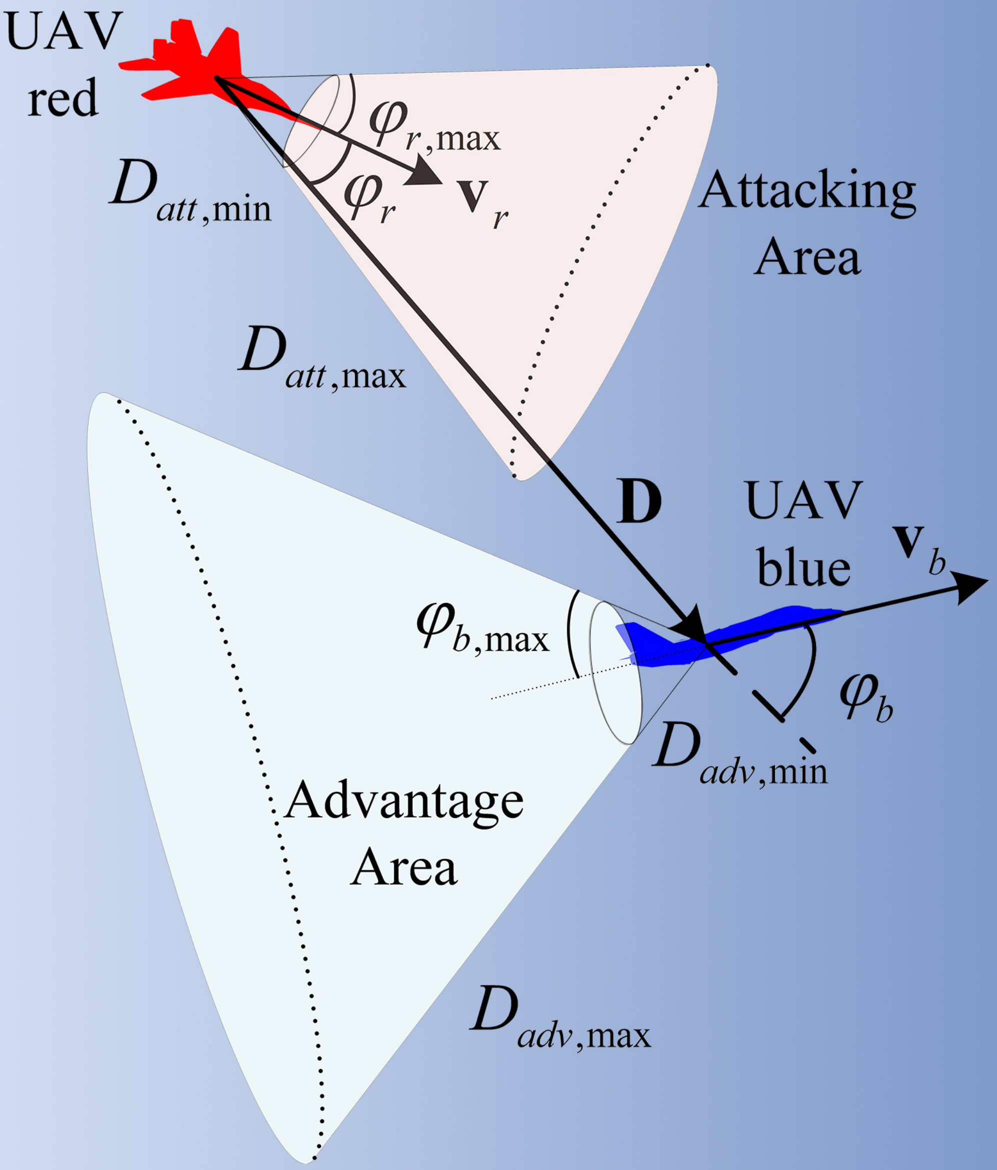
Volume 3, Issue 1, March 2023

INTELLIGENCE & ROBOTICS



UAV maneuver decision-making via deep reinforcement learning for short-range air combat

Zhiqiang Zheng, Haibin Duan*

Open Access

ISSN 2770-3541 (Online)

www.intellrobot.com

Editor-in-Chief

Simon X. Yang

Prof. Simon X. Yang is currently the Head of the Advanced Robotics and Intelligent Systems Laboratory at the University of Guelph. His research interests include artificial intelligent, robotics, sensors and multi-sensor fusion, wireless sensor networks, control systems, bio-inspired intelligence, machine learning, neural networks, fuzzy systems, and computational neuroscience.

Our Features

(1) Gold Open Access
(2) Strong Editorial Board
(3) Rigorous Peer-review
(4) Discounted English language Editing Service
(5) Online First Once Accepted
(6) Wide Promotion (Twitter\LinkedIn\WeChat\Facebook)

Editorial Board

- 1 Editor-in-Chief
- 2 Executive Editor
- 6 Advisory Editorial Members
- 49 Associate Editors
- 24 Youth Editorial Board Members

Scope

Top-quality unpublished original technical and non-technical application-focused articles are welcome from intelligence and robotics, particularly on the interdisciplinary areas of intelligence and robotics, including but not limited to the following areas:

- biological, bio-inspired, and artificial intelligence;
- neural networks, fuzzy systems, and evolutionary algorithms;
- sensing, multi-sensor fusion, localization, data analysis, modeling, planning, and control for various mobile, aerial, and underwater robotic systems;
- robot cooperation, teleoperation, and human-machine interactions;
- development and maintenance of real-world intelligent and robotic systems by multidisciplinary teams of scientists and engineers.

Journal Home
https://intellrobot.com/

Submission Link
https://oaemesas.com/login?JournalId=ir

EDITORIAL BOARD

EDITORIAL BOARD

GENERAL INFORMATION

About the Journal

Intelligence & Robotics (*IR*), ISSN 2770-3541 (Online), publishes top-quality unpublished original technical and non-technical application-focused articles on intelligence and robotics, particularly on the interdisciplinary areas of intelligence and robotics. The Journal seeks to publish articles that deal with the theory, design, and applications of intelligence and robotics, ranging from software to hardware. The scope of the Journal includes, but is not limited to, biological, bio-inspired, and artificial intelligence; neural networks, fuzzy systems, and evolutionary algorithms; sensing, multi-sensor fusion, localization, data analysis, modeling, planning, and control for various mobile, aerial, and underwater robotic systems; and robot cooperation, teleoperation and human-machine interactions. The Journal would be interested in distributing development and maintenance of real-world intelligent and robotic systems by multidisciplinary teams of scientists and engineers.

Information for Authors

Manuscripts should be prepared in accordance with Author Instructions.
Please check https://intellrobot.com/pages/view/author_instructions for details.
All manuscripts should be submitted online at https://oaemesas.com/login?JournalId=ir.

Copyright

Permissions

For information on how to request permissions to reproduce articles/information from this journal, please visit www.intellrobot.com.

Disclaimer

The information and opinions presented in the journal reflect the views of the authors and not of the journal or its Editorial Board or the Publisher. Publication does not constitute endorsement by the journal. Neither the *IR* nor its publishers nor anyone else involved in creating, producing or delivering the *IR* or the materials contained therein, assumes any liability or responsibility for the accuracy, completeness, or usefulness of any information provided in the *IR*, nor shall they be liable for any direct, indirect, incidental, special, consequential or punitive damages arising out of the use of the *IR*. *IR*, nor its publishers, nor any other party involved in the preparation of material contained in the *IR* represents or warrants that the information contained herein is in every respect accurate or complete, and they are not responsible for any errors or omissions or for the results obtained from the use of such material. Readers are encouraged to confirm the information contained herein with other sources.

Published by

OAE Publishing Inc.
245 E Main Street Ste 107, Alhambra CA 91801, USA
Website: www.oaepublish.com

Contacts

E-mail: editorial@intellrobot.com
Website: www.intellrobot.com

Intelligence & Robotics

CONTENTS

Volume 3, Issue 1

Review

1-22 **Formation control of multiple autonomous underwater vehicles: a review**
Tao Yan, Zhe Xu, Simon X. Yang, S. Andrew Gadsden
Intell Robot 2023;3(1):1-22. http://dx.doi.org/10.20517/ir.2023.01

23-55 **An overview of intelligent image segmentation using active contour models**
Yiyang Chen, Pengqiang Ge, Guina Wang, Guirong Weng, Hongtian Chen
Intell Robot 2023;3(1):23-55. http://dx.doi.org/10.20517/ir.2023.02

Research Article

56-75 **GMAW welding procedure expert system based on machine learning**
Xuewu Wang, Qian Chen, Hao Sun, Xiuwei Wang, Huaicheng Yan
Intell Robot 2023;3(1):56-75. https://dx.doi.org/10.20517/ir.2023.03

76-94 **UAV maneuver decision-making via deep reinforcement learning for short-range air combat**
Zhiqiang Zheng, Haibin Duan
Intell Robot 2023;3(1):76-94. http://dx.doi.org/10.20517/ir.2023.04

95-112 **Human gait tracking for rehabilitation exoskeleton: adaptivefractional order sliding mode control approach**
Yuan Zhou, Zhe Sun, Bo Chen, Guangpu Huang, Xiang Wu, Tian Wang
Intell Robot 2023;3(1):95-112. http://dx.doi.org/10.20517/ir.2023.05

Yan *et al*. *Intell Robot* 2023;3(1):1-22
DOI: 10.20517/ir.2023.01

Intelligence & Robotics

Review

Open Access

Formation control of multiple autonomous underwater vehicles: a review

Tao Yan[1], Zhe Xu[1], Simon X. Yang[1], S. Andrew Gadsden[2]

[1]School of Engineering, University of Guelph, Guelph, Ontario N1G 2W1, Canada.
[2]Department of Mechanical Engineering, McMaster University, Hamilton, Ontario L8S 4L8, Canada.

Correspondence to: Prof. Simon X. Yang, School of Engineering, University of Guelph, 50 Stone Road East, Guelph, Ontario N1G 2W1, Canada. E-mail: syang@uoguelph.ca

How to cite this article: Yan T, Xu Z, Yang SX, Gadsden SA. Formation control of multiple autonomous underwater vehicles: a review. *Intell Robot* 2023;3(1):1-22. http://dx.doi.org/10.20517/ir.2023.01

Received: 24 Sep 2022 **First Decision:** 9 Nov 2022 **Revised:** 23 Nov 2022 **Accepted:** 19 Dec 2022 **Published:** 14 Jan 2023

Academic Editor: Chen Peng **Copy Editor:** Yinyin Han **Production Editor:** Yinyin Han

Abstract

This paper presents a comprehensive overview of recent developments in formation control of multiple autonomous underwater vehicles (AUVs). Several commonly used structures and approaches for formation coordination are listed, and the advantages and deficiencies of each method are discussed. The difficulties confronted in synthesis of a practical AUVs formation system are clarified and analyzed in terms of the characteristic of AUVs, adverse underwater environments, and communication constraints. The state-of-the-art solutions available for addressing these challenges are reviewed comprehensively. Based on that, a brief discussion is made, and a list of promising future work is pointed out, which aims to be helpful for the further promotion of AUVs formation applications.

Keywords: Autonomous underwater vehicles (AUVs), formation control, challenges and difficulties, state-of-the-art solutions

1. INTRODUCTION

Unmanned underwater vehicles (UUVs) as effective devices have played a key role in exploration and exploitation of marine resources for human beings since about 1960s. In particular, remotely operated vehicles (ROVs)[1,2] as the typical UUVs, tethered with a cable used for data transmission and power supply, have been widely applied in the oil and gas industry and other common fields where underwater inspection, maintenance and intervention necessitate. Nonetheless, such a vehicle does not seem to be cost-effective, since in order to execute missions, an associated support vessel and operators are always needed. While there has been a so-called fully autonomous ROV recently without requiring any intervention from operators, the tether or supply ship can still not be removed, which sometimes yields the most expenditure[3,4]. To overcome this drawback while improving working performance, autonomous underwater vehicles (AUVs) have received increasing attention in the past few decades from both industry and academia. As an obvious distinction from the ROVs, AUVs get rid of the tethers, operate automatically for the assigned missions, and require neither human intervention nor support ships. Therefore, the cost of each operation can be reduced to a great extent, and moreover, due to the characteristics mentioned, AUVs can even be adopted to access restrictive areas, e.g., shallow water or under-ice areas. The growing applications in practice can even be found as the efficient alternative to ROVs[5–10]; for example, they are used frequently to perform various survey and inspection tasks in the deep sea, such as hydrographic survey (i.e., positioning or locating the underwater surroundings), submarine cable inspection, oil and gas pipeline inspection and maintenance, etc. Besides, such mechatronic systems are also helpful for the science purpose; that is, a great many marine organisms can be discovered and identified with the assistance of high-quality cameras, and many ancient shipwrecks can be surveyed and excavated using these effective tools. In addition to that, AUVs have even been employed by several navies for military use, such as mine neutralization, intelligence gathering, reconnaissance or even nuclear bomb searching, etc.

However, as the complexity of missions grows, it is becoming nearly impossible for a single AUV system to attain a satisfactory result. As a consequence, along with the recent advances of the multi-agent systems theory, the developments of multiple autonomous underwater vehicle systems have received much attention in communities of control and ocean engineering over the past few decades[11,12]. It is natural that a vast amount of difficulties can be handled efficiently by employing a group of AUVs. Besides that, such a multi-agent system has many beneficial inherent features, including high degree of flexibility, ease of extension and maintenance, and better robustness against perturbations and failures[13,14], etc.

The research efforts on a swarm of AUVs are mainly focused on designing efficient protocols and algorithms such that some useful collective behaviors among individuals can emerge, which is roughly categorized into problems such as formation control, flocking, hunting, pursuit-evasion[15–20], etc. It is observed from literature review that formation control of AUVs fleet has attracted the most attention among those research topics, due to its considerable potential in practical maritime operations, which also motivates this brief survey. Notice that there are several definitions of formation control found in the literature[21]. For example, some definitions are either forcing the constraints on relative positions of agents or just simply demand speed synchronization of each individual (without the need to maintain a specific distance from each other). In this article, formation control is referred to as designing controllers for a fleet of AUVs so that some prescribed formation shapes can be formed and kept, and meanwhile, the velocities of each AUV in the group are agreed to move along with a desired route as a whole. In short, formation control attempts to control the relative distance and bearing between vehicles while maneuvering together. It is worth noting that AUV formation is totally different from the problem of AUV-assisted underwater acoustic networks (UANs)[22,23], while they both seek to employ multiple AUVs to construct a communication network. In vehicle assisted UANs, AUVs play a role in shortening the distance of information connection from the stationary nodes to the surface vessels or buoy beacons. In such a way, the communication quality can be improved considerably through AUV relay. Nonetheless, in this process, the shapes of the AUVs fleet are not necessarily constrained, and indeed people are concerned more about the AUVs path planning or task assignment so as to obtain the shortest routes with relatively lower energy consumption[24]. In contrast, as mentioned earlier, the AUVs formation control is mainly focusing on the design of efficient and robust control techniques for AUVs in order to strictly meet the prescribed formation constraints. The recent advances in sensor technologies and algorithms used for underwater localization and navigation can be found in the survey papers[25,26], which also serve as critical factors contributing to better formation control performance. This paper is mainly from a control point of view to examine the AUV formation applications. In addition, it is worth noting that there exist three types of AUVs, i.e., fish-like AUV, underwater glider, and torpedo-like AUV. Due to a balanced performance of torpedo-like AUV as reported in literature[27], in this brief we are specifically concerned with the formation control of this type of AUV. For more details on performance comparison among three types of AUVs, the readers of interest are referred to the paper[27]. In what follows, for convenience, the term AUV is referred to as the torpedo-like AUV.

In fact, it is not easy to develop high-performance formation controllers for a group of AUVs[5–7]. Major technical problems may arise from the following several aspects. First, it is notorious that the dynamics of AUVs are highly nonlinear with 6 degrees of freedom (DOF), but not fully actuated; that is, the control actions are not directly applied in the sway and heave motion directions. Besides, subject to the severe effects of hydrodynamics, a set of hydro-related plant parameters in AUV dynamic model are always time-varying, and furthermore, the ocean waves and currents have significant impacts on the dynamics of AUVs as well. Those mentioned factors even make the motion control of a single AUV fairly challenging[28]. More importantly, to control the formation of AUVs fleet, it is imperative to establish a communication network to exchange information between the vehicles. Nevertheless, due to the underwater environments, radio frequency and optical based communication technologies are usually inefficient, when the communication zone becomes broad, which is the case in AUVs formation in order to attain maximum coverage. In such a case, acoustic-based technology is regarded as the most suitable way to provide communication support for AUVs formation applications[27,29,30]. As a result, the communication constraints induced, e.g., low data rate, high propagation delays, path loss, noises, Doppler effect, etc., cannot be neglected. There have been a variety of surveys discussing and summarizing the formation control techniques available for multi-agent systems, especially including integrator-modeled systems[11,31], unmanned ground vehicles (UGVs)[31–33], unmanned aerial vehicles (UAVs)[34–37], unmanned spacecraft[38,39]. It is clear that due to the distinct applications as well as the characteristics, the technical challenges encountered in these systems may have an apparent difference from the autonomous underwater vehicles. For example, as a stark contrast, the communication channels for formation control of UGVs, UAVs and spacecraft are much better than those of AUVs, since for the former, the base stations and satellites can offer a stable high data rate and lower transmit errors. Consequently, the communication constraints may not be considered critical or pressing for such systems in order to achieve a high-precision formation performance. In addition, most of the existing available surveys on AUVs formation are mainly concerned with the motion control techniques and formation coordination strategies[29,40–43], but with limited focuses on communication problems which are, as mentioned, vital for the AUVs to reach a robust formation performance in underwater environments.

In light of the aforementioned observations, we believe that it is timely and helpful to present a brief overview of recent advances in AUVs formation control techniques that includes, particularly, a sufficient survey in handling underwater communication constraints. Notice also that due to the vast amount of the literature, it would be intractable to extensively review the existing results. We are thus concentrating on the major technical challenges and practical issues that significantly affect the formation control quality. In particular, we discuss the influence arising from the underwater communication constraints, which may deteriorate the formation performance in practice, but not well-studied in previous surveys. The contributions of this paper are listed: (1) Several popular formation coordination structures and approaches used for AUVs formation are presented, and their advantages and drawbacks with respect to the implementation, analysis, robustness and flexibility are discussed in-depth; (2) The difficulties in the development of a practical AUVs formation system are classified and analyzed in terms of the characteristic of AUVs, adverse underwater conditions, and communication constraints. Based on that, a comprehensive literature review of recent advances to handle these challenges is conducted; (3) According to the results of the survey, a summary is made, and several promising research directions are pointed out, which may be beneficial to promote the development of this field.

The rest of the paper is arranged in the following. Section 2 provides some preliminaries on AUVs formation control systems. Several widely used formation coordinating structures and approaches are presented in Section 3. The technical problems and practical considerations in applications are clarified, and the existing research solutions are reviewed extensively in Section 4. Section 5 gives a summary and points out some valuable future works. Section 6 concludes the paper.

2. PRELIMINARY

In this section, some preliminaries are presented. Particularly, graph theory is introduced, which serves as a useful tool to represent and analyze the interaction of AUV networks. Then, the mathematical model of the AUVs is given to describe the motion of each autonomous underwater vehicle.

2.1. Basic knowledge on graph theory

To analyze the properties of a AUVs formation system, graph theory can be used as a useful tool. This subsection aims to introduce some fundamental concepts in graph theory. The graph, denoted by a triple $G = \{V, E, A\}$, can be used to represent the communication topology among a AUVs fleet, including the vertex set $V = \{v_1, v_2, \dots, v_N\}$, edge set $E \subseteq V \times V$ and weighted adjacency matrix $A = \left[a_{ij}\right] \in \mathbb{R}^{\mathrm{N}\times\mathrm{N}}$. In particular, the element v_i in the vertex set V, termed a node, represents the AUV i in the group, where i belongs to an accountable index set $\Gamma = \{1, \dots, N\}$. The element $\left(v_i, v_j\right)$ in the edge set E describes the interaction between AUVs i and j, and associated with weights $a_{ij} = a_{ji} > 0$, which are the entries of adjacency matrix A. In such a case, we call AUV j a neighbor of AUV i, and all the neighbors of AUV i can be described by the set $N_i = \left\{j | \left(v_i, v_j\right) \in E\right\}$. If there is no connection between AUVs i and j, then let $a_{ij} = a_{ji} = 0$ and $\left(v_i, v_j\right)$ is not the element of E. We may further define $a_{ii} = 0$ for all $i \in \Gamma$, and out-degree of the node i as $d_i = \sum_{j\in N_i} a_{ij}$, after which the degree matrix and the Laplacian matrix of the graph G can then be defined as $D = \mathrm{diag}\left\{d_1, \dots, d_N\right\} \in \mathbb{R}^{\mathrm{N}\times\mathrm{N}}$ and $L = D - A$, respectively.

In addition, a path in graph is defined by a sequence that contains a set of successive adjacent nodes, starting from the initial node and ending at the final node. If there exists at least one path between any two nodes in a graph G, then, say, graph G is connected. Furthermore, in order to make the AUVs fleet move along with a desired path as a whole, a reference trajectory must be defined ahead of time. Thus, the availability of the information of reference trajectory for i-th AUV is characterized by a parameter b_i; that is, if AUV i is permitted to access this information, then $b_i > 0$; otherwise, $b_i = 0$, and define $B = \mathrm{diag}\left\{b_1, \dots, b_N\right\}$. Based on that, we may have the following important lemma, which is useful to help analyze the stability of AUVs formation systems based on the graph theory.

Lemma 1 *For the considered AUVs formation control network, described by graph G, if G is connected and there is at least one AUV able to access the information of the reference trajectory, i.e., the elements of B are not all equal to zero, then the matrix L + B is positive definite.*

2.2. Mathematical models of AUVs

The kinematics of each AUV is described as[28]

$$
\begin{aligned}
\dot{x}_i &= \cos\theta_i \cos\psi_i u_i - \sin\psi_i v_i + \sin\theta_i \cos\psi_i w_i,\\
\dot{y}_i &= \cos\theta_i \sin\psi_i u_i + \cos\psi_i v_i + \sin\theta_i \sin\psi_i w_i,\\
\dot{z}_i &= -\sin\theta_i u_i + \cos\theta_i w_i,\\
\dot{\theta}_i &= q_i,\\
\dot{\psi}_i &= \frac{1}{\cos\theta_i} r_i,
\end{aligned}
\tag{1}
$$

where (x_i, y_i, z_i) and (θ_i, ψ_i) are the position and orientation of the i-th vehicle ($i \in \Gamma$), respectively, expressed in the earth-fixed frame $E^{\mathrm{I}} = \left\{e_o^{\mathrm{I}}, e_x^{\mathrm{I}}, e_y^{\mathrm{I}}, e_z^{\mathrm{I}}\right\}$, and (u_i, v_i, w_i) and (q_i, r_i) are the linear and angular velocities

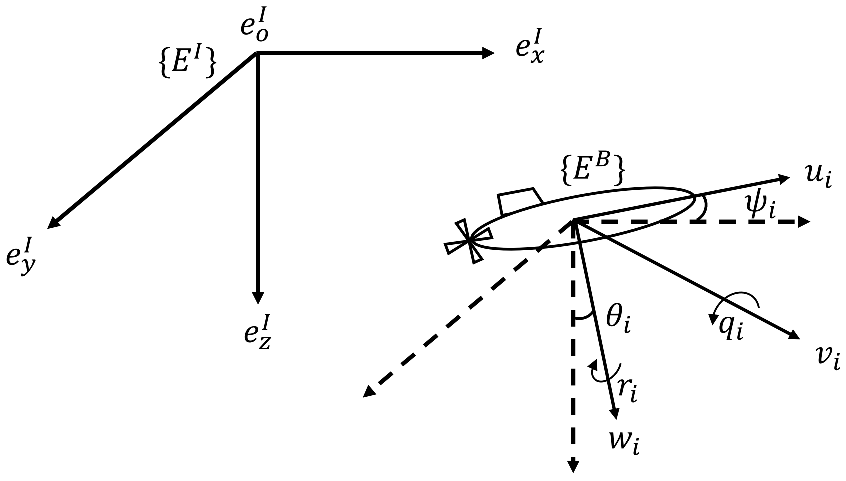


Figure 1. Earth-fixed frame and body-fixed frame systems of i-th AUV

of the i-th vehicle, respectively, which is expressed in the body-fixed frame $E^{\mathrm{B}} = \left\{e^{\mathrm{B}}_{o,i}, e^{\mathrm{B}}_{x,i}, e^{\mathrm{B}}_{y,i}, e^{\mathrm{B}}_{z,i}\right\}$, as shown in Figure 1.

The dynamics of the i-th vehicle is modeled by

$$\begin{aligned}
m_{i1}\dot{u}_i &= m_{i2}v_i r_i - m_{i3}w_i q_i - \beta_{ui}u_i + \tau_{i1} + d_{i1},\\
m_{i2}\dot{v}_i &= -m_{i1}u_i r_i - \beta_{vi}v_i + d_{i2},\\
m_{i3}\dot{w}_i &= m_{i1}u_i q_i - \beta_{wi}w_i + d_{i3},\\
m_{i4}\dot{q}_i &= (m_{i3} - m_{i1})\, u_i w_i - \beta_{qi}q_i - \beta_{bi}\sin\theta_i + \tau_{i2} + d_{i4},\\
m_{i5}\dot{r}_i &= (m_{i1} - m_{i2})\, u_i v_i - \beta_{ri}r_i + \tau_{i3} + d_{i5},
\end{aligned} \tag{2}$$

where $m_{i1} = m_i - \beta_{\dot{u}i}$, $m_{i2} = m_i - \beta_{\dot{v}i}$, $m_{i3} = m_i - \beta_{\dot{w}i}$, $m_{i4} = I_{yi} - \beta_{\dot{q}i}$ and $m_{i5} = I_{zi} - \beta_{\dot{r}i}$; m_i is the mass of the i-th vehicle; I_{yi} and I_{zi} are the moments of inertia around the axes of $e^{\mathrm{B}}_{y,i}$ and $e^{\mathrm{B}}_{z,i}$, respectively; $\beta_{(\cdot)}$ is set of hydrodynamic related terms associated with the i-th vehicle, $\tau_i = [\tau_{i1}, \tau_{i2}, \tau_{i3}]^{\mathrm{T}} \in \mathbb{R}^3$ is the control input, and $d_i = [d_{i1}, d_{i2}, d_{i3}, d_{i4}, d_{i5}]^{\mathrm{T}} \in \mathbb{R}^5$ is the disturbance acting on the i-th vehicle.

Remark 1 *It is worthwhile noting that the complete motion of the equation of an AUV is of 6 DOF, but, as we surveyed, almost all of the literature studying 3-dimensional (3D) formation applications employ the AUV model, as presented in Equation (1) and Equation (2), with 5 DOF. This relies on the fact that for an AUV formation fleet, the rotational motion around $e^{\mathrm{B}}_{x,i}$ axis, i.e., roll motion, sometimes is not required in many practical AUV maneuvering, and hence the equation governing the roll motion is intentionally omitted, which does not cause loss of the practicality. Indeed, it is easy to check that the roll motion is passively bounded*[28]*. Particularly, its impacts on the other DOF can be treated as unmodeled dynamics and handled effectively by some disturbance rejection techniques.*

Remark 2 *It is clear that the dynamics of AUVs are highly nonlinear and underactuated as in Equation (2); in other words, the sway and heave velocities, i.e., v_i and w_i, are not fully actuated and there are no actual control actions allowed to be applied. In the sequel, this underactuation feature will be demonstrated as a major challenge to designing high-performance formation controllers for AUVs. In addition, besides the environmental disturbances described by d_i, as observed in Equation (2), there are many system parameters subject to the perturbation due to the effect of hydrodynamics, which is characterized by the time-varying parameters $\beta_{(\cdot)}$. Such uncertainties in the dynamic model of AUVs give rise to another significant difficulty for the formation control design. Furthermore, the angle of θ_i is constrained and not allowed to take values at $\pm\pi/2$ in order to avoid the singularities, which* *should be guaranteed by the formation controller. Therefore, from the control point of view, addressing such a system continues to be challenging.*

2.3. Problem statement

This subsection aims to formulate the considered formation control problem of a fleet of AUVs. As mentioned earlier, one of the requirements of formation control is to drive the AUVs to form a prescribed static or time-varying geometric shape and maintain it. Typically, the expected geometric pattern can be determined by the appropriate assignments of the relative positions between vehicles, denoting as $\Delta_{ij} = \left[\delta_{x,ij}, \delta_{y,ij}, \delta_{z,ij}, \delta_{\theta,ij}, \delta_{\psi,ij}\right]^{\mathrm{T}}$ for AUV i $(i \in \Gamma)$ and its neighboring nodes $j \in N_i$. To this end, the formation control is cast to the problem of controlling relative positions and orientations of vehicles with respect to their neighbors so that Δ_{ij} can be achieved as time tends to infinity, and particularly these desired relative poses could even be set to be time-varying as necessary. Another practical requirement is that it is most desirable for an AUVs fleet in applications to track a reference trajectory as a group, in which case not only the positions, but also the velocities of the vehicles in the fleet are needed to be in consensus; that is, $(\dot{x}_i, \dot{y}_i, \dot{z}_i)$ converges to a common reference speed $(\dot{x}^d, \dot{y}^d, \dot{z}^d)$, as $t \longrightarrow +\infty$. Such an objective is also referred to as the problem of formation tracking control in the literature.

In short, the formation control objective can be summarized as designing controllers for AUVs such that a set of desired relative positions and orientations, i.e., Δ_{ij} $(i, j \in \Gamma)$, can be achieved and maintained while the AUVs fleet tracks a common reference trajectory, moving in the same speed together.

3. AUV FORMATION CONTROL PROTOCOLS

In order to achieve the preceding formation objective, an appropriate formation control protocol is a must to coordinate the motions of AUVs efficiently. In general, multi-agent coordinating strategies can be roughly categorized into two forms depending on the information used, i.e., centralized coordination and decentralized coordination[11,31]. In centralized form, the control commands or control actions of each AUV are planned through a central control procedure which is allowed access to the global information of the whole formation system. On the contrary, in a decentralized structure, there is no such a controller planning actions for every AUV in the group, and instead, AUVs make their own decisions based on their local information accessed. In fact, on the basis of an extensive review, there are few research efforts made to design centralized coordination protocols for AUVs formation applications, which owes to the fact that each AUV employed has sufficient capabilities to sense the surroundings, plan its control actions accordingly and communicate with neighbors.

Based on this observation, we are mainly concerned with the approaches adopting a decentralized coordinating control structure, which includes leader-following structure, virtual structure, behavior-based approach, artificial potential field approach, and other common structures.

3.1. Leader-following structure

Leader-following structure is one of the most popular schemes used for the formation control of multiple agent systems because of its straightforward descriptions[11]. In such a scheme, one or several agents are selected as the leaders, and the rest of the agents are grouped into followers, as depicted in Figure 2. The desired reference signal is merely known to the leaders, and in conventional leader-following structure, the goal of the leaders is simply to track this prescribed reference and there is no explicit interaction between their following agents. The sole goal of the remainders is aimed to keep the desired relative pose (i.e., position and heading) with respect to their leading agents. As such, the formation control objective, as stated in section 2.3. can be achieved if each vehicle's goal is reached. The major advantages of such a method are that it is easy to be implemented and fairly flexible to add or remove vehicles in the fleet; besides, since there are no direct interactions between neighboring vehicles, the stability of the whole formation system can be easily analyzed based on the graph

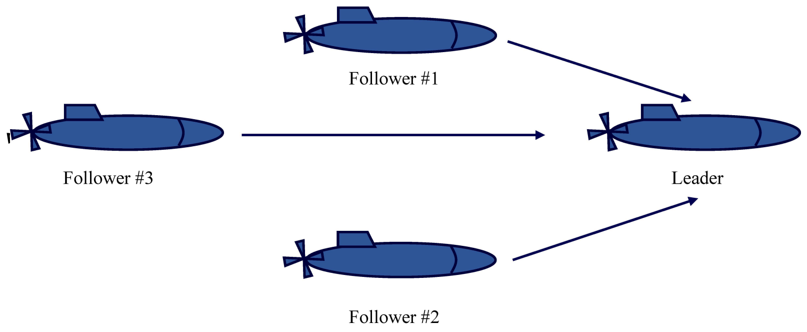


Figure 2. A typical topology of leader-following structure.

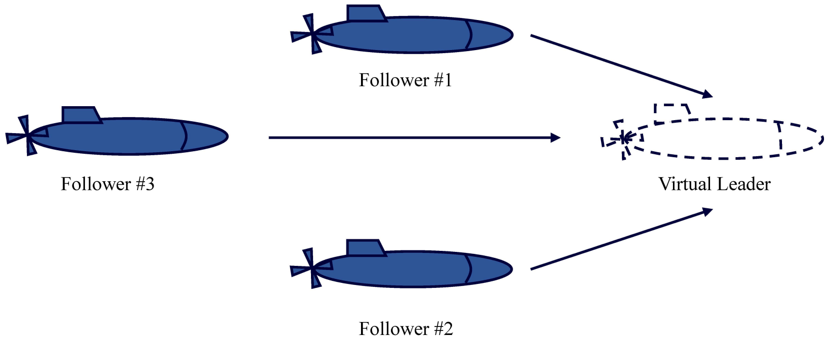


Figure 3. A typical topology of virtual leader structure.

theory, as presented in section 2.1. Because of those nice features, there is a great amount of literature on AUV formation control adopting the leader-following structure[44–48].

Such an approach, however, suffers from a major defect that the performance of the overall formation system depends highly on the behavior of the leaders and the quality of the communication. In other words, once the lead agents or communication network fail to work as usual due to unpredictable faults, which is often the case in underwater environments, the entire formation system may be disabled. To overcome this issue and improve the robustness of the leader-following approach, a virtual leader based method is proposed, in which there are no physical vehicles employed to lead the group and, therefore, the above-mentioned issue can be appropriately addressed[49–52]. A typical virtual leader based formation structure can be referred to the Figure 3. Another critical consideration regarding this type of structure is that it is always assumed that every vehicle in the group is permitted to obtain the trajectory information of the virtual leaders, which is a strong assumption and may not be fulfilled in many realistic applications.

3.2. Virtual structure approaches

Similar to the virtual leader approach, virtual structure coordination is another common method used to coordinate the multi-agent formation, which was first reported in[53,54] to address the cooperative control of multiple mobile robots. In this method, a set of virtual points are defined corresponding to each vehicle, which is determined by the desired formation configuration as well as the trajectory to be tracked. Since each vehicle is assigned its own reference point, the formation tracking problem is then converted into the tracking control problem associated, the goal of which is to drive the vehicles to minimize the errors between their actual positions and desired ones. A typical realization of such an approach is illustrated in Figure 4. Due to the fact that such a method is also straightforward and simple to analyze and realize, there have been many results reported to date based on this method to achieve the formation control requirements[54–59].

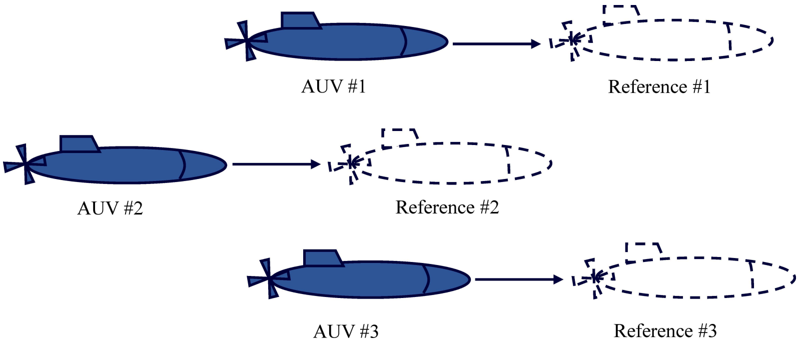


Figure 4. A typical topology of virtual structure approach.

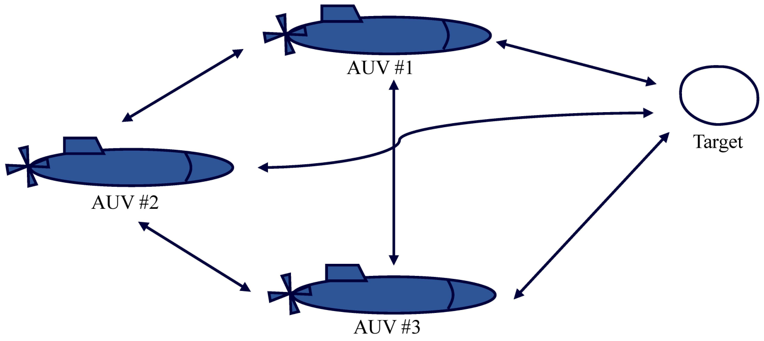


Figure 5. A typical topology of behavior-based approach.

The main drawbacks of the virtual structure approach may lie in several aspects: (1) Like the virtual leader approach, this method relies closely on the desired reference trajectories, which appears to be unrealistic in many practical scenarios; (2) It is not easy to expand the AUV formation, since the desired virtual reference points are designed in advance based on the prescribed formation pattern; (3) Due to the lack of information exchanges between the neighboring vehicles, there is no cooperation occurred in the formation system, which degrades the coordination performance.

3.3. Behavior-based approaches

Different from the above two methods, as shown in Figure 5, there exists an explicit mutual communication in formation systems synthesized using the behavior-based coordination approach. Instead of directly prescribing *a priori* reference trajectories, in behavior-based approach, each vehicle in the group makes its own decisions based on the local information (e.g., its own states, surroundings, and neighbors' states) and the goals predefined. The goals usually include target reaching, collision or obstacle avoidance, distance maintaining, etc. Particularly, the overall control actions of the vehicles are then generated by a weighted combination of achieving these different goals. Due to the multi-objective and distributed features, behavior-based approach has attracted extensive attention over the past few decades in the research areas of multi-agent cooperation and coordination[60–64].

Although the behavior-based scheme is turned out to be able to achieve multiple objectives and is merely dependent on the limited local information to calculate control activities, it is hard to analyze the stability properties of the overall formation system based on such a method when more vehicles and behaviors are involved. This restricts its practical applications greatly.

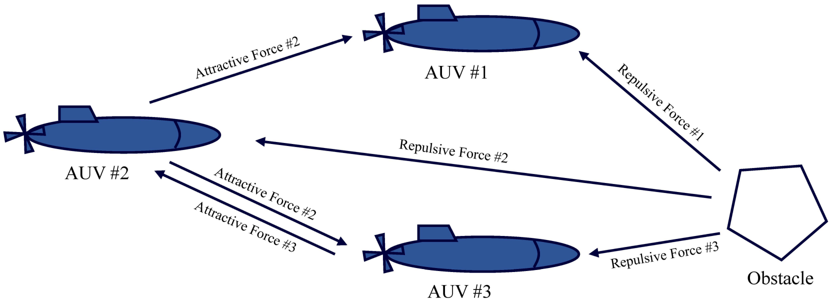


Figure 6. A typical topology of APF-based approach.

3.4. Artificial potential field approaches

Artificial potential field (APF) approach is first invented by Khatib[65] in order to design an algorithm to generate an obstacle-free route for manipulator and mobile robot path planning. The main feature of the method is that a series of artificial potential functions are defined intentionally with the purpose of reaching the target and meanwhile avoiding obstacles. Analogous to the potential energy in physics, the APF functions defined can generate corresponding potential forces as well. Typically, there are two types of APF functions involved: One aims to generate attractive potential forces to bring the vehicles to the targets, and another attempts to yield the repulsive potential forces to make the vehicles keep away from the obstacles, which is shown in Figure 6. As a consequence, under both the attractive and repulsive forces, the prescribed objective can be attained. Inspired by such a formulation of clear physical significance, APF approach is also introduced in various multi-agent systems to help organize the cooperation and coordination[66–72].

Similar to the behavior-based approach, it is relatively easy for the APF approach to synthesize distributed controllers that achieve multiple goals depending only on the local information. However, one major drawback is that it has the chance to trap into points at which the resulting net force applied on vehicles is zero, which is also known as the issue of ”local minima”. Likewise, the stability analysis of APF based multi-agent systems is not easy as well, compared to the leader-follower structure and virtual structure approaches, when the scale of the group grows larger.

3.5. Other approaches

By a comprehensive literature review, there are some other commonly used approaches to achieving the multi-agent cooperation and coordination, while these kinds of schemes can be, in some sense, regarded as variants of those already presented. For example, the so-called formation reference point (FRP) method is actually a type of virtual leader method, in which a reference point is defined and parameterized with desired velocity profile, and then the control objective of vehicles in the group is simply to maintain a specific distance and bearing with respect to the reference point[73,74]. Like the conventional virtual leader method, there are no explicit interactions took place between neighboring vehicles.

In the field of multi-agent coordination, there exist a fundamental problem, termed consensus problem, which characterizes how the inter-agent cooperation can be emergent by merely using the local information (i.e., interacting with neighbors)[75]. Specifically speaking, it is possible that the state of the entire multi-agent system can be ultimately in consensus; that is, each agent's state converges to the same equilibrium point under a proper local control law that is designed only based on the neighboring information. Moreover, such a problem can be well formulated and tackled by the graph theory, as mentioned in section 2.1, including the basic issues such as the existence of solutions, stability and robustness properties of multi-agent systems. For this reason, a vital amount of related research results on networked multi-agent systems coordination based on consensus approach are reported both in theoretical and practical dimensions[31,76–81]. Such a problem is then extended to the case of formation control design, where a virtual leader is introduced to guide the vehicle group

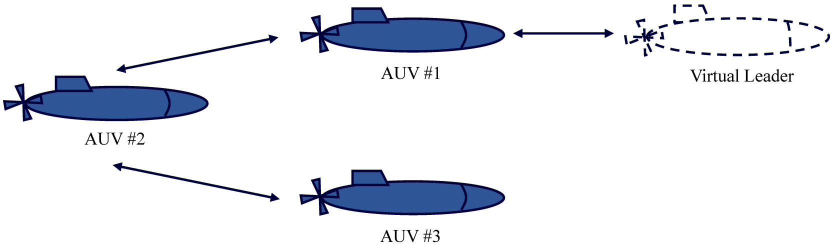


Figure 7. A typical topology of consensus-based approach.

to move along with a reference trajectory together, and meanwhile, the formation is formed by the exchange of information between neighbors[82–84]. A typical communication topology of consensus-based formation control is illustrated in Figure 7.

It should be noted that in stark contrast with the conventional virtual leader structure, there exists local information flow in the consensus based approach, and more importantly, it is not necessary that all members in the group have access to the information of reference trajectory[85–87]. It is, in effect, the sole condition that the graph associated has a spanning tree[11]. Those mentioned features seemingly make the consensus based method more practical and beneficial for real-word applications among various formation control methods. While such a method has attracted much attention in the studies of groups of unmanned ground vehicles[88–92], unmanned aerial vehicles[93–98], and spacecraft[99–102], there is still a lack of sufficient research efforts to apply consensus based approach to the multi-AUV formation where more practical challenges associated should be addressed further, most of which will be discussed in what follows.

4. PRACTICAL ISSUES AND CHALLENGES IN AUV FORMATION CONTROL

This section discusses the major technical challenges and several practical issues encountered in the synthesis of AUVs formation control systems in terms of motion control of AUVs, addressing the adverse underwater conditions and communication constraints. A great variety of existing available results to tackle these difficulties are reviewed comprehensively, and the features of each type of method are pointed out.

4.1. Nonlinear constrained dynamics of AUVs

As mentioned earlier, different from the general multi-agent systems[11] where a point-mass model is typically used to describe the motion of agents, the governing equations of AUVs are much more complicated as explained in Remark2, including nonlinearity, underactuation and system constraints. As a result, such systems may suffer from some extra design complexities in respect of motion control. To deal with the nonlinearity, employing an approximated model is a likely choice by linearizing the nonlinear model at a specific operating point, and then linear control theory can be used to design the motion controller for AUVs. Based on this idea, the authors[103] proposed a nonlinear gain scheduling controller for heading control of AUVs in the horizontal plane. In this study, a finite number of linear static feedback controllers were derived at distinct speed conditions, after which the parameters of the controller were interpolated upon speed. The resulting scheduled controller was designed using the D-methodology[104] to guarantee the stability of the overall closed-loop system. To obtain an optimal control performance with respect to a quadratic-type objective, a linear-quadratic regulator (LQR) algorithm was implemented[105], and an optimal state feedback gain was figured out to regulate the AUVs' depth as well as stabilize the roll and pitch angles. In order to account for the approximating errors and the time-varying environmental conditions, a robust H_∞ based depth control was proposed[106], in which the aim of the controller is to minimize the cost function under the maximum effects of the parametric perturbations, and the resulting robust performance is verified by simulation.

Another class of optimization-based control methods, termed model predictive control (MPC), is inherently effective in handling internal uncertainties. Due to the preceding horizon formulation, an online optimization mechanism is enabled in MPC using real-time state feedback against the potential modeling errors. Furthermore, owing to a shortened horizon employed in optimization, such a method is also allowed to handle the state and input constraints, which is a very beneficial and useful feature in designing practical controllers as nearly all of the physical systems are subject to actuation saturation. By means of the MPC policy, the authors[107] studied three-dimension (3D) underwater tracking control of a fully-actuated AUV with the practical constraints in both state and input, and the simulation results verified the satisfactory performance in the presence of various disturbances. In underwater conditions, the measurement noises may be another non-negligible factor affecting the control performance. To this end, linear quadratic Gaussian (LQG) control, integrated with the Kalman filtering (KF) technique, serves as an efficient optimal control strategy to ensure that the controlled system operates at the optimum status in the sense of variance minimization[108]. Recently, a model predictive tracking controller was combined with an extended state based Kalman filter for a constrained remotely operated underwater vehicle in order to achieve a high-precision tracking performance, while the system nominal model is uncertain and subject to both sensor noises and external disturbances[109].

However, most of the aforementioned approaches are designed based on a linearized model at some specific operating points, which necessitate assumptions that the AUVs move at a relatively low speed and the impacts of disturbances, caused by the ocean waves and currents, are much limited such that the locally stable results can be obtained. Such a hypothesis may be quite conservative and, in effect, not always hold in practical situations. In view of that, nonlinear control techniques have been the research focal point over past decades to help develop high-performance motion controllers for AUVs with global stability guarantee. The researchers[110] proposed a nonlinear tracking controller for a remotely controlled AUV based on the feedback linearization technique, under which the resulting system becomes a decoupled linear system, and furthermore, an optimal error correcting term is added in the feedback loop, based on LQR approach, to compensate the uncertainty. To derive adaptive formation tracking controllers for a group of underactuated AUVs in a horizontal plane, the backstepping design procedure and neural network technique are used[46]. The resulting nonlinear formation controllers developed for each AUV are based on a virtual leader structure without knowledge of the leader's velocity and dynamics, and the overall AUVs formation is proved to be uniformly ultimately bounded stable using the Lyapunov stability theory. Due to the obvious benefits of MPC approach in handling systems constraints, a nonlinear MPC controller was presented for the tracking control of an AUV in 2-dimension (2D) case[111,112], where an auxiliary nonlinear control law is proposed using the backstepping-based Lyapunov synthesis, and as a result, the linearization procedure can be avoided for standard MPC design, which improves the region of attraction of resulting system greatly. Furthermore, the resulting Lyapunov-based MPC is guaranteed with properties of iterative feasibility and stability. Following the same ideas, the authors[113] developed a leader-follower based receding horizon kinematic controller for formation control of a constrained underactuated AUV. Taking into consideration the uncertainties in modeling, the authors combined the extended state observer (ESO) with nonlinear MPC to deal with the formation tracking of AUVs[114].

Another class of important methods to address complex systems is intelligent control, including fuzzy logic control, neural network control, and data-driven control (sometimes in the literature also called machine learning-based control), etc. For instance, the authors[115] developed a nonlinear fuzzy logic based proportional-integral-derivative (PID) controller to regulate an AUV's heading and depth, and in particular, Mamdani fuzzy rules are used to tune the control gains of PID adaptively to address the nonlinearity and uncertainties in AUV modeling. As well, to address a similar issue, 3D path following of an underactuated AUV is investigated using fuzzy logic based backstepping controller[116]. It is worth noting that AUVs path following problem is slightly different from the tracking problem, while they are both expected to follow a desired route. The former does not impose a strict temporal constraint on the reference trajectory. Thus, an extra degree of freedom in the speed of the trajectory can be exploited to design controllers, which could be practical in some applications[6].

To handle the nonlinear and unknown dynamics of AUVs, a data-driven model-free predictive approach[117] was proposed for the accurate trajectory tracking control, in which the authors suppose that both parameters and order of the plant are unavailable to control design. To address a similar problem, an adaptive line of sight (LOS) guided reinforcement learning approach based on the long short-term memory (LSTM) neural network model was devised[118]. Considering the unknown nonlinearity in AUV dynamic model, underactuation and input limitation, the author[119] designed a neural network based robust formation controller using virtual structure for a group of underactuated AUVs with actuation saturation, and furthermore, rigorous stability analysis was provided.

4.2. Adverse underwater environments

In addition to the nonlinear, uncertain and underactuated characteristics of the AUVs dynamics as discussed above, the complicated underwater conditions (e.g., marine disturbances, hydrodynamic effects, unpredictable static or dynamic obstacles) may also pose a great difficulty in AUVs formation design. In the preceding discussions, we have already mentioned numerous methods to address the uncertainties in AUV dynamic model, most of which adopt an adaptive idea, e.g., adaptive control methods, data-driven methods, neural network-based methods, etc., to estimate parameters (either control gains or model-related parameters) in a real-time manner based on the measured data. This class of methods, although superior, is basically suitable for the case when the structure of the uncertainties is known, but the associated parameters are required to be estimated. On the other hand, due to the integration with online parameter estimation, most existing adaptive approaches result in a nonlinear time-varying system whose robustness properties regarding the modeling errors are hard to be guaranteed. Nonetheless, as for the marine disturbances induced by the ocean waves, currents and winds, as well as the effects from the hydrodynamics, their interaction patterns with the AUVs are complicated and quite difficult to understand. In view of that, nonlinear robust control techniques, e.g., sliding mode control (SMC), have received much attention in various mechatronic control systems, for which a static switching-like feedback control law is used and the corresponding control gains are calculated to ensure that all types of disturbances, so long as within a given bound, can be handled[120–122].

Based on this paradigm, there is a variety of literature employing SMC schemes to study the robust tracking control problem of AUVs subject to both unknown disturbances and modeling uncertainties[123–126]. In particular, considering the potential time delays between surface ships and vehicles, which happens quite often in practical situations, the authors proposed a discrete-time quasi-SMC method for AUVs depth control to guarantee the stability and robustness even in the presence of large sampling intervals, and concurrently, the chattering was addressed using the so-called equivalent control region[127]. Besides the robustness, in order to meet a fast converging requirement near the equilibrium point, terminal sliding mode control (TSMC) techniques are used by researchers to achieve the finite time control[128–130]. Due to the possibility of singularity appearing in TSMC schemes, some non-singular TSMC approaches were investigated to avoid this issue[131–133]. While SMC based controllers exhibit a stunning robust performance in handling unknown environmental disturbances as well as inherent modeling uncertainties, a major drawback is that the chattering issue occurring in control activities should be carefully addressed, as it may inevitably deteriorate the performance in practical cases. Towards this end, higher-order SMC techniques were proposed to attenuate the high-frequency chattering to make SMC schemes more practical, and such methods were also applied to the control problems of AUVs tracking and formation[134–138]. To completely eliminate the chattering while obtaining good robustness, the authors developed a distributed neuro-dynamics-based sliding mode controller for the consensus formation tracking control of a group of AUVs[139]. In addition to the static robust control schemes that are considered more conservative, there are numerous disturbance observer-based controllers used for AUVs motion and formation control, which behave more actively in addressing the disturbances and hydrodynamic coefficients[140–145]. As well, observer-based schemes can also be used as an effective tool to tackle the output feedback control where the velocity sensing is unavailable, which is fairly beneficial in terms of reducing cost and meanwhile improving the performance of AUVs formation[146–149]. Furthermore, observer or estimator based techniques are also of critical importance and act as useful tools in designing systems of diagnosis and accommodation for sensor errors and faults. It is well known that failure to detect and isolate the errors and faults may lead to inaccurate control performance and even loss of the AUVs. To overcome it, the authors[150,151] employed certain observers and filters on the basis of vehicles' models to identify the errors in sensors and provide an estimate correspondingly, and thus a high-quality motion control objective can still be maintained.

Except for the handling of unknown disturbances and hydrodynamic uncertainties, avoiding obstacles and inter-vehicle collisions is another realistic aspect in synthesis of formation systems due to the poor knowledge of the obstacle distribution. To prevent unmanned aerial vehicles (UAVs) formation from any collisions, a collision avoidance mechanism was incorporated into the UAVs formation control strategy. In this approach, an additional velocity term is involved based on the mechanical impedance principle to avoid potential collisions, once the obstacles detected are within a certain range[152]. A new dual-mode strategy is designed to achieve cooperative UAV formation flying. In an obstacle-free environment, safe mode is activated to achieve global optimization. When faced with obstacles, danger mode enables a modified Grossberg neural network to plan an obstacle-free route, and a model predictive controller is also developed to achieve the route planned[153]. An obstacle avoidance strategy was proposed for UAV formation control design based on the artificial potential field (APF). An attractive potential field is designed for a leader to track the moving target, and the rest of the AUVs aims to follow the leader using the attractive potential force, while repulsive potential forces are also defined for UAVs to avoid obstacles[154]. Employing the same idea of APF, the authors proposed an obstacle-free formation controller for multi-agent systems based on virtual structure. To achieve the path planned by APF, a backstepping controller was developed based on the neural network and finite-time control technique[155]. Besides the APF-based obstacle avoiding strategies, as aforementioned, MPC-type controllers are capable of handling constraints effectively, which can be utilized to design collision-free formation controllers. This can be achieved by either extending the objective function or adding extra avoidance constraints into the receding horizon optimization procedure[156–159]. Other available approaches, such as ant colony algorithms[160,161], particle swarm optimization[162], machine learning-based algorithms[163], etc., were also reported to optimize an obstacle-free path, yet such intelligent optimization-based algorithms quite often necessitate a great computational effort and take a relatively long time, which greatly restricts their applications in AUVs formation where computation resources are limited and rapid response is needed to handle the variable surroundings.

4.3. Communication constraints

Different from the multiple land robots, surface vehicles and unmanned aerial vehicles, the signals of the global position system (GPS) or base stations cannot be received by AUVs in underwater conditions, and therefore the localization and communication in AUVs fleet may become an unavoidable issue. It has been turned out that the traditional acoustic technologies still serve as the most efficient way to provide the wireless connecting services[27], in which case the communication capability, thereby, has major impact on the performance of AUVs formation. The effects of the communication constraints mainly result from the following several aspects.

- Propagation delay: The speed of acoustic waves is nearly 1.5 kilometers per second, which is far slower than the speed of electromagnetic waves in air, and thus the effects of propagation delays cannot be neglected. Moreover, since the speed of the acoustic propagation relies highly on environmental factors (e.g., pressure, temperature, salinity, disturbances), the calculation for propagation delays may be rather complicated and often time-varying.
- Path loss: As the acoustic waves propagate, their energy spreads and attenuates in the medium. This process is called path loss, especially the higher the frequency, the easier the path loss. More path loss means a shorter communication range.
- Limited bandwidth: For a specific transition channel, its bandwidth is always limited and related to the range of communication. Therefore, it is necessary to allocate the interactive data reasonably (e.g., sensor

data, control data, localization data) in order to achieve an effective formation control.

- Multipath: During propagation, the acoustic signals could be reflected possibly by the sea surface, seabed and obstacles, which causes the multipath. Signals in multipath will result in distortion, consequently reducing the measurement or transmission accuracy.
- Doppler effect: Since the AUVs formation is always in motion, the frequency of the wave being transmitted by a moving base may be varied on its receiving side, which is called the Doppler effect.

There are a number of research efforts made to overcome the above communication constraints so that a better formation performance can be achieved. By assuming that the delays of propagation are bounded and the bounds on delays are less than the sampling period of AUVs, it was shown [164] that the leader-following based formation system can tolerate the delays and still achieve the formation stabilization, although it takes a longer time to stabilize than the case without delays. To provide a lower bound on control performance of an inspection ROV with sensor time delays, the authors employed an input-output controllability analysis to show that at which amount of delays the ROV system can tolerate irrespective of the forms of designed controllers, while the results are obtained based on an approximated linear model of the ROV [165]. As well, the researchers demonstrated by experiments that there exists a dominant time delay in the underwater acoustic positioning system powered by the short baseline (SBL) with four transducers, and that the traditional Smith predictor failed to estimate due to the time-varying properties. To handle it, an online identification was proposed in their work to estimate the variation of the delay, thus minimizing its impacts as much as possible [166]. To be able to account for the package dropouts, the authors [167] treated it as an extra delay out of the total delays and a compensating law was designed in the H_∞ controller to ensure a robust formation performance against delays. To characterize the delays more precisely, researchers [168] represent the delays as a differentiable function and suppose that its time derivative is less than one, under which a leader-following controller is designed to reach the formation requirements. To overcome the bandwidth limitation, an observer-based formation controller is quite often used to reduce the amount of information exchange, and furthermore, a fault function is introduced to describe the effect of the multipath in communication [169]. Nonetheless, as indicated [27], due to the fact that the information will be compressed before transmission, reducing control or sensor data do not contribute a lot to the relaxation of the bandwidth limitation. The authors provided a leader-following algorithm to optimize the formation configuration in terms of coverage efficacy and communication power consumption utilizing a calculation model for the path loss [170]. To reduce the Doppler effect, a Doppler effect compensation system is designed using an efficient multi-rate sampling technique [171]. Similarly, using diffident resampling strategies, reduction of the Doppler effect for the moving platform can be achieved [172,173]. Due to the unpredictable ambient disturbances and long communication distance, it is possible for the AUVs fleet to encounter a period of communication disconnection, in which case dynamic network topologies can be used to describe it. To achieve a dynamic topology, the authors predefine a set of topologies and put a random mechanism on it, and the objective is to design a formation controller such that the formation system can be stabilized under a jumped network topology [174,175]. Instead of random switching, a distance-based dynamic topology is defined among the group [176], in which merely the vehicles within the communication range can create the connection.

