西安交通大学

本科毕业设计（论文）

基于自适应动态规划的互联系统安全保障控制研究

学院（部、中心）：　　电气工程学院

专　　业：　　电气工程及其自动化

班　　级：　　电气 910

学生姓名：　　谭浚楷

学　　号：　　2194313163

指导教师：　　曹晖

2023 年 06 月

摘 要

随着工业系统规模的增大以及人工智能技术特别是机器学习的发展，大规模复杂系统的控制方法在多个领域得到应用。作为复杂系统中的基础问题，互联系统稳定控制以及智能体博弈纳什均衡往往具有很强的耦合性以及非线性，也一直是多智能体领域的研究热点。大规模系统通常是重点安全控制系统，需要通过一定的优化约束条件来满足系统的安全运行要求。为了保障系统运行的安全稳定，可以在系统的控制层面加入安全保障机制。安全保障控制器就是一种在原有的系统层级控制策略架构上拓展的决策机制，通过对系统危险倾向进行反向补偿，实现整体状态的安全稳定运行，因此安全保障控制器的研究对重点安全的大规模系统稳定安全具有很高的研究价值。

本文主要设计了一种针对包括互联系统、非零和博弈在内的复杂系统的安全保障控制器，不仅能够满足凸集合的安全边界约束要求，还可以满足传统安全控制器无法满足的非凸集合安全约束要求。同时安全保障控制补偿项可以和任意名义控制器结合，灵活适用各种系统，对系统参数的变化具有良好的抗干扰性，对安全边界的约束具有一定的自适应性。本文分别针对复杂系统中的非零和博弈系统以及互联系统的特性进行安全保障控制器设计。这种保障安全的控制器设计主要分为三个部分。首先设计一种基于障碍函数梯度的安全保障的控制器补偿项。其次根据自适应动态规划理论建立名义控制器，非零和博弈系统中需要满足多个智能体非零和博弈下的纳什均衡，而互联系统中则需要利用分布式稳定性理论设计分布式稳定控制器。最后利用并行学习的更新律对控制器逼近的神经网络进行迭代，借助单一评判网络方法，在得到适用性广泛的安全保障控制器的同时，减少了神经网络逼近参数迭代过程中一半的计算量。

在此基础上，本文借助李雅普诺夫稳定性理论分别证明了非零和博弈与互联系统在安全保障控制器作用下的稳定性，并且证明得到并行学习更新律的参数最终一致有界。本文通过数值仿真实验对互联系统以及非零和博弈的理论分析结果进行验证，表明在所设计的基于障碍函数的安全保障控制器的作用下，复杂系统能够在非凸性以及凸性安全约束下保障状态变量的安全。

为了进一步验证基于障碍函数的安全保障策略在实际应用场景下的效果，本文设计了无人机的硬件模拟实验对安全保障策略进行验证，表明在所设计的安全策略作用下，无人机复杂系统能够在避开固定障碍物以及无人机互相避让控制下保障安全。

关 键 词：自适应控制；安全强化学习；障碍函数；互联系统；非零和博弈

ABSTRACT

With the increasing scale of industrial systems and the development of artificial intelligence technology, especially machine learning, control methods for large-scale complex systems have been applied in several fields. As a fundamental problem in complex systems, interconnected control as well as multi-agent Nash equilibrium are often highly coupled as well as nonlinear. Large-scale systems are usually focused on safety control systems, which require certain optimization constraints to meet the safe operation requirements of the system. In order to guarantee the safe and stable operation of the system, a safety assurance mechanism can be added to the control level of the system. The safety assurance controller is a kind of decision mechanism that comes from upgrading on the original system level control strategy architecture to achieve safe and stable operation by the compensation term, so the research of the safety assurance controller has high research value for the safety of large-scale system.

In this paper, a safety assurance controller is designed for complex systems including interconnected systems, non-zero-sum games, etc. It meets the safety boundary constraint requirements of convex and non-convex sets. The safety assurance control compensation term is combined with nominal controller to apply to various systems, with good anti-interference to the changes of system parameters and certain adaptiveness to the constraints of the safety boundary. In this paper, the safety-securing controller design is carried out for non-zero-sum game systems and interconnected systems, respectively. Firstly, a controller compensation term is designed based on the gradient of the barrier function for security assurance. Secondly, a nominal controller is established, which is required to satisfy the Nash equilibrium under non-zero-sum game, while a distributed stability controller is required to be designed using distributed stability theory in interconnected system. Finally, using the update law of concurrent learning to iterate the neural network for controller approximation, with the help of single network method, the computational effort is reduced by half.

This paper demonstrates the stability of non-zero-sum games and interconnected systems under the security controllers with the Lyapunov stability theory, and proves that the parameters of the concurrent learning are ultimately bounded. The theoretical analysis results are verified by numerical simulation experiments, which show that the complex system can secure the state variables under the nonconvex and convex security constraints under the action of the designed safety assurance controller based on the barrier function.

To further verify the effectiveness of the safety assurance strategy in practical application scenarios, this paper designs a UAV hardware simulation experiment, showing that under the effect of the designed safety strategy, the UAV complex system can ensure safety avoiding fixed obstacles and mutual avoidance of UAVs.

KEY WORDS: Adaptive control; safe reinforcement learning; barrier functions; interconnected systems; non-zero-sum games

目 录

1 绪论

1.1 研究背景及意义

当今的社会活动以及工业生产中广泛存在着复杂系统。例如强复杂性的交通枢纽、强关联性的电力系统、通信网络、工业制造和航天航空等[1–3]。由于这些系统中通常包含多个控制器和传感器，因而系统具有高度的非线性和耦合性。进入本世纪以来，具有广泛实际应用场景的复杂互联系统受到工业界、学术界的极大关注，其高度耦合性与非线性对控制策略的建立提出了极高的要求。原有的集中式控制策略难以进一步拓展到具有更多互联项的复杂系统之中，因此分布式控制策略得以发展，即对复杂系统中的每一个子系统设计独立的控制策略，各个子系统的决策相互独立，这种控制架构极大的节省了整体通讯资源，并且提高了复杂系统的鲁棒性。相比传统的集中式控制策略，针对非线性互联系统，分布式控制策略的控制器设计更加方便有效。

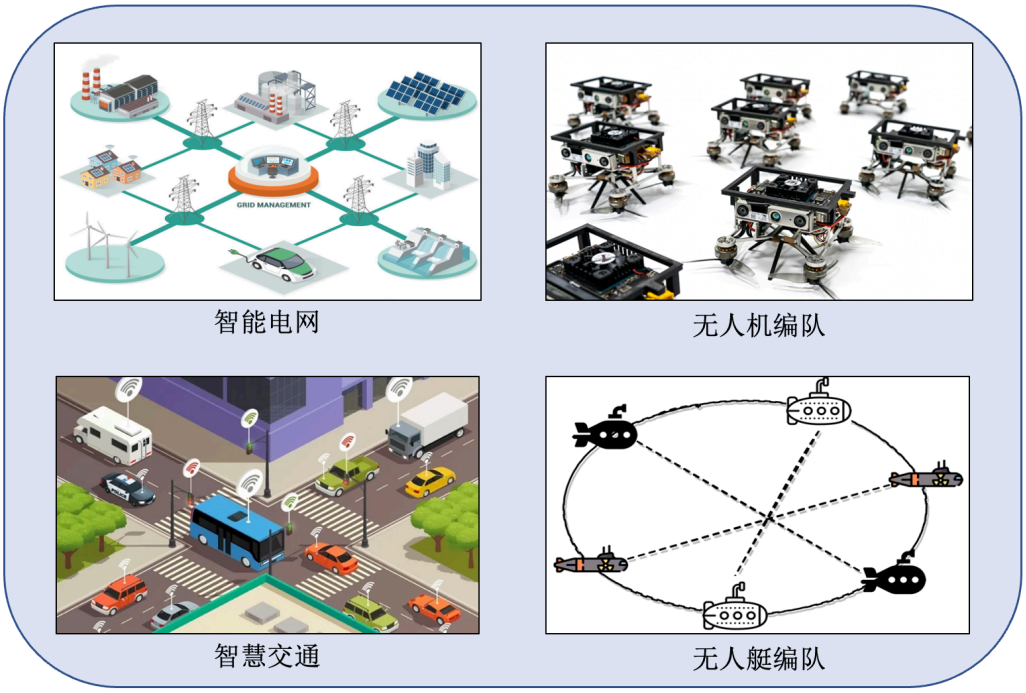

图 1-1 互联系统应用场景示意

多智能体系统控制是近年来控制领域的热门问题[4–6]，其分布式协作的控制机制在智能电网、无人机编队、无人艇编队以及智慧交通等实际场景被广泛应用，如图 1-1 所示。单个智能体的学习称为强化学习，控制领域又称之为自适应动态规划。通过智能体与环境的交互，智能体不断学习如何在当前环境下做出相对最优的决策。但是复杂系统下往往存在大量的智能体，智能体之间的利益存在一定的强耦合性与复杂性，对多智能体的分布式稳定架构提出了挑战，特别是存在一定危险的重点安全系统，需要严格保障系统的状态安全的前提下进行稳定性控制。故复杂多智能体系统的安全性要求

对分布式控制器提出了很大的挑战，研究安全保障控制机制对于复杂系统在各方面的应用具有极大的价值。

1.2 基于自适应动态规划的互联系统抗干扰分析研究现状

1.2.1 互联系统与多智能体博弈

随着现代工业生产规模的增大，以电网、交通网络和互联系统[7–9]为主要代表的众多网络系统受到工业界和学术界的广泛关注，其中互联系统是由多个互相耦合的复杂子系统交互组成，如图 1-2 所示，相比于传统控制中针对的单一系统的研究更具挑战性。互联系统相关控制设计中的关键是针对子系统之间的互联项进行稳定性设计。现代控制理论提出，互联系统的稳定性控制有两种主要的研究途径，一是集中式控制，二是分布式控制。其中集中式控制利用中心点收集系统整体的参数，统一处理并由其产生控制信号，能够高效处理数据，但是其具有成本高、通信难的显著缺点。而分布式控制器通过构建多个局部的控制器，分而治之从而实现互联系统的总体控制，相比集中式控制具有成本更低、抗干扰能力强等特点。因此分布式控制成为了工业领域、控制学界的一个研究热点。

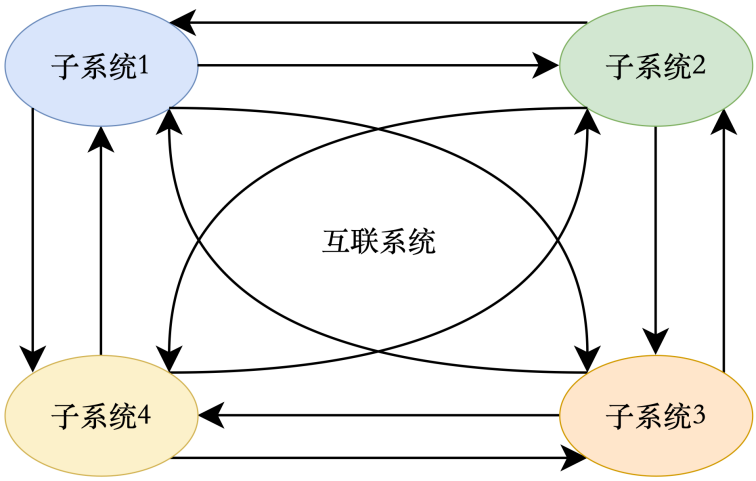

图 1-2 互联系统结构示意图

文献[10]中提出，互连的虚拟-现实系统已经成为一个重要的研究领域。这种由虚拟元素和现实世界组件组成的互连子系统提供了一种独特的组合，使系统的运行更加灵活。文献[11]使用了在线学习最优控制方法实现了连续时间非线性互连系统的稳定控制。文献[12]将非线性互联系统的分散跟踪问题被转化为增强子系统的最优调节问题，并针对其设计稳定控制器，对连续事件采样从而反馈信息。文献[13]中提出了一类具有强互连性的非线性互连系统的近似最优分布式控制方案。

由于复杂工业系统中通常包含多个控制器，故互联系统中的每个控制器可以等同于一个智能体，多个智能体的控制系统可以称为多智能体博弈系统。每个智能体会权衡在合作与冲突中的利益，做出利益最大化的选择。根据具体的利益资源分配，博弈系统可以分为零和博弈问题与非零和博弈问题。其中零和博弈的资源总数有限，多个智

能体需要在有限的资源下将自己的利益最大化，所有智能体之间为竞争关系。而非零和博弈的资源不受限制，又可以具体分为完全合作博弈与部分合作博弈。无论是零和博弈还是非零和博弈，系统控制的最终目的是找到一个最终的稳定状态，在经济学上，这个平衡点被称为纳什均衡。为了寻求这些均衡点，通常需要求解复杂的汉密尔顿-雅各比-伊萨克斯（Hamilton-Jacobi-Issacs，HJI）方程或者汉密尔顿-雅各比（Hamilton-Jacobi，HJ）方程，但是由于这些方程具有的强耦合性以及高度的非线性，传统的动态规划方法难以获得精确解。因此，利用自适应动态规划方法研究复杂系统下的多智能体博弈具有一定的理论价值与工业实际意义。

包含多个控制器的复杂系统同样可以看成是一个博弈问题。在多个智能体博弈中，多个控制器相互作用，追求纳什均衡，同时确保控制权保持在安全限度内。文献[14]提出了一个基于障碍函数的坐标系转换机制，以保障不对称边界下的状体变量安全。文献[15]研究了一种在线的“执行-评判”算法，包含多个独立价值函数，以及多个智能体的控制器。文献[16]提出了一种新颖的鲁棒深度神经网络（DNN），该网络具有控制执行器和近似值函数的辨识器。文献[17]的研究提出了一种在线博弈学习的 ACI 架构，该架构放宽了传统方法对于持续激励信号要求。文献[18]研究了一种在线策略迭代方法，实现同时评估非线性零和博弈中的两个智能体的策略和价值。文献[19]针对离散时间系统提出了一种离线迭代策略的方法，在完全合作博弈中实现状态和输入同时受限下的控制。文献[20]提出了一种使用单一评判网络的最优控制方法，该方法实现仅仅利用一个神经网络逼近价值、控制和干扰。

1.2.2 动态规划与自适应动态规划

动态规划（Dynamic Programming，DP）是由贝尔曼于 1957 年提出的一种求解非线性系统最优策略的现代控制方法[21]。其核心是贝尔曼方程[22]，通过求解贝尔曼最优性方程，高维度的优化问题被转化为多个低维的优化问题，最优策略应当根据时间尺度首先从一个时刻向未来时刻进行前推，再反向递推得到，如图 1-3 和图 1-4 所示。

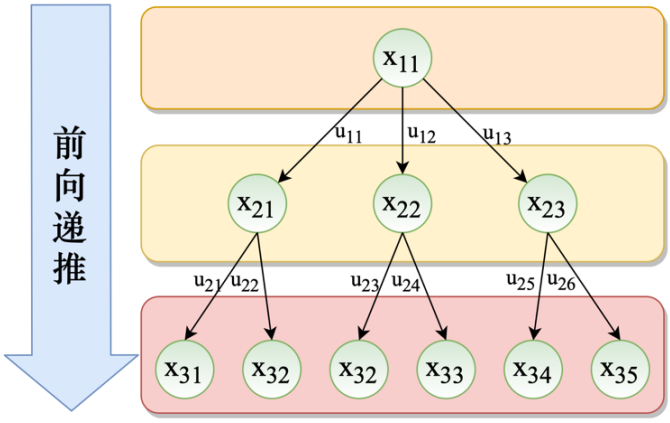

图 1-3 动态规划中的前向递推

但是随着复杂系统的发展，控制器、传感器的数量呈爆炸式增长，系统的非线性程

度进一步提高，控制器传感器之间的耦合性也进一步深化，整体系统功能呈现高度的复杂性。此时动态规划对于计算资源的需求也极大的增大，精度受到严重影响。针对如今具有大量实时数据的复杂系统，动态规划需要融合新的技术降低其对求解计算资源的需求。

动态规划最初针对的是采样得到的离散系统，后续推广到连续系统中，由线性系统推广到非线性系统，其系统状态、控制策略的维度不断增加，其中的汉密尔顿-雅可比-贝尔曼(Hamilton-Jacobi-Bellman，HJB)方程的求解计算量急剧增大，计算精度受到严重影响，学术界称其为“维数灾难”。为了解决维数灾难问题，韦伯斯提出利用神经网络等工具近似逼近动态规划中的值函数，由于利用了神经网络的自适应性，故称此方法为自适应动态规划。

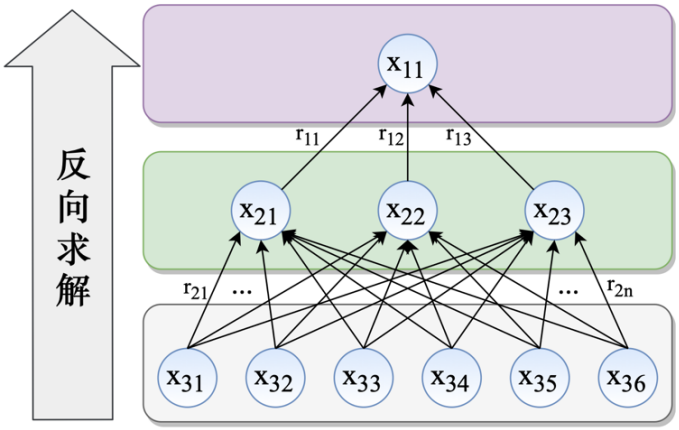

图 1-4 动态规划中的反向求解

自适应动态规划（Adaptive Dynamic Programming，ADP）是对动态规划方法的改进，结合了强化学习的“奖励-学习”机制，借助神经网络的强大拟合逼近能力，实现未知先验知识下的决策求解，同时有效避免“维数灾难”。自适应动态规划的核心是收集在线亦或离线数据，采用神经网络对优化决策过程中的复杂价值函数进行逼近，并将逼近的价值函数用于求解最优性决策。其最主要的特性有两点：一是模型未知性，自适应动态规划可以在无先验模型的前提下，利用大量“智能体-环境”交互数据训练智能体的决策生成机制；二是逼近性，针对高维度、非线性和高耦合性的复杂系统，自适应动态规划借助神经网络的强大拟合性能，能有效求解决策问题，节约决策生成所需要的计算资源。

1.2.3 强化学习与安全强化学习

强化学习（Reinforcement Learning，RL）是一种基于环境-智能体交互的机器学习方法。强化学习的核心思想是模拟高等生物大脑的“奖励-学习”机制，建立一个可以根据外界反馈不断迭代策略的智能体，智能体与外界环境进行探索式互动获取一定的奖励反馈，对其自身的控制策略生成机制进行更新。由于强化学习可以解决环境未知、无模型的非先验决策优化问题，强化学习被广泛用于求解具有高度非线性项、复杂的

未知模型的最优决策问题。下图描述了智能体与动态环境之间的交互过程，智能体根据自身的动作控制机制生成决策，以实际物理形态作用于动态系统。动态系统受到智能体的控制决策后，发生一定变化并反馈奖励或者惩罚值，反馈给智能体。智能体的类脑模型，即评判机制对惩罚值进行一定处理，生成对整体价值形势的估计，即价值 V。每次智能体与环境交互后，智能体根据自身判断估计的价值 V 更新自己的控制决策机制，迭代形成新的控制策略，具体交互过程如图 1-5 所示。随着人工智能领域的不断扩充，人工神经网络技术得到进一步发展，通过利用神经网络的逼近性质，强化学习的研究也得以从离散系统拓展到连续空间。人工神经网络具有高度的容错性、自适应性和拟合性，可以利用大量有效数据进行迭代学习，从而实现系统辨识、数值函数拟合等，广泛应用于优化、决策生成领域。

