

Article

Real-Time Weight Optimization of a Nonlinear Model Predictive Controller Using a Genetic Algorithm for Ship Trajectory Tracking

Dunjing Yu , Fang Deng *, Hongyan Wang *, Xiuhui Hou , Hualin Yang  and Tikun Shan

College of Mechanical and Electrical Engineering, Qingdao University of Science and Technology, Qingdao 266061, China

* Correspondence: dengfhelen@163.com (F.D.); wanghy0546@163.com (H.W.)

Abstract: This paper presents a weight optimization method for a nonlinear model predictive controller (NMPC) based on the genetic algorithm (GA) for ship trajectory tracking. The weight coefficients Q and R of the objective function in NMPC are obtained via the real-time optimization of the genetic algorithm instead of the trial and error method, which improves the efficiency and accuracy of the controller. In addition, targeted improvements are made to the internal crossover operator, mutation operator, crossover rate, and mutation rate of the genetic algorithm. The simulation comparison of trajectory tracking between NMPC with real-time-optimized weight coefficients and the one with constant coefficients is performed. Finally, the simulation result shows that the controller with real-time-optimized weight coefficients has a better trajectory tracking effect than that with constant weight coefficients.

Keywords: nonlinear model predictive control; genetic algorithm; ship trajectory tracking; parameter optimization



Citation: Yu, D.; Deng, F.; Wang, H.; Hou, X.; Yang, H.; Shan, T. Real-Time Weight Optimization of a Nonlinear Model Predictive Controller Using a Genetic Algorithm for Ship Trajectory Tracking. *J. Mar. Sci. Eng.* **2022**, *10*, 1110. <https://doi.org/10.3390/jmse10081110>

Academic Editor: Vincenzo Crupi

Received: 6 July 2022

Accepted: 31 July 2022

Published: 12 August 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the rapid development of deep-sea applications, ship automatic control has received unprecedented attention. Trajectory tracking is one of the key technologies used in ship intelligent control. Therefore, the research and design of trajectory tracking control to ensure the accurate tracking of the desired trajectory plays an important role in the safe and effective implementation of marine engineering applications [1–3].

Relevant scholars have made many efforts and achievements in the field of ship trajectory tracking control. Katayama and Aoki [4] used the characteristics of the Euler approximation model of ship dynamics to design a reduced order observer and output feedback controller, which can effectively track a preset straight-line trajectory. Qu et al. [5] proposed an unmanned ship exponential tracking controller based on an observer to estimate external disturbances and system uncertainties, with the controller able to ensure exponential convergence and strong robustness. Liu et al. [6] improved the nonlinear robust control algorithm based on sliding mode control theory to simulate the maneuverability of the surface ship.

However, the above control methods cannot take the system input constraints directly into account due to the physical limits of the thrusters. Considering that the model predictive control (MPC) has a superior ability to establish constraints on optimization problems, it has been widely used in various fields, such as semiconductor, energy, and aerospace, to achieve advanced control [7–10]. In the field of marine vessel control, Oh and Sun [11] combined the LOS guidance algorithm with MPC to implement the tracking control in the presence of input constraints. Li and Sun [12] proposed a new disturbance-compensated MPC algorithm for ship heading control. Kapetanović et al. [13] proposed applied linear model predictive control (LMPC) for the trajectory tracking of underactuated ships

under disturbance and input constraint saturation. Furthermore, since ship dynamics is highly nonlinear, in order to improve the control performance, it stimulates the development of nonlinear model predictive control (NMPC) [14]. Abdelaal et al. [15] proposed a combined NMPC for the position and velocity tracking of underactuated surface vessels. Yang et al. [16] designed a robust NMPC scheme based on a nonlinear disturbance observer for the trajectory tracking control of a dynamic positioning ship.

It can be seen from the above research that NMPC can perform well for trajectory tracking. However, in traditional NMPC, the most important parameters of weight coefficients in objective function are generally obtained through trial or experience, which is time-consuming and easily falls into local optimization. Therefore, many studies have been conducted for the optimal weight tuning of model predictive control. Based on the real-time weight adjustment strategy, Zhao et al. [17] designed a spacing control law for the adaptive cruise control system, which can adjust the weight coefficient according to actual output demand. In addition, the development of intelligent algorithms also provides good ideas for this. For example, the genetic algorithm (GA) is combined with NMPC for parameter optimization because of its superior characteristics of global optimization [18]. In addition, Ramasamy et al. [19] used GA to determine the MPC weights to minimize overall energy utilization, reduce tracking error, and realize optimal control of the cement kiln. Essa et al. [20] adopted the combination of the Rhododendron search algorithm and GA to adjust the weight coefficients of MPC, and gained strong stability and robustness in the electro-hydraulic servo system. Zhang and Zhuan [21] proposed an improved genetic algorithm of the variable chromosome length coevolution method to optimize the weight matrices Q and R in the electromagnetic isolation system, and received better damping performance.

Aiming to improve the optimization performance of GA, the crossover operator and mutation operator have been improved upon in many studies. For example, Tang and Tseng [22] proposed an adaptive directed mutation operator, which can enhance the global search ability and accelerate the convergence of GA by integrating local directional and adaptive random search strategy. Chuang et al. [23] proposed the use of the direction-based crossover operator and the dynamic random mutation operator. The crossover operator uses the ranking information of chromosome fitness value in the population to generate offspring chromosomes. The mutation operator generates new chromosomes through an adaptively adjusted mutation step and random disturbance vector. However, the crossover operators cannot effectively produce offspring in some specific areas where the initial population is located, which reduces the performance of the algorithm. For the mutation operators, due to their random variability, on the one hand, the optimal solution of the current population may be mutated, thus destroying the optimal solution. On the other hand, it may not make any mutation to the worst solution, which cannot improve the population diversity. In addition, the crossover rate and the mutation rate are two important control parameters in GA, and their values affect the convergence of the algorithm. The traditional determination method is to take a constant value according to the results of many experiments, which cannot ensure the algorithm to finally converge to the global optimal solution.

On the basis of the above analysis, this paper proposes a novel real-time weight optimization method of a nonlinear model predictive controller using a genetic algorithm for ship trajectory tracking, and the main contributions are as follows:

- (1) NMPC is applied to the ship trajectory tracking control, and the genetic algorithm is used to optimize the weight coefficients of the objective function in NMPC in real time;
- (2) The crossover operator, mutation operator, crossover rate, and mutation rate in the genetic algorithm are improved to enhance the performance of the genetic algorithm.

The rest of the paper is organized as follows: Section 2 introduces the structure of NMPC for ship control. Section 3 puts forward the GA-based real-time weight optimization

algorithm of NMPC. In Section 4, simulations are conducted and results are discussed. The last section gives the conclusions.

