Preprints of the 10th IFAC International Symposium on Dynamics and Control of Process Systems
The International Federation of Automatic Control
December 18-20, 2013. Mumbai, India

Automatic Loop Shaping in QFT using Hybrid Optimization and Consistency Technique

R. Jeyasenthil *, P.S.V. Nataraj **

* *Systems and Control Engineering Group, IIT Bombay, India (Tel: +91-22-25764884; e-mail: jeya@sc.iitb.ac.in).*
** *Systems and Control Engineering Group, IIT Bombay, India (Tel: +91-22-25767887; e-mail: nataraj@sc.iitb.ac.in).*

Abstract: This paper proposes an efficient algorithm for automatic loop shaping in Quantitative Feedback Theory(QFT). The proposed method uses hybrid optimization and consistency techniques. Hull consistency is used to prune the input domain by removing the inconsistent values which are not a part of the solution. The hybrid optimization part combines interval global optimization and nonlinear local optimization methods. The proposed method is demonstrated on uncertain DC motor plant model and performance is compared with those of existing interval methods.

Keywords: Interval consistency technique, Quantitative feedback theory, global optimization, robust control.

1. INTRODUCTION

The Quantitative Feedback Theory(QFT) approach to robust control synthesis is a frequency domain approach and has been widely applied in industry [Horowitz, 1993]. The key step in QFT is the gain-phase loop shaping technique used to synthesis the controller. In this step,a controller is synthesized to satisfy the magnitude-phase QFT bounds on the nominal loop transmission function. The bounds capture the robust stability and robust performance specifications via quadratic inequalities. Various automatic loop shaping design are available in the QFT literature [Bryant and Halikias, 1995],[Gera and Horowitz, 1980].

A reliable method of automatic loopshaping uses deterministic global optimization method [Nataraj and Tharewal, 2007] based on interval branch and bound method. The approach is computationally slow due to usage of interval arithmetic. To speedup this approach, Nataraj and Kubal [2007] proposed a method based on hybrid optimization and geometric constraint propagation ideas. More recently, the QFT based controller synthesis problem was formulated as an constraint satisfaction problem(CSP) with quadratic inequality constraints [Nataraj and Kalla, 2010]. The CSP has been solved using a consistency technique that finds all the feasible controller parameter solutions in an initial search box based on branch and prune method. Among all the feasible solutions, the optimal solution is defined as the one having least high-frequency gain. The main drawback of this approach is its computational demand, since it looks for all feasible solutions, and not just the optimal solution.

In contrast to the above methods, the proposed method combines hybrid optimization and consistency technique ideas to solve the optimization problem in a branch and bound framework. The hybrid optimization part combines nonlinear local optimization methods with interval global optimization methods. Local optimization helps as it gives early knowledge of the (approximate) global minimum.

The salient features of the proposed approach can be given:

- The method is fully automated.
- It enables the designer to specify the controller structure in advance.
- For the specified controller structure and initial search box, this approach finds the optimal controller solution.

The rest of the paper is organized as follows: Section 2 deals with the QFT basics, the QFT controller synthesis problem and consistency techniques. Section 3 presents the key components of the proposed algorithm such as feasibility check,hull consistency and local optimization. The proposed algorithm based on hybrid optimization and consistency technique is then presented in section 4, and demonstrated on a DC motor example in section 5. The conclusion of the work is given in section 6.

2. BACKGROUND

This section gives an outline of QFT basics, the QFT controller problem and consistency techniques.

2.1 QFT basics

QFT [Horowitz, 1993] is a frequency domain based robust control techniques. The basic idea in QFT is to convert the given performance specifications into bounds in the Nichols chart. Using any gain-phase loop shaping method, a controller is designed to satisfy all the bounds at each design frequency. The aim of QFT is to minimize the cost of feedback i.e., to reduce the high frequency controller gain.