5. SUMMARY AND FUTURE WORK

We first provided several most commonly used coordinating structures and approaches to solve the AUVs formation control problem, that is, leader-following structure, virtual structure, behave-based approach, artificial potential field approach, and other approaches. It is noted that all of these mentioned approaches adopt a decentralized control architecture, i.e., each vehicle makes its own decisions based on the information received. In leader-following structure, there are one or several agents defined as leaders in advance to guide the whole formation system to meet the control requirements. Due to the advantages (e.g., easy implementation and analysis, and good flexibility), such a method has attracted much attention in most of the AUVs formation syntheses. A major deficiency is that the entire formation performance relies strongly on the leaders' behavior, thus often resulting in poor robustness. Although a virtual leader takes the place of the physical leaders, it still needs an assumption that each vehicle in the group is able to access the information of the leaders, which is obviously not realistic in real applications. Similarly, virtual structure approach seeks to convert the coupled coordination problem into the single vehicle tracking problem. In such a way, the analysis of the behavior of the overall formation system becomes trivial, since the stability and robustness of the overall system are simply ensured by the properties of single tracking systems. While straightforward and simple to realize, formation scaling is not that flexible compared to the leader-following approach, as the virtual reference points are predefined in accordance with the specified formation shape and cannot be changed during the formation process. In behavior-based approach, an apparent difference from above-mentioned methods is that mutual communication between neighbors is introduced, whereby it is possible for the agents to fulfill the formation demands using only the local information rather than the leaders' information, which seems to be more beneficial. Another benefit is that multi-goal can be achieved in this framework, e.g., formation keeping, target seeking, obstacle avoidance, etc. However, the major restriction of this method is that it is hard to analyze the behavior of the overall formation systems, especially in the case that the number of agents becomes large and the goals of the agents are complex, and thus cannot claim any theoretical guarantees to support its practical applications. Likewise, artificial potential field approach can be used to achieve the multi-objectives by means of the local interactions based on the so-called artificial potential field function. The analysis of the evolution of resulting systems is not easy as well due to the fact that the introduced potential functions are most likely to be nonlinear. Recently, the consensus-based approach has received enough attention, since it is able to provide a good trade-off between theoretical analysis and coordination performance. While the studies regarding consensus in the context of general muli-agent systems are extensive, because of the difficulties and special characteristics of the AUVs formation systems as mentioned above in Section 4, the research work adopting a consensus-based formation design is very limited.

Next, we categorized the several technical challenges and practical considerations in AUVs formation, that is, nonlinear uncertain dynamics of AUVs, complex underwater conditions, unknown disturbances and faults, and communication constraints. It is quite necessary to address them suitably and efficiently in order to obtain a better formation performance. Based on that, we reviewed in-depth the recent advances and developments in handling these pressing challenges. While, as observed above, there are an increasing number of research results being made to partly tackle some of these issues, due to the interdisciplinary nature of AUVs formation control systems, it is still far from mature and needs further research efforts. Based on the brief literature survey, some promising research directions are pointed out as follows.

- Due to the fact that the dynamics of AUVs are inherently nonlinear and uncertain subject to the underactuation and numerous constraints, high-quality motion control continues to be challenging. While there are some good frameworks that have been proven to be able to address these issues concurrently by combining the nonlinear control techniques (e.g., feedback linearization and backstepping design procedure) with MPC strategy[111–113], the performance of the resulting systems is dependent of the proposed auxiliary controllers, which may somehow tighten the region of attraction, and meanwhile such methods are usually computationally demanding. Thus, it is interesting to explore efficient nonlinear MPC schemes or other optimization-based approaches that are free from the auxiliary control laws simultaneously with fewer computation demands.
- Most of the existing solutions proposed are based on either the kinematic model or a reduced 2D dynamic model in the horizontal or vertical plane, which lack practicality, and therefore it is necessary to propose methods suitable for the general 3D case with more dynamic maneuvering tasks. Besides the stability, the systems' robustness properties against disturbances, noise and faults should be further established in order to synthesize more practical and robust controllers, but it seems that many existing results ignore the analysis on this aspect.
- In order to simplify the stability and robustness analysis, a good many available results are obtained based

on the leader-following structure. As aforementioned, such a method, although easy to implement, is not realistic and short of cooperation as it is assumed that the leaders' information needs to be received. In view of that, the consensus-based formation tracking control deserves further investigation.

- It is observed that many advanced formation control protocols developed assume perfect communication. As indicated above, addressing communication constraints (e.g., time-varying delays) is one of the most important points, as it is ubiquitous to encounter bad communication channels due to underwater conditions. Therefore, we believe that incorporating delay estimation and compensation techniques into formation control or designing delay-tolerant formation systems is of great significance in practice.
- While there are some primary proposals that provide several delay compensating strategies[165–168], most of them may be lack practicality due to the employment of an unrealistic delay model. Thus, more practical delay models are required, and then can be used to compensate for the impacts of the propagation delays. As suggested[27], most of the communication constraints can be ultimately grouped into the effects of delays, while the delays usually can be much more complicated. Thus, developing an effective estimation strategy for delays contributes to better formation control performance.
- In addition to the delays, due to the unpredictable underwater situations, it is possible to encounter a communication disconnection with neighboring agents, in which case the traditional fixed topology-based formation coordination strategy may fail to work. For this reason, developments of variable topology-based formation control protocols are of more practicality and robustness. While there exist some potential techniques suitable for handling this issue, as reported in *Tomera*'s work[177]where hybrid switching control theory was used to attempt to construct safety controller for a training ship operating in many modes, we believe more research efforts are still needed and relevant on this topic to derive a realistic switching mechanism for formation applications with consideration of characteristics of AUVs and communication.

In summary, because of the interdisciplinary feature of the AUVs formation applications, including the fields of mechanical, electrical, control and communication, many challenges and practical issues, as mentioned above, have not been well resolved and studied. In order to synthesize a practical and robust formation control system, we believe that it is imperative to take into consideration a suitable formation coordination structure, a realistic dynamic model of AUVs, disturbances, fault-tolerance and communication constraints.

6. CONCLUSION

This paper reviews the current advances and developments in the area of AUVs formation control systems, especially from the control point of view. We start with presenting the preliminaries on graph theory and AUVs mathematical models, both of which are essential to the analysis and synthesis of AUVs formation systems. Based on that, the formation control problem is stated. Then, several commonly used formation control protocols are listed, and the pros and cons of each method are summarized. After that, the technical challenges and practical issues confronted in AUVs formation control design are identified, and the existing available results to overcome these challenges are comprehensively surveyed, whereby we believe that the field is still at the early stage due to an apparent gap between theoretical progress and real-world applications. Finally, we summarize the paper and point out some prospective research directions worth further investigation.

DECLARATIONS

Authors' contributions

Made substantial contributions to the research and investigation process, reviewed and summarized the literature, wrote and edited the original draft: Yan T, Xu Z
Performed oversight and leadership responsibility for the research activity planning and execution as well as developed ideas and provided critical review, commentary and revision: Yang SX, Gadsden SA

Availability of data and materials

Not applicable.

Financial support and sponsorship

This work was supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada.

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Copyright

REFERENCES

1. Goheen KR, Jefferys ER. Multivariable self-tuning autopilots for autonomous and remotely operated underwater vehicles. *IEEE J Oceanic Eng* 1990;15:144–51. DOI
2. García-Valdovinos LG, Salgado-Jiménez T, Bandala-Sánchez M, Nava-Balanzar L, Hernández-Alvarado R, Cruz-ledesma JA. Modelling, Design and robust control of a remotely operated underwater vehicle. *Int J Adv Robot Syst* 2014;11:1. DOI
3. Reisenbichler KR, Chaffey MR, Cazenave F, et al. Automating MBARI's midwater time-series video surveys: The transition from ROV to AUV. In: OCEANS 2016 MTS/IEEE Monterey; 2016. pp. 1–9. DOI
4. Petillot YR, Antonelli G, Casalino G, Ferreira F. underwater robots: from remotely operated vehicles to intervention-autonomous underwater vehicles. *IEEE Robot Automat Mag* 2019;26:94–101. DOI
5. Wynn RB, Huvenne VAI, Le Bas TP, et al. Autonomous underwater vehicles (AUVs): their past, present and future contributions to the advancement of marine geoscience. *Marine Geology* 2014;352:451–68. DOI
6. Shi Y, Shen C, Fang H, Li H. Advanced control in marine mechatronic systems: a survey. *IEEE/ASME Trans Mechatron* 2017;22:1121-31. DOI
7. Li J, Xu Z, Zhu D, et al. Bio-inspired intelligence with applications to robotics: a survey. *Intell Robot* 2021;1:58–83. DOI
8. Fossen TI. Guidance and control of ocean vehicles. University of Trondheim, Norway, Printed by John Wiley & Sons, Chichester, England, ISBN: 0 471 94113 1, Doctors Thesis 1999. DOI
9. Bogue R. Underwater robots: a review of technologies and applications. *Indus Robot* 2015;42:186-91. DOI
10. Gafurov SA, Klochkov EV. Autonomous unmanned underwater vehicles development tendencies. *Procedia Engineering* 2015;106:141–48. DOI
11. Oh KK, Park MC, Ahn HS. A survey of multi-agent formation control. *Automatica* 2015;53:424–40. DOI
12. Alam K, Ray T, Anavatti SG. A brief taxonomy of autonomous underwater vehicle design literature. *Ocean Engineering* 2014;88:627-30. DOI
13. Sousa J, Cruz N, Matos A, Pereira FL. Multiple AUVs for coastal oceanography. In: Oceans' 97. MTS/IEEE Conference Proceedings. vol. 1. IEEE; 1997. pp. 409–14. DOI
14. Singh H, Catipovic J, Eastwood R, et al. An integrated approach to multiple AUV communications, navigation and docking. In: OCEANS 96 MTS/IEEE Conference Proceedings. The Coastal Ocean-Prospects for the 21st Century. vol. 1. IEEE; 1996. pp. 59–64. DOI
15. Sotzing CC, Evans J, Lane DM. A multi-agent architecture to increase coordination efficiency in multi-auv operations. In: OCEANS 2007-Europe. IEEE; 2007. pp. 1–6. DOI
16. Yu W, Chen G, Cao M. Distributed leader–follower flocking control for multi-agent dynamical systems with time-varying velocities. *Syst Contr Letters* 2010;59:543–52. DOI
17. Yamaguchi H. A cooperative hunting behavior by mobile-robot troops. *Int J Robot Res* 1999;18:931–40. DOI
18. Vidal R, Shakernia O, Kim HJ, Shim DH, Sastry S. Probabilistic pursuit-evasion games: theory, implementation, and experimental evaluation. *IEEE Trans Robot Automat* 2002;18:662-9. DOI
19. Chung TH, Hollinger GA, Isler V. Search and pursuit-evasion in mobile robotics. *Auton Robot* 2011;31:299-316. DOI
20. Hespanha JP, Kim HJ, Sastry S. Multiple-agent probabilistic pursuit-evasion games. In: Proceedings of the 38th IEEE Conference on Decision and Control (Cat. No. 99CH36304). vol. 3. IEEE; 1999. pp. 2432–37. DOI
21. Guler S, Fidan B, Gazi V. Adaptive swarm coordination and formation control. In: Tan Y, editor. Handbook of Research on Design, Control, and Modeling of Swarm Robotics. IGI Global; 2016. pp. 175-206. DOI

22. Yan J, Yang X, Luo X, Chen C. Energy-efficient data collection over AUV-assisted underwater acoustic sensor network. *IEEE Syst J* 2018;12:3519-30. DOI
23. Huang M, Zhang K, Zeng Z, Wang T, Liu Y. An AUV-assisted data gathering scheme based on clustering and matrix completion for smart ocean. *IEEE Internet Things J* 2020;7:9904-18. DOI
24. Duan R, Du J, Ren J, et al. VoI based information collection for AUV assisted underwater acoustic sensor networks. In: ICC 2020-2020 IEEE International Conference on Communications (ICC). IEEE; 2020. pp. 1–6. DOI
25. Tan HP, Diamant R, Seah WK, Waldmeyer M. A survey of techniques and challenges in underwater localization. *Ocean Engineering* 2011;38:1663-76. DOI
26. Kinsey JC, Eustice RM, Whitcomb LL. A survey of underwater vehicle navigation: Recent advances and new challenges. In: IFAC conference of manoeuvering and control of marine craft. vol. 88. Lisbon; 2006. pp. 1–12. DOI
27. Yang Y, Xiao Y, Li T. A survey of autonomous underwater vehicle formation: Performance, formation control, and communication capability. *IEEE Commun Surv Tutorials* 2021;23:815-41. DOI
28. Fossen TI. Handbook of marine craft hydrodynamics and motion control. Chichester: John Wiley & Sons, Ltd; 2011. pp. 343-415. DOI
29. Das B, Subudhi B, Pati BB. Cooperative formation control of autonomous underwater vehicles: an overview. *Int J Autom Comput* 2016;13:199-225. DOI
30. Wei X, Wang X, Bai X, Bai S, Liu J. Autonomous underwater vehicles localisation in mobile underwater networks. *IJSNET* 2017;23:61. DOI
31. Chen YQ, Wang Z. Formation control: a review and a new consideration. In: 2005 IEEE/RSJ International conference on intelligent robots and systems. IEEE; 2005. pp. 3181–86. DOI
32. Issa B, Rashid AT. A survey of multi-mobile robot formation control. *IJCA* 2019;181:12-6. DOI
33. Zhang Y, Mehrjerdi H. A survey on multiple unmanned vehicles formation control and coordination: Normal and fault situations. In: 2013 International conference on unmanned aircraft systems (ICUAS). IEEE; 2013. pp. 1087–96. DOI
34. Do HT, Hua HT, Nguyen MT, et al. Formation control algorithms for multiple-uavs: a comprehensive survey. *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems* 2021;8:e3–e3. DOI
35. Ziquan Y, Zhang Y, Jiang B, Jun F, Ying J. A review on fault-tolerant cooperative control of multiple unmanned aerial vehicles. *Chinese J Aeronaut* 2022;35:1–18. DOI
36. Cai G, Dias J, Seneviratne L. A survey of small-scale unmanned aerial vehicles: recent advances and future development trends. *Un Sys* 2014;02:175-99. DOI
37. Dong X, Yu B, Shi Z, Zhong Y. Time-varying formation control for unmanned aerial vehicles: Theories and applications. *IEEE Trans Contr Syst Technol* 2015;23:340-8. DOI
38. Scharf DP, Hadaegh FY, Ploen SR. A survey of spacecraft formation flying guidance and control. part ii: control. In: Proceedings of the 2004 American control conference. vol. 4. Ieee; 2004. pp. 2976–85. DOI
39. Liu GP, Zhang S. A survey on formation control of small satellites. *Proc IEEE* 2018;106:440-57. DOI
40. Yuh J. Design and control of autonomous underwater robots: a survey. Autonomous Robots 2000;8:7–24. DOI
41. Li X, Zhu D, Qian Y. A survey on formation control algorithms for multi-AUV system. *Un Sys* 2014;02:351-9. DOI
42. Hadi B, Khosravi A, Sarhadi P. A review of the path planning and formation control for multiple autonomous underwater vehicles. *J Intell Robot Syst* 2021;101. DOI
43. Wang X, Zerr B, Thomas H, Clement B, Xie Z. Pattern formation of multi-AUV systems with the optical sensor based on displacement-based formation control. *Int J Syst Sci* 2020;51:348-67. DOI
44. Edwards D, Bean T, Odell D, Anderson M. A leader-follower algorithm for multiple AUV formations. In: 2004 IEEE/OES Autonomous Underwater Vehicles (IEEE Cat. No. 04CH37578). IEEE; 2004. pp. 40–46. DOI
45. Ren W, Sorensen N. Distributed coordination architecture for multi-robot formation control. *Robot Auton Syst* 2008;56:324–33. DOI
46. Cui R, Ge SS, How BVE, Choo YS. Leader–follower formation control of underactuated autonomous underwater vehicles. *Ocean Engineering* 2010;37:1491–502. DOI
47. Zheng J, Huang Y, Xiao Y. The effect of leaders on the consistency of group behaviour. *IJSNET* 2012;11:126–35. DOI
48. Cao X, Guo L. A leader–follower formation control approach for target hunting by multiple autonomous underwater vehicle in three-dimensional underwater environments. *Int J Adv Robot Syst* 2019;16:1729881419870664. DOI
49. Shi H, Wang L, Chu T. Virtual leader approach to coordinated control of multiple mobile agents with asymmetric interactions. *Physica D: Nonlinear Phenomena* 2006;213:51–65. DOI
50. Droge G. Distributed virtual leader moving formation control using behavior-based MPC. In: 2015 American Control Conference (ACC). IEEE; 2015. pp. 2323–28. DOI
51. Zheping Y, Yibo L, Jiajia Z, Gengshi Z. Moving target following control of multi-auvs formation based on rigid virtual leader-follower under ocean current. In: 2015 34th Chinese control conference (CCC). IEEE; 2015. pp. 5901–6. DOI
52. Li J, Du X. Underactuated multi-AUV robust formation control based on virtual leader. In: 2018 IEEE International Conference on Mechatronics and Automation (ICMA). IEEE; 2018. pp. 1568–73. DOI
53. Tan KH, Lewis MA. Virtual structures for high-precision cooperative mobile robotic control. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS'96. vol. 1. IEEE; 1996. pp. 132–39. DOI
54. Lewis MA, Tan KH. High precision formation control of mobile robots using virtual structures. *Auton Robot* 1997;4:387–403. DOI
55. Ren W, Beard RW. Decentralized scheme for spacecraft formation flying via the virtual structure approach. *Journal of Guidance, Control, and Dynamics* 2004;27:73–82. DOI

56. Yuan J, Tang GY. Formation control for mobile multiple robots based on hierarchical virtual structures. In: IEEE ICCA 2010. IEEE; 2010. pp. 393–98. DOI
57. Zhang Lc, Wang J, Wang T, Liu M, Gao J. Optimal formation of multiple AUVs cooperative localization based on virtual structure. In: OCEANS 2016 MTS/IEEE Monterey. IEEE; 2016. pp. 1–6. DOI
58. Zhen Q, Wan L, Li Y, Jiang D. Formation control of a multi-AUVs system based on virtual structure and artificial potential field on SE (3). *Ocean Engineering* 2022;253:111148. DOI
59. Yuan C, Licht S, He H. Formation learning control of multiple autonomous underwater vehicles with heterogeneous nonlinear uncertain dynamics. *IEEE Trans Cybern* 2018;48:2920-34. DOI
60. Balch T, Arkin RC. Behavior-based formation control for multirobot teams. *IEEE Trans Robot Automat* 1998;14:926–39. DOI
61. Monteiro S, Bicho E. A dynamical systems approach to behavior-based formation control. In: Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292). vol. 3. IEEE; 2002. pp. 2606–11. DOI
62. Xiaomin M, Yang D, Xing L, Sentang W. Behavior-based formation control of multi-missiles. In: 2009 Chinese Control and Decision Conference. IEEE; 2009. pp. 5019–23. DOI
63. Xu D, Zhang X, Zhu Z, Chen C, Yang P. Behavior-based formation control of swarm robots. *Mathematical Problems in Engineering* 2014;2014:1-13. DOI
64. Hacene N, Mendil B. Behavior-based autonomous navigation and formation control of mobile robots in unknown cluttered dynamic environments with dynamic target tracking. *Int J Autom Comput* 2021;18:766-86. DOI
65. Khatib O. Real-time obstacle avoidance for manipulators and mobile robots. In: Cox IJ, Wilfong GT, editors. Autonomous Robot Vehicles. New York: Springer; 1990. pp. 396-404. DOI
66. Gazi V. Swarm aggregations using artificial potentials and sliding-mode control. *IEEE Trans Robot* 2005;21:1208-14. DOI
67. Fiorelli E, Leonard NE, Bhatta P, Paley DA, Bachmayer R, et al. Multi-AUV control and adaptive sampling in Monterey Bay. *IEEE J Oceanic Eng* 2006;31:935-48. DOI
68. Barnes L, Fields M, Valavanis K. Unmanned ground vehicle swarm formation control using potential fields. In: 2007 Mediterranean Conference on Control & Automation. IEEE; 2007. pp. 1–8. DOI
69. Pereira AR, Hsu L. Adaptive formation control using artificial potentials for Euler-Lagrange agents. *IFAC Proceedings Volumes* 2008;41:10788–93. DOI
70. Sabattini L, Secchi C, Fantuzzi C. Arbitrarily shaped formations of mobile robots: artificial potential fields and coordinate transformation. *Auton Robot* 2011;30:385-97. DOI
71. Nair RR, Behera L, Kumar V, Jamshidi M. Multisatellite formation control for remote sensing applications using artificial potential field and adaptive fuzzy sliding mode control. *IEEE Syst* J2014;9:508–18. DOI
72. Ying Z, Xu L. Leader-follower formation control and obstacle avoidance of multi-robot based on artificial potential field. In: The 27th Chinese Control and Decision Conference (2015 CCDC). IEEE; 2015. pp. 4355–60. DOI
73. Ihle IA, Skjetne R, Fossen TI. Nonlinear formation control of marine craft with experimental results. In: 2004 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601). vol. 1. IEEE; 2004. pp. 680–85. DOI
74. Cui R, Xu D, Yan W. Formation control of autonomous underwater vehicles under fixed topology. In: 2007 IEEE International Conference on Control and Automation. IEEE; 2007. pp. 2913–18. DOI
75. Yu W, Wen G, Chen G, Cao J. Distributed cooperative control of multi-agent systems. John Wiley & Sons; 2017.
76. Fink A, Kosecoff J, Chassin M, Brook RH. Consensus methods: characteristics and guidelines for use. *Am J Public Health* 1984;74:979-83. DOI
77. Ren W, Beard RW, Atkins EM. A survey of consensus problems in multi-agent coordination. In: Proceedings of the 2005, American Control Conference, 2005. IEEE; 2005. pp. 1859–64. DOI
78. Ren W, Beard RW, McLain TW. Coordination variables and consensus building in multiple vehicle systems. In: Kumar V, Leonard N, Morse AS, editors. Cooperative Control. Berlin: Springer Berlin Heidelberg; 2005. pp. 171-88. DOI
79. Olfati-Saber R, Fax JA, Murray RM. Consensus and cooperation in networked multi-agent systems. *Proc IEEE* 2007;95:215-33. DOI
80. Anderson BD, Yu C, Fidan B, Hendrickx JM. Rigid graph control architectures for autonomous formations. *IEEE Control Syst* 2008;28:48-63. DOI
81. Mesbahi M, Egerstedt M. Graph theoretic methods in multiagent networks. In: Graph Theoretic Methods in Multiagent Networks. Princeton University Press; 2010. DOI
82. Ren W. Consensus based formation control strategies for multi-vehicle systems. In: 2006 American Control Conference. IEEE; 2006. pp. 6–pp. DOI
83. Porfiri M, Roberson DG, Stilwell DJ. Tracking and formation control of multiple autonomous agents: a two-level consensus approach. *Automatica* 2007;43:1318–28. DOI
84. Luo X, Han N, Guan X. Leader-following consensus protocols for formation control of multi-agent network. *J Syst Eng Electron* 2011;22:991-7. DOI
85. Dong R, Geng Z. Consensus based formation control laws for systems on Lie groups. *Syst Contr Letters* 2013;62:104–11. DOI
86. Dong R, Geng Z. Consensus for formation control of multi-agent systems. *Int J Robust Nonlinear Control* 2015;25:2481-501. DOI
87. Falconi R, Sabattini L, Secchi C, Fantuzzi C, Melchiorri C. Edge-weighted consensus-based formation control strategy with collision avoidance. *Robotica* 2015;33:332–47. DOI
88. Listmann KD, Masalawala MV, Adamy J. Consensus for formation control of nonholonomic mobile robots. In: 2009 IEEE international conference on robotics and automation. IEEE; 2009. pp. 3886–91. DOI

89. Wang W, Huang J, Wen C, Fan H. Distributed adaptive control for consensus tracking with application to formation control of nonholonomic mobile robots. *Automatica* 2014;50:1254–63. DOI
90. Peng Z, Wen G, Rahmani A, Yu Y. Distributed consensus-based formation control for multiple nonholonomic mobile robots with a specified reference trajectory. *Int J Syst Sci* 2015;46:1447–57. DOI
91. Peng Z, Wen G, Yang S, Rahmani A. Distributed consensus-based formation control for nonholonomic wheeled mobile robots using adaptive neural network. *Nonlinear Dynamics* 2016;86:605–22. DOI
92. Cheng Y, Jia R, Du H, Wen G, Zhu W. Robust finite-time consensus formation control for multiple nonholonomic wheeled mobile robots via output feedback. *Int J Robust Nonlinear Control* 2018;28:2082-96. DOI
93. Kuriki Y, Namerikawa T. Consensus-based cooperative formation control with collision avoidance for a multi-UAV system. In: 2014 American Control Conference. IEee; 2014. pp. 2077–82. DOI
94. Kuriki Y, Namerikawa T. Formation control with collision avoidance for a multi-UAV system using decentralized MPC and consensus-based control. *SICE Journal of Control, Measurement, and System Integration* 2015;8:285–94. DOI
95. Mu B, Li H, Ding J, Shi Y. Consensus in second-order multiple flying vehicles with random delays governed by a Markov chain. *Journal of the Franklin Institute* 2015;352:3628–44. DOI
96. Du H, Zhu W, Wen G, Duan Z, Lü J. Distributed formation control of multiple quadrotor aircraft based on nonsmooth consensus algorithms. *IEEE Trans Cybern* 2019;49:342-53. DOI
97. Kuo CW, Tsai CC, Lee CT. Intelligent leader-following consensus formation control using recurrent neural networks for small-size unmanned helicopters. *IEEE Trans Syst Man Cybern, Syst* 2021;51:1288-301. DOI
98. Wu Y, Gou J, Hu X, Huang Y. A new consensus theory-based method for formation control and obstacle avoidance of UAVs. *Aerospace Science and Technology* 2020;107:106332. DOI
99. Ren W. Distributed attitude consensus among multiple networked spacecraft. In: 2006 American control conference. IEEE; 2006. pp. 6–pp. DOI
100. Ren W. Distributed attitude alignment in spacecraft formation flying. *Int J Adapt Control Signal Process* 2007;21:95-113. DOI
101. Guiming L, Liangdong L. Coordinated multiple spacecraft attitude control with communication time delays and uncertainties. Chinese J Aeronaut 2012;25:698–708. DOI
102. Nazari M, Butcher EA, Yucelen T, Sanyal AK. Decentralized consensus control of a rigid-body spacecraft formation with communication delay. *Journal of Guidance, Control, and Dynamics* 2016;39:838-51. DOI
103. Silvestre C, Pascoal A. Control of the INFANTE AUV using gain scheduled static output feedback. *Control Engineering Practice* 2004;12:1501–9. DOI
104. Kaminer I, Pascoal AM, Khargonekar PP, Coleman EE. A velocity algorithm for the implementation of gain-scheduled controllers. *Automatica* 1995;31:1185–91. DOI
105. Mohamed SA, Osman AA, Attia SA, Maged SA. Dynamic model and control of an autonomous underwater vehicle. In: 2020 International Conference on Innovative Trends in Communication and Computer Engineering (ITCE). IEEE; 2020. pp. 182–90. DOI
106. Nag A, Patel SS, Kishore K, Akbar S. A robust H-infinity based depth control of an autonomous underwater vehicle. In: 2013 International Conference on Advanced Electronic Systems (ICAES). IEEE; 2013. pp. 68–73. DOI
107. Zhang Y, Liu X, Luo M, Yang C. MPC-based 3-D trajectory tracking for an autonomous underwater vehicle with constraints in complex ocean environments. *Ocean Engineering* 2019;189:106309. DOI
108. Naeem W, Sutton R, Ahmad S. LQG/LTR control of an autonomous underwater vehicle using a hybrid guidance law. *IFAC Proceedings Volumes* 2003;36:31–36. DOI
109. Long C, Qin X, Bian Y, Hu M. Trajectory tracking control of ROVs considering external disturbances and measurement noises using ESKF-based MPC. *Ocean Engineering* 2021;241:109991. DOI
110. Chellabi A, Nahon M. Feedback linearization control of undersea vehicles. In: Proceedings of OCEANS '93; 1993. pp. I410–ol.1. DOI
111. Shen C, Shi Y, Buckham B. Trajectory tracking control of an autonomous underwater vehicle using Lyapunov-based model predictive control. *IEEE Trans Ind Electron* 2018;65:5796-805. DOI
112. Shen C, Shi Y, Buckham B. Nonlinear model predictive control for trajectory tracking of an AUV: A distributed implementation. In: 2016 IEEE 55th Conference on Decision and Control (CDC). IEEE; 2016. pp. 5998–6003. DOI
113. Li H, Xie P, Yan W. Receding horizon formation tracking control of constrained underactuated autonomous underwater vehicles. *IEEE Trans Ind Electron* 2017;64:5004-13. DOI
114. Wei H, Shen C, Shi Y. Distributed Lyapunov-based model predictive formation tracking control for autonomous underwater vehicles subject to disturbances. *IEEE Trans Syst Man Cybern, Syst* 2019;51:5198–208. DOI
115. Khodayari MH, Balochian S. Modeling and control of autonomous underwater vehicle (AUV) in heading and depth attitude via self-adaptive fuzzy PID controller. *J Mar Sci Technol* 2015;20:559-78. DOI
116. Liang X, Qu X, Wan L, Ma Q. Three-dimensional path following of an underactuated AUV based on fuzzy backstepping sliding mode control. *Int J Fuzzy Systl* 2018;20:640-9. DOI
117. Nan D, Weng Y, Liu Y, Wang X. Accurate trajectory tracking control of unknown autonomous underwater vehicles: A data-driven predictive approach. In: 2021 IEEE 10th Data Driven Control and Learning Systems Conference (DDCLS). IEEE; 2021. pp. 1241–45. DOI
118. Wang D, He B, Shen Y, Li G, Chen G. A Modified ALOS Method of Path Tracking for AUVs with Reinforcement Learning Accelerated by Dynamic Data-Driven AUV Model. *J Intell Robot Syst* 2022;104:1–23. DOI
119. Shojaei K. Neural network formation control of underactuated autonomous underwater vehicles with saturating actuators. *Neurocom-*

puting 2016;194:372–84. DOI

120. Young KD, Utkin VI, Ozguner U. A control engineer's guide to sliding mode control. *IEEE Trans Contr Syst Technol* 1999;7:328–42. DOI
121. Edwards C, Spurgeon S. Sliding mode control: theory and applications. Crc Press; 1998. DOI
122. Perruquetti W, Barbot JP. Sliding mode control in engineering. vol. 11. Marcel Dekker New York; 2002. DOI
123. Yan Z, Wang M, Xu J. Robust adaptive sliding mode control of underactuated autonomous underwater vehicles with uncertain dynamics. Ocean Engineering 2019;173:802–9. DOI
124. Guo Y, Qin H, Xu B, Han Y, Fan QY, et al. Composite learning adaptive sliding mode control for AUV target tracking. *Neurocomputing* 2019;351:180–86. DOI
125. Huang B, Yang Q. Double-loop sliding mode controller with a novel switching term for the trajectory tracking of work-class ROVs. *Ocean Engineering* 2019;178:80–94. DOI
126. Yan Y, Yu S. Sliding mode tracking control of autonomous underwater vehicles with the effect of quantization. *Ocean Engineering* 2018;151:322–28. DOI
127. Lee PM, Hong SW, Lim YK, et al. Discrete-time quasi-sliding mode control of an autonomous underwater vehicle. *IEEE J Oceanic Eng*1999;24:388–95. DOI
128. Zhihong M, Yu XH. Terminal sliding mode control of MIMO linear systems. *IEEE Trans Circuits Syst I* 1997;44:1065-70. DOI
129. Wang Y, Gu L, Gao M, Zhu K. Multivariable output feedback adaptive terminal sliding mode control for underwater vehicles. *Asian J Contr* 2016;18:247–65. DOI
130. Elmokadem T, Zribi M, Youcef-Toumi K. Terminal sliding mode control for the trajectory tracking of underactuated Autonomous Underwater Vehicles. textitOcean Engineering 2017;129:613–25. DOI
131. Qiao L, Zhang W. Adaptive non-singular integral terminal sliding mode tracking control for autonomous underwater vehicles. textitIET Control Theory & Applications 2017;11:1293–306. DOI
132. Patre B, Londhe P, Waghmare L, Mohan S. Disturbance estimator based non-singular fast fuzzy terminal sliding mode control of an autonomous underwater vehicle. textitOcean Engineering 2018;159:372–87. DOI
133. Rangel MAG, Manzanilla A, Suarez AEZ, Muñoz F, Salazar S, et al. Adaptive non-singular terminal sliding mode control for an unmanned underwater vehicle: Real-time experiments. textitInt J Control Autom Syst 2020;18:615-28. DOI
134. Salgado-Jimenez T, Spiewak JM, Fraisse P, Jouvencel B. A robust control algorithm for AUV: based on a high order sliding mode. In: Oceans' 04 MTS/IEEE Techno-Ocean'04 (IEEE Cat. No. 04CH37600). vol. 1. IEEE; 2004. pp. 276–81. DOI
135. Li X, Zhu D. Formation control of a group of AUVs using adaptive high order sliding mode controller. In: OCEANS 2016-Shanghai. IEEE; 2016. pp. 1–6. DOI
136. Guerrero J, Antonio E, Manzanilla A, Torres J, Lozano R. Autonomous underwater vehicle robust path tracking: Auto-adjustable gain high order sliding mode controller. textitIFAC-PapersOnLine 2018;51:161–66. DOI
137. Wang J, Wang C, Wei Y, Zhang C. Sliding mode based neural adaptive formation control of underactuated AUVs with leader-follower strategy. textitApplied Ocean Research 2020;94:101971. DOI
138. Borlaug ILG, Pettersen KY, Gravdahl JT. Comparison of two second-order sliding mode control algorithms for an articulated intervention AUV: Theory and experimental results. textitOcean Engineering 2021;222:108480. DOI
139. Yan T, Xu Z, Yang SX. Consensus Formation Control for Multiple AUVSystems Using Distributed Bioinspired Sliding Mode Control. textitIEEE Trans Intell Veh 2022:1–1. DOI
140. Xu H, Zhang Gc, Cao J, Pang S, Sun Ys. Underactuated AUV nonlinear finite-time tracking control based on command filter and disturbance observer. textitSensors (Basel) 2019;19:4987. DOI
141. Guerrero J, Torres J, Creuze V, Chemori A. Adaptive disturbance observer for trajectory tracking control of underwater vehicles. textitOcean Engineering 2020;200:107080. DOI
142. Gao Z, Guo G. Fixed-time sliding mode formation control of AUVs based on a disturbance observer. textitIEEE/CAA J Autom Sinica 2020;7:539-45. DOI
143. Su B, Wang Hb, Wang Y. Dynamic event-triggered formation control for AUVs with fixed-time integral sliding mode disturbance observer. textitOcean Engineering 2021;240:109893. DOI
144. Wang H, Su B. Event-triggered formation control of AUVs with fixed-time RBF disturbance observer. textitApplied Ocean Research 2021;112:102638. DOI
145. Li J, Du J, Chang WJ. Robust time-varying formation control for underactuated autonomous underwater vehicles with disturbances under input saturation. textitOcean Engineering 2019;179:180–88. DOI
146. Gao Z, Guo G. Velocity free leader-follower formation control for autonomous underwater vehicles with line-of-sight range and angle constraints. textitInformation Sciences 2019;486:359–78. DOI
147. Wang J, Wang C, Wei Y, Zhang C. Observer-Based Neural Formation Control of Leader–Follower AUVs With Input Saturation. textitIEEE Syst J 2021;15:2553-61. DOI
148. Chen B, Hu J, Zhao Y, Ghosh BK. Finite-time velocity-free observer-based consensus tracking for heterogeneous uncertain AUVs via adaptive sliding mode control. textitOcean Engineering 2021;237:109565. DOI
149. Yan Z, Zhang C, Tian W, Cai S, Zhao L. Distributed observer-based formation trajectory tracking method of leader-following multi-AUV system. textitOcean Engineering 2022;260:112019. DOI
150. Filaretov V, Zhirabok A, Zuev A, Procenko A. The development of system of accommodation to faults of navigation sensors of underwater vehicles with resistance to disturbance. In: 2014 14th International Conference on Control, Automation and Systems (ICCAS 2014);

2014. pp. 1548–53. DOI

151. Xia Y, Xu K, Wang W, et al. Optimal robust trajectory tracking control of a X-rudder AUV with velocity sensor failures and uncertainties. textitOcean Engineering 2020;198:106949. DOI
152. Liu Z, Yu X, Yuan C, Zhang Y. Leader-follower formation control of unmanned aerial vehicles with fault tolerant and collision avoidance capabilities. In: 2015 international conference on unmanned aircraft systems (ICUAS). IEEE; 2015. pp. 1025–30. DOI
153. Wang X, Yadav V, Balakrishnan SN. Cooperative UAV Formation Flying With Obstacle/Collision Avoidance. textitIEEE Trans Contr Syst Technol 2007;15:672–79. [DOI: 10.1109/TCST.2007.899191] DOI
154. Chang K, Xia Y, Huang K. UAV formation control design with obstacle avoidance in dynamic three-dimensional environment. textit-SpringerPlus 2016;5:1–16. DOI
155. Shou Y, Xu B, Lu H, Zhang A, Mei T. Finite-time formation control and obstacle avoidance of multi-agent system with application. textitIntl J Robust & Nonlinear 2022;32:2883-901. DOI
156. Lobo Pereira F, Borges de Sousa J, Gomes R, Calado P. A model predictive control approach to AUVs motion coordination. In: van Schuppen JH, Villa T, editors. Coordination Control of Distributed Systems. Cham: Springer International Publishing; 2015. pp. 9-18. DOI
157. Guo H, Shen C, Zhang H, Chen H, Jia R. Simultaneous trajectory planning and tracking using an MPC method for cyber-physical systems: A case study of obstacle avoidance for an intelligent vehicle. textitIEEE Trans Ind Inf 2018;14:4273-83. DOI
158. Wang X, Yao X, Zhang L. Path planning under constraints and path following control of autonomous underwater vehicle with dynamical uncertainties and wave disturbances. textitJ Intell Robot Syst 2020;99:891–908. DOI
159. Lindqvist B, Mansouri SS, Agha-mohammadi Aa, Nikolakopoulos G. Nonlinear MPC for collision avoidance and control of UAVs with dynamic obstacles. textitIEEE Robot Autom Lett 2020;5:6001–8. DOI
160. Zhang Gl, Jia Hm. Global path planning of AUV based on improved ant colony optimization algorithm. In: 2012 IEEE International Conference on Automation and Logistics. IEEE; 2012. pp. 606–10. DOI
161. Lin C, Wang H, Yuan J, Fu M. An online path planning method based on hybrid quantum ant colony optimization for AUV. textitInt J Robot Autom 2018;33:435–44. DOI
162. Phung MD, Quach CH, Dinh TH, Ha Q. Enhanced discrete particle swarm optimization path planning for UAV vision-based surface inspection. textitAutom Construction 2017;81:25–33. DOI
163. Wang D, Fan T, Han T, Pan J. A two-stage reinforcement learning approach for multi-UAV collision avoidance under imperfect sensing. textitIEEE Robot Autom Lett 2020;5:3098-105. DOI
164. Yan Z, Yang Z, Yue L, et al. Discrete-time coordinated control of leader-following multiple AUVs under switching topologies and communication delays. textitOcean Engineering 2019;172:361–72. DOI
165. Sørensen FF, von Benzon M, Liniger J, Pedersen S. A quantitative parametric study on output time delays for autonomous underwater cleaning operations. textitJMSE 2022;10:815. DOI
166. Pedersen S, Liniger J, Sørensen FF, Schmidt K, von Benzon M, et al. Stabilization of a ROV in three-dimensional space using an underwater acoustic positioning system. textitIFAC-PapersOnLine 2019;52:117–22. DOI
167. Millán P, Orihuela L, Jurado I, Rubio FR. Formation control of autonomous underwater vehicles subject to communication delays. textitIEEE Trans Contr Syst Technol 2013;22:770–77. DOI
168. Yan Z, Pan X, Yang Z, Yue L. Formation control of leader-following multi-UUVs with uncertain factors and time-varying delays. textitIEEE Access 2019;7:118792–805. DOI
169. Chen S, Ho DW. Consensus control for multiple AUVs under imperfect information caused by communication faults. textitInformation Sciences 2016;370-371:565-77. DOI
170. Burlutskiy N, Touahmi Y, Lee BH. Power efficient formation configuration for centralized leader–follower AUVs control. textitJ Mar Sci Technol 2012;17:315–29. DOI
171. Sharif BS, Neasham J, Hinton OR, Adams AE. A computationally efficient Doppler compensation system for underwater acoustic communications. textitIEEE J Oceanic Eng 2000;25:52–61. DOI
172. Li B, Zhou S, Stojanovic M, Freitag L, Willett P. Multicarrier communication over underwater acoustic channels with nonuniform Doppler shifts. textitIEEE J Oceanic Eng 2008;33:198–209. DOI
173. Yoshizawa S, Saito T, Mabuchi Y, Tsukui T, Sawada S. Parallel resampling of OFDM signals for fluctuating doppler shifts in underwater acoustic communication. textitJ Electr Compu Eng 2018;2018:1-11. DOI
174. Hu Z, Ma C, Zhang L, et al. Formation control of impulsive networked autonomous underwater vehicles under fixed and switching topologies. textitNeurocomputing 2015;147:291–98. DOI
175. Yan Z, Xu D, Chen T, Zhang W, Liu Y. Leader-follower formation control of UUVs with model uncertainties, current disturbances, and unstable communication. textitSensors 2018;18:662. DOI
176. Seuret A, de Wit CC, et al. Contraction control of a fleet circular formation of AUVs under limited communication range. In: Proceedings of the 2010 American Control Conference. IEEE; 2010. pp. 5991–96. DOI
177. Tomera M. Hybrid switching controller design for the maneuvering and transit of a training ship. textitInt J Appl Mathem Compu Sci 2017;27:63–77. DOI

Intelligence & Robotics

Review

Open Access

An overview of intelligent image segmentation using active contour models

Yiyang Chen[1], Pengqiang Ge[1], Guina Wang[1], Guirong Weng[1], Hongtian Chen[2]

[1]School of Mechanical and Electrical Engineering, Soochow University, Suzhou, Jiangsu 215137, China.
[2]Department of Chemical and Materials Engineering, University of Alberta, Edmonton, AB T6G 1H9, Canada.

Correspondence to: Dr. Guina Wang, School of Mechanical and Electrical Engineering, Soochow University, No. 8, Jixue Road, Suzhou, Jiangsu 215137, China. E-mail: wangguina@suda.edu.cn; Dr. Hongtian Chen, Department of Chemical and Materials Engineering, University of Alberta, Edmonton, AB T6G 1H9, Canada. E-mail: chtbaylor@163.com; ORCID: 0000-0002-8600-9668

How to cite this article: Chen Y, Ge P, Wang G, Weng G, Chen H. An overview of intelligent image segmentation using active contour models. *Intell Robot* 2023;3(1):23-55. http://dx.doi.org/10.20517/ir.2023.02

Received: 9 Dec 2022 **First Decision:** 12 Jan 2023 **Revised:** 4 Feb 2023 **Accepted:** 7 Feb 2023 **Published:** 22 Feb 2023

Academic Editor: Simon Yang **Copy Editor:** Yin Han **Production Editor:** Yin Han

Abstract

The active contour model (ACM) approach in image segmentation is regarded as a research hotspot in the area of computer vision, which is widely applied in different kinds of applications in practice, such as medical image processing. The essence of ACM is to make use ofuse an enclosed and smooth curve to signify the target boundary, which is usually accomplished by minimizing the associated energy function by means ofthrough the standard descent method. This paper presents an overview of ACMs for handling image segmentation problems in various fields. It begins with an introduction briefly reviewing different ACMs with their pros and cons. Then, some basic knowledge in of the theory of ACMs is explained, and several popular ACMs in terms of three categories, including region-based ACMs, edge-based ACMs, and hybrid ACMs, are detailedly reviewed with their advantages and disadvantages. After that, twelve ACMs are chosen from the literature to conduct three sets of segmentation experiments to segment different kinds of images, and compare the segmentation efficiency and accuracy with different methods. Next, two deep learning-based algorithms are implemented to segment different types of images to compare segmentation results with several ACMs. Experimental results confirm some useful conclusions about their sharing strengths and weaknesses. Lastly, this paper points out some promising research directions that need to be further studied in the future.

Keywords: Active contour model, level set, energy function, intensity inhomogeneity, deep learning

1. INTRODUCTION

Image segmentation is a significant component in image processing, and serves as the foundation for image analysis and image understanding. The accuracy of image segmentation hugely affects the quality of subsequent image processing procedures. Its major role is to separate the input images into a series of disjoint sub-regions with unique features, and extract objects of interest. Therefore, image segmentation has been extensively employed in a variety of areas such as medical image processing[1–4], target recognition[5–8], moving target tracking[9–12], *etc*.

In the last decade, active contour model (ACM) using the level set approach has become one of the most efficient tools for image segmentation, which has been extensively employed in tasks of image segmentation. The image segmentation algorithm based on ACM is an image processing technique that combines upper-level and various prior knowledge for stable image segmentation, which can add image grayscale and edge information during the process of optimization. It provides a piece-wise smooth closed contour as the final outcome, which haswith superior performance such as diverse forms and flexible structures. ACM converts the image segmentation problem into the process of solving a minimization problem with the energy function. The contour of the target object is expressed by means of the zero level set in the execution process, which is convenient to dealfor dealing with the topological deformation during the curve evolution. Nevertheless, the topology of the segmented region changes in an automatic and uncontrollable manner can either be an advantage or an inconvenience according to different applications. The essence of ACM is to employ a continuous and closed curve to represent object boundary, which is achieved through the standard gradient descent approach to minimize the associated energy function.

ACMs are generally comprised of two categories: parametric ACMs and geometric ACMs. In parametric ACMs[13,14], the evolution curve is described in the parametric form to obtain object boundary. However, parametric ACMs can only deal with images that include a sole target object with an obvious boundary through the process of parameterization. Most importantly, they cannot automatically handle topology changes during the process of curve evolution. The geometric ACM is also named as the level set method, which guides evolution curves to evolve towards the target boundary through the geometric measurement parameters. The introduction of level set methods makes it possible to segment images with multiple target objects simultaneously, and solves the issue of topology changes (merging or broken curves) caused by the process of parameterization in parametric ACMs. This paper mainly pays attention to the existing geometric ACMs, and they can be further categorized into three types: region-based ACMs[15–17], which aim at identifying each region of interest using a defined region descriptor to guide the evolution motion of active contour; edge-based ACMs[18,19], which utilize gradient information of the target boundary as the major driving force to attract the active contour to the object boundary; and hybrid ACMs[20–24], which combine local region and edge information together to instruct evolution curve to move towards target boundary.

The Chan-Vese (CV) model[15] utilized the average gray values of the outside and inside areas of the contour to characterize the foreground and background of the input image, respectively. As a classical region-based ACM, CV model does not need to utilize image gradient information, which makes it very suitable to segmentfor segmenting images with blurred or discontinuous edges. However, for images with uneven grayscale such as images subjected to uneven illumination, CV model could obtain undesired segmentation results in the form of having difficulty extracting meaningful objects out of images and falling into local minima[25]. To solve this issue, the pieceiwse piecewise smooth (PS) model[26] was developed to segment images with intensity non-uniformity to some degree due to the consideration of image local attributes. Nevertheless, PS model is sensitive to different initial contours and inefficient due to the complex computation process. The geometric active contours (GAC) model[27,28] translated curve evolution into the evolution of level set function through variational methods, which effectively solves topology change problems. However, this model has to continuously re-initialize the level set function to zero level set, which results in inefficient segmentation and possible boundary leakage. In addition, the segmentation results of medical images that usually contain noise and blurred or discontinuous edges are poor. The fast global minimization (FGM) model[29] defined global minimizers to overcome the drawback of falling into local minima in the local optimization approach such as CV model[15], which enables the FGM model to be independent of random positions of initial contours and gets rid of frequent re-initialization of distance function in GAC model[28]. The model[30] was competent to obtain the global minimum of an ACM between two sides, which enables the initialization process easier and reduces the chance of falling into a local minimum at the false edge. The key of this model is to take advantage of a novel numerical method to compute the minimum route, which is the global minimum of the associated energy function among all routes connecting the two end points.

To make the level set function inherently stable, the distance regularized level set evolution (DRLSE) model[31] added a distance regularization term, which controls the deviation between the level set function and the standard signed distance function during the curve evolution. In addition, this model avoided the problem of constant re-initialization during the curve evolution. Nevertheless, this model has no self-adjustment ability during the process of energy minimization due to uni-directional area term, and remains sensitive to different selections of initial contours. The bias correction (BC) model[32] was designed to segment the image and compute the bias field simultaneously to correct unevenly distributed intensity in medical images, which is more precise and has less segmentation time than the famous PS model. However, this model is nowadays inefficient and less accurate than many newly developed ACMs. In addition, it is not very effective in segmenting natural images taken from nature. The local binary fitting (LBF)[33] and region- scalable fitting (RSF)[34] models were constructed to segment images with intensity non-uniformity, which use a kernel function to design a local binary fitting energy and embeds information of local area to guide the motion of level set function. In addition, these two models incorporates a penalty term in the energy function, which avoids the periodic re-initialization process and greatly improves algorithm efficiency. However, the introduced kernel function only calculates the grayscale value of image locally, which makes it possible to get trapped into local minimum during the procedure of minimizing its energy. In other words, these two models are sensitive to initial contours. In addition, it takes time to calculate the two fitting functions that need to be continuously updated during each iteration, resulting in the inefficient segmentation of RSF model. The local image fit (LIF) model[35] considered the technique of Gaussian filtering and local image information to segment different images with intensity non-uniformity, which segments images faster than RSF model due to only two convolution operations during each iteration. However, this model still remains susceptible to different initial contours. Specifically, an inappropriate initial contour may result in a wrong segmentation due to the fact that the majority of existing ACMs have non-convex energy functions. To solve the issue of non-convex functions, the approach[36] was designed to translate non-convex function to convex function, which handles the problem of local minima frequently occurred occurring in non-convex function. Nevertheless, this approach is too complex and time-consuming to be applied in practice. In addition, the method[37] numerically tracked an accurate numerical approximation of the most optimized solution for some relaxed problems, which is capable of providing a close bound between the calculated solution and the real minimizer. Nevertheless, this model is not guaranteed to obtain a global minimizer of the minimal partition problem (also known as spatially continuous Potts model).

The local and global intensity fitting (LGIF) model[38] was defined as a linear combination of local image fit (LIF) energy and global image fit (GIF) energy. By choosing the appropriate weights that are used to control the ratio of LIF energy and GIF energy, this model can effectively handle the grayscale non-uniformity and has good initialization robustness. However, the weights of LIF and GIF models are unpredictable for different images and often need to be manually calibrated with respect to the degree of grayscale non-uniformity. The segmentation will fail if it is chosen poorly[39]. The local gaussian distribution fitting (LGDF) model[40] defined a fitting energy based on the mean and variance of the local gray values. Compared with RSF model, this model is able to segment local areas with the same mean gray value but different variances. However, this model is less efficient than RSF model due to the fact that more time is consumed to compute the variances. In addition, this model is also sensitive to different initial contours[41].

The core of local region Chan-Vese (LRCV) model[42] was to replace two fitting constants in the CV model with the two fitting functions in RSF model. In addition, this model utilizes the segmentation result of degraded CV model as the initial contour, which can reduce the dependence on the initial contour to a certain extent and accelerate the segmentation speed at the same time. Considering that many targets and backgrounds in real images are random, the local histogram fitting (LHF) model[43] took the advantage of two fitted histograms to approximate the distribution of the target and background, which can be used to segment regions with unpredictable distributions. However, it is inefficient because it needs to calculate the histogram distribution for each gray level (0-255). Similarly, it is sensitive to the initial contours. The local and global Gaussian distribution fitting (LGGDF) model[44] constructed a linear combination of a local and global Gaussian fit energies with a changeable weight to balance the local and global energies, which further decreases the dependence on the choices of initial contours. However, it is computationally intensive and the adaptive weight does not work well for some images. The local likelihood image fitting (LLIF) model[45] mainly utilized mean intensity and variance information of the local region. In fact, LLIF model is a combination of LIF model and LGDF model, which has enhanced applicability for segmenting images. However, the segmentation efficiency is relatively low, while the robustness to initialization is not appealing[46].

The RSF&LoG model[39] combined RSF model with optimized Laplacian of Gaussian (LoG) energy to improve segmentation results, which further improve sensitivity to different initial contours. Nevertheless, the segmentation time of this model is relatively long[47] due to the unoptimized computation procedure. The local pre-fitting (LPF) model[48] pre-calculated mean intensities of local regions ahead of iteration to obtain faster segmentation speed. Nevertheless, this model still faces some common issues such as stagnation of false boundaries, under- segmentation[49]. Therefore, the segmentation accuracy of this model still has space to be further improved. The LPF&FCM model[41] locally fitted out two fuzzy center points inside and outside the evolution curve ahead of iteration through the fuzzy c-means (FCM) clustering algorithm, which reduces computation cost and improves segmentation efficiency. In addition, this model puts combines an adaptive edge indicator function and an adaptive sign function together to resolve the issue of single direction of evolution contour to realize bidirectional motion.

The super-pixel based via a local similarity factor and saliency (SLSFS) model[50] linked super-pixel with FCM clustering algorithm to create initial contours, which is competent to create adaptive initial contour in the neighborhood of the target and effectively protect weak edge information. The model[51] constructed an adaptive weight ratio to calibrate the relationship between local energy part and global energy part, which is capable of automatically calibrating the direction of curve evolution with respect to the location of the target region. Nevertheless, the initial contour still has to be manually labeled during the process of curve evolution. The approach[52] associated the level set method (LSE) model[32] with region and edge synergetic level set framework (RESLS) model[53] to improve segmentation results, which is able to efficiently segment images with unevenly distributed intensity and extends the two-phase model to multi-phase model. However, this model is sensitive to the choice of parameters and incompetent to in effectively processing natural images with complicated background information. The method[54] employed self- organizing maps (SOM) to cluster the input image into two regions: foreground and background regions, which decreases the interference of noise and enhances system robustness. However, compared with K-mean clustering algorithm, SOM algorithm may obtains relatively smaller lower computation precision owing to the update of neighborhood nodes. The global and local fuzzy image fitting (GLFIF) model[55] utilized a combination of global and local fitting energy to process images with noise and non-uniform intensity, which hugely decreases the influences of background noise and intensity non-uniformity to obtain accurate segmentation result.

The additive bias correction (ABC) model[56] employed the theory of bias field correction to effectively segment images with unevenly distributed intensity and achieved good segmentation results. However, the issue of under- segmentation may occur while segmenting images with Gaussian noise interference, as described in Section 4, which means that the anti-noise robustness of this model still has space to be optimized. The pre-fitting energy (PFE) model[47] calculated median intensities of local regions before iteration begins began to decrease segmentation time. In addition, this model contains a novel single well potential function and its corresponding evolution speed function to facilitate the evolution speed of the level set function to achieve fast image segmentation. However, issues of stagnation of false boundaries and under- segmentation may take placeoccur during the process of evolution process. The above said issues are illustrated and explained in detail in Section 4. Therefore, this model still has room for improvement in terms of system robustness and segmentation accuracy. The pre-fitting bias correction (PBC) model[57] utilized an optimized FCM algorithm to pre-calculate the bias field before iteration, which is able to effectively segmenting images with unevenly distributed intensity and greatly reduces segmentation time. However, the segmentation accuracy and efficiency may be inversely affected if the FCM algorithm has bad performanceperforms poorly. The local and global Jeffreys divergence (LGJD) model[58] put local and global data fitting energies together to measure the difference between input image and fitted image through the Jeffreys divergence theory, which can effectively segment natural and medical images with intensity non-uniformities. However, this model has a long segmentation time[47] due to the unoptimized computation process. The adaptive local pre-fitting energy function, based on Jeffreys divergence (APFJD) model[49], embedded two pre-fitting functions to construct an enhanced energy function. This model replaces the traditional Euclidean distance with the theory of Jeffreys divergence to measure the distance between real image and fitted image, which is proved to be more capable of segmenting images with intensity non-uniformity efficiently and effectively. Nevertheless, the matter of under- segmentation sometimes happens when segmenting images with Gaussian noise, as described and explained in detail in Section 4, which indicates that this model still has room for improvement regarding robustness against noise interference.