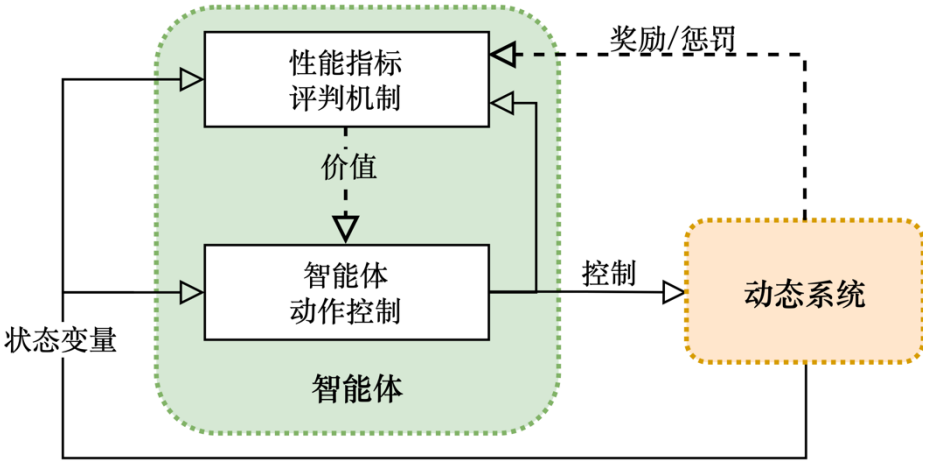

图 1-5 智能体与环境的交互

现有的决策生成方法中，大多数控制器都是针对系统的稳定或者追踪任务进行设计的，其性能指标仅仅考虑所需实现的单一功能。然而从系统安全以及整体稳定的角度考虑，这样绝对理想化的方法势必会对系统整体的控制性能造成一定影响，同时会大量增加系统用于稳定的资源。

障碍函数是一种优化决策方法，将优化过程中存在的一定的约束条件，转化为非连续/连续的障碍函数，以此将优化过程的可行域限制在约束条件下，同时利用障碍函数与目标函数的结合惩罚不满足约束条件的解。这样原本的优化问题就可以在满足一定的约束条件的前提下，求解得到保障安全的控制决策。为了保障系统的整体安全性，学者们提出了基于障碍函数的安全保障机制，包括基于障碍函数的坐标系转化[23]、基于障碍函数的奖励机制[24]、基于类 Lyapunov 障碍函数的补偿控制器[25]等。这些基于障碍函数的安全保障方法能有效处理凸性/非凸性约束问题，具有很强的普适性，在重点安全系统以及高精度控制系统中有着广泛的应用。文献[26]提出了一种在离散时间系统下，利用状态约束来系统安全的 RL 方法。文献[27]中提出了一种在传感器和执行器攻击下的安全控制方法。文献[28]通过对 Bellman 误差进行采样，并使用稀疏神经网络进行训练，改进了文章[14]的仿真结果，减少了相应的计算压力。

在复杂系统的多个智能体的博弈中，激励信号的风险往往被忽视。对于多智能体

博弈系统，并行学习是解决持续激励问题的一种常见方法。文献[29]提出了一种有效利用历史和即时数据学习得到控制器的方法。文献[30]提出了一种非严格要求激励条件的在线学习最优控制方法。文献[31]结合了 RL 和经验回放方法，该研究在相比文献[32]取得了更好的控制性能的同时，避免了激励暂态可能导致的系统不稳定。文献[33]设计了一种基于模型的方法来提高控制决策的计算效率。文献[34]提出一种基于 DNN 和并行学习的方法，可以有效地求解最优跟踪问题。

互联系统通常具有大量非线性项，因而通过基于与环境的互动的强化学习获取的最优策略往往具有良好的模型适应性。然而，传统的基于强化学习的方法侧重于性能优化，没有明确考虑安全约束[35,36]，这可能会导致学习过程中系统的不安全状况[37]。安全保障的控制是互连系统的一个关键方面，因为无安全约束的故障或事故会导致严重的后果[38]。由于互联系统的子系统之间复杂的耦合性以及所需满足的各种安全约束，确保这些系统的安全运行具有挑战性[39]。文献[25]研究了一种新型的控制屏障功能，通过这种新型的障碍屏障方法运用于系统，增强了基于学习的控制策略的安全性，以确保安全。文献[40]提出了引入障碍惩罚/奖励的安全控制器。文献[41]提出了一个在安全保障下，适应未知复杂环境追踪博弈。因此，在以安全为首要前提的互联系统控制中，控制障碍函数的强约束性保障系统的硬性状态安全。文献[42]提出了一种基于卷积残差神经网络（Residual Network，Resnet）的自适应控制策略，并且针对每一层参数设计了基于李雅普诺夫函数的自适应更新律。相比于以往经常使用的一阶更新律，文献[43]提出了一种针对高阶系统的自适应控制方法，通过提出一个基于加速梯度下降的更新律，迭代得到自适应控制策略，可以在保障系统具有渐进稳定的能力的同时，实现高效地追踪控制。文献[44]提出了一种实时的深度神经网络自适应控制律。通过基于李雅普诺夫函数的自适应更新律实时更新外层输出参数，通故数据驱动的监督学习方法实现离线逼近内层参数。总的来说，自适应动态规划、强化学习等控制策略的安全性有待考量，有安全保障的控制决策具有一定的研究价值。

1.3 本论文主要内容及结构安排

根据上文对互联系统的自适应动态规划的研究现状的分析，为了保障复杂互联系统的状态安全，通常直接将控制器设计为能够满足一定安全特性的安全控制器，但是这种集成式控制器对于系统随机变化没有良好的自适应性，随着系统部分参数的变化与扰动的加入，控制器往往需要重新设计，不能在真实环境中达到保障安全的目的。其次，传统的安全控制器对安全的定义较为单一，通常将安全定义为不违反特定边界的约束，这些边界约束一般为凸性集合的边界，但实际的系统状态约束几乎不可能为凸集合。在重点安全的互联系统中，安全保障控制器的设计将对系统的性能有极大的影响。

基于以上分析，本论文将设计一种针对包括互联系统、非零和博弈系统的安全保障控制器，能够同时实现凸集合和非凸集合的安全边界约束要求。本章的主要内容结构图如下图 1-6 所示：

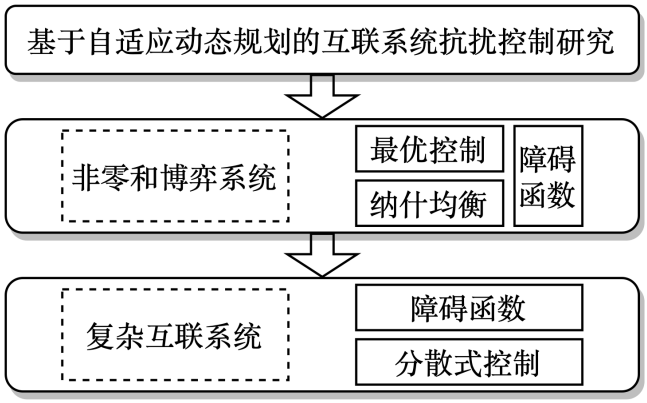

图 1-6 本文研究的内容结构图

本文的结构安排如下：

第一章，绪论。本章主要介绍了基于自适应动态规划互联系统的安全抗干扰控制，简要总结了此领域的研究现状。在此基础上，分析了现有安全强化学习的优缺点以及研究进展。

第二章，预备知识。本章主要内容为本文涉及的自适应动态规划算法，神经网络原理和障碍函数理论，给出分布式稳定理论，明确本文要解决的安全控制问题的基础。

第三章，非零和博弈下的保障安全控制。本章节主要针对同为复杂系统的非零和博弈系统进行控制器设计，为后续更复杂的互联系统安全保障控制器设计提供研究基础。首先提出基于障碍函数梯度的安全保障的控制器补偿项。根据 ADP 理论建立名义控制器，实现非零和博弈下的纳什均衡。最后基于并行学习设计逼近参数更新律。

第四章，基于保障安全控制器的互联系统安全稳定控制。本章主要针对更加复杂的互联系统进行控制器设计。首先提出基于障碍函数的安全补偿项。其次设计分布式稳定的名义项，可得到安全保障控制器。最后利用并行学习对逼近网络进行迭代。

第五章，数值仿真验证。本章通过数值仿真实验对互联系统以及非零和博弈的理论分析结果进行验证，表明在所设计的基于障碍函数的安全保障控制器的作用下，复杂系统能够在非凸性以及凸性安全约束下保障状态变量的安全。

第六章，硬件模拟仿真验证。本章通过无人机硬件模拟对基于障碍函数的安全保障控制进行验证，在安全策略作用下，无人机系统能够避开固定障碍物并相互避让。

第七章，结论与展望。本章对本论文的研究内容以及结果进行了总结，得出本研究的结论，并指出了今后的探索方向。

2 预备知识

本章的主要内容为文中所涉及的自适应动态规划的基础算法，神经网络原理和障碍函数相关理论，给出了分布式互联系统的稳定性理论，明确本文所要解决的互联系统稳定安全控制问题的基础。

2.1 自适应动态规划

自适应动态规划（ADP）是一种解决最优控制问题的机器学习技术。这种控制理论与强化学习结合的方法使用神经网络逼近最优控制策略。基于动态规划的概念，利用数学优化方法，将复杂的问题分解成更小、更容易解决的子问题。

在具有大量不确定性因素和复杂项的系统中，比如机器人、金融和航空航天领域中的系统，都可以利用 ADP 进行求解。因此该方法是解决现实世界问题的有力工具，其具有从环境中学习并适应不断变化的环境的能力。

为了体现研究的一般性，针对具有一般性一般离散时间非线性系统：

$$x_{k+1} = F(x_k, u_k) \tag{2-1}$$

根据最优控制理论，我们设置如下形式的性能指标函数：

$$J(x_k) = \sum_{i=k}^{\infty} U(x_i, u_i) \tag{2-2}$$

其中U为效用函数，代表每个离散时刻环境或者系统反馈给智能体的奖励值或惩罚。

为了获取最优控制器，根据庞特里亚金极小值原理，可知最优性能指标是最小值，利用动态规划的形式写成如下形式：

$$J^*[x(t), t] = \min_{u(t)}\{U[x(t), u(t), t] + \gamma J^*[x(t+1), t+1]\} \tag{2-3}$$

故最优控制策略即为性能指标取得最小值时的控制u：

$$u^*(t) = \operatorname{argmin}_{u(t)}\{U[x(t), u(t), t] + \gamma J^*[x(t+1), t+1]\} \tag{2-4}$$

但是根据韦伯斯教授提出的维数灾难理论，一般的动态规划利用的前向求解，反向计算方法，会出现计算维度随着变量增大而呈现指数型增长。为了解决维数灾难问题，韦伯斯提出，利用迭代更新的方式获取控制策略，以便减少计算所需要的资源，同时保障控制策略逼近最优策略。

接下来介绍自适应动态规划的典型算法之一——广义值迭代的算法步骤。

基于值函数的迭代逼近是强化学习解决“维数灾难”的最基础的方法。值函数的逼近一般使用非线性逼近方法，由于神经网络具有良好的逼近性能，通常使用当个神经网络对值函数进行近似。就具体步骤而言，基于值函数的迭代算法分为四个步骤：第一步是初始化一个半正定的价值函数$V_0(x_k) \equiv \Psi(x_k)$。接着获取对应和这个价值函数的控制策略：

$$v_0(x_k) = \operatorname{argmin}_{u_k}\{U(x_k, u_k) + V_0(x_{k+1})\} \tag{2-5}$$

由于执行策略后，价值函数会发生变化，故需重新计算性能指标：

$$V_1(x_k) = U\big(x_k, v_0(x_k)\big) + V_0\big(F\big(x_k, v_0(x_k)\big)\big) \tag{2-6}$$

重复已上步骤，依次获得控制策略v：

$$v_i(x_k) = \text{argmin}_{u_k}\{U(x_k, u_k) + V_i\big(F(x_k, u_k)\big)\} \tag{2-7}$$

以及通过不断迭代获得性能指标V：

$$V_{i+1}(x_k) = \min_{u_k}\{U(x_k, u_k) + V_i\big(F(x_k, u_k)\big)\} \tag{2-8}$$

具体的广义值迭代算法步骤如表 1.1 所示。需要注意的是，表 1.1 的第四步为判定步骤，设置阈值ε，该步骤通过比较$|V_{i+1}(x_k) - V_i(x_k)|$的大小是否小于阈值判断迭代过程是否收敛，决定是否终止迭代算法。

表 2-1 广义值迭代算法

步骤	内容
Step1:	**初始化(Initialization)** 设半正定函数Ψ作为初始值：$V_0(x_k) \equiv \Psi(x_k)$
Step2:	**策略评估(Policy evaluation)** 获得相应步骤的迭代控制律：$v_i(x_k) = \text{argmin}_{u_k}\{U(x_k, u_k) + V_i\big(F(x_k, u_k)\big)\}$
Step3:	**策略改进(Policy improvement)** 更新对应的计算性能指标：$V_{i+1}(x_k) = \min_{u_k}\{U(x_k, u_k) + V_i\big(F(x_k, u_k)\big)\}$
Step4:	重复策略评估与策略改进，直到$\|V_{i+1}(x_k) - V_i(x_k)\| < \varepsilon$，否则返回 Step2，继续在控制$v$和性能指标$V$之间不断迭代。

具体的算法结构图如图 2-1 所示，算法在执行每一步迭代的过程中，首先，由当前控制网络生成控制策略$u(k)$，作用于动态模型，获得下一个时刻$(k+1)$的状态变量$x(k+1)$，利用该状态变量值输入评判网络，即可获取价值函数的逼近值$J(k+1)$，接着，再利用价值函数$J(k+1)$与上一时刻的存储价值函数$J(k)$的差值来更新评判网络，最后更新控制策略的生成网络。

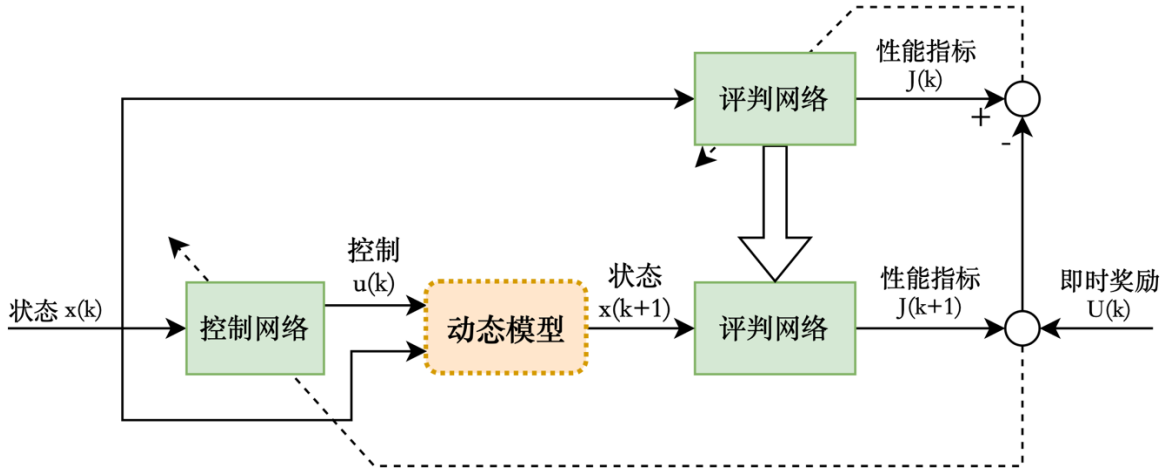

图 2-1 广义值迭代算法结构图

接下来介绍自适应动态规划的典型算法之一——广义策略迭代算法的步骤。

随着对基于值函数的迭代逼近算法的深入研究，学术界发现值函数的逼近方法在解决一些复杂系统的优化决策问题存在许多的困难，因此另一种基于策略迭代逼近的动态规划算法开始出现。

策略迭代算法主要包括策略改进和策略评估两个关键步骤，其基本流程是：首先初始设置$v_0(x_k)$为一个容许的控制律，接着设置一个相应的初始效用函数：

$$V_0(x_k) = U\big(x_k, v_0(x_k)\big) + V_0\big(F\big(x_k, v_0(x_k)\big) \tag{2-9}$$

通过更新获得下一步的迭代控制策略：

$$v_1(x_k) = \text{argmin}_{u_k \in \mathbb{R}^m}\{U(x_k, u_k) + V_0(x_{k+1})\} \tag{2-10}$$

利用已得到的控制策略计算本次的性能指标$V_i(x_k)$：

$$V_i(x_k) = U\big(x_k, v_i(x_k)\big) + V_i\big(F\big(x_k, v_i(x_k)\big) \tag{2-11}$$

以及下一步的控制律$v_{i+1}(x_k)$：

$$v_{i+1}(x_k) = \text{argmin}_{u_k \in \mathbb{R}^m}\big\{U(x_k, u_k) + V_i\big(F(x_k, u_k)\big)\big\} \tag{2-12}$$

具体的广义策略迭代算法步骤如表 1.2 所示。需要注意的是，表 1.2 的第四步为判定步骤，设置阈值ε，该步骤通过比较$|V_{i+1}(x_k) - V_i(x_k)|$的大小是否小于阈值判断迭代过程是否收敛，决定是否终止迭代算法。

表 2-2 广义策略迭代算法

步骤	内容
Step1:	**初始化(Initialization)** 设半正定函数Ψ作为初始值：$V_0(x_k) \equiv \Psi(x_k)$
Step2:	**策略改进(Policy improvement)** 获得相应步骤的计算性能指标：$V_i(x_k) = \min_{u_k}\big\{U(x_k, u_k) + V_i\big(F(x_k, u_k)\big)\big\}$
Step3:	**策略评估(Policy evaluation)** 更新对应的迭代控制律：$v_{i+1}(x_k) = \text{argmin}_{u_k}\big\{U(x_k, u_k) + V_i\big(F(x_k, u_k)\big)\big\}$
Step4:	重复策略改进与策略评估，直到$\|V_{i+1}(x_k) - V_i(x_k)\| < \varepsilon$，否则返回 Step2，继续在控制$v$和性能指标$V$之间不断迭代。

具体的算法结构图如图 2-2 所示，算法在执行每一步迭代的过程中，首先将控制策略$u(k)$作用于动态模型，获得下一个时刻$(k+1)$的状态变量$x(k+1)$，接着更新控制网络，即新的控制策略的逼近值$u(k)$，再利用更新完成的控制策略来更新价值函数的生成网络，将该控制策略的逼近值$u(k)$输入评判网络，即可获取价值函数的逼近值$J(k+1)$，接着，再利用价值函数$J(k+1)$与上一时刻的存储价值函数$J(k)$的差值来更新评判网络。通过上述评判网络与控制网络的不断更新迭代，对应的策略评估与策略

改进的也在同时不断重复，最终两个用于逼近的神经网络收敛，几乎与理论上的价值函数一致，神经网络达到逼近的效果。

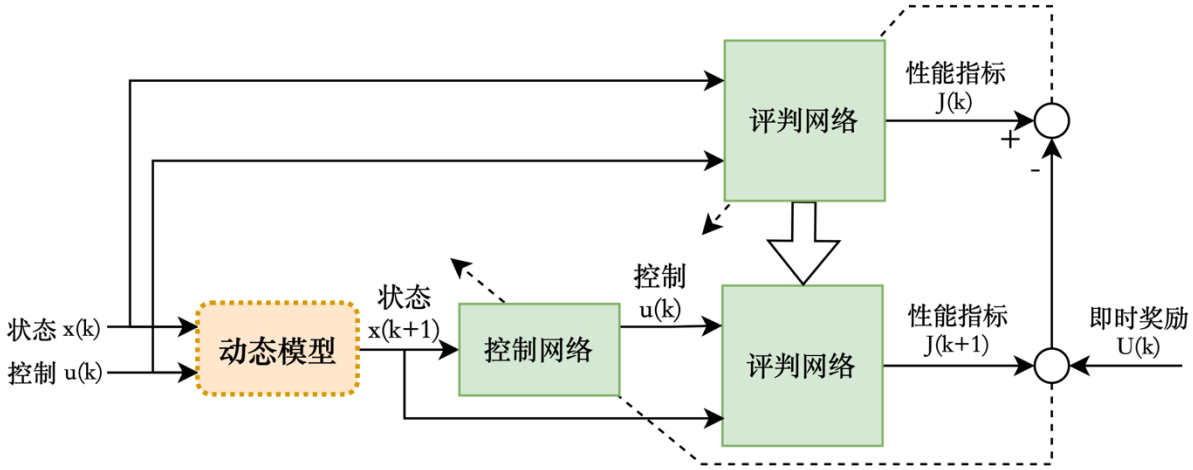

图 2-2 广义策略迭代算法结构图

2.2 神经网络逼近

经典的 ADP 算法需要使用到两个神经网络，即评判网络和执行策略网络，由于神经网络的逼近特性是 ADP 相对 DP 的主要不同点，故本小节将就神经网络的基本原理进行介绍。