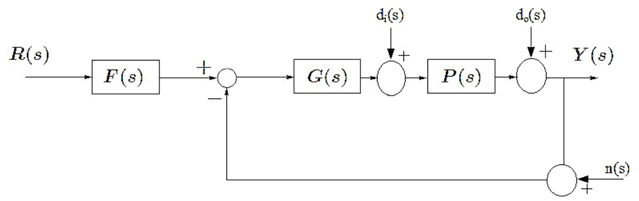

Fig. 1. The two degree-of-freedom structure in QFT

Consider the two degree of freedom configuration shown in Fig. 1, where $G(s)$ and $F(s)$ are the controller and prefilter respectively. The uncertain linear time-invariant plant $P(s)$ is given by $P(s) \in \{P(s,\lambda) : \lambda \in \boldsymbol{\lambda}\}$, where $\lambda \in \mathcal{R}^l$ is a vector of plant parameters whose values vary over a parameter box $\boldsymbol{\lambda}$ given by

$$\lambda = \{\lambda \in \mathcal{R}^l : \lambda_i \in [\underline{\lambda_i}, \overline{\lambda_i}], \underline{\lambda_i} \leq \overline{\lambda_i}, i = 1, ..., l\} \quad (1)$$

This gives rise to a parametric plant family or set

$$\mathcal{P} = \{P(s,\lambda) : \lambda \in \boldsymbol{\lambda}\}$$

The open loop transmission function is defined as

$$L(s,\lambda) = G(s)P(s,\lambda) \quad (2)$$

$$L(j\omega,\lambda) = g(j\omega)e^{j\phi(j\omega)}p(j\omega)e^{j\theta(j\omega)} \quad (3)$$

The nominal open loop transmission function is

$$L_0(j\omega) = g(j\omega)e^{j\phi(j\omega)}p_0(j\omega)e^{j\theta_0(j\omega)} \quad (4)$$

or

$$L_0(j\omega) = l_0(j\omega)e^{j\psi_0(j\omega)} \quad (5)$$

where

$$l_0(j\omega) = g(j\omega)p_0(j\omega)$$

and

$$\psi_0(j\omega) = \phi_0(j\omega) + \theta_0(j\omega)$$

The objective in QFT is to synthesize $G(s)$ and $F(s)$ such that the various stability and performance specifications are met for all $P(s) \in \mathcal{P}$ and at each $\omega \in \Omega$ is the design frequency set. In general, the following specifications are considered in QFT:

(1) Robust stability margin

$$\left|\frac{L(j\omega)}{1+L(j\omega)}\right| \leq w_s \quad (6)$$

w_s = desired stability margin specification

(2) Robust Input disturbance rejection

$$\left|\frac{P(j\omega)}{1+L(j\omega)}\right| \leq w_{d_i}(\omega) \quad (7)$$

$w_{d_i}(\omega)$ = Input disturbance specification

(3) Control effort constraint

$$\left|\frac{G(j\omega)}{1+L(j\omega)}\right| \leq w_c(\omega) \quad (8)$$

$w_c(\omega)$ = Control Effort specification

(4) Robust tracking performance

$$|T_L(j\omega)| \leq \left|\frac{F(j\omega)L(j\omega)}{1+L(j\omega)}\right| \leq |T_U(j\omega)| \quad (9)$$

where $T_L(j\omega)$ and $T_U(j\omega)$ are the lower and upper tracking performance specifications. The above specification can be converted into corresponding quadratic inequalities [Chait and Yaniv, 1993]

(1) Robust stability margin

$$g^2p^2\left(1 - \frac{1}{{w_s}^2}\right) + 2gp\cos(\phi+\theta) + 1 \geq 0 \quad (10)$$

where g and p are the magnitude of the controller and plant, respectively, ϕ and θ are the phase angle of the controller and plant, respectively.(see equation 3)

(2) Robust Input disturbance rejection

$$g^2p^2 + 2gp\cos(\phi+\theta) + (1 - \frac{p^2}{{w^2}_{d_i}}) \geq 0 \quad (11)$$

(3) Control Effort Constraint

$$g^2p^2 + 2gp\cos(\phi+\theta) + (1 - \frac{g^2}{{w^2}_c}) \geq 0 \quad (12)$$

(4) Robust tracking performance

$$g^2{p_k}^2{p_i}^2\left(1 - \frac{1}{\delta^2(\omega)}\right) + 2gp_kp_i\left[p_k\cos(\phi+\theta_i) - \frac{p_i}{\delta^2(\omega)}\cos(\phi+\theta_k)\right] + \left({p_k}^2 - \frac{{p_i}^2}{\delta^2(\omega)}\right) \geq 0 \quad (13)$$

where

$$\delta(\omega) = \left|\frac{T_U(j\omega)}{T_L(j\omega)}\right|$$

The above constraints are to be met for all $P(s) \in \mathcal{P}$, and for all ω. The last inequality should be met for each possible pair $P_i(j\omega) = p_i(j\omega)e^{j\theta_i(j\omega)}$,$P_k(j\omega) = p_k(j\omega)e^{j\theta_k(j\omega)}$ of plants from the uncertain plant set $\mathcal{P}$.

