(I + w_I\Delta_I)$
Multiplicative output	$\hat{P} = (I + w_O\Delta_O)P$
Inverse additive	$\hat{P} = (I + w_{IA}P\Delta_{IA})^{-1}P$
Inverse multiplicative input	$\hat{P} = P(I + w_{II}\Delta_{II})^{-1}$
Inverse multiplicative output	$\hat{P} = (I + w_{IO}\Delta_{IO})^{-1}P$

Multiplicative input uncertainty represents inaccuracies associated with the actuators, whereas multiplicative output uncertainty represents inaccuracies associated with the measurements. Additive and multiplicative output uncertainties are the most commonly used to represent unmodeled process dynamics. The "inverse" uncertainties allow for processes in which it is not known with certainty whether poles near the imaginary axis are unstable or stable. Inverse multiplicative output uncertainty provides a convenient mathematical means to address performance specifications within the context of robust stability (this is explained in Section 2.4).

Each uncertainty block is of dimension compatible with the nominal model P. This implies that Δ_A has the same dimensions as P, Δ_I and Δ_{II} are square matrices of dimensions equal to the number of actuators, and Δ_O and Δ_{IO} are square matrices of dimensions equal to the number of sensing locations.

Each uncertainty block can have structure. In the literature, additive uncertainty (typically representing unmodeled process dynamics) is normally represented as a full matrix, whereas multiplicative uncertainties are treated as being either full or diagonal. Further, diagonal uncertainty blocks can be represented as having diagonal elements that are independent scalars, $\Delta_i = \mathrm{diag}\{\delta_{ij}\}$, or repeated scalars, $\Delta_i = \delta_i I$. A repeated diagonal uncertainty description may be appropriate for modeling inaccuracies in the sensor model, since the sensor is usually of the tracking type, with the same sensor being used to take all measurements. An independent diagonal uncertainty description would be more appropriate for representing inaccuracies in the actuator models [19] since each actuator is expected to have somewhat different dynamic response.

The uncertainty weights in Table 1 assume that components of the same type (for example, slice lip screws) have the same level of uncertainty associated with their respective models. This is a good assumption for sheet/film machine components, since each component of a particular type is almost always manufactured by the same company to provide the same level of reproducibility. Note that this assumption does not necessarily require that the models for each component of a particular type are precisely equal for all plants within the uncertainty description, only that the level of inaccuracy of each component is the same. The selection of uncertainty weights is described in several references [43,45,46]. Procedures have been developed for sheet and film processes for identifying both nominal models and uncertainty weights based on process data [41,50].

Now we define a non-traditional additive uncertainty description for a sheet and film process, in which the

pseudo-singular values in (2) are uncertain. The uncertainty description is represented as independent diagonal additive uncertainty (that is, $\Delta_{DA} = \mathrm{diag}\{\delta_k\}$, $k = 1, \ldots, n$)

$$\hat{P} = U(\Sigma(s) + \Sigma_{DA}\Delta_{DA})V^T = P + U\Sigma_{DA}\Delta_{DA}V^T, \qquad (7)$$

where Σ_{DA} is a diagonal weighting matrix, not necessarily equal to a scalar multiplied by the identity matrix. Theoretical justifications of this uncertainty description, including methods to compute Σ_{DA} from experimental data, are provided elsewhere [41,50].

2.4. Robust stability and performance

Algebraic manipulations performed either by hand [43,46,51] or with programs [52–55] can be used to collect the uncertainties associated with various components in the system into the block-diagonal $\Delta(s)$ shown in Fig. 3. The *generalized plant* $G(s)$ is defined by the nominal model $P(s)$, the performance specifications, and the magnitude and location of the uncertainties. The generalized plant $G(s)$ and the controller $K(s)$ can be combined to produce the nominal closed loop system matrix $M(s)$. If $G(s)$ is partitioned to be compatible with $K(s)$, then $M(s)$ is described by the linear fractional transformation (LFT), where s has been suppressed for brevity,

$$M = F_l(G, K) = G_{11} + G_{12}K(I - G_{22}K)^{-1}G_{21}. \qquad (8)$$

The LFT $F_l(G, K)$ is defined for any well-posed system [this is equivalent to the existence of the inverse of $(I - G_{22}K)$].

Eq. (6) implies that each block-diagonal matrix $\Delta(s)$ within the uncertainty description is in the set $\mathbf{\Delta}$, where

$$\mathbf{\Delta} \equiv \{\mathrm{diag}\{\Delta_k(s)\} \mid \|\Delta_k(s)\|_\infty \leqslant 1;\ \Delta_k(s)\ \text{stable}, k = 1, \ldots, u\} \qquad (9)$$

where each $\Delta_k(s)$ has the same dimensions as $P(s)$, and u is the number of uncertainty types. The structure of each $\Delta_k(s)$ can be repeated diagonal, independent diagonal, or full block.

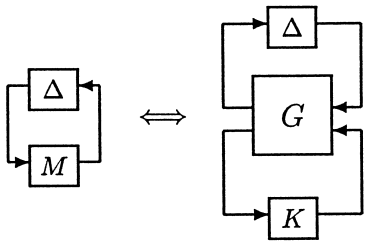

Fig. 3. Equivalent system representations (dependence on s suppressed by brevity).

The closed loop system is said to satisfy *robust stability* if it is stable for all stable norm-bounded perturbations $\Delta(s) \in \mathbf{\Delta}$. The closed loop system is said to be satisfy *robust performance* if the performance specification (3) holds for all $\Delta \in \mathbf{\Delta}$. The closed loop system is robustly stable to linear time invariant (LTI) perturbations if and only if the nominal closed loop system is stable (that is, the poles of $M(s)$ are in the open left half plane) and the structured singular value $\mu_{\mathbf{\Delta}}(M(j\omega))$ is less than 1 for all frequencies (see [43,46,56,57] for more details). The value of the matrix function $\mu_{\mathbf{\Delta}}(M(j\omega))$ at each frequency depends on both the elements of the matrix $M(s)$ and the structure of $\mathbf{\Delta}$. The corresponding test for robust performance is exactly as for the robust stability test, except with the performance specification treated as though it were an additional inverse multiplicative output uncertainty (that is, w_{IO} is set equal to w_P, with full block Δ_{IO} representing the performance specification).

It is a key idea that μ provides a general analysis tool for determining robust stability and performance with respect to LTI uncertainty [58–60]. Any system with uncertainty adequately modeled as in (6) can be put into $M - \Delta$ form, with robust stability and robust performance written as a μ-test. Although exact computation of the matrix function μ can be computationally expensive [61,62], upper and lower bounds for μ can be computed in polynomial time (M can always be augmented with zeros to a square matrix with the same value of μ, so without loss of generality M will be taken to be square in what follows):

$$\max_{U \in \mathbf{U}} \rho(MU) = \mu_{\mathbf{\Delta}}(M) \leqslant \inf_{D \in \mathbf{D}} \bar{\sigma}\left(DMD^{-1}\right), \qquad (10)$$

where $\mathbf{U}$ is the set of unitary matrices with the same block diagonal structure as $\mathbf{\Delta}$, $\rho(A)$ is the spectral radius of A, and $\mathbf{D}$ is the set of all matrices that commute with every $\Delta \in \mathbf{\Delta}$, that is, $\mathbf{D} = \{D \mid D\Delta = \Delta D$ for all $\Delta \in \mathbf{\Delta}\}$ [56,63]. This definition implies that each $D \in \mathbf{D}$ is a block-diagonal matrix with u blocks, the structure of each block defined by the corresponding block of $\Delta \in \mathbf{\Delta}$. In particular, D_k is full block for repeated scalar Δ_k, D_k is repeated scalar for full block Δ_k, and D_k is independent scalar for independent scalar Δ_k.

The maximization in (10) is not convex, and existing algorithms either provide only a local maximum or are computationally expensive [64,65], hence the reference to the maximization as being a "lower bound," although the equality in (10) holds [52]. The upper bound can be formulated as a linear matrix inequality and is solvable in polynomial time using either ellipsoid or interior point algorithms [66,67]. The computed lower and upper bounds are usually tight. However, computational experience indicates that the bounds become more conservative as the system dimension increases [64,65]. Robust suboptimal controllers are almost always computed using the upper bound.

The H_∞-optimal control problem is to compute a stabilizing $K(s)$ that minimizes $\|F_l(G, K)\|_\infty$ (see Fig. 3). The state-space approach for solving the H_∞ control problem is implemented in off-the-shelf software [52,53]. The DK-iteration method (often called μ-synthesis) is an ad hoc method that attempts to minimize the upper bound of μ, that is, it attempts to solve [52,53]

$$\inf_{K(s)\in\mathbf{K}_s^n}\ \inf_{D(s)\in D_s^{nu}}\ \sup_{s=j\omega}\bar{\sigma}\big(D(s)F_l(G(s), K(s))D^{-1}(s)\big), \tag{11}$$

where $\mathbf{K}_s^n$ is the set of all internally stabilizing controllers of dimension $n\times n$, and $\mathbf{D}_s^{nu}$ is the set of all $nu\times nu$ stable minimum phase transfer functions that satisfy $D\Delta=\Delta D$ at each frequency. The approach in DK-iteration is to alternatively minimize

$$\sup_{s=j\omega}\bar{\sigma}\big(D(s)M(s)D^{-1}(s)\big) = \sup_{s=j\omega}\bar{\sigma}\big(D(s)F_l(G(s), K(s))D^{-1}(s)\big) \tag{12}$$

for either $K(s)$ or $D(s)$ while holding the other constant. For fixed $D(s)$, the controller synthesis is solved via H_∞-optimization. For fixed $K(s)$, the quantity (12) is minimized for each $D(s)$ using linear matrix inequalities [66,67] or some other approach [52,53]. The resulting invertible stable minimum-phase transfer function $D(s)$ is wrapped back into the nominal interconnection structure $G(s)$. This increases the number of states of the scaled $G(s)$, which causes the second H_∞-synthesis step to produce a higher order controller. The iterations between $D(s)$ and $K(s)$ stop after the quantity (12) is less than 1 or is no longer diminished. The resulting high-order controller is typically reduced using Hankel model reduction [68]. Although the DK-iteration method is not guaranteed to converge to a global minimum, it has been used to design robust controllers for many mechanical systems, e.g. flexible space structures [69], missile autopilots [70,71], and rockets [72].

Besides being an approximation to the original μ condition for LTI perturbations, (11) is also interesting in its own right, as its objective less than one is a necessary and sufficient condition for robustness to arbitrarily slow linear time varying (SLTV) perturbations [48] when all the perturbations are full block. Also, the objective in (11) less than one is a necessary and sufficient condition for robustness to fast linear time varying (FLTV), nonlinear time invariant (NLTI), or nonlinear time varying (NLTV) perturbations when the matrices in $\mathbf{D}$ are restricted to be constant matrices [49].

3. Optimal robust controller design

To state the results, it is useful to recall that G is an open loop transfer function matrix defined by the uncertainty weights $w_j(s)$, the uncertainty locations (in Table 1), and the open loop nominal model $P(s)$. For the uncertainty types in Table 1 and in (7), $G(s)$ can be written in terms of submatrices that include only the following terms (including multiplications of the terms): $P(s)$, $w_j(s)I_n$, I_n, 0_n, U, and V, where I_n is the $n\times n$ identity matrix, and 0_n is the $n\times n$ matrix of zeros. Define the n lower dimension transfer functions $\tilde{G}^i(s)$, which are constructed from $G(s)$ by the following substitutions [see (45) and (49) for example]:

$$\begin{aligned} P(s) &\longleftrightarrow \Sigma_{P,ii}(s)\\ w_j(s)I_n &\longleftrightarrow w_j(s)\\ I_n &\longleftrightarrow 1\\ 0_n &\longleftrightarrow 0\\ U &\longleftrightarrow 1\\ V &\longleftrightarrow 1 \end{aligned} \tag{13}$$

Each of the $\tilde{G}^i(s)$ corresponds to a pseudo-singular value $\Sigma_{P,ii}(s)$ of the plant $P(s)$. To simplify the statement of the results, $P(s)$ will be treated as being square. As discussed earlier, this is without loss in generality.

The results of this section are of two types. First, it is shown that for various uncertainty types the robust controller of the form

$$K(s) = V\Sigma_{\mathrm{K}}(s)U^T \tag{14}$$

is optimal. Second, it is shown how controllers of this form simplify robustness analysis and synthesis by either partially or completely decoupling the MIMO controller design problem into SISO control problems, or a single SISO control problem. The robustness analysis and synthesis results are first presented for sheet and film processes with general interactions. Then somewhat stronger results are stated for symmetric nominal models.

3.1. Processes with general interactions matrix

For the case where all the uncertainty blocks are full and arbitrarily-slow linear time varying, the following theorem provides conditions for which the robust optimal controller has the form $K(s)=V\Sigma_{\mathrm{K}}(s)U^T$, and describes how this simplifies the computation of the robust optimal controller. Proofs of all results are in the appendix.

Theorem 1 (Robust optimality with SLTV Δ). *Consider a nominal model $P(s)=U\Sigma_P(s)V^T$, where U and V are real orthogonal matrices and $\Sigma_{\mathrm{P}}(s)$ is a diagonal transfer function matrix. Suppose there are multiple full block uncertainties of the forms listed in Table 1 and a diagonal additive uncertainty of the form (7). Then a controller of the form $K(s)=V\Sigma_{\mathrm{K}}(s)U^T$ minimizes*

$$\inf_{D(s)\in\mathbf{D}_s^{nu}}\ \big\|D(s)F_l(G(s), K(s))D^{-1}(s)\big\|_\infty \tag{15}$$

where u is the number of uncertainties and

$$\begin{aligned}&\mathbf{D}_s^{nu} \equiv \operatorname{diag}\{D_k\}, k = 1, \ldots, u,\\ &D_1 = \operatorname{diag}\{d_{1i}(s)\}, i = 1, \ldots, n,\\ &D_k = \operatorname{diag}\{d_k(s) I_n\}, k = 2, \ldots, u.\end{aligned} \tag{16}$$

The generalized plant G(s) is constructed from the nominal model P(s), the types of uncertainties, and the uncertainty weights, with the rows and columns of G(s) arranged such that the independent diagonal additive uncertainty is the upper block of $\mathbf{\Delta}$.

Furthermore,

$$\begin{aligned}&\inf_{K(s)\in\mathbf{K}_s^n} \inf_{D(s)\in\mathbf{D}_s^{nu}} \left\| D(s) F_l(G(s), K(s)) D^{-1}(s) \right\|_\infty\\ &= \inf_{\substack{d_k(s)\\ k=2,\ldots,u}} \max_{i=1,\ldots,n} \left\{ \inf_{\Sigma_{K,ii}(s)\in\mathbf{K}_s^1} \inf_{d_{1i}(s)} \left\| \hat{D}^i(s) F_l\left(\tilde{G}^i(s), \Sigma_{K,ii}(s)\right) \right.\right.\\ &\quad \left.\left. \times \left(\hat{D}^i(s)\right)^{-1} \right\|_\infty \right\},\end{aligned} \tag{17}$$

where $\hat{G}^i(s)$ *is constructed from G(s) as defined in (13),* $\hat{D}^i = \{\operatorname{diag}(d_{1i}(s), d_2(s), \ldots, d_u(s)) | d_k(s)$ *stable and minimum phase;* $k = 1i, 2, \ldots, u\}$, *and* $\Sigma_{K,ii}(s)$ *are the diagonal elements of* $\Sigma_K(s)$. *For the case with no independent diagonal additive uncertainty, the* d_{1i} *and the corresponding infimum in (17) are dropped.*

Theorem 2 (Robust optimality with NLTV, NLTI, and LTV Δ). *Consider the assumptions of Theorem 1, except with the SLTV perturbations replaced by NLTV, NLTI, or LTV perturbations. All results of Theorem 1 hold, with the scaling matrices* $\mathbf{D}_s^{nu}$ *restricted to be constant matrices.*

For SLTV, NLTV, NLTI, and LTV full block uncertainties, Theorems 1 and 2 indicate that the robust controller synthesis problem for $K(s)$ can be reduced to n mildly coupled SISO robust controller synthesis problems for the $\Sigma_{\mathrm{K},ii}(s)$. If DK-iteration is used to design to a robust suboptimal controller, then the K step consists of n independent SISO H_∞-optimal control problems, one for each of the SISO subplants $\Sigma_{\mathrm{P},ii}(s)$ of $P(s)$. The D step is coupled, since many of the elements of $D(s)$ enter in more than one of the SISO H_∞-optimal control problems. After the DK iterations have converged to result in the final $\Sigma_{\mathrm{K},ii}(s)$, they are collected into a diagonal matrix $\Sigma_{\mathrm{K}}(s)$, and the final controller computed from (14).

The next result is for the case where the uncertainties are linear time invariant.

