Proceedings of the
American Control Conference
San Francisco, California • June 1993

A CACSD PACKAGE FOR $\mathcal{H}_\infty$ AND LTR DESIGN

Hans Henrik Niemann and Jakob Stoustrup
Mathematical Institute, Technical University of Denmark, DK-2800 Lyngby, Denmark.

ABSTRACT.

A CACSD Matlab package for $\mathcal{H}_2$, $\mathcal{H}_\infty$ and Loop Transfer Recovery (LTR) design and related methods/tools is presented. The $\mathcal{H}_2$ and $\mathcal{H}_\infty$ design methods are implemented in the most general form (i.e. the singular approach), where no conditions are imposed on the direct terms in the controlled systems. The LTR methods are based both on classical methods, such as LQG or eigenstructure assignment, and on $\mathcal{H}_2$, $\mathcal{H}_\infty$ methods. All methods are implemented in .m functions in the Matlab standard way. The numerical methods for the .m functions are also described.

1. INTRODUCTION.

Today, one of the most popular ways to make new theoretical results in the field of control systems available, is by implementing the results in numerical programs on standard control software platforms. One of the most used standard control software packages is the MATLAB program from MathWorks [22]. Several control toolboxes have been developed, commercial as well as non-commercial, for the MATLAB package.

In this paper we present a non-commercial CACSD (Computer Aided Controller System Design) package, based on the MATLAB program, for $\mathcal{H}_2$ and $\mathcal{H}_\infty$ design, LTR design and related methods. The control package is developed in the last two years in close connection with our theoretical research, especially in the field of LTR design, [13,18]. With this purpose in mind, only state space methods have been implemented for the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ problems. For alternative approaches the present toolbox comply to a wide extent with existing toolboxes. The specialized solution algorithms have been implemented, however, in order to threat specific numerical difficulties related to the design problem dealt with.

The toolbox is a result of a theoretical research in the field of LTR design methods, especially those based on $\mathcal{H}_2$ or $\mathcal{H}_\infty$ optimization. Using a systematic description of the LTR problem based on recovery errors [13], the related $\mathcal{H}_2$ and $\mathcal{H}_\infty$ standard design problems are in general singular, i.e. the direct feedthrough terms does not necessarily have maximal rank. Normally, one of the direct feedthrough terms will be zero and the other terms depend on the specific LTR design problem formulation, [13,18]. Singular $\mathcal{H}_2$ or $\mathcal{H}_\infty$ problems can be solved in two ways: Perturbing the direct feedthrough terms and using the regular theory developed by Doyle et.al. [6] or using a method which explicitly takes care of the singularity in the direct feedthrough terms. Such a method has been derived by Stoorvogel [16,17] called the singular $\mathcal{H}_\infty$ approach. Our toolbox is based on the singular approach due to the need of explicit solutions in the LTR design cases. Moreover the regular approach is also included in the singular case as a special case [16,17]. As a comparison, the two Matlab toolboxes by Balas et.al. [1] and Chiang and Safonov [4] are only based on the regular $\mathcal{H}_2$ or $\mathcal{H}_\infty$ approach.

The .m functions are divided into two main groups. In the first group the general .m functions are collected. Here we have functions for: Solving Quadratic Matrix Inequalities, (QMI, related to the singular approach), solving regular and singular Riccati equations, almost disturbance decoupling problems, $\mathcal{H}_2$ and $\mathcal{H}_\infty$-norm calculation. The second includes more special .m functions which are based on the general functions and the functions available in MATLAB. Here we have functions for: Transformations between different controller configurations, $\mathcal{H}_2$ and $\mathcal{H}_\infty$ design of different problems, LTR-related methods special based on $\mathcal{H}_2$ and $\mathcal{H}_\infty$ optimization and functions for calculating transfer functions.

In this paper, we will first introduce the singular $\mathcal{H}_\infty$ approach and the LTR design concept based on recovery errors. Afterwards, the structure of the toolbox and the different functions will be described.

This paper is <u>not</u> written as a manual for the toolbox. The intention is to give an overview of the facilities in the toolbox.

2. THE STANDARD FOUR BLOCK PROBLEM.

The standard four block setup is shown in fig. 2.1, where w is the exogenous input, u is the control signal, z is the output to be controlled and y is the measured output. Σ_G represents a generalized plant and Σ_K represents a controller.

Let the transfer function of the controller Σ_K be given by K(s). Then the closed loop transfer function from w to z becomes:

$$G_{zw}(s) = T_{zw}(s) + T_{zu}(s)K(s)(I - T_{yu}(s)K(s))^{-1}T_{yw}(s) \tag{1}$$

where $T_{zw}(s)$, $T_{zu}(s)$, $T_{yw}(s)$ and $T_{yu}(s)$ are the open loop transfer functions from $w \to z$, $u \to z$, $u \to y$ and $w \to y$ respectively.

It is assumed that the plant Σ is finite dimensional, linear, time invariant, (FDLTI) and, hence, can be represented by the following state-space realization:

$$\Sigma: \begin{cases} \dot{x} = Ax + Bu + Ew \\ y = C_1x + D_{11}u + D_{12}w \\ z = C_2x + D_{21}u + D_{22}w \end{cases} \tag{2}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $w \in \mathbb{R}^l$, $y \in \mathbb{R}^q$ and $z \in \mathbb{R}^p$.