神经网络是一种仿生计算模型，旨在模拟人脑神经元之间的相互作用。它的起源可以追溯到 1943 年数学家麦库洛克和皮茨提出的基础理论。当时，他们希望通过使用电子元件来模拟神经元之间的相互作用来解释神经元的工作原理。由于神经网络具有非常强大的映射逼近能力，故对于处理复杂系统中的非线性项和复杂耦合部分，神经网络为复杂网络控制设计提供了强有力的设计途径，其具有以下几种优点：

(1) 从理论角度出发，神经网络可以在神经元的数量不受限的前提下，实现对任意非线性函数的逼近，同时可以利用各种先进算法实现效率更高、精度更高的逼近效果。

(2) 由于神经网络具有大量可调参数，神经网络可以根据相应的约束自适应调节权重，具有很强的自适应能力。

(3) 鲁棒性随着神经网络的层数和神经元数量的不断增加而提升，具有较强的容错性和抗干扰能力。

典型的神经网络一般包括三层结构，分别为输入层、隐含层和输出层，具体原理图如图 2-3 所示。输入层的输入为整体网络的变量，在本文所使用的神经网络中，所有输入变量都是系统的状态变量及其变体。在经典的全连接网络中，输入层每一个变量将于下一层，也就是隐含层的所有神经元都有连接，输入层和隐含层之间一共有n^2个连接线。隐含层中有 m 个激活函数，将输入的 n 个变量映射到非线形空间中，增强整体网络的非线性拟合能力与理论极限逼近效果。最后乘以一定权重，从输出层输出神经网络的逼近结果，在本文中，输出的结果即为对非线性、强耦合的价值函数的捏合，这

里我们用$f(x)$表示神经网络的输出结果。

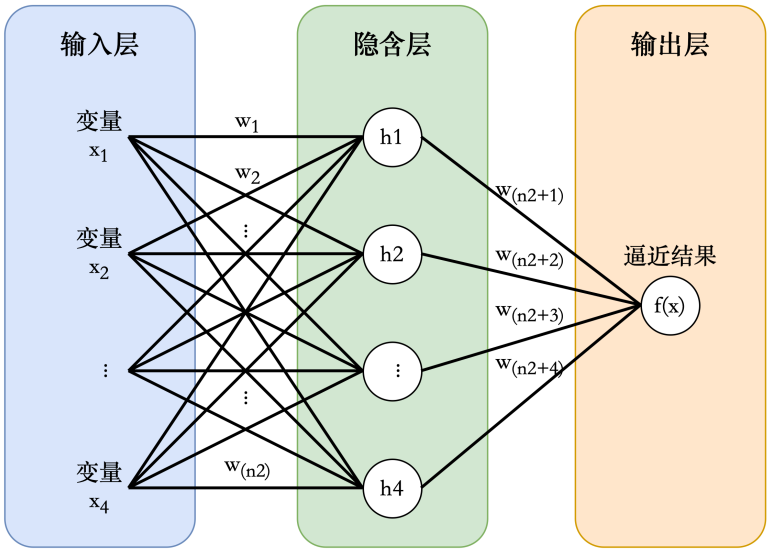

图 2-3 控制理论中用于逼近非线性项的神经网络原理图

根据上述神经网络的基本原理可知，神经网络可以用来近似逼近任意紧凑集合上的平滑函数，考虑如下矩阵形式的函数逼近：

$$V = \sigma(w^T h(x) + \varepsilon(x)) \tag{2-13}$$

其中$\omega \in \mathbb{R}^p$是网络的理想权重，p是隐藏层的神经元数量，$\varepsilon(x)$是网络的近似误差。$h(x) \in \mathbb{R}^{n\times p}$是激活函数的向量，典型激活函数的有 Sigmoid、tanh 和 ReLu 等。

但是由于实际网络的神经元数量不能为无穷，且数值逼近存在一定误差，逼近理想神经网络的估计值可以表示为：

$$\hat{V} = \sigma(\hat{w}^T h(x)) \tag{2-114}$$

其中$\hat{\omega} \in \mathbb{R}^p$是网络的估计权重，在神经网络中实现对实际值的估计。并且一般情况下，网络的近似误差$\varepsilon(x)$不等于 0，但是神经网络对于函数的近似误差可以随着神经网络的信息容量的增大而减小，即神经元数目、网络层数的适当增加可以提高神经网络的逼近性能。

2.3 障碍函数理论

首先，定义一个集合$c \subset \mathbb{R}^n$的前向不变性：如果对于任意的$x_0 \in c$，在一个预先定义的时期$t \in \mathcal{I}(x_0)$内，其中$\mathcal{I}(x_0)$是初始状态x_0对应的某个预设时间区间，系统动态的解满足$x(t) \in c$，则称集合c具有前向不变性。具有前向不变性的集合c是一个“安全的集合”，它由内部和边界两部分组成，可以表示为如下数学形式：

$$c = \{x \in \mathbb{R}^n \mid h(x) \geq 0\} \tag{2-11}$$

$$\partial c = \{x \in \mathbb{R}^n \mid h(x) = 0\} \tag{2-12}$$

$$\mathrm{Int}\,(c) = \{x \in \mathbb{R}^n \mid h(x) > 0\} \tag{2-13}$$

其中 $h \in \mathbb{R}^n$是边界函数，在集合c的边界上趋向于 0。如任何状态变量满足$x(t) \in \partial c$，则这个系统是一个安全的系统。

定义 2.1：如果一个连续函数$b(x)$满足以下三个重要特性，则称为障碍函数

1. 函数$b(x)$不到无穷大时，$x \in \mathrm{Int}(c)$时，即$||b(x)|| < \infty$。
2. 当状态变量x接近前向不变集的边界时，即$x \to \partial c$时，函数$b(x)$趋向于无穷大，可表示为 $\lim_{x \to \partial c} b(x) = \infty$。
3. 障碍函数在原点时均衡值消失，即$b(0) = 0$。

为了方便后续的安全保障控制器的设计。我们选择如下具体形式的障碍函数$b(x)$的形式：

$$b(x) = \left(\frac{1}{h(x)} - \frac{1}{h(0)}\right)^2 \tag{2-14}$$

其中$h(x)$是具有连续性的边界函数，确保$b(x)$满足定义 1 的所有三个性质。

在机器人的实际应用中，有一种与基于障碍函数的安全保障控制方法类似的机制，即人工势场法，该方法的原理为通过设置一定的势能场，利用与目标位置相互吸引，与危险区域相互排斥的势力作用达到安全保障的作用。在人工势场法中，势能的数学形式与障碍函数的性质相符合，故势能函数也是障碍函数定义中的一种。

2.4 分布式系统稳定理论

下文的稳定性分析针对具有N个孤立子系统的互联系统，其动态方程数值形式如下所示：

$$\dot{x}_i(t) = f_i(x_i(t)) + g_i(x_i(t))u_i(x_i(t)), i = 1,2,\ldots,N \tag{2-15}$$

下图 2-4 展示的互联系统由三部分组成：设计的分布式的控制器，N个子系统及其互联项：

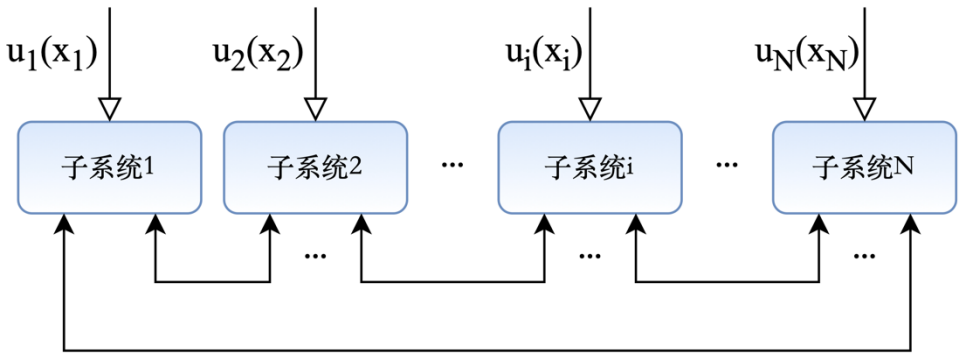

图 2-4 具有N个子系统的互联系统分布式控制结构图

在下文中，我们将表明，通过增大局部子系统的反馈增益，可以为上述互连系统建立一个稳定的分布式控制架构。接下来我们给出以下定理，说明如何选取反馈增益，以保障孤立子系统的渐近稳定性。

定理 2.1 考虑本节上述的孤立子系统，设计如下形式的反馈控制律：

$$\bar{u}_i(x_i) = \hat{\pi}_i \mu_i^*(x_i) = -\frac{1}{2}\pi_i R_i^{-1} g_i^T(x_i) \nabla J_i^*(x_i), i = 1,2,\dots,N \tag{2-16}$$

当$\pi_i \geqslant 1/2$（$i = 1,2,\dots,N$）时，该反馈控制策略可以确保N个闭环孤立的子系统在任意初始状态下都是渐进稳定的。因此针对本节定义的互联系统，存在N个正数$\pi_i^* > 0$（$i = 1,2,\dots,N$），使得对于任何 $\pi_i \geq \pi_i^*$（$i = 1,2,\dots,N$），上述反馈控制策略可以确保闭环互联系统是渐进稳定的。换而言之，该控制策略$\left(\bar{u}_1(x_1), \bar{u}_2(x_2), \dots, \bar{u}_N(x_N)\right)$是复杂系统的分散控制策略。下面给出该控制策略作用下，互联系统渐进稳定的证明。

证明：在此先定义如下简化表述的矩阵：

$$\Theta = \text{diag}\{\theta_1, \theta_2, \dots, \theta_N\} \tag{2-17}$$

$$\Lambda = \begin{bmatrix} \lambda_{11} & \lambda_{12} & \cdots & \lambda_{1N} \\ \lambda_{21} & \lambda_{22} & \cdots & \lambda_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{N1} & \lambda_{N2} & \cdots & \lambda_{NN} \end{bmatrix} \tag{2-18}$$

$$\Pi = \text{diag}\left\{\frac{1}{2}\left(\pi_1 - \frac{1}{2}\right), \frac{1}{2}\left(\pi_2 - \frac{1}{2}\right), \dots, \frac{1}{2}\left(\pi_N - \frac{1}{2}\right)\right\} \tag{2-19}$$

并且引入一个$2N$-维矢量：

$$\xi = \begin{bmatrix} Q_1(x_1) \\ Q_2(x_2) \\ \vdots \\ Q_N(x_N) \\ \left\| \left(\nabla J_1^*(x_1)\right)^T g_1(x_1) R_1^{-1/2} \right\| \\ \left\| \left(\nabla J_2^*(x_2)\right)^T g_2(x_2) R_2^{-1/2} \right\| \\ \vdots \\ \left\| \left(\nabla J_N^*(x_N)\right)^T g_N(x_N) R_N^{-1/2} \right\| \end{bmatrix} \tag{2-20}$$

系统渐进稳定的条件是如下形式的关于时间导数的 Lyapunov 函数为负定：

$$\begin{aligned} \dot{L}(x) &\leq -\xi^T \begin{bmatrix} \Theta & -\frac{1}{2}\Lambda^T\Theta \\ -\frac{1}{2}\Theta\Lambda & \Theta\Pi \end{bmatrix} \xi \\ &\triangleq -\xi^T \mathcal{A} \xi \end{aligned} \tag{2-21}$$

最终可以得到紧凑形式的稳定性条件，即有足够大的 π_i（$i = 1,2,\dots,N$），可以使矩阵$\mathcal{A}$是正定的，那么此时$\dot{L}(x) < 0$。因此，在上述分布式控制策略的控制作用下，闭环互连系统是渐进稳定的。至此，定理 2.1 证明完毕。

2.5 本章小结

本章主要阐述了本论文所涉及的自适应动态规划的基础算法，神经网络原理和障碍函数相关理论，给出了分布式互联系统的稳定性理论，明确本文所要解决的互联系统稳定安全控制问题的基础。

3　非零和博弈下的保障安全控制器设计

本章设计了一种安全保障控制器，以确保在受限状态空间中系统探索行为的安全性。该控制器通过一个基于已知模型的 RL 架构来获得非零和博弈的反馈纳什均衡解。为了解决持续激励的不确定性，采用了并行学习方法，在学习过程中同时使用历史和实时数据。为了减轻计算负荷，该方法利用单一评判网络进行近似计算。为了证明所方法的有效性，后续建立了一个两智能体的非零和博弈，分别针对凸性和非凸性的安全状态空间约束设计了安全保障控制器。

3.1 问题描述

在实际工业工程中，很多复杂系统通常包含大量的“传感器-控制器”。对于分布式的系统而言，每一个独立控制器可以视为一个智能体，其受到其他控制器作用的影响，同时又通过自身的决策动作改变其他传感器的观测量。

智能体之间具有十分复杂的强耦合性与相互作用关系。这种多智能体的博弈系统属于广义上的多智能体-互联系统，其分布式的控制结构对于后续章节的互联系统安全保障控制器有一定理论价值，故本章首先针对多智能体的非零和博弈进行分析，设计其安全保障控制器，以便后续互联系统控制器的分析与设计。

本章后续分析针对 N 智能体博弈问题，考虑非线性动力学连续时间仿射系统：

$$\dot{x} = f(x) + \sum_{i=1}^{N} g_i(x)u_i \tag{3-1}$$

其中$x(t) = [x_1, x_2, \dots, x_N] \in \mathbb{R}^n$，$u_i(t) \in \mathbb{R}^{m_j}$，$g_i(x) \in \mathbb{R}^{n\times m_j}$。$f(x) \in \mathbb{R}^n$是非线性项。并且假设$f(0) = 0$，$f(x)$具有局部 Lipschitz 性。设$U = [u_1, u_2, \dots, u_N]$是可行控制集合。

设置该智能体i的成本函数为V_i，其具体数值形式为：

$$V_i(x(0), u_1, u_2, \dots, u_N) = \int_t^{\infty} r_i(x(t), u_1, u_2, \dots, u_N)dt \tag{3-2}$$

其中$r_i \in \mathbb{R}_{\geq 0}$是即刻奖励函数，定义为：

$$r(x(t), u_1, u_2, \dots, u_N) = Q_i(x) + \sum_i^N u_j^T R_{ij} u_j \tag{3-3}$$

非零和博弈的目标是找到一个纳什均衡解的控制器组，即

$$U^* = [u_1^*, u_2^*, \dots, u_N^*] \tag{3-4}$$

该控制器组合使得整体的价值函数最小，相应的价值函数可以表示为：

$$V_i^*(x(0), u_1^*, u_2^*, \dots, u_N^*) = \min_{u_i} \int_t^{\infty} r_i(x(t), u_1^*, u_2^*, \dots, u_N^*)dt \tag{3-5}$$

相应的控制可以表示为：

$$u_i^* = \underset{u_i}{\operatorname{argmin}} V_i \tag{3-6}$$

为了得到控制策略的解析解，我们对价值函数V^*进行微分，得到汉密尔顿-雅各比方程，具体形式可以表示为：

$$0 = r_i(x(t), u_1, \dots, u_N) + \left(\Delta V_i^{*T}\left(f(x) + \sum_{j=1}^{N} g_j(x)u_i\right)\right. \tag{3-7}$$

根据最优控制理论，纳什均衡控制方案 $U^* = [u_1^*, u_2^*, \dots, u_N^*]$可以表示为：

$$u_i^* = -\frac{1}{2}R_{ii}^{-1}g_i^T(\Delta V_i)^T \tag{3-8}$$

将式（3-6）代入式（3-5），闭环的汉密尔顿-雅各比方程可以表示为：

$$0 = r_i(x(t), u_1, \dots, u_N) + \left(\Delta V_i^{*T}\left(f(x) + \frac{1}{4}\sum_{j=1}^{N} u_j^T R_{ij} u_j^*\right)\right) \tag{3-9}$$

3.2 基于障碍函数的保障安全控制器设计

3.2.1 障碍函数梯度方法

上一节介绍了多智能体非零和博弈系统和障碍函数的定义。受文献[25]的启发，下面设计一种基于障碍函数的补偿控制器：

$$u_b(x) = -\alpha_i g_i(x)^T \Gamma(\nabla b(x)^T) \tag{3-10}$$

其中α_i是选定的控制增益，Γ是防止梯度达到无穷大的投影。

上述公式的函数$b(x)$是我们在上一节中定义的障碍函数。若$b(x)$倾向于接近无穷大，我们可以选择 tanh 或 sigmoid 函数。

引理 3.1[25] 对于 N 个智能体的式（3-1）所表示的动态系统，假设内部集Int (c)包含原点x_0。如果对于所有$t \in \mathcal{I}(x_0)$，障碍函数不接近无穷大，即：

$$||b(x(t))|| < \infty \tag{3-11}$$

则内部集Int (c)具有向前不变的特性。

根据定理 1 的事实，在障碍函数是有限的条件下，系统的安全性得到保障。为了设计特定的多智能体非零和博弈的控制器，我们给出了以下假设。

假设 3.1 对于 N 个智能体的式（3-1）所展示的动态系统，给定一个前向不变集c，假设以下属性成立：

1 非线性动态$f(x)$是由一个非负增加的函数来约束的$\bar{f} \in \mathbb{R}_{\geq 0}$，即$||f(x)|| \leq \bar{f}(x)$和$\lim_{x \to \partial c} \bar{f}(x) < \infty$。

2 存在一个下限为$g(x)$的下限，即$\underline{g} \leq ||g(x)||$对于所有$x \in c$，其中$g \in \mathbb{R}_{>0}$是一个正常数。

3 边界集的非零邻域∂c被定义为$\mathcal{N}(\partial c)$，它满足对所有$x \in \mathcal{N}(\partial c)$，安全保障控制器不会消失，即$||\Gamma(\nabla b(x))g(x)|| \neq 0$。

基于假设 3.1 和定理 3.1，我们有如下定理来获得安全控制策略，它使式（3-1）所

展示的动态系统的内部集Int (x)对系统(1)来说是前向不变的。

引理 3.2[25] 对于 N 个智能体的式（3-1）所展示的动态系统，一个前向不变集$c \subset \mathbb{R}^n$，其满足$0 \in$ Int (c)，并定义b作为多智能体博弈的障碍函数。基于假设 3.1 成立，式（3-8）的安全保障控制器$u = u_b(x)$确保内部集Int (c)是向前不变的，其中（3-1）的安全性得到保障。上述结果表明，当安全保障项被用作控制器时，内部集Int (c)被确保是向前不变的。接下来，我们得到了一个常规的 ADP 控制器来解决纳什均衡问题。之后，它与安全保障控制器相结合，保障在任何凸/非凸集的状态约束下安全探索。

定义 3.1 假设连续时间控制器$u_i(x,t)$被设计为在状态空间中是局部 Lipschitz 的，并且满足以下条件$u_i(0,t) = 0$对于$t \in \mathcal{I}(x_0)$。当假设 1 成立，输入增益矩阵的动态有界且为$\|g(x)u_i(x,t)\| \le \bar{g}_u$，其中$\bar{g}_u$与$\bar{f}$的定义为函数的上限。得到的控制器为如下形式：

$$u_{b,i} = u_i(x,t) + u_b(x) \tag{3-12}$$

上述控制器确保内部集合Int (C)是互联动态系统的前向不变集。该控制器并且还保障了状态空间的原点是（3-1）的最终均衡解。

通过一个稳定控制器和一个安全补偿项，我们得出了一个控制器$u_{b,i}$使 Int(c)保持其正向不变性。在下一小节中，本文将详细介绍使用 RL 方法对名义控制器进行在线逼近以避免求解过程中的维数灾难。