The main steps of the QFT design process are

(1) **Generating plant templates**: For a given uncertain plant $P(s) \in \mathcal{P}$, at each design frequency $\omega_i \in \Omega$, calculate the template or value set for set of plants $P(j\omega)$ in the complex plane.
(2) **Computation of QFT bounds**: At each design frequency ω_i, translate the stability and performance specifications using the plant templates to obtain the stability and performance bounds in the Nichols plane.The bound at ω_i is denoted as $B_i(\angle L_0(j\omega), \omega_i)$, or simply $B_i(\omega_i)$.
(3) **Design of controller**: Design a controller $G(s)$ using loop shaping technique such that
    - The bound constraints at each design frequency ω_i are satisfied.
    - The nominal closed loop system is stable.
(4) **Design of prefilter**: Design a prefilter $F(s)$ such that the robust tracking specifications in (9) are satisfied. Two inequalities for robust tracking specifications result from (9) :

$$|T_L(j\omega)| - |F(j\omega)||T(j\omega)| \leq 0; \quad (14)$$

$$|F(j\omega)||T(j\omega)| - |T_U(j\omega)| \leq 0; \quad (15)$$

where $|T(j\omega)| = |\frac{L(j\omega)}{1+L(j\omega)}|$

2.2 QFT Controller synthesis - Problem Formulation

The QFT controller synthesis problem can be posed as a constrained optimization problem with the objective function as the high frequency gain of the controller, and the constraints set for the optimization as the set of possibly nonconvex, nonlinear magnitude-phase QFT bounds at the various design frequencies [Nataraj and Tharewal, 2007].

In this work, the controller structure is represented in the gain-pole-zero form

$$G(s,x) = \frac{k_G \prod_{i_1=1}^{n_z} (s + z_{i_1})}{\prod_{k_1=1}^{n_p} (s + p_{k_1})} \tag{16}$$

where the controller parameter vector is

$$x = (k_G, z_1, ...z_{n_z}, p_1, ..., p_{n_p}). \tag{17}$$

Here z_{i_1} and p_{k_1} can be real or complex values.

The magnitude and phase functions of $G(s,x)$ are defined as

$$g_{mag}(\omega, x) = |G(j\omega, x)|;\ g_{ang}(\omega, x) = \angle G(j\omega, x) \tag{18}$$

For the given robust stability and performance specifications, the initial QFT design method is followed up to the QFT bound generation step.

Proceeding with the minimization of k_G as the objective, the QFT controller synthesis can be formulated as the following constrained global optimization problem:

find min k_G

subject to $H(x) <= 0$

Where x is the vector of controller parameters in (17).

The QFT bound $B_i(\angle L_0(j\omega_i, x), \omega_i)$ at each design frequency ω_i is to be respected, leading to the set of inequality constraints $H(x)$. Typically, the constraint set $H(x)$ is nonlinear and non-convex and is given by $H(x) = \{h_i(x)\}$, where $h_i(x)$ represents a single-valued bound constraint

$$h_i^u(x) = |L_0(j\omega_i, x)| - B_i(\angle L_0(j\omega_i, x), \omega_i) \leq 0 \tag{19}$$

$$h_i^l(x) = B_i(\angle L_0(j\omega_i, x), \omega_i) - |L_0(j\omega_i, x)| \leq 0 \tag{20}$$

Where h_i^u,h_i^l denotes the single-valued upper and lower bound constraints respectively. A multiple-valued bound can be split in terms of h_i^u and h_i^l, and represented in the above form.

2.3 Consistency Techniques

Well known consistency methods are Hull and Box consistency approach. Hull consistency is a constraint inversion procedure with respect to each variable in a constraint and it checks the consistency only on the bounds of the variable domains. HC4 is the most important hull consistency method available in the literature [Benhamou et al., 1999]. It works in two steps, called as forward evaluation and backward propagation on a binary constraint tree. In the forward phase, tree traversal is from the bottom to top node, evaluating at each node the natural interval evaluation of that sub-term of the constraint. In the backward phase, the tree is traversal from the top to bottom node, projecting on each node the effects of interval narrowing already performed on its parent node.

For further details, readers can refer to this link [1].

