

Techniques for Tuning Track-keeping Controllers of an Autonomous In-scale Fast-ferry Model

Francisco Jesús Velasco, Elías Revestido, Eloy López, Emiliano Moyano and M. Haro Casado

Abstract—In the present paper, optimization techniques are used to tune heading controllers for the purpose of track-keeping. Several simulations point out the improvement obtained with this short of techniques with respect to other. The heading control is based on a model obtained from data provided by a platform for marine vehicles. This platform has an in-scale fast-ferry model TF-120, which was developed to be autonomous and is controlled remotely from a PC using Wi-Fi communications. A sensitivity approach has been used for design of an input signal based on a previous parameter estimates. Different classical control structures for heading autopilot and track-keeping are studied and results obtained from optimization techniques are compared to genetic algorithm techniques.

Keywords— Optimization, tuning, heading control, track-keeping.

I. INTRODUCTION

A marine vehicle Web-Wi-Fi platform was developed in [2, 15] to carry out tests in open waters that can not be made in a model basin and for at-scale experimentation of coordination between sea vessels.

As a part of the platform, a physical in-scale model of the TF-120 turboferry (figure 2) was designed to be autonomous

Manuscript received December 31, 2008.

This work has been partially supported by MEC with the project DPI2006-11835.

Francisco Jesús Velasco is with the Dept. Electronic Technology & Systems Engineering and Automatic Control, Univ. Cantabria, E.T.S. de Náutica - C/ Gamazo 1 - 39004 Santander (Cantabria), Spain, (e-mail: velascof@unican.es).

Elías Revestido is with the Dept. Electronic Technology & Systems Engineering and Automatic Control, Univ. Cantabria, E.T.S. de Náutica - C/ Gamazo 1 - 39004 Santander (Cantabria), Spain, (e-mail: revestidoe@unican.es).

Eloy López is with the Dept. Sciences and Navigation Techniques, Engines and Naval Architecture, Univ. País Vasco (UPV/EHU), E.T.S. de Náutica y Máquinas Navales - C/ Ma Diaz de Haro 68 - 48920 Portugalete (Vizcaya), SPAIN, (e-mail: eloy.lopez@ehu.es).

Emiliano Moyano is with the Dept. Applied Mathematics and Computing Sciences, Univ. Cantabria, E.T.S. de Ingenieros Industriales y Telecomunicaciones - Avda. de los Castros, s/n - 39005 Santander (Cantabria) SPAIN, (e-mail: moyanoe@unican.es).

M. Haro Casado is with the Dept. Systems Engineering and Automatic Control, Electronic Technology & Electronics, University of Cádiz Faculty of Nautical Sciences - Polígono Río San Pedro s/n, Edificio C.A.S.E.M. - 11510. Puerto Real. Cádiz, Spain.

and controlled remotely from a Laptop in order to perform sea vessel trials [16, 19] with research purposes.

With the data from sea trials, in the coastline of the bay of Santander, with the marine Web-Wi-Fi platform a maneuvering model was identified [17]. The tuning of the autopilot is based on this model and also the design of an input signal to improve the previous parameter estimates.

In this work different optimization techniques are used to tune heading controllers such as: Non-linear least squares optimization and minimizing the maximum objective, in order to design a track-keeping system based on waypoint guidance by line of sight (LOS).

Moreover, classical controllers are obtained for the mathematical ship heading model which have been tested in simulation. Genetic algorithm techniques have been used to be compared with optimization techniques. In addition, the possibilities of the platform for experimentation of marine vehicles with research and educational purposes are presented.

II. PLATFORM ELEMENTS

Figure 1 shows a schema of the on-board and on-land elements which make up marine vehicle Web-Wi-Fi platform in order to do sea trials.

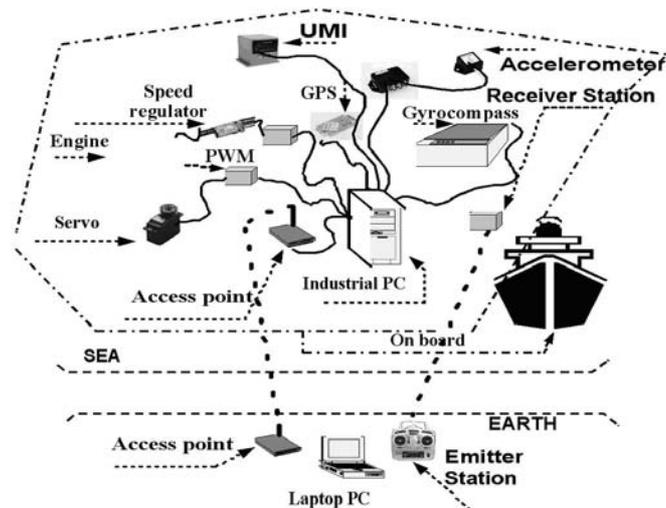


Fig. 1 Elements which make up the platform.

On-board Elements:

Components for the propulsion and steering of the vessel

(motors, servos, speed controller, turbojets).

Components of the control circuitry: PWM = Pulse Width Modulation and receiver station.

Sensors (GPS, electronic gyrocompass, UMI and accelerometers).

Communication components: industrial PC and access point.

On-land Communication Elements:

One laptop with an access point.

One radio control station which takes over the steering of the vessel in case of failure of the Wi-Fi network.

With the data from the gyrocompass a heading model was identified.

III. HEADING AUTOPILOT

In the next sections optimization techniques are developed to tune an autopilot so the heading control problem must be explained before.

An automatic pilot must fulfill two functions: course-keeping and change of course. In the first case, the objective is to maintain the trajectory of the vessel following the desired heading ($\psi(t) = \text{constant}$) of the ship (see figure 2). In the second case, the objective is to perform the change of heading without excessive oscillations and in the minimum time possible. In both situations, the correct functioning of the system must be independent from the disturbances produced by the wind, the waves and the currents.

The trajectory followed by a vessel can be specified by means of a second order reference model:

$$\ddot{\psi}(t) + 2\zeta\omega_n\dot{\psi}(t) + \omega_n^2\psi(t) = \omega_n^2\psi_r \quad (1)$$

where ω_n is the natural frequency and ζ ($0,8 \leq \zeta \leq 1$), the desired damping coefficient system in a closed loop.