2. Nonlinear Model Predictive Control for Trajectory Tracking

2.1. Ship Mathematical Model

For surface ships, the heave, pitch, and roll motions are neglected, and only the horizontal motions of surge, sway, and yaw need to be considered. According to Fossen [24,25], the inertial earth-fixed frame and the body-fixed frame attached to the moving vessel are constructed to build the nonlinear mathematical model of a surface ship as:

$$\dot{\eta} = J(\psi)v, \quad (1)$$

$$M\dot{v} = -Dv + \tau + J^T(\psi)d, \quad (2)$$

where $\eta = [x, y, \psi]^T$ represents the ship's position and heading vector in the inertial earth-fixed frame, $v = [u, v, r]^T$ is the velocity vector in the body-fixed frame, $J(\psi)$ is the transfer matrix, M is the mass matrix, including the added mass, and D is the damping matrix. These matrices are given as:

$$\begin{aligned} J(\psi) &= \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}, M = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 \\ 0 & m - Y_{\dot{v}} & mX_g - Y_{\dot{r}} \\ 0 & mX_g - Y_{\dot{r}} & I_z - N_{\dot{r}} \end{bmatrix}, \\ D &= \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & -Y_r \\ 0 & -N_v & -N_r \end{bmatrix}, \end{aligned} \quad (3)$$

In addition, $\tau = [\tau_u, \tau_v, \tau_r]^T$ is the control force vector, composed of surge force τ_u , sway force τ_v , and yaw moment τ_r , which are limited by the nonlinear saturation characteristic of the propeller:

$$\tau_{min} \leq \tau \leq \tau_{max}, \quad (4)$$

where $\tau_{min} = [\tau_{umin}, \tau_{vmin}, \tau_{rmin}]^T$, $\tau_{max} = [\tau_{umax}, \tau_{vmax}, \tau_{rmax}]^T$ are the minimal and maximal force vectors, respectively, that can be provided by the thrusters. Furthermore, $d = [d_1, d_2, d_3]^T$ refers to the unknown time-varying disturbance vector composed of wind, wave, and current environmental interference and model uncertainties.

Generally, the surface ship model can be expressed by the state-space equation as:

$$\begin{aligned} \dot{x}(t) &= A(t)x + B(t)u + \Gamma(t)d(t), \\ z(t) &= Cx(t), t \geq 0, \end{aligned} \quad (5)$$

where $x = [x, y, \psi, u, v, r]^T$ is the state vector, $u = \tau$ is the control input vector, and z is the output vector. In combination with Equation (1), $A(t)$, $B(t)$, $C(t)$, $\Gamma(t)$ can be expressed as:

$$\begin{aligned} A(t) &= \begin{bmatrix} 0_{3 \times 3} & J(\psi(t)) \\ 0_{3 \times 3} & -M^{-1}D \end{bmatrix}, B(t) = \begin{bmatrix} 0_{3 \times 3} \\ M^{-1} \end{bmatrix}, \\ \Gamma(t) &= \begin{bmatrix} 0_{3 \times 3} \\ M^{-1}J^T(\psi(t)) \end{bmatrix}, C(t) = [I_{3 \times 3}, 0_{3 \times 3}], \end{aligned} \quad (6)$$

2.2. Discrete-Time NMPC Scheme

In general, the NMPC scheme is designed based on the discrete-time model of the system. Therefore, referring to Zhu et al. [26] and Veerasamy et al. [27], the ship model (5) can be discretized utilizing a sampling time of T_s as:

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}(k)\mathbf{x}(k) + \mathbf{B}(k)\mathbf{u}(k) + \mathbf{\Gamma}(k)\mathbf{d}(k), \\ \mathbf{z}(k) &= \mathbf{C}\mathbf{x}(k), \end{aligned} \quad (7)$$

where k is the current time step with regard to the current time t , $\mathbf{x}(k) = [x(k), y(k), \psi(k), u(k), v(k), r(k)]^T$, $\mathbf{u}(k) = [\tau_u(k), \tau_v(k), \tau_r(k)]^T$, $\mathbf{d}(k) = [d_1(k), d_2(k), d_3(k)]^T$, $\mathbf{z}(k) = [x(k), y(k), \psi(k)]^T$ are corresponding vectors of the continuous-time model (5), and the discrete-time matrices $\mathbf{A}(k)$, $\mathbf{B}(k)$, as well as $\mathbf{\Gamma}(k)$, can be determined at each time step as:

$$\begin{aligned} \mathbf{A}(k) &= e^{\mathbf{A}(t)T_s}, \\ \mathbf{B}(k) &= \mathbf{A}^{-1}(t)(\mathbf{A}(k) - \mathbf{I})\mathbf{B}(t), \\ \mathbf{\Gamma}(k) &= \mathbf{A}^{-1}(t)(\mathbf{A}(k) - \mathbf{I})\mathbf{\Gamma}(t), \end{aligned} \quad (8)$$

For trajectory tracking, the control objective is to operate the ship by thrusters to track the predefined reference trajectory under external disturbances and input saturation constraints. That is, for a given reference trajectory $\mathbf{z}_r = [x_r, y_r, \psi_r]^T$, we need to determine the control input to make $\lim_{x \rightarrow \infty} \|\mathbf{z}(t) - \mathbf{z}_r(t)\| = 0$. Thus, the objective function of NMPC can be constructed as:

$$\begin{aligned} \min_{\mathbf{U}(k)} V(k) &= \sum_{n=1}^{N_p} (\|\mathbf{z}_n(k) - \mathbf{z}_r(k)\|_Q^2 + \|\mathbf{u}_n(k)\|_R^2), \\ \text{s.t. } \mathbf{x}_{n+1}(k) &= \mathbf{A}(k)\mathbf{x}_n(k) + \mathbf{B}(k)\mathbf{u}_n(k) + \mathbf{\Gamma}(k)\mathbf{d}_n(k), n = 0, \dots, N_p - 1, \\ \mathbf{z}_n(k) &= \mathbf{C}\mathbf{x}_n(k), n = 1, \dots, N_p, \\ \tau_{min} &\leq \mathbf{u}_n(k) \leq \tau_{max}, \end{aligned} \quad (9)$$

where $\mathbf{U}(k) = [u_0(k)^T, \dots, u_{N_c-1}(k)^T]^T$ is the control input sequence in the control horizon N_c , and $\mathbf{Z}(k) = [z_0(k)^T, \dots, z_{N_p}(k)^T]^T$ is the predictive output sequence in the predictive horizon N_p . Since N_c is less than N_p , it is assumed that, when $N_c \geq n \geq N_p$, $u_n(k) = u_{n-1}(k)$. In addition, $Q = \text{diag}(q_1, q_2, q_3)$ and $R = \text{diag}(r_1, r_2, r_3)$ are weight matrices that are positive and definite matrices, which proportionally allocate the state vector and the input vector in the objective function (9).