Theorem 3 (Robust optimality with LTI Δ). *Consider a nominal model* $P(s) = U\Sigma_P(s)V^T$, *where U and V are real orthogonal matrices and* $\Sigma_P(s)$ *is a diagonal transfer function matrix. Suppose there is any combination of uncertainties of the following forms: (i) one full block uncertainty of any type, (ii) any number of repeated diagonal multiplicative and inverse multiplicative uncertainties of the forms listed in Table 1, (iii) an independent diagonal additive uncertainty of the form (7). Then a controller of the form* $K(s) = V\Sigma_K(s)U^T$ *minimizes*

$$\sup_{s=j\omega} \mu_{\mathbf{\Delta}}(F_l(G(s), K(s))) \tag{18}$$

where the generalized plant G(s) is constructed from the nominal model P(s), the types of uncertainties, and the uncertainty weights. Furthermore,

$$\begin{aligned}&\inf_{K(s)\in\mathbf{K}_s^n} \sup_{s=j\omega} \mu_{\mathbf{\Delta}}(F_l(G(s), K(s)))\\ &= \max_{i=1,\ldots,n} \left\{ \inf_{\Sigma_{K,ii}(s)\in\mathbf{K}_s^1} \sup_{s=j\omega} \mu_{\tilde{\mathbf{\Delta}}}\left(F_l\left(\tilde{G}^i(s), \Sigma_{K,ii}(s)\right)\right) \right\},\end{aligned} \tag{19}$$

where $\tilde{\mathbf{\Delta}} = \left\{\operatorname{diag}\{\delta_k\} \middle| |\delta_k| \leqslant 1; \delta_k \in \mathcal{C}; k = 1, \ldots, u\right\}$ *and* $\tilde{G}^i(s)$ *is constructed from G(s) as defined in (13).*

For some sheet and film processes, Theorem 3 indicates that the robust controller synthesis problem for $K(s)$ can be reduced to n completely independent SISO robust controller synthesis problems for $\Sigma_{\mathrm{K},ii}(s)$, one for each of the SISO subplants $\Sigma_{\mathrm{P},ii}(s)$ of $P(s)$. To make the comparison with Theorem 1 clearer, consider the Corollary.

Corollary 1 (Robust optimality with LTI Δ). *Consider the conditions in Theorem 3, with the additional condition that* μ *is equal to its upper bound. Then*

$$\begin{aligned}&\inf_{K(s)\in\mathbf{K}_s^n} \sup_{s=j\omega} \mu_{\mathbf{\Delta}}(F_l(G(s), K(s)))\\ &= \max_{i=1,\ldots,n} \left\{ \inf_{\Sigma_{K,ii}(s)\in\mathbf{K}_s^1} \inf_{\hat{D}^i(s)\in\hat{\mathbf{D}}_s^u} \right.\\ &\quad \left. \left\| \hat{D}^i(s) F_l\left(\tilde{G}^i(s), \Sigma_{K,ii}(s)\right) \left(\hat{D}^i(s)\right)^{-1} \right\|_\infty \right\}\end{aligned} \tag{20}$$

where $\hat{\mathbf{D}}_s^u = \left\{ \hat{D}(s) \middle| \hat{D}(s) = \operatorname{diag}\{d_k(s)\};\ d_k(s)$ *stable and minimum phase;* $k = 1, \ldots, u\right\}$.

It is much simpler to solve for the controller in (20) than in (17), although (17) has fewer variables to optimize over. The SISO problems in (17) are coupled while those in (20) are completely decoupled. If DK-iteration were applied in both cases, the computation for D in (17) is coupled, while the computation for each D in (20) is not. In both cases, the K step is decoupled.

As discussed in the Background section, it is common for μ to be equal to or nearly equal to its upper bound.

The next result assumes this to generalize Theorem 1 to address a wider range of uncertainty structures.

Theorem 4 (Robust optimality with SLTV or LTI Δ). *Consider a nominal model $P(s) = U\Sigma_P(s)V^T$, where U and V are real orthogonal matrices and $\Sigma_P(s)$ is a diagonal transfer function matrix. Suppose there is any combination of uncertainties of the following forms: (i) multiple full block uncertainties and repeated diagonal multiplicative and inverse multiplicative uncertainties of the forms listed in Table 1, (ii) an independent diagonal additive uncertainty of the form (7). Assume that μ is equal to its upper bound. Then a controller of the form $K(s) = V\Sigma_K(s)U^T$ minimizes*

$$\sup_{s=j\omega} \mu_{\mathbf{\Delta}}(F_l(G(s), K(s))) = \inf_{D(s)\in\mathbf{D}_s^{nu}} \left\| D(s)F_l(G(s), K(s))D^{-1}(s) \right\|_\infty \tag{21}$$

where the generalized plant $G(s)$ is constructed from the nominal model $P(s)$, the types of uncertainties, and the uncertainty weights.

Let f refer to the number of full blocks, and d refer to the number of repeated and independent scalar diagonal blocks, and let the rows and columns of G be arranged such that all the full blocks appear as the lower blocks in $\mathbf{\Delta}$. Then

$$\inf_{K(s)\in\mathbf{K}_s^n} \inf_{D(s)\in\mathbf{D}_s^{nu}} \left\| D(s)F_l(G(s), K(s))D^{-1}(s) \right\|_\infty = \inf_{\hat{D}_f(s)\in\hat{\mathbf{D}}_s^f} \max_{i=1,\ldots,n} \left\{ \inf_{\Sigma_{K,ii}(s)\in\mathbf{K}_s^1} \inf_{\hat{D}_d^i(s)\in\hat{\mathbf{D}}_s^d} \left\| \begin{bmatrix} \hat{D}_d^i(s) & \\ & \hat{D}_f^i(s) \end{bmatrix} F_l\left(\tilde{G}^i(s), \Sigma_{K,ii}(s)\right) \begin{bmatrix} \left(\hat{D}_d^i(s)\right)^{-1} & \\ & \left(\hat{D}_f^i(s)\right)^{-1} \end{bmatrix} \right\|_\infty \right\} \tag{22}$$

where

$$\hat{\mathbf{D}}_s^f = \left\{ \mathrm{diag}\left\{d_{f,k}(s)\right\} \middle| d_{f,k}(s) \text{ stable and minimum phase; } k = 1, \ldots, f \right\}, \tag{23}$$

$$\hat{\mathbf{D}}_s^d = \left\{ \mathrm{diag}\left\{d_{d,k}(s)\right\} \middle| d_{d,k}(s) \text{ stable and minimum phase; } k = 1, \ldots, d \right\}, \tag{24}$$

$\tilde{G}^i(s)$ is constructed from $G(s)$ as defined in (13), and $\Sigma_{K,ii}(s)$ are the diagonal elements of $\Sigma_K(s)$.

The upper bound is not exactly equal to μ for many problems, at which case the assumption of Theorem 4 will be an approximation. However, this approximation is a widely accepted one, and is used in all existing off-the-shelf software for robust controller synthesis [52,53].

The next results show that, under increased restrictions on the uncertainties, it is possible to construct the multivariable robust optimal controller by solving a *single* SISO robust synthesis problem.

Theorem 5 (Robust optimality with multiplicative LTI uncertainties). *Consider the conditions of Theorem 3 with the additional conditions that: (i) all the uncertainties are multiplicative or inverse multiplicative (the full block uncertainty must correspond to a multiplicative or inverse multiplicative uncertainty), (ii) the $\Sigma_{P,ii}(s) \neq 0 \forall i$, and (iii) the $\Sigma_{P,ii}(s)$ have same right half plane (RHP) poles and zeros, $\forall i$. Define $\Sigma_{K,\overline{i}\overline{i},opt}$ as the optimal controller for any of the SISO robust synthesis problems in the right hand side of (19). Then the other $n-1$ SISO robust optimal controllers can be computed by*

$$\Sigma_{K,ii,opt}(s) = \frac{\Sigma_{K,\overline{i}\overline{i},opt}(s)\Sigma_{P,\overline{i}\overline{i}}(s)}{\Sigma_{P,ii}(s)} \tag{25}$$

Theorem 6 (Robust optimality with additive LTI uncertainties). *Consider the conditions of Theorem 3 with the additional conditions that: (i) there is one additive, inverse additive, or diagonal additive uncertainty, (ii) the $\Sigma_{P,ii}(s) \neq 0 \forall i$, and (iii) the $\Sigma_{P,ii}(s)$ have same RHP poles and zeros, $\forall i$. Define $\Sigma_{K,\overline{i}\overline{i},opt}$ as the optimal controller for any of the SISO robust synthesis problems in the right hand side of (19). Then the other $n-1$ SISO robust optimal controllers can be computed by*

$$\Sigma_{K,ii,opt}(s) = \frac{\Sigma_{K,\overline{i}\overline{i},opt}(s)\Sigma_{P,\overline{i}\overline{i}}(s)}{\Sigma_{P,ii}(s)} \tag{26}$$

Assumption (iii) of Theorems (5) and (6) is not restrictive, as sheet and film processes have the same dynamics for each pseudo-singular value, and so share the same poles and zeros. Assumption (ii) only requires that a pseudo-singular value is not precisely equal to zero so that the ratios in (25) and (26) are well-defined [note that this assumption does allow $\Sigma_{P,ii}(s)$ to have zeros]. When a pseudo-singular value is exactly zero, which occurs for some square and all non-square interactions matrices, then the corresponding SISO controller $\Sigma_{K,ii}(s)$ should be set equal to zero, since that pseudo-singular value and the corresponding columns of U and V are uncontrollable [41,50,73].

3.2. Symmetric nominal models

Somewhat broader uncertainty types than those considered in Theorems 3 and 4 are applicable to sheet and film processes with symmetric nominal models. More specifically, in this case the results hold for diagonal uncertainties of *any* of the forms listed in Table 1.

Corollary 2 (Robust optimality with LTI Δ for symmetric nominal models). *Assume the conditions of Theorem 3 with the additional condition that $U = V$. Then the results of Theorem 3 hold for any combination of uncertainties of the following forms: (i) one full block uncertainty of any type, (ii) any number of repeated diagonal uncertainties of the forms listed in Table 1, (iii) an independent diagonal additive uncertainty of the form (7).*

Corollary 3 (Robust optimality with SLTV or LTI Δ for symmetric nominal models). *Assume the conditions of Theorem 4 with the additional condition that $U = V$. Then the results of Theorem 4 hold for any combination of uncertainties of the following forms: (i) multiple full block uncertainties and repeated diagonal uncertainties of the forms listed in Table 1, (ii) an independent diagonal additive uncertainty of the form (7).*

3.3. Remarks

All of the results in this section yield controllers that are superoptimal [74–76], that is, the H_∞ norm is minimized in n directions. This is in contrast to the H_∞ controllers computed by commercial software packages, which only minimize the H_∞ norm in the worst-case direction [52]. From a practical point of view, this means that the superoptimal H_∞ will give much better closed loop response to most disturbances, although it will have the same overall H_∞-norm as a non-superoptimal controller.

The controller design theorems in Sections 3 and 4 yield controllers of the form $K(s) = V\Sigma_K(s)U^T$. The robustness for the overall system is by minimizing the robustness margin for the SISO control problems. The pseudo-singular values of P_{CD} that are nearly zero cannot be reliably controlled [41,50]. Separate relationships for the SISO controllers $\Sigma_{K,ii}(s)$ are given depending on the magnitude of $\Sigma_{P,ii}(0)$. For $\Sigma_{P,ii}(0)$ close to zero the corresponding SISO controller $\Sigma_{K,ii}(s)$ is set equal to zero. Otherwise, $\Sigma_{K,ii}(s)$ is computed according to the appropriate theorem from Section 3 or 4. The parameter ϵ defines the boundary between controllable and effectively uncontrollable pseudo-singular values, and can be computed from experimental data using a Monte Carlo algorithm [50]. The SISO robust control problems associated with the uncontrollable pseudo-singular values should not be included in the robustness margin calculations, that is, the multivariable performance specification is only applied to the controllable plant directions.

It was shown in previous work that explicit constraint handling is not always needed when robust control design methods are used [41,50,77]. This is because directions corresponding to low gains are not manipulated by the SVD controller. Also, designing of the SVD controller to be robust prevents overly large dynamic excursions in the manipulated variables. A recent paper provides explicit criteria for determining when constraint-handling is necessary [78]. In cases where constraint-handling is needed, any of the well-established multivariable anti-windup procedures can be applied [79–83]. This results in a simple controller implementation (see the end of the next section for more details).

When used for controller design via DK-iteration, the theorems in Section 3 may yield controllers of unacceptably high order. In practice, low order controllers are often desirable. Low order controllers can be achieved by using model reduction techniques to reduce the controller order or by fixing the controller order in the synthesis step. The theorems provided above are suitable for the former approach, while the theorems in the next section are suitable for the latter. Fixing the controller order in the synthesis step leads to further simplifications in robust controller design. As will be seen in the examples section, this simplification can be with a small loss in closed loop performance.

4. Algorithms for low order robust controller design

The results of the previous section can be used to compute robust suboptimal controllers using the DK-iteration method. It is unlikely, however, that any controller design method, irrespective of complexity, will produce a controller that gives precisely the desired stability and performance for all disturbances and all operating conditions (for example, during startup or grade changes). This motivates the development of controllers which have parameters that can be tuned (or detuned) on-line when necessary. Secondly, controllers produced by DK-iteration tend to have very high order, while low order controllers are easier to implement.

That an SVD controller optimizes robust performance for a variety of uncertainty types suggests that such low order tunable controllers should be selected to have the SVD structure. In this way, the low order tunable controller will have the optimal directionality. The algorithms for low order robust controller design for the LTI uncertainty types considered in Theorem 3 require less computation and are presented first, followed by the algorithms for the uncertainty types considered in Theorems 1, 2, and 4–6.

4.1. Low order robust controller design for LTI uncertainty

For the LTI uncertainty types covered by Theorem 3, the following result shows that any SVD controller (14) decouples the multivariable robust control synthesis into independent SISO control problems.

Corollary 4 (Robustness analysis with LTI Δ). *Consider the conditions and notation in Theorem 3. Then*

$$\begin{aligned}&\sup_{s=j\omega}\mu_{\mathbf{\Delta}}(F_l(G(s),K(s)))\\&\quad=\sup_{s=j\omega}\max_i\left\{\mu_{\hat{\mathbf{\Delta}}}\left(F_l\left(\tilde{G}^i(s),\Sigma_{K,ii}(s)\right)\right)\right\}\\&\quad=\max_i\left\{\sup_{s=j\omega}\mu_{\hat{\mathbf{\Delta}}}\left(F_l\left(\tilde{G}^i(s),\Sigma_{K,ii}(s)\right)\right)\right\}\end{aligned}\tag{27}$$

holds for any controller of the form $K(s)=V\Sigma_K(s)U^T$.

The robustness for the overall system is optimized by minimizing the robustness margin for the SISO control problems. A low order multivariable controller can be designed by designing low order SISO controllers $\Sigma_{\mathrm{K},ii}(s)$. The controller $\Sigma_{\mathrm{K},ii}(s)$ for each SISO problem can be designed by any robust controller design method; here we describe the use of internal model control (IMC) tuning [46] for scalar dynamics described by first order plus time delay (this is by far the most commonly used model for describing sheet and film process dynamics [10,84], for more complex models see [46]:

$$p(s)=\frac{\mathrm{e}^{-\theta s}}{\tau s+1}.\tag{28}$$

Without loss of generality, the steady-state gain of $p(s)$ has been scaled so that $p(0)=1$.

The internal model control-proportional integral derivative (IMC-PID) control form is

$$\Sigma_{\mathrm{K},ii}(s)=\begin{cases}\dfrac{1}{\Sigma_{\mathrm{P},ii}(0)}\cdot\dfrac{1+\tau_{\mathrm{D}}s+\dfrac{1}{\tau_I s}}{\tau_{\mathrm{F},i}s+1}\cdot\dfrac{2\tau+\theta}{2(\lambda_i+\theta)} & \text{if } \left|\Sigma_{\mathrm{P},ii}(0)\right|>\epsilon\\ 0 & \text{if } \left|\Sigma_{\mathrm{P},ii}(0)\right|\leqslant\epsilon\end{cases}\tag{29}$$

$$\tau_{\mathrm{I}}=\tau+\frac{\theta}{2};\quad \tau_{\mathrm{D}}=\frac{\tau\theta}{2\tau+\theta};\quad \tau_{\mathrm{F},i}=\frac{\lambda_i\theta}{2(\lambda_i+\theta)};\tag{30}$$

where ϵ is the tolerance as described in Section 3.3. If a lower order controller is desired, the IMC-PI form is

$$\Sigma_{\mathrm{K},ii}(s)=\begin{cases}\dfrac{\tau}{\lambda_i\Sigma_{\mathrm{P},ii}(0)}\left(1+\dfrac{1}{\tau s}\right) & \text{if } \left|\Sigma_{\mathrm{P},ii}(0)\right|>\epsilon\\ 0 & \text{if } \left|\Sigma_{\mathrm{P},ii}(0)\right|\leqslant\epsilon\end{cases}\tag{31}$$

The SISO controllers $\Sigma_{\mathrm{K},ii}(s)$ are stacked up as the diagonal elements of a matrix $\Sigma_K(s)$, with the overall SVD controller computed from (14). The number of states in $K(s)$ constructed using the IMC-PID form (29) is less than or equal to $2n$, whereas using the IMC-PI form (31) results in $K(s)$ having not greater than n states.

