Design of Sparse Digital Finite-Precision Controller Structures Based on an Improved Closed-Loop Stability Related Measure *

J. Wu† and S. Chen
Department of Electronics and Computer Science
University of Southampton, Southampton SO17 1BJ, U.K.
jw01v@ecs.soton.ac.uk   sqc@ecs.soton.ac.uk

ABSTRACT

An improved closed-loop stability measure is derived for digital controller structures with finite-word-length (FWL) implementation, which takes into account the number of trivial elements in a controller realization. A practical procedure is presented to design sparse controller realizations with good FWL closed-loop stability characteristics. A case study shows that the proposed design procedure yields computationally efficient controller realizations with enhanced FWL closed-loop stability performance.

KEY WORDS

Digital controller, finite word length, closed-loop stability, sparse realization, optimization, real-time computation.

1. Introduction

A designed stable control system may achieve a lower than predicted performance or even become unstable when the controller is implemented with a finite-precision device. In real-time applications where computational efficiency is critical, a digital controller implemented in fixed-point arithmetic has some advantages. With a fixed-point processor, the detrimental FWL effects are markedly increased due to a reduced precision. It is well-known that FWL effects on the closed-loop stability depend on the controller realization structure. This fact can be used to find “optimal” realizations of controllers based on various FWL stability measures [1]-[7]. However, these design methods usually yield fully parameterized controller structures.

It is highly desirable that a controller realization has a sparse structure with many trivial elements of 0, 1 or -1. This is particularly important for real-time applications with high-order controllers, as it will achieve better computational efficiency. A canonical controller realization has sparse structure but may not have the required FWL stability robustness. This poses a complex problem of finding sparse controller realizations with good FWL closed-loop stability characteristics. In the works [8],[9], a design procedure has been given to obtain sparse controller realizations based on a FWL pole-sensitivity stability measure.

This study derives an improved FWL closed-loop stability measure, which takes into account the number of trivial elements in a controller realization. A practical procedure is proposed, which first maximizes a lower bound of the proposed stability measure and the resulting controller realization is then made sparse using an iterative stepwise algorithm originally developed for filter design [2],[10]. The proposed method has some advantages over the existing methods [5],[8],[9], as it is more accurate in estimating the robustness of the FWL closed-loop stability and the computational complexity is considerably reduced. A design example is used to test the proposed method.

2. The problem formulation

Consider the discrete-time closed-loop control system with a linear time-invariant plant $P(z)$ and a digital controller $C(z)$. The plant $P(z)$ is strictly proper with a state-space description $(\mathbf{A}_P, \mathbf{B}_P, \mathbf{C}_P)$, where $\mathbf{A}_P \in \mathcal{R}^{m\times m}$, $\mathbf{B}_P \in \mathcal{R}^{m\times l}$ and $\mathbf{C}_P \in \mathcal{R}^{q\times m}$. Let $(\mathbf{A}_C, \mathbf{B}_C, \mathbf{C}_C, \mathbf{D}_C)$ be a state-space description of the controller $C(z)$, with $\mathbf{A}_C \in \mathcal{R}^{n\times n}$, $\mathbf{B}_C \in \mathcal{R}^{n\times q}$, $\mathbf{C}_C \in \mathcal{R}^{l\times n}$ and $\mathbf{D}_C \in \mathcal{R}^{l\times q}$. Given the transfer function matrix $C(z)$, there are infinite state-space descriptions. In fact, if $(\mathbf{A}_C^0, \mathbf{B}_C^0, \mathbf{C}_C^0, \mathbf{D}_C^0)$ is a state-space description of $C(z)$, all the state-space descriptions of $C(z)$ form a *realization* set

$$\mathcal{S}_C \triangleq \{(\mathbf{A}_C, \mathbf{B}_C, \mathbf{C}_C, \mathbf{D}_C) \mid \mathbf{A}_C = \mathbf{T}^{-1}\mathbf{A}_C^0\mathbf{T}, \mathbf{B}_C = \mathbf{T}^{-1}\mathbf{B}_C^0, \mathbf{C}_C = \mathbf{C}_C^0\mathbf{T}, \mathbf{D}_C = \mathbf{D}_C^0\} \quad (1)$$

where $\mathbf{T} \in \mathcal{R}^{n\times n}$ is any non-singular matrix. Denote

$$\mathbf{X} \triangleq \begin{bmatrix} \mathbf{D}_C & \mathbf{C}_C \\ \mathbf{B}_C & \mathbf{A}_C \end{bmatrix} = \begin{bmatrix} x_1 & \cdots & x_{N-l-n+1} \\ x_2 & \cdots & x_{N-l-n+2} \\ \vdots & \cdots & \vdots \\ x_{l+n} & \cdots & x_N \end{bmatrix} \quad (2)$$

where $N = (l+n)(q+n)$. The stability of the closed-loop control system depends on the eigenvalues of the closed-loop system matrix

$$\overline{\mathbf{A}}(\mathbf{X}) = \begin{bmatrix} \mathbf{A}_P & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{B}_P & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_n \end{bmatrix} \mathbf{X} \begin{bmatrix} \mathbf{C}_P & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_n \end{bmatrix}$$

*The work was supported by the U.K. Royal Society under a KC Wong fellowship (RL/ART/CN/XFI/KCW/11949).

†On leave from Institute of Industrial Process Control, Zhejiang University, Hangzhou, 310027, P. R. China

$$\triangleq \mathbf{M}_0 + \mathbf{M}_1\mathbf{X}\mathbf{M}_2 \quad (3)$$

where $\mathbf{0}$ is the zero matrix of appropriate dimension and $\mathbf{I}_n$ the $n \times n$ identity matrix. All the different realizations $\mathbf{X}$ in $\mathcal{S}_C$ have the same set of closed-loop poles if they are implemented with infinite precision. Since the closed-loop system has been designed to be stable, all the eigenvalues $\lambda_i(\overline{\mathbf{A}}(\mathbf{X}))$, $1 \le i \le m+n$, are within the unit disk.