In the beginning of this paper, the authors have briefly reviewed the diverse ACMs (region-based ACMs, edge-based ACMs, and hybrid ACMs) in the area of image segmentation with their pros and cons. Then, several typical models in region-based ACMs, edge-based ACMs, and hybrid ACMs have beenwere reviewed with their advantages and disadvantages, respectively. After that, 12 typical ACMs chosen from the literature review have beenwere selected to conduct three comparison experiments on different kinds of images (synthetic images, medical images, and natural images). Next, two deep-learning based algorithms have been implemented to segment double-phase images and multi-phase images, whose experimental results are compared with several ACMs to demonstrate their strengths and weaknesses. Lastly, some promising research directions and works have been recommended to subsequent researchers. The rest of this paper is arranged as follows: Section 2 explains some basic knowledge of the ACMs theory. Section 3 reviews several popular ACMs in three categories, including region-based ACMs, edge-based ACMs, and hybrid ACMs, with their advantages and disadvantages. Section 4 describes experimental results with respect to segmentation experiments of synthetic images, medical images, and natural images. Section 5 presents several possible research directions.

2. RELATED KNOWLEDGE

2.1. Curve evolution

Geometric ACMs[59–62] are mainly on the basis of partial differential equations (PDEs) and variational method, whose essence is to continuously evolve toward the direction of energy minimum under the constraint of image information and give conditions. The segmentation process is generally as follows: a closed curve is initialized on the given image. Then, the curve evolves under the combined effect of internal and external energies, and stops evolving when the energy function achieves a minimal value through gradient descent method. Lastly, the zero level set coincides with the target edge to complete segmentation.

The goal of the level set approach is to find out the zero level set, which represents the target boundary as energy function is minimized through standard descent method. In other words, this level set method utilizes zero level set one dimension higher to express the evolution result of low-dimensional target. During the curve evolution, the points on the curve move towards their normal directions at a certain velocity respectively, with time as a variable respectively according to a certain velocity. In addition, the speed and direction of the motion are mainly controlled by two parameters: curvature and unit normal vector.

A closed and smooth curve C is defined in two dimensions[63] as follows:

$$\frac{dC}{dp} = T, \frac{d^2C}{dp} = kN, \tag{1}$$

where p is the curve parameter, k denotes the curvature, T signifies the tangent line, and N represents the normal line. Note that $T(p)$ and $N(p)$ are perpendicular to each other, so the direction and magnitude of the motion of any point on the curve C can be represented by these two vectors. By adding the time variable t, the evolution of the curve is represented as

$$\frac{dC(t)}{dt} = \alpha_1 T + \alpha_2 N, \tag{2}$$

where α_1 denotes the point speed on the curve in the tangential direction, and α_2 signifies the point speed on the curve in the normal direction. Since the shape change and geometric properties of the curve during evolution process are only related to the speed in the normal direction. Therefore, only the normal speed is taken into consideration, while the velocity component in the tangential direction is chosen to be ignored for better segmentation efficiency in practical applications.

Therefore, Equation (2) can be simplified as

$$\frac{dC(t)}{dt} = F_n N, \tag{3}$$

where F_n is the speed function used to represent the motion speed of all points on the curve.

2.2. Level set function

The fundamental idea of level set method is to express the evolution of a closed curve C in the plane as the evolution of the intersection of a higher dimensional function with the horizontal plane by using the expression of an implicit function, which performs interface tracing and shape modeling by solving the zero level set function[64]. Specifically, the level set method employs a level set function a dimension higher to implicitly express a two-dimensional closed curve, or a three-dimensional surface, or a multi-dimensional hyper-surface, which transforms the process of curve evolution into the evolution problem of level set function one dimension higher.

The level set function is always a valid function when the topology of the closed curve or surface embedded in the level set function changes. Instead of tracking the position of the evolved curve, the level set function is continuously updated under the action of solving a partial differential evolution equation to figure out its zero level set when image segmentation is performed by the level set method. The zero level set at that moment is derived when the evolution process stops under some certain criteria, which means the position of the zero level set is the location of the object contour after segmentation.

2.3. Energy function

The internal energy is determined by the internal properties of the curve, which defines an enlargeable and bendable curve deformation energy term, and maintains the continuity and smoothness of the contour curve by adjusting the weights to control the consistency of the elastic tensor of curve bending and the rigid tensor of stretching. The external energy determined by image information consists of image constraint energy term and image potential energy term[65]. There is no fixed expression formula for the constraint energy term, which is usually constructed according to users' demands or image features. The external energy determines the evolution direction of the active contour, which guides the evolution contour line to evolve towards the target boundary.

3. ACTIVE CONTOUR MODEL

In this section, some representative ACMs in of three types (region-based ACMs, edge-based ACMs, and hybrid ACMs) are reviewed with their pros and cons in detail.

3.1. Region-based ACMs

3.1.1. Mumford Shah model

MS model[26] unifies image data, initial estimation and target contour in a feature extraction process under the constraint of knowledge, which is capable of autonomously converging to the energy minimum energy state after proper initialization. This model converts image segmentation issue into minimization of the energy function as follows:

$$e^{MS}(v, K) = p \int_{\Omega} (v - I)^2 dx + q \int_{\Omega \backslash K} |\nabla v|^2 dx + r|K|, \tag{4}$$

where v is the fitted image, I is the original input image, ∇ denotes gradient operator, $|K|$ is the length of contour line K, and p, q, r are positive coefficients to control the associated segments.

The energy function in Equation (4) is comprised of three terms: the first data fidelity term ($p \int_{\Omega}(v - I)^2 dx$) maintains the similarity between original input image and segmentation result; the second curve smoothing term ($q \int_{\Omega \backslash K} |\nabla v|^2 dx$) makes segmentation result smooth, and the third length constraint term ($r|K|$) constrains the curve length. Among these terms, data fidelity term and curve smoothing term utilize the feature of local region information to get rid of unnecessary contours. The most optimized contour K is obtained through the minimization of Mumford and Shah energy function in Equation (4), which segments the original input image I into several non-overlapping areas, and a fitted image v after the process of smoothing. However, it may have the issue of several local minima since that $e^{MS}(v, K)$ is not convex. In addition, it is time-consuming and inefficient to solve Equation (4) because of incompatible dimensions of v and K[66].

3.1.2. Chan Vese model

CV model[15] considers the image global characteristics and image statistical information inside and outside the evolution curve to drive the curve to approach the contour of the target area, which achieves success in the segmentation of images with blurred edges and small gradient changes and remains insensitive to noise. The CV energy function is proposed as

$$e^{CV}(c_1, c_2, C) = \int_{\text{outside}(\ C)} (I - c_1)^2\, dx + \int_{\text{inside }(C)} (I - c_2)^2\, dx + a|C|, \tag{5}$$

where a is a constant; c_1 and c_2 denotes the grayscale averages of the outer and inner regions of the curve, respectively; $|C|$ represents the length of evolution curve; the first two terms in Equation (5) are data-driven terms that are utilized to guide the curve to evolve towards target boundary, and the last term in Equation (5) is length constraint term that is used to control the curve length as well as smooth it. According to Equation (5), the energy function e^{CV} reaches the minimum value when curve C is on the edge of target boundary. In the process of minimizing the CV energy, the curve C can be represented by the level set function ϕ, which generates the following rewritten energy function as follows:

$$E^{CV}(\phi, c_1, c_2) = \int_{\Omega} |I - c_1|^2 H_{\varepsilon}(\phi(x)) dx + \int_{\Omega} |I - c_2|^2 \left[1 - H_{\varepsilon}(\phi(x))\right] dx + u \int_{\Omega} \delta_{\varepsilon}(\phi(x)) |\nabla \phi(x)| dx, \tag{6}$$

where $H_\varepsilon(\phi(x))$ and $\delta_\varepsilon(\phi(x))$ are approximated Heaviside and Dirac functions defined as

$$H_\varepsilon(x) = \frac{1}{2}\left(1 + \frac{2}{\pi}\arctan\left(\frac{x}{\varepsilon}\right)\right), \tag{7}$$

$$\delta_\varepsilon(x) = H'_\varepsilon(x) = \frac{\varepsilon}{\pi\left(\varepsilon^2 + x^2\right)}. \tag{8}$$

Utilizing the standard gradient descent approach to minimize the energy function in Equation (6), therefore, the issue of minimizing the energy function is transformed into solving the gradient descent function, which obtains the following gradient flow function (level set evolution function) as follows:

$$\frac{\partial\phi}{\partial t} = -\delta_\varepsilon(\phi)\left[(I(x) - c_1)^2 - (I(x) - c_1)^2\right] + a\delta_\varepsilon(\phi)\,\mathrm{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right), \tag{9}$$

with c_1 and c_2 being

$$\begin{cases} c_1 = \frac{\int_\Omega I(x)\cdot H_\varepsilon(\phi(x))dx}{\int_\Omega H_\varepsilon(\phi(x))dx}, \\ c_2 = \frac{\int_\Omega I(x)\cdot[1-H_\varepsilon(\phi(x)]dx}{\int_\Omega[1-H_\varepsilon(\phi(x))]dx}. \end{cases} \tag{10}$$

Lastly, the zero level set can be obtained through iteratively solving $\phi^{k+1} = \phi^k + \Delta t \cdot \partial\phi/\partial t$. The iteration process will stop either when the convergence criteria are satisfied or the maximum iteration number is reached.

CV model has fair segmentation speed and initialization robustness[42,67]. However, c_1 and c_2 are only related to the global gray value of the input image. Therefore, the segmentation result will be wrong if the gray values inside and outside the curve C are different. In other words, this model cannot segment images with intensity non-uniformity, which limits its application scope.

3.1.3. Region scalable fitting model

RSF model employs Gaussian kernel function to extract image characteristics locally, which can effectively process images with uneven grayscale. To overcome the drawback of CV model, RSF model[34] is proposed. The RSF energy function based on local gray values is proposed as

$$\varepsilon_x^{\mathrm{Fit}}(C, f_1(x), f_2(x)) = \lambda_1 \int_{\mathrm{outside}\ (C)} K(x-y)\,|I(y) - f_1(x)|^2\,dy + \lambda_2 \int_{\mathrm{inside}\ (C)} K(x-y)\,|I(y) - f_2(x)|^2\,dy, \tag{11}$$

where λ_1 and λ_2 are constant values; $f_1(x)$ and $f_2(x)$ signify local fitting functions outside and inside curve C; image intensity $I(y)$ denotes local region centered at point x, whose size is controlled by Gaussian kernel function K. In fact, $\varepsilon_x^{\mathrm{Fit}}$ denotes the weighted average squared error between the fitted values $f_1(x)$ and $f_2(x)$ and the truth grayscale values. Therefore, given a centroid x, the fitted energy $\varepsilon_x^{\mathrm{Fit}}$ is minimized when the fitted values $f_1(x)$ and $f_2(x)$ are the best approximation of the local image grayscale values on both sides of the contour C, which means that the contour C is on the target boundary. For all points x in the image domain, the total energy e^{RSF} can be computed by integrating $\int \varepsilon_x^{Fit}(C, f_1(x), f_2(x))\,dx$, which is expressed as follows:

$$\begin{aligned} e^{RSF}(\phi, f_1(x), f_2(x)) =& \lambda_1 \int_\Omega\left(\int_\Omega K_\sigma(x-y)\,|I(y) - f_1(x)|^2\,H_\varepsilon(\phi(y))dy\right)dx \\ &+ \lambda_2 \int_\Omega\left(\int_\Omega K_\sigma(x-y)\,|I(y) - f_2(x)|^2\,[1 - H_\varepsilon(\phi(y))]\,dy\right)dx, \end{aligned} \tag{12}$$

where $H_\varepsilon(\phi(x))$ and $\delta_\varepsilon(\phi(x))$ are approximated Heaviside and Dirac functions defined in Equation (7) and Equation (8), respectively. In addition, length constraint term $L(\phi)$ is added to smooth and shorten the contour C while distance regularization term $P(\phi)$ is introduced to maintain the regularity of level set function ϕ to avoid its re-initialization. Therefore, the total energy of RSF model is defined as

$$F^{RSF}(\phi, f_1(x), f_2(x)) = e^{RSF}(\phi, f_1(x), f_2(x)) + a_1 L(\phi) + a_2 P(\phi), \tag{13}$$

where a_1 and a_2 are constant values related to the length constraint term $L(\phi)$ and the distance regularization term $P(\phi)$, respectively, which are defined as

$$L(\phi) = \int_\Omega \delta_\varepsilon(\phi(y))|\nabla\phi(y)|dx, \tag{14}$$

$$P(\phi) = \int_\Omega \frac{1}{2}(|\nabla\phi(y)| - 1)^2 dx. \tag{15}$$

Applying the standard gradient descent method[68] to minimize energy F^{RSF}. Firstly, fix level set function ϕ and minimize energy F^{RSF} with respect to $f_1(x)$ and $f_2(x)$ through partial derivative respectively, which generates following functions as

$$\begin{cases} f_1(x) = \frac{\int_\Omega K_\sigma(x-y)[H_\varepsilon(\phi(y))\cdot I(y)]dy}{\int_\Omega K_\sigma(x-y)H_\varepsilon(\phi(y))dy}, \\ f_2(x) = \frac{\int_\Omega K_\sigma(x-y)[(1-H_\varepsilon(\phi(y))\cdot I(y)]dy}{\int_\Omega K_\sigma(x-y)[1-H_\varepsilon(\phi(y))]dy}. \end{cases} \tag{16}$$

Secondly, fix $f_1(x)$ and $f_2(x)$ and minimize energy F^{RSF} with respect to level set function ϕ through partial derivative respectively, which generates following gradient flow function as follow:

$$\frac{\partial\phi^{RSF}}{\partial t} = -\delta_\varepsilon(\phi)\left(\lambda_1 e_1 - \lambda_2 e_2\right) + a_1\delta_\varepsilon(\phi)\operatorname{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) + a_2\left(\nabla^2\phi - \operatorname{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right)\right), \tag{17}$$

with $e_1(x)$ and $e_2(x)$ being

$$\begin{cases} e_1(x) = \int_\Omega K_\sigma(y-x)\left|I(y) - f_1(x)\right|^2 dy, \\ e_2(x) = \int_\Omega K_\sigma(y-x)\left|I(y) - f_2(x)\right|^2 dy. \end{cases} \tag{18}$$

In Equation (17), a_1, a_2 are positive constants, and the gradient flow is composed of three terms: the first term $-\delta_\varepsilon(\phi)\left(\lambda_1 e_1 - \lambda_2 e_2\right)$ represents the data-driven term that drives curve C towards target boundary to complete segmentation; the second term $a_1\delta_\varepsilon(\phi)\operatorname{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right)$ signifies arc length of the contour C, which is used to smooth or shorten the length of the contour C; the third term $a_2\left(\nabla^2\phi - \operatorname{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right)\right)$ denotes the regularization term of level set function, which is utilized to maintain the regularity of level set function.

RSF model sufficiently takes advantage of local image information through Gaussian kernel function, which enables it to effectively segment images with intensity non-uniformity. However, the incorporated kernel function only calculates the grayscale values of local image regions, which renders the energy function F^{RSF} to easily fall into the local minimum during the process of iteration. Accordingly, this model is very susceptible to the selection of initial contour. In addition, at least 4 convolutions have been performed to update the 2 fitting functions $f_1(x) and f_2(x)$ in Equation (16) during each iteration, which leads to inefficient segmentation.

3.1.4. Local image fitting model

To reduce the computation time in RSF model, LIF model[35] is put forward to modify and optimize the calculation procedure of fitting functions in RSF model, which greatly reduces the number of convolution operations required to update the fitting functions.

The LIF energy function is constructed to minimize the difference between the fitted image and the actual one, which is expressed as

$$e^{\mathrm{LIF}}(\phi) = \frac{1}{2}\int_\Omega \left|I(y) - I_f(x,y)\right|^2 dx, \tag{19}$$

where $I_f(x)$ is the local fitted image defined as

$$I_f(x,y) = m_1(x)H(\phi(y)) + m_2(x)(1 - H(\phi(y))). \tag{20}$$

Note that $H_\varepsilon(\phi(y))$ and $\delta_\varepsilon(\phi(y))$ are approximated Heaviside and Dirac functions defined in Equation (7) and Equation (8), respectively; local fitting functions $m_1(x)$ and $m_2(x)$ are

$$\begin{cases} m_1(x) = \text{mean}\left(I \in (\{y \in \Omega \mid \phi(y) < 0\} \cap \Omega_k(x))\right), \\ m_2(x) = \text{mean}\left(I \in (\{y \in \Omega \mid \phi(y) > 0\} \cap \Omega_k(x))\right), \end{cases} \tag{21}$$

where x signifies the center point of initial contour while y denotes all point in a specific chosen region; $\Omega_k(x)$ is a truncated Gaussian window K_σ with size $(4k+1) \times (4k+1)$ and standard deviation σ. In fact, $m_1(x)$ and $m_2(x)$ serves as two local fitting functions outside and inside the contour C. Note that K_σ is a Gaussian kernel function with size $k \times k$ and standard deviation σ. The parameter σ is used to control local region size with respect to image features. Because of the localization property of the Gaussian kernel function, the contribution of image intensity $I(y)$ fades away if the distance between the center point x and the point y is far. In other words, the image intensity of point y in the vicinity of the center point x mainly contributes to the existence of LIF energy. Therefore, this model is capable of precisely handling images with unevenly distributed intensity.

Utilizing the steepest descent method[68] to minimize the energy function $e^{\text{LIF}}(\phi)$ in Equation (19), which generates the gradient flow equation as follows:

$$\frac{\partial \phi^{LIF}}{\partial t} = \left(I - I_f(x)\right)\left(m_1(x) - m_2(x)\right)\delta_\varepsilon(\phi), \tag{22}$$

where z_1, z_2 are positive constants; $\delta_\varepsilon(\phi(x))$ are approximated Dirac functions described in Equation (8).

The main contribution of LIF model is to re-write data-driven term $\lambda_1 e_1 - \lambda_2 e_2$ in RSF model in Equation (17), which reduces the convolution number of each iteration to update the fitting functions from 4 to 2 to save a huge amount of computation time.

The LIF model re-writes data-driven term $\lambda_1 e_1 - \lambda_2 e_2$ in RSF model in Equation (17) as follows:

$$\lambda_1 e_1 - \lambda_2 e_2 = (\lambda_1 - \lambda_2)\, I^2(y)\, [K_\sigma(x) * \mathbf{1}] - 2I(y)\, [K_\sigma(x) * (\lambda_1 f_1(x) - \lambda_2 f_2(x))] + K_\sigma(x) * \left(\lambda_1 f_1^2(x) - \lambda_2 f_2^2(x)\right), \tag{23}$$

with $e_1(x)$ and $e_2(x)$

$$e_1(x) = \int_\Omega K_\sigma(y-x)\, |I(y) - f_1(x)|^2 \; \mathrm{d}y = I^2(y)\, [K_\sigma(x) * \mathbf{1}] - 2I(y)\, [K_\sigma(x) * f_1(x)] + K_\sigma(x) * f_1^2(x), \tag{24}$$

$$e_2(x) = \int_\Omega K_\sigma(y-x)\, |I(y) - f_2(x)|^2 \; \mathrm{d}y = I^2(y)\, [K_\sigma(x) * \mathbf{1}] - 2I(y)\, [K_\sigma(x) * f_2(x)] + K_\sigma(x) * f_2^2(x). \tag{25}$$

In Equation (23), the first convolution term $K_\sigma(x) * \mathbf{1}$ only needs to be calculated once before iteration begins. Note that $\mathbf{1}$ is a matrix full of ones, and $K_\sigma(x) * \mathbf{1} = \int K_\sigma(y-x)dy$, which equals to constant 1 anywhere but the edge of the image region Ω. Therefore, there are only two convolution terms left $K_\sigma(x) * (\lambda_1 f_1(x) - \lambda_2 f_2(x))$ and $K_\sigma(x) * \left(\lambda_1 f_1^2(x) - \lambda_2 f_2^2(x)\right)$ to be calculated in each convolution.

Compared with RSF model, although LIF model does not contain length constraint and distance regularization terms, it utilizes Gaussian filtering to smooth the curve C as well as regularize the level set function, which reduces the possibility of the occurrence of local minima. In addition, there are only 2 convolutions to update fitting functions in Equation (23) in LIF model during each iteration instead of 4 convolution to update fitting functions in Equation (16) in RSF model, which saves a great amount of computation time. However, the incorporated Gaussian kernel function also only computes the grayscale values of local image area, which also makes it easy to get stuck at a local minimum. That is to say, this the model maintains sensitivity to different initial contours.

3.2. Edge-based ACMs

3.2.1. Geodesic active contour model

GAC model[28] constructively integrates the concept of edge indicator function into energy function, which can flexibly deal with topology changes and guide the contour line to converge at the target boundary.

GAC energy function[28] based on edge indicator function is defined as

$$e(C) = \int_0^1 \left(e_{\text{int}}\left(C'(q) + e_{\text{ext}}\left(C(q)\right)\right) dq, \tag{26}$$

where e_{int} is length constraint term and e_{ext} is area term that are defined respectively as

$$e_{\text{int}}\left(C'(q)\right) = \alpha_1 \left|C'(q)\right|^2, \tag{27}$$

$$e_{\text{ext}}\left(C(q)\right) = \gamma_1 g_\beta(I) \left|\nabla I\left(C(q)\right)\right|^2. \tag{28}$$

Note that α_1 and γ_1 are constant values; g_β is the edge indicator defined as

$$g_\beta(I) = \frac{1}{1 + \left|\nabla\left(K_\sigma * I\right)\right|^2}, \tag{29}$$

where K_σ is the Gaussian kernel function with standard deviation σ. Utilizing the standard gradient method to minimize energy function in Equation (26), which generates gradient flow function as

$$\frac{\partial \phi}{\partial t} = \alpha_1 |\nabla\phi| \operatorname{div}\left(g_\beta \frac{\nabla\phi}{|\nabla\phi|}\right) + \gamma_1 g_\beta(I)|\nabla\phi|. \tag{30}$$

GAC model obtains a closed curve (the zero level set) by continuously updating level set function under certain rules, which can flexibly handles changes in curve topology. However, this model cannot realize adaptive segmentation and requires human intervention. Specifically, the sign and magnitude of evolution speed need to be determined manually with respect to the location of initial contour (inside or outside the target boundary), which leads to the issue of repetitive re-initialization of level set function during iteration process and possible boundary leaking. In addition, this model highly depends on the boundary gradient as well as initial position, which means that only those boundary pixels with relatively strong great gradient changes are likely to be detected.

3.2.2. Distance regularized level set evolution model

To solve the problem of repetitive re-initialization of level set function in GAC model, DRLSE model[31] incorporates a distance regularization term into the classic ACM to calibrate the deviation between the level set function and the standard symbolic distance function (SDF) in the curve evolution process, so that the level set function can maintain its internal stability, and finally avoids the problem of continuous re-initialization in the curve evolution process.

DRLSE energy function is described as

$$E^{DRLSE}(\phi) = \mu_1 \int_\Omega p_2(|\nabla\phi|)dx + \mu_2 \int_\Omega g_\beta \delta_\varepsilon(\phi)|\nabla\phi|dx + \mu_3 \int_\Omega g_\beta H_\varepsilon(-\phi)dx, \tag{31}$$

where μ_1, μ_2, μ_3 are constant values; ∇ denotes gradient operator; ϕ is the level set function; g_β is the edge indicator function defined in Equation (29); $H_\varepsilon(x)$ and $\delta_\varepsilon(x)$ are regularized Heaviside and Dirac functions denoted in Equation (7) and Equation (8), respectively; and the double well potential function $p(s)$ and its associated derivative $p_2'(s)$ are defined as follows respectively:

$$p_2(s) = \begin{cases} \frac{1}{(2\pi)^2}(1 - \cos(2\pi s)), & s \le 1, \\ \frac{1}{2}(s-1)^2, & s \ge 1, \end{cases} \tag{32}$$

$$p_2'(s) = \begin{cases} \frac{1}{2\pi}\sin(2\pi s), & s \le 1, \\ s-1, & s \ge 1. \end{cases} \tag{33}$$

Employing the steepest gradient descent method to minimize energy function in Equation (31), which obtains the following gradient descent flow equation as

$$\frac{\partial \phi}{\partial t} = \mu_1 \operatorname{div}\left(d_{p_2}(|\nabla\phi|)\nabla\phi\right) + \mu_2\delta_\varepsilon(\phi)\operatorname{div}\left(g\frac{\nabla\phi}{|\nabla\phi|}\right) + \mu_3 g\delta_\varepsilon(\phi), \tag{34}$$

where div(s) denotes vector divergence; the evolution speed function $d_{p_2}(s)$ is defined as

$$d_{p_2}(s) = \frac{p_2'(s)}{s} = \begin{cases} \frac{1}{2\pi s}\sin(2\pi s), & s \le 1, \\ 1-\frac{1}{s}, & s \ge 1. \end{cases} \tag{35}$$

DRLSE model incorporates the distance rule function to solve the deviation between the level set function and the singed distance function, which means that the level set function no longer requires the re-initialization operation in the iterative process. However, this model has several drawbacks, as follows:

- The area term utilized to facilitate the evolution speed of the zero level set is a single value (positive or negative), which can be only chosen either from positive to zero or negative to zero during the process of energy minimization. In a word, this model has no self-adjustment ability and cannot realize adaptive segmentation.
- The area and length terms are highly dependent on the edge indicator function that is constructed by the gradient of the input image. The edge indicator function will be almost zero if the gradient is big, which renders the target boundary after Gaussian filtering blurry and wider. In this case, the target boundaries may be interconnected due to Gaussian smoothing when the distance between targets is very close, which results in segmentation failure.
- The constant μ_3 must be set manually, which has a great influence on the segmentation results. the The evolution speed will be slowed down if μ_3 is chosen too small, and the evolution speed will be too large that the target boundary leaks if μ_3 is set too big.
- The evolution speed function $d_{p_2}(s)$ has a maximum value of 1 when $s = 0$, which renders the evolution curve evolve so quickly that it may intrude into the target. In addition, the evolution speed function $d_{p_2}(s)$ has a small slope when $s = 1$, which leads to slow evolution speed.

3.2.3. Adaptive level set evolution model

To solve the issue of unidirectional motion of area term in DRLSE model, ALSE model[69] adds an adaptive sign variable parameter to the area term of the energy function, so that the evolution curve can iterate according to the current position and choose the direction independently. Its corresponding gradient flow function is defined as

$$\frac{\partial \phi^{ALSE}}{\partial t} = \mu\left(\Delta\phi - \operatorname{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right)\right) + \lambda g div\left(\frac{\nabla\phi}{|\nabla\phi|}\right)\delta(\phi) + v\left(I, c_3, c_4\right) g\delta(\phi), \tag{36}$$

where μ and λ are constants; div(s) signifies the divergence of vector; the edge indicator function g is denoted as

$$g(I) = \exp\left(-\frac{|\nabla I_\sigma|}{k_3}\right), \tag{37}$$

where k_3 is a positive constant used to control the slope of edge indicator function, and the area term $v\left(I, c_3, c_4\right) g\delta_\varepsilon(\phi)$ is defined as

$$v\left(I, c_3, c_4\right) = \alpha \operatorname{sign}\left(I(x,y) - \frac{c_3+c_4}{2}\right), \tag{38}$$

with

$$\begin{cases} c_3 = \frac{\int_\Omega I(x,y)H_\varepsilon(-\phi)\mathrm{d}x\ \mathrm{d}y}{\int_\Omega H_\varepsilon(-\phi)\mathrm{d}x\ \mathrm{d}y}, \\ c_4 = \frac{\int_\Omega I(x,y)H_\varepsilon(\phi)\mathrm{d}x\ \mathrm{d}y}{\int_\Omega H_\varepsilon(\phi)\mathrm{d}x\ \mathrm{d}y}. \end{cases} \tag{39}$$

In Equation (36), the gradient flow function is composed of three parts: the first part $\mu\left(\Delta\phi - \operatorname{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right)\right)$ based on the distance rule term is used to reduce the error between the level set function and the signed distance function, which gets rid of re-initialization during the process of iteration; the second part $\lambda g div\left(\frac{\nabla\phi}{|\nabla\phi|}\right)\delta(\phi)$ is the length constraint term that is utilized to enhance the effect of shortening and smoothing the zero-level contour, which effectively maintains the regularity of the evolution curve; the third part $v\left(I, c_3, c_4\right) g\delta(\phi)$ is the area term with variable coefficients, which is used to adjust the magnitude and direction of evolution contour line.

Compared with DRLSE model, this model introduces the weighted coefficient $v\left(I, c_3, c_4\right)$ in Equation (38) to substitute constant value μ_3 in area term in Equation (34). The direction of this sign function $v\left(I, c_3, c_4\right)$ is determined by the mean difference between $I(x, y)$ and the mean values of the images outside and inside the contour line. Therefore, the gradient flow function can adjust the direction of evolution motion with respect to the image grayscale information inside and outside the initial contour, which solves the issue of unidirectional motion of the zero level set in DRLSE model and improves the robustness of the initial contour. However, the issues such as the tendency to fall into false boundaryboundaries, leaking from weak edges, and poor anti-noise ability remain unsolved.

3.2.4. Fuzzy c-means model

To realize bidirectional motion of zero level set to accomplish adaptive segmentation, FCM model[41] links optimized FCM algorithm that calculates local image intensity with optimized adaptive functions, which resolves the issues of leaking from vulnerable boundary and slow computation process.

FCM energy function is constructed as

$$\begin{aligned} E^{FCM}(\phi) =& k_1 R_{p_w}(\phi) + k_2 L_{g_{\beta_1}}(\phi) + \varphi\left(I_{\sigma_1}, C_1, C_2\right) A_{g_{\beta_1}}(\phi) \\ =& k_1 \int_\Omega p_w(|\nabla\phi|)dx + k_2 \int_\Omega g_{\beta_1}\delta_\varepsilon(\phi)|\nabla\phi|dx + \varphi\left(I_{\sigma_1}, C_1, C_2\right)\int_\Omega g_{\beta_1} H_\varepsilon(-\phi)dx, \end{aligned} \tag{40}$$

where k_1 and k_2 are positive constants; ϕ is the level set function; I_{σ_1} is the image vector after Gaussian filtering; C_1, C_2 denotes the FCM energy; $H_\varepsilon(x)$ and $\delta_\varepsilon(x)$ are regularized Heaviside and Dirac functions denoted in Eq. 7 and Eq. 8, respectively; the adaptive edge indicator function $g_{\beta_1}(I)$ is described as

$$g_{\beta_1}(I) = \frac{1}{1 + \left|\nabla I_{\sigma_1}\right|^2 / \beta_1^2}, \tag{41}$$

with

$$\beta_1(I) = \frac{1 + \sqrt{S\left(I_{\sigma_1}\right)}}{3}, \tag{42}$$

where S denotes standard deviation value of image after Gaussian filtering, the adaptive sign function $\varphi\left(I_{\sigma_1}, C_1, C_2\right)$ in the area term is defined as

$$\varphi\left(I_{\sigma_1}, C_1, C_2\right) = \eta \arctan\left[\left(I_{\sigma_1} - \frac{C_1 + C_2}{2}\right)/\tau\right]. \tag{43}$$

In Equation (43), η and τ are positive constant values; $I_{\sigma_1} = G_\sigma * I$; the two clustering results C_1, C_2 are obtained through the cluster centroids $c_{j,1}$ and membership function $\mu_j\left(x_i\right)$, which are described respectively as follows:

$$c_{j,1} = \frac{\sum_{i=1}^{n\times(2\omega+1)^2}\left[\mu_j\left(x_i\right)\right]^\alpha x_i}{\sum_{i=1}^{n\times(2\omega+1)^2}\left[\mu_j\left(x_i\right)\right]^\alpha}, \tag{44}$$

$$\mu_j\left(x_i\right)=\frac{\sum_{s=1}^{k}\left|x_i-c_{j,1}\right|^{\frac{-2}{\alpha-1}}}{\sum_{s=1}^{k}\left|x_i-c_{s,1}\right|^{\frac{-2}{\alpha-1}}}, \tag{45}$$

where image size of $I(x)$ is $m \times n$; x_i signifies ith pixel in the first row of image region; the weighted ratio α equals to 2; $n \times (2\omega+1)^2$ is the total number of elements in particular sample; $(2\omega+1)^2$ signifies the width of square frame. In FCM model, the cluster number k equals to 2, and $C_1 = c_{1,1}$ and $C_2 = c_{2,1}$.

For the purpose of maintaining evolution stability, the potential function $p_w(s)$ and its corresponding $p_w{}'(s)$ are denoted respectively as follows:

$$p_w(s)=\frac{1}{2}s^2+\frac{w}{2}\exp\left[-\left(\frac{s-0.75}{w}\right)^2\right]+0.375\sqrt{\pi}\operatorname{erf}\left(\frac{s-0.75}{w}\right), \tag{46}$$

$$p_w{}'(s)=\frac{4}{3}s\left\{0.75-\exp\left[-\frac{(s-0.75)^2}{w^2}\right]\right\}, \tag{47}$$

where erf$(\cdot)$ denotes the Gaussian error function; w is equal to 0.465; the evolution speed function $d_{p_w}(s)$ is denoted as

$$d_{p_w}(s)=\frac{p_w{}'(s)}{s}=\frac{4}{3}\left\{0.75-\exp\left[-\frac{(s-0.75)^2}{w^2}\right]\right\}. \tag{48}$$

The evolution speed function $d_{p_w}(s)$ in Equation (48) is inspired by evolution speed functions $d_{p2(s)}$ in Equation (35). In DRLSE model[31], the evolution speed functions $d_{p2(s)}$ in Equation (35) has a slow final convergence speed due to a small slope at one well potential $|\nabla\phi| = 1$ as shown in Figure 2. Note that the one well potential is defined at $|\nabla\phi| = 1$ by convention[31]. The motivation of $d_{p_w}(s)$ is to raise the slope at one well potential $|\nabla\phi| = 1$ to solve the issue of slow convergence speed of $d_{p2(s)}$ in Equation (35), which also increases the sensitivity of distance regularized term $k_1\int_\Omega p_w(|\nabla\phi|)dx$ in Equation (40).

Applying the gradient descent method to minimize energy function in Equation (40), which obtains the flowing gradient descent flow equation as follows:

$$\frac{\partial\phi^{FCM}}{\partial t}=-\frac{\partial E^{FCM}(\phi)}{\partial\phi}=k_1\operatorname{div}\left(d_{p_w}(|\nabla\phi|)\nabla\phi\right)+k_2\delta_\varepsilon(\phi)\operatorname{div}\left(g_{\beta_1}\frac{\nabla\phi}{|\nabla\phi|}\right)+\varphi\left(I_{\sigma_1},C_1,C_2\right)g_{\beta_1}\delta_\varepsilon(\phi). \tag{49}$$

In Equation (49), the gradient descent flow function is mainly made up of two components. The first component is the internal energy part that contains distance regularized term ($k_1\operatorname{div}\left(d_{p_w}(|\nabla\phi|)\nabla\phi\right)$), which offsets the deviation between the sign distance function (SDF) and level set function to resolve the problem of repeated initialization during the process of evolution. The second component is the external energy part that consists of length constraint part ($k_2\delta_\varepsilon(\phi)\operatorname{div}\left(g_{\beta_1}\frac{\nabla\phi}{|\nabla\phi|}\right)$) and area part ($\varphi\left(I_{\sigma_1},C_1,C_2\right)g_{\beta_1}\delta_\varepsilon(\phi)$). The length constraint term is utilized to guide the zero level to evolve towards target boundary as well as control contour length due to the effect of the adaptive edge indicator function g_{β_1}. The area term is used to adjust contour velocity through the effect of adaptive sign function $\varphi\left(I_{\sigma_1},C_1,C_2\right)$, which achieves bidirectional evolution with respect to image grayscale value.

To better understand the working mechanism of FCM model, the corresponding flow chart is illustrated in Figure 1. Note that the convergence criterion is set as $|(S_{i+5}-S_i)/S| < 10^{-5}$, and S signifies the entire area of input image.

FCM model is characterized by pre-fitting the fuzzy two centroids inside and outside the contour line using the local area-based fuzzy C-mean clustering principle before iteration to construct an adaptive edge indicator function, which solves the one-way motion problem of the edge level set model. However, FCM algorithm applied in this model is unoptimized and complex, which results in relatively long segmentation time. In addition, segmentation may fail in the forms of falling into false boundaryboundaries, if FCM algorithm has poor performance.

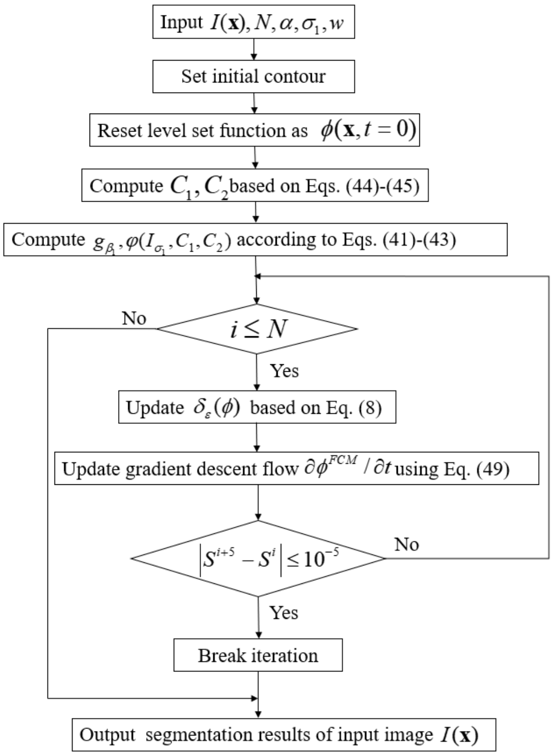

Input $I(\mathbf{x}), N, \alpha, \sigma_1, w$
Set initial contour
Reset level set function as $\phi(\mathbf{x}, t=0)$
Compute C_1, C_2based on Eqs. (44)-(45)
Compute $g_\beta, \varphi(I_{\sigma_1}, C_1, C_2)$ according to Eqs. (41)-(43)
No
$i \le N$
Yes
Update $\delta_\varepsilon(\phi)$ based on Eq. (8)
Update gradient descent flow $\partial\phi^{FCM} / \partial t$ using Eq. (49)
$|S^{i+5} - S^i| \le 10^{-5}$
No
Yes
Break iteration
Output segmentation results of input image $I(\mathbf{x})$

Figure 1. The flow chart of FCM model.

3.3. Hybrid ACMs

3.3.1. Optimized local pre-fitting image model

To achieve better segmentation accuracy and reduce computation cost, OLPFI model[70] is proposed to associate region-based attributes and edge-based attributes through mean local pre-fitting functions, which is capable of effectively segments segmenting images with uneven intensity and noise disturbance.

OLPFI energy function is defined as

$$e^{\text{OLPFI}}\left(\phi(x)\right) = \frac{A}{2}\int_{\Omega}\int_{\Omega} K_\sigma(y-x)\left|I(x) - f^{LPFI}(x,\phi(x))\right|^2 g_e(x)dxdy. \tag{50}$$

Note that A is a positive variable used to manually adjust segmentation speed according to the target size; K_σ is the Gaussian kernel function with standard deviation σ; the edge indicator function $g_e(\mathrm{x})$ is defined as

$$g_e(x) = 1 - \frac{2}{\pi}\arctan\left(\left|\nabla\left(I * K_\sigma\right)\right|^2/\tau\right), \tag{51}$$

where ϕ is the level set function; $\tau = \text{std}\,2(I(x))$ is the standard deviation of the image in the matrix form.

There are 4 variables A, σ, w, k to be manually calibrated to meet the ideal segmentation results. Specifically, variable A is used to adjust segmentation speed according to target size to prevent issues of under- segmentation or over- segmentation; variable σ in Gaussian kernel function K_σ is properly adjusted to collect image information locally with respect to object size to prevent issues of under- segmentation or over- segmentation; variable k is the magnitude of average filter, which is appropriately calibrated to filter out irrelevant information and obtain a smoothed final contour; variable w is the size of small local region, which should be increased or decreased properly to entirely cover targets in the input image. In addition, variable w should be increased to filter out noise and unrelated pixel information while segmenting images with strong noise disturbance.

In Equation (50), the local pre-fitted image (LPFI) function is defined as

$$f^{LPFI}(x,\phi(x)) = L_1(x)H_\varepsilon(\phi(x)) + L_2(x)\left(1 - H_\varepsilon(\phi(x))\right), \tag{52}$$

with $L_1(x)$ and $L_2(x)$

$$\begin{cases} L_1(x) = \min\left[I(y) \mid y \in \Omega_x\right], \\ L_2(x) = \max\left[I(y) \mid y \in \Omega_x\right], \end{cases} \tag{53}$$

where Ω_x denotes a small rectangular local area with size $(2w+1)^2$ at center point x; $I(\mathrm{y})$ denotes the gray values of all points y in Ω_x; these two pre-fitting functions $L_1(x), L_2(x)$ of OLPFI model are inspired by local fitting function $m_1(x), m_2(x)$ of LIF model in Equation (21). Specifically, $m_1(x), m_2(x)$ in Equation (21) have to be computed for curve evolution during each iteration, which means that n iterations will calculate $m_1(x), m_2(x)$ n times. Therefore, LIF model has slow segmentation speed and heavy computation cost due to unoptimized fitting functions $m_1(x), m_2(x)$. To address this issue, pre-fitting functions $L_1(x), L_2(x)$ in in Equation (53) quickly pre-calculates the foreground and background of the input image ahead of iteration process and are independent of iteration process, which saves a great amount of computation time and confers much faster segmentation speed than LIF model. Utilizing the standard descent method to minimize the energy function in Equation (50), which obtains the following gradient descent flow function as follows:

$$\frac{\partial\phi}{\partial \mathrm{t}} = -\frac{\partial E^{OLPFI}}{\partial\phi} = -A\delta_\varepsilon(\phi)\cdot(L_1 - L_2)\cdot g_e(x)\int_\Omega K_\sigma(y-x)\left(I - f^{LPFI}\right)dy. \tag{54}$$

Note that the global minimizer can be computed point by point by simply solving $\phi(x) := \underset{\psi\in\mathbb{R}}{\operatorname{argmin}}\left|I(x) - f^{LPFI}(x,\psi)\right|$. However, to follow up the convention, the method of partial derivative equation (PDE) is applied to conduct the optimization process. The PDE approach also provides an opportunity to regularize the level set function at each time step as described in Equation (58), which improves the segmentation performance.

In order to improve segmentation performance, Equation (54) is rewritten as

$$\frac{\partial\phi^{OLPFI}}{\partial \mathrm{t}} = -A\delta_\varepsilon(\phi)\cdot \mathrm{esign}\left(\frac{h(x)}{\tau}\cdot \mathrm{g}_e(x)\right), \tag{55}$$

where esign $(\cdot)$ and $h(x)$ respectively defined as

$$\mathrm{esign}(x) = \mathrm{sign}(x)\left(1 - e^{-x^2}\right), \tag{56}$$

$$h(x) = (L_1 - L_2)\left[I\left(K_\sigma * \mathbf{1}\right) - K_\sigma * \left((L_1 - L_2)\,H_\varepsilon(\phi)\right) - K_\sigma * L_2\right]. \tag{57}$$

In order to effectively regularize the level set function and smooth evolution curve, a regularization function ϕ_R and a length constraint function ϕ_L are respectively defined as

$$\begin{cases} \phi_R = \mathrm{esign}\left(8\cdot\phi^{i+1}\right), \\ \phi_L = \mathrm{mean}\left(\phi_R(y) \mid y \in \Omega_{\mathrm{x}}\right), \end{cases} \tag{58}$$

where the regularization function ϕ_R is used to regularize the level set function ϕ to generate a more stable evolution environment; length constraint function ϕ_L is utilized to get rid of unrelated curves as well as shorten and smooth evolution curves through an average filter with size $k \times k$.

In Equation (58) ϕ^{i+1} is the level set function ϕ at (i+1)-th iteration, which follows level set evolution function defined as

$$\phi^{i+1} = \phi^i + \Delta t\cdot\frac{\partial\phi^{OLPFI}}{\partial t}, \tag{59}$$

where Δt is time interval; ϕ^i is the level set function ϕ at i-th iteration; $\partial\phi^{OLPFI}/\partial t$ is defined in Equation (55). OLPFI model combines local pre-fitting image functions based on mean intensity and edge indicator function to segment images with uneven intensity and noise interference, which achieves relatively high segmentation accuracy. The pre-fitting function pre-computes local image intensity ahead of iteration, which achieves fast image segmentation and greatly reduces the computation cost. However, the issue of under- segmentation may take place when segmenting images with noise interference due to the traditional Euclidean distance. In addition, boundary leaking may sometimes occur in the form of broken curves when segmenting images with large objects.

3.3.2. Pre-fitting energy model

To obtain better segmentation precision and decrease CPU elapsed time, PFE model[49] combines median pre-fitting functions with optimized adaptive functions, which solves the issue of unidirectional motion of evolution curve and hugely decreases computation cost.

PFE energy function is constructed as

$$\begin{aligned} E^{PFE}(\phi) =& n_1 R_{p_3}(\phi) + n_2 L_{g_m}(\phi) + \varphi_1\left(I_\sigma, c_l, c_s\right) A_{g_m}(\phi) \\ =& n_1 \int_\Omega p_3(|\nabla\phi|)dx + n_2 \int_\Omega g_m \delta_\varepsilon(\phi)|\nabla\phi|dx + \varphi_1\left(I_\sigma, c_l, c_s\right) \int_\Omega g_{\beta_m} H_\varepsilon(-\phi)dx, \end{aligned} \tag{60}$$

where n_1, n_2, n_3 are positive constants; ϕ is the level set function; I_σ is the image after Gaussian filtering; $H_\varepsilon(\phi(x))$ and $\delta_\varepsilon(\phi(x))$ are approximated Heaviside and Dirac functions defined in Equation (7) and Equation (8) respectively; $\varphi_1\left(I_\sigma, c_l, c_s\right)$ denotes the adaptive sign function $\varphi\left(I_\sigma, c_l, c_s\right)$ that is descried as

$$\varphi\left(I_\sigma, c_l, c_s\right) = n_3 \arctan\left[\left(I_\sigma - \frac{c_l + c_s}{2}\right)/\tau\right]. \tag{61}$$

Note that $\tau = \operatorname{std} 2(I(\mathrm{x}))$ is the standard deviation of the image in the matrix form; the adaptive edge indicator function $g_m(I)$ is

$$g_m(I) = 1 - \tanh\frac{|\nabla K_\sigma * I|^2}{m}, \tag{62}$$

$$m(I) = 2S\left(I_\sigma\right), \tag{63}$$

with S denoting the standard deviation of image after Gaussian filtering; $I_\sigma = K_\sigma * I$ is the image after Gaussian filtering, and K_σ is a Gaussian filtering template with standard deviation σ; two pre-fitting functions c_l,c_s are defined as

$$\begin{cases} f_{\text{median}}(\mathbf{x}) = \operatorname{median}\left(I(\mathbf{y}) \mid \mathbf{y} \in \Omega_{\mathbf{x}}\right), \\ c_l(\mathbf{x}) = \operatorname{mean}\left(I(\mathbf{y}) \mid \mathbf{y} \in \Omega_l\right), \\ c_s(\mathbf{x}) = \operatorname{mean}\left(I(\mathbf{y}) \mid \mathbf{y} \in \Omega_s\right), \end{cases} \tag{64}$$

where f_{median} denotes the median intensity in a local area $\Omega_{\mathbf{x}}$ centered at point $\mathbf{x}$ with radius w; c_l and c_s are average intensities in Ω_l and Ω_s, respectively; $I(\mathbf{y})$ signifies a local area at center point $\mathbf{y}$; parameter $\mathbf{x}$ denotes the center point of initial contour. As known that the fitting functions of RSF model in Equation (16) are unoptimized and complex, which results in huge computation costs and low segmentation efficiency. To solve this drawback, these pre-fitting functions of PFE model in Equation (64) quickly fit out the foreground and background before iteration process takes place and are independent of it, which dramatically saves computation cost and increases segmentation efficiency.

In Equation (64), Ω_l and Ω_s are respectively defined as follows:

$$\begin{cases} \Omega_l = \left\{\mathbf{y} \mid \left(I(\mathbf{y}) > f_{\text{median}}(\mathbf{x})\right\} \cap \Omega_{\mathbf{x}}, \\ \Omega_s = \left\{\mathbf{y} \mid \left(I(\mathbf{y}) < f_{\text{median}}(\mathbf{x})\right\} \cap \Omega_{\mathbf{x}}. \end{cases} \tag{65}$$

Note that Ω_l is the local region inside $\Omega_{\mathbf{x}}$, where the image intensities are all bigger than f_{median} in $\Omega_{\mathbf{x}}$; Ω_s is the local region inside $\Omega_{\mathbf{x}}$, where the image intensities are all less than f_{median} in $\Omega_{\mathbf{x}}$.

The single potential function $p_3(s)$ and its corresponding evolution speed function $d_{p_3}(s)$ are constructed as

$$p_3(s) = \begin{cases} \frac{1}{3}s^3 + \frac{1}{\pi^3}\sin(\pi(s-1)) - \frac{s}{\pi^2}\cos(\pi(s-1)), \\ -\frac{1}{2}s^2 + \frac{1}{6} + \frac{1}{\pi^2}, s \in [0,1], \\ \frac{1}{2}s^2 - \arctan\left(s^2\right) - \frac{1}{2} + \frac{\pi}{4}, s \in [1,+\infty), \end{cases} \tag{66}$$

$$d_{p_3}(s) = \frac{p_3'(s)}{s} = \begin{cases} s + \frac{1}{\pi}\sin(\pi(s-1)) - 1, s \in [0,1], \\ 1 - \frac{2}{1+s^4}, s \in (1,\infty). \end{cases} \tag{67}$$

The evolution speed function $d_{p_3}(s)$ in Equation (67) is inspired by evolution speed functions $d_{p_2(s)}$ of DRLSE model in Equation (35), $d_{p_w(s)}$ of FCM model in Equation (48). In order to visualize these complex evolution speed functions and explain the differences among them, all three evolution speed functions $d_{p_2(s)}$, $d_{p_w(s)}$, $d_{p_3(s)}$ are plotted in Figure 2. In this figure, the evolution speed functions $d_{p_2(s)}$, $d_{p_w(s)}$ achieve the maximum value of 1 at zero well potential ($|\nabla\phi| = 0$), which causes the evolution speed of $d_{p_2(s)}$, $d_{p_w(s)}$ to be too fast, and the evolution curve may invade inside target interior. On the contrary, the evolution speed function $d_{p_3(s)}$ obtains the minimum value of -1, which decelerates evolution speed to achieve stable evolution and avoid wrong segmentation. In addition, the evolution speed function $d_{p_3(s)}$ has the steepest slope at one well potential ($|\nabla\phi| = 1$) among all three evolution speed functions $d_{p_2(s)}$, $d_{p_w(s)}$, $d_{p_3(s)}$, which means the evolution speed function $d_{p_3(s)}$ facilitates the convergence process in a faster speed than $d_{p_2(s)}$, $d_{p_w(s)}$ as well as enables the distance regularization term more sensitive.

Applying the steepest descent approach to minimize the energy function in Equation (60), which achieves the following gradient descent flow function as

$$\frac{\partial\phi}{\partial t} = -\frac{\partial E^{PFE}(\phi)}{\partial\phi} = n_1 \operatorname{div}\left(d_{p_3}(|\nabla\phi|)\nabla\phi\right) + n_2\delta_\varepsilon(\phi)\operatorname{div}\left(g_m\frac{\nabla\phi}{|\nabla\phi|}\right) + \varphi_1\left(I_\sigma, c_l, c_s\right) g_m\delta_\varepsilon(\phi). \tag{68}$$

In Equation (68), the gradient descent flow function consists of three parts. The first part denotes internal energy ($n_1 \operatorname{div}\left(d_{p_3}(|\nabla\phi|)\nabla\phi\right)$) on the basis of distance regularized term, which is used to minimize the difference between level set function and sign distance function to solve the issue of repeated re-initialization during iteration process. The second part signifies external energy on the foundation of length constraint part ($n_2\delta_\varepsilon(\phi)\operatorname{div}\left(g_m\frac{\nabla\phi}{|\nabla\phi|}\right)$) and area part ($\varphi_1\left(I_\sigma, c_l, c_s\right) g_m\delta_\varepsilon(\phi)$). Specifically, the length constraint part guides the contour line to evolve towards the target boundary due to the effect of adaptive edge indicator function as well and adjusts the length of contour line. The area part controls the velocity of contour line with respect to image gray-scale information due to the effect of adaptive sign function.

PFE model combines energy function based on median pre-fitting functions with adaptive functions, which realizes and accelerates the bidirectional evolution of contour line and reduces the probability of edge leakage. In addition, this model is able to effectively handle images with uneven intensity. However, the issues of falling into false boundary boundaries and insufficient segmentation at the boundary edge may sometimes happen while segmenting images with a large target.

4. EXPERIMENTAL RESULTS

Different kinds of ACMs have been reviewed in Section 3 and Section 4, and 12 representative of those models (BC[32], RSF[34], LIF[35], LPF[48], RSF&LoG[39], OLPFI[70], PBC[57], LPF&FCM[41], LGJD[58], ABC[56], PFE[47], APFJD[49]) are selected to conduct comparison experiment to segment various images including synthetic images, medical images, and natural images and compare their segmentation results (The CPU running

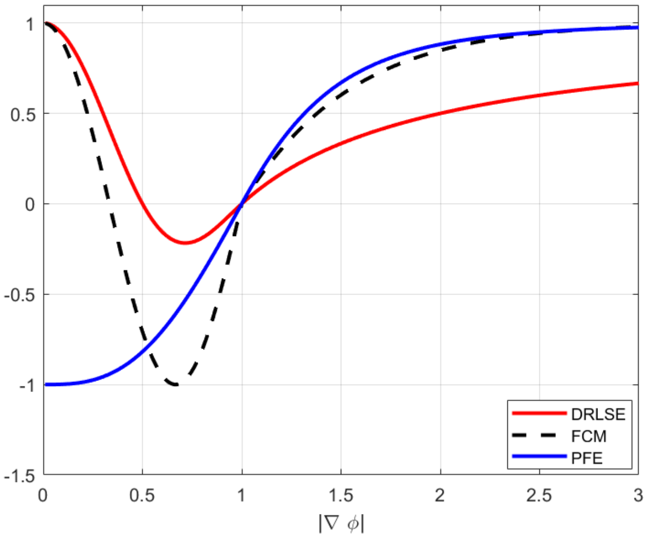


Figure 2. Contrast of DRLSE, FCM, PFE models on evolution speed equations $d_{p_2(s)}$, $d_{p_w(s)}$, $d_{p_3(s)}$.

time T, iteration number N, IOU, and DSC). All the models wereas programmed utilizing MATLAB 2021a and run ran on an AMD Ryzen7 5800H 3.2GHz CPU, 16G RAM, a NVIDIA GeForce RTX 3060 6G GPU, and 64 bit Windows 11 operating system. To explain the time consumed by convolution operation, using APFJD model on image (a) in Figure 4, the CPU running time T is 1.528 seconds, and the computation of convolutions accounts for 70% of that time. The relevant codes are available on the website https://github.com/sdjswgr.

Common evaluation criteria for assessing different segmentation approaches are segmentation time and segmentation quality. The authors evaluate segmentation time through the CPU running time T and iteration number N. The smaller their values, the less segmentation time and better segmentation efficiency will be. In addition, the segmentation quality of segmented image is measured through Intersection over union (IOU), which is described as

$$\mathrm{IOU} = \frac{A_1 \cap A_1^G}{A_1 \cup A_1^G}. \tag{69}$$

Note that A_1 signifies the foreground region of the segmented image while A_1^G denotes the foreground region of the ground-truth image. The IOU value is used to measure the similarity between the foreground region of segmented image and ground-truth image to evaluate the segmentation quality. The range of this coefficient is bounded in $[0, 1]$, and the closer it is to 1, the better segmentation quality it is.

4.1. Dataset characteristic

All images utilized in this paper are downloaded from a public open source image library called Berkeley segmentation data set and Benchmarks 500 (BSDS500), which can be reached on the website https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/ for more details. Specifically, for medical images (a-h) in Figure 4, images (a-b) are blood capillaries, image (c) is CT of bone, images (d) is bacteria embryo, image (e) is kidney, image (f) is the internal structure of the brain, and images (g-h) are B-ultrasound of uterus. For natural images (a-h) in Figure 5, image (a) is a piece of maple leaf, image (b) is a shell, image (c) is a starfish, image (d) is a polar bear, image (e) is a bradypod, image (f) is a stone bench, image (g) is plane, and image (h) is an eagle.