Note that the system is nonlinear; thus, the system matrices are time-varying during the predictive horizon. However, since the navigation state of the ship changes slowly, it can be assumed that $\mathbf{A}(k)$, $\mathbf{B}(k)$, and $\mathbf{\Gamma}(k)$ remain unchanged during the prediction horizon. Therefore, the predictive outputs can be calculated like a linear model, and the nonlinear optimization problem will be transformed into a relatively simple quadratic optimization problem. Thus, the optimal control sequence $\mathbf{U}^*(k) = [u_0^*(k)^T, \dots, u_{N_c-1}^*(k)^T]^T$ is obtained by optimizing the objective function (9), and the first value $u_0^*(k)$ is substituted into the process.

The estimated value of disturbance is introduced into the online optimization calculation of NMPC, so as to eliminate the influence of disturbance and realize robust control. Based on Yang et al. [16], the calculation method of interference estimation value is as follows:

$$\begin{aligned} \hat{\mathbf{d}}(t) &= \mathbf{p}(t) + \mathbf{L}_0\mathbf{J}(\psi)\mathbf{M}\mathbf{v}, \\ \dot{\mathbf{p}}(t) &= -\mathbf{L}_0\mathbf{p}(t) - \mathbf{L}_0[\dot{\mathbf{J}}(\psi)\mathbf{M}\mathbf{v} + \mathbf{J}(\psi)(-\mathbf{D}\mathbf{v} + \boldsymbol{\tau}) + \mathbf{L}_0\mathbf{J}(\psi)\mathbf{M}\mathbf{v}], \end{aligned} \quad (10)$$

where $\hat{d}(t)$ is the estimate of $d(t)$, $p(t)$ is an auxiliary state vector of the disturbance observer, and L_0 is a positive definite symmetric matrix. In addition, the predictive output sequence $Z(k)$ can be obtained from Equations (7) and (8) as:

$$Z(k) = \Omega x(k) + \Phi U(k) + \Lambda O_d(k), \quad (11)$$

where $O_d(k) = \Gamma(k)\hat{d}(k)$, and:

$$\Gamma = \begin{bmatrix} CA(k) \\ CA^2(k) \\ \vdots \\ CA^{N_p}(k) \end{bmatrix}, \Omega = \begin{bmatrix} C \\ C(A(k) + I_{6 \times 6}) \\ \vdots \\ C(\sum_{i=1}^{N_p-1} A^i k + I_{6 \times 6}) \end{bmatrix},$$

$$\Phi = \begin{bmatrix} CA(k) & 0 & \cdots & 0 \\ CA(k)B(k) & CB(k) & \cdots & 0 \\ \vdots & \ddots & \cdots & \vdots \\ CA^{N_c-1}(k)B(k) & CA^{N_c-2}(k)B(k) & \cdots & CB(k) \\ CA^{N_c}(k)B(k) & CA^{N_c-1}(k)B(k) & \cdots & C(A(k) + I_{6 \times 6})B(k) \\ \vdots & \ddots & \cdots & \vdots \\ CA^{N_p-1}(k)B(k) & CA^{N_p-2}(k)B(k) & \cdots & C(\sum_{i=1}^{N_p-N_c} A^i k + I_{6 \times 6})B(k) \end{bmatrix}, \quad (12)$$

3. Genetic Algorithm-Based Weight Tuning of NMPC

The weight coefficients in objective function (9) are very important for the NMPC online optimization calculation. If not selected properly, it will lead to a non-optimal solution, or even non-convergence. However, the weight matrices Q and R are usually determined by trial or experience, which is usually time-consuming and cannot be adjusted in real time with the change of control state to maintain the optimal solution [28]. Therefore, it is necessary to adjust the weight matrices Q and R online in real time to improve the control accuracy of NMPC. In this way, the adjusted parameters will have the direct physical significance of making the optimization process faster and the results more accurate [29]. Considering that the genetic algorithm (GA) has the characteristics of strong adaptability and automatic search [30], it is selected to optimize the weight matrices Q and R .

3.1. General Principle of GA

GA is a method of searching an optimal solution by simulating the process of natural evolution. It transforms the problem-solving process into a process that is similar to the crossover and mutation of chromosome genes in biological evolution. The main components of GA are fitness function, selection operator, crossover operator, and mutation operator, etc. Fitness function is an index that is used to judge the quality of individuals in the group. It is evaluated according to the objective function of the problem. The function of the selection operator is to screen out individuals with excellent genes for follow-up. The crossover operator refers to the operation of replacing and reorganizing some structures of two parent individuals to generate new individuals. The content of the mutation operator is to change the gene value on some loci of the individual string in the population.

Generally, when using GA to solve an optimization problem, a population composed of multiple individuals should be constructed first. Each chromosome of individuals in the population represents a feasible solution to the problem. Additionally, each solution is substituted into the fitness function to evaluate the fitness of the individual. The solutions with low fitness are gradually eliminated, and those with high fitness are added. Then, the next generation is generated through replication, crossover, mutation, and other operations. In this way, individuals with high fitness will be obtained after n generations of evolution; thus, a relatively accurate solution to the problem can be obtained.

Herein, to obtain online the optimal weight factors of the weight matrices Q and R in the NMPC objective function, the weight factors $q_1, q_2, q_3, r_1, r_2, r_3$ can be continuously optimized in GA. To improve the solution efficiency, a novel GA algorithm is proposed, considering the actual needs of ship track control, and utilizing the improved crossover operator, mutation operator, crossover rate, and mutation rate.