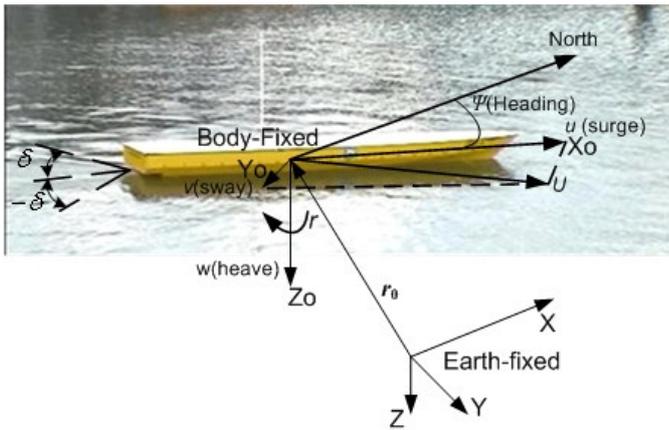


Fig. 2 Variables that describe the movement of the physical model TF-120 on the horizontal plane.

The tuning of the autopilot is based on a heading model. This model was proposed by Nomoto [6] and fit the following equations:

$$T\ddot{\psi} + \dot{\psi} = K\delta \quad (2)$$

whose transfer function is:

$$\frac{\psi}{\delta}(s) = \frac{K}{s(1+Ts)} \quad (3)$$

where,

ψ : Heading.

δ : Turbojets angle.

and the corresponding parameters, were identified in a previous work [20] with a turning circle manoeuvre:

$$K = -0.94828, T = 4.2497 \quad (4)$$

IV. INPUT DESIGN

In this section is explained the sensitivity approach used for design of an input signal and how it is applied to the heading model exposed in previous sections to improve the parameter estimates in (4).

A. Parameter Sensitivity Approach

A sensitivity approach for design of input signals and evaluation of the accuracy of parameter estimates was proposed by Knudsen [9]. It is a measure of the parameter sensitivity with respect to the model error, valid in a local range around the optimal estimate $\hat{\theta}_{opt}$. The usual approach for parameter estimation is to minimize a quadratic performance function $V(\theta)$ of the model error $\varepsilon(k)$

$$\begin{aligned} \varepsilon(k) &= y(k) - \hat{y}(k) \\ V(\theta) &= \frac{1}{N} \sum_{k=1}^N \varepsilon^2(k, \theta) \end{aligned} \quad (5)$$

However, in the sensitivity approach the model error is divided into parts, a part ε_0 caused by noise and model structure errors, and a parameter dependent part ε_p :

$$\begin{aligned} \varepsilon(k, \hat{\theta}) &= \varepsilon_0(k) + \varepsilon_p(k) \\ &= y(k) - \hat{y}(k, \hat{\theta}_{opt}) + \hat{y}(k, \hat{\theta}_{opt}) - \hat{y}(k, \hat{\theta}) \end{aligned} \quad (6)$$

For the sensitivity to be independent of the operating point, it is necessary to introduce a so called parameter dependent part of the model error, and to perform a linearization in the parameters of the model using Taylor expansion to the ε_p .

Thus, it is calculated:

$$\tilde{y}(k, \hat{\theta}) \cong \hat{y}(k, \hat{\theta}_{opt}) + \nabla(k, \hat{\theta}_{opt})(\hat{\theta} - \hat{\theta}_{opt}) \quad (7)$$

Where the model gradient is

$$\nabla(k, \hat{\theta}_{opt}) = \frac{d\hat{y}(k, \theta)}{d\theta} \Big|_{\theta=\hat{\theta}_{opt}} \quad (8)$$

The root mean square error is defined as:

$$\varepsilon_{p,RMS}(\theta) = \sqrt{\frac{1}{N} \sum_{k=1}^N \varepsilon_p^2(k, \theta)} \quad (9)$$

Where the approximated Hessian matrix $H(\hat{\theta}_{opt})$ is

$$H(\hat{\theta}_{opt}) = \frac{1}{N} \sum_{k=1}^N \nabla(k, \hat{\theta}_{opt}) \nabla^T(k, \hat{\theta}_{opt}) \quad (10)$$

For more details, for the assessing of the RMS expression, see Knudsen [10].

A minimization of $\varepsilon_{p,RMS}(\theta)$ provides the same value, as minimization of the typical $V(\theta)$, equation (5). The advantage of (5) over (9), for the sensitivity study, is that $\varepsilon_{p,RMS}(\theta)$ is linear in parameters θ .

As the values of the individual parameters may be very different, relative parameter sensitivity are to be used. Accordingly relative parameters θ_r are introduced

$$\theta_r = L^{-1}\theta, \quad L \approx \text{diag}(\hat{\theta}) \quad (11)$$

And

$$\varepsilon_{p,RMS}(\theta_r) = \sqrt{(\theta_r - I)^T H_r(\hat{\theta})(\theta_r - I)} \quad (12)$$

$$I = \{1 \dots 1\}^T$$

where

$$H_r(\theta_N) = L^T H(\theta_N) L \quad (13)$$

The relative sensitivity of $\varepsilon_{p,RMS}(\theta)$ relative to one parameter θ_i is then

$$S_i = \frac{d\varepsilon_{p,RMS}}{d\theta_i / \hat{\theta}_i} = \sqrt{h_{rii}}, \quad h_{rii} = \{H_r(\hat{\theta})\}_{ii} \quad (14)$$

This means that the relative sensitivity of the i th parameter is equivalent to the square root of the i th diagonal element of the relative H -matrix, defined by (13). To obtain accurate parameter estimates, large sensitivities of the individual parameters, is a necessary requirement. But it is not sufficient, due to the fact that the sensitivity of a combination of two or more parameters may be much smaller than the individual

sensitivities, indicating a high correlation between parameters.

Other important definitions in the general $n\theta$ -dimensional space, where the relative sensitivity in the directions of the main axis are determined by an orthogonal transformation to diagonal form

$$\theta_{dr} = T^T \theta_r = T^T L^T \theta \quad (15)$$

$$H_{dr} = T^T H_r(\theta) T = T^T L^T H(\theta) L T$$

Where T is the orthogonal transformation matrix, containing the eigenvectors of H_r as columns.