The design problem is to design a FDLTI controller $u(s) = K(s)y(s)$, if possible, such that a suitable norm of the resulting closed-loop transfer function from w to z is smaller than a specified level.

In the following, it will be assumed that the diagonal terms in D (D_{11} and D_{22}) is zero due to the applied $\mathcal{H}_2$ and $\mathcal{H}_\infty$ design methods. The system Σ with all non-zero D-terms can be transformed into a new system Σ_{LS} with $D_{11} = D_{22} = 0$ using Loop Shifting [16]. Controller design for Σ can instead be derived for the loop shifted system Σ_{LS} without affecting the closed-loop $\mathcal{H}_\infty$ norm.

The four block problem setup allows directly to formulate control design problems such as e.g. robust stability design, servo design, feedforward design, etc. in a simple way and in the <u>same</u> setup.

Design methods based on the setup in fig. 2.1 are considered in next section.

3. DESIGN METHODS.

In the following the singular $\mathcal{H}_\infty$ design method will be considered followed by a short description of the LTR design principle.

3.1 The singular $\mathcal{H}_\infty$ approach.

Consider a FDLTI system:

$$\Sigma: \begin{cases} \dot{x} = Ax + Bu + Ew, & x \in \mathbb{R}^n, \ u \in \mathbb{R}^m, \ w \in \mathbb{R}^l \\ y = C_1x + D_{12}w, & y \in \mathbb{R}^q \\ z = C_2x + D_{21}u, & z \in \mathbb{R}^p \end{cases} \tag{3}$$

The singular approach taken will be based on the results in [16] which is a very general approach, and in particular it does not impose any assumption on the direct feedthrough term of the standard problem (the four block problem), as it has been done in other approaches as for instance [6]. The general approach where nothing is assumed about the 'D' matrices is normally referred to as the singular approach, contrary to the regular approach, where the direct feedthrough term has to have a special form. The necessary preliminaries for the singular $\mathcal{H}_\infty$ method will be introduced in the following.

Our design problems are as follows:

The $\mathcal{H}_\infty$ case.

We assume that $\gamma > 0$ has been given. We wish to design, if possible, an internally stabilizing FDLTI controller $u = K(s)y$ such that the $\mathcal{H}_\infty$ norm of the resulting closed-loop transfer function from w to z is smaller than γ.

Note that only suboptimal controllers is obtained in the $\mathcal{H}_\infty$ case, due to the problem formulation.

First, let's define the following matrix functions:

$$F(\gamma,P) = \begin{bmatrix} A^TP + PA + C_2^TC_2 + \gamma^{-2}PEE^TP & PB + C_2^TD_{21} \\ B^TP + D_{21}^TC_2 & D_{21}^TD_{21} \end{bmatrix} \tag{4}$$

$$G(\gamma,Q) = \begin{bmatrix} AQ + QA^T + EE^T + \gamma^{-2}QC_2^TC_2Q & QC_1^T + ED_{12}^T \\ C_1Q + D_{12}E^T & D_{12}D_{12}^T \end{bmatrix}$$

This work is supported in part by the Danish Technical Research Council under grant no. 16-4885 and grant no. 26-1830.

$$L(\gamma,P,s) = \begin{bmatrix} sI - A - \gamma^{-2}EE^TP & -B \end{bmatrix}$$

$$M(\gamma,Q,s) = \begin{bmatrix} sI - A - \gamma^{-2}QC_2^TC_2 \\ -C_1 \end{bmatrix}$$

$$G(s) = C_2(sI - A)^{-1}B + D_{21}, \quad H(s) = C_1(sI - A)^{-1}E + D_{12}$$

Assumption 3.1.
It is assumed that the systems (A,B,C_2,D_2) and (A,E,C_1,D_1) have no invariant zeros in C^0.

Lemma 3.2.
Consider a system Σ which satisfies Assumption 3.1. Let $\gamma_\infty > 0$ be given. Then, there exists a FDLTI controller $u = K(s)y$ for which the resulting closed loop system is internally stable, and for which the transfer function from w to z has $\mathcal{H}_\infty$ norm smaller than γ_∞ if and only if there exist $P \geq 0$ and $Q \geq 0$ for which:

(1) $F(\gamma,P) = \begin{bmatrix} C_{2,P}^T \\ D_P^T \end{bmatrix} \times \begin{bmatrix} C_{2,P} & D_P \end{bmatrix} \geq 0$

(2) $G(\gamma,Q) = \begin{bmatrix} E_Q \\ D_Q \end{bmatrix} \times \begin{bmatrix} E_Q^T & D_Q^T \end{bmatrix} \geq 0$

(3) $\text{rank } F(\gamma,P) = \text{normrank } G$

(4) $\text{rank } G(\gamma,Q) = \text{normrank } H$

(5) $\text{rank}\begin{bmatrix} L(\gamma,P,s) \\ F(\gamma,P) \end{bmatrix} = n + \text{normrank } G, \ \forall s \in \bar{C}^+$

(6) $\text{rank}\begin{bmatrix} M(\gamma,Q,s) & G(\gamma,Q) \end{bmatrix} = n + \text{normrank } H, \ \forall s \in \bar{C}^+$

(7) $\rho(PQ) < \gamma^2$

The proof of Lemma 3.2 can be found in [16]. We shall refer to condition (1) as the Quadratic Matrix Inequality, and any $P \geq 0$ satisfying (1) will be called a solution to the Quadratic Matrix Inequality. Analogously we shall call (2) the Dual Quadratic Matrix Inequality, and refer to solutions of the Dual Quadratic Matrix Inequality any $Q \geq 0$ satisfying (2). Conditions (3) and (5) guarantees that a solution to the Quadratic Matrix Inequality is unique and of minimal rank (and dually for the Dual Quadratic Matrix Inequality with (4) and (6)). (7) is a typical $\mathcal{H}_\infty$ coupling condition, which also appears in [6].

