

分类号：TU834

单位代码：10363

密 级：公开

学 号：2210330149

题 目 智能楼宇风平衡控制技术研究

题 目 Research on Intelligent Building
Air Balancing Control Technology

学 生 姓 名：陈刚

校 内 导 师：程凡永（副教授）

校 外 导 师：刘海广（工程师）

专业学位类别：电子信息

研究方向（领域）：人工智能与系统集成

论文答辩日期：2024年5月19日

安徽工程大学学位论文原创性声明

本人郑重声明：我恪守学术道德，崇尚严谨学风。所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已明确注明和引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品及成果的内容。论文为本人亲自撰写，我对所写的内容负责，并完全意识到本声明的法律后果由本人承担。

学位论文作者签名：陈刚

日期：2024年5月24日

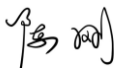
安徽工程大学学位论文版权使用授权书

学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅或借阅。本人授权安徽工程大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

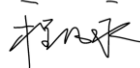
保密 ☐，在 ____ 年解密后适用本版权书。

本学位论文属于

不保密 ☒。

学位论文作者签名： 

日期：2024 年 5 月 24 日

指导教师签名： 

日期：2024 年 5 月 24 日

智能楼宇风平衡控制技术研究

摘要

智能楼宇中的风平衡控制是降低通风管道系统能耗，提高室内舒适性和健康环境质量的关键技术和研究热点。多支路通风管道系统是一类强耦合和非线性的多输入、输出系统。由于系统内部管道拓扑结构复杂多样，支路间气流的强耦合作用使得风平衡调试困难。针对多支路通风管道系统风量平衡收敛速度慢、难以准确控制，且系统物理建模复杂度高的问题，提出了风机自适应的分布式协同风平衡控制方法和基于强化学习的风平衡控制方法。从提高风量平衡的收敛速度、准确控制和通用性出发，开展了以下研究工作：

（1）针对通风管道系统多区域风平衡问题，分析多支路通风管道系统风量控制原理和分布式协同控制在多智能体领域的应用，提出风机自适应的分布式协同控制方法(DCAC-AB)。首先，基于共识算法，实现各相邻支路间规定信息交换，通过分布式协同控制风阀使支路空气流量与设定流量比值达成一致。然后，设计风机自适应控制算法调节系统总进风量，解决风机控制中由于多目标风量控制的时变性导致风量平衡收敛困难问题。最后，根据通风系统风量控制原理，设计风机自适应的分布式协同风平衡控制方法步骤。实验结果表明，该方法不需系统拓扑结构和管道参数，是一种快速有效的风平衡控制方法。

(2) 针对高精度目标风量控制问题, 分析强化学习在多支路通风管道系统中风量平衡控制理论和模型, 提出基于强化学习的风平衡控制方法(DRL-AB)。首先, 分析深度强化学习算法, 根据系统风量控制问题的多解性质, 设计了具有马尔可夫性质的系统风量控制过程, 设置了以减小风量误差为导向的系统状态、动作, 设计了控制误差的奖励函数。然后, 基于预训练-在线学习架构, 设计了单支路动态目标训练机制, 减小了训练代价, 提高模型的训练效率和面对不同目标风量的泛化能力。设计在线学习过程, 以预训练过程中的网络结构和参数作为初始化, 保证实际运用中的控制效率。最后通过实验测试和对比实验, 得出所提方法可实现大范围的高精度风量目标控制要求, 且具有较快收敛速度和较高的通用性, 是一种快速、精准的风平衡控制方法。

论文提出的两种风平衡方法, 适用于不同的楼宇建筑环境要求: DCAC-AB 法可在系统正常运行时实施, 不中断服务; DRL-AB 法经系统环境预训练后, 可实现对目标风量的高精度控制。两种方法不仅有助于降低能耗, 还能提高室内舒适度, 可在实际应用中推广。

关键词: 多支路通风管道系统, 风平衡, 分布式协同控制, 风机自适应控制, 深度强化学习, 单支路动态目标训练机制

RESEARCH ON INTELLIGENT BUILDING AIR BALANCING CONTROL TECHNOLOGY

ABSTRACT

In intelligent buildings, air balancing control is a key technology and research focus for reducing energy consumption of ventilation duct systems, enhancing indoor comfort, and improving the quality of the indoor environment. Multi-branch ventilation duct systems represent a type of strongly coupled and nonlinear multi-input, multi-output system. Due to the complex and diverse internal pipe topology of the system, the strong coupling between branches makes air balancing tuning challenging. To address the issues of slow convergence of air balancing and difficulty in accurate control in multi-branch ventilation duct systems, as well as the high complexity of system physical modeling, two approaches have been proposed: fan-adaptive distributed cooperative air balancing control method and reinforcement learning-based air balancing control method. From the perspective of improving the convergence speed, accuracy, and universality of air balancing, the following research work has been conducted:

(1) For the issue of multi-zone air balancing in ventilation duct systems, this paper analyzes the principles of airflow control in multi-branch ventilation duct systems and the application of distributed

cooperative control in the multi-agent domain. It proposes a fan-adaptive distributed cooperative control method (DCAC-AB). Firstly, based on consensus algorithms, it achieves information exchange between adjacent branches, and through distributed cooperative control of flowrate, it ensures that the airflow ratio of each branch matches the set airflow ratio. Then, it designs a fan adaptive control algorithm to regulate the total air intake of the system, addressing the difficulty of convergence in air balancing due to the time-varying nature of multi-target airflow control in fan control. Finally, based on the principles of airflow control in ventilation systems, it outlines the steps of fan-adaptive distributed cooperative air balancing control method. Experimental results demonstrate that this method does not require system topology and pipeline parameters, making it a rapid and effective air balancing control method.

(2) For the problem of high-precision target airflow control, this paper analyzes the theory and model of air balancing control in multi-branch ventilation duct systems using reinforcement learning and proposes a reinforcement learning-based air balancing control method (DRL-AB). Firstly, by analyzing deep reinforcement learning algorithms and considering the multi-solution nature of the system's airflow control problem, a system airflow control process with Markov properties is designed. This process sets system states and actions aimed at reducing

airflow errors and designs a reward function for control errors. Then, based on the pre-training-online learning architecture, a single-branch dynamic target training mechanism is designed to reduce training costs, improve model training efficiency, and enhance the model's ability to generalize to different target airflow conditions. An online learning process is designed to utilize the network structure and parameters from the pre-training phase as initialization to ensure control efficiency in practical applications. Finally, through experimental testing and comparative analysis, it is concluded that the proposed method achieve a wide range of high-precision airflow target control requirements with fast convergence speed and high versatility, making it a rapid and accurate air balancing control method.

The two air balancing methods proposed in the paper are suitable for different building environments: the DCAC-AB method can be implemented while the system is operating normally without interrupting services; the DRL-AB method, after pre-training in the system environment, can achieve high-precision control of target airflow. Both methods not only contribute to energy savings but also enhance indoor comfort, making them suitable for practical application and promotion.

Chen Gang(Electronic Information)

Supervised by Cheng Fanyong

KEY WORDS: multi-branch ventilation duct system, air balancing,
distributed cooperative control, fan adaptive control, deep
reinforcement learning, single-branch dynamic target training
mechanism

目 录

第 1 章 绪论.....	1
1.1 研究背景及研究意义.....	1
1.2 多支路通风管道系统风平衡控制技术的发展与研究现状.....	2
1.3 主要研究对象与内容.....	5
1.4 本文结构安排.....	6
第 2 章 通风管道系统模型分析及实验平台介绍.....	9
2.1 多支路通风管道系统的管道模型.....	9
2.1.1 拓扑结构模型.....	9
2.1.2 系统组件及其风量-压力模型.....	10
2.2 实验平台介绍.....	13
2.3 本章小结.....	16
第 3 章 风机自适应的分布式协同风平衡控制方法.....	17
3.1 引言.....	17
3.2 分布式协同控制理论.....	17
3.2.1 共识算法.....	18
3.2.2 风阀分布式协同控制.....	20
3.3 风机自适应控制.....	21
3.3.1 PID 控制算法.....	21
3.3.2 自适应 PID 控制.....	22
3.4 实验设计与步骤.....	26
3.5 实验验证.....	29
3.6 本章小结.....	31
第 4 章 基于强化学习的风平衡控制方法.....	32
4.1 引言.....	32
4.2 强化学习背景知识.....	32
4.2.1 马尔可夫决策过程.....	33
4.2.2 深度强化学习算法.....	36
4.2.2.1 Deep Q Network 算法.....	36
4.2.2.2 Double Deep Q Network 算法.....	39
4.2.2.3 探索和利用.....	40
4.3 基于强化学习的风平衡控制方法.....	41
4.3.1 基于强化学习通风管道系统 MDP 建模.....	41
4.3.1.1 状态特征.....	41
4.3.1.2 动作空间.....	43
4.3.1.3 奖励函数.....	43
4.3.1.4 终止条件.....	44
4.3.2 单支路动态目标训练机制.....	45
4.4 实验设计与步骤.....	47
4.5 实验验证.....	51

4.6 本章小结.....	57
第 5 章 风平衡控制系统 Web 端部署	59
5.1 系统需求分析.....	59
5.2 Web 端总体设计.....	59
5.2.1 开发环境介绍.....	59
5.2.2 远程监控平台架构.....	59
5.3 客户端设计.....	60
5.3.1 登录界面.....	60
5.3.2 系统简介界面.....	61
5.3.3 平台监控界面.....	61
5.3.4 远程控制界面.....	62
5.3.5 风阀信息界面.....	63
5.3.6 风阀状态控制界面.....	64
5.3.7 历史数据查询界面.....	64
5.4 本章小结.....	65
第 6 章 总结与展望.....	66
6.1 总结.....	66
6.2 展望.....	66
参考文献.....	68
学术成果.....	75
致谢	76

第 1 章 绪论

1.1 研究背景及研究意义

作为当前能源消耗量最大的行业之一，建筑业在全球范围内所消耗的能源相当巨大，大约占到了全球能源消耗总量的 40%左右，且运行中所排放的二氧化碳量约占到全球碳排放总量的 30%以上^[1-2]。2022 年底，中国建筑节能协会能耗专委会发布了《中国建筑能耗研究报告(2022)》，该报告对中国建筑行业的能源消耗情况进行了深入研究，指出 2022 年建筑运行阶段能耗为 10.6 亿吨标准煤当量（Ton of standard Coal Equivalent，简称 TCE），占全国能源消费总量的比重为 21.3%；建筑运行阶段碳排放 21.6 吨二氧化碳，占全国碳排放总量的比重为 21.7%^[3]，并且能耗和排放量逐年增加。习近平总书记在党的二十大报告中指出，推动绿色发展，促进人与自然和谐共生。国家发展和改革委员会（National Development and Reform Commission，简称 NDRC）将提高能源利用效率和实行节能作为支持经济高质量发展的一个重要方向。NDRC 指出，提高能源利用效率和实行节能不仅是构建美丽中国、应对气候变化等目标的前提，也是扩大绿色发展新动能的重要因素。此外，提高能源利用效率和实行节能也是推动人民普遍福祉改善的重要内容^[4-5]。为响应国家政策，实现绿色发展，建筑领域的节能发展是实现新型城镇高质量发展的重要一环。

供暖、通风和空调（Heating, Ventilation and Air Conditioning，简称 HVAC）系统是建筑运行中的主要能源消耗者，高达 51%^[6]。HVAC 系统广泛应用于商场办公、工厂生产等环境中，用于调控室内的温度、湿度和其他环境参数，以满足不同的需求，是建筑运行阶段的第一耗能设备^[7-9]。通风管道系统是 HVAC 系统中最大的耗能子系统之一，是保证室内空气质量的关键^[10-12]。通风管道系统通过风机将室外的新鲜空气泵进建筑内，再由管道将空气分配到各个区域中。不同建筑环境中通风管道系统管道结构如图 1-1 所示。在实际的通风管道系统中，气流分布通常是不均匀的。通风不足会导致室内空气循环不足，从而患上病态建筑综合征（如干咳、头痛）^[13]；过度通风则会导致能源的过度浪费^[14-15]。因此，开发高效的风平衡技术变得非常必要，这不仅可以提升能效，还可以提

高室内的健康舒适性。



(a) 工业建筑环境中的管道结构 (b) 办公建筑环境中的管道系统

图 1-1 通风管道系统管道结构图

多支路通风管道系统是具有多条支路末端的大型通风系统，主要由风机、管道、管道配件和风阀组成。风平衡是一种通过管道将适当的气流分配到所需支路终端的技术，它能有效地避免能量浪费，提高通风管道系统的能源效率，很多学者对风平衡技术开展了相关研究^[16-24]。但是，由于多支路通风管道系统各支路间拓扑连接方式以及管道中气流之间的耦合作用，使得各个系统终端的空气流量难以同时达到设定值。此外，不同的建筑环境有着不同的管道拓扑结构，随着建筑环境规模的扩大其复杂程度也会提高，系统的风平衡控制过程也变得困难，需要为每个通风管道系统开发针对性的风平衡策略，导致开发成本高、工作量大。

目前，多支路通风管道系统风平衡控制的实现都是通过基于迭代的方法实现。该方法由安装工人根据支路末端风量不断试错调整阻尼器角度，最终实现多区域的风量平衡。但是这种方法高度依赖工人的经验，管道系统结构的复杂性对调整时间影响较大，因此昂贵且费时费力，且这种方法一经安装后难以进行二次调整。随着建筑行业的发展，这种依靠工人经验实现风平衡的迭代方法逐渐不适用于规模庞大、结构复杂的通风管道环境，人们迫切需要能够快速调节、准确实现风量控制、泛化性强、调节范围广的风平衡控制策略。

综上所述，建筑行业的绿色发展正逐渐引起人们的重视。多支路通风管道系统作为 HVAC 系统最大的耗能子系统之一，快速的调节速度和精准的风量控制是建筑节能运行的研究热点。如何开发一种适用于多支路风量控制且泛化性强的风平衡控制策略成为重要的研究问题。