H_{dr} is a diagonal matrix with the diagonal elements equal to the eigenvalues $\lambda_1 \dots \lambda_d$. Then a measure of the sensitivity variation is calculated:

$$R = \frac{S_{\max}}{S_{\min}} = \frac{\sqrt{\lambda_{\max}}}{\sqrt{\lambda_{\min}}} \quad \lambda_i = \{H_{dr}(\theta)\}_{ii} \quad (16)$$

And the minimum sensitivity of θ_i is:

$$S_{i\min} = \frac{1}{\theta_{i\min}} = \sqrt{\{H_r^{-1}(\theta_N)\}_{ii}^{-1}} \quad (17)$$

As a measure of the correlation between a parameter θ_i and the other parameters is used R_i , defined as:

$$R_i = S_i / S_{i\min} \quad (18)$$

The most characteristic sensitivity measures are outlined in Table I, together with references for their calculation, and their best values for accurate parameter estimation.

TABLE I
 PARAMETER SENSITIVITY MEASURES

Sensitivity	Equations	Best Values
S_{\min}	(15)(16)	As large as possible
$S_{i\min}$	(13)(17)	As large as possible
R	(15)(16)	As close as 1 as possible
R_i	(13)(14) (18)	As close as 1 as possible

B. Frequency-domain considerations

It is interesting and easier to relate physical information and insight to frequency-domain. Thus, a relation between accuracy of parameter estimates is established, as the parameter sensitivity function of the frequency response. Frequency-domain considerations can be applied for linear models or models linearized in the signals.

As the problem is to determine accurate estimates of the model parameters, a relation between parameter accuracy and a frequency-domain expression must be established. This relationship is the parameter sensitivity function of frequency response, because, a necessary condition for accurate

parameter estimates is that the model frequency response is accurate in the frequency range where the parameter sensitivity is high. Thus, the target is to determine the relative parameter sensitivity of the magnitude of $G(j\omega)$ to each parameter θ_i :

$$S_i(\omega) = abs\left(\frac{d|G(j\omega)|}{d\theta_i} \frac{\theta_i}{|G(j\omega)|}\right) \quad (19)$$

C. Procedure for Input Design

Based on previous sensitivity considerations, the following steps for design a good input signal (not optimal) must be developed:

Step 1: Obtain approximative parameter estimates or acquire a priori parameter values.

Step 2: Determine parameter sensitivity in the frequency domain, (simulation). Calculate and plot parameter dependent sensitivities.

Step 3: Choose a class of preliminary input signals with a feasible power spectra. The input signals shall depend on few input signal.

Step 4: Calculate and plot some of the characteristic sensitivity measures as a function of input signal parameters and choose the best values of these according to Table I.

Step 5: Use the determined input signal on the real system, obtaining an improved parameter estimate.

Step 6: If required, repeat step 4-5 or 2-5.

D. Application of the Sensitivity Approach

The sensitivity approach is applied to a Nomoto model defined in (3) with the previous parameter estimation assessed in (4) but considering the yaw rate $r = d\psi/dt$ instead of heading, therefore:

$$\frac{r}{\delta}(s) = \frac{K}{(1+Ts)} = G_{nomoto}(s) \quad (20)$$

The frequency domain sensitivity of $|G_{nomoto}(j\omega)|$ to K and T , are assessed using (19) and are plotted in figure 3.

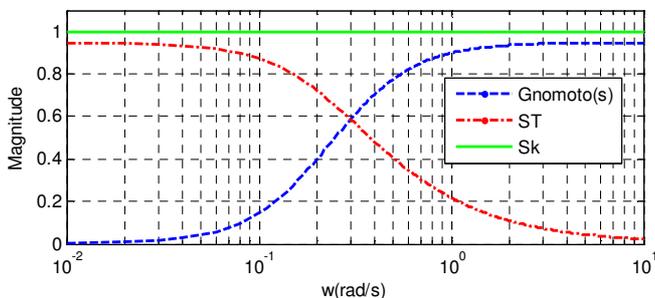


Fig. 3 Magnitude $|G_{nomoto}(j\omega)|$, sensitivity of $|G_{nomoto}(j\omega)|$ to K , (Sk), and sensitivity $|G_{nomoto}(j\omega)|$ to T , (ST).

A square signal is chosen as an input. According to figure 3 it seems to be good to select a high frequency for the input

signal where the sensitivity S_T is greater, e.g. $\omega > 3$; in the case of S_k it doesn't matter what frequency is chosen as long as it is constant for all the frequencies. However, for smaller value of $f = 0.025$ in figure 4 the S_{min} has a maximum, the R a minimum and the $S_{i min}$ have high values and R_i has values close to 1, which obeys the criteria established in Table I.

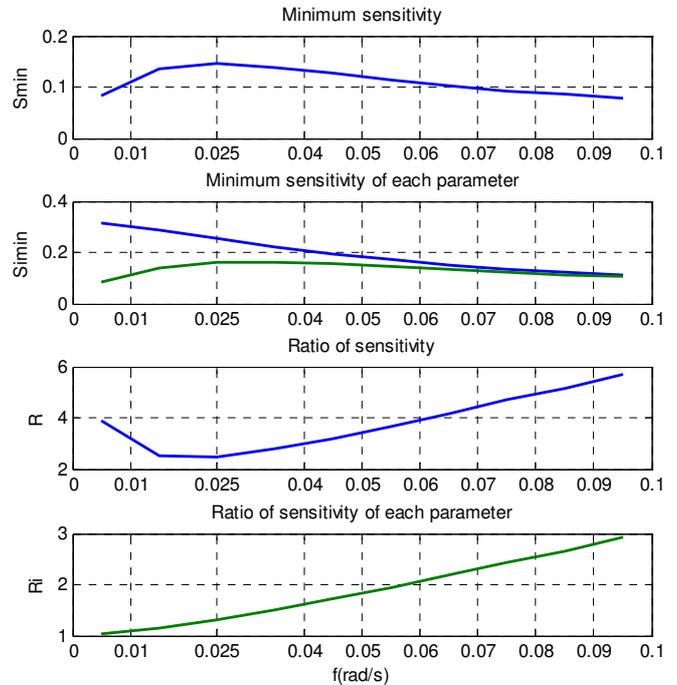


Fig. 4 Sensitivity measures as a function of frequency f of the input signal.

V. TUNING TECHNIQUES

To tune the parameters of a PID controller the diagram of the figure 5 is followed in which the optimization algorithm takes data from the output ψ and from the input ψ_d .