When $\mathbf{X}$ is implemented with a fixed-point processor of B_s bits, it is perturbed to $\mathbf{X}+\Delta\mathbf{X}$ due to the FWL effect. Each element of $\Delta\mathbf{X}$ is bounded by $\pm\varepsilon/2$, that is,

$$\mu(\Delta\mathbf{X}) \triangleq \max_{j\in\{1,\cdots,N\}} |\Delta x_j| \le \varepsilon/2 \quad (4)$$

Let $B_s = B_i + B_f$, where B_i ensures that the absolute value of each element of $2^{-B_i}\mathbf{X}$ is no larger than 1. Thus, B_i are bits required for the integer part of a number and B_f are bits used to implement the fractional part of a number. It can be shown that $\varepsilon = 2^{-B_f}$. With the perturbation $\Delta\mathbf{X}$, $\lambda_i(\overline{\mathbf{A}}(\mathbf{X}))$ is moved to $\lambda_i(\overline{\mathbf{A}}(\mathbf{X}+\Delta\mathbf{X}))$. If a pole of $\overline{\mathbf{A}}(\mathbf{X}+\Delta\mathbf{X})$ is outside the open unit disk, the closed-loop system becomes unstable with B_s-bit implemented $\mathbf{X}$. Another important consideration is the sparseness of $\mathbf{X}$. Those elements of $\mathbf{X}$, which have values 0, 1 or -1, are *trivial* parameters. A trivial parameter requires no operations in the fixed-point implementation and does not cause any computational error at all. Thus $\Delta x_j = 0$ when $x_j = 0, 1$ or -1. Let we define an indicator function as

$$\delta(x) = \begin{cases} 0, & \text{if } x = 0, 1 \text{ or } -1 \\ 1, & \text{otherwise} \end{cases} \quad (5)$$

When the FWL error $\Delta\mathbf{X}$ is small,

$$\Delta\left|\lambda_i\right| \triangleq \left|\lambda_i(\overline{\mathbf{A}}(\mathbf{X}+\Delta\mathbf{X}))\right| - \left|\lambda_i(\overline{\mathbf{A}}(\mathbf{X}))\right|$$

$$\approx \sum_{j=1}^{N} \frac{\partial\left|\lambda_i\right|}{\partial x_j}\Delta x_j \delta(x_j), \ \ \forall i \in \{1,\cdots,m+n\} \quad (6)$$

where $\frac{\partial|\lambda_i|}{\partial x_j}$ is evaluated at $\mathbf{X}$. It follows from the Cauchy inequality that

$$|\Delta|\lambda_i|| \le \sqrt{N_s \sum_{j=1}^{N} \left|\frac{\partial\left|\lambda_i\right|}{\partial x_j}\right|^2 \left|\Delta x_j\right|^2 \delta(x_j)}$$

$$\le \mu(\Delta\mathbf{X})\sqrt{N_s \sum_{j=1}^{N} \left|\frac{\partial\left|\lambda_i\right|}{\partial x_j}\right|^2 \delta(x_j)}, \ \ \forall i \quad (7)$$

where N_s is the number of the nontrivial elements in $\mathbf{X}$. This leads to the following stability measure

$$\mu_1(\mathbf{X}) = \min_{i\in\{1,\cdots,m+n\}} \frac{1-\left|\lambda_i(\overline{\mathbf{A}}(\mathbf{X}))\right|}{\sqrt{N_s \sum_{j=1}^{N} \delta(x_j)\left|\frac{\partial|\lambda_i|}{\partial x_j}\right|^2}} \quad (8)$$

If $\mu(\Delta\mathbf{X}) < \mu_1(\mathbf{X})$, it follows from (7) and (8) that $|\Delta|\lambda_i|| < 1 - \left|\lambda_i(\overline{\mathbf{A}}(\mathbf{X}))\right|$. Therefore

$$\left|\lambda_i(\overline{\mathbf{A}}(\mathbf{X}+\Delta\mathbf{X}))\right| \le |\Delta|\lambda_i|| + \left|\lambda_i(\overline{\mathbf{A}}(\mathbf{X}))\right| < 1 \quad (9)$$

which means that the closed-loop system remains stable under the FWL error $\Delta\mathbf{X}$. The larger $\mu_1(\mathbf{X})$ is, the larger FWL errors the closed-loop system can tolerate. Hence, $\mu_1(\mathbf{X})$ is a stability measure describing the FWL closed-loop stability robustness of a controller realization $\mathbf{X}$.

Noting the result of how to calculate $\frac{\partial\lambda_i}{\partial\mathbf{X}}$ [5],[7] and the following relationship

$$\frac{\partial|\lambda_i|}{\partial\mathbf{X}} = \frac{1}{|\lambda_i|}\mathrm{Re}\left[\lambda_i^* \frac{\partial\lambda_i}{\partial\mathbf{X}}\right] \quad (10)$$

leads to the following proposition, which shows that, given a $\mathbf{X}$, the value of $\mu_1(\mathbf{X})$ can easily be calculated.