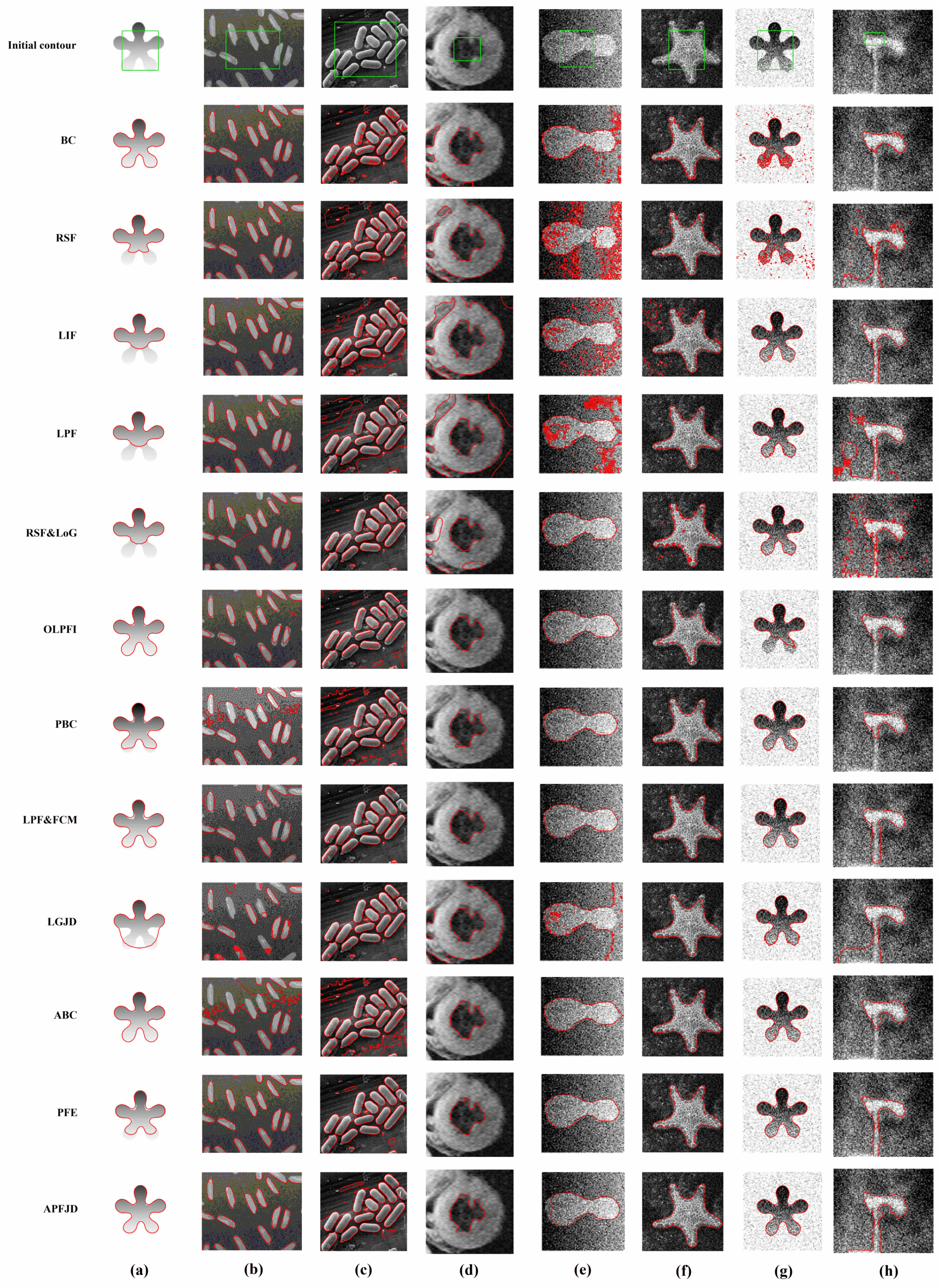


Figure 3. The segmentation results of the first comparative experiment to segment synthetic images. The 1st row represents initial contours, the 2nd to 12th rows denote segmentation results of BC [32], RSF [34], LIF [35], LPF [48], RSF&LoG [39], OLPFI [70], PBC [57], LPF&FCM [41], LGJD [58], ABC [56], PFE [47], and APFJD [49], respectively.

Table 1. Numerical analysis of segmentation results (The CPU running time T, iteration number N, and IOU) of the first comparative experiment in images a-h in Figure 3.

	Image(a)(100 × 100)	**Image(b)**(132 × 103)	**Image(c)**(256 × 233)	**Image(d)**(136 × 132)	**Image(e)**(103 × 97)	**Image(f)**(214 × 209)	**Image(g)**(100 × 100)	**Image(h)**(127 × 107)
BC	2.159/20/0.928	7.726/180/0.799	10.288/200/0.711	1.015/30/0.552	9.912/200/0.836	7.445/180/0.914	7.831/180/0.820	18.086/300/0.581
RSF	2.875/300/0.718	5.928/300/0.784	18.380/300/0.309	12.776/220/0.207	23.527/500/0.256	15.856/300/0.929	14.628/300/0.832	19.251/300/0.499
LIF	1.468/200/0.917	**2.658/150/0.835**	13.895/200/0.408	5.315/180/0.211	15.563/200/0.547	10.814/200/0.729	10.412/200/0.941	1.047/120/0.688
LPF	5.221/300/0.662	5.117/300/0.780	10.429/500/0.433	4.585/220/0.267	5.606/300/0.553	23.248/500/0.927	**3.148/150/0.945**	1.425/120/0.431
RSF&LoG	5.751/200/0.674	6.853/200/0.683	10.635/300/0.701	4.856/180/0.554	7.963/200/0.944	20.835/300/0.879	4.258/100/0.949	1.125/100/0.503
OLPFI	0.961/60/0.945	4.754/200/0.727	8.758/200/0.394	1.021/60/0.880	5.761/200/0.945	8.635/200/0.758	7.468/180/0.714	1.249/65/0.604
PBC	6.142/300/0.882	5.856/300/0.587	8.967/300/0.593	1.165/90/0.796	4.821/200/0.951	8.617/300/0.935	5.804/200/0.918	0.346/65/0.571
LPF&FCM	3.494/300/0.876	5.108/300/0.661	7.752/300/0.717	0.543/85/0.875	3.264/280/0.949	9.822/300/0.920/0.958	3.365/200/0.913/0.954	0.449/60/0.731
LGJD	3.952/200/0.725	4.585/200/0.515	5.856/250/0.874	3.658/200/0.230	8.635/300/0.611	7.423/300/0.894	10.528/300/0.939	1.437/120/0.474
ABC	**0.641/20/0.958**	6.589/300/0.415	15.254/300/0.576	0.196/20/0.855	8.111/300/0.931	**3.856/150/0.941**	5.964/250/0.934	7.215/280/0.575
PFE	2.964/150/0.803	3.589/180/0.834	**5.545/200/0.879**	**0.248/90/0.891**	**3.132/180/0.954**	12.826/600/0.902	3.792/180/0.926	**0.237/90/0.812**
APFJD	0.855/65/0.917	1.856/100/0.795	6.982/250/0.820	0.285/65/0.838	3.915/200/0.936	4.570/200/0.878	3.253/180/0.924	0.958/100/0.610

4.2. Segmentation experiment of synthetic images

Intensity non-uniformity and noise interference often occur in image segmentation. In Figure 3, the segmentation results of the 12 ACMs on synthetic images (a-h) are described in Figure 3. The former 4 images (a-d) represent images with intensity non-uniformity, while the latter 4 images (e-h) symbolize images with noise interference. The associated segmentation quality (IOU) and segmentation time (the CPU running time T and iteration number N) are concluded in Table 1. From this table, for images with intensity non-uniformity (a-d), ABC, LIF, and PFE models respectively obtain better segmentation results than other models respectively. In addition, for images with noise interference (e-h), PFE, ABC, LPF achieve the best segmentation results respectively. Particularly, PFE model achieves the most stable segmentation results for all images with noise interference (a-h) in Figure 3. In fact, PFE model takes advantage of novel pre-fitting functions to quickly approximate the background and foreground of the input image ahead of iteration process, which improves the stability of segmenting noisy images and saves computation costs.

Table 2. Numerical analysis of segmentation results (The CPU running time T, iteration number N, and IOU) of the second comparative experiment in images a-h in Figure 4.

	Image(a)(111 × 110)	**Image(b)**(103 × 131)	**Image(c)**(112 × 224)	**Image(d)**(152 × 128)	**Image(e)**(124 × 66)	**Image(f)**(119 × 78)	**Image(g)**(200 × 227)	**Image(h)**(95 × 93)
BC	5.725/380/0.733	2.407/180/0.766	2.958/250/0.815	4.176/280/0.899	5.285/300/0.835	15.992/150/0.607	1.437/30/0.889	2.564/200/0.633
RSF	2.458/300/0.844	2.152/140/0.825	1.886/120/0.852	4.537/200/0.898	4.852/200/0.756	23.974/200/0.622	14.733/220/0.207	10.511/220/0.433
LIF	1.525/130/0.851	1.234/100/0.826	1.458/100/0.868	3.172/150/0.909	2.912/130/0.723	15.254/150/0.675	2.305/120/0.714	2.859/100/0.572
LPF	1.621/140/0.862	1.575/100/0.802	**2.245/150/0.932**	5.109/180/0.916	1.852/120/0.781	1.653/120/0.635	0.995/100/0.442	1.595/120/0.447
RSF&LoG	1.062/100/0.870	1.250/80/0.821	**5.582/150/0.934**	9.926/200/0.829	2.952/135/0.779	16.582/200/0.718	8.289/200/0.849	1.252/120/0.601
OLPFI	**0.805/80/0.875**	**0.905/85/0.837**	**1.584/90/0.935**	0.925/85/0.925	0.848/95/0.850	1.465/120/0.604	2.048/150/0.835	**0.652/65/0.918**
PBC	1.689/120/0.866	1.612/100/0.831	4.773/250/0.926	3.593/180/0.931	3.525/150/0.845	**1.081/100/0.784**	1.509/100/0.867	1.653/100/0.906
LPF&FCM	0.952/100/0.861	**0.896/85/0.838**	2.545/150/0.896	3.862/180/0.905	2.465/150/0.893	1.868/120/0.653	7.269/200/0.864	1.058/100/0.889
LGJD	3.759/280/0.796	2.225/150/0.815	**2.587/100/0.932**	2.726/200/0.910	3.582/220/0.724	3.341/280/0.689	2.758/180/0.204	3.582/200/0.513
ABC	1.259/100/0.885	**1.036/95/0.838**	**2.848/120/0.930**	2.033/85/0.930	**0.629/35/0.928**	3.258/120/0.692	**0.821/85/0.919**	1.275/95/0.898
PFE	1.028/100/0.869	**0.855/80/0.841**	**2.257/110/0.933**	**0.846/90/0.942**	0.911/90/0.902	2.586/200/0.573	0.986/100/0.898	0.856/95/0.912
APFJD	1.528/105/0.728	1.043/95/0.815	1.585/110/0.899	1.852/120/0.851	**0.506/30/0.926**	1.962/120/0.662	1.399/100/0.889	1.124/100/0.894

4.3. Segmentation experiment of medical images

ACMs are extensively applied to process medical images to find out the location of the lesion. Consequently, the 12 ACMs are utilized to segment 6 medical images (a-h) in Figure 4, and the associated segmentation quality (IOU) and segmentation time (the CPU running time T and iteration number N) are described in Table 2. From this table, for image (a), OLPFI has the best performance. For image (b), OLPFI, LPF&FCM, ABC, PFE

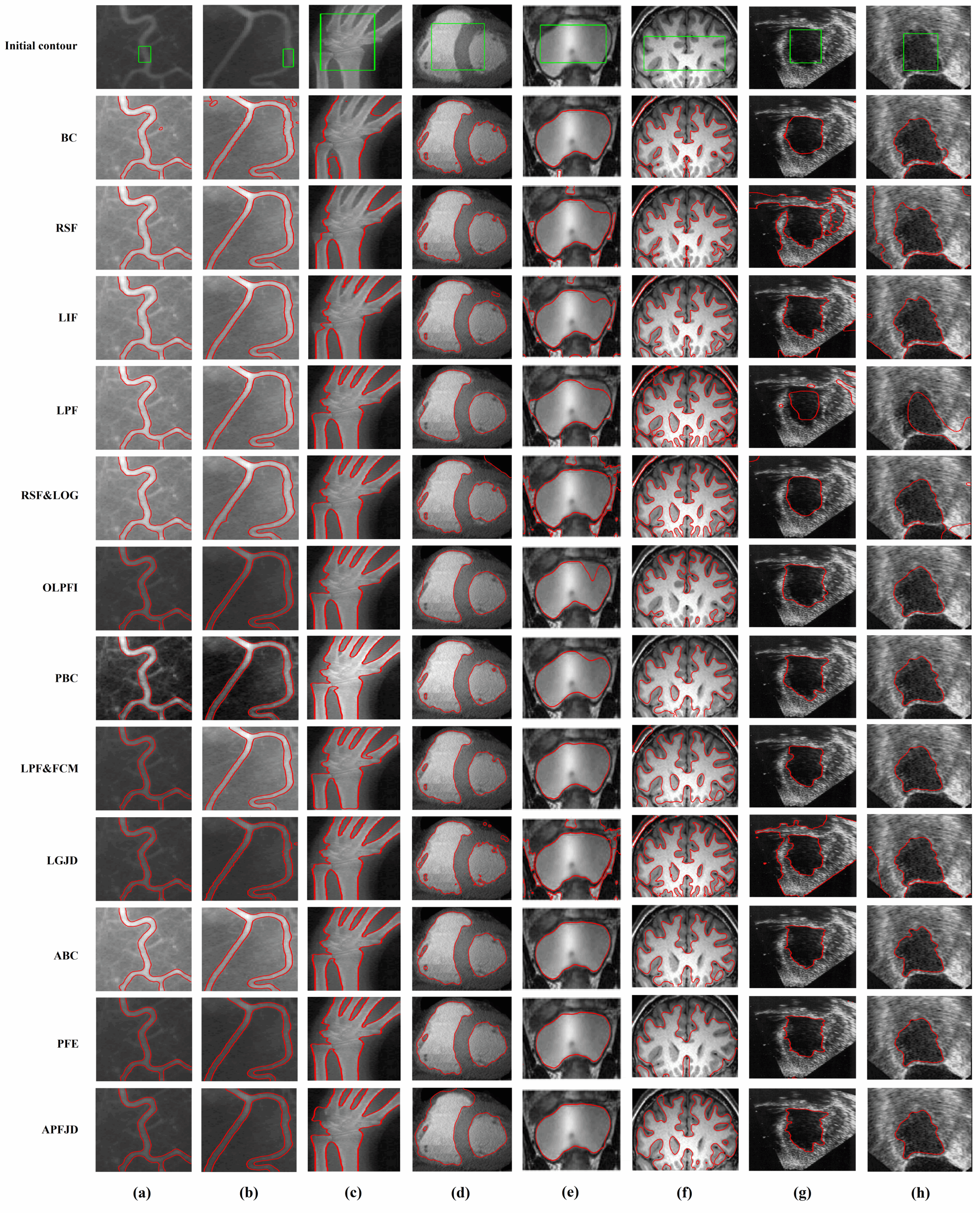


Figure 4. The segmentation results of the second comparative experiment to segment medical images. The 1st row represents initial contours, the 2nd to 12th rows denote segmentation results of BC [32], RSF [34], LIF [35], LPF [48], RSF&LoG [39], OLPFI [70], PBC [57], LPF&FCM [41], LGJD [58], ABC [56], PFE [47], and APFJD [49], respectively.

models have very similar segmentation results. However, PFE model is ranked first in terms of the least CPU running time T and iteration number N and the largest IOU value. For image (c), LPF, RSF&LOG, OLPFI, LGJD, ABC, PFE models have similar segmentation results. Nevertheless, OLPFI has the best performance with a small advantage. For image (d), the performance of PFE model is ranked first in terms of all evaluation criteria. For image (e), the IOU value of ABC model are is the biggest, while the CPU running time T and iteration number N of APFJD model is are the least. For image (f), PBC obtains the best segmentation results. For image (g), ABC model obtains the best performance in terms of all evaluation criteria. For image (h), OLPFI model has the best segmentation results in terms of all evaluation criteria. Particularly, ABC model

Table 3. Numerical results of segmentation outcomes (The CPU running time T, iteration number N, and IOU) of the third comparative experiment in images a-h in Figure 5.

	Image(a) (300×203)	**Image(b)** (300×225)	**Image(c)** (481×321)	**Image(d)** (481×321)	**Image(e)** (481×321)	**Image(f)** (481×321)	**Image(g)** (481×321)	**Image(h)** (481×321)
BC	1.221/10/0.950	2.967/50/0.734	5.775/120/0.602	2.315/20/0.161	2.940/25/0.231	7.567/150/0.162	7.491/150/0.289	8.722/150/0.837
RSF	1.043/220/0.952	14.658/250/0.709	15.589/250/0.255	**8.152/180/0.926**	7.254/165/0.543	8.952/180/0.582	18.892/380/0.637	10.866/150/0.826
LIF	1.716/150/0.964	2.536/200/0.879	7.895/180/0.405	9.255/500/0.657	5.752/150/0.592	6.895/150/0.841	7.125/180/0.775	6.525/150/0.870
LPF	**1.115/90/0.969**	1.531/100/0.841	9.528/300/0.617	**10.588/380/0.922**	12.592/300/0.377	15.281/300/0.581	4.338/120/0.543	5.450/120/0.781
RSF&LoG	5.741/100/0.893	6.952/120/0.941	8.258/180/0.524	7.528/120/0.840	7.592/150/0.587	10.896/180/0.844	6.882/120/0.505	3.317/95/0.817
OLPFI	**0.506/40/0.955**	**0.731/65/0.951**	**1.638/95/0.889**	**0.546/40/0.927**	**1.867/95/0.794**	**1.105/85/0.794**	1.983/100/0.757	1.513/100/0.753
PBC	0.696/85/0.954	1.984/95/0.929	7.595/200/0.824	**1.856/95/0.924**	8.215/200/0.791	3.148/120/0.823	**4.768/150/0.904**	1.233/95/0.845
LPF&FCM	2.524/120/0.958	3.158/150/0.889	13.752/300/0.515	7.181/180/0.871	10.537/200/0.858	9.905/200/0.863	9.389/200/0.801	7.851/180/0.831
LGJD	0.715/95/0.886	1.985/100/0.919	5.589/180/0.563	2.755/120/0.135	11.762/300/0.791	3.556/120/0.815	2.789/120/0.876	2.511/120/0.873
ABC	0.785/80/0.951	1.259/95/0.954	9.785/200/0.507	1.895/100/0.903	7.892/150/0.847	**2.048/100/0.886**	**2.638/100/0.902**	2.032/100/0.836
PFE	0.748/65/0.937	2.468/120/0.638	2.685/150/0.531	**3.522/150/0.925**	**4.896/180/0.870**	8.896/200/0.834	9.541/230/0.816	3.522/120/0.877
APFJD	0.591/30/0.971	**1.167/100/0.966**	**2.592/120/0.927**	1.047/100/0.924	**1.972/100/0.869**	**2.925/120/0.913**	**1.045/95/0.906**	**1.161/100/0.886**

obtains the most stable segmentation results for all images (a-h) in Figure 4. Actually, ABC model utilizes a novel regularization function to normalize the energy range of data driven term, which enables it to effectively process medical images with intensity non-uniformity.

4.4. Segmentation experiment of natural images

The 12 ACMs are applied to segment natural images (a-h) in Figure 5, and the associated segmentation quality (IOU) and segmentation time (the CPU running time T and iteration number N) are described in Table 3.

In Table 3, for image (a), the IOU value of LPF model is the largest while the CPU running time T and iteration number N of OLPFI model are the least. For image (b), OLPFI model is ranked first in terms of the CPU running time T and iteration number N, and APFJD model is ranked first with respect to the IOU value. For image (c), OLPFI model has the best performance in terms of the CPU running time T and iteration number N, while APFJD model has the best performance with respect to the IOU value. For image (d), RSF, LPF, OLPFI, PBC, PFE, and APFJD models achieve similar segmentation results. However, OLPFI model is ranked first with respect to all evaluation criteria. For image (e), the IOU value of the PFE model are is the biggest, while the CPU running time T and iteration number N of OLPFI model are the least. For image (f), the IOU value of APFJD model is the biggest, while the CPU running time T and iteration number N of OLPFI model are the least. For image (g), although the IOU value of PBC, ABC, and APFJD models are similar, APFJD model has the best segmentation results with the biggest IOU value and lowest CPU running time T and iteration number N. For image (h), APFJD model has the best segmentation results in terms of all evaluation criteria. On average, APFJD model acquires the most stable segmentation results for all natural images (a-h) in Figure 5. In fact, APFJD model employs an adaptive regularization function to normalize the ranges of the level set function and data driven term, which renders it to efficiently process natural images with complex background information.

4.5. Comparison experiments with Deep learning-based algorithms

To compare the segmentation results between ACMs and deep learning-based algorithms, DeepLabv3+[71] and Mask R-CNN algorithms[72] are selected to segment 6 images (a-f) in Figure 6. Note that DeepLabv3+ and Mask R-CNN algorithms are capable of recognizing all pixels subordinated to the target and painting the target

[0] Bias correction (BC), Region scalable fitting (RSF), Local image fitting (LIF), Local pre-fitting (LPF), Region scalable fitting and optimized Laplacian of Gaussian (RSF&LoG), Optimized local pre-fitting image (OLPFI), Pre-fitting bias field (PBC), Local pre-fitting and fuzzy c-means (LPF&FCM), Local and global Jeffreys divergence (LGJD), Additive bias correction (ABC), Pre-fitting energy (PFE), and Adaptive pre-fitting function and Jeffreys divergence (APFJD).

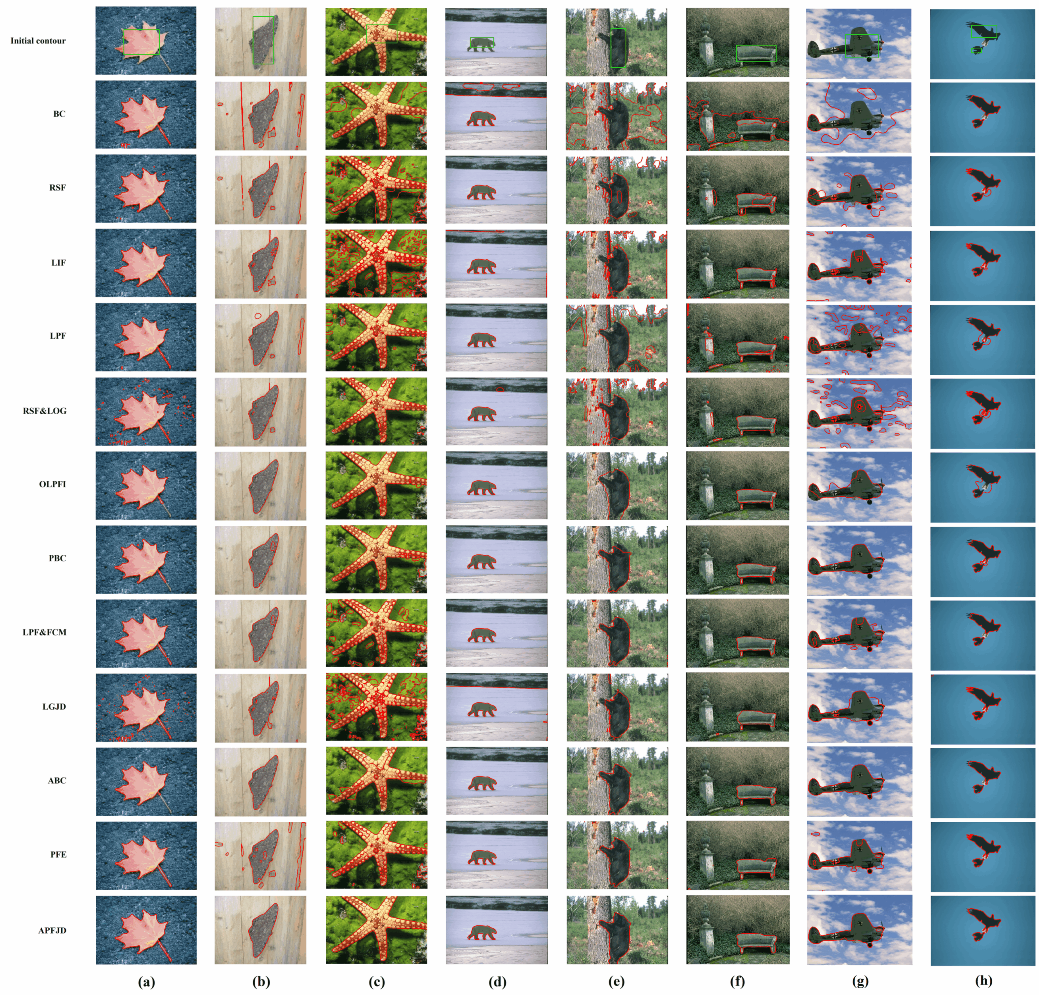


Figure 5. The segmentation results of the third comparative experiment to segment synthetic images. The 1st row represents initial contours, the 2nd to 12th rows denote segmentation results of BC[32], RSF[34], LIF[35], LPF[48], RSF&LoG[39], OLPFI[70], PBC[57], LPF&FCM[41], LGJD[58], ABC[56], PFE[47], and APFJD[49], respectively.

into multiple random colors to demonstrate the final segmentation result. For the pre-training stage of these two deep learning-based algorithms, the DeepLabv3+ algorithm have has used PASCAL Visual Object Classes 2007 (VOC20007), which can be found on the website http://host.robots.ox.ac.uk/pascal/VOC/voc2007/. The entire duration of pre-training stage of DeepLabv3+ neural network lasts roughly 9 hours under the framework of PyTorch deep learning environment. In addition, the Mask R-CNN algorithm utilizes the (Common Objects in Context) COCO dataset to conduct the pre-training process, which can be downloaded from the website https://cocodataset.org/. The whole duration of pre-training stage of DeepLabv3+ neural network lasts roughly 18 hours under the framework of PyTorch deep learning environment. Once the pre-trainings are completed, the trained DeepLabv3+ and Mask R-CNN neural networks are utilized to segment and visualize 6 images (a-f) in Figure 6 and compute their associated IOUs, which is illustrated in Figure 6. Meanwhile, the authors select 3 ACMs (RSF[34], LGJD[58], and APFJD[49]) to segment and visualize 6 images (a-f) in Figure 6 and calculate their corresponding IOUs. The numerical segmentation results (IOUs) of the experiments are all listed in Table 4.

In Table 4, for image (a), the DeepLabv3+ algorithm obtains the biggest IOU value due to the most excellent segmentation result. For image (b), the Mask R-CNN algorithm, RSF model and APFJD achieve similar IOU

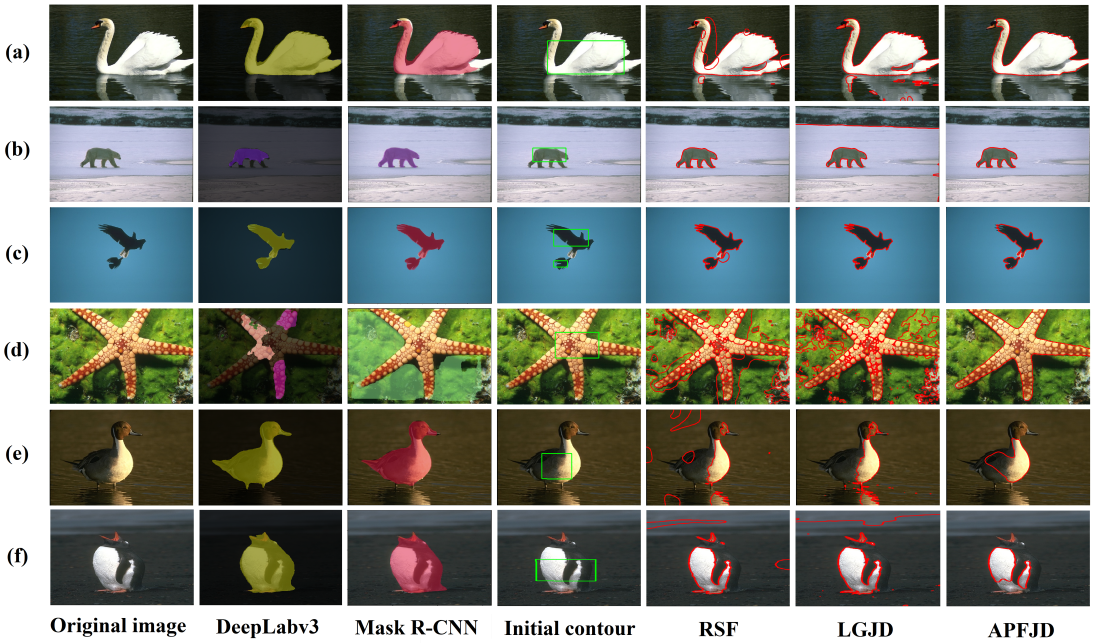


Figure 6. The segmentation results between DeepLabv3+ algorithm[71], Mask R-CNN algorithm[72], RSF[34] model, LGJD[58] model and APFJD[49] model. The 1st column represents original images, the 2nd to 3rd columns signify segmentation results of DeepLabv3+ algorithm and Mask R-CNN algorithm, respectively, the 4th column denotes initial contours of ACMs, and the 5th to 7th columns represent segmentation results of RSF model, LGJD model and APFJD model, respectively.

Table 4. Numerical analysis of IOUs between DeepLabv3 algorithm, Mask R-CNN algorithm, RSF model, LGJD model, APFJD model in images (a-f) in Figure 6.

	DeepLabv3+	Mask R-CNN	RSF	LGJD	APFJD
Image a(481×321)	0.935	0.807	0.681	0.853	0.920
Image b(481×321)	0.655	0.922	0.925	0.135	0.930
Image c(481×321)	0.875	0.883	0.826	0.873	0.886
Image d(481×321)	0.215	0.015	0.255	0.563	0.927
Image e(481×321)	**0.911**	**0.895**	0.328	0.321	0.468
Image f(321×481)	**0.905**	**0.915**	0.479	0.353	0.385

values. For image(c), the RSF model obtains the smallest IOU value due to the issue of edge leakage, and similar results are achieved by the remaining models. For image (d), the DeepLabv3+ and Mask R-CNN algorithms acquire very unsatisfactory IOU values due to failed segmentation. On the contrary, the APFJD model attains the largest IOU value because of clear and clean segmentation. For image (e), the DeepLabv3+ and Mask R-CNN algorithms demonstrate the advantages of segmentation of multi-phase images, which obtains much bigger IOU values than 3 ACMs (RSF, LGJD, APFJD models). The DeepLabv3+ algorithm attains the largest IOU value due to a more fully segmented target. For image (f), the Mask R-CNN algorithm obtains the best segmentation result in terms of the biggest IOU value.

4.6. Summary

Since the images with unevenly distributed intensity, the area inside and outside evolution curves are not intensity uniform. The calculated intensity averages are incapable tocannot represent intensity distribution. Therefore, BC model estimates a bias field to process images with unevenly distributed intensity, which works well with images (a-b) in Figure 3. However, common issues such as falling into false boundary boundaries may occur in images (c-d) in Figure 3. In addition, this model cannot effectively process images with noise interference such as images (e, g, h) in Figure 3. BesidesAdditionally, under- segmentation may take place as image (h) in Figure 3 and images (c, g) in Figure 4 shows. In addition, BC model leaks from target boundary when it segments natural images, as the second row in the Figure 5 indicates.

RSF model is capable of segmenting images with uneven intensity as image (b) in Figure 3. However, the incorporated kernel function only computes the gray values of image local image regions, which makes it easy to fall into a local minimum during the process of energy minimization such as images (a, c) in Figure 3. Nevertheless, falling into a false area still remains unsolved as images (c, d, g) in Figure 3. In addition, the segmentation time is long due to at least four convolution operations to update fitting functions during each iteration. BesidesIn addition, this model has poor anti-noise ability, which is vulnerable to the influence of noise interference, as images (e, g, h) illustrate. Moreover, the issues of under- segmentation and leaking from weak edge still exists, as shown in images (c, e) in Figure 4 and images (g, h) in Figure 4, respectively. Lastly, this model obtains poor segmentation results when segmenting natural images, as shown in the third row in the Figure 5.

Compared with RSF model, LIF model only utilizes two convolution operations to update fitting functions, which greatly reduces the CPU running time T and iteration number N according to Tables 1, 2, and 3. The Gaussian kernel function is also used in this model, so common issues such as boundary leakage and falling into local minimum also occur in this model in some cases (as illustrated in images (a, d, e) in Figure 3). However, this model is still sensitive to noise interference, as shown in images (e, f, h) in Figure 3. BesidesAdditionally, the problem of under- segmentation has been improved to some degree as shown in image (c) in Figure 4. In additionFurthermore, this model has very poor segmentation results while segmenting natural images as indicated in images (b, c, e, g) in Figure 5.

LPF model locally computes average image intensity ahead of iteration process, which reduces the computation cost to some degree. However, the Gaussian kernel function is also used in this model to update the level set function, which results in falling into false boundary boundaries (as described in image (a) in Figure 3) and edge leakage (as illustrated in images (b, c, d) in Figure 3). In addition, this model has further improved in terms of anti-noise ability as shown in images (f, g) in Figure 3. However, the issue of boundary leakage still exists in images (e, h) in Figure 3 and images (g, h) in Figure 4 and images (e, f, g, h) in Figure 5. Besides, the problem of trapping into false boundary boundaries still occurs in images (b, c) in Figure 5.

RSF&LoG model combines RSF model and Laplacian of Gaussian (LoG) energy to smooth the homogeneous areas and enhance boundary characteristics simultaneously to segment images with uneven intensity, which can segment images with uneven intensity to some extent. However, this model may create some common issues such as under- segmentation and falling into false boundary boundaries in images (a, b, c) in Figure 3 and images (d, h) in Figure 3, respectively. In addition, boundary leakage may occur in some cases (as shown in image (h) in Figure 4 and images (c, e) in Figure 5).

OLPFI model calculates the mean intensity of the selected local regions before iteration starts, which dramatically decreases segmentation time. This model puts region-based attributes and edge-based attributes together to handle images with intensity non-uniformity, which obtains excellent results as shown in image (a, d) in Figure 3. However, under- segmentation often occurs in images (b, c, f, g, h) in Figure 3, image (e) in Figure 4, and images (a, g, h) in Figure 5. this This model has relatively poor anti-noise ability in the form of under-segmentation as indicated in images (f, g, h) in Figure 3. Lastly, this model greatly reduces the possibility of boundary leakage and falling into fake false boundaryboundaries.

PBC model utilizes the optimized FCM algorithm to estimate the bias field before iteration process, which gets rid of time-consuming convolution operation during each iteration and greatly reduces segmentation time. In addition, this model can segment images with uneven intensity in images (a, d) in Figure 3. However, boundary leakage may occur in some cases (as indicated in images (b, c) in Figure 3). Moreover, this model is capable of effectively segmenting images with noise interference with high segmentation quality. Nevertheless, under-segmentation may occur in some cases (as indicated in images (h) in Figure 3, image (e) in Figure 4 and images (a, e) in Figure 5). Lastly, falling into fake false boundary boundaries may take place in some cases (as shown in images (b, c) in Figure 5).

LPF&FCM model employs the FCM algorithm and adaptive sign function to solve the issue of boundary leakage, which obtains outstanding performance to segment images with intensity non-uniformity in images (a, d) in Figure 3. However, issues such as under- segmentation and falling into local minima may occur in some cases (as shown in images (b, c) in Figure 3 and images (b, c, g) in Figure 5, respectively). In addition, this model has strong robustness to images with noise (as indicated in images (e-h) in Figure 3). Lastly, this model is capable of effectively segmenting medical images (a-h) in Figure 4 with high accuracy.

LGJD model utilizes the changeable weights to control the local and global data fitting energies based on Jeffreys divergence (JD), which is capable of segmenting images with intensity non-uniformity to some degree. However, this model also has common issues such as over- segmentation or under- segmentation in some cases (as shown in images (a, b, c) in Figure 3 and images (e, f, g) in Figure 5). In addition, boundary leakage may occur during the process of segmenting images with noise interference in some cases (as illustrated in image (e, h) in Figure 3). Besides, the issue of strapping into false boundary boundaries frequently takes place in some cases (as shown in images (d) in Figure 3, images (g, h) in Figure 4, and images (a-d) in Figure 5).

ABC model applies the theory of bias field to segment images with unevenly distributed intensity, which obtains excellent segmentation performance in terms of handling images with intensity non-uniformity in images (a, d) in Figure 3. However, this model has issues of leaking from weak boundary boundaries in some cases (as illustrated in images (b, c) in Figure 3). In addition, this model can effectively handle images with noise interference due to the effect of additive bias correction as shown in images (e, f, g) in Figure 3. Nevertheless, the problem of under- segmentation may happen in image (h) in Figure 4 and images (a, c) in Figure 5. Lastly, this model is also able to effectively segment medical images (a-h) in Figure 4 with high precision.

PFE model computes the median intensity of the chosen local areas ahead of iteration process, which greatly reduces computation cost. According to the twelfth row of Figure 3, this model is able to deal with images with uneven intensity and has excellent noise resistivity. However, common issues such as falling into fake false boundary boundaries and boundary leakage may take place in some cases (as indicated in image (c) in Figure 3 and images (b, c, g) in Figure 5). Lastly, this model is also competent to effectively segment medical images (a-h) in Figure 4 with high efficiency.

APFJD model computes average intensity of selected areas before iteration takes place, which dramatically decreases segmentation time. This model can effectively segment images with uneven intensity and noise interference due to the effect of JD, as shown in images (a-b, d) and images (e-f) in Figure 3. However, the issue of strapping into false boundary boundaries may happen as indicated in image (c) in Figure 3). In addition, under- segmentation may occur in some cases (as indicated in image (e) in Figure 3 and image (c, f) in Figure 4). Lastly, this model segments natural images (a-h) in Figure 5 with excellent accuracy.

To conclude the characteristic of above said ACMs, the calculation processes of BC, RSF, LIF, LPF, RSF&LoG, and LGJD models are too complex to be implemented in practice, which have poor anti-noise capability and spend a huge amount of time for curve evolution. In addition, the computation processes to compute pre-fitting functions in OLPFI, PBC, PFE, and APFJD models are optimized, which enables them to quickly segment different kinds of images within a short amount of time. LPF&FCM takes advantages of FCM clustering to divide the input image region into background region and foreground region before iteration begins, which greatly reduces the computational overhead. ABC model implements K-means ++ clustering to separate input image area into background and foreground regions before iteration starts, which also hugely decreases the computational expense. Nevertheless, LPF&FCM and ABC models may generate unexpected segmentation outcomes such as redundant curves, if the FCM and K-means ++ clustering algorithms have bad performance.

Although ACMs can effectively segment double-phase images with fair segmentation results, the majority of existing ACMs are not able to segment multi-phase images as indicated in images (e,f) in Figure 6. Note that double-phase image means a target in an image either contains black pixels or white pixels, while multi-phase image means a target in an images contains black and white pixels at the same time as shown in images (e,f) in Figure 6. According to Figure 6, the deep learning-based algorithms (DeepLabv3+ and Mask R-CNN algorithms) exhibit an advantage on in segmenting multi-phase images. Specifically, ACMs (RSF, LGJD, APFJD models) can barely segment multi-phase images (e,f) in in Figure 6, while DeepLabv3+ and Mask R-CNN algorithms obtain excellent segmentation results in terms of much bigger IOU value. However, failed segmentation in the form of under- segmentation may take placeoccur while implementing DeepLabv3+ and Mask R-CNN algorithms to segment double-phase images as illustrated in images (b,d) in Figure 6.

5. CHALLENGES AND PROMISING FUTURE DIRECTIONS

Nowadays, there are still various common issues waiting for solutions in the field of image segmentation in practice. The review of the above ACMs points out some common issues and concludes some promising future directions. It is believed that this discussion will be useful for later researchers in this field to design more advanced models.

5.1. The combination of deep learning models

Inspired by the general idea of ACMs, the work[73–75] incorporates region and length constraint terms into the cost or loss function of convolutional neural network (CNN) model based on traditional Dense U-Net for image segmentation, which combines geometric attributes such as edge length with region information to achieves better segmentation accuracy. In addition, compared with traditional ACMs requiring iterations to solve PDEs, the employment of CNN greatly reduces computation cost in image segmentation, although its training process is generally long. In addition, later researchers embed some loss functions in deep learning[76–81] in region-based level set energy functions to improve segmentation efficiency and accuracy. Therefore, one can put the energy function of diverse ACMs mentioned in this paper and other segmentation models in deep learning together to design some new hybrid energy functions to further improve segmentation performance, which is recommended as a promising future research direction in the area of image segmentation.

5.2. The combination of edge-based and region-based ACMs

As discussed in this paper, active contour method can be grouped into two types: the region-based ACM and the edge-based ACM. The region-based ACM utilizes a pre-defined region descriptor or a contour representation to recognize each region on the image, while the edge-based ACM uses the differential property or gradient information of boundary points to construct a contour representation. The region-based ACMs generally utilize regional information (the pixel grayscale information) of the image to construct energy functions, which improves their system robustness and effectiveness. However, the region-based ACMs cannot deal with contours that do not evolve from region boundaries. The edge-based ACMs mainly use the gradient information of the target boundary points as the main driving force to guide the motion of the evolution curve, which is capable of handling topology changes adaptively. Nevertheless, the edge-based ACMs generally need to reinitialize the level set function periodically during the evolution process, which will affect the computational accuracy and slow down segmentation efficiency. In this case, the zero level set may not be able to move towards the target boundary, and how and when to initialize it still remains unsolved.

Therefore, it is necessary to combine the strengths of the region-based and region-based ACMs to obtain better segmentation outcomes. Recently, several hybrid ACMs[41,47,70,82,83] are constructed to take advantage of both metrics of the region-based and edge-based ACMs to achieve higher segmentation efficiency and lower computation cost. It is recommended that future researchers design more hybrid ACMs on the basis of the hybrid ones mentioned above.

5.3. Fast and stable optimization algorithm

The general optimization process of ACMs is to minimize the associated energy function through gradient or steepest descent method. However, it should be aware that it may be hard to figure out the global minima if the energy function is non-convex[33,84–87], which may cause a failed segmentation in the form of falling into a local minima. Specifically, the traditional gradient or steepest descent approach is initialized by the initial level set function and then descends at each iteration, ; the descending direction is controlled by the slope or the derivative of the evolution curve. It is possible to replace the traditional one with other gradient descent methods to design a brand-new series of ACMs, which is capable of optimizing the evolution curve and avoiding falling into local minima.

6. CONCLUSIONS

This paper has presented an overview of different kinds of ACMs in image segmentation in Sections 1, which helps readers to obtain a comprehensive understanding of different kinds of ACMs. Then, some fundamental theory of ACMs has been explained in Section 2. Specifically, region-based ACMs, edge-based ACMs, and hybrid ACMs are reviewed in Section 3. After that, three comparison experiments of 12 different ACMs in terms of several evaluation criteria (the CPU running time T, iteration number N, IOU and DSC) have been conducted to compare their segmentation performance on different kinds of images (synthetic images, medical images, and natural images) in Section 4. In addition, two deep learning-based algorithms (DeepLabv3+ and Mask R-CNN) have been implemented to segment double-phase images and multi-phase images, whose segmentation results in terms of IOU values have been compared with several ACMs to demonstrate their advantages and disadvantages. According to the experimental results of these comparison experiments, the hybrid ACMs appear to be more suitable for large- scale image segmentation applications due to higher segmentation efficiency and accuracy. In addition, the deep learning-based algorithms (DeepLabv3+ and Mask R-CNN) obtain much superior segmentation results than ACMs while segmenting multi-phase images. Lastly, some challenges and promising future research works in the field of image segmentation have been discussed in Section 5.

DECLARATIONS

Authors' contributions

Writing- Original draft preparation: Chen Y, Ge P
Writing-Reviewing and Editing: Wang G, Weng G
Conceptualization: Chen Y, Wang G, Chen H
Methodology: Chen Y, Wang G, Chen H
Project administration: Chen Y, Wang G
Recourses: Chen Y, Chen H
Supervision: Weng G, Chen H
Data curation: Chen Y, Ge P
Software: Ge P, Weng G
Investigation: Ge P

Visualization: Ge P

Availability of data and materials

Not applicable.

Financial support and sponsorship

This research paper was supported in part by National Natural Science Foundation of China under Grant 62103293, in part by Natural Science Foundation of Jiangsu Province under Grant BK20210709, in part by Suzhou Municipal Science and Technology Bureau under Grant SYG202138, and in part by Entrepreneurship and Innovation Plan of Jiangsu Province under Grant JSSCBS20210641.

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Copyright

REFERENCES

1. Son LH, Tuan TM. Dental segmentation from X-ray images using semi-supervised fuzzy clustering with spatial constraints. *Eng Appl of Art Int* 2017;59:186–95. DOI
2. Civit-Masot J, Luna-Perejón F, Corral JMR, et al. A study on the use of Edge TPUs for eye fundus image segmentation. *Eng Appl Art Int* 2021;104:104384. DOI
3. Akbari Y, Hassen H, Al-Maadeed S, Zughaier SM. COVID-19 lesion segmentation using lung CT scan images: comparative study based on active contour models. *Applied Sciences* 2021;11:8039. DOI
4. Guo Q, Wang L, Shen S. Multiple-channel local binary fitting model for medical image segmentation. *Chin J Electron* 2015;24:802-6. DOI
5. Zhang D, Li J, Li X, Du Z, Xiong L, Ye M. Local–global attentive adaptation for object detection. *Eng Appl Art Int* 2021;100:104208. DOI
6. Yang C, Wu L, Chen Y, Wang G, Weng G. An active contour model based on retinex and pre-Fitting reflectance for fast image segmentation. *Symmetry* 2022;14:2343. DOI
7. Chen H, Liu Z, Alippi C, Huang B, Liu D. Explainable intelligent fault diagnosis for nonlinear dynamic systems: from unsupervised to supervised learning. *IEEE Trans Neur Netw Lear Syst* 2022;Early Access. DOI
8. Ge P, Chen Y. An automatic detection approach for wearing safety helmets on construction site based on YOLOv5. In: 2022 IEEE 11th Data Driven Control and Learning Systems Conference (DDCLS). IEEE; 2022. pp. 140–45. DOI
9. Cao Y, Wang G, Yan D, Zhao Z. Two algorithms for the detection and tracking of moving vehicle targets in aerial infrared image sequences. *Remote Sensing* 2016;8:28. DOI
10. Wu S. A traffic motion object extraction algorithm. *Int J Bifurcation Chaos* 2015;25:1540039. DOI
11. Paragios N, Deriche R. Geodesic active contours and level sets for the detection and tracking of moving objects. *IEEE Trans Pattern Anal Machine Intell* 2000;22:266-80. DOI
12. Wu Z, Tian E, Chen H. Covert attack detection for LFC systems of electric vehicles: a dual time-varying coding method. *IEEE/ASME Trans Mechatron* 2022:1–11. DOI
13. Cootes TF, Edwards GJ, Taylor CJ. Active appearance models. *IEEE Trans Pattern Anal Machine Intell* 2001;23:681–85. DOI
14. Mille J. Narrow band region-based active contours and surfaces for 2D and 3D segmentation. *Compu Vis Image Und* 2009;113:946-65. DOI
15. Chan TF, Vese LA. Active contours without edges. *IEEE Trans Image Process* 2001;10:266-77. DOI
16. Tsai A, Yezzi A, Willsky AS. Curve evolution implementation of the Mumford-Shah functional for image segmentation, denoising, interpolation, and magnification. *IEEE Trans Image Process* 2001;10:1169-86. DOI

17. Wang G, Zhang F, Chen Y, Weng G, Chen H. An active contour model based on local pre-piecewise fitting bias corrections for fast and accurate segmentation. *IEEE Trans Instrum Meas* 2023;72:1-13 DOI
18. Xiang Y, Chung ACS, Ye J. An active contour model for image segmentation based on elastic interaction. *J Comput Phys* 2006;219:455–76. DOI
19. Huang AA, Abugharbieh R, Tam R. A Hybrid Geometric–Statistical Deformable Model for Automated 3-D Segmentation in Brain MRI. *IEEE Trans Biomed Eng* 2009;56:1838-48. DOI
20. Pluempitiwiriyawej C, Moura JMF, Wu YJL, Ho C. STACS: new active contour scheme for cardiac MR image segmentation. *IEEE Trans Med Imaging* 2005;24:593-603. DOI
21. Bowden A, Sirakov NM. Active contour directed by the poisson gradient vector field and edge tracking. *J Math Imaging Vis* 2021;63:665-80. DOI
22. Fahmi R, Jerebko A, Wolf M, Farag AA. Robust segmentation of tubular structures in medical images. In: Reinhardt JM, Pluim JPW, editors. SPIE Proceedings. SPIE; 2008. pp. 691443–1443-7. DOI
23. Zhang H, Morrow P, McClean S, Saetzler K. Coupling edge and region-based information for boundary finding in biomedical imagery. *Pattern Recogn* 2012;45:672-84. DOI
24. Wen J, Yan Z, Jiang J. Novel lattice Boltzmann method based on integrated edge and region information for medical image segmentation. *Biomed Mater Eng* 2014;24:1247-52. DOI
25. Lv H, Zhang Y, Wang R. Active contour model based on local absolute difference energy and fractional-order penalty term. *Appl Math Model* 2022;107:207-32. DOI
26. Mumford D, Shah J. Optimal approximations by piecewise smooth functions and associated variational problems. *Comm Pure Appl Math* 1989;42:577-685. DOI
27. Caselles V, Catte F, Coll T, Dibos F. A geometric model for active contours in image processing. *Numer Math* 1993;66:1-31. DOI
28. Caselles V, Kimmel R, Sapiro G. Geodesic active contours. *Int J Compu Vis* 1997;22:61–79. DOI
29. Bresson X, Esedoglu S, Vandergheynst P, Thiran JP, Osher S. Fast global minimization of the active contour/snake model. *Math Imaging Vis* 2007;28:151-67. DOI
30. Cohen LD, Kimmel R. Global minimum for active contour models: a minimal path approach. *Int J Compu Vis* 1997;24:57–78. DOI
31. Li C, Xu C, Gui C, Fox MD. istance regularized level set evolution and its application to image segmentation. *IEEE Trans Image Process* 2010;19:3243-54. DOI
32. Li C, Huang R, Ding Z, et al. A level set method for image segmentation in the presence of intensity inhomogeneities with application to MRI. *IEEE Trans Image Process* 2011;20:2007-16. DOI
33. Li C, Kao CY, Gore JC, Ding Z. Implicit active contours driven by local binary fitting energy. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition. IEEE; 2007. pp. 1–7. DOI
34. Li C, Kao CY, Gore JC, Ding Z. Minimization of region-scalable fitting energy for image segmentation. *IEEE Trans Image Process* 2008;17:1940-9. DOI
35. Zhang K, Song H, Zhang L. Active contours driven by local image fitting energy. *Pattern Recogn* 2010;43:1199–206. DOI
36. Chan TF, Esedoglu S, Nikolova M. Algorithms for finding global minimizers of image segmentation and denoising models. *SIAM J Appl Math* 2006;66:1632-48. DOI
37. Chambolle A, Cremers D, Pock T. A convex approach to minimal partitions. *SIAM J Imaging Sci* 2012;5:1113-58. DOI
38. Wang L, Li C, Sun Q, Xia D, Kao CY. Active contours driven by local and global intensity fitting energy with application to brain MR image segmentation. *Comput Med Imaging Graph* 2009;33:520-31. DOI
39. Ding K, Xiao L, Weng G. Active contours driven by region-scalable fitting and optimized Laplacian of Gaussian energy for image segmentation. *Signal Proce* 2017;134:224-33. DOI
40. Wang L, He L, Mishra A, Li C. Active contours driven by local Gaussian distribution fitting energy. *Signal Proce* 2009;89:2435-47. DOI
41. Jin R, Weng G. Active contours driven by adaptive functions and fuzzy c-means energy for fast image segmentation. *Signal Proce* 2019;163:1-10. DOI
42. Liu S, Peng Y. A local region-based Chan–Vese model for image segmentation. *Pattern Recogn* 2012;45:2769-79. DOI
43. Liu W, Shang Y, Yang X. Active contour model driven by local histogram fitting energy. *Pattern Recognit Lett* 2013;34:655-62. DOI
44. Wang H, Huang TZ, Xu Z, Wang Y. An active contour model and its algorithms with local and global Gaussian distribution fitting energies. *Inform Sciences* 2014;263:43-59. DOI
45. Ji Z, Xia Y, Sun Q, Cao G, Chen Q. Active contours driven by local likelihood image fitting energy for image segmentation. *Inform Sciences* 2015;301:285-304. DOI
46. Yang Y, Ren H, Hou X. Level set framework based on local scalable Gaussian distribution and adaptive-scale operator for accurate image segmentation and correction. *Signal Processing: Image Communication* 2022;104:116653. DOI
47. Ge P, Chen Y, Wang G, Weng G. A hybrid active contour model based on pre-fitting energy and adaptive functions for fast image segmentation. *Pattern Recogn Lett* 2022;158:71–79. DOI
48. Ding K, Xiao L, Weng G. Active contours driven by local pre-fitting energy for fast image segmentation. *Pattern Recogn Lett* 2018;104:29–36. DOI
49. Ge P, Chen Y, Wang G, Weng G. An active contour model driven by adaptive local pre-fitting energy function based on Jeffreys divergence for image segmentation. *Expert Syst Appl* 2022;210:118493. DOI
50. Liu G, Jiang Y, Chang B, Liu D. Superpixel-based active contour model via a local similarity factor and saliency. *Measurement* 2022;188:110442. DOI

51. Chen H, Zhang H, Zhen X. A hybrid active contour image segmentation model with robust to initial contour position. *Multimed Tools Appl* 2022 sep. DOI
52. Yang Y, Hou X, Ren H. Efficient active contour model for medical image segmentation and correction based on edge and region information. *Expert Syst Appl* 2022;194:116436. DOI
53. Zhang W, Wang X, You W, et al. RESLS: Region and edge synergetic level set framework for image segmentation. *IEEE Trans Image Process* 2020;29:57-71. DOI
54. Dong B, Weng G, Jin R. Active contour model driven by Self Organizing Maps for image segmentation. *Expert Syst Appl* 2021;177:114948. DOI
55. Fang J, Liu H, Liu J, et al. Fuzzy region-based active contour driven by global and local fitting energy for image segmentation. *Applied Soft Comput* 2021;100:106982. DOI
56. Weng G, Dong B, Lei Y. A level set method based on additive bias correction for image segmentation. *Expert Syst Appl* 2021;185:115633. DOI
57. Jin R, Weng G. A robust active contour model driven by pre-fitting bias correction and optimized fuzzy c-means algorithm for fast image segmentation. *Neurocomputing* 2019;359:408–19. DOI
58. Han B, Wu Y. Active contour model for inhomogenous image segmentation based on Jeffreys divergence. *Pattern Recogn* 2020;107:107520. DOI
59. Asim U, Iqbal E, Joshi A, Akram F, Choi KN. Active contour model for image segmentation with dilated convolution filter. *IEEE Access* 2021;9:168703–14. DOI
60. Costea C, Gavrea B, Streza M, Belean B. Edge-based active contours for microarray spot segmentation. *Proce Compu Sci* 2021;192:369–75. DOI
61. Fang J, Liu H, Zhang L, Liu J, Liu H. Region-edge-based active contours driven by hybrid and local fuzzy region-based energy for image segmentation. *Inform Sciences* 2021;546:397–419. DOI
62. Yu H, He F, Pan Y. A novel segmentation model for medical images with intensity inhomogeneity based on adaptive perturbation. *Multimed Tools Appl* 2019;78:11779-98. DOI
63. Sirakov NM. A new active convex hull model for image regions. *J Math Imaging Vis* 2006;26:309-25. DOI
64. Osher S, Sethian JA. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *J Compu Phys* 1988;79:12–49. DOI
65. Kass M, Witkin A, Terzopoulos D. Snakes: active contour models. *Int J Comput Vision* 1988;1:321-31. DOI
66. Bresson X, Vandergheynst P, Thiran JP. A variational model for object segmentation using boundary information and shape prior driven by the mumford-shah functional. *Int J Comput Vision* 2006;68:145-62. DOI
67. Deriche M, Amin A, Qureshi M. Color image segmentation by combining the convex active contour and the Chan Vese model. *Pattern Anal Applic* 2019;22:343-57. DOI
68. Aubert G, Kornprobst P. Mathematical problems in Image Processing. New York: Springer; 2006. DOI
69. Wang Y, He C. An adaptive level set evolution equation for contour extraction. *Appl Math Comput* 2013;219:11420–29. DOI
70. Yan X, Weng G. Hybrid active contour model driven by optimized local pre-fitting image energy for fast image segmentation. *Appl Math Model* 2022;101:586–99. DOI
71. Chen LC, Zhu Y, Papandreou G, Schroff F, Adam H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. European Conference on Computer Vision 2018 Feb.
72. He K, Gkioxari G, Dollar P, Girshick R. Mask R-CNN. arXiv 2017 Mar.
73. Chen X, Williams BM, Vallabhaneni SR, et al. Learning active contour models for medical image segmentation. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE; 2019. pp. 11624–32. DOI
74. Ma J, He J, Yang X. Learning geodesic active contours for embedding object global information in segmentation CNNs. *IEEE Trans Med Imaging* 2021;40:93-104. DOI
75. Gu J, Fang Z, Gao Y, Tian F. Segmentation of coronary arteries images using global feature embedded network with active contour loss. *Comput Med Imaging Graph* 2020;86:101799. DOI
76. Gur S, Wolf L, Golgher L, Blinder P. Unsupervised microvascular image segmentation using an active contours mimicking neural network. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV). IEEE; 2019. pp. 10721–30. DOI
77. Kim B, Ye JC. Mumford–Shah Loss Functional for Image Segmentation With Deep Learning. *IEEE Trans Image Process* 2020;29:1856–66. DOI
78. Tao H, Qiu J, Chen Y, Stojanovic V, Cheng L. Unsupervised cross-domain rolling bearing fault diagnosis based on time-frequency information fusion. *J Franklin Ins* 2023;360:1454-77 DOI
79. Chen H, Li L, Shang C, Huang B. Fault detection for nonlinear dynamic systems With consideration of modeling errors: a data-Driven approach. *IEEE Trans Cybern* 2022;1–11. DOI
80. Qu F, Tian E, Zhao X. Chance-Constrained H_∞ State Estimation for Recursive Neural Networks Under Deception Attacks and Energy Constraints: The Finite-Horizon Case. *IEEE Trans Neural Netw Learn Syst* 2022;1–12. DOI
81. Chen Y, Jiang W, Charalambous T. Machine learning based iterative learning control for non-repetitive time-varying systems. *Int J Robust Nonlinear* 2022;Early Veiw. DOI
82. Han B, Wu Y. A hybrid active contour model driven by novel global and local fitting energies for image segmentation. *Multimed Tools Appl* 2018;77:29193-208. DOI
83. Yang X, Jiang X, Zhou L, Wang Y, Zhang Y. Active contours driven by Local and Global Region-Based Information for Image Segmen-

tation. *IEEE Access* 2020;8:6460–70. DOI

84. Chen Y, Zhou Y. Machine learning based decision making for time varying systems: Parameter estimation and performance optimization. *Knowledge-Based Systems* 2020;190:105479. DOI
85. Chen Y, Zhou Y, Zhang Y. Machine Learning-Based Model Predictive Control for Collaborative Production Planning Problem with Unknown Information. *Electronics* 2021;10:1818. DOI
86. Chen H, Chai Z, Dogru O, Jiang B, Huang B. Data-Driven Designs of Fault Detection Systems via Neural Network-Aided Learning. *IEEE Trans Neur Net Lear Syst* 2021:1–12. DOI
87. Jiang W, Chen Y, Chen H, Schutter BD. A Unified Framework for Multi-Agent Formation with a Non-repetitive Leader Trajectory: Adaptive Control and Iterative Learning Control. TechRxiv 2023 jan. DOI

Intelligence & Robotics

Research Article

Open Access

GMAW welding procedure expert system based on machine learning

Xuewu Wang[1], Qian Chen[1], Hao Sun[2], Xiuwei Wang[2], Huaicheng Yan[1]

[1]Key Laboratory of Smart Manufacturing in Energy Chemical Process, Ministry of Education, East China University of Science and Technology, Shanghai 201907, China.
[2]Lincoln Electric Management (Shanghai) Co. Ltd, Shanghai 201907, China.