One way to solve this problem is to minimize the error between the output ψ and the input signal ψ_d . The variables are the parameters of the Proportional Integral Derivative (PID) controller. Then, the goal is to minimize the error for all time steps from t_0 (initial instant of time) to t_f (final instant of time), thus producing a multiobjective function (one function for each time step) [18].

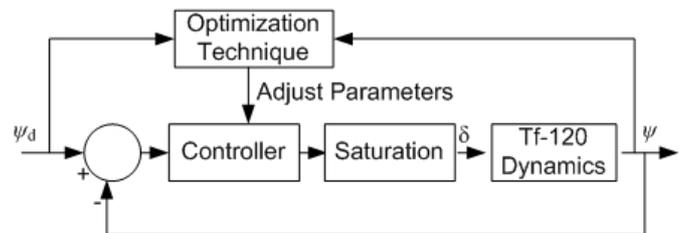


Fig. 5 Close loop diagram for the optimization process.

Another approach to optimizing the control parameters is instead of minimizing the error between the output ψ and the

input signal ψ_d , it is minimized the maximum value of the output ψ at any time t between t_0 and t_f .

A. Initializing the Parameters

As the optimization algorithm needs initial parameters for the first iteration, in this section is presented a classical method to obtain parameters for a PID controller.

With the first order Nomoto model (3) and its parameters (4), a PID controller has been tuned to maintain the constant heading ψ_d .

For the correct functioning of the system, a damping ratio in the range $0.8 \leq \zeta \leq 1$ is selected. ω_n is selected as the rate of turbojets rotation angle ω_δ (rad./sec) and the dynamics $1/T$ (rad./sec) of the identified in-scale model, according to the equation:

$$\frac{1}{T} < \omega_n \sqrt{1 - 2\zeta^2} + \sqrt{4\zeta^4 - 4\zeta^2 + 2} < \omega_\delta \quad (21)$$

Next, a damping value of $\zeta = 0.9$ is selected and the following range of values fitted to ω_n is obtained:

$$0.1128 \frac{\text{rad.}}{\text{sec}} < \omega_n \cdot 0.746 < 2 \frac{\text{rad.}}{\text{sec}} \quad (22)$$

In the selection of the ω_n value, security reasons have been taken into account [1]. In restricted waters and for collision avoidance, the course-changing manoeuvre should have a clear start, in order to show other ships the intention of the manoeuvre and, for that reason, that manoeuvre should preferably completed without overshoot. Therefore, ω_n should not be much greater than 0.6, which would provide a response too much fast. Then, in keeping with equation (32), a value of $\omega_n = 0.6$ rad/sec can be selected.

B. PID Control scheme

The schema of a conventional PID does not work very well when the derivate mode amplifies the noise measurement. To avoid this problem, the following PID control schema is used [14]:

$$\frac{\delta}{E}(s) = K_p \left[1 + \frac{1}{T_i s} + \frac{T_d s}{\alpha T_d s + 1} \right] \quad (23)$$

Equation (23) shows that the derivate action is multiplied by the term $1/(\alpha T_d s + 1)$. The manufacturers give as typical values of α the range of 0.05 to 0.2, so $\alpha = 0.1$ is selected. This can be interpreted as an ideal derivative that is filtered using a first order system with the time constant αT_d . For a small s the transfer function is approximately $K_p T_d s$, and for large s is equal to K_p/α . The approximation acts as a derivative for low-frequency signal components, and the high-frequency gain is limited to K_p/α .

Developing a procedure proposed in [6], the parameters for

the PID controller are obtained:

$$K_p = 1.6133, T_i = 16.666, T_d = 1.3463 \quad (24)$$

Since the results from the tuning, given in the following sections, indicate that the integral time is high, it is interesting to tune a first order network (25) in which the integral action does not appear.

$$\frac{\delta}{E}(s) = K \left[\frac{s+z}{s+p} \right] \quad (25)$$

C. PI-D Control scheme

In the ideal PID controller, if the input ψ_d is a step, due to the derivative term, the output signal δ will have an impulse function. If instead of a pure derivative term $T_d s$, an expression like in equation (23) is used, then the output signal δ will have a pulse function. In order to avoid this phenomenon the derivative term can be included in the feedback (figure 6). In this case the differentiation only occurs with the output signal ψ and not with the input ψ_d .

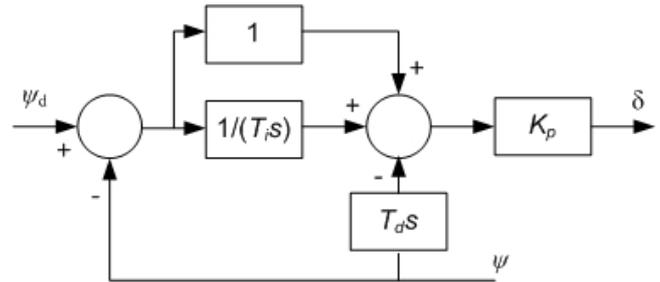


Fig. 6 PI-D control system.

Thus, the manipulated signal will be given by

$$\delta(s) = K_p \left(1 + \frac{1}{T_i s} \right) \psi_d(s) - K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \psi(s) \quad (26)$$

D. Tuning with Genetic Algorithms

In this section the heading controllers are tuned by means of genetic algorithms (GA) [11] so results provided by GA will be compared to results obtained with optimization techniques. In the case of the first order network, parameters calculated by GA will initialize the optimization algorithm.

The aim of the design is that the ship should make a fast course change following, without oscillations, the course determined by the values $\zeta = 0.9$ and $\omega_n = 0.6$ rad/sec in equation (1). The cost function selected was:

$$J(\theta) = \sum_{i=1}^n |\Delta \psi_i + \lambda \delta_i| \quad (27)$$

Where θ is the vector of the controller parameters, n is the total number of iterations in the control system simulations, $\Delta\psi_i$ the i th heading angle error between the desired and obtained heading, λ is a scaling factor ($\lambda = 0.2$ in this case) and δ_i the i th rudder angle deflection. The term δ_i has been included in order to take into account also the minimization of the control effort. A simulation time of 60 sec has been used.

Each individual is represented by a parameter vector $\theta = [K_p \ T_i \ T_d]$ of the PID controller and $\theta = [k \ z \ p]$ for the first order network. The chromosomes are of the binary type. The selection of the range of values of the parameters was performed with a view to avoiding an excessive saturation of the actuators and to ensuring stable controllers.

