Chapter 1

Stable controller coefficient perturbation in floating point implementation

Jun Wu
National Key Laboratory of Industrial Control Technology
Institute of Advanced Process Control
Zhejiang University
Hangzhou 310027
P. R. China
jwu@iipc.zju.edu.cn

Sheng Chen
Department of Electronics and Computer Science
University of Southampton, Highfield
Southampton SO17 1BJ
United Kingdom
sqc@ecs.soton.ac.uk

1.1 Description of the problem

For real matrix $\mathbf{X} = [x_{ij}]$, denote

$$\|\mathbf{X}\|_{\max} = \max_{i,j} |x_{ij}|. \tag{1.1}$$

For real matrices $\mathbf{X} = [x_{ij}]$ and $\mathbf{Y} = [y_{ij}]$ of the same dimension, denote the Hadamard product of $\mathbf{X}$ and $\mathbf{Y}$ as

$$\mathbf{X} \circ \mathbf{Y} = [x_{ij} y_{ij}]. \tag{1.2}$$

A square real matrix is said to be Schur stable if its eigenvalues are all in the interior of the unit disc.

Consider a stable discrete-time closed-loop control system, consisting of a linear time invariant plant $P(z)$ and a digital controller $C(z)$. The plant model $P(z)$ is assumed to be strictly proper with a state-space description

$$\begin{cases} \mathbf{x}_P(k+1) = \mathbf{A}_P\mathbf{x}_P(k) + \mathbf{B}_P\mathbf{u}(k) \\ \mathbf{y}(k) = \mathbf{C}_P\mathbf{x}_P(k) \end{cases} \tag{1.3}$$

where $\mathbf{A}_P \in \mathcal{R}^{m\times m}$, $\mathbf{B}_P \in \mathcal{R}^{m\times l}$ and $\mathbf{C}_P \in \mathcal{R}^{q\times m}$. The controller $C(z)$ is described by

$$\begin{cases} \mathbf{x}_C(k+1) = \mathbf{A}_C\mathbf{x}_C(k) + \mathbf{B}_C\mathbf{y}(k) \\ \mathbf{u}(k) = \mathbf{C}_C\mathbf{x}_C(k) + \mathbf{D}_C\mathbf{y}(k) \end{cases} \tag{1.4}$$

where $\mathbf{A}_C \in \mathcal{R}^{n\times n}$, $\mathbf{B}_C \in \mathcal{R}^{n\times q}$, $\mathbf{C}_C \in \mathcal{R}^{l\times n}$ and $\mathbf{D}_C \in \mathcal{R}^{l\times q}$. It can be shown easily that the transition matrix of the closed-loop system is

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_P + \mathbf{B}_P\mathbf{D}_C\mathbf{C}_P & \mathbf{B}_P\mathbf{C}_C \\ \mathbf{B}_C\mathbf{C}_P & \mathbf{A}_C \end{bmatrix} \in \mathcal{R}^{(m+n)\times(m+n)}. \tag{1.5}$$

It is well known that a discrete-time closed-loop system is stable if and only if its transition matrix is Schur stable. Since the closed-loop system, consisting of (1.3) and (1.4), is designed to be stable, $\mathbf{A}$ is Schur stable. Let

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}_P & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \in \mathcal{R}^{(m+n)\times(l+n)}, \tag{1.6}$$

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_P & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \in \mathcal{R}^{(q+n)\times(m+n)}, \tag{1.7}$$

$$\mathbf{W} = \begin{bmatrix} \mathbf{D}_C & \mathbf{C}_C \\ \mathbf{B}_C & \mathbf{A}_C \end{bmatrix} \in \mathcal{R}^{(l+n)\times(q+n)}, \tag{1.8}$$

where $\mathbf{0}$ and $\mathbf{I}$ denote the zero and identity matrices of appropriate dimensions, respectively. Define the set

$$\mathcal{S} = \{\mathbf{\Delta} : \mathbf{\Delta} \in \mathcal{R}^{(l+n)\times(q+n)}, \mathbf{A} + \mathbf{B}(\mathbf{W} \circ \mathbf{\Delta})\mathbf{C} \text{ is Schur stable}\} \tag{1.9}$$

and further define

$$\mu = \inf\{\|\mathbf{\Delta}\|_{\max} : \mathbf{\Delta} \in \mathcal{R}^{(l+n)\times(q+n)}, \mathbf{\Delta} \notin \mathcal{S}\}. \tag{1.10}$$

The open problem is: calculate the value of μ.

