World Journal of

Engineering

World Journal of Engineering 2 (2004)8 - 17

Automatic loop-shaping in quantitative feedback theory using genetic algorithms

Wenhua Chen[1], Yun Li[2]

1 Department of Aeronautical and Automotive Engineering Loughborough University, Loughborough, Leicestershire LE11 3TU
E-mail: w.chen@lboro.ac.uk

2 Centre for Systems and Control and Department of Electronics and Electrical Engineering, Engineering University of Glasgow, Glasgow G12 8QQ UK
E-mail: Y.Li@elec.gla.ac.uk

(Received 16 November 2004; accepted 6 December 2004)

Abstract

Design automation in Quantitative Feedback Theory (QFT) is addressed in this paper. An automatic loop shaping procedure based on Genetic Algorithms (GAs) is developed, where a robust controller for uncertain plants can be designed automatically such that the cost of feedback is minimised and all robust stability and performance specifications are satisfied. The developed approach can improve the current QFT design in at least two aspects. One is in the design of an initial controller for complicated plants, which might be difficult even to find a stabilising controller manually. The other is in proving the initial manual design by optimisation of the performance index under the prescribed requirements within the neighbourhood of the manual design. An illustrative example which compares manual loop shaping with automatic loop shaping is presented.

Key words: *Quantitative feedback theory*, *Design automation*, *Genetic algorithms*, *Stability*

1. Introduction

Quantitative Feedback Theory (QFT) has been applied in many engineering systems successfully since it was developed by Horowitz (Horowitz, 1973; Horowitz and Sidi, 1978). The basic idea in QFT is to convert design specifications of a closed-loop system and plant uncertainties into robust stability and performance bounds on the open-loop transmission of the nominal system and then to design a controller by using the gain-phase loop shaping technique.

The most important feature of QFT is that it is able to deal with the design problem of complicated uncertain plants. However, loop shaping is currently performed in computer aided design environments manually and it is usually a trial and error procedure (Chait, 1997). Whether or not the design is successful mainly depends on the experiences of the designer. Moreover for uncertain unstable and non-minimum phase plants, it is difficult to design a controller to satisfy all specifications (even a stabilising controller) manually. This also occurs for systems with a large number

ISSN: 1708-5284

of resonances, pure delays, etc. where a high order and complex controller is necessary. This issue was highlighted recently by the design campaign organised by European Space Agency (ESA), where H_∞ and QFT are chosen as candidate design methods for robust control of spacecraft with large flimsy appendages. It was shown that although QFT outperforms H_∞ methods in many aspects and QFT is chosen as the preferred design method for SISO (single-input-single-output) robust control design for future space missions, lack of design automation is one of the main obstacles for extending QFT for MIMO (Multi-Input-Multi-Output) systems, in particular for spacecraft with 6 degrees of freedom (Bodineau *et al.*, 2004)

To solve this problem and to unleash the power of QFT, design automation will be studied in this paper. It can improve the design of QFT controllers in at least two aspects. The first is in generating an initial controller on which the manual loop-shaping is based. This is necessary for unstable and non-minimum phase plants or plants with complicated characteristics where it might be difficult to find a stabilising controller. The designer starts from this initial controller and designs a final controller according to his experiences and design requirements, e.g., the controller order, robustness, and the sensitivity to the sensor noise, etc.. The second is that after an initial controller is designed manually, the optimisation may be used to maximise the performance/minimise the cost of feedback under the prescribed requirements within the neighbourhood of the manual design. The work of Thompson and Nwokah (1994) belongs to this case. Combining these aspects, complete design automation from scratch may be achieved.

Several researchers have investigated the design automation of QFT. Horowitz and Gera (1980) and Ballance and Gawthrop (1991) both proposed the use of Bode integrals within an iterative approach to loop shaping. Thompson and Nwokah (1994) proposed an analytical approach to loop shaping when the templates are approximated by boxes. A linear programming approach for automatic loop shaping was proposed by Bryant and Halikias (1995). The most recent results about the design automation in QFT are given by Chait (1997) where, to overcome the non-convexity of the bounds on the open-loop transmission, the design is based on the closed-loop bounds. The main disadvantage of these approaches is attempting to solve a complicated nonlinear optimisation with a convex or linear programming. This results in imposing unrealistic assumptions or very conservative design. For example, the denominator of the closed-loop transfer function must be specified in advance in the approach of Chait (1997).