3.2. Application and Improvement of GA for Real-Time Weight Tuning

3.2.1. Fitness Function

Fitness function determines the direction of GA evolution, and is set according to practical problems. In NMPC, the best choice of Q and R is required to minimize the objective function (9) for the best trajectory tracking effect. Therefore, the Q and R optimized by the fitness function in this novel GA needs to meet the requirements of minimizing the objective function, and the fitness function is defined as follows:

$$Fitness = \frac{1}{v(k)} = \frac{1}{\sum_{n=1}^{N_p} (\|z_n(k) - z_r(k)\|_Q^2 + \|u_n(k)\|_R^2)}, \quad (13)$$

3.2.2. Improvement of Crossover Operator

In order to ensure that the genes of better individuals can be inherited to the offspring, the tournament selection strategy is adopted for the genetic operator. The selected individuals will generate a new individual through the crossover operator, and the standard crossover algorithm can be expressed as follows:

$$Q_1 = (T_1 + T_2)/2 + \beta \times (T_1 - T_2)/2, \quad (14)$$

where Q_1 is the offspring chromosome; T_1 and T_2 are parent chromosomes. β is a random number between $[0,1]$. However, there is a certain chance that the tournament strategy will repeatedly choose parent chromosomes, which indirectly leads to insufficient diversity. In order to solve this problem, by introducing T_3 , the improved crossover strategy is as follows:

$$Q_2 = (T_1 + T_2 + T_3)/3 + \beta \times (2T_1 - T_2 - T_3)/3, \quad (15)$$

Comparing Equations (14) and (15), the former is conducive to individual stability and less individual change range, due to less utilization of parent information. In Equation (15), each individual uses more information from the surrounding individuals, so that it is easy to quickly search a certain area in the early stage of the algorithm. Therefore, if the current number of iterations is less than $T \times Gen_{max}$, the crossover algorithm of Equation (15) is adopted. Otherwise, Formula (14) is adopted. The specific T and Gen_{max} will be determined in subsequent experiments.

3.2.3. Adoption of Mutation Operator

Aiming to solve the problems of mutation operator, an improvement is made on the basis of the method proposed by Tsoulos et al. [31]. Firstly, chromosomes in the population are sorted according to their fitness values from high to low, and then the mutation scale is determined according to the mutation rate P_m . In order to purposefully mutate the chromosome with low fitness rather than the optimal one, the mutation operator equation calculated using the PSO optimization method is as follows:

$$c'_i = l_1 m_1 (c_i^b - c_i) + l_2 m_2 (c_i^b - c_i), \quad (16)$$

where l_1 and l_2 are two positive constants, m_1 and m_2 are uniform random numbers between $[0,1]$, and c^b represents the chromosome with largest fitness currently found. The improvement can ensure that only the worst solutions will be mutated. This will improve

the gene, increase the diversity of the population, and prevent the algorithm from falling into a local optimal solution.

3.2.4. Improvement of Crossover and Mutation Rate

According to the Darwinian evolution principle, in the early stage of genetic algorithm evolution, the crossover operator can enhance the global search ability of the algorithm, which can avoid falling into the local optimal solution. In the later stage of evolution, when getting closer to optimal solution, the crossover operator will affect the convergence speed. At this time, the mutation operator plays a more important role. Herein, the crossover rate P_c and the mutation rate P_m can be defined as follows:

$$\begin{aligned} P_c &= P_1 - 0.35 \times \left(\frac{gen}{Gen_{max}} \right), \\ P_m &= P_2 + 0.05 \times \left(\frac{gen}{Gen_{max}} \right), \end{aligned} \quad (17)$$

where P_1 is the initial crossover rate and P_2 is the initial mutation rate. Gen_{max} is the maximum number of iterations and gen is the current number of iterations. With the progress of the search process, the crossover rate is getting smaller and smaller, while the mutation rate is getting larger and larger.

3.3. Optimization Procedure of GA-NMPC

The genetic algorithm-based nonlinear model predictive control (GA-NMPC) for vessel trajectory tracking is shown in Figure 1. The weight matrices Q and R of NMPC are optimized in real time using the genetic algorithm, and the implementation steps are concluded as follows.

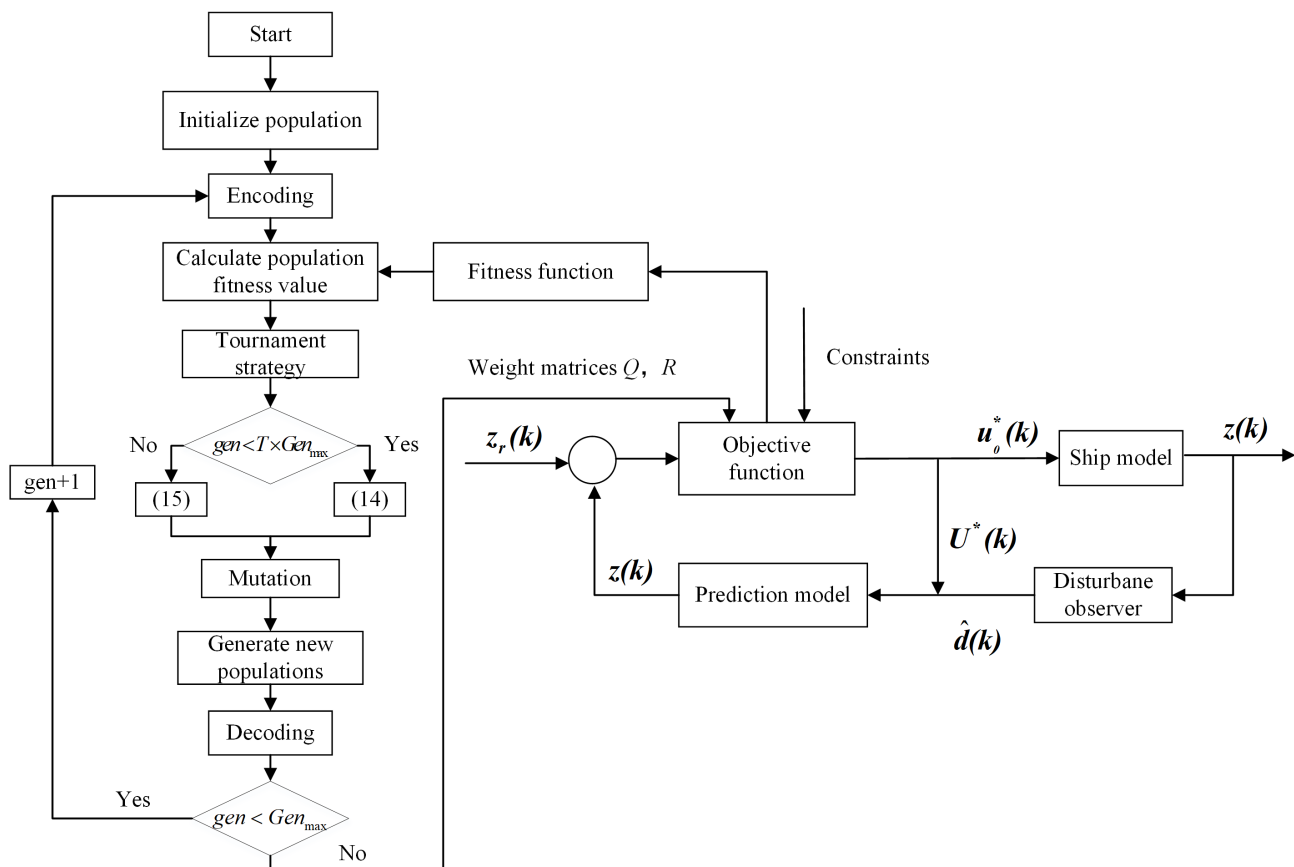


Figure 1. GA-NMPC schematic diagram.

