INTELLIGENT CONTROL STRATEGIES USED IN FAST TIME SHIP MANOEUVRING SIMULATIONS

Changyuan Chen
PhD student, Ghent University, Changyuan.Chen@ugent.be
Marc Vantorre
Professor, Ghent University, Marc.Vantorre@ugent.be
Evert Lataire
Post-doctoral assistant, Ghent University, Evert.Lataire@ugent.be
Maxim Candries
Post-doctoral researcher, Ghent University, Maxim.Candries@ugent.be
Katrien Eloot
Professor, Ghent University, Senior Researcher, Flanders Hydraulics Research
Katrien.Eloot@mow.vlaanderen.be
Jeroen Verwilligen
Senior Researcher, Flanders Hydraulics Research, Jeroen.Verwilligen@mow.vlaanderen.be

Abstract: Fast time simulations are an efficient method to assess and analyze ship manoeuvring behaviour in confined water. The track-controller plays an important role in controlling the ship trajectory in these fast time simulations. In order to improve track-controller accuracy and adaptability in different nautical environments, different intelligent control strategies are considered to be used in the simulation. The principle and characteristics of the track-controller as used at Flanders Hydraulics Research (FHR) and Ghent University are firstly described, then advanced and smart control methods depending on its characteristics are reviewed and compared. Finally, the grey relational decision-making approach is applied in selecting the proper control strategy for the track-controller. Based upon the results from comparison, the fuzzy logic control algorithm is the most suitable for track-controller designing due to its simple control structure and good adaptability.

Keywords: Fast time simulation; Track-controller; Intelligent control strategies; Autopilot; Ship manoeuvring

1. Introduction

Fast time ship manoeuvring simulation is a common technique for checking new navigation areas, for assessing channel safety, and for analyzing confined and shallow water effects on ships. The simulation offers a cost-effective and efficient approach for identifying potential risks when designing a fairway. In a fast time simulation, the human input is eliminated and replaced by a control algorithm, here named “track-controller”. Track-controller is preferred over “autopilot” in order to avoid confusion with the on-board device with the same name. The track-controller is a control model used in the fast time simulation system, which was initially developed for track-keeping (within restricted deviations) in a confined channel. This algorithm does not only provide new control actions (rudder deflection) at discrete time intervals with the help of a prediction model, which can deal with complex, highly non-linear ship behaviour, but also controls the engine speed thus setting the propeller rate.

The track-controller, developed and in use at Flanders Hydraulics Research (FHR) and Ghent University(UGent), is taken as a case study. It is rather advanced, although the performance of this control system does not always lead to a satisfactory result. In order to optimize the present model, intelligent and advanced control strategies need to be taken into account. Apart from the classic PID control theory, fast time simulation track controllers can be based on fuzzy adaptive logic control, neural network control, and other more advanced or sophisticated tracking control algorithms.

The present paper intends to review and highlight the special requirements for fast time simulation systems with the focus on advanced and intelligent control algorithms. This paper will start with an overview of the development of control algorithms in fast time simulations, then existing intelligent control strategies are outlined and classified according to their functions, methodologies and applications. Six different control strategies are compared based on eight predefined properties.
Finally, the grey relational decision-making method is used to select the most suitable control strategy for the fast time simulation system.

2. Overview of fast time simulations

2.1 Application status of fast time simulation systems

Fast time simulation systems have been widely used. Numerous universities, government, research institutes and commercial companies have developed their own fast time simulation programs, each for different applications. Recent development of fast time simulation systems are briefly listed in Appendix A.

2.2 Principle of track-controller

The version of the track controller presently in use at FHR for fast time simulations (FTS) is a joint development of FHR and UGent. In the FTS system, the vessel’s trajectory is controlled by a track-controller which is able to run in the track keeping mode, making it possible to performance typical waterway and harbour manoeuvres [1].

Fig.1 Principle of track controller

The track-controller consists of input signals, a prediction model, a cost function as well as output signals. Fig.1 presents the principle of a track-controller in detail. Specific steps are as follows:

Step 1: Input signals
The input signals include position and course angle \((x_0, y_0, \psi_0)\), velocity components \((u_0, v_0, r_0)\), acceleration components \((\dot{u}, \dot{v}, \dot{r})\), rudder angle \((\delta)\), and propeller rate \((n)\). The desired track and speed of the vessel alongside the track are also given as input variables.