[1] http://www.sc.iitb.ac.in/~jeya/ProposedHybridAlgorithm.pdf

3. PROPOSED ALGORITHM COMPONENTS

The proposed algorithm has three main components (a) Feasibility check, (b) Hull consistency, and (c) local optimization. These are explained below.

3.1 Feasibility Check

Let $|B_i|_{max}$ and $|B_i|_{min}$ be the top most and bottom most value of the single valued QFT bound for the entire phase interval $\angle L_0(j\omega_i, \mathbf{z})$. Based on the location of the L_0 box with respect to this bound, one of the following cases arises, see Fig. 2

(1) If the entire L_0 box lies on or above $|B_i|_{max}$ (box A in Fig. 2) then h_i^l is satisfied for any controller parameter vector $z \in \mathbf{z}$, so that the entire box $\mathbf{z}$ is *feasible* at ω_i. Here $\mathbf{z}$ = x.
(2) If the entire L_0 box lies below $|B_i|_{min}$ (box B in Fig. 2) then h_i^l is not satisfied for any controller parameter vector $z \in \mathbf{z}$, so that the entire box $\mathbf{z}$ is *infeasible* at ω_i.
(3) Else box $\mathbf{z}$ is indeterminate (box C in Fig. 2).

An algorithm based on above ideas is as follows

FC ALGORITHM : $flag_z$=FC(L_{0mag}, L_{0phase}, B_i, Ω, $\mathbf{z}$)
Inputs: The controller parameter box $\mathbf{z}$, interval extensions [Moore, 1966] $L_{0mag}(\omega_i, \mathbf{z})$, $L_{0phase}(\omega_i, \mathbf{z})$, QFT bounds B_i and design frequency vector Ω.

Output: $flag_z$ describes the feasibility status of the box $\mathbf{z}$.

BEGIN FC ALGORITHM

(1) FOR $i = 1, ..., n$
 (a) Evaluate $|L_0(j\omega_i, \mathbf{z})|$ and $\angle L_0(j\omega_i, \mathbf{z})$
 (b) Evaluate $|B_i|_{max}$, $|B_i|_{min}$
 (c) IF inf $|L_0(j\omega_i, \mathbf{z})| \geq |B_i|_{max}$ THEN set $flag(\omega_i) =$ *feasible*.
 ELSE IF sup $|L_0(j\omega_i, \mathbf{z})| \leq |B_i|_{min}$ THEN set $flag(\omega_i) =$ *infeasible* and go to Step 3.
 END FOR
(2) IF $flag(\omega_i) =$ *feasible* for all ω_i, $i = 1, ..., n$ THEN set $flag_z =$ *feasible* and EXIT FC ALGORITHM.
 ELSE set $flag_z =$ *indeterminate* and EXIT FC ALGORITHM.
(3) Set $flag_z =$ *infeasible* and EXIT Algorithm.

END FC ALGORITHM

3.2 Hull Consistency

HC4 filter operates on each QFT quadratic inequality constraint [Nataraj and Kalla, 2010] separately as explained in section 2.3. It works untill there is no reduction in any of the input parameter domain in a iteration.

Inputs: The controller parameter box $\mathbf{z}$ and quadratic inequality constraints ($\mathbf{C_i}$).
Output: A reduced controller parameter box $\mathbf{z}$, Reduction flag ($flag_{HC}$).
HC ALGORITHM : ($\mathbf{z}$, $flag_{HC}$) = HC4($\mathbf{C_i}$, $\mathbf{z}$).

BEGIN HC ALGORITHM

(1) Forward Evaluation Phase (see Section 2.3)

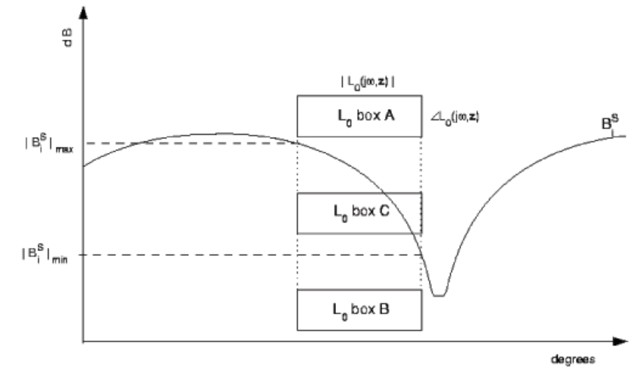

Fig. 2. Feasibility conditions for different locations of L_0 box with respect to single valued lower bound B_i at ω_i.