1.2 多支路通风管道系统风平衡控制技术的发展与研究现状

根据美国供暖、制冷和空调协会（American Society of Heating, Refrigerating and Air-Conditioning Engineers, 简称 ASHRAE）提出的标准，风平衡要求通风管道各支路出口的风量与设定风量的误差为 10% 以内。为实现室内各区域的通风要求，在通风管道设计之初，可以采用摩擦法和静态恢复法^[25-26] 等方法设计管道尺寸来达到风量要求目的，但由于存在建模或安装误差，整个系统在实际中可能无法达到预期效果，而且这种通过改变管道设计实现风量需求的方法随着时间的推移和使用情况的改变，系统可能会再次出现不平衡，因此需要进行定期的风平衡维护任务。

风平衡是在调试阶段进行测试、调整和平衡（Test, Adjust and Balance, 简称 TAB），确保系统在投入使用后能够按设计要求工作。比例法是现代建筑中最常使用的风平衡控制法，被众多的学会（ASHRAE^[27]、美国国家环境平衡局^[28]等）所采纳。该方法通过调节风机电压保持总出风量与目标风量一致，再逐步调整各支路中的风阀阻尼器角度，以此来实现不同区域的风量需求。比例法本质上是一种迭代调控的方法，需要反复调节尝试来平衡每个支路末端气流至设定值。由于管道系统中各支路间的气流存在较强的耦合作用，一条支路上风量的改变会导致已经满足目标风量的支路发生变化，导致系统的风平衡难以实现。实际应用中，比例法持续时间和控制效果在很大程度上取决于技术人员，导致该方法具有较大的局限性，且其风量控制的精确程度和调整过程中迭代次数也难以得到保证和控制。

为解决比例法存在调节迭代次数不可预测的问题，Federico 等人提出一种非迭代的标签检测方法，即“渐进流法”^[29]，鉴于通风管道中最末端的气压最小，空气流量低，该方法从最末端的风阀开始依次向前调整，在最远的终端气流与风机转速之间建立一个控制回路，以确保在平衡过程中整个下游的工况不变性。管道中风阀的调整是一个非重复的过程，即每个风阀只调整一次，对于一个有 n 个端子 m 个分支的通风管道系统，渐进流法相较于比例法减少了 $2n+m-1$ 次测量。根据这种反向调整方法，Tamminen 等人引入了一种无需测量终端的非迭代方法，称为基于风机压力的渐进流方法^[30]。该方法在平衡过程中控制风机静压保持恒定，通过风机 QP 曲线来估计气流量，对所有风阀逐步调节，直到估计气流量满足当前的设计值。然而，该方法仅适用于公用管道较短的小规模通风

管道系统，并且其精度不可控，依赖于制造商提供的数据和控制系统的精度。虽然上述两种方法一定程度上提升了 TAB 效率，但仅适用于公共管道段且支路少的通风管道系统中，在面对复杂的建筑环境时调节效果受到局限。

非迭代的风平衡控制方法在一定程度上提高了多支路管道间的调节效率，但是管道内部气流间的物理规律尚不清楚，难以实现高精度的风量控制。许多学者对通风管道系统支路节点处的压力与风量进行建模，提出了基于模型的风平衡控制方法。Small 等人建立了一种数学模型，可以精确计算出满足设计风量要求的风阀阻尼器角度^[31]。该模型通过风量守恒定律和压降方程将节点压力与分支气流联系起来。系统参数由风机特性和每个分支的最小摩擦损失系数这两组测量值决定，然后将设计值代入数学模型，得到所需的风阀阻尼器角度的预测值。然而，该方法仅适用于没有连接的简单管道系统，在复杂情况下可能无法提供可靠的预测。Chen 等人在数据采集效率和参数识别精度方面进一步改进了 Small 的方法^[32-33]，对该模型进行了数值计算和实验验证。但这种方法在很大程度上依赖于优化算法，并且需要很高的计算能力，在大多数移动设备上都无法实现。此外，从人工输入或建筑信息模型(BIM)中获取模型结构操作复杂，需要较强的专业知识，无法大面积普及应用。Jing 等人提出了一种改进方法^[34-35]，将系统本身的参数与风阀参数分开，未确定的模型参数利用有监督的机器学习方法来识别，避免了求解非线性方程组时遇到的困难。然而，这些方法对拓扑结构复杂的通风管道系统，需要巨大的离线训练数据，收集足够的训练数据耗时且昂贵，而且有时可能由于缺乏适当的自动化设备而无法获得，难以得到广泛使用。

随着对风平衡控制理论的不断研究，许多学者将模糊控制算法引入风平衡控制以解决多支路通风管道中风量精确控制问题。为实现风机最低能耗下的风平衡，Zhang 等人以风阀阻尼器最大开口角度作为静压标准，对风机电压进行调整，提出基于最大阻尼器角度位置反馈的静压复位风平衡模糊控制方法^[36]。为进一步节约能耗，Li 等人建立了风阀阻尼器角度与风机静压的模型，提出基于模糊规则的模型静压风平衡控制方法^[37]。通过利用风阀阻尼器开口角度对风机静压进行调整，从而实现多支路风平衡策略。上述这两种方法虽具备较强的抗干扰能力，在部分多支路系统中有较好收敛性能，但其收敛效果受经验调节

参数的影响。

为实现多支路环境中的风平衡控制，一些学者将目光投向了正快速发展的机器学习领域，提出了许多基于数据驱动的风平衡控制方法。通过获取风阀阻尼器角度与风量间的数据，无需建立准确的通风管道系统模型，由所提供数据学习输入到输出间的映射关系实现多支路风量平衡。Jing 等人提出了基于向量的通风系统风平衡方法^[34]。该方法通过构建管道静压模型，由经过数据拟合过的机器学习模型预测出支路中风阀的位置。为实现更精准的风量控制，Zhang 等人提出了利用神经网络实现通风系统的空气平衡方法^[38]。该方法通过建立一个多层感知器模型来预测阻尼器位置，利用阻尼器处压降和气流速率之间的关系实现风平衡。然而这种方法需要在每个支路处安装压力传感器，根据风压确定风阀阻尼器角度。为实现对各支路风量的精准控制，Cui 等人提出了基于全数据化的风平衡策略^[39]。直接在通风管道系统内建立分支风量到支路风阀的数据模型，无需借助支路风阀处风压信息。Mu 等人提出了多区域预测风阀角度的风平衡控制方法^[40]。避免风机泵入新风时的静压波动，实现风机的快速最优调节。

以上基于数据驱动的风平衡控制方法，通过拟合数据间的数学关系来实现各支路的风量控制。然而这些方法需要获取大量标签数据建立模型，一些数据收集难且标记成本高，控制方法的收敛性能也受采集数据的质量影响。另外，对于不同拓扑结构下的管道支路，需要针对该系统重新收集数据，导致基于数据驱动的风平衡方法缺乏通用性。

1.3 主要研究对象与内容

风平衡是一种通过管道将适当的气流分配到所需终端的技术，能够有效避免能量的浪费，提高通风管道系统的能源效率。现有的风平衡方法，大多无法应对拓扑结构复杂的通风管道系统，且收敛速度较慢，无法大范围调节风量，在系统开始运行时便不能对系统进行实时调节，这极大地限制了商场和工厂等环境对通风的不同需求。本课题在已经搭建好的智能物联网风平衡平台的基础上，针对复杂的管道拓扑结构建筑环境，对风平衡控制理论与技术方法展开研究，提升风平衡的工作效率和实用性。

通风管道系统一般有四部分组成：风机、管道、管道配件和风阀，如图 1-2 所示。风机为室内提供处理后的新风；管道一般包括一个主管道和多个分支管道，管道间通过管道配件连接。分支管道连接带有供气出口的终端，将新风输送到所需区域；支路末端安装风阀，通过改变风阀内部阻尼器开口角度来调节区域供风量，以满足空气质量要求。

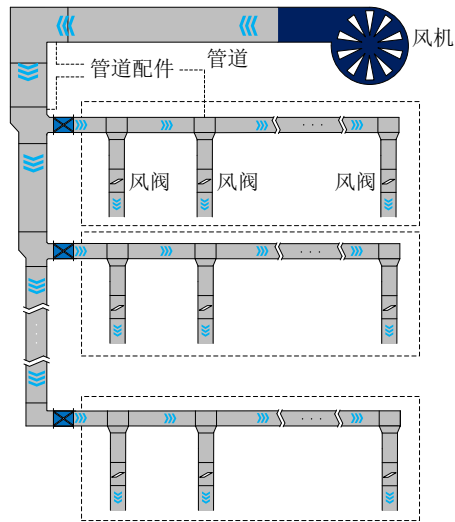


图 1-2 通风管道系统结构图

为实现多支路通风管道系统风量的精准控制，保证系统的节能运行，本课题在考虑平衡收敛速度慢和系统准确模型及数据标签难以获取的问题，研究了分布式协同控制方法理论与深度强化学习算法。本文通过研究通风管道系统的运行机理和多支路风量控制问题的多解性质，分别设计了风机自适应的分布式协同风平衡方法和基于强化学习的风平衡控制方法。针对多支路系统难以平衡收敛问题，本文引入共识算法，以实现多风阀间的分布式协同控制，并使用风机自适应控制方法使各支路快速收敛至目标值。本文设计了具有马尔可夫性质的通风管道系统风平衡控制过程，设计了单支路动态目标训练机制，通过预训练-在线学习框架实现对风量的精准控制，提高了风量平衡控制的收敛速度、精准控制和通用性。

1.4 本文结构安排

本文共分为六章用来介绍课题内容，论文结构安排如图 1-3 所示。每章的安排如下：

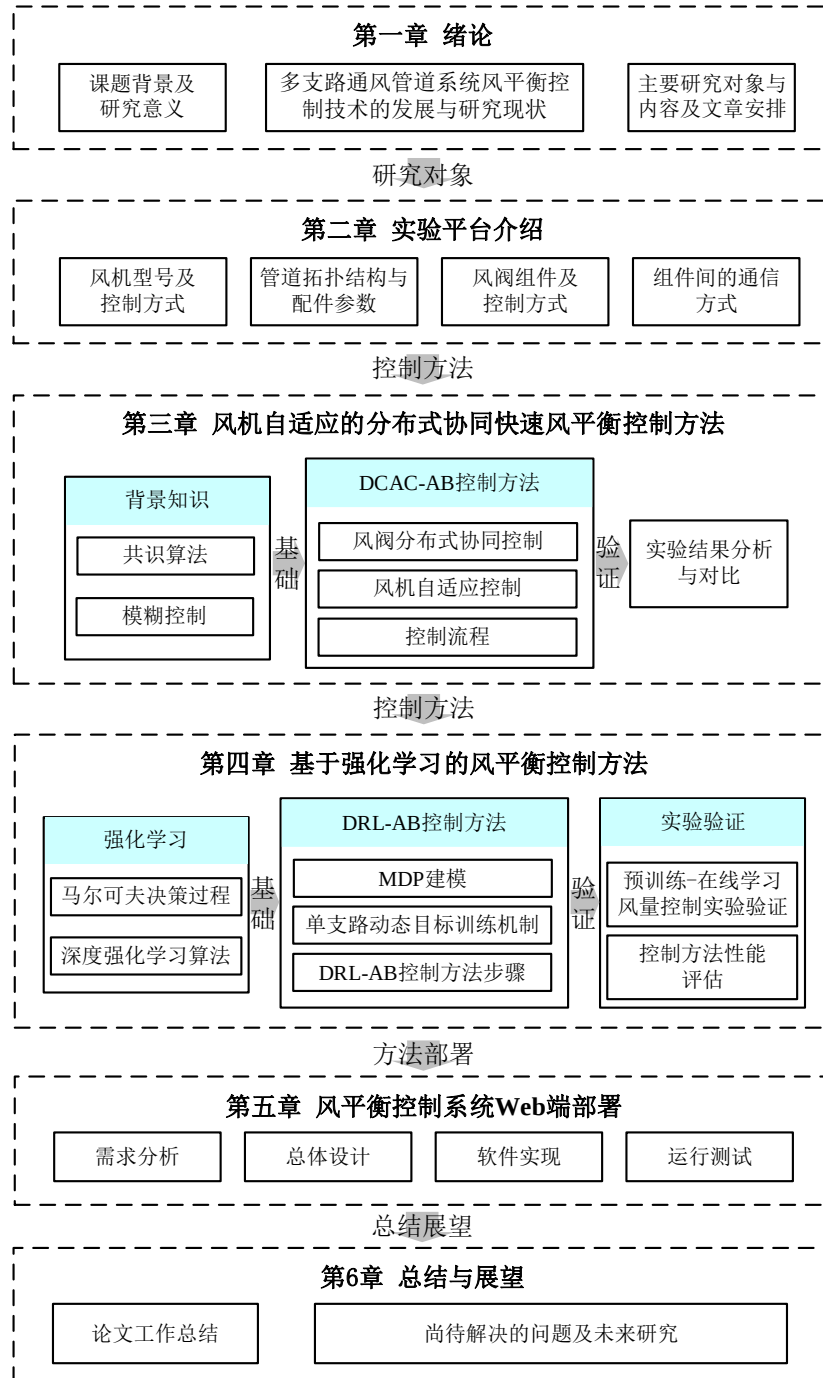


图 1-3 论文结构安排

第一章是绪论。该部分首先介绍了本课题的研究背景及其意义，接着介绍了智能楼宇风平衡控制方法的发展与研究现状，然后介绍了本课题所研究的对象与内容，最后给出了本文主要内容与结构安排。

第二章分析了多支路通风管道系统模型。首先，对其中的管道拓扑结构模型进行了介绍，随后分析了通风管道系统组件和风量-压力模型。最后对本课题的实验平台进行介绍，包括风机类型、管道参数和风阀内部结构，为后续控制

方法部署提供参考。

第三章提出了风机自适应的分布式协同风平衡控制方法。首先，介绍了共识算法，并将该算法引入多支路通风管道系统，使相邻风阀间进行信息交换，实现分布式协同控制。随后，根据风机的电压-风量特性，设计了风机的模糊控制表，用以实现风机的自适应控制。接着对该方法的具体实现步骤进行介绍。最后，通过实验平台测试本方法的控制结果，验证所提风机自适应的分布式协同风平衡控制方法的可行性。

