Position tracking of container ship using evolutionary-based robust controller design

Kamran Alishaei[1*]

Abstract: **In recent years, there has been sustained interest in the development of robust controllers for the waypoint tracking problem of a container ship. Although highly efficient, high-order robust controllers may not be feasible for real-time implementation due to hardware and computational limitations. This paper presents an evolutionary algorithm–based robust controller design for the waypoint tracking system of a container ship. The developed algorithm enables the implementation of simple-structure controllers satisfying robust stability and performance requirements. The performance of controller is validated and compared with that of existing approaches using the ship steering system. Simulations demonstrate the effectiveness of the proposed design.**

Keywords: **Container ship, Tracking system, μ-synthesis, Evolutionary algorithm, H_∞ control**

I Introduction

The purpose of a motion control problem is to make the ship track a desired trajectory. For the motion control of ships, various control goals may be considered, such as trajectory tracking, path following, and way-point tracking [1]. The aim of trajectory tracking is to design appropriate control laws such that the ship tracks a desired time-parameterized trajectory. Path following requires the path to be independent of time and expressed in terms of its geometric description [2,3]. In the way-point tracking control problem, it is necessary to guide the ship using a series of waypoints between the starting point and the desired final position. Recently, various studies have been reported to address the motion control problem of container ships [4–10]. While moving in oceans, ship motions are frequently influenced by exogenous environmental disturbances, such as winds, waves, and ocean currents, resulting in uncertainties [11]. Thus, the most significant objectives in the field of tracking are to ensure stability and performance in the presence of such uncertainties [12]. Robust control techniques are the most widely known approaches to achieving these goals. The key technique most frequently applied for designing robust controllers is μ-synthesis [13]. μ-synthesis, which employs the structured singular value, can guarantee robust stability and performance against different types of uncertainties [14, 15]. This synthesis requires a linearized model of the system and may include several weighting functions for shaping the exogenous signals and representing the performance specifications of the closed-loop system. The results reported in the literature confirm that this method provides significantly better results than conventional classical controllers [16]. Generally, μ-synthesis controllers lead to high-order controllers that may not be feasible for real-time applications due to hardware and computational limitations.

[1] Marun Petrochemical Company (MPC), Mahshahr, Iran
* Corresponding Author, Email: kalishaei@mpc.ir

The $D-K$ iteration algorithm, as the most well-known approach for solving the μ-synthesis problem, suffers from a major disadvantage: it may fail to converge in some cases, leading to suboptimality of the designed controller [17]. Consequently, the selection of control structures and controller parameters constitutes the main topic that motivates the present research. To design a robust controller that satisfies the performance specifications, the control problem is first formulated as a constrained optimization problem with a predefined controller structure, such as a Proportional-Integral-Derivative (PID) controller [18]. Evolutionary Algorithms (EAs) are then adopted to determine the controller parameters. High-dimensional optimization problems have been successfully tackled by EAs rather than by manual parameter tuning. Several EAs, such as genetic algorithms, ant colony optimization, and particle swarm optimization, have received considerable attention for solving multi-objective optimization problems [19–22].

Motivated by these considerations, the contribution of this paper is to design a swarm-based robust controller for position tracking of a container ship in the presence of uncertainties and external disturbances. The main advantage of the proposed algorithm is its ability to design robust controllers with a simple structure by formulating a constrained optimization problem solved via EAs. This significantly reduces the controller order. The performance of the designed controller is validated through comparison with the $D-K$ iteration, a common approach for solving the μ-synthesis problem, and with an H_∞ controller. Simulation results demonstrate the satisfactory performance of the proposed control algorithm in the position tracking problem.

The rest of this paper is organized as follows. Section II states the preliminaries. Section III describes the modeling of the container-ship used in this paper. Section IV elaborates the proposed control algorithm. Section V gives the simulation results. Finally, Section VI outlines the main conclusions.

II Preliminaries

In this section, first, the control system topology is described. Then, the principle of μ-synthesis is demonstrated.

A Control System Topology

Fig. 1 presents a general framework for robust controller design considering multiplicative uncertainty. In this framework, $G(s)$ and $\tilde{G}(s)$ denote the nominal and real models of the system, respectively; $W_\Delta(s)$ is a weighting function with a fixed stable transfer function; $\Delta(s)$ is a stable transfer function satisfying $\|\Delta\|_\infty \leq 1$; and $K(s)$ is the fixed-structure controller, r, d, e, y, and u represent the input, disturbance, error, output, and control effort signals, respectively. The weighting functions W_p and W_u correspond to the performance and control input indices, respectively, and the corresponding weighted signals of these variables are denoted by $\tilde{e}$ and $\tilde{u}$. Fig. 1 can be represented by the general control configuration shown in Fig. 2. In Fig. 2, P is the generalized plant, K is the controller, and w and z are the bi-dimensional input and output vectors used in the Linear Fractional Transformation (LFT) structure definition. The diagonal matrix Δ, which satisfies $\|\Delta\|_\infty \leq 1$, contains the feasible uncertainties. To analyze the system for a given controller, the M structure can be used, as depicted in Fig. 2b. The block N, formulated as a lower LFT, is expressed as

$$N = F_l(P, K) = P_{11} + P_{12}K(I - P_{22}K)^{-1}P_{21} \tag{1}$$

An upper LFT representation can also be applied to the M structure, as illustrated in Fig. 2b. Here, the corresponding transfer function from w to z is defined as [23].

