ITERATIVE MODEL AND CONTROLLER REDUCTION USING CLOSED-LOOP BALANCING, WITH APPLICATION TO A COMPACT DISC MECHANISM

PEPIJN M. R. WORTELBOER1*, MAARTEN STEINBUCH1 AND OKKO H. BOSGRA2

1Philips Electronics N.V., P.O. Box 218, 5600 MD, Eindhoven, The Netherlands
2Mechanical Engineering Systems and Control Group, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands

SUMMARY

The standard configuration for model-based control design is reformulated into a twin feedback configuration with the model and controller written in separate feedback loops. Closed-loop balanced reduction is developed in this framework. A scheme with alternating model reduction, optimal control synthesis and controller reduction steps is designed to find high-performance low-order controllers. CD-player tracking controllers with orders below 10 have been found starting from a 120th-order model. Copyright © 1999 John Wiley & Sons, Ltd.

Key words: linear dynamic systems; optimal control design; order reduction

1. INTRODUCTION

This paper proposes a tractable iterative procedure to incorporate order reduction of both the model and controller in the control design.

Consider the optimization-based control design problem of Figure 1, where a specific norm is minimized with respect to $K$, the controller that closes the lower loop around the so-called standard plant $N$. This general representation will be denoted by the standard controller synthesis configuration (SCSC).

The standard plant $N$ comprises the model ($G$) of the system to be controlled and performance weights that are used to ensure that minimization of the influence of $w$ on $z$ leads to the desired controlled behaviour. Often, the order of the model is high and a reduction is needed before any controller can be calculated. Here we assume that the original model has high accuracy and is the best we can get; the only problem is its order. The weights that are involved, however, are to be shaped depending on the performance measures; they are only auxiliary in arriving at a satisfactory closed-loop behaviour. This strongly suggests that the dynamics of the real system that are given by nature and the weights that are to be tuned in the control design phase, should be isolated from each other. Hence, we introduce the twin-feedback configuration.1

In the twin-feedback configuration the interconnecting system matrix $M$ exclusively contains the performance and robustness weights involved in the controller synthesis problem. We will call

*Correspondence to: P. M. R. Wortelboer, Philips Electronics N.V., P.O. Box 218, 5600 MD, Eindhoven, The Netherlands.

CCC 1049-8923/99/030123–20$17.50
Copyright © 1999 John Wiley & Sons, Ltd.
it the master weight. The controller synthesis configuration can be thought of as the twin-feedback configuration with freedom to choose the controller in the lower feedback loop. The performance is related to the norm of the twin-feedback configuration: the lower this closed-loop system norm, the better the performance.

1.1. Order reduction

Controller reduction has been an active area for research, as the use of model-based control design techniques applied to high-order system models leads to controllers having also a high order. We refer to the literature and the references therein for more information on existing model order reduction methods.

Various controller reduction approaches have been proposed. It seems to have been accepted by several authors that controller reduction is best performed by incorporating frequency weights. However, it must be realized that most of these methods do not provide a guaranteed bound on the performance degradation, nor provide insight into an appropriate selection of the reduced order. Notable exceptions aim at performance preservation. In practical applications, performance weights are instruments to find a sufficiently low-order controller which achieves a good intuitive measure of performance. The straight-forwardness of closed-loop balanced reduction allows a quick search for the ‘best’ performance over different reduced orders. It has been shown that additional frequency weights to penalize performance degradation in critical frequency ranges due to closed-loop balanced reduction, can be very effective in practice. This approach has much in common with a control design approach: the weights are adjusted in an iterative fashion, after each adjustment the obtained performance of the reduced controller is inspected, and the weight adjustment and/or order selection is continued until the best compromise between order and reduced-order performance is obtained.

This paper incorporates the closed-loop balanced reduction technique together with model-based optimal control synthesis techniques, into a practical procedure for finding controllers with satisfactory performance and limited complexity. This procedure will be explained concisely and illustrated by means of a practical CD-player control problem. The computation of $H_2$-norm optimal fixed-order controllers is included as a final step in the procedure. Special attention is paid to the relation between closed-loop balanced reduction, plain balanced and frequency-weighted balanced reduction.

The organization of the paper is as follows. After the introduction of the notation for basic operations (Section 2), the balanced order reduction in the twin-feedback configuration is introduced in Section 3. Section 4 gives a detailed description of the application of the iterative model/controller reduction procedure to a compact disc mechanism. The main results are summarized in the form of conclusions in Section 5.
2. PRELIMINARIES

2.1. Truncation and projection of realizations

Continuous-time finite-dimensional time-invariant linear systems can be written in state space as

\[ G = (A, B, C, D), \quad \dot{x} = Ax + Bu \]
\[ y = Cx + Du \]

For clarity we will write \( G_n \) for a realization of \( n \)th order.

Truncation of a system realization underlies both balanced reduction and modal reduction. Let matrix subscripts between parantheses indicate index vectors. With

\[ \Gamma_r = \begin{bmatrix} I_r & 0 \\ 0 & 0 \end{bmatrix} = I_{n(1:r)} \]

the truncation of realization \( G_n \) to order \( r \) can be written as

\[ G_r = \mathcal{B}_{[1,r]} (G_n) \triangleq (\Gamma_r^T A \Gamma_r, \Gamma_r^T B, C \Gamma_r, D) \]

\[ = (A_{[1:r,1:n]}, B_{[1:r]}, C_{(1,1:r)}, D) \triangleq \mathcal{B}_r (G_n) \]

In the same format a state transformation can be written as \( \tilde{G}_n = \mathcal{B}_{[1,T]} (G_n) \), and a projection of dynamics that is governed by the projection pair \([L_r, R_r] \) obeying \( L_r^* R_r = I_r \) can be written as \( \tilde{G}_r = \mathcal{B}_{[L_r,R_r]} (G_n) \).