The ranges of values of the parameters S selected were:

$$S = \{0.1 \leq K_p \leq 10, 1 \leq T_i \leq 5000, 0.1 \leq T_d \leq 10\} \quad (28)$$

for the PID controller and

$$S = \{0.1 \leq k \leq 100, 0.01 \leq z \leq 10, 0.1 \leq p \leq 20\} \quad (29)$$

for the first order network.

For the optimization, a population of 30 individuals over 500 generations is used with a probability of crossover of 50% and mutation of 5%. The GA evaluates the cost function in each iteration, after running the Simulink model, with the controller. A roulette wheel was used in the selection, and the principle of elitism was also used, keeping for the next generation the best two individuals of the previous population, and selecting them for the crossover and mutation.

The resulting parameters of the PID controller (23) are:

$$K_p = 1.21172, T_i = 4951.352255, T_d = 2.300027 \quad (30)$$

And the resulting parameters the first order network in (25) are:

$$k = 32.603814, z = 0.42214, p = 8.696483 \quad (31)$$

E. Non-Linear Least Squares Optimization

As it was introduced before, one of the problems is to minimize the error between the output ψ and the input signal ψ_d . This can be done using least squares optimization in this way:

$$\min_x \|f(x)\|_2^2 = \min_x (f_1(x)^2 + f_2(x)^2 + \dots + f_n(x)^2) \quad (32)$$

where,

$$f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_n(x) \end{bmatrix} = \begin{bmatrix} \psi_1(x) - \psi_{d1}(x) \\ \psi_2(x) - \psi_{d2}(x) \\ \vdots \\ \psi_n(x) - \psi_{dn}(x) \end{bmatrix} \quad (33)$$

and, n = number of time instants from t_0 to t_f

It has been used the matlab function lsqnonlin [13], which solves this kind of nonlinear least-squares problems in equation (15). It has been chosen the large-scale algorithm. This algorithm is a subspace trust-region method and is based on the interior-reflective Newton method described in [3] and [4]. Each iteration involves the approximate solution of a large linear system using the method of preconditioned conjugate gradients (PCG) [5].

Applying this method and following the diagram in figure 5, for time interval from $t_0 = 0$ to $t_f = 160$, the next results are calculated:

$$K_p = 2.5046, T_i = 16.716, T_d = 1.314 \quad (34)$$

The algorithm was initialized with the parameters in (24), after several iterations the parameters obtained in (34) present a small difference between the ones in (24). This is due to the optimization algorithm finds a local minimum near the initialization point.

In the same way the algorithm is applied to optimize the parameters of a 1st order network. In this case, the algorithm is initialized with the parameters calculated with GA in (31):

$$k = 54.389, z = 0.89414, p = 2.5549 \quad (35)$$

These results are different from de initial point in (31) and simulations in the next section will prove the improvement managed with the optimization technique.

F. Minimizing the Maximum Objective

The other approach is to minimize the maximum value of the output ψ at any time t between t_0 and t_f as follows:

$$\min_x \max_i F_i(x) \text{ such that } c(x) \leq 0 \quad (36)$$

where,

$$F_i(x) = \begin{bmatrix} F_1(x) \\ F_2(x) \\ \vdots \\ F_n(x) \end{bmatrix} = \begin{bmatrix} \psi_1(x) \\ \psi_2(x) \\ \vdots \\ \psi_n(x) \end{bmatrix} \quad (37)$$

Taking into account that the constant of time of the system is 4.2497 (see parameters 4), it can be stated the next constrain after the 20 seconds response of the system for a step input signal δ of 20°:

$$\psi(t) \geq 0.95 * 20, 20 \leq t \leq t_f \quad (38)$$

Thus, the output ψ must be greater or equal to 95% of the final value of ψ when it reaches its stationary state. As the constrains are in the form $c(x) \leq 0$ in equation (36), the

expression in (38) must be rearranged in the form:

$$c(x) \leq -\psi(t) + 0.95 * 20, 20 \leq t \leq t_f \quad (39)$$

This problem is solved using the matlab function fminimax. This function reformulates the minimax problem (36) into an equivalent Nonlinear Linear Programming problem by appending additional reformulation constraints of the form $F_i(x) \leq \gamma$ to the constraints given (36), and then minimizing γ over x . It is used a sequential quadratic programming (SQP) method [8] to solve this problem.

As it was done in section V.C the algorithm gets these results for the PID controller:

$$K_p = 98.055, T_i = 240.18, T_d = 0.71093 \quad (40)$$

Figure 7 shows the results, in each iteration, given by matlab. Since the 18th iteration, the objective function converges to 20.25, very close value to the input signal $\psi_d = 20^\circ$. Moreover, the directional derivative is very close to zero, which means that a minimum is reached.

For 1st order network the results are:

$$k = 54.952, z = 0.98296, p = 5.5161 \quad (41)$$

From the 12th iteration in figure 8, the objective function converges to 20, the value of the input signal $\psi_d = 20^\circ$. Then, the error between the response of the output ψ and the input ψ_d is zero.

Iter	F-count	Objective value	Max constraint	Line search steplength	Directional derivative
0	5	0	24.1086		
1	11	21.5	1.532	1	21.5
2	17	22.01	0.617	1	0.518
3	23	22.13	-0.01256	1	0.113
4	29	22.01	-0.002174	1	-0.119
5	35	21.81	0.0533	1	-0.203
6	42	21.81	0.0268	0.5	0.00173
7	52	21.79	0.04332	0.0625	-0.182
8	58	21.16	0.1234	1	-0.638
9	64	21.06	0.02684	1	-0.0948
10	70	21.02	0.003785	1	-0.0418
11	76	20.74	0.2441	1	-0.283
12	82	20.34	0.1965	1	-0.401
13	88	20.32	0.1618	1	-0.0121
14	94	20.19	0.2049	1	-0.136
15	100	20.3	0.2778	1	0.116
16	106	20.15	0.1037	1	-0.153
17	112	20.23	0.06968	1	0.075
18	118	20.25	0.007371	1	0.0209
19	124	20.25	0.0422	1	-5.73e-005
20	130	20.25	0.03263	1	-5.73e-005
21	136	20.25	0.03135	1	0.00367

Fig. 7 Tuning a PID with with optimization technique (fminimax) and genetic algorithm.

