List of Contributors

Koldo Basterretxea
Department of Electronics and Telecommunications
University of the Basque Country
Plaza de la Casilla 3
48012 Bilbao, Vizcaya
Spain

Shankar P. Bhattacharyya
Department of Electrical Engineering
Texas A&M University
College Station, TX 77843
USA

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

Joseph R. Corrado
Raytheon Systems Co.
Tucson, AZ 85734-1337
USA
e-mail:
jcorrado@west.raytheon.com

Inés del Campo
Department of Electricity and Electronics
University Of The Basque Country
48940 Leioa, Vizcaya
Spain

Stéphane Dussy
EADS Launch Vehicles
66, route de Verneuil
78130 Les Mureaux
France

Ian J. Fialho
Dynacs Engineering - Boeing ISS
2100 Space Park Drive, MS HS44
Houston, TX 77058
USA

Tryphon T. Georgiou
Department of Electrical Engineering
University of Minnesota
Minneapolis, MN 55455
USA

Ryszard Gessing
Institute of Automatic Control
Silesian Technical University
Ul. Akademicka 16, 44-101 Gliwice
Poland

Wassim M. Haddad
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
USA
e-mail:
wm.haddad@aerospace.gatech.edu

Robert H. Istepanian
Department of Electronic and Computer Engineering
Brunel University
Uxbridge, Middlesex, UB8 3PH
United Kingdom
e-mail:
Robert.Istepanian@brunel.ac.uk

Lee H. Keel
Centre of Excellence in Information Systems
Tennessee State University
Nashville, TN 37203
USA

Gang Li
School of Electrical and Electronic Engineering
Nanyang Technological University
Nanyang Avenue, Singapore 639798
Singapore

Pertti M. Mäkilä
Automation and Control Institute
Tampere University of Technology
PO Box 692, FIN-33101 Tampere
Finland

Juha Paattilammi
Automation and Control Institute
Tampere University of Technology
PO Box 692, FIN-33101 Tampere
Finland

Yeng Chai Soh
School of Electrical and Electronic Engineering
Nanyang Technological University
Nanyang Avenue, Singapore 639798
Singapore
e-mail: eycsoh@ntu.edu.sg

José M. Tarela
Department of Electricity and Electronics
University Of The Basque Country
48940 Leioa, Vizcaya
Spain

James F. Whidborne
Department of Mechanical Engineering
King's College London
Strand, London WC2R 2LS
United Kingdom
e-mail:
james.whidborne@kcl.ac.uk

Darrell Williamson
Faculty of Engineering and Information Technology
The Australian National University
Canberra, ACT 0200
Australia
e-mail:
Darrell.Williamson@anu.edu.au

Jian Liang Wang
School of Electrical and Electronic Engineering
Nanyang Technological University
Nanyang Avenue, Singapore 639798
Singapore
e-mail: ejlwang@ntu.edu.sg

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

Guang-Hong Yang
School of Electrical and Electronic Engineering
Nanyang Technological University
Nanyang Avenue, Singapore 639798
Singapore
e-mail: egyang@ntu.edu.sg

8 The Determination of Optimal Finite Precision Controller Realizations Using a Global Optimization Strategy: a Pole-sensitivity Approach

Sheng Chen[1] and Jun Wu[2]

[1] Department of Electronics and Computer Science, University of Southampton, U.K.
[2] Institute of Advanced Process Control, Zhejiang University, Hangzhou, P.R. China

Abstract. The pole-sensitivity approach is a general method for analyzing the stability of the discrete-time control system with a finite wordlength (FWL) implemented digital controller. It leads to a non-smooth and non-convex optimization framework, where an optimal controller realization can be designed by maximizing some stability related measure. In this contribution, a new stability related measure is derived, which is more accurate in estimating the closed-loop stability robustness of an FWL implemented controller than the existing measures of pole-sensitivity analysis. This improved stability related measure provides a better criterion to find the optimal FWL realizations for a generic controller structure that includes output-feedback and observer-based controllers. An efficient global optimization strategy called the adaptive simulated annealing (ASA) is adopted to solve for the resulting optimization problem. A numerical example is included to verify the theoretical analysis and to illustrate the design procedure.

8.1 Introduction

The classical controller design methodology often assumes that the controller is implemented exactly, even though in reality a control law can only be realized in finite precision. The justification of this assumption is usually on the grounds that the plant uncertainty is the most significant source of uncertainty in the control system. However, researchers have realized that the controller uncertainty caused by finite-precision implementation has significant influence on the performance of the control system. A designed stable control system may achieve a lower than predicted performance or even become unstable when the control law is implemented with a finite-precision device due to the FWL effects. This is highlighted in the so-called fragility puzzles [1]–[3]: certain high-performance robust optimal controllers are known to be fragile. Ironically, these controllers have been designed to tolerate uncertainty in the plant, and yet small perturbations on the controller parameters may cause the designed closed-loop system to go unstable.

The fragility issues are strongly related to and interconnected with the FWL controller implementation issues. Although the number of controller implementations using floating-point processors is increasing due to their reduced price, for reasons of cost, simplicity, speed, memory space and power consumption, the use of fixed-point processors is more desired for many industrial and consumer applications. Furthermore, due to their reliability and well-understood properties, fixed-point processors predominate in safety-critical systems. With a fixed-point processor, however, the detrimental FWL effects are markedly increased due to a reduced precision. The problem can become serious when a high sampling rate and a high-order controller are used. It has been noted that a controller design can be implemented with different realizations and that the FWL effect on the closed-loop stability depends on the controller realization structure. This property can be utilized to select controller realization in order to improve the robustness of closed-loop stability under controller perturbations. Currently, two approaches exist for determining the optimal controller realizations under different criteria, namely pole-sensitivity measures [4]-[8] and complex stability radius measures [9],[10].

In the first approach, pole-sensitivity measures [5],[6] are used to quantify the FWL effect, leading to a non-convex and non-smooth optimization problem in finding an optimal FWL controller realization. The need to solve for a non-convex and non-smooth optimization problem had been seen as a disadvantage, as conventional optimization algorithms [11],[12], which are better known to the control community, may not guarantee to find a true optimal realization. However, the efficient global optimization techniques [13]-[18] to tackle this kind of difficult optimization problems are now widely available. Moreover, the pole-sensitivity approach is very general and can be applied to output-feedback and observer-based controllers as well as the controllers that are parameterized either by the usual shift operator or the delta operator [8], [19]–[21]. More recently, Fialho and Georgiou [10] used the complex stability radius measure to formulate an optimal FWL controller realization problem that can be represented as a special $\mathcal{H}_\infty$-norm minimization problem and solved for with the method of linear matrix inequality. In this second approach, the FWL perturbations are assumed to be complex-valued. Although this assumption is somewhat artificial and the approach can only be applied to shift-operator based output-feedback controllers, the method does not require to solve for a nonlinear optimization problem and has certain attractive features. For a detailed treatment of this approach, see Chapter .