The admissible controllers will be given in the following in terms of the matrices for certain transformations of Σ. For this, we will need the following matrices:

$$\begin{aligned} A_P &= A + \gamma^{-2}EE^TP \\ C_{1,P} &= C_1 + \gamma^{-2}D_1E^TP \\ A_{P,Q} &= A_P + \gamma^{-2}(I-\gamma^{-2}QP)^{-1}QC_{2,P}^TC_{2,P} \\ B_{P,Q} &= B + \gamma^{-2}(I-\gamma^{-2}QP)^{-1}QC_{2,P}^TD_P \\ E_{P,Q} &= (I-\gamma^{-2}QP)^{-1}E_Q \end{aligned} \tag{5}$$

where $C_{2,P}$, D_P and E_Q are given by Lemma 3.2.

We shall refer to the system where A_P, $C_{1,P}$, $C_{2,P}$ and D_P substitute A, C_1, C_2 and D_{21} as the full information transform of the system Σ. Substituting $A_{P,Q}$, $B_{P,Q}$, $E_{P,Q}$ and $D_{P,Q}$ for the corresponding variables in previous system will be referred to as the full control transformation. Note that the two transformations are totally independent in the $\mathcal{H}_2$ case. Moreover, only the matrices E, C_2, D_{12} and D_{21} are changed.

Now, consider the transformed system Σ_{PQ}:

$$\Sigma_{PQ} : \begin{cases} \dot{x}_{P,Q} = A_{P,Q}x_{P,Q} + B_{P,Q}u_{P,Q} + E_{P,Q}w \\ y_{P,Q} = C_{1,P}x_{P,Q} + D_Qw \\ z_{P,Q} = C_{2,P}x_{P,Q} + D_Pu_{P,Q} \end{cases} \tag{6}$$

The connection between the original system Σ and the transformed system is given in the following lemma:

Lemma 3.3.
Let's use an arbitrary controller $u = K(s)y$. Then, the following two statements are equivalent:

1. The controller $K(s)$ applied on Σ is internally stabilizing and the resulting closed loop transfer function from w to z is strictly proper and has $\mathcal{H}_\infty$ norm $< \gamma_\infty$.

2. The controller $K(s)$ applied to the transformed system Σ_{PQ} is internally stabilizing and the resulting closed loop transfer functions from w to $z_{P,Q}$ is strictly proper and has $\mathcal{H}_\infty$ norm $< \gamma_\infty$.

In terms of the transformed system matrices we can compute the desired $\mathcal{H}_\infty$ controllers. In this paper we will only look at full-order observer based controllers, because a systematic design method exist for obtaining the specified norms for the transformed system. However, it is not always necessary to use full-order controllers (or minimal-order controllers) especially if we don't go too close to the infimally achievable norms.

Theorem 3.4.
Let $A_{P,Q}$, $B_{P,Q}$ and $C_{1,P}$ be as above. Let L be a state feedback, such that $A_{P,Q} + B_{P,Q}L$ is stable, and such that:

$$\|(C_{2,P} + D_PL)(sI - A_{P,Q} - B_{P,Q}L)^{-1}\|_\infty < \gamma/(3\|E_{P,Q}\|) \tag{7}$$

Let M be an output injection, such that $A_{P,Q} + MC_{1,P}$ is stable and further:

$$\|(sI - A_{P,Q} - MC_{1,P})^{-1}(E_{P,Q} + MD_Q)\|_\infty < \tilde{\gamma} \tag{8}$$

where

$$\tilde{\gamma} = \min\{\gamma/(3\|D_PL\|), \|E_{P,Q}\|/\|B_{P,Q}L\|\}$$

Then the controller:

$$u = -L(sI - A_{P,Q} - B_{P,Q}L - MC_{1,P})^{-1}My \tag{9}$$

makes the $\mathcal{H}_\infty$ norm of the resulting closed loop transfer function smaller than γ_∞.

The significance of Theorem 3.4 is to transform the original $\mathcal{H}_2$ or $\mathcal{H}_\infty$ problem to two disturbance attenuation problems, which can be solved by well known methods, see e.g. [16,17]. A thorough, numerical analysis of this approach is given in [14].

3.2. LTR design.

The LTR design method has first been introduced by Doyle and Stein [7] as an attractive design principle for full-order observer-based controllers. Afterwards, the LTR principle has been considered in many papers, see e.g. [4,8,10,15,18,19].

The LTR design method consists of two steps. First, a target static state feedback gain is determined such that the design specifications are satisfied. Second, a dynamic controller (normally a full-order observer-based controller) must be derived such that the properties from the target loop is recovered as well as possible in the final loop (the LTR step).

The motivation for applying two steps design methods, is the extra freedom available in the target design compared to one shot methods as $\mathcal{H}_2$ or $\mathcal{H}_\infty$ methods. The choice of design method for the static state feedback is free. One possibility is to use an optimization method for the calculation of the feedback gain, such that some explicit conditions (i.e. limit gains, time response conditions etc.) are satisfied. When the target loop is recovered in the LTR step, the conditions, satisfied by the target loop will also be recovered to some extent in the final loop, (but there is no guarantee how well in general).