Proposition 1 Let $\overline{\mathbf{A}}(\mathbf{X}) = \mathbf{M}_0 + \mathbf{M}_1\mathbf{X}\mathbf{M}_2$ given in (3) be diagonalisable, and have eigenvalues $\{\lambda_i\} = \{\lambda_i(\overline{\mathbf{A}}(\mathbf{X}))\}$. Denote $\mathbf{p}_i$ a right eigenvector of $\overline{\mathbf{A}}(\mathbf{X})$ related to the eigenvalue λ_i. Define $\mathbf{M}_p \triangleq [\mathbf{p}_1\ \mathbf{p}_2 \cdots \mathbf{p}_{m+n}]$ and $\mathbf{M}_y \triangleq [\mathbf{y}_1\ \mathbf{y}_2 \cdots \mathbf{y}_{m+n}] = \mathbf{M}_p^{-H}$, where $\mathbf{y}_i$ is the reciprocal left eigenvector related to λ_i. Then

$$\frac{\partial|\lambda_i|}{\partial\mathbf{X}} = \begin{bmatrix} \frac{\partial|\lambda_i|}{\partial x_1} & \cdots & \frac{\partial|\lambda_i|}{\partial x_{N-l-n+1}} \\ \vdots & \cdots & \vdots \\ \frac{\partial|\lambda_i|}{\partial x_{l+n}} & \cdots & \frac{\partial|\lambda_i|}{\partial x_N} \end{bmatrix}$$

$$= \frac{1}{|\lambda_i|}\mathbf{M}_1^T \mathrm{Re}\left[\lambda_i^* \mathbf{y}_i^* \mathbf{p}_i^T\right]\mathbf{M}_2^T \quad (11)$$

By considering the sensitivity of eigenvalue moduli rather than the sensitivity of eigenvalues, the stability measure (8) is different from the existing measure [5],[8],[9], and it generally provides a more accurate estimate for the robustness of FWL closed-loop stability. It is also worth pointing out that this improved measure has considerable computational advantages over the existing one. This is because $\frac{\partial|\lambda_i|}{\partial\mathbf{X}}$ is real-valued while $\frac{\partial\lambda_i}{\partial\mathbf{X}}$ is complex-valued. Thus the optimization process and sparse transformation procedure, discussed in the next section, require much less computation than the previous approach [5],[8],[9], unless all the system eigenvalues are real-valued in which case $\mu_1(\mathbf{X})$ and the existing measure become identical.

3. The design procedure

The optimal sparse controller realization with a maximum tolerance to FWL perturbation in principle is the solution of the following optimization problem:

$$\upsilon \triangleq \max_{\mathbf{X}\in\mathcal{S}_C} \mu_1(\mathbf{X}) \quad (12)$$

However, it is difficult to solve for the above optimization problem because $\mu_1(\mathbf{X})$ includes $\delta(x_j)$ and is not a continuous function with respect to controller parameters x_j.

Consider a lower bound of $\mu_1(\mathbf{X})$ defined by

$$\underline{\mu_1}(\mathbf{X}) = \min_{i\in\{1,\cdots,m+n\}} \frac{1-\left|\lambda_i(\overline{\mathbf{A}}(\mathbf{X}))\right|}{\sqrt{N\sum_{j=1}^{N}\left|\frac{\partial|\lambda_i|}{\partial x_j}\right|^2}} \tag{13}$$

Obviously, $\underline{\mu_1}(\mathbf{X}) \le \mu_1(\mathbf{X})$ and $\underline{\mu_1}(\mathbf{X})$ is a continuous function of controller parameters. It is relatively easy to optimize $\underline{\mu_1}(\mathbf{X})$ (e.g. [7]). Let the "optimal" controller realization $\mathbf{X}_{\rm opt}$ be the solution of the optimization problem

$$\omega \triangleq \max_{\mathbf{X}\in\mathcal{S}_C} \underline{\mu_1}(\mathbf{X}) \tag{14}$$

$\mathbf{X}_{\rm opt}$ is generally not the optimal solution of (12) and does not have a sparse structure. However, it can readily be attempted by the following optimization procedure.

3.1 Optimization of the lower-bound measure

Assume that an initial realization has been obtained by some design procedure and is denoted as $\mathbf{X}_0$. According to (1)–(3), a similarity transformation of $\mathbf{X}_0$ by $\mathbf{T}$ is

$$\mathbf{X} = \mathbf{X}(\mathbf{T}) = \begin{bmatrix} \mathbf{I}_l & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^{-1} \end{bmatrix} \mathbf{X}_0 \begin{bmatrix} \mathbf{I}_q & \mathbf{0} \\ \mathbf{0} & \mathbf{T} \end{bmatrix} \tag{15}$$

where $\det(\mathbf{T}) \neq 0$. The closed-loop system matrix for the realization $\mathbf{X}$ is

$$\overline{\mathbf{A}}(\mathbf{X}) = \begin{bmatrix} \mathbf{I}_m & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^{-1} \end{bmatrix} \overline{\mathbf{A}}(\mathbf{X}_0) \begin{bmatrix} \mathbf{I}_m & \mathbf{0} \\ \mathbf{0} & \mathbf{T} \end{bmatrix} \tag{16}$$

Obviously $\overline{\mathbf{A}}(\mathbf{X})$ has the same set of eigenvalues as $\overline{\mathbf{A}}(\mathbf{X}_0)$, denoted as $\{\lambda_i^0\}$. Applying proposition 1 to (16) results in

$$\left.\frac{\partial|\lambda_i|}{\partial \mathbf{X}}\right|_{\mathbf{X}(\mathbf{T})} = \begin{bmatrix} \mathbf{I}_l & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^{T} \end{bmatrix} \left.\frac{\partial|\lambda_i|}{\partial \mathbf{X}}\right|_{\mathbf{X}_0} \begin{bmatrix} \mathbf{I}_q & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^{-T} \end{bmatrix} \tag{17}$$

For a complex-valued matrix $\mathbf{M} \in \mathcal{C}^{(l+n)\times(q+n)}$ with elements m_{sk}, denote the Frobenius norm

$$\|\mathbf{M}\|_F \triangleq \sqrt{\sum_{s=1}^{l+n}\sum_{k=1}^{q+n} m_{sk}^* m_{sk}} \tag{18}$$

