A new approach for approximating the free transfer function in Q-parametrization.

Gustavo Sánchez and José Ferrer

Abstract— This paper presents a new approach for approximating the free transfer function in Q-parametrization. Its main advantage is that the number of parameters to be tuned during the design process is reduced, when compared to others methods. As a design example, the "two-mass and spring" benchmark control problem was solved, using GESA: an evolutionary non-linear optimization technique.

Index Terms — Q-parametrization. Benchmark control problem. Evolutionary optimization.

NOTATION

- $\mathfrak{C}_-$ / $\overline{\mathfrak{C}}_-$ ≡ open / closed left-half complex plan.
- $\mathfrak{C}_+$ / $\overline{\mathfrak{C}}_+$ ≡ open / closed right-half complex plan
- $RH_\infty^{p\times m}$: The set of matrix-valued real functions

$$\hat{G} : \overline{\mathfrak{C}}_+ \rightarrow \mathfrak{C}^{p\times m}$$

satisfying the following properties:

· $\hat{G}$ is analytic in $\overline{\mathfrak{C}}^+$

· For almost $\omega \in \mathcal{R}$,

$$\hat{G}(j\omega) = \lim_{\sigma\rightarrow 0^+} \hat{G}(\sigma + j\omega)$$

· $ess \sup_{s\in\overline{\mathfrak{C}}^+} \overline{\sigma}[\hat{G}(s)]$ is finite.

- $\underline{n}$: The set $\{1, 2, ..., n\}$

I. INTRODUCTION

Most practical control design problems can be solved using numerical optimization techniques [1]. For example, the convex optimization approach, in which all stabilizing controllers are parametrized in terms of the Youla parameter $Q \in RH_\infty$, has been now deeply studied [2], [3]. In this approach, Q is usually approximated by "basis functions in order to reduce the computational complexity" [4]. Let

$$Q \simeq \sum_{i=1}^{n_q} \nu_i Q_i \qquad (1)$$

where

- $Q_i \in RH_\infty$, $i \in \underline{n_q}$, are fixed stable maps.
- $\nu_i \in \mathcal{R}$, $i \in \underline{n_q}$, are called "decision variables".

This approach always finds a solution if one exists and if a suitable Q_i set has been selected. However, designers need choose the most suitable Q_i for each application.

This work was supported by the Universidad Simón Bolivar, Venezuela. Please send regular mail to: CCS:91537. PO.BOX 025323. MIAMI FL. 33102-5323. USA. E-mail: gsanchez@usb.ve

In this paper, a new approach for approximating the free transfer function in the Q-parametrization is proposed. This approach reduce the number of parameters during the interactive design procedure. Particularly, designers need not select a suitable Q_i set. The drawback is that, however, the optimization problem becomes harder to solve.

The exposition is organized as follows. The problem formulation is presented in section II. The proposed design method is described in section III. In section IV, it is applied to solve the "two-mass and spring" benchmark control problem, using GESA: an evolutionary non-linear optimization technique. Finally, some conclusions are given in section V.

II. PROBLEM FORMULATION

The integrated continuous-time closed-loop design model, including all frequency-dependent weights, is shown in figure1. There, G, K and Δ denote the given plant model, the controller and the uncertainty model respectively.

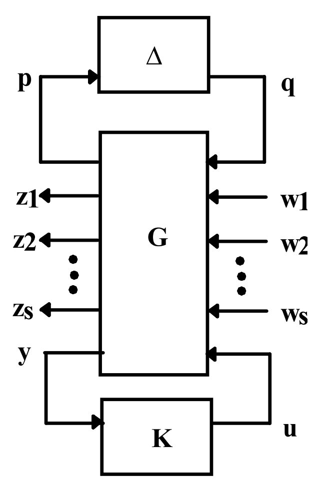

Fig. 1.- Integrated continuous-time design model

As usually, the vectors $w_1, w_2, ..., w_s$ denote the exogenous inputs (sensor noises, disturbances, references, etc.). The vectors $z_1, z_2, ..., z_s$ represent the outputs to be regulated. The vectors u and y represent the control inputs and the measured outputs respectively. It is assumed that G is real rational, proper, stabilizable from u and detectable from y. For each $k \in \underline{s}$, let $T_k(K)$ be a closed-loop transfer function from w_k to z_k (any interesting transfer function can be expressed in this way). Let