$$\begin{aligned} z &= F_u(N, \Delta)w, \\ F_u(N, \Delta) &= N_{22} + N_{21}\Delta(I - N_{11}\Delta)^{-1}N_{12} \end{aligned} \tag{2}$$

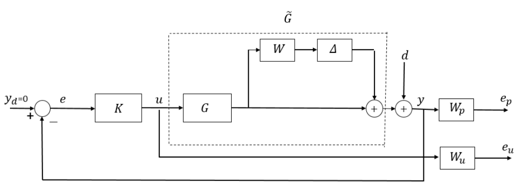
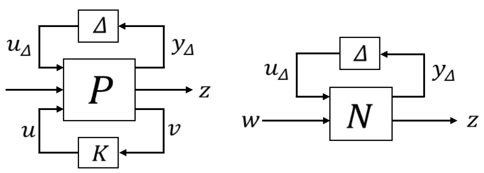


Figure 1. General closed-loop framework.



(a) Synthesis (b) Analysis

Figure 2. General control topology.

B Principal of μ-Synthesis

In the μ-synthesis control framework, the structured singular value μ is used to measure the robustness of the system under consideration. A key advantage is its ability to assess robustness in terms of both stability and performance simultaneously. Let Δ denote the structured uncertainty, composed of an n-th order block diagonal matrix, as

$$\begin{aligned} \mathbf{\Delta} = \{&\mathrm{diag}[\delta_1 I_{r_1}, \ldots, \delta_s I_{r_s}, \Delta_1, \ldots, \Delta_g], \\ &\delta_i \in \mathbb{C}, \Delta_j \in \mathbb{C}^{m_j \times m_j}\} \end{aligned} \tag{3}$$

where $\mathbb{C}$ denotes the set of complex numbers, s denotes the number of scalar blocks, and g denotes the number of full complex blocks. The structured singular value, denoted by $\mu_\Delta(N)$, is then computed as

$$\mu_\Delta(N) = \frac{1}{\min\{\sigma_{\max}(\Delta) : \det(I - N\Delta) = 0\}} \tag{4}$$

According to (4), $\mu_\Delta(N)$ is a measure of the smallest Δ that causes the system shown in Fig. 2a to become unstable. Within the framework shown in Fig. 2b, and using μ_Δ, the objectives of the controller design can be stated as follows:

$$\begin{aligned} \mathrm{RS} &\iff \mu_\Delta(N_{11}) < 1, \quad \forall\omega \\ \mathrm{RP} &\iff \mu_{\tilde{\Delta}}(N) < 1, \quad \forall\omega \end{aligned} \tag{5}$$

where RS and RP stand for robust stability and performance, respectively. Moreover, $\tilde{\Delta} = \mathrm{diag}(\Delta, \Delta_F)$, in which Δ_F is an artificial perturbation matrix representing the H_∞ performance specification [24].

C D-K Iteration Algorithm

Owing to the presence of multiple local minima in the optimization control problem, (4) is inadequate for directly calculating μ [24]. The $D-K$ iteration algorithm is the most common approach to tackling this problem, utilizing the upper bound of μ, which is expressed as

$$\mu_\Delta(N) \leq \sigma_{\max}(DND^{-1}) \tag{6}$$

where $D = \mathrm{diag}(d_1 I_{r_1}, d_s I_{r_s}, d_1 I_{m_1}, d_F I_{m_F})$, $d_i > 0, d_j > 0$. The $D-K$ iteration algorithm, formulated as a convex optimization problem at each frequency ω, is expressed by

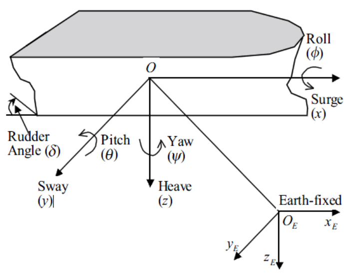


Figure 3. Ship dynamics with 6 DOF motions.

$$\inf_{K(s)} \sup_{\omega} \sigma_{\max}(D(j\omega)F_l(P,K)D^{-1}(j\omega)) \tag{7}$$

where $K(s)$ is a given fixed-structure controller. Indeed, (7) is an alternative method for solving the problem formulated in (8).

$$\inf_{K(s)} \|DF_l(P,K)D^{-1}\|_\infty \tag{8}$$

The procedure of the $D-K$ algorithm can be summarized as follows:
Step 1: Set $D = I$.
Step 2: Minimize $\|DF_l(P,K)D^{-1}\|_\infty$ with D fixed.
Step 3: Compute D to minimize $\sigma_{\max}(DF_l(P,K)D^{-1})$ at each frequency for a fixed K.
Step 4: Find a stable minimum-phase transfer function via curve fitting over the magnitude of each element of $D(j\omega)$.
Step 5: Return to Step 2 and iterate until the stopping criterion is satisfied, namely, $\sup_\omega \mu_\Delta < 1$.