Let's apply the LTR technique method for the controller design of the standard four block problem in (3). First, the case of full state information is considered, i.e. $C_1 = I$, $D_{12} = 0$, and a (target) state feedback controller $K(s) = F$ is designed. The closed-loop target transfer function is given by $G_{zw,F}(s)$:

$$G_{zw,F}(s) = (C_2 + D_{21}F)(sI - A - BF)^{-1}E \tag{10}$$

Second, the LTR step is performed where in a dynamic controller $K(s)$ is designed such that the closed-loop transfer function $G_{zw}(s)$ recover the target loop $G_{zw,F}(s)$ as well as possible in a suitable sense. Using the recovery error description [13], we are now able to formulate a four block LTR design problem.

Problem 3.5.
Let the target closed-loop transfer function and the full-loop transfer function be given by $G_{zw,F}(s)$ and $G_{zw}(s)$, respectively. The closed-loop recovery error $E_{zw}(s)$ is defined by:

$$E_{zw}(s) = G_{zw,F}(s) - G_{zw}(s) \tag{11}$$

Find, if possible, a FDLTI controller $K(s)$ which minimize a suitable norm or make a suitable norm of the recovery error $E_{zw}(s)$ smaller than a specified level.

The recovery error in (11) has the following state-space realization in the standard four block setup (i.e. fig. 2.1):

$$\Sigma_{E}: \begin{cases} \dot{x} = \begin{bmatrix} A+BF & 0 \\ 0 & A \end{bmatrix} x + \begin{bmatrix} 0 \\ B \end{bmatrix} u + \begin{bmatrix} E \\ E \end{bmatrix} w \\ y = \begin{bmatrix} 0 & -C_1 \end{bmatrix} x - D_{12} w \\ z = \begin{bmatrix} C_2+D_{21}F & C_2 \end{bmatrix} x - D_{21} u \end{cases} \tag{12}$$

With this state space description of the recovery error $E_{zw}(s)$, controllers designed by $\mathcal{H}_2$ or $\mathcal{H}_\infty$ optimization techniques can directly be applied. This has been done in [18] where the sensitivity and the closed-loop recovery design problem has been solved by $\mathcal{H}_\infty$ optimization. The LTR design problem formulated in Problem 3.5 has also been considered in [4]. Let's instead apply a Luenberger observer described by, [8]:

$$\Sigma_L: \begin{cases} \dot{z} = Jz + Nu + Py \\ u = Rz + Vy \end{cases} \tag{13}$$

where the Luenberger matrices J,N,P,R and V have to satisfy:

$$\begin{aligned} &\lambda(J) \subset \mathbb{C}^- \\ &TA - JT = PC_1 \\ &N = TB \\ &F = RT + VC_1 \end{aligned} \tag{14}$$

for some matrix T.

When the Luenberger observer based controller is applied, the recovery error in Problem 3.5 can be written in a more convenient form.

Lemma 3.6.

Define

$$M_f(s) = R(sI - J)^{-1}(TE - PD_{12}) - VD_{12} \tag{15}$$

Then

$$E_{zw}(s) = G_{zu,F}(s)M_f(s),\ G_{zu,F}(s) = (C_2+D_{21}F)(sI-A-BF)^{-1}B+D_{21} \tag{16}$$

Proof. See [4]. Lemma 3.6 is a generalization of the closed loop recovery error functions derived in [13].

The matrix valued function $M_f(s)$ plays a very central role in LTR design of observer based controllers [4,13,19]. Therefore we will refer to $M_f(s)$ as the recovery matrix.

Using the recovery error description as in Lemma 3.6, we get the following central results:

Lemma 3.7.

Assume that $G_{zu,F}(s)$ has full row rank for almost every $s \in \mathbb{C}$ and let the recovery error be as in Lemma 3.6. Exact recovery, i.e. $E_{zw}(s) = 0$, is obtained if and only if:

$$M_f(s) = 0 \tag{17}$$

Further, asymptotic recovery is possible if and only if for all $\varepsilon > 0$ there exist a controller $C_\varepsilon(s)$ such that:

$$\|E_{zw,\varepsilon}(s)\|_{H} < \varepsilon \tag{18}$$

or equivalently, if and only if for all $\ell > 0$ there exists K_ℓ such that:

$$\|M_{f,\ell}(s)\|_{H} < \ell \tag{19}$$

$\|\cdot\|_H$ is either $\|\cdot\|_2$ or $\|\cdot\|_\infty$.

Proof. Lemma 3.7 is trivial by observing that $G_{zu,F}(s)$ depend only on the system Σ_T and the target design F, not of the observer design.

To obtain a more general LTR design problem than in Lemma 3.6, let's introduce a FDLTI system, which is left invertible (as a rational matrix) given by:

$$W(s) = D_w + C_w(sI-A_w)^{-1}B_w \tag{20}$$

It is further assumed that Σ_w is stabilizable and detectable. Now we apply W(s) as a weight function for $M_f(s)$.

Definition 3.8.