In view of the non-convexity of the loop shaping problem in QFT design, this paper addresses the design automation problem in QFT by the use of Genetic Algorithms (GAs), which are multi-objective global optimisation techniques. The design objective is to design a controller for the uncertain plants such that the cost of feedback is minimised and all robust stability and performance specifications are satisfied.

This paper is organised as follows. First the QFT and Genetic Algorithms are briefly introduced in Section 2. In Section 3, the procedure of automatic loop shaping is developed. In converting the QFT problem into the evaluating computing formulation, several problems, including how to ensure the nominal case is stabilised, how to avoid the right half plane zero/pole cancellations, and how to ensure the resultant controller is rational, are answered. The design method developed is illustrated by a benchmark example in QFT Toolbox in Section 4 and we close the paper with a brief conclusion in Section 5.

2. QFT method and Genetic Algorithms

2.1. *QFT overview*

The two-degree-freedom feedback system configuration of QFT is given in Figure 1.

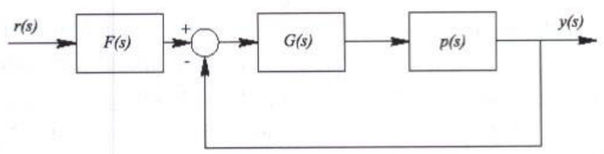

Fig. 1. Feedback control system Configuration for QFT.

where $G(s)$ and $F(s)$ are referred to as the controller and the prefilter respectively. $P(s)$ denotes the uncertain plant, which belongs to a given plant family P. It can contain structured, unstructured or mixed uncertainties.

One view of the QFT approach is if there is no uncertainties and noise, the feedback is unnecessary and we can achieve the prescribed performance specifications by the prefilter $F(s)$, which can be designed via open-loop shaping. The main role of the controller, $G(s)$, *is therefore to reduce uncertainties and disturbances by using feedback. The QFT design is thereby divided into two steps. The first step is to design the controller*, $G(s)$, such that uncertainties and noise on the closed-loop system is reduced to an acceptable level which is determined by the closed-loop robust stability and performance specifications. The prefilter is then designed to achieve the desired frequency responses.

In general three kinds of specifications are considered in QFT:

1. Robust Stability Margin

$$\left|\frac{L(j\omega)}{1+L(j\omega)}\right| \le \gamma \text{ for all } P(s)\in P \quad \omega\in[0,\infty) \tag{1}$$

2. *Tracking Performance*

$$|a(w)| \le \left|\frac{F(j\omega)L(j\omega)}{1+L(j\omega)}\right| \le |b(\omega)| \text{ for all } P(s)\in P \quad \omega\in[0,\infty) \tag{2}$$

3. Disturbance Attenuation Performance

$$\left|\frac{P(j\omega)}{1+L(j\omega)}\right| \le |d(\omega)| \text{ for all } P(s)\in P \quad \omega\in[0,\infty) \tag{3}$$

where γ, $a(\omega)$, $b(\omega)$ and $d(\omega)$ are stability performance specifications, which are robust stability margin, the low and upper tracking performance bounds and disturbance attenuation bounds respectively. The open loop transmission is defined as

$$L(s) = P(s)G(s)。$$

2.2. QFT design procedure

1. Generating templates. For a given uncertain plant $P(s) \in P$, select a series of frequency points, ω_i, $i = 1, \cdots, l$, according to the plant characteristics and the specifications. Calculate the value sets of the plant $p(j\omega_i)$ in the complex plane, i.e., the so called plant templates, at all frequency point ω_i;

2. Computation of QFT bounds. An arbitrary member in the plant set is chosen as the nominal case. At each selected frequency point, combining the stability and performance specifications with the plant template yields stability margin and performance bounds in term of the nominal case. Intersection of all such bounds, i.e., the worst case bound, at the same frequency point yields a single QFT bound. Compute such a QFT bound for all frequency points ω_i, $i = 1, \cdots, l$. Hence the specifications of the closed-loop systems for all $P(s) \in P$ are translated into that of the open loop nominal case;

3. Loop shaping for QFT controllers. The design of the QFT controller, $G(s)$, is accomplished on the Nichols Chart. The phase gain loop shaping technique is employed to design of the controllers, $G(s)$, until the QFT bounds at all-frequencies are satisfied and the closed-loop nominal system is stable;

4. Design of prefilters $F(s)$. The final step in QFT is to design the prefilters, $F(s)$, such that the performance specifications are satisfied.