3.2.2 神经网络近似

为了获得控制政策的分析解u_i和价值函数的分析解V_i的分析解，本小节利用一个单一评判网络来逼近价值函数V_i，其形式为：

$$V_i = \omega_i^T \phi(x) + \varepsilon_i(x)^T \tag{3-13}$$

其中$\omega_i \in \mathbb{R}^{p_i}$是单一评判网络的理想权重，而$\phi(x) \in \mathbb{R}^{n \times p_i}$是激活函数的向量，$p_i$是隐藏层的神经元数量，$\varepsilon_i(x)$是评判网络的近似误差。

价值函数V_i的梯度表示为：

$$\nabla V_i = \nabla \phi(x)^T \omega_i + \nabla \varepsilon_i(x) \tag{3-14}$$

理想价值函数的估计近似值V_i定义为：

$$\hat{V}_i = \hat{\omega}_i^T \phi(x) \tag{3-15}$$

其中$\hat{\omega}_i \in \mathbb{R}^{p_i}$是单一网络的估计权重，在单一网络中实现对值$V_i$的估计。

为了减少计算负荷，控制的近似是通过单网络方法实现的，其形式为：

$$u_i = -\frac{1}{2} R_{ii}^{-1} g_i^T (\Delta \phi_i^T(x) \omega_i + \Delta \varepsilon_i^T(x)) \tag{3-16}$$

利用估计值的梯度，使用权重ω_i，实际的控制器可以用以下形式表示：

$$\hat{u}_i = -\frac{1}{2} R_{ii}^{-1} g_i^T \nabla \phi_i^T(x) \hat{\omega}_i \tag{3-17}$$

然后，将安全保障项（3-8）添加到控制策略的项中得到最终的安全保障控制器：

$$u_{b,i} = \hat{u}_i - \frac{\alpha_i}{2} R_{ii}^{-1} g_i(x)^T \nabla b(x)^T \tag{3-18}$$

3.2.3 基于并行学习的参数更新律

基于式（3-7），式（3-12）和式（3-14），可以定义汉密尔顿-雅各比方程的近似误差，其形式为：

$$\delta_i = \Omega_i^{\mathrm{T}} \sigma_i + x^{\mathrm{T}} Q_i x + \sum_{j=1}^{N} \frac{1}{4} \omega_j^{\mathrm{T}} \sigma_j' G_{ij} \sigma_j'^{\mathrm{T}} \omega_j + \nabla \varepsilon_i^T \Omega_i \tag{3-19}$$

其中$G_j = g_j R_{jj}^{-1} g_j^{\mathrm{T}}$，$G_{ij} = g_j R_{jj}^{-1} R_{ij} R_{jj}^{-1} g_j^{\mathrm{T}}$，$\sigma_j = \nabla \phi(x)\left(f + \sum_{k=1}^{N} g_k u_{b,k}\right)$和$\Omega_i = \sigma_i' f - \frac{1}{2}\sum_{j=1}^{N} \sigma_i' G_j \sigma_j'^{\mathrm{T}} \hat{\omega}_j$。

我们简化接下来简化符号表示表示得到：

$$e_i = \Omega_i^{\mathrm{T}} \sigma_i + x^{\mathrm{T}} Q_i x + \sum_{j=1}^{N} \frac{1}{4} \hat{\omega}_j^{\mathrm{T}} \sigma_j' G_{ij} \sigma_j'^{\mathrm{T}} \hat{\omega}_j \tag{3-20}$$

$$\nabla \varepsilon_i^T \Omega_i = -\varepsilon_{\mathrm{ham},i} \tag{3-21}$$

进一步表示为：

$$\delta_i = e_i - \varepsilon_{\mathrm{ham},i} \tag{3-22}$$

为了得到一个可控的控制策略u，并方便后续的优化过程，我们首先将历史数据和实时数据以总能量E_i的形式结合起来，可以表示为：

$$E_i = \frac{1}{2}\left[\frac{\sigma_i^2}{\left(1+\sigma_i^T \sigma_i\right)^2} + \sum_{k=1}^{M} \frac{\left(\sigma_i^k\right)^2}{\left(1+\left(\sigma_i^k\right)^T \sigma_i^k\right)^2}\right] \tag{3-23}$$

其中σ_i^k是第k个历史数据。M是历史数据σ_i的总数。

根据上述目标函数的性质，我们可以得到基于最小二乘法的适应法，对估计的评判网络权重$\hat{\omega}_i$的更新律如下所示:

$$\dot{\hat{\omega}}_i = -\beta_i \frac{\partial E_i}{\partial \omega_i} = -\beta_i \frac{\sigma_i e_i}{\left(1+\sigma_i^T \sigma_i\right)^2} - \beta_i \sum_{k=1}^{M} \frac{\sigma_i^k e_i^k}{\left(1+\left(\sigma_i^k\right)^T \sigma_i^k\right)^2} \tag{3-24}$$

其中β_i是每个智能体的学习收益，确定每个智能体的单网络权重的收敛速度ω_i。

虽然较高的学习率β_i可能会加快收敛速度，但由于安全保障控制器的谨慎性，过高的学习率也可能是一次模型训练的灾难。

3.3 控制器性能分析

本节为了研究所提出的名义 ADP 控制器的稳定性，本节基于李雅普诺夫方法建立了控制器的稳定性分析。首先，我们引入以下假设，以方便稳定性的证明。

假设 3.2 为了方便下面的李雅普诺夫分析，假设以下有界条件成立。

1. 单一网络的理想权重ω_i的单一评判网络是有界的，即$\|\omega_i\| \leq \bar{\omega}_i$。
2. 单一网络的近似误差$\varepsilon_i(x)$的单一评判网络和其相应的梯度是有界的，即

$\|\varepsilon_i(x)\| \le \bar{\varepsilon}_i$和$\|\nabla\varepsilon_i(x)\| \le \nabla\bar{\varepsilon}_{\max,i}$。

3. 单一网络的激活向量$\phi_i(x)$的激活向量及其相应的梯度是有边界的，即$\|\phi_i(x)\| \le \bar{\phi}_i$和$\|\nabla\phi_i(x)\| \le \nabla\bar{\phi}_i$。
4. 汉密尔顿残差是有界的，即是$\|\varepsilon_{\text{ham},i}\| \le \bar{\varepsilon}_{\text{ham},i}$。
5. $g_i(x)$是有边界Ω，即$g_i(x) \le \bar{g}_i$。

定理 3.1 对于N智能体的式（3-1）所展示的动态系统，$c \subset \mathbb{R}^n$为前向不变集，从对边界的定义中可以看出，满足$0 \in \text{Int}(c)$，并定义b作为多智能体博弈的障碍函数。基于假设 3.1 和 3.2 成立，并且满足：

$$\bar{g}_i\bar{\phi}_j < 0 \tag{3-25}$$

$$\rho < 0 \tag{3-26}$$

$$\beta_i\left(\frac{p+1}{2} - 2\lambda_{\min}(\Gamma_k)\right) < 0 \tag{3-27}$$

其中$\rho = \sum_{i=1}^{N}\left[\beta_i\frac{p+1}{2}\bar{\varepsilon}_i^2 - (\bar{\omega}_i\bar{\phi}_i + \bar{\varepsilon}_i)\sum_{j=1}^{N}\left(\frac{1}{2}G_j\bar{\phi}_i\|\hat{\omega}_j\| - g_i\bar{\varepsilon}_i\right)\right]$。

因此式（3-15）中的控制策略和式（3-19）中的基于学习的并行更新法，保障了多智能体博弈的动态表式（3-1）的内部集$\text{Int}(c)$对多智能体博弈的动态表式（3-1）来说是前向不变的。此外，存在一个全局均衡点，状态渐进收敛为零。

证明：我们定义以下李雅普诺夫函数用于稳定性分析：

$$V_L = \sum_{i=1}^{N}\left(V_i + V_{\omega,i}\right) \tag{3-28}$$

其中$V_{\omega,i} = \frac{1}{2}\tilde{\omega}_i^{\mathrm{T}}\tilde{\omega}_i$是单一评判网络权重的附加误差项。

对于每个智能体，我们有

$$\dot{V}_i = \left(\frac{\partial V_i(x)}{\partial x}\right)^T\left[f(x) - \frac{1}{2}\sum_{j=1}^{N} g_j(x)R_{jj}^{-1}g_j^T\nabla\phi_i^T(x)\hat{\omega}_j\right] \tag{3-29}$$

结合控制器（3-14）、汉密尔顿-雅各比方程（3-7）和假设 3.2，我们可以得到

$$\dot{V}_i \le -r_i - (\bar{\omega}_i\bar{\phi}_i + \bar{\varepsilon}_i)\sum_{j=1}^{N}\left(\frac{1}{2}G_j\bar{\phi}_i\|\hat{\omega}_j\| - g_i\bar{\varepsilon}_i\right) \tag{3-30}$$

将每个智能体的权重误差项进行微分$V_{\omega,i}$得出以下公式

$$\dot{V}_{\omega,i} = \tilde{\omega}_i^T\dot{\tilde{\omega}}_i \tag{3-31}$$

然后，基于更新法则的式（3-19），每个智能体的单一评判网络权重误差的动态可以表示为：

$$\dot{\tilde{\omega}}_i = -\beta_i[\Gamma_a(t) + \Gamma_k]\tilde{\omega}_i(t) + \beta_i\Lambda_a \tag{3-32}$$

其中的符号可以表示为：

$$\Gamma_a(t) = \frac{\sigma_i(\sigma_i)^T}{[1+(\sigma_i)^T\sigma_i]^2}, \tag{3-33}$$

$$\Gamma_k = \sum_{k=1}^{p} \frac{\sigma_i\left(\sigma_i^k\right)^{\mathrm{T}}}{\left[1+\left(\sigma_i^k\right)^{\mathrm{T}}\sigma_i\right]^2} \tag{3-34}$$

$$\Lambda_a = \frac{\sigma_i \varepsilon_{\mathrm{ham},i}}{[1+(\sigma_i)^T\sigma_i]^2} + \sum_{k=1}^{p} \frac{\sigma_i^k \varepsilon_{\mathrm{ham},i}^k}{\left[1+\left(\sigma_i^k\right)^T\sigma_i^k\right]^2} \tag{3-35}$$

将式（3-25）插入式（3-24）中，得到

$$\dot{V}_{\omega,i} \le \beta_i \left[\frac{p+1}{2} - 2\lambda_{\min}(\Gamma_k)\right] \|\tilde{\omega}_i\|^2 + \beta_i \frac{p+1}{2} \bar{\varepsilon}^2_{\mathrm{hmax},i} \tag{3-36}$$

将不等式（3-23）和式（3-28）结合起来，可以得到

$$\dot{V} \le -\sum_{i=1}^{N} r_i + \rho + \sum_{i=1}^{N} \left[\bar{g}_i \bar{\phi}_i + \beta_i \left(\frac{p+1}{2} - 2\lambda_{\min}(\Gamma_k)\right)\right] \|\tilde{\omega}_i\|^2 \tag{3-37}$$

对于每个智能体，如果假设 3.2 中的式（3-20）成立，我们有$\dot{V}_L \le 0$。那么，根据李雅普诺夫稳定性定理，所提出的控制器（3-15）的稳定性得到了保障。根据前向不变性和渐进稳定性的特性，可以保障迫使多智能体博弈的动态表式（3-1）达到纳什均衡的安全性。至此，定理 3.1 证明完毕。

3.4 本章小结

本章主要针对同为复杂系统的非零和博弈系统进行研究，为后续更加复杂的互联系统安全保障控制器设计提供研究基础。本章首先设计了一种基于障碍函数的安全保障控制器的补偿项，以保障在受限状态空间中的安全探索。其次通过建立一个基于已知模型的强化学习架构，获得 N 个智能体的非零和博弈的纳什均衡解。为了处理持久性激励的不确定性，本章应用了同时使用历史和实时数据的并行学习方法训练无激励风险的网络。为降低对计算资源需求，本章利用了单评判网络对价值函数进行逼近。

4　基于保障安全控制器的互联系统安全稳定控制

针对上一章节与互联系统同为复杂系统的非零和博弈系统，我们研究了其安全保障控制器，在此类安全保障控制器设计的启发下，本章提出了一种用于复杂互连系统基于强化学习的稳定控制策略。本章的方法建立在安全 RL 和并行学习的基础上。本章的主要内容分为三个部分：首先建立了一个基于在线学习的强化学习架构，用于求解最优控制问题的 HJB 方程。其次基于前文博弈系统的设计，引入基于障碍函数的安全保障补偿项来实现对部分子系统的安全控制。最后使用单一评判神经网络技术与并行学习方法实现对价值函数的逼近。首先需要对后续研究的互联系统的具体形式进行定义。

4.1 问题描述

本章节的研究对象为互连系统，这种系统是由N子系统组成，其动力学特性描述为：

$$\dot{x}_i(t) = f_i(x_i(t)) + g_i(x_i(t))(u_i(x_i(t)) + \mathbf{I}(x(t))) \tag{4-1}$$

其中$i = 1, \dots, N$，是每个子系统的状态。$x_i(t) \in \mathbb{R}^n$是每个子系统的状态，$u_i(x_i(t)) \in \mathbb{R}^n$是第$i$个子系统的控制输入向量，$x(t) = [x_1(t), \dots, x_N(t)]$是一个由所有子系统状态组成的状态向量。此外，$f_i(x_i)$代表非线性动态，$g_i(x_i)$代表输入增益矩阵，$\mathbf{I}(x(t))$为互联项。函数$f_i(\cdot)$和$g_i(\cdot)$具有局部 Lipschitz 性。

对于$i = 1, \dots, N$，设定$x_i = 0$为第i个子系统的平衡状态。假设当$x_i = 0$时，控制输入向量为$u_i(x_i) = 0$，这意味着一旦系统达到平衡点，控制器的输入将会停止。

为了实现互联系统的控制目标，本章首先设计了将系统引导到平衡点的最优控制策略。由于子系统在控制过程中最终会达到相同的目标点，我们将子系统的最优控制器设计为相同的通用控制器。因此，考虑与系统（4-1）相对应的孤立的子系统，表示为：

$$\dot{x}_i(t) = f_i(x_i(t)) + g_i(x_i(t))u_i(x_i(t)), i = 1,2, \dots, N \tag{4-2}$$

我们假设每个子系统i都是可控的，并且存在一个在$\Omega \in \mathbb{R}^n$连续的控制策略，可以渐进地稳定子系统i。为了处理无限时间视角的最优控制问题，本小节的设计目标是找到控制策略$u_i(x_i)$（$i = 1, \dots, N$），使下列的局部成本函数最小化：

$$J_i(x_i, u(\cdot)) = \int_0^\infty Q_i^2(x_i(\tau)) + u_i^T(x_i(\tau))R_i u_i(x_i(\tau))d\tau \tag{4-3}$$

其中$i = 1,2, \dots, N$，$Q_i(x_i)$是一个正定函数，并且满足$q_i(x_i) \le Q_i(x_i)$。

基于最优控制理论，设计的反馈控制不仅要使子系统在Ω_i（$i = 1, \dots, N$）稳定，而且还要保障成本函数是有限的。对于任意一组可以实现上述稳定目标的控制策略$\mu_i \in \Phi_i(\Omega_i)$（$i = 1, \dots, N$），如果如下形式相应的成本函数的估计值$\hat{J}_i(x_i, \mu(\cdot))$是连续可微的：

$$\hat{J}_i(x_i, \mu(\cdot)) = \int_0^\infty Q_i^2(x_i(\tau)) + \mu_i^T(x_i(\tau))R_i\mu_i(x_i(\tau))d\tau \tag{4-4}$$

则可以通过对上述成本函数两边进行求导得到每个孤立子系统的汉密尔顿函数：

$$H_i\left(x_i,\mu_i,\nabla\hat{J}_i(x_i,\mu_i)\right)=Q_i^2(x_i)+\mu_i^T(x_i)R_i\mu_i(x_i)+\left(\nabla\hat{J}_i(x_i)\right)^T\dot{x}_i \tag{4-5}$$

其中$i=1,\dots,N$。故每个子系统的最优成本函数可以表示为：

$$V_i=\min_{\mu_i\in\Phi_i(\Omega_i)}\int_0^\infty\{Q_i^2(x_i(\tau))+\mu_i^T(x_i(\tau))R_i\times\mu_i(x_i(\tau))\}d\tau \tag{4-6}$$

其中$i=1,\dots,N$，$J_i^*(x_i)$满足以下HJB方程：

$$0=\min_{\mu_i\in\Psi_i(\Omega_i)}H_i(x_i,\mu_i,\nabla V_i) \tag{4-7}$$

其中$V_i=\partial V_i/\partial x_i$，那么子系统的最优控制策略即为：

$$u_i^*(x_i)=\arg\min_{\mu_i\in\Psi_i(\Omega_i)}H_i(x_i,\mu_i,\nabla V_i)=-\frac{1}{2}R_i^{-1}g_i^T(x_i)\nabla V_i \tag{4-8}$$

根据文献[11]中的定理 1，分布式控制子系统的反馈控制表示为如下形式：

$$u_i=\beta_i u_i^*(x_i)=-\frac{1}{2}R_i^{-1}g_i^T(x_i)\nabla J_i^*(x_i),i=1,\dots,N \tag{4-9}$$

其中β_i是非负常数，即满足$\beta_i\geq 0$。

4.2 基于障碍函数的保障安全控制器设计

4.2.1 障碍函数梯度方法

本节接下来介绍安全保障项。受文献[25]的启发，本章直接给出安全控制器中基于障碍函数的补偿项，如下所示：

$$u_b^*(x_i)=-\alpha_i g_i^T(x_i)\nabla b^T(x_i),i=1,\dots,N \tag{4-10}$$

其中α_i是所选择的控制增益，$b(x)$是我们在前文中定义的障碍函数。需要保障安全的子系统的反馈控制可以被表述为：

$$u_i=\beta_i u_i^*(x_i)+\gamma_i u_b^*(x_i),i=1,\dots,N \tag{4-11}$$

其中γ_i是由用户设定的正常数。根据文献[11]中的定理 1，通过选择适当的γ_i和β_i，所设计的控制器可以使互连系统达到渐进稳定。因此，本节设计的安全保障控制器理论上可以保障子系统在设定的状态范围内运行，即始终处于安全区域c。

4.2.2 神经网络近似

为了得到控制策略的解析解u_i和价值函数V_i，本小节利用单一评判网络技术来实现对价值函数V_i的逼近，具体形式为：

$$V_i=W_i^T\phi(x)+\epsilon_i(x) \tag{4-12}$$

其中$W_i\in\mathbb{R}^{p_i}$是单一评判网络的理想权重，而$\phi(x)\in\mathbb{R}^{n\times p_i}$是激活函数的向量，$p_i$是隐藏层的神经元数量，$\epsilon_i(x)$是评判网络的近似误差，$V_i$的梯度可以表示为：

$$\nabla V_i=\nabla\phi(x)^T W_i+\nabla\epsilon_i^T(x) \tag{4-13}$$

理想值函数的估计近似值V_i被定义为：

$$\hat{V}_i = \hat{W}_i^T \phi(x) \tag{4-14}$$

其中$\hat{W}_i \in \mathbb{R}^{p_i}$是单一评判网络的估计权重，在单一评判网络中实现对实际值的估计，即 $\hat{V}$。为了减少计算量，控制器u_i利用已定义的单一神经网络实现：

$$u_i = -\frac{1}{2} R_i^{-1} g_i^T (\nabla \phi_i^T(x) W_i + \nabla \epsilon_i^T(x)) \tag{4-15}$$

利用估计值的梯度，使用价值函数的逼近权重W_i，实际的控制器可以表示为：

$$\hat{u}_i = -\frac{1}{2} R_i^{-1} g_i^T \nabla \phi_i^T(x) \hat{W}_i \tag{4-16}$$

根据式（4-14）和式（4-15），本节将汉密尔顿-雅各比方程的近似误差定义为:

$$H_i(x, u_i, W_i) = u_i^T R_i u_i + Q_i^2 + [W_i^T \nabla \phi_i + (\nabla \epsilon_i)^T] \dot{x}_i = -\nabla \epsilon_i^T (f_i + g_i u_i) \tag{4-17}$$