Then the lower-bound measure (13) can be rewritten as

$$\begin{aligned} \underline{\mu_1}(\mathbf{X}) = & \\ \min_{i\in\{1,\cdots,m+n\}} & \frac{1-|\lambda_i^0|}{\sqrt{N}\left\|\begin{bmatrix} \mathbf{I}_l & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^{T} \end{bmatrix} \left.\frac{\partial|\lambda_i|}{\partial \mathbf{X}}\right|_{\mathbf{X}_0} \begin{bmatrix} \mathbf{I}_q & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^{-T} \end{bmatrix}\right\|_F} \\ = \min_{i\in\{1,\cdots,m+n\}} & \frac{1}{\sqrt{N}\left\|\begin{bmatrix} \mathbf{I}_l & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^{T} \end{bmatrix} \mathbf{\Phi}_i \begin{bmatrix} \mathbf{I}_q & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^{-T} \end{bmatrix}\right\|_F} \end{aligned} \tag{19}$$

where

$$\mathbf{\Phi}_i \triangleq \frac{\left.\frac{\partial|\lambda_i|}{\partial \mathbf{X}}\right|_{\mathbf{X}_0}}{1-|\lambda_i^0|} \tag{20}$$

If we introduce the cost function

$$\begin{aligned} f(\mathbf{T}) &= \underline{\mu_1}(\mathbf{X}) \\ = \min_{i\in\{1,\cdots,m+n\}} & \frac{1}{\sqrt{N}\left\|\begin{bmatrix} \mathbf{I}_l & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^{T} \end{bmatrix} \mathbf{\Phi}_i \begin{bmatrix} \mathbf{I}_q & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^{-T} \end{bmatrix}\right\|_F} \end{aligned} \tag{21}$$

the optimal similarity transformation $\mathbf{T}_{\rm opt}$ can be obtained by solving for the unconstrained optimization problem:

$$\omega = \max_{\mathbf{T}\in\mathcal{R}^{n\times n}} f(\mathbf{T}) \tag{22}$$

with a measure of monitoring the singular values of $\mathbf{T}$ to make sure that $\det(\mathbf{T}) \neq 0$. The unconstrained optimization problem (22) can be solved, for example, using the adaptive simulated annealing (ASA) algorithm [11]. With $\mathbf{T}_{\rm opt}$, the corresponding optimal realization $\mathbf{X}_{\rm opt}$ that is the solution of (14) can readily be computed.

3.2 Stepwise transformation for sparse realizations

As the optimal sparse realization that maximizes μ_1 is difficult if not impossible to obtain, we search for a suboptimal solution of (12). Since $\mathbf{X}_{\rm opt}$ maximizes $\underline{\mu_1}$ and $\underline{\mu_1}$ is a lower-bound of μ_1, $\mathbf{X}_{\rm opt}$ will produce a satisfactory large value of μ_1, although it usually contains no trivial elements. We make $\mathbf{X}_{\rm opt}$ sparse by changing one nontrivial element of $\mathbf{X}_{\rm opt}$ into a trivial one at a step, under the condition that the value of $\underline{\mu_1}$ does not reduce too much. This process produces a sparse realization $\mathbf{X}_{\rm spa}$ with a satisfactory value of $\underline{\mu_1}$. Clearly $\mathbf{X}_{\rm spa}$ is not a true optimal solution of (12). Notice that, even though $\underline{\mu_1}(\mathbf{X}_{\rm spa}) \le \underline{\mu_1}(\mathbf{X}_{\rm opt})$, it is possible that $\mu_1(\mathbf{X}_{\rm spa}) \ge \mu_1(\mathbf{X}_{\rm opt})$. In other words, $\mathbf{X}_{\rm spa}$ may actually have better FWL stability performance than $\mathbf{X}_{\rm opt}$. The stepwise procedure for obtaining $\mathbf{X}_{\rm spa}$ is:

Step 1: Set τ to a very small positive real number (e.g. 10^{-5}). The transformation matrix $\mathbf{T} \in \mathcal{R}^{n\times n}$ is initially set to $\mathbf{T}_{\rm opt}$ so that $\mathbf{X}(\mathbf{T}) = \mathbf{X}_{\rm opt}$.

Step 2: Find out all the trivial elements $\{\eta_1, \cdots, \eta_r\}$ in $\mathbf{X}(\mathbf{T})$ (a parameter is considered to be trivial if its distance to 0, 1 or -1 is less than a tolerance, say 10^{-8}). Denote ξ the nontrivial element in $\mathbf{X}(\mathbf{T})$ that is the nearest to 0, 1 or -1.

Step 3: Choose $\mathbf{S} \in \mathcal{R}^{n\times n}$ such that

i) $\underline{\mu_1}(\mathbf{X}(\mathbf{T}+\tau\mathbf{S}))$ is close to $\underline{\mu_1}(\mathbf{X}(\mathbf{T}))$.

ii) $\{\eta_1, \cdots, \eta_r\}$ in $\mathbf{X}(\mathbf{T})$ remain unchanged in $\mathbf{X}(\mathbf{T}+\tau\mathbf{S})$.

iii) ξ in $\mathbf{X}(\mathbf{T})$ is changed as nearer as possible to 0, 1 or -1 in $\mathbf{X}(\mathbf{T}+\tau\mathbf{S})$.

iv) $\|\mathbf{S}\|_F = 1$.

If $\mathbf{S}$ does not exist, $\mathbf{T}_{\text{spa}} = \mathbf{T}$ and terminate the algorithm.

Step 4: $\mathbf{T} = \mathbf{T} + \tau\mathbf{S}$. If ξ in $\mathbf{X}(\mathbf{T})$ is nontrivial, go to step 3. If ξ becomes trivial, go to step 2.