第四章提出了基于强化学习的风平衡控制技术。首先，介绍了强化学习的基本知识和本文章涉及到的深度强化学习算法。随后，根据多支路通风管道系统中风量控制问题的多解性，本章提出了基于强化学习的风平衡控制方法。本章设计了具有马尔可夫性质的支路风量控制过程，设计了以减小风量误差为导向设置强化学习的状态、动作及奖励函数。本文设计了一种单支路动态目标训练机制应用于预训练阶段实时与通风管道系统进行交互，学习风量控制网络结构参数，降低了训练代价，提高了模型学习效率。再将训练完成的单支路风量控制模型应用于多支路通风管道系统中，进入在线学习阶段，完善风平衡控制策略，保证控制方法的真实性能。最后，通过实验平台上对所提出的方法进行验证，验证结果表明所提方法有较强的泛化性，保证了实际应用中风量的精准、快速控制。

第五章，将两种控制方法部署至 Web 端。设计本系统的可视化图形界面，实现对系统风机、风阀和各支路风量信息的实时监测，用户可根据需要选择合适的风平衡方法运行模式，并可由 Web 端对通风管道系统组件在线控制。

第六章，总结与展望。该章内容对本文所提出的风平衡控制方法进行总结，梳理了目前尚未解决的技术难题，对下一阶段的研究任务进行展望。

第2章 通风管道系统模型分析及实验平台介绍

2.1 多支路通风管道系统的管道模型

通风管道系统中的管道模型，描述了管道内驱动压力与风量分布之间的关系以及管道内压力损失的情况。在该模型中，包含风量 q （单位： m^3/h ）、管道压降 Δp （单位： Pa ）与节点压力 p （单位： Pa ）三个变量，其中风量 q 指单位时间内空气的流通量，管道压降 Δp 指两个流通截面间风量静压的变化，节点压力 p 则表示多个配件之间连接点处的压力。

2.1.1 拓扑结构模型

在多支路通风管道系统中，新风由风机进入系统，通过系统内管道到达通风区域。其典型拓扑结构如图 2-1 所示。图中蓝色箭头为新风流经管道，其管道配件参数固定不变。图中共有 n 条分支管道，分支末端连接需风区域。

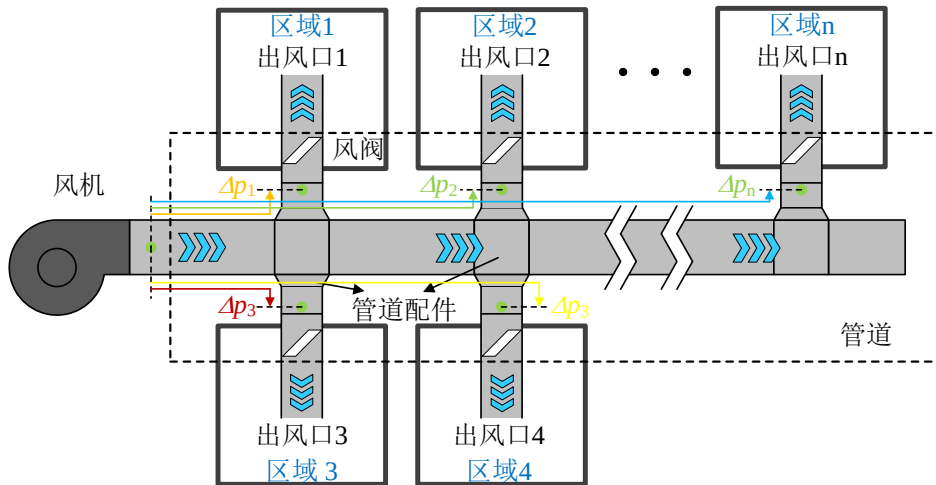


图 2-1 多支路通风管道系统的典型拓扑结构

通风管道系统中，节点之间至少由两个管道组件连接，空气通过两节点间的连接管道流至分支管道，且系统内风量遵守质量守恒定律。由于通风管道系统内支路间的空气流量与电路中电流存在相似性，利用这种特性并结合节点分析法^[41]对多支路通风管道系统建立管道模型，该模型可描述各分支处风量与节点压力的关系。其数学表达式如下：

$$\begin{cases} Aq = 0 \\ \Delta p = A^T p \\ \Delta p_i = \sum_j \Delta p_{ij}(q, \beta) \end{cases} \quad (2-1)$$

方程中， A 是关联矩阵，用来表示管道系统分支间的拓扑结构， q 是分支中的空气流速， p 为管道节点处的压强， Δp 是分支间的压降， Δp_{ij} 是关于空气流速与管道参数 β 的非线性函数。

对于一个具有 M 个节点和 N 条分支的管道系统，存在 $M \times N$ 维的关联矩阵 A ，组成元素 A_{ij} 定义为：

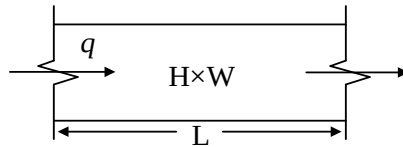
$$A_{ij} = \begin{cases} -1 & \text{分支}j\text{气流流入节点}i \\ 0 & \text{分支}j\text{不与节点}i\text{相连} \\ 1 & \text{分支}j\text{气流流出节点}i \end{cases} \quad (2-2)$$

通过通风管道系统中管道拓扑结构和管道组件参数，得到系统每个节点处的压力和管道内气流量。

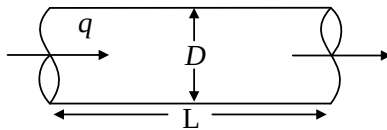
2.1.2 系统组件及其风量-压力模型

多支路通风管道系统由多种管道、管道配件、风阀和风机连接组成。

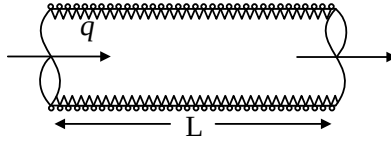
通风管道系统中管道为系统输送新风。常见的管道类型有圆形、矩形和螺旋管道，如图 2-2 所示。圆形管道空气摩擦损失小，因为其圆形截面，制造和安装相对简单，成本较低。矩形管道在空间有限的情况下更容易适应，可以更好利用有限资源，但接头连接处成本较大。螺旋管道兼具上述两种管道优点，且能够减少震动和噪音。



(a) 矩形管道图



(b) 圆形管道图



(c) 螺旋管道图

图 2-2 管道类型图

空气在管道内流通会与管壁摩擦造成压力损失，其压力损失可由达西公式表示^[42]：

$$\Delta p_{\text{duct}} = C_f^{\text{duct}} \frac{L}{D_h} \frac{1}{2} \rho v^2 \quad (2-3)$$

式中， C_f^{duct} 为达西摩擦系数， L 为管道长度， D_h 为水利直径，不同管道的水利直径与其长度、阻力和风量相关， ρ 为管道内空气密度， v 为平均风速，与风量关系为：

$$q = \pi \left(\frac{D_h}{2} \right)^2 v \quad (2-4)$$

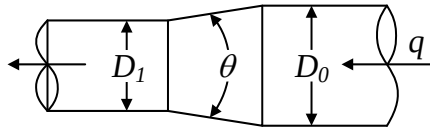
摩擦力系数 C_f 可由经验公式估计^[43]：

$$\frac{1}{\sqrt{C_f}} = -2 \log_{10} \left(\frac{\varepsilon}{3.7 D_h} + \frac{2.51}{\text{Re} \sqrt{C_f}} \right) \quad (2-5)$$

式中， ε 为管道内部粗糙程度， Re 为雷诺数：

$$\text{Re} = 66.4 \times D_h v \quad (2-6)$$

管道系统中管道间通过组件连接，如图 2-3 所示。不同尺寸大小的管道间用变径管相连，当管道系统需改变方向或产生分支时，可用弯头和 T 形接头实现。所有组件内部皆呈弧形，用来减小压损。



(a) 变径管

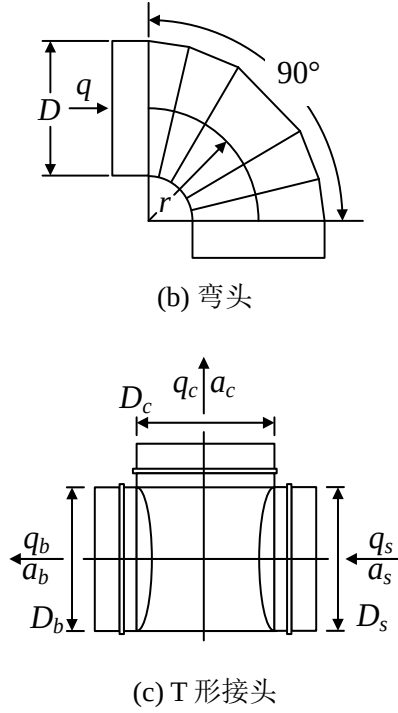


图 2-3 管道组件

通过查询 ASHRAE 中管道数据库(ASHRAE-DFDB)^[44]可以得到常用管道配件公式和 $C_f^{fitting}$ ，以此来估计各类型管道中的压力损失：

$$\Delta p_{fitting} = \frac{1}{2} C_f^{fitting} \rho v^2 \quad (2-7)$$

每条末端支路的出风量通过风阀来调节。常见的风阀类型有蝶阀和多叶阀等，如图 2-4 所示。

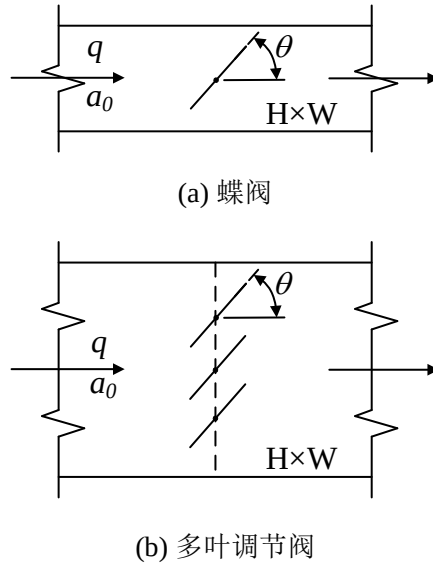


图 2-4 风阀类型图

风阀通过调节内部阻尼器的角度 θ 来改变风阀的摩擦系数 C_f^{damper} ，以此改

变出口处的风压从而控制风量。不同角度下的摩擦系数 C_f^{damper} 可由 ASHRAE 中管道数据库(ASHRAE CD9-1)查询。

风机将室外新风输入至管道系统中。目前通风管道系统中最常用的风机类型为离心风机，其结构简单且易控制，如图 2-5 所示：

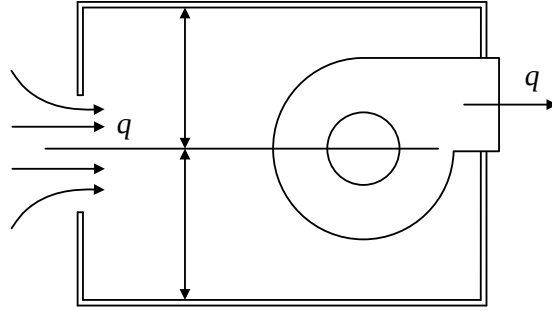


图 2-5 离心风机

风机为系统提供最为重要的新风能源，通过叶轮或旋转叶片，将外部空气输送至管道中，克服系统内部阻力使空气流通至管道各处。当支路末端的压强大于大气压时，风机静压 p_{fan} 为管道内的压力损失之和：

$$p_{fan} = \sum (\Delta p_{duct} + \Delta p_{fitting}) \quad (2-8)$$

2.2 实验平台介绍

为测试所提出风平衡控制方法在实际应用中的性能，本课题通过对实际商场和工厂通风设备的研究，结合实际建筑环境，搭建通风管道系统实验平台，如图 2-6 所示。该实验平台包含风机、管道和风阀。平台原理图如图 2-7 所示。风机向系统输送新风，歧管将主管道内气流分配到分支管道中，每个支路末端安装一个风阀，用以控制该支路出口的风量。实验平台中，共有 11 个分支管道。图 2-7 中，T1~T3 为长方形截面管道，T4~T6 为正方形截面管道，T7~T11 为圆形截面管道。在图 2-6 中，蓝色箭头为新风流经管道，红色标注为风阀，用来控制每条支路出风量。

本实验平台使用的风机型号为 B3P310-EC137-010，具备出色的调节性能。该风机最大可提供 $4270 \text{ m}^3/\text{h}$ ($1.19 \text{ m}^3/\text{s}$) 的气流速率，可模拟出各种高风量的应用环境需要。风机转速由 NI 电压输出模块(NI 9264)产生的 0-10 VDC 信号进行控制。通过控制电压的大小，可以精确地调节风机的转速，从而实现对气流速

率的控制。NI 电压输出模块(NI 9264)是一种高性能的控制设备，能够生成稳定可靠的电压信号，用于控制外部设备的运行。通过与风机控制系统的连接，该模块可以实现对风机转速的精确控制，确保实验平台的稳定运行和可靠性。



图 2-6 通风管道系统实验平台

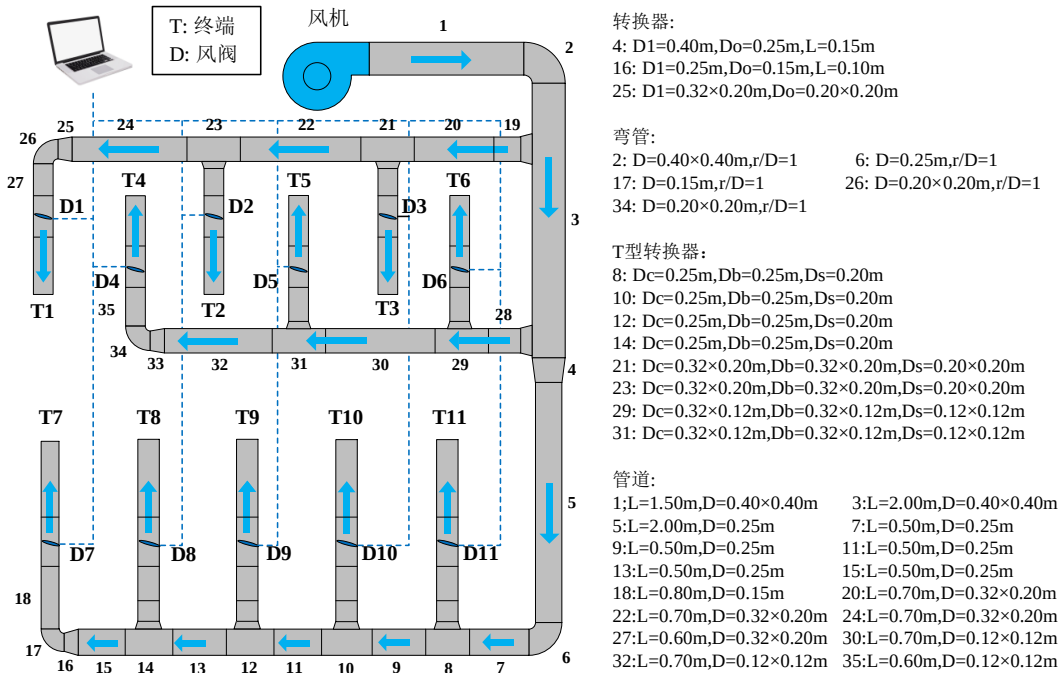


图 2-7 实验平台原理图

实验平台中所有管道采用 0.6 mm Z22 SGCC 镀锌钢螺旋管组成，具有较高的耐腐蚀性，螺旋管的结构使得管道具有较高的强度和刚性，内部能够承受较大风压。平台主管道由方形和圆形管道连接组成，主管道分出多条分支，每条分支连接不同尺寸的支路出风口，支路出口安装有风阀控制出风量。不同形状