若代入的控制策略改为其估计值，则汉密尔顿-雅各比方程的近似误差变为：

$$H_i\left(x, \hat{u}_i, \hat{W}_i\right) = \hat{u}_i^T R_i \hat{u}_i + Q_i^2 + \left(\hat{W}_i^T \nabla \phi_i\right)(f_i + g_i \hat{u}_i) = e_i \tag{4-18}$$

为了简化符号，可以简化如下表述：$\nabla \epsilon_i^T (f_i + g_i u_i) = -e_{H,i}$和$\omega_i = \nabla \phi_i (f_i + g_i \hat{u}_i)$。为了得到一个可接受的控制策略$u$并便于进行下面的优化，首先将历史数据和实时数据结合起来，以总能量目标的形式E_i表示：

$$E_i = \frac{1}{2}\left[\frac{e_i^2}{\left(1+\omega_i^T \omega_i\right)^2} + \sum_{k=1}^{M} \frac{\left(e_i^k\right)^2}{\left(1+\left(\omega_i^k\right)^T \omega_i^k\right)^2}\right] \tag{4-19}$$

其中ω_i^k是ω_i的第k个历史数据。M 是历史数据的总数。设$\bar{\omega}_i = [\omega_i^1 \ldots \omega_i^M]$是历史数据栈。

根据上述目标函数的性质，可以得到基于最小二乘法的自适应更新律，故得到如形式的评判网络权重估计值$\hat{W}_i$的更新律：

$$\dot{\hat{W}}_i = -a_i \frac{\partial E_i}{\partial \hat{W}_i} = -a_i \frac{\omega_i e_i}{\left(1+\omega_i^T \omega_i\right)^2} - a_i \sum_{k=1}^{M} \frac{\omega_i^k e_i^k}{\left(1+\left(\omega_i^k\right)^T \omega_i^k\right)^2} \tag{4-20}$$

其中a_i是每个子系统的学习增益，确定每个单一评判网络权重W_i的收敛速度。

4.3 控制器性能分析

为了研究上文设计控制器的稳定性，本小节基于李雅普诺夫方法对前文的控制方法进行稳定性分析。首先给出$\tilde{W}_i$的误差动态：

$$\dot{\tilde{W}}_i(t) = -a_i \frac{\omega_i}{\omega_i^T \omega_i + 1}\left[\frac{\omega_i^T \tilde{W}(t) + e_{H,i}}{\omega_i^T \omega_i + 1}\right] - a_i \sum_{k=1}^{M} \frac{\omega_i^k}{\left(\omega_i^k\right)^T \omega_i^k + 1}\left[\frac{\omega^T(t_i) \tilde{W}(t) + e_{H,i}^k}{\left(\omega_i^k\right)^T \omega_i^k + 1}\right] \tag{4-21}$$

定理 4.1 在以下条件下，评判性权重是一致最终有界的：

1．$\mathrm{rank}(\bar{\omega}_i) = p_i$；

2．$e_{H,i}$的上界为 $e_{H\max,i}$

证明：定义如下形式的李雅普诺夫函数，以用于后续稳定性分析：

$$V_i(t)=\frac{1}{2a_i}\tilde{W}_i^T\tilde{W}_i \tag{4-22}$$

对于每个子系统，可以得到如下形式李雅普诺夫函数的导数：

$$\dot{V}_i=-\tilde{W}_i^T(\zeta_a(t)+\zeta_i(t))\tilde{W}_i+\tilde{W}_i^T\eta_i \tag{4-23}$$

其中实时数据形成的归一化二次型项为：

$$\zeta_a(t)=\frac{\omega_i(\omega_i)^T}{[1+(\omega_i)^T\omega_i]^2} \tag{4-24}$$

由存储的 p 个历史数据形成的归一化二次型项为：

$$\zeta_i=\sum_{k=1}^{p}\frac{\omega_i\left(\omega_i^k\right)^{\mathrm{T}}}{\left[1+\left(\omega_i^k\right)^{\mathrm{T}}\omega_i\right]^2} \tag{4-25}$$

并且实时数据与历史数据相结合的误差项可以表示为：

$$\eta_i=\frac{\omega_i e_{H,i}}{[1+(\omega_i)^T\omega_i]^2}+\sum_{k=1}^{p}\frac{\omega_i^k e_{H,i}^k}{\left[1+\left(\omega_i^k\right)^T\omega_i^k\right]^2} \tag{4-26}$$

并且由于$\zeta_a>0$，可以得到不等式关系：

$$\dot{V}_i\le-\tilde{W}_i^T\zeta_i(t)\tilde{W}_i+\tilde{W}_i^T\eta_i \tag{4-27}$$

根据假设条件 1，即$\mathrm{rank}\,(\bar{\omega}_i)=k$，可以进一步得到：

$$\dot{V}_i\le-\lambda_{\min}(\zeta_i)\|\tilde{W}_i\|^2+\|\tilde{W}_i\|\left(\frac{M+1}{2}\right)e_{H\max,i} \tag{4-28}$$

故在$\|\tilde{W}_i\|\ge\frac{(M+1)e_{H\max,i}}{2\lambda_{\min}(\zeta_i)}$的条件下，$\dot{V}_i$可以保障为负值。因此权重的误差$\tilde{W}_i$在上述条件下为一致最终有界的。至此，定理 4.1 证明完毕。

4.4 本章小结

针对更加复杂、耦合性更强的互联系统，本章设计了一种基于模型已知强化学习的安全保障且可以稳定的控制器。首先建立了包含多个子系统的互连系统和安全保障、稳定优化问题。利用在线强化学习求解最优控制的 HJB 方程。其次基于上文的研究，设计一种新颖的安全保障项，实现子系统的安全保障控制。为了训练没有激励风险的网络，本章使用了并行学习方法，训练用于逼近值函数的单一评判网络，得到适用性广泛的安全保障控制器。最后，本章证明了评判者网络的估计误差动态是最终有边界的。接下来，第五章的仿真例子将展示本控制方法的安全性和稳定性。

5　数值仿真验证

本章主要内容为对上文所设计的基于自适应动态规划的互联系统抗干扰策略的安全性和稳定性设置数值仿真验证，并对数值仿真结果进行分析。

5.1 仿真参数设定

5.1.1 非零和博弈系统仿真参数

本小节将对非零和博弈系统的数值仿真进行参数设定。本仿真将考虑一个包含两个智能体的博弈系统。两个智能体的利益为近似的，属于完全协作博弈。具体的仿真参数设置如下所示：

$$\dot{x} = f(x) + g_1(x)u_1 + g_2(x)u_2 \tag{5-1}$$

其中$x \in \mathbb{R}^2$，$u_1, u_2 \in \mathbb{R}$。我们选择如下形式的非线性动态：

$$f = \begin{bmatrix} x_2 - 2x_1 \\ -\frac{1}{2}x_1 - x_2 + \frac{1}{4}x_2(\cos(2x_1) + 2)^2 \\ +\frac{1}{4}x_2(\sin(4x_1^2) + 2)^2 \end{bmatrix} \tag{5-2}$$

两个智能体的动力学输入矩阵分别设置为：

$$g_1 = \begin{bmatrix} 0 \\ \cos(2x_1) + 2 \end{bmatrix} \tag{5-3}$$

$$g_2 = \begin{bmatrix} 0 \\ \sin(4x_1^2) + 2 \end{bmatrix} \tag{5-4}$$

价值函数的具体构造形式选择为：

$$\begin{aligned} V_i(x(0), u_1, u_2, \dots, u_N) &= \int_t^\infty r_i(x(t), u_1, u_2, \dots, u_N)dt \\ &= \int_t^\infty \left(Q_i(x) + \sum_i^N u_j^T R_{ij} u_j\right)dt \end{aligned} \tag{5-5}$$

其中的权重矩阵大小为$Q_1 = 2Q_2 = 2I_2$和$R_{11} = R_{12} = 2R_{21} = 2R_{22} = 2$。可以看出两个智能体的价值取向一致，故为完全协作博弈。

为了稳定两个智能体的非零和博弈，我们提出的控制器的目标是保障状态$x(t)$趋近于零，同时确保$x(t)$不超出安全边界集∂c。在这个数值模拟中，边界集∂c有一个指定的边界函数$h(x) = px_2^2 - x_1 + 1$，详细定义可见第二章，其中p是决定安全集的凸性属性的系数。为简化表示，我们选择$p = -1$表示凸集，而$p = 1$为非凸集。

选取初始状态为$x_0 = [-4, 2.2]$，这是一个非常接近安全集合边界的状态。通过建立障碍函数 $b(x) = (1/h(x) - 1/h(0))^2$和设置控制器增益$\alpha_i = 0.1$，每次评估学习都会得到一个安全保障控制器。每个参与者的学习率被选为$\beta_1 = 1$和$\beta_2 = 0.1$，将权重初始化为$\hat{\omega}_i(t_0) = [0.5, 0.5, 0.5]^T$。

5.1.2 互联系统仿真参数

本小节设置一个互联系统仿真实例来说明第四章建立的分布式控制安全保障策略的适用性，该系统由两个互联的子系统组成，具体的数值表示如下所示：

$$\dot{x}_1 = f_1(x) + \begin{bmatrix} 0 \\ \cos(2x_{11}) + 2 \end{bmatrix} (u_1 + (x_{11} + x_{22}) \sin x_{12}^2 \cos(0.5x_{21})) \quad (5\text{-}6)$$

$$\dot{x}_2 = f_2(x) + \begin{bmatrix} 0 \\ x_{21} \end{bmatrix} \left(u_2 + 0.5(x_{12} + x_{22}) \cos\left(e^{x_{21}^2}\right)\right) \quad (5\text{-}7)$$

$$f_1(x) = \begin{bmatrix} -x_{11} + x_{12} \\ -0.5(x_{11} + x_{12}) - 0.5x_{12}(\cos(2x_{11}) + 2)^2 \end{bmatrix} \quad (5\text{-}8)$$

$$f_2(x) = \begin{bmatrix} x_{22} \\ -x_{21} - 0.5x_{22} + 0.5x_{21}^2 x_{22} \end{bmatrix} \quad (5\text{-}9)$$

其中$x_1 = [x_{12}, x_{11}]^T \in \mathbb{R}^2$表示子系统 1 的状态，$u_1$表示子系统 1 的控制输入。$x_2 = [x_{21}, x_{22}]^T \in \mathbb{R}^2$表示子系统 2 的状态，$u_2$表示子系统 2 的控制输入。在这个数值模拟中，边界集∂c有一个指定的边界函数$h(x_i) = x_{i,2}^2 - x_{i,1} + 1$。初始状态被选择为$x_{i,0} = [1.2, 0.47]$。通过选择障碍函数为 $b(x_i) = (1/h(x_i) - 1/h(0))^2$的增益，可以得到一个安全保障控制器，并且令$\alpha_i = 1$就可以得到一个安全保障控制器。每个子系统的学习率选取为$a_1 = 1$和$a_2 = 1$，而权重初始化为$\hat{\omega}_i(t_0) = [1,1,1]^T$。

5.2 数值仿真结果

5.2.1 非零和博弈下的保障安全性验证

为了证明所提出的技术的有效性，在非凸/凸状态约束的情况下，对有和没有安全保障控制器的系统进行了模拟。对于凸的情况，我们设定$p = -1$，主要结果见图 5-1，学习过程见图 2。如图 5-1 所示，通过利用式（3-15）中的控制策略和增加的安全保障项，状态$x(t)$被稳定在零状态，同时永远不会离开安全集。相比之下，没有安全保障项的控制策略将状态稳定到零，但在开始时，状态轨迹违反了安全约束。

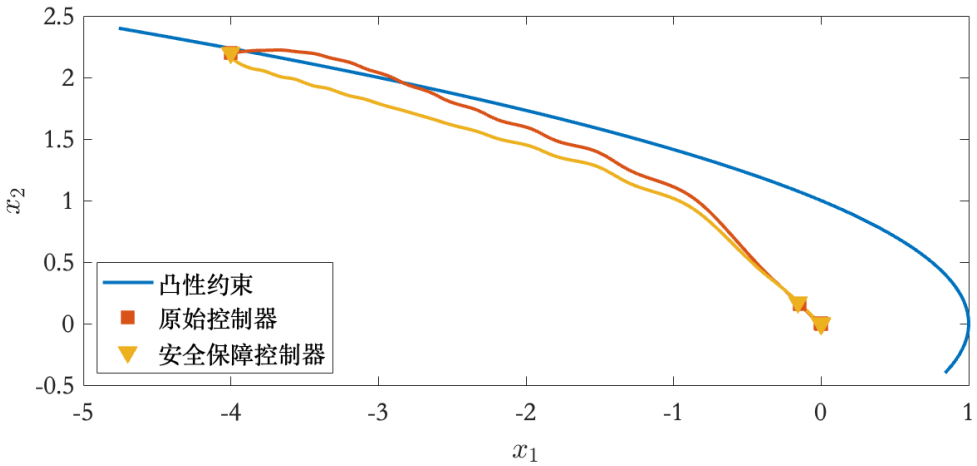

图 5-1 具有凸性安全边界的两智能体非零和博弈系统的轨迹

如图 5-2 和图 5-3 所示，智能体 1 和智能体 2 的价值函数的权重趋于[0.5000,0,1.0002]和[0.2500,0,0.5001]，分别得到的价值函数如下所示：

$$\hat{V}_1 = \begin{bmatrix} 0.5000 \\ 0 \\ 1.0002 \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} x_1^2 \\ x_1 x_2 \\ x_2^2 \end{bmatrix} \tag{5-10}$$

$$\hat{V}_2 = \begin{bmatrix} 0.2500 \\ 0 \\ 0.5001 \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} x_1^2 \\ x_1 x_2 \\ x_2^2 \end{bmatrix} \tag{5-11}$$

两个智能体分别对应的控制器如下所示：

$$\hat{u}_1 = -\frac{1}{2} R_{11}^{-1} g_1^{\mathrm{T}} \begin{bmatrix} 2x_1 & 0 \\ x_2 & x_1 \\ 0 & 2x_2 \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} 0.5000 \\ 0 \\ 1.0002 \end{bmatrix} \tag{5-12}$$

$$\hat{u}_2 = -\frac{1}{2} R_{22}^{-1} g_2^{\mathrm{T}} \begin{bmatrix} 2x_1 & 0 \\ x_2 & x_1 \\ 0 & 2x_2 \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} 0.2500 \\ 0 \\ 0.5001 \end{bmatrix} \tag{5-13}$$

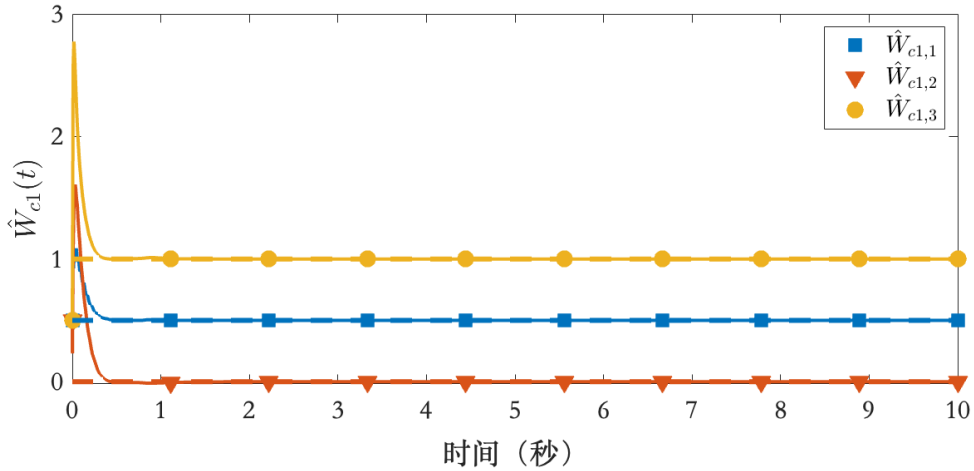

图 5-2 智能体 1 的价值函数的权重

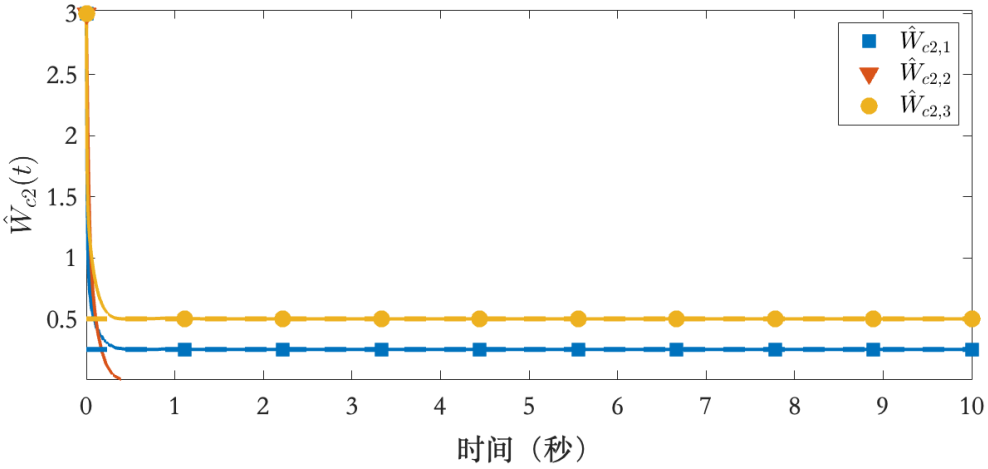

图 5-3 智能体 2 的价值函数的权重

最后，为了证明安全保障控制器的通用性，我们还设置了一个非凸的边界限制。图5-4展示了带有/不带有安全保障项的控制器的控制效果的差异。初始状态为$x_0 = [1.5,3]$，在初始阶段，两个控制器的状态轨迹是相同的。然而，当安全保障控制器接近边界时，安全保障项将状态轨迹拖离障碍函数的反向梯度方向。当离边界的距离逐渐增加时，安全保障项的影响逐渐消失，最后收敛为零。然而，原始控制器的状态轨迹直接越过了状态限制，没有任何倾向于远离边界的迹象。在该场景设置下，如果使用原始控制器，即未加入安全保障的控制补偿项，非零和博弈系统的状态轨迹突破安全边界，如图5-4中的红色状态轨迹所示，这将会对系统的安全性提出重大的挑战。

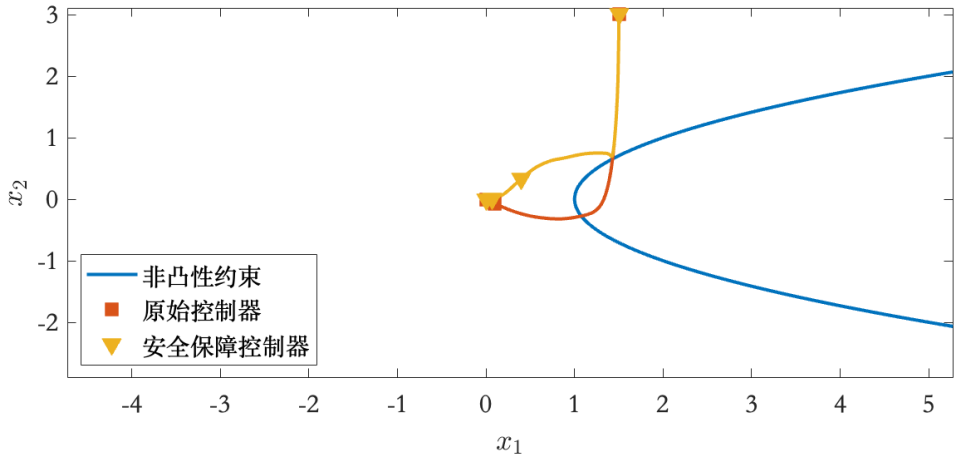

图5-4 具有非凸性安全边界的两智能体非零和博弈系统的轨迹

5.2.2 互联系统功下的保障安全性验证

为了证明安全保障控制器的通用性，我们设置了一个非凸的安全边界。学习过程中每一个子系统的单一评判网络的具体权重变化如图 5-5 和图 5-6 所示。从图中可以看出，子系统 1 和子系统 2 值函数的逼近结果分别都是收敛的，满足分布式系统的稳定性要求。故本文第四章提出的分散式控制方法首先满足基本的稳定性。其次我们再分析其特定场景下的安全保障特性。