2.2. Balancing

The theory of balancing is now well established: see for instance References 4, 24 and 25. The balancing idea hinges explicitly on the state co-ordinates. The system dynamics are analysed in two parts. The controllability part measures the influence of input \( u \) on the state co-ordinates assuming \( x_0 = 0 \), and the observability part measures the influence of \( x_0 \) on the output \( y \) assuming \( u = 0 \).

The controllability Gramian \( P_n \) and observability Gramian \( Q_n \) can be solved uniquely for stable systems from the following Lyapunov equations:

\[ A P_n + P_n A^* + B B^* = O \quad (1a) \]
\[ A^* Q_n + Q_n A + C^* C = O \quad (1b) \]

A balancing transformation \( \mathcal{L}_n \) (with \( \mathcal{L}_n \triangleq \mathcal{L}^{-*} \)) exists, such that

\[ \mathcal{L}_n^* P_n \mathcal{L}_n = \mathcal{L}_n^* Q_n \mathcal{L}_n = \text{diag} (\sigma_n) \]

with

\[ \sigma_n = \sqrt{\lambda(P_n Q_n)} \]

the so-called Hankel singular value (HSV) vector. The HSVs are system invariants (realization independent). This transformation is exclusively based on \( P_n \) and \( Q_n \),

\[ [\mathcal{L}_n, \mathcal{R}_n] = \tilde{\mathcal{F}} (P_n, Q_n) \]
By definition, $\tilde{G}_n = \mathcal{R}_{[L_n, \hat{R}_n]}(G_n)$ is a balanced realization, and
$$\tilde{G}_r = \mathcal{R}_r(\tilde{G}_n) = \mathcal{R}_{[L_r, \hat{R}_r]}(G_n) \triangleq \text{bal } \mathcal{R}_r(G_n)$$
defines balanced reduction, with $\hat{R}_r = \hat{R}_n \Gamma_r, \hat{L}_r = L_n \Gamma_r$ satisfying $\hat{L}_r^* \hat{R}_r = I_r$.

For frequency-weighted balanced reduction, we start from frequency-weighted Gramians and perform the balancing and reduction in the same way as for plain balanced reduction. A formal definition is given in the appendix.

### 2.3. Performance configurations and optimal control

We use linear fractional transformations and refer to Figure 2 for notation.

First consider the open-loop case, i.e. $K = [\ ]$, then we can use the upper feedback loop to define the upper linear fractional transformation
$$N = \mathcal{F}_u(M, G) \triangleq \begin{bmatrix} M_{yw} & M_{yu} \\ M_{yw} & M_{yy} \end{bmatrix} + \begin{bmatrix} M_{rs} \\ M_{sr} \end{bmatrix} G(I - M_{ws} G)^{-1} \begin{bmatrix} M_{uw} & M_{un} \end{bmatrix}$$

Next, we use this $N$ in conjunction with $K$ in the lower feedback loop to define the lower linear fractional transformation
$$F = \mathcal{F}_l(N, K) \triangleq N_{zw} + N_{zw} K (I - N_{yw} K)^{-1} N_{yw}$$

From linear fractional transformation theory we know that
$$\mathcal{F}(G, M, K) = \mathcal{F}_l(\mathcal{F}_u(M, G), K) = \mathcal{F}_u(\mathcal{F}_l(M, K), G)$$

The theory of optimal control in the $H_2$ and $H_{\infty}$ case is rather complete now. We refer to Reference 4 for all details. The state-space approaches are coded, for instance, in the $\mu$-tools for use with MATLAB. Our approach to low-order control design uses an order reduction procedure around standard optimal full-order control synthesis. In this paper we will use for the CD player example an $H_2$ design, and we denote in the sequel an $H_2$ (full-order) control synthesis operation as $H_2 \mathcal{F}(G, M, -)$.

### 3. ORDER REDUCTION IN A CLOSED-LOOP SETTING

First, we consider the objectives of closed-loop order reduction: in Section 3.1 the model reduction step is analysed, and in Section 3.2 we take a closer look at the controller reduction

![Figure 2. Twin-feedback configuration and its symbolic representation (underlined symbols for lower feedback loop and over-lined symbols for upper feedback loop)](image-url)

step. The algorithm that is used for both model and controller reduction, closed-loop balanced reduction, is explained in Section 3.3. Finally, Section 3.4 gives a procedure for using the new order reduction facility in connection with optimal controller synthesis.

3.1. Model reduction in closed loop

As mentioned earlier, model reduction is only an auxiliary step in low-order control design. It is needed to enable an optimal controller synthesis step. The main issue is that there is not a clear measure, like the performance measure, to quantify the loss of information by model reduction. Minimizing some open-loop error \( \| G - G_r \| \) is not appropriate since open-loop irrelevant dynamics may appear to be crucial in closed loop and vice versa (spill-over problem). A better approach is to minimize the closed-loop changes due to model reduction. This, however, requires a controller. Although we do not have the controller we are looking for yet, we often have a preliminary stabilizing controller. Note that most servo-systems are designed for feedback operation with fairly simple feedback controllers. Our problem then is not mere closed-loop stability, but performance improvement (often with a limitation on the controller complexity). We state that model reduction for control design should make a trade-off between the model order and the change in closed-loop behaviour (both preferably low). The change in closed-loop behaviour can be measured by

\[
c_m \triangleq \| J(G_h, M, K) - J(G_m, M, K) \| \tag{4}
\]

with \( G_h \) and \( G_m \) the high-order and moderate-order model, respectively. For ease of interpretation, we introduce a relative error with

\[
\gamma_h \triangleq \| J(G_h, M, K) \| \tag{5}
\]

We define

\[
\delta_m \triangleq c_m/\gamma_h = \frac{\| J(G_h, M, K) - J(G_m, M, K) \|}{\| J(G_h, M, K) \|} \tag{6}
\]

Note that \( \gamma_m = \| J(G_m, M, K) \| \) itself is not a good measure, since minimization of \( \gamma_m \) yields the ‘best-controllable’ reduced-order system; a solution might even be a zeroth-order model \( G_0 = O \).