$$T_{s+1}(K) = T_{pq}(K) \tag{2}$$

be the transfer function from the uncertainty model output to its input, when Δ is removed from the model in figure 1. We state the general multiple objective control problem as follows:

$$\min_{\text{stabilizing } K} \max_{k\in\underline{s+1}} \left(\frac{\Phi_k[T_k(K)] - \varepsilon_k}{\varepsilon_k} \right)_{\varepsilon_k>0} \tag{3}$$

where $\Phi_k : RH_\infty \rightarrow \mathcal{R}$ are objective functionals and ε_k are design goals that represent the "largest tolerable values of the objective functions Φ_k" [5]. Particularly, the "robust stability" condition (small-gain theorem) can be stated as:

$$\Phi_{s+1} = \|T_{pq}(K)\|_\infty < 1 \tag{4}$$

As shown in [6] the set of all controllers that stabilize a given plant G is given parametrically as follows: K stabilizes G if and only if

$$K = (X_r - QN_l)^{-1}(Y_r - QD_l) \tag{5}$$

where,

$$Q, D_r, N_r, D_l, N_l, X_r, Y_r, X_l, Y_l \ \in RH_\infty \tag{6}$$

and

$$\begin{pmatrix} X_r & -Y_r \\ -N_l & D_l \end{pmatrix} \begin{pmatrix} D_r & Y_l \\ N_r & X_l \end{pmatrix} = I \tag{7}$$

In terms of the Q-parametrization, each transfer function T_k , $k \in \underline{s+1}$, becomes affine in Q, i.e,

$$T_k = T_{k1} + T_{k2}QT_{k3} \tag{8}$$

where T_{k1} , T_{k2}, and $T_{k3} \in RH_\infty$ can be obtained easily [6]. So, the above problem (3) reduces to:

$$\min_{Q\in RH_\infty} \max_{k\in\underline{s+1}} \left(\frac{\Phi_k[T_k(Q)] - \varepsilon_k}{\varepsilon_k} \right)_{\varepsilon_k>0} \tag{9}$$

III. DESIGN METHOD

Unlike the usual convex approach, we propose to construct Q by introducing the operator "vec", as follows. Let Θ, with $n_q, n_r, n_i \in \mathcal{Z}_0^+$, $n_{qu}, n_{qy} \in \mathcal{Z}^+$and

$$n_q = n_r + 2n_i \tag{10}$$

$$n_\theta = n_q(1 + n_{qu} + n_{yu}) + n_{qu} \times n_{qy} \tag{11}$$

be the set of $\theta \in \mathcal{R}^{n_\theta}$, such that:

1) $\theta_{i_{min}} \le \theta_i \le \theta_{i_{\max}}$, $i \in \underline{n_\theta}$, where $\theta_{i_{min}}, \theta_{i_{\max}}$ are bounds of the search space.

2) The vector

$$p^{n_q} = \begin{pmatrix} \theta_1 \\ \theta_2 \\ . \\ . \\ \theta_{n_r} \\ \theta_{n_r+1} + j\theta_{n_r+2} \\ \theta_{n_r+1} - j\theta_{n_r+2} \\ \theta_{n_r+3} + j\theta_{n_r+4} \\ \theta_{n_r+3} - j\theta_{n_r+4} \\ . \\ . \\ \theta_{n_r+2n_i-1} + j\theta_{n_r+2n_i} \\ \theta_{n_r+2n_i-1} - j\theta_{n_r+2n_i} \end{pmatrix} \tag{12}$$

satisfies,

- $p_i \in \mathfrak{C}^-$, $i \in \underline{n_q}$
- $Im(p_i) = 0$, $i \in \underline{n_r}$
- $\overline{p_i} = p_{i+1}$, for $i = n_r + 1, n_r + 3, ..., n_r + 2n_i - 1$