Step 2: Trajectory control
The track-controller makes its decisions based on a mathematical simulation model, named the “prediction model”, which predicts the ship’s position after each time interval \((\Delta t)\), where \(\Delta t = \xi/u\), \(\xi\) is the anticipation distance, \(u\) the velocity component). A discrete number of rudder \((\delta)\) and propeller control \((n)\) actions are combined together to estimate the future ship’s position.
Step 3: Evaluation

The cost function \( C \) is applied to evaluate the effect of each control action of the track-controller, which is a function of the predicted position. The cost function is also used to minimize the cross-track error between the real trajectory and the reference or desired trajectory.

\[
C = c_F \Delta \eta_F^2 + c_M \Delta \eta_M^2 + c_A \Delta \eta_A^2
\]  

where \( c_F, c_M \) and \( c_A \) are weight coefficients, and \( \eta_F, \eta_M \) and \( \eta_A \) are the distance at the fore perpendicular, amidship and aft perpendicular relative to the reference trajectory.

Step 4: Output signals

The output signals consist of position, heading, rudder angle and propeller rate, velocity, acceleration and forces, etc.

2.3 Case study: Canal Ghent-Terneuzen

A full-scale measurement campaign and several fast time simulations were carried out on a bulk carrier (LxRxT 229.5x36.9x12.5 m³) sailing southbound on the Canal Ghent-Terneuzen. In Fig.2, the positions of the bulk carrier from the full-scale measurement (green trajectory) is plotted together with the positions generated by the track-controller in the fast time simulations (blue trajectory). There are small deviations between the two trajectories due to track-controller parameters such as the chosen weight coefficients and anticipation time.

![Fig.2 Trajectory from the full-scale measurements and track-controller fast time simulations](image)

In order to get the optimal trajectory, different weight coefficients of cost function and anticipation times are selected.

(1) Different weight coefficients \( \{c_F, c_M \text{ and } c_A\} \) of cost function are chosen to compare the effect of track-controller in Tab.1, and trajectories of the ship’s fore perpendicular, amidship and aft perpendicular are plotted in Fig.3.

<table>
<thead>
<tr>
<th>Weight Coefficients</th>
<th>( c_F )</th>
<th>( c_M )</th>
<th>( c_A )</th>
<th>Trajectory Color</th>
</tr>
</thead>
<tbody>
<tr>
<td>CF111</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Black</td>
</tr>
<tr>
<td>CF151</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>Pink</td>
</tr>
<tr>
<td>CF1101</td>
<td>1</td>
<td>10</td>
<td>1</td>
<td>Blue</td>
</tr>
<tr>
<td>Real trajectory</td>
<td></td>
<td></td>
<td></td>
<td>Red</td>
</tr>
</tbody>
</table>
Fig.3 Comparison results among different weight coefficients

(2) Different anticipation times of the track-controller are selected to compare the effect of the track-controller in Tab.2, and trajectories of the ship’s fore perpendicular, amidship and aft perpendicular are plotted in Fig.4.

Tab.2 Comparison among different anticipation times

<table>
<thead>
<tr>
<th>Types</th>
<th>Anticipation Times (s)</th>
<th>Trajectory Color</th>
</tr>
</thead>
<tbody>
<tr>
<td>LH5</td>
<td>5</td>
<td>Red</td>
</tr>
<tr>
<td>LH50</td>
<td>50</td>
<td>Black</td>
</tr>
<tr>
<td>LH100</td>
<td>100</td>
<td>Blue</td>
</tr>
<tr>
<td>Real trajectory</td>
<td></td>
<td>Pink</td>
</tr>
</tbody>
</table>

Fig.4 Comparison results among anticipation times

2.4 Characteristics of the track-controller

After comparing and analyzing the fast time simulation results, the characteristics of the track-controller can be summarized as follows:
In comparison to real time simulations, the following observations are made:

- No visuals have to be developed, the simulations do not need to be performed at a full mission bridge simulator.
- The track controller does not have to perform the simulations in real time, so that many runs can be performed in a much shorter time span and can quickly provide the ship trajectory distribution in the channel.
- Results are objective and repeatable.