Let $M_f(s)$ be the recovery matrix corresponding to a given Luenberger observer based controller, and let $W(s)$ be a transfer function as above. Then we define the (general) recovery error $E(s)$ by:

$$E(s) = W(s)M_f(s) \tag{21}$$

It is important to note that the LTR formulation in the version given in Problem 3.5 does not include the same freedom as the above LTR formulation based on Luenberger observers.

To make the LTR problem constructively, we will apply the Q-observer [13] which gives the following recovery matrix:

$$\begin{aligned} M_f(s) = {} & F(sI-A-KC_1)^{-1}(E+KD_{12}) \\ & + Q(s)[C_1(sI-A-KC_1)^{-1}(E+KD_{12})+D_{12}] \end{aligned} \tag{22}$$

where K is a preliminary observer design, Q, $Q \in \mathcal{RH}_\infty$ is the controller to be designed. The recovery error in Definition 3.8 has the following state-space form:

$$\Sigma_{E}: \begin{aligned} \dot{x} &= \begin{bmatrix} A+KC_1 & 0 \\ B_wF & A_w \end{bmatrix} x + \begin{bmatrix} 0 \\ B_w \end{bmatrix} u + \begin{bmatrix} E+KD_{12} \\ 0 \end{bmatrix} w \\ y &= \begin{bmatrix} C_1 & 0 \end{bmatrix} x + D_{12} w \\ z &= \begin{bmatrix} D_wF & C_w \end{bmatrix} x + D_w u \end{aligned} \tag{23}$$

The above LTR design problem is a standard problem where e.g. $\mathcal{H}_2$ or $\mathcal{H}_\infty$ optimization can be applied directly. However, by using the special structure in (23), the $\mathcal{H}_2$ or $\mathcal{H}_\infty$ optimization can be simplified, because some of the involved equations get very simple [13,18]. This can be used both for simplifying the computer programs for calculating the LTR controllers and also reducing the order of the final LTR controllers.

If an optimization of (23) has been done directly, the final LTR controllers will be of order $2n + n_w$ (a full-order observer + Q(s)). Applying the structure in (23), the final LTR controller can be reduced to order $n + n_w$ without any approximations [13,18].

4. THE CACSD PACKAGE.

The MATLAB CACSD package [11] is described in the following. First, the structure of the package is considered followed by a description of the .m functions included in the program package.

4.1 The Program Structure.

The program package is build upon the standard .m functions in MATLAB together with some new .m functions solving some central equations/inequalities as e.g. Quadratic Matrix Inequalities, Riccati equations etc. Based on these .m functions, some more special .m functions are derived for solving more specialized problems as e.g. different $\mathcal{H}_2$/LTR or $\mathcal{H}_\infty$/LTR design problems, regular and singular $\mathcal{H}_2$ and $\mathcal{H}_\infty$ design etc. The structure of the package and it's connection with MATLAB is shown in fig. 4.1. Note that both the general and the specialized functions are directly available to the user.

The .m functions in the two boxes will be described in the following two sections.

4.2. General .m Functions.

The following general .m functions shown in Table 4.1 are derived for the CACSD package:

functions:	solution of:
h2norm	$\mathcal{H}_2$-norm computation
hinfnorm	$\mathcal{H}_\infty$-norm computation
ricc	Riccati equation
singric	Singular Riccati equation
qmi	Quadratic Matrix Inequality
tsqmi	Totally singular QMI
loopsh	Loop Shifting
factor	Factorization
addp	Almost Disturbance Decoupling Problem

Table 4.1. General functions in the package.

The function for $\mathcal{H}_2$ norm calculation can be applied to stable systems described by Σ_u: (A_u, B_u, C_u). The function use the controllability Grammian L_c of (A_u, B_u) and the observability Grammian of (C_u, A_u) L_o defined by:

$$\begin{aligned} A_uL_c + L_cA_u^T + B_uB_u^T = 0 \\ A_u^TL_o + L_oA_u + C_u^TC_u = 0 \end{aligned} \tag{24}$$

The $\mathcal{H}_2$-norm of G(s) is the given by [6]:

$$\|G_u(j\omega)\|_2^2 = \mathrm{trace}(C_uL_cC_u^T) = \mathrm{trace}(B_u^TL_oB_u) \tag{25}$$

with $G_u(s) = C_u(sI-A_u)^{-1}B_u$.

The function for calculating the $\mathcal{H}_\infty$ norm of a stable system described by Σ: (A,B,C,D) is based on the bisection algorithm in [2].
The algorithm is as follows:

1) Calculate an upper and an lower bound for the $\mathcal{H}_\infty$ norm, γ_l, γ_u.
2) $\gamma = (\gamma_l + \gamma_u)/2$.
3) Form M_γ and calculate the eigenvalues of M_γ.
4) If M_γ has no imaginary eigenvalues, $\gamma_u = \gamma$ else $\gamma_l = \gamma$.
5) If $(\gamma_u - \gamma_l) > 2\varepsilon\gamma_l$ goto 2
   else
6) Output γ.

where M_γ is given by:

$$M_\gamma = \begin{bmatrix} A_z - B_z R^{-1} D_z^T C_z & -\gamma B_z R^{-1} B_z^T \\ \gamma C_z^T S^{-1} C_z & -A_z^T + C_z^T D_z R^{-1} B_z^T \end{bmatrix}, \tag{26}$$
$$R = D_z^T D_z - \gamma^2 I,\ S = D_z D_z^T - \gamma^2 I$$

Note that γ is calculated with a relative accuracy of ε. The upper and lower bounds for γ in step 1 can be calculated using Hankel singular values. Another method can be found in [3].