3.2. Controller reduction in closed loop

Controller reduction in closed loop has the same objective as the original control design problem. Given \( M \) and \( G \), find a \( K_r \) that minimizes \( \| J(G, M, K_r) \| \). The only difference is that we have a high-order controller available. The assessment of the reduction result is much easier than in the model reduction case since we can use the performance criterion directly

\[
\gamma_r = \| J(G, M, K_r) \| \tag{7}
\]

Note that it is also possible to strive for reduced-order controllers that change the closed-loop minimally as in the model reduction case, but this may result in controllers that have worse performance.
3.3. Closed-loop balanced reduction

First, we derive the algorithm and then we state some of its properties.

3.3.1. The algorithm. To define balanced reduction within the twin-feedback configuration, we make a realization $F = \mathcal{I}(G_h, M, K_n)$ in which the state vector is built from the state vectors of $G_h$ (length $h$), $M$, and $K_n$ (length $n$) in that precise order: $x_F^T = [x_h^T x_M^T x_n^T]$. Balanced reduction of $G$ within $F$ follows the standard balanced reduction procedure with the difference that instead of taking the entire Gramians of $F$, denoted by $P_F = \mathcal{P}(F)$ and $Q_F = \mathcal{Q}(F)$, only specific parts of these Gramians are used. The scheme for $G_h$ reduction within $I(G_h, M, K)$ hinges on taking the left-upper parts of the Gramians of the interconnected system realization and proceeds along classical lines from then on (notation: the reduction procedure is applied to the underlined system):

$$
P_G = \mathcal{P}(\mathcal{I}(G_h, M, K_n))_{(1:h,1:h)} \triangleq \mathcal{P}(G_{\underline{h}}, M, K_n)
$$

$$
Q_G = \mathcal{Q}(\mathcal{I}(G_h, M, K_n))_{(1:h,1:h)} \triangleq \mathcal{Q}(G_{\underline{h}}, M, K_n)
$$

$$
[\tilde{L}_h, \tilde{R}_h] = \mathcal{J}(P_G, Q_G)
$$

$$
\tilde{L}_m = [\tilde{L}_h]_{(1:m)}; \tilde{R}_m = [\tilde{R}_h]_{(1:m)}
$$

$$
\tilde{G}_m = \mathcal{R}_{[\tilde{L}_r, \tilde{R}_r]}(G_{\underline{h}}) \triangleq \mathcal{R}_m(\mathcal{I}(G_h, M, K_n))
$$

Explaining this procedure in words, but now for closed-loop controller reduction:

- take the controller state part of the closed-loop Gramians, $P_K = \mathcal{P}(\mathcal{I}(G_h, M, K_n))$ and $Q_K = \mathcal{Q}(\mathcal{I}(G_h, M, K_n))$,
- extract a balancing transformation, $[\tilde{L}_n, \tilde{R}_n] = \mathcal{J}(P_K, Q_K)$,
- truncate the last $n - r$ columns of $\tilde{L}_n$ and $\tilde{R}_n$ yielding $\tilde{L}_r$ and $\tilde{R}_r$ and
- apply a projection of $K_n$ by means of $\tilde{L}_r$ and $\tilde{R}_r$.

This entire procedure is summarized and denoted as

$$
\tilde{R}_r = \mathcal{R}_r(\mathcal{I}(G_h, M, K_n))
$$

Since the closed-loop balancing transformations for reducing $G$ and $K$ are computed independently, we can also balance $G$ and $K$ simultaneously. In Figure 3 the block-diagonal structure of the similarity transformation group that is allowed in closed-loop transformation is visualized, $\tilde{G}$ and $\tilde{K}$ are closed-loop balanced if they induce diagonal blocks in the Gramians of $\mathcal{I}(\tilde{G}, M, \tilde{K})$:

$$
\mathcal{P}((\tilde{G}, M, \tilde{K})) = \begin{bmatrix}
\Sigma_G & & \\
& & \\
& & \Sigma_K
\end{bmatrix}
$$

$$
\mathcal{Q}((\tilde{G}, M, \tilde{K})) = \begin{bmatrix}
& & \\
& & \\
& & \Sigma_K
\end{bmatrix}
$$
Figure 3. Diagonal structure of similarity transformation group

with $\Sigma_G = \text{diag}(\sqrt{\lambda(P_GQ_G)})$ the $\mathcal{H}(G, M, K)$-HSVs and $\Sigma_K = \text{diag}(\sqrt{\lambda(P_KQ_K)})$ the $\mathcal{H}(G, M, K)$-HSVs. Note that $\mathcal{H}(\hat{G}, M, \hat{K}) \neq \mathcal{H}(\hat{G}, M, \hat{K})$ generically.

3.3.2. Properties. Plain balanced reduction has three nice properties. First, stability is always preserved, second, the reduction error measured in $H_\infty$ norm is bounded by twice the sum of the discarded HSVs, and finally, the truncation of a balanced realization is still a balanced realization.

These properties do not hold for closed-loop balanced reduction. The reduction results obtained for our CD-player example prove that models and controllers resulting from closed-loop balanced reduction may get unstable themselves and also may lead to an unstable closed-loop system.

A property that each reduction method should have is that hidden modes are identified. In the sequel we investigate what happens if HSVs go to zero.

If $K_n$ has $n - r$ uncontrollable state co-ordinates then $\mathcal{H}(G, M, K_n)$ has at least $n - r$ uncontrollable state co-ordinates and $\mathcal{H}(\hat{G}, M, \hat{K}_n)$ has at most rank $r$. The same holds for $G$. Of course, the observability case is completely dual. In case $G$, $M$, and $K$ are all minimal realizations, there may still be a chance that $\mathcal{H}(G, M, K)$ is not minimal. This situation is often referred to as ‘cancellation’. It is widely recognized nowadays that by using a well-defined SCSC, the danger of cancellations can be minimized.