The **Step 3** guarantees that $\mathbf{X}(\mathbf{T}_{\text{spa}})$ has good performance as measured by $\underline{\mu_1}$ and contains many trivial parameters. The key is how to obtain $\mathbf{S}$. Denote $\mathrm{Vec}(\cdot)$ the column stacking operator. With a very small τ, condition i) means that

$$\left(\mathrm{Vec}\left(\frac{d\underline{\mu_1}}{d\mathbf{T}}\right)\right)^T \mathrm{Vec}\left(\mathbf{S}\right) = 0 \tag{23}$$

and condition ii) means that

$$\begin{cases} \left(\mathrm{Vec}\left(\frac{d\eta_1}{d\mathbf{T}}\right)\right)^T \mathrm{Vec}\left(\mathbf{S}\right) = 0 \\ \quad\quad\quad\vdots \\ \left(\mathrm{Vec}\left(\frac{d\eta_r}{d\mathbf{T}}\right)\right)^T \mathrm{Vec}\left(\mathbf{S}\right) = 0 \end{cases} \tag{24}$$

Denote the matrix

$$\mathbf{E} \triangleq \begin{bmatrix} \left(\mathrm{Vec}\left(\frac{d\underline{\mu_1}}{d\mathbf{T}}\right)\right)^T \\ \left(\mathrm{Vec}\left(\frac{d\eta_1}{d\mathbf{T}}\right)\right)^T \\ \vdots \\ \left(\mathrm{Vec}\left(\frac{d\eta_r}{d\mathbf{T}}\right)\right)^T \end{bmatrix} \in \mathcal{R}^{(r+1)\times n^2} \tag{25}$$

$\mathrm{Vec}(\mathbf{S})$ must belong to the null space $\mathcal{N}(\mathbf{E})$ of $\mathbf{E}$. If $\mathcal{N}(\mathbf{E})$ is empty, $\mathrm{Vec}(\mathbf{S})$ does not exist and the algorithm is terminated. If $\mathcal{N}(\mathbf{E})$ is not empty, it must have basis $\{\mathbf{b}_1, \cdots, \mathbf{b}_t\}$, assuming that the dimension of $\mathcal{N}(\mathbf{E})$ is t. Condition iii) requires moving ξ to its desired value (0, 1 or -1) as fast as possible, and we should choose $\mathrm{Vec}(\mathbf{S})$ as the orthogonal projection of $\mathrm{Vec}\left(\frac{d\xi}{d\mathbf{T}}\right)$ onto $\mathcal{N}(\mathbf{E})$. Noting condition iv), we can compute $\mathrm{Vec}(\mathbf{S})$ as follows:

$$a_i = \mathbf{b}_i^T \mathrm{Vec}\left(\frac{d\xi}{d\mathbf{T}}\right) \in \mathcal{R},\ \ \forall i \in \{1, \cdots, t\} \tag{26}$$

$$\mathbf{v} = \sum_{i=1}^{t} a_i \mathbf{b}_i \in \mathcal{R}^{n^2} \tag{27}$$

$$\mathrm{Vec}(\mathbf{S}) = \pm \frac{\mathbf{v}}{\sqrt{\mathbf{v}^T\mathbf{v}}} \in \mathcal{R}^{n^2} \tag{28}$$

The sign in (28) is chosen in the following way. If ξ is larger than its nearest desired value, the minus sign is taken; otherwise, the plus sign is used.

For calculating the required derivatives $\frac{d\underline{\mu_1}}{d\mathbf{T}}$, $\frac{d\xi}{d\mathbf{T}}$, $\frac{d\eta_1}{d\mathbf{T}}, \cdots, \frac{d\eta_r}{d\mathbf{T}}$, the following well-known fact is useful. Given any element y_{ij} in a nonsingular $\mathbf{Y} \in R^{n\times n}$ with $i \in \{1, \cdots, n\}$ and $j \in \{1, \cdots, n\}$,

$$\frac{\partial \mathbf{Y}}{\partial y_{ij}} = \mathbf{e}_i\mathbf{e}_j^T \quad \text{and} \quad \frac{\partial \mathbf{Y}^{-1}}{\partial y_{ij}} = -\mathbf{Y}^{-1}\mathbf{e}_i\mathbf{e}_j^T\mathbf{Y}^{-1} \tag{29}$$

where $\mathbf{e}_i$ denotes the ith coordinate vector. In (15), define

$$\mathbf{U}_1 = \begin{bmatrix} \mathbf{I}_l & \mathbf{0} \\ \mathbf{0} & \mathbf{T} \end{bmatrix} \quad \text{and} \quad \mathbf{U}_2 = \begin{bmatrix} \mathbf{I}_q & \mathbf{0} \\ \mathbf{0} & \mathbf{T} \end{bmatrix} \tag{30}$$

For any element x_{ks} in $\mathbf{X} = \mathbf{U}_1^{-1}\mathbf{X}_0\mathbf{U}_2$, where $k \in \{1, \cdots, l+n\}$ and $s \in \{1, \cdots, q+n\}$, and any t_{ij} in $\mathbf{T}$, where $i \in \{1, \cdots, n\}$ and $j \in \{1, \cdots, n\}$,

$$\begin{aligned} \frac{\partial x_{ks}}{\partial t_{ij}} &= \mathbf{e}_k^T \frac{\partial \mathbf{U}_1^{-1}}{\partial t_{ij}} \mathbf{X}_0\mathbf{U}_2\mathbf{e}_s + \mathbf{e}_k^T\mathbf{U}_1^{-1}\mathbf{X}_0\frac{\partial \mathbf{U}_2}{\partial t_{ij}}\mathbf{e}_s \\ &= -\mathbf{e}_k^T\mathbf{U}_1^{-1}\mathbf{e}_{l+i}\mathbf{e}_{l+j}^T\mathbf{U}_1^{-1}\mathbf{X}_0\mathbf{U}_2\mathbf{e}_s + \mathbf{e}_k^T\mathbf{U}_1^{-1}\mathbf{X}_0\mathbf{e}_{q+i}\mathbf{e}_{q+j}^T\mathbf{e}_s \\ &= -\mathbf{e}_k^T\mathbf{U}_1^{-1}\mathbf{e}_{l+i}\mathbf{e}_{l+j}^T\mathbf{X}\mathbf{e}_s + \mathbf{e}_k^T\mathbf{U}_1^{-1}\mathbf{X}_0\mathbf{e}_{q+i}\mathbf{e}_{q+j}^T\mathbf{e}_s \end{aligned} \tag{31}$$