In general the first two steps can be carried out by numerical evaluations on computer. For a large class of systems with nonlinear uncertainties, a systematic method for generating the plant templates and its symbolic computation procedure has been developed by Chen and Ballance (Chen and Ballance, 1998a; Ballance and Chen, 1998). The main difficulty in QFT design procedure lies in Step 3 in Section 2.1, it is normally performed manually with the help of computer aided design (CAD) environments, e.g., the QFT Toolbox for MATLAB (Borghesani *et al.*, 1995). The main advantages of this method are that the design experiences can be used and the design procedure is transparent to the designer. The designer can consider many factors, which might be difficult to represent by analytical expressions or quantitatively. However, as pointed out in Section 1, when the plant has unstable zero/pole or complicated characteristics, it may be difficult to design a stabilising controller manually. In addition, whether or not the design is successful mainly depends on designer's experience applied to the trials. If the controllers, which ensure all QFT bounds are satisfied, have been worked out in Step 3, it is quite easy to design the prefilters by manual loop shaping. Thus, only the QFT controller design, Step 3, is considered in this paper.

2.3. *Genetic algorithms*

Clearly, the manual loop-shaping procedure may be replaced, and thus automated, by a computerised optimisation procedure. However, such a nonlinear, non-convex multi-objective optimisation problem can hardly be solved using conventional, gradient-based techniques, as these suffer from the following deficiencies:

1. The objective function may not be well-behaved and a smooth derivative may not exist;

2. Direct domain constraints (such as parameter range requirements and fixed relationships) and indirect inequalities (such as control constraints) often exist, which impose difficulties in synthesising an appropriate objective func tion;

3. Even when an appropriate objective function exists and is well-behaved, gradient guidance can usually leads to a local optimum and is noise prone;

4. Conventional techniques can only deal with one, or one composite, objective; and

5. It is difficult to make use of engineers' existing knowledge on certain parameters for a globally optimal solution.

One approach to this problem could be exhaustive search. Suppose that, therefore, 10 parameters need to be tuned, each of which has 10 candidate values to evaluate. This requires repeatedly running a CAD toolbox 10^{10} times. In practice, however, such a process could take 4 months to complete, if each evaluation on the CAD simulator takes 1 mini-second! This approach would thus be impractical.

The most powerful computational intelligence technique, evolutionary computation, mimics the human intelligent trial-and-error procedure in an emulated 'survival-of-the-fittest' evolutionary process (Li *et al.*, 1996; Li and Haeussler, 1996). In contrast to conventional optimisation and search algorithms, a GA uses multiple points (in the form of a 'population' of candidate solutions) to conduct parallel search every time. It thus offers a better opportunity to arrive at the global optimum and to deal with multiple objectives. Note that a constrained simulation problem is much easier to solve than a constrained synthesis problem. As long as the CAD simulator works, candidate designs can be evaluated and selected. A GA incorporates biased selection and replication mechanisms using the 'survival- of-the-fittest' Darwinian principle. It thus guides the search in a trial-and-error based a-posteriori process, requiring no gradient information. After replicating better performing candidates, the GA then diverges the search using an operation called 'crossover' by exchanging coordinates or parameters among surviving candidates. It also diverges the search by altering some parameter values in an operation called 'mutation'. This way, a new 'generation' of candidate designs will be formed and the emulated evolutionary cycle continues until no meaningful improvements in the design may be found.

GAs offer an exponentially increased effficiency compared with exhaustive search. The GA paradigm has been successfully applied to solve many control engineering problems (Li *et al.*, 1996; Li and Haeussler, 1996). A GA can recommend several top-performing candidates. Multiple solutions to multiple design objectives provide sensitivity and reliability information on the designs and also provide design transparency for 'minimum commitment' at the CAD stage. Thus the application of an optimisation procedure is not intended to replace an engineer's job. It is to assist the engineer to arrive at a preliminary design quickly and to assess whether an existing design may be improved based on simulation results.