Correspondence to: Assoc. Prof. Xuewu Wang, Key Laboratory of Smart Manufacturing in Energy Chemical Process, Ministry of Education, East China University of Science and Technology, Shanghai 201907, China. E-mail: wangxuew@ecust.edu.cn

How to cite this article: Wang X, Chen Q, Sun H, Wang X, Yan H. GMAW welding procedure expert system based on machine learning. *Intell Robot* 2023;3(1):56-75. https://dx.doi.org/10.20517/ir.2023.03

Received: 14 Dec 2022 **First Decision:** 31 Jan 2023 **Revised:** 12 Feb 2023 **Accepted:** 22 Feb 2023 **Published:** 1 Mar 2023

Academic Editors: Simon Yang, Jianjun Ni **Copy Editor:** Ying Han **Production Editor:** Ying Han

Abstract

In order to simplify the robot preparation before welding and improve the automation of the whole welding process, an intelligent expert system for Gas Metal Arc Welding is designed in this paper. In the system, the user inputs the initial welding information and the output interface displays suitable welding procedure parameter schemes. The user can choose the schemes according to the actual requirements or directly generate the welding procedure specification required by the enterprise format for direct use. In addition, the system also combines the database technology and XGBoost algorithm in the field of machine learning, migrates the model trained on the data set to predict the welding raw data, accumulates more data for daily use to optimize the model, which makes the whole system more systematic and intelligent, and achieves the goal of more accurate use.

Keywords: Welding, expert system, machine learning, database

1. INTRODUCTION

With the development of the industrial internet of things, big data, artificial intelligence, machine learning and other technologies, information technology has been popularized, and intelligent manufacturing has also developed rapidly. Digitalization and intelligence have become the basis of manufacturing production. Welding is the basic processing method in the manufacturing industry. The combination of them promotes the automation and intelligence of welding process. With the urgent demand for data sharing and professional knowledge in welding enterprises, the welding expert system plays an increasingly important role in intelligent welding systems[1].

An expert system is a computer program that combines the experience and knowledge of experts in a specific field. As early as the mid-1960s, a large number of researchers began to study the expert system. Welding technology is the product of multi-disciplines, ranging from weldability analysis before welding[2,3] and preparation of welding process documents[4] to the selection of process parameters in the welding process[5-8] and real-time control of welding[9], defect diagnosis and treatment after welding[10], welding quality assessment[11,12] and calculation of welding material consumption[13]. Each stage involves a large amount of data, knowledge and models, which is an ideal field for expert system application[14,15].

GMAW is a welding method that uses the arc heat generated between the welding wire and the weldment to melt the welding wire and the base material continuously fed by the welding torch under shielding gas to achieve the welding purpose. This method has the advantages of low cost, strong adaptability, easy operation and easy automation. The main advantage of GMAW lies in its metal deposition per hour, which greatly reduces labor costs. In addition, this technology is a clean process because it does not use flux, and it is easy for operators to observe the situation of the arc and molten pool and make adjustments at any time. The advantages of GMAW make it widely used in many industrial applications[16]. Metal transfer modes are the way to transfer the molten electrode droplet to the molten pool. The modes are generally classified into three forms, which are often used for welding quality control and defect detection. Different metal transfer modes correspond to different welding parameters, especially current parameters. In this expert system, the current is mainly used as the output parameter, so the metal transfer modes are not considered as the input parameter[17].

Welding procedure design expert system has always been a hot topic in the research of the welding expert system. A complete welding process involves many parameters. If developers can collect key information such as welding joint, groove, size and position through expert systems, process the data with the help of expert experience and knowledge, study relevant welding standards in the industry, and combine data mining and machine learning technology to generate specific welding process parameter schemes, which could avoid plenty of repetitive work. The use of the expert system improves the storage mechanism of welding data. By constantly updating the expert system with new technologies in this field, knowledge beyond expert experience will be found, which plays an important role in promoting the design of welding technology. In this case, research on the welding expert system is very necessary.

From the traditional expert system, which can only answer the results already in the system, to the expert system combining machine learning algorithms, software and hardware technology[18-22]. Due to the rapid development of new computer technology, researchers in the world have made many attempts. The expert system can help users to make decisions by simulating expert thinking and predict answers to questions that are not involved in the knowledge base.

The design of the knowledge base, database and inference engine is the key part of an expert system design. The sustainability and extensibility of development are the difficulties in designing each module. As for the knowledge base design, Zhang[23] uses the traditional design pattern and divides the design into knowledge representation and knowledge acquisition. Liu *et al.* use knowledge graph technology to build an expert system[24]. The system crawls the data on the Internet, performs format conversion, entity recognition and other processing on the data, which accumulates frequently asked questions and their solutions to form the knowledge base, and provides a new idea for the knowledge acquisition of the expert system. Considering that the filling and modification of knowledge in the future may completely depend on the developer if only the database is used, which is not intelligent, and only applying crawler technology does not meet the parameter design requirements of accuracy and reliability. Therefore, the knowledge base design of this system combines the two methods, which employ the algorithm to improve knowledge acquisition and knowledge representation using the database. Regarding the design of the database and inference engine, Xiao *et al.* uses a transfer learning model[25]. Firstly, the system learns the sample dataset to form the training model, and then migrates it to the system. The daily sampled data are further learned by the deep learning algorithm to continuously optimize the model. Similarly, this system designs a self-learning module based on the transfer learning model. It overcomes the problem that there is no accumulated data when the software is just used, and the number of reasonable solutions is insufficient in complex welding conditions, which leads to the inaccuracy of recommended solutions. Some machine learning models provide black-box prediction, and the prediction accuracy is satisfactory, but the model explanation is insufficient, which leads to users' lack of confidence in the reliability of the results. Ahmed *et al.* puts forward a framework to work together with the established machine learning model and expert knowledge[26]. Based on the generated model, the framework extracts decision rules and adopts expert suggestions, makes the dataset self-sufficient and constantly generates new rules by checking data sets. This system also takes into account the expert suggestions to improve the prediction accuracy of the system and interfaces some key parameters, which allows experts to adjust parameters according to experience and the effect of parameters in actual welding, so as to achieve the desired output.

The welding process is complicated, the final output parameters of the expert system are only a part of the welding process, and many parameters affect the welding process, so it is usually fundamental to visually monitor the welding process. Human welders can adjust the parameters in real time by observing the welding process to obtain the desired output. In addition to using sensing technology to simulate human welders to monitor the welding process, the welding procedure design expert system combined with the machine learning algorithm only needs to give the initial information and expected output under the actual welding conditions, and can simulate the intermediate process by itself based on the data without the user's understanding of the welding process, and obtain the expected output. The training data set often comes from the combination of parameters with a good welding effect, and with more data accumulated in the system, the algorithm learns again and optimizes the existing model parameters, which is also the correction of ideal output and actual welding results, and realizes the intelligent adaptation of the welding process to the environment to some extent[27].

In recent years, XGBoost has gained great popularity in various data science competitions. The gradient boosting principle behind XGBoost is a representative algorithm of the boosting method in the ensemble algorithm, and XGBoost can be applied to both classification and regression problems. As verified by Guan[28], compared with BP neural network, ARMA model and KNN algorithm, XGBoost model has higher accuracy and faster operation efficiency. In this system, XGBoost, with the best prediction performance, is selected as the machine learning model. With the adoption of different algorithms, expert systems have been widely studied and applied in various fields of science, engineering, agriculture and medicine, providing valuable experience for the research and development of welding expert systems in this paper[29-36].

The overall architecture design of the GMAW welding procedure expert system, and the main functional modules will be summarized first. The second section will describe the knowledge base and the methods to supplement and update it. The database technology will be introduced in the third section, describe the main components and show some sample tables. The fourth section will introduce the reasoning logic of this system, and expound on the algorithm principle of welding parameter intelligent recommendation module. The fifth section will apply the expert system to practice, introduce the system workflow, and display the key output parts, as well as the concrete effect drawings obtained by actual welding. The last part is the summary and prospect of this paper.

2. OVERALL DESIGN OF GMAW EXPERT SYSTEM

The composition of the expert system usually includes the human-computer interaction interface, knowledge base, inference engine, and interpretation engine part. Faced with questions posed by users, computer programs simulate the thinking of experts to solve problems. Through the knowledge acquisition module, the human-computer interface is the carrier of getting user input information, calling the system's storage of knowledge and rules, relying on the inference engine for the logical reasoning calculations, with the aid of the interpretation engine all the knowledge and logic used to conclude can be viewed, makes the whole operation process is determined.

The overall structure of the expert system is shown in Figure 1. Considering the welding process expert system involves many parameters, the human-computer interaction interface, knowledge base database, and inference engine's realization are the main parts of the system design.

Human-computer interaction interface is the dialogue window between the user and the program. The system involves many parameters of input and calculation, the interface will be an important part of the system design, and considering the actual welding environment is not clear, the interface design and function realization is mainly on the Qt Creator platform, and packaged to generate software that can be used offline, can cope with the complex working environment.

The knowledge base and database are the core part of the expert system. The completeness of system knowledge determines the reliability and accuracy of an expert system decision, and the sustainability of knowledge base development determines the service life of the expert system. This system adopts the mode of knowledge base and database work together. The software has its own basic knowledge base and database at first, and users then use the system to produce new data and timely feedback to the system database. When accumulating a certain amount of new data, the expert system uses the machine learning algorithm to learn new knowledge from new data, and optimize the existing systems. The knowledge base and database can be updated constantly to achieve the goal of more accuracy and more consistency with the actual production scene.

To evaluate the correctness of an expert system's decision, in addition to the need for a good knowledge base design, it is more important to make the most of the existing professional knowledge. The inference engine is the whole knowledge and control strategy used to deduce the conclusion, and it is the key to integrating knowledge into the program. There are many welding parameters in a complete welding process, and the coupling degree of parameters is high. Changing one parameter will lead to the recalculation of a series of parameters. Therefore, the reasoning mechanism of this system combines accuracy and fuzzy. When the initial information is sufficient, the system calls the knowledge base and database to use forward reasoning to achieve accurate reasoning. When the initial information is insufficient to reason, the system uses the database data and combines the machine learning model to make a fuzzy prediction of the result.

Based on the overall architecture design of the system, the specific functional modules of the system mainly include the following parts:

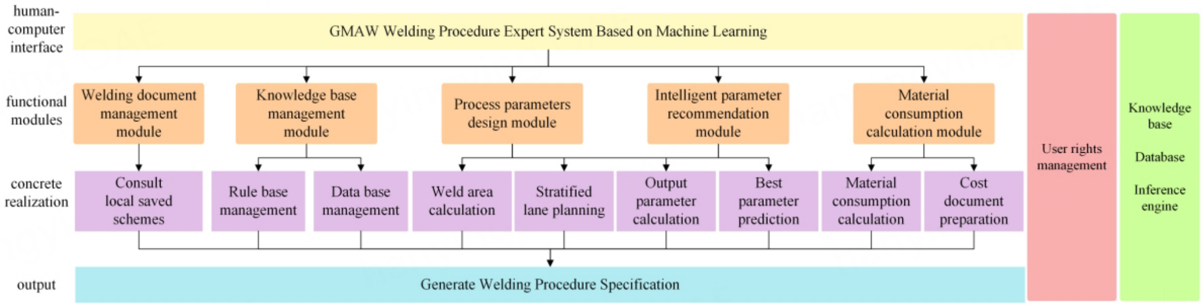

human-computer interface
functional modules
concrete realization
output
GMAW Welding Procedure Expert System Based on Machine Learning
Welding document management module
Knowledge base management module
Process parameters design module
Intelligent parameter recommendation module
Material consumption calculation module
Consult local saved schemes
Rule base management
Data base management
Weld area calculation
Stratified lane planning
Output parameter calculation
Best parameter prediction
Material consumption calculation
Cost document preparation
Generate Welding Procedure Specification
User rights management
Knowledge base
Database
Inference engine

Figure 1. Overall system design.

(1) User management module

This module mainly divides the identity of users into ordinary user, engineer, and administrator roles. Ordinary users can use basic functional modules. On the basis of ordinary users, engineers can modify the knowledge base data on the interface, and generate more customized parameter schemes for the user's working environment. In addition, the administrator can manage user information, enhance user' rights, and carry out daily system maintenance.

(2) Document management module

The module includes consulting the industry welding standards and weldability documents of welding materials, managing previous welding schemes and statistical information on weld cost corresponding to product functions, and providing help for users' pre-welding preparation.

(3) Knowledge base module

This module is responsible for the management of the rule base and knowledge database. Users modify relevant parameters according to their actual needs, and the system calculates the modified parameters in real time to generate a more guiding scheme for the work site environment. In addition, users can save the expert system parameters with expected welding effects in the actual welding site to the database.

(4) Procedure parameters design module

The welding procedure design module is the key part of this expert system, which also includes sub-functional modules such as calculation of weld cross-sectional area, planning of multi-layer and multi-pass welding, calculation of welding parameters and intelligent recommendation of parameter schemes. The weld cross-sectional area calculation module prepares data for groove stratified separation, where the stratified separation scheme under different welding parameters is various, the parameters of the recommended scheme based on different modes, are also combined with the cost calculation module to meet the user's need.

(5) Material consumption calculation module

This module mainly includes the input and preservation of weld information, calculation of weld cost, and generation of cost files. For a specific product, users input the relevant information according to the system, where the cost can be divided into two parts, artificial operation cost and material cost. Save input information into the corresponding cost file, and summarize the total length and mass, welding wire consumption and other information of all welds under the current product. The module also provides two working modes: single weld and weld cost comparison mode. The calculation method also has different evaluation indicators to meet different needs. The flow chart of the module is shown in Figure 2.

In the system, users input relevant information under the prompt, and the system integrates the information and displays it in a user-friendly manner. The output interface provides a set of appropriate welding procedure design schemes, which supports the user to modify the data and real-time calculation, but also can directly export the WPS for direct use. The extension module of the consumable cost calculation provides more reference indexes to meet the demand of actual welding conditions. With the use of this expert system, the formulation of the welding parameter scheme is not limited to experienced engineers, but enables new technical personnel to complete reasonable and effective welding through the welding scheme recommended by the system, avoiding the dependence on experienced personnel before actual welding.

3. KNOWLEDGE BASE DESIGN

An expert system integrates the knowledge of multiple experts in related fields. The accuracy of an expert system depends mainly on the completeness of the knowledge base and the ability of programmers to convert expert knowledge into codes. The content of the knowledge base can be divided into explicit knowledge and tacit knowledge. Explicit knowledge refers to that formulaic, regular and easy-to-express knowledge, while tacit knowledge refers to empirical, heuristic and difficult-to-convert knowledge. Tacit knowledge in this system mainly comes from experts' experience and the enlightening knowledge obtained by reading welding documents. The explicit knowledge in this system mainly includes the part shown in Figure 3.

In terms of designing the knowledge base, considering the wide variety of welding technologies and welding equipment, different instruments, operators and environmental conditions will lead to differences in the welding process. Even if the same parameters are set, the welding effect is different under different environments. In the realization of the knowledge part of the system, the tacit knowledge is incorporated into the program, the explicit knowledge is usually stored in the knowledge base, and key parameters and formulas that may need to be modified are made into interfaces to wait for the adjustment of experts, so as to meet the desired output under the actual welding conditions and make the whole system more flexible.

The knowledge acquisition part is the expansion of the knowledge base. The knowledge acquisition of this system mainly comes from the following aspects:

1. Basic rule base and database after downloading the software;

2. New welding data is added to the database after users use this expert system;

3. New knowledge learned by machine learning algorithm on new welding data;

4. Update data on the server at irregular intervals.

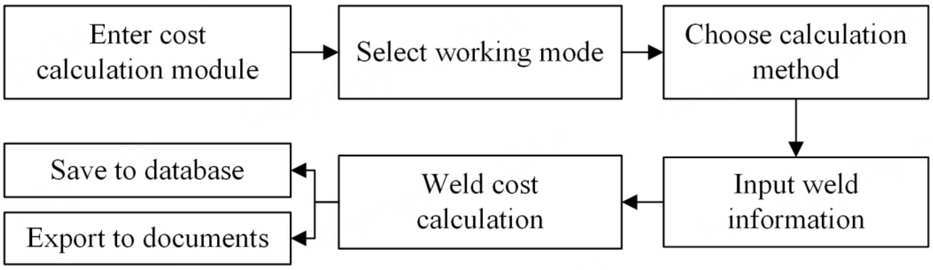


Figure 2. Flow chart of weld cost calculation.

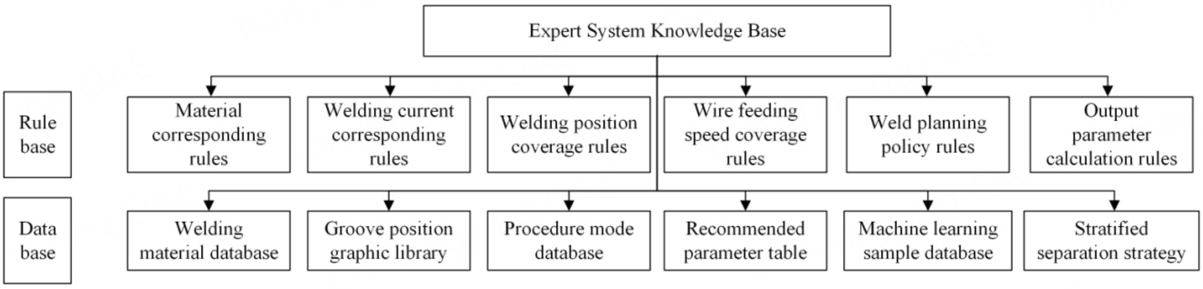

Expert System Knowledge Base
Rule base
Material corresponding rules
Welding current corresponding rules
Welding position coverage rules
Wire feeding speed coverage rules
Weld planning policy rules
Output parameter calculation rules
Data base
Welding material database
Groove position graphic library
Procedure mode database
Recommended parameter table
Machine learning sample database
Stratified separation strategy

Figure 3. Part of knowledge base display.

4. DATABASE DESIGN

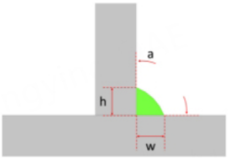
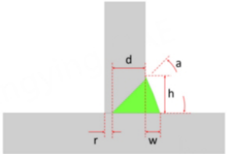
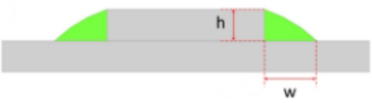
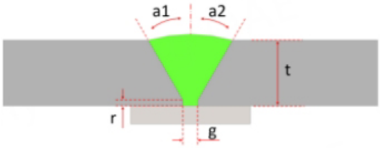
The database part of the system combines MySQL database and SQLite database, which includes three major contents: a user login information table, welding information database, and cost calculation database. Considering the unclear welding environment, the basic functions can be realized by using the local SQLite database provided by the software when there is no network. When the network is available, users connect to MySQL remote database and automatically update the local database and upload user data to complete the update of the knowledge base and database on the client and server. In the design of the expert system database, different databases are selected, so as to improve the adaptability of the system to different welding environments. The database contents of some recommended parameter settings are shown in Figure 4.

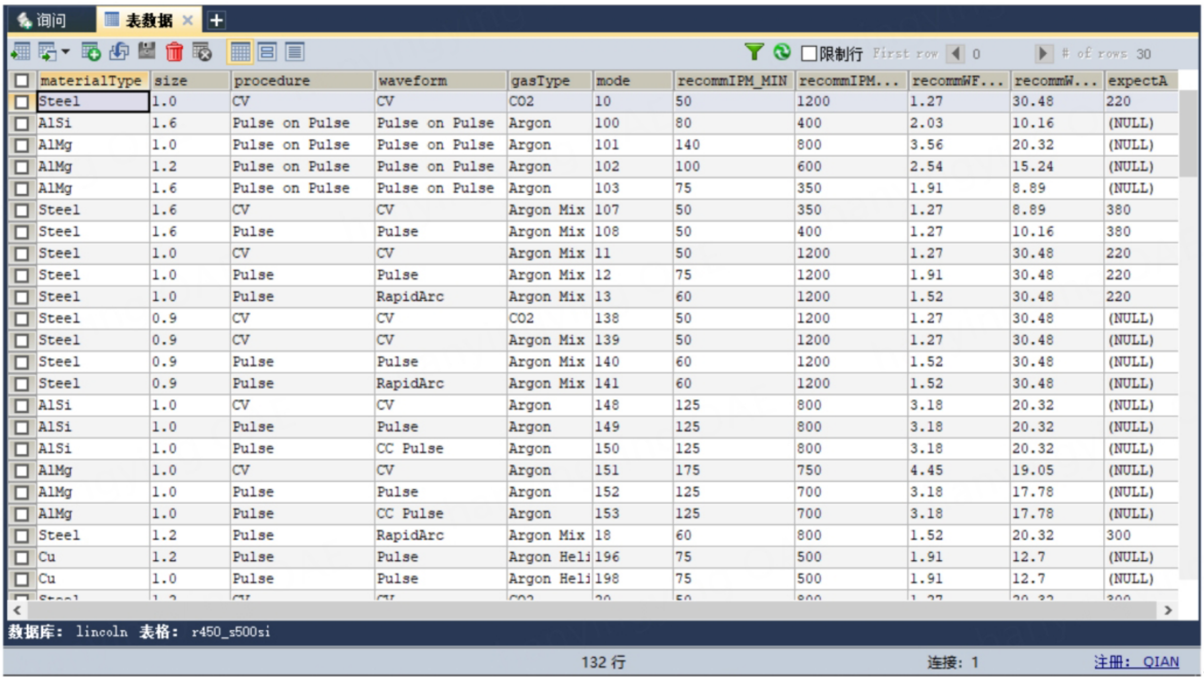
Among the tables as the main content of the database, the welding information database includes a series of tables, such as the groove information table, welding position graph table, layered lane division strategy table, welding setting reference table and so on. The cost calculation database stores specific information and corresponding cost statistics of each weld under different products. Table 1 shows only part of the database.

5. INFERENCE ENGINE DESIGN

The reasoning logic of this system is mainly shown in Figure 5. According to the initial information of joints, welding wires and positions input by users, combined with the corresponding rule base and database, the system carries out multi-layer and multi-pass welding planning for the joint groove of specific welding positions, and obtains detailed data information of each weld. The key parameters set are loaded and calculated in real time to obtain the important parameters guiding robot welding. Users can modify the data information in each weld to generate welding parameters that meet the user's expectations.

Table 1. Sketches of common welded joint

Joint type	Groove type	Position	Weld sketch	Dimension parameter
T-joint	I	2F		h: weld leg size w: weld leg size a: groove angle
T-joint	Single-V	2F		a: groove angle d: groove depth r: root face w: weld leg size h: weld leg size
Lap-joint	I	2F		h: weld leg size w: weld leg size
Butt-joint	V	1G		t: thickness a1: groove angle a2: groove angle r: root face g: root gap



materialType	size	procedure	waveform	gasType	mode	recommIPM_MIN	recommIPM...	recommWF...	recommW...	expectA
Steel	1.0	CV	CV	CO2	10	50	1200	1.27	30.48	220
AlSi	1.6	Pulse on Pulse	Pulse on Pulse	Argon	100	80	400	2.03	10.16	(NULL)
AlMg	1.0	Pulse on Pulse	Pulse on Pulse	Argon	101	140	800	3.56	20.32	(NULL)
AlMg	1.2	Pulse on Pulse	Pulse on Pulse	Argon	102	100	600	2.54	15.24	(NULL)
AlMg	1.6	Pulse on Pulse	Pulse on Pulse	Argon	103	75	350	1.91	8.89	(NULL)
Steel	1.6	CV	CV	Argon Mix	107	50	350	1.27	8.89	380
Steel	1.6	Pulse	Pulse	Argon Mix	108	50	400	1.27	10.16	380
Steel	1.0	CV	CV	Argon Mix	11	50	1200	1.27	30.48	220
Steel	1.0	Pulse	Pulse	Argon Mix	12	75	1200	1.91	30.48	220
Steel	1.0	Pulse	RapidArc	Argon Mix	13	60	1200	1.52	30.48	220
Steel	0.9	CV	CV	CO2	138	50	1200	1.27	30.48	(NULL)
Steel	0.9	CV	CV	Argon Mix	139	50	1200	1.27	30.48	(NULL)
Steel	0.9	Pulse	Pulse	Argon Mix	140	60	1200	1.52	30.48	(NULL)
Steel	0.9	Pulse	RapidArc	Argon Mix	141	60	1200	1.52	30.48	(NULL)
AlSi	1.0	CV	CV	Argon	148	125	800	3.18	20.32	(NULL)
AlSi	1.0	Pulse	Pulse	Argon	149	125	800	3.18	20.32	(NULL)
AlSi	1.0	Pulse	CC Pulse	Argon	150	125	800	3.18	20.32	(NULL)
AlMg	1.0	CV	CV	Argon	151	175	750	4.45	19.05	(NULL)
AlMg	1.0	Pulse	Pulse	Argon	152	125	700	3.18	17.78	(NULL)
AlMg	1.0	Pulse	CC Pulse	Argon	153	125	700	3.18	17.78	(NULL)
Steel	1.2	Pulse	RapidArc	Argon Mix	18	60	800	1.52	20.32	300
Cu	1.2	Pulse	Pulse	Argon Heli	196	75	500	1.91	12.7	(NULL)
Cu	1.0	Pulse	Pulse	Argon Heli	198	75	500	1.91	12.7	(NULL)

数据库： lincoln 表格： r450_s500si

132 行 连接：1 注册： QIAN

Figure 4. Part of database display.

The design of the inference engine uses the principle of accurate and fuzzy reasoning. The concrete realization is that when the user inputs enough initial information, the inference engine calls rules and data, carries on the forward inference and obtains the accurate output. When the user inputs only part of the data, the system invokes the machine learning model built on the sample dataset and predicts the output. When more data instances are accumulated, the expert system will use the self-learning module integrated

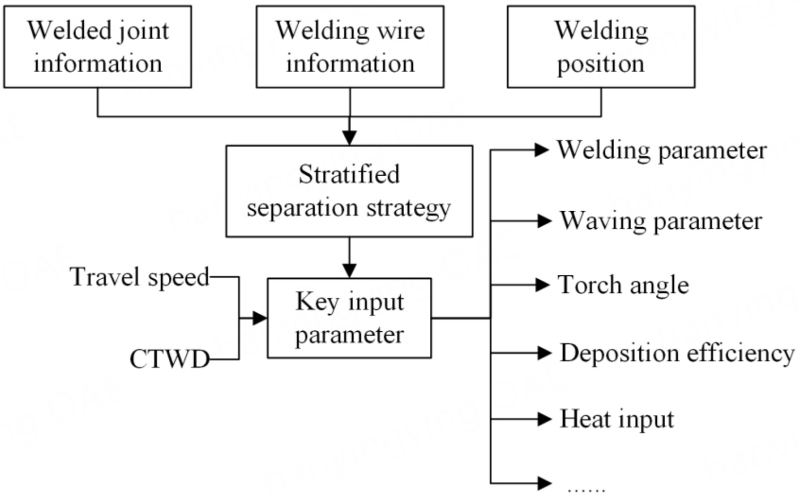


Figure 5. Expert system inference logic.

into the system to re-learn the model on the new data on the basis of the existing model, adjust the parameters of the model, and continuously improve the prediction accuracy.

5.1 XGBoost algorithm principle

XGBoost means eXtreme Gradient Boosting algorithm, and its essence is a binary tree based on the Gradient Boosting algorithm. Its model realization is to add a new tree at each iteration, and increase the weight of the sample data with the wrong prediction of the new tree, to achieve the goal of improving model accuracy after continuous iteration.

The model's objective function innovatively introduces a regularization term to calculate the model complexity, which avoids the natural overfitting characteristic of the binary tree model, and also greatly improves the operation speed of the model, making the operation speed more than ten times faster than the other models, which meets the real-time requirements of the expert system. Moreover, the algorithm adopts cache access mode and data compression processing, which supports the expansion of the model to more than billions of instances, and meets the requirements of multiple parameters and fast data growth in the welding field.

$$obj=\sum_{i=1}^{m}L(y_i,\hat{y}_i)+\sum_{k=1}^{K}\Omega(f_k) \quad (1)$$

$$\Omega(\mathrm{f})=\gamma\mathrm{T}+\frac{1}{2}\lambda\sum_{\mathrm{j}=1}^{\mathrm{T}}\omega_{\mathrm{j}}^{2} \quad (2)$$

In Equation (1), function L is the loss function of the model, which measures the difference between the true value y_i and the predicted value $\hat{y}_i$. m is the total number of samples. The Ω part measures the complexity of the model, and f_k represents the tree structure that retains the best performance in the kth iteration. This model was designed by Chen *et al.*[37]. The objective function consists of two parts: traditional loss function and model complexity calculation. As shown in Equation (2), the calculation of model complexity consists of the number of leaf nodes T of each tree as L1 regularization term and the cumulative sum of squares of leaf weights as L2 regularization term. By increasing the coefficients of the two terms as penalty terms, users can avoid the overfitting phenomenon caused by too many branches of the tree model. The above two formulas aim to strike a balance between the loss function and the relatively simple model structure, and attempt to obtain the model parameters with less computational resources and higher model accuracy.

$$f_t\left(X_i\right)=\omega_{q\left(X_i\right)} \tag{3}$$

$$\hat{y}_i=\sum_{k=1}^{K} f_k\left(X_i\right) \tag{4}$$

Chen TQ innovatively proposed the concepts of leaf nodes and leaf weights on trees. For each tree created by XGBoost, the structure f_t performs the best in that iteration. By classifying the sample dataset on the tree, a tree is divided into q classes. The predicted value of each sample X_i on the tree is taken from the mean of all the samples on the leaf, which is the leaf weight $\omega_{q(X_i)}$, as shown in Equation (3). The total prediction of the entire model for a specific sample value is expressed by summing the predicted value of the sample on all trees, as shown in Equation (4).

$$obj^{(t)}=\sum_{i=1}^{m} L\left(y_i^{(t)},\hat{y}_i^{(t-1)}+f_t\left(X_i\right)\right)+\Omega\left(f_t\right) \tag{5}$$

Based on the above formulas, the objective function at the tth iteration is shown in Equation (5). According to the principle of XGBoost, now $y_i^{(t)}$ $y_i^{(t-1)}$ is a constant, so the optimization of the function is to optimize the tree structure generated by the next iteration each time. The smaller the partial value of the loss function of the XGBoost model, the smaller the deviation of prediction. The simpler the structure of the model, the smaller the corresponding prediction variance, which makes the model show an expected prediction effect no matter what type of data is applied.

5.2 Intelligent parameter recommendation

A complete set of the machine learning process includes data preprocessing, data exploration, feature engineering, machine learning modeling, parameter tuning, and test set validation. In this paper, for the intelligent prediction module of parameters, since this is a supervised regression problem, five regression prediction models containing linear regression, decision tree regression, random forest, polynomial regression and XGBoost are used to compare the model accuracy and operation speed on the same dataset. The dataset includes wire type, wire diameter, distance from conducting nozzle to the workpiece, wire feeding speed, and current, where current is used as the prediction column in this regression. There are 492 samples in the dataset.

(1) Modeling steps

Firstly, considering that the dataset comes from the reasonable parameter scheme of the welding site, there are no outliers or missing values. After that, the data were mapped in MATLAB to make a study of the data relationship. The features were selected in combination with the relevant knowledge and experience of parameter calculation provided by experts. The linear regression, decision tree, random forest, polynomial regression and XGBoost model, which are widely used in the current regression machine learning algorithms, were used to fit the data. In the model-building process, grid search and learning curve drawing were used to optimize the parameters.

(2) Model parameter optimization

The self-learning module of the system mainly realizes that when more instance data is accumulated in the system, the system calls the internally packaged function to optimize and reset the parameters of the machine learning model. XGBoost model has many parameters. Table 2 contains parameters mainly used for the process of parameter adjustment in this expert system.

As shown in Table 2, the XGBoost model has many parameters, different model has different parameters, and parameter adjustment is also different. In the design process of this expert system, grid search and learning curve drawing methods are combined to find the best parameters.

In Figure 6, the GridSearchCV library in the sklearn module is called to arrange all parameters to find the combination that makes the scoring index perform best. The drawing learning curve method used in Figure 7 mainly relies on the programmer's judgment of the effect. Lines in the figure represent the effect of XGBoost under default parameters, grid search recommended parameters, and learning curve manual parameter adjustment.

From the trend of the lines in Figure 7, XGBoost has a good model effect only with the default parameters, which shows the power of the model design. The parameters determined by drawing the learning curve significantly reduced the degree of the overfitting phenomenon and performed best. The parameters by grid search reduce the error on the training set compared with the learning curve parameters. Grid search is the permutation and combination of all parameters, which is more time-consuming on medium and large datasets, but the advantage is that it is convenient to program, and users do not need to adjust parameters manually. By default, this expert system uses the grid search style to update machine learning models. Due to the time-consuming, the system will update the new data in the background when the data reaches the set threshold. After updating, the system will use the pickle library to save the new model for the next direct invocation.

(1) Model evaluation index

In the whole process, the ratio of 70% training set and 30% test set was used to divide the dataset. The same training set and test set for preliminary debugging are used to get the current parameter combination of the model, and the obtained parameter is used to train each model. After training, the mean absolute error (MAE), mean percentage of absolute error (MAPE), mean square error (MSE), and R^2 (a common indicator of regression models in machine learning) were used to evaluate the accuracy of the model.

The calculation formulas of the model evaluation index in Table 3 are as follows:

$$MAE = \frac{\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right|}{n} \tag{6}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\% \tag{7}$$

$$MSE = \frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}{n} \tag{8}$$

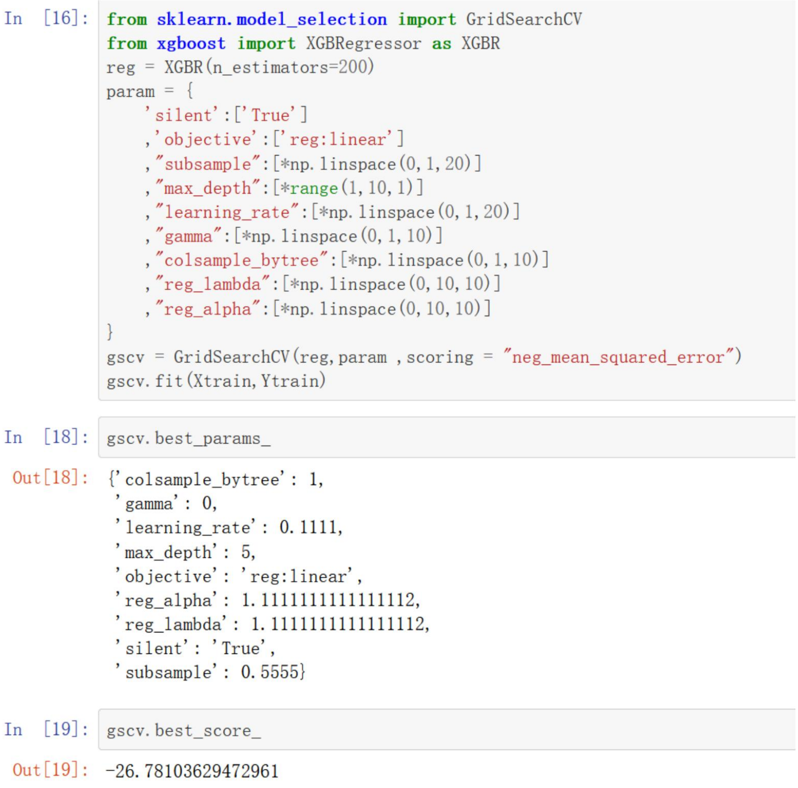
$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{n}\left(y_i - \bar{y}\right)^2} \tag{9}$$

Table 2. XGBoost model parameter list

Model parameters	Instructions
n_estimators	Number of weak evaluators in the model
learning_rate	To control the iteration rate of the tree
subsample	The proportion of sampling from samples
max_depth	Maximum tree depth of the weak evaluators
objective	The objective function
gamma	A decrease in objective function to minimize branching
reg_alpha	L1 regularization coefficient in the objective function
reg_lambda	L2 regularization coefficient in the objective function
n_jobs	Number of parallel threads used to run model
random_state	Random seed
colsample_bytree	The proportion of features selected each time

Table 3. Each model's performance on the test set

Evaluation index	MAE	MAPE	MSE	R^2
	Test set			
Linear regression	22.1175	11.6555%	870.8522	0.9148
Decision tree regression	2.8114	1.4815%	43.3126	0.9958
Random forest	4.5157	2.3797%	57.0616	0.9944
Polynomial regression	2.9628	1.5614%	14.4916	0.9986
XGBoost	2.4408	1.2863%	12.3635	0.9988



```
In  [16]: from sklearn.model_selection import GridSearchCV
          from xgboost import XGBRegressor as XGBR
          reg = XGBR(n_estimators=200)
          param = {
              'silent':['True']
              ,'objective':['reg:linear']
              ,"subsample":[*np.linspace(0,1,20)]
              ,"max_depth":[*range(1,10,1)]
              ,"learning_rate":[*np.linspace(0,1,20)]
              ,"gamma":[*np.linspace(0,1,10)]
              ,"colsample_bytree":[*np.linspace(0,1,10)]
              ,"reg_lambda":[*np.linspace(0,10,10)]
              ,"reg_alpha":[*np.linspace(0,10,10)]
          }
          gscv = GridSearchCV(reg,param ,scoring = "neg_mean_squared_error")
          gscv.fit(Xtrain,Ytrain)

In  [18]: gscv.best_params_

Out[18]: {'colsample_bytree': 1,
           'gamma': 0,
           'learning_rate': 0.1111,
           'max_depth': 5,
           'objective': 'reg:linear',
           'reg_alpha': 1.1111111111111112,
           'reg_lambda': 1.1111111111111112,
           'silent': 'True',
           'subsample': 0.5555}

In  [19]: gscv.best_score_

Out[19]: -26.78103629472961
```

Figure 6. Grid-search codes and results.

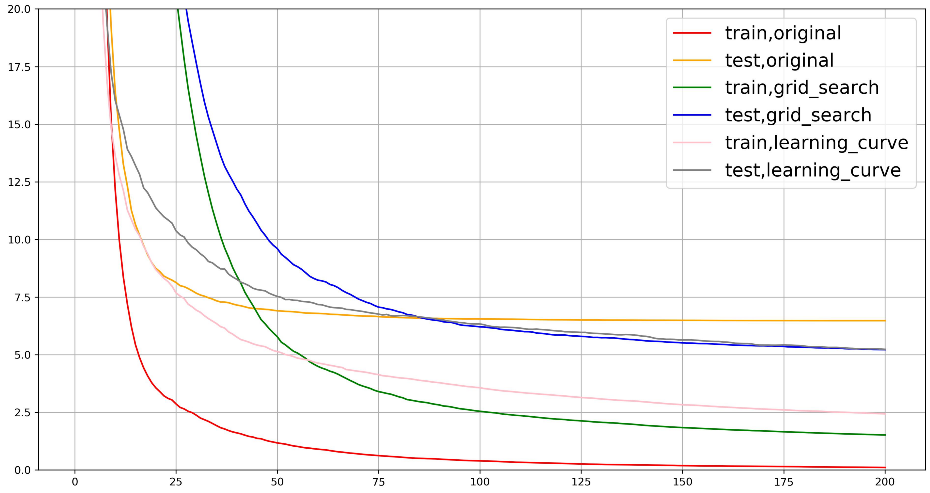


Figure 7. Learning curve search parameters. The X-axis represents the number of weak estimators used in the current model, and the Y-axis shows the mean square error of the model.

(2) Model effect analogy

In Figure 8, the linear regression model has a large variance and low stability after fitting. The fitting effect of the regression tree and random forest regressor is similar, which is more stable than that predicted by the linear regression model, but some data are still not fitted. Obviously, the dataset performs better on polynomial regression and XGBoost, and the model prediction has higher accuracy. Combined with the accuracy in Table 3, the XGBoost model is superior in all aspects. The model is integrated by multiple weak estimators and has a more stable prediction accuracy regardless of large or small datasets. Based on this, XGBoost is adopted as the parameter prediction model in this system, and the actual welding verification shows that the model has a good prediction effect.

6. SYSTEM TEST AND APPLICATION

6.1 System workflow

The software is mainly divided into two functional modules: welding parameter design and weld cost calculation. The parameter design module includes three interfaces, and the user can input information on the first two interfaces according to the actual welding conditions. The system workflow is shown in Figure 9, and the final output is shown in Figure 10. The button on the interface supports the integration of all parameter information into WPS for direct use [Figure 11].

The interface of the cost calculation module is shown in Figure 12. The user completes the input of weld, procedure and cost related parameters on the interface, and the total cost is calculated and displayed in real time. Save, export and import buttons support the function of saving the cost information entered by the user into the database, generating the weld cost file, and reading the unfinished weld information to the work area to continue writing. The statistics button supports statistics of all parameters under the current product. Single mode displays the cost information of a single weld, while the comparison mode can display the cost comparison of two welds.

6.2 Actual application effect

The components of the robot system used are shown in Figure 13. After users input information, the welding effect diagram of the WPS recommended by the expert system in the actual scene is shown in

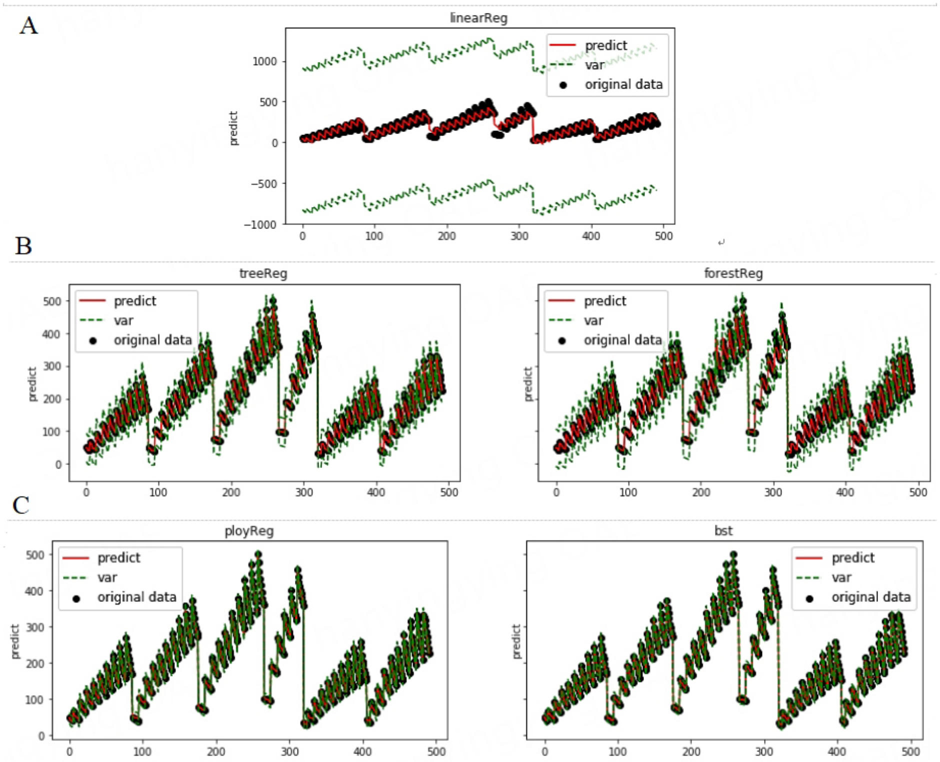


Figure 8. Model fitting data renderings. The X-axis represents the id of the data sample, and the Y-axis represents the value. A: linear regression model; B: respectively are the fitting effects of decision tree regression and random forest, both of which are based on binary trees; C: the model of polynomial regression and XGBoost, both have high prediction accuracies.

Figure 14, and the welding parameters used are the output parameters in Figure 10.

It is easy to see that the welds are well formed with good consistency and less spatter. The weld has no external defects such as edge bite, pore and weld tumor. The measured size of the leg length is 16 × 16, which is consistent with the setting of the expert system. As shown in the macro section photos of the weld in Figure 15 below, the internal fusion of the weld is good, and there are no cracks, pores and other internal defects in the weld. Therefore, the research and development of the expert system are quite necessary.

7. CONCLUSION

After the machine learning model is trained on the initial training set, the model migrates to the expert system, and users use the system to generate new data instances later. When the number of data instances accumulates to a certain extent, the software optimizes and updates the model parameters in the background, continuously improves the prediction accuracy, and avoids waiting a long time for model updates. After downloading, the software carries the initial rule base and database, which can realize all functions of the expert system. With the self-learning module, the rule base and database can be dynamically updated to cope with different welding scenes.

Compared with other procedure design expert systems, this system improves the expansion mechanism of the knowledge base and database, realizes the dynamic update by using machine learning, and can adapt to different situations by selecting different databases. This system will be easy to modify knowledge has been made into an interface; users do not need to operate the database by themselves, but the rest of the

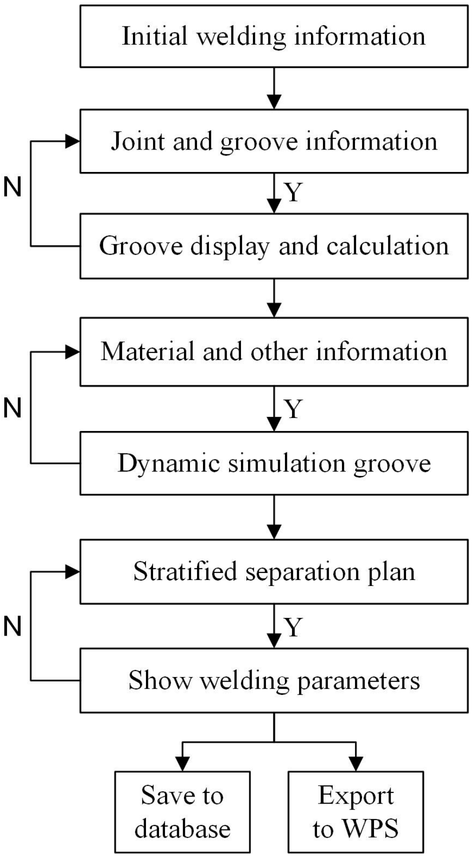
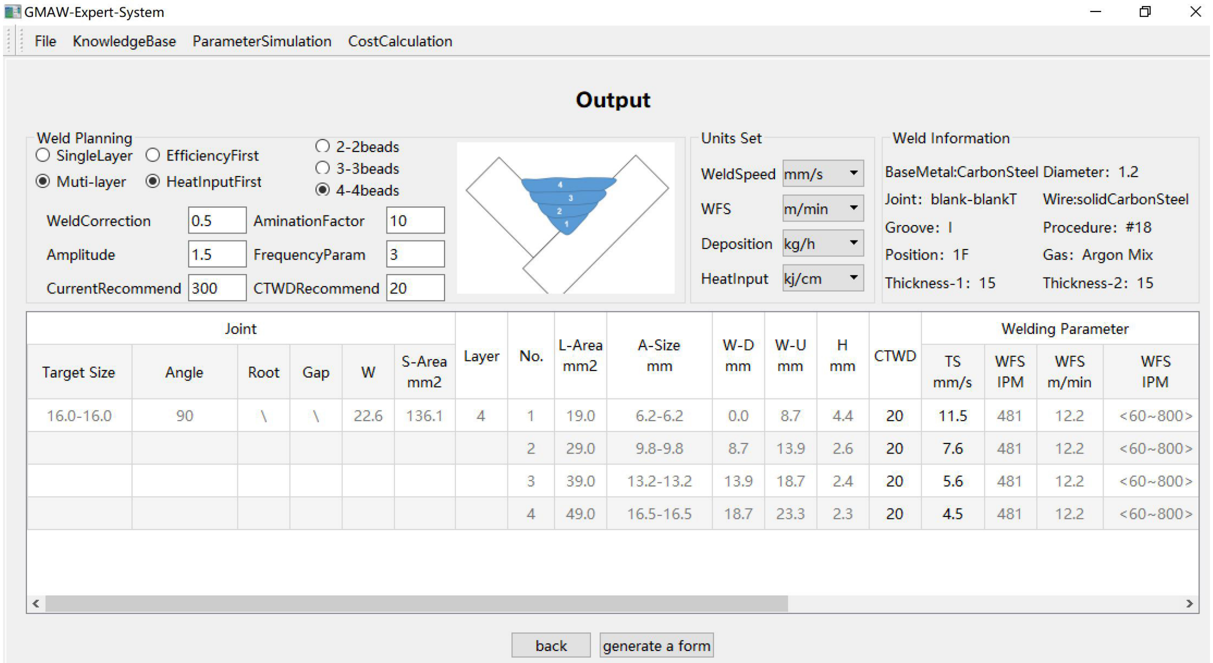

Initial welding information
Joint and groove information
Y
N
Groove display and calculation
Material and other information
Y
N
Dynamic simulation groove
Stratified separation plan
Y
N
Show welding parameters
Save to database
Export to WPS

Figure 9. Work flow chart.



Output

Weld Planning
○ SingleLayer ○ EfficiencyFirst
● Muti-layer ● HeatInputFirst
○ 2-2beads
○ 3-3beads
● 4-4beads

WeldCorrection	0.5	AminationFactor	10
Amplitude	1.5	FrequencyParam	3
CurrentRecommend	300	CTWDRecommend	20

Units Set

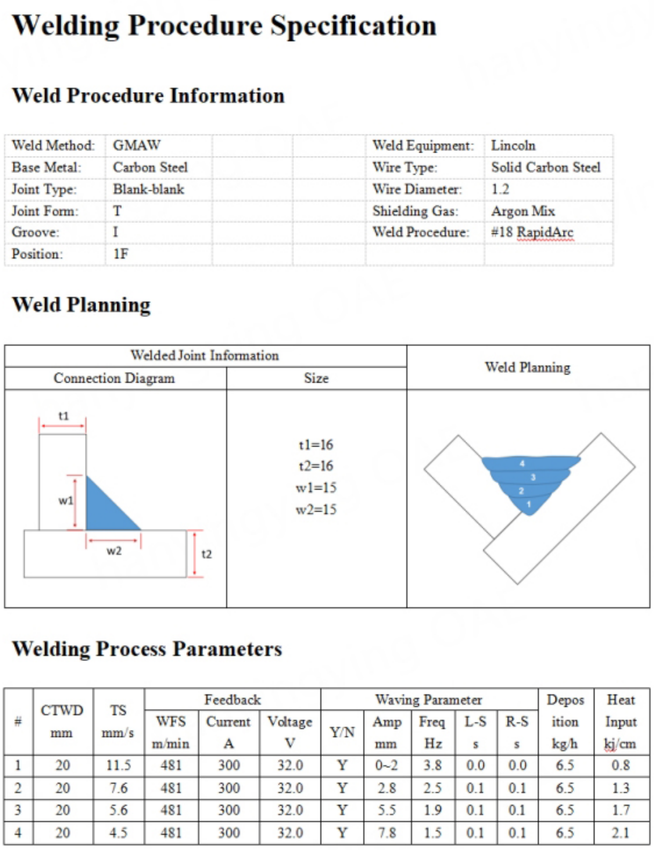
WeldSpeed	mm/s
WFS	m/min
Deposition	kg/h
HeatInput	kj/cm

Weld Information

BaseMetal:CarbonSteel	Diameter: 1.2
Joint: blank-blankT	Wire:solidCarbonSteel
Groove: I	Procedure: #18
Position: 1F	Gas: Argon Mix
Thickness-1: 15	Thickness-2: 15

Joint						Layer	No.	L-Area mm2	A-Size mm	W-D mm	W-U mm	H mm	CTWD	Welding Parameter			
Target Size	Angle	Root	Gap	W	S-Area mm2									TS mm/s	WFS IPM	WFS m/min	WFS IPM
16.0-16.0	90	\	\	22.6	136.1	4	1	19.0	6.2-6.2	0.0	8.7	4.4	20	11.5	481	12.2	<60~800>
							2	29.0	9.8-9.8	8.7	13.9	2.6	20	7.6	481	12.2	<60~800>
							3	39.0	13.2-13.2	13.9	18.7	2.4	20	5.6	481	12.2	<60~800>
							4	49.0	16.5-16.5	18.7	23.3	2.3	20	4.5	481	12.2	<60~800>

Figure 10. Output parameter interface.