(2) Backward Propagation Phase (see Section 2.3)

END HC ALGORITHM

3.3 Local Optimization

Local constrained optimization methods are used to quickly locates the approximate global minimum, provided the obtained local solution is a feasible one. In our context, the gain intervals are clipped to have local solution at their supremum since the objective is to seek the solution with minimum high frequency gain. And other controller parameters with local solutions by clipping their end points.

Local Optimization ($z_{local}, flag_{z_{local}}$)= LO($\mathbf{z}$, f, c_i)

Input: The controller parameter box $\mathbf{z}$, Objective function(f) and bound constraints($H(x)$). Here $f = \inf(z(1))$.

Output: A Locally optimum solution z_{local}, and value of a feasibility flag $flag_{z_{local}}$ of the locally optimum solution.

BEGIN LO ALGORITHM

(1) Let z= Mid $\mathbf{z}$, where Mid = Midpoint
   (a) With z as the starting point, call a nonlinear constrained local optimization routine to solve the optimization problem. For instance, we can call fmincon routine of MATLAB [MathWorks, 2009]
   (b) Augment z: $z \leftarrow (z_{local}, z)$.

END LO ALGORITHM

3.4 Usefulness of HC4 in QFT Design

Let us consider the fixed(for simplicity) plant transfer function for the speed loop of a DC Motor [Borghesani et al., 1995]

$$P(s) = \frac{10}{s(s+1)};$$

The design specification is the bandwidth specification i.e.

$$\left|\frac{L(j\omega)}{1+L(j\omega)}\right| \leq 0.707;\ \omega > 10$$

The design frequency set is

$$\Omega = [0.01, 0.05, 0.1, 0.2, 1, 2, 10, 12, 15, 35, 100]$$

A gain controller(K) with input domain $[0, 100]$ is chosen. For bandwidth specifications, the quadratic inequality can be written as

$$(K^2) * \left(1 - \frac{1}{{w_b}^2}\right) * p^2 + 2 * \cos(\phi) * K * p + 1 = [0, \infty]$$

Fig 3 shows the backward propagation phase of the constraint tree where arrows are used to represents the top down phase. The projection of a variable with multiple occurrence in a constraint tree is different.

At Node 9 :
Father Node = $N6$ = 2*Cos(ϕ)*K ; Brother Node = $N8$= 2*cos(ϕ)

$$\mathbf{Projection_{N9}} = \frac{Backwardpropagation_{N6}}{ForwardEvaluation_{N8}}$$
$$= \frac{[-16.73.., 0]}{[-1.994.., -1.994..]}$$
$$\mathbf{Projection_{N9}} = [0, 8.3892..]$$

$$\mathbf{N9_{new}} = \mathbf{N9_{old}} \cap \mathbf{Projection_{N9}}$$
$$= [0, 100] \cap [0, 8.3892]$$
$$\mathbf{N9_{new}} = [0, 8.3892]$$

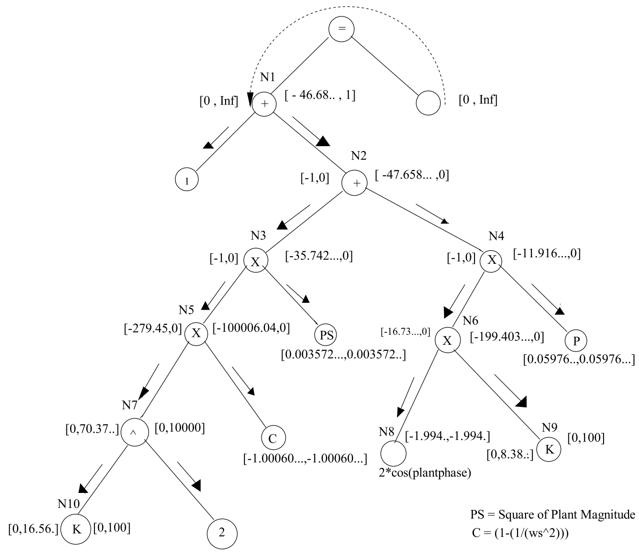

Fig. 3. Backward Propagation Phase

At Node 10 :
Father Node = $N7 = K^2$; Brother Node = 2 (Power Term)

$$\mathbf{Projection_{N10}} = \sqrt{(BackwardPropagation_{N7})}$$
$$= \sqrt{[0, 70.37..]}$$
$$\mathbf{Projection_{N10}} = [-16.56.., 16.56..]$$

$$\mathbf{N10_{new}} = \mathbf{N10_{old}} \cap \mathbf{Projection_{N10}}$$
$$= [0, 100] \cap [-16.56.., 16.56..]$$
$$\mathbf{N10_{new}} = [0, 16.56..]$$

After the first backward propagation phase, the gain value at Node 9 is $[0, 8.3892]$ where as at Node 10, its value is $[0, 16.56..]$. Thus, after one iteration, the gain interval reduces from $[0, 100]$ to $[0, 8.3892]$.