Then, let

$$vec : \theta \in \Theta \rightarrow RH_\infty^{n_{qu}\times n_{qy}} \tag{13}$$

be the operator such that:

$$\begin{aligned} vec(\theta) &= Q^{n_r,n_i} \in RH_\infty^{n_{qu}\times n_{qy}} \\ &= C_Q(sI - A_0 + B_0K_p)^{-1}B_Q + D_Q \end{aligned} \tag{14}$$

where,

- $\deg(Q^{n_r,n_i}) = n_r + 2n_i = n_q$
- $A_0 \in \mathcal{R}^{n_q\times n_q}, B_0 \in \mathcal{R}^{n_q\times n_{qu}}$ are fixed matrices such that

$$rank \begin{pmatrix} B_0 & A_0B_0 & \cdots & A_0^{n_q-1}B_0 \end{pmatrix} = n_q \tag{15}$$

- K_p is such that

$$p^{n_q} = eig(A_0 - B_0K_p) \tag{16}$$
$$= poles\left\{(sI - A_0 + B_0K_p)^{-1}\right\} \in \overline{\mathfrak{C}} \tag{17}$$

- $B_Q \in \mathcal{R}^{n_q\times n_{qu}}$, $C_Q \in \mathcal{R}^{n_{qy}\times n_q}$, $D_Q \in \mathcal{R}^{n_{qy}\times n_{qu}}$ are shown in Appendix B.

Resuming, the control problem (9) can now be expressed in terms of $\theta \in \Theta$, as follows:

$$\min_{\theta\in\Theta} \left\{ \max_{k\in\underline{s+1}} \left(\frac{\Phi_k[vec(\theta)] - \varepsilon_k}{\varepsilon_k} \right) \right\} \tag{18}$$

Notes:

- θ is formed by stacking p^{n_q}, the columns of B_Q, the rows of C_Q and the columns of D_Q into one single vector-column.
- When $n_q = 0$, $\Theta = \mathcal{R}^4$:

$$Q^{0,0} = D_Q = \begin{pmatrix} \theta_1 & \theta_3 \\ \theta_2 & \theta_4 \end{pmatrix} \tag{19}$$

- Designers just need select n_r, n_i, A_0 and B_0 (see Appendix A).

- Each $\theta \in \Theta$ represents a single $Q^{n_r,n_i} \in RH_\infty^{n_{qu}\times n_{qy}}$. However, each $Q^{n_r,n_i} \in RH_\infty^{n_{qu}\times n_{qy}}$ can be represented by infinite $\theta \in \Theta$.

IV. A DESIGN EXAMPLE

A. The "two-mass / spring" benchmark control problem

To illustrate the proposed method, the "two-mass and spring" benchmark control problem was tackled [7]. The regulated output is the position x_2 of body 2. Disturbance w_1 is a measurement noise ($y = x_2 + w_1$) and w_2 a disturbance force acting on body 2, the control input u acting on body 1 (see figure 2).

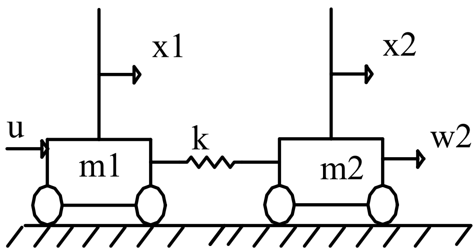

Fig.2.- Two mass and spring system.

Uncertain parameters are the spring constant and the two masses:

$$\begin{aligned} k &= k_0(1+\sigma_k\delta_k) \\ m_1 &= m_{10}(1+\sigma_{m_1}\delta_{m_1}) \\ m_2 &= m_{20}(1+\sigma_{m_2}\delta_{m_2}) \end{aligned} \quad (20)$$

with

$$k_0 = m_{10} = m_{20} = 1 \quad (21)$$

where $\sigma_k, \sigma_{m_1}, \sigma_{m_2}$ are measures of the relative uncertainty size. We consider

$$\Delta = \begin{pmatrix} \delta_k & 0 & 0 \\ 0 & \delta_{m_1} & 0 \\ 0 & 0 & \delta_{m_2} \end{pmatrix}$$

with

$$\begin{aligned} |\delta_k| &< 1 \\ |\delta_{m_1}| &< 1 \\ |\delta_{m_2}| &< 1 \end{aligned}$$