That is,

$$\begin{aligned} \frac{dx_{ks}}{d\mathbf{T}} = &\begin{bmatrix} \mathbf{e}_k^T\mathbf{U}_1^{-1} & & \\ & \ddots & \\ & & \mathbf{e}_k^T\mathbf{U}_1^{-1} \end{bmatrix} \times \\ &\left(\begin{bmatrix} \mathbf{X}_0\mathbf{e}_{q+1}\mathbf{e}_{q+1}^T & \cdots & \mathbf{X}_0\mathbf{e}_{q+1}\mathbf{e}_{q+n}^T \\ \vdots & \cdots & \vdots \\ \mathbf{X}_0\mathbf{e}_{q+n}\mathbf{e}_{q+1}^T & \cdots & \mathbf{X}_0\mathbf{e}_{q+n}\mathbf{e}_{q+n}^T \end{bmatrix}\right. \\ &\left. - \begin{bmatrix} \mathbf{e}_{l+1}\mathbf{e}_{l+1}^T\mathbf{X} & \cdots & \mathbf{e}_{l+1}\mathbf{e}_{l+n}^T\mathbf{X} \\ \vdots & \cdots & \vdots \\ \mathbf{e}_{l+n}\mathbf{e}_{l+1}^T\mathbf{X} & \cdots & \mathbf{e}_{l+n}\mathbf{e}_{l+n}^T\mathbf{X} \end{bmatrix}\right) \begin{bmatrix} \mathbf{e}_s & & \\ & \ddots & \\ & & \mathbf{e}_s \end{bmatrix} \end{aligned} \tag{32}$$

Thus, we can readily calculate $\frac{d\xi}{d\mathbf{T}}, \frac{d\eta_1}{d\mathbf{T}}, \cdots, \frac{d\eta_r}{d\mathbf{T}}$. Let

$$i_0 = \arg\min_{i\in\{1,\cdots,m+n\}} \frac{1}{\sqrt{N}\left\| \begin{bmatrix} \mathbf{I}_l & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^T \end{bmatrix} \boldsymbol{\Phi_i} \begin{bmatrix} \mathbf{I}_q & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^{-T} \end{bmatrix} \right\|_F} \tag{33}$$

Similar to the derivation of $\frac{dx_{ks}}{d\mathbf{T}}$, for any element w_{ks} in $\mathbf{W} = \mathbf{U}_1^T\boldsymbol{\Phi}_{i_0}\mathbf{U}_2^{-T}$, where $k \in \{1, \cdots, l+n\}$ and $s \in \{1, \cdots, q+n\}$, we have

$$\begin{aligned} \frac{dw_{ks}}{d\mathbf{T}} = &\begin{bmatrix} \mathbf{e}_k^T & & \\ & \ddots & \\ & & \mathbf{e}_k^T \end{bmatrix} \times \\ &\left(\begin{bmatrix} \mathbf{e}_{l+1}\mathbf{e}_{l+1}^T\boldsymbol{\Phi}_{i_0} & \cdots & \mathbf{e}_{l+n}\mathbf{e}_{l+1}^T\boldsymbol{\Phi}_{i_0} \\ \vdots & \cdots & \vdots \\ \mathbf{e}_{l+1}\mathbf{e}_{l+n}^T\boldsymbol{\Phi}_{i_0} & \cdots & \mathbf{e}_{l+n}\mathbf{e}_{l+n}^T\boldsymbol{\Phi}_{i_0} \end{bmatrix}\right. \\ &\left. - \begin{bmatrix} \mathbf{W}\mathbf{e}_{q+1}\mathbf{e}_{q+1}^T & \cdots & \mathbf{W}\mathbf{e}_{q+n}\mathbf{e}_{q+1}^T \\ \vdots & \cdots & \vdots \\ \mathbf{W}\mathbf{e}_{q+1}\mathbf{e}_{q+n}^T & \cdots & \mathbf{W}\mathbf{e}_{q+n}\mathbf{e}_{q+n}^T \end{bmatrix}\right) \times \\ &\begin{bmatrix} \mathbf{U}_2^{-T}\mathbf{e}_s & & \\ & \ddots & \\ & & \mathbf{U}_2^{-T}\mathbf{e}_s \end{bmatrix} \end{aligned} \tag{34}$$

Since

$$\underline{\mu_1} = \frac{1}{\sqrt{N}\sqrt{\sum_{k=1}^{l+n}\sum_{s=1}^{q+n} w_{ks}^* w_{ks}}} \tag{35}$$

We can calculate

$$\frac{d\underline{\mu}_1}{d\mathbf{T}} = -\frac{1}{\sqrt{N}\,\|\mathbf{W}\|_F^3}\mathrm{Re}\left[\sum_{k=1}^{l+n}\sum_{s=1}^{q+n} w_{ks}^* \frac{dw_{ks}}{d\mathbf{T}}\right] \tag{36}$$

As in [6],[7], an estimated minimum bit length for guaranteeing closed-loop stability based on $\underline{\mu}_1(\mathbf{X})$ is

$$\hat{B}_{s,\min} = B_i + \mathrm{Int}[-\log_2(\underline{\mu}_1(\mathbf{X}))] - 1 \tag{37}$$

where the integer $\mathrm{Int}[x] \geq x$.