的管道用来模拟不同环境对出风口尺寸的要求。每条分支初始端安装有比例阀，实现对各分支的总开关。管道中所有弯头均为 90°焊接弯头，用以提供平滑的流动路径，减少气流的阻力和压力损失。此外，管道系统中没有内部通风口。

通风管道系统中关键部件是智能风阀，根据管道支路类型设计有长方形、正方形、圆形三种结构，如图 2-8 所示。其中圆形和方形智能风阀设计图如图 2-9 所示，风阀控制器是一种用于感知测量和控制风流量的集成设备，它主要包括平均皮托管(APT)、压差变送器、阻尼器执行器等。APT 是一个双板皮托管，采用流线型设计兼顾了低流速的灵敏度和高流速的低噪音，根据压差和叶片角度校准，进行流量测量，测量误差在 5%以内。嵌入式压差传感器为欧姆龙 D6F-PH0505AD3，读取精度为 $\pm 3\%$ ，测量范围为 ± 50 Pa。阻尼器执行器是由伺服电机驱动的单板叶片，在 0°（全开）到 90°（全闭）之间旋转，角分辨率为 $\pm 0.18^\circ$ 。

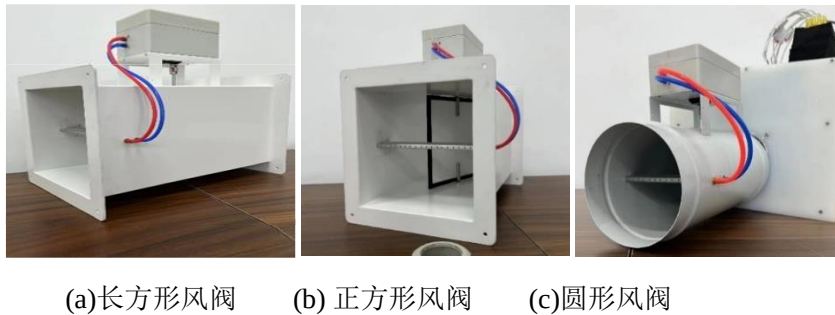


图 2-8 风阀控制器

智能风阀控制器通过检测平均皮托管输出的差压转换为空气流量进行控制，来调整阻尼叶片角度以实现闭环风流量控制。PWM 驱动器驱动伺服电机实现阻尼叶片控制，调节风量达到风平衡。再由通信模块实现终端与监控计算机之间的无线双向数据传输。

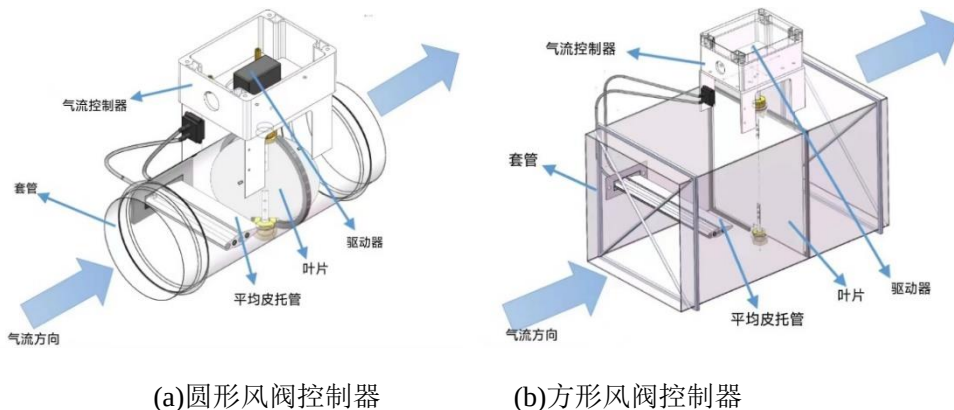


图 2-9 风阀控制器

2.3 本章小结

本章分析了多支路通风管道系统的管道模型。通过对管道拓扑结构的分析，管道内支路间连接方式会引起内部空气流量的相互影响，从而产生强耦合现象。根据对不同系统组件的数学模型分析发现，管道内部空气流量的大小与内部压强有着复杂的非线性关系。管道系统是由不同类型的管道和配件连接而成，通风管道系统内风量与压强间也是非线性关系。接着本章介绍了模拟实际环境搭建的通风管道系统实验平台，对平台中的管道、风阀和风机类型进行了详细介绍。本文中所提出的风平衡控制方法皆是基于该实验平台进行实验测试。

第3章 风机自适应的分布式协同风平衡控制方法

3.1 引言

根据第二章对实验平台的介绍和系统模型分析，通风管道系统由多条支路通过不同的拓扑结构连接而成。由于系统具备连续输入、多输出、强耦合的物理特性，导致通风管道系统难以使每条支路出风量同时达到设定要求。典型的风平衡控制方法为比例法，但是其控制效果和持续时间受安装工程师经验影响。目前，一些学者提出基于物理模型的风量控制法，但是这类方法的性能受构建模型的准确性影响。此外，部分学者提出基于数据驱动的风平衡控制方法，但该种方法需要获取大量数据标签，其数据收集困难，标记成本高。

为解决多支路通风管道系统风平衡收敛困难问题，本章将开展风机自适应的分布式协同风平衡控制方法的研究。首先，本章介绍了共识算法理论知识，并将共识算法引入多支路通风系统，规定相邻风阀间进行特定信息交换，使系统各支路实际风量与目标风量比值达到一致。接着设计了基于通风系统的风机自适应控制方法，控制系统总进风量实现风平衡控制。最后本章设计了风机自适应的分布式协同风平衡控制流程，并通过实验平台验证了所提方法的收敛性能。

本章的后续内容安排如下：首先，在 3.2 小节介绍了分布式协同控制理论知识，将共识算法引入多智能体控制体系，实现风阀的分布式协同控制。然后，3.3 小节对 PID 算法进行介绍，在此基础上设计了基于通风系统的风机自适应控制方法。接着在 3.4 小节对所选用实验平台分支进行描述，并设计风机自适应的分布式协同风平衡控制方法实现步骤。最后在 3.5 小节验证该方法的可行性与控制效果。

3.2 分布式协同控制理论

分布式协同控制的目标是为多智能体系统制定一个控制协议，以某种规定的方式实现所有智能体的同步行为。本节介绍一种基于共识性的分布式控制协议，其中多智能体动态系统由固定的通信图拓扑相互连接，每个智能体只依赖

于自身和相邻智能体的信息。

3.2.1 共识算法

通信网络图一般用 $G(V, E)$ 来表示，其中 V 是通信网络图中的一个顶点， E 是两个顶点间的一组边。边 E 用 (v_i, v_j) 来表示，即从 v_i 指向 v_j 。若 $(v_i, v_j) \in E$ ，则称该图为有向图，否则称为无向图。节点 i 的数量即是顶点 V 的邻居数量，用 $\deg(v_i)$ 表示。 G 的度矩阵为 $n \times n$ 的对角矩阵，用 $\Delta = \text{diag}\{\deg(v_i)\}$ 表示。通信网络图例如图 3-1 所示。

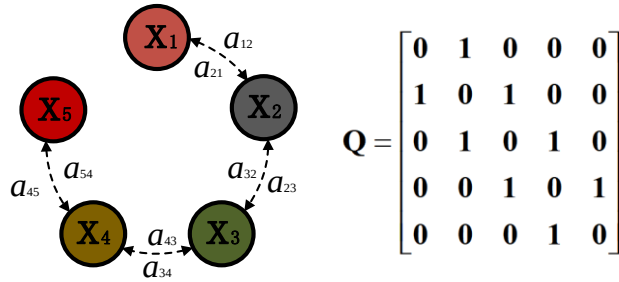


图 3-1 通信网络图例

通信网络图的结构用邻接矩阵 A 来表示， $A = [a_{ij}] \in R^{n \times n}$, a_{ij} 可表示为：

$$a_{ij} = \begin{cases} 1 & (v_i, v_j) \in E \\ 0 & \text{其他} \end{cases} \quad (3-1)$$

对于一个无向图，邻接矩阵是对称的，即 $A = A^T$ 。如果每对节点之间存在一条路径，则该通信图称为连通图。在本文中， G 是一个没有自环的简单通信网络图，即 $(v_i, v_i) \notin E$ ，这表示 $a_{ii} = 0$ ，没有多重边。

拉普拉斯矩阵 L 定义为：

$$L = \Delta - A \quad (3-2)$$

式中， L 中的元素组成定义为：

$$l_{ii} = \begin{cases} -1 & j \in N_i \\ |N_i| & j = i \end{cases} \quad (3-3)$$

式中， N_i 表示与节点 i 相邻的个数。拉普拉斯矩阵 L 中所有行的和为 0，即 $\sum_j l_{ij} = 0$ 。所以拉普拉斯矩阵 L 有一个重要特征，即 $v_0 = (1, \dots, 1)^T$ 是零特征值相关联的特征向量^[45]。且因为无向图具有对称的拉普拉斯矩阵 L ，所以它的特征值为实数^[46]。上述网络图例中拉普拉斯矩阵为：

$$L = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}$$

将每个节点赋予一个值 x_i ，用 $x = (x_1, \dots, x_n)^T$ 可以表示通信网络图 G 的状态。通信网络图的拉普拉斯势^[47]定义为：

$$\psi(x) = x^T Lx \quad (3-4)$$

可具体表示为：

$$x^T Lx = \frac{1}{2} \sum_{(v_i, v_j) \in E} a_{ij} (x_i - x_j)^2 \quad (3-5)$$

通信网络图的拉普拉斯势量化了所有节点（智能体）之间的总不一致。当且仅当多智能体系统达到共识时，即 $\psi(x) = 0$ ，表明该系统逐渐收敛于一维协议空间^[48]，其特征为：

$$x_1 = x_2 = \dots = x_n \quad (3-6)$$

本文中引用了一种简单的无通信时滞的连续时间积分器网络的简单线性一致协议^[48-50]：

$$\dot{x} = -\nabla \psi(x) = -Lx \quad x(0) \in \mathbb{R}^n \quad (3-7)$$

可表示为：

$$\dot{x}_i(t) = - \sum_{j \in N_i} a_{ij} (x_i(t) - x_j(t)) \quad (3-8)$$

对于离散时间的智能体，该协议表达如下：

$$x_i(k+1) = x_i(k) - T_s \sum_{j \in N_i} a_{ij} (x_i(k) - x_j(k)) \quad (3-9)$$

式中， T_s 是时间步长。

智能体通过共识协议与相邻的智能体进行局部通信，实现所有智能体收敛到集体决策。如果智能体间实现共识，则系统平衡的状态即为所有节点达成共识的状态：

$$x^* = (\alpha, \dots, \alpha)^T \quad (3-10)$$

对公式(3-8)中所有节点 i 求和得到 $\sum_i x_i = 0$ ，由此可知 $\dot{\bar{x}} = 0$ 。表明 $\bar{x}$ 为不变量，即 $\bar{x}(\infty) = \bar{x}(0)$ 。因此，可得到如下公式：

$$\alpha = \frac{1}{n} \sum_i x_i(0) \quad (3-11)$$

满足这种特定不变性的共识算法称为平均共识算法^[51]。该算法在负载平衡、传感器融合、移动自主智能体的协调等方面有广泛应用^[52-54]。

3.2.2 风阀分布式协同控制

风平衡的目的是确保空气均匀分布在管道系统中。均匀分布是指系统可以按照每个支路终端设定要求提供相匹配的空气流量。因为系统中支路管道间的空气流量相互影响，可将管道系统视为一个由固定通信网络图拓扑相互连接的多智能体动态系统，在这个系统中，每条支路中的风阀（智能体）空气流量与其相邻的风阀空气流量合作达成共识协议：

$$\frac{q_1}{q_1^*} = \dots = \frac{q_n}{q_n^*} = \alpha \quad (3-12)$$

式中， q 为实际空气流量， q^* 为设计空气流量， α 是最终共识值。公式(3-12)的实现可应用公式(3-9)中的平均共识算法来实现。将公式(3-9)中的 x 设为流量比，

即 $x_i = \frac{q_i}{q_i^*}$ ，可以得到：

$$\frac{q_i(k+1)}{q_i^*} - \frac{q_i(k)}{q_i^*} = -T_s \sum_{j \in N_i} a_{ij} \left(\frac{q_i(k)}{q_i^*} - \frac{q_j(k)}{q_j^*} \right) \quad (3-13)$$

公式(3-13)左侧差值可转换为风阀角度 θ 的增量，即：

$$\frac{q_i(k+1)}{q_i^*} - \frac{q_i(k)}{q_i^*} = \frac{J_{ii}(\theta_i(k+1)) - \theta_i(k)}{q_i^*} \quad (3-14)$$

式中 J_{ii} 是雅可比矩阵第 i 个对焦元素， $J = \text{diag}\{J_{ii}\}$ 。公式(3-13)可转化为：

$$\theta_i(k+1) - \theta_i(k) = \kappa_i \sum_{j \in N_i} a_{ij} \left(\frac{q_i(k)}{q_i^*} - \frac{q_j(k)}{q_j^*} \right) \quad (3-15)$$