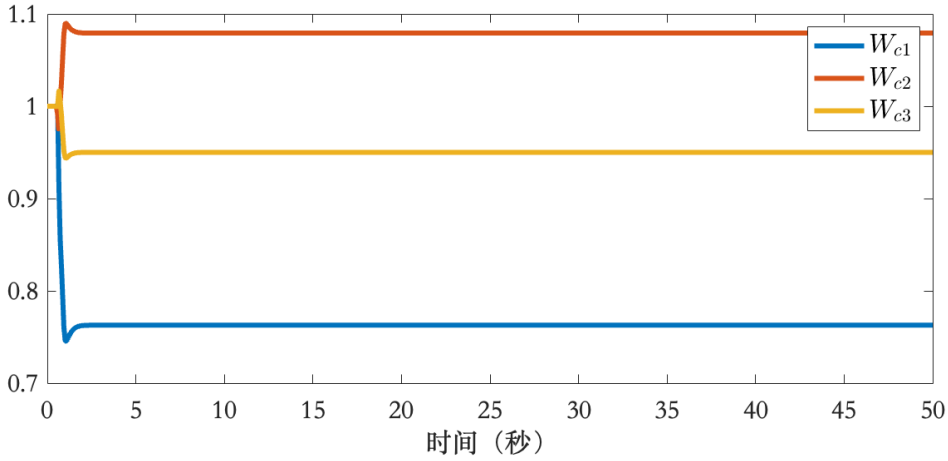

图5-5 子系统 1 评判网络的权重变化图

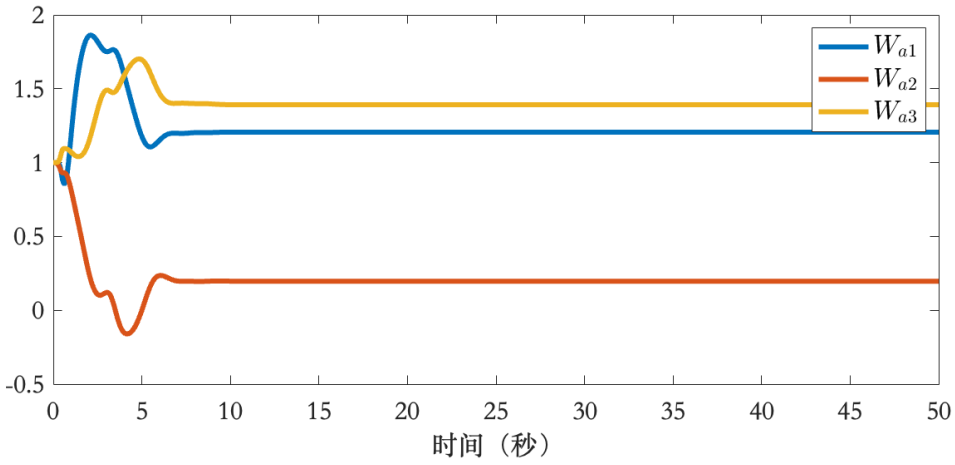

图 5-6 子系统 2 评判网络的权重变化图

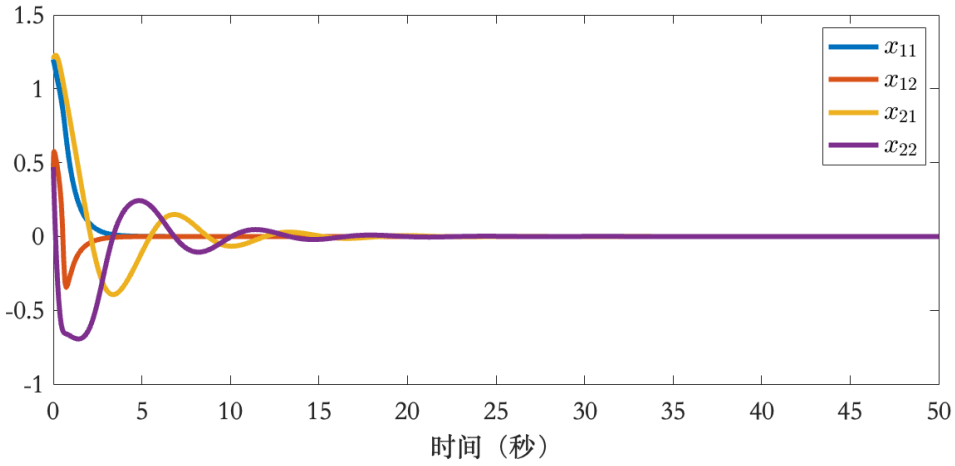

图 5-7 两个子系统的状态变化

图 5-7 展示了增加/未增加安全保障控制项的两种情况下，其子系统 1 和子系统 2 分别的状态变化曲线，可以看出，无论是否加入安全保障控制项，子系统都终将收敛。

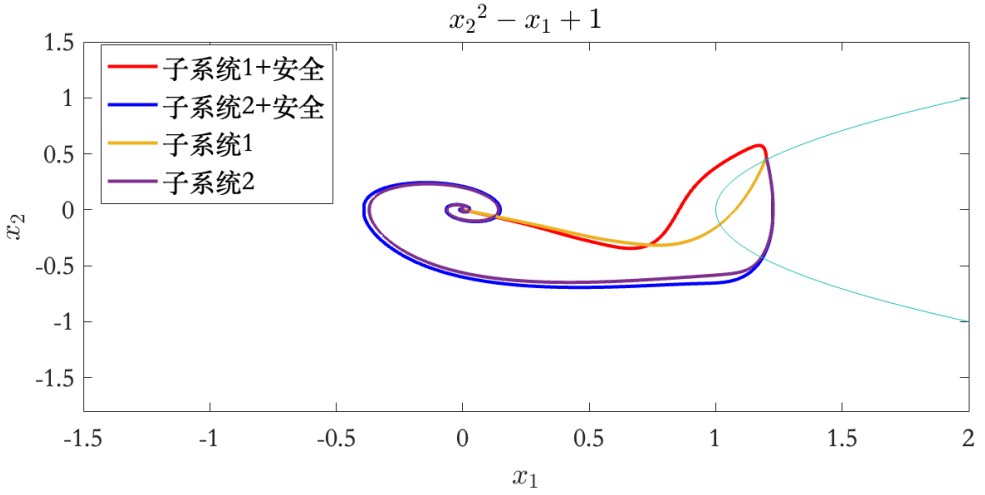

图 5-8 在非凸性安全边界集合中的子系统状态轨迹

图 5-8 为本文互联系统安全保障控制器设计的主要结果，其中浅蓝色实线代表安全边界。在图 5-8 中，对于是否增加安全保障控制器，子系统 1 的状态轨迹几乎相同。然而，对于子系统 2，当增加安全保障控制器的子系统接近边界时，安全保障项将状态轨迹拖离障碍函数的反向梯度方向。当与边界的距离逐渐增加时，安全保障项的影响就会消失。最后，状态收敛为零。与此不同的是，原始未加入安全控制项的子系统的状态轨迹直接穿过状态限制，没有任何避免与边界碰撞的迹象。

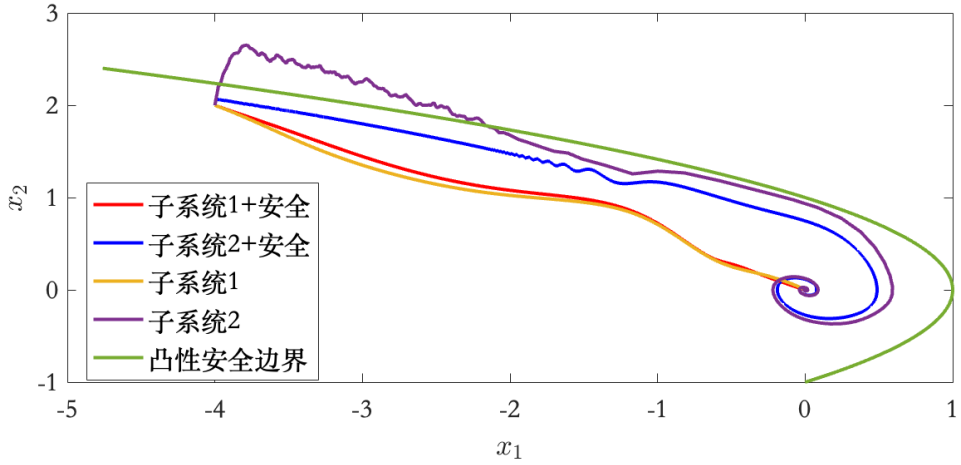

图 5-9 在凸性安全边界集合中的子系统状态轨迹

图 5-9 为安全保障控制器在凸性安全边界集合中的结果，其中的绿色实线代表安全边界。在图 5-8 中，对于是否增加安全保障控制器，子系统 1 的状态轨迹几乎相同。然而，对于子系统 2，当增加安全保障控制器的子系统接近边界时，安全保障项将状态轨迹拖离障碍函数的反向梯度方向。当与边界的距离逐渐增加时，安全保障项的影响就会消失。最后，状态收敛为零。与此不同的是，原始未加入安全控制项的子系统的状态轨迹直接穿过状态限制，没有任何避免与边界碰撞的迹象。可以看到，针对凸性边界和非凸性边界，本文设计的互联系统安全保障控制器均具有良好的保障状态变量的轨迹处于安全边界以内的效果，可以对重点安全的复杂系统形成很好的保护作用。

5.3 本章小结

本章通过数值仿真来对第三章、第四章的理论分析与设计的结果进行验证，表明在所设计的基于障碍函数的安全保障控制策略的作用下，非零和博弈系统和互联系统都能够在非凸集/凸集安全边界的情况下能够实现安全、稳定控制。仿真结果表明所设计的安全保障控制器能够确保严格避开不安全的状态区域，所设计控制器在广义的分布式多智能体中是安全有效的。

6　硬件模拟仿真验证

本章主要内容为对上文所设计的基于自适应动态规划的互联系统抗干扰策略的安全性和稳定性设置硬件模拟仿真验证，并对仿真结果进行分析。

6.1 无人机硬件仿真验证

为了验证该基于障碍函数的安全保障控制器在实际场景下的安全性，本节选取了卓翼公司飞思实验室的 F450 无人机集群作为仿真验证对象。仿真平台方面，采用 Matlab/Simulink 作为控制器的数值求解器，负责实时解算无人机上的$x-y-z$三个轴的运动轨迹控制以及俯仰角、倾角两个角度；使用 CopterSim 作为底层模型的仿真解算器，其主要功能分为两方面，一是提供固定无人机的模型，可以根据机架类型、整机质量、机架轴距、电机电池等因素构建接近实际的仿真模型；二是提供了与其他软件通讯的接口，能够与 Matlab、3Ddisplay 等软件实时接收、发送信息，实现在线控制器控制以及实时运动轨迹显示。

根据 F450 无人机的机身特性，基于横坐标x、纵坐标y、高度h和角度ψ五个变量，可以给出如下形式的单个无人机的数值模型：

$$\ddot{x}+3.9811\dot{x}+0.529635x=41.1632u_x \tag{6-1}$$

$$\ddot{y}+3.89105\dot{y}+0.529839y=41.1633u_y \tag{6-2}$$

$$\dot{h}+13.2188h=13.2244u_h \tag{6-3}$$

$$\dot{\psi}+5.43658\psi=5.45157u_\psi \tag{6-4}$$

后续仿真分为两个部分，一是单机实验，在简单场景下验证基于障碍函数的安全保障控制器，基于式（6-1）~（6-4）提供的简单动力学模型进行决策求解与控制；二是多机互联实验，两台无人机在避开固定障碍物的同时避让另外一台无人机，无人机的动力学模型选择 CopterSim，实现更贴近实际飞行环境综合影响的避障模拟实验。

图 6-1 仿真软件 RflySim3D 利用物理引擎对 F450 无人机的建模

6.2 单无人机安全保障实验

6.2.1 单无人机安全保障控制器设计

当前的状态包括横坐标x、纵坐标y、高度h，为简化后续设计流程，在此写成集中形式，用字符ξ表示。无人机的目标位置为ξ_{wp}。定义当前状态与目标状态的误差为$\tilde{\xi}_{wp} \triangleq \xi - \xi_{wp}$，当前状态与圆形障碍物中心点的误差为$\tilde{\xi}_0 \triangleq \xi - \xi_0$。

根据无人机的物理模型属性可知，无人机的动态方程为二阶方程，底层传感器-电机的控制决策求解较为复杂，故本节将从顶层决策角度出发，对无人机的路径目标设置进行规划与决策，利用位置误差设计无人机控制器。本章节的核心为验证基于障碍函数的安全保障控制器，为了简化后续的控制决策设计过程，在此给固定增益定k_1，作为未实施安全保障机制的名义控制器：

$$u_{nomial} = -k_1 \tilde{\xi}_{wp} \tag{6-5}$$

为实现单一无人机在达到目标位置的同时保障避开障碍，本小节将基于障碍函数方法设计安全保障控制器。为了研究具有通性的障碍边界，本节设置一个圆形的虚拟障碍边界和一个圆形的实际障碍边界，设圆的中心点为ξ_0，半径分别为r和r'。实际边界为硬性的边界约束，无人机不能触碰实际边界，一旦触碰实际边界，无人机将直接陷入危险状态，例如撞击建筑物或者其他飞行器。而虚拟障碍边界是非硬性的边界约束，在控制决策的过程中，需要提前考虑可能发生的危险，预留一定的安全距离，防止一定小概率事件的出现而导致的安全事故。为了更好的保障对无人机控制决策的安全性，本节以虚拟边界作为障碍函数的约束，设置障碍函数的边界函数h为$h = \left(\|\tilde{\xi}_o\| - r'\right)$，故可得对应的障碍函数为：

$$b(\xi) = \frac{1}{\left(\|\tilde{\xi}_o\| - r'\right)^2} \tag{6-6}$$

根据第四章互联系统下基于障碍函数的安全保障控制器设计，针对，我们设计如下形式的基于障碍函数的安全保障项：

$$u_b = k_2 \frac{1}{\left(\|\tilde{\xi}_o\| - r'\right)^2} \frac{\tilde{\xi}_0}{\|\tilde{\xi}_0\|} \tag{6-7}$$

接下来给出关于控制输入的求解过程，为了实现安全保障的同时保证系统的渐进稳定性，本文设计给出如下形式的李雅普诺夫函数：

$$V_2 = \frac{k_1}{2} \tilde{\xi}_{wp}^T \tilde{\xi}_{wp} + k_2 \frac{1}{\|\tilde{\xi}_o\| - r'} \tag{6-8}$$

对该李雅普诺夫函数进行求导可得：

$$\dot{V} = k_1 \tilde{\xi}_{wp}^T \bar{u} - k_2 \frac{1}{\left(\|\tilde{\xi}_o\| - r'\right)^2} \frac{\tilde{\xi}_o^T}{\|\tilde{\xi}_o\|} \bar{u} = \left(a\tilde{\xi}_{wp} - b\tilde{\xi}_o\right)^T \bar{u} \tag{6-9}$$

为了使得李雅普诺夫函数为负定，需要整体的控制输入满足如下公式表示的形式：

$$\bar{u} = -a_0\left(a\tilde{\xi}_{wp} - b\tilde{\xi}_o\right) \tag{6-10}$$

此时可以保证如下的不等式成立：

$$\dot{V} = -\left(a\tilde{\xi}_{wp} - b\tilde{\xi}_o\right)^T a_0\left(a\tilde{\xi}_{wp} - b\tilde{\xi}_o\right) \le 0 \tag{6-11}$$

故该安全保障控制策略可以保障避开障碍物的硬性约束边界：

$$\|\tilde{p}_0\| > r_0 \tag{6-12}$$

并且当无人机抵达设定的目标点ξ_{origin}时，控制输入消失，即该控制策略不会使无人机陷入除$V = 0$以外的其他平衡点：

$$a\left(\xi_{origin} - \xi_{wp}\right) + b\left(\xi_{origin} - \xi_0\right) = 0_{2\times 1}$$

实际控制器的输出大小受到一定限制，由于该限制一般为平滑连续性变化，故可以使用饱和函数$sat(\cdot)$表示，单个 F450 无人机的安全保障控制器可以表示为：

$$u = -sat\left(a_0 \times \left(a\tilde{\xi}_{wp} - b\tilde{\xi}_0\right)\right) \tag{6-13}$$

6.2.2 单无人机的安全保障仿真实验

本小节实验采用飞行界面软件 FlightGear 进行实时展示，如图 6-2 所示。利用 Simulink 进行底层模型求解与控制决策，无人机的动态模型选用式（6-1）~（6-4）。

图 6-2　飞行界面 FlightGear 展示 F450 无人机的运动

具体参数选取上，我们选择$a_0 = 1$，$k_0 = 1$，$k_1 = 0.5$，$k_2 = 10$，$r = 2m$，$r' = 4m$。仿真时长为 20 秒，时长足够无人机接近最终目标位置。控制器的饱和量选取为$u_{x,max} = 2$，$u_{y,max} = 2$，$u_{z,max} = 2$和$u_{\psi,max} = 2$。

单个 F450 多旋翼无人机的仿真避障实验如下图 6-3 所示，从图中可以看到，无人机从初始状态位置$(0,0)$出发，最终到达目标位置$(-6m, 25m)$，并且完成了对实际的固定障碍物的避障任务。

在无人机的整体控制决策过程中，如图 6-3 所示，我们可以看到无人机初始化后，即刚开始出发的小段时间内，无人机是沿着初始位置与终点位置的连线前进，但当无人机接近固定障碍物时，无人机的期望位置轨迹与实际位置轨迹都发生了一定的偏移，

轨迹朝着远离障碍物的方向发生移动，保持轨迹不与硬性障碍约束以及软障碍约束发生接触，最后到达目标终点。

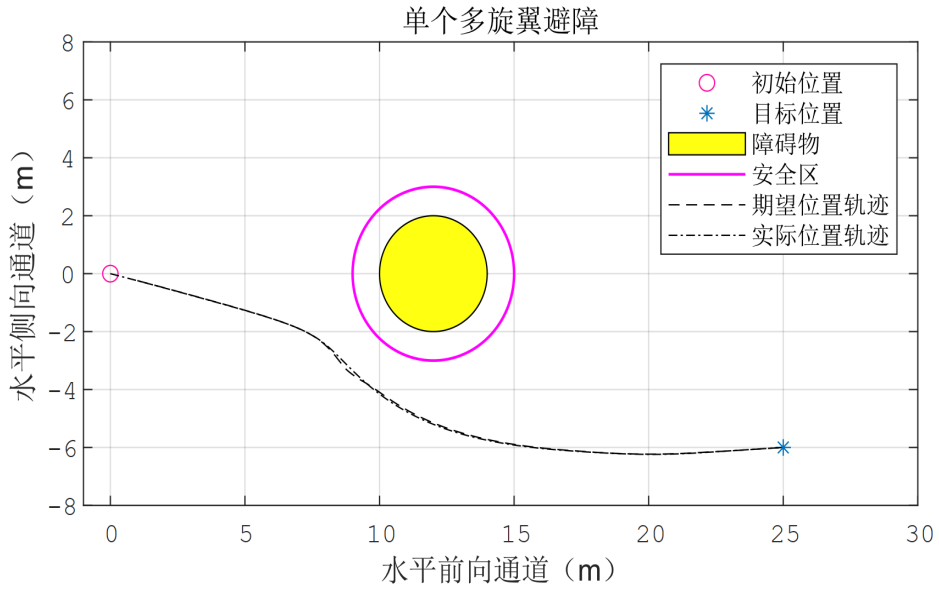

图 6-3 单 F450 无人机避障轨迹

无人机在 y 轴上的实际与期望轨迹，如图 6-4 所示，为单方向朝着目标点前进。

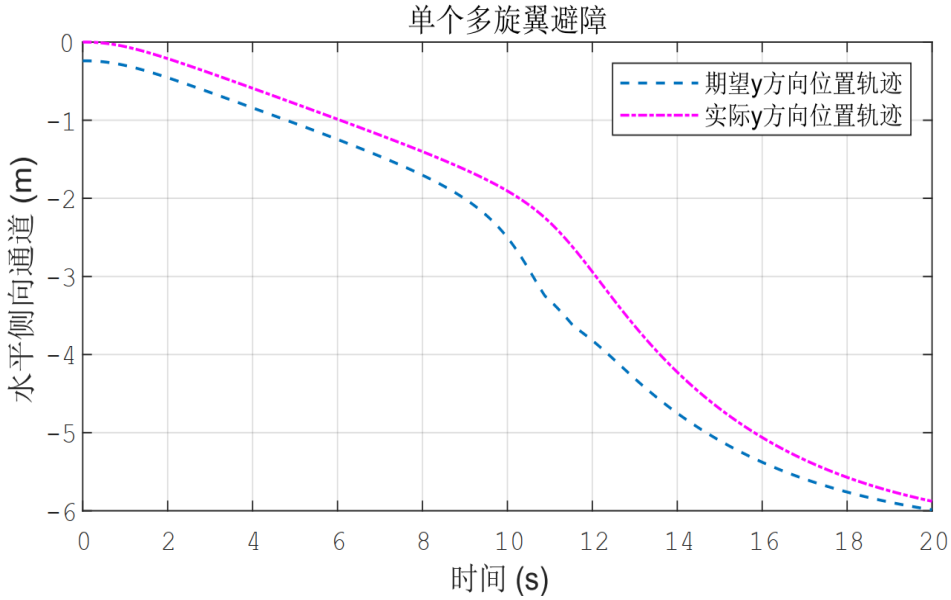

图 6-4 单 F450 无人机避障的在 y 轴方向上的轨迹

6.3 多无人机互联系统的安全保障控制器设计

6.3.1 多无人机安全保障控制器设计

为实现多无人机的协作控制下的安全保障控制，下面将针对两架 F450 无人机的情况设计安全保障控制器，实现两种形式的安全保障行为：1、避开硬性约束的障碍物；2、避免与其他无人机等飞行器碰撞。