The IMC tuning parameters λ_i can be selected either as fast as possible while maintaining robust stability [43], or to maximize robust performance. If the λ_i are used to optimize robust performance, then care must be taken to ensure that the combined uncertainty-performance description is not too conservative.

The IMC tuning rules used in (29) and (31) are known to provide poor load disturbance suppression for processes which have the open loop time constant τ larger than the desired closed loop time constant λ [85]. For most sheet and film processes, the time delay dominates the open loop dynamics and τ is relatively small, so that λ will be greater than τ for a robust control system [10]. For those rare sheet and film processes where robust performance allows $\lambda<\tau$, the IMC-tuning rules used in (29) should be replaced by the modified IMC-PID rules [86].

4.2. Low order robust controller design for the SLTV, NLTV, NLTI, LTV uncertainties

Here we consider low order controller design for the uncertainty types considered by Theorems 1, 2, and 4.

Corollary 5 (Robustness analysis with SLTV Δ). *Consider the conditions and notation in Theorem 1. Then*

$$\begin{aligned}&\inf_{D(s)\in\mathbf{D}_s^{nu}}\left\|D(s)F_l(G(s),K(s))D^{-1}(s)\right\|_\infty\\&\quad=\inf_{\substack{d_k(s)\\k=2,\ldots,u}}\max_i\\&\qquad\left\{\inf_{d_{1i}(s)}\left\|\hat{D}^i(s)F_l\left(\tilde{G}^i(s),\Sigma_{K,ii}(s)\right)\left(\hat{D}^i(s)\right)^{-1}\right\|_\infty\right\},\end{aligned}\tag{32}$$

holds for any controller of the form $K(s)=V\Sigma_K(s)U^T$

Corollary 6 (Robustness analysis with NLTV, NLTI, and LTV Δ). *Consider the conditions and notation in Theorem 2. Then*

$$\begin{aligned}&\inf_{D\in\mathbf{D}^{nu}}\left\|DF_l(G(s),K(s))D^{-1}\right\|_\infty\\&\quad=\inf_{\substack{d_k\\k=2,\ldots,u}}\max_i\left\{\inf_{d_{1i}}\left\|\hat{D}^iF_l\left(\tilde{G}^i(s),\Sigma_{K,ii}(s)\right)\left(\hat{D}^i\right)^{-1}\right\|_\infty\right\},\end{aligned}\tag{33}$$

holds for any controller of the form $K(s) = V\Sigma_K(s)U^T$, *where* $\mathbf{D}^{nu}$ *is the set of constant matrices with the same structure as* $\mathbf{D}_s^{nu}$.

Corollary 7 (Robustness analysis with SLTV or LTI Δ). *Consider the conditions and notation in Theorem 4. Then*

$$\inf_{D(s)\in\mathbf{D}_s^{nu}} \left\| D(s)F_l(G(s), K(s))D^{-1}(s)\right\|_\infty = \inf_{\hat{D}_f(s)\in\hat{\mathbf{D}}_s^f} \max_i \left\{ \inf_{\hat{D}_d^i(s)\in\hat{\mathbf{D}}_s^d} \left\| \begin{bmatrix} \hat{D}_d^i(s) & \\ & \hat{D}_f^i(s) \end{bmatrix} F_l\left(\tilde{G}^i(s), \Sigma_{K,ii}(s)\right) \begin{bmatrix} \left(\hat{D}_d^i(s)\right)^{-1} & \\ & \left(\hat{D}_f^i(s)\right)^{-1} \end{bmatrix} \right\|_\infty \right\} \tag{34}$$

holds for any controller of the form $K(s) = V\Sigma_K(s)U^T$.

If μ defines a robust performance objective (with Δ_{IO} representing the performance specification), then low order tunable controllers can be designed by solving the appropriate optimization problem (32), (33), or (34), with the $\Sigma_{\mathrm{K},ii}(s)$ restricted to be a low order controllers, such as (29) or (31). A procedure similar to DK-iteration can be used to compute a high quality suboptimal solution to the nonconvex optimization problems. In the K step, the H_∞ optimization over the controller is replaced by an optimization over the λ_i. The optimizations over the λ_i are independent, and can be easily automated. Moreover, since the SISO control problems are nearly decoupled, each λ_i behaves similarly as in tuning a SISO IMC controller. In particular, for reasonable uncertainty and performance weights, the SISO robust performance objectives will be large when λ_i is either small (poor stability robustness) or large (poor performance). Extensive experience with IMC tuning of time delay processes indicates that the optimization of the μ upper bound over λ_i will usually have a unique minimum. Also, given that the $\Sigma_{\mathrm{P},ii}(s)$ have the same dynamics with a nearly continuous range of gains from low to high singular values of P_{CD}, the minimizing λ_i for one optimization can be used as an initial condition for the adjacent optimization (λ_{i+1}). In the D step, fitting the D-scale at each frequency to a transfer function is unnecessary, since the IMC-PI/PID $\Sigma_{\mathrm{K},ii}(s)$'s are not computed from the transfer functions $d_i(s)$, but only from their values at each frequency. Thus the modified DK-iteration procedure avoids both the D-fitting and the H_∞-synthesis procedures, which are the steps in standard DK-iteration that can cause numerical inaccuracies [23].

An alternative to the modified DK-iteration procedure will be to directly optimize the overall μ upper bound over the λ_i using a generic optimization procedure. This would require re-computing the D-scales every time the λ_i are updated. The modified DK-iteration procedure, on the other hand, requires a limited number of D-scale computations if properly initialized. The independent design procedure in Section 4.1 can be used to initialize the algorithm.

If the μ robustness measure defines a robust stability objective (without inverse multiplicative input or output uncertainties), then it is desired to select the IMC tuning parameters λ_i as fast as possible while maintaining robust stability. This optimization problem can be posed as:

$$\inf_{\hat{D}(s)\in\hat{\mathbf{D}}_s} \max_i \left\{ \inf_{\Sigma_{\mathrm{K},ii}(s)\in\mathbf{K}_s^1} \left| \left\| \hat{D}F_l\left(\tilde{G}^i(s), \Sigma_{\mathrm{K},ii}(s)\right)\hat{D}^{-1}\right\|_\infty - 1\right| \right\}. \tag{35}$$

A modified DK-iteration procedure similar to that described in the previous section can be used to solve this optimization problem. The robust stability objective is achievable if and only if the optimal value of the objective function in (35) is zero (in practice, some tolerance close to zero is used). If the optimal value of the objective function in (35) is greater than zero, then the uncertainty set must be reduced (for example, through increased data collection [77]).

4.3. Low order robust controller design for multiplicative or additive LTI uncertainties

Here we consider low order controller design for the uncertainty types considered by Theorems 5 and 6.

Corollary 8 (Robustness analysis with multiplicative LTI uncertainties). *Consider the conditions and notation in Theorem 5. Then*

$$\sup_{s=j\omega} \mu_{\mathbf{\Delta}}(F_l(G(s), K(s))) = \max_i \left\{ \sup_{s=j\omega} \mu_{\tilde{\mathbf{\Delta}}}\left(F_l\left(\tilde{G}^i(s), \Sigma_{K,ii}(s)\right)\right) \right\}, \tag{36}$$

holds for any controller of the form $K(s) = V\Sigma_K(s)U^T$. *Furthermore, all SISO controllers* $\Sigma_{K,ii}(s)$ *can be constructed from a single SISO controller design problem. Let the low order controller designed be denoted* $\bar{i}$. *The other controllers are given by*

$$\Sigma_{k,ii,opt}(s) = \frac{\Sigma_{K,\bar{i}\bar{i},opt}(s)\Sigma_{P,\bar{i}\bar{i}}(s)}{\Sigma_{P,ii}(s)} \tag{37}$$

Corollary 9 (Robustness analysis with additive LTI uncertainties). *Consider the conditions and notation in Theorem 6. Then*

$$\sup_{s=j\omega} \mu_{\mathbf{\Delta}}(F_l(G(s), K(s))) = \max_i \left\{ \sup_{s=j\omega} \mu_{\tilde{\mathbf{\Delta}}}\left(F_l\left(\tilde{G}^i(s), \Sigma_{K,ii}(s)\right)\right)\right\}, \tag{38}$$

holds for any controller of the form $K(s) = V\Sigma_K(s)U^T$. *Furthermore, all SISO controllers* $\Sigma_{K,ii}(s)$ *can be constructed from a single SISO controller design problem. Let the low order controller designed be denoted* $\bar{i}$. *The other controllers are given by*

$$\Sigma_{K,ii,opt}(s) = \frac{\Sigma_{K,\bar{i}\bar{i},opt}(s)\Sigma_{P,\bar{i}\bar{i}}(s)}{\Sigma_{P,ii}(s)} \tag{39}$$

For the uncertainty descriptions treated by Theorems 5 and 6, a controller of the form $K(s) = V\Sigma_K(s)U^T$ decouples the process into n independent SISO problems. If low order controllers are desired, $\Sigma_{K,ii}(s)$ may be selected to have the form of (29) or (31) and only one $\Sigma_{K,ii}(s)$ needs to be synthesized. The other controllers are constructed as multiples of that one controller.

4.4. Implementation

SVD controllers (14) can be implemented in the form of a static decoupler U^T in series with a diagonal dynamics matrix $\Sigma_K(s)$ in series with another static decoupler V. The implementation for the PI and PID SVD controllers is particularly simple — the technology for implementing static decouplers and noninteracting PI/PID controllers has been available for over two decades.

Sheet and film processes usually have min-max and second-order spatial constraints on their manipulated variables to prevent excessive stresses (such as in a die or slice lip) or flow instabilities [10]. These constraints can be addressed by applying any of the well-established multivariable anti-windup procedures [79–83] to the SVD controllers. The SVD controllers with anti-windup are implementable in real time on large scale sheet and film processes using existing hardware [10].

5. Applications

Here the robust controller design theorems developed in the previous sections are applied to a model developed from industrial data that captures many of the realities of an industrial paper machine.

5.1. Paper machine model

Many of the features of this model are common to other sheet and film processes (e.g. constant interaction matrix, scalar dynamics, edge effects). The model was developed from industrial identification data reported by Heaven et al. [84] who studied the slice lip to weight profile transfer function of a fine paper machine (see [32] for details):

$$y(s) = \frac{e^{-2s}}{0.533s + 1} Pu(s). \tag{40}$$

where y is the vector of measurements of basis weight (in lbs), u is the vector of actuator positions (in mils), and P_{CD} is the interaction matrix (with units of lbs/mil). The actuators are motors which change the slice lip openings and the weight profile is measured by a scanning sensor at the reel of the machine. The interaction matrix P_{CD} is of the form

$$P_{CD} = C\Lambda_G \tag{41}$$

where the matrix

$$C = \begin{bmatrix}
c_2 & c_7 & c_{12} & \cdots & c_{37} & 0 & \cdots & \cdots & 0 & 0 \\
c_1 & c_6 & c_{11} & \cdots & c_{36} & 0 & \cdots & \cdots & 0 & 0 \\
c_0 & c_5 & c_{10} & \cdots & c_{35} & 0 & \cdots & \cdots & 0 & 0 \\
c_1 & c_4 & c_9 & \cdots & c_{34} & 0 & \cdots & \cdots & 0 & 0 \\
c_2 & c_3 & c_8 & \cdots & c_{33} & c_{38} & 0 & \cdots & 0 & 0 \\
c_3 & c_2 & c_7 & \cdots & c_{32} & c_{37} & 0 & \cdots & 0 & 0 \\
c_4 & c_1 & c_6 & \cdots & c_{31} & c_{36} & 0 & \cdots & 0 & 0 \\
c_5 & c_0 & c_5 & \cdots & c_{30} & c_{35} & 0 & \cdots & 0 & 0 \\
c_6 & c_1 & c_6 & \cdots & c_{29} & c_{34} & 0 & \cdots & 0 & 0 \\
c_7 & c_2 & c_7 & \cdots & c_{28} & c_{33} & c_{38} & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
c_{35} & c_{30} & c_{25} & \cdots & c_0 & c_5 & c_{10} & \cdots & \vdots & \vdots \\
c_{36} & c_{31} & c_{26} & \cdots & c_1 & c_4 & c_9 & \cdots & \vdots & \vdots \\
c_{37} & c_{32} & c_{27} & \cdots & c_2 & c_3 & c_8 & \cdots & \vdots & \vdots \\
c_{38} & c_{33} & c_{28} & \cdots & c_3 & c_2 & c_7 & \cdots & \vdots & \vdots \\
0 & c_{34} & c_{29} & \cdots & c_4 & c_1 & c_6 & \cdots & \vdots & \vdots \\
0 & c_{35} & c_{30} & \cdots & c_5 & c_0 & c_5 & \cdots & \vdots & \vdots \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\
\vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & c_2 & c_3 \\
\vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & c_3 & c_2 \\
\vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & c_4 & c_1 \\
\vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & c_5 & c_0 \\
\vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & c_6 & c_1 \\
0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & c_7 & c_2
\end{bmatrix} \tag{42}$$

$$c_i = \begin{cases} -0.1322i + 1 & i = 0, 1, \ldots, 10 \\ \left.\begin{array}{l} -1.3178 \times 10^{-8} i^6 \\ +2.1221 \times 10^{-6} i^5 \\ -1.4006 \times 10^{-4} i^4 \\ +4.8607 \times 10^{-3} i^3 \\ -9.4066 \times 10^{-2} i^2 \\ +0.97362i - 4.3108 \end{array}\right\} & i = 11, 12, \ldots, 38 \end{cases} \tag{43}$$

represents the interactions between 130 actuators and 650 downstream measurement locations, and the diagonal matrix Λ_G captures the variation of the actuator gains across the machine:

$$\Lambda_{G,ii} = \begin{cases} \left.\begin{array}{l} 7.4167 \times 10^{-4}(i^2) \\ -2.1971 \times 10^{-3} i \\ +0.3015 \end{array}\right\} & i = 1, 2, \ldots, 10 \\ \left.\begin{array}{l} -1.1392 \times 10^{-6} i^6 \\ +4.1401 \times 10^{-8} i^5 \\ -5.9244 \times 10^{-6} i^4 \\ +4.2284 \times 10^{-4} i^3 \\ -1.5624 \times 10^{-2} i^2 \\ +0.2751i - 1.2015 \end{array}\right\} & i = 11, 12, \ldots, 43 \\ 0.4692 & i = 44, 45, \ldots, 87 \\ \Lambda_{G,130-i+1,130-i+1} & i = 88, 89, \ldots, 130 \end{cases} \tag{44}$$

We consider the case where there is uncertainty in both the input and the output of the process (see Fig. 4). This uncertainty includes inaccuracies in the actuators and sensors, as well as uncertainty associated with the actual process. The operators Δ_I and Δ_O are unity norm bounded and assumed to be linear time invariant (LTI).

The magnitude of the uncertainty is set by the weights W_I and W_O. Each uncertainty weight (W_I, W_O) was chosen to represent up to 10% steady state error and up to 100% dynamic error. The uncertainty weights also cover model error due to replacing the time delay with a 3rd order Pade approximation.

The performance weight is selected to ensure less than 0.4% steady-state error and a closed loop time constant of $\tau_p = 5$ min. Eq. (5) indicates that the maximum disturbance amplification will be less than 2 at all frequencies, and that the bandwidth of the closed loop system will be at least 0.2. Rearranging the block diagram in Fig. 4 and including a performance block results in the generalized plant matrix

$$G = \begin{bmatrix} 0 & 0 & 0 & -W_I \\ W_O P & 0 & 0 & -W_O \\ W_P P & W_P & -W_P & -W_P P \\ P & I & -I & -P \end{bmatrix} \tag{45}$$

where

$$W_I = W_O = \frac{0.1(10s+1)}{s+1} I \tag{46}$$

$$W_P = \frac{0.5(\tau_p s + 1)}{\tau_p s + 0.002} I \tag{47}$$

$$\tau_p = 5 \tag{48}$$

Controllers designed to be robust to the uncertainty description will also be insensitive to measurement noise, as the uncertainty specifications require a rolloff of the complementary sensitivity function.