It is worth noting that the major drawback of the $D-K$ iteration is that the resulting controller may be non-optimal [17]. To address this issue, it is necessary to introduce an alternative approach for solving the μ-synthesis problem.

III Container Ship Modeling

A Motion Equations

Fig. 3 illustrates the ship dynamics with six-degree-of-freedom motions, where x, y, z, ϕ, θ, and ψ denote the surge, sway, heave, roll, pitch, and yaw motions, respectively, while xyz and $x_E y_E z_E$ represent the body-fixed and earth-fixed reference frames. The equations of motion for the ship can be written as [10]:

$$M\dot{v} + C(v)v = \tau \tag{9}$$

where M represents the inertia matrix, $v = [u,v,w,p,q,r]^T$ denotes the linear and angular velocity vector in the body frame, $C(v)$ stands for the Coriolis and centripetal effects, and τ is the control signal vector. It is important to note that in practice, only the ship's motion on the horizontal plane is considered. Here, we assume that

A.1. The origin of the body frame lies on the ship's centerline.
A.2. The mass distribution is assumed to be homogeneous.
A.3. The xz-plane is assumed to be symmetric.
A.4. The modes corresponding to roll, pitch, and heave are ignored.

The nonlinear motion equations corresponding to surge, sway, and yaw, , respectively, are given by [2].

$$m(-vr - x_G r^2 + \dot{u}) = X \tag{10}$$
$$m(x_G \dot{r} + \dot{v} + ur) = Y \tag{11}$$
$$m x_G(\dot{v} + ur) + I_z \dot{r} = N \tag{12}$$

where m stands for the mass of the ship, u and v represent the velocities associated with surge and sway, respectively, r denotes the yaw rate, X and Y are the components of the hydrodynamic forces along the x- and y-axes, respectively, x_G is the center of gravity, I_z is the moment of inertia about the z-axis, and N denotes the z-component of the hydrodynamic moment. The hydrodynamic forces and moment are described by

$$X = f_1(u,v,r,\dot{u},\dot{v},\dot{r},\delta) \tag{13}$$
$$Y = f_2(u,v,r,\dot{u},\dot{v},\dot{r},\delta) \tag{14}$$
$$N = f_3(u,v,r,\dot{u},\dot{v},\dot{r},\delta) \tag{15}$$

The environmental disturbances, including wind, waves, and ocean currents, directly affect the motion of the ship, leading to perturbations. Accordingly, it is necessary to include these uncertainties in the model description in order to design an appropriate controller. In what follows, the impact of these disturbances is described. The combined effects can be considered using the superposition principle.

B Wind Impact

The wind can generate the forces and moments, which are expressed as [25]:

$$X_a = (1/2)\,\rho_A C_x(\theta_a) A_T V_A^2 \tag{16}$$
$$Y_a = (1/2)\,\rho_A C_y(\theta_a) A_L V_A^2 \tag{17}$$
$$N_a = (1/2)\,\rho_A C_n(\theta_a) A_L V_A^2 \tag{18}$$

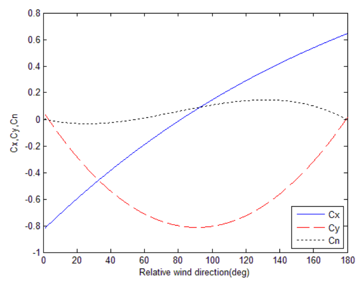


Figure 4. Wind force and moment coefficients vs. relative wind direction.

where ρ_A denotes the density of air, A_T and A_L denote the transverse and longitudinal projected areas, respectively, γ_A and V_A stand for the relative direction and speed of the wind, respectively, and C_x, C_y, and C_n are the force and moment coefficients along the X, Y, and N directions, respectively. Fig. 4 displays the variations of C_x, C_y, and C_n with respect to the relative wind direction.

C Ocean Current

Here, a two-dimensional current model with two parameters, namely the average current speed V_c and the current direction γ_c, is adopted. The irrotational current velocities in the body-fixed frame are expressed as [25]:

$$\begin{aligned} v_c &= V_{0c}\sin(\gamma_c - \psi), \\ u_c &= V_{0c}\cos(\gamma_c - \psi) \end{aligned} \tag{19}$$

satisfying

$$\dot{V}_{0c}(t) + \mu_0 V_{0c}(t) = l(t) \tag{20}$$

in which $\mu_0 \geq 0$ and $l(t)$ is a Gaussian white noise sequence with zero mean.