Finally, if the SCSC is asymptotically stable and minimal, the Gramians of the SCSC are both positive definite. Then, a small $\|K\|$ can only occur if the coupling of the last ($n$th) closed-loop balanced controller state co-ordinate with the other co-ordinates is sufficiently weak. And this means that truncation of the closed-loop balanced controller to order $n - 1$ in the SCSC gives almost the same result as truncation of the SCSC by plain balanced reduction.

The relation between closed-loop balanced reduction and frequency-weighted balanced reduction is worked out in the appendix, see also Reference 20. We conclude that for controller reduction

$$\text{bal } \mathcal{H}(G, M, K) = \text{bal } \mathcal{H}(W_{\delta K}KV_{\delta K})$$

with the weights as in (22), see the appendix. Note, however, that the computational scheme for closed-loop balanced reduction is more efficient and that unstable $G$ and $K$ can be reduced as long as the twin feedback configuration is strictly stable.

3.4. A combined order reduction—control design strategy

Here we describe the main cycle in obtaining high-performance low-order control starting from a high-order model (Figure 4). The model reduction step can be repeated a number of times to
Given: $K_s$ and $G_h$ \quad \text{s} \ll \text{h}
Start: $K_n = K_s$

**Model Reduction**

$G_m$?

\[
g_m < h \quad \frac{\|\mathcal{I}(G_h, M, K_s) - \mathcal{I}(G_m, M, K_n)\|}{\|\mathcal{I}(G_h, M, K_n)\|} < \epsilon
\]

**Controller Synthesis**

$K_n$?

\[
\min_{K_n} \|\mathcal{I}(G_m, M, K_n)\|
\]

**Controller Reduction**

$K_r$?

\[
\|\mathcal{I}(G_h, M, K_r)\| < \gamma
\]

Figure 4. The iteration scheme for low-order control design starting from a high-order model using interactive order reduction.

find an appropriate $m$ together with $G_m$. The evaluation of the order reduction effects is by means of (6)

\[
\delta_m \triangleq \frac{\|\mathcal{J}(G_h, M, K_s) - \mathcal{J}(G_m, M, K_n)\|}{\|\mathcal{J}(G_h, M, K_n)\|}
\]

The control design step can be performed by $H_2$ or $H_{\infty}$ optimal control if $m$ is sufficiently small. With a new $K_n$ we can go back to the model reduction phase and verify if $G_m$ is still appropriate. If this iteration has converged we can proceed with the controller reduction iteration. After each step the performance is measured by means of $\mathcal{I}(G_h, M, K_r)$. For the $H_2$ case we end with a search for the optimal fixed-order controller. For the $H_{\infty}$ case, we do not have a satisfactory numerical algorithm to derive an optimal fixed-order $H_{\infty}$ controller starting from the optimal full-order controller. Yet, one idea is to find better performing low-order controllers by closed-loop balanced reduction of each controller that is generated during the bisection procedure of a $H_{\infty}$ control algorithm. The idea behind this is to relax the performance requirement deliberately, thus hoping to find a full-order controller that can be reduced more easily.

It is important to realize that balanced reduction is heuristic in the end and requires a thorough embedding in an evaluation and manipulation environment.

4. APPLICATION TO A COMPACT DISC MECHANISM

4.1. Tracking control problem

The control task of a CD mechanism is to achieve track following, which basically amounts to pointing the laser spot to the track of pits on the CD that is rotating. The mechanism treated here, consists of a swing arm on which a lens is mounted, see Figure 5.
The rotation of the arm in the horizontal plane enables reading of the spiral-shaped disc track, and the suspended lens is used to focus the spot on the disc. Due to the fact that the disc is not perfectly flat, and due to irregularities in the spiral of pits on the disc, a feedback system is needed. The challenge is to find a low-cost controller that can make the servo-system faster and less sensitive to external shocks. Performance improvements are sought via model-based control design in the frequency domain. Robustness is an important issue in practical control design, in fact, design for robustness against specific variations in the CD-system has already been achieved. Here, we concentrate on low-order control design for a single high-order model.

A detailed model is needed to describe the vibrational behaviour of the electro-mechanical system over a large frequency range in order to anticipate the interaction with a controller of possible high bandwidth. A finite element model was built, containing 60 vibration modes ($n = 120$), and has two inputs (actuation of arm and of focus lens), and two outputs (tracking error and focus error). The model can be obtained by FTP. With respect to the disc, we can discern a radial part and a focus part of the feedback loop. Ideally, these parts have no interaction, but in practice there is (some) mutual interference. From a control design viewpoint, the radial loop poses a much more difficult problem compared to the focus loop. This is due to the more pronounced mechanical resonances. In this paper, we concentrate our investigation on this loop only.

In Figure 6 a block-diagram of the radial control loop is shown. The difference between the track position $w_{\text{track}}$ and the spot position is detected by the optical system (summing junction in Figure 6); it generates a radial error $e$. A controller $K(s)$ feeds the system with the current $i$, generating a force resulting in an acceleration of the actuator. The disturbance $w_{\text{dist}}$ accounts for mechanical shocks and vibrations. The transfer function from the current $i$ to the radial displacement of the mechanism is called $G(s)$. Only the control-error signal $e$ and controller output $i$ are available for measurement. In Figure 13 the (magnitude) frequency response of a finite element based 120th-order model is plotted. At low frequencies the actuator transfer function from current input $i$ to position error output $e$ is a critically stable system with a phase lag of $180^\circ$ (rigid body mode). At higher frequencies the plot shows parasitic dynamics.