5.1.1. The inadequacy of commercial software

The commercial software packages for designing robust controllers are the Matlab μ-toolbox [52] and the Robust Control Toolbox [53]. It is impossible to even form the G matrix (45) for the large scale paper machine in Matlab on a Sparc Ultra 2200 computer with 64 MB of RAM and 240 MB of swap space — the computer runs out of memory.

It is instructive however to estimate the time required to design a robust controller using the standard DK-iteration procedure [43,46,52,53,87] if it were possible to perform these calculations. For only 20 actuators, one DK-iteration step took 77 min. One H_∞ synthesis step took 20 min, μ analysis took 57 min for 50 frequency

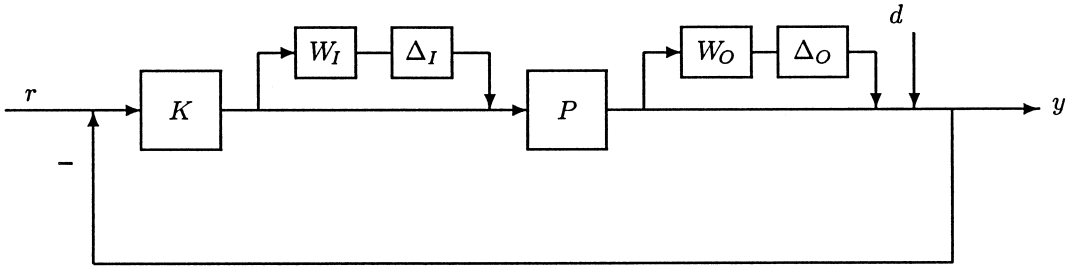

Fig. 4. Block diagram with both input and output uncertainty.

points, and the D-fitting step took 2 s. Assuming that scaling up to 130 actuators follows an $O(n^3)$ increase in computation time, and that six DK-iteration steps are necessary, then DK-iteration for 130 actuators would require more than 2000 h of computation. Note that assuming an $O(n^3)$ increase in computation time is a lower bound — it is likely that a higher order would occur in practice. For example, for 40 actuators the μ analysis step took more than 30 min per frequency point.

The conservative timing estimates above are for the case where the uncertainties are all full block. DK iteration for repeated scalar uncertainties is not implemented in the commercial software packages. If it were implemented, the D-fitting step for repeated scalar uncertainties would take much longer, as in this case the number of degrees of freedom to be computed grows very rapidly (quadratically) as a function of plant input-output dimension. This high computational expense is likely why the D-fitting step for repeated scalar uncertainties is not implemented in commercial packages.

Even if a supercomputer with GBs of RAM and/or swap space were available, and if time to compute the robust optimal controller was not a concern, the paper machine control problem has a large enough dimensionality that the DK iteration algorithm would likely produce highly suboptimal results (the algorithm would have difficulty converging). This behavior has been demonstrated on much smaller problems in past work [23]. Also, the resulting controller would be of very high order and would be expensive to implement.

This motivates the robust controller design procedures presented in this manuscript. The dimensionality reduction theorems given here allow robust controllers to designed for systems in which no other design techniques are suitable. The total computation time of the following algorithms is on the order of minutes on a Sun Workstation or Pentium II.

5.2. Full order controller design for repeated scalar input and output uncertainties

If Δ_{I} and Δ_{O} are treated as being repeated scalar, then the input–output uncertainty description satisfies the conditions of Theorem 5, and the robust controller design problem reduces to the design of a single $\Sigma_{\mathrm{K},ii}(s)$. The n lower dimensional transfer functions $\tilde{G}^i(s)$ are constructed as shown in (13):

$$\tilde{G}^i(s) = \begin{bmatrix} 0 & 0 & 0 & -w_{\mathrm{I}}(s) \\ w_{\mathrm{O}}(s)\Sigma_{\mathrm{P},ii}(s) & 0 & 0 & -w_{\mathrm{O}}(s)\Sigma_{\mathrm{P},ii}(s) \\ w_{\mathrm{P}}(s)\Sigma_{\mathrm{P},ii}(s) & w_{\mathrm{P}}(s) & -w_{\mathrm{P}}(s) & -w_{\mathrm{P}}(s)\Sigma_{\mathrm{P},ii}(s) \\ \Sigma_{\mathrm{P},ii}(s) & 1 & -1 & -\Sigma_{\mathrm{P},ii}(s) \end{bmatrix};$$

$$\hat{\Delta} = \begin{bmatrix} \delta_I & 0 & 0 \\ 0 & \delta_O & 0 \\ 0 & 0 & \delta_{\mathrm{P}} \end{bmatrix}. \tag{49}$$

The multivariable robust control problem decouples into independent SISO robust control problems as defined in (19), where

$$F_l\left(\tilde{G}^i(s), \Sigma_{\mathrm{K},ii}(s)\right) =$$

$$\begin{bmatrix} \dfrac{-w_{\mathrm{I}}(s)\Sigma_{\mathrm{K},ii}(s)\Sigma_{\mathrm{P},ii}}{1+\Sigma_{\mathrm{K},ii}(s)\Sigma_{\mathrm{P},ii}(s)} & \dfrac{-w_{\mathrm{I}}(s)\Sigma_{\mathrm{K},ii}(s)}{1+\Sigma_{\mathrm{P},ii}(s)\Sigma_{\mathrm{K},ii}(s)} & \dfrac{w_{\mathrm{I}}(s)\Sigma_{\mathrm{K},ii}(s)}{1+\Sigma_{\mathrm{P},ii}(s)\Sigma_{\mathrm{K},ii}(s)} \\ \dfrac{w_{\mathrm{O}}(s)\Sigma_{\mathrm{P},ii}(s)}{1+\Sigma_{\mathrm{K},ii}(s)\Sigma_{\mathrm{P},ii}(s)} & \dfrac{-w_{\mathrm{O}}(s)\Sigma_{\mathrm{P},ii}(s)\Sigma_{\mathrm{K},ii}(s)}{1+\Sigma_{\mathrm{K},ii}(s)\Sigma_{\mathrm{P},ii}(s)} & \dfrac{-w_{\mathrm{O}}(s)\Sigma_{\mathrm{P},ii}(s)\Sigma_{\mathrm{K},ii}(s)}{1+\Sigma_{\mathrm{K},ii}(s)\Sigma_{\mathrm{P},ii}(s)} \\ \dfrac{w_{\mathrm{P}}(s)\Sigma_{\mathrm{P},ii}(s)}{1+\Sigma_{\mathrm{K},ii}(s)\Sigma_{\mathrm{P},ii}(s)} & \dfrac{w_{\mathrm{P}}(s)}{1+\Sigma_{K,ii}(s)\Sigma_{\mathrm{P},ii}(s)} & \dfrac{-w_{\mathrm{P}}(s)}{1+\Sigma_{\mathrm{K},ii}(s)\Sigma_{\mathrm{P},ii}(s)} \end{bmatrix} \tag{50}$$

Since the number of uncertainties in the SISO problem (49) is less than four, μ is equal to its upper bound. DK-iteration can be used to compute a μ-suboptimal solution for the SISO controller design problem.

Using the μ-toolbox [52], DK-iteration was applied to one of the SISO robust control problems defined in (49). The frequency-dependent D scales, $D(s)$, were allowed to be up to third order. The DK-iteration procedure was stopped after six steps, at which point the maximum value of μ was 0.96. DK-iteration for this SISO system required about 10 s of computation per iteration on Sparc Ultra 2200. The state space matrices for the SISO controller are given in the Appendix. The other robust SISO controllers $\Sigma_{\mathrm{K},jj}(s)$ were constructed as shown in (25), and the robust multivariable controller constructed as shown in (14). The value of μ for the SISO problem is equal to μ for the multivariable system (see Fig. 5).

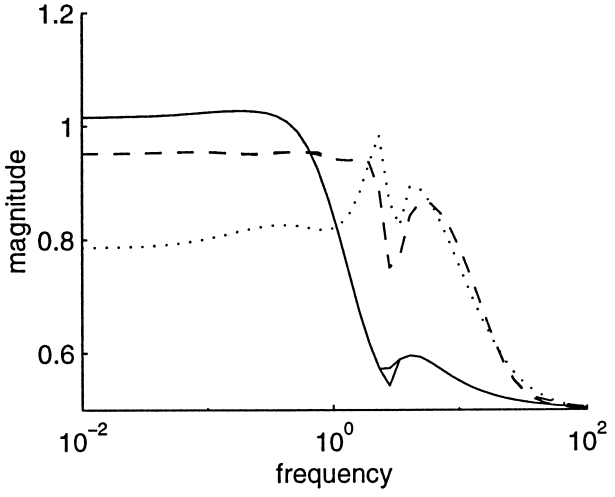

Fig. 5. μ as a function of frequency for the full order controller with repeated scalar uncertainties (dashed), the low order controller with repeated scalar uncertainties (solid), and the full order controller with full block uncertainties (dotted).

Fig. 7 is the closed loop response to the disturbance in Fig. 6, where the controller attempts to control all of the pseudo-singular values of the paper machine. For industrial paper machines, some of the pseudo-singular values are usually uncontrollable (see Section 4.1). The controller used in the closed loop simulations shown in Fig. 8 does not attempt to control the pseudo-singular values of the paper machine corresponding to the singular values of the interaction matrix P_{CD} (42) smaller than $\epsilon = 0.12$ (algorithms to compute ϵ directly from experimental data are provided in [50]). In this case, μ for robust performance applies only for the controllable pseudo-singular values. The loss in closed loop time domain performance in not controlling the smallest pseudo-singular values is negligible. Both controllers are insensitive to high frequency measurement noise (measurement noise was not included in the time domain simulations so that the details of the closed loop responses would be clear).

Figs. 9 and 10 illustrate the robustness of the control to model uncertainties. The dynamic perturbations were selected to be time delays because the dynamics associated with time delays are known to be particularly difficult to handle by most control systems. The closed loop responses demonstrate similar robustness for a wide variety of other model perturbations [88].

5.3. Low order robust controller design

Here the same uncertainty description for the paper machine is assumed as in Section 5.2, but the SISO robust controllers are designed to be in the IMC-PI form (31).

For the selected input–output uncertainty description with the SVD control structure, Corollary 8 holds. Only

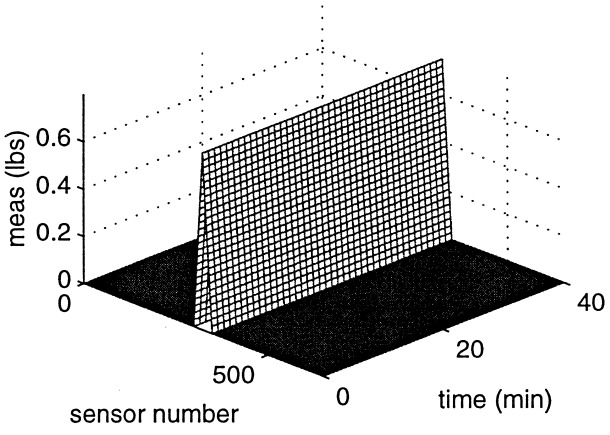

Fig. 6. The process disturbance, which represents a streak down the middle of the paper machine. Such disturbances are commonly encountered in industrial paper machines.

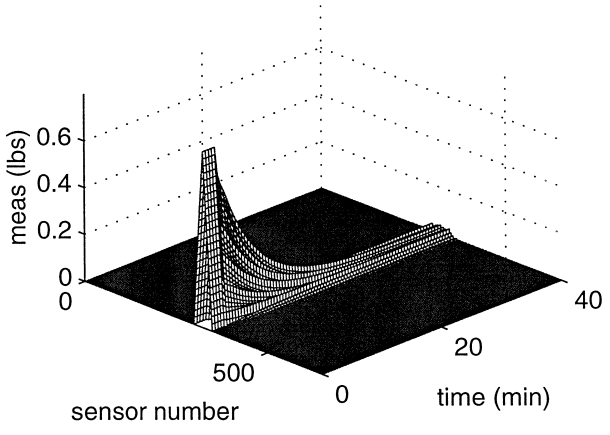

Fig. 7. The closed loop response of the paper machine to the process disturbance in Fig. 6 for the nominal model. The controller was designed via DK-iteration to control all the pseudo-singular values of the process.

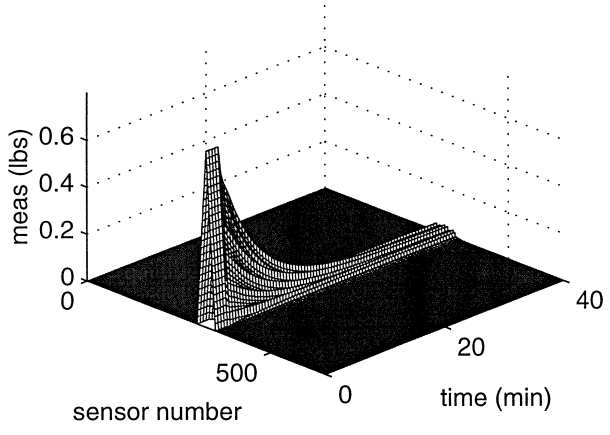

Fig. 8. The closed loop response of the paper machine to the process disturbance in Fig. 6 for the nominal model. The controller was designed via DK-iteration to control a subset of the pseudo-singular values of the process.

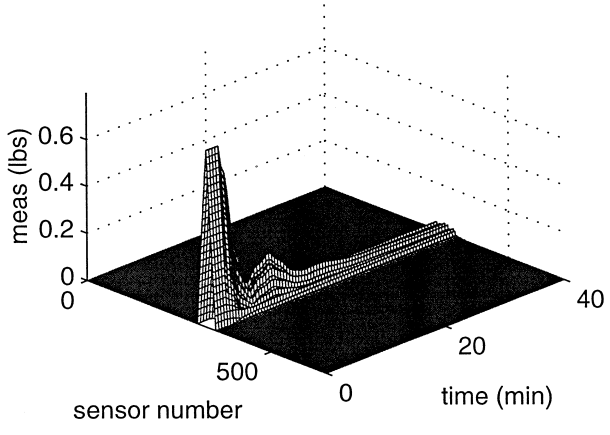

Fig. 9. The closed loop response of the paper machine to the process disturbance in Fig. 6 for repeated scalar input uncertainty Δ_I and output uncertainty Δ_O equal to a 3rd order Pade approximation for a time delay of 2 min times the identity matrix. The controller was designed via DK-iteration to control a subset of the pseudo-singular values of the process.

one SISO PI controller $\Sigma_{K,ii}(s)$ needs to be designed. The multivariable robustness margin reduces to the calculation of μ for a single 3×3 transfer function matrix, which took less than 0.2 seconds for each frequency on a Sparc Ultra 2200. The *single* IMC tuning parameter $\lambda = \lambda_i$ was selected to minimize the value of μ. This resulted in $\lambda_i = 8.26$ min, with a multivariable μ value of 1.028 (see Fig. 5). A rescaling of the uncertainty and performance weights by 3% would give $\mu < 1$. The low order controller has μ greater than one for the given uncertainty and performance weights because restricting the controller to be low order is suboptimal. The multivariable SVD controller was constructed from the SISO controller as described in the last section.

The multivariable closed loop responses to a variety of perturbations are similar to those for the full order controller in Figs. 7–10 [88]. The low order controller required less computations to design, and has a simple tuning parameter, λ, which can be re-tuned on-line should the uncertainty description have been too optimistic or too conservative.

5.4. Full-block input and output uncertainties

Now let Δ_I and Δ_O be full blocks. In this case, since there are less than four full blocks, the robustness margins for LTI and SLTV are equal [56] and Theorem 1 applies. The multivariable robust control synthesis problem can be replaced by the coupled SISO problems in (17) with $\tilde{G}^i$ defined by (49).

If DK-iteration is used to compute a μ-suboptimal controller, only two transfer functions $d_1(s)$ and $d_2(s)$ need to be fitted in the D-step. It is more expensive to compute each $d_k(s)$ than in the repeated scalar case (5.2), since each $d_k(s)$ appears in multiple SISO H_∞-synthesis problems. All the $\tilde{G}^i(s)$ and $\Sigma_{K,ii}(s)$ are used to compute the $d_1(s)$ and $d_2(s)$ for the next iteration. The D-scale computation could be posed as a highly structured Linear Matrix Inequality (LMI) optimization (for more details, see [66,89,90]), the dimension of which is equal to the dimension of the original multivariable robust control synthesis problem. The K-step, on the other hand, consists of independent SISO robust controller synthesis problems.

The wide range of gains of the pseudo-singular values in the coupled SISO problems caused the optimization over the dynamics of d_1 and d_2 to provide negligible improvement over constant D-scales. Optimization over even a constant d_2 had negligible effect as well. This is not surprising since the robustness of the overall system is much more sensitive to full-block input uncertainty than full-block output uncertainty when the plant is poorly conditioned, as it is in this case. This restricted the number of degrees of freedom in the D-scales enough that it was necessary to relax the performance weight to $\tau_p = 8$ in order to get $\mu < 1$. The optimization over the D-scales gave $d_1 = 0.15$ and $d_2 = 1$, with $\mu = 0.99$ (see Fig. 5). With the D-scales fixed, about 10 s was required to compute the H_∞ controller for each SISO subproblem on a Sparc Ultra 2200.