D Wind Generated Waves

The wave's motion can be formulated as [2]:

$$\psi_H(s) = \frac{K_w s}{(s^2 + 2\zeta\omega_0 s + \omega_0^2)w_H(s)} \tag{21}$$

where $K_w = 2\zeta\omega_0\sigma_w$ is a constant parameter, σ_w is the wave intensity, $w_H(s)$ is Gaussian white noise with zero mean, ω_0 is the wave frequency, and ζ is the damping coefficient. The state-space representation of (21) is given by

$$\begin{aligned} \dot{\zeta}_H &= \psi_H \\ \dot{\psi}_H &= -2\zeta\omega_0\psi_H - {\omega_0}^2 - \zeta_H + K_w w_H \end{aligned} \tag{22}$$

In the context of ship maneuvering, the wave frequency ω_0 is modified as follows:

$$\omega_e(U, \omega_0, B) = \omega_0 - \frac{\omega_0^2 U}{g}\cos(B) \tag{23}$$

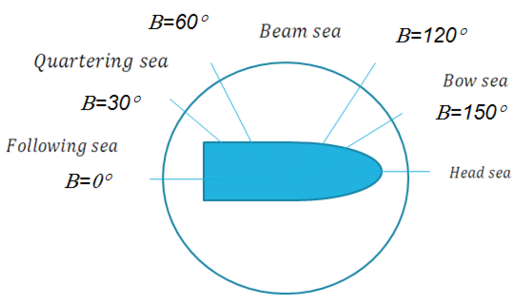


Figure 5. Definition of heading angle μ.

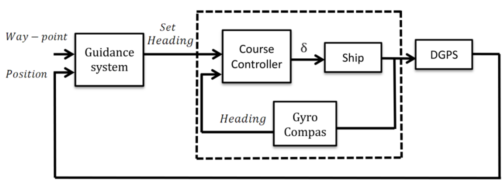


Figure 6. Track keeping system architecture.

where ω_e is the encounter frequency, g is the gravitational acceleration, U is the total speed of the ship, and B is the angle between the direction and the heading, as defined in Fig. 5.

E Tracking Systems

There are two control loops, namely guidance and control, in ship track-keeping systems, as illustrated in Fig. 6. In these systems, waypoints are defined to construct the ship trajectory. Line-of-sight (LOS) guidance is the most widely used path control approach, wherein the desired heading angle is computed based on the relative position between the ship and the next waypoint, as given by [8].

$$\psi_d(t) = \tan^{-1}(\theta) \tag{24}$$

where $\theta = \frac{y_d(k)-y(t)}{x_d(k)-x(t)}$. The key task is to select ψ_d in the appropriate quadrant. If the ship lies within an acceptance circle with radius ρ_0 around $(x_d(k), y_d(k))$, then the subsequent waypoint can be determined by satisfying the following relationship for the ship's position at time t. Note that a typical choice for the radius is $\rho_0 = 2L$.

IV Proposed Control Algorithm

In this section, evolutionary algorithms (EAs) are first briefly described. Without loss of generality, the Particle Swarm Optimization (PSO) algorithm is selected due to its simple structure and few control parameters. Then, the procedure for controller design is presented.

A Optimization Algorithm

Evolutionary algorithms (EAs) have been successfully applied to a wide range of complex real-world problems. Their primary strengths lie in their nature-inspired mechanisms, stochastic search behavior, and independence from the gradient of the objective function [26]. Among EAs, particle swarm optimization (PSO) is one of the most widely known algorithms, inspired by the social foraging behavior of bird flocks. At each iteration, every particle is updated based on historical information, specifically its personal best position and the global best position of the entire swarm. The best position found by a given particle up to the current generation is termed the personal best p_{best}, whereas the best position discovered by the swarm across all previous generations is the global best g_{best}. The velocity and position update rules for the i-th particle are given as follows [27].

$$\begin{aligned} V_i(k+1) &= \omega V_i(k) + c_1 r_1 (p_{\text{best}_i} - X_i(k)) \\ &\quad + c_2 r_2 (g_{\text{best}} - X_i(k)) \\ X_i(k+1) &= X_i(k) + V_i(k+1) \end{aligned} \tag{25}$$

where k is the current generation, ω is the inertia weight, c_1 and c_2 are positive constant parameters, and r_1 and r_2 are uniformly distributed random variables in the interval $[0, 1]$.