3. Automated QFT design

3.1. Problem formulation

The design automation problem considered in this paper can be stated as, given the QFT bounds and the nominal plant, to develop a controller automatically satisfied and the given performance index is minimised.

A good automatic design procedure, we believe, should be flexible and transparent to the designer. That is the designer should know how to control the optimisation process to achieve the specific requirements for a problem in hand by adjusting the parameters provided by the optimisation procedure, for example, the order of the controller, whether or not an integral is included, etc.

3.2. Choice of optimisation variables

One way to optimal QFT design is to use the Bode integrals, which enforces the analytical relationship between the gain and phase for minimum and stable plants and can be easily extended to non-minimum phase and unstable cases.

This approach is inspired by the original work of Horowitz who shows the optimal QFT design is achieved when the open-loop transmission lies on the corresponding QFT bound at each frequency (Horowitz, 1973). However in general it gives irrational controllers. Using Bode Integrals in automatic QFT design produces an approximated frequency response of the optimal controllers and requires rational function approximation (Horowitz and Gera, 1980; Ballance and Gawthrop, 1991). The disadvantages of this approach are:

1. control by the designer over the order of the controller is lost;

2. unstable poles and non-minimum phase zeros cancellation between the controller and the plant may occur;

3. it requires rational function approximation.

The second disadvantage means that although by appropriately chosen Bode Integrals it can be ensured that the number of the unstable poles of the resultant open-loop system are the same as that of the nominal plant, it does not imply that there are no right-half plane pole/zero cancellations between the controller and the plant. That is, it is possible that the controller cancels an unstable pole in the nominal plant by a non-minimum phase zero and introduces another new unstable pole in the resultant open-loop transmission.

The automatic design is based on the open-loop transmission, $L(s)$, in most of the QFT methods (e.g., see Bryant and Halikias (1995) and Chait (1997)). The use of the open-loop naturally follows because the QFT bounds are given in term of the open-loop transmission. All existing optimal design methods in QFT are based on the open-loop transmission $L(s)$ and thereby the controller order is equal to the plant order plus the resultant open-loop transmission order. In contrast, in this paper the controller $G(s)$ is designed directly. Direct design of $G(s)$ is one of the main features of this paper, and, as shown in later, greatly simplifies the optimisation problem. The order of the controller can be prespecified, which enables a low order controller to be designed. In this paper, the optimisation variables are the coefficients of the controller's transfer function

$$G(s) = \frac{brs^r + \cdots + b_1 s + b_0}{a_m s^m + \cdots + a_1 s + a_0} \qquad (4)$$

Other advantages of this approach are that it gives control over the control struc ture to designers, e.g., whether or not an integral is included, the relative order in the controller. In general a_m is set equal to 1.

3.3. Stability of the nominal case

According to the Zero Inclusion Theorem (for example see Jayasuriya and Zhao (1994)), the sufficient and necessary condition for the robust stability of the closed-loop systems is that the nominal system is stable and the open-loop transmission under the prescribed plant set does not intersect the $-1+j0$ point in the complex plane. The later is guaranteed in QFT by the robust stability margin condition (1).

It is easy to check the stability of systems in the Nichols Chart since the stability criteria in the Nyquist plot is extended to the Nichols Chart (Cohen *et al*., 1992). However it is a graphical criteria and not easily tested numerically. In our method, the stability of the closed-loop nominal system is simply tested by solving the roots of the characteristic polynomial. The function

$$J_{sta} = \begin{cases} 0 & \text{if stable} \\ 1 & \text{if unstable} \end{cases} \quad (5)$$

is defined as the stability index.