Welding Procedure Specification

Weld Procedure Information

Weld Method:	GMAW		Weld Equipment:	Lincoln
Base Metal:	Carbon Steel		Wire Type:	Solid Carbon Steel
Joint Type:	Blank-blank		Wire Diameter:	1.2
Joint Form:	T		Shielding Gas:	Argon Mix
Groove:	I		Weld Procedure:	#18 RapidArc
Position:	1F			

Weld Planning

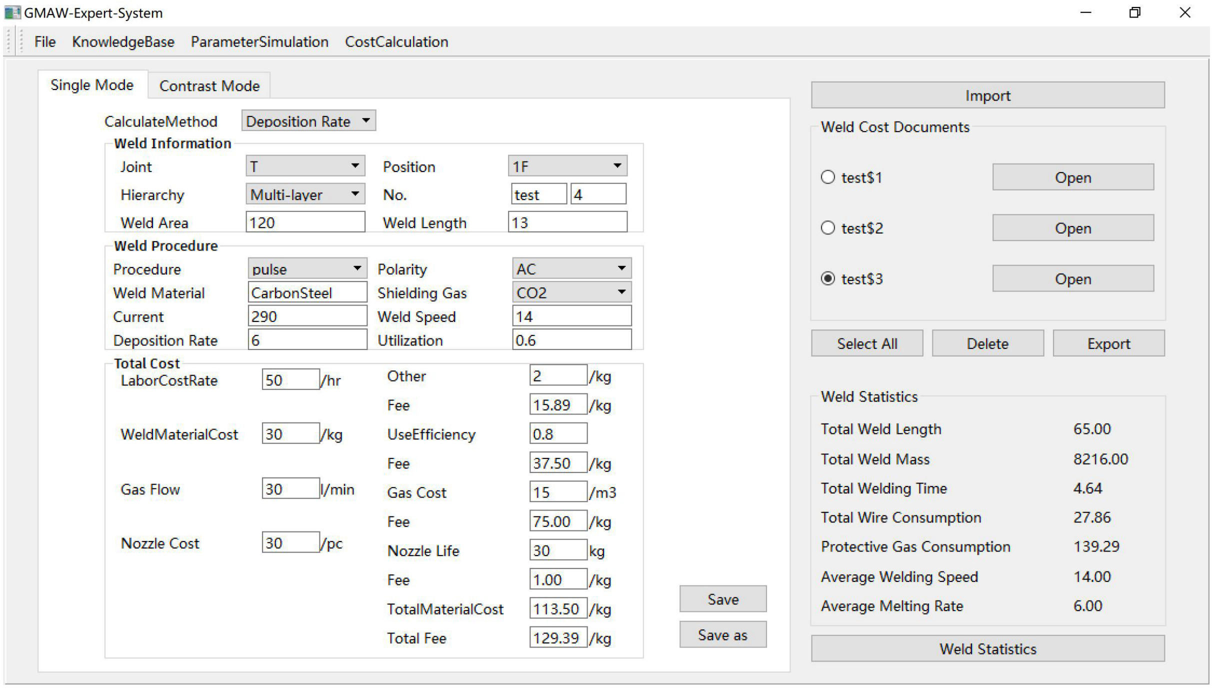
Welded Joint Information		Weld Planning
Connection Diagram	Size	

	t1=16
	t2=16
	w1=15
	w2=15	

Welding Process Parameters

#	CTWD	TS	Feedback			Waving Parameter					Depos	Heat
	mm	mm/s	WFS m/min	Current A	Voltage V	Y/N	Amp mm	Freq Hz	L-S s	R-S s	ition kg/h	Input kj/cm
1	20	11.5	481	300	32.0	Y	0~2	3.8	0.0	0.0	6.5	0.8
2	20	7.6	481	300	32.0	Y	2.8	2.5	0.1	0.1	6.5	1.3
3	20	5.6	481	300	32.0	Y	5.5	1.9	0.1	0.1	6.5	1.7
4	20	4.5	481	300	32.0	Y	7.8	1.5	0.1	0.1	6.5	2.1

Figure 11. The generated WPS.



CalculateMethod: Deposition Rate

Weld Information

Joint	T	Position	1F
Hierarchy	Multi-layer	No.	test 4
Weld Area	120	Weld Length	13

Weld Procedure

Procedure	pulse	Polarity	AC
Weld Material	CarbonSteel	Shielding Gas	CO2
Current	290	Weld Speed	14
Deposition Rate	6	Utilization	0.6

Total Cost

LaborCostRate	50 /hr	Other	2 /kg
		Fee	15.89 /kg
WeldMaterialCost	30 /kg	UseEfficiency	0.8
		Fee	37.50 /kg
Gas Flow	30 l/min	Gas Cost	15 /m3
		Fee	75.00 /kg
Nozzle Cost	30 /pc	Nozzle Life	30 kg
		Fee	1.00 /kg
		TotalMaterialCost	113.50 /kg
		Total Fee	129.39 /kg

Weld Cost Documents

○ test$1
○ test$2
● test$3

Weld Statistics

Total Weld Length	65.00
Total Weld Mass	8216.00
Total Welding Time	4.64
Total Wire Consumption	27.86
Protective Gas Consumption	139.29
Average Welding Speed	14.00
Average Melting Rate	6.00

Figure 12. Cost calculation interface.

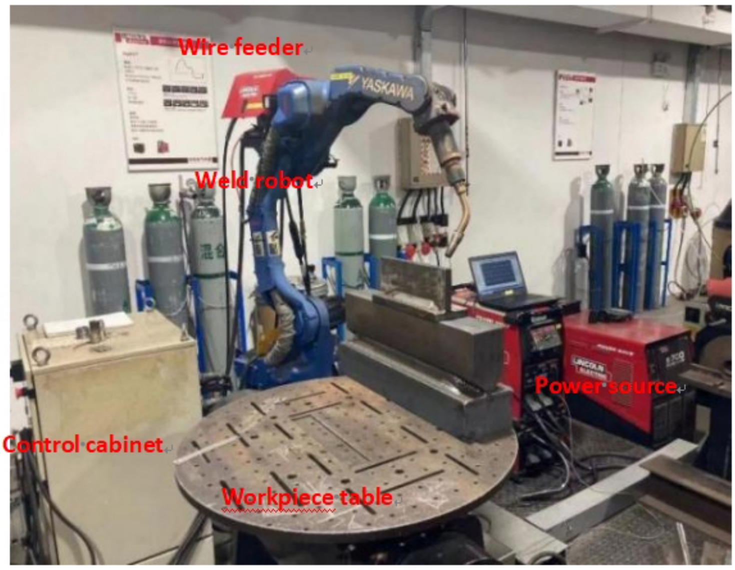


Figure 13. Robot system.

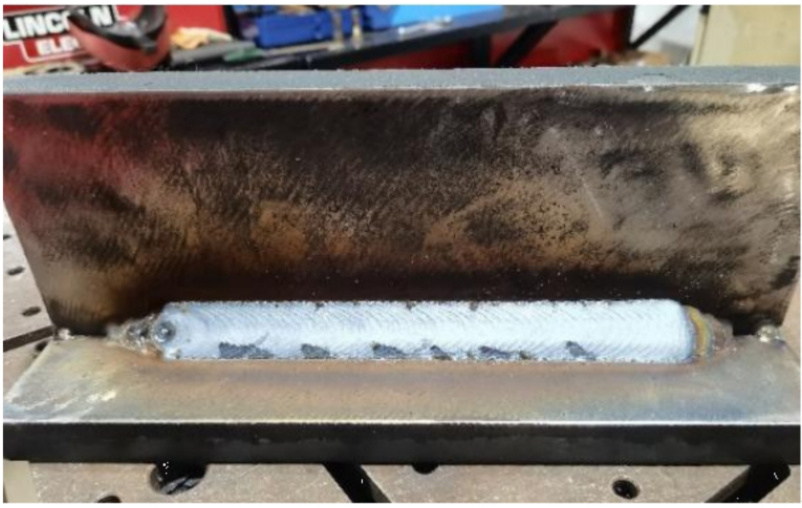


Figure 14. Welding renderings.

knowledge representation, modification of the latter mainly relies on programmers to achieve, and it is hoped that in the future the formation of a welding knowledge representation system for the welding field can be considered. To realize the full automation of knowledge base representation and acquisition and avoid the repeated development of enterprise-oriented and data-oriented expert systems.

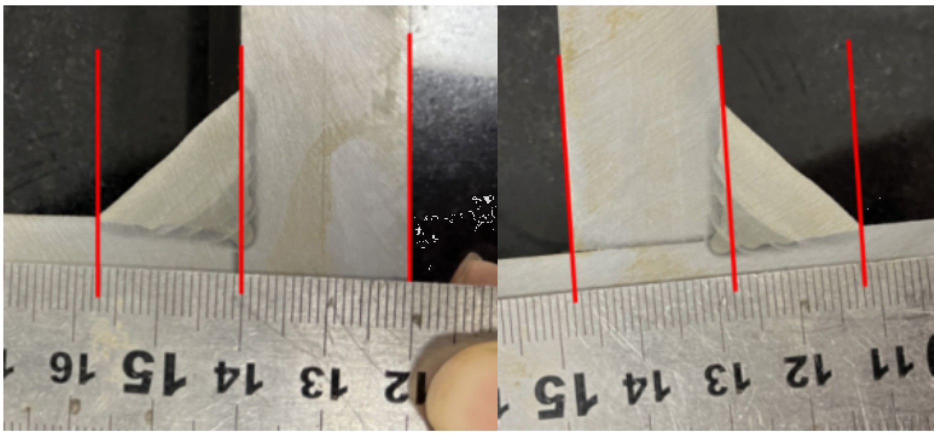


Figure 15. Weld macro section figures.

DECLARATIONS

Authors' contributions

Made substantial contributions to conception and design of the study and performed data analysis and interpretation: Wang X, Chen Q, Yan H

Performed data acquisition, as well as provided administrative, technical, and material support: Sun H, Wang X

Availability of data and materials

In this paper, data comes from references and is mainly provided by enterprises. It constitutes the content of knowledge base and database, including the original welding process parameters and recommended threshold parameters and so on. It is the core data of welding enterprises and cannot be shared.

Financial support and sponsorship

The authors appreciate the support of the National key research and development program (2022YFB4602104), the National Natural Science Foundation of China (62076095, 61973120).

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

All the authors have consented to the publication of this manuscript.

Copyright

REFERENCES

1. Wang B, Li Y, Freiheit T. Towards intelligent welding systems from a HCPS perspective: a technology framework and implementation roadmap. *J Manuf Syst* 2022;65:244-59. DOI
2. Madavi K, Jogi B, Lohar G. Metal inert gas (MIG) welding process: a study of effect of welding parameters. *Mater Today Process* 2022;51:690-8. DOI
3. Huysmans S, Peeters E, De Bruycker E, De Prins K. Weldability study of additive manufactured 316L austenitic stainless steel

components-welding of AM with conventional 316L components. *Weld World* 2021;65:1415-27. DOI

4. Dong Z, Paul S, Tassenberg K, Melton G, Dong H. Transformation from human-readable documents and archives in arc welding domain to machine-interpretable data. *Comput Ind* 2021;128:103439. DOI
5. Trotta G, Cacace S, Semeraro Q. Optimizing process parameters in micro injection moulding considering the part weight and probability of flash formation. *J Manuf Process* 2022;79:250-8. DOI
6. Weng H, Jiang J, Feng M, Yao M, Chen C, Lian G. Multi-objective optimizations of the Q355C steel gas metal arc welding process based on the grey correlation analysis. *Int J Adv Manuf Technol* 2022;121:3567-82. DOI
7. Nguyen DS, Park HS, Lee CM. Optimization of selective laser melting process parameters for Ti-6Al-4V alloy manufacturing using deep learning. *J Manuf Process* 2020;55:230-5. DOI
8. Sparham M, Sarhan AA, Mardi N, Hamdi M, Dahari M. ANFIS modeling to predict the friction forces in CNC guideways and servomotor currents in the feed drive system to be employed in lubrication control system. *J Manuf Process* 2017;28:168-85. DOI
9. Ding D, Shen C, Pan Z, et al. Towards an automated robotic arc-welding-based additive manufacturing system from CAD to finished part. *CAD Comput Aided Des* 2016;73:66-75. DOI
10. Bologna F, Tannous M, Romano D, Stefanini C. Automatic welding imperfections detection in a smart factory via 2-D laser scanner. *J Manuf Process* 2022;73:948-60. DOI
11. Kenda M, Klobčar D, Bračun D. Condition based maintenance of the two-beam laser welding in high volume manufacturing of piezoelectric pressure sensor. *J Manuf Syst* 2021;59:117-26. DOI
12. Liu L, Chen H, Chen S. Quality analysis of CMT lap welding based on welding electronic parameters and welding sound. *J Manuf Process* 2022;74:1-13. DOI
13. Miler D, Hoić M, Žeželj D. Optimisation of welded beams: how cost functions affect the results. *Proc Des Soc: Des Conf* 2020;1:2531-40. DOI
14. Kuklik J, Mente T, Wippo V, et al. Laser welding of additively manufactured thermoplastic components assisted by a neural network-based expert system. In: High-Power Laser Materials Processing: Applications, Diagnostics, and Systems XI, Proceedings of SPIE - The International Society for Optical Engineering. SPIE; 2022 Feb 20-24:The Society of Photo-Optical Instrumentation Engineers (SPIE). DOI
15. Shahriari D, Zolfaghari A, Jahazi M, Bocher P. Development of an expert system to characterize weld defects identified by ultrasonic testing. In: ASME 2013 Pressure Vessels and Piping Conference, PVP 2013 Jul 14-18, 2013. Vol 5. American Society of Mechanical Engineers, Pressure Vessels and Piping Division (Publication) PVP. American Society of Mechanical Engineers (ASME); 2013: Nondestructive Evaluation Engineering Division; Pressure Vessels and Piping Division. DOI
16. Cheng Y, Yu R, Zhou Q, Chen H, Yuan W, Zhang Y. Real-time sensing of gas metal arc welding process - a literature review and analysis. *J Manuf Process* 2021;70:452-69. DOI
17. Cullen M, Zhao S, Ji J, Qiu X. Classification of transfer modes in gas metal arc welding using acoustic signal analysis. *Int J Adv Manuf Technol* 2021;115:3089-104. DOI
18. Lughofer E, Zavoianu A, Pollak R, et al. Autonomous supervision and optimization of product quality in a multi-stage manufacturing process based on self-adaptive prediction models. *J Process Control* 2019;76:27-45. DOI
19. Rønsch GØ, Dybdahl M, Kulahci M. Real-time adjustment of injection molding process settings by utilizing Design of Experiment, time series profiles and PLS-DA. *Qual Eng* 2022;34:215-29. DOI
20. Jagdeesh Patil S. Expert system: a fault diagnosis expert system for high-power industrial production platform. In: Shetty NR, Patnaik LM, Nagaraj HC, Hamsavath PN, Nalini N, editors. Emerging research in computing, information, communication and applications. Singapore: Springer; 2022. p. 317-23. DOI
21. Silva CW. Intelligent robotics - misconceptions, current trends, and opportunities. *Intell Robot* 2021;1:3-17. DOI
22. Su C, Yu G, Wang J, Yan Z, Cui L. A review of causality-based fairness machine learning. *Intell Robot* 2022;2:244-74. DOI
23. Zhang Y, Wei Y, Pu J, Yuan X. Development and application of knowledge-based software for railcar frame welding process. *Int J Adv Manuf Technol* 2021;112:273-84. DOI
24. Liu G, Hong G, Huang M, Xia T, Chen Z. Integrated modelling of automobile maintenance expert system based on knowledge graph. *J Phys: Conf Ser* 2021;1983:012118. DOI
25. Xiao H, Ogai H, Wang W. Multi-channel domain adaptation deep transfer learning for bridge structure damage diagnosis. *IEEJ Transactions Elec Engng* 2022;17:1637-47. DOI
26. Ahmed F, Kim K. Recursive approach to combine expert knowledge and data-driven RSW weldability certification decision making process. *Robot Comput-Integr Manuf* 2023;79:102428. DOI
27. Zhang Y, Wang Q, Liu Y; American Welding Society. Adaptive intelligent welding manufacturing. *Weld J* 2021;100:63-83. DOI
28. Guan K, Yang G, Du L, Li Z, Yang X. Method for fusion of neighborhood rough set and XGBoost in welding process decision-making. *J Intell Manuf* 2023;34:1229-40. DOI
29. Jiang J, Xiong Y, Zhang Z, Rosen DW. Machine learning integrated design for additive manufacturing. *J Intell Manuf* 2022;33:1073-86. DOI
30. Jafarian M, Vahdat SE. A fuzzy multi-attribute approach to select the welding process at high pressure vessel manufacturing. *J Manuf Process* 2012;14:250-6. DOI
31. Koyee RD, Heisel U, Eisseler R, Schmauder S. Modeling and optimization of turning duplex stainless steels. *J Manuf Process* 2014;16:451-67. DOI

32. Martínez R, Alvarez Bestard G, Martins Almeida Silva A, Absi Alfaro SC. Analysis of GMAW process with deep learning and machine learning techniques. *J Manuf Process* 2021;62:695-703. DOI
33. Horváth CM, Botzheim J, Thomessen T, Korondi P. Bead geometry modeling on uneven base metal surface by fuzzy systems for multi-pass welding. *Expert Syst Appl* 2021;186:115356. DOI
34. Farahani S, Khade V, Basu S, Pilla S. A data-driven predictive maintenance framework for injection molding process. *J Manuf Process* 2022;80:887-97. DOI
35. Wang B. A study on spot welding quality judgment based on hidden Markov model. *Proc Inst Mech Eng Part E J Process Mech Eng* 2021;235:208-18. DOI
36. Jiang JC, Yu CL, Xu X, Ma YS, Liu JK. Achieving better connections between deposited lines in additive manufacturing via machine learning. *Math Biosci Eng* 2020;17:3382-94. DOI PubMed
37. Chen T, Guestrin C. XGBoost: a scalable tree boosting system. In: 22nd Acm SIGKDD International Conference on Knowledge Discovery and Data Mining. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Association for Computing Machinery; 2016 Aug 16-17; New York; 2016.p.785-94. DOI

Intelligence & Robotics

Research Article

Open Access

UAV maneuver decision-making via deep reinforcement learning for short-range air combat

Zhiqiang Zheng, Haibin Duan

School of Automation Science and Electrical Engineering, Beihang University, Beijing 100083, China.

Correspondence to: Prof. Haibin Duan, School of Automation Science and Electrical Engineering, Beihang University, No. 37, Xueyuan Road, Haidian District, Beijing 100083, China. E-mail: hbduan@buaa.edu.cn; ORCID: 0000-0002-4926-3202

How to cite this article: Zheng Z, Duan H. UAV maneuver decision-making via deep reinforcement learning for short-range air combat. *Intell Robot* 2023;3(1):76-94. http://dx.doi.org/10.20517/ir.2023.04

Received: 6 Jan 2023 **First Decision:** 16 Feb 2023 **Revised:** 26 Feb 2023 **Accepted:** 9 Mar 2023 **Published:** 18 Mar 2023

Academic Editor: Simon Yang **Copy Editor:** Yin Han **Production Editor:** Yin Han

Abstract

The unmanned aerial vehicle (UAV) has been applied in unmanned air combat because of its flexibility and practicality. The short-range air combat situation is rapidly changing, and the UAV has to make the autonomous maneuver decision as quickly as possible. In this paper, a type of short-range air combat maneuver decision method based on deep reinforcement learning is proposed. Firstly, the combat environment, including UAV motion model and the position and velocity relationships, is described. On this basic, the combat process is established. Secondly, some improved points based on proximal policy optimization (PPO) are proposed to enhance the maneuver decision-making ability. The gate recurrent unit (GRU) can help PPO make decisions with continuous timestep data. The actor network's input is the observation of UAV, however, the input of the critic network, named state, includes the blood values which cannot be observed directly. In addition, the action space with 15 basic actions and well-designed reward function are proposed to combine the air combat environment and PPO. In particular, the reward function is divided into dense reward, event reward and end-game reward to ensure the training feasibility. The training process is composed of three phases to shorten the training time. Finally, the designed maneuver decision method is verified through the ablation study and confrontment tests. The results show that the UAV with the proposed maneuver decision method can obtain an effective action policy to make a more flexible decision in air combat.

Keywords: Short-range air combat, unmanned aerial vehicle, deep reinforcement learning, maneuver decision, proximal policy optimization, flight simulation

1. INTRODUCTION

The unmanned aerial vehicle (UAV) has been applied in many fields for its low cost and high efficiency, including the military domain [1,2]. As sensors, artificial intelligence (AI) and other related technologies are developed and applied to UAV, the serviceable range of UAV in the military has been significantly expanded [3]. Air combat is one of the fields where the UAV is utilized.

The air combat is incredibly complex due to the difficulty of predicting the various scenarios that may arise unpredictably. During the combat, especially in short-range air combat, the UAV performs violent maneuvers which make the combat scenarios change instantly. There are roughly three categories of methods, optimization methods, game theory methods and AI methods, to solve the short-range air combat maneuver decision-making problem [3,4]. For the optimization methods, the maneuver decision problem is turned into an optimization problem, and solved by the optimization theories, like optimization algorithms [4,5]. However, the optimization problem for air combat is a high-dimensional and large-scale problem that is usually so difficult and complex that most optimization-based decision-making algorithms cannot be executed in real-time and adapt to practical constraints. Game theory methods, especially differential games [6,7], are another popular method to solve air combat maneuver decision problems. Whereas, the mathematical models of game theory methods are difficult to establish and their solutions are always hard to prove the adequacy and necessity [4]. For the complex air combat problem, AI methods catch the scholars for their flexibility and operability. The expert system method [8] is one of the AI methods, which tries to map the human knowledge and experience into the flight rule library to complete the maneuver decision. However, the mapping process is complex because human knowledge and experience are always hard to generalize into rules and describe mathematically. On the other hand, once the rule library has been built, the maneuver policy is fixed and inflexible [3]. The methods based on reinforcement learning (RL) are popular for air combat problems recently.

RL is a type of machine learning method that improves its action policy with respect to the reward obtained by repeated trial and error in an interactive environment [9]. In recent years, the neural network has been combined with RL, which is called deep reinforcement learning (DRL). Many types of DRL algorithms have been proposed, like deep Q network (DQN), deep deterministic policy gradient (DDPG), proximal policy optimization (PPO), etc. DRL has been applied in UAV path control [10], quadrupedal locomotion control [11], autonomous platoon control [12], etc. At the same time, DRL has been used to improve the operational efficiency of air combat [9,13,14]. In ref. [3], the DQN is used to solve one-to-one short-range air combat with an evaluation model and maneuver decision model, and basic-confrontation training is presented because of the huge computation load. The PPO is used to learn the continuous 3-DoF short-range air combat strategy in ref. [15], and it can adapt to the combat environment to beat the enemy who uses the minmax strategy. With the DRL method, the UAV can adapt to the changing combat situation and make a reasonable maneuver decision. However, the huge computation load and slow training speed are still the main issues that needs to be addressed when combining DRL with air combat problems.

In this paper, the problem of one-to-one UAV short-range air combat maneuver decision-making is studied. The main contributions are summarized as follows.

(1) The air combat environment with the attacking and advantage areas is designed to describe the relationship between the UAVs. And to increase the confrontment difficulty, the attacking conditions, blood values and the enemy action policy consisting of prediction and decision are introduced.

(2) The GRU layer is applied to design the neural networks which are used as the PPO's actor and critic networks. On the other hand, the observation of the UAV for actor network and combat state information for critic network are designed to separate their roles.

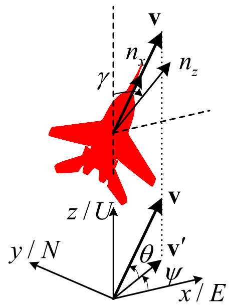


Figure 1. The UAV's motion model in the ground coordinate system. UAV: unmanned aerial vehicle

(3) To improve flexibility and intelligence during the confrontment, the reward function is divided into three parts: dense reward, event reward and end-game reward. Then, the phased training process is designed from easy to difficult to ensure the feasibility of training.

The remainder of this paper is organized as follows. In section II, the UAV motion model, air combat environment and its process are introduced. Then the method designed is explained in section III, which includes the PPO algorithm, some improved points and the design for state, action and reward function to combine the PPO algorithm with the air combat problem. The action policy for the enemy and training process is also introduced. Next, the training results and simulation analysis are presented in section IV. Finally, the conclusion is presented in section V.

2. PROBLEM FORMULATION

2.1. UAV motion model

The three-degree-of-freedom UAV model is considered because the main consideration in short-range air combat problem is the position and velocity relationship between the two sides[3]. The motion model is established in the ground coordinate system as East-North-Up (ENU) coordinate system, which is shown in Figure 1.

To simplify the problem, assume that the velocity direction is fixed with the -axis of the body coordinate system, and the UAV's motion model is shown as[14,16]

$$\begin{cases} \dot{x} = v\cos\theta\cos\psi \\ \dot{y} = v\cos\theta\sin\psi \\ \dot{z} = v\sin\theta \\ \dot{v} = g\left(n_x - \sin\theta\right) \\ \dot{\theta} = \dfrac{g}{v}\left(n_z\cos\gamma - \cos\theta\right) \\ \dot{\psi} = \dfrac{g n_z \sin\gamma}{v\cos\theta} \end{cases}, \tag{1}$$

where x, y and z are the UAV's position coordinate values and $\mathbf{p} = [x, y, z]^T$, $v = \|\mathbf{v}\|$ is the velocity, $\dot{x}$, $\dot{y}$ and $\dot{z}$ are the values of $\mathbf{v}$ on the ground coordinate axes, γ, θ and ψ represent the flight-path bank angle, flight-path angle and heading angle respectively, g is the acceleration of gravity, n_x is the overload in velocity direction, and n_z is the normal overload. Noting that the heading angle ψ is the angle between **v'**, the projection of **v** on the xoy plane, and ox axis. The basic control parameters n_x, n_z and γ in the motion model can be expressed as

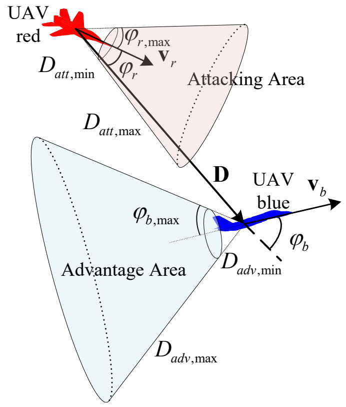


Figure 2. The relationship between the red and blue sides during the battle.

a control input vector $\mathbf{u} = [n_x, n_z, \gamma]^T \in \mathbb{R}^3$, which can be used to control the UAV's position and velocity in the air combat maneuver decision making[16]. At the same time, the UAV's motion must satisfy the constraints which are expressed as[3]

$$\begin{cases} v_{\min} \leqslant v \leqslant v_{\max} \\ \theta_{\min} \leqslant \theta \leqslant \theta_{\max} \\ -\pi < \gamma \leqslant \pi \\ 0 \leqslant \psi < 2\pi \\ n_{x\min} \leqslant n_x \leqslant n_{x\max} \\ n_{z\min} \leqslant n_z \leqslant n_{z\max} \end{cases}, \tag{2}$$

where subscript *min* and *max* mean the minimum and maximum values, and the parameters are set as $v_{\min} = 30m/s$, $v_{\max} = 150m/s$, $\theta_{\min} = -\pi/4$, $\theta_{\max} = \pi/4$, $n_{x\min} = -1$, $n_{x\max} = 2.5$, $n_{z\min} = -4$ and $n_{z\max} = 4$. These constraints are judged and processed when the control input vector $\mathbf{u}$ is input to the UAV motion model and the UAV state, including v, θ and ψ, is updated.

By giving the control parameters' values and UAV's state at the time step t, the UAV's state at the time step $t+1$ can be easily obtained by the Runge-Kutta method[14].

2.2. Air combat environment

In the one-to-one short-range air combat environment, there are two UAVs, divided into red and blue. The red aims to gain the advantage situation over the blue until the blue side is destroyed by its weapon, and the blue aims to do the opposite[4]. In this paper, the red UAV is controlled by the proposed decision method based on DRL algorithm. The relationship between the red and blue sides during the battle, which is shown in Figure 2, is mainly described by both sides' velocity vectors, the red's velocity $\mathbf{v}_r$ and the blue's velocity $\mathbf{v}_b$, and the relative position vector $\mathbf{D}$, which can be expressed as

$$\mathrm{D} = \mathbf{p}_b - \mathbf{p}_r, \tag{3}$$

where $\mathbf{p}_r$ and $\mathbf{p}_b$ are the red and blue's position vectors respectively.

The angle φ_r, named the attacking angle, is between $\mathbf{v}_r$ and $\mathbf{D}$, and the formula can be expressed as[16]

$$\varphi_r = \arccos \frac{\mathbf{v}_r \cdot \mathbf{D}}{\|\mathbf{v}_r\| \cdot \|\mathbf{D}\|}. \tag{4}$$

During the confrontation, the red has a chance to attack and deal damage to the blue only if the blue is in its attacking area. φ_r can be used to describe whether the blue in the red's attacking area and the conditions can be described as

$$\begin{cases} D_{att,\min} \leqslant \|\mathbf{D}\| \leqslant D_{att,\max} \\ \varphi_r \leqslant \varphi_{att,\max} \end{cases}, \tag{5}$$

where $D_{att,\min}$ and $D_{att,\max}$ are the minimum and maximum attacking distances, and $\varphi_{att,\max}$ is the maximum attacking angle that has a chance to deal damage.

The angle φ_b, named escaping angle, is between $\mathbf{v}_b$ and $\mathbf{D}$, and the formula is expressed as[16]

$$\varphi_b = \arccos \frac{\mathbf{v}_b \cdot \mathbf{D}}{\|\mathbf{v}_b\| \cdot \|\mathbf{D}\|}. \tag{6}$$

During the confrontation, the red not only needs to keep the blue in its attacking area but also tries to avoid being attacked by the blue. Thus, the advantage area for the red is defined behind the blue and can be described as

$$\begin{cases} D_{adv,\min} \leqslant \|\mathbf{D}\| \leqslant D_{adv,\max} \\ \varphi_b \leqslant \varphi_{esp,\max} \end{cases}, \tag{7}$$

where $D_{adv,\min}$ and $D_{adv,\max}$ are the minimum and maximum advantage distances, and $\varphi_{esp,\max}$ is the maximum escaping angle that the red is in its advantage area.

During air combat, the UAV has limited attacking resource and needs attack only under certain conditions. The enemy should be both within the red's attacking area and hard to escape. These conditions can be described as[15]

$$\begin{cases} D_{att,\min} \leqslant \|\mathbf{D}\| \leqslant D_{att,\max} \\ \varphi_r \leqslant \varphi_{att,\max} \\ \varphi_b \leqslant \varphi_{esp,att} \end{cases}, \tag{8}$$

where $\varphi_{esp,att}$ is the maximum escape angle for the enemy if the red wants to make a successful attack. On the other hand, not every attack could cause damage to the other side, and it will probably take more than an attack to destroy the other side. Thus, it is assumed that the UAVs have blood value B_0 when initialized, and every attack has the probability p_{att} to cause some damage ΔB, which is shown as

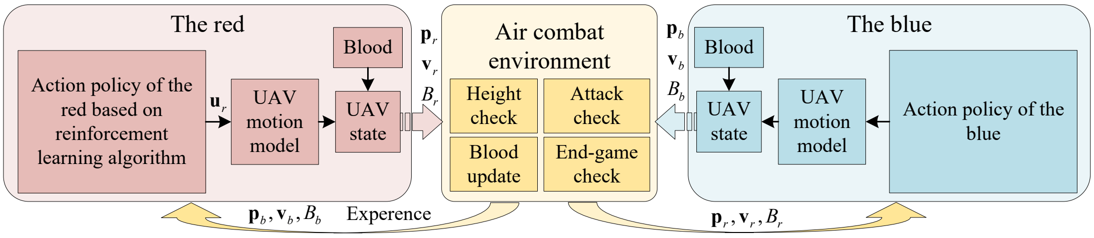


Figure 3. Short-range air combat framework.

$$\Delta B = \begin{cases} -B_1 & 0 \leq p < p_{att1} \\ -B_2 & p_{att1} \leq p < p_{att2} \\ -B_3 & p_{att2} \leq p < p_{att} \\ 0 & p \geq p_{att} \end{cases}, \tag{9}$$

where $p \in [0, 1]$ is a random number, B_1, B_2, B_3 are the reduced blood values after an attack, and p_{att1} and p_{att2} are the threshold of probability.

Note that Figure 1 is shown from the red perspective, and the relative relationship between the two sides can also be defined from the blue perspective in the same way. In the view of the blue, distance vector $\mathbf{D}'$, attacking angle φ'_b, and escaping angle φ'_r are defined like Equations (4-8), and the attacking conditions are defined as

$$\begin{cases} D_{att,\min} \leqslant \|\mathbf{D}'\| \leqslant D_{att,\max} \\ \varphi'_b \leqslant \varphi_{att,\max} \\ \varphi'_r \leqslant \varphi_{esp,\max} \end{cases}. \tag{10}$$

2.3. Air combat process

For an episode of one-to-one short-range air combat, the red and the blue are initiated, then confront in the combat environment until the conditions for the end of the episode are satisfied[17]. An episode ends when the red or the blue is damaged or the maximum decision step $t_{\max}$ is reached. If the UAV is out of the range of height or its blood value B is reduced to zero, this UAV will be judged as damaged. There are three types of results for the red, win, loss and draw, at the end of an episode. The red wins over the blue only when the blue is damaged before reaching the $t_{\max}$. Similarly, the red is lost when it is damaged before the blue within $t_{\max}$ time steps. The draw result means that both sides are still undamaged when the maximum decision step $t_{\max}$ is reached. The duration for each step is t_{step}. At every step in an episode, the red and the blue will make a maneuver decision by its action policy respectively to get the $\mathbf{u}_r$ and $\mathbf{u}_b$. During a step, the UAVs will continually execute $\mathbf{u}_r$ and $\mathbf{u}_b$ until another maneuver decision-making starts. The flag *done* shows when the episode ends. In this paper, the action policy of the red is actor network which is updated by the PPO algorithm, and the blue's action policy is described in section 3.6. The air combat framework is shown in Figure 3 and the process in an episode is shown in Algorithm 1.

3. AIR COMBAT DECISION METHOD DESIGN

3.1. PPO algorithm and improvement

The PPO algorithm is a type of DRL algorithm that has been used in many types of problems. In this part, the basic PPO algorithm and its usage in short-range air combat are introduced.

Algorithm 1 Air combat process in an episode

```
Input: position: $\mathbf{p}_r$, $\mathbf{p}_b$; velocity: $\mathbf{v}_r$, $\mathbf{v}_b$; blood: $B_r = B_b = B_0$; number of steps: $t_{\max}$; step: $t = 1$; damage:
    $dam_r = dam_b = False$; height range: $[0, z_{\max}]$
Output: confrontation result in the view of the red
for all $t \leqslant t_{\max}$ do
    set $done = False$
    use the red's action policy to get $\mathbf{u}_r$
    use the blue's action policy to get $\mathbf{u}_b$
    update the red's and the blue's positions and velocities by Equation (1)
    if $z_r \notin [0, z_{\max}]$ or $B_r \leqslant 0$ then
        set $B_r = 0$ and $dam_r = True$
    end if
    if $z_b \notin [0, z_{\max}]$ or $B_b \leqslant 0$ then
        set $B_b = 0$ and $dam_b = True$
    end if
    if $dam_r$ is $False$ then
        calculate $\varphi_r$ and $\varphi_b$ in the view of the red by Equations (4) and (6)
        if satisfy Equation (8) then
            get a random number $p \in [0, 1]$
            calculate $\Delta B$ by Equation (9)
            set $B_b = B_b + \Delta B$
        end if
    end if
    if $dam_b$ is $False$ then
        calculate $\varphi_r'$ and $\varphi_b'$ in the view of the blue
        if satisfy Equation (10) then
            get a random number $p \in [0, 1]$
            calculate $\Delta B$ by Equation (9)
            set $B_r = B_r + \Delta B$
        end if
    end if
    if $dam_r$ is $False$ and $dam_b$ is $True$ then
        set $done = True$
        return red win
    else if $dam_r$ is $True$ and $dam_b$ is $False$ then
        set $done = True$
        return red loss
    else if $dam_r$ is $True$ and $dam_b$ is $True$ then
        set $done = True$
        return tie
    else if $t = t_{\max}$ then
        set $done = True$
        return tie
    else
        set $t = t + 1$
    end if
end for
```

3.1.1. The PPO algorithm

The PPO algorithm is based on the actor-critic framework and the policy gradient method (PG), which can be applied to continuous or discrete motion space problems.[18] PG-based algorithms maximize the action policy's expected return by updating the action policy directl[19]. The PPO algorithm's main objective is[18]

$$L(\theta) = \mathbb{E}\left[L_{actor}(\theta) - c_1 L_{critic}(\theta) + c_2 S_\theta(o_t)\right], \tag{11}$$

where L_{actor} and L_{critic} are the loss function for actor network and critic network, θ is the parameters of networks, c_1 and c_2 are coefficients, S_θ is the entropy bonus which is used to ensure sufficient exploration, and o_t is the observation of the actor network[18]. By considering these terms, the loss calculated by Equation (11) is related to the parameters of actor and critic networks. L_{actor} is defined as

$$L_{actor}(\theta) = \mathbb{E}\left[\min\left(r(\theta)\hat{A}, \text{clip}\left(r(\theta), 1-\varepsilon, 1+\varepsilon\right)\hat{A}\right)\right], \tag{12}$$

where $r(\theta) = \pi_\theta(a_t|o_t)/\pi_{\theta old}(a_t|o_t)$ is the probability ratio. It is the probability of a_t with o_t under latest action policy π_θ and action policy before update $\pi_{\theta old}$, is the clipping function which is to ensure the policy from $\pi_{\theta old}$ to π_θ doesn't change too much. $\hat{A}$ is the estimator of the advantage function, and the generalized advantage estimation (GAE) is used to calculate it[20]. The L_{critic} is defined as[18]

$$L_{critic}(\theta) = \frac{1}{2}\left(V_\theta(s) - R(s)\right)^2, \tag{13}$$

where V_θ is the state-value function, which means the critic network in the PPO algorithm, and R is the return.

In this paper, the red's action policy is based on the PPO algorithm. Thus, $\mathbf{u}_r$ is generated based on the actor network. The output of actor network is a probability distribution $dist$, which is the selection probability of each action for the red under the observation. Then, the action a is sampled from the $dist$, which means the index of selected action in the action space[21]. Finally, $\mathbf{u}_r$ can be generated by the designed action space. Noting that the $\log\_prob$ is the logarithm of $dist$. The usage of the PPO algorithm in short-range air combat is shown as Algorithm 2[21].

3.1.2. Improved points

To improve the training effect, some improvement points are adopted in this paper, and the framework of PPO algorithm for short-range air combat is shown in Figure 4.

The first improvement point is considering the historical combat data when making decision. During the confrontation, the red must gradually accumulate the situational advantages over the blue and finally beat it. Therefore, decision on current action should take into account the previous air combat situations. The link between the action decision and historical air combat experience is established by adding the gate recurrent unit (GRU)[22] to the neural network. The GRU is a type of recurrent neural network (RNN), which can adaptively capture the time's dependencies in different scales[22], similar to long short-term memory (LSTM). However, the GRU is easier to train and more efficient than LSTM. The GRU layer, the hidden layer using GRU, is used in both actor network and critic network. The networks' inputs, observation and state, are firstly processed by the fully connected network to extract the inputs' features. Then the features are fused with historical features by the GRU layer to obtain the integrated features considering the historical situational features. The specific process is shown in Figure 4.

Algorithm 2 The usage of the PPO algorithm in short-range air combat

```
initialize the PPO's hyperparameters, including epoch $K$, the number of the minimum experience for training $N$, etc.
initialize the air combat environment including the UAVs' positions, velocities, blood values, etc.
initialize the number of experiences in experience buffer $i = 0$
for all $episode \in training\_episodes$ do
    execute the process shown in Algorithm 1, including updating the UAVs' positions, velocities, blood values and damage flags. For each time step, $o$, state and reward are generated, and the experiences are generated, then set $i = i + 1$
    if $i \geq N$ then
        calculate the return for every step and normalize the advantage
        set $k = 1$
        for all $k \leq K$ do
            sample from the experience buffer based on the batch size
            calculate the loss of each batch by Equation (11)
            update the networks' parameters by Adam optimizer
            set $k = k + 1$
        end for
        set $i = 0$ and clear the experience buffer
    end if
end for
```

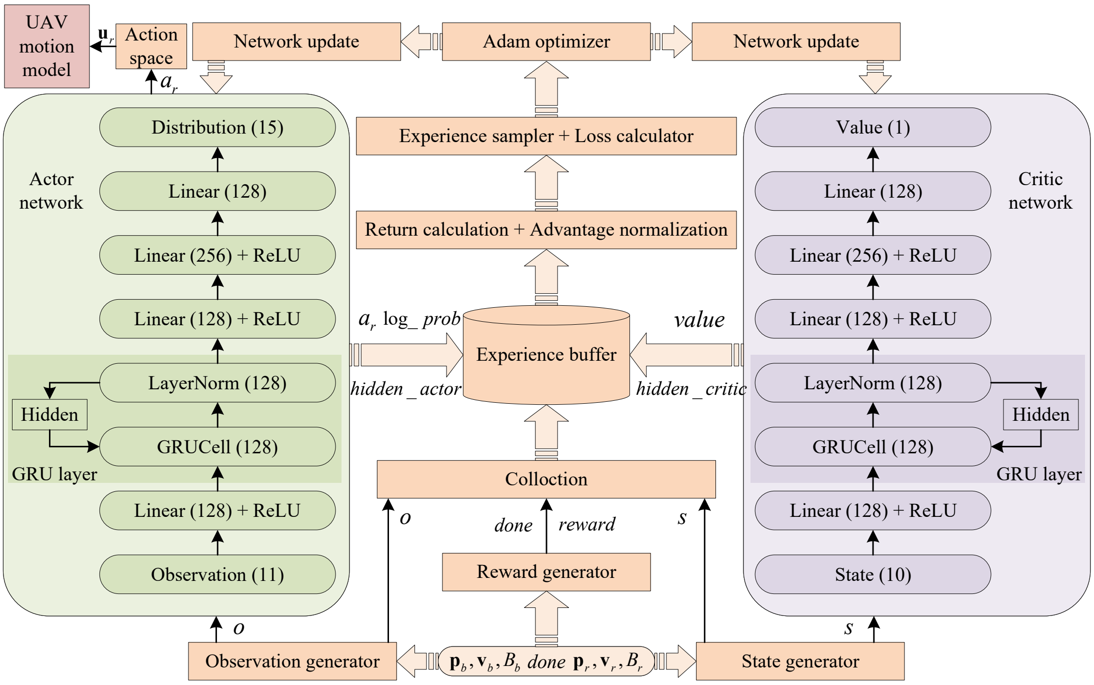


Figure 4. The PPO algorithm framework for short-range air combat.

The second improvement point is to differentiate the neural network inputs. For the basic PPO algorithm, the critic network uses the same input as the actor network, which is named state space[20]. However, the actor and critic networks play different roles in the algorithm. The actor network's state space is in the view of the red because the actor network's input is the red's observation of the air combat environment. On the other hand, the critic network is used to evaluate the output of the actor network by its output, the critic value, based on the current air combat situation[23]. Thus, the input of critic network can be more objective and include some information that cannot be observed by the red. In this vein, the observation o from the observation generator and the state s from the state generator are designed for actor network and critic network respectively, as shown in Figure 4.

The third improvement point is a variable-sized experience buffer. For each time step, the experiences are stored in the experience buffer in order. During the training, it is divided into thousands of episodes, and each episode is over if it satisfies the ending condition. Thus, the length of each episode may be different. To make the training more general, the network parameters are updated until enough experiences are stored. However, the return calculation should be calculated on the basis of the complete experiences. Therefore, when the networks are updated, the numbers of experiences, which are larger than the number of the minimum experience for training N, are different to make sure that the same episode's experiences are stored in the buffer.

The fourth improvement point is the phased training process, which is discussed in section 3.6.

3.2. State space design

The state space of air combat should contain the information of the red and the blue UAVs, and a suitable designed state space can speed up the convergence of training. In this part, two state spaces designed for actor network and critic network are introduced.

3.2.1. *The actor network's state space*

The designed state space consists of two parts, position information s_{pos} and velocity information s_{vel} which is expressed as

$$s_{actor} = \left[s_{pos}^T, s_{vel}^T\right]^T. \tag{14}$$

The s_{pos} is the position relationship between the UAVs, which is defined as[24]

$$s_{pos} = [x_r - x_b, y_r - y_b, z_r, D, \theta_D, \psi_D]^T, \tag{15}$$

where the subscript r and b represent the information for the red and blue, $D = \|\mathbf{D}\|$, and θ_D and ψ_D are the flight-path angle and heading angle for $\mathbf{D}$ respectively, which are similar to the θ and ψ for $\mathbf{v}$ in Figure 2. The θ_D and ψ_D are described as[14]

$$\begin{aligned} \theta_D &= \arcsin\frac{z_b - z_r}{D}, \\ \psi_D &= \begin{cases} \operatorname{atan2}(y_b - y_r, x_b - x_r) & \operatorname{atan2}(y_b - y_r, x_b - x_r) > 0 \\ 2\pi + \operatorname{atan2}(y_b - y_r, x_b - x_r) & \operatorname{atan2}(y_b - y_r, x_b - x_r) < 0 \end{cases}, \end{aligned} \tag{16}$$

where $\operatorname{atan2}(y, x) \in [-\pi, \pi]$ returns the angle between $[x, y]^T$ and x axis. The s_{vel} is the velocity relationship between the UAVs, which is defined as[14]

$$s_{vel} = [\dot{x}_r - \dot{x}_b, \dot{y}_r - \dot{y}_b, \dot{z}_r - \dot{z}_b, \varphi_b, \varphi_r]^T. \tag{17}$$

At the same time, to avoid the difference between the values of each state variable from being too large to affect the learning efficiency of the network, the normalization for every state variable is adopted. The threshold vector δ_{actor} for state variables is selected as

$$mathbf\delta_{actor} = [D_{th}, D_{th}, H_{th}, D_{th}, \pi, 2\pi, v_{\max} - v_{\min}, v_{\max} - v_{\min}, v_{\max} - v_{\min}, \pi, \pi]^T, \tag{18}$$

where D_{th} and H_{th} are the threshold values for distance and height. Noting that the elements of δ_{actor} and s_{actor} correspond one to one. Then, the normalization is executed by[3]

$$o_i = \frac{s_{actor,i}}{\delta_{actor,i}} \cdot a - b \quad i = 1, 2, \cdots, 11, \tag{19}$$

where o is the normalized state vector, which is usually called the observation of the actor network. a and b are constants and satisfy $a = 2b$.

3.2.2. *The critic network's state space*

The actor network gets action according to the relationship in the view of the red, but the critic network gets the evaluation value based on the state of the air combat environment, which can include the information that cannot be observed. Thus, the input for the critic network s_{critic} is defined as[17]

$$s_{critic} = [D, \varphi_r, \varphi_b, z_b - z_r, \psi_D, \psi_{br}, \theta_{br}, v_r - v_b, B_r, B_{rb}]^T, \tag{20}$$

where subscript rb means the red's value minus the blue's and br is on the contrary, and B_r is the red's residual blood. The critic network's state variables are also normalized and the threshold vector δ_{critic} is selected as

$$\delta_{critic} = [D_{th}, \pi, \pi, H_{th}, 2\pi, \pi, \pi, v_{\max} - v_{\min}, B_0, B_0/2]^T. \tag{21}$$

Then, the s_{critic} is normalized like Equation (19), and the normalized result is s.

3.3. Action space design

In the air combat problem, the UAV's actions are always summed up as a maneuver library, which consists of a series of tactical actions, such as high yo-yo, cobra maneuvering and so on[3]. Pilots can choose from the library according to the combat situation. However, the establishment of the library is difficult and complex, and these tactical actions can be disassembled into basic actions. Thus, fifteen basic actions $a_1, a_2, \cdots, a_{15}$ are adopted to form the action space A, which includes five types of directions, forward, upward, downward, left turn and right turn, and three types of speed control, maintenance, acceleration and deceleration[25]. Every basic action in the designed library is a set of values of control parameters, $[n_x, n_z, \gamma]^T$, for the UAV motion model. The designed maneuver library is shown in Table 1. Therefore, the action space is discrete and its dimension is 15.

3.4. Reward function design

The aim of RL is to maximize the cumulative reward obtained from the environment. Therefore, the reward function is the bridge to communicate the training result requirements to the DRL algorithm and its design is extremely important[25]. In this paper, the reward function is well-designed and divided into three parts:

Table 1. The basic action's values in the designed action space

No.	Action	Values for $[n_x, n_z, \gamma]^T$	No.	Action	Values for $[n_x, n_z, \gamma]^T$
1	Forward, maintain	0; 1; 0	2	Forward, accelerate	2; 1; 0
3	Forward, decelerate	-1; 1; 0	4	Upward, maintain	0; 3.5; 0
5	Upward, accelerate	2; 3.5; 0	6	Upward, decelerate	-1; 3.5; 0
7	Downward, maintain	0; -3.5; 0	8	Downward, accelerate	2; -3.5; 0
9	Downward, decelerate	-1; -3.5; 0	10	Left turn, maintain	0; 3.5; arccos (2/7)
11	Left turn, accelerate	2; 3.5; arccos (2/7)	12	Left turn, decelerate	-1; 3.5; arccos (2/7)
13	Right turn, maintain	0; 3.5; - arccos (2/7)	14	Right turn, accelerate	2; 3.5; - arccos (2/7)
15	Right turn, decelerate	-1; 3.5; - arccos (2/7)			

dense reward, event reward and end-game reward. Different types of rewards are triggered under different conditions to transmit different expectations. In this part, the reward function designed for short-range air combat is introduced.

3.4.1. Dense reward

The red receives a dense reward from the air combat environment after completing the action for every step. Thus, a properly designed dense reward can improve the red's exploration efficiency and speed up the training process. The dense reward is based on the air combat situatio[16] after the execution of the red's and blue's actions and can be considered as the immediate situation value for the maneuvering decision-making. The dense reward r_{dense} is defined as

$$r_{dense} = (w_a r_a + w_d r_d + w_h r_h + w_v r_v - 1) \cdot w_{dense}, \tag{22}$$

where r_a, r_d, r_h and r_v are the angle reward, distance reward, height reward and velocity reward respectively, and w_a, w_d, w_h, w_v and w_{dense} are the weights. By giving a negative reward as a penalty term at every step, the red will try to find ways to reduce the penalty by trying its best to increase r_a, r_d, r_h and r_v as quickly as possible. Thereby, the red can be trained more efficiently. r_a represents the evaluation value of the azimuth relationship between the red and blue, and is defined as

$$r_a = \frac{\pi - \varphi_r}{\pi} \cdot \frac{\pi - \varphi_b}{\pi}. \tag{23}$$

r_d represents how good the distance relationship is, which consists of two parts, r_{d1} and r_{d2}, and satisfies $r_d = r_{d1} + r_{d2}$. r_{d1} is defined as

$$r_{d1} = \begin{cases} 0.25 & \Delta D < 0, D > D_{mid} \\ 0 & other \end{cases}, \tag{24}$$

where ΔD is the distance difference from the previous time, and $D_{mid} = \left(D_{att,\min} + D_{att,\max}\right)/2$ is the desired distance during air combat. This means the red can receive the reward r_{d1} only when the distance D is larger than D_{mid} and the red is closer to the blue than the previous time. r_{d2} is defined as[16]

$$r_{d2} = \begin{cases} 0.25 \cdot \left(a_{D1}\left(D - D_{adv,\max}\right)^2 + 1\right) & D_{adv,\max} < D \leqslant D_{th} \\ 0.25 + 0.25 \cdot \left(a_{D2}\left(D - D_{att,\max}\right)^2 + 1\right) & D_{att,\max} < D \leqslant D_{adv,\max} \\ 0.5 + 0.25 \cdot a_{D3}\left(D - D_{att,\min}\right)\left(D - D_{att,\max}\right) & D_{att,\min} < D \leqslant D_{att,\max} \\ 0 & other \end{cases}, \tag{25}$$

where a_{D1}, a_{D2} and a_{D3} are the parameters and defined as

$$\begin{aligned} a_{D1} &= -1\Big/\left(D_{th} - D_{adv,\max}\right)^2 \\ a_{D2} &= -1\Big/\left(D_{adv,\max} - D_{att,\max}\right)^2 \\ a_{D3} &= 1/\left(\left(D_{mid} - D_{att,\min}\right)\left(D_{mid} - D_{att,\max}\right)\right) \end{aligned} . \tag{26}$$

r_h represents the height advantage and is defined as

$$r_h = \begin{cases} 0.1 & H_{\max} < z_r - z_b < D_{att,\max} \\ h_1(z_r - z_b - H_{adv})^2 + 1 & H_{adv} < z_r - z_b \leqslant H_{\max} \\ 1 & H_{att} < z_r - z_b \leqslant H_{adv} \\ h_2(z_r - z_b - H_{att})^2 + 1 & H_{\min} < z_r - z_b \leqslant H_{att} \\ 0 & other \end{cases} , \tag{27}$$

where $H_{\max}$, H_{adv}, H_{att} and $H_{\min}$ are maximum desired height, desired advantage height, desired attacking height and minimum desired height during the combat, and they satisfy $H_{\max} > H_{adv} > H_{att} > H_{\min}$. h_1 and h_2 are parameters that are defined as

$$\begin{aligned} h_1 &= -0.9\Big/(H_{\max} - H_{adv})^2 \\ h_2 &= -1\Big/(H_{\min} - H_{att})^2 \end{aligned} . \tag{28}$$

As for r_v, it is the evaluation of the velocity for both sides and is defined as

$$r_v = \begin{cases} 0.1 & v_r/v_b > 1.5 \\ 1 & 1.0 \leqslant v_r/v_b < 1.5 \\ 5 \cdot v_r/v_b - 4 & 0.8 \leqslant v_r/v_b < 1.0 \\ 0 & other \end{cases} . \tag{29}$$

3.4.2. Event reward

During air combat, there are many types of events[24,26], such as attacking successfully, reaching the advantage area, making the enemy in the attacking area and so on. By continuously triggering these events, the red will beat the blue finally. Thus, the event reward is necessary to make the red consciously trigger these events to maintain the advantage. This paper designs two types of event rewards: advantage area reward r_{adv} and attacking reward r_{att}. r_{adv} encourages the red for getting in the advantage area, and is defined as

$$r_{adv} = w_{adv} \cdot \left(0.6 \cdot \frac{D_{adv,\max} - D}{D_{adv,\max} - D_{adv,\min}} + 0.4 \cdot \frac{\pi - \varphi_r}{\pi}\right), \tag{30}$$

where w_{adv} is the weight. By giving a changeable r_{adv}, the red is encouraged to keep in the advantage area and get closer to the blue. Analogously, the blue can also be in the blue's advantage area, which is harmful to the red. In this situation, the penalty r'_{adv}, which satisfies $r'_{adv} < 0$ and $|r'_{adv}| > w_{adv}$, will be given to the red to encourage it to turn over the situation as soon as possible. When the red succeeds to attack the blue, r_{att} will reward it. r_{att} is a positive const number. On the other hand, if the red is attacked by the blue, it will get the penalty r'_{att} which satisfies $r'_{att} < 0$ and $|r'_{att}| > r_{att}$.

3.4.3. End-game reward

During the training, the draw result is regarded as a case in which the red loses and is used to motivate the red to beat the blue. When the flag $done = True$, which means an episode is over, the end-game reward will be generated, and is defined as[25]

$$r_{end} = \begin{cases} r_0 + r_1 \cdot \dfrac{t_{\max} - t}{t_{\max}} + r_2 \cdot \dfrac{B_r}{B_0} & \text{win} \\ -r_{loss} & \text{loss} \end{cases}. \tag{31}$$

where r_0, r_1, r_2 and r_{loss} are positive numbers, and satisfy $r_0 + r_1 + r_2 \leqslant r_{loss}$.

3.5. Action policy for the Blue

During the training, a policy is adopted as the blue's action policy, which consists of prediction and decision. In the prediction step, the blue will predict which action the red will do at the next time step and then estimate the red's position and velocity at the next time step based on the predicted action. At the decision step, the blue will find which action it should take to confront the red. To find which action is better, the thread function T is defined[27], which consists of angle thread T_φ, velocity thread T_v, distance thread T_d and height thread T_h. Hence, T is calculated by

$$T = 0.41 \cdot T_\varphi + 0.26 \cdot T_v + 0.19 \cdot T_d + 0.14 \cdot T_h. \tag{32}$$

Noting that the definitions of T_φ, T_v, T_d and T_h are the same as in ref.[27]. The blue's action policy is described as Algorithm 3.

3.6. Training process

In the confrontation training, the red and the blue confront each other for thousands of episodes in the air combat environment. Every episode works as Algorithm 1 and for every step the blue makes the maneuver decision as Algorithm 3. To train the red's action policy with the DRL algorithm, the experience for every step is stored. When satisfying the training conditions, the experiences will be used to update the red's action policy. To make sure the training is successful and to obtain satisfactory results, the training is divided into three phases, basic, dominant and balanced[3]. The initial states for these phases are shown in Table 2.

The three phases constitute a progressive relationship, which means the later training is based on the training results of the former training. The red's actor network and critic network are loaded with the former trained networks' parameters before starting the training.

4. RESULTS

4.1. Parameters setting

The hyperparameters setting for the DRL algorithm is shown in Table 3[18].