In the ricc.m function the Riccati equation:

$$XA + A^TX + XRX + Q = 0 \tag{27}$$

is solved using Schur decomposition.

In the singric.m function the unique solution X to the following Riccati equation:

$$XA + A^TX - XBB^TP = 0 \tag{28}$$

satisfying $A - BB^TX$ is stable, is calculated. It is assumed that A has no imaginary axis eigenvalues and that (A,B) is a stabilizable pair. The solution can be given explicit by [18]:

$$X = -\Pi^T(\Pi G_c \Pi^T)^{-1}\Pi \tag{29}$$

where G_c is the controllability Grammian of (A,B) and Π is the orthogonal projection onto $X_-(A)^\perp$ along $X_-(A)$, the space of generalized stable eigenvectors of A. The singular Riccati equation in (28) appears in LTR design, see section 4.3.

Two functions (one general and one for the total singular case, D = 0) are available for calculating the solution of the Quadratic Matrix Inequality for the system Σ given by:

$$\Sigma: \begin{cases} \dot{x} = Ax + Bu + Ew, & x \in \mathbb{R}^n, u \in \mathbb{R}^m, w \in \mathbb{R}^q \\ z = Cx + Du, & z \in \mathbb{R}^p \end{cases} \tag{30}$$

such that the following three conditions are satisfied:

$$\begin{bmatrix} A^TP + PA + C^TC + \gamma^{-2}PEE^TP & PB + C^TD \\ B^TP + D^TC & D^TD \end{bmatrix} =: \begin{bmatrix} C_P^T \\ D_P^T \end{bmatrix} [\, C_P \;\; D_P \,] \geq 0$$

$$\text{rank } C_P = p \tag{31}$$

$(A + \gamma^{-2}EE^TP, B, C_P, D_P)$ is a minimum phase system.

The positive semidefinite solution for the Quadratic Matrix Inequality is derived by using the algorithm in [12]. If $\gamma > 0$ has been selected to small (in the $\mathcal{H}_\infty$ case), one or more of the conditions in (31) can not be satisfied which will be indicated by the function.

Functions for loop shifting based on the equations in [16] are implemented in different versions depending of one or both the diagonal elements in D (D_{11} and D_{22}) are non-zero.

The factor.m function calculate the factorization of a positive semidefinite matrix X:

$$X = Y^TY \tag{32}$$

The function is used for calculation of the transformed system Σ_{PQ} based on the solutions of the Quadratic Matrix Inequality and the Dual Quadratic Matrix Inequality.

In the last step of calculating $\mathcal{H}_2$ or $\mathcal{H}_\infty$ controllers by using the singular approach, involves solving one or two Almost Disturbance Decoupling Problems (ADDPs). Let's consider the minimum phase system Σ given by:

$$\Sigma: \begin{cases} \dot{x} = Ax + Bu + Ew, & x \in \mathbb{R}^n, u \in \mathbb{R}^m, w \in \mathbb{R}^q \\ z = Cx + Du, & z \in \mathbb{R}^p \end{cases} \tag{33}$$

The ADDP for Σ is to determine a state feedback gain u = Fx such that:

$$\|(C+DF)(sI-A-BF)^{-1}E\|_\infty < \varepsilon \tag{34}$$

where $\varepsilon > 0$ is specified.

The ADDP functions are based on three different methods for design the state feedback gain F which satisfies the norm inequality in (34). The applied methods are LQ-design, Eigenstructure assignment [19] and geometric methods [16,20,21]. It should be noted that the .m functions only handle minimum-phase systems, although of it is also possible sometimes to solve the ADDP for non-minimum phase systems, [20]. The reason is that the ADDPs appearing in $\mathcal{H}_2$ or $\mathcal{H}_\infty$ design are always related to minimum-phase systems.

4.3. Special .m Functions.

The package includes different types of special .m functions which are listed in table 4.2.

Translation functions
$\mathcal{H}_2$ and $\mathcal{H}_\infty$ related functions
LTR related functions
Transfer function calculations

Table 4.2. The categories of special .m functions.

The .m functions in the four groups are shortly described.

<u>Translation functions:</u>

This first group of functions has been created as a compromise between generality of functions, the need for specific implementations, and a not too large number of functions. For reducing the number of functions, the Luenberger observer has been applied in every function as the controller, except in the functions for design of specific controllers. The translation functions take care of the calculation of the Luenberger matrices for a variety of specific controller types. Functions for the following observer-based controllers has been derived:

- Full-order observer-based controllers
- Minimal-order observer-based controllers
- Q-observer-based controllers
- Full-order PI observer-based controllers
- Minimal-order PI observer-based controllers

Both matrices for the normal and the dual Luenberger observer can be calculated [13].

Moreover, some functions for calculating discrete-time systems from continuous-time systems and vice versa based on the Tustin approximation are also included.

<u>$\mathcal{H}_2$ and $\mathcal{H}_\infty$ related functions:</u>

The $\mathcal{H}_2$ and $\mathcal{H}_\infty$ related functions is one of the two main groups of special functions.

Different functions for $\mathcal{H}_2$ and $\mathcal{H}_\infty$ controller design for the four block standard setup described in (1) are available. In both the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ case we have functions for state-feedback, i.e. $C_1 = I$ and $D_{12} = 0$, for observer design, i.e. B = I and $D_{21} = 0$, and for full-order observer design. Further, depending of the design problem, functions for regular design (i.e. the direct terms has maximal rank), singular design and for totally singular design (i.e. the direct terms are zero), are available. In the singular case (general) case there is no conditions on the direct terms, so the singular case include the two other cases as special cases.