4. PROPOSED ALGORITHM

The proposed algorithm begins with an initialization part, by assigning the value of initial search domain x to the current box $\mathbf{z}$. The feasibility of box z is tested as described in section 3.1. A triple $(\mathbf{z}, z, flag_z)$ is formed where z is the lower bound of objective function over the box $\mathbf{z}$ and the initialization part is completed by putting this triple in the working list(L). The algorithm then calls the hull consistency algorithm as explained in section 3.2. The reduced box is given to local optimization as described in section 3.3.

After this, the current box under process $\mathbf{z}$ is split into two subboxes $\mathbf{v}^1$,$\mathbf{v}^2$ and feasibility test is carried out on each subbox. The subbox found to be infeasible is discarded and any remaining one is added to the working list(L). The list is arranged such that the second members of all triples i.e. minimum value of objective function over respective boxes in list do not decrease. If, at any iteration, the box $\mathbf{z}$ of the leading triple is of sufficiently small width(ϵ) or the List(L) is empty, then algorithm is terminated after printing out the optimal controller solution Z. We now present the algorithm.

ALGORITHM: QFT CONTROLLER DESIGN

Inputs: The tracking and stability bounds at the various design frequencies obtained using the QFT toolbox [Borghesani et al., 1995], an initial search box $\mathbf{x}$ of controller parameter values, interval extensions $L_{0mag}(\omega, \mathbf{x})$, $L_{0phase}(\omega, \mathbf{x})$, Quadratic inequality constraints c_i, $i = 1, 2, ...m$, the design frequency vector $\Omega = \{\omega_i,\ i = 1, ..., n\}$ and a prespecifiedbox tolerance ϵ.

Outputs: One of the following

- Message 'No feasible solution exists in the given search box $\mathbf{x}$'.
- An optimal controller parameter vector Z satisfying all the QFT bounds.

BEGIN ALGORITHM

(1) Set $\mathbf{z} = \mathbf{x}$ and initialize $z_v \leftarrow \{\}$
(2) Execute Feasibility Check algorithm,
$flag_z$=FC($L_{0mag}\, L_{0phase}$, B_i, Ω, $\mathbf{z}$).(see Section 3.1)
(3) IF $flag_z$ = *infeasible*, then PRINT 'No feasible solution exists in the given search box' and EXIT ALGORITHM.
IF $flag_z$ = *feasible*, then PRINT 'controller parameter box is $\mathbf{z}$' and EXIT ALGORITHM.
IF $flag_z$ = *indeterminate*, then ENTER Iterative Part(Steps 4-16).
(4) Initialize list $L := \{\mathbf{z},\ z,\ flag_z\}$ where $z = \inf \mathbf{z}(1)$.
(5) Execute Hull Consistency (HC4) Algorithm,
$(\mathbf{z},\ flag_{HC})$ = HC4($\mathbf{C_i}$, $\mathbf{z}$).(see Section 3.2).
(6) Execute Local optimization part,
$(z_{local}, flag_{z_{local}})$= LO($\mathbf{z}$, f, c_i).(see Section 3.3)
(7) Bisect the current box $\mathbf{z}$ in the longest direction, into two subboxes $\mathbf{v}^1$ and $\mathbf{v}^2$ such that $\mathbf{z} = \mathbf{v}^1 \bigcup \mathbf{v}^2$.
(8) Execute Feasibility Check algorithm,
$flag_{v^j}$=FC(L_{0mag}, L_{0phase}, B_i, Ω, $\mathbf{v}^j$), $j = 1, 2$.
(9) Set $v^1 = \inf \mathbf{v}^1(1)$, $v^2 = \inf \mathbf{v}^2(1)$ and form the triples $\{\mathbf{v}^1,\ v^1,\ flag_{z^1}\}$, $\{\mathbf{v}^2,\ v^2,\ flag_{z^2}\}$.
(10) IF $flag_{v^j}$ = *infeasible* THEN discard the triple $\{\mathbf{v}^j,\ v^j,\ flag_{z^j}\}$, $j = 1, 2$.
(11) Remove the triple $\{\mathbf{z},\ z,\ flag_z\}$ from the list L.
(12) Add any remaining triple(s) from step 10 to the list L.
(13) Sort the list L such that the second members of all pairs of L do not decrease.
(14) Denote the first item in the list as $\{\mathbf{z},\ z,\ flag_z\}$.
(15) IF Width(box)$\leq \epsilon$ OR list L is empty, THEN PRINT 'Optimal controller parameter vector is Z' and EXIT algorithm.
(16) Go to Step 5.