The parameters of the designed air combat environment are set as follows[15]. For the attacking distance, it is set as $D_{att,\min} = 40m$ and $D_{att,\max} = 900m$. The maximum attacking angle $\varphi_{att,\max} = \pi/6$ and the maximum escape

Algorithm 3 Action policy of the blue

```
Input: position: $\mathbf{p}_r$, $\mathbf{p}_b$; velocity: $\mathbf{v}_r$, $\mathbf{v}_b$; action space: $A$; prediction time: $t_{pred}$
Output: the blue's action $\mathbf{u}_b$
initialize the thread set $\Xi$
for all action $a$ in $A$ do
    calculate the red's $\mathbf{p}'_r$ and $\mathbf{v}'_r$ after $t_{pred}$ with action $a$ by Equation (1)
    calculate thread value by Equation (32) in the view of the red with $\mathbf{p}'_r$, $\mathbf{v}'_r$, $\mathbf{p}_b$ and $\mathbf{v}_b$
    append the thread value to $\Xi$
end for
set $ind$ equal to the index of the maximum value in $\Xi$
set $\mathbf{u}'_r$ equal to $ind$-th element in $A$
calculate the red's $\mathbf{p}'_r$ and $\mathbf{v}'_r$ after $t_{pred}$ with action $\mathbf{u}'_r$ by Equation (1)
initialize the thread set $\Xi$
for all action $a$ in $A$ do
    calculate the blue's $\mathbf{p}'_b$ and $\mathbf{v}'_b$ after $t_{pred}$ with action $a$ by Equation (1)
    calculate thread value by Equation (32) in the view of the blue with $\mathbf{p}'_r$, $\mathbf{v}'_r$, $\mathbf{p}'_b$ and $\mathbf{v}'_b$
    append the thread value to $\Xi$
end for
set $ind$ equal to the index of the maximum value in $\Xi$
set $\mathbf{u}_b$ equal to $ind$-th element in $A$
return $\mathbf{u}_b$
```

Table 2. The initial states of the UAVs in three training phases

Initial states Phases	Camp	x (m)	y (m)	z (m)	v (m/s)	θ (deg)	ψ (deg)
Basic	Red	[200, 1800]	[1500, 3500]	[700, 2500]	60	0	$[-0.5\pi, 0.5\pi]$
	Blue	[1500, 2200]	[1500, 3500]	[700, 2500]	60	0	$[-0.3\pi, 0.3\pi]$
Dominant	Red	[200, 1800]	[1500, 3500]	[700, 2500]	60	0	$[-0.5\pi, 0.5\pi]$
	Blue	[1500, 2200]	[1500, 3500]	[700, 2500]	60	0	$[-0.3\pi, 0.3\pi]$
Balanced	Red	[800, 4800]	[0, 4500]	[800, 3500]	60	0	$[0, 2\pi]$
	Blue	[800, 4800]	[0, 4500]	[800, 3500]	60	0	$[0, 2\pi]$

Table 3. The hyperparameters setting for the DRL algorithm

Hyperparameter	Value	Hyperparameter	Value
Learning rate	0.00025	GAE paramerter	0.95
Discount	0.99	Minimum buffer size	$N = 8192$
Number of batches	4	Epoch	$K = 5$
Clip parameter	0.1	Main objective's coefficients	$c_1 = 0.5; c_2 = 0.01$

angle when attacking $\varphi_{esp,att} = \pi/3$. For the advantage area, it is set as $D_{adv,\min} = 40m$, $D_{adv,\max} = 1300m$ and $\varphi_{esp,\max} = \pi/3$. For the blood, the probabilities are set as $p_{att1} = 0.1$, $p_{att2} = 0.4$ and $p_{att} = 0.8$, and the damage values are set as $B_0 = 300$, $B_1 = 51$, $B_2 = 21$ and $B_3 = 11$. For an episode, the maximum decision step $t_{\max} = 400s$ and the time step $t_{step} = 0.5s$. The threshold values are set as $D_{th} = H_{th} = 5000m$. For the reward function, the weights are set as $r_a = 0.15$, $r_d = 0.6$, $r_v = 0.1$, $r_h = 0.15$, $w_{dense} = 0.05$ and $w_{adv} = 0.05$, the parameters about height are set as $H_{\max} = 500m$, $H_{adv} = 300m$, $H_{att} = 100m$ and $H_{\min} = -300m$, and the parameters in end-game reward are set as $r_0 = 5$, $r_1 = 3$, $r_2 = 6$ and $r_{loss} = 15$.

4.2. Training results in the phases

The four cases are trained with the hyperparameters in Table 3 and initial states in Table 2. Four cases are compared in this paper, which are case I PPO, case II PPO with GRU, case III PPO with GRU and state input

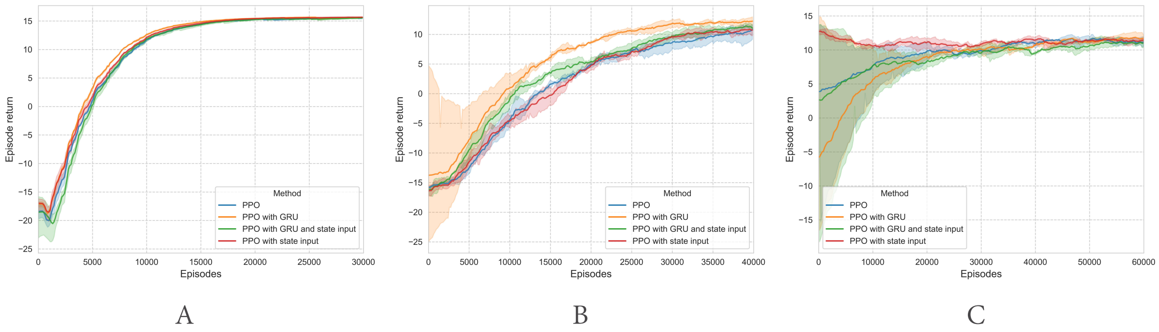


Figure 5. The episode returns of every episode in the three phases while training. A: the episode returns in the basic phase; B: the episode returns in the dominant phase; C: the episode returns in the balanced phase.

and case IV PPO with state input. The state input means the inputs of actor and critic networks are different, and if there is no GRU, the GRU layer in Figure 4 is replaced with linear layer with 128 units and ReLU.

In the basic phase, the UAVs' states are shown in the first line of Table 2. The blue always performs the forward and maintain action a_0 and the red is initialized behind the blue. Four cases are trained with 30000 episodes respectively. In the dominant phase, the UAVs' states are shown in the second line of Table 2. The blue uses the policy as Algorithm 2 and red is also initialized behind the blue. Four cases are trained with 40000 episodes respectively, and every case trains based on their own training result in the basic phase. In the balanced phase, the UAVs' states are shown in the third line of Table 2, and the initial states of the red and the blue are the same. Four cases are trained with 60000 episodes respectively, and every case trains based on their own training result in the dominant phase. The returns of every episode in the three phases are shown in Figure 5.

It can be seen that the PPO algorithm can go to converge faster when it is combined with GRU. In the basic phase, a relatively simple scenario, all four cases can easily find a better action policy to enclose the blue and beat it. And in the more complex scenario, it can find a better policy faster with GRU. But there are also larger episode reward variations in its infancy. Therefore, the state input is introduced to reduce the episode reward variations, which will synchronize to slightly reduce the final episode reward.

In addition, to test the final training result for the four cases, the training results in the balanced phase are reloaded and the initial states of the UAVs are set as

$$\begin{array}{l} x_r = 1000m, y_r = 2500m, z_r = 1200m, v_r = 90m/s, \theta_r = 0, \psi_r = 0 \\ x_b = 4200m, y_b = 2500m, z_b = 1200m, v_b = 90m/s, \theta_b = 0, \psi_b = 0 \end{array}. \tag{33}$$

And the maneuvering trajectories are shown in Figure 6. The steps for the four cases are 146, 90, 123 and 193. For Figure 6B, it is shown that the blue and the red collide. It is obvious that the red can beat the blue in a smaller space range when trained with PPO with GRU, and by adding state input, the red can be more flexible to avoid collision. But the cost is the increase in time steps, which explains the decrease of final episode reward. In case III, the red uses hover and altitude variations to lure the blue closer and gain an advantage situation, instead of pursuing the blue.

4.3. Confrontation tests

To compare the final training result of four cases, the confrontation tests for the four cases are conducted in this part when all of the training is finish. To accelerate the confrontation tests, the B_0 is set as 100 during the test, and the UAVs' initial states are the same as they are in the balanced phases. The tie air combat result is

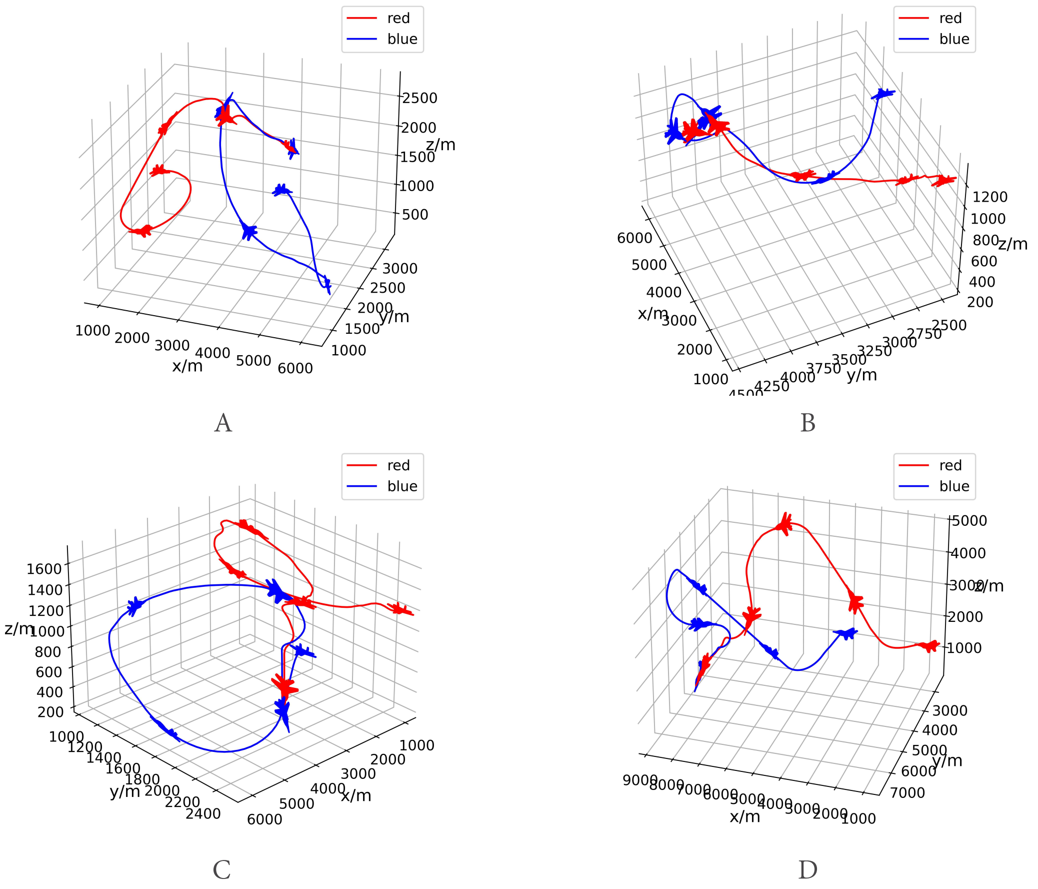


Figure 6. The maneuvering trajectories in the training test for four cases. A: the maneuvering trajectories for the case I PPO; B: maneuvering trajectories for case II PPO with GRU; C: maneuvering trajectories for case III PPO with GRU and state input; D: maneuvering trajectories for case IV PPO with state input.

Table 4. The confrontment test results between four cases

Algorithm case	Results (100 episodes)			
	Win rate	Loss rate	Tie rate	
			Tie win rate	Tie loss rate
PPO with GRU and state input *vs.* PPO	74%	12%	9%	5%
PPO with GRU and state input *vs.* PPO with GRU	57%	1%	17%	25%
PPO with GRU and state input *vs.* PPO with state input	52%	14%	14%	20%

divided into two types, tie win if $B_r > B_b$ and tie loss if $B_r \leqslant B_b$. All of the tests are conducted in the air combat environment for 100 episodes, and the results are shown in Table 4.

It can be seen that by combining PPO with GRU and state input, the UAV can get a more flexible and intelligent action policy even though the training process is the same. It is proved that training the action policy by the PPO with proposed improve points can help the UAV gain an advantage situation more quickly and greater operational capability in short-range air combat confrontation, and the action policy can be more intelligent to adapt to the blue's uncertain policy.

5. CONCLUSIONS

In this article, a maneuver decision method for UAV air combat is proposed based on the PPO algorithm. To enhance the PPO's performance, the GRU layer and different compositions of networks' inputs are adopted. At the same time, to accelerate training, some designs are applied. The action space is discretized into 15 basic actions, and the reward function is well-designed with three parts. Further, the training process is divided into several progressively more complex phases. To illustrate the advantages of the designed method, ablation experiments and UAV air combat tests are conducted in this paper. The episode rewards and confrontment test results show that the designed maneuver decision method can generate a more intelligent action policy for the UAV to win short-range air combat. By combining the PPO with the improved points, the training feasibility is improved and convergence is more efficient. The proposed maneuver decision-making method is always is able to achieve a win rate of more than 50% and a loss rate of less than 15%.

In future, the more complex six-degree-of-freedom UAV motion model and tighter UAV performance constraints could be introduced to improve accuracy. On the other hand, the multiple-to-multiple air combat problem, including multi-UAV coordinated attacking and tactical decisions, is the focus of future research.

DECLARATIONS

Authors' contributions

Made substantial contributions to the research, idea generation, software development, and conduct the DRL experiments. Wrote and edited the original draft: Zheng Z

Performed process guidance responsibility for the planning and execution of the study, as well as the evolution of overarching research aims, critical review, and material support. Review and revise the original draft: Duan H

Availability of data and materials

Not applicable.

Financial support and sponsorship

This work was partially supported by National Natural Science Foundation of China under grant #U20B2071, #91948204, #T2121003 and #U1913602.

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Copyright

REFERENCES

1. Ayass T, Coqueiro T, Carvalho T, Jailton J, Araújo J, Francês R. Unmanned aerial vehicle with handover management fuzzy system for 5G networks: challenges and perspectives. *Intell Robot* 2022;2:20-6. DOI
2. Zhang JD, Yang QM, Shi GQ, Lu Y, Wu Y. UAV cooperative air combat maneuver decision based on multi-agent reinforcement learning. *J Syst Eng Electron* 2021;6:1421–88. DOI

3. Yang QM, Zhang JD, Shi GQ, Hu JW, Wu Y. Maneuver decision of UAV in short-range air combat based on deep reinforcement learning. *IEEE Access* 2020;8:363–78. DOI
4. Ruan WY, Duan HB, Deng YM. Autonomous maneuver decisions via transfer learning pigeon-inspired optimization for UCAVs in dogfight engagements. *IEEE/CAA J Autom Sinica* 2022;9:1639–57. DOI
5. Yang Z, Zhou DY, Piao HY, Zhang K, Kong WR, Pan Q. Evasive maneuver strategy for UCAV in beyond-visual-range air combat based on hierarchical multi-objective evolutionary algorithm. *IEEE Access* 2020;8:46605–23. DOI
6. Xu GY, Liu Q, Zhang HM. The application of situation function in differential game problem of the air combat. 2018 Chinese Automation Congress (CAC); 2018 Nov 30-Dec 2; Xi'an, China. IEEE; 2019. pp. 1190–5. DOI
7. Başpınar B, Koyuncu E. Differential flatness-based optimal air combat maneuver strategy generation. AIAA Scitech 2019 Forum; 2019 Jan 7-11; San Diego, CA, USA. AIAA; 2019. pp. 1–10. DOI
8. Wang D, Zu W, Chang HX, Zhang J. Research on automatic decision making of UAV based on Plan Goal Graph. 2016 IEEE International Conference on Robotics and Biomimetics (ROBIO); 2016 Dec 3-7; Qingdao, China. IEEE; 2016. pp. 1245–9. DOI
9. Özbek MM, Koyuncu E. Reinforcement learning based air combat maneuver generation; 2022. Available from: http://arxiv.org/abs/2201.05528. [Last accessed on 15 Mar 2023] DOI
10. Zhang YT, Zhang YM, Yu ZQ. Path following control for UAV using deep reinforcement learning approach. *Guid Navigat Control* 2021;1:2150005. DOI
11. Zhang H, He L, Wang D. Deep reinforcement learning for real-world quadrupedal locomotion: a comprehensive review. *Intell Robot* 2022;2:275–97. DOI
12. Boin C, Lei L, Yang SX. AVDDPG - Federated reinforcement learning applied to autonomous platoon control. *Intell Robot* 2022;2:145–67. DOI
13. Li YF, Shi JP, Jiang W, Zhang WG, Lyu YX. Autonomous maneuver decision-making for a UCAV in short-range aerial combat based on an MS-DDQN algorithm. *Def Technol* 2022;9:1697–714. DOI
14. Li Y, Han W, Wang YG. Deep reinforcement learning with application to air confrontation intelligent decision-making of manned/unmanned aerial vehicle cooperative system. *IEEE Access* 2020;8:67887–98. DOI
15. Li LT, Zhou ZM, Chai JJ, Liu Z, Zhu YH, Yi JQ. Learning continuous 3-DoF air-to-air close-in combat strategy using proximal policy optimization. 2022 IEEE Conference on Games (CoG); 2022 Aug 21-24; Beijing, China. IEEE; 2022. pp. 616–9. DOI
16. Kang YM, Liu Z, Pu ZQ, Yi JQ, Zu W. Beyond-visual-range tactical game strategy for multiple UAVs. 2019 Chinese Automation Congress (CAC); 2019 Nov 22-24; Hangzhou, China. IEEE; 2019. pp. 5231–6. DOI
17. Ma XT, Xia L, Zhao QC. Air-combat strategy using deep Q-learning. 2018 Chinese Automation Congress (CAC); 2018 Nov 30-Dec 2; Xi'an, China. IEEE; 2019. pp. 3952–7. DOI
18. Schulman J, Wolski F, Dhariwal P, Radford A, Klimov O. Proximal policy optimization algorithms; 2017. Available from: http://arxiv.org/abs/1707.06347. [Last accessed on 15 Mar 2023] DOI
19. Wu JT, Li HY. Deep ensemble reinforcement learning with multiple deep deterministic policy gradient algorithm. *Math Probl Eng* 2020;2020:1–12. DOI
20. Yuksek B, Demirezen MU, Inalhan G, Tsourdos A. Cooperative planning for an unmanned combat aerial vehicle fleet using reinforcement learning. *J Aerosp Inform Syst* 2021;18:739–50. DOI
21. Xing JW. RLCodebase: PyTorch codebase for deep reinforcement learning algorithms; 2020. Available from: https://github.com/KarlXing/RLCodebase. [Last accessed on 15 Mar 2023]
22. Chung JY, Gulcehre C, Cho K, Bengio Y. Empirical evaluation of gated recurrent neural networks on sequence modeling; 2014. Available from: http://arxiv.org/abs/1412.3555. [Last accessed on 15 Mar 2023] DOI
23. Pope AP, Ide JS, Micovic D, et al. Hierarchical reinforcement learning for air-to-air combat; 2021. Available from: https://arxiv.org/abs/2105.00990. [Last accessed on 15 Mar 2023] DOI
24. Sun ZX, Piao HY, Yang Z, et al. Multi-agent hierarchical policy gradient for air combat tactics emergence via self-play. *Eng Appl Artif Intel* 2021;98:104112. DOI
25. Hu JW, Wang LH, Hu TM, Guo CB, Wang YX. Autonomous maneuver decision making of dual-UAV cooperative air combat based on deep reinforcement learning. *Electronics* 2022;11:467. DOI
26. Jing XY, Hou MY, Wu GL, Ma ZC, Tao ZX. Research on maneuvering decision algorithm based on improved deep deterministic policy gradient. *IEEE Access* 2022;10:92426-45. DOI
27. Yang AW, Li ZW, Li B, Xi ZF, Gao CQ. Air combat situation assessment based on dynamic variable weight. *Acta Armamentarii* 2021;42:1553-63. (in Chinese) DOI

Zhou *et al*. *Intell Robot* 2023;3(1):95-112
DOI: 10.20517/ir.2023.05

Intelligence & Robotics

Research Article

Open Access

Human gait tracking for rehabilitation exoskeleton: adaptive fractional order sliding mode control approach

Yuan Zhou[1,2], Zhe Sun[1,2], Bo Chen[1,2], Guangpu Huang[1,2], Xiang Wu[1,2], Tian Wang[3]

[1]Department of Automation, Zhejiang University of Technology, Hangzhou 310023, Zhejiang, China.
[2]Zhejiang Provincial United Key Laboratory of Embedded Systems, Zhejiang University of Technology, Hangzhou 310023, Zhejiang, China.
[3]Hangzhou RoboCT Technology Development Co., Ltd, Hangzhou 310000, Zhejiang, China.

Correspondence to: Prof. Bo Chen, Department of Automation, Zhejiang University of Technology, Hangzhou 310023, Zhejiang, China. E-mail: bchen@aliyun.com; ORCID: 0000-0001-6150-3881

How to cite this article: Zhou Y, Sun Z, Chen B, Huang G, Wu X, Wang T. Human gait tracking for rehabilitation exoskeleton: adaptive fractional order sliding mode control approach. *Intell Robot* 2023;3(1):95-112. http://dx.doi.org/10.20517/ir.2023.05

Received: 1 Feb 2023 **First Decision:** 2 Mar 2023 **Revised:** 16 Mar 2023 **Accepted:** 23 Mar 2023 **Published:** 31 Mar 2023

Academic Editor: Simon Yang **Copy Editor:** Yin Han **Production Editor:** Yin Han

Abstract

To improve the rehabilitation training effect of hemiplegic patients, in this paper, a discrete adaptive fractional order fast terminal sliding mode control approach is proposed for the lower limb exoskeleton system to implement high-precision human gait tracking tasks. Firstly, a discrete dynamic model is established based on the Lagrange system discretization criterion for the lower limb exoskeleton robot. Then, in order to design a discrete adaptive fractional order fast terminal sliding mode controller, the Grünwald–Letnikov fractional order operator is introduced to combine with fast terminal attractor to construct a fractional order fast terminal sliding surface. An adaptive parameter adjustment strategy is proposed for the reaching law of sliding mode control, which drives the sliding mode to the stable region dynamically. Moreover, the stability of the control system is proved in the sense of Lyapunov, and the guidelines for selecting the control parameters are given. Finally, the simulations are tested on the MATLAB-Opensim co-simulation platform. Compared with the conventional discrete sliding mode control and discrete fast terminal sliding mode control, the results verify the superiority of the proposed method in improving lower limb rehabilitation training.

Keywords: Lower-limb exoskeleton, adaptive discrete-time sliding mode, fractional order control, finite-time convergence

1. INTRODUCTION

In recent years, the aging population of many countries in the world has increased sharply, and the health problems of the elderly have been widely concerned by the public[1]. Stroke is a common disease in the elderly population, which will lead to the paralysis of the lower limbs of patients. If the patient can get timely and effective exercise rehabilitation treatment, the patient's motor function can be restored[1,2]. Traditional rehabilitation training mainly relies on physical therapists to provide patients with highly repetitive training. However, the number of therapists is seriously insufficient to meet the social requirements. Furthermore, the traditional method is a mostly subjective evaluation, which is inefficient and cannot guarantee effectiveness. In this situation, a lower limb exoskeleton robot is useful for the patient to conduct repetitive rehabilitation training, greatly reduce the burden of therapists, and assist doctors in accurately observing patients' rehabilitation status. Generally, rehabilitation can be divided into three stages based on the degree of spinal cord injury: initial, intermediate, and terminal. In the initial stage, due to the weak mobility of the patient, the patient needs to wear the lower limb exoskeleton robot to accurately follow the predetermined gait trajectory for rehabilitation training. With the gradual recovery of mobility, the patient can enter the middle and final stages of active rehabilitation training[2]. Therefore, the initial stage is crucial for the entire rehabilitation training. However, the uncertainty and disturbance caused by the unexpected behavior of stroke patients will seriously impact the initial rehabilitation training.

To improve rehabilitation efficiency at the initial stage and eliminate the influence of external disturbance, many control algorithms can be applied to exoskeleton robots. For instance, the PID control, adaptive control, robust control[3], fuzzy control[4], active disturbance rejection control[5], neural network control[6], Master-Slave Synchronization[7], and sliding mode control methods[8]. Among the methods, sliding mode control has the characteristics of fast response, insensitivity to uncertainties, and easy implementation in motion control applications. In particular, the sliding mode control can overcome the problems of external disturbances and uncertainties by constructing the reaching law and sliding mode surface in theory, so that the controlled system can achieve higher tracking accuracy. Non-singular terminal sliding mode control[9] and fast terminal sliding mode control[10] were applied to overcome parameter uncertainty and external disturbances to realize gait tracking control of lower limb exoskeleton rehabilitation robot, and theoretically analyzed the stability of controller design and tracking accuracy of trajectory. Sliding mode control technology can also be combined with the neural network, a recurrent neural network-based robust nonsingular sliding mode control is proposed for the non-holonomic spherical robot, it can enhance the robustness to control the system[11]. To obtain higher accuracy, fractional order sliding mode control is introduced to deal with uncertainties and external disturbances. Fractional order sliding mode control has the characteristics of global memory and elimination of jitter, so it is widely used in industry, such as micro gyroscope[12], manipulator control[13], and permanent magnet synchronous motor control[14]. In addition, the fractional order control algorithm can be applied in the field of robot control in combination with other technical methods. For example, a fractional neural integral sliding-mode controller based on the Caputo-Fabrizio derivative and Riemann–Liouville integral for a robot manipulator mounted on a free-floating satellite[15] and a method based on the nested saturations technique and the Caputo-Fabrizio derivative for a quadrotor aircraft[16]. In chaotic systems, the application of fractional order can endow the system with more degrees of freedom[17], help to study the dynamic behavior of the system, combined with robust control methods[18], eliminate external interference, and effectively solve the synchronization problem of the system[19]. However, the general fractional order sliding mode control strategy is designed based on the continuous time state of the controlled object and is directly been tested on the digital computer system, so the design of the controller ignoring the sampling interval will lead to the loss of the control system precision[20,21]. Moreover, the use of fractional operators to construct sliding mode functions also introduces the influence of uncertainty. If the fractional operators are defined by Grünwald–Letnikov (GL), there are non-physical initial conditions in the experiment. The fractional definition of Caputo is feasible in engineering applications, but the definition of Caputo can only be implemented in the approximation method based on the Laplace transform, which will introduce additional approximation errors to the control system[22].

In this paper, a novel discrete adaptive fractional order fast terminal sliding mode controller (AFOFTSMC) is designed for high-precision gait trajectory tracking tasks. To reduce the difference between theoretical design and practical application of digital computer systems, the controller designed in this paper derived the discrete-time object model based on the Lagrange discretization criterion. In addition, to preserve the global memorability of fractional operators, Grünwald–Letnikov fractional difference operators are used to construct discrete sliding mode surfaces. Considering the uncertainty of parameters and the boundedness of disturbances, a new adaptive terminal sliding mode approach law is proposed to drive the sliding mode dynamics to the region of finite step size. In this paper, the theoretical analysis of the system entering the stable state in finite time is given, and the validity of the algorithm is tested on the co-simulation platform of MATLAB and Opensim software.

The rest of this article is structured as follows. The second part describes the lower extremity exoskeleton discrete model based on the Lagrange system discrete criterion. The design and stability analysis of the discrete adaptive fractional order fast terminal sliding mode controller is presented in Section III. In Section IV, the simulation results are analyzed to prove the effectiveness of the controller. Section V summarizes the thesis.

2. DYNAMICS MODEL OF LOWER LIMB EXOSKELETON

The swing leg dynamic model is considered according to a two-degree-of-freedom (2-DOF) lower limb exoskeleton diagram shown in Figure 1. Based on the motion mechanism of human lower limbs, the hip and knee joints are designed as active joints, and the ankle joint is designed as a passive joint. The physical parameters of the 2-DOF lower limb exoskeleton in Figure 1 are explained as follows. $O(0,0)$ represents the coordinate origin, q_i $(i = 1, 2)$ denotes the angle of the hip or knee joint, l_{ci} $(i = 1, 2)$ represents the distance between the centroid of thigh or shank segment and the hip joint or knee joint, l_i $(i = 1, 2)$ corresponds to the length of the thigh or shank segment, m_i $(i = 1, 2)$ denotes the mass of thigh or shank segment.

To achieve high-precision motion control of the lower limb exoskeleton rehabilitation robot. We established a dynamic model of the lower limb exoskeleton using the Lagrange equation of motion. The equation of general forms is expressed as follows[23]:

$$L = T - V \tag{1}$$

$$\tau = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\right) - \frac{\partial L}{\partial q} \tag{2}$$

where L denotes the Lagrangian, T and V represent kinetic energy and potential energy respectively. τ represents the torque of the system.

The equations of motion of a lower limb exoskeleton robot are described according to the Lagrange equation (2):

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + N(q,\dot{q},\ddot{q}) = \tau \tag{3}$$

where q, $\dot{q}$ and $\ddot{q}$ denote the joint angle, angular velocity, and angular acceleration vectors respectively, $M(q) \in R^{2\times2}$ is the positive definite inertia matrix, $C(q,\dot{q}) \in R^{2\times2}$ is the Coriolis and centrifugal force matrix, $G(q) \in R^{2\times1}$ is the gravity matrix[3], $\tau \in R^2$ is the torque vector, $N(q,\dot{q},\ddot{q}) \in R^{2\times1}$ denotes the uncertainty of model parameters and external disturbances. $N(q,\dot{q},\ddot{q})$ can be expressed as:

$$N(q,\dot{q},\ddot{q}) = \Delta M(q)\ddot{q} + \Delta C(q,\dot{q})\dot{q} + \Delta G(q) - \tau_d. \tag{4}$$

The uncertainty of model parameters and external disturbances should be considered in the actual lower limb exoskeleton dynamics model. $\Delta M(q) \in R^{2\times2}$, $\Delta C(q,\dot{q}) \in R^{2\times2}$, $\Delta G(q) \in R^{2\times1}$ denotes the uncertain inertia

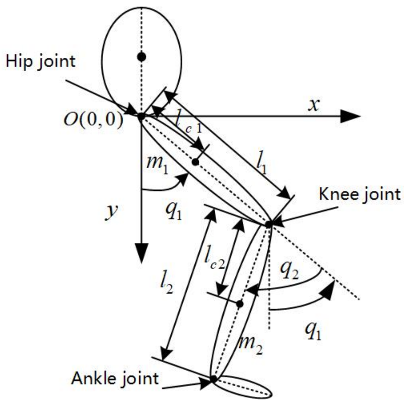


Figure 1. Simplified diagram of 2-DOF lower limb exoskeleton. 2-DOF: two-degree-of-freedom.

part, uncertain Coriolis and centrifugal force part, uncertain gravitational part respectively, and the external disturbances as $\tau_d \in R^{2\times 1}$.

To design digital motion control systems, it is crucial to obtain nominal discretization dynamics. In this paper, the discretization substitution criterion of the Lagrangian system is used to discretize the dynamics model. The discretization criterion is as follows[24]:

$$\frac{d}{dt}(\frac{\partial L}{\partial \dot{q}_i}) \rightarrow \frac{1}{T}[(\frac{\partial L}{\partial \dot{q}_i})_{k+1} - (\frac{\partial L}{\partial \dot{q}_i})_k] \tag{5}$$

$$\frac{\partial L}{\partial q_i} \rightarrow \frac{1}{2}[\frac{M(q_{k+1}) - M(q_k)}{q_{i,k+1} - q_{i,k}}]\dot{q}_{i,k+1}\dot{q}_{i,k} \tag{6}$$

where T denotes the sampling period, k denotes time step. Then, the explicit form of the discrete dynamics model is derived as follows[25]:

$$\begin{cases} q_{k+1} = q_k + T\dot{q}_k \\ \dot{q}_{k+1} = M^{-1}(q_k + T\dot{q}_k)M(q_k)\dot{q}_k + TM^{-1}(q_k + T\dot{q}_k)C(q_k \\ \qquad\qquad + T\dot{q}_k, q_k, \dot{q}_k)\dot{q}_k + TM^{-1}(q_k + T\dot{q}_k)\tau_k - \\ \qquad\qquad TM^{-1}(q_k + T\dot{q}_k)G(q_k + T\dot{q}_k, q_k) \end{cases} \tag{7}$$

the description of the system given in (7) can be expressed in state representation form as:

$$x(k+1) = f(k) + b(k)u(k) + d(k) \tag{8}$$

$$f(k) = \begin{bmatrix} I_{2\times 2} & I_{2\times 2}T \\ 0_{2\times 2} & M^{-1}(q_k + T\dot{q}_k)M(q_k) + TM^{-1}(q_k + T\dot{q}_k)C - TM^{-1}(q_k + T\dot{q}_k)G \end{bmatrix} x(k) \tag{9}$$

$$b(k) = \begin{bmatrix} 0_{2\times 2} \\ M^{-1}(q_k + T\dot{q}_k) \end{bmatrix} \tag{10}$$

where, $x(k) = [q_{1,k}, q_{2,k}, \dot{q}_{1,k}, \dot{q}_{2,k}]^T$ is the system state vector, $u(k) = [\tau_{1,k}, \tau_{2,k}]^T$ is the system control input vector, $f(k) \in R^{4\times 1}$ is the nonlinear state transition matrix, and $b(k) \in R^{4\times 2}$ is the control matrix, $I_{2\times 2}$ is represented as the second-order identity matrix, and $0_{2\times 2}$ is a second-order zero matrix, $d(k) \in R^{4\times 1}$ is the set of parameter uncertainties and external disturbances of the system. Assuming that $d(k)$ is bounded, then $||d(k)|| \leq d_\rho$.

3. CONTROLLER DESIGN AND STABILITY ANALYSIS

In the previous section, we modeled and analyzed the dynamics model of the lower limb exoskeleton robot. In this section, we will design an appropriate controller for the lower limb exoskeleton robot. The design of the controller is divided into five parts, and the process of controller construction, stability proof, and parameter selection are explained completely. Firstly, before introducing the fractional order sliding mode surface, the definition of fractional order needs to be introduced. Fractional order has the advantage of global memorability and plays an important role in the construction of AFOFTSMC. In the second part, the fractional order fast terminal sliding mode surface and adaptive terminal control law is proposed to construct an AFOFTSMC controller for the lower limb exoskeleton robot. In the third part, the stability of the controller is proved in detail, and it is proved that both the sliding mode variables and the system errors can converge in the bounded region. The fourth part gives some guiding opinions on the parameter selection of the controller. Finally, conventional sliding mode control (CSMC) and fast terminal sliding mode control (FTSMC) controllers are designed to compare with AFOFTSMC.

3.1. Preliminaries

In order to design a discrete adaptive fractional order fast terminal sliding mode controller for the lower limb exoskeleton system, the sliding mode surface function should be designed according to the properties of the fractional order operator, and the adaptive sliding mode control law should be designed to form the controller. Therefore, we will elaborate on the basic definition and related nomenclatures of the fractional operator in detail.

Definition 1: The Grünwald–Letnikov fractional order operator is defined as follows[22]:

$$ {}_{a}^{GL}D_{t}^{\lambda}f(t)=\frac{1}{h^{\lambda}}\sum_{j=0}^{[(t-a)/h]}(-1)^{j}\binom{\lambda}{j}f(t-jh) \tag{11}$$

where λ is the arbitrary order of function $f(t)$, the value of λ will affect the calculus properties of fractional order operators. When $\lambda > 0$, the fractional order operator is a differentiator, while $\lambda < 0$, the fractional order operator is an integrator[19]. a is the initial value of the integral, and generally, zero initial condition can be assumed, that is, $a = 0$. h is the sampling time interval, and $\binom{\lambda}{j}$ is the binomial coefficient. The specific calculation method is as follows:

$$\binom{\lambda}{j}=\begin{cases}1 & j=0\\ \frac{\lambda(\lambda-1)\cdots(\lambda-j+1)}{j!} & j=1,2,3,\dots\end{cases}. \tag{12}$$

However, storing all motion data to calculate fractional integrals in practical engineering applications consumes hardware resources and makes the calculation inefficient. Therefore, to improve the operation efficiency and ensure the global memory of the fractional operator, the fractional integral can be calculated by storing part of the motion data, as shown below:

$$ {}_{a}^{GL}D_{t}^{\lambda}f(t)=\frac{1}{h^{\lambda}}\sum_{j=0}^{L}(-1)^{j}\binom{\lambda}{j}f(t-jh) \tag{13}$$

where L represents the limited amount of data stored.

Lemma 1[22]: The sum of binomial coefficients in equation (13) can be expressed by gamma function $\Gamma(z) = \int_0^\infty e^{-t}t^{z-1}dt$ as:

$$\sum_{j=0}^{L}(-1)^{j}\binom{\lambda}{j}=\frac{\Gamma(L+1-\lambda)}{\Gamma(1-\lambda)\Gamma(L+1)}. \tag{14}$$

3.2. Controller design

To synthesize the advantages of the fractional order sliding mode surface and the fast terminal sliding mode control law to construct the controller, the appropriate fractional order sliding mode surface should be selected. Several fractional order sliding mode surfaces have been described in the literature[8,12–14,22]. Inspired by the above strategies, the discrete fractional order sliding mode surface selected are as follows:

$$s(k) = c_1 e_1(k) + e_2(k) + c_2 D^{\lambda}[|e_1(k)|^{\beta} sgn(e_1(k))] \tag{15}$$

where $e_1(k) = [x_{d1}(k) - x_1(k), x_{d2}(k) - x_2(k)]^T$ is the tracking error between the desired position and real position, $e_2(k) = [x_{d3}(k) - x_3(k), x_{d4}(k) - x_4(k)]^T$ is the tracking error between desired velocity and real velocity, $x_d(k) \in R^{4\times1}$ is the reference signal vector, $c_1 = diag(c_{1i})$ $(i = 1, 2)$, $c_2 = diag(c_{2i})$ $(i = 1, 2)$ are selected constant matrices, $0 < \beta = \frac{q_\beta}{p_\beta} < 1$ with p_β, q_β being both odd positive integers.

Remark 1: For a nonlinear system, when the system state is far from the equilibrium point, the fractional order terminal sliding mode surface proposed by Sun *et al.*[22] can ensure that the system converges in a finite time. However, considering that the system state is close to the equilibrium point, the terminal attractor cannot guarantee the fast convergence of the system. In this paper, a linear term $c_1 e_1(k)$ is introduced into the sliding mode surface, when the system state is close to the equilibrium point, the convergence time is mainly determined by the linear term $c_1 e_1(k)$, which can accelerate the convergence of the system. Therefore, the sliding mode surface designed in this paper not only makes the system state converge in a finite time but also preserves the rapidity of the linear sliding mode when it is close to the equilibrium point.

To make the system stable, for the system model (8), the ideal quasi-sliding mode band should meet the following requirements: $s(k + 1) = 0$, then the controller can be obtained as follows:

$$C_1 = \begin{bmatrix} c_{11} & 0 & 1 & 0 \\ 0 & c_{12} & 0 & 1 \end{bmatrix} \tag{16}$$

$$C_1[x_d(k + 1) - f(k) - b(k)u(k) - d(k)] + c_2 D^{\lambda}[|e_1(k)|^{\beta} sgn(e_1(k))] = 0. \tag{17}$$

The equivalent control law is:

$$u_{eq}(k) = [C_1 b(k)]^{-1}[C_1(x_d(k + 1) - f(k)) + c_2 D^{\lambda}[|e_1(k)|^{\beta} sgn(e_1(k))]]. \tag{18}$$

To eliminate the influence brought by system parameter uncertainty and external disturbance, the new adaptive terminal sliding mode reaching law used for system model (8) is:

$$s(k + 1) = PQs(k) - TP\Phi|s(k)|^{\alpha} sgn[s(k)] \tag{19}$$

where, $P = diag(P_i)$ with $P_i = 1 - exp\left(-[\frac{s_i(k)}{\epsilon}]^{2m}\right)$, $Q = diag(Q_i)$ with $Q_i = 1 - \sigma_i T$, $\Phi = diag(\Phi_i)$ with $\Phi_i = \delta|s_i(k)|$, $\delta > 0$, $0 < 1 - \sigma_i T < 1$, $0 < \alpha < 1$, ϵ and m are positive real numbers, $i = 1, 2$.

Then the switching control law u_{sw} is as follows:

$$u_{sw}(k) = -[C_1 b(k)]^{-1}[PQs(k) - TP\Phi|s(k)|^{\alpha} sgn[s(k)]]. \tag{20}$$

Combining (8), (15) and the reaching law (19), an AFOFTSMC law is obtained, and the corresponding control input can be expressed as

$$u(k) = u_{eq}(k) + u_{sw}(k). \tag{21}$$

Substituting equation (21) into the sliding mode function $s(k + 1)$, which can be written as:

$$s(k + 1) = PQs(k) - TP\Phi|s(k)|^{\alpha} sgn[s(k)] - \xi(k) \tag{22}$$

where $\xi(k) = C_1 d(k)$, $|\xi(k)| \leq \rho_f$, $|\xi_i(k)| \leq \rho_i$.

3.3. Stability analysis

Lemma 2[20]: If $0 < \alpha < 1$, $\psi(\alpha)$ is a function in equation (23), then $\psi(\alpha) - x\psi(\alpha) - 1 + x^{\alpha}\psi(\alpha)^{\alpha} \geq 0$ holds for any $x \in [0, 1]$.

$$\psi(\alpha) = 1 + \alpha^{\frac{\alpha}{1-\alpha}} - \alpha^{\frac{1}{1-\alpha}} \tag{23}$$

with $0 \leq \alpha \leq 1$, $1 < \psi(\alpha) < 2$.

Proof: Let $f(x) = \psi(\alpha) - x\psi(\alpha) - 1 + x^{\alpha}\psi(\alpha)^{\alpha}$, then to prove whether the minimum value of $f(x)$ is greater than zero. From the properties of $\psi(\alpha)$, we know that $\psi(\alpha) > 1$, then $F(1) = -1 + \psi(\alpha)^{\alpha} > 0$, and $f(0) = \psi(\alpha)$. Then, the extreme value of $f(x)$ can be obtained from $\dot{f}(x) = 0$. When $\dot{f}(x) = 0$, $x = \alpha^{\frac{1}{1-\alpha}}(\psi(\alpha))^{-1}$ can be obtained, and then the minimum value $f(x) = \psi(\alpha) - \alpha^{\frac{1}{1-\alpha}} - 1 + \alpha^{\frac{\alpha}{1-\alpha}} = \psi(\alpha) - 1 + (\frac{1}{\alpha} - an)\alpha^{\frac{1}{1-\alpha}} > 0$, then the prove is completed.

Lemma 3[20]: If $0 < \alpha < 1$, $\psi(\alpha)$ is a function in equation (23), then $\psi(\alpha) + x\psi(\alpha) - 1 - x^{\alpha}\psi(\alpha)^{\alpha} \geq 0$ holds for any $x \in [0, 1]$.

The proof of Lemma 3 is similar to the proof of Lemma 2.

Theorem 1: For system model (8) with uncertainties and external disturbances, the following sliding mode motion properties can be guaranteed by using control law (21) :

1) The discrete-time sliding variable can be driven into the domain Ω within a finite number of steps K^*, and Ω can be expressed as follows.

$$\Omega = \{s(k) || s_i(k)| < \psi(\alpha)\varpi_i\} \tag{24}$$

$$\begin{aligned} \varpi_i &= max\left\{\left(\frac{\rho_i}{TP_i(k)\Phi_i(k)}\right)^{\frac{1}{\alpha}}, \left(\frac{T\Phi_i(k)}{1-\sigma_i T}\right)^{\frac{1}{1-\alpha}}\right\} \\ &= max\left\{\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}}, \left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}\right\} \end{aligned} \tag{25}$$

where, $Z_i(k) = TP_i(k)\Phi_i(k)$, $K^* = \lfloor \frac{s_i^2(0) - (\psi(\alpha)\varpi_i)^2}{(\rho_i\Phi^{\alpha} - \rho_i)^2} \rfloor + 1$, $K^* \in N^+$.

2) Once the sliding mode moves into the domain Ω, it will stay in the domain and will not escape, that is, when $|s_i(k)| \leq \psi(\alpha)\varpi_i$, then $|s_i(k+1)| \leq \psi(\alpha)\varpi_i$.

3) When the sliding mode variables move in the domain Ω, the errors will converge to the bounded region, as follows:

$$|e_{1i}(k)| \leq \psi(\beta) \cdot max\left\{\left(\frac{\vartheta_i T^{\lambda}}{c_{2i}}\right)^{\frac{1}{\beta}}, \left(\frac{c_{2i}}{(1-c_{1i}T)T^{\lambda}}\right)^{\frac{1}{1-\beta}}\right\} \tag{26}$$

where ϑ_i is a bounded variable, which can be known from the following analysis.

Proof: Choose the discrete Lyapunov function as $V(k) = s^T(k)s(k)$, then

$$\begin{aligned} \triangle V(k) &= V(k+1) - V(k) \\ &= \sum_{i=1}^{2} \left[s_i(k+1) - s_i(k)\right]\left[s_i(k+1) + s_i(k)\right]. \end{aligned} \tag{27}$$

To prove the reaching and existence of the control law, we discuss the following cases.

1) When the sliding mode moves outside the domain Ω, the following two conditions exist:

Case 1: Considering the situation that $s_i(k) > \psi(\alpha)\varpi_i > 0$, then equation (27) can be rewritten as:

$$\triangle V(k) = -\sum_{i=1}^{2}\left[s_i(k) - s_i(k+1)\right]\left[s_i(k) + s_i(k+1)\right]. \tag{28}$$

Since $s_i(k) > \psi(\alpha)\varpi_i$, then $s_i(k) > \psi(\alpha)\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}}$, and $Z_i(k)s_i^{\alpha}(k) > \rho_i\psi^{\alpha}(\alpha)$, thus $Z_i(k)s_i^{\alpha}(k) > \rho_i\psi^{\alpha}(\alpha)$, we can obtain:

$$\begin{aligned}
&s_i(k) - s_i(k+1)\\
&= s_i(k) - P_i(k)Q_i(k)s_i(k) + Z_i(k)|s_i(k)|^{\alpha}sgn[s_i(k)] + \xi_i(k)\\
&= [1 - P_i(k)Q_i(k)]s_i(k) + Z_i(k)|s_i(k)|^{\alpha}sgn[s_i(k)] + \xi_i(k)\\
&\geq [1 - P_i(k)Q_i(k)]s_i(k) + Z_i(k)|s_i(k)|^{\alpha} - |\xi_i(k)|\\
&\geq [1 - P_i(k)Q_i(k)]s_i(k) + \rho_i\psi^{\alpha}(\alpha) - \rho_i\\
&\geq \rho_i\psi^{\alpha}(\alpha) - \rho_i\\
&> 0
\end{aligned} \tag{29}$$

$$\begin{aligned}
&s_i(k) + s_i(k+1)\\
&= [1 + P_i(k)Q_i(k)]s_i(k) - Z_i(k)|s_i(k)|^{\alpha}sgn[s_i(k)] - \xi_i(k)\\
&\geq s_i(k) + Z_i(k)[\psi^{1-\alpha}(\alpha){s_i}^{\alpha}(k) - |s_i(k)|^{\alpha}sgn[s_i(k)]] - \xi_i(k)\\
&\geq {s_i}^{1-\alpha}(k){s_i}^{\alpha}(k) - \xi_i(k)\\
&\geq \frac{Z_i(k)}{P_i(k)Q_i(k)}{s_i}^{\alpha}(k) - \xi_i(k)\\
&\geq Z_i(k){s_i}^{\alpha}(k) - \xi_i(k)\\
&\geq Z_i(k){s_i}^{\alpha}(k) - |\xi_i(k)|\\
&\geq \rho_i\psi^{\alpha}(\alpha) - \rho_i\\
&\geq 0.
\end{aligned} \tag{30}$$

It can be seen from the above derivation that $s_i(k) - s_i\,(k+1) > 0$ and $s_i\,(k+1) + s_i(k) > 0$ are tenable, we can easy to deduce that:

$$\Delta V(k) = -\sum_{i=1}^{2}\left[s_i(k) - s_i\,(k+1)\right]\left[s_i\,(k+1) + s_i(k)\right] < 0. \tag{31}$$

Case 2: Moreover, another situation is that $s_i(k) < -\psi(\alpha)\varpi_i < 0$, similar to the proof for case 1, because $s_i(k) < -\psi(\alpha)\varpi_i$ and $s_i(k) < -\psi(\alpha)\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}}$, then $Z_i(k)\,|s_i(k)|^{\alpha} > \rho_i\psi^{\alpha}(\alpha)$, we can get that $-Z_i(k)\,|s_i(k)|^{\alpha} + |\xi_i(k)| < -\rho_i\psi^{\alpha}(\alpha) + \rho_i < 0$.

Therefore, $\Delta V(k) = -\sum_{i=1}^{2}\left(\rho_i\psi^{\alpha} - \rho_i\right)^2 < 0$ holds in this case. Through the analysis of the above knowledge, the system will enter the domain Ω in K^* step, the $s_i^2\,(K^*) - s_i^2(0) < -K^*\,(\rho_i\psi^{\alpha} - \rho_i)^2$, then we get $s_i^2\,(K^*) < s_i^2(0) - K^*\,(\rho_i\psi^{\alpha} - \rho_i)^2 < \left(\psi(\alpha)\varpi_i\right)^2$, available $K^* = \left\lfloor \frac{s_i^2(0) - (\psi(\alpha)\varpi_i)^2}{(\rho_i\psi^{\alpha} - \rho_i)^2} \right\rfloor + 1$, and $K^* \in N^+$.

2) When the sliding variables enter into the domain Ω, $|s_i(k)| \leq \psi(\alpha)\varpi_i$. To prove $|s_i\,(k+1)| \leq \psi(\alpha)\varpi_i$, it is essential to divide the analyses due to the location of $s_i(k)$.

Case 1: Consider $\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}} \geq \left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}$, suppose $s_i(k) = \psi(\alpha)\theta\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}}$, $0 < |\theta| < 1$, show that $|s_i(k+1)| \leq \psi(\alpha)\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}}$. $|\xi_i(k)| \leq \rho_i$, we have

$$\begin{aligned} s_i(k+1) &= P_i(k)\left(Q_i\right)s_i(k) - Z_i(k)|s_i(k)|^{\alpha} - \xi_i(k) \\ &\leq P_i(k)Q_i s_i(k) - \left(\psi(\alpha)\theta\right)^{\alpha}\rho_i + \rho_i sgn[s_i(k)] + \rho_i \end{aligned} \tag{32}$$

$$\begin{aligned} s_i(k+1) &= P_i(k)\left(Q_i\right)s_i(k) - Z_i(k)|s_i(k)|^{\alpha} - \xi_i(k) \\ &\geq P_i(k)Q_i s_i(k) - \left(\psi(\alpha)\theta\right)^{\alpha}\rho_i + \rho_i sgn[s_i(k)] - \rho_i. \end{aligned} \tag{33}$$

Since $\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}} \geq \left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}$, then

$$\begin{aligned} &\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}} \geq \left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}} \\ &\Rightarrow \left(\frac{\rho_i}{Z_i(k)}\right)^{1-\alpha} \geq \left(\frac{T\Phi_i(k)}{Q_i}\right)^{\alpha} \\ &\Rightarrow {\rho_i}^{1-\alpha}\left[P_i(k)\left(Q_i\right)\right]^{\alpha} \geq Z_i(k) \\ &\Rightarrow \rho_i \leq P_i(k)\left(Q_i\right)\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}}. \end{aligned} \tag{34}$$

Since $0 < \theta < 1$, lead to $\psi(\alpha)\theta \geq 1$ or $0 < \psi(\alpha)\theta < 1$. When $\psi(\alpha)\theta \geq 1$, we have

$$\begin{aligned} s_i(k+1) &\leq P_i(k)Q_i s_i(k) - \left(\psi(\alpha)\theta\right)^{\alpha}\rho_i + \rho_i \\ &\leq P_i(k)Q_i\psi(\alpha)\theta\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}} \\ &\leq \psi(\alpha)\theta\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}} \\ &\leq \psi(\alpha)\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}} \end{aligned} \tag{35}$$

$$\begin{aligned} s_i(k+1) &\geq P_i(k)Q_i s_i(k) - \left(\psi(\alpha)\theta\right)^{\alpha}\rho_i - \rho_i \\ &\geq P_i(k)Q_i s_i(k) - \left[\left(\psi(\alpha)\theta\right)^{\alpha} + 1\right]P_i(k)Q_i\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}} \\ &\geq \left[\psi(\alpha)\theta - \left(\psi(\alpha)\theta\right)^{\alpha} - 1\right]P_i(k)Q_i\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}} \\ &\geq -P_i(k)Q_i\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}} \\ &\geq -\psi(\alpha)\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}}. \end{aligned} \tag{36}$$

We consider the situation that $0 < \psi(\alpha)\theta < 1$, according to Lemma 3, we can also obtain $|s_i(k+1)| \leq \psi(\alpha)\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}}$.

From the above proof, we have $|s_i(k+1)| \leq \psi(\alpha)\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}}$, when $\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}} \geq \left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}$ and $0 < \theta < 1$, $0 < s_i(k) \leq \psi(\alpha)\varpi_i$.

When $-\psi(\alpha)\varpi_i \leq s_i(k) < 0$ and $-1 < \theta < 0$, $s_i(k) = \psi(\alpha)\theta\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}} = -\psi(\alpha)|\theta|\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}}$, then

$$s_i(k+1) = P_i(k)Q_i s(k) - Z_i(k)|s_i(k)|^{\alpha} sgn[s_i(k)] - \xi_i(k). \tag{37}$$

Similar to the above proof process, we can obtain $|s_i\,(k+1)| \leq \psi(\alpha)\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}}$.

From the above proof, it can be seen that $|s_i\,(k+1)| \leq \psi(\alpha)\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}}$ is still satisfied with the conditions $\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}} \geq \left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}$, $-1 < \theta < 0$ and $-\psi(\alpha)\varpi_i \leq s_i(k) < 0$.

Case 2: If $\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}} < \left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}$, suppose $s_i(k) = \psi(\alpha)\theta\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}$, $0 < |\theta| < 1$, then prove $|s_i\,(k+1)| \leq \psi(\alpha)\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}$. Submit $s_i(k)$ to $s_i(k+1)$, we can obtain that:

$$s_i(k+1) = P_i(k)Q_i s_i(k) - Z_i(k)\left(\psi(\alpha)\theta\right)^{\alpha}\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{\alpha}{1-\alpha}} - \xi_i(k) \tag{38}$$

$$\begin{aligned}
&\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}} < \left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}\\
&\Rightarrow \left(\frac{\rho_i}{Z_i(k)}\right)^{1-\alpha} < \left(\frac{T\Phi_i(k)}{Q_i}\right)^{\alpha}\\
&\Rightarrow {\rho_i}^{1-\alpha} < \frac{Z_i(k)}{[P_i(k)Q_i]^{\alpha}}\\
&\Rightarrow \rho_i < P_i(k)Q_i\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}.
\end{aligned} \tag{39}$$

When $\psi(\alpha)\theta \geq 1, \psi(\alpha)\theta \geq (\psi(\alpha)\theta)^{\alpha}$, then, based on Lemma 2

$$\begin{aligned}
&s_i(k+1)\\
&\leq P_i(k)Q_i s_i(k) - P_i(k)Q_i\left(\psi(\alpha)\theta\right)^{\alpha}\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}} + \rho_i\\
&\leq \left[\psi(\alpha)\theta - (\psi(\alpha)\theta)^{\alpha} + 1\right]P_i(k)Q_i\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}\\
&\leq \psi(\alpha)P_i(k)Q_i\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}\\
&\leq \psi(\alpha)\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}
\end{aligned} \tag{40}$$

$$\begin{aligned}
&s_i(k+1)\\
&\geq P_i(k)Q_i s_i(k) - P_i(k)Q_i\left(\psi(\alpha)\theta\right)^{\alpha}\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}} - \rho_i\\
&\geq \left[\psi(\alpha)\theta - (\psi(\alpha)\theta)^{\alpha} - 1\right]P_i(k)Q_i\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}\\
&\geq -P_i(k)Q_i\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}\\
&\geq -\psi(\alpha)F_i(k)\left(Q_i\right)\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}.
\end{aligned} \tag{41}$$

After $\psi(\alpha)\theta \geq 1$ with $\psi(\alpha)\theta \geq (\psi(\alpha)\theta)^{\alpha}$ proved, we will discuss $0 < \psi(\alpha)\theta < 1$ with $\psi(\alpha)\theta \leq (\psi(\alpha)\theta)^{\alpha}$. According to Lemma 3, we can obtain that $|s_i\,(k+1)| \leq \psi(\alpha)\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}$.

It can be seen from the above proof that under conditions $\left(\frac{\rho_i}{Z_i(k)}\right)^{\frac{1}{\alpha}} < \left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}$ and $0 < \theta < 1$, namely, $0 < s_i(k) \leq \psi(\alpha)\varpi_i$, $|s_i\,(k+1)| \leq \psi(\alpha)\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}$ holds.