We will show the effectiveness of our closed-loop reduction strategy in finding a (model-based) $K_r$ that gives $\mathcal{J}(G_{120}, M, K_r)$ a favourable disturbance attenuation behaviour. As an example we will show results for optimal $H_2$ control. In Reference 28 also the results for $H_{\infty}$ control are shown. The performance objective is defined in terms of a four-block transfer function matrix,
Figure 6. Configuration of the radial control loop

being the transfer from disturbances on setpoint $w_{\text{track}}$ and plant input $w_{\text{dist}}$ (shocks and vibration), with weights $W_r$ and $W_d$, respectively, to (radial) error $e$ and controller output $i$ as signals to be controlled, with weights $W_s$ and $W_t$, respectively.

In the sequel we will use the twin-feedback configuration:

$$ F = f(G, M, K) $$

with the master weight

$$ M = \begin{bmatrix} O & O & W_d & I_f \\ W_s & W_s W_r & O & O \\ O & O & O & W_t \\ I_u & W_r & O & O \end{bmatrix} $$

Note that $M$ can also be written in a matrix product form with each weight only occurring once.\(^{28}\)

4.2. Iterative model and controller reduction

The procedure for achieving high-performance low-order controllers can be summarized as follows: we start with the 120th-order model $G_{120}$, a master weight $M$, including the weights $W_s, W_r, W_d$ and a stabilizing (PID) feedback $K_3$. Figure 7 shows the magnitudes of the frequency responses of the weights

$$ W_s(s) = \frac{3000\pi}{s + 2\pi} $$

$$ W_t(s) = 625 \frac{s^2 + 1.2(4000*2\pi)s + (4000*2\pi)^2}{s^2 + 2(1e5*2\pi)s + (1e5*2\pi)^2} $$

where $W_s$ is used to force low-frequent disturbance attenuation and $W_t$ to obtain robustness against high-frequent parasitics. Both $W_s$ and $W_d$ are chosen equal to 1.

The stabilizing controller $K_3$ is a PID controller with first-order low-pass at high frequencies:

$$ K_3(s) = -7.6746 \times 10^4 \frac{(s + 3.1447*10^2)(s + 8.6207*10^2)}{s(s + 6.6225*10^3)(s + 6.2832*10^4)} $$

Applying closed-loop balanced reduction we can obtain a low-order model, for which an $H_2$ optimal controller can be calculated, which in turn can be reduced in closed loop. With this low-order controller the procedure is restarted. As a final step the low-order controller is used as
The six steps of model reduction (1,4), controller design (2,5), and controller reduction (3,6):

1. balR_{20}(\mathcal{I}(G_{120}, M_3, K_3)) \quad aa \quad (CPU: 7.47 s)

2. H_2K_{23}(\mathcal{I}(G_{20}, M_3, \_\_\_)) \quad b \quad (CPU: 0.82 s)

3. balR_{14}(\mathcal{I}(G_{20}, M_3, K_{28})) \quad cc \quad (CPU: 0.59 s)

4. balR_{14}(\mathcal{I}(G_{20}, M_3, K_{11})) \quad dd \quad (CPU: 0.61 s)

5. H_2K_{17}(\mathcal{I}(G_{14}, M_3, \_\_\_)) \quad e \quad (CPU: 0.92 s)

6. balR_{9}(\mathcal{I}(G_{14}, M_3, K_{12})) \quad ff \quad (CPU: 0.47 s)

starting value in a non-linear optimization routine to achieve better performance and possibly obtain the global minimum. In Figure 8 the procedure is shown in the form of a flow chart. The key routines for closed-loop balanced reduction and $H_2$ optimal fixed order calculation are included in the WOR-toolbox \cite{22} (weighted order reduction). This toolbox makes use of the $\mu$-toolbox \cite{26} and parts of it are used in the QFT-toolbox \cite{29}.

In the following section we will describe all the subsequent steps as mentioned in Figure 8 (computing times in the figure from MATLAB4.2 on an HP700).

**Step 1. Model reduction:** $G_{20} = \text{bal} R_{20}(\mathcal{F}(G_{120}, M_3, K_3))$. The criterion in the model reduction step is the relative change of the closed-loop transfer function matrix measured in $H_2$-norm

$$\delta(r) = \frac{\|\mathcal{F}(G_{120}, M_3, K_3) - \mathcal{F}(G_r, M_3, K_3)\|_2}{\|\mathcal{F}(G_{120}, M_3, K_3)\|_2} \lesssim 1$$

Figure 9 shows this relative error $\delta(r)$ as a function of the order $r$ of the model. The dashed bars indicate results where the reduced-order model is not stabilized by the controller.
A good approximation is possible with \( r = 20 \). Of course, it is worthwhile to investigate even lower-order approximations. However, in this stage we should be very careful about deleting any dynamics in the model which are not excited by the simple controller \( K_3 \), but which might become important in the final stage. So here we will choose a model with a relatively high order, and in a second iteration we will try to further reduce the model, using \( H_2 \) optimal controllers in the calculation of the closed-loop error transfer function.

The new model \( G_{120} \) enables the calculation of an \( H_2 \) optimal controller.

**Step 2. Controller calculation:**

\[
K_{23} = H_2 \mathcal{K}_{23} (\mathcal{J}(G_{20}, M_3, K))
\]

\[
(23) \quad \gamma(23) = \| \mathcal{J}(G_{20}, M_3, K_{23}) \|_2 = 913.76
\]

**Step 3. Controller reduction:**

\[
K_{11} = \text{bal} \mathcal{R}_{11} (\mathcal{J}(G_{20}, M_3, K_{23}))
\]

In the controller reduction step the criterion for choosing an appropriate \( r \) is the \( H_2 \)-norm of the closed-loop system \( F = \mathcal{J}(G_{20}, M_3, K_r) \):

\[
\gamma(r) = \| \mathcal{J}(G_{20}, M_3, K_r) \|_2
\]

In Figure 10 the \( H_2 \)-norm of the closed-loop system is shown, as a function of controller order. Again, the dashed bars indicate unstable (closed-loop) results.