4. A numerical example

This was a single-input single-output fluid power speed control system studied in [12],[13]. The plant model was in the continuous-time form and a continuous-time H_∞ optimal controller was designed in [12]. We obtained a discrete-time plant $P(z)$ and a discrete-time controller $C(z)$ by sampling the continuous-time plant and H_∞ controller with a sampling rate of 2 kHz. The discrete-time plant $P(z)$ was given by

$$\mathbf{A}_P = \left[\begin{array}{rrr} 9.9988e-1 & 1.9432e-5 & 5.9320e-5 \\ -4.9631e-7 & 2.3577e-2 & 2.3709e-5 \\ -1.5151e-3 & 2.3709e-2 & 2.3751e-5 \\ 1.5908e-3 & 2.3672e-2 & 2.3898e-5 \end{array}\right.$$

$$\left.\begin{array}{r} -6.2286e-5 \\ 2.3672e-5 \\ 2.3898e-5 \\ 2.3667e-5 \end{array}\right], \quad \mathbf{B}_P = \left[\begin{array}{r} 3.0504e-03 \\ -1.2373e-02 \\ -1.2375e-02 \\ -8.8703e-02 \end{array}\right],$$

$$\mathbf{C}_P = \left[\begin{array}{cccc} 1 & 0 & 0 & 0 \end{array}\right]$$

The initial realization of the controller $C(z)$ given in a controllable canonical form was

$$\mathbf{X}_0 = \left[\begin{array}{rrr} -8.0843e-4 & -1.6112e-3 & -1.5998e-3 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{array}\right.$$

$$\left.\begin{array}{rr} -1.5885e-3 & -1.5773e-3 \\ 0 & -3.3071e-1 \\ 0 & 1.9869e+0 \\ 0 & -3.9816e+0 \\ 1 & 3.3255e+0 \end{array}\right]$$

The closed-loop transition matrix $\overline{\mathbf{A}}(\mathbf{X}_0)$ was formed using (3), from which the eigenvalues and the corresponding eigenvectors of the ideal (infinite-precision) closed-loop system were computed. The optimisation problem (22) was constructed, and the ASA algorithm [11] obtained a solution $\mathbf{T}_{\mathrm{opt}}$. The corresponding controller realization, which maximises the lower-bound measure $\underline{\mu}_1$, was

$$\mathbf{X}_{\mathrm{opt}} = \left[\begin{array}{rrr} -8.0843e-4 & 6.4378e-2 & -1.1974e-2 \\ 2.7588e-3 & 1.0010e+0 & -1.4054e-2 \\ -2.2776e-4 & -5.8175e-2 & 3.3649e-1 \\ -2.5200e-4 & 1.0668e-3 & 1.6778e-2 \\ 8.1179e-3 & 5.1520e-3 & 3.1311e-2 \end{array}\right.$$

$$\left.\begin{array}{rr} -1.1493e-2 & -2.2104e-1 \\ 1.0924e-3 & -8.9552e-3 \\ 7.5457e-2 & 1.3962e-3 \\ 9.9766e-1 & 1.5423e-3 \\ -3.8681e-3 & 9.9031e-1 \end{array}\right]$$

The stepwise transformation was then applied to make $\mathbf{X}_{\mathrm{opt}}$ sparse, which yielded a similarity transformation matrix $\mathbf{T}_{\mathrm{spa}}$ and corresponding controller realization

$$\mathbf{X}_{\mathrm{spa}} = \left[\begin{array}{rrr} -8.0843e-4 & 1.6372e-2 & -5.4228e-4 \\ 0 & 1 & 0 \\ 0 & -6.8678e-2 & 3.3285e-1 \\ 0 & -5.6623e-6 & -7.6002e-4 \\ 2.3061e-2 & -8.1961e-6 & 0 \end{array}\right.$$

$$\left.\begin{array}{rr} -1.8348e-3 & -6.9866e-2 \\ 0 & -1.4073e-3 \\ 4.2230e-1 & 5.8895e-4 \\ 1 & 0 \\ 4.5476e-5 & 9.9262e-1 \end{array}\right]$$

Table 1 compares the FWL closed-loop stability performance and the number of non-trivial elements for the three controller realizations $\mathbf{X}_0$, $\mathbf{X}_{\mathrm{opt}}$ and $\mathbf{X}_{\mathrm{spa}}$, respectively. We also exploited the true minimum bit length that guaranteed closed-loop stability for a controller realization $\mathbf{X}$ using the following computer simulation. Starting with a large enough bit length, e.g. $B_s = 100$, we rounded the controller $\mathbf{X}$ to B_s bits and checked the stability of the closed-loop system, i.e. observing whether the closed-loop poles were within the open unit disk. Reduced B_s by 1 and repeated the process until there appeared to be closed-loop instability at B_u bits. Then $B_{s,\min} = B_u + 1$. The values of $B_{s,\min}$ for the three realizations are given in Table 1. Notice that for $B_s \geq B_{s,\min}$, the B_s-bit implemented controller will always guarantee closed-loop stability. However, there may exist some $B_s < B_u$, which regains closed-loop stability. For example, for the initial realization $\mathbf{X}_0$, $B_u = 32$, i.e. when the bit length is smaller than 33, the closed-loop becomes unstable. At $B_s = 16$ or 15, the closed-loop becomes stable again. With $B_s < 15$ instability is observed again.