When $-1 < \theta < 0$, $s_i(k) = \psi(\alpha)\left(\frac{T\Phi_i(k)}{1-q_iT}\right)^{\frac{1}{1-\alpha}} = -\psi(\alpha)|\theta|\left(\frac{T\Phi_i(k)}{1-q_iT}\right)^{\frac{1}{1-\alpha}}$.

$$
\begin{aligned}
& s_i(k+1) \\
&= P_i(k)Q_is_i(k) - Z_i(k)\left(\psi(\alpha)\theta\right)^{\alpha}\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{\alpha}{1-\alpha}} - \xi_i(k) \\
&= P_i(k)Q_is_i(k) + P_i(k)Q_i\left(\psi(\alpha)|\theta|\right)^{\alpha}\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}} - \xi_i(k).
\end{aligned}
\tag{42}
$$

Similar to the above proof process, we can obtain $|s_i\,(k+1)| \leq \psi(\alpha)\left(\frac{T\Phi_i(k)}{Q_i}\right)^{\frac{1}{1-\alpha}}$.

To this end, when $s_i(k) \in \Omega$, $s_i(k+1) \in \Omega$.

3) After the sliding mode variables enter the determination domain Ω, we will further discuss the bounded convergence region of the tracking errors. Before proving, it is necessary to introduce an important lemma about the discrete fast terminal sliding mode surface, as follows.

Lemma 4[20]: Consider a scalar dynamical system

$$
z(k+1) = vz(k) - lz(k)^{\alpha} + g(k)
\tag{43}
$$

where $v > 0$, $l > 0$ and $0 < \alpha < 1$. if $|g(k)| < \gamma$, $\gamma > 0$, then the state $z(k)$ is always bounded and there is a finite step to guarantee

$$
|z(k)|{\leq}\psi(\alpha){\cdot}max\left\{\left(\frac{\gamma}{l}\right)^{\frac{1}{\alpha}},\left(\frac{l}{v}\right)^{\frac{1}{1-\alpha}}\right\}.
\tag{44}
$$

When the sliding mode variable enters the domain Ω, combined with the analysis of the fractional order fast terminal sliding mode surface, we can see

$$
c_1e_1(k) + e_2(k) + c_2D^{\lambda}[|e_1(k)|^{\beta}sgn(e_1(k))] = g(k).
\tag{45}
$$

By the definition of GL fractional order operator,

$$
c_1e_1(k) + e_2(k) + \frac{c_2}{T^{\lambda}}\sum_{j=0}^{L}(-1)^j\binom{\lambda}{j}|e_1(k-j)|^{\beta}sgn(e_1(k)) = g(k).
\tag{46}
$$

According to Lemma 1 and equation (17), the following expression can be obtained:

$$
e_{1i}(k+1) = (1-c_{1i}T)e_{1i}(k) + \frac{c_{2i}}{T^{\lambda}}|e_{1i}(k)|^{\beta}sgn(e_{1i})(k)) + \Upsilon_i(k)
\tag{47}
$$

where $|\Upsilon_i(k)| \leq \vartheta_i$, Lemma 1 tells us, $\vartheta_i = \psi(\alpha)\varpi_i + \frac{\eta_i K_i}{T^\lambda}$, $\eta_i{\geq}max\left\{|e_{1i}(k-j)|^\beta\right\}$, $K_i = \sum_{j=1}^{L}(-1)^j\binom{\lambda}{j} = \frac{\Gamma(L+1-\lambda)}{\Gamma(1-\lambda)\Gamma(L+1)} - 1$, based on Lemma 4,

$$|e_{1i}(k)|{\leq}\psi(\beta){\cdot}max\left\{\left(\frac{\vartheta_i T^\lambda}{c_{2i}}\right)^{\frac{1}{\beta}}, \left(\frac{c_{2i}}{(1-c_{1i}T)T^\lambda}\right)^{\frac{1}{1-\beta}}\right\}. \tag{48}$$

According to the above analysis, the system errors will also converge within the bounded region when the sliding variables enter the domain.

3.4. Selection of control parameters

Through detailed control input exhibition and stability proof accomplished so far, choosing befitting control parameters concerning the factors including control input smoothness and measuring noises is also important in the acquisition of outstanding performance. Hence, a parameter selection guideline is provided here.

Selection of c_{1i}: When the sliding variables enter the equilibrium states, the parameter c_{1i} can make the sliding variables decay exponentially rapidly to ensure that the system states converge in a finite number of steps and realize the sliding variables converge quickly and precisely to the equilibrium states. Increasing c_{1i} can improve the rapidness of the system convergence, but too large a value will lead to serious system chattering. Given this trade-off, we set $c_{11} = 15$ and $c_{12} = 10$.

Selection of c_{2i}: In equation (47), if c_{2i} is too large or too small, the convergence limit of errors will be affected and chattering will occur in the system. To achieve a balance, we set $c_{21} = 100$, $c_{22} = 100$.

Selection of λ: The smaller the parameter λ is, the higher the tracking accuracy will be, but too small will seriously make the system chattering problem. For better performance, we set $\lambda = -1.7$.

Selection of β: $\beta = \frac{q_\beta}{p_\beta}$, the selection of p and q must satisfy the odd number of $p_\beta > q_\beta > 0$, making $u(k)$ have no switching item, which can effectively eliminate chattering. we set $p_\beta = 5$ and $q_\beta = 3$.

Selection of ϵ, m, δ: In equation (19), the parameter ϵ, m, δ are the system overcomes the main parameters perturbation and external disturbance, but the parameter selection inappropriate tends to cause system chattering. In order to get better performance, we set $\epsilon = 500$, $m = 2$, $\delta = 160$.

Selection of σ_i: The larger the parameter σ_i is, the faster the system reaches the sliding mode surface, but it also causes chattering. Taking this tradeoff into consideration, we set $\sigma_1 = 0.6$ and $\sigma_2 = 0.6$.

Selection of α: The smaller the parameter α is, the smaller the sliding mode bandwidth will be obtained. But it is better to set $\alpha = 0.5$ so that the boundary layer of the sliding variable is $O(T^2)$.

Remark 2: For the method proposed in this paper, the stability of convergence in finite steps has been proved in the above parts. This method combines the global memory characteristics of fractional order operators, accelerates the convergence rate of the system, and makes the system errors approach zero quickly. Moreover, adaptive law is added to adjust the width of the quasi-sliding mode band in real time to reduce the chattering of the system. Compared with CSMC algorithm[26] and FTSMC algorithm[27], it can improve the dynamic response ability of the system.

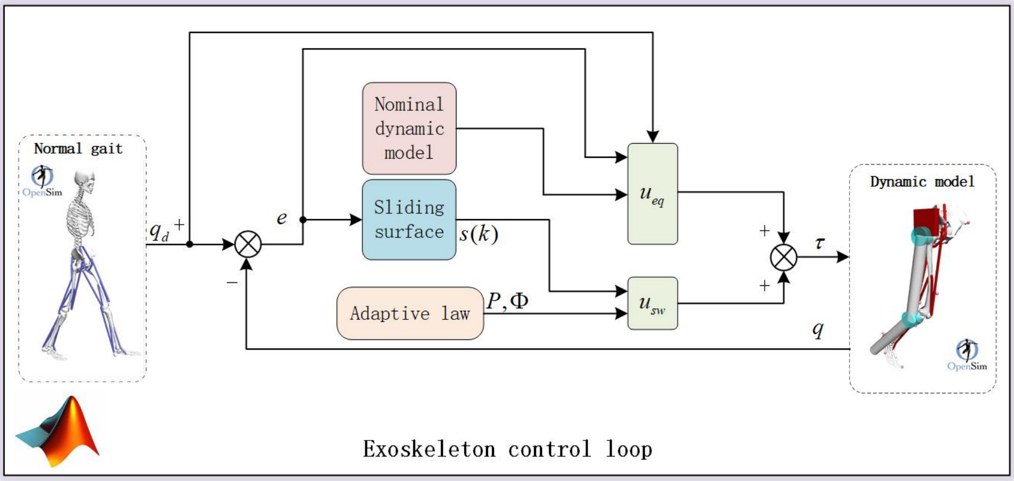


Figure 2. 2-DOF lower limb exoskeleton dynamic simulator block diagram. The exoskeleton controllers are implemented in MATLAB interfacing with the human and exoskeleton models defined in Opensim. 2-DOF: two-degree-of-freedom.

4. SIMULATION

To verify the effectiveness of the AFOFTSM control algorithm proposed in this paper on the suppression of external disturbances, we used OpenSim and Matlab software to develop a lower limb exoskeleton co-simulation control system to simulate the motion state of the human body wearing lower limb exoskeleton rehabilitation robot, which is depicted in Figure 2. OpenSim is open-source software for modeling, simulating, controlling, and analyzing the human neuromusculoskeletal system, developed by the National Institutes of Health (NIH) Center for Biomedical Computing at Stanford University[28,29]. OpenSim is a developable platform. Using the template and experimental data provided by OpenSim, we built a human musculoskeletal model with a height of 1814mm and a weight of 72.6kg and wore the lower extremity exoskeleton on the model[30]. At the same time, the controller algorithm was written in Matlab to calculate the input torque of each exoskeleton joint and control the lower limb exoskeleton to drive the human body to move together.

In the following simulation, the established simulation system was run in Matlab software, the simulation time was 3s, the simulation step size was 0.001s, and the standard hip and knee joints of a healthy subject when walking horizontally were used as the reference track.

At the same time, the discrete CSMC algorithm[26] and the discrete FTSMC algorithm[27] are respectively applied to the dynamics model of the lower limb exoskeleton robot for comparison. The design methods of CSMC and FTSMC are as follows:

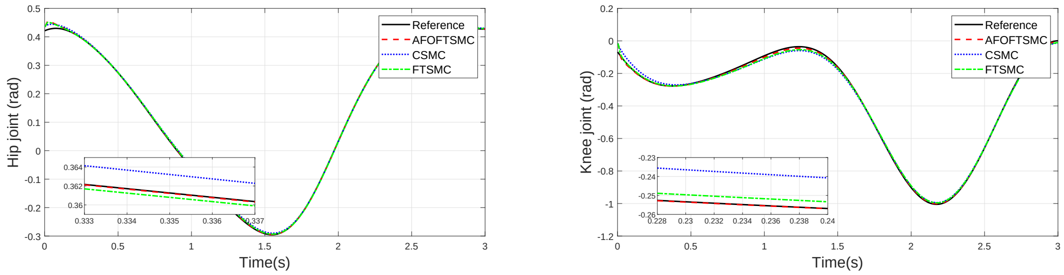
1) CSMC

For simplicity, the CSMC[26] control input u_{csmc} is given directly:

$$u_{csmc} = [C_{ec}b(k)]^{-1} [C_{ec}x_d (k+1) - C_{ec}f(k) - (1 - q_cT) s_c(k) + \varepsilon_c T sgn (s_c(k))] . \tag{49}$$

In the formula (49), $s_c(k)$ is the sliding mode variable, which is defined as follows:

$$s_c(k) = c_c e_c(k) + \dot{e}_c(k) \tag{50}$$



(a) Tracking profiles of hip joint

(b) Tracking profiles of knee joint

Figure 3. Trajectory tracking performance of the robotic exoskeleton by using CSMC, FTSMC, and AFOFTSMC control strategy. (a) trajectory tracking of the hip joint. (b) trajectory tracking of the knee joint. CSMC: conventional sliding mode control; FTSMC: fast terminal sliding mode control; AFOFTSMC: adaptive fractional order fast terminal sliding mode controller.

$e_c(k)$ is the tracking error under the control of CSMC, $c_c = \begin{bmatrix} 15 & 10 \end{bmatrix}$, $\varepsilon_c = \begin{bmatrix} 0.01 & 0.01 \end{bmatrix}$, $C_{ec} = \begin{bmatrix} 15 & 0 & 1 & 0 \\ 0 & 10 & 0 & 1 \end{bmatrix}$.

2) FTSMC

An FTSMC control law[27] is also designed for comparison, and the control input u_{ftsmc} is shown below:

$$\begin{aligned} u_{ftsmc} =& [C_{fc}b(k)]^{-1}[C_{fc}(x_d(k+1) - f(k)) - (1 - q_cT)s_f(k) \\ & + \varepsilon_c T sgn(s(k)) - C_{fc2}|s_f(k)|^{\alpha} sgn[s_f(k)]]. \end{aligned} \tag{51}$$

In Formula (51), s_f is a sliding mode variable, defined as follows:

$$s_f(k) = c_{fc1}e_f(k) + \dot{e}_f(k) + C_{fc2}\left|e_f(k)\right|^{\alpha} sgn\left[e_f(k)\right] \tag{52}$$

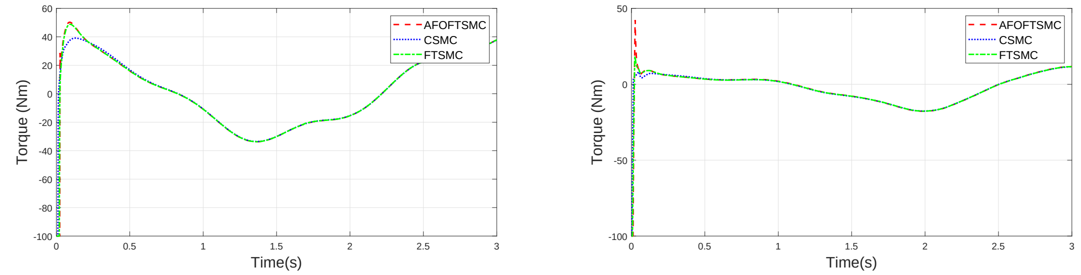
$e_f(k)$ is the tracking error under FTSM control, $c_{fc1} = \begin{bmatrix} 15 & 10 \end{bmatrix}$, $c_{fc2} = \begin{bmatrix} 0.05 & 0.05 \end{bmatrix}$, $C_{fc} = \begin{bmatrix} 15 & 0 & 1 & 0 \\ 0 & 10 & 0 & 1 \end{bmatrix}$.

To test the robustness of the three algorithms to external disturbances, the same external disturbances are set for the three controllers respectively, and the external disturbances are set as:

$$\xi(k) = \begin{bmatrix} 5sin(k) \\ -5sin(k) \end{bmatrix} (Nm). \tag{53}$$

The simulation results are shown in Figure 3-6, which are position tracking of the hip joint and knee joint, control input signals, position tracking errors, and sliding mode surface function respectively. In Figure 3, compared with CSMC and FTSMC, the control algorithm proposed in this paper can track the gait trajectory more accurately, and the tracking trajectory of AFOFTSMC is closer to the reference trajectory.

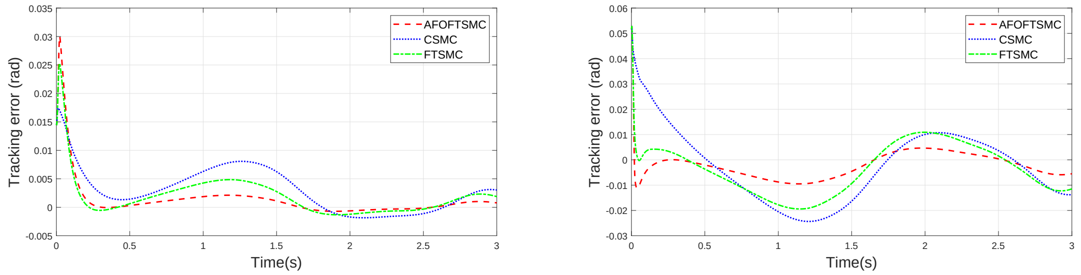
Moreover, it can be seen from Figure 4 that the control inputs of AFOFTSMC do not need to give greater control efforts to maintain higher track tracking accuracy and effectively eliminate the impact of external interference. According to the comparison of tracking errors in Figure 5, the AFOFTSMC algorithm has the smallest steady-state tracking error, which can ensure that the system error converges in a finite time. In Figure 6, the sliding mode variables of the three controllers can move rapidly into the quasi-sliding mode band, and the control algorithm proposed in this paper can effectively reduce the width of the quasi-sliding mode band.



(a) Control inputs of hip joint

(b) Control inputs of knee joint

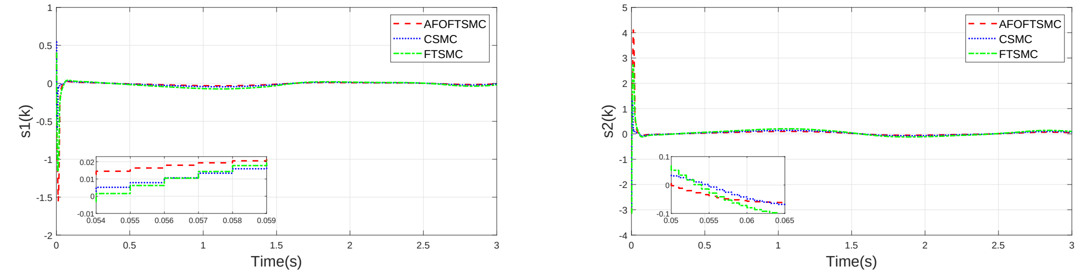
Figure 4. AFOFTSMC control input comparison with CSMC and FTSMC. CSMC: conventional sliding mode control; FTSMC: fast terminal sliding mode control; AFOFTSMC: adaptive fractional order fast terminal sliding mode controller.



(a) Tracking errors of hip joint

(b) Tracking errors of knee joint

Figure 5. Tracking errors comparison of CSMC, FTSMC, and AFOFTSMC in simulations. (a) errors comparison of the hip joint; (b) errors comparison of the knee joint. CSMC: conventional sliding mode control; FTSMC: fast terminal sliding mode control; AFOFTSMC: adaptive fractional order fast terminal sliding mode controller.



(a) Sliding variables $s_1(k)$

(b) Sliding variables $s_2(k)$

Figure 6. Sliding variables for the three sliding surfaces.

To provide more quantitative proof, three indicators are compared in Table 1 to evaluate the performance of three different control strategies. Mean absolute error (MAE) with $MAE = \frac{1}{N}\sum_{k=1}^{N}|e(k)|$, maximum tracking error (MTE) with $MTE = max|e(k)|$, root mean square error (RMSE) with $RMSE = \sqrt{\frac{1}{N}\sum_{k=1}^{N}[e(k)]^2}$. The results are shown in Table 1. The discrete adaptive fractional order fast terminal sliding mode control algorithm proposed in this paper is applied to the lower limb exoskeleton robot to achieve better trajectory tracking control. The generated RMSE and MAE are both minima, but the MTE is maximum, which is due to overdrive caused by fast response speed.

To enhance SMC's ability to resist external disturbances, we use fractional order sliding mode surfaces to accel-

Table 1. Performance indicators of the three controllers

Controllers	RMSE of hip	MAE of hip	MTE of hip	RMSE of knee	MAE of knee	MTE of knee
AFOFTSMC (Proposed)	0.0056 rad	0.0026 rad	0.0300 rad	0.0075 rad	0.0057 rad	0.0529 rad
CSMC [26]	0.0066 rad	0.0056 rad	0.0174 rad	0.0189 rad	0.0166 rad	0.0529 rad
FTSMC [27]	0.0056 rad	0.0370 rad	0.0253 rad	0.0128 rad	0.0105 rad	0.0529 rad

erate the convergence of system errors and make adaptive adjustments to control law parameters. Compared with the traditional control methods, AFOFTSMC has better potential to be applied to the actual exoskeleton rehabilitation robot and help patients to carry out rehabilitation training in the early stage of rehabilitation. In addition to the field of rehabilitation robots, the algorithm can also be extended to other fields to achieve better control effects, such as chaotic systems, robot manipulators, and quadrotor UAVs.

5. CONCLUSIONS

In this paper, we study a human gait trajectory-tracking control issue of lower limb exoskeleton rehabilitation robot. Firstly, the dynamic properties of the lower limb exoskeleton rehabilitation robot were analyzed. Then, a new AFOFTSMC algorithm was developed for the lower limb exoskeleton robot with uncertain parameters and unknown external interference, where the fractional order fast terminal sliding mode function was introduced to achieve rapid convergence in finite time. Particularly, the unknown dynamic part of the exoskeleton was processed by adaptive law, and the width of the quasi-sliding mode band was adjusted in real-time to ensure that the sliding mode variables quickly enter the quasi-sliding mode band. Moreover, the stability of the whole control system was verified in the Lyapunov sense. To illustrate the effectiveness of the proposed controller, we compared the simulation results of CSCM, FTSMC, and AFOFTSMC on the MATLAB-Opensim co-simulation platform. The simulation results showed that the adaptive fractional order fast terminal sliding mode controller has the characteristics of high precision, fast response, and strong robustness for robot trajectory tracking.

DECLARATIONS

Authors' contributions

Implemented the methodologies presented and wrote the paper: Zhou Y, Sun Z

Performed oversight and leadership responsibility for the research activity planning and execution, as well as developed ideas and evolution of overarching research aims: Sun Z, Chen B, Wang T

Performed critical review, commentary, and revision, as well as providing technical guidance: Sun Z, Chen B, Wu X, Huang G

All authors have revised the text and agreed to the published version of the manuscript.

Availability of data and materials

Not applicable.

Financial support and sponsorship

This work was supported by the Primary Research and Development Plan of Zhejiang Province (No. 2022C03029)

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Copyright

REFERENCES

1. Liu WL, Yin BL, Yan BB. A survey on the exoskeleton rehabilitation robot for the lower limbs. In: 2016 2nd International Conference on Control, Automation and Robotics (ICCAR); 2016 Apr 28-30; Hong Kong, China. IEEE; 2016. pp. 90–94.
2. Meng W, Liu Q, Zhou ZD, Ai QS, Sheng B, Xie SQ. Recent development of mechanisms and control strategies for robot-assisted lower limb rehabilitation. *Mechatronics* 2015;31:132–45. DOI
3. Han J, Yang SY, Xia L, Chen YH. Deterministic adaptive robust control with a novel optimal gain design approach for a fuzzy 2-DOF lower limb exoskeleton robot system. *IEEE Trans Fuzzy Syst* 2021;29:2373-87. DOI
4. Sun W, Lin JW, Su SF, Wang N, Er MJ. Reduced adaptive fuzzy decoupling control for lower limb exoskeleton. *IEEE Trans Cybern* 2021;51:1099-109. DOI
5. Long Y, Du ZJ, Cong L, Wang WD, Zhang ZM, Dong W. Active disturbance rejection control based human gait tracking for lower extremity rehabilitation exoskeleton. *ISA Trans* 2017;67:389-97. DOI
6. Asl HJ, Narikiyo T, Kawanishi M. Neural network-based bounded control of robotic exoskeletons without velocity measurements. *Contr Eng Pract* 2018;80:94-104. DOI
7. Torres FJ, Guerrero GV, García CD, Gomez JF, Adam M and Escobar RF. Master-slave synchronization of robot manipulators driven by induction motors. *IEEE Latin Am Trans* 2016;14:3986-91. (in Spanish) DOI
8. Ahmed S, Wang HP, Tian Y. Robust adaptive fractional-order terminal sliding mode control for lower-limb exoskeleton. *Asian J of Contr* 2019;21:473-82. DOI
9. Narayan J, Abbas M, Patel B, Dwivedy SK. A Singularity-free terminal sliding mode control of an uncertain paediatric exoskeleton system. In: 2022 5th International Conference on Advanced Systems and Emergent Technologies (IC_ASET); 2022 Mar 22-25; Hammamet, Tunisia. IEEE; 2022. pp. 198–203.
10. Cao SB, Cao GZ, Zhang YP, Ling ZQ, He BB, Huang SD. Fast-terminal sliding mode control based on dynamic boundary layer for lower limb exoskeleton rehabilitation robot. In: 2021 IEEE 11th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER); 2021 July 27-31; Jiaxing, China. IEEE; 2021. pp. 453–458.
11. Chen SB, Beigi A, Yousefpour A, et al. Recurrent neural network-based robust nonsingular sliding mode control with input saturation for a non-holonomic spherical robot. *IEEE Access* 2020;8:188441-53. DOI
12. Fei JT, Feng ZL. Fractional-order finite-time super-twisting sliding mode control of micro gyroscope based on double-loop fuzzy neural network. *IEEE Trans Syst Man Cybern, Syst* 2021;51:7692-706. DOI
13. Wang YY, Gu LY, Xu YH, Cao XX. Practical tracking control of robot manipulators with continuous fractional-order nonsingular terminal sliding mode. *IEEE Trans Ind Electron* 2016;63:6194-204. DOI
14. Yang Y, Chen YQ, Chu YZ, Wang Y, Liang Q. Fractional order adaptive sliding mode controller for permanent magnet synchronous motor. In: 2016 35th Chinese Control Conference (CCC); 2016 July 27-29; Chengdu, China. IEEE; 2016. pp. 3412–3416.
15. Lavín-Delgado JE, Chávez-Vázquez S, Gómez-Aguilar JF, Alassafi MO, Alsaadi FE, Ahmad AM. Intelligent Neural Integral Sliding-mode Controller for a space robotic manipulator mounted on a free-floating satellite. *Adv Space Res* 2022; Epub ahead of print. DOI
16. Lavín-Delgado JE, Beltrán ZZ, Gómez-Aguilar JF, Pérez-Careta E. Controlling a quadrotor UAV by means of a fractional nested saturation control. *Adv Space Res* 2022; Epub ahead of print. DOI
17. Li JF, Jahanshahi H, Kacar S, et al. On the variable-order fractional memristor oscillator: Data security applications and synchronization using a type-2 fuzzy disturbance observer-based robust control. *Chaos, Solitons & Fractals* 2021;145:110681. DOI
18. Wang YL, Jahanshahi H, Bekiros S, Bezzina F, Chu YM, Aly AA. Deep recurrent neural networks with finite-time terminal sliding mode control for a chaotic fractional-order financial system with market confidence. *Chaos, Solitons & Fractals* 2021;146:110881. DOI
19. Xiong PY, Jahanshahi H, Alcaraz R, Chu YM, Gómez-Aguilar JF, Alsaadi FE. Spectral entropy analysis and synchronization of a multi-stable fractional-order chaotic system using a novel neural network-based chattering-free sliding mode technique. *Chaos, Solitons & Fractals* 2021;144:110576. DOI
20. Li SH, Du HB, Yu XH. Discrete-time terminal sliding mode control systems based on euler's discretization. *IEEE Trans Automat Contr* 2014;59:546-52. DOI
21. Chen B, Hu GQ, Ho DWC, Yu L. Distributed Estimation and Control for Discrete Time-Varying Interconnected Systems. *IEEE Trans Automat Contr* 2022;67:2192-207. DOI
22. Sun GH, Ma ZQ, Yu JY. Discrete-time fractional order terminal sliding mode tracking control for linear motor. *IEEE Trans Ind Electron* 2018;65:3386-94. DOI
23. Ajjanaromvat N, Parnichkun M. Trajectory tracking using online learning LQR with adaptive learning control of a leg-exoskeleton for disorder gait rehabilitation. *Mechatronics* 2018;51:85-96. DOI
24. Neuman CP, Tourassis VD. Discrete dynamic robot models. *IEEE Trans Syst , Man, Cybern* 1985;SMC-15:193-204. DOI

25. Zhou Y, Hu ZY, Sun Z, Wang T, Chen B. Covariance intersection fusion approach for gait estimation of lower limb rehabilitation Exoskeleton Robot. In: 2022 5th International Symposium on Autonomous Systems (ISAS); 2022 April 08-10; Hangzhou, China. IEEE; 2022. pp. 1–6.
26. Gao WB, Wang YF, A HMF. Discrete-time variable structure control systems. *IEEE Trans Ind Electron* 1995;42:117–22. DOI
27. Wang Z, Li SH, Li Q. Discrete-time fast terminal sliding mode control design for DC–DC buck converters with mismatched disturbances. *IEEE Trans Ind Inf* 2020;16:1204–13. DOI
28. Delp SL, Anderson FC, Arnold AS, et al. OpenSim: open-source software to create and analyze dynamic simulations of movement. *IEEE Trans Biomed Eng* 2007;54:1940-50. DOI
29. Seth A, Hicks JL, Uchida TK, et al. OpenSim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement. *PLoS Comput Biol* 2018;14:e1006223. DOI
30. Mi WM, Zhang T. Fuzzy variable impedance adaptive robust control algorithm of exoskeleton robots. In: 2019 Chinese Control Conference (CCC); 2019 July 27-30; Guangzhou, China. IEEE; 2019. pp. 4302–7.

AUTHOR INSTRUCTIONS

1. Submission Overview

Before you decide to publish with *Intelligence & Robotics* (*IR*), please read the following items carefully and make sure that you are well aware of Editorial Policies and the following requirements.

1.1 Topic Suitability

The topic of the manuscript must fit the scope of the journal. Please refer to Aims and Scope for more information.

1.2 Open Access and Copyright

The journal adopts Gold Open Access publishing model and distributes content under the Creative Commons Attribution 4.0 International License. Copyright is retained by authors. Please make sure that you are well aware of these policies.

1.3 Publication Fees

IR is an open access journal. When a paper is accepted for publication, authors are required to pay Article Processing Charges (APCs) to cover its editorial and production costs. The APC for each submission is $1200. There are no additional charges based on color, length, figures, or other elements. For more details, please refer to OAE Publication Fees.

1.4 Language Editing

All submissions are required to be presented clearly and cohesively in good English. Authors whose first language is not English are advised to have their manuscripts checked or edited by a native English speaker before submission to ensure the high quality of expression. A well-organized manuscript in good English would make the peer review even the whole Editorial handling more smoothly and efficiently.

If needed, authors are recommended to consider the language editing services provided by OAE to ensure that the manuscript is written in correct scientific English before submission. An extra charge is required to enjoy this service. Please visit https://www.oaepublish.com/index/author_services or contact English-Editing@oaepublish.com for more details.

1.5 Work Funded by the National Institutes of Health

If an accepted manuscript was funded by National Institutes of Health (NIH), the author may inform editors of the NIH funding number. The editors are able to deposit the paper to the NIH Manuscript Submission System on behalf of the author.

2. Submission Preparation

2.1 Cover Letter

A cover letter is required to be submitted accompanying each manuscript. Here is a guideline of a cover letter for authors' consideration:
List the highlights of the current manuscript and no more than 5 short sentences;
All authors have read the final manuscript, have approved the submission to the journal, and have accepted full responsibilities pertaining to the manuscript's delivery and contents;
Clearly state that the manuscript is an original work on its own merit, that it has not been previously published in whole or in part, and that it is not being considered for publication elsewhere;
No materials are reproduced from another source (if there is material in your manuscript that has been reproduced from another source, please state whether you have obtained permission from the copyright holder to use them);
Conflicts of interest statement;
If the manuscript is contributed to a Special Issue, please also mention it in the cover letter;
If the manuscript was presented partly or entirely in a conference, the author should clearly state the background information of the event, including the conference name, time, and place in the cover letter.

2.2 Types of Manuscripts

There is no restriction on the length of manuscripts, number of figures, tables and references, provided that the manuscript is concise and comprehensive. The journal publishes Research Article, Review, Technical Note, etc. For more details about paper type, please refer to the following table.

Manuscript Type	Definition	Word Limit	Abstract	Keywords	Main Text Structure

Research Article	A Research Article is a seminal and insightful research study and showcases that often involves modern techniques or methodologies. Authors should justify that their work is of novel findings.	8000 max	The abstract should state briefly the purpose of the research, the principal results and major conclusions. No more than 250 words.	3-8 keywords	The main content should include four sections: Introduction, Methods, Results and Discussion.
Review	A Review should be an authoritative, well balanced, and critical survey of recent progress in an attractive or a fundamental research field.	5000 max /10000 max	Unstructured abstract. No more than 250 words.	3-8 keywords	The main text may consist of several sections with unfixed section titles. We suggest that the author include an "Introduction" section at the beginning, several sections with unfixed titles in the middle part, and a "Conclusions" section at the end.
Technical Note	A Technical Note is a short article giving a brief description of a specific development, technique, or procedure, or it may describe a modification of an existing technique, procedure or device applied in research.	3500 max	Unstructured abstract. No more than 250 words.	3-8 keywords	/
Editorial	An Editorial is a short article describing news about the journal or opinions of senior Editors or the publisher.	1000 max	None required	None required	/
Commentary	A Commentary is to provide comments on a newly published article or an alternative viewpoint on a certain topic.	2500 max	Unstructured abstract. No more than 250 words.	3-8 keywords	/
Perspective	A Perspective provides personal points of view on the state-of-the-art of a specific area of knowledge and its future prospects.	2000 max	Unstructured abstract. No more than 250 words.	3-8 keywords	/

2.3 Manuscript Structure

2.3.1 Front Matter

2.3.1.1 Title

The title of the manuscript should be concise, specific and relevant, with no more than 16 words if possible.

2.3.1.2 Authors and Affiliations

Authors' full names should be listed. The initials of middle names can be provided. The affiliations and email addresses for all authors should be listed. At least one author should be designated as the corresponding author. In addition, corresponding authors are suggested to provide their Open Researcher and Contributor ID upon submission. Please note that any change to authorship is not allowed after manuscript acceptance. The authors' affiliations should be provided in this format: department, institution, city, postcode, country.

2.3.1.3 Abstract

The abstract should be a single paragraph with word limitation and specific structure requirements (for more details please refer to Types of Manuscripts). It usually describes the main objective(s) of the study, explains how the study was done, including any model organisms used, without methodological detail, and summarizes the most important results and their significance. The abstract must be an objective representation of the study: it is not allowed to contain results that are not presented and substantiated in the manuscript, or exaggerate the main conclusions. Citations should not be included in the abstract.

2.3.1.4 Graphical Abstract

The graphical abstract is essential as this can catch first view of your publication by readers. We recommend you submit an eye-catching figure. It should summarize the content of the article in a concise graphical form. It is recommended to use it because this can make online articles get more attention.

The graphical abstract should be submitted as a separate document in the online submission system. Please provide an image with a minimum of 531 × 1328 pixels (h × w) or proportionally more. The image should be readable at a size of 5 cm × 13 cm using a regular screen resolution of 96 dpi. Preferred file types: TIFF, PSD, AI, JPEG, and EPS files.

2.3.1.5 Keywords

Three to eight keywords should be provided, which are specific to the article, yet reasonably common within the subject discipline.

Sections 2.3.1.1 and 2.3.1.2 should appear in all manuscript types.

2.3.2 Main Text

Manuscripts of different types are structured with different sections of content. Please refer to Types of Manuscripts to make sure which sections should be included in the manuscripts.

2.3.2.1 Introduction

The introduction should contain background that puts the manuscript into context, allow readers to understand why the study is important, include a brief review of key literature, and conclude with a brief statement of the overall aim of the work and a comment about whether that aim was achieved. Relevant controversies or disagreements in the field should be introduced as well.

2.3.2.2 Methods

The methods should contain sufficient details to allow others to fully replicate the study. New methods and protocols should be described in detail while well-established methods can be briefly described or appropriately cited. Statistical terms, abbreviations, and all symbols used should be defined clearly. Protocol documents for clinical trials, observational studies, and other non-laboratory investigations may be uploaded as supplementary materials.

2.3.2.3 Results

This section contains the findings of the study. Results of statistical analysis should also be included either as text or as tables or figures if appropriate. Authors should emphasize and summarize only the most important observations. Data on all primary and secondary outcomes identified in the section Methods should also be provided. Extra or supplementary materials and technical details can be placed in supplementary documents.

2.3.2.4 Discussion

This section should discuss the implications of the findings in context of existing research and highlight limitations of the study. Future research directions may also be mentioned.

2.3.2.5 Conclusion

It should state clearly the main conclusions and include the explanation of their relevance or importance to the field.

2.3.3 Back Matter

The following sections should appear in all manuscript types.

2.3.3.1 Acknowledgments

Anyone who contributed towards the article but does not meet the criteria for authorship, including those who provided professional writing services or materials, should be acknowledged. Authors should obtain permission to acknowledge from all those mentioned in the Acknowledgments section. This section is not added if the author does not have anyone to acknowledge.

2.3.3.2 Authors' Contributions

Each author is expected to have made substantial contributions to the conception or design of the work, or the acquisition, analysis, or interpretation of data, or the creation of new software used in the work, or have drafted the work or substantively revised it.

Please use Surname and Initial of Forename to refer to an author's contribution. For example: made substantial contributions to conception and design of the study and performed data analysis and interpretation: Salas H, Castaneda WV; performed data acquisition, as well as providing administrative, technical, and material support: Castillo N, Young V.

If an article is single-authored, please include "The author contributed solely to the article." in this section.

2.3.3.3 Availability of Data and Materials

In order to maintain the integrity, transparency and reproducibility of research records, authors should include this section in their manuscripts, detailing where the data supporting their findings can be found. Data can be deposited into data repositories or published as supplementary information in the journal. Authors who cannot share their data should state that the data will not be shared and explain it. If a manuscript does not involve such issues, please state "Not applicable." in this section.

2.3.3.4 Financial Support and Sponsorship

All sources of funding for the study reported should be declared. The role of the funding body in the experiment design, collection, analysis and interpretation of data, and writing of the manuscript should be declared. Any relevant grant numbers and the link of funder's website should be provided if any. If the study is not involved with this issue, state "None." in this section.

2.3.3.5 Conflicts of Interest

Authors must declare any potential conflicts of interest that may be perceived as inappropriately influencing the representation or interpretation of reported research results. If there are no conflicts of interest, please state "All authors declared that there are no conflicts of interest." in this section. Some authors may be bound by confidentiality agreements. In such cases, in place of itemized disclosures, we will require authors to state "All authors declared that they are bound by confidentiality agreements that prevent them from disclosing their conflicts of interest in this work.". If authors are unsure whether conflicts of interest exist, please refer to the "Conflicts of Interest" of *IR* Editorial Policies for a full explanation.

2.3.3.6 Consent for Publication

Manuscripts containing individual details, images or videos, must obtain consent for publication from that person, or in the case of children, their parents or legal guardians. If the person has died, consent for publication must be obtained from the next of kin of the participant. Manuscripts must include a statement that written informed consent for publication was obtained. Authors do not have to submit such content accompanying the manuscript. However, these documents must be available if requested. If the manuscript does not involve this issue, state "Not applicable." in this section.

2.3.3.7 Copyright

Authors retain copyright of their works through a Creative Commons Attribution 4.0 International License that clearly states how readers can copy, distribute, and use their attributed research, free of charge. A declaration "© The Author(s) 2023." will be added to each article. Authors are required to sign License to Publish before formal publication.

2.3.3.8 References

References should be numbered in order of appearance at the end of manuscripts. In the text, reference numbers should be placed in square brackets and the corresponding references are cited thereafter. List all authors when the number of authors is less than or equal to six, if there are more than six authors, only the first three authors' names should be listed, other authors' names should be omitted and replaced with "et al.". The journal's name should be required to be italicized and the journal references should have corresponding DOI numbers. Information from manuscripts accepted but not published should be cited in the text as "Unpublished material" with written permission from the source. Journal names should be abbreviated according to the List of Title Word Abbreviations.

References should be described as follows, depending on the types of works:

Types	**Examples**
Journal articles by individual authors	Cao MS, Pan LX, Gao YF, et al. Neural network ensemble-based parameter sensitivity analysis in civil engineering systems. *Neural Comput Applic* 2017;28:1583-90. [DOI: 10.1007/s00521-015-2132-4]

Organization as author	Diabetes Prevention Program Research Group. Hypertension, insulin, and proinsulin in participants with impaired glucose tolerance. *Hypertension* 2002;40:679-86. [DOI: 10.1161/01.HYP.0000035706.28494.09]
Both personal authors and organization as author	Vallancien G, Emberton M, Harving N, van Moorselaar RJ; Alf-One Study Group. Sexual dysfunction in 1,274 European men suffering from lower urinary tract symptoms. *J Urol* 2003;169:2257-61. [PMID: 12771764 DOI: 10.1097/01.ju.0000067940.76090.73]
Journal articles not in English	Mao X, Ding YK. Sentiment feature analysis and harmonic sense evaluation of images. *J Electronic* 2001;29:23-7. (in Chinese)
Journal articles ahead of print	Cong Y, Gu CJ, Zhang T, Gao YJ. Underwater Robot Sensing Technology: A Survey. *Fundamental Res* 2021; Epub ahead of print [DOI: 10.1016/j.fmre.2021.03.002]
Books	Gaydon AG, Wolfhard HG. Flames. 2nd ed. London: Chapman and Hall Ltd.; 1960. pp. 10-20.
Book chapters	Goel AK, Fitzgerald T, Parashar P. Analogy and metareasoning: Cognitive strategies for robot learning. In: Lawless WF, Mittu R, Sofge DA, Editors. Human-Machine Shared Contexts. Academic Press; 2020. pp. 23-44.
Online resource	Intel Technology Journal. Developing smart toys - from idea to product. Available from: https://www.intel.com/content/dam/www/public/us/en/documents/research/2001-vol05-iss-4-intel-technology-journal.pdf. [Last accessed on 20 Feb 2021]
Conference proceedings	Harnden P, Joffe JK, Jones WG, Editors. Germ cell tumours V. Proceedings of the 5th Germ Cell Tumour Conference; 2001 Sep 13-15; Leeds, UK. New York: Springer; 2002.
Conference paper	Christensen S, Oppacher F. An analysis of Koza's computational effort statistic for genetic programming. In: Foster JA, Lutton E, Miller J, Ryan C, Tettamanzi AG, editors. Genetic programming. EuroGP 2002: Proceedings of the 5th European Conference on Genetic Programming; 2002 Apr 3-5; Kinsdale, Ireland. Berlin: Springer; 2002. pp. 182-91.
Unpublished material	Tian D, Araki H, Stahl E, Bergelson J, Kreitman M. Signature of balancing selection in Arabidopsis. *Proc Natl Acad Sci U S A*. Forthcoming 2002.

The journal also recommends that authors prepare references with a bibliography software package, such as EndNote to avoid typing mistakes and duplicated references.

2.3.3.9 Supplementary Materials

Additional data and information can be uploaded as Supplementary Materials to accompany the manuscripts. The supplementary materials will also be available to the referees as part of the peer-review process. Any file format is acceptable, such as data sheet (word, excel, csv, cdx, fasta, pdf or zip files), presentation (powerpoint, pdf or zip files), image (cdx, eps, jpeg, pdf, png or tiff), table (word, excel, csv or pdf), audio (mp3, wav or wma) or video (avi, divx, flv, mov, mp4, mpeg, mpg or wmv). All information should be clearly presented. Supplementary materials should be cited in the main text in numeric order (e.g., Supplementary Figure 1, Supplementary Figure 2, Supplementary Table 1, Supplementary Table 2, etc.). The style of supplementary figures or tables complies with the same requirements on figures or tables in main text. Videos and audios should be prepared in English, and limited to a size of 500 MB.

2.4 Manuscript Format

2.4.1 File Format

Manuscript files can be in DOC and DOCX formats and should not be locked or protected.

Manuscript prepared in LaTex must be collated into one ZIP folder (including all source files and images, so that the Editorial Office can recompile the submitted PDF).

When preparing manuscripts in different file formats, please use the corresponding Manuscript Templates.

2.4.2 Length

The word limit is specified in the item “Types of Manuscripts”. There are no restrictions on number of figures or number of supporting documents. Authors are encouraged to present and discuss their findings concisely.

2.4.3 Language

Manuscripts must be written in English.

2.4.4 Multimedia Files

The journal supports manuscripts with multimedia files. The requirements are listed as follows:

Video or audio files are only acceptable in English. The presentation and introduction should be easy to understand. The frames should be clear, and the speech speed should be moderate;

A brief overview of the video or audio files should be given in the manuscript text;

The video or audio files should be limited to a size of up to 500 MB;

Please use professional software to produce high-quality video files, to facilitate acceptance and publication along with the submitted article. Upload the videos in mp4, wmv, or rm format (preferably mp4) and audio files in mp3 or wav format.

2.4.5 Figures

Figures should be cited in numeric order (e.g., Figure 1, Figure 2) and placed after the paragraph where it is first cited;
Figures can be submitted in format of TIFF, PSD, AI, EPS or JPEG, with resolution of 300-600 dpi;
Figure caption is placed under the Figure;
Diagrams with describing words (including, flow chart, coordinate diagram, bar chart, line chart, and scatter diagram, *etc.*) should be editable in word, excel or powerpoint format. Non-English information should be avoided;
Labels, numbers, letters, arrows, and symbols in figure should be clear, of uniform size, and contrast with the background;
Symbols, arrows, numbers, or letters used to identify parts of the illustrations must be identified and explained in the legend;
Internal scale (magnification) should be explained and the staining method in photomicrographs should be identified;
All non-standard abbreviations should be explained in the legend;
Permission for use of copyrighted materials from other sources, including re-published, adapted, modified, or partial figures and images from the internet, must be obtained. It is authors' responsibility to acquire the licenses, to follow any citation instruction requested by third-party rights holders, and cover any supplementary charges.

2.4.6 Tables

Tables should be cited in numeric order and placed after the paragraph where it is first cited;
The table caption should be placed above the table and labeled sequentially (e.g., Table 1, Table 2);
Tables should be provided in editable form like DOC or DOCX format (picture is not allowed);
Abbreviations and symbols used in table should be explained in footnote;
Explanatory matter should also be placed in footnotes;
Permission for use of copyrighted materials from other sources, including re-published, adapted, modified, or partial tables from the internet, must be obtained. It is authors' responsibility to acquire the licenses, to follow any citation instruction requested by third-party rights holders, and cover any supplementary charges.

2.4.7 Abbreviations

Abbreviations should be defined upon first appearance in the abstract, main text, and in figure or table captions and used consistently thereafter. Non-standard abbreviations are not allowed unless they appear at least three times in the text. Commonly-used abbreviations, such as DNA, RNA, ATP, *etc.*, can be used directly without definition. Abbreviations in titles and keywords should be avoided, except for the ones which are widely used.

2.4.8 Italics

General italic words like *vs.*, *et al.*, *etc.*, *in vivo*, *in vitro*; *t* test, *F* test, *U* test; related coefficient as *r*, sample number as *n*, and probability as *P*; names of genes; names of bacteria and biology species in Latin.

2.4.9 Units

SI Units should be used. Imperial, US customary and other units should be converted to SI units whenever possible. There is a space between the number and the unit (i.e., 23 mL). Hour, minute, second should be written as h, min, s.

2.4.10 Numbers

Numbers appearing at the beginning of sentences should be expressed in English. When there are two or more numbers in a paragraph, they should be expressed as Arabic numerals; when there is only one number in a paragraph, number < 10 should be expressed in English and number > 10 should be expressed as Arabic numerals. 12345678 should be written as 12,345,678.

2.4.11 Equations

Equations should be editable and not appear in a picture format. Authors are advised to use either the Microsoft Equation Editor or the MathType for display and inline equations.
Display equations should be numbered consecutively, using Arabic numbers in parentheses;
Inline equations should not be numbered, with the same/similar size font used for the main text.

2.4.12 Headings

In the main body of the paper, three different levels of headings may be used.
Level one headings: they should be in bold, and numbered using Arabic numbers, such as **1. INTRODUCTION**, and **2. METHODS**, with all letters capitalized;
Level two headings: they should be in bold and numbered after the level one heading, such as **2.1 Statistical analyses**, **2.2** ..., **2.3**..., *etc.*, with the first letter capitalized;
Level three headings: they should be italicized, and numbered after the level two heading, such as *2.1.1 Data distributions*,and *2.1.2 outliers and linear regression*, with the first letter capitalized.

2.4.13 Text Layout

As the electronic submission will provide the basic material for typesetting, it is important to prepare papers in the general editorial style of the journal.
The font is Times New Roman;
The font size is 12pt;
Single column, 1.5× line spacing;
Insert one line break (one Return) before the heading and paragraph, if the heading and paragraph are adjacent, insert a line break before the heading only;
No special indentation;
Alignment is left end;
Insert consecutive line numbers;
For other details please refer to the Manuscript Templates.

2.5 Submission Link

Submit an article via https://oaemesas.com/login?JournalId=ir.

3. Publication Ethics Statement

OAE is a member of the Committee on Publication Ethics (COPE). We fully adhere to its Code of Conduct and to its Best Practice Guidelines.

The Editors of this journal enforce a rigorous peer-review process together with strict ethical policies and standards to guarantee to add high-quality scientific works to the field of scholarly publication. Unfortunately, cases of plagiarism, data falsification, image manipulation, inappropriate authorship credit, and the like, do arise. The Editors of *IR* take such publishing ethics issues very seriously and are trained to proceed in such cases with zero tolerance policy.

Authors wishing to publish their papers in *IR* must abide by the following:

The author(s) must disclose any possibility of a conflict of interest in the paper prior to submission;
The authors should declare that there is no academic misconduct in their manuscript in the cover letter;
Authors should accurately present their research findings and include an objective discussion of the significance of their findings;
Data and methods used in the research need to be presented in sufficient detail in the manuscript so that other researchers can replicate the work;
Authors should provide raw data if referees and the Editors of the journal request;
Simultaneous submission of manuscripts to more than one journal is not tolerated;
Republishing content that is not novel is not tolerated (for example, an English translation of a paper that is already published in another language will not be accepted);
The manuscript should not contain any information that has already been published. If you include already published figures or images, please get the necessary permission from the copyright holder to publish under the CC-BY license;
Plagiarism, data fabrication and image manipulation are not tolerated;
Plagiarism is not acceptable in OAE journals.

Plagiarism involves the inclusion of large sections of unaltered or minimally altered text from an existing source without appropriate and unambiguous attribution, and/or an attempt to misattribute original authorship regarding ideas or results, and copying text, images, or data from another source, even from your own publications, without giving credit to the source.

As to reusing the text that is copied from another source, it must be between quotation marks and the source must be cited. If a study's design or the manuscript's structure or language has been inspired by previous studies, these studies must be cited explicitly.

If plagiarism is detected during the peer-review process, the manuscript may be rejected. If plagiarism is detected after publication, we may publish a Correction or retract the paper.

Falsification is manipulating research materials, equipment, or processes, or changing or omitting data or results so that the findings are not accurately represented in the research record.

Image files must not be manipulated or adjusted in any way that could lead to misinterpretation of the information provided by the original image.

Irregular manipulation includes: introduction, enhancement, moving, or removing features from the original image; the grouping of images that should be presented separately, or modifying the contrast, brightness, or color balance to obscure, eliminate, or enhance some information.

If irregular image manipulation is identified and confirmed during the peer-review process, we may reject the manuscript. If irregular image manipulation is identified and confirmed after publication, we may publish a Correction or retract the paper.

OAE reserves the right to contact the authors' institution(s) to investigate possible publication misconduct if the Editors find conclusive evidence of misconduct before or after publication. OAE has a partnership with iThenticate, which is the most trusted similarity checker. It is used to analyze received manuscripts to avoid plagiarism to the greatest extent possible. When plagiarism becomes evident after publication, we will retract the original publication or require modifications, depending on the degree of plagiarism, context within the published article, and its impact on the overall integrity of the published study. Journal Editors will act under the relevant COPE guidelines.

4. Authorship

Authorship credit of *IR* should be solely based on substantial contributions to a published study, as specified in the following four criteria:
1. Substantial contributions to the conception or design of the work, or the acquisition, analysis, or interpretation of data for the work;
2. Drafting the work or revising it critically for important intellectual content;
3. Final approval of the version to be published;
4. Agreement to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

All those who meet these criteria should be identified as authors. Authors must specify their contributions in the section Authors' Contributions of their manuscripts. Contributors who do not meet all the four criteria (like only involved in acquisition of funding, general supervision of a research group, general administrative support, writing assistance, technical editing, language editing, proofreading, *etc*.) should be acknowledged in the section of Acknowledgement in the manuscript rather than being listed as authors.

If a large multiple-author group has conducted the work, the group ideally should decide who will be authors before the work starts and confirm authors before submission. All authors of the group named as authors must meet all the four criteria for authorship.
AI and AI-assisted technologies should not be listed as an author or co-author.

5. Reviewers Exclusions

You are welcome to exclude a limited number of researchers as potential Editors or reviewers of your manuscript. To ensure a fair and rigorous peer review process, we ask that you keep your exclusions to a maximum of three people. If you wish to exclude additional referees, please explain or justify your concerns—this information will be helpful for Editors when deciding whether to honor your request.

6. Editors and Journal Staff as Authors

Editorial independence is extremely important and OAE does not interfere with Editorial decisions. Editorial staff or Editors shall not be involved in processing their own academic work. Submissions authored by Editorial staff/Editors will be assigned to at least two independent outside reviewers. Decisions will be made by the Editor-in-Chief, including Special Issue papers. Journal staff are not involved in the processing of their own work submitted to any OAE journals.

7. Policy of the Use of AI and AI-assisted Technologies in Scientific Writing

Generative AI and AI-assisted technologies (e.g., large language models) are expected to be increasingly used to create content. In the writing process of manuscripts, using AI and AI-assisted technologies to complete key researcher work, such as producing scientific insights, analyzing and interpreting data or drawing scientific conclusions, is not allowed, and they should only be used to improve the readability and language of manuscripts.
AI and AI-assisted technologies should be used under human control and supervision as they may generate incorrect or prejudiced output, and they should not be listed as an author or co-author, nor cited as an author.
The use of AI and AI-assisted technologies should be disclosed by authors in their manuscripts, and a statement will be required in the final publication.
OAE will keep monitoring the development and adjust the policy when necessary.

8. Conflict of Interests

OAE journals require authors to declare any possible financial and/or non-financial conflicts of interest at the end of their manuscript and in the cover letter, as well as confirm this point when submitting their manuscript in the submission system. If no conflicts of interest exist, authors need to state “All authors declared that there are no conflicts of interest”. We also recognize that some authors may be bound by confidentiality agreements, in which cases authors need to state “All authors declared that they are bound by confidentiality agreements that prevent them from disclosing their competing interests in this work”.

9. Editorial Process

9.1. Pre-Check

New submissions are initially checked by the Managing Editor from the perspectives of originality, suitability, structure and formatting, conflicts of interest, background of authors, *etc*. Poorly prepared manuscripts may be rejected at this stage. If your manuscript does not meet one or more of these requirements, we will return it for further revisions.

Once your manuscript has passed the initial check, it will be assigned to the Assistant Editor, and then the Editor-in-Chief, or an Associate Editor in the case of a conflict of interest, will be notified of the submission and invited to review. Regarding Special Issue paper, after passing the initial check, the manuscript will be successively assigned to the Assistant Editor, and then to the Editor-in-Chief, or an Associate Editor in the case of conflict of interest for the Editor-in-Chief to review. The Editor-in-Chief, or the Associate Editor may reject manuscripts that they deem highly unlikely to pass peer review without further consultation. Once your manuscript has passed the Editorial assessment, the Associate Editor will start to organize peer-review.

All manuscripts submitted to *IR* are screened using CrossCheck powered by iThenticate to identify any plagiarized content. Your study must also meet all ethical requirements as outlined in our Editorial Policies. If the manuscript does not pass any of these checks, we may return it to you for further revisions or decline to consider your study for publication.

9.2. Peer Review

IR operates a single-blind review process, which means that reviewers know the names of authors, but the names of the reviewers are hidden from the authors. The scientific quality of the research described in the manuscript is assessed by a minimum of two independent expert reviewers. The Editor-in-Chief is responsible for the final decision regarding acceptance or rejection of the manuscript.

All information contained in your manuscript and acquired during the review process will be held in the strictest confidence.

9.3. Decisions

Your research will be judged on scientific soundness only, not on its perceived impact as judged by Editors or referees. There are three possible decisions: Accept (your study satisfies all publication criteria), Invitation to Revise (more work is required to satisfy all criteria), and Reject (your study fails to satisfy key criteria and it is highly unlikely that further work can address its shortcomings). All of the following publication criteria must be fulfilled to enable your manuscript to be accepted for publication:

Originality
The study reports original research and conclusions.
Data availability
All data to support the conclusions either have been provided or are otherwise publicly available.
Statistics
All data have been analyzed through appropriate statistical tests and these are clearly defined.
Methods
The methods are described in sufficient detail to be replicated.
Citations
Previous work has been appropriately acknowledged.
Interpretation
The conclusions are a reasonable extension of the results.
Ethics
The study design, data presentation, and writing style comply with our Editorial Policies.

9.4. Revisions

Authors are required to submit the revised manuscript within one week if minor revision is recommended while two weeks if major revision recommended or one month if additional experiments are needed. If authors need more than one point-to-point response to all comments of reviewers and the Editor-in-Chief or the Associate Editor should be supplied along with the revised manuscript to allow quick assessment of your revised manuscript. This document should outline in detail how each of the comments was addressed in the revised manuscript or should provide a rebuttal to the criticism. Manuscripts may or may not be sent to reviewers after revision, dependent on whether the reviewer requested to see the revised version. Apart from in exceptional circumstances, *IR* only supports a round of major revision per manuscript.

10. Contact Us

Journal Contact

Intelligence & Robotics Editorial Office
Suite 1504, Plaza A, Xi'an National Digital Publishing Base,
No. 996 Tiangu 7th Road, Gaoxin District, Xi'an 710077, Shaanxi, China.

Managing Editor

Wen Zhang
editorial@intellrobot.com

Last updated on 6 June, 2023