The basic PSO offers significant advantages, such as fast convergence. However, it suffers from a lack of a systematic mechanism for velocity control. The inertia weight plays a key role in balancing the exploitation and exploration capabilities of the swarm. A smaller inertia weight results in faster convergence of the particles, which may lead to entrapment in a local optimum. Therefore, various PSO variants that adapt the inertia weight during the search process have been introduced, including linearly decreasing, nonlinearly decreasing, and time-varying inertia weight strategies [28–32]. The choice of a particular variant directly depends on the problem under study, taking into account factors such as computational burden. In the problem at hand, computational burden is a significant concern. Compared with the aforementioned PSO variants, the Linearly Decreasing Weight PSO (LDW-PSO) exhibits superior performance in terms of both solution quality and computational efficiency. Accordingly, this variant is adopted in the present paper, with the following linear adaptation law for the inertia weight ω [33]:

$$\omega_k = \omega_{\min} + \frac{iter_{\max} - k}{iter_{\max}}.(\omega_{\max} - \omega_{\min}) \tag{26}$$

where the maximum number of iterations is denoted by $iter_{\max}$, and $\omega_{\min}$ and $\omega_{\max}$ are the minimum and maximum values of ω, respectively.

B Evolutionary-Based Controller Design

Robustness is a crucial property that any control system must possess, particularly when designing low-order robust controllers. In this paper, we develop an EA-based robust controller design using μ-synthesis to achieve this objective. The μ-synthesis problem is formulated as a constrained optimization problem. The controller parameters are evolved using EAs based on an appropriately defined objective function, where the dimension of the search space corresponds to the number of controller parameters. As mentioned earlier, we adopt the PSO algorithm to solve this problem. The first step is to define the objective function along with the associated constraints, as explained below. The objective function and the constraint are defined based on robust performance (RP) and robust stability (RS), respectively, quantified via μ-performance and μ-stability analyses. Accordingly, we define the following objective function:

$$\begin{aligned} &\min_{K(s)} \ \|N(s)\|_\mu \\ &\text{subject to } \|N_{11}(s)\|_\mu < 1 \end{aligned} \tag{27}$$

The second step is to define the control structure, i.e., a fixed-structure controller. The controller parameters are then determined using an EA (i.e., PSO) to minimize the objective function in (27). The proposed control algorithm is summarized in the following steps:
Step 1: Specify the proper fixed-structure controller.
Step 2: Initialize the EA parameters.
Step 3: Compute p_{best}, g_{best}, and the objective function corresponding to the RP for each individual.
Step 4: Update ω based on the iteration number using (26).
Step 5: Update the velocities and positions of each individual using (25) and (26), respectively.
Step 6: If the new position does not satisfy the stability condition, retain the previous position; otherwise, adopt the new one.
Step 7: Check the termination criterion. If the stopping criterion is met, the best controller parameters are obtained. Otherwise, return to Step 3.
Step 8: Verify the performance of the obtained controller. If it is satisfactory, stop. Otherwise, increase the controller order and return to Step 1.

In the next section, we carry out the controller design and verify whether these criteria are satisfied.

V Simulations And Discussions

A Experimental Set up

In this section, a robust fixed-structure controller is designed for the ship tracking problem. The controller parameters are determined by solving a constrained optimization problem using MATLAB's Robust Control Toolbox. We adopt a PID controller as the fixed-structure form, which is expressed as follows [34]:

$$K(s) = K_1 + \frac{K_2}{s} + \frac{K_3 s}{\tau s + 1} \tag{28}$$

where K_1, K_2, and K_3 are the positive proportional, integral, and derivative gains, respectively, and τ is the time constant of the derivative filter. PID controllers are widely employed in many industrial applications due to their simple structure and the physical interpretability of their parameters. As mentioned in the previous section, the LDW-PSO algorithm was adopted for optimization, with $c_1 = c_2 = 2$, $\omega_{\min} = 0.4$, and $\omega_{\max} = 0.9$ [35]. To validate the proposed control scheme, the PSO algorithm was run 40 times independently, and all resulting solutions were found to be satisfactory. The maximum number of iterations $iter_{\max}$ was set to 50 as the stopping criterion. The search space for the controller parameters is provided in Table 1. To handle the constraints, various techniques have been reported in the literature, such as incorporating penalty functions into the objective function. Employing this penalty-based approach, as adopted from [36], the stability condition in (7) is penalized with a sufficiently large positive constant set to 10^3.

Table 1. Range of the controller parameters

Controller parameters	K_1	K_2	K_3	τ
Search space	[0 50]	[0 10]	[0 100]	[0 2]

B Controller Design

The aim of the control algorithm is to regulate the vessel's heading angle ψ using the rudder angle δ. The relationship between these angles is given by [27]:

$$\dddot{\psi}(t)+(\frac{1}{T_1}+\frac{1}{T_2})\ddot{\psi}(t)+\frac{1}{T_1T_2}\dot{\psi}(t) = \frac{K}{T_1T_2}(\tau_3\dot{\delta}(t)+\delta(t)) \tag{29}$$

where T_1 and T_2 denote the time constants, and K is the gain constant. Owing to exogenous environmental disturbances and defining $a_1=\frac{KT_3}{T_1T_2}$, $a_2=\frac{K}{T_1T_2}$, $a_3=\frac{T_1+T_2}{T_1T_2}$, $a_4=\frac{1}{T_1T_2}$, the transfer function associated with (28) can be expressed as