The nominal closed loop response to the disturbance in Fig. 6 was similar to those in the previous sections. A closed loop response is shown in Fig. 11 for the paper machine model with full block input and output perturbations. That is, the input and output uncertainties are full random matrices with norm one. Attempts to simulate a process with anti-diagonal time delay perturbations failed because the computer did not have enough memory to create the uncertain transfer function. It was possible to create the transfer function matrix in Matlab for a constant anti-diagonal perturbation, but the time domain simulation would not converge.

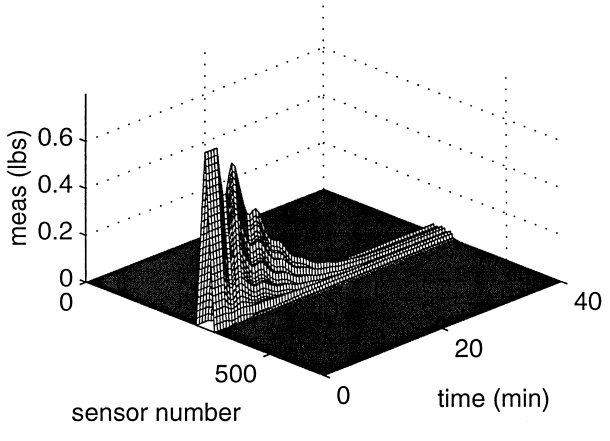

Fig. 10. The closed loop response of the paper machine to the process disturbance in Fig. 6 for repeated scalar input uncertainty Δ_I and output uncertainty Δ_O equal to minus one times a 3rd order Pade approximation for a time delay of 2 min times the identity matrix. The controller was designed via DK-iteration to control a subset of the pseudo-singular values of the process.

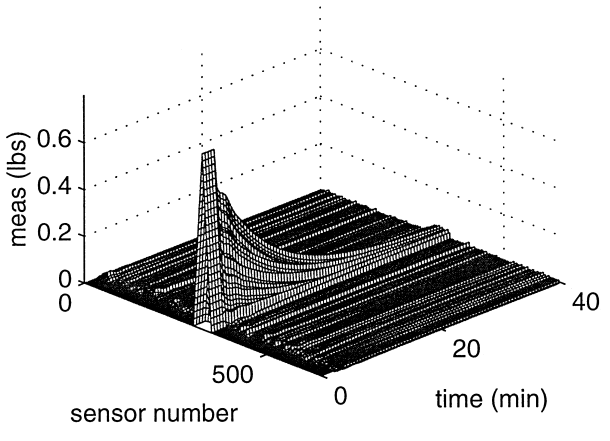

Fig. 11. The closed loop response of the paper machine to the process disturbance in Fig. 6 for full block input uncertainty Δ_I and output uncertainty Δ_O equal to a worst-case norm random matrix. The controller was designed via DK-iteration to control a subset of the pseudo-singular values of the process.

Time domain responses for other random full block uncertainties demonstrated similar robustness [88]. In all cases, the robust controllers derived from the theorems in this paper achieved monotonic or near monotonic rejection of the disturbance within the desired settling time.

6. Conclusions

Control algorithms for sheet and film processes based on modified DK-iteration procedures were presented that address model uncertainties in a numerically efficient and effective manner. Alternative algorithms requiring less computations were presented for the design of robust low order tunable controllers. The low order controllers were of the form of two static decoupling matrices in series with either a diagonal PI or PID controller. The algorithms are applicable to large scale sheet and film processes with arbitrary interaction matrices and very general uncertainty structures. These results, based on refinements of theorems by Hovd, Braatz, and Skogestad [23,37,38], are substantially more general and less conservative than previous approaches. Contributions of this manuscript include:

- The theorems are constructive.
- In many cases a more complete model reduction is derived, in that the robust multivariable controller can be constructed from the solution of a *single* SISO robust control problem.
- Low order tunable controllers can be designed. For the simple types of dynamics usually associated with sheet and film processes, the resulting controllers will give nearly the same robust performance as high order controllers, while being simpler to implement.
- More general processes and uncertainty structures are considered. This includes non-square processes and other processes with a singular interactions matrix, and perturbations which are nonlinear and/or time-varying.
- For the first time, these algorithms are applied to a simulated paper machine which has a realistic description of interactions across the machine. This example is of substantially higher dimensionality than that of any robust control problem ever considered.

Acknowledgements

The first author acknowledges support from the UIUC Computational Science and Engineering Program and the National Center for Supercomputing Applications. The third author acknowledges support from the DuPont Young Faculty Award.

Appendix

The Appendix contains some preliminary mathematics need in the proofs of the theorems and corollaries, and the state space matrices for the robust controllers computed in the examples.

A1. Preliminary mathematics

First we present a Lemma that will be used in proving our results.

Lemma 1 (Optimality of the SVD controller). *Consider the robust synthesis problem*

$$\inf_{\tilde{K}(s)\in\mathbf{K}_s^n} \sup_{s=j\omega} \bar{\sigma}\Big(F_l\Big(\tilde{G}(s), \tilde{K}(s)\Big)\Big), \tag{51}$$

where $\tilde{G}(s)$ is composed of diagonal sub-blocks of dimension $n \times n$. Then a diagonal controller $\tilde{K}(s)$ is optimal.

Note that the Lemma also holds if the H_∞ norm is replaced by a μ problem which has a single full block $\mathbf{\Delta}$, since these objectives are equivalent [56].

Without loss in generality, each proof considers one of each type of uncertainty ($\mathbf{\Delta} = \text{diag}\{\mathbf{\Delta}_k\} = \text{diag}\{\mathbf{\Delta}_{\text{DA}}, \mathbf{\Delta}_{\text{A}}, \mathbf{\Delta}_{\text{IA}}, \mathbf{\Delta}_{\text{O}}, \mathbf{\Delta}_{\text{IO}}, \mathbf{\Delta}_{\text{I}}, \mathbf{\Delta}_{\text{II}}\}$. Below is some preliminary algebra which is used in the proofs. For brevity, dependence on s will be suppressed.

The G matrix for the system shown in Fig. 12 is:

$$G = \left[\begin{array}{cccccccc} 0 & 0 & 0 & 0 & 0 & V^T & -V^T & -V^T \\ 0 & 0 & 0 & 0 & 0 & w_{\mathrm{A}} I & -w_{\mathrm{A}} I & -w_{\mathrm{A}} I \\ 0 & 0 & -w_{\mathrm{IA}} P & 0 & 0 & w_{\mathrm{IA}} P & -w_{\mathrm{IA}} P & -w_{\mathrm{IA}} P \\ w_{\mathrm{O}} U \Sigma_{\mathrm{DA}} & w_{\mathrm{O}} I & -w_{\mathrm{O}} P & 0 & 0 & w_{\mathrm{O}} P & -w_{\mathrm{O}} P & -w_{\mathrm{O}} P \\ w_{\mathrm{IO}} U \Sigma_{\mathrm{DA}} & w_{\mathrm{IO}} I & -w_{\mathrm{IO}} P & w_{\mathrm{IO}} I & -w_{\mathrm{IO}} I & w_{\mathrm{IO}} P & -w_{\mathrm{IO}} P & -w_{\mathrm{IO}} P \\ 0 & 0 & 0 & 0 & 0 & 0 & -w_{\mathrm{I}} I & -w_{\mathrm{I}} I \\ 0 & 0 & 0 & 0 & 0 & 0 & -w_{\mathrm{II}} I & -w_{\mathrm{II}} I \\ \hline U \Sigma_{\mathrm{DA}} & \mathrm{I} & -P & I & -I & P & -P & -P \end{array}\right]. \tag{52}$$

This can be written as $G = U_w \tilde{G} V_w^T$ where

$$U_w = \mathrm{diag}(I, V, U, U, U, V, V, U) \tag{53}$$

$$V_w^T = \mathrm{diag}\left(I, U^T, V^T, U^T, U^T, V^T, V^T, V^T\right). \tag{54}$$

and $\tilde{G}$ is partitioned compatibly with G and has diagonal $n \times n$ sub-blocks. Also define

$$U_{w1} = \mathrm{diag}(I, V, U, U, U, V, V,) \tag{55}$$

$$V_{w1}^T = \mathrm{diag}\left(I, U^T, V^T, U^T, U^T, V^T, V^T\right) \tag{56}$$

The scaled $\tilde{G}$ is also partitioned compatibly with G and has diagonal sub-blocks:

$$\left[\begin{array}{cc} D & 0 \\ 0 & I \end{array}\right] \tilde{G} \left[\begin{array}{cc} D^{-1} & 0 \\ 0 & I \end{array}\right] =$$

$$\left[\begin{array}{cccccccc} 0 & 0 & 0 & 0 & 0 & \frac{1}{d_6} D_1 & \frac{-1}{d_7} D_1 & -D_1 \\ 0 & 0 & 0 & 0 & 0 & \frac{d_2 w_{\mathrm{A}} I}{d_6} & \frac{-d_2 w_{\mathrm{A}}}{d_7} I & -d_2 w_{\mathrm{A}} I \\ 0 & 0 & -w_{\mathrm{IA}} \Sigma_{\mathrm{P}} & 0 & 0 & \frac{d_3 w_{\mathrm{IA}}}{d_6} \Sigma_{\mathrm{P}} & \frac{-d_3 w_{\mathrm{IA}}}{d_7} \Sigma_{\mathrm{P}} & -d_3 w_{\mathrm{IA}} \Sigma_{\mathrm{P}} \\ d_4 w_{\mathrm{O}} \Sigma_{\mathrm{DA}} D_1^{-1} & \frac{d_4 w_{\mathrm{O}}}{d_2} I & \frac{-d_4 w_{\mathrm{O}}}{d_3} \Sigma_{\mathrm{P}} & 0 & 0 & \frac{d_4 w_{\mathrm{O}}}{d_6} \Sigma_{\mathrm{P}} & \frac{-d_4 w_{\mathrm{O}}}{d_7} \Sigma_{\mathrm{P}} & -d_4 w_{\mathrm{O}} \Sigma_{\mathrm{P}} \\ d_5 w_{\mathrm{IO}} \Sigma_{\mathrm{DA}} D_1^{-1} & \frac{d_5 w_{\mathrm{IO}}}{d_2} I & \frac{-d_5 w_{\mathrm{IO}}}{d_3} \Sigma_{\mathrm{P}} & \frac{d_5 w_{\mathrm{IO}}}{d_4} I & -w_{\mathrm{IO}} I & \frac{d_5 w_{\mathrm{IO}}}{d_6} \Sigma_{\mathrm{P}} & \frac{-d_5 w_{\mathrm{IO}}}{d_6} \Sigma_{\mathrm{P}} & -d_5 w_{\mathrm{IO}} \Sigma_{\mathrm{P}} \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{-d_6 w_{\mathrm{I}}}{d_7} I & -d_6 w_{\mathrm{I}} I \\ 0 & 0 & 0 & 0 & 0 & 0 & -w_{\mathrm{II}} I & -d_7 w_{\mathrm{II}} I \\ \hline \Sigma_{\mathrm{DA}} D_1^{-1} & \frac{1}{d_2} I & \frac{-1}{d_3} \Sigma_{\mathrm{P}} & \frac{1}{d_4} I & \frac{-1}{d_6} \Sigma_{\mathrm{P}} & \frac{1}{d_6} \Sigma_{\mathrm{P}} & \frac{-1}{d_7} \Sigma_{\mathrm{P}} & -\Sigma_{\mathrm{P}} \end{array}\right] \tag{57}$$

A2. Proof of Lemma 1

This Lemma essentially shows that a decentralized controller is optimal for a decentralized plant with decentralized weights (costs).

The optimal controller solves

$$\inf_{\tilde{K}\in\mathbf{K}_s} \left\| F_l\left(\tilde{G}, \tilde{K}\right)\right\|_\infty \tag{58}$$

where $\mathbf{K}_s$ represents the set of all stabilizing controllers.

The key to a rigorous proof that a diagonal controller $\tilde{K}$ can be chosen to be optimal is to reparameterize the above optimization over $\tilde{K}$ as an optimization over the Youla matrix Q, and then use matrix dilation theory to show that Q can be taken to be diagonal. The set of all stabilizing $\tilde{K}$ is given by

$$\mathbf{K}_s = \left\{K : K = (Y - TQ)(X - SQ)^{-1}, Q \in \mathcal{RH}_\infty\right\} \tag{59}$$

$$= \left\{K : K = \left(\tilde{X} - Q\tilde{S}\right)^{-1}\left(\tilde{Y} - Q\tilde{T}\right), Q \in \mathcal{RH}_\infty\right\} \tag{60}$$

where (S, T) and $(\tilde{S}, \tilde{T})$ are right and left coprime factors of $\tilde{G}_{22}$, respectively (i.e. $\tilde{G}_{22} = ST^{-1} = \tilde{T}^{-1}\tilde{S}$), and $\left(X, Y, \tilde{X}, \tilde{Y}\right)$ is a solution to the following Bezout identity:

$$\begin{bmatrix} \tilde{X} & -\tilde{Y} \\ -\tilde{S} & \tilde{T} \end{bmatrix}\begin{bmatrix} T & Y \\ S & X \end{bmatrix} = I \tag{61}$$

Note that, since $\tilde{G}_{22}$ is diagonal, we may choose $T, S, \tilde{X}, \tilde{Y}, X, Y, \tilde{T}, \tilde{S}$ to all be diagonal (to do this, first construct the right and left coprime factors of each subsystem and stack these on the diagonal to construct right and left coprime factors of the overall system).

Using the parameterization (59) and (60), (58) becomes

$$\inf_{Q\in\mathcal{RH}_\infty} \left\| G_{11} + G_{12}QG_{21}\right\|_\infty \tag{62}$$

where

$$G_{11} = \tilde{G}_{11} + \tilde{G}_{12}T\tilde{Y}\tilde{G}_{21} \tag{63}$$

$$G_{12} = \tilde{G}_{12}T \tag{64}$$

$$G_{21} = \tilde{T}\tilde{G}_{21} \tag{65}$$

The only restriction on Q is that it should be analytic in the closed RHP.

The matrix G_{11} consists of diagonal blocks because $\tilde{G}_{11}$, $\tilde{G}_{12}$, and $\tilde{G}_{21}$ consist of diagonal blocks and T and $\tilde{Y}$ are diagonal. Similarly, G_{12} and G_{21} also consist of diagonal blocks. Thus, each entry of $G_{11} + G_{12}QG_{21}$ will have one Q_{ij} in it, and the rows and columns of this matrix can be permuted so that the permuted matrix can be partitioned with only one Q_{ij} in each partition (permuting the rows and columns of a matrix does not change the value of its unitary-invariant norm). Call this permuted matrix $P(Q)$ and let $P_{ij}\left(Q_{ij}\right)$ be the partition containing Q_{ij}. Then

$$\inf_{Q\in\mathcal{RH}_\infty} \left\| G_{11} + G_{12}QG_{21}\right\|_\infty = \inf_{Q\in\mathcal{RH}_\infty} \left\| P(Q)\right\|_\infty \tag{66}$$

The maximum singular value of a matrix [in this case, $P(Q)$] is greater than the maximum singular values of each partition P_{ij} of $P(Q)$ [40], that is,

$$\inf_{Q\in\mathcal{RH}_\infty} \left\| P(Q)\right\|_\infty = \inf_{Q\in\mathcal{RH}_\infty \text{ and } Q \text{ full}} \sup_\omega \left(P(Q)\big|_{s=j\omega}\right) \tag{67}$$

$$\geqslant \inf_{Q_{ij}\in\mathcal{RH}_\infty} \max_{i,j} \left\{ \sup_{\omega} \bar{\sigma}\Big(P_{ij}(Q_{ij})\big|_{s=j\omega}\Big)\right\} \tag{68}$$

$$\geqslant \inf_{Q_{ii}\in\mathcal{RH}_\infty} \max_{i} \left\{ \sup_{\omega} \bar{\sigma}\Big(P_{ii}(Q_{ii})\big|_{s=j\omega}\Big)\right\} \tag{69}$$

$$= \inf_{Q\in\mathcal{RH}_\infty \text{ and } Q \text{ diagonal}} \sup_{\omega} \bar{\sigma}\Big(P(Q)\big|_{s=j\omega}\Big) \tag{70}$$

$$= \inf_{Q\in\mathcal{RH}_\infty \text{ and } Q \text{ diagonal}} \sup_{\omega} \bar{\sigma}\big(G_{11}+G_{12}QG_{21}\big|_{s-j\omega}\Big). \tag{71}$$

Thus minimizing over diagonal Q gives an H_∞-norm less than or equal to the value obtained by minimizing over full Q. Since Q being diagonal is more restrictive than allowing Q to be full, the above inequalities are equalities and the optimal Q can be taken to be diagonal. That diagonal Q corresponds to diagonal K can be seen from (59) and (60), that is

$$\inf_{Q\in\mathcal{RH}_\infty \text{ and } Q \text{ diagonal}} \sup_{\omega} \bar{\sigma}\Big(G_{11}+G_{12}QG_{21}\big|_{s=j\omega}\Big) \tag{72}$$

$$= \inf_{\tilde{K}\in\mathbf{K}_s \text{ and } \tilde{K} \text{ diagonal}} \Big\|F_l\Big(\tilde{G},\tilde{K}\Big)\Big\|_\infty. \tag{73}$$

QED.