第一种避障的安全保障控制项已经在单一 F450 无人机的安全保障控制器设计中提出，为了实现第二种安全保障情况，即实现避免与其他 F450 无人机的碰撞，本小节根据基于障碍函数的安全保障控制器，首先定义当前状态下无人机i与无人机j之间的误差为$\tilde{\xi}_{ij} \triangleq \xi_i - \xi_j$，无人机$i$与障碍中心之间的误差为$\tilde{\xi}_{i,o} \triangleq \xi_i - \xi_o$。为简化控制决策建立过程，突出安全保障控制器的设计，在此给固定增益定k_1，给出未保障安全的控制器：

$$u_{nomial,i} = -k_1 \tilde{\xi}_{wp,i} \tag{6-14}$$

为了避免无人机i与无人机j互相碰撞，定义一个圆形的硬约束，该约束在本仿真实例中等价于无人机的碰撞区域，若进入这个区域，将与该无人机发生碰撞，设置该约束的半径为r_{ij}。由于无人机的安全保障控制实际需要一定的余量，在此定义一个软约束的障碍边界，该约束随着无人机的运动同时相对地面移动，即使进入该软约束区域但不进入硬性约束区域，无人机的有一定概率发生碰撞事故，设置该约束的半径为r_{ij}'。选择如下形式的固定障碍物边界函数：

$$h_o(\xi) = \|\tilde{\xi}_{ij}\| - r_{ij}', i,j = 1,\cdots,n \tag{6-15}$$

给出编号为i的无人机与其他无人机之间的障碍函数，其具体形式如下公式所示：

$$b_{ij}(\xi) = \frac{1}{(\|\tilde{\xi}_{ij}\| - r_{ij\prime})^2}, i,j = 1,\cdots,n \tag{6-16}$$

其中n为需要与编号为i的无人机进行避障决策无人机的总数。

根据第四章互联系统的安全保障控制器设计的内容以及上一节关于单个无人机的安全保障控制器的设计内容，本节提出如下形式的基于障碍函数的安全保障控制项：

$$u_{b,ij} = k_2 \frac{1}{(\|\tilde{\xi}_{ij}\| - r_{o,i\prime})^2} \frac{\tilde{\xi}_{ij}}{\|\tilde{\xi}_{ij}\|}, i,j = 1,\cdots,n \tag{6-17}$$

同样的，如同上文关于无人机i与无人机j之间的障碍约束相类似的定义，在此定义无人机i与固定障碍物之间的硬性约束与软约束皆为圆形边界函数的约束，其半径分别为$r_{o,i}$和$r_{o,i}'$。为了满足本文第二章中相关障碍函数定义，在此选择如下形式的边界函数：

$$h_{o,i}(\xi) = \|\tilde{\xi}_{i,o}\| - r_{o,i}', i = 1,\cdots,n \tag{6-18}$$

给出编号为i的无人机与固定障碍物之间的障碍函数，其具体形式如下公式所示：

$$b_{o,i}(\xi) = \frac{1}{(\|\tilde{\xi}_{i,o}\| - r_{o,i\prime})^2}, i = 1,\cdots,n \tag{6-19}$$

其中n为需要与编号为i的无人机进行避障决策无人机的总数。根据第四章互联系统的安全保障控制器设计的内容以及上一节关于单个无人机的安全保障控制器的设计内容，本节提出如下形式的基于障碍函数的安全保障控制项：

$$u_{b,io} = k_2 \frac{1}{(\|\tilde{\xi}_{i,o}\| - r_{o,i\prime})^2} \frac{\tilde{\xi}_{i,o}}{\|\tilde{\xi}_{i,o}\|}, i = 1,\cdots,n \tag{6-20}$$

接下来给出关于控制输入的求解过程，为了实现安全保障的同时保证系统的渐进稳

定性，本文设计给出如下形式的李雅普诺夫函数：

$$V_i = \frac{k_1}{2}\tilde{\xi}_{wp,i}^T\tilde{\xi}_{wp,i} + k_2\frac{1}{\|\tilde{\xi}_{ij}\|-r_{ij'}} + k_2\frac{1}{\|\tilde{\xi}_{i,o}\|-r_{o,i'}} \tag{6-21}$$

对该李雅普诺夫函数进行求导可得：

$$\dot{V}_{2,i} = k_1\tilde{\xi}_{wp,i}^T\bar{u}_i - k_2\frac{1}{(\|\tilde{\xi}_{i,o}\|-r_{o,i'})^2}\frac{\tilde{\xi}_{i,o}}{\|\tilde{\xi}_{i,o}\|}\bar{u}_i - k_2\sum_{j=1}^{n}\frac{1}{(\|\tilde{\xi}_{ij}\|-r_{ij'})^2}\frac{\tilde{\xi}_{ij}}{\|\tilde{\xi}_{ij}\|}\bar{u}_i \tag{6-22}$$

合并并且简写可得：

$$\dot{V}_{2,i} = \left(a\tilde{\xi}_{wp,i} - b_{o,i}\tilde{\xi}_{o,i} - \sum_{j=1}^{n} b_{ij}\tilde{\xi}_{ij}\right)^T\bar{u}_i \tag{6-23}$$

为了使得李雅普诺夫函数为负定，需要整体控制输入满足如下公式表示的形式：

$$\bar{u} = -a_0\left(a\tilde{\xi}_{wp,i} - b_{o,i}\tilde{\xi}_{o,i} - \sum_{j=1}^{n} b_{ij}\tilde{\xi}_{ij}\right) \tag{6-24}$$

此时可以保证如下的不等式成立：

$$-\left(a\tilde{\xi}_{wp,i} - b_{o,i}\tilde{\xi}_{o,i} - \sum b_{ij}\tilde{\xi}_{ij}\right)^T a_0\left(a\tilde{\xi}_{wp,i} - b_{o,i}\tilde{\xi}_{o,i} - \sum b_{ij}\tilde{\xi}_{ij}\right) \le 0 \tag{6-25}$$

故该安全保障控制策略可以保障避开障碍物以及其他无人机的硬性约束边界：

$$\|\tilde{\xi}_{ij}\| > r_{ij} \tag{6-26}$$

$$\|\tilde{\xi}_{i,o}\| > r_{0,i} \tag{6-27}$$

并且当无人机抵达设定的目标点ξ_{origin}时，控制输入消失，即该控制策略不会使无人机陷入除$v = 0$以外的其他平衡点：

$$a\left(\xi_{origin} - \xi_{wp,i}\right) + b_{o,i}\left(\xi_{origin} - \xi_{0,i}\right) + \sum b_{ij}\left(\xi_{origin} - \tilde{\xi}_{ij}\right) = 0_{2\times1} \tag{6-28}$$

实际控制器的输出大小受到一定限制，由于该限制一般为平滑连续性变化，可以使用饱和函数$sat(\cdot)$表示，故针对具有多个 F450 无人机的互联系统，可以将其安全保障控制器由单一无人机形式进一步改进为：

$$u_i = -sat\left(a_0 \times \left(a\tilde{\xi}_{wp,i} - b_{o,i}\tilde{\xi}_{o,i} - \sum b_{ij}\tilde{\xi}_{ij}\right)\right) \tag{6-29}$$

6.3.2 多无人机的安全保障仿真实验

本实验选取了两台 F450 无人机，作为多无人机实验的主要控制决策对象，并且设置了一个位于两个无人机中央的圆形固定障碍物，无人机 1 与无人机 2 之间的硬性约束为$r_{ij} = 2m$，软约束为$r'_{ij} = 4m$。固定障碍物的硬性约束半径为$r_{o,i} = 2m$，软约束$r'_{o,i} = 4m$。无人机 1 的初始位置为$(0, 0m)$，无人机 2 的初始位置为$(0, 30m)$，固定障碍物的中心点在$(0, 18m)$。

在控制器参数选取上，给定$a_0 = 1$，$a = 0.1$。在增益选取上，本节选择$k_0 = 2$，$k_1 = 2$，$k_2 = 10$。实验的仿真时长选取为 610 秒，足够两台无人机到达终点位置并且稳定。两台无人机的初始位置之间的距离为 30m，两台无人机的其他特性均一致。

在有固定障碍物的场景下，两台多旋翼 F450 无人机具体避障的实验如下面两张图

所示。图 6-5 为全局避障轨迹展示，在基于障碍函数的安全保障控制器的作用下，两台无人机都避免了对固定障碍的碰撞，并且两台无人机在互不碰撞的前提下，实现了互换位置，即无人机 1 最后到达无人机 2 的初始位置，无人机 2 最后到达无人机 1 的初始位置，两台无人机的路径选择较优，并未出现大幅度偏差。

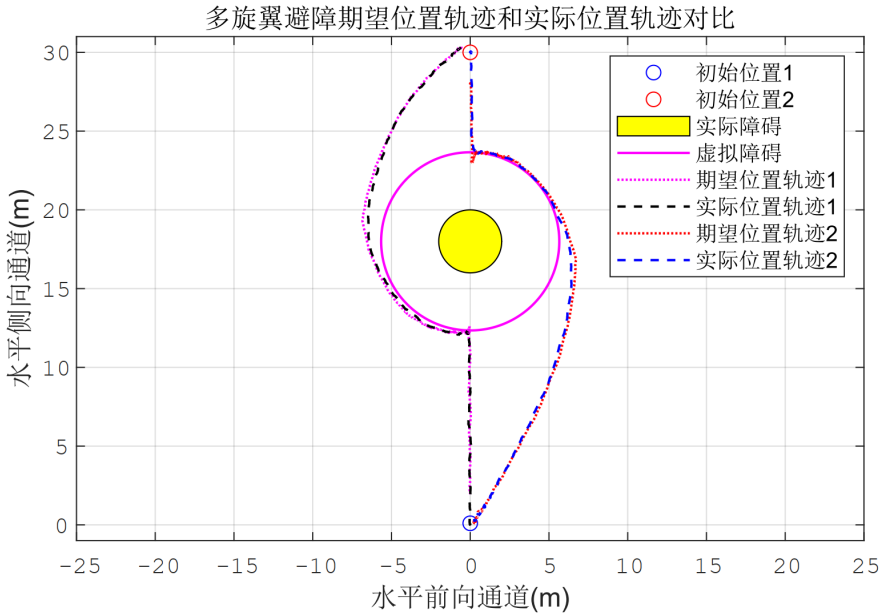

图 6-5 两架 F450 无人机的避障轨迹

为展示基于障碍函数的安全保障控制器的效果，图 6-6 展示了在软约束边界附近的无人机轨迹，其中黑色虚线为编号为 1 号的 F450 无人机的实际位置轨迹，粉色细点线为期望决策轨迹。可以看到，期望轨迹略有超过软约束边界，但是实际轨迹未超过软约束边界，保障了无人机飞行过程中的安全。

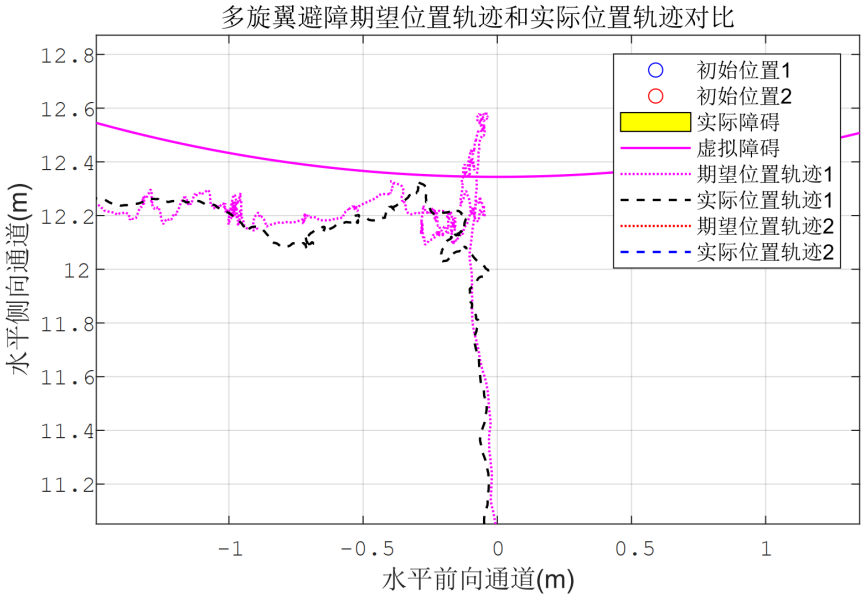

图 6-6 无人机 1 在接近软约束边界时的局部轨迹

利用 QGroudContract 软件作为地面站，对无人机仿真进行实时定位跟踪，无人机 1 和无人机 2 在地面站的模拟 GPS 跟踪信号的轨迹分别如图 6-7 和图 6-8 所示。这两个无人机的运动轨迹与图 6-5 全局避障中的两天实际轨迹对应，模拟了实际无人机在飞行控制时反馈的飞行动态。

图 6-7 无人机 1 在地面站软件中显示的轨迹

图 6-8 无人机 2 在地面站软件中显示的轨迹

具体的实验仿真过程同样在 Rflysim3D 中实时显示，本节选取了三个时刻的场景进行展示，图 6-9 为初始时刻的场景，两台无人机相距 30m，障碍物处于两台无人机的连线上，并且无人机 1 距离障碍物 18m，无人机 2 距离障碍物 12m。

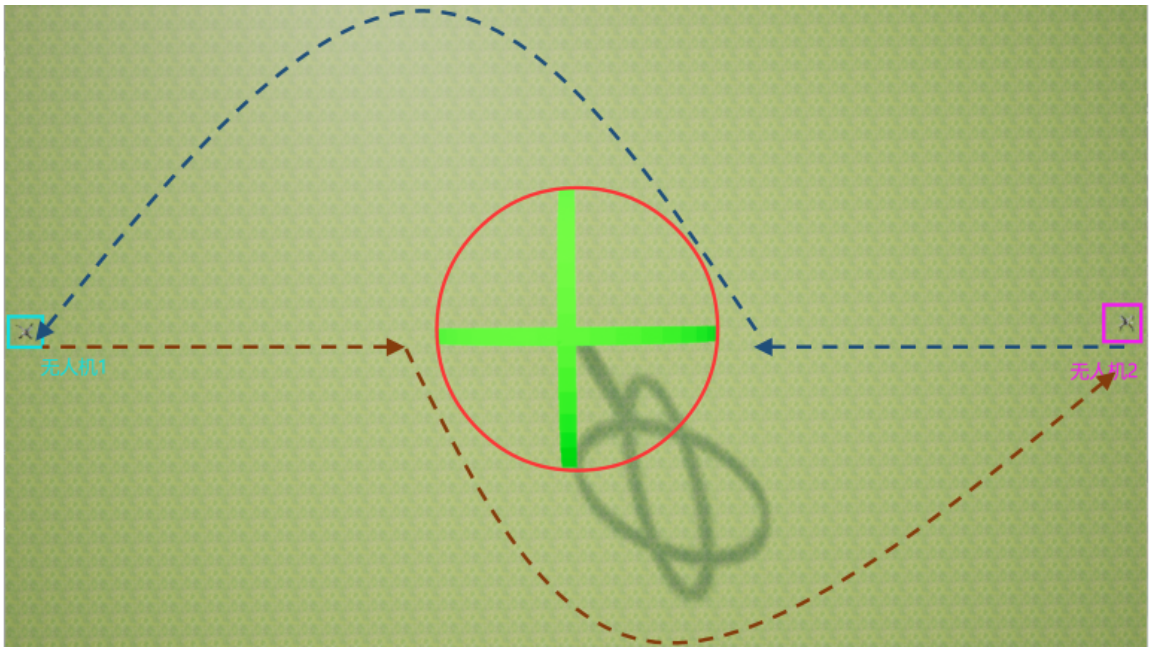

图 6-9 两台无人机避障在 RflySim3D 中的初始场景

图 6-10 展示无人机 1 和无人机 2 正在绕过固定障碍物，在整个过程中未碰撞障碍物，无人机首先以趋近于直线的路径逼近障碍物，然后因为障碍的过于接近，导致无人机受到向障碍物两侧方向的控制作用，使得无人机往障碍物的两侧飞行，从而实现避开障碍物的效果。

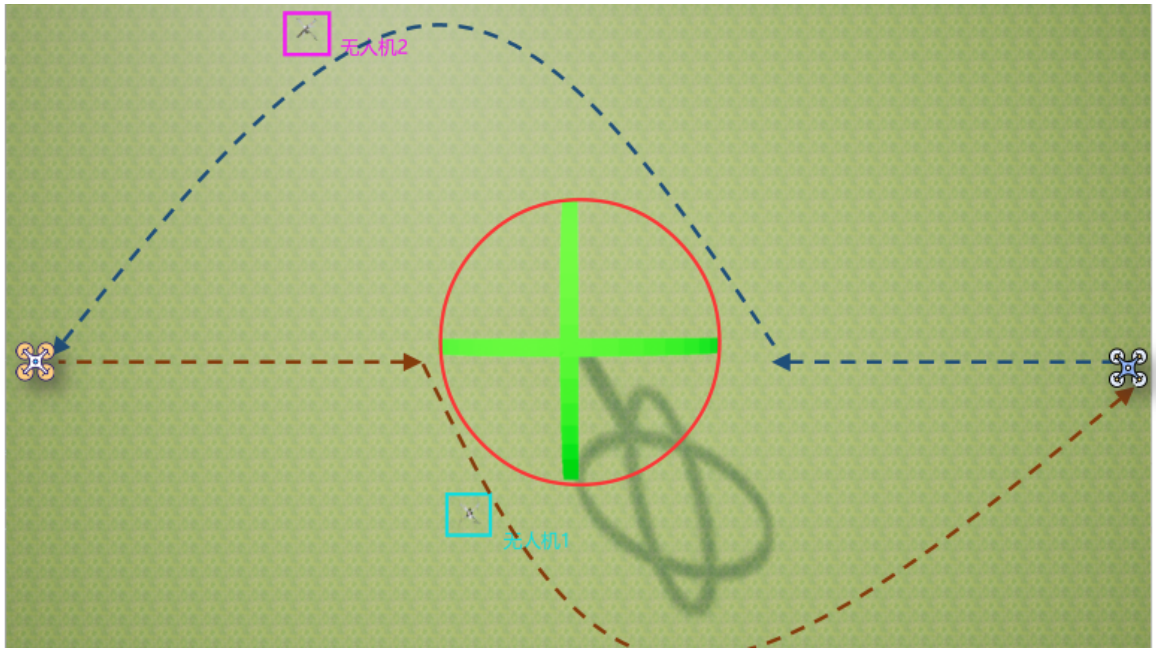

图 6-10 两台无人机避障在 RflySim3D 中的绕过障碍物

图 6-11 为两架无人机到达终点时刻的场景，此时无人机 1 达到无人机 2 的初始位置，无人机 2 到达无人机 1 的初始位置，两台无人机实现位置互换的同时避免相互碰撞以及与固定障碍物的碰撞。