- (1) Initialize the weight matrices Q and R , and obtain $\mathbf{U}(k)$ and $\mathbf{Z}(k)$ through Equations (9) and (10), respectively.
- (2) Define parent population size as N_{pop} , and $q_1, q_2, q_3, r_1, r_2, r_3$ are the genes carried by each individual's chromosome.
- (3) Encode chromosomes and calculate the fitness values through Equation (13), and sort the chromosomes from high to low according to fitness values.
- (4) Determine the number of chromosomes that need to be crossed according to crossover rate P_c . Select a corresponding number of chromosomes with high fitness values through the tournament strategy, and then cross these chromosomes according to Equations (14) and (15) to obtain new chromosomes. Finally, the newly generated chromosomes are used to replace the chromosomes with the lowest fitness values in the parent population.
- (5) After crossover, the chromosome will be mutated according to the mutation rate P_m . The individuals with low fitness values in the parent population are mutated based on Equation (16). Thus far, a new parent population has emerged.
- (6) If the number of iterations does not reach the maximum number of iterations Gen_{max} , continue the cycle. Otherwise, the individual with largest fitness value is introduced into objective function (9) as weight matrices Q and R . The optimal solution sequence $\mathbf{U}^*(k) = [u_0^*(k)^T, \dots, u_{N_c-1}^*(k)^T]^T$ is obtained by solving the objective function (9), and then the first element $u_0^*(k)$ is input into the closed-loop system to complete the ship trajectory tracking.

4. Simulation Evaluation and Discussion

In this section, the simulation evaluation of the trajectory tracking performance using the proposed GA-NMPC is implemented. The CSII ship, which is a replica of the supply ship with a ratio of 1:70, is chosen as the simulation object. Its a priori identified dynamic parameters are shown in Table 1.

Table 1. CSII ship parameters [32].

Parameter	Value	Unit	Parameter	Value	Unit
m	23.8	kg	Y_r	−0.1	kg·m/s
I_z	1.76	kg·m ²	$Y_{\dot{\theta}}$	−10.0	kg
X_g	0.046	m	$Y_{\ddot{\theta}}$	−0.0	kg·m
X_u	−2	kg/s	N_v	−1.0	kg·m/s
$X_{\dot{u}}$	−2.0	kg	N_r	−0.5	kg·m ² /s
Y_v	−0.7	kg/s	$N_{\ddot{r}}$	−1.0	kg·m ²

4.1. Main Parameters Setting

The maximal and minimal thrust limit in (4) is set as $\tau_{max} = [2.0\text{N}, 2.0\text{N}, 1.5\text{N}\cdot\text{m}]^T$, $\tau_{min} = [-2.0\text{N}, -2.0\text{N}, -1.5\text{N}\cdot\text{m}]^T$, respectively. The sample time step is set as $T_s = 0.1$. In GA-NMPC, the population size $N_{pop} = 30$, and the initial crossover rate and mutation rate are $P_c = 0.9$, $P_m = 0.2$, respectively. The parameters defined in Section 3.2.2 are set as $T = 0.6$, $Gen_{max} = 100$.

To demonstrate the effectiveness of the proposed real-time weight optimization algorithm for NMPC, the control effects of NMPC with different Q and R will be compared. Herein, three cases are s

- (1) The proposed GA-NMPC algorithm with real-time online weight optimization with initial $Q = \text{diag}(50, 50, 50)$, $R = \text{diag}(1, 1, 1)$;
- (2) NMPC with constant weight matrices $Q = \text{diag}(100, 100, 100)$, $R = \text{diag}(1, 1, 1)$;
- (3) NMPC with constant weight matrices $Q = \text{diag}(5, 5, 5)$, $R = \text{diag}(1, 1, 1)$.

In addition, two different reference trajectories are used to verify the performance, and they are set as follows:

- (1) Circle trajectory, with the expression described as:

$$\begin{aligned}x_r &= 3 - 3 \cos(0.025t), \\y_r &= 4 \sin(0.025t), t \geq 0,\end{aligned}\quad (18)$$

- (2) Sine curve trajectory, with the expression described as:

$$\begin{aligned}x_r &= 0.05t, \\y_r &= 3 \sin(0.025t), t \geq 0.\end{aligned}\quad (19)$$

The desired heading ψ_r of the two trajectories are both taken as the direction of the tangent vector along the path.

4.2. Results Analysis

The simulation results of the three cases based on the two trajectories are shown in Figures 2–5, where the results of cases (1)–(3) are represented by red, blue, and pink solid lines, respectively, while the reference trajectory is represented by green dotted lines. Figures 2a and 4a are ship position and heading. Figures 2b and 4b represent ship velocities in surge, sway, and yaw. Figures 3 and 5 show the trajectory tracking results under different cases. For the circle trajectory, it can be seen from Figures 2a and 3 that the trajectory tracking results of the three cases are ideal. However, by observing the local enlarged picture, we can see that the error of the result obtained by the controller using constant weight matrices is slightly larger than that of GA-NMPC. In addition, the heading angle of NMPC with constant matrices Q and R cannot return to zero in time after completing a circle, so there will be obvious fluctuations when starting a new voyage. This problem can also be reflected in Figure 2b that, at the end of a tracking cycle, the ship velocities will fluctuate significantly. However, the adoption of real-time-optimized Q and R solves this problem well.

Figures 4 and 5 show the tracking results of the sine curve trajectory. In Figures 4a and 5, the three cases have a good tracking effect on the trajectory. However, in contrast, the control result of NMPC with constant Q and R is slightly unstable compared with that of real-time-optimized Q and R . Additionally, when the element values in matrix Q are small, as can be seen from Figure 4b, the velocity v fluctuates greatly. Therefore, by receiving ship feedback to optimize the weight matrices in real time, the tracking error can be reduced and obtain a better control effect.

Figures 2c,d and 4c,d respectively show the matrices Q and R optimized in real time by GA-NMPC during ship tracking of the two trajectories. It can be seen that, at every step, the genetic algorithm adjusts elements in the Q and R according to the tracking error to improve population fitness, which can achieve the best control effect. Although using constant weight matrices can also obtain a relatively ideal result, it needs many attempts and adjustments, which is time-consuming and laborious. In addition, the final control effect may not be the best, due to the invariance of weights.