As well as before, the directional derivative is very close to zero.

Iter	F-count	Objective value	Max constraint	Line search steplength	Directional derivative
0	5	0	20.0003		
1	11	28.43	2.598	1	28.4
2	17	23.89	-0.8699	1	-4.54
3	23	21.21	0.5008	1	-2.68
4	29	20.39	1.055	1	-0.826
5	35	20.9	0.02136	1	0.514
6	41	20.74	0.000423	1	-0.163
7	49	20.63	0.09675	0.25	-0.443
8	55	20.68	0.01436	1	0.051
9	61	20.03	0.1387	1	-0.653
10	67	20.04	0.02673	1	0.0171
11	74	20.02	0.01596	0.5	-0.0432
12	80	20	0.001379	1	-0.0223
13	86	20	-5.769e-008	1	-1.56e-005

Fig. 8 Tuning a 1st order network with optimization technique (fminimax) and genetic algorithms.

VI. 2-DIMENSIONAL LOS GUIDANCE SYSTEM

Systems for guidance are systems consisting of a waypoint generator with human interface. One solution to design this system is to store the selected way points in a way-point database and use them to generate a trajectory (path) for the ship Fig. 9. Other systems can be linked to this waypoint guidance system as the case of weather routing, collision and obstacle avoidance, mission planning, etc.

LOS schemes have been applied to surface ships by [7, 12]. In this methodology it is computed a LOS vector as the vector from the body-fixed origin (x, y) to the next way-point (x_k, y_k) . This suggests that the set-point for the heading autopilot should be chosen as:

$$\psi_d(t) = \tan^{-1} \left(\frac{y_k - y(t)}{x_k - x(t)} \right) \quad (42)$$

where (x, y) is the vessel position usually measured with a GPS. In this article, the position of the TF-120 model is calculated with its kinematical equations [6] for constant speed, so $u = u_0 = 0.4\text{m/sec}$. Equation (42) requires a sign test to ensure that $\psi_d(t)$ is in the proper quadrant. The autopilot follows the heading by guiding the TF-120 model from way-point to way-point.

When moving along the path a switching mechanism for selecting the next way point is needed. The way-point (x_{k+1}, y_{k+1}) can be selected on a basis of whether the ship lies within a circle of acceptance with radius R_0 around the way point (x_k, y_k) . Moreover if the vehicle positions $(x(t), y(t))$ at time t satisfy:

$$[x_k - x(t)]^2 + [y_k - y(t)]^2 \leq R_0^2 \quad (43)$$

A guideline could be to choose R_0 equal to two ship lengths (L_{pp}), in the case of the in-scale physical model TF-120 model $L_{pp} = 4.4\text{m}$.

VII. COMPUTER SIMULATIONS

Computer simulations are performed in Simulink, in order to indicate the improvement of the response of the system ψ ,

using optimization techniques. In the simulations the optimization technique is compared to GA technique. After that, simulations of a track-keeping system are made with different structures of heading controllers and then the best heading controller is selected for the track-keeping system.

The Tuning of a 1st order network with optimization technique (lsqnonlin) in figure 10 shows a faster response than

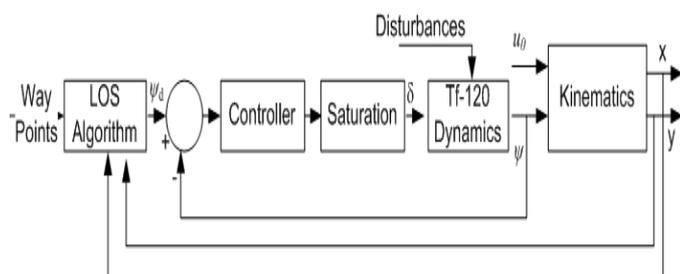


Fig. 9 Conventional LOS Guidance system.

the one with GA and small overshoot around 5%.

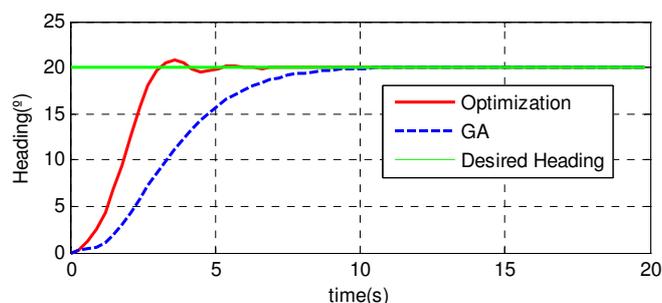


Fig. 10 Tuning a 1st order network with the optimization technique (lsqnonlin) and genetic algorithms.

In figure 11 a PID controller is tuned with the optimization technique (fminimax), where the response is faster than with GA and without any overshoot and oscillations.

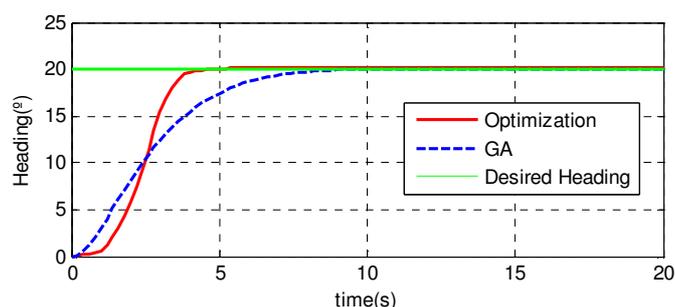


Fig. 11 Tuning a PID with the optimization technique (fminimax) and genetic algorithms.

A 1st order network with the optimization technique (fminimax) is performed in figure 12; in this simulation the response is faster than with GA and it hasn't overshoot and oscillations. This is the best result due to the response with (fminimax) for a 1st order network is faster than in figure 10 with (fminimax) for a PID and it hasn't overshoot as in

figure 9 with (lsqnonlin) for a 1st order network. Therefore, the best election is to tune a 1st order network with (fminimax).

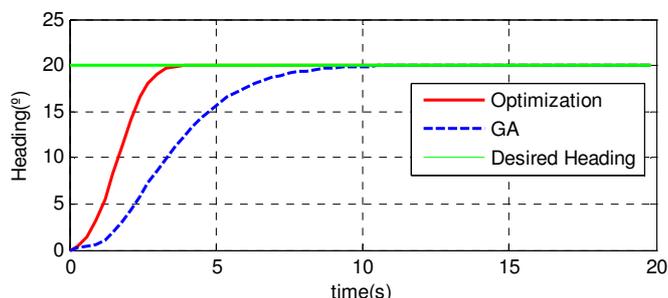


Fig. 12 Tuning a 1st order network with the optimization technique (fminimax) and genetic algorithms.