式中, $\kappa_i = -\frac{T_s q_i^*}{J_{ii}}$, J 为对角矩阵, 用来忽略其他支路阻尼器角度变化导致的风

量变化, 即支路末端风量值的改变只考虑调节自身支路的阻尼器。通过实验分析风阀气流与角度($q-\theta$)的特性曲线, 可以得到雅可比矩阵, 其中 J_{ii} 是由 $q(\theta)$ 平均导数得到。公式(3-12)中 α 的值是由最终的流量比所决定的, 当 α 的值等于 1 时, 表明平衡完成后的实际风量总值等于风量总设定值。

3.3 风机自适应控制

3.3.1 PID 控制算法

目前风机系统中, 最常采用的控制算法为 PID 控制算法, 用来控制风机给定电压, 其原理方框图如图 3-2 所示。

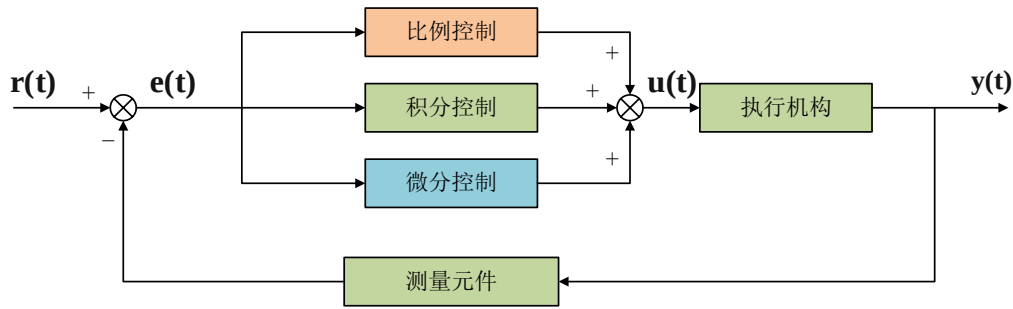


图 3-2 PID 控制系统原理方框图

传统 PID 输出结果为被控对象的设定值 $r(t)$ 与实际值 $y(t)$ 间的偏差 $e(t)$, 其控制目的使 $e(t)$ 越来越小, 直至消失。 $e(t)$ 计算公式如下:

$$e(t) = r(t) - y(t) \quad (3-16)$$

通过对偏差 $e(t)$ 做比例-积分-微分控制, 使其值能够随时间的推移逐渐减小得到输出值 $u(t)$ ^[55], 其计算公式如下:

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right] \quad (3-17)$$

式中, K_p 是比例因子, T_i 为积分时间, T_d 为微分时间。取积分因子 K_i 、微分因子 K_d :

$$K_i = K_p / T_i \quad (3-18)$$

$$K_d = K_p \cdot T_d \quad (3-19)$$

将公式(3-17)改写为:

$$u(t) = K_p e(t) + K_i \int_0^1 e(t) + K_d \frac{de(t)}{dt} \quad (3-20)$$

位置式 PID 在积分阶段对过去偏差 $e(t)$ 的累加, 通过计算得到输出值^[56]。

其计算公式为:

$$u(k) = K_p e(k) + K_i \sum_{j=0}^k e(j) + K_d [e(k) - e(k-1)] \quad (3-21)$$

由位置式 PID 控制算法得到离散 PID 表达式:

$$u(k-1) = K_p e(k-1) + K_i \sum_{j=0}^{k-1} e(j) + K_d [e(k-1) - e(k-2)] \quad (3-22)$$

将式(3-21)、(3-22)相减得到 $k-1$ 时刻控制量:

$$\begin{aligned} \Delta u(k) &= K_p [e(k) - e(k-1)] + K_i e(k) + K_d [e(k) - 2e(k-1) + e(k-2)] \\ \Delta u(k) &= K_p \Delta e(k) + K_i e(k) + K_d [\Delta e(k) - \Delta e(k-1)] \end{aligned} \quad (3-23)$$

3.3.2 自适应 PID 控制

传统 PID 控制方法, 由于其控制参数设定后无法改变, 在应对干扰和时变系统时存在一定的不足。因此, 引入模糊控制控制方法以提升收敛性能。本文设计出自适应 PID 控制算法, 使系统在有高精度和快速响应的同时, 还具有较强鲁棒性和参数自整能力, 从而实现通风管道系统中风机可根据不同的风量设定, 做出快速且精准的控制^[57]。

将系统输出偏差 $e(t)$ 先经模糊控制器处理, 再将输出结果输送至执行机构, 如图 3-3 模糊控制系统组成所示, 输出结果由四部分推导得出^[58]:

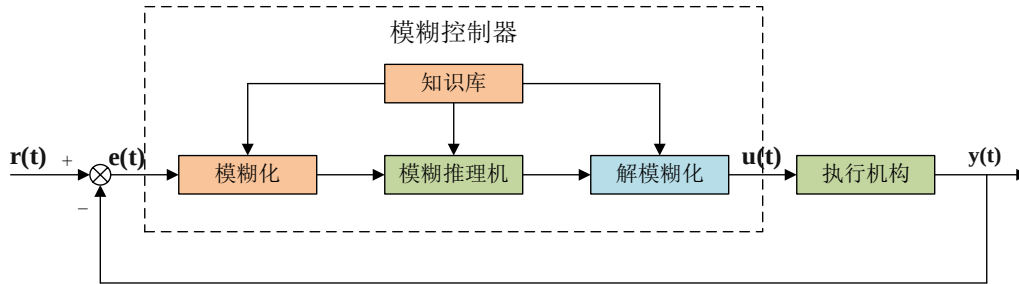


图 3-3 模糊控制系统组成

(1) 模糊化：这一过程是由具体输入值得到模糊量。首先对输入值进行处理，将其比例转化映射至论域中，再将其转换为模糊语言。

(2) 知识库：由数据库和模糊规则库构成。将具体输入值与模糊规则表相对应。

(3) 模糊推理机：通过模糊规则库选出当前偏差下最合适的控制参数。

(4) 解模糊化：将控制参数的模糊量转换为具体数值。

针对通风管道系统中的风机控制，将 PID 控制与模糊控制相结合，设计出根据风量误差及误差变化率计算出符合当前状态的 PID 控制参数值，从而实现目标风量的快速收敛。通过分析风机控制系统，将风量偏差 e 和偏差变化率 ec 作为系统输入，以 K_p 、 K_i 和 K_d 的变化量作为输出，设计风机自适应控制算法，其原理方框图如图 3-4 所示。

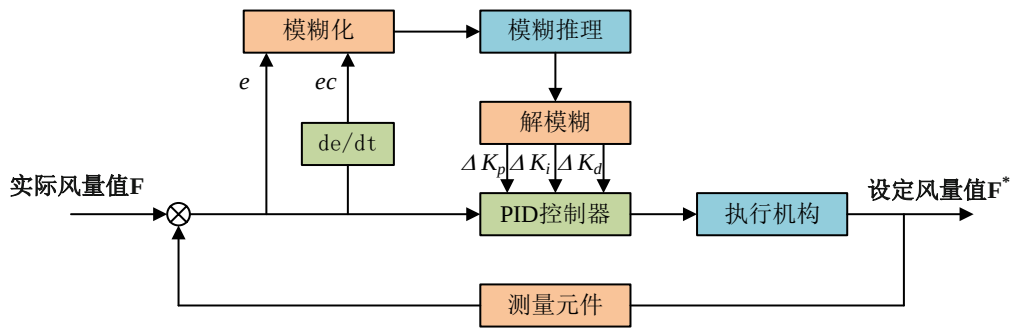


图 3-4 自适应 PID 控制原理方框图

通过对系统进行实验测试，系统的输入 e 和 ec 的基本论域分别为 $[-400, 400]$ 、 $[-800, 800]$ ，输入量的模糊论域统一取为 $[-3, 3]$ 。由于模糊控制器不能对真实语言进行数值处理，需进行离散化。将论域设置为：负大[NB]、负中[NM]、负小[NS]、零[ZO]、正小[PS]、正中[PM]、正大[PB]，这 7 个语言变量用来表示其模糊子集。将 e 和 ec 映射到论域上统一处理，根据设定的模糊子集来先确定 e 和 ec 的隶属度。通过平台电机在不同设定下的运行情况，得到 K_p 、 K_i 和 K_d 的取值范围分别为 $[0.1, 0.46]$ ， $[0.06, 0.12]$ ， $[0.016, 0.022]$ 。同样对这三个模糊输出也设置为 7 个模糊语言变量{NB, NM, NS, ZO, PS, PM, PB}。

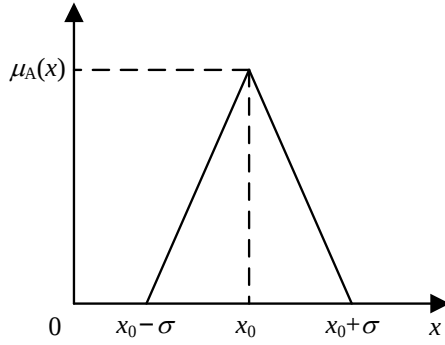


图 3-5 三角形隶属度函数

隶属度函数用于描述输入和输出变量之间的模糊关系。它将输入变量和输出变量转换为隶属度值，进行模糊推理和模糊控制。通风管道系统中风机需对风量的变化迅速做出调整，因此需要较高的精确度和较快的反应时间，三角隶属度函数因其图形简洁的特点，可对输入信号进行快速响应。因此，本文采用三角隶属度函数，如图 3-5 所示，数学表达式如下：

$$\mu_A(x) = \begin{cases} \frac{x - x_0}{\sigma} + 1 & x_0 - \sigma \leq x \leq x_0 \\ \frac{x_0 - x}{\sigma} + 1 & x_0 < x \leq x_0 + \sigma \end{cases} \quad (3-24)$$

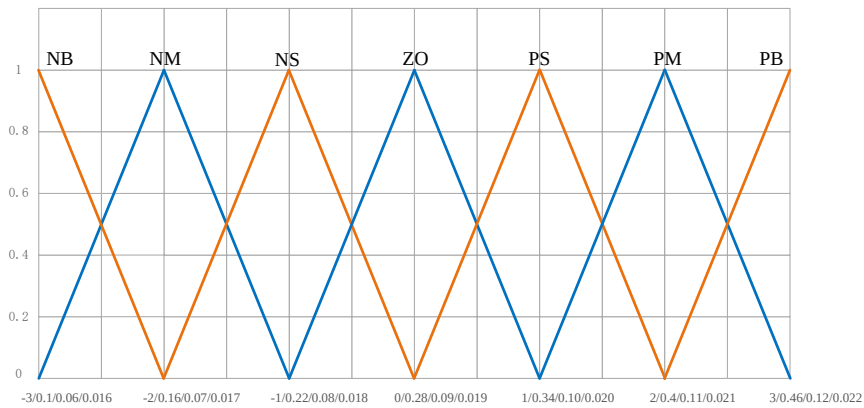


图 3-6 模糊控制论域

由公式(3-24)可知隶属度函数对模糊控制器的运算表现产生较大影响，一般而言，隶属度函数中 σ 若偏低，则所对应的三角形图像的顶角角度将减小，由此系统输出值会显著增加，反之，系统输出值会显著降低。借助上述语言变量以及三角形隶属度函数，可以对输入与输出量进行模糊分割，进而获取如图 3-6 中的七个模糊子集。

通风管道系统中，为适应系统的时变性，使系统具有最优动态响应特性，需要对 PID 参数进行在线动态整定。在风机调节前期，实际风量和目标风量之间的误差较大，为快速调节当前风量到达设定值且产生较小震荡，应选择比例系数大、积分系数小和微分系数较大的控制参数。在风机调节的过程中，实际风量正逐渐接近设定值，在这期间比例系数过大会使系统超调，积分系数过大会使系统剧烈震荡，应选择比例系数小、积分系数小的控制参数。在风机调节的后期，过小的比例系数会使系统出现静态误差，为避免这一现象出现，应当选择比例系数较大、积分系数较大的控制参数。根据以上表达式来制定 ΔK_p 、 ΔK_i 和 ΔK_d 的模糊控制规则表。如表 3-1、3-2 和 3-3 所示。

表 3-1 ΔK_p 模糊控制规则表

ΔK_p		ec						
		NB	NM	NS	ZO	PS	PM	PB
e	NB	PB	PB	PM	PM	PS	ZO	ZO
	NM	PB	PB	PM	PS	PS	ZO	NS
	NS	PM	PM	PM	PS	ZO	NS	NS
	ZO	PM	PM	PS	ZO	NS	NM	NM
	PS	PS	PS	ZO	NS	NS	NM	NM
	PM	PS	ZO	NS	NM	NM	NM	NB
	PS	ZO	ZO	NM	NM	NM	NB	NB

表 3-2 ΔK_i 模糊控制规则表

ΔK_i		ec						
		NB	NM	NS	ZO	PS	PM	PB
e	NB	NB	NB	NM	NM	NS	ZO	ZO
	NM	NB	NB	NM	NS	NS	ZO	ZO
	NS	NB	NM	NS	NS	ZO	PS	PS
	ZO	NM	NM	NS	ZO	PS	PM	PM
	PS	NM	NS	ZO	PS	PS	PM	PB
	PM	ZO	ZO	PS	PS	PM	PB	PB
	PS	ZO	ZO	PS	PM	PM	PB	PB

表 3-3 ΔK_d 模糊控制规则表

ΔK_d	ec						
	NB	NM	NS	ZO	PS	PM	PB
NB	PS	NS	NB	NB	NB	NM	PS
NM	PS	NS	NS	NM	NM	NS	ZO
NS	ZO	NS	NM	NM	NS	NS	ZO
ZO	ZO	NS	NS	NS	NS	NS	ZO
PS	PB	ZO	ZO	ZO	ZO	ZO	ZO
PM	PB	NS	PS	PS	PS	PS	PB
PS	PB	PM	PM	PM	PS	PS	PB

误差 e 和误差变化率 ec 经过模糊化处理后形成 3 个模糊集合，需要对模糊集合的数值进行相应代数运算，输出准确值进行实时控制。目前常用的解模糊化方法有面积中心法（重心法）、面积平分法、最大隶属度法^[59]等。考虑本系统需要较高的精确度，因此选择重心法较合适。重心法原理为：取横坐标与隶属度函数围成图形面积的重心为模糊推理的输出值。计算公式如下：

$$V_o = \frac{\sum_{i=0}^n M_i \times F_i}{\sum_{i=0}^n M_i} \quad (3-25)$$

式中， M 为隶属度， F 为模糊量化值， V_o 为解模糊后的输出值。将修正后的 K_p 、 K_i 和 K_d 输入至 PID 控制器中，通过 PID 控制器的输出电压实现对风机的自适应控制。