$$G(s)=\frac{a_1s+a_2}{s\left(s^2+a_3s+a_4\right)} \tag{30}$$

with $a_i=\bar{a}_i\left(1+p_{a_i}\delta_{a_i}\right)$, where $\bar{a}_i$, $i=1,\dots,4$, denote the nominal values of a_i, and p_{a_i}, with $|\delta_{a_i}|\le 1$, represent the potential perturbations on these parameters. To represent the uncertain system within the framework of Fig. 2, the uncertain parameters are expressed in the form of an upper LFT as $a_i=F_U\left(M_{a_i},\delta_{a_i}\right)$, where $M_{a_i}=\begin{bmatrix}0 & \bar{a}_i\\ p_{a_i} & \bar{a}_i\end{bmatrix}$. The nominal parameters $\bar{a}_i$ are 0.0051, 0.00023, 0.192, and 0.0016, respectively, with $p_{a_i}=50\%$. Note that the container ships experience large variations in loading conditions, which can alter the mass and inertial properties by 30–50%. Accordingly, the uncertain system can be expressed as

$$y=F_u(G(s),\Delta) \tag{31}$$

where Δ is the perturbation matrix given by

$$\Delta=diag(\delta_{a_1},\delta_{a_2},\delta_{a_3},\delta_{a_4}) \tag{32}$$

in which $\|\Delta\|_\infty\le 1$.

In what follows, the weighting functions W_p and W_u are determined. Ideally, the sensitivity function S should be small to achieve good disturbance rejection—for instance, to satisfy a maximum tracking error requirement at specific frequencies. These specifications can be captured by imposing a magnitude bound on the sensitivity function, expressed as $\|W_pS\|_\infty\le 1$. Furthermore, to limit the control input magnitude u, an upper bound on the magnitude of KS must also be enforced, i.e., $\|W_uKS\|_\infty\le 1$. Consequently, to satisfy all of the aforementioned requirements, the following mixed-sensitivity condition must hold:

$$\begin{aligned}\|W_pS\|_\infty&\le 1\\ \|W_uKS\|_\infty&\le 1\end{aligned} \tag{33}$$

Assuming the nominal values are $\bar{a}=0.0051$, $\bar{b}=0.00023$, $\bar{c}=0.192$, and $\bar{d}=0.0016$, with $p_a=p_b=p_c=p_d=50\%$ and $-1\le\delta_a,\delta_b,\delta_c,\delta_d\le 1$, we have $G(s)=\dfrac{0.0051s+0.00023}{s^3+0.192s^2+0.0016s}$, and the associated weighting functions are chosen as follows:

$$\begin{aligned}W(s)&=2.4\frac{(s+0.03)}{(s+2)}\\ W_p(s)&=0.03\frac{s^3+10s^2+7s+1}{s^3+1.2s^2+0.09s+0.0001}\\ W_u(s)&=2\times10^{-3}\end{aligned} \tag{34}$$

C Results and Comparisons

Following the procedure described in Section B, the proposed fixed-structure controller is given by

$$K(s)=11.9741+\frac{0.0216}{s}+\frac{74.6978s}{0.1808s+1} \tag{35}$$

Fig. 7 shows the evolution profile of the objective function in (27). Figs. 8 and 9 illustrate the heading angle and waypoint tracking performance of the proposed controller, respectively. To further assess the performance of the proposed controller, it is compared with the H_∞ [3] and $D-K$ controllers. The numerator and denominator polynomials of the controllers are

$$\begin{aligned}N_{DK}(s)&=5.793s^6+19.74s^5+18.3s^4+4.122s^3\\&\quad+0.2721s^2+0.003031s+9.619\text{e}^{-6}\\ D_{DK}(s)&=s^7+4.95s^6+8.58s^5+6.122s^4+1.591s^3\\&\quad+0.1333s^2+0.003338s+3.598\text{e}^{-6}\\ N_{H_\infty}(s)&=112.372s^4+153.9816s^3+42.7038s^2\\&\quad+3.4954s+0.0274\\ D_{H_\infty}(s)&=s^5+2.9469s^4+3.6700s^3+2.0013s^2\\&\quad+0.2099s+0.0059\end{aligned}$$

The numerator and denominator of the reduced $D-K$ controller, obtained via Hankel norm approximation, are

$$\begin{aligned}N_{reduced}(s)&=5.793s^4+1.669s^3+0.1177s^2\\&\quad+0.001324s+4.217\text{e}^{-6}\\ D_{reduced}(s)&=s^5+1.831s^4+0.6286s^3\\&\quad+0.05672s^2+0.001462s+1.577\text{e}^{-6}\end{aligned}$$

It is apparent that the controllers are of relatively high order compared with the proposed fixed-structure controller in (35). The primary objective in designing these controllers is to guarantee the robustness of the overall closed-loop system. Figures 10 and 11 illustrate these aspects for the designed controllers. From Fig. 10, it can be observed that RS is achieved for all three design methods, and the controllers exhibit satisfactory stability margins. Among these, the best stability margin corresponds to the proposed controller. As shown in Fig. 10, the proposed and H_∞ controllers exhibit identical behavior over the entire frequency range. Similarly, as depicted in Fig. 11, the proposed and $D-K$ controllers yield comparable results, although the proposed controller performs slightly better. Nevertheless, the H_∞ controller fails to maintain RS.