1.2 Motivation of the problem

The classical digital controller design methodology often assumes that the controller is implemented exactly, even though in reality a control law can only be realized with a digital processor of finite word length (FWL). It may seem that the uncertainty resulting from finite-precision computing of the digital controller is so small, compared to the uncertainty within the plant, such that this controller "uncertainty" can simply be ignored. Increasingly, however, researchers have realized that this is not necessarily the case. Due to the FWL effect, a casual controller implementation may degrade the designed closed-loop performance or even destabilize the designed stable closed-loop system, if the controller implementation structure is not carefully chosen [1, 2].

With decreasing in price and increasing in availability, the use of floating-point processors in controller implementations has increased dramatically. When a real number x is implemented in a floating-point format, it is perturbed to $x(1+\delta)$ with $|\delta| < \eta$, where η is the maximum round-off error of the floating-point representation [3]. It can be seen that the perturbation resulting from finite-precision floating-point arithmetic is multiplicative.

For the closed-loop system described in section 1.1, when $C(z)$ is implemented in finite-precision floating-point format, the controller realization $\mathbf{W}$ is perturbed to $\mathbf{W} + \mathbf{W} \circ \boldsymbol{\Delta}$. Each element of $\boldsymbol{\Delta}$ is bounded by $\pm\eta$, that is,

$$\|\boldsymbol{\Delta}\|_{\max} < \eta. \tag{1.11}$$

With the perturbation $\boldsymbol{\Delta}$, the transition matrix of the closed-loop system becomes $\mathbf{A} + \mathbf{B}(\mathbf{W} \circ \boldsymbol{\Delta})\mathbf{C}$. If an eigenvalue of $\mathbf{A} + \mathbf{B}(\mathbf{W} \circ \boldsymbol{\Delta})\mathbf{C}$ is outside the open unit disc, the closed-loop system, designed to be stable, becomes unstable with the FWL floating-point implemented $\mathbf{W}$.

It is therefore critical to know the ability of the closed-loop stability to tolerate the coefficient perturbation $\boldsymbol{\Delta}$ in $\mathbf{W}$ resulted from finite-precision implementation. This means that we would like to know the largest "cube" in the perturbation space, within which the closed-loop system remains stable. The measure μ defined in (1.10) gives the exact size of the largest "stable perturbation cube" for $\mathbf{W}$. If the value of μ can be computed, it becomes a simple matter to check whether $\mathbf{W}$ is "robust" to FWL errors, because $\mathbf{A}+\mathbf{B}(\mathbf{W}\circ\boldsymbol{\Delta})\mathbf{C}$ remains Schur stable when $\mu > \eta$.

Furthermore, $\mathbf{W}$ or $(\mathbf{A}_C, \mathbf{B}_C, \mathbf{C}_C, \mathbf{D}_C)$ is a realization of the controller $C(z)$. The realizations of $C(z)$ are not unique. Different realizations are all equivalent if they are implemented in infinite precision. In fact, if $(\mathbf{A}_C^0, \mathbf{B}_C^0, \mathbf{C}_C^0, \mathbf{D}_C^0)$ is a realization of $C(z)$, all the realizations of $C(z)$ form a set

$$\mathcal{S}_C = \left\{ \mathbf{W} : \mathbf{W} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{D}_C^0 & \mathbf{C}_C^0 \\ \mathbf{B}_C^0 & \mathbf{A}_C^0 \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{T} \end{bmatrix} \right\} \tag{1.12}$$

where the transformation matrix $\mathbf{T} \in \mathcal{R}^{n\times n}$ is an arbitrary non-singular matrix. A useful observation is that different $\mathbf{W}$ have different values of μ. Provided that the value of μ is computationally tractable, an optimal realization of $C(z)$, which has a maximum tolerance to FWL errors, can be obtained via optimization.

The open problem defined in section 1.1 was first seen in [3]. At present, there exists no available result. An approach to bypass the difficulty in computing μ is to define some approximate upper bound of μ using a first-order approximation, which is computationally tractable (see [3]).

Acknowledgements

The authors gratefully acknowledge the support of the United Kingdom Royal Society under a KC Wong fellowship (RL/ART/CN/XFI/KCW/11949).

Bibliography

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

[2] R.S.H. Istepanian and J.F. Whidborne, Eds.,*Digital Controller Implementation and Fragility: A Modern Perspective*. London: Springer Verlag, 2001.

[3] J. Wu, S. Chen, J.F. Whidborne and J. Chu, "Optimal realizations of floating-point implemented digital controllers with finite word length considerations," submitted to *Automatica*, 2001.
Available from http://www.bib.ecs.soton.ac.uk/people/sqc@ecs.soton.ac.uk