3.4 实验设计与步骤

通过上述理论研究，该部分就所提出风机自适应的分布式协同风平衡控制方法(DCAC-AB)实现步骤展开设计。对第二章中所描述的实验平台进行改造，选取内部空气耦合作用较强的管道分支验证 DCAC-AB 法的收敛性能，如图 3-7 所示。图 3-7 中最上面一排管道是系统的主管道，主管道右边最末端连接着风机，风机泵进来的空气通过主管道进入系统。主管道左边连接 90°弯头，将主管道的空气分配到五个分支管道，每个分支管道末端都安装有一个圆形智能风阀，每个风阀通过改变内部阻尼器角度来实现控制分支口出风量。图中蓝色箭头为风机输送进的空气流经的主管道和分支管道。红色注释为通风管道系统中

的智能风阀。本实验中，所有测试基于图 3-7 中蓝色箭头流经的管道，其余管道支路进行屏蔽处理，以防影响实验结果。

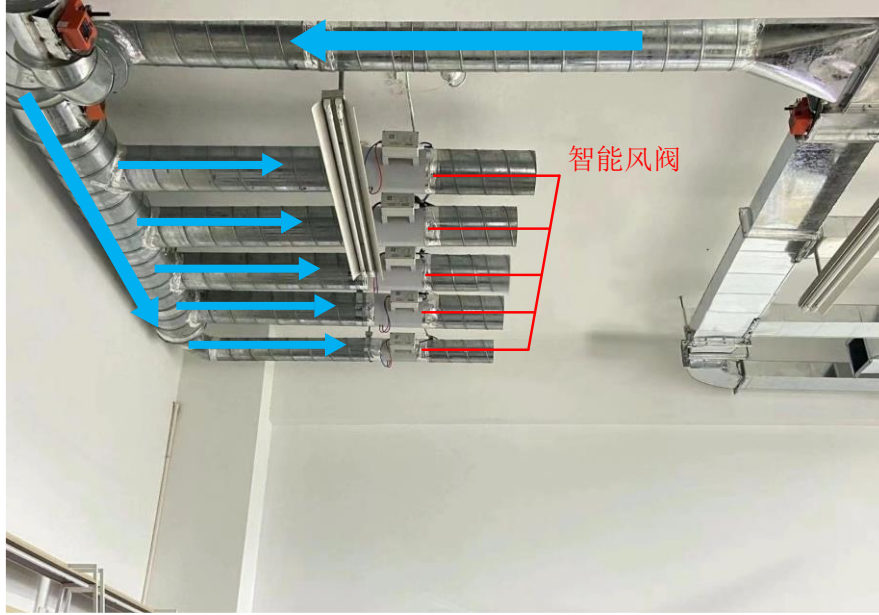


图 3-7 实验平台分支图

在进行实验之前，对风阀和风机的 Wi-Fi 模块进行检查，确保终端之间的数据传送稳定。风机控制信号连接到 NI DAQ 设备，先对风机发送起始信号，并初始化所有风阀阻尼器角度，等待各支路风量稳定后开始进行数据采集和测试。在实验过程中设定合适的时间步长，用来消除系统内支路间产生的气流震荡。实验完成的标准为各支路风量与目标风量误差在 10%以内。实验完成后保持各设备状态稳定，记录实验结果。

算法 1

算法 1: 分布式协同控制算法

Initialize target airflow F_{1-5} , fan voltage u , damper angle θ_{1-5} , β

For $ratio_1 = \dots = ratio_5$

 For i in $[1, 2, \dots, 5]$

 Adjust the i -th damper angle θ_n^* by Eq (3-15)

 Count $ratio_{1-5}$

 Wait for flowrate to stabilize

 End for

所提 DCAC-AB 方法分两阶段完成。第一阶段，由分布式协同控制算法对所有风阀阻尼器进行角度调节，通过公式(3-15)交换相邻支路间的气流信息，调整风阀阻尼器的角度，直至各支路出风口的实际空风量和目标风量比值一致

后停止调节，如算法 1 所示。第二阶段，由风机自适应算法调节风机给定电压，根据调节过程中不同状态下的 e 和 ec ，模糊规则表选出符合当前状态下的 K_p 、 K_i 和 K_d 控制参数，用以控制风机给定电压，使风机总出风量等于各支目标风量之和，如算法 2 所示。

算法 2

算法 2：风机自适应算法

Initialize fan voltage u

For u in U

Figure out e and ec

Adjust PID parameter K_p 、 K_i and K_d

Wait for the flowrate to stabilize

End for

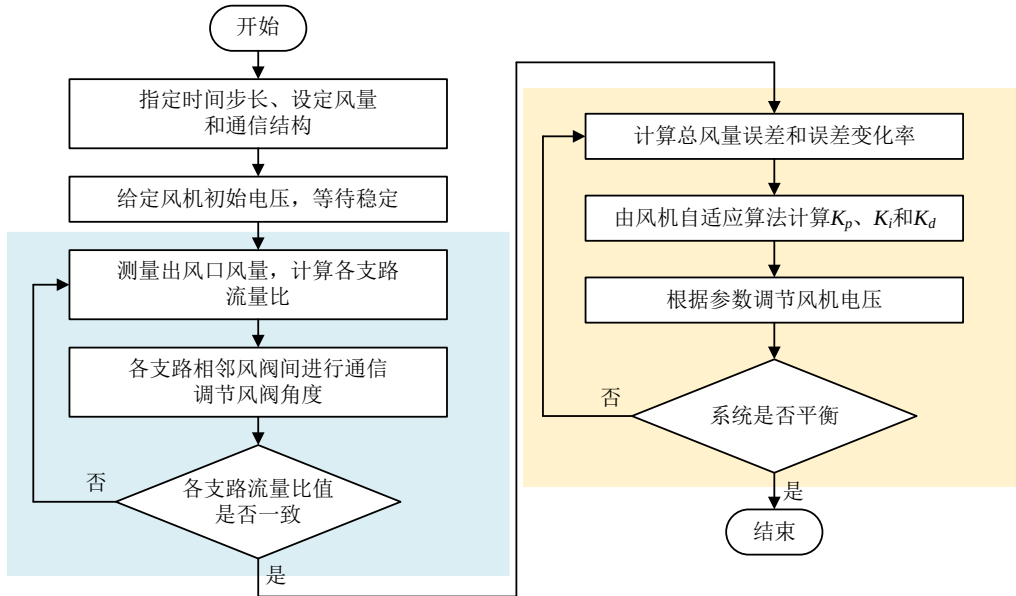


图 3-8 DCAC-AB 方法流程图

基于上述数学公式推导和实验平台结构，所提出的 DCAC-AB 方法的实现步骤流程图如图 3-8 所示，可以总结如下：

步骤 1：设定各支路空气流量 F_{1-5} ，初始所有风阀阻尼器角度 θ_{1-5} ，给定风机初始电压 u 。

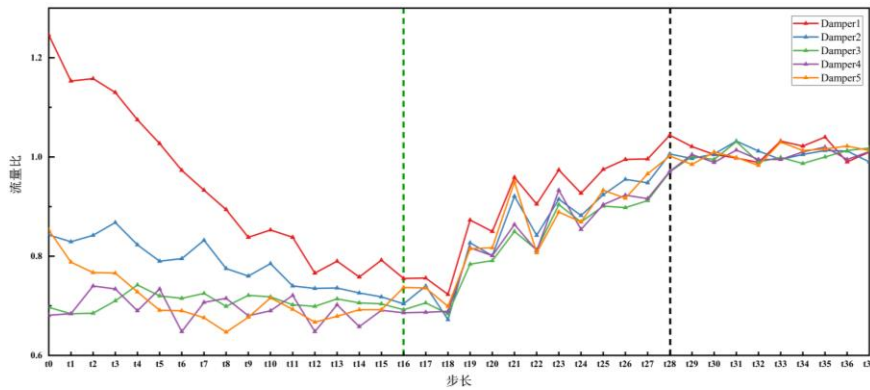
步骤 2：由算法 1 对各支路风阀阻尼器角度进行调整，等待调节完毕后得到所有风阀阻尼器角度 θ_{1-5}^* 。

步骤 3：由算法 2 对风机给定电压 u 进行调节，得到平衡状态下风机电压 u^* 。

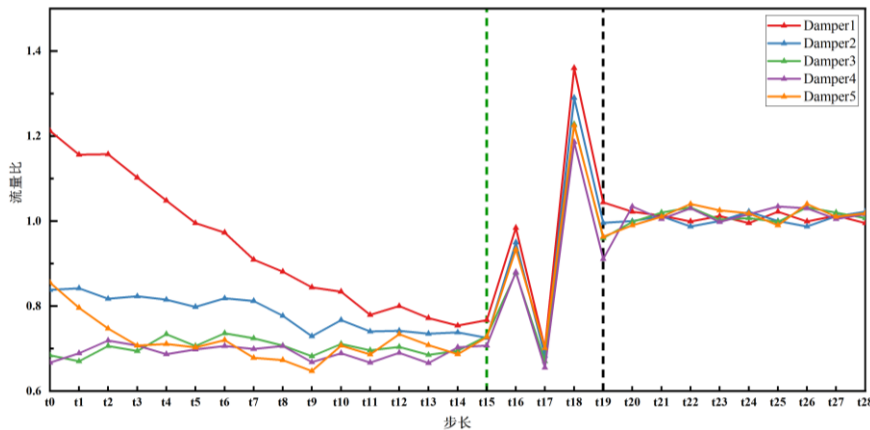
步骤 4: 风阀阻尼器角度 θ_{1-5}^* 和风机电压 u^* 保持恒定, 实现风平衡并记录实验结果。

3.5 实验验证

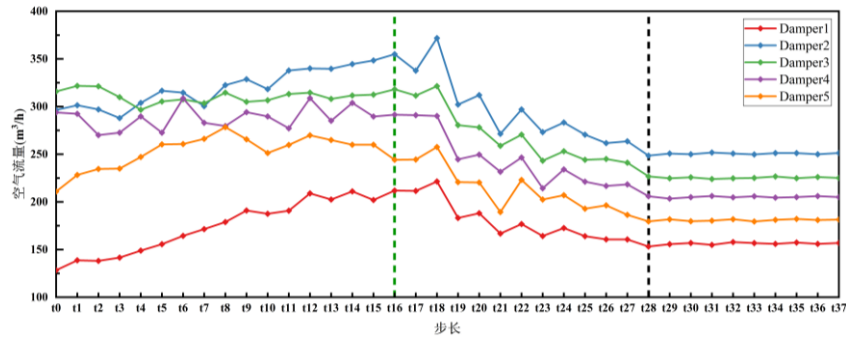
为验证本文提出的 DCAC-AB 方法的平衡收敛性能, 在实验平台进行实验验证, 并将其收敛速度与传统的分布式协同风平衡控制方法(DCC)进行了比较。表 3-4 和表 3-5 记录了各支路风量和风阀角度信息, 图 3-9 是各支路流量比和风量的平衡收敛图。表 3-4 和表 3-5 中, q_{int} 为系统各支路初始风量, q^* 为各支路目标风量, q_f 为系统平衡后的风量, θ 为各支路风阀阻尼器初始角度, θ_f 为系统平衡后各支路风阀阻尼器角度。由表 3-4 和表 3-5 可知, 两实验初始状态和各支路目标风量设置相同。对于系统风量控制的准确性, 本文采用实际风量与目标风量之间的相对误差对其进行衡量。经实验测试, 两种方法对应的最终支路风量与目标风量之间的误差均在 $\pm 10\%$ 以内, 符合 ASHRAE 误差标准。



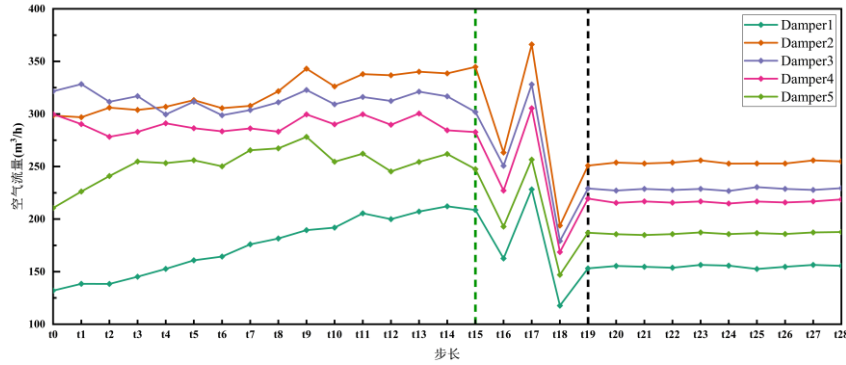
(a) DCC 法各支路空气流量比



(b) DCAC-AB 法各支路空气流量比



(c) DCC 法各支路风量



(d) DCAC-AB 法各支路风量

图 3-9 实验结果对比图

表 3-4 DCC 实验结果

Terminal	$q_{int}(m^3/h)$	$q^*(m^3/h)$	$q_f(m^3/h)$	$\theta(^{\circ})$	$\theta_f(^{\circ})$
1	128.4	160.0	156.9	45.0	19.3
2	296.5	250.0	251.3	45.0	27.8
3	315.8	220.0	225.2	45.0	34.6
4	293.9	200.0	205.1	45.0	32.6
5	210.9	180.0	181.5	45.0	20.9

表 3-5 DCAC-AB 实验结果

Terminal	$q_{int}(m^3/h)$	$q^*(m^3/h)$	$q_f(m^3/h)$	$\theta(^{\circ})$	$\theta_f(^{\circ})$
1	130.5	160.0	158.3	45.0	21.4
2	298.5	250.0	252.4	45.0	29.3
3	321.6	220.0	221.8	45.0	36.1
4	299.9	200.0	204.7	45.0	32.9
5	210.6	180.0	183.3	45.0	24.2