The next simulations are performed to show the behavior of different types of heading controllers in a track-keeping system. In this case, It has been simulated a realistic level of noise measurement for the heading with a white noise additive input, which its mean value is zero and its variance is 1.

Figure 12 exhibits the behavior of the PID controller with a derivative term equal to $T_d s$ which presents a large overshoot at $t = 208$ sec and at $t = 366$ sec. As it was explained in section V.C, this problem is corrected with the PI-D scheme where the derivative term is included in the feedback (figure 6) to avoid an impulse function in the output signal δ . Figure 13 presents a way point guidance by line of sight where the vehicle mission is given by a set of way points $x_d(k) = [20 -20 -30 20 0]$ $y_d(k) = [25 30 40 70 80]$. It is appreciated, for the PID controller, how the large overshoot in the heading control causes a strange behavior in the guidance around the way point $[x_d(k), y_d(k)] = [-30, 40]$, meanwhile for the PI-D control the guidance seems to be good. The heading controller with a filtered derivative term (23) does not present a large overshoot, that is why the first order term attenuates high frequencies.

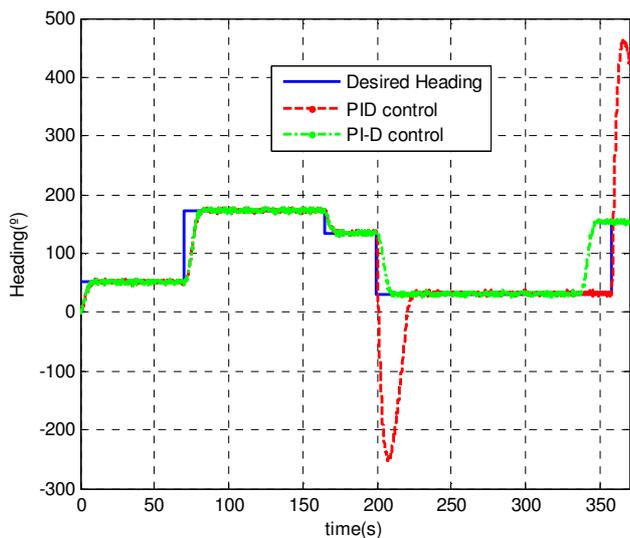


Fig. 13 Heading control of the track-keeping system using equation (26) as a switching mechanism.

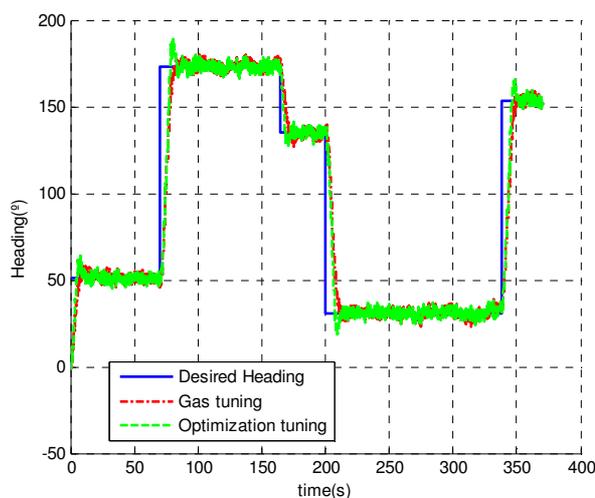


Fig. 15 Heading control of the track-keeping system using equation (26) as a switching mechanism for a first order network heading controller.

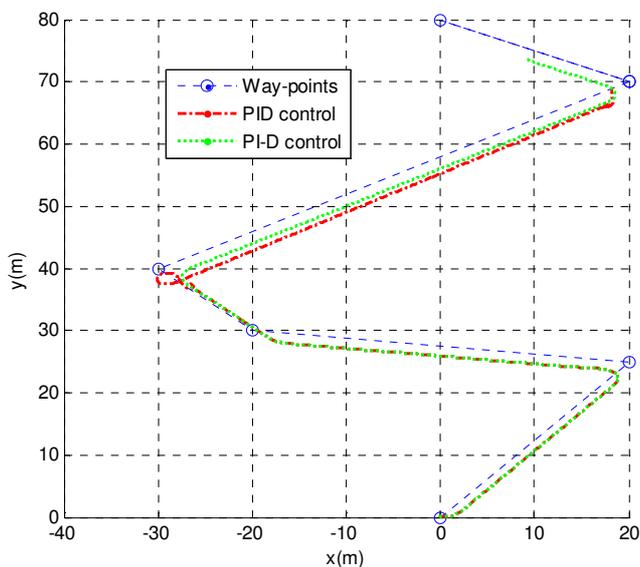


Fig. 14 xy plot of the simulated trajectory of the TF-120 model and desired geometrical path made up of way-points.

In the simulations (figure 15 and 16) a guidance system is shown, using a first order network as a heading controller tuned with the optimization technique (fminimax) as it presented the best behavior in the previous simulations. In the guidance simulation (figure 16) the trajectory described by the system tuned with GA is very similar to the one with the Optimization tuning due to the fact that in both situations the error in the heading control is close to zero. The difference is in the heading response, in the case of optimization tuning the response is faster but it has a bigger overshoot comparing with the GA tuning. In a real case, the changes of heading are not as big as in figure 15. In restricted waters, where the marine traffic is considerably high, the changes of heading are highly likely around 20° like in figure 12 so that the ship describes a soft trajectory. Thus, the improvement managed with the tuned first order network is very interesting when the changes of

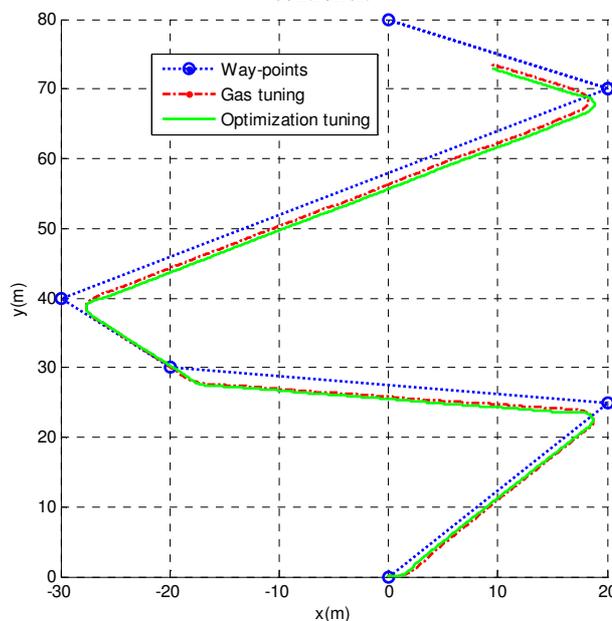


Fig. 16 xy plot of the simulated trajectory of the TF-120 model and desired geometrical path made up of way-points. for a first order network heading controller.

course are small, because the response is faster and without overshoot.