3.4. Right half plane pole/zero cancellation

In order to ensure internal stability, it is desired that a minimum phase and stable controller is designed. This can guarantee the internal stability and no unstable pole and non-minimum phase zero cancellations. Our method is to limit all coefficients of the transfer function $G(s)$ to be positive. This is a necessary condition for a polynomial having negative real part roots. Certainly we can calculate the poles and the zeros of the controllers and avoid right half pole/zero cancellation by comparing them with all right-half plane poles/zeros (if any) of the nominal case. Another way is to use the Horowitz method for QFT design of unstable and non-minimum phase plants, i.e., translation of QFT bounds for an unstable/non-minimum phase nominal plant to that for a stable and minimum phase plant (Horowitz and Sidi, 1978; Horowitz, 1992; Chen and Ballance, 1998b). Then it is impossible to have right half plane pole/zero cancellations since the new nominal plant is stable and minimum phase.

3.5. QFT bounds

In general the QFT bounds are very complicated and are non-convex (Horowitz, 1963). It is difficult to give analytical expressions of the QFT bounds except for some special templates (Thompson and Nwokah, 1994). In our approach, the QFT bounds are generated by the computation procedure in the QFT Toolbox and numerical bounds are yielded. The properties of GAs allows these numerical bounds to be used directly in our design. At each frequency, the gain and phase of the open loop transmission $L(j)$ is calculated and then checked to see whether or not the QFT bound at this frequency is satisfied by interplotation. A bound index is defined by

$$J_{bi} = \begin{cases} 0 & \text{if the QFT bound at } \omega_i \text{ is satisfied} \\ 1 & \text{otherwise} \end{cases}$$
$$i = 1, \cdots, l \quad (6)$$

A Universal High-frequency Bound (UHB) is widely used in QFT. There are several interpretations about UHB including the robust stability margin, the maximum overshoot and disturbance attenuation (Ballance, 1992; Ballance and Gawthrop, 1992). To ensure the open-loop transmission does not intersect the UHB, a number of frequencies near or greater than the largest frequency in ω_i, $i = l+1, \ldots, h$, are added. The gain and phase of the open loop transmission $(j\omega_i)$, $i = l+1, \ldots, h$, are computed and the UHB is tested at those frequency points. This is not to add much computational burden since no new QFT bounds need to be calculated.

3.6. Performance index

Among all loop transmissions, $L(j\omega_i)$, which satisfy the QFT bounds and the stability requirement of the nominal system, the optimum is taken to be any $L(j\omega_i)$ whose magnitude as a function of frequency decreases as fast as possible (Horowitz, 1992). The justification for this is to consider the effects of high frequency sensor noise and the unmodelled high-frequency dynamics/parasitics, which may result, with unnecessarily large bandwidth, in actuator saturation and instability. It follows that the cost-function to be minimized is the high-frequency gain of the open loop transmission $L(s)$ where the high-frequency gain is defined by

$$\lim_{s \to \infty} s^q L(s) \quad (7)$$

when the relative order of the transfer function L

(s) is q. It is also called the cost of feedback in QFT. Since the nominal plant is fixed, this is equivalent to the high frequency gain of the controller, given by

$$J_{hg} = b_r / a_m \quad (8)$$

This performance index is widely adopted in QFT optimisation (Horowitz, 1991; Chait, 1997; Thompson and Nwokah, 1994; Bryant and Halikias, 1995).

3.7. *Fitness in the GA*

The fitness in a GA for the QFT design should reflect the stability and performance requirements, and the performance index, given by

$$J = J_{hg} + \sum_{i=1}^{h} \gamma_i J_{bi} + \gamma_0 J_{sta} \quad (9)$$

where $\gamma_i, i = 0, 1, \cdots, h$ are weighting factors. In general γ_i, $i = 0, 1, \cdots, h$ should be reasonably large. In this paper we choose ro is about $2h$ times than the biggest in γ_i, $i = 1, \ldots, h$, and γ_i is chosen as the maximum of all controller parameters.

3.8. *Choice of parameter space and coding*

After the controller order and the control structure in (4) are specified by the designer, the variables to be optimised are fixed, given by

$$P = [b_r, \cdots, b_1, b_0, a_{m-1} \cdots, a_1, a_0] \in R^{r+m+1} \quad (10)$$

when a_m is fixed equal to 1 in this paper.