For this example, the canonical realization $\mathbf{X}_0$ is the most sparse with 9 non-trivial parameters, but its FWL closed-loop stability measure $\underline{\mu}_1(\mathbf{X}_0)$ is very poor. The realization $\mathbf{X}_{\mathrm{opt}}$ has a much better FWL stability robustness as indicated by $\underline{\mu}_1(\mathbf{X}_{\mathrm{opt}})$, but its all 25 elements are non-trivial. The realization $\mathbf{X}_{\mathrm{spa}}$ has the largest $\mu_1(\mathbf{X}_{\mathrm{spa}})$

Table 1. Comparison of the three controller realizations.

realization	$\mathbf{X}_0$	$\mathbf{X}_{\mathrm{opt}}$	$\mathbf{X}_{\mathrm{spa}}$
N_s	9	25	16
$\underline{\mu}_1$	2.6045e-12	6.8629e-05	6.1081e-05
μ_1	4.4179e-12	6.8629e-05	1.3489e-04
$\hat{B}_{s,\min}$	39	14	13
$B_{s,\min}$	33	11	11

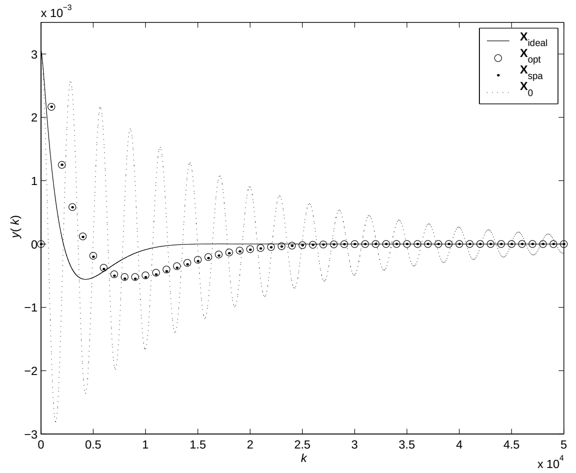

Figure 1. Comparison of unit impulse response of the infinite-precision controller implementation $\mathbf{X}_{\rm ideal}$ with those of the three 16-bit implemented controller realizations $\mathbf{X}_0$, $\mathbf{X}_{\rm opt}$ and $\mathbf{X}_{\rm spa}$.

and, moreover, it is sparse with 16 non-trivial parameters. Although this example only has a pair of complex eigenvalues, comparing with the results given in [8] confirms that the proposed μ_1 ($\underline{\mu_1}$ respectively) is less conservative in estimating the robustness of FWL closed-loop stability than the previous measure (its lower-bound respectively). We also computed the unit impulse response of the closed-loop control system when the controllers were the infinite-precision implemented $\mathbf{X}_0$ and 16-bit implemented three different controller realizations. Any realization $\mathbf{X} \in \mathcal{S}_C$ implemented in infinite precision will achieve the exact performance of the infinite-precision implemented $\mathbf{X}_0$, which is the *designed* controller performance. For this reason, the the infinite-precision implemented $\mathbf{X}_0$ is referred to as the *ideal* controller realization $\mathbf{X}_{\rm ideal}$. Fig. 1 compares the unit impulse response of the plant output $y(k)$ for the ideal controller $\mathbf{X}_{\rm ideal}$ with those of the 16-bit implemented $\mathbf{X}_0$, $\mathbf{X}_{\rm opt}$ and $\mathbf{X}_{\rm spa}$.

5. Conclusions

We have presented a design procedure for constructing sparse controller realizations with good FWL closed-loop stability characteristics, based on an improved stability measure. This new measure yields a more accurate estimate for the robustness of FWL closed-loop stability. An example confirms that the proposed design procedure produces computationally efficient controller structures suitable for FWL implementation in real-time applications.

References

[1] P. Moroney, A.S. Willsky and P.K. Houpt, The digital implementation of control compensators: the coefficient wordlength issue, *IEEE Trans. Automatic Control*, 25(8), 1980, 621–630.

[2] M. Gevers and G. Li, *Parameterizations in Control, Estimation and Filtering Problems: Accuracy Aspects* (London: Springer Verlag, 1993).

[3] I.J. Fialho and T.T. Georgiou, On stability and performance of sampled data systems subject to word length constraint, *IEEE Trans. Automatic Control*, 39(12), 1994, 2476–2481.

[4] A.G. Madievski, B.D.O. Anderson and M. Gevers, Optimum realizations of sampled data controllers for FWL sensitivity minimization, *Automatica*, 31(3), 1995, 367–379.

[5] G. Li, On the structure of digital controllers with finite word length consideration, *IEEE Trans. Automatic Control*, 43(5), 1998, 689–693.

[6] R.H. Istepanian, G. Li, J. Wu and J. Chu, Analysis of sensitivity measures of finite-precision digital controller structures with closed-loop stability bounds, *IEE Proc. Control Theory and Applications*, 145(5), 1998, 472–478.

[7] J. Wu, S. Chen, G. Li, R.H. Istepanian and J. Chu, An improved closed-loop stability related measure for finite-precision digital controller realizations, *IEEE Trans. Automatic Control*, 46(7), 2001, 1162–1166.

[8] J. Wu, S. Chen, G. Li and J. Chu, Digital finite-precision controller realizations with sparseness considerations, *Proc. 3rd Chinese World Cong. Intelligent Control and Intelligent Automation*, Hefei, China, 2000, 2869–2873.

[9] R.H. Istepanian, J. Wu and S. Chen, Sparse realizations of optimal finite-precision teleoperation controller structures, *Proc. ACC'2000*, Chicago, USA, 2000, 687–691.

[10] D.S.K. Chan, Constrained minimization of roundoff noise in fixed-point digital filters, *Proc. ICASSP'79*, 1979, 335–339.

[11] S. Chen and B.L. Luk, Adaptive simulated annealing for optimization in signal processing applications, *Signal Processing*, 79(1), 1999, 117–128.

[12] I. Njabeleke, R.F. Pannett, P.K. Chawdhry and C.R. Burrows, H_∞ control in fluid power, *IEE Colloquium Robust Control – Theory, Software and Applications*, London, U.K., 1997, 7/1–7/4.

[13] J.F. Whidborne, J. Wu and R.S.H. Istepanian, Finite word length stability issues in an l_1 framework, *Int. J. Control*, 73(2), 2000, 166–176.