This contribution focuses on the pole-sensitivity analysis method and emphasizes a unified approach for analyzing the sensitivity of closed-loop stability with respect to FWL effects. A generic digital controller structure is

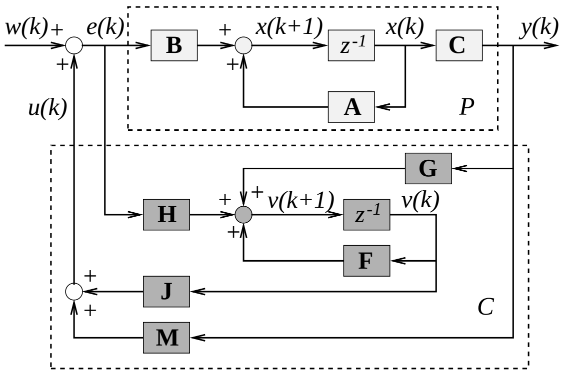

Fig. 8.1. Discrete-time closed-loop system with a generic digital controller.

considered that includes output-feedback and observer-based controllers, and a new stability related measure is proposed for the unified controller structure. An efficient global optimization procedure based on the ASA algorithm [16]–[18] is developed to find the optimal controller realization that maximizes this new measure. Through theoretical analysis and numerical results, it is shown that this improve measure is less conservative in estimating the FWL closed-loop stability robustness of a controller realization than the existing pole-sensitivity measures [5],[6].

8.2 Problem Formulation

Consider the discrete-time closed-loop control system shown in Fig. 8.1, where the linear time-invariant plant P is described by

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{e}(k) \\ \mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) \end{cases} \tag{8.1}$$

which is completely state controllable and observable with $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times p}$ and $\mathbf{C} \in \mathbb{R}^{q\times n}$; and the generic digital stabilizing controller C is described by

$$\begin{cases} \mathbf{v}(k+1) = \mathbf{F}\mathbf{v}(k) + \mathbf{G}\mathbf{y}(k) + \mathbf{H}\mathbf{e}(k) \\ \mathbf{u}(k) = \mathbf{J}\mathbf{v}(k) + \mathbf{M}\mathbf{y}(k) \end{cases} \tag{8.2}$$

with $\mathbf{F} \in \mathbb{R}^{m\times m}$, $\mathbf{G} \in \mathbb{R}^{m\times q}$, $\mathbf{J} \in \mathbb{R}^{p\times m}$, $\mathbf{M} \in \mathbb{R}^{p\times q}$ and $\mathbf{H} \in \mathbb{R}^{m\times p}$. The output-feedback and observer-based controllers can be unified in this general structure: C is an output-feedback controller when $\mathbf{H} = \mathbf{0}$; a full-order observer-based controller when $\mathbf{F} = \mathbf{A} - \mathbf{G}\mathbf{C}$, $\mathbf{M} = \mathbf{0}$ and $\mathbf{H} = \mathbf{B}$; a reduced-order observer-based controller, otherwise [22],[23]. Notice that, for notational simplicity, we have restricted to the controller structure with the shift operator z parameterization. All the results, however, can readily be extended to the controller structure with the delta operator parameterization [19]–[21].

Assume that a realization $(\mathbf{F}_0, \mathbf{G}_0, \mathbf{J}_0, \mathbf{M}_0, \mathbf{H}_0)$ of C has been designed. It is well-known that the realizations of C are not unique. All the realizations of C form the set:

$$\mathcal{S} = \{(\mathbf{F}, \mathbf{G}, \mathbf{J}, \mathbf{M}, \mathbf{H}) : \mathbf{F} = \mathbf{T}^{-1}\mathbf{F}_0\mathbf{T}, \mathbf{G} = \mathbf{T}^{-1}\mathbf{G}_0, \mathbf{J} = \mathbf{J}_0\mathbf{T}, \mathbf{M} = \mathbf{M}_0, \mathbf{H} = \mathbf{T}^{-1}\mathbf{H}_0\} \tag{8.3}$$

where $\mathbf{T} \in \mathbb{R}^{m\times m}$ is any real-valued non-singular matrix, called a similarity transformation. Let $\mathbf{w}_\mathbf{F} = \mathrm{Vec}(\mathbf{F})$, with $\mathrm{Vec}(\cdot)$ defining the column stacking operator. Denote

$$\mathbf{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_N \end{bmatrix} \triangleq \begin{bmatrix} \mathbf{w}_\mathbf{F} \\ \mathbf{w}_\mathbf{G} \\ \mathbf{w}_\mathbf{J} \\ \mathbf{w}_\mathbf{M} \\ \mathbf{w}_\mathbf{H} \end{bmatrix}, \quad \mathbf{w}_0 \triangleq \begin{bmatrix} \mathbf{w}_{\mathbf{F}_0} \\ \mathbf{w}_{\mathbf{G}_0} \\ \mathbf{w}_{\mathbf{J}_0} \\ \mathbf{w}_{\mathbf{M}_0} \\ \mathbf{w}_{\mathbf{H}_0} \end{bmatrix} \tag{8.4}$$

where $N = (m+p)(m+q) + mp$. We also refer to $\mathbf{w}$ as a realization of C. The stability of the closed-loop system in Fig. 8.1 depends on the eigenvalues of the matrix

$$\bar{\mathbf{A}}(\mathbf{w}) = \begin{bmatrix} \mathbf{A} + \mathbf{BMC} & \mathbf{BJ} \\ \mathbf{GC} + \mathbf{HMC} & \mathbf{F} + \mathbf{HJ} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^{-1} \end{bmatrix} \bar{\mathbf{A}}(\mathbf{w}_0) \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{T} \end{bmatrix} \tag{8.5}$$

All the different realizations $\mathbf{w}$ in $\mathcal{S}$ are completely equivalent and, in particular, achieve exactly the same set of closed-loop poles, if they are implemented with infinite precision. Since the closed-loop system will have been designed to be stable, the eigenvalues

$$|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))| = |\lambda_i(\bar{\mathbf{A}}(\mathbf{w}_0))| < 1, \ \forall i \in \{1, \dots, m+n\} \tag{8.6}$$

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

$$\|\Delta\mathbf{w}\|_\infty \triangleq \max_{i\in\{1,\cdots,N\}} |\Delta w_i| \le \epsilon/2 \tag{8.7}$$

For a fixed point processor of B_s bits, let $B_s = B_i + B_f$, where 2^{B_i} is a "normalization" factor to make the absolute value of each element of $2^{-B_i}\mathbf{w}$ 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 easily be seen that

$$\epsilon = 2^{-B_f} \tag{8.8}$$

With the perturbation $\Delta\mathbf{w}$, $\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))$ is moved to $\lambda_i(\bar{\mathbf{A}}(\mathbf{w} + \Delta\mathbf{w}))$. If an eigenvalue of $\bar{\mathbf{A}}(\mathbf{w} + \Delta\mathbf{w})$ is outside the open unit disk, the closed-loop system, designed to be stable, becomes unstable with B_s-bit implemented $\mathbf{w}$.

It is therefore critical to know when the FWL error will cause closed-loop instability. This ultimately means that we would like to know the largest open "sphere" in the controller perturbation space, within which closed-loop remains stable. The size or radius of this "sphere" is defined by:

$$\mu_0(\mathbf{w}) \triangleq \inf\{\|\Delta\mathbf{w}\|_\infty : \bar{\mathbf{A}}(\mathbf{w}+\Delta\mathbf{w}) \text{ is unstable}\} \tag{8.9}$$

From the definition of $\mu_0(\mathbf{w})$, it is obvious that $\bar{\mathbf{A}}(\mathbf{w}+\Delta\mathbf{w})$ remains stable for any $\Delta\mathbf{w}$ with $\|\Delta\mathbf{w}\|_\infty < \mu_0(\mathbf{w})$. The larger $\mu_0(\mathbf{w})$ is, the larger FWL error the closed-loop stability can tolerate. Hence, $\mu_0(\mathbf{w})$ constitutes a FWL stability measure.