实验中，设定每个步长的时间为 20 秒，用以消除因风量变化引起的各支路间气流震荡。本文使用系统调节步数衡量方法的快速性。如图 3-9 中 (a)和(b)所示：DCC 控制方法在第 16 步各分支的风量比达到一致，在第 28 步各分支的流量比收敛于 1；DCAC-AB 方法在第 15 步各分支的风量比达到一致，在第 19 步

各分支的流量比收敛于 1。如图 3-9 (c)和 (d)所示：DCC 方法在第 16 步进行风机调整，由于参数不能实时更新，风机调节受固定控制参数的限制，风量只能逐步调整，共经过 28 个步长达到风量设定值。DCAC-AB 方法首先由共识算法实现风阀间的分布式协同控制，经过 15 个步长达到各支路间风量共识，再由风机自适应控制调节控制参数，使其随着风量误差的减小而实时更新，最终经过 19 个步长达到风量设定值。

表 3-6 DCC 法与 DCAC-AB 法的对比结果

控制方法	相对误差		调节时间 (min)
	最大误差(%)	平均误差(%)	
DCC 控制方法	2.55	1.64	9.33
DCAC-AB 控制方法	2.35	1.40	6.33

在相同的初始条件和步长下，DCAC-AB 方法首先对风阀进行分布式协同控制，再由自适应算法调节风机电压。由表 3-6 可知，DCAC-AB 方法经过 6.33 min 便可满足风量设定要求，而 DCC 方法需要 9.33 min 才能满足风量设定要求。通过两种控制方法的实验结果对比，DCAC-AB 方法相比 DCC 方法在收敛速度方面提升了 32.12%，是一种快速的风平衡控制方法。

3.6 本章小结

本章针对多支路通风管道系统开展了风机自适应的分布式协同风平衡控制方法研究，以解决因管道拓扑结构复杂难以收敛和平衡速度缓慢的问题。首先，本章介绍了共识算法并进行公式推导，将其应用至多支路通风管道系统中使风阀间达成一致性协议。然后，通过对模糊 PID 的分析，提出风机自适应控制算法，解决风机收敛速度慢问题。最后，在搭建的实验平台进行实验，测试所提方法的平衡收敛性能。

通过对实验平台的改造，选取耦合作用严重的支路管道进行实验，以验证所提 DCAC-AB 方法性能。本文在通风管道系统风量控制任务中以相同的初始化条件和控制目标，对比了 DCC 控制方法。实验结果发现，在 ASHRAE 标准 (10%)下，DCAC-AB 方法收敛速度明显快于 DCC 方法，是一种快速的风平衡控制方法，且该方法不需要管道系统的拓扑结构和管道配件参数，是一种无模型的方法，易应用于实际环境中。

第 4 章 基于强化学习的风平衡控制方法

4.1 引言

根据本文第三章的分析，文章所提出的风机自适应的分布式协同风平衡控制方法可以在系统正常运行时不中断服务，在线实现多支路管道系统的风量控制，且在目标风量调节过程中，有着快速的收敛性能。然而在实验室、电子厂房等环境中，由于环境需要、设备密集等原因，一些区域需要较高精度目标风量以确保安全问题，在面对大范围目标风量调整时，DCAC-AB 方法在控制精度上难以达到高标准要求。

为提高风平衡方法的泛化能力，解决实际应用中大范围风量调节精度需求，本章提出基于强化学习的风平衡控制方法。首先，本章介绍了强化学习的基本知识，包括马尔可夫决策过程及深度强化学习算法。然后，基于对深度强化学习算法的分析，根据通风管道系统风量控制问题的多解性，设计了具有马尔可夫性质的通风管道风平衡控制过程。以减小风量误差为导向设置了强化学习的状态与动作，设计基于控制误差的奖励函数。本章节提出了预训练-在线学习框架，在预训练阶段提出了单支路动态目标训练机制，减小了训练成本并提升了训练效率，增强了面对不同目标风量时的泛化能力。在线学习阶段将训练阶段完成的网络结构应用至多支路风阀控制中，通过与环境交互自适应调整网络参数，保证控制效率，提升收敛速度。最后，本章通过实验平台验证了基于强化学习的风平衡控制方法的控制性能。

本章后续内容安排如下：首先，在 4.2 小节中对强化学习的基础知识进行了介绍，内容包括了马尔可夫决策过程以及本章节所涉及到的深度强化学习算法。接着，在 4.3 小节中设计了具有马尔可夫性质的通风管道系统风量控制过程，设计单支路动态目标训练机制。4.4 小节中设计了基于强化学习的风平衡控制方法和实现步骤。最后，在 4.5 小节中验证了所提方法的控制性能。

4.2 强化学习背景知识

4.2.1 马尔可夫决策过程

马尔可夫过程又称马尔可夫链，过程中的每个状态都具有马尔可夫性质，即指下一个系统状态只与当前状态相关，而与前几次的状态没有关系。马尔可夫可由公式表示：

$$\mathbb{P}(S_{t+1} | S_t) = \mathbb{P}(S_{t+1} | S_1, \dots, S_t) \quad (4-1)$$

马尔可夫过程中状态 s 和其下一个状态 s' 之间的状态转移概率定义为：

$$P_{ss'} = \mathbb{P}(S_{t+1} = s' | S_t = s) \quad (4-2)$$

式中 P 为状态转移矩阵， P 中的元素为状态 s 到状态 s' 的转移概率：

$$P = \begin{bmatrix} P_{11} & \cdots & P_{1n} \\ \vdots & \ddots & \vdots \\ P_{n1} & \cdots & P_{nn} \end{bmatrix} \quad (4-3)$$

马尔可夫过程可以描述为一个集合 $\{S, P\}$ ，集合中 S 为马尔可夫过程中的有限状态集， P 为状态转移矩阵。马尔可夫奖励过程较马尔可夫过程多了奖励集合 R 和折扣系数 γ ，可表示为 $\{S, P, R, \gamma\}$ 。状态 $S_t = s$ 下的奖励在 $t+1$ 时刻获得。 $S_t = s$ 状态下的期望奖励可表示为：

$$R_s = \mathbb{E}[r_{t+1} | S_t = s] \quad (4-4)$$

奖励 R_t 表示在马尔可夫奖励过程中，从 t 时刻开始往后所有衰减奖励的和：

$$R_t = r_{t+1} + \gamma r_{t+2} + \dots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \quad (4-5)$$

其中， γ 为折扣系数，是一个在 $[0, 1]$ 范围内的值，用于衡量未来奖励的重要性。当 γ 接近 0 时，意味对未来奖励的贴现程度较高。相反，当 γ 接近 1 时，意味着对未来奖励的贴现程度较低，更加注重长期规划和长远利益。折扣系数的设置主要考虑以下两个方面：首先，从数学分析的角度来看，由于马尔可夫过程在没有终止状态下，其奖励值 R_t 会不断进行累加，折扣系数 γ 的引入可避免该事件发生。其次，合适的折扣系数可以降低复杂模型中不确定奖励对决策的影响。

在马尔可夫奖励过程中，会有一个终止状态结束该过程。奖励 R_t 代表了从时间 t 到终止状态的所有衰减奖励的总和。一个完整的马尔可夫奖励过程，从

初始状态到终止状态，被定义为一个回合。可以由奖励 R_t 定义状态 s 的价值函数，即从该状态开始的马尔可夫奖励过程中所获得的期望回报：

$$V(s) = \mathbb{E}[R_t | S_t = s] \quad (4-6)$$

将衰减奖励 R_t 展开可得：

$$V(s) = \mathbb{E}[r_{t+1} + \gamma V(s') | S_t = s] \quad (4-7)$$

状态转移矩阵可以表示下个状态 s' 的价值函数，即贝尔曼方程：

$$V(s) = R_s + \gamma \sum_{s' \in S} P_{ss'} V(s') \quad (4-8)$$

马尔可夫决策过程（Markov Decision Process, 简称 MDP）定义为 $\{S, A, P, R, \gamma\}$ ^[60]，较之马尔可夫奖励过程多了一个动作集合 A 。所以 MDP 过程中的状态转移概率与奖励函数都有动作集合 A 参与，可表示为：

$$P_{ss'}^a = \mathbb{P}[S_{t+1} = s' | S_t = s, A_t = a] \quad (4-9)$$

$$R_s^a = \mathbb{E}[R_{t+1} | S_t = s, A_t = a] \quad (4-10)$$

策略 $\pi(a|s)$ 定义为在状态 s 下采取动作 a 的奖励，即：

$$\pi(a | s) = \mathbb{P}[A_t = a | S_t = s] \quad (4-11)$$

MDP 中从状态 s 开始根据策略 π 选择动作 a 所获得的期望奖励为价值函数，记为 $V_\pi(s)$ ，可表示为：

$$V_\pi(s) = \mathbb{E}_\pi[R_t | S_t = s] \quad (4-12)$$

策略 π 下的状态价值函数可用贝尔曼方程表示为：

$$V_\pi(s) = \sum_{a \in A} \pi(a | s) \left(R_s^a + \gamma \sum_{s' \in S} P_{ss'}^a V_\pi(s') \right) \quad (4-13)$$

状态-价值函数 $Q_\pi(s, a)$ 表示在给定策略 π 的情况下，从状态 s 开始，选择指定动作 a 后，所有动作遵循策略 π 能够获得的期望奖励。 $Q_\pi(s, a)$ 可以评估在状态 s 下选择动作 a 的好坏程度。 $Q_\pi(s, a)$ 表示为：

$$Q_\pi(s, a) = \mathbb{E}_\pi[R_t | S_t = s, A_t = a] \quad (4-14)$$

状态价值函数 $V_\pi(s)$ 与状态-动作价值函数 $Q_\pi(s, a)$ 之间的关系可用贝尔曼方程表示为：

$$Q_{\pi}(s, a) = R_s^a + \gamma \sum_{s' \in S} P_{ss'}^a V_{\pi}(s') \quad (4-15)$$

根据上述公式可得，策略 π 对于状态 s 下所能获得的期望奖励有重要影响。策略 π 更新过程实为寻找在新策略 π' 下所获得期望奖励不低于原策略 π 的过程，即：

$$V_{\pi'}(s) \geq V_{\pi}(s), \forall s, \pi \quad (4-16)$$

因此在 MDP 中，至少存在一个策略 π 使获得的期望奖励不小于任一策略 π ，称该策略为最优策略 π^* ，其对应的状态价值函数也称之为最优状态价值函数，表示为：

$$V_{\pi^*}(s) = \max_{\pi} V_{\pi}(s), s \in S \quad (4-17)$$

其对应的贝尔曼最优方程为：

$$V_{\pi^*}(s) = \max_a \left(R_s^a + \gamma \sum_{s' \in S} P_{ss'}^a V_{\pi^*}(s') \right) \quad (4-18)$$

同理，最优策略 π^* 下对应的最优状态-动作价值函数 $Q_{\pi^*}(s, a)$ 定义为：

$$Q_{\pi^*}(s, a) = R_s^a + \gamma \sum_{s' \in S} P_{ss'}^a \max_a Q_{\pi^*}(s', a') \quad (4-19)$$

图 4-1 以图形化的方式表示了贝尔曼最优方程(4-18)与(4-19)。

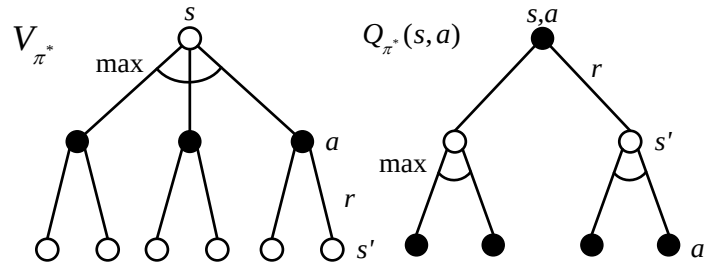


图 4-1 最优状态价值函数与最优动作-价值函数回溯图

由于最优状态-动作价值函数给出了任意状态 s 下，执行每个动作 a 的期望回报，故最优策略 π^* 为选择使 $Q_{\pi^*}(s, a)$ 值最大的动作。根据状态-动作价值函数求得最优策略表示为：

$$\pi^* = \arg \max_a Q_{\pi^*}(s, a) \quad (4-20)$$

在基于模型的 MDP 中，若已知环境的状态转移矩阵和建立函数，则可以通过动态规划来求解最优策略。但实际应用中，由于环境状态的未知性，获取准确的状态转移矩阵变得极为困难。如通风管道系统中，管道内部压力值以及

各支路风量数据的准确度都存在一定程度的不确定性。针对此类问题，需要使用基于无模型的算法以求解最优策略。通过与环境进行互动，不需要了解环境变化规律，算法根据经验进行学习，最终获得最优策略。

4.2.2 深度强化学习算法

强化学习中，智能体是指与环境进行交互并获得奖励的实体。在强化学习的过程中，智能体从环境处获取当前状态 s_t ，根据 s_t 采取相应动作 a_t ，而后环境反馈奖励 r_t 并转移到新的状态 s_{t+1} 。智能体通过最大化累积奖励 r_t ，学习出最优策略 π^* 。图 4-2 为强化学习智能体与环境交互的过程。

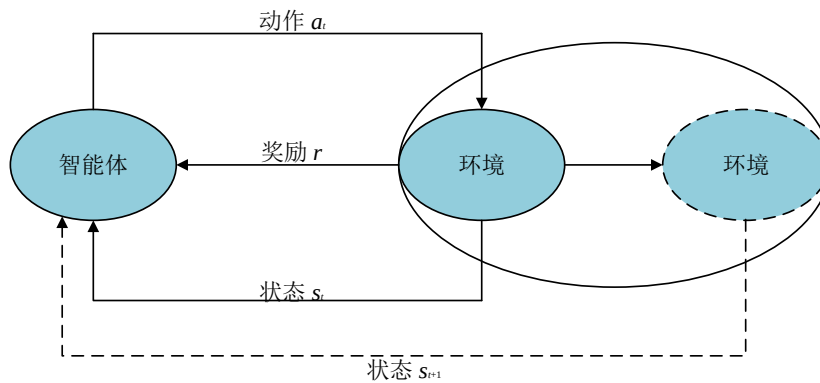


图 4-2 智能体与环境交互过程

在 Q-Learning 中，状态-动作价值函数 $Q_\pi(s, a)$ 使用一个固定维度的表格存储，其中每个条目中的 Q 值在每次迭代运算中更新^[61-63]。在 Q-Learning 强化学习中主要存在两个缺点：（1）表格强化学习在处理状态空间较大或连续动作问题时，需要使用大量内存来存储状态-动作价值函数，这导致存储和计算的开销非常高，并且难以找到足够的训练样本来覆盖整个状态空间；（2）表格强化学习缺乏估计价值的能力，在与环境交互时，只有探索到对应的状态-动作对时，才能对 $Q_\pi(s, a)$ 进行更新。