On the contrary, a number of challenges have been identified during recent applications:

- The three weight coefficients (2 independent) of the cost function are difficult to be selected to gain the best performance of fast time simulation.
- A different anticipation time is the key part of the fast time track controller. The anticipation time parameter need to be manually selected instead of an automatic selection for different navigation environments.

In order to optimize the present track-controller, intelligent control algorithms should be taken into consideration like adaptive fuzzy logic, neural network and genetic algorithm as well as their hybrid, etc. Intelligent control algorithms are used to design and optimize the existing track-controller to satisfy various needs, such as berthing, unberthing and overtaking manoeuvres.

3. Intelligent control strategies review

Different control approaches have been applied in control systems. This section reviews and discusses different control methods in order to improve fast time simulation system performance.

3.1 PID control strategy

PID controller description

A PID (Proportional-Integral-Derivative) controller is a classic control method that is probably most widely applied in ship control systems, due to its simple structure, ease of design, and low cost in implementation [2].

The principle of a PID controller is to continuously calculate an error value as the distance between a measured position and desired position, in order to minimize the error by adjusting input course, the PID controller theory is as follows [3].

\[
\delta(t) = k_p e(t) + k_i \int_0^t e(t) \, dt + k_d \frac{de(t)}{dt}
\]

(2)

\[e(t) = \varphi_{in}(t) - \varphi_{out}(t)
\]

(3)

Where \( k_p \) is the proportional coefficient, \( k_i \) is the integral coefficient, \( k_d \) is the derivative coefficient, and \( e(t) \) is an error between a measured course and desired course.

Brief review of PID controller application

The PID controller was firstly implemented in automatic ship steering system by Minorsky in 1922, which was a pioneer in the field of ship control [4].

Kumar et al. reviewed classic PID control theory in the chronological order. The advantages and drawbacks of PID controller were reported. Since classic PID controller cannot deal with complex and nonlinear systems, fuzzy PID controller was designed, analyzed and discussed in detail [5].

Regarding PID controller gain tuning methods, Zhang et al. introduced a novel approach to solve PID parameters based on closed-loop gain shaping algorithm [6]. The algorithm did not only have simple calculating procedure and obvious physical sense, but also had good control performance and robustness. Apart from that, several approaches were applied in tuning PID parameters, such as the “Good Gain” method and the “Ziegler-Nichols” method based on experiments [7] and the Skogestad’s method based on transfer function model [8].
A PID compensating model was proposed by Kwon et al. Comparing with the conventional PID control algorithms, the designed control model can compensate nonlinear characteristics and reduce the overshoot. The simulation results achieved good control performance [9].

Fossen designed a PID control algorithm used in SISI and MIMO ship control systems, and optimized PID control algorithm with the aid of acceleration feedback [10]. The research topic was also discussed by Lindegaard [11].

An adaptive PID controller had been designed and analyzed based on model reference adaptive control algorithm and self-tuning regulator control algorithm by Nguyen et al. The result of simulation showed that there was notable improvement in accuracy, nonlinear control and smaller tuning time [12].

A robust PID control algorithm was designed based on the closed-loop gain shaping approach in order to enhance control performance and system stability. Comparing with the fuzzy control method, the designed robust PID control algorithm was simple and easy to apply for practical engineering [13].

Zribi et al. presented a new PIDNN (PID Neural Network) control system based on the adaptive tuning approach [14].

### 3.2 Fuzzy Logic control (FLC) strategy

**Fuzzy Logic controller description**

![Fuzzy Logic Controller Diagram](image)

**Fig.5 Principle of fuzzy logic controller**

The qualitative knowledge of fuzzy logic controller is used to design a practical controller. The fuzzy logic control algorithm has been considered as an intelligent method to deal with a complex system including nonlinearity and uncertain dynamics, etc. It consists of five major components: fuzzifier, knowledge base, rule base, inference engine, defuzzifier [15] [16]. Fig.5 shows the principle of a fuzzy logic controller. The structure is composed of the following elements:

- **Fuzzifier** – The role of the fuzzifier is to convert the crisp input values into fuzzy values.
- **Fuzzy Knowledge Base** – It stores the knowledge about all the input-output fuzzy relationships.
- **Fuzzy Rule Base** – It stores the knowledge about the operation of the process of the domain.
- **Inference Engine** – It acts as a kernel of any FLC. Basically it simulates human decisions by performing approximate reasoning.
- **Defuzzifier** – The role of the defuzzifier is to convert the fuzzy values into crisp values getting from the fuzzy inference engine.