The figure clearly shows that the reduced-order controllers for \( r \geq 11 \) are almost as good as the original \( K_{23} \). The ninth-order controller would be appropriate and we could stop here. To see how the iteration proceeds we take the 11th-order to do a further model reduction step.

**Step 4. Model reduction:**

\[
G_{14} = \text{bal} \mathcal{R}_{14} (\mathcal{J}(G_{20}, M_3, K_{11}))
\]

Again in this model reduction step we use as criterion the \( H_2 \)-norm of the relative error, but now with the 20th-order model as starting point (and not \( G_{120} \) because of the computational burden involved), and the controller \( K_{11} \) from the previous step

\[
\delta(r) = \frac{\| \mathcal{J}(G_{20}, M_3, K_{11}) - \mathcal{J}(G_r, M_3, K_{11}) \|_2}{\| \mathcal{J}(G_{20}, M_3, K_{11}) \|_2} \ll 1
\]

Figure 11 shows this relative error \( \delta(r) \) as a function of the order \( r \) of the model.
A very small approximation error is made with a 14th-order model. Using $G_{14}$ we can again calculate an $H_2$ optimal controller. Note that $G_{14}$ is unstable and that (a) $G_{14}$ could also have been selected directly based on Figure 9.

**Step 5. Controller calculation:**

$$ K_{17} = H_2\mathcal{H}_{17}(\mathcal{J}(G_{14}, M_3, \ldots)). $$  

(\text{e})

$$ \gamma(17) = \| \mathcal{J}(G_{14}, M_3, K_{17}) \|_2 = 913.77 $$

**Step 6. Controller reduction:**

$$ K_9 = \text{bal} \mathcal{R}_9(\mathcal{J}(G_{14}, M_3, K_{17})). $$  

(\text{ff})

In Figure 12 the $H_2$-norm $\gamma(r) = \| \mathcal{J}(G_{14}, M_3, K_r) \|_2$ of the closed-loop system is shown, as a function of controller order.

From the figure the choice for $r = 9$ is evident. Notice that a similar norm would have been obtained if in step 3 (ee) a ninth-order controller would have been taken. Hence, steps 5 and 6 could have been skipped in this case. Since the final reduced-order controller will be further optimized using fixed-order $H_2$ optimization it is only relevant to have a reasonable starting value for the controller parameters. Nevertheless, the 11th order controller from step 3 still was relevant for the model reduction in step 4, leading to a low-order approximation of the plant $G$, enabling a faster calculation for the final optimization. Besides the fact that two different roads lead to approximately the same result is a strong indication for the robustness of the proposed scheme.
4.3. Fixed-order $H_2$ optimization

Using the reduced-order controllers as initial values, we are now able to start the fixed-order $H_2$-optimization. The results are summarized in Tables I and II below.

Table I shows the results obtained using the reduced-order model ($G_{20}$), indicating that further optimization improves results significantly for $r < 10$. This also holds for the (computational much more involved) case with the full 120th-order model, see Table II. It should be noted that...
the convergence of the algorithm is fast provided the closed-loop balanced low-order controllers are close to the minimizing reduced-order controllers. It is emphasized that these results may be only local minimizers, still the improvement over closed-loop balanced results is general.
5. CONCLUSIONS

Using modern optimal control schemes and newly developed reduction schemes it is now possible to derive controllers of limited complexity that achieve almost optimal performance. Prerequisites are the availability of an accurate model and the possibility to translate the control objectives into a minimization problem.

Straightforward extensions of frequency weighted balanced reduction are very suitable for application of order reduction within controlled systems. Combined model and controller reduction in a twin-feedback configuration is ideally suited for model-based fixed-order control design based on high-order models since the interconnection structure incorporates the same weighting functions as the standard control design configuration. The key algorithm for closed-loop balanced reduction is almost as fast as standard balanced reduction. The iterative process for fixed-order control design starting from high-order models can be executed in an interactive way using a new toolbox for use with MATLAB.

APPENDIX A. COMPARISON OF BALANCING CONCEPTS USING TRANSFER FUNCTIONS

To enable a transparent comparison of the balancing concepts presented in this paper, we formulate the different Gramians in terms of transfer functions; the controllability Gramian is defined in terms of the transfer function from external inputs to the state of concern, the observability Gramian is expressed in terms of the transfer function from the initial state to all external outputs.

Let \( y(s) = G(s)u(s) \) with \( G(s) \) strictly proper and let \( G = (A, B, C, O) \) be a realization of \( G(s) \) hinging on state vector \( x \). Next, we introduce a transfer function formulation of the state-space dynamics.

The Laplace transform of the state \( x \) for non-zero initial condition is

\[
x(s) = (sI - A)^{-1}Bu(s) + (sI - A)^{-1}x_0(s)
\]

Thus the transfer function for \( x_0 \) from \( u \) to \( x \) is \( (sI - A)^{-1} \). The anti-causal transfer function for \( u = 0 \) from \( x \) to \( x_0 \) is \( (sI - A)^{-1} \). The transfer function for \( u = 0 \) from \( x_0 \) to \( y \) is \( C(sI - A)^{-1} \).

We start the derivation with plain balancing, next we derive the frequency weighted balancing case, and then we introduce the closed-loop balancing variant. We conclude with the comparison.