Let $B_s^{\min}$ be the smallest wordlength, when used to implement $\mathbf{w}$, can guarantee the closed-loop stability. An estimate of $B_s^{\min}$ can be obtained as

$$\hat{B}_{s,0}^{\min} = B_i + \mathrm{Int}[-\log_2(\mu_0(\mathbf{w}))] - 1 \tag{8.10}$$

where the integer $\mathrm{Int}[x] \geq x$. It can easily be seen that the closed-loop system remains stable if $\mathbf{w}$ is implemented with a fixed-point processor of at least $\hat{B}_{s,0}^{\min}$. Moreover, $\mu_0(\mathbf{w})$ is a function of the controller realization $\mathbf{w}$, we could search for an optimal realization that maximizes $\mu_0(\mathbf{w})$. However, it is not known yet how to compute the value of $\mu_0(\mathbf{w})$ given a realization $\mathbf{w}$. A practical solution is to consider a lower bound of the stability measure $\mu_0(\mathbf{w})$ in some sense, which is computationally tractable. This in effect defines a smaller but known stable "sphere" or region in the $\Delta\mathbf{w}$ space. Obviously, the closer such a lower bound is to $\mu_0(\mathbf{w})$, the better. Two existing pole-sensitivity measures [5],[6] can both be regarded as such lower bounds and, hence, termed stability related measures. It should be emphasized that the approach based on the complex stability radius measure [10] is also "conservative" in that the region defined by the complex stability radius measure is generally smaller than that defined by $\mu_0(\mathbf{w})$.

8.3 A New Pole-sensitivity Stability Related Measure

Roughly speaking, how easily the FWL error $\Delta\mathbf{w}$ can cause a stable control system to become unstable is determined by how close $|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))|$ are to 1 and how sensitive they are to the controller parameter perturbations. We propose the following FWL stability related measure[1]

$$\mu_{1I}(\mathbf{w}) \triangleq \min_{i\in\{1,\cdots,m+n\}} \frac{1-|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))|}{\alpha_i(\mathbf{w})} \tag{8.11}$$

with

$$\alpha_i(\mathbf{w}) \triangleq \sum_{\mathbf{X}=\mathbf{F},\mathbf{G},\mathbf{J},\mathbf{M},\mathbf{H}} \left\| \frac{\partial |\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))|}{\partial \mathbf{X}} \right\|_1 \tag{8.12}$$

[1] This measure, as shown later, is an improved version of the existing measure μ_1 given in [6] and hence is denoted with μ_{1I}.

where, for a matrix $\mathbf{X} \in \mathbb{C}^{s \times v}$, the 1-norm $\|\mathbf{X}\|_1$ is defined as

$$\|\mathbf{X}\|_1 \triangleq \sum_{i=1}^{s} \sum_{j=1}^{v} |x_{ij}| \tag{8.13}$$

It remains to be shown how $\mu_{1I}(\mathbf{w})$ can be regarded as an FWL stability related measure, under what conditions $\mu_{1I}(\mathbf{w})$ is a lower bound of $\mu_0(\mathbf{w})$, and that $\mu_{1I}(\mathbf{w})$ is computationally tractable.

In practice, those controller perturbations $\Delta\mathbf{w}$ that will not cause closed-loop instability are most important. These $\Delta\mathbf{w}$ lie in the bound region:

$$\mathcal{Q}(\mathbf{w}) \triangleq \{\Delta\mathbf{w} : \|\Delta\mathbf{w}\|_\infty < \mu_0(\mathbf{w})\} \tag{8.14}$$

Define a perturbation subset to the controller realization $\mathbf{w}$ to be

$$\mathcal{P}(\mathbf{w}) \triangleq \{\Delta\mathbf{w} : \left|\lambda_i(\bar{A}(\mathbf{w} + \Delta\mathbf{w}))\right| - \left|\lambda_i(\bar{A}(\mathbf{w}))\right| \le \|\Delta\mathbf{w}\|_\infty \cdot \alpha_i(\mathbf{w}), \forall i\} \tag{8.15}$$

It is straightforward to prove the following proposition.

Proposition 8.1. $\bar{\mathbf{A}}(\mathbf{w} + \Delta\mathbf{w})$ is stable if $\Delta\mathbf{w} \in \mathcal{P}(\mathbf{w})$ and $\|\Delta\mathbf{w}\|_\infty < \mu_{1I}(\mathbf{w})$.

Thus, $\mu_{1I}(\mathbf{w})$ is a stability measure for $\Delta\mathbf{w} \in \mathcal{P}(\mathbf{w})$. The requirement for $\Delta\mathbf{w} \in \mathcal{P}(\mathbf{w})$ is not over restricted. Similar to the discussions in [19],[8], it can be proved that $\mathcal{P}(\mathbf{w})$ exists and at least a large part of $\mathcal{Q}(\mathbf{w})$ is covered by $\mathcal{P}(\mathbf{w})$. Defining

$$\rho\left(\mathcal{P}(\mathbf{w})\right) \triangleq \inf_{\Delta\mathbf{w} \notin \mathcal{P}(\mathbf{w})} \|\Delta\mathbf{w}\|_\infty \tag{8.16}$$

we have the following corollary, the proof of which is also straightforward.

Corollary 8.1. $\mu_{1I}(\mathbf{w}) \le \mu_0(\mathbf{w})$ if $\rho(\mathcal{P}(\mathbf{w})) > \mu_0(\mathbf{w})$.

It can be seen that $\mu_{1I}(\mathbf{w})$ is a lower bound of $\mu_0(\mathbf{w})$, provided that $\mu_0(\mathbf{w})$ is small enough. The assumption of small $\mu_0(\mathbf{w})$ is generally valid, especially for control systems with fast sampling. Given a controller realization $\mathbf{w}$, the value of $\mu_{1I}(\mathbf{w})$ can readily be calculated. This is summarized in the following theorem, the proof of which is given in Appendix A.

Theorem 8.1. Let $\mathbf{x}_i(\bar{\mathbf{A}}(\mathbf{w}))$ and $\mathbf{y}_i(\bar{\mathbf{A}}(\mathbf{w}))$ be the right and reciprocal left eigenvectors related to the $\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))$, respectively, and

$$\mathbf{L}_i(\mathbf{w}) = \frac{\mathrm{Re}\left[\lambda_i^*(\bar{\mathbf{A}}(\mathbf{w}))\mathbf{y}_i^*(\bar{\mathbf{A}}(\mathbf{w}))\mathbf{x}_i^T(\bar{\mathbf{A}}(\mathbf{w}))\right]}{\left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))\right|} \tag{8.17}$$

where * denotes the conjugate operator, ^{T} the transpose operator, and $\mathrm{Re}[\cdot]$ the real part. Then,

$$\frac{\partial\left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))\right|}{\partial\mathbf{F}} = [\,\mathbf{0} \quad \mathbf{I}\,]\,\mathbf{L}_i(\mathbf{w}) \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} \tag{8.18}$$