In the regular case, the functions directly give the matrices for the controllers, whereas in the two other cases, the matrices for the transformed system Σ_{PQ} are calculated. These matrices for Σ_{PQ} can directly be applied for the ADDPs. The general ADDP functions can be used directly in the state-feedback case and the observer case. In the full-order controller case, ADDP functions, based on the general ADDP functions, are available. In every $\mathcal{H}_2$ design function, γ_2^* is calculated.

<u>LTR related functions:</u>

The second main group of special functions is the LTR related functions. .m functions for both "classical" LTR design as well as for $\mathcal{H}_2/\mathcal{H}_\infty$ based LTR functions are derived. In the group of classical LTR design methods we have: LQG/LTR and ES/LTR (eigenstructure assignment based LTR [19]) design for both full-order as well as for minimal-order observer-based controllers. Further, functions for LTR design at both the input and the output loop breaking point are available.

In the group of $\mathcal{H}_2/\mathcal{H}_\infty$ based LTR functions, we have both functions for direct as well as for indirect LTR design. By indirect LTR design means an optimization of the recovery matrix (only for observer-based controllers), whereas direct LTR design is an optimization of a specific recovery error, [18]. The functions available for $\mathcal{H}_2$/LTR and $\mathcal{H}_\infty$/LTR design are based both on Problem 3.5, where the controller type is not specified, and the recovery error in Definition 3.8, where an observer-based controller is applied. Further, LTR design based on some specific choice of W(s) in (23) are also implemented. The specific LTR functions are : W(s) = I, (indirect design) $W(s) = S_{TR}(s)$, (sensitivity recovery) and $W(s) = G_{IO,TR}(s)$ (input-output recovery).

As mentioned above, the structure in the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ LTR problems are used in the .m functions for reducing the number of numerical operations and, more important, for getting more explicit expressions for the controller gains.
$\mathcal{H}_2$ or $\mathcal{H}_\infty$ LTR design based on the recovery error in Definition 3.8:

$$E(s) = W(s)M_f(s) \tag{35}$$

can be reduced to find the solution to a n_w'th order Quadratic Matrix Inequality associated with the system Σ_W:

$$\Sigma_W: \begin{cases} \dot{x}_w = A_w x_w + B_w u_w \\ z_w = C_w x_w + D_w u_w \end{cases} \tag{36}$$

and the usual Dual Quadratic Matrix Inequality for Σ_F in (23) of order $n + n_w$. Note that the Quadratic Matrix Inequality for Σ_w is independent of γ which mean that solvability of the corresponding $\mathcal{H}_\infty$/LTR problem depend only on solvability of the Dual Quadratic Matrix Inequality for Σ_F. Further, if D_w is injective, the Quadratic Matrix Inequality is equivalent with the singular Riccati equation given by:

$$A_w^T P + PA_w - PB_w(D_w^T D_w)^{-1}B_w^T P = 0 \tag{37}$$

The "static-feedback" gain R in the Luenberger observer (13) can always be determined as [18]:

$$R = [\, F \;\; -L_2 \,] \tag{38}$$

where F is the target feedback gain and L_2 must satisfies an n_w'th order ADDP. If D_w is injective, L_2 can be determined as:

$$L_2 = -D_F^{+} C_{2F} \tag{39}$$

which solves the DDP exactly.

An equivalent simplification can also be obtained when the $\mathcal{H}_\infty$/LTR design problem is based on the recovery error in (11), where the controller type is not specified. This is applied in the implementation of the .m functions for this LTR approach.

The output from the LTR .m functions is the controller matrices/gains, if they can be calculated directly/exactly (i.e. regular $\mathcal{H}_\infty$/LTR, LQG/LTR etc.), else the transformed system matrices to be applied for the ADDP design.

Transfer functions calculation.

In this group we have functions for calculating different transfer functions. Further, also the maximal and minimal singular values at specified frequencies are calculated. Special functions are derived for calculation of transfer functions related to the LTR design concept as e.g. the recovery matrix, the recovery error, the target loop, the full-loop transfer function etc.

5. DISCUSSION.

The presented CACSD package for $\mathcal{H}_2$, $\mathcal{H}_\infty$ and LTR design is based on the singular $\mathcal{H}_2$/$\mathcal{H}_\infty$ approach by Stoorvogel [16,17]. In [12] a numerical study based on the toolbox has been presented.

The reason for using the singular approach is twofold. First, it is natural to select the singular approach, since the LTR design problems normally give rise to singular problems. By using the singular approach, we avoid to perturb the system to make it regular. Secondly, from a numerical point of view, we are normally satisfied if we can find a reasonable solution by using the perturbation technique. It can been shown that the singular problem (state feedback) can be solved by using the cheap control principle from a theoretical point of view. Further, it has also been shown that the solution for the cheap control problem will converget to the solution of the equivalent Quadratic Matrix Inequality. In [12] it has been shown, however, that the numerical problems will spoil this convergence.

The main conclusion of our work, both in theoretical as well as in practical (numerical) research, [12], is that the singular approach must always be applied when the design problem is singular. The cheap control principle should be applied only when the equivalent singular theory/numerical algorithms are not available.