In the preparation phase for GAs, all the parameters need to be coded by an integer string to form a chromosome. The choice of the range of the parameters is important in the automatic loop shaping, which determines the size of the searching space. Since all parameters are positive according to the stable minimum phase controller requirement, the minimum is chosen as close to zero and the maximum of all parameters can be chosen as a reasonably large number if there is no prior information about the ranges of the parameters. To improve the convergence of the Genetic Algorithms, noting that all controller parameters are positive, we optimise these parameters in the logarithmic space rather than the original parameter space. That is, first the parameters, P, are translated into the logarithmic space. For any element Pi in (10), let

$$\overline{p}_i = \log10(p_i), i = 1, \cdots, r + m + 1$$

The $\overline{P} = [\overline{P}_1, \cdots, \overline{p}_{r+m+1}]$ is coded to form a chromosome for GAs. Our expeirence shows this greatly saves the optimisation time.

4. Design example

The above method is applied to design a controller for the Benchmark Example 2 in the QFT Toolbox of MATLAB (Borghesani *et al.*, 1995). The automation of the design of a QFT controller for this example is also investigated by Chait (1997).

The uncertain plant is described by

$$P = \{ p(s) = \frac{ka}{s(s+a)} : k \in [1,10], a \in [1,10] \} \quad (11)$$

The closed-loop specifications include the robust stability margin and the tracking performance, which are specified by

$$\left| \frac{p(j\omega) G(j\omega)}{1 + P(j\omega) G(j\omega)} \right| \leq 1.2, \text{ for all } P \in p, \omega \geq 0 \quad (12)$$

and

$$T_L(\omega) \leq \left| F(j\omega) \frac{P(j\omega) G(j\omega)}{1 + P(j\omega)(Gj\omega)} \right| \leq T_U(\omega) \quad (13)$$

respectively, where

$$T_U(\omega) = \left| \frac{0.6854(j\omega + 30)}{(jw)^2 + 4(jw) + 19.752} \right| \quad (14)$$

and

$$T_L = \left| \frac{120}{(jw)^3 + 17(jw)^2 + 828(j\omega) + 120} \right| \quad (15)$$

The design objective in this paper is to find a controller such that all the closed loop specifications are satisfied and the cost of feedback is as small as possible. As in the QFT Toolbox, the QFT bounds on the frequencies $\omega = [0.1, 0.5, 1, 2, 15, 100]$ are calculated. In order to ensure the whole loop transmission satisfies the UHB (which is yielded by the robust stability margin requirement), in the automatic loop shaping, the frequency points ω = 120, 200, 300, 400, 500, 1000, 2000, 3000, 5000. are added to test the UHB. Since a third order controller is designed in the QFT Toolbox, we try to design a lower order controller for this plant and specify a second order controller.

For a second order controller, there are four parameters when $a_2 = 1$. Since there is no prior information about the range of the coefficients in the controller, the quite large ranges of all parameters are chosen, or $p_i \in [0.001, 10^{10}]$. Following the suggestion in Section 3, before coding the parameters to form a chromosome in GAs, we translate them into the logarithmic space. It therefore becomes $\overline{pi} \in [-3, 10]$. Since the maximum of the parameters are 10^{10}, γ_i, $i = 1, 2, \cdots$, is chosen as 10^{10} as well. γ_0 is selected as 10^{11} since the stability is more important than one QFT bound.

The design method developed in this paper gives, after 75 generations, a second order controller:

$$G(s) = \frac{6.753 \times 10^6 s + 1.3947 \times 10^7}{s^2 + 3.4834 \times 10^3 s + 1.6218 \times 10^6} \quad (16)$$

The loop shaping result is given in Figure 2. It can be shown that all robust stability margin and the performance specifications are satisfied. The nominal system is also stable. This controller would not be easy to design by other optimization methods since the coefficients vary in quite large ranges.