VIII. CONCLUSIONS

Heading controllers have been tuned with optimization techniques such as: Non-linear least squares optimization and minimizing the maximum objective. With these techniques results from GA are improved.

The heading control is based on a model obtained from data provided by a platform for marine vehicles.

Simulations have been carried out to verify which control structure and which optimization method present the best

results. Moreover, the possibilities of the platform for experimentation of marine vehicles with research and educational purposes are presented.

In this section is presented a conventional LOS guidance system simulation. It has been used a first order network as a heading controller with the parameters calculated in previous sections. It is compared the tuning with GA method to the optimization method (fminimax) and it is exhibited how the optimization has a faster response without overshoot for small changes of heading, which is a typical case of the manoeuvres developed in restricted waters.

ACKNOWLEDGMENT

This paper has been partially supported by MEC with the project DPI2006-11835.

REFERENCES

- [1] Amerongen J. V., *Adaptative Steering of Ships: A model-reference approach to improved manoeuvring and economical course keeping*, PhD thesis, 1982.
- [2] Aranda, J. González de Santos, P. and De la Cruz, J.M. (Ed.) *Robotics and Automation in the Maritime Industries*, Chapter 12: Platform for remote experimentation of autonomous high speed craft models with a combined technology for Wi-Fi and internet communications programmed in LabVIEW, UCM, Madrid, Spain. 2006, pp 275-293.
- [3] Brayton, R.K., S.W. Director, G.D. Hachtel, and L.Vidigal, "A New Algorithm for Statistical Circuit Design Based on Quasi-Newton Methods and Function Splitting," *IEEE Trans. Circuits and Systems*, Vol. CAS-26, pp. 784-794, Sept. 1979.
- [4] Coleman, T.F. and Y. Li, "An Interior, Trust Region Approach for Nonlinear Minimization Subject to Bounds," *SIAM Journal on Optimization*, Vol. 6, pp. 418-445, 1996.
- [5] Coleman, T.F. and Y. Li, "On the Convergence of Reflective Newton Methods for Large-Scale Nonlinear Minimization Subject to Bounds," *Mathematical Programming*, Vol. 67, Number 2, pp. 189-224, 1994.
- [6] Fossen, T. I., *Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles (Marine Cybernetics AS)*. Trodheim, Norway, 2002.
- [7] Fossen, T. I. Breverik, M., Skjetne, R., "Line of sight path following of underactuated marine craft". in: *Proc. 6th IFAC Conf. on Maneuvering and Control of Marine Crafts*, Girona, Spain, 2003, pp. 244-249.
- [8] Jorge Nocedal, Stephen J. Wright. *Numerical optimization*, Springer, 2006.
- [9] Knudsen M., *A Sensitivity Approach for Estimation of Physical Parameters*, SYSID'94 Copenhagen Denmark, 1994.
- [10] Knudsen M., *Estimation of Physical Parameters in Linear and Nonlinear Dynamic Systems*. PhD-report, Aalborg University, 1993.
- [11] Man, K.F., Tang, K.S., Kwong, S., *Genetics Algorithms: concepts and designs*. Springer, 1999.
- [12] McGookin, E.W., Murray-Smith, D.J., Lin, Y., Fossen, T.I., "Ship steering control system optimization using genetic algorithms", *Journal of Control Engineering Practice*, Aug. 1998, pp. 429-443
- [13] *Optimization Toolbox™ 4 User's Guide*, mathworks, 2007.
- [14] Smith, C. A., Corripio, A. B., *Principles and Practice of Automatic Process Control*, John Wiley & Sons, Canadá, 1997.
- [15] Velasco, F. J., Rueda, T. M., Revestido, E., Esquibel, L. Á. *Marine Vehicles Web-Wi-Fi Platform for Remote Experimentation Programmed in LabVIEW*, 'ACE 06' 7th IFAC Symposium on Advances in Control Education, Madrid, 2006.
- [16] Velasco, F. J., Rueda, T. M., Revestido, E., Esquibel, L. Á. *Platform of Trials in an Open Environment for the Cooperation of Marine Vehicles Programmed in Labview with Wireless Connectivity by Means of Wifi.*, WSEAS/IASME Int. Conf. on Electrosience and Technology for Naval Engineering and All-Electric Ship, Miami, 2005.
- [17] Velasco, F. J., Revestido, E., López E., Moyano E., Haro Casado M., *Autopilot and Track-Keeping Simulation of an Autonomous In-scale Fast-ferry Model*, 12th WSEAS International Conference on SYSTEMS. Heraklion- Crete, Greece, 2008.
- [18] Velasco, F. J., Revestido, E., López E., Moyano E., Haro Casado M., "Optimization techniques for Tuning Heading Controllers of an Autonomous In-scale Fast-ferry Model", 8th WSEAS International Conference on Simulation, Modelling and Optimization (SMO'08), Santander, Spain, 2008.
- [19] Velasco, F. J., Rueda, T. M., Revestido, E., Esquibel, L. Á., "Turning Circles of an Autonomus High Speed Craft Model", in 5th WSEAS Int. Conf. on System Science and Simulation in Engineering. 1st WSEAS International Symposium on Advances in Naval Science, Research and Education. Tenerife, Spain, 2006.
- [20] Velasco, F. J., Revestido, E., López E., Moyano E., Haro Casado M., "Simulations of an Autonomous In-scale Fast-ferry Model", *International Journal of Systems Applications, Engineering & Development*, issue 3, Volume 2, 2008.