$$\frac{\partial\left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))\right|}{\partial \mathbf{G}} = [\,\mathbf{0} \quad \mathbf{I}\,]\,\mathbf{L}_i(\mathbf{w}) \begin{bmatrix} \mathbf{C}^T \\ \mathbf{0} \end{bmatrix} \tag{8.19}$$

$$\frac{\partial\left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))\right|}{\partial \mathbf{J}} = [\,\mathbf{B}^T \quad \mathbf{H}^T\,]\,\mathbf{L}_i(\mathbf{w}) \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} \tag{8.20}$$

$$\frac{\partial\left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))\right|}{\partial \mathbf{M}} = [\,\mathbf{B}^T \quad \mathbf{H}^T\,]\,\mathbf{L}_i(\mathbf{w}) \begin{bmatrix} \mathbf{C}^T \\ \mathbf{0} \end{bmatrix} \tag{8.21}$$

$$\frac{\partial\left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))\right|}{\partial \mathbf{H}} = [\,\mathbf{0} \quad \mathbf{I}\,]\,\mathbf{L}_i(\mathbf{w}) \begin{bmatrix} \mathbf{C}^T\mathbf{M}^T \\ \mathbf{J}^T \end{bmatrix} \tag{8.22}$$

Similar to (8.10), an estimate of $B_s^{\min}$ can be provided with $\mu_{1I}(\mathbf{w})$ by

$$\hat{B}_{s,1I}^{\min} = B_i + \mathrm{Int}[-\log_2(\mu_{1I}(\mathbf{w}))] - 1 \tag{8.23}$$

Provided that the conditions of Proposition 8.1 and Corollary 8.1 are met, $\hat{B}_{s,1I}^{\min} \geq \hat{B}_{s,0}^{\min} \geq B_s^{\min}$. That is, $\hat{B}_{s,1I}^{\min}$ is a conservative estimate of the minimum bit length, compared with $\hat{B}_{s,0}^{\min}$. Unlike $\hat{B}_{s,0}^{\min}$, however, $\hat{B}_{s,1I}^{\min}$ can be computed easily. We now show that $\mu_{1I}(\mathbf{w})$ is a closer lower bound of $\mu_0(\mathbf{w})$ than the two existing pole-sensitivity measures [5],[6], denoted as $\mu_2(\mathbf{w})$ and $\mu_1(\mathbf{w})$, respectively. Since it has been demonstrated [6] that $\mu_1(\mathbf{w})$ is a closer lower bound of $\mu_0(\mathbf{w})$ than $\mu_2(\mathbf{w})$, we only need to compare $\mu_{1I}(\mathbf{w})$ with $\mu_1(\mathbf{w})$. The stability related measure $\mu_1(\mathbf{w})$ is define as [6]:

$$\mu_1(\mathbf{w}) \triangleq \min_{i\in\{1,\cdots,m+n\}} \frac{1-\left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))\right|}{\beta_i(\mathbf{w})} \tag{8.24}$$

with

$$\beta_i(\mathbf{w}) \triangleq \sum_{\mathbf{X}=\mathbf{F},\mathbf{G},\mathbf{J},\mathbf{M},\mathbf{H}} \left\| \frac{\partial\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))}{\partial \mathbf{X}} \right\|_1 \tag{8.25}$$

An estimate of $B_s^{\min}$ is provided with $\mu_1(\mathbf{w})$ by

$$\hat{B}_{s,1}^{\min} = B_i + \mathrm{Int}[-\log_2(\mu_1(\mathbf{w}))] - 1 \tag{8.26}$$

The key difference between $\mu_{1I}(\mathbf{w})$ and $\mu_1(\mathbf{w})$ is that the former considers the sensitivity of $\left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))\right|$ while the latter considers the sensitivity of $\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))$. It is well known that the stability of a linear discrete-time system depends only on the moduli of its eigenvalues. As $\mu_1(\mathbf{w})$ includes the unnecessary eigenvalue arguments in consideration, it is expected that $\mu_1(\mathbf{w})$ is conservative in comparison with $\mu_{1I}(\mathbf{w})$. This can strictly be verified with the following. Noting

$$\frac{\partial\left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))\right|}{\partial w_j} = \mathrm{Re}\left[\lambda_i^*(\bar{\mathbf{A}}(\mathbf{w}))\frac{\partial\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))}{\partial w_j}\right] / \left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))\right| \tag{8.27}$$

gives rise to

$$\left| \frac{\partial \left| \lambda_i(\bar{\mathbf{A}}(\mathbf{w})) \right|}{\partial w_j} \right| \leq \frac{\left| \lambda_i^*(\bar{\mathbf{A}}(\mathbf{w})) \frac{\partial \lambda_i(\bar{\mathbf{A}}(\mathbf{w}))}{\partial w_j} \right|}{\left| \lambda_i(\bar{\mathbf{A}}(\mathbf{w})) \right|} = \left| \frac{\partial \lambda_i(\bar{\mathbf{A}}(\mathbf{w}))}{\partial w_j} \right| \tag{8.28}$$

which means that $\alpha_i(\mathbf{w}) \leq \beta_i(\mathbf{w})$. This leads to:

Theorem 8.2. $\mu_1(\mathbf{w}) \leq \mu_{1I}(\mathbf{w})$ and $\hat{B}_{s,1}^{\min} \geq \hat{B}_{s,1I}^{\min}$.

8.4 Optimization Procedure

When a controller is designed, it will have satisfied certain performance criteria and, in particular, ensures closed-loop stability. The design, however, is usually done under the infinite or at least high precision assumption. As actual implementation can only be finite precision, the real controller performance may be quite different from the designed one and, if the bit length is too small, the closed-loop stability may even be lost. Given a designed controller realization, denoted as $\mathbf{w}_0$, there are infinite many realizations $\mathbf{w}$ related to $\mathbf{w}_0$ by (8.3). All these realizations are completely equivalent under infinite precision implementation, but they may perform differently under FWL implementation. The problem naturally arisen is to find an "optimal" realization, denoted as $\mathbf{w}_{\text{opt}}$, such that $\mu_{1I}(\mathbf{w})$ is maximized. This is of practical importance, since the controller implemented with $\mathbf{w}_{\text{opt}}$ can tolerate a maximum FWL error. This optimal realization problem is formally defined as

$$\mathbf{w}_{\text{opt}} = \arg \max_{\mathbf{w} \in \mathcal{S}} \mu_{1I}(\mathbf{w}) \tag{8.29}$$