**Brief review of fuzzy logic controller application**

In 1955, Zadeh was a pioneer who proposed the "Fuzzy Sets" theory, which laid the foundation for the application of the fuzzy logic algorithm [17]. In 1965, Lotfi A. Zadeh published "Fuzzy Sets" book which laid out the mathematics of fuzzy set theory [18] [19].

Van Amerongen firstly investigated the ship control algorithm designed with fuzzy set theory in 1997. Two inputs, five linguistic variables together with a fixed rule base were used in the proposed control
system. Comparing with the traditional PID controller, the fuzzy controller presented a significantly enhanced performance in external disturbance. The pioneering work was followed by various applications in the area of ship control system [20].

An automatic ship controller using a multivariable fuzzy logic algorithm was proposed by Seema Singh et al. The control approach was applied in path following, track keeping and collision avoidance systems, which performed better and eased to be utilized in control system [21].

Fraga et al. designed a double fuzzy logic controller to steer the ship following a desired path. The proposed algorithm was composed of inner and outer loop controllers considering environmental disturbances (wind, current, wave). The stability, robustness and effectiveness were verified by numerical simulations [22].

A ship control regulator in manoeuvring simulations with three fuzzy logic controllers was presented by Morawski. Straight and turning manoeuvring of a ship were controlled by the proposed regulator, which was established in Matlab with the aid of the graphical interface [23].

Bahttacharyya designed a fuzzy logic control algorithm for surface ship motions, and its performance for straight line or curved trajectories were verified [24]. Zalewski presented a fuzzy fast time simulation (FTS) algorithm used in confined waters. The novel model was a fuzzy logic controller based on an expert database, which was collected from real time simulation and expertise. Fuzzy FTS showed satisfactory results after several trials [25][26].

3.3 Genetic algorithm control strategy

**Genetic controller description**

A genetic algorithm (GA) reflects the process of natural selection where the fittest individuals are selected for reproduction [27]. The following steps are included in a GA:

1. Randomly generate the initial population (t)
2. Determine fitness of generated population (t)
3. Repeat
   - Select parents from population (t)
   - Perform crossover on parents to create population (t+1)
   - Perform mutation in population (t+1)
   - Determine population fitness (t+1)
4. Loop until an individual is good enough to meet the criteria.

Fig.6 shows the flow diagram of genetic algorithm.

![Fig.6 Generic algorithm flow diagram](image)

**Brief review of genetic controller application**
Larrazabal et al. presented an intelligent rudder control approach based on the genetic algorithm [4]. The algorithm used PID controller for different control points with optimized parameters by genetic algorithm. Compared to the conventional PID control algorithm and fuzzy logic control algorithm, the designed control algorithm led to more satisfactory results. The simulation results showed that the combination of intelligent and traditional control algorithms was a good way to address environmental uncertainty.

McGookin et al. optimized the non-linear control system by genetic algorithm to take changing environment factors into account [28], which was a GA controller applied in track keeping and course changing systems.

The genetic algorithm was also used to optimize gains of PID systems. Manual tuned gain values of PID controller were compared by the genetic algorithm [29-31]. Simulation results showed that genetic algorithm performed better and was more suitable to design a controller with unfamiliar control system.

Xiao et al. designed a ship course controller based on active disturbance rejection approach. The proposed controller was optimized by the genetic algorithm to improve the controller’s adaptive ability. The adaptabilities and robustness of the control strategies were validated by numerical simulations [32].

A nonlinear control system was designed based on backstepping approach. The parameters of the nonlinear controller were tuned and optimized by genetic algorithm to get better nonlinear control performance compared with conventional PID controller [33].

Alfaro-Cid et al. applied sliding mode controller to the propulsion and navigation systems [34]. Parameters of sliding mode controller was optimized by genetic algorithm. The robustness and capacity of resisting disturbance of proposed controller were significantly enhanced.