END ALGORITHM

5. DESIGN EXAMPLE

The proposed algorithm is demonstrated on a DC motor plant model having uncertain plant parameters. To show the capabilities of the proposed approach, we also compare its performance with existing interval based algorithms ([Nataraj and Tharewal, 2007],[Nataraj and Kalla, 2010]) in terms of performance metrics such as number of iterations and average computational time.

Consider the uncertain plant transfer function of a DC motor [Zolotas and Halikias, 1999]

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

where the nominal parameter values are $k_0 = 1$, $a_0 = 1$. Our aim is to design a robust PID controller meeting the closed-loop specifications given below

(1) Robust Stability margin specification: $\omega_s = 1.2$ in (6)
(2) Tracking performance specifications:
In (9), we have

$$T_L(s) = \frac{0.6584(s+30)}{s^2 + 4s + 19.752};$$

$$T_U(s) = \frac{8400}{(s+3)(s+4)(s+10)(s+70)};$$

The initial search domain for the controller parameter is taken as $[k_p] = [5, 10^3]$; $[k_i] = [0.1, 10]$; $[k_d] = [3, 5]$. The controller solutions are to be found to an accuracy $= 0.1$. The design frequency(rad/sec) set is chosen as $\omega = [.5, 1, 2, 10, 30, 60]$.

For this problem Zolotas and Halikias designed a PID controller (singular value decomposition approach) given by

$$G_a(s) = (12.6 + 3.95s + \frac{4.46}{s}) \quad (21)$$

we now apply the proposed algorithm. For some insight into the algorithm working, consider the following situation be at some iteration. Let the controller parameter box

$$[k_p, k_i, k_d] = \{[7.021.., 9.0431..], [0.1, 3.76..], [3, 5]\}. \quad (22)$$

After HC4 pruning, the box reduced to

$$[k_p, k_i, k_d] = \{[7.02.., 7.19..], [0.1, 0.79..], [3.88., 5]\}. \quad (23)$$

Now the mid point of reduced box is taken as starting point for local optimization solver and the local solution is

Table 1. Performance comparison

Algorithm	Iterations	Time(sec)
HC4 [Nataraj and Kalla, 2010]	201	204.13
IGO [Nataraj and Tharewal, 2007]	342	70
Proposed Hybrid Optimization	9	10.3

IGO = Interval Global Optimization

$(k_p, k_i, k_d) = \{7.10, 0.44, 4.44\}$. Update the parameter box with feasible local solution as below,

$$[k_p, k_i, k_d] = \{[7.02.., 7.10..], [0.1, 0.44..], [3.88.., 4.44..]\}. \tag{24}$$

The Proposed algorithm in section 4 gives the PID controller

$$G_b(s) = (7.03 + 3.89s + \frac{0.1}{s}) \tag{25}$$

For further details, readers can refer to this link [2].

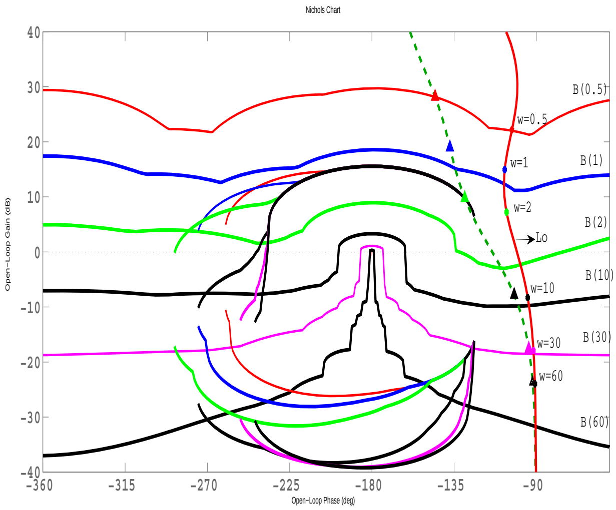

Fig. 4. Plots of the nominal loop transmission functions (a) Zolotas and Halikias algorithm (dash-dot line) (b) Proposed algorithm (solid line)