Given the design $\mathbf{w}_0$, $\forall i \in \{1, \cdots, m+n\}$, partition $\mathbf{x}_i(\bar{\mathbf{A}}(\mathbf{w}_0))$ and $\mathbf{y}_i(\bar{\mathbf{A}}(\mathbf{w}_0))$:

$$\mathbf{x}_i(\bar{\mathbf{A}}(\mathbf{w}_0)) = \begin{bmatrix} \mathbf{x}_{i,1}(\bar{\mathbf{A}}(\mathbf{w}_0)) \\ \mathbf{x}_{i,2}(\bar{\mathbf{A}}(\mathbf{w}_0)) \end{bmatrix} \tag{8.30}$$

$$\mathbf{y}_i(\bar{\mathbf{A}}(\mathbf{w}_0)) = \begin{bmatrix} \mathbf{y}_{i,1}(\bar{\mathbf{A}}(\mathbf{w}_0)) \\ \mathbf{y}_{i,2}(\bar{\mathbf{A}}(\mathbf{w}_0)) \end{bmatrix} \tag{8.31}$$

where $\mathbf{x}_{i,1}(\bar{\mathbf{A}}(\mathbf{w}_0)), \mathbf{y}_{i,1}(\bar{\mathbf{A}}(\mathbf{w}_0)) \in \mathbb{C}^n$ and $\mathbf{x}_{i,2}(\bar{\mathbf{A}}(\mathbf{w}_0)), \mathbf{y}_{i,2}(\bar{\mathbf{A}}(\mathbf{w}_0)) \in \mathbb{C}^m$. It is easily seen from (8.5) that

$$\mathbf{x}_i(\bar{\mathbf{A}}(\mathbf{w})) = \begin{bmatrix} \mathbf{x}_{i,1}(\bar{\mathbf{A}}(\mathbf{w}_0)) \\ \mathbf{T}^{-1}\mathbf{x}_{i,2}(\bar{\mathbf{A}}(\mathbf{w}_0)) \end{bmatrix} \tag{8.32}$$

$$\mathbf{y}_i(\bar{A}(\mathbf{w})) = \begin{bmatrix} \mathbf{y}_{i,1}(\bar{\mathbf{A}}(\mathbf{w}_0)) \\ \mathbf{T}^T\mathbf{y}_{i,2}(\bar{\mathbf{A}}(\mathbf{w}_0)) \end{bmatrix} \tag{8.33}$$

From (8.18)–(8.22), we have

$$\frac{\partial\left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))\right|}{\partial\mathbf{F}} = \mathbf{T}^T\mathbf{L}_{i,2,2}(\mathbf{w}_0)\mathbf{T}^{-T} \tag{8.34}$$

$$\frac{\partial\left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))\right|}{\partial\mathbf{G}} = \mathbf{T}^T\mathbf{L}_{i,2,1}(\mathbf{w}_0)\mathbf{C}^T \tag{8.35}$$

$$\frac{\partial\left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))\right|}{\partial\mathbf{J}} = \left(\mathbf{B}^T\mathbf{L}_{i,1,2}(\mathbf{w}_0) + \mathbf{H}_0^T\mathbf{L}_{i,2,2}(\mathbf{w}_0)\right)\mathbf{T}^{-T} \tag{8.36}$$

$$\frac{\partial\left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))\right|}{\partial\mathbf{M}} = \left(\mathbf{B}^T\mathbf{L}_{i,1,1}(\mathbf{w}_0) + \mathbf{H}_0^T\mathbf{L}_{i,2,1}(\mathbf{w}_0)\right)\mathbf{C}^T \tag{8.37}$$

$$\frac{\partial\left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}))\right|}{\partial\mathbf{H}} = \mathbf{T}^T\left(\mathbf{L}_{i,2,1}(\mathbf{w}_0)\mathbf{C}^T\mathbf{M}_0^T + \mathbf{L}_{i,2,2}(\mathbf{w}_0)\mathbf{J}_0^T\right) \tag{8.38}$$

where

$$\mathbf{L}_{i,j,l}(\mathbf{w}_0) = \frac{\mathrm{Re}\left[\lambda_i^*(\bar{\mathbf{A}}(\mathbf{w}_0))\mathbf{y}_{i,j}^*(\bar{\mathbf{A}}(\mathbf{w}_0))\mathbf{x}_{i,l}^T(\bar{\mathbf{A}}(\mathbf{w}_0))\right]}{\left|\lambda_i(\bar{\mathbf{A}}(\mathbf{w}_0))\right|}, \quad j,l = 1,2 \tag{8.39}$$

Define the following cost function:

$$f(\mathbf{T}) \triangleq \min_{i\in\{1,\cdots,m+n\}} \frac{1 - |\lambda_i(\bar{\mathbf{A}}(\mathbf{w}_0))|}{\alpha_i(\mathbf{w})} = \mu_{1I}(\mathbf{w}) \tag{8.40}$$

The optimal realization problem (8.29) can then be posed as the following optimization problem of finding an optimal similarity transformation matrix:

$$\mathbf{T}_{\mathrm{opt}} = \arg \max_{\substack{\mathbf{T}\in\mathbb{R}^{m\times m}\\ \det(\mathbf{T})\neq 0}} f(\mathbf{T}) \tag{8.41}$$

Although $f(\mathbf{T})$ is non-smooth and non-convex, efficient global optimization methods exist for solving for this kind of optimization problem. The ASA [17],[18] is such an algorithm and is adopted to search for a true global optimum $\mathbf{T}_{\mathrm{opt}}$ of the problem (8.41). The detailed ASA algorithm is provided in Appendix B. With $\mathbf{T}_{\mathrm{opt}}$, the optimal controller realization $\mathbf{w}_{\mathrm{opt}}$ can readily be obtained using the relationship (8.3).

8.5 A Numerical Example

A numerical example was used to illustrate the FWL optimal design procedure based on the pole-sensitivity approach. The plant model used was a modification of the plant studied in [5], which was a single-input single-output system. One more output, the first state in the original plant model, was added. The state-space model of this modified plant was given by

$$\mathbf{A} = \begin{bmatrix} 3.2439e-1 & -4.5451e+0 & -4.0535e+0 & -2.7003e-3 & 0 \\ 1.4518e-1 & 4.9477e-1 & -4.6945e-1 & -3.1274e-4 & 0 \\ 1.6814e-2 & 1.6491e-1 & 9.6681e-1 & -2.2114e-5 & 0 \\ 1.1889e-3 & 1.8209e-2 & 1.9829e-1 & 1.0000e+0 & 0 \\ 6.1301e-5 & 1.2609e-3 & 1.9930e-2 & 2.0000e-1 & 1 \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} 1.4518e-1 \\ 1.6814e-2 \\ 1.1889e-3 \\ 6.1301e-5 \\ 2.4979e-6 \end{bmatrix},$$

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & 1.6188e+0 & -1.5750e-1 & -4.3943e+1 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

The closed-loop poles as given in [5] were used in design, and the designed reduced-order observer-based controller obtained using a standard design procedure [23] had the form:

$$\mathbf{F}_0 = \begin{bmatrix} 0 & 1 \\ -9.3303e-1 & 1.9319e+0 \end{bmatrix},$$

$$\mathbf{G}_0 = \begin{bmatrix} 4.1814e-2 & 2.7132e+2 \\ 3.9090e-2 & 1.0167e+3 \end{bmatrix},$$

$$\mathbf{J}_0 = \begin{bmatrix} 3.0000e-4 & 5.0000e-4 \end{bmatrix},$$

$$\mathbf{M}_0 = \begin{bmatrix} 0 & 6.1250e-1 \end{bmatrix}, \quad \mathbf{H}_0 = \begin{bmatrix} 7.8047e+1 \\ 7.3849e+1 \end{bmatrix}$$

With this initial controller realization $\mathbf{w}_0$ and the plant model, the optimization problem (8.41) was formed and solved for, giving rise to the following optimal similarity transformation matrix:

$$\mathbf{T}_{\text{opt}} = \begin{bmatrix} 1.4714e+1 & 3.2071e+1 \\ 1.3588e+1 & 3.0531e+1 \end{bmatrix}$$