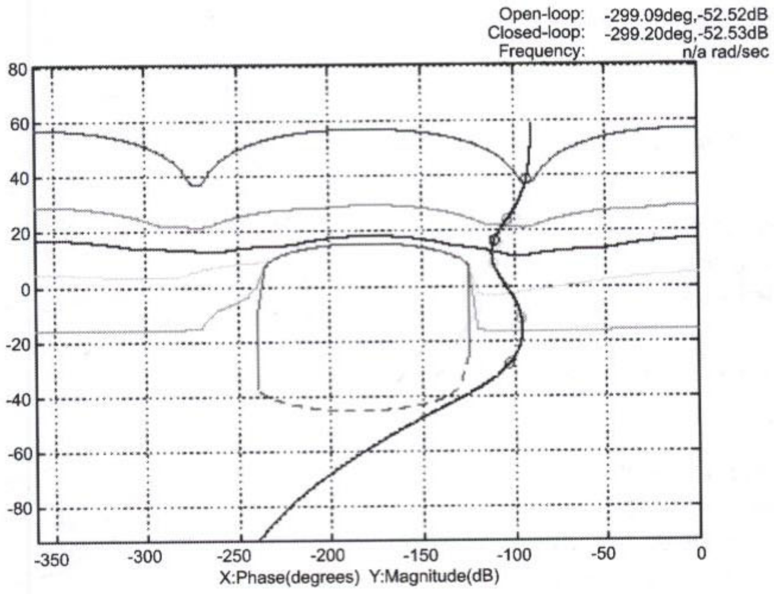

Fig. 2. Loop shaping via Genetic Algorithms: Second order controller.

The third order controller for this plant in the QFT Toolbox is given by

$$G(s) = \frac{3.0787 \times 10^6 s^2 + 3.5365 \times 10^8 s + 3.8529 \times 10^8}{s^3 + 1.5288 \times 10^3 s^2 + 1.0636 \times 10^6 s + 4.2810 \times 10^7} \quad (17)$$

The loop shaping result under this controller is shown in Figure 3. Only the high frequency gain of the new second order controller is a little higher than that of the third order controller in QFT Toolbox.

Chait (1997) also designs a controller for this plant by the design automation procedure. In Chait's method the poles of the nominal closed-loop transfer function must be prescribed. Certainly this imposes an unnecessary requirement on the design and thereby the design result is quite conservative. The controller order yielded by this method is the sum of the prescribed nominal closed-loop transfer function and the order of the nominal open-loop plant. As expected, the order of the controller is high. In Chait (1997) a fifth order and a ninth order controller are designed.

5. Conclusion

A genetic algorithm based automatic design procedure for QFT has been reported in this paper. Using this approach, an engineer does not need to approximate a complicated non-convex optimisation problem of QFT design by a linear and convex programming problem. The GA based approach overcomes the the disadvantages of the previous automatic design methods in QFT that results in imposing unrealistic assumptions or very conservative designs. The design is accomplished directly on the controller instead of on the loop transmission, as in the traditional QFT design and other automatic design methods. In converting a QFT design problem into a nonlinear optimisation problem, several key problems including the stability of the nominal plant, QFT robust stability and performance bounds, the internal stability of the uncertain systems etc. are discussed. The design result shows the introduction of genetic algorithm in QFT can efficiently overcome the difficulties in loop shaping and improve the QFT design. It provides an alternative approach to loop shaping in QFT design.

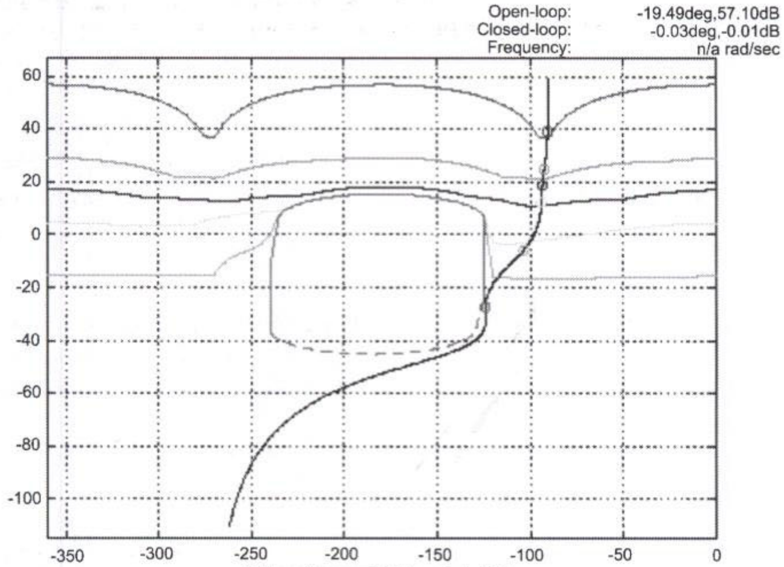

Fig.3. Loop shaping in the QFT Toolbox: Third order controller.