An autopilot was designed based on a fuzzy logic controller whose fuzzy rules were tuned and optimized by the genetic algorithm [35]. The autopilot was used to pilot the ship following the “visual course angle” calculated from desired course, real course and track deflection. Simulation results showed that the proposed control algorithm enabled the ship to converge to desired track satisfactorily.

3.4 Neural network (NN) control strategy

Neural network controller description

Neural network systems are one of the major intelligent algorithms used in the field of ship controlling. Neural networks mimic the human learning process to solve problems including complicated and highly non-linear systems. After training, neural networks can be utilized to predict the output data based on input data [36]. Neural network’s structure consists of an input layer, a hidden layer and an output layer (see Fig.7).

![Fig.7 Neural network structure](image)
Brief review of neural network controller application

McCulloch and Pitts designed the first neural network system in 1943, various engineering applications were discussed [37].

Zhang et al. proposed the SIMO neural network control strategy for the ship track-keeping issue [38]. It utilized the rudder to regulate both ship heading error and tracking error. Simulation results illustrated a better training effect and robust performance could be achieved.

Zhang et al. designed a ship course-keeping controller by combining neural network algorithm with backstepping algorithm in order to deal with environmental disturbances and uncertainty for nonlinear control system [39]. The effectiveness and robustness of the proposed control algorithm were verified by Matlab simulation and full scale measurement test.

An autonomous collision avoidance approach based on convolution neural network was proposed by Xu et al. The aim of the proposed algorithm was to improve the automation level of marine ship. It can predict the ship position, guide the ship, and control the ship navigation. Effectiveness and stability of the neural network control strategy was verified by simulation [40].

Velagic designed an efficient artificial neural network algorithm to identify a nonlinear ship controller and a fuzzy logic ship controller [41]. The comparison indicated the effectiveness and superiority of proposed neural network algorithm.

Bong et al. proposed an neural network algorithm based on out-feedback controller in order to estimate the velocity of the underactuated surface vessels with input saturation and uncertainties [42]. Feedback controller and Lyapunov analysis were applied to improve the performance of the neural network controller.

Lacki provided an intelligent method to predict ships maneuvering in confined water to alert the navigator of incoming threat and increase navigation safety. It is based on neuron evolution and artificial neural network theory [43].

Li et al. introduced an adaptive neural network controller of surface vessels for fine trajectory tracking. Radial basis neural network was used to approximate the unknown nonlinear dynamics of ships. Lyapunov theory and backstepping method were applied to analyze the system stability [44]. The proposed controller could converge both tracking errors and velocity error to zero.

3.5 $H^\infty$ robust control strategy

$H^\infty$ robust controller description

Robust control theory is an approach to design robust controllers, which can assess the performance changes of a control system with changing system parameters. The aim of designing robust controllers is to enhance the robustness and stability of systems.

Brief review of $H^\infty$ robust controller application

In 1981, Zames introduced the $H\infty$ robust control theory, which marked the birth of it [45]. The robust controller was applied in automatic steering of ships in order to achieve system stability and robust performance with environmental disturbance. Weighting functions were selected to deal with disturbance rejection and rudder saturation limit [46].

X K. Zhang designed a track-keeping controller with the aid of robust control theory and Matlab robust control toolbox [47]. The robustness of the controller was presented by satisfactory performance under various environmental disturbances, loading conditions and ship speeds, etc.

Sheng et al. designed a robust controller to handle uncertainty of system parameters and environmental disturbance [48]. Compared with the PID controller, the designed system had better robust performance and higher course precision.

Wang, et al. applied mixed-sensitivity approach of $H\infty$ robust control to simplify the design of path tracking controllers and address nonlinear dynamic model problems of an autonomous underwater vehicle [49].
Hu et al. designed a nonlinear robust controller by using $H_{\infty}$ input/output linearization formulation and $\mu$-synthesis approach [50]. The aim of the control model was to improve robust stability and robust performance for various uncertainties.

### 3.6 Expert system control strategy

#### Expert system controller description

An expert system is a evaluating system, which can simulate behavior and judgment of a human who has experience and expert knowledge with the aid of artificial intelligent technologies (a computer program) [51]. An expert system is an important component and widely applied in designing the ship track controller.