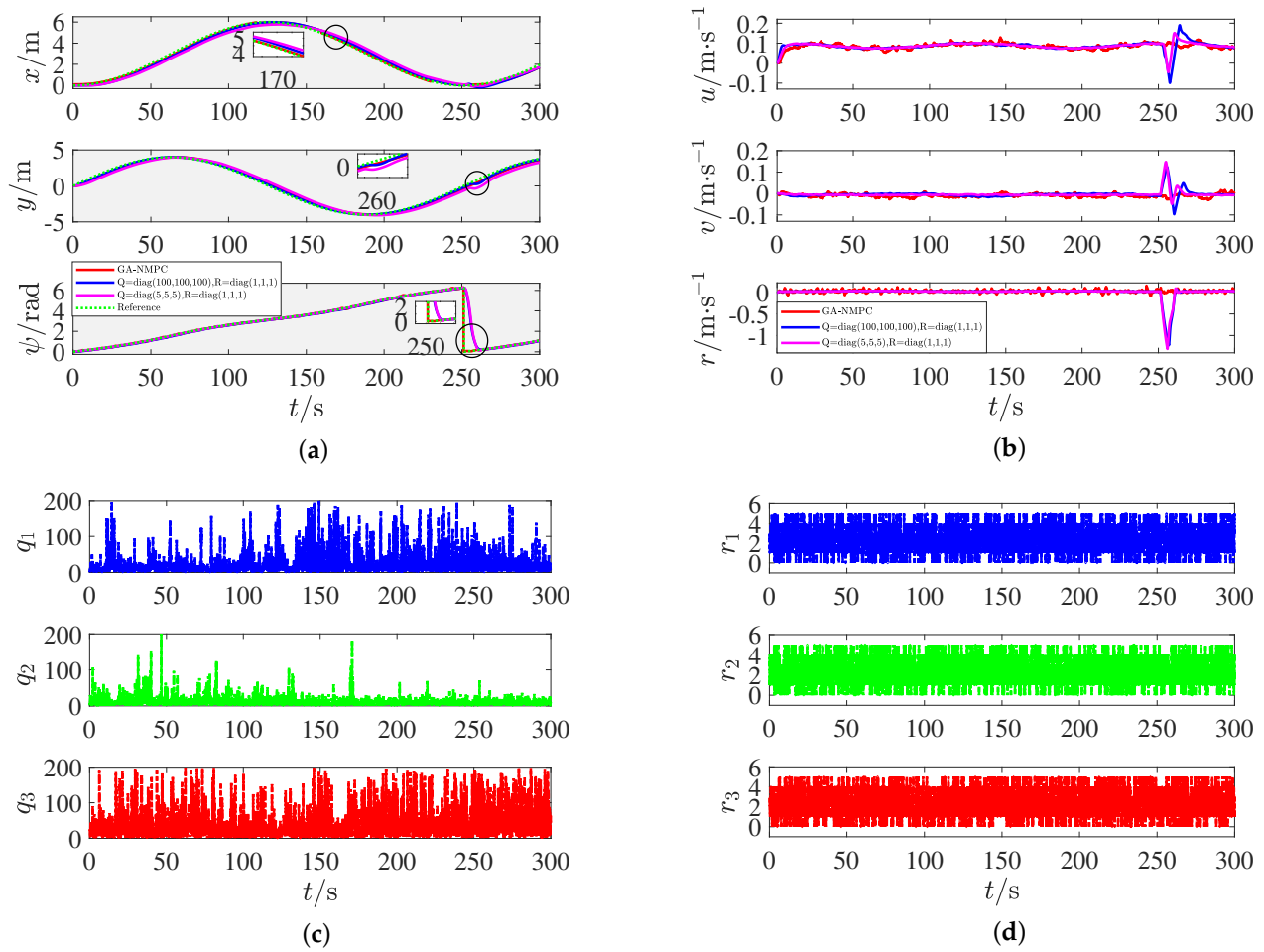


Figure 2. Simulation results for circle trajectory. (a) ship position and heading; (b) ship velocities in surge, sway, and yaw; (c) real-time optimized matrix Q ; (d) real-time optimized matrix R .

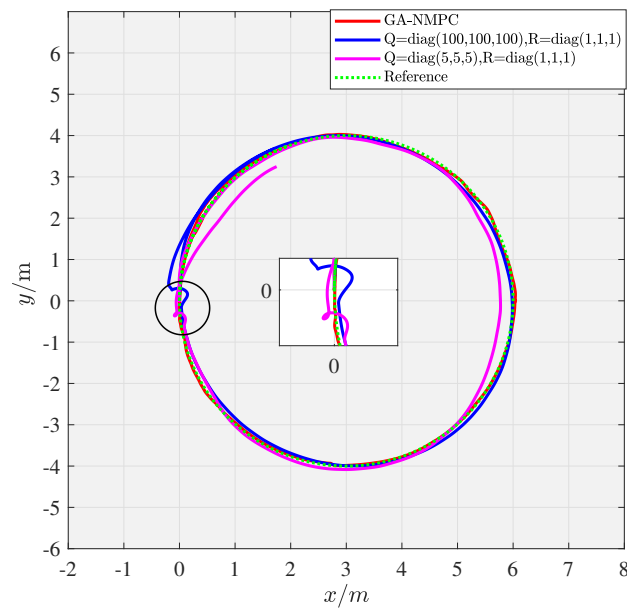


Figure 3. Circle trajectory tracking results under different cases.