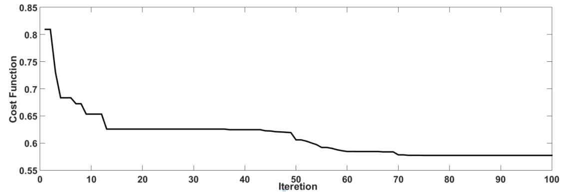


Figure 7. Profile of the objective function J.

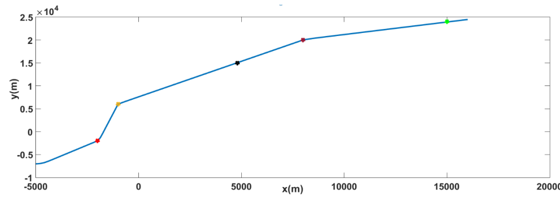


Figure 8. Heading angle response with the proposed controller.

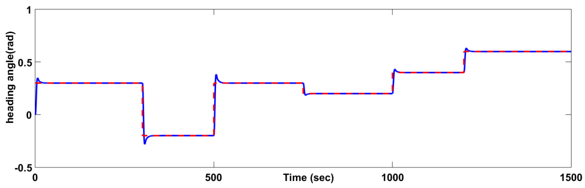


Figure 9. Way-point tracking performance of the proposed controller.

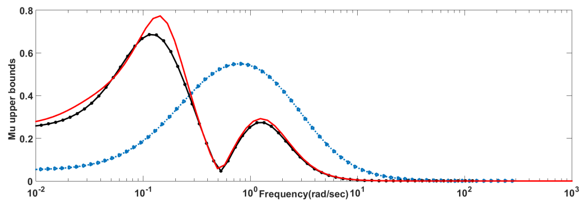


Figure 10. Robust stability comparison.

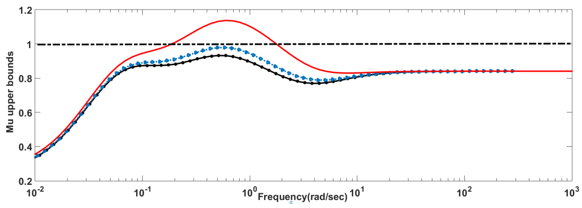


Figure 11. Robust performance comparison.

VI Conclusion

In this paper, a robust fixed-structure controller is designed for a container ship subject to model perturbations and external disturbances during waypoint tracking. The control problem is formulated as a constrained optimization problem, and the controller parameters are determined using EAs. Comparative simulation results demonstrate the satisfactory performance and effectiveness of the proposed controller. Future work will examine the application of a fractional-order fixed-structure controller to this problem.

References

[1] Fossen, T. *Guidance and control of ocean vehicles*, New York: Wiley, 1994.

[2] Pettersen, K.Y., Nijmeijer, H. Underactuated ship tracking control: theory and experiments, *International Journal of Control*, vol. 14, p: 200174:1435–46.

[3] Alfi, A., Shokrzadeh A, Asadi, M. Reliability analysis of H-infinity control for a container ship in way-point tracking control, *Applied Ocean Research*, vol. 52, pp. 309–316, 2015.

[4] Yurchenko, D., Alevras, P. Stability, control and reliability of a ship crane payload motion, *Probabilistic Engineering Mechanics*, vol. 38, pp. 173–179, 2014.

[5] Abdel-Rahman, E., Nayfeh, A., Masoud, Z. Dynamics and control of cranes: A review, *Journal of Vibration and Control*, vol. 9, pp. 863–908, 2003.

[6] Wu, J., et al. Ship's tracking control based on nonlinear time series model, *Applied Ocean Research*, vol. 36, pp.1–11, 2012.

[7] Do, K., Pan J. Global tracking control of underactuated ships with nonzero off-diagonal terms in their system matrices, *Automatica*, vol. 41, pp. 87–95, 2005.

[8] Moreira, L., Fossen, T.I., Soares, C.G. Path following control system for a tanker ship model, *Ocean Engineering*, vol. 34, pp. 2074–2085, 2007.

[9] L. Sheng, L., et al. Application of H_∞ control to ship steering system, *Journal of Marine Science and Application*, vol. 5, pp. 6–11, 2006.

[10] Velagic, J, Vukic, Z., Omerdic, E. Adaptive fuzzy ship autopilot for track-keeping, *Control Engineering Practice*, vol. 11, pp. 433–443, 2003.

[11] Faltinsen, O.M. *Sea loads on ships and offshore structures*, Cambridge: Cambridge University Press, 1990.