A.1. Plain balancing

First verify that

\[
[T_{xa}(s)] = C(sI - A)^{-1}B = [C(sI - A)^{-1}][sI - A][(sI - A)^{-1}B] = [T_{ya}(s)][T_{xa}(s)][T_{wa}(s)]
\]

In this decomposition we can identify the transfer functions \( [T_{ya}(s)] \) and \( [T_{wa}(s)] \) that define the controllability and observability Gramian:

\[
P_n = \mathcal{P}(G_n) = \frac{1}{2\pi} \int_{-\infty}^{\infty} [T_{wa}(j\omega)][\cdot]^* \, d\omega \tag{11a}
\]

\[
Q_n = \mathcal{Q}(G_n) = \frac{1}{2\pi} \int_{-\infty}^{\infty} [T_{ya}(j\omega)][\cdot]^* \, d\omega \tag{11b}
\]

These Gramians can be solved uniquely for stable systems from the Lyapunov equations (1).
A.2. Frequency-weighted balancing

Let \( y(s) = W(s)G(s)V(s)u(s) \) with \( V \) and \( W \) an input weight and an output weight, respectively. Further, let the realization for \( GV \) have state-space matrices
\[
A_{GV} = \begin{bmatrix} A & BC_V \\ O & A_V \end{bmatrix}, \quad B_{GV} = \begin{bmatrix} O \\ B_V \end{bmatrix}
\]
and let the realization for \( WG \) have state-space matrices
\[
A_{WG} = \begin{bmatrix} A_W & B_wC \\ O & A \end{bmatrix}, \quad C_{WG} = [C_w \ 0]
\]
In frequency-weighted balanced reduction, we consider the input to state transfer function (assuming \( x_0 = x_{0V} = 0 \))
\[
T_{ux}(s) = (sI - A)^{-1}BV(s) = [I \ O](sI - A_{GV})^{-1}B_{GV}
\]
and the initial-state-to-output transfer function (assuming \( u = x_{0W} = x_{0V} = 0 \))
\[
T_{yx_0} = W(s)C(sI - A)^{-1} = C_{WG}(sI - A_{WG})^{-1} \begin{bmatrix} O \\ I \end{bmatrix}
\]
(12)

With \( T_{ux} = (sI - A) \), we can verify the following:
\[
T_{yx_0}T_{ux}(s)T_{ux}(s) = W(s)C(sI - A)^{-1}BV(s) = W(s)G(s)V(s) = T_{yn}(s)
\]
The frequency-weighted Gramians can be defined in exactly the same way as in the plain balancing case:
\[
P_G = \frac{1}{2\pi} \int_{-\infty}^{\infty} [T_{ux}(j\omega)]^* \cdot \cdot \cdot \ d\omega
\]
\[
= [I \ O] \frac{1}{2\pi} \int_{-\infty}^{\infty} (sI - A_{GV})^{-1}B_{GV}B_{GV}^*(sI - A_{GV})^{-*} \cdot \cdot \cdot \ d\omega \begin{bmatrix} 1 \\ 0 \end{bmatrix}
\]
\[
= [I \ O] P_{GV} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = [P_{GV}]_{upper, left}
\]
\[
Q_G = \frac{1}{2\pi} \int_{-\infty}^{\infty} [T_{yn}(j\omega)]^*[\cdot \cdot \cdot] \ d\omega
\]
\[
= [O \ I] \frac{1}{2\pi} \int_{-\infty}^{\infty} (sI - A_{WG})^{-*}C_{WG}C_{WG}(sI - A_{WG})^{-1} \cdot \cdot \cdot \ d\omega \begin{bmatrix} O \\ I \end{bmatrix}
\]
\[
= [O \ I] Q_{WG} \begin{bmatrix} O \\ I \end{bmatrix} = [Q_{WG}]_{lower, right}
\]
The above reformulation shows that \( P_G \) and \( Q_G \) can be computed by first solving \( P_{GV} \) and \( Q_{WG} \) from appropriate Lyapunov equations and next taking the relevant blocks on the diagonal.

A.3. Closed-loop balancing

Consider \( z(s) = \mathcal{J}(G(s), M(s), K(s))w(s) \) with the twin-feedback system realization \( F = \mathcal{J}(G, M, K) \) in terms of the state vector \( x_0^T = \begin{bmatrix} x_0^T M x_0^T \end{bmatrix} \), and define
\[
\Phi_u = \begin{bmatrix} I_u & O_{uw} & O_{ux} \end{bmatrix}, \quad \Phi_z = \begin{bmatrix} O_{zn} & I_z & O_{zx} \end{bmatrix}, \quad \Phi_z = \begin{bmatrix} O_{zn} & O_{zx} & I_z \end{bmatrix}
\]
\[
\Phi_y = \begin{bmatrix} I_y \\ O_{wy} \\ O_{wy} \end{bmatrix}, \quad \Phi_u = \begin{bmatrix} O_{uw} & I_u \end{bmatrix}, \quad \Phi_y = \begin{bmatrix} O_{uw} \\ O_{uw} \end{bmatrix}
\]
With
\[ T_{x,w} = \Phi_u(sl - A_F)^{-1}B_F \tag{13} \]
\[ T_{x,w} = C_F(sl - A_F)^{-1}\Phi_y \tag{14} \]
we can define the closed-loop Gramians for \( G \):
\[ P_G = \frac{1}{2\pi} \int_{-\infty}^{+\infty} [T_{x,w}(j\omega)]^* \cdot \omega = \Phi_uP_F\Phi_u^* \]
\[ Q_G = \frac{1}{2\pi} \int_{-\infty}^{+\infty} [T_{x,w}(j\omega)]^* \omega = \Phi_y^*Q_F\Phi_y \]
This formulation reveals why \( P_G \) is the upper-left part of the controllability Gramian of \( F \), and similarly why \( Q_G \) is the lower-right part of \( Q_F \). Using \( N = \mathcal{F}(M, K) \), we can also write
\[ T_{x,w} = N_{sy}C(sl - A - BN_{\pi\pi}C)^{-1} \]
\[ = N_{sy}C((sl - A)(I - (sl - A)^{-1}BN_{\pi\pi}C))^{-1} \]
\[ = N_{sy}(I - (sl - A)^{-1}BN_{\pi\pi}C)^{-1}B(I - A)^{-1} \]
\[ = N_{sy}(I - C(sl - A)^{-1}BN_{\pi\pi}C)^{-1}C(sl - A)^{-1} = N_{sy}(I - GN_{\pi\pi})^{-1}C(sl - A)^{-1} \tag{15} \]
This expression will be used in the next section. Similarly, we have
\[ T_{x,w} = (sl - A)^{-1}B(I - N_{\pi\pi}G)^{-1}N_{uw} \tag{16} \]
Note that \( T_{x,w} \) (13) and \( T_{x,w} \) (14) are stable for stable \( F \), whereas the formulas in (15) and (16) seem to point to instability for \( G \) unstable; this means that cancellation of all unstable poles occur.