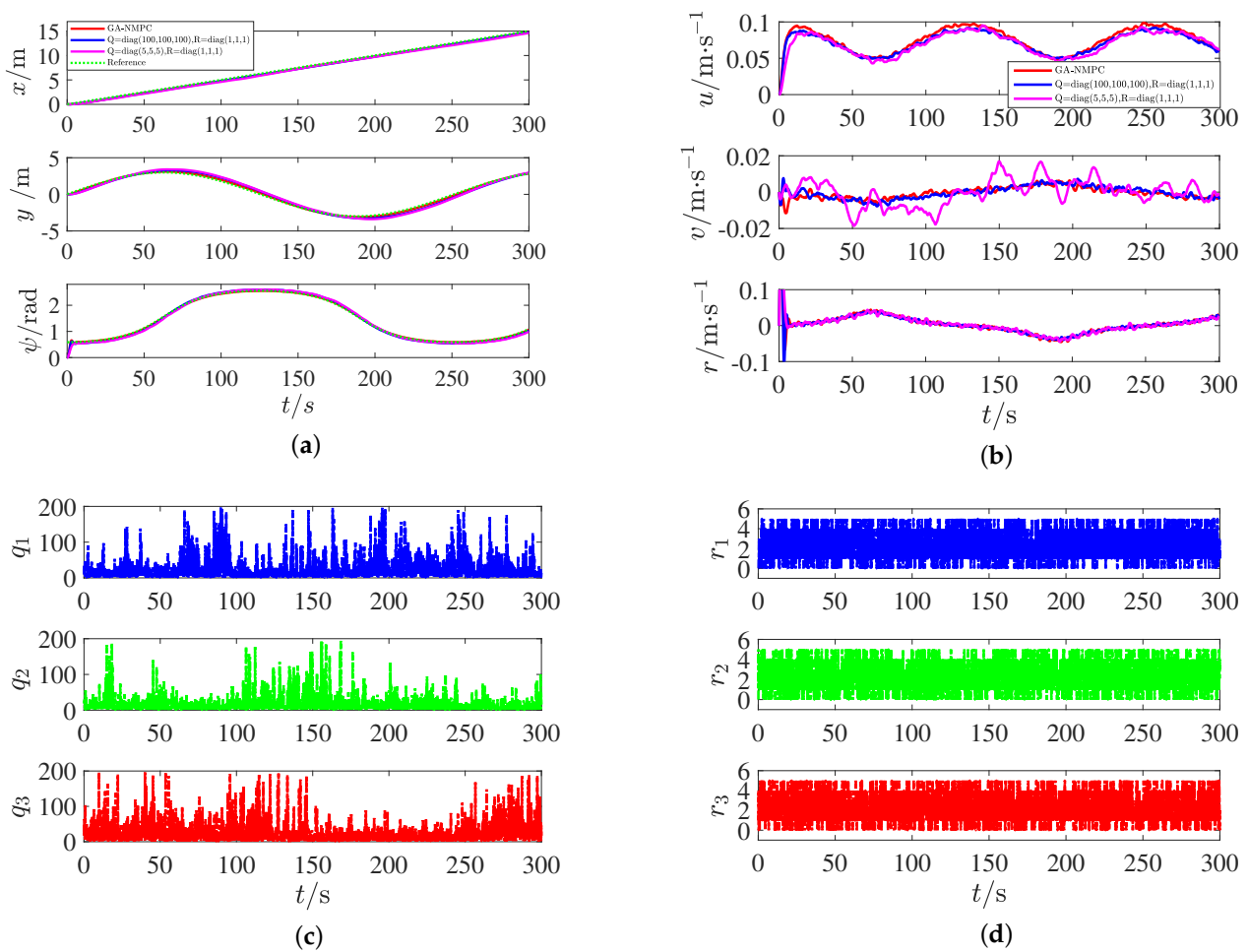


Figure 4. Simulation results for the sine curve trajectory. (a) ship position and heading; (b) ship velocities in surge, sway and yaw; (c) real-time optimized matrix Q ; (d) real-time optimized matrix R .

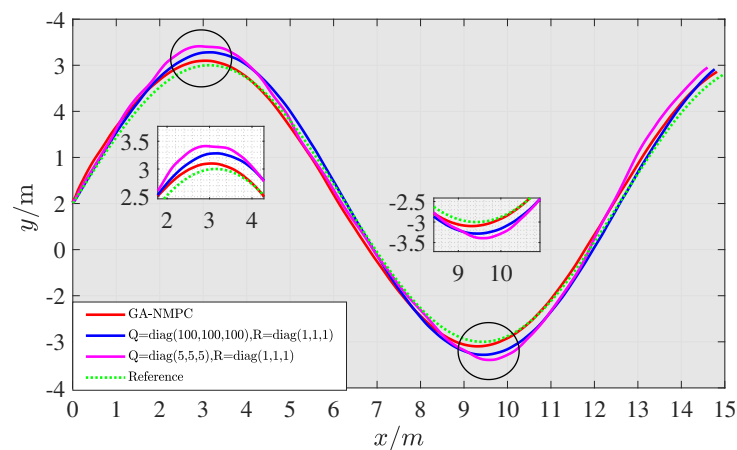


Figure 5. Sine trajectory tracking results under different cases.

5. Conclusions

In this paper, a real-time weight tuning method based on the genetic algorithm (GA) is proposed and applied in a ship nonlinear model predictive controller for ship trajectory tracking. The weight matrices Q and R to be optimized in the NMPC objective function are taken as individuals in GA. The individuals with the highest fitness are obtained as weight factors in the matrices of Q and R ; thus, the NMPC will solve the objective

function online and obtain the optimal control action. In addition, the crossover operator, mutation operator, crossover rate, and mutation rate of the genetic algorithm are improved to enhance the diversity of the population and the efficiency of searching for the optimal solution. This optimization method adjusts the weight proportion of the input value in the objective function in real time according to the actual navigation and makes up for the defects of the traditional trial and error method, as well as the empirical method. The controller optimized by this method is simulated and compared with the traditional NMPC. The results show that the proposed GA-NMPC method makes it easy to implement NMPC and improves the control efficiency of ship trajectory control.