A3. Proof of Theorem 1

The necessity and sufficiency of (15) as a test for robustness to arbitrarily-slow linear time varying full block uncertainties was shown by Poolla and Tikku [48]. Now

$$\inf_{K\in\mathbf{K}_s^n} \inf_{D\in\mathbf{D}_s^{nu}} \big\|DF_l(G,K)D^{-1}\big\|_\infty = \tag{74}$$

$$\inf_{K\in\mathbf{K}_s^n} \inf_{D\in\mathbf{D}_s^{nu}} \Big\|DF_l\Big(U_w\tilde{G}V_w^T,K\Big)D^{-1}\Big\|_\infty = \tag{75}$$

$$\inf_{K\in\mathbf{K}_s^n} \inf_{D\in\mathbf{D}_s^{nu}} \Big\|DU_{w1}F_l\Big(\tilde{G},V^TKU\Big)V_{w1}^TD^{-1}\Big\|_\infty = \tag{76}$$

$$\inf_{K\in\mathbf{K}_s^n} \inf_{D\in\mathbf{D}_s^{nu}} \Big\|U_{w1}DF_l\Big(\tilde{G},V^TKU\Big)D^{-1}V_{w1}^T\Big\|_\infty = \tag{77}$$

$$\inf_{K\in\mathbf{K}_s^n} \inf_{D\in\mathbf{D}_s^{nu}} \Big\|DF_l\Big(\tilde{G},V^TKU\Big)D^{-1}\Big\|_\infty = \tag{78}$$

$$\inf_{K\in\mathbf{K}_s^n} \inf_{D\in\mathbf{D}_s^{nu}} \left\|F_l\left(\begin{bmatrix} D & 0\\ 0 & I\end{bmatrix}\tilde{G}\begin{bmatrix} D^{-1} & 0\\ 0 & I\end{bmatrix},VKU^T\right)\right\|_\infty = \tag{79}$$

$$\inf_{\Sigma_K\in\mathbf{K}_s^n} \inf_{D\in\mathbf{D}_s^{nu}} \left\|F_l\begin{bmatrix} D & 0\\ 0 & I\end{bmatrix}\tilde{G}\begin{bmatrix} D^{-1} & 0\\ 0 & I\end{bmatrix},\Sigma_K\right\|_\infty = \tag{80}$$

$$\inf_{\substack{d_k\\ k=2,\ldots,u}} \inf_{\substack{d_{1i}\\ i=1,\ldots,n}} \max_{i=1,\ldots,n} \left\{ \inf_{\Sigma_{K,ii}\in\mathbf{K}_s^1} \left\|F_l\left(\begin{bmatrix} D^i & 0\\ 0 & I\end{bmatrix}\tilde{G}^i\begin{bmatrix} (D^i)^{-1} & 0\\ 0 & I\end{bmatrix},\Sigma_{K,ii}\right)\right\|_\infty\right\} = \tag{81}$$

$$\inf_{\substack{d_k\\ k=2,\ldots,u}} \max_{i=1,\ldots,n} \left\{ \inf_{\Sigma_{K,ii}\in\mathbf{K}_s^1} \inf_{d_{1i}} \left\|F_l\left(\begin{bmatrix} D^i & 0\\ 0 & I\end{bmatrix}\tilde{G}^i\begin{bmatrix} (D^i)^{-1} & 0\\ 0 & I\end{bmatrix},\Sigma_{K,ii}\right)\right\|_\infty\right\} = \tag{82}$$

$$\inf_{\substack{d_k \\ k=2,\ldots,u}} \max_{i=1,\ldots,n} \left\{ \inf_{\Sigma_{\mathrm{K},ii}\in\mathbf{K}_s^1} \inf_{d_{1i}} \left\| D^i F_l\left(\tilde{G}^i, \Sigma_{\mathrm{K},ii}\right)\left(D^i\right)^{-1} \right\|_\infty \right\}. \tag{83}$$

The fact that (80) is equal to (81) follows from Lemma 1. QED.

A4. Proof of Theorem 2

That constant scaling matrices provide a necessary and sufficient condition for robustness to LTV, NLTI, and NLTV uncertainties was shown in [49]. The rest of the proof follows the same steps as that of Theorem 1. QED.

A5. Proof of Theorem 3

Consider $G = U_w \tilde{G} V_w^T$ where $\tilde{G}$ has diagonal sub-blocks. Then

$$\inf_{K\in\mathbf{K}_s^n} \sup_{s=j\omega} \mu_{\mathbf{\Delta}}(F_l(G,K)) = \inf_{K\in\mathbf{K}_s^n} \sup_{s=j\omega} \mu_{\mathbf{\Delta}}\left(F_l\left(U_w \tilde{G} V_w^T, K\right)\right) \tag{84}$$

$$= \inf_{K\in\mathbf{K}_s^n} \sup_{s=j\omega} \mu_{\mathbf{\Delta}}\left(U_{w1} F_l\left(\tilde{G}, V^T K U\right) V_{w1}^T\right) \tag{85}$$

$$= \inf_{\Sigma_\mathrm{K}\in\mathbf{K}_s^n} \sup_{s=j\omega} \mu_{\mathbf{\Delta}}\left(F_l\left(\tilde{G}, V^T K U\right)\right) \tag{86}$$

The last step follows from two observations. For the sub-blocks of Δ which are repeated diagonal, the corresponding sub-blocks of U_{w1} and V_{w1}^T commute with the sub-block of Δ and cancel. The sub-blocks of U_{w1} and V_{w1}^T corresponding to a full uncertainty block can be absorbed into the uncertainty to produce an equivalent full uncertainty block (that is, it will have the same set).

By assumption Δ has at most one full block. Absorb $u-1$ diagonal blocks of Δ into G. The remaining block can be either full or diagonal without affecting the value of μ. By taking the remaining block as full, Lemma 1 implies that a diagonal $V^T K U = \Sigma_\mathrm{K}$ is optimal for all values of the diagonal uncertainties, and hence is optimal for (86). Now by taking the remaining block as diagonal, (86) is equivalent to

$$\inf_{\Sigma_{\mathrm{K},ii}\in\mathbf{K}_s^1} \sup_{s=j\omega} \max_{i=1,\ldots,n} \left\{ \mu_{\tilde{\mathbf{\Delta}}}\left(F_l\left(\tilde{G}^i, \Sigma_{\mathrm{K},ii}\right)\right)\right\} = \tag{87}$$

$$\inf_{\Sigma_{\mathrm{K},ii}\in\mathbf{K}_s^1} \max_{i=1,\ldots,n} \left\{ \sup_{s=j\omega} \mu_{\tilde{\mathbf{\Delta}}}\left(F_l\left(\tilde{G}^i, \Sigma_{\mathrm{K},ii}\right)\right)\right\} = \tag{88}$$

$$\max_{i=1,\ldots,n} \left\{ \inf_{\Sigma_{\mathrm{K},ii}(s)\in\mathbf{K}_s^1} \sup_{s=j\omega} \mu_{\tilde{\mathbf{\Delta}}}\left(F_l\left(\tilde{G}^i, \Sigma_{\mathrm{K},ii}\right)\right)\right\} \tag{89}$$

QED.

A6. Proof of Corollary 1

$$\inf_{K\in\mathbf{K}_s^n} \sup_{s=j\omega} \mu_{\Delta}(F_l(G,K)) = \tag{90}$$

$$\max_{i=1,\ldots,n} \left\{ \inf_{\Sigma_{\mathrm{K},ii}\in\mathbf{K}_s^1} \sup_{s=j\omega} \mu_{\tilde{\mathbf{\Delta}}}\left(F_l\left(\tilde{G}^i, \Sigma_{\mathrm{K},ii}\right)\right)\right\} = \tag{91}$$

$$\max_{i=1,\ldots,n} \left\{ \inf_{\hat{D}^i\in\hat{\mathbf{D}}_s^u} \inf_{\Sigma_{\mathrm{K},ii}\in\mathbf{K}_s^1} \left\| \hat{D}^i F_l\left(\tilde{G}^i, \Sigma_{\mathrm{K},ii}\right)\left(\hat{D}^i\right)^{-1} \right\|_\infty \right\} \tag{92}$$

QED.

A7. Proof of Theorem 4

The equations are given first and comments follow.

$$\inf_{K\in\mathbf{K}_s^n}\sup_{s=j\omega}\mu_{\mathbf{\Delta}}(F_l(G,K))= \tag{93}$$

$$\inf_{K\in\mathbf{K}_s^n}\sup_{s=j\omega}\mu_{\mathbf{\Delta}}\Big(F_l\Big(U_w\tilde{G}V_w^T,K\Big)\Big)= \tag{94}$$

$$\inf_{K\in\mathbf{K}_s^n}\sup_{s=j\omega}\mu_{\mathbf{\Delta}}\Big(U_{w1}F_l\Big(\tilde{G},V^TKU\Big)V_{w1}^T\Big)= \tag{95}$$

$$\inf_{K\in\mathbf{K}_s^n}\sup_{s=j\omega}\mu_{\tilde{\mathbf{\Delta}}}\Big(F_l\Big(\tilde{G},V^TKU\Big)\Big)= \tag{96}$$

$$\inf_{K\in\mathbf{K}_s^n}\sup_{s=j\omega}\sup_{\|\tilde{\mathbf{\Delta}}_d\|_\infty\leqslant 1}\mu_{\tilde{\Delta}_f}\Big(F_l\Big(F_u\Big(\tilde{G}^s,\tilde{\Delta}_d\Big),V^TKU\Big)\Big)= \tag{97}$$

$$\inf_{\Sigma_{\mathrm{K},ii}\in\mathbf{K}_s^n}\inf_{D_f\in\mathbf{D}_s^f}\sup_{s=j\omega}\sup_{\|\tilde{\mathbf{\Delta}}_d\|_\infty\leqslant 1}\bar{\sigma}\Big(D_fF_l\Big(F_u\Big(\tilde{G}^s,\tilde{\Delta}_d\Big)\Sigma_K\Big)D_f^{-1}\Big)= \tag{98}$$

$$\inf_{\Sigma_{\mathrm{K}}\in\mathbf{K}_s^n}\inf_{D_f\in\mathbf{D}_s^f}\sup_{s=j\omega}\sup_{\|\tilde{\mathbf{\Delta}}_d\|_\infty\leqslant 1}\mu_{\tilde{\Delta}_{\mathrm{BF}}}\Big(D_fF_l\Big(F_u\Big(\tilde{G}^s,\tilde{\Delta}_d\Big),\Sigma_{\mathrm{K}}\Big)D_f^{-1}\Big)= \tag{99}$$

$$\inf_{\Sigma_{\mathrm{K}}\in\mathbf{K}_s^n}\inf_{D_f\in\mathbf{D}_s^f}\sup_{s=j\omega}\mu_{\begin{bmatrix}\tilde{\mathbf{\Delta}}_d & \\ & \tilde{\mathbf{\Delta}}_{\mathrm{BF}}\end{bmatrix}}\left(\begin{bmatrix}I & D_f\end{bmatrix}F_l\Big(\tilde{G},\Sigma_{\mathrm{K}}\Big)\begin{bmatrix}I & \\ D_f^{-1} & \end{bmatrix}\right)= \tag{100}$$

$$\inf_{\Sigma_{\mathrm{K}}\in\mathbf{K}_s^n}\inf_{D_f\in\mathbf{D}_s^f}\sup_{s=j\omega}\mu_{\begin{bmatrix}\tilde{\mathbf{\Delta}}_d & \\ & \tilde{\mathbf{\Delta}}_{\mathrm{BF}}\end{bmatrix}}\left(F_l\left(\begin{bmatrix}I & & \\ & D_f & \\ & & I\end{bmatrix}\tilde{G}\begin{bmatrix}I & & \\ & D_f^{-1} & \\ & & I\end{bmatrix},\Sigma_{\mathrm{K}}\right)\right)= \tag{101}$$

$$\inf_{D_f\in\mathbf{D}_s^f}\inf_{\Sigma_{\mathrm{K}}\in\mathbf{K}_s^n}\sup_{s=j\omega}\mu_{\begin{bmatrix}\tilde{\mathbf{\Delta}}_d & \\ & \tilde{\mathbf{\Delta}}_{\mathrm{BF}}\end{bmatrix}}\left(F_l\left(\begin{bmatrix}I & & \\ & D_f & \\ & & I\end{bmatrix}\tilde{G}\begin{bmatrix}I & & \\ & D_f^{-1} & \\ & & I\end{bmatrix},\Sigma_{\mathrm{K}}\right)\right)= \tag{102}$$

$$\inf_{D_f\in\mathbf{D}_s^f}\max_{i=1,\ldots,n}\left\{\inf_{\Sigma_{\mathrm{K},ii}\in\mathbf{K}_s^1}\inf_{D\in\mathbf{D}_s^{d+1}}\left\|D\left(F_l\begin{bmatrix}1 & & \\ & \hat{D}_f^i & \\ & & 1\end{bmatrix}\tilde{G}^i\begin{bmatrix}1 & & \\ & \left(\hat{D}_f^i\right)^{-1} & \\ & & 1\end{bmatrix},\Sigma_{\mathrm{K},ii}\right)D^{-1}\right\|_\infty\right\}= \tag{103}$$

$$\inf_{D_f\in\mathbf{D}_s^f}\max_{i=1,\ldots,n}\left\{\inf_{\Sigma_{\mathrm{K},ii}\in\mathbf{K}_s^1}\inf_{\hat{D}_d^i\in\hat{\mathbf{D}}_s^d}\left\|\begin{bmatrix}\hat{D}_d^i & \\ & d_{\mathrm{BF}}\hat{D}_f^i\end{bmatrix}F_l\Big(\tilde{G}^i,\Sigma_{\mathrm{K},ii}\Big)\begin{bmatrix}\left(\hat{D}_d^i\right)^{-1} & \\ & d_{\mathrm{BF}}^{-1}\left(\hat{D}_f^i\right)^{-1}\end{bmatrix}\right\|_\infty\right\}= \tag{104}$$

$$\inf_{D_f\in\hat{\mathbf{D}}_s^f}\max_{i=1,\ldots,n}\left\{\inf_{\Sigma_{\mathrm{K},ii}\in\mathbf{K}_s^1}\inf_{\hat{D}_d^i\in\hat{\mathbf{D}}_s^d}\left\|\begin{bmatrix}\hat{D}_d^i & \\ & \hat{D}_f^i\end{bmatrix}F_l\Big(\tilde{G}^i,\Sigma_{\mathrm{K},ii}\Big)\begin{bmatrix}\left(\hat{D}_d^i\right)^{-1} & \\ & \left(\hat{D}_f^i\right)^{-1}\end{bmatrix}\right\|_\infty\right\} \tag{105}$$

In (97) we have absorbed the diagonal blocks of Δ into $\tilde{G}$ which produces a diagonally scaled $\tilde{G}$ denoted by $\tilde{G}^s$ (this scaling is the same as that used in skewed-μ, for details see [91–94]. For any fixed values of the diagonal blocks of Δ, Theorem 1 may be applied to show that a diagonal controller is optimal. If it is optimal for any fixed values, it is optimal for the worst case values.

In (99) we replaced $\bar{\sigma}$ with $\mu_{\tilde{\Delta}_{\mathrm{BF}}}$ where $\tilde{\Delta}_{\mathrm{BF}}$ is one large full block [56]. The step from (102) to (103) holds as a direct application of Corollary 1. The step from (104) to (105) holds since D_d has a extra degree of freedom (thus d_{BF} may be set to one without loss of generality). QED.