The integrated continuous-time open-loop design model (see figure 1), including all weights is described by the following state-space matrices:

$$\left(\begin{array}{c|c|c|c|c} A & B_p & B_{w_1} & B_{w_2} & B_u \\ \hline C_q & D_{qp} & D_{qw_1} & D_{qw_2} & D_{qu} \\ \hline C_z & D_{zp} & D_{zw_1} & D_{zw_2} & D_{zu} \\ \hline C_y & D_{yp} & D_{yw1} & D_{yw2} & D_{yu} \end{array}\right)$$

In fact, numerical values of these matrices can be found in [8]. Control requirements were selected as follows : Design a linear feedback controller such that the closed-loop system satisfies the following properties:

1) For a unit impulse in w_1 (measurement noise) and for the nominal system, x_2 satisfies

$$\Phi_1 = \|x_2(t)\|_2^2 \le 25$$

2) For a unit impulse in w_2 (disturbance exerted on body 2) and for the nominal system, x_2 has a 5% settling time of 15s

$$\Phi_2 = \max_{t>15} |x_2(t)| \le 0.05$$

3) For a unit impulse in w_2 (disturbance exerted on body 2) and for the nominal system, reasonable control effort is used

$$\Phi_3 = |u(t)| \le 10$$

4) The closed-loop is stable for

$$\begin{aligned} 0.8 &< k < 1.2 \\ 0.8 &< m_1 < 1.2 \\ 0.8 &< m_2 < 1.2 \end{aligned}$$

that is

$$\Phi_4 = \|T_{pq}\|_\infty < 1$$

with

$$\sigma_k = \sigma_{m_1} = \sigma_{m_2} = 0.2$$

Finally, the control problem can be expressed as follows:

$$\min_{\theta\in\Theta} \left\{ \max_{k\in\{1,2,3,4\}} \left(\frac{\Phi_k(\theta) - \varepsilon_k}{\varepsilon_k} \right) \right\} \quad (22)$$

with

$$\varepsilon = \begin{pmatrix} 25 \\ 0.05 \\ 10 \\ 1 \end{pmatrix}$$

B. An evolutionary optimization technique: Guided Evolutionary Simulated Annealing (GESA).

The "Guided Evolutionary Simulated Annealing" (GESA) is a scalar hybrid technique "involving both simulated evolution, genetic algorithms and simulated annealing" [9]. This algorithm has a mechanism "for measuring the qualities of many regions and automatically focuses the search onto regions with higher chances of attaining the optimum". There are two levels of competition. One is the local competition: the children in the same family compete with each other and only the one with the lowest objective value survives. The second level is the competition among families. The number of children to be generated in the next generation depends on the fitness of each family. An algorithm outline is shown next.

GESA Algorithm

- Set initial temperature T.
- Randomly select N initial parents.
- Generate children from the parents and find the best child for each parent.
- For each family, the best child is accepted as the parent for the next generation if

$$v_1 < v_2 \text{ or } exp(\frac{v_2 - v_1}{T}) < \rho$$

where:

v_1 is the best child objective value
v_2 is the parent objective value
$\rho \in [0, 1]$ is a random number

- Find the number of children that will be generated from the parents of the next generation.
- Decrease T and repeat Step 3 to Step 6 until an acceptable solution has been found or until a certain number of iterations has been reached.

For more information about how to generate children from parents and GESA pros and cons with respect to other optimization methods, see [9]. This technique was applied to solve problem (22) with $n_q = 6$, $n_r = 2, n_i = 2, n_{qu} = 1, n_{qy} = 1$,

$$A_0 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & -720 \\ 1 & 0 & 0 & 0 & 0 & -1764 \\ 0 & 1 & 0 & 0 & 0 & -1624 \\ 0 & 0 & 1 & 0 & 0 & -735 \\ 0 & 0 & 0 & 1 & 0 & -175 \\ 0 & 0 & 0 & 0 & 1 & -21 \end{pmatrix}$$

$$B_0 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

The algorithm was initialized at different random points, with initial temperature $T = 10^6$ (other temperature values were also tested) and 100 initial parents.