From $\mathbf{T}_{\text{opt}}$, the corresponding optimal controller realization $\mathbf{w}_{\text{opt}}$ was determined:

$$\mathbf{F}_{\text{opt}} = \begin{bmatrix} 9.8677e-1 & 1.4943e-2 \\ -2.9047e-2 & 9.4511e-1 \end{bmatrix},$$

$$\mathbf{G}_{\text{opt}} = \begin{bmatrix} 1.7066e-3 & -1.8080e+3 \\ 5.2084e-4 & 8.3794e+2 \end{bmatrix},$$

$$\mathbf{J}_{\text{opt}} = [\,1.1208e-2 \quad 2.4887e-2\,],$$

$$\mathbf{M}_{\text{opt}} = [\,0 \quad 6.1250e-1\,], \quad \mathbf{H}_{\text{opt}} = \begin{bmatrix} 1.0691e+0 \\ 1.9430e+0 \end{bmatrix}$$

For the initial and optimal controller realizations, the true minimal bit lengths $B_s^{\min}$ that can guarantee the closed-loop stability were also determined using a computer simulation method. Table 8.1 compares the values of the two stability related measures, corresponding estimated minimum bit lengths and true minimum bit lengths for the initial and optimal controller realizations. The results clearly show that the new measure μ_{1I} is much less conservative than the existing measure μ_1 in estimating the true minimum bit length.

Table 8.1. Comparison of the two stability related measures, corresponding estimated minimum bit lengths and true minimum bit lengths for the initial and optimal controller realizations.

realization	μ_{1I}	$\hat{B}_{s,1I}^{\min}$	μ_1	$\hat{B}_{s,1}^{\min}$	$B_s^{\min}$
$\mathbf{w}_0$	2.556877e-6	28	4.050854e-7	31	22
$\mathbf{w}_{\text{opt}}$	8.696940e-5	24	3.012354e-6	29	21

The unit impulse response of the closed-loop control system when the controllers were the infinite-precision implemented $\mathbf{w}_0$ and various FWL implemented realizations were also computed. Notice that any realization $\mathbf{w} \in \mathcal{S}$, implemented in infinite precision, will achieve the exact performance of the infinite-precision implemented $\mathbf{w}_0$, which is the *designed* controller performance. For this reason, the infinite-precision implemented $\mathbf{w}_0$ is referred to as the *ideal* controller realization $\mathbf{w}_{\text{ideal}}$. Fig. 8.2 compares the unit impulse response of the first plant output $y_1(k)$ for the ideal controller implementation $\mathbf{w}_{\text{ideal}}$ with those of 21-bit implemented realizations $\mathbf{w}_0$ and $\mathbf{w}_{\text{opt}}$. It can be seen that the closed-loop became unstable with a 21-bit implemented controller realization $\mathbf{w}_0$. However, the closed-loop system remained stable with the 21-bit implemented $\mathbf{w}_{\text{opt}}$.

8.6 Conclusions

The pole-sensitivity approach has been adopted to address the stability issue of the closed-loop discrete-time control system where a digital controller is implemented with a fixed-point processor. A new FWL closed-loop stability related measure has been derived, which is a less conservative lower bound of the computationally intractable true stability measure than other existing measures of the pole-sensitivity approach. As this new stability related measure is a function of the controller realization, it can be used as a cost

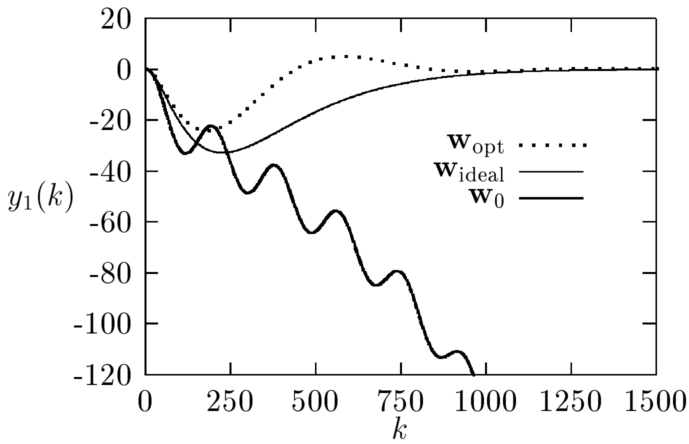

Fig. 8.2. Comparison of unit impulse response for the infinite-precision controller implementation $\mathbf{w}_{\text{ideal}}$ with those for the 21-bit implemented controller realizations $\mathbf{w}_0$ and $\mathbf{w}_{\text{opt}}$.

function for obtaining an optimal controller realization that maximizes the proposed measure. An efficient optimization strategy has been developed for optimizing a unified controller structure which includes output-feedback and observer-based controllers.

Acknowledgements

The authors wish to thank the support of the UK Royal Society under a KC Wong fellowship (RL/ART/CN/XFI/KCW/11949).

References

1. L.H. Keel and S.P. Bhattacharryya, "Robust, fragile, or optimal?" *IEEE Trans. Automatic Control*, Vol.42, No.8, pp.1098–1105, 1997.
2. P.M. Mäkilä, "Comments on 'Robust, fragile, or optimal?'," *IEEE Trans. Automatic Control*, Vol.43, No.9, pp.1265–1267, 1998.
3. P.M. Mäkilä, "Fragility and robustness puzzles," in *Proc. American Control Conf.* (San Diego, CA, USA), June 2-4, 1999, pp.2914–2919.
4. M. Gevers and G. Li, *Parameterizations in Control, Estimation and Filtering Problems: Accuracy Aspects.* London: Springer Verlag, 1993.
5. G. Li, "On the structure of digital controllers with finite word length consideration," *IEEE Trans. Automatic Control*, Vol.43, pp.689–693, 1998.
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*, Vol.145, No.5, pp.472–478, 1998.