#### Brief review of expert system controller application

Shuo et al. reviewed intelligent control algorithms including expert system, which was applied in ship control system [52].

Nikitakos et al. presented a method of adjusting ship control system based on expert system [53].

Haibin et al. designed a PID autopilot whose parameters were optimized by a self-setting expert system. Comparing with conventional PID control strategies, the proposed control system could perform better in gain tuning and adaptive ability [54] [55].

### 3.7 Comparison among different control strategies

PID, fuzzy logic, neural network and genetic algorithms are important branches of intelligent control technologies. Different control technologies have their own significant advantages, but there are still some shortcomings in various intelligent control strategies. Taking neural network algorithm as an example, the structure and function of biological neural networks are mimicked to approximate unknown functions and identify nonlinear and uncertainty systems, etc. Its outstanding advantages are excellent function approximation, self-learning, adaptive and fault tolerance abilities. At the same time, the neural network algorithm has good optimization and online processing capabilities. On the contrary, neural network algorithms cannot be calculated with accurate mathematical description, so the algorithm has poor knowledge expression.

After reviewing relevant literatures, eight properties (learning ability, processing nonlinearity, processing uncertainty,...) are selected to compare the disadvantages and advantages in mentioned control algorithms. The ability for each control strategy is appreciated in five classes: poor, fair, average, good and excellent based upon their characteristics of six controllers. Tab.3 and Fig.8 shows the comparison results.

#### Tab.3 Comparison among different control strategies

<table>
<thead>
<tr>
<th></th>
<th>PID</th>
<th>$H_{\infty}$</th>
<th>FLC</th>
<th>NN</th>
<th>GA</th>
<th>ES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mathematical Description</td>
<td>Excellent</td>
<td>Excellent</td>
<td>Good</td>
<td>Poor</td>
<td>Poor</td>
<td>Poor</td>
</tr>
<tr>
<td>Learning Ability</td>
<td>Poor</td>
<td>Poor</td>
<td>Poor</td>
<td>Excellent</td>
<td>Good</td>
<td>Poor</td>
</tr>
<tr>
<td>Optimization Ability</td>
<td>Fair</td>
<td>Fair</td>
<td>Poor</td>
<td>Good</td>
<td>Excellent</td>
<td>Poor</td>
</tr>
<tr>
<td>Knowledge Expression</td>
<td>Fair</td>
<td>Fair</td>
<td>Excellent</td>
<td>Poor</td>
<td>Fair</td>
<td>Excellent</td>
</tr>
<tr>
<td>Fault Tolerance</td>
<td>Poor</td>
<td>Excellent</td>
<td>Excellent</td>
<td>Excellent</td>
<td>Excellent</td>
<td>Poor</td>
</tr>
<tr>
<td>Realtime Performance</td>
<td>Excellent</td>
<td>Excellent</td>
<td>Excellent</td>
<td>Good</td>
<td>Fair</td>
<td>Poor</td>
</tr>
<tr>
<td>Processing Nonlinearity</td>
<td>Poor</td>
<td>Poor</td>
<td>Excellent</td>
<td>Excellent</td>
<td>Excellent</td>
<td>Fair</td>
</tr>
<tr>
<td>Processing Uncertainty</td>
<td>Poor</td>
<td>Excellent</td>
<td>Excellent</td>
<td>Excellent</td>
<td>Excellent</td>
<td>Poor</td>
</tr>
</tbody>
</table>

Excellent(5); Good(4); Average(3); Fair(2); 5.Poor(1).
4. Grey relational decision making model

In order to choose the suitable control algorithm for fast time simulation system, the grey relational decision-making model is used. This is a classic multi-criteria decision making method, which can select best model according to the properties of different models [56].