[12] Zhou, K.M., Doyle, J.C., Glover, K. *Robust and optimal control*, Beijing: National Defence Industry Press, 2002.

[13] Juang, J.Y., Chang, B.C. Robust control theory applied to ship maneuvering, in: *IEEE Conference on Decision and Control*, 1999.

[14] Packard, A., Doyle, J.C. The complex structured singular value, *Automatica*, vol.29, pp. 71–109, 1993.

[15] Packard, A., Doyle, J.C., Balas, G. Linear, multivariable robust control with a μ perspective, *Journal of Dynamic Systems, Measurement and Control*, vol. 115, pp. 426–438, 1993.

[16] Buso, S. Design of a robust voltage controller for a buck-boost converter using μ-synthesis, *IEEE Transactions on Control Systems Technology*, vol. 7, pp. 222–229, 1999.

[17] Stein, G., Doyle, J.C. Beyond singular values and loop-shapes, *AIAA J Guid Control*, vol. 14, pp. 5–16, 1991.

[18] Alfi, A., et al. Design and implementation of robust-fixed structure controller for telerobotic systems, *Journal of Intelligent & Robotic Systems*, vol. 83, no. 2, pp. 253-269, 2016.

[19] Arab, A., Alfi, A. An adaptive gradient descent-based local search in memetic algorithm for solving engineering optimization problems, *Information Sciences*, vol. 299, pp. 117–142, 2015.

[20] Shokri-Ghaleh, H., et al. Unequal limit cuckoo optimization algorithm applied for optimal design of nonlinear field calibration problem of a triaxial accelerometer, *Measurement*, vol. 164, p. 107963, 2020.

[21] Pahnehkolaei, S.M., Alfi, A., Machado, J.A. Particle swarm optimization algorithm using complex-order derivative concept: A comprehensive study, *Applied Soft Computing*, vol. 101, p. 107641, 2021.

[22] Shahri, E.S.A., Alfi, A., Machado, J.T. Fractional fixed-structure H_∞ controller design using augmented Lagrangian particle swarm optimization with fractional order velocity, Applied Soft Computing, vol. 77, pp.688-695, 2019.

[23] Zhou, K., Doyle, J. *Essentials of Robust Control*, Prentice-Hall, 1998.

[24] Doyle, J.C., Packard, A. Uncertain multivariable systems from a state space perspective, in: *American Control Conference*, Minneapolis, pp. 2147-2152, 1987.

[25] Fossen, T.I., Strand, J.P. Nonlinear ship control, in: *IFAC Conference on Control Application in Marine Systems*, pp. 1–75, 1998.

[26] Alfi, A. PSO with adaptive mutation and inertia weight and its application in parameter estimation of dynamic systems, *Acta Automatica Sinica*, vol. 37, pp. 541–549, 2011.

[27] Alfi, A., Fateh, M.M. Identification of nonlinear systems using modified particle swarm optimization: A hydraulic suspension system, *Vehicle Systems Dynamic*, vol. 46, pp. 871–887, 2011.

[28] Alfi, A., Modares, H. System identification and control using adaptive particle swarm optimization, *Applied Mathematical Modeling*, vol. 35, pp. 1210–1221, 2011.

[29] Alfi, A. Chaos suppression on a class of uncertain nonlinear chaotic systems using an optimal adaptive PID controller, *Chaos Solitons Fractals*, vol. 42, pp. 351–357, 2012.

[30] Khooban, M.H., Alfi, A., Nazari Maryam Abadi, D. Control of a class of non-linear uncertain chaotic systems via an optimal Type-2 fuzzy proportional integral derivative controller, *IET Science, Measurement & Technology*, vol. 7, pp. 50–58, 2013.

[31] Shokri-Ghaleh, H., Alfi, A. A comparison between optimization algorithms applied to synchronization of bilateral teleoperation systems against time delay and modeling uncertainties, *Appled Soft Computing*, vol. 24, pp. 447–456, 2014.

[32] Kennedy, J., Eberhart, R.C. Particle swarm optimization, in: *IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, 1995.

[33] Ratnaweera, A., Halgamuge, S.K., Watson, H.C. Self-organizing hierarchical particle swarm optimizer with time varying acceleration coefficients, *IEEE Transactions on Evolutionary Computation*, vol. 8, pp. 240–255, 2004.

[34] Kaitwanidvilai, S., Parnichkun, M. Design of structured controller satisfying H_∞ loop shaping using evolutionary optimization: Application to a pneumatic robot arm, *Engineering Letters*, vol. 16, pp. 193-201, 2008.

[35] Shi, Y., Eberhart, R. Empirical study of particle swarm optimization, in: *Congress on Evolutionary Computation*, pp. 1945–1950, 1999.

[36] Michalewicz, Z., et al. Evolutionary algorithms for constrained engineering problems, *Computers & Industrial Engineering*, vol. 30, pp. 851–870, 1996.