7. S. Chen, J. Wu, R.H. Istepanian and J. Chu, "Optimizing stability bounds of finite-precision PID controller structures," *IEEE Trans. Automatic Control*, Vol.44, No.11, pp.2149–2153, 1999.
8. J. Wu, S. Chen, G. Li and J. Chu, "Optimal finite-precision state-estimate feedback controller realization of discrete-time systems," *IEEE Trans. Automatic Control*, Vol.45, No.7, July 2000.
9. I.J. Fialho and T.T. Georgiou, "On stability and performance of sampled data systems subject to word length constraint," *IEEE Trans. Automatic Control*, Vol.39, No.12, pp.2476–2481, 1994.
10. I.J. Fialho and T.T. Georgiou, "Optimal finite worldlength digital controller realization," in *Proc. American Control Conf.* (San Diego, CA, USA), June 2-4, 1999, pp.4326-4327.
11. G.S.G. Beveridge and R.S. Schechter, *Optimization: Theory and Practice.* McGraw-Hill, 1970.
12. L.C.W. Dixon, *Nonlinear Optimisation.* London: The English Universities Press Ltd, 1972.
13. D.E. Goldberg, *Genetic Algorithms in Search, Optimisation and Machine Learning.* Addison Wesley, 1989.
14. K.F. Man, K.S. Tang and S. Kwong, *Genetic Algorithms: Concepts and Design.* London: Springer-Verlag, 1998.
15. C.M. Fonseca and P.J. Fleming, "Multiobjective optimization and multiple constraint handling with evolutionary algorithms – Part I: A unified formulation," *IEEE Trans. Systems, Man, and Cybernetics Part A: Systems and Humans*, Vol.28, No.1, pp.26–37, 1998.
16. L. Ingber, "Simulated annealing: practice versus theory," *Mathematical and Computer Modelling*, Vol.18, No.11, pp.29–57, 1993.
17. L. Ingber, "Adaptive simulated annealing (ASA): lessons learned," *J. Control and Cybernetics*, Vol.25, No.1, pp.33-54, 1996.
18. S. Chen and B.L. Luk, "Adaptive simulated annealing for optimization in signal processing applications," *Signal Processing*, Vol.79, No.11, pp.117-128, 1999.
19. S. Chen, J. Wu, R.H. Istepanian, J. Chu and J.F. Whidborne, "Optimizing stability bounds of finite-precision controller structures for sampled-data systems in the delta operator domain," *IEE Proc. Control Theory and Applications*, Vol.146, No.6, pp.517–526, 1999.
20. S. Chen, R.H. Istepanian, J. Wu and J. Chu, "Comparative study on optimizing closed-loop stability bounds of finite-precision controller structures with shift and delta operators," *Systems and Control Letters*, Vol.40, No.3, pp.153–163, 2000.
21. J. Wu, S. Chen, G. Li, R.H. Istepanian and J. Chu, "Shift and delta operator realizations for digital controllers with finite-word-length considerations," *IEE Proc. Control Theory and Applications*, to appear, 2001.
22. T. Kailath, *Linear Systems.* Prentice-Hall, 1980.
23. J. O'Reilly, *Observers for Linear Systems.* London: Academic Press, 1983.

A Appendix - Theorem Proof

Proof. Let the real-valued square matrix $\bar{\mathbf{A}} = \mathbf{V}_0 + \mathbf{V}_1\mathbf{X}\mathbf{V}_2$ be diagonalisable, where all the matrices concerned are real-valued with proper dimensions, and $\mathbf{V}_0$, $\mathbf{V}_1$ and $\mathbf{V}_2$ are independent of $\mathbf{X}$. From Lemma 1 in [5],

$$\frac{\partial \lambda_i(\bar{\mathbf{A}})}{\partial \mathbf{X}} = \mathbf{V}_1^T \mathbf{y}_i^*(\bar{\mathbf{A}})\mathbf{x}_i^T(\bar{\mathbf{A}})\mathbf{V}_2^T \tag{8.42}$$

where $\lambda_i(\bar{\mathbf{A}})$ denotes the i-th eigenvalue of $\bar{\mathbf{A}}$, $\mathbf{x}_i(\bar{\mathbf{A}})$ and $\mathbf{y}_i(\bar{\mathbf{A}})$ the related right and reciprocal left eigenvectors, respectively. Noting

$$|\lambda_i(\bar{\mathbf{A}})| = \sqrt{\lambda_i^*(\bar{\mathbf{A}})\lambda_i(\bar{\mathbf{A}})} \tag{8.43}$$

leads to

$$\frac{\partial |\lambda_i(\bar{\mathbf{A}})|}{\partial \mathbf{X}} = \frac{1}{2\sqrt{\lambda_i^*(\bar{\mathbf{A}})\lambda_i(\bar{\mathbf{A}})}} \left(\frac{\partial \lambda_i^*(\bar{\mathbf{A}})}{\partial \mathbf{X}} \lambda_i(\bar{\mathbf{A}}) + \lambda_i^*(\bar{\mathbf{A}}) \frac{\partial \lambda_i(\bar{\mathbf{A}})}{\partial \mathbf{X}} \right)$$

$$= \frac{1}{2|\lambda_i(\bar{\mathbf{A}})|} \left(\left(\frac{\partial \lambda_i(\bar{\mathbf{A}})}{\partial \mathbf{X}} \right)^* \lambda_i(\bar{\mathbf{A}}) + \lambda_i^*(\bar{\mathbf{A}}) \frac{\partial \lambda_i(\bar{\mathbf{A}})}{\partial \mathbf{X}} \right)$$

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

$$= \frac{1}{|\lambda_i(\bar{\mathbf{A}})|} \mathbf{V}_1^T \mathrm{Re} \left[\lambda_i^*(\bar{\mathbf{A}}) \mathbf{y}_i^*(\bar{\mathbf{A}}) \mathbf{x}_i^T(\bar{\mathbf{A}}) \right] \mathbf{V}_2^T \tag{8.44}$$

The closed-loop system matrix (8.5) has the following equivalent forms:

$$\bar{\mathbf{A}}(\mathbf{w}) = \begin{bmatrix} \mathbf{A}+\mathbf{BMC} & \mathbf{BJ} \\ \mathbf{GC}+\mathbf{HMC} & \mathbf{HJ} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} \mathbf{F} [\, \mathbf{0} \quad \mathbf{I} \,] \tag{8.45}$$

$$\bar{\mathbf{A}}(\mathbf{w}) = \begin{bmatrix} \mathbf{A}+\mathbf{BMC} & \mathbf{BJ} \\ \mathbf{HMC} & \mathbf{F}+\mathbf{HJ} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} \mathbf{G} [\, \mathbf{C} \quad \mathbf{0} \,] \tag{8.46}$$

$$\bar{\mathbf{A}}(\mathbf{w}) = \begin{bmatrix} \mathbf{A}+\mathbf{BMC} & \mathbf{0} \\ \mathbf{GC}+\mathbf{HMC} & \mathbf{F} \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{H} \end{bmatrix} \mathbf{J} [\, \mathbf{0} \quad \mathbf{I} \,] \tag{8.47}$$

$$\bar{\mathbf{A}}(\mathbf{w}) = \begin{bmatrix} \mathbf{A} & \mathbf{BJ} \\ \mathbf{GC} & \mathbf{F}+\mathbf{HJ} \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{H} \end{bmatrix} \mathbf{M} [\, \mathbf{C} \quad \mathbf{0} \,] \tag{8.48}$$

$$\bar{\mathbf{A}}(\mathbf{w}) = \begin{bmatrix} \mathbf{A}+\mathbf{BMC} & \mathbf{BJ} \\ \mathbf{GC} & \mathbf{F} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} \mathbf{H} [\, \mathbf{MC} \quad \mathbf{J} \,] \tag{8.49}$$

Using (8.44) in (8.45)–(8.49) leads to (8.18)–(8.22). □

B Appendix - Adaptive Simulated Annealing

The ASA is a global optimization scheme for solving for the following general optimization problem:

$$\min_{\mathbf{x}\in\mathcal{X}} J(\mathbf{x}) \tag{8.50}$$

It evolves a single point $\mathbf{x} = [x_1 \cdots x_D]^T$ in the parameter or state space $\mathcal{X}$. The seemingly random search is guided by certain underlying probability distributions. Specifically, the general algorithm is described by three functions.

1. Generating probability density function

$$G(x_i^{\text{old}}, x_i^{\text{new}}, T_i; 1 \le i \le D) \tag{8.51}$$

This determines how a new state $\mathbf{x}^{\text{new}}$ is created, and from what neighbourhood and probability distributions it is generated, given the current state $\mathbf{x}^{\text{old}}$. The generating "temperatures" T_i describe the widths or scales of the generating distribution along each dimension x_i of the state space.