References

Ballance, D. J., 1992. Comments on the papers "A new approach to optimum loop synthesis" and "On the determination of plant variation bounds for optimum loop synthesis". *Int. J. Control* **55(1)**, 241-248.

Ballance, D. J., Gawthrop, P. J., 1991. Control systems design via a quantitative feedback theory approach. In: *Proceedings of IEEE Control*'91. **Vol. 1**. Edinburgh, UK. pp. 476-480.

Ballance, D. J., Gawthrop, P. J., 1992. QFT, the UHB, and the choice of the template nominal point. In: Proceedings of Qunatitative Feedback Theory.

Ballance, D.J., W.H. Chen, 1998. Symbolic computation in value sets of plants with uncertain parameters. In: Proceedings *of IEE Control*'98. Swansea, UK. pp. 1322-1327.

Bodineau, G., Boullade, S., Frapard, B., Chen, W. H., S. Salehi, Ankersen, F., 2004. Robust control of large flexible appendages for future space missions. In: *Proceedings of 6th International Conference on Dynamics and Control of Systems and Structure in Space*. Italy.

Borghesani, C., Chait, Y., Yaniv, O., 1995. *Quantitative Feedback Theory Toolbox User Manual*. The Math Work Inc.

Bryant, G. F., Halikias, G. D., 1995. Optimal loop-shaping for systems with large parameter uncertainty via linear programming. *Int. J. Control* **62(3)**, 557-568.

Chait, Y., 1997. QFT loop shaping and minimization of the high-frequency gain via convex optimization. In: *Proceedinds of the Symposium on Quantitative Feedback Theory and other Frequency Domain Methods and Applications*. Glasgow, Scotland. pp. 13-28.

Chen, W.H., Ballance, D.J., 1998a. Plant template generation in Quantitative Feedback Theory. In: *Proceedings of IEE Control*'98. Swansea, UK. pp. 810-815.

Chen, W.H., Balance, D.J., 1998b. QFT design for uncertain non-minimum phase and unstable plants. In: *Proceedings of American Control Conference*. pp. 2486-2490.

Cohen, N., Chait, Y., Yaniv, O., Borghesani, C., 1992. Stability analysis using nichols charts. In: *Proceedings of the Symposium on Qauntitative Feedback Theory*. Ohio, USA. pp. 80-103.

Horowitz, I. M., 1963. Synthesis of Feedback Systems. Academic Press.

Horowitz, I. M., 1973. Optimum loop transfer function in single-loop minimumphase feedback systems. *Int. J. Control* **18**, 97-113.

Horowitz, I. M., 1991. Survey of quantitative feedback theory (QFT). *Int. J Control* **53(2)**, 255-291.

Horowitz, I. M., 1992. *Quantitative Feedback Design Theory (QFT)*. Vol. **1**. QFT Publications, 4470 Grinnel Ave., Boulder, Colorado 80303, USA.

Horowitz, I. M., Gera, A., 1980. Optimization of the loop transfer function. *Int. J. Control* **31**, 89-398.

Horowitz, I. M., Sidi, M., 1978. Optimum synthesis of non-minimum phase feedback system with plant uncertainty. *Int. J. Control* **27**, 361-386.

Jayasuriya, S., Zhao, Y., 1994. Robust stability of plants with mixed uncertainties and quantitative feedback theory. *AMSE Journal Dynamic Systems, Measurement, and Control* **116(2)**, 10-16.

Li, Y., Ng, K.C., Murray-Smith, D.J., Gray, G.J., and K.C. Sharman, 1996. Genetic algorithm automated approach to design of sliding mode control systems. *Int. J. Control* **63(4)**, 721-739.

Li, Y., Alexander Haeussler, 1996. Artificial evolution of neural networks and its application to feedback control. Artificial Intelligence in Engineering **10** **(2)**, 143-152.

Thompson, D. F., Nwokah, O. D. I., 1994. Analytical loop shaping methods in quantitative feedback theory. *Journal of Dynamic Systems, Measurement, and Control* **116**, 169-177.