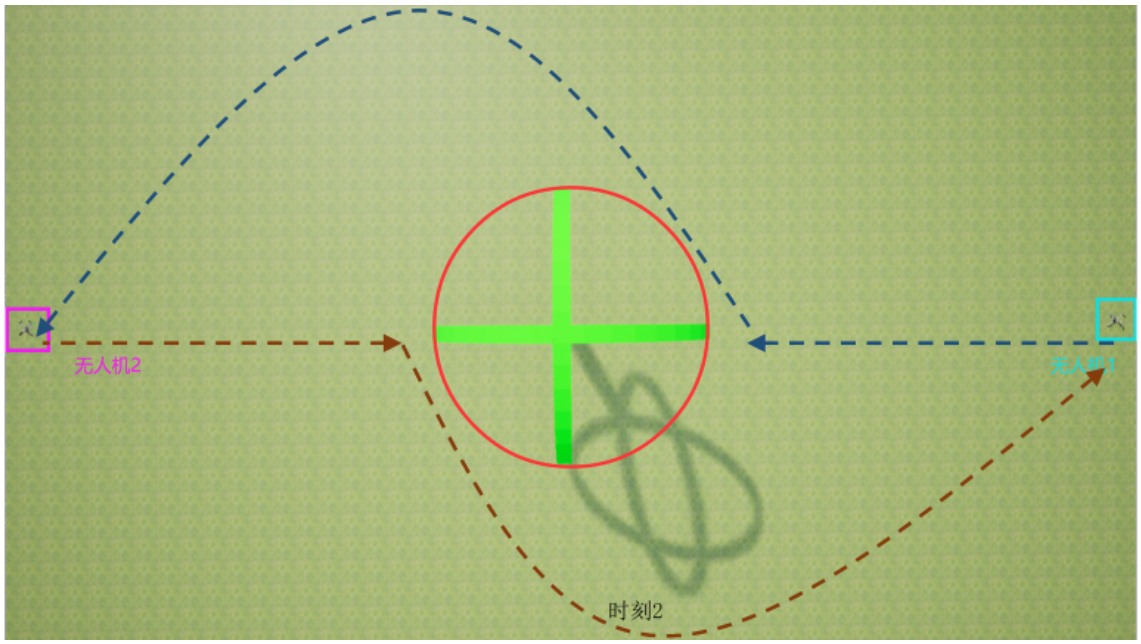

图 6-11 两台无人机避障在 RflySim3D 中的最终场景

两台无人机在 x 轴、y 轴和 z 轴上的轨迹变化分别如图 6-12、6-13 和 6-14 所示，可以看到在 x 轴方向上，两台无人机方向相反但是保持单向前进。在 y 轴方向上，无人机 1 和 2 都有通过离开$y = 0$以避障的操作，但是最终都将回到$y = 0$附近。在 z 轴方向上，无人机 1、2 这两台无人机在初始化之后上升到$z = 5m$的高度并保持。

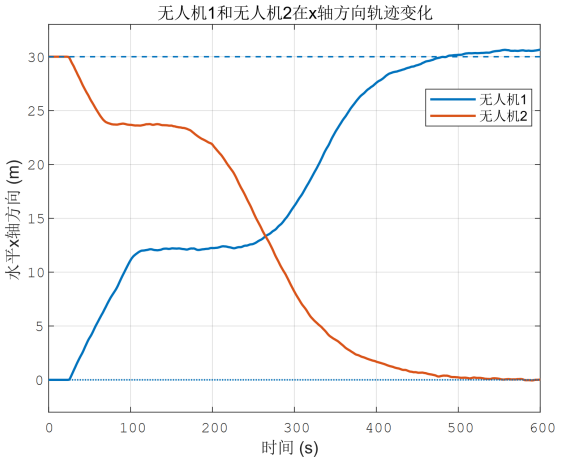

图 6-12 无人机 1 和 2 在 x 轴方向上的轨迹

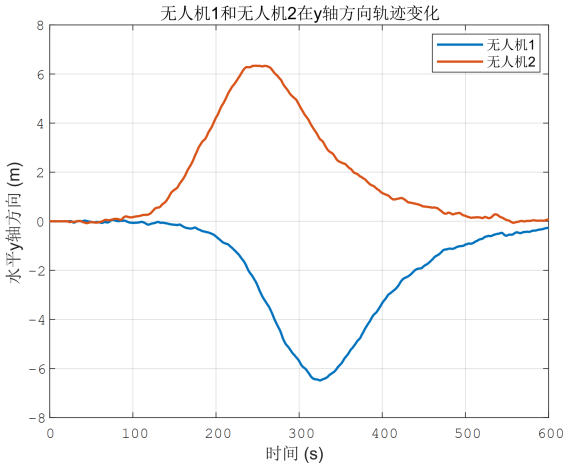

图 6-13 无人机 1 和 2 在 y 轴方向上的轨迹

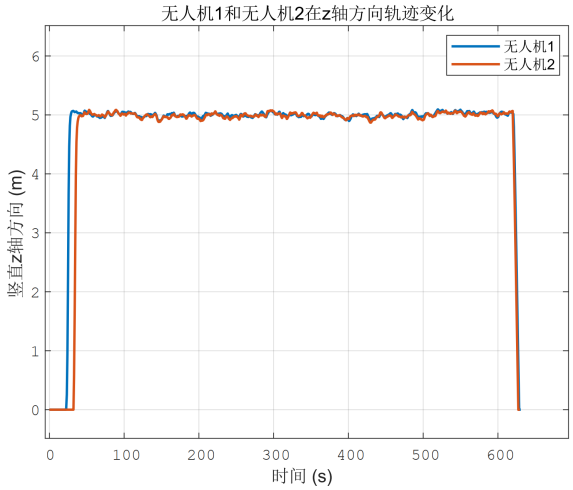

图 6-14 无人机 1 和 2 在 z 轴方向上的轨迹

6.4 本章小结

本章通过硬件模拟仿真来对第三章、第四章的理论分析与设计的结果进行验证，表明在所设计的基于障碍函数的安全保障控制策略的作用下，单个无人机系统和多个无人机的互联系统都能够在凸集安全边界下能够实现安全保障。仿真结果表明所设计的基于障碍函数的安全保障控制项能够确保避开不安全的区域，所设计安全策略在实际应用场景中是安全有效的。

7 结论与展望

7.1 结论

本文主要设计了一种针对包括互联系统和非零和博弈等复杂系统的安全保障控制器，不仅能够满足凸集合的安全边界约束要求，还可以满足传统安全控制器无法满足的非凸集合安全约束要求。同时安全保障控制补偿项可以和任意名义控制器结合，灵活适用各种系统，对系统参数的变化具有良好的抗干扰性，对安全边界的约束具有一定的自适应性，本文的主要工作分为以下三部分。

1）本文分别针对复杂系统中的非零和博弈系统以及互联系统的特性进行安全保障控制器设计。这种保障安全的控制器设计主要分为三步。首先设计一种基于障碍函数梯度的安全保障的控制器补偿项。其次根据自适应动态规划理论建立名义控制器，非零和博弈系统中需要满足多个智能体非零和博弈下的纳什均衡，而互联系统中则需要利用分布式稳定性理论设计分布式稳定控制器。最后利用并行学习的更新律对控制器逼近的神经网络进行迭代，借助单一评判网络方法，在得到适用性广泛的安全保障控制器的同时，减少了神经网络逼近参数迭代过程中一半的计算量。

2）本文借助李雅普诺夫稳定性理论分别证明了非零和博弈与互联系统在安全保障控制器作用下的稳定性，并且证明得到并行学习更新律的参数最终一致有界。数值仿真实验对互联系统以及非零和博弈的理论分析结果进行验证，表明在所设计的基于障碍函数的安全保障控制器的作用下，复杂系统能够在非凸性以及凸性安全约束下保障状态变量的安全。

3）为了进一步验证基于障碍函数的安全保障策略在实际应用场景下的效果，本文设计了无人机的硬件模拟实验对安全保障策略进行验证，表明在所设计的安全策略作用下，无人机复杂系统能够在避开固定障碍物以及无人机互相避让下保障安全。

7.2 展望

本文的研究仍有可以提高的部分，后续进一步的研究可以有以下三个方向：

1)本文所设计的保障安全控制器在神经网络的选取上,选择了单一评判网络方法，其计算负担较低，但是对于特定复杂场景的逼近性能可能存在不足，需要继续研究如何结合最新的深度神经网络技术，即深度学习方法。

2）本文关于性能指标的选取可以进一步拓展，考虑安全边界与状态变量实时距离对性能指标的影响。

3）在后续的研究中，可针对安全保障补偿器设计触发机制，实现名义控制与安全保障控制的策略切换。并且由于硬件仿真实验对控制器设计的要求较高，后续研究中可以针对强化学习策略对硬件部分进行改进。

致　谢

在我完成本科毕业设计之际，我想感谢身边的每一个人，相遇即是缘分。

首先，我要由衷地感谢我的导师曹晖教授。自接触科研以来，我一直在曹老师的关怀与支持下成长。感谢您在整个毕业设计期间给我的指导和支持。您的鼓励和指导帮助我克服了科研路上许多的困难。

感谢薛霜思师兄，在我步入科研的道路上给予了我极大的帮助，感谢薛师兄在这一年以来对我的关心帮助与指导，帮助我渡过了重重难关，并且让我更加深入地理解了我的研究课题。

其次我还要感谢我的同学、同门以及舍友，感谢周围帮助过我、理解关心过我的所有人，感谢我的朋友们。

最后，我还有感谢我的父母家人，你们是我最幸福的港湾与最坚实的后盾，有你们的呵护，我才能长大成人。

毕设的完成是一段充实难忘的回忆，在此，我再次向所有支持、帮助和关心过我的人表示由衷的感谢。

参考文献

[1] 李金娜, 程薇燃. 基于强化学习的数据驱动多智能体系统最优一致性综述[J]. 智能科学与技术学报, 2020, 2(4): 327-340.

[2] 赵冬斌, 刘德荣, 易建强. 基于自适应动态规划的城市交通信号优化控制方法综述[J]. 自动化学报, 2009, 35(6): 676-681.

[3] 魏文军, 尉晶波. 拓扑切换下 IT2 T-S 模糊非线性多智能体系统全局逆最优控制[J]. 控制理论与应用, 2022, 39(10): 1985-1994.

[4] 蒋伟进, 陈君鹏. 移动群智感知中基于任务质量的多任务分发参与者选择[J]. 控制与决策, 2022, 37(10): 2667-2676.

[5] 王睿, 孙秋野, 张化光. 微电网的电流均衡/电压恢复自适应动态规划策略研究[J]. 自动化学报, 2022, 48(2): 479-491.

[6] 王敏, 黄龙旺, 杨辰光. 基于事件触发的离散 MIMO 系统自适应评判容错控制[J]. 自动化学报, 2022, 48(5): 1234-1245.

[7] Qu B, Wang Z, Shen B, et al. Event-Based Joint State and Unknown Input Estimation for Energy Networks: Handling Multi-Machine Power Grids[J]. IEEE Transactions on Network Science and Engineering, 2023, 10(1): 253-264.

[8] Wang D, Fang X, Wan Y, et al. Distributed Optimization Algorithms for MASs With Network Attacks: From Continuous-Time to Event-Triggered Communication[J]. IEEE Transactions on Network Science and Engineering, 2022, 9(5): 3332-3344.

[9] Zhang X M, Han Q L, Ge X, et al. Networked control systems: a survey of trends and techniques[J]. IEEE/CAA Journal of Automatica Sinica, 2020, 7(1): 1-17.

[10] Zhang H, Luo G, Tian Y, et al. A Virtual-Real Interaction Approach to Object Instance Segmentation in Traffic Scenes[J]. IEEE Transactions on Intelligent Transportation Systems, 2021, 22(2): 863-875.

[11] Liu D, Wang D, Li H. Decentralized Stabilization for a Class of Continuous-Time Nonlinear Interconnected Systems Using Online Learning Optimal Control Approach[J]. IEEE Transactions on Neural Networks and Learning Systems, 2014, 25(2): 418-428.

[12] Qu Q, Zhang H, Feng T, et al. Decentralized adaptive tracking control scheme for nonlinear large-scale interconnected systems via adaptive dynamic programming[J]. Neurocomputing, 2017, 225: 1-10.

[13] Narayanan V, Sahoo A, Jagannathan S, et al. Approximate Optimal Distributed Control of Nonlinear Interconnected Systems Using Event-Triggered Nonzero-Sum Games[J]. IEEE Transactions on Neural Networks and Learning Systems, 2019, 30(5): 1512-1522.

[14] Yang Y, Ding D W, Xiong H, et al. Online barrier-actor-critic learning for H∞ control with full-state constraints and input saturation[J]. Journal of the Franklin Institute, 2020, 357(6): 3316-3344.

[15] Vamvoudakis K G, Lewis F L. Multi-player non-zero-sum games: Online adaptive learning solution of coupled Hamilton–Jacobi equations[J]. Automatica, 2011, 47(8): 1556-1569.

[16] Johnson M, Kamalapurkar R, Bhasin S, et al. Approximate N-Player Nonzero-Sum Game Solution for an Uncertain Continuous Nonlinear System[J]. IEEE Transactions on Neural Networks and Learning Systems, 2015, 26(8): 1645-1658.

[17] Kamalapurkar R, Klotz J R, Dixon W E. Concurrent learning-based approximate feedback-Nash equilibrium solution of N-player nonzero-sum differential games[J]. IEEE/CAA Journal of Automatica Sinica, 2014, 1(3): 239-247.

[18] Vamvoudakis K G, Lewis F L. Online actor–critic algorithm to solve the continuous-time infinite

horizon optimal control problem[J]. Automatica, 2010, 46(5): 878-888.
[19] Liu S, Liu L, Yu Z. Safe reinforcement learning for discrete-time fully cooperative games with partial state and control constraints using control barrier functions[J]. Neurocomputing, 2023, 517: 118-132.
[20] Zhang Q, Zhao D, Zhu Y. Event-Triggered H∞ Control for Continuous-Time Nonlinear System via Concurrent Learning[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2017, 47(7): 1071-1081.
[21] 张化光, 张欣, 罗艳红, 等. 自适应动态规划综述[J]. 自动化学报, 2013, 39(4): 303-311.
[22] 王鼎. 基于学习的鲁棒自适应评判控制研究进展[J]. 自动化学报, 2019, 45(06): 1031-1043.
[23] Yang Y, Yin Y, He W, et al. Safety-Aware Reinforcement Learning Framework with an Actor-Critic-Barrier Structure[C]//2019 American Control Conference (ACC). 2019: 2352-2358.
[24] Marvi Z, Kiumarsi B. Safe reinforcement learning: A control barrier function optimization approach[J]. International Journal of Robust and Nonlinear Control, 2021, 31(6): 1923-1940.
[25] Cohen M H, Belta C. Safe exploration in model-based reinforcement learning using control barrier functions[J]. Automatica, 2023, 147: 110684.
[26] Xu J, Wang J, Rao J, et al. Twin Deterministic Policy Gradient Adaptive Dynamic Programming for Optimal Control of Affine Nonlinear Discrete-time Systems[J]. International Journal of Control, Automation and Systems, 2022, 20(9): 3098-3109.
[27] Zhou Y, Vamvoudakis K G, Haddad W M, et al. A Secure Control Learning Framework for Cyber-Physical Systems Under Sensor and Actuator Attacks[J]. IEEE Transactions on Cybernetics, 2021, 51(9): 4648-4660.
[28] Greene M L, Deptula P, Nivison S, et al. Sparse Learning-Based Approximate Dynamic Programming With Barrier Constraints[J]. IEEE Control Systems Letters, 2020, 4(3): 743-748.
[29] Chowdhary G, Yucelen T, Mühlegg M, et al. Concurrent learning adaptive control of linear systems with exponentially convergent bounds[J]. International Journal of Adaptive Control and Signal Processing, 2013, 27(4): 280-301.
[30] Mahmud S M N, Nivison S A, Bell Z I, et al. Safe Model-Based Reinforcement Learning for Systems With Parametric Uncertainties[J]. Frontiers in Robotics and AI, 2021, 8: 733104.
[31] Modares H, Lewis F L, Naghibi-Sistani M B. Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems[J]. Automatica, 2014, 50(1): 193-202.
[32] Abu-Khalaf M, Lewis F L. Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach[J]. Automatica, 2005, 41(5): 779-791.
[33] Kamalapurkar R, Andrews L, Walters P, et al. Model-Based Reinforcement Learning for Infinite-Horizon Approximate Optimal Tracking[J]. IEEE Transactions on Neural Networks and Learning Systems, 2017, 28(3): 753-758.
[34] Greene M L, Bell Z I, Nivison S, et al. Deep Neural Network-based Approximate Optimal Tracking for Unknown Nonlinear Systems[J]. IEEE Transactions on Automatic Control, 2023: 1-8.
[35] Gao W, Deng C, Jiang Y, et al. Resilient reinforcement learning and robust output regulation under denial-of-service attacks[J]. Automatica, 2022, 142: 110366.
[36] Liu C, Zhang H, Xiao G, et al. Integral reinforcement learning based decentralized optimal tracking control of unknown nonlinear large-scale interconnected systems with constrained-input[J]. Neurocomputing, 2019, 323: 1-11.
[37] Yang Y, Vamvoudakis K G, Modares H. Safe reinforcement learning for dynamical games[J]. International Journal of Robust and Nonlinear Control, 2020, 30(9): 3706-3726.
[38] Ding L, Li J, Ye M, et al. Fully Distributed Resilient Cooperative Control of Vehicular Platoon Systems

Under DoS Attacks[J]. IEEE/CAA Journal of Automatica Sinica, 2022, 9(5): 937-940.

[39] Li B, Wen S, Yan Z, et al. A Survey on the Control Lyapunov Function and Control Barrier Function for Nonlinear-Affine Control Systems[J]. IEEE/CAA Journal of Automatica Sinica, 2023, 10(3): 584-602.

[40] Marvi Z, Kiumarsi B. Barrier-Certified Learning-Enabled Safe Control Design for Systems Operating in Uncertain Environments[J]. IEEE/CAA Journal of Automatica Sinica, 2022, 9(3): 437-449.

[41] Kokolakis N M T, Vamvoudakis K G. Safety-Aware Pursuit-Evasion Games in Unknown Environments Using Gaussian Processes and Finite-Time Convergent Reinforcement Learning[J]. IEEE Transactions on Neural Networks and Learning Systems, 2022: 1-14.

[42] Patil O S, Le D M, Griffis E J, et al. Deep Residual Neural Network (ResNet)-Based Adaptive Control: A Lyapunov-Based Approach[C]//2022 IEEE 61st Conference on Decision and Control (CDC). Cancun, Mexico: IEEE, 2022: 3487-3492.

[43] Le D M, Patil O S, Nino C F, et al. Accelerated Gradient Approach For Neural Network Adaptive Control Of Nonlinear Systems[C]//2022 IEEE 61st Conference on Decision and Control (CDC). 2022: 3475-3480.

[44] Sun R, Greene M L, Le D M, et al. Lyapunov-Based Real-Time and Iterative Adjustment of Deep Neural Networks[J]. IEEE Control Systems Letters, 2022, 6: 193-198.