A8. Proof of Theorem 5

Under the assumptions, (19) of Theorem 3 holds. Pick any $\bar{i} \in [1, n]$ and define

$$\hat{\Sigma}_{\mathrm{K},ii} = \frac{\Sigma_{\mathrm{K},ii}\Sigma_{\mathrm{P},ii}}{\Sigma_{\mathrm{P},\bar{i}\bar{i}}} \tag{106}$$

$$H\left(\Sigma_{\mathrm{P},ii}, \Sigma_{\mathrm{K},ii}\right) = \Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}/\left(1 + \Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) \tag{107}$$

$$S\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) = 1/\left(1 + \Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) \tag{108}$$

Then

$$\inf_{\Sigma_{\mathrm{K},ii}\in\mathbf{K}_s^1} \sup_{s=j\omega} \mu_{\tilde{\Delta}}\left(F_l\left(\tilde{G}^i, \Sigma_{\mathrm{K},ii}\right)\right) = \tag{109}$$

$$\inf_{\Sigma_{\mathrm{K},ii}\in\mathbf{K}_s^1} \sup_{s=j\omega} \mu_{\tilde{\Delta}}\left(\begin{bmatrix} 1 & & & \\ & 1 & & \\ & & \Sigma_{\mathrm{P},ii} & \\ & & & \Sigma_{\mathrm{P},ii} \end{bmatrix} F_l\left(\tilde{G}^i, \Sigma_{\mathrm{K},ii}\right) \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & \Sigma_{\mathrm{P},ii}^{-1} & \\ & & & \Sigma_{\mathrm{P},ii}^{-1} \end{bmatrix}\right) = \tag{110}$$

$$\inf_{\Sigma_{\mathrm{K},ii}\in\mathbf{K}_s^1} \sup_{s=j\omega} \mu_{\tilde{\Delta}}\left(\begin{bmatrix} w_{\mathrm{O}}H\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) & -w_{\mathrm{O}}H\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) & -w_{\mathrm{O}}S\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) & w_{\mathrm{O}}S\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) \\ w_{\mathrm{IO}}S\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) & -w_{\mathrm{IO}}S\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) & w_{\mathrm{IO}}S\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) & -w_{\mathrm{IO}}S\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) \\ -w_{\mathrm{I}}H\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) & w_{\mathrm{I}}H\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) & -w_{\mathrm{I}}H\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) & w_{\mathrm{I}}H\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) \\ w_{\mathrm{II}}H\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) & -w_{\mathrm{II}}H\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) & -w_{\mathrm{II}}S\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) & w_{\mathrm{II}}S\left(\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}\right) \end{bmatrix}\right) = \tag{111}$$

$$\inf_{\Sigma_{\mathrm{K},ii}\in\mathbf{K}_s^1} \sup_{s=j\omega} \mu_{\tilde{\Delta}}\left(\begin{bmatrix} w_{\mathrm{O}}H\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) & -w_{\mathrm{O}}H\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) & -w_{\mathrm{O}}S\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) & w_{\mathrm{O}}S\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) \\ w_{\mathrm{IO}}S\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) & -w_{\mathrm{IO}}S\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) & w_{\mathrm{IO}}S\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) & -w_{\mathrm{IO}}S\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) \\ -w_{\mathrm{I}}H\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) & w_{\mathrm{I}}H\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) & -w_{\mathrm{I}}H\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) & w_{\mathrm{I}}H\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) \\ w_{\mathrm{II}}H\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) & -w_{\mathrm{II}}H\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) & -w_{\mathrm{II}}S\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) & w_{\mathrm{II}}S\left(\Sigma_{\mathrm{P},\bar{i}\bar{i}}\hat{\Sigma}_{\mathrm{K},ii}\right) \end{bmatrix}\right) \tag{112}$$

since, for the assumed uncertainty types, the above matrices contain $\Sigma_{\mathrm{P},ii}$ and $\Sigma_{\mathrm{K},ii}$ only as the product $\Sigma_{\mathrm{P},ii}\Sigma_{\mathrm{K},ii}$. Since (112) is the SISO control problem for $\Sigma_{\mathrm{P},\bar{i}\bar{i}}$, we have

$$\hat{\Sigma}_{\mathrm{K},ii,\mathrm{opt}} = \Sigma_{\mathrm{K},\bar{i}\bar{i},\mathrm{opt}} \;\Rightarrow\; \Sigma_{\mathrm{K},ii,\mathrm{opt}} = \frac{\Sigma_{\mathrm{P},\bar{i}\bar{i},\mathrm{opt}}\Sigma_{\mathrm{K},\bar{i}\bar{i}}}{\Sigma_{\mathrm{P},ii}}. \tag{113}$$

Note that assumptions (ii) and (iii) were required to ensure complete SISO control problem equivalence (that is, internal stability as well as the μ condition is satisfied). At the surface it may appear that it would also be required that $\Sigma_{\mathrm{P},ii}$ have no zeros or poles at $s = 0$. However, the continuity of μ allows the construction of a limit argument to show that zeros or poles at $s = 0$ are allowed [95].

QED

A6. Proof of Theorem 6

Under the assumptions, (19) of Theorem 3 holds. Pick any $\bar{i} \in [1, n]$ and define

$$\hat{\Sigma}_{\mathrm{K},ii} = \frac{\Sigma_{\mathrm{K},ii}\Sigma_{\mathrm{P},ii}}{\Sigma_{\mathrm{P},\bar{i}\bar{i}}}. \tag{114}$$

For any i

$$\arg \inf_{\Sigma_{\mathrm{K},ii} \in \mathbf{K}_s^1} \sup_{s=j\omega} \mu_{\tilde{\mathbf{\Delta}}}\left(F_l\left(\tilde{G}^i, \Sigma_{\mathrm{K},ii}\right)\right) = \arg \inf_{\hat{\Sigma}_{\mathrm{K},ii} \in \mathbf{K}_s^1} \sup_{s=j\omega} \mu_{\tilde{\mathbf{\Delta}}}\left(F_l\left(\tilde{G}^{\bar{i}}, \hat{\Sigma}_{\mathrm{K},ii}\right)\right). \tag{115}$$

Now apply that same argument as used in the proof of Theorem 5. QED.

A10. Proof of Corollary 2
Similar to the proof of Theorem 3. QED.

A11. Proof of Corollary 3
Similar to the proof of Theorem 4. QED.

A12. Proof of Corollary 4
Similar to the proof of Theorem 3. QED.

A13. Proof of Corollary 5
Similar to the proof of Theorem 1. QED.

A14. Proof of Corollary 6
Similar to the proof of Theorem 2. QED.

A15. Proof of Corollary 7
Similar to the proof of Theorem 4. QED.

A.16. Proof of Corollary 8
Similar to the proof of Theorem 5. QED.

A.17. Proof of Corollary 9
Similar to the proof of Theorem 6. QED.

A.18. State space matrices of the controller designed in Section 5.2

Ak =

Columns 1−3

−5.880071907852288e + 02	6.148744672296147e + 02	4.604740029212040e + 02
5.604233917995811*e* + 01	−6.283292865721634e + 01	−1.691823275285767e + 01
−1.546954497030094e + 01	2.191712894846478 + 01	−3.890468609915896e − 01
9.311545220904512e − 16	−1.002746785079390e − 15	−7.189989990010580e − 16
1.747568665938097e − 15	−1.881931322784988e − 15	−1.349400224867352e − 15
5.168031888201593e − 01	−1.164315286585576e + 00	−8.021795349521191e − 01
−1.138880502746125e + 00	2.565804560859658e + 00	1.767765083154003e + 00
3.016528151784629e + 00	−6.795991037819960e + 00	−4.682241136113864e + 00
−1.490982062881223e − 01	3.359060558044675e − 01	2.314295506872773e − 01
4.534129462957375e − 01	−1.021502258360954e + 00	−7.037854917868097e − 01
−2.408891108259889e + 00	5.427034510893976e + 00	3.73906909275055e + 00
2.227914152170288e + 00	−5.019308240907330e + 00	−3.458158109630253e + 00
−3.218198676091172e − 15	3.465631428124776e − 15	2.484959877015252e − 15
−1.301471937665606e − 15	1.401536233143548e − 15	1.004942786841387e − 15
1.740120485458225e − 15	−1.873910485368964e − 15	−1.343649048040776e − 15
6.361093229963488e − 16	−6.850168940399087e − 16	−4.911773026273162e − 16
6.277618792368714e − 16	−6.760276530547976e − 16	−4.847317518998088e − 16

Columns 4 – 6

−2.518737189454669e + 02	−8.207425911351102e + 02	4.016757110238605e + 03
1.540021980744796e + 01	5.018235472019094e + 01	−2.455950651373719e + 02
−4.364834033951118e + 00	−1.422302100393165e + 01	6.960820769347728e + 01
−8.265644312716370e + 01	−1.347353451334145e + 00	−6.634331338467195e − 15
1.786532630293183e + 00	−5.903219215036291e − 02	−1.190116941665417e − 14
−5.954429810346211e − 01	1.454723830235523e + 00	−7.174092503276716e + 00
−1.512726989370992e + 00	−3.228108488833820e + 00	1.712162732350863e + 01
−5.555230480618331e + 00	8.474637242388923e + 00	−4.179160438722221e + 01
−4.334599934426976e − 01	−4.244734476200322e − 01	2.065637361398801e + 00
8.606041586127435e − 01	1.287221971883328e + 00	−6.281676656796931e + 00
−2.092695851781515e + 00	−6.819149782834415e + 00	3.337327521665450e + 01
8.896187620953473e + 00	6.361857614333061e + 00	−3.086598306768993e + 01
1.838522761346047e + 01	1.454077465700355e − 01	2.199019483247496e − 14
3.037564364282468e + 00	2.401086839222032e − 02	8.874789698245302e − 15
−6.705799569379678e + 00	−5.300696598177999e − 02	−1.187920560130718e − 14
−3.294305308792737e − 01	−2.604031445170019e − 03	−4.376380741688733e − 15
2.931507431542018e − 01	2.317252597418788e − 03	−4.277264465118866e − 15

Columns 7 – 9

1.504246426113544e + 02	1.525070510577801e + 03	−3.329230683724810e + 03
−9.197357193003622e + 00	−9.324681107297295e + 01	2.035578961303343e + 02
2.606776929184614e + 00	2.642863930629627e + 01	−5.769375009046735e + 01
2.234494381634972e − 14	−1.546415674605307e − 14	−5.687376785619688e − 01
−7.641468864095658e − 16	−4.332731699650277e − 15	5.050899772695990e − 03
−1.572125549019853e + 00	−2.885616450796306e + 00	5.941956540594664e + 00
4.422437901327094e − 01	3.628293611098791e + 00	−1.305437165084760e + 01
1.013673413206383e + 00	−1.808353651471117e + 01	3.451939524547544e + 01
7.735662210222391e − 02	1.198427614464494e + 00	−1.744775419349829e + 00
−2.352442381169413e − 01	−3.644460983966932e + 00	5.318505537686632e + 00
1.249804969396672e + 00	1.267106685255740e + 01	−2.766095355491361 + 01
−1.155908695591889e + 00	−1.790761880431088e + 01	2.599254552566666e + 01
−4.568064743158970e − 15	1.142558079306990e − 14	1.468590234726399e − 01
−5.389191681212471e − 16	3.860318418220820e − 15	2.928977227012811e − 02
1.520714377337405e − 15	−5.631857556122544*e* − 15	−5.350055242693297e − 02
1.883703905232138e − 16	−1.948351299877517e − 15	6.509477441801849e − 02
−2.391325201017346e − 16	−1.581266453034795e − 15	4.186260809490598e − 03

Columns 10 – 12

−6.033542782401723e + 02	−2.361824502053449e + 03	−9.699111577159437e + 02
3.689066308928619e + 01	1.444081448057488e + 02	5.930291213016883e + 01
−1.045580022885556e + 01	−4.092912913639285e + 01	−1.680803082128656e + 01
2.148937879454401e + 01	−1.276600856798962e − 11	−9.597327131896650e + 01
−2.849345083709308e − 01	1.757648657146881e − 13	1.349787265190995e + 00
1.322063592077839e + 00	4.016883123307948e + 00	1.900748850110523e − 01
−2.305584665278505e + 00	−8.852015563970417e + 00	−4.871374333183550e + 00
8.749969190898078 + 00	2.344614214078176e + 01	7.095440459624101e − 01
−5.698009108790048e − 02	−1.158874560974575e + 00	−8.359547880219935e − 01

1.354668360317635e − 02	3.524178742052303e + 00	2.956546775033679e + 00
−5.012976345900557e + 00	−1.972324754951022e + 01	−8.0585186292282359e + 00
1.693829607087231e + 00	1.731659353430797e + 01	1.030008363738264e + 01
−5.195583450271068e + 00	3.076586046944202e − 12	2.291375800934379e + 01
−8.732600801466369e − 01	5.150827891630203e − 13	3.708399980908524e + 00
1.893891079549973e + 00	−1.119196544599048e − 12	−8.353515348056909e + 00
−1.129085201254176e − 01	8.346623946347433e − 14	−1.422334750303090e + 00
08.84114471291422e − 02	5.546306742616683e − 14	3.375768808475222e − 01

Columns 13 − 15

2.612824770728370e + 02	−2.458118279523622e + 02	−1.509455164352034e + 02
−1.597549595727122e + 01	1.502957989259916e + 01	9.229204784776213e + 00
4.527882675403263e + 00	−4.259784772648334e + 00	−2.615803388170683e + 00
−4.126827894911816e − 16	3.882476620161116e − 16	2.384110005444475e − 16
−7.745132464884936e − 16	7.286539802651294e − 16	4.474440968520103e − 16
6.832400925277543e − 02	4.708152797602322e − 01	2.598064579192431e − 01
−1.505657931120921e − 01	−1.037536830486723e + 00	−5.725361526643380e − 01
3.988003592324785e − 01	2.748101359299717e + 00	1.516464122673138e + 00
−1.971154096305844e − 02	−1.358306512495912e − 01	−7.495440758842742e − 02
5.994349688429951e − 02	4.130658397146136e − 01	2.279390183732075e − 01
2.170868632911270e + 00	−2.042330556869504e + 00	−1.254132655885890e + 00
2.945415787680354e − 01	2.029662446995587e + 00	1.120013376358902e + 00
1.536076500855985e + 00	−1.999611752484199e + 01	5.253082020888732e + 00
1.999611752484184e + 01	−2.401527592458446e + 01	1.091516833465946e + 01
−5.253082020888680e + 00	1.091516833465946e + 01	−8.277776883445661e + 00
−2.819203081848754e − 16	2.652276841071862e − 16	1.628681991585485e − 16
−2.782207649897801e − 16	2.617471924739046e − 16	1.607309358241906e − 16

Columns 16 − 17

−3.521949820207279e + 02	1.202207848415532e + 03
2.153412496114261e + 01	−7.350614108273361e + 01
−6.103346750693381e + 00	2.083360564419734e + 01
1.649825182048802e − 01	1.304128260855715e − 03
3.305304236890853e − 02	2.612725707489746e − 04
6.440432501452810e − 01	−2.142815582291262e + 00
−1.352629057058886e + 00	4.722655003392209e + 00
3.770668974448044e + 00	−1.250730815729212e + 01
−1.533177392850852e − 01	6.184611286261353e − 01
4.203244851449394e − 01	−1.881125263626151e + 00
−2.926216284012265e + 00	9.988558504203477e + 00
2.533748524177969e + 00	−9.239495217611880e + 00
−1.722476028671700e − 01	−1.361556177024859e − 03
−9.413569841909747e − 02	−7.441092910887845e − 04
6.232887804508652e − 02	4.926876629717610e − 04
−6.996318151809803e + 00	−7.492870120896803e − 01
7.155243215128683e − 01	−6.749813277420426e − 02

Bk =

4.628508634446622e − 16
1.346687187543125e − 15
1.164045765169753e − 15
7.163600221044353e − 01
1.435174976090861e − 01
7.057191151708615e − 02
1.338802016092666e − 01
4.616522236195802e − 01
−2.289907570187057e + 00
−9.773274724129235e − 01
2.846333616870248e − 13
−8.272529134556158e − 01
7.479052807530189e − 01
−4.087405849665713e − 01
2.706342280378746e − 01
−3.820079746246349e + 00
−1.164024408225487e − 01

Ck =

Columns 1−3

−3.278655389103304e − 01 7.386542249660075e − 01 5.089113829393147e − 01

Columns 4−6

−2.848293353182419e − 01 −9.281300473869561e − 01 4.542316930222445e + 00

Columns 7−9

1.701064769673434e − 01 1.724613515296431e + 00 −3.764833293193810e + 00

Columns 10−12

−6.822982514891273e − 01 −2.548354148065294e + 00 −1.096816101047014e + 00

Columns 13−15

2.954691525044453e − 01 −2.779742954611004e − 01 −1.706955028714858e − 01

Columns 16−17

−3.982768152683027e − 01 1.359506914068743e + 00

Dk = 0

References

[1] K.J. Åström, U. Borisson, L. Ljung, B. Wittenmark, Theory and applications of self-tuning regulators, Automatica 13 (1977) 457–476.

[2] K.J. Åström, B. Wittenmark, On self tuning regulators, Automatica 9 (1973) 185–199.