Just as an example, a particular solution θ^* was found after $\simeq 2000$ objective function evaluation, satisfying:

$$\begin{pmatrix} \Phi_1(\theta^*) \\ \Phi_2(\theta^*) \\ \Phi_3(\theta^*) \\ \Phi_4(\theta^*) \end{pmatrix} = \begin{pmatrix} 19.17 \\ 0.007 \\ 1.82 \\ 0.99 \end{pmatrix}$$

with values:

$$\theta^* = \begin{pmatrix} -5.903 \\ -5.621 \\ -0.609 \\ -0.351 \\ -5.460 \\ 2.568 \\ -132.75 \\ -11.226 \\ 16.624 \\ -11.456 \\ -0.514 \\ -0.324 \\ 1.789 \\ 0.035 \\ 8.414 \\ -8.402 \\ 175.3 \\ -1053.6 \\ -29.229 \end{pmatrix}$$

Figures 3 and 4 (next page) represent the closed-loop impulse responses with three different plant parameter values, using the designed controller. Note that robust stability is, in fact, achieved.

V. CONCLUSIONS AND FUTURE WORK

A simpler approach for approximating the free transfer function in Q-parametrization was proposed, although the problem convexity was sacrificed. Particularly, designers need not choose suitable function basis. However, the proposed parametrization depends on how many real and conjugate complex poles Q should have. This choice normally has a big influence on potential controller performance. Hence it is not so easy to select.

In other hand, although the dimension of the decision vector can be longer when compared to other methods, the problem can still be solved using conventional computation equipment.

An evolutionary algorithm, like GESA, was able to find "good" sub-optimal solutions, taking into account the method simplicity. Of course, much more examples are needed in order to get results that have some statistical significance. Our intention is just to show how the proposed design method can be implemented in practice.

At the present time, authors are developing a MATLAB toolbox that allows using other powerful optimization tools based on Computational Intelligence: Genetic Algorithms, Simulated Annealing, Fuzzy Logic, etc. Besides, they are developing a Human/Machine interface allowing evaluation of solutions, modifying search parameters, etc.

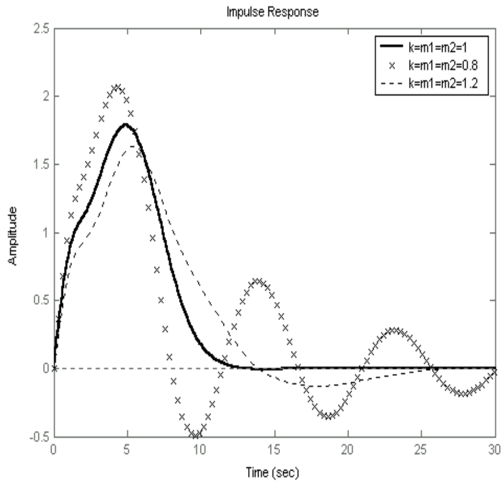

Fig. 3.- Output response x2(t) to a unit-impulse disturbance acting on body 2

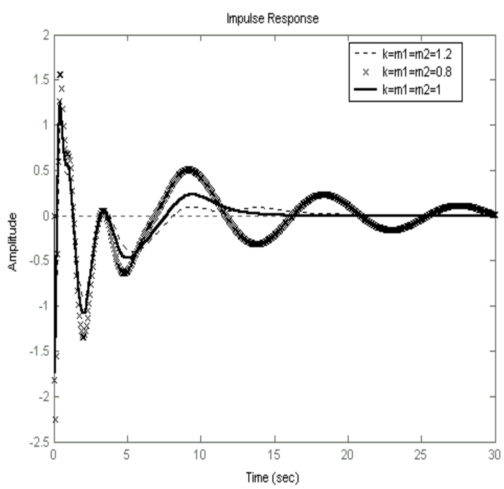

Fig 4.- Input command u(t) to a unit-impulse disturbance acting on body 2

References

[1] S. Boyd and C. Barrat, "Linear controller design: Limits of performance". Prentice Hall Inc, 1991.