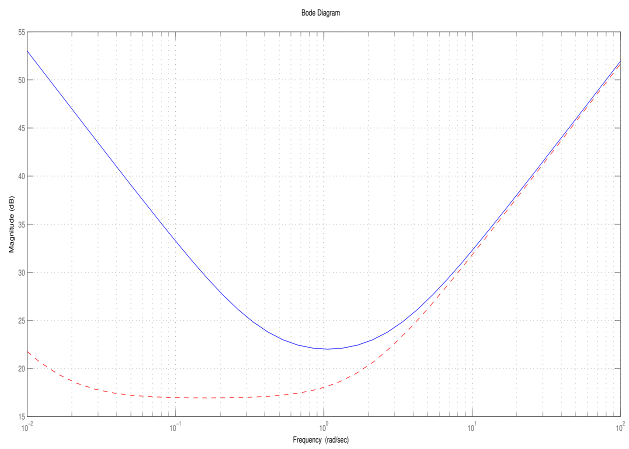

Fig. 5. Controller Magnitude Plots (a) Zolotas and Halikias algorithm (solid line) (b) Proposed algorithm(dash-dot line)

[2] http://www.sc.iitb.ac.in/∼jeya/ProposedHybridAlgorithm.pdf

A controller high frequency gain reduction of up-to 55 percent is obtained as compared to Zolotas and Halikias approach. And upto 5 times speed up is obtained with proposed algorithm than existing interval algorithms ([Nataraj and Tharewal, 2007],[Nataraj and Kalla, 2010]).

Table 1 compares the performance metrics given by the existing interval algorithms and the proposed hybrid approach.

6. CONCLUSIONS

A computationally efficient method for automatic loop shaping in QFT is proposed. The proposed method uses hybrid optimization techniques and consistency ideas. The method operates on the various QFT quadratic inequalities describing the robust stability and performance specifications , over the set of uncertain plants at each design frequency. The proposed algorithm outperforms the existing algorithms in terms of performance metrics such as number of iteration and average computation time. Thus aids in producing globally optimum controller solution in a reasonable time for a given controller structure and initial search domain. The proposed algorithm can be extended to multivariable systems with fair numerical efficiency.

REFERENCES

F. Benhamou, F. Goualard, and L. Granvilliers. Revising hull and box consistency. In *Proc. of ICLP'99 - MIT Press*, pages 230–244, 1999.

C. Borghesani, Y. Chait, and O. Yaniv. *The quantitative feedback theory toolbox for MATLAB*. The MathWorks, Inc., MA, USA, 1995.

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

Y. Chait and O. Yaniv. Multi-input/single-output computer aided control design using Quantitative Feedback Theory. *International Journal of Robust and Nonlinear Control*, 3(1):47–54, 1993.

A. Gera and I. M. Horowitz. Optimization of the loop transfer function. *Int. J. Control*, 31:389–398, 1980.

I. M. Horowitz. *Quantitative Feedback Theory (QFT)*. QFT Publications, Boulder, Colorado, 1993.

MathWorks. *Optimization toolbox user's guide for MATLAB version 4.2*. The MathWorks Inc, MA, USA, 2009.

R. E. Moore. *Interval Analysis*. Prentice-Hall, Englewood Cliffs, New Jersey, 1966.

P. S. V. Nataraj and R. Kalla. Synthesis of fractional-order qft controllers using interval constraint satisfaction technique. In *Proc. of 4th IFAC Workshop on Fractional Differentiation and its Applications*, Badajoz, Spain, 2010.

P. S. V. Nataraj and N. Kubal. Automatic loop shaping in qft using hybrid optimization and constraint propagation techniques. *International Journal of Robust and Nonlinear Control*, 17:251–264, 2007.

P. S. V. Nataraj and S. Tharewal. An interval analysis algorithm for automated controller synthesis in QFT designs. *ASME Journal of Dynamic Systems,Measurement and Control*, 129:311–321, 2007.

A. Zolotas and G. Halikias. Optimal design of pid controllers uaing the qft method. In *IEE Proc.of Control Theory Applications*, pages 585–589, UK, 1999.