Author Contributions: Conceptualization, F.D.; Methodology, D.Y.; Formal analysis, D.Y.; Writing—Original Draft Preparation, D.Y.; Writing—Review and Editing, H.Y. and D.Y.; Software, D.Y. and X.H.; Investigation, T.S.; Funding Acquisition: F.D. and H.W.; Resources: F.D. and H.W.; Validation, X.H.; Supervision, F.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (No. 52101401), the National Key R&D Program of China (No. 2021YFB3401400), and the Collaborative Innovation Center of Intelligent Green Manufacturing Technology and Equipment, Shandong (No. IGSD-2020-011).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- Moreira, L.; Fossen, T.I.; Soares, C.G. Path following control system for a tanker ship model. *Ocean Eng.* **2007**, *34*, 2074–2085. [\[CrossRef\]](#)
- Consolini, L.; Tosques, M. A minimum phase output in the exact tracking problem for the non-minimum phase underactuated surface ship. *IEEE Trans. Autom. Control* **2012**, *57*, 3174–3180. [\[CrossRef\]](#)
- Zhu, G.; Du, J. Global robust adaptive trajectory tracking control for surface ships under input saturation. *IEEE J. Ocean. Eng.* **2018**, *45*, 442–450. [\[CrossRef\]](#)
- Katayama, H.; Aoki, H. Straight-line trajectory tracking control for sampled-data underactuated ships. *IEEE Trans. Control Syst. Technol.* **2013**, *22*, 1638–1645.
- Qu, Y.; Xiao, B.; Fu, Z.; Yuan, D. Trajectory exponential tracking control of unmanned surface ships with external disturbance and system uncertainties. *ISA Trans.* **2018**, *78*, 47–55. [\[CrossRef\]](#) [\[PubMed\]](#)
- Liu, Y.; Bu, R.; Gao, X. Ship trajectory tracking control system design based on sliding mode control algorithm. *Pol. Marit. Res.* **2018**, *25*, 26–34. [\[CrossRef\]](#)
- Wang, W.; Rivera, D.E.; Kempf, K.G. Model predictive control strategies for supply chain management in semiconductor manufacturing. *Int. J. Prod. Econ.* **2007**, *107*, 56–77. [\[CrossRef\]](#)
- Brdys, M.A.; Grochowski, M.; Gminski, T.; Konarczak, K.; Drewa, M. Hierarchical predictive control of integrated wastewater treatment systems. *Control Eng. Pract.* **2008**, *16*, 751–767. [\[CrossRef\]](#)
- Salsbury, T.; Mhaskar, P.; Qin, S.J. Predictive control methods to improve energy efficiency and reduce demand in buildings. *Comput. Chem. Eng.* **2013**, *51*, 77–85. [\[CrossRef\]](#)
- Keviczky, T.; Balas, G.J. Receding horizon control of an F-16 aircraft: A comparative study. *Control Eng. Pract.* **2006**, *14*, 1023–1033. [\[CrossRef\]](#)
- Oh, S.R.; Sun, J. Path following of underactuated marine surface vessels using line-of-sight based model predictive control. *Ocean Eng.* **2010**, *37*, 289–295. [\[CrossRef\]](#)
- Li, Z.; Sun, J. Disturbance compensating model predictive control with application to ship heading control. *IEEE Trans. Control Syst. Technol.* **2011**, *20*, 257–265. [\[CrossRef\]](#)
- Kapetanović, N.; Bibuli, M.; Mišković, N.; Caccia, M. Real-time model predictive line following control for underactuated marine vehicles. *IFAC-PapersOnLine* **2017**, *50*, 12374–12379. [\[CrossRef\]](#)
- Henson, M.A. Nonlinear model predictive control: Current status and future directions. *Comput. Chem. Eng.* **1998**, *23*, 187–202. [\[CrossRef\]](#)
- Abdelaal, M.; Fränzle, M.; Hahn, A. Nonlinear model predictive control for trajectory tracking and collision avoidance of underactuated vessels with disturbances. *Ocean Eng.* **2018**, *160*, 168–180. [\[CrossRef\]](#)

16. Yang, H.; Deng, F.; He, Y.; Jiao, D.; Han, Z. Robust nonlinear model predictive control for reference tracking of dynamic positioning ships based on nonlinear disturbance observer. *Ocean Eng.* **2020**, *215*, 107885. [\[CrossRef\]](#)
17. Zhao, R.C.; Wong, P.K.; Xie, Z.C.; Zhao, J. Real-time weighted multi-objective model predictive controller for adaptive cruise control systems. *Int. J. Automot. Technol.* **2017**, *18*, 279–292. [\[CrossRef\]](#)
18. Rankovic, V.; Radulović, J.; Grujovic, N.; Divac, D. Neural network model predictive control of nonlinear systems using genetic algorithms. *Int. J. Comput. Commun. Control* **2012**, *7*, 540–549. [\[CrossRef\]](#)
19. Ramasamy, V.; Sidharthan, R.K.; Kannan, R.; Muralidharan, G. Optimal Tuning of Model Predictive Controller Weights Using Genetic Algorithm with Interactive Decision Tree for Industrial Cement Kiln Process. *Processes* **2019**, *7*, 938. [\[CrossRef\]](#)
20. Essa, M.E.S.M.; Aboelela, M.A.S.; Hassan, M.A.; Abdrabbo, S.M. Design of model predictive force control for hydraulic servo system based on cuckoo search and genetic algorithms. *Proc. Inst. Mech. Eng. Part I J. Syst. Control Eng.* **2020**, *234*, 701–714. [\[CrossRef\]](#)
21. Zhang, L.; Zhuan, X. Model predictive control method of a parallel electromagnetic isolation system based on the improved genetic algorithm. *J. Vib. Control* **2020**, *26*, 2001–2012. [\[CrossRef\]](#)
22. Tang, P.H.; Tseng, M.H. Adaptive directed mutation for real-coded genetic algorithms. *Appl. Soft Comput. J.* **2013**, *13*, 600–614. [\[CrossRef\]](#)
23. Chuang, Y.C.; Chen, C.T.; Hwang, C. A real-coded genetic algorithm with a direction-based crossover operator. *Inf. Sci.* **2015**, *305*, 320–348. [\[CrossRef\]](#)
24. Fossen, T.I. *Guidance and Control of Ocean Vehicles*; John Wiley & Sons: Chichester, UK, 1999; pp. 5–56.
25. Fossen, T.I. *Handbook of Marine Craft Hydrodynamics and Motion Control*; John Wiley & Sons: Chichester, UK, 2011; pp. 52–59.
26. Zhu, H.; Zhao, G.; Sun, L.; Lee, K.Y. Nonlinear predictive control for a boiler–turbine unit based on a local model network and immune genetic algorithm. *Sustainability* **2019**, *11*, 5102. [\[CrossRef\]](#)
27. Veerasamy, G.; Kannan, R.; Siddharthan, R.K.; Muralidharan, G.; Sivanandam, V.; Amirtharajan, R. Integration of genetic algorithm tuned adaptive fading memory Kalman filter with model predictive controller for active fault-tolerant control of cement kiln under sensor faults with inaccurate noise covariance. *Math. Comput. Simul.* **2022**, *191*, 256–277. [\[CrossRef\]](#)
28. Mohammadi, A.; Asadi, H.; Mohamed, S.; Nelson, K.; Nahavandi, S. Multiobjective and interactive genetic algorithms for weight tuning of a model predictive control-based motion cueing algorithm. *IEEE Trans. Cybern.* **2018**, *49*, 3471–3481. [\[CrossRef\]](#) [\[PubMed\]](#)
29. Maran, F. Model-Based Control Techniques for Automotive Applications. Ph.D. Thesis, University of Padova, Padova, Italy, 2013.
30. Liu, J.; Yang, L.; Zhang, X.; Yan, R.; Chen, X. Intelligent time-domain parameters matching for shock response spectrum and its experimental validation in active vibration control systems. *Shock Vib.* **2019**, *2019*, 2920845. [\[CrossRef\]](#)
31. Tsoulos, I.G.; Tzallas, A.; Tsalikakis, D. PDoublePop: An implementation of parallel genetic algorithm for function optimization. *Comput. Phys. Commun.* **2016**, *209*, 183–189. [\[CrossRef\]](#)
32. Fannemel, Å.V. Dynamic Positioning by Nonlinear Model Predictive Control. Master’s Thesis, Institutt for Teknisk Kybernetikk, Trondheim, Norway, 2008.