[2] S. Boyd, V. Balakrishan, C. Barrat, N. Khraishi, X. Li, D. Meyer, and S. Norman, “"A new CAD method and associated architectures for linear controllers",” IEEE Transactions on Automatic Control, Vol AC-33, No.3, March, 1988.

[3] E. Polak and E. Salcudean, “"On the design of multivarable feedback systems via constrained nondifferentiable optimization in Hinf spaces",” IEEE Transactions on Automatic Control, Vol. 34, No.3, pp. 268–276, 1989.

[4] Z. Hu, E. Salcudean, and P. Loewen, “"Numerical solution of the multiple objective control system design problem for SISO systems".,” Proceedings of the American Control Conference ACC-95 Seattle, Washington, USA, pp. 1458,1462, June, 1995.

[5] J. Whidborne, “"Multi-objective robust control system design",” Proceedings of the 25th Nathiagali Summer School on Physics and Contemporary Needs, Pakistan., July 2000.

[6] M. Vidsayagar, “"Control systems synthesis: A factorization approach",” M.I.T Press, 1984.

[7] B. Wie and D. Bernstein, “Benchmark problems for robust control design,” American Institute of Aero. and Astro., vol. 15, no. 5, pp. 1057–1059, 1992.

[8] L. E. Ghaoui and J. Flocher, “Robust multiobjective lti control design for systems with structured perturbations,” Proceedings IEEE of the International Federation of Automatic Congress, 1996.

[9] P. Yip, The role of regional guidance in optimization: The Guided Evolutionary Simulated Annealing approach. PhD thesis, Department of Electrical Engineering and Applied Physics, Case Western Reserve University, 1993.

VI. APPENDIX A

In the scalar case, it is easy to construct a random controllable pair (A_0, B_0). For example, let

$$p = s^n + a_1 s^{n-1} + ... + a_{n-1}s + a_n$$

be a characteristic polynome with $roots(p) \in \mathfrak{C}_-, i \in \underline{n}$. Then, the companion pair (A_0, B_0) with

$$A_0 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & -a_n \\ 1 & 0 & 0 & 0 & 0 & -a_{n-1} \\ 0 & 1 & 0 & 0 & 0 & \bullet \\ 0 & 0 & 1 & 0 & 0 & \bullet \\ 0 & 0 & 0 & 1 & 0 & \bullet \\ 0 & 0 & 0 & 0 & 1 & -a_1 \end{pmatrix}$$

$$B_0 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

is fully controllable.

VII. APPENDIX B

$$B_Q = \begin{pmatrix} \theta_{n_q+1} & \theta_{2n_q+1} & \cdot & \cdot & \theta_{n_{qu}\times n_q+1} \\ \theta_{n_q+2} & \theta_{2n_q+2} & \cdot & \cdot & \theta_{n_{qu}\times n_q+2} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \theta_{2n_q} & \theta_{3n_q} & \cdot & \cdot & \theta_{n_{qu}\times n_q+n_q} \end{pmatrix}$$

$$C_Q = \begin{pmatrix} \theta_{(n_{qu}+1)\times n_q+1} & \cdot & \cdot & \theta_{(n_{qu}+1)\times n_q+n_q} \\ \theta_{(n_{qu}+2)\times n_q+1} & \cdot & \cdot & \theta_{(n_{qu}+2)\times n_q+n_q} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \theta_{(n_{qu}+n_{qy})\times n_q+1} & \cdot & \cdot & \theta_{(n_{qu}+n_{qy}+1)\times n_q} \end{pmatrix}$$

$$D_Q = \begin{pmatrix} \theta_{n_{d0}+1} & \cdot & \cdot & \theta_{n_{d0}+(n_{qu}-1)n_{qy}+1} \\ \theta_{n_{d0}+2} & \cdot & \cdot & \theta_{n_{d0}+(n_{qu}-1)n_{qy}+2} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \theta_{n_{d0}+n_{qy}} & \cdot & \cdot & \theta_{n_{d0}+n_{qy}\times n_{qu}} \end{pmatrix}$$

with $n_{d0} = (n_{qu} + n_{qy} + 1) \times n_q$.