Often a cost function has different sensitivities along different dimensions of the state space. Ideally, the generating distribution used to search a steeper and more sensitive dimension should have a narrower width than that of the distribution used in searching a dimension less sensitive to change. The ASA adopts a so-called reannealing scheme to periodically re-scale T_i, so that they optimally adapt to the current status of the cost function. This is an important mechanism, which not only speeds up the search process but also makes the optimization process robust to different problems.

2. Acceptance function

$$P_{\text{accept}}(J(\mathbf{x}^{\text{old}}), J(\mathbf{x}^{\text{new}}), T_c) \tag{8.52}$$

This gives the probability of $\mathbf{x}^{\text{new}}$ being accepted. The acceptance temperature T_c determines the frequency of accepting new states of poorer quality.

Probability of acceptance is very high at very high temperature T_c, and it becomes smaller as T_c is reduced. At every acceptance temperature, there is a finite probability of accepting the new state. This produces occasionally uphill move, enables the algorithm to escape from local minima, and allows a more effective search of the state space to find a global minimum. The ASA also periodically adapts T_c to best suit the status of the cost function. This helps to improve convergence speed and robustness.

3. Reduce temperatures or annealing schedule

$$\left.\begin{array}{l} T_c(k_c) \longrightarrow T_c(k_c+1) \\ T_i(k_i) \longrightarrow T_i(k_i+1),\ 1 \le i \le D \end{array}\right\} \tag{8.53}$$

where k_c and k_i are some annealing time indexes. The reduction of temperatures should be sufficiently gradual in order to ensure that the algorithm finds a global minimum.

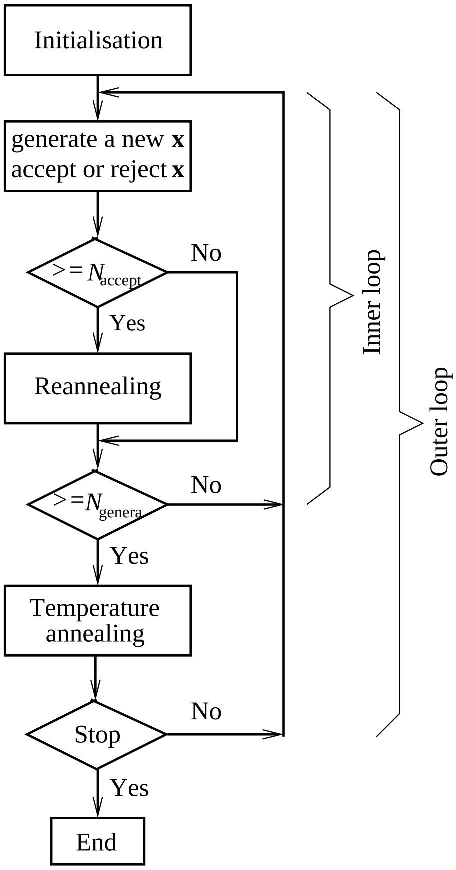

Fig. 8.3. Flow chart of the adaptive simulated annealing.

This mechanism is based on the observations of the physical annealing process. When the metal is cooled from a high temperature, if the cooling is sufficiently slow, the atoms line themselves up and form a crystal, which is the state of minimum energy in the system. The annealing process usually must converge very slowly to ensure a global optimum. The ASA, however, can employ a very fast annealing schedule, as it has self adaptation ability to re-scale temperatures.

An implementation of the ASA algorithm, shown in Fig. 8.3, is detailed as follows.

(i) Initialisation An initial $\mathbf{x}$ is randomly generated, the initial temperature of the acceptance probability function, $T_c(0)$, is set to the initial value of the cost function $J(\mathbf{x})$, and the initial temperatures of the parameter generating probability functions, $T_i(0)$, $1 \leq i \leq D$, are set to 1.0.

A control parameter c in annealing process is given, and the annealing times, k_i for $1 \le i \le D$ and k_c, are all set to 0.

(ii) Generating The algorithm generates a new point in the parameter space with:

$$x_i^{\text{new}} = x_i^{\text{old}} + g_i\,(U_i - V_i) \text{ and } x_i^{\text{new}} \in [U_i,\ V_i],\ 1 \le i \le D. \tag{8.54}$$

Here U_i and V_i are the lower and upper bounds for x_i, g_i is calculated as

$$g_i = \text{sgn}\left(u_i - \frac{1}{2}\right) T_i(k_i) \left(\left(1 + \frac{1}{T_i(k_i)}\right)^{|2u_i-1|} - 1\right), \tag{8.55}$$

and u_i a uniformly distributed random variable in $[0,\ 1]$. The value of the cost function $J(\mathbf{x}^{\text{new}})$ is then evaluated and the acceptance probability function of $\mathbf{x}^{\text{new}}$ is given by

$$P_{\text{accept}} = \frac{1}{1 + \exp\left(\left(J(\mathbf{x}^{\text{new}}) - J(\mathbf{x}^{\text{old}})\right)/T_c(k_c)\right)}. \tag{8.56}$$

A uniform random variable P_{unif} is generated in $[0,\ 1]$. If $P_{\text{unif}} \le P_{\text{accept}}$, $\mathbf{x}^{\text{new}}$ is accepted; otherwise it is rejected.

(iii) Reannealing After every N_{accept} acceptance points, calculating the sensitivities:

$$s_i = \left|\frac{J(\mathbf{x}^{\text{best}} + \mathbf{1}_i\,\delta) - J(\mathbf{x}^{\text{best}})}{\delta}\right|, \quad 1 \le i \le D, \tag{8.57}$$

where $\mathbf{x}^{\text{best}}$ is the best point found so far, δ is a small step size, the D-dimensional vector $\mathbf{1}_i$ has unit ith element and the rest of elements of $\mathbf{1}_i$ are all zeros. Let $s_{\max} = \max\{s_i,\ 1 \le i \le D\}$. Each T_i is scaled by a factor $s_{\max}/s_i$ and the annealing time k_i is reset:

$$T_i(k_i) = \frac{s_{\max}}{s_i} T_i(k_i),\ k_i = \left(-\frac{1}{c}\log\left(\frac{T_i(k_i)}{T_i(0)}\right)\right)^D. \tag{8.58}$$

Similarly, $T_c(0)$ is reset to the value of the last accepted cost function, $T_c(k_c)$ is reset to $J(\mathbf{x}^{\text{best}})$ and the annealing time k_c is rescaled accordingly:

$$k_c = \left(-\frac{1}{c}\log\left(\frac{T_c(k_c)}{T_c(0)}\right)\right)^D. \tag{8.59}$$

(iv) Annealing After every N_{genera} generated points, annealing takes place with

$$\left.\begin{aligned} k_i &= k_i + 1 \\ T_i(k_i) &= T_i(0)\exp\left(-c k_i^{\frac{1}{D}}\right) \end{aligned}\right\}\quad 1 \le i \le D \tag{8.60}$$

and

$$\left.\begin{array}{l} k_c = k_c + 1 \\ T_c(k_c) = T_c(0)\exp\left(-c k_c^{\frac{1}{D}}\right) \end{array}\right\} . \qquad (8.61)$$

Otherwise, goto step (ii).

(v) Termination The algorithm is terminated if the parameters has remained unchanged for a few successive reannealings or a preset maximum number of cost function evaluations has been reached; Otherwise, goto step (ii).

Index