4.1 Grey relational decision-making model

(1) Initial decision matrix

n different alternatives and m attributes are selected to compose the initial decision matrix. In initial decision matrix, the reference series $X_0 = (x_0(1), x_0(2), \ldots, x_0(m))$ is determined, $x_0(i)$ is the maximum in $i^{th}$ attributes [57].

$$(X_0, X_1, \ldots, X_n) = \begin{bmatrix} x_0(1) & x_1(1) & \cdots & x_n(1) \\ x_0(2) & x_1(2) & \cdots & x_n(2) \\ \vdots & \vdots & \ddots & \vdots \\ x_0(m) & x_1(m) & \cdots & x_n(m) \end{bmatrix} \quad (0 \leq i \leq n, 1 \leq j \leq m) \quad (4)$$

(2) Non-dimensional initial matrix

$$(X'_0, X'_1, \ldots, X'_n), X'_i(j) = X_i(j)/X_0(j) \quad (5)$$

$$(X'_0, X'_1, \ldots, X'_n) = \begin{bmatrix} x'_0(1) & x'_1(1) & \cdots & x'_n(1) \\ x'_0(2) & x'_1(2) & \cdots & x'_n(2) \\ \vdots & \vdots & \ddots & \vdots \\ x'_0(m) & x'_1(m) & \cdots & x'_n(m) \end{bmatrix} \quad (0 \leq i \leq n, 1 \leq j \leq m) \quad (6)$$

(3) Calculate the difference $\Delta_0(i)$ between the alternative series and reference series [58].

$$\Delta_0(i) = \left| x'_0(i) - x'_i(j) \right|, \quad (0 \leq i \leq n, 1 \leq j \leq m) \quad (7)$$
Grey relational coefficient

Calculate the grey relational coefficient between the alternative series and reference series using the following equation:

\[ \xi_{0i}(j) = \frac{\min_i \Delta_{0i}(j) + \rho \max_i \Delta_{0j}(j)}{\Delta_{0i}(j) + \rho \max_j \Delta_{0j}(j)} \]  

where the identification coefficient \( \rho = 0.5 \), which provides a good stability for calculation.

The grey relational coefficient matrix is as follows:

\[
\xi = \begin{pmatrix}
\xi_{01}(1) & \xi_{02}(1) & \cdots & \xi_{0u}(1) \\
\xi_{01}(2) & \xi_{02}(2) & \cdots & \xi_{0u}(2) \\
\vdots & \vdots & \ddots & \vdots \\
\xi_{01}(m) & \xi_{02}(m) & \cdots & \xi_{0u}(m)
\end{pmatrix}
\]

(10)

5 Calculate the grey relational degree \( \gamma_{0i} \)

\[ \gamma_{0i} = \frac{1}{m} \sum_{j=1}^{m} \xi_{0i}(j) \]  

The grey relational degree \( \gamma_{0i} \) can be used to rank the alternatives by the similarity between the alternative series and reference series. The best alternative can be selected according to the maximum of grey relational degree \( \gamma_{0i} \) [59].

4.2 Application of grey relational making model

Six different control strategies (PID, fuzzy logic...) are compared based upon eight predefined properties (learning ability, processing nonlinearity, processing uncertainty,...).

(1) Suppose six different control strategies as alternative series and eight different predefined properties as attribute series, with evaluation: Excellent (5); Good (4); Average (3); Fair (2); Poor (1).

The initial decision matrix is as follows:

\[
\left( X_0, X_1, \cdots X_n \right) = \begin{pmatrix}
5 & 5 & 5 & 4 & 1 & 1 & 2 \\
5 & 1 & 1 & 1 & 5 & 4 & 1 \\
5 & 2 & 2 & 1 & 4 & 5 & 1 \\
5 & 2 & 2 & 5 & 1 & 2 & 5 \\
5 & 1 & 1 & 5 & 5 & 5 & 1 \\
5 & 5 & 5 & 4 & 2 & 1 & 5 \\
5 & 1 & 1 & 5 & 5 & 5 & 2 \\
5 & 1 & 1 & 5 & 5 & 5 & 1
\end{pmatrix}
\]

(12)

(2) Non-dimensional initial matrix:
(3) Difference \( \Delta_0 \) matrix between the alternative series and reference series

\[
\Delta = \begin{bmatrix}
0 & 0 & 0.2 & 0.8 & 0.8 & 0.6 \\
0.8 & 0.8 & 0.8 & 0 & 0.2 & 0.8 \\
0.6 & 0.6 & 0.8 & 0.2 & 0 & 0.8 \\
0.6 & 0 & 0.8 & 0.6 & 0 \\
0.8 & 0 & 0 & 0 & 0 & 0.8 \\
0 & 0 & 0.2 & 0.6 & 0.8 & 0.8 \\
0 & 0 & 0 & 0 & 0.8 & 0 \\
0 & 0 & 0 & 0 & 0 & 0.8 \\
\end{bmatrix}
\]  