A.4. Comparison of different balancing concepts

First, we investigate the relation between closed-loop balanced reduction and frequency weighted balanced reduction. Therefore, we write the twin feedback system in an elementary feedback form in which the model \( G \) and controller \( K \) are taken together:
\[ F = \mathcal{F}(G, M, K) = \Phi_xE\Phi_w \]
with
\[ E = M(I - \Lambda M)^{-1} \tag{17} \]
\[ \Lambda = \text{diag}(G, O_{xx}, K) \tag{18} \]
With \( E_0 = M(I - \Lambda_0M)^{-1} \) and \( \bar{E} = M(I - \bar{\Lambda}M)^{-1} \), we can derive
\[ \bar{E} - E_0 = \bar{E}(\bar{\Lambda} - \Lambda_0)E_0 \tag{19} \]
This means that
\[ \mathcal{F}(\bar{G}, M, \bar{K}) - \mathcal{F}(G_0, M, K_0) = \Phi_x\bar{E}(\bar{\Lambda} - \Lambda_0)E_0\Phi_w \hat{=} W_{\Delta\Delta}(\bar{\Lambda} - \Lambda_0)V_{\Delta\Delta} \]
with
\[ W_{\Delta\Delta} = \Phi_x\bar{E}, \ V_{\Delta\Delta} = E_0\Phi_w \]
We conclude that we managed to rewrite the error due to any reduction, as a weighted reduction with the input weight known and the output weight depending on the reduced models in \( \bar{\Lambda} \). We may remove this
dependence, if \( \Phi_d(\bar{E} - E_0) \) is sufficiently small; let \( \delta \Lambda = \bar{\Lambda} - \Lambda_0 \), \( V_{\delta \Lambda} = E_0 \Phi_w = V_{\Lambda \Lambda} \) and \( W_{\delta \Lambda} = \Phi_y E_0 \), then
\[
\mathcal{F}(\bar{G}, M, \bar{K}) - \mathcal{F}(G_0, M, K_0) \approx W_{\delta \Lambda} \delta \Lambda V_{\delta \Lambda}
\] (20)

Another way to derive this result is to write \( \bar{E} \) as a series expansion,
\[
\bar{E} = E_0(I + [(\bar{\Lambda} - \Lambda_0)E_0] + [(\bar{\Lambda} - \Lambda_0)E_0]^2 + [\cdot]^3 + \cdots)
\]
which shows that for \((\bar{\Lambda} - \Lambda_0)E_0 \) sufficiently small, we may use the following first-order approximation
\[
\bar{E} - E_0 \approx E_0(\bar{\Lambda} - \Lambda_0)E_0
\]

We continue with rewriting (20) as
\[
\mathcal{F}(\bar{G}, M, \bar{K}) \approx \mathcal{F}(G_0, M, K_0) + W_{\delta G} \delta GV_{\delta G} + W_{\delta K} \delta KV_{\delta K}
\] (21)

with
\[
W_{\delta G} = \Phi_x E \Phi_y, \quad V_{\delta G} = \Phi_x E \Phi_w, \quad W_{\delta K} = \Phi_y E \Phi_w, \quad V_{\delta K} = \Phi_y E \Phi_w
\] (22)

The weights \( V_{\delta G} \) and \( W_{\delta G} \) are used implicitly in closed-loop balanced reduction of \( G \). This will be derived for \( W_{\delta G} \). First note that
\[
\mathcal{F}(I, K) \triangleq N = M(I - \Phi_y K \Phi_x M)^{-1} = MR^{-1}
\]
with \( R = I - \Phi_y K \Phi_x M \). Then
\[
W_{\delta G} = \Phi_x E \Phi_y = \Phi_x M(I - \Lambda M)^{-1} \Phi_y
\]
\[= \Phi_x M(I - \Phi_y G \Phi_w K \Phi_x M)^{-1} \Phi_y
\]
\[= \Phi_x M((I - \Phi_y G \Phi_w M R^{-1}) R)^{-1} \Phi_y
\]
\[= \Phi_x M R^{-1} \Phi_y (I - G \Phi_w M R^{-1} \Phi_y)^{-1}
\]
\[= \Phi_x N \Phi_y (I - G \Phi_w N \Phi_y)^{-1}
\]
\[= N_{\delta G}(I - G N_{\delta G})^{-1}
\]

This means that (15)
\[
T_{\text{exa}} = W_{\delta \Lambda} C(sI - A)^{-1}
\]
and this is exactly the transfer function (12) governing the observability Gramian in the frequency-weighted case with output weight \( W_{\delta \Lambda} \).

We may conclude that closed-loop balanced model reduction can be reformulated as a frequency-weighted balanced reduction with specific weights
\[
\text{bal } \mathcal{R}(\mathcal{F}(G, M, K)) = \text{bal } \mathcal{R}(N, G) = \text{bal } \mathcal{R}(W_{\delta G} GV_{\delta G})
\] (23)

Notice that any frequency-weighted balanced reduction can be formulated as a closed-loop balanced reduction
\[
\text{bal } \mathcal{R}(WGV) = \text{bal } (\mathcal{F}_u(N, G))
\]

with
\[
N = \begin{bmatrix} O & V \\ W & O \end{bmatrix}
\]

Thus, frequency-weighted balanced reduction is a special case of closed-loop balanced reduction.
REFERENCES