[3] W.L. Bialkowski, Application of steady state Kalman filters — theory with field results, in: Proc. of the Joint Automatic Control Conf., Piscataway, NJ, IEEE Press, 1978. pp. 361–374.

[4] W.L. Bialkowski, Application of Kalman filters to the regulation of dead time processes, IEEE Trans. on Automatic control 28 (1983) 400–406.

[5] T. Cegrell, T. Hedqvist, Successful adaptive control of paper machines, Automatica 11 (1975) 53–59.

[6] G.A. Dumont, Application of advanced control methods in the pulp paper industry — a survey, Automatica 22 (1986) 143–153.

[7] M.S. Ma, D.C. Williams, A simplified adaptive model predictive controller, Tappi J. 71 (1988) 190–194.

[8] R.F. Sikora, W.L. Bialkowski, J.F. MacGregor, P.A. Tayler, A self-tuning strategy for moisture control in papermaking, in: Proc. of the American Control Conf., IEEE Press, Piscataway, NJ, 1984, pp. 54–61.

[9] D.B. Brewster, Profile control by distributed control systems: dream or reality?, Tappi J. 72 (1989) 75–81.

[10] R.D. Braatz, B.A. Ogunnaike, A.P. Featherstone. Identification, estimation, and control of sheet and film processes, in: Proc. of the IFAC World Congress, Tarrytown, New York, Elsevier Science Inc., 1996, pp. 319–324.

[11] L.G. Bergh, J.F. MacGregor, Spatial control of sheet and film forming processes, Can. J. of Chem. Eng. 65 (1987) 148–155.

[12] A. Halouskova, M. Karny, I. Nagy, Adaptive cross-direction control of paper basis weight, Automatica 29 (1993) 425–429.

[13] M.L. Tyler, M. Morari, Estimation of cross directional properties: scanning versus stationary sensors, AIChE J. 41 (1995) 846–854.

[14] ABB advertisement for AccuRay Hyperscan. Tappi J. 81 (1) (1998) 3–4.

[15] M. Matsuda, A new decoupling matrix method for paper basis weight profile control, in: Proc. of the IEEE Conf. on Decision and Control. Piscataway, NJ, IEEE Press, 1990, pp. 1592–1594.

[16] D.A. McFarlin, Control of cross-machine sheet properties on paper machines, in: Proc. of the 3rd Int. Pulp and Paper Process Control Symp., Vancouver, BC, Canada, 1983, pp. 49–54.

[17] K.J. Åström, Introduction to Stochastic Control Theory, Academic Press, San Diego, CA, 1970.

[18] A.E. Beecher, R.A. Bareiss, Theory and practice of automatic control of basis weight profiles, Tappi 53 (1970) 847–852.

[19] S.R. Duncan, The design of robust cross directional control systems for paper making. Control Systems Centre Report 807, UMIST, Manchester, UK, 1994.

[20] R. Gunawan, E.L. Russell, R.D. Braatz. Comparison of theoretical and computational characteristics of dimensionality reduction methods for large scale uncertain systems, J. of Process Control, in press.

[21] E.L. Russell, R.D. Braatz, Model reduction for the robustness margin computation of large scale uncertain systems, Computers and Chemical Engineering 22 (1998) 913–926.

[22] E.L. Russell, R.D. Braatz. The average-case identifiability and controllability of large scale systems, J. of Process Control, in press.

[23] M. Hovd, R.D. Braatz, S. Skogestad, SVD controllers for H_{2-}, H_{∞}-, and μ-optimal control, Automatica 33 (1996) 433–439.

[24] R.W. Brockett, J.L. Willems, Discretized partial differential equations: Examples of control systems defined by modules, Automatica 10 (1974) 507–515.

[25] A. Rigopoulos, Y. Arkun, F. Kayihan, Full CD profile control of sheet forming processes using adaptive PCA and reduced order MPC design, in: Proc of ADCHEM97, Banff, Canada, 1997, pp. 396–401.

[26] A. Rigopoulos, Y. Arkun, F. Kayihan, Model predictive control of CD profiles in sheet forming processes using full profile disturbance models identified by adaptive PCA, in: Proc of the American Control Conf., Piscataway, NJ, IEEE Press, 1997. pp. 1468–1472.

[27] P. Dave, D.A. Willig, G.K. Kudva, J.F. Pekny, F.J. Doyle, LP methods in MPC of large scale systems: application to paper-machine CD control, AIChE J. 43 (1997) 1016–1031.

[28] F.J. Doyle III, J.F. Pekny, P. Dave, S. Bose, Specialized programming methods in the model predictive control of large scale systems, Computers and Chemical Engineering 21 (1997) S847–S852.

[29] C.V. Rao, J.C. Campbell, J.B. Rawlings, S.J. Wright, Efficient implementation of model predictive control for sheet and film forming processes, in: Proc. of the American Control Conf., Piscataway, NJ, IEEE Press, 1997, pp. 2940–2944.

[30] J.G. VanAntwerp, R.D. Braatz, Model predictive control of large scale processes., J. of Process Control 10 (2000) 1–8.

[31] R.D. Braatz, J.G. VanAntwerp, Advanced cross-directional control, Pulp and Paper Canada 98 (7) (July 1997) T237–239.

[32] J.G. VanAntwerp, R.D. Braatz, Fast model predictive control of sheet and film processes, IEEE Trans. on Control Systems Technology 8 (2000) 408–417.

[33] J.G. VanAntwerp, R.D. Braatz, Model predictice control of large scale processes, in: Dynamics and Control of Process Systems, Vol. 1, Elsevier Science, Kidlington, UK, 1999, pp. 153–158.

[34] M. Hovd, R.D Braatz, S. Skogestad, Optimal and Robust Control of SVD Processes, Technical report, University of Trondheim, Trondheim, Norway, 1996.

[35] D. Laughlin, M. Morari, R.D. Braatz, Robust performance of cross-directional basis-weight control in paper machines, Automatica 29 (1993) 1395–1410.

[36] M.J. Grimble, J. Fotakis, The design of strip shape control systems for Sendzimir mills, IEEE Trans. on Automatic Control 27 (1982) 656–666.

[37] M. Hovd, R.D. Braatz, S. Skogestad, On the structure of the robust optimal controller for a class of problems, in: Proc. of the IFAC World Congress, Vol. IV, Tarrytown, New York, Elsevier Science Inc., 1993, pp. 27–30.

[38] M. Hovd, R.D. Braatz, S. Skogestad, SVD controllers for H_{2-}, H_{∞}-, and μ-optimal control, in: Proc. of the American Control Conf., Piscataway, NJ, IEEE Press, 1994, pp. 1233–1237.

[39] G.E. Stewart, D.M. Gorinevsky, G.A. Dumont, Spatial loop-shaping: A case study on cross-directional profile control, in: Proc. of the American Control Conf., Piscataway, NJ, IEEE Press, 1999, pp. 3098–3103.

[40] G.H. Golub, C.F. van Loan, Matrix Computations, John Hopkins University Press, Baltimore, MD, 1983.

[41] A.P. Featherstone, R.D. Braatz, Control-oriented modeling of sheet and film processes, AIChE J. 43 (1997) 1989–2001.

[42] R. Bellman, Introduction to Matrix Analysis, McGraw-Hill, New York, 1960.

[43] S. Skogestad, I. Postlethwaite, Multivariable Feedback Control: Analysis and Design, John Wiley & Sons, New York, 1996.

[44] R.G. Wilhelm Jr., M. Fjeld, Control algorithms for cross directional control, in: Proc. of the 5th Int. IFAC/IMEKO Conf. on Instrum. and Autom. in the Paper, Rubber, Plastics, and Polymerization Industries, Antwerp, Belgium, 1983, pp. 163–174.

[45] P. Lundstrom, S. Skogestad, Z.-Q. Wang, Trans. Inst. Modeling Control 13 (1998) 241–252.

[46] M. Morari, E. Zafiriou, Robust Process Control. Prentice Hall, Englewood Cliffs, NJ, 1989.

[47] R.D. Braatz, M. Morari, A multivariate stability margin for systems with mixed time-varying parameters, Int. J. of Robust and Nonlinear Control 7 (1997) 105–112.

[48] K. Poola, A. Tikku, Robust performane against time-varying structured perturbations, IEEE Trans. on Automatic Control 40 (1995) 1589–1602.

[49] P.M. Young, M.A. Dahleh, Robust l_p stability and performance, Systems and Control Letters 25 (1995) 305–312.
[50] A.P. Featherstone, R.D. Braatz, Integrated robust identification and control of large scale processes, Ind. Eng. Chem. Res. 37 (1998) 97–106.
[51] K. Zhou, J.C. Doyle, K. Glover, Robust and Optimal Control. Prentice Hall, Engelwood Cliffs, NJ, 1995.
[52] G.J. Balas, J.C. Doyle, K. Glover, A.K. Packard, R.S.R. Smith, μ-Analysis and Synthesis Toolbox (μ-Tools): Matlab Functions for the Analysis and Design of Robust Control Systems. The MathWorks, Inc, Natick, MA, 1992.
[53] R.Y. Chiang, M.G. Safonov, Robust Control Toolbox: For Use With MATLAB, The Mathworks Inc., Natick, MA, 1992.
[54] E.L. Russell, C.P.H. Power, R.D. Braatz, Multidimensional realizations of large scale uncertain systems for multivariable stability margin computation, Int. J. of Robust and Nonlinear Control 7 (1997) 113–125.
[55] E.L. Russell, R.D. Braatz, Analysis of large scale systems with model uncertainty, actuator and state constraints, and time delays, in: *AIChE Annual Meeting*, Chicago, IL, November 1996, Paper 45a.
[56] J.C. Doyle, Analysis of feedback systems with structured uncertainties, IEE Proc. D, Control Theory and Applications 129 (1982) 242–250.
[57] A. Packard, J.C. Doyle, The complex structured singular value, Automatica 29 (1993) 71–109.
[58] R.D. Braatz, O.D. Crisalle, Robustness analysis for systems with ellipsoidal uncertainty, Int. J. of Robust and Nonlinear Control 8 (1998) 1113–1117.
[59] J. Chen, M.K.H. Fan, C.N. Nett, Structured singular values and stability analysis of uncertain polynomials, part 1: the generalized μ, Systems and Control Letters 23 (1994) 53–65.
[60] J. Chen, M.K.H. Fan, C.N. Nett, Structured singular values and stability analysis of uncertain polynomials, part 2: a missing link, Systems and Control Letters 23 (1994) 97–109.
[61] R.D. Braatz, E.L. Russell, Robustness margin computation for large scale systems, Computers and Chemical Engineering 23 (1999) 1021–1030.
[62] R.D. Braatz, P.M. Young, J.C. Doyle, M. Morari, Computational complexity of μ calculation, IEEE Trans. on Automatic Control 39 (1994) 1000–1002.
[63] M.G. Safonov, Stability margins of diagonally perturbed multivariable feedback systems, IEE Proc. D, Control Theory and Applications 129 (1982) 251–256.
[64] J.G. VanAntwerp, R.D. Braatz, N.V. Sahinidis, Globally optimal robust control for systems with nonlinear time-varying pertubations, Computers & Chemical Engineering 21 (1997) S125–S130.
[65] J.G. VanAntwerp, R.D. Braatz, N.V. Sahinidis, Globally optimal robust process control, J. of Process Control 9 (1999) 375–383.
[66] S. Boyd, L. El Ghaoui, E. Feron, V. Balakrishnan, Linear Matrix Inequalities in System and Control Theory, volume 15 of Studies in Applied Mathematics. SIAM, Philadelphia, PA, 1994.
[67] Y. Nesterov, A. Nemirovskii, Interior Point Polynomial Algorithms in Convex Programming, volume 13 of Studies in Applied Mathematics, SIAM, Philadelphia, PA, 1994.
[68] K. Glover, All optimal Henkal-norm approximations of linear multivariable systems and their L^{∞}- error bounds, Int. J. of Countrol 39 (1984) 1115–1193.
[69] Balas, G.J. Robust Control of Flexible Structures: Theory and Experiments, PhD thesis, California Institute of Technology, Pasadena, 1990.
[70] R.T. Reichert, Robust autopilot design using μ-synthesis, in: Proc. of the American Control Conf. Piscataway, NJ. IEEE Press, 1991, pp. 2368–2373.
[71] Jackson, P. Applying μ-synthesis to missile autopilot design, in: Proc. of the IEEE Conf. on Decision and Control, Piscataway, NJ, IEEE Press, 1990, pp. 2993–2998.
[72] D.F. Enns, Structured singular value synthesis design example: rocket stabilization, in: Proc. of the American Control Conf, Piscataway, NJ IEEE Press, 1987. pp. 2514–2520.
[73] A.P. Featherstone, R.D. Braatz, Modal-based cross-directional control, Tappi J. 82 (1999) 203–207.
[74] D.-W. Gu, M.C. Tsai, I. Postlethwaite, Improved formulae for the 2-block super-optimal solution, Automatica 26 (1990) 437–440.
[75] H. Kwakernaak, A polynomial approach to minimax frequency domain optimization of multivariable feedback systems, Int. J. of Control 44 (1986) 117–156.
[76] M.C. Tsai, D. Gu, I. Postlethwaite, A state-space approch to super-optimal H_{∞} control problems, IEEE Trans. on Automatic Control 33 (1988) 833–843.
[77] A.P. Featherstone, R.D. Braatz, Input design for large scale sheet and film processes, Ind. Eng. Chem. Res. 37 (1998) 449–454.
[78] J.G. VanAntwerp, D.L. Ma, R.D. Braatz, When is constrained control necessary for large scale processes, In: Proc. of the American Control Conf. Piscataway, NJ, IEEE Press, 2000, pp. 4244–4248.
[79] K.J. Åström, B. Wittenmark, Computer Controlled Systems: Theory and Design, Prentice Hall, Englewood Cliffs, NJ, 1984.
[80] R.D. Braatz, M.L. Tyler, M. Morari, F.R. Pranckh, L. Sartor, Identification and cross-directional control of coating processes, AIChE J. 38 (1992) 1329–1339.
[81] P.J. Campo, M. Morari, Robust control of processes subject to saturation nonlinearities, Computers and Chemical Engineering 14 (1990) 343–358.
[82] S. Mhatre, C.B. Brosilow, Multivariable model state feedback, in: Proc. of the IFAC World Congress. Volume M, Piscataway, NJ, IEEE Press, 1996, pp. 139–144.
[83] A. Zheng, M.V. Kothare, M. Morari, Antiwindup design for internal model control, Int. J. of Control 60 (1994) 1015–1024.
[84] E.M. Heaven, T.M. Kean, I.M. Jonsson, M.A. Manness, K.M. Vu, R.N. Vyse, Applications of system identification to paper machine model development and controller design, in: Second IEEE Conf. on Control Applications, Vancouver, BC, Canada, 13–16 September 1993, pp. 227–233.
[85] R.D. Braatz, Internal model control, in: W.S. Levine (Ed.), The Control Handbook, CRC Press, Boca Raton, FL, 1995, pp. 215–224.
[86] I.G. Horn, J.R. Arulandu, C.J. Gombas, J.G. VanAntwerp, R.D. Braatz, Improved filter design in internal model control, Ind. Eng. Chem. Res. 35 (1996) 3437–3441.
[87] J.C. Doyle, Structured uncertainty in control system design, in: Proc. of the IEEE Conf. on Decision and Control, Piscataway, NJ, IEEE Press, 1985, pp. 260–265.
[88] J.G. VanAntwerp, Globally Optimal Robust Control for Large Scale Systems, PhD thesis, University of Illinois, Urbana, IL, 1999.
[89] J.H. Ly, R.Y. Chiang, K.C. Goh, M.G. Safonov, Multiplier K_m/μ-analysis — LMI approach, in: Proc. of the American Control Conf., Piscataway, NJ, IEEE Press, June 1995, pp. 431–436.
[90] J.G. VanAntwerp, R.D. Braatz, A tutorial on linear and bilinear matrix inequalities, J. of Process Control 10 (2000) 363–385.
[91] R.D. Braatz, M. Morari, μ-Sensitivities as an aid for robust identification, in: Proc. of the American Control Conf., Piscataway, NJ, IEEE Press, 1991, pp. 231–236.
[92] M.K.H. Fan, A.L. Tits, A measure of worst-case H_{∞} performance and of largest acceptable uncertainty, Systems and Control Letters 18 (1992) 409–421.
[93] S. Skogestad, M. Morari, Robust performance of decentralized control systems by independent designs, Automatica 29 (1989) 119–125.
[94] R.S.R. Smith, Model Validation for Uncertain Systems, PhD thesis, California Institute of Technology, Pasadena, 1990.
[95] A. Packard, P. Pandey, Continuity properties of the real/complex structured singular value, IEEE Trans. on Automatic Control 38 (1993) 415–428.