Current work is being carried out on a discrete time version of the toolbox.

REFERENCES.

[1] G.J. Balas et. al. 1991: "*μ-analysis and synthesis toolbox, User's Giude*", The MathWorks, Inc., South Natick, MA, USA.

[2] S.P. Boyd, V. Balakrihnan and P. Kabamba,1989:" *A bisection method for computing the $\mathcal{H}_\infty$ norm of a transfer matrix and related problems*", Mathematics of Control, Signals and Systems, MCSS, vol. 1 no. 2.

[3] N.A. Bruinsma and M. Steinbuch, 1989:"*A fast algorithm to compute the $\mathcal{H}_\infty$ norm of a transfer function*", Philips Research Laboratories Eindhoven, The Netherlands.

[4] B. Chen, A. Saberi and P. Sannuti,1993: "Loop Transfer Recovery: Analysis and design", Springer-Verlag.

[5] R.Y. Chiang and M.G. Safonov, 1990: "*Robust Control Toolbox, User's Giude*", The MathWorks, Inc., South Natick, MA, USA.

[6] J. Doyle, K. Glover, P. Khargonekar and B.A.Francis,1989: "*State-space solutions to standard $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control problems*", IEEE Transact. on Aut Control, vol AC-34, no. 8.

[7] J. Doyle and G. Stein,1981: "*Multivariable feedback design: Concepts for a classical/modern synthesis*", IEEE Transact. on Aut. Control, vol AC-26, p.4-16.

[8] D.G. Luenberger,1971:" *An introduction to observers*", IEEE Transact. on Aut. Control vol. AC-16, no. 6.

[9] J.B. Moore and T.T. Tay,1989:" *Loop recovery via $\mathcal{H}_\infty$/$\mathcal{H}_2$ sensitivity recovery*", Int. J. Control, vol 49, p. 1249-1271.

[10] H.H. Niemann and J. Stoustrup, 1991: "*A general $\mathcal{H}_\infty$/LTR design problem*", Proc. 30th IEEE Conference on Decision and Control, Brighton, England.

[11] H.H. Niemann and J. Stoustrup, 1993: "*The $\mathcal{H}_\infty$ and LTR Matlab toolbox, User's guide*", Mathematical Institute, Technical University of Denmark.

[12] H.H. Niemann and J. Stoustrup, 1993: "*Regular vs. singular methods in cheap $\mathcal{H}_\infty$ control: A numerical study*", to be presented at ECC-93, The Netherlands.

[13] H.H. Niemann, P. Søgaard-Andersen, J. Stoustrup,1991:" *Loop Transfer Recovery for General Observer Architectures*", Int. J. Control, vol.53, no. 5.

[14] E. Schijfs, 1990:"*A numerical analysis of the singular $\mathcal{H}_\infty$ control problem*", Masters Thesis, Eindhoven University of Technology, Dept. Math. and Comp. Sci., The Netherlands.

[15] G. Stein and M. Athans,1987:" *The LQG/LTR procedure for multivariable feedback control design*", IEEE Transact. Aut. Control, vol AC-32.

[16] A.A. Stoorvogel,1992:" *The $\mathcal{H}_\infty$ control problem: A state space approach*", Prentice Hall, London.

[17] A.A. Stoorvogel,1992:" *The singular $\mathcal{H}_2$ control problem*", AUTOMATICA, vol 28, no. 3, pp. 627-632.

[18] J. Stoustrup and H.H. Niemann,1993:" *State space solutions to the $\mathcal{H}_\infty$/LTR design problem*", to appear in Int. J. Robust and Non-linear Control.

[19] P. Søgaard-Andersen,1989:" *Loop transfer recovery - An eigenstructure interpretation*", Control - Theory and Adv. Technology, C-TAT, vol 5, no. 3.

[20] J.C. Willems,1981:" *Almost invariant subspaces: An approach to high gain feedback design -part 1: Almost controlled invariant subspaces*", IEEE Transact. on Aut. Control, vol AC-26, p. 235-252.

[21] W.M. Wonham,1985: "*Linear multivariable control: A geometric approach*", Springer-Verlag, New York, third edition.

[22] "*MATLAB User's Guide*" The MathWorks, Inc., South Natick, MA, USA.

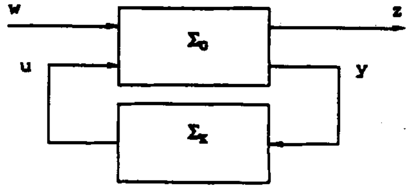

Fig. 2.1. The standard four block problem.

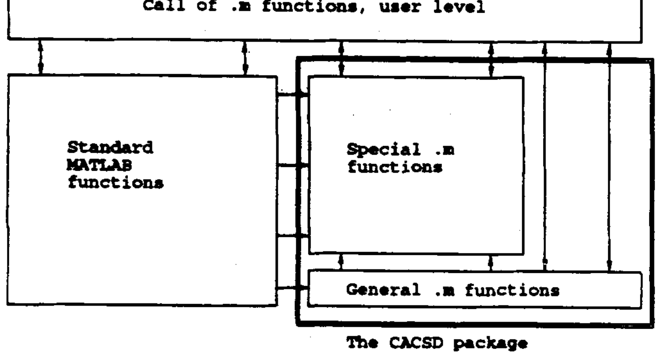

Fig. 4.1. The structure of the CACSD package.