(14)

(4) Grey relational coefficient matrix

\[
\bar{\gamma} = \begin{bmatrix}
1.000 & 1.000 & 0.667 & 0.333 & 0.333 & 0.400 \\
0.333 & 0.333 & 0.333 & 1.000 & 0.667 & 0.333 \\
0.400 & 0.400 & 0.333 & 0.667 & 1.000 & 0.333 \\
0.400 & 0.400 & 1.000 & 0.333 & 0.400 & 1.000 \\
0.333 & 1.000 & 1.000 & 1.000 & 1.000 & 0.333 \\
1.000 & 1.000 & 1.000 & 0.667 & 0.400 & 0.333 \\
0.333 & 0.333 & 1.000 & 1.000 & 1.000 & 0.400 \\
0.333 & 1.000 & 1.000 & 1.000 & 1.000 & 0.333 \\
\end{bmatrix}
\]  

(15)

(5) Grey relational degree \( \gamma_0 \)

\[
\gamma_0 = \begin{bmatrix} 0.517 & 0.683 & 0.792 & 0.750 & 0.725 & 0.433 \end{bmatrix}
\]  

(16)

<table>
<thead>
<tr>
<th>Alternative</th>
<th>PID</th>
<th>H(\infty)</th>
<th>FLC</th>
<th>ANN</th>
<th>GA</th>
<th>ES</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\gamma_0)</td>
<td>0.517</td>
<td>0.683</td>
<td>0.792</td>
<td>0.750</td>
<td>0.725</td>
<td>0.433</td>
</tr>
<tr>
<td>Rank</td>
<td>5</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>6</td>
</tr>
</tbody>
</table>

From Tab.4, the grey relational degree value of the fuzzy logic control algorithm is the highest followed by neural network and genetic systems, and the value of expert system is the lowest. Calculation results shows the fuzzy logic control algorithm is more suitable to the fast time simulation system.

5. Conclusions

This paper concentrates on enhancing the performance of the track controller in a fast time simulation system. Various intelligent control approaches (PID, Fuzzy logic, and ANN, etc.) are reviewed and
compared according to their properties. The grey relational decision-making method is applied in selecting an appropriate control strategy for fast time simulation system. Comparing and analysing the results illustrate that the fuzzy logic control algorithm can perform better than other control algorithms, with the maximum of grey relational degree as 0.792. This is due to its outstanding abilities to deal with nonlinearity, uncertainty and environmental disturbances system like the FTS system. The algorithm is simple, easy to implement, and has potential application in the FTS system.

6. Acknowledgements

This work was supported by the program of China Scholarships Council (No.201706570007) and Special Research Fund - Cofunding for Chinese candidates holding a CSC-grant (01SC8418).

Appendix A.

Tab.5 Application status of fast time simulation systems

<table>
<thead>
<tr>
<th>NO.</th>
<th>Name of Simulator</th>
<th>Research Institutes</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Fast time simulator</td>
<td>FHR and UGent</td>
<td><a href="https://biblio.ugent.be/publication/398813">https://biblio.ugent.be/publication/398813</a></td>
</tr>
<tr>
<td>3</td>
<td>SAMMON ISSIMS GmbH</td>
<td><a href="https://issims-gmbh.com/yoomla/products/simopt">https://issims-gmbh.com/yoomla/products/simopt</a></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>SEAMAN SSPA</td>
<td></td>
<td><a href="https://www.sspa.se/tools-and-methods/simulations">https://www.sspa.se/tools-and-methods/simulations</a></td>
</tr>
<tr>
<td>11</td>
<td>Fast Time Simulator Marine Institute of Memorial University</td>
<td></td>
<td><a href="https://www.mi.mun.ca/departments/centreformariesimulation/simulators/fasttimesimulator/">https://www.mi.mun.ca/departments/centreformariesimulation/simulators/fasttimesimulator/</a></td>
</tr>
</tbody>
</table>

7. References