为解决 Q-Learning 面对高维状态-动作空间时难以进行控制和决策的缺点，强化学习中引入深度神经网络，称为深度强化学习。深度强化学习具备直接从高维数据中提取低维特征的能力，且拟合复杂系统输入和输出之间的非线性关系，从而有效解决高维状态空间中由于状态空间过大导致难以决策的问题。

4.2.2.1 Deep Q Network 算法

Deep Q Network (DQN)^[64]算法中使用卷积神经网络来逼近状态-动作价值函数，即 Q 值函数，用来处理高纬度状态信息，并共享权重来降低模型复杂度。 Q 值函数通过参数为 θ 的 Q 网络输出代替，可表示为：

$$Q(s, a, \theta) \approx Q_{\pi}(s, a) \quad (4-21)$$

在 DQN 中，神经网络非线性逼近值函数时，会造成算法不稳定或不收敛问题。为解决这一现象，在深度 Q 网络中加入经验回放机制。该机制其核心思想就是保存智能体在环境中的经验，在训练过程中重复使用这些经验^[65]。具体来说，经验回放缓存区会存储智能体在环境中的状态、动作、奖励以及下一个状态等信息。当智能体进行训练时，会从这些经验中随机抽取一部分进行训练，以此来提高训练效果。通过经验回放机制，智能体可以更好的利用之前的经验，避免了只依赖当前经验的模型过拟合问题。

在训练过程中将产生的经验数据 (s, a, r, s') 存放到经验池中，智能体可以再次使用这些数据信息，可以使数据满足独立分布。由于经验池中任意两个连续的经验数据之间呈弱相关，为确保系统能够快速稳定收敛且防止相邻状态数据间的连续性， Q 网络每次训练从经验池中随机抽取经验数据。由于神经网络梯度下降不会尝试不同方向，引入经验池后，随机抽取数据，再根据抽取数据求梯度，可有效避免单一方向的梯度下降问题。

在训练模型时，从经验池中随机抽取数据时一般选择均匀采样法。存储的样本序列都是源于智能体与环境交互的数据，在时间上存在相关性，然而值函数表示的是执行该动作得到的累积奖励，但每次的样本信息只是当前一次的采样过程，不代表所有轨迹，所以得到的估计值和目标值会有误差。随着训练时间的增加，产生的误差也会变大。

在深度强化学习训练时，使用一个神经网络来估计状态-动作价值函数。如果直接使用当前神经网络计算目标 Q 值，由于神经网络的参数更新是基于当前策略的采样数据，会导致目标值的变化，从而引发训练不稳定的问题，即算法可能会发散或收敛到次优解。DQN 算法中引入目标网络(Target Network)，该目标网络目标函数为 $Q(s, a, \theta^-)$ ，其中 θ^- 为目标网络参数。在 DQN 中会使用到两个主要网络：估计网络和目标网络。估计网络用于计算当前状态下 Q 值，而目标网络用于计算目标 Q 值。目标网络中权重的更新频率相对较低，一般不会对

每个训练步骤中更新。在训练过程中，会周期性地从 Q 网络拷贝权重信息。这种延迟更新的方式可以减少目标值的波动，提高算法稳定性。

DQN 算法的值函数 Q 是一组向量，用 θ 表示神经网络权值，用 $Q(s, a, \theta)$ 表示当前网络的输出，目标网络的输出用 $Q^-(s_{t+1}, a_{t+1}, \theta^-)$ 表示。其中，当前网络参数表示为 θ ，目标网络参数表示为 θ^- 。DQN 目标函数的计算公式定义为：

$$Y_t^{DQN} = r + \gamma \max_a Q(s_{t+1}, a; \theta^-) \quad (4-22)$$

在目标网络中， $\max_a Q(s_{t+1}, a, \theta^-)$ 表示在下一状态 s_{t+1} 中所有可选动作中，选择具有最大 Q 值的动作 a ，其计算是基于目标网络的参数 θ^- 完成。

利用状态价值函数迭代计算 Q 值函数，通过 Q 值函数更新当前网络的权重参数，在经过一定周期后将估计网络权重拷贝更新至目标网络。通过这种不断拷贝更新迭代达到目标网络更新的目的。更新如下：

$$Q(s, a; \theta) = (1 - \alpha)Q(s, a; \theta) + \alpha \left(r + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta^-) \right) \quad (4-23)$$

式中， α 是学习率。学习率 α 的值会直接影响梯度下降过程中更新参数的速度和方向。在一定数值范围内，较大的 α 会导致参数变化幅度大，从而导致训练的不稳定性或发散现象的产生；而较小的 α 则会使得参数变化幅度小，相应的训练过程的收敛速度也会变慢。DQN 算法中通常使用较小的学习率，以确保稳定的训练过程。此外，还可以采用学习率衰减策略，逐渐降低学习率，以进一步提高算法的稳定性和收敛性。

神经网络中的参数会随着反复迭代过程进行更新，直到损失函数的值最小化。DQN 中通过反向传播实现梯度更新。通过计算当前策略网络的预测 Q 值与目标 Q 值之间的差异，计算损失函数。损失函数计算如下：

$$L_i(\theta_i) = \mathbb{E} \left(r + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta_{i-1}) - Q(s_i, a_i; \theta_i) \right)^2 \quad (4-24)$$

然后将损失函数对网络参数 θ 求偏导，计算公式如下：

$$\frac{\partial L(\theta_i)}{\partial \theta_i} = \mathbb{E} \left[\left(r + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta_{i-1}) - Q(s_i, a_i; \theta_i) \right) \nabla_{\theta} Q(s_i, a_i; \theta_i) \right] \quad (4-25)$$

DQN 算法中使用两个神经网络模型来处理时间差分中的 TD 误差。这种方法可以解决相关性参数的问题，有利于算法收敛。具体来说，估计网络用于预测 Q 值函数，根据当前状态和动作预测下个状态的最大价值，并将其与实际获

得的奖励进行比较，计算 TD 误差。而目标网络用于计算目标 Q 值，其神经网络参数会滞后一段时间，以减少目标函数和实际值间的相关性，从而提高算法稳定性。DQN 算法结构图如图 4-3 所示。

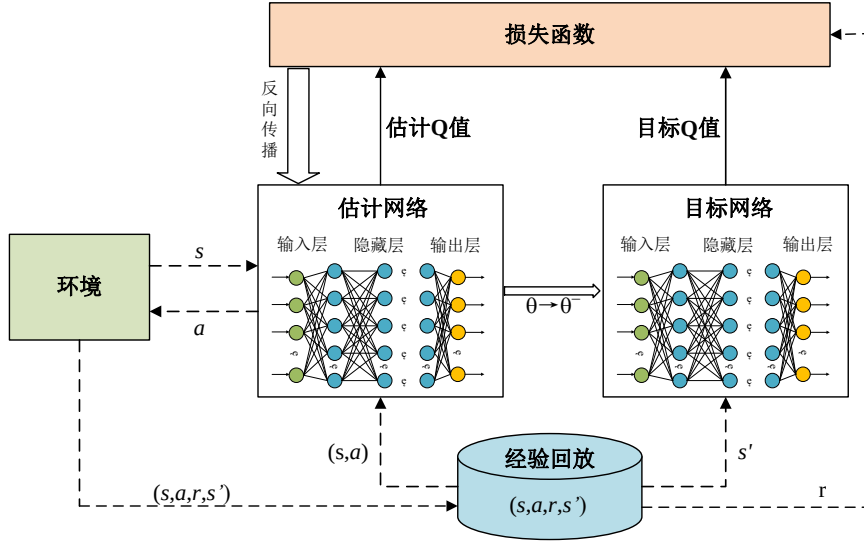


图 4-3 DQN 算法结构图

4.2.2.2 Double Deep Q Network 算法

DQN 算法在选择动作时容易高估 Q 值。这是由于 DQN 算法中选择最大值来使 Q 值更接近可能的目标，而目标网络参数的滞后更新导致目标网络无法及时跟上策略网络的更新。当策略网络参数更新很快时，目标网络的更新滞后，可能导致估计的 Q 值过高，进而影响策略的选择和决策过程。为了解决这个问题，Double Deep Q Network (Double DQN) 算法^[66]将选择动作与 Q 值的评估解耦。通过交替更新两个函数中的一个，Double DQN 可以学习到两个值函数，如两个 Q 网络，并得到两组网络权重： θ 和 θ^- 。其中， θ 用于确定要采取的动作， θ^- 用于计算目标 Q 值。通过估计选取最大 Q 值对应的动作，公式表达如下：

$$a_{\max} = \arg \max_a Q(s_{t+1}, a; \theta) \quad (4-26)$$

然后由目标网络根据 $a_{\max}$ 计算目标 Q 值。Double DQN 目标函数计算公式如下：

$$Y_t^{\text{DoubleDQN}} = r + \gamma \max_a Q\left(s_{t+1}, \arg \max_a Q(s_{t+1}, a; \theta); \theta^-\right) \quad (4-27)$$

Double DQN 中，首先使用当前策略网络 θ 选择在给定状态 s 下的最佳动作 a 。然后，利用目标网络 θ^- 计算在下一个状态 s_{t+1} 下，根据当前策略网络 θ 选择的动作 a 所得到的 Q 值。通过这种方式，选择动作和计算 Q 值的过程解耦开来，

避免了选择动作时对 Q 值的高估问题。在训练过程中，Double DQN 交替更新当前策略网络 θ 和目标网络 θ^- 的参数。通过比较预测 Q 值和目标 Q 值的差异，使用梯度下降等优化算法来更新当前策略网络 θ 的参数。定期地将当前策略网络的参数复制到目标网络中，以保持两个网络的一致性。Double DQN 算法结构图如图 4-4 所示。损失函数计算如下：

$$L_i(\theta_i) = \mathbb{E} \left(\omega_i \left(Y_t^{\text{DoubleDQN}} - Q(s, a; \theta_i) \right)^2 \right) \quad (4-28)$$

式中 ω_i 为权重系数，用于调整每个样本重要性。

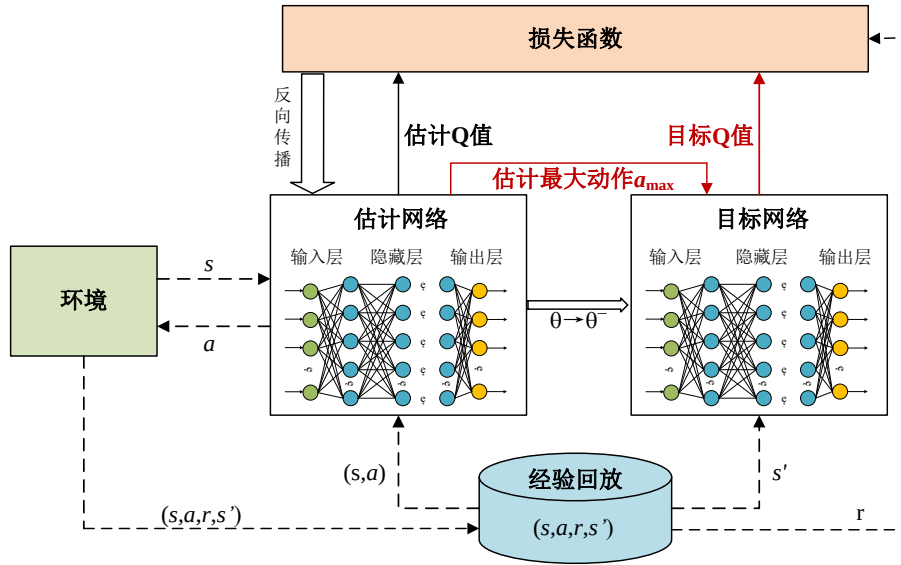


图 4-4 Double DQN 算法结构图

4.2.2.3 探索和利用

探索和利用是应用强化学习需要解决的两个重要问题。探索是选择当前非最优的行动，并探索新的行动。这样可以增加行动值，以获取长期下更多的奖励。而利用则是从已经获得的经验知识中选择那些当前认为是最优的行动，以获取短期内更多的奖励。常见的方法有 ϵ -贪心法、 ϵ -递减策略和贪心法等。

大多数强化学习算法都采用 ϵ -贪心法，其概率随时间的推移而降低，探索逐渐转向利用。 ϵ -贪心法会以 $1-\epsilon$ 的概率选择当前已知的行动（利用），以 ϵ 的概率随机行动（探索）。当进行最后一轮探索时，它变成一种贪心法并选择最优动作。基于此策略的动作选择如下：

$$a = \begin{cases} \text{random} & (0, \epsilon) \\ \arg \max_{a'} Q(a') & [\epsilon, 0) \end{cases} \quad (4-29)$$

通过这种方式， ε -贪心法可以在一定程度上平衡探索和利用，以保证对未知状态和行动的探索，同时又能够利用已有的经验知识获得更高的奖励。

4.3 基于强化学习的风平衡控制方法

根据第二章对系统模型的分析，多支路的通风管道系统具备连续输入、多输出、强耦合的物理特性。系统的风量控制具备多解的性质，这导致其快速、准确控制及其困难。Double DQN 算法利用神经网络拟合控制策略与价值函数。通过设计预训练-在线学习的强化学习框架，实现对风量的精准控制。通过使用经验回放缓存区来存储之前的经验，加快训练速度。因此，Double DQN 可应用于解决具备非线性、强耦合特征的多支路通风管道系统风量控制问题。

4.3.1 基于强化学习通风管道系统 MDP 建模

智能体与通风管道系统环境交互的过程如图 4-5 所示。在交互过程中，通风管道系统在 t 时刻的状态 s_t 被智能体所观测，在该状态下智能体选出动作 a_t 与系统进行交互，系统在完成动作 a_t 后得到新状态 s_{t+1} ，并将新状态传递给智能体，智能体由 s_{t+1} 得到奖励 r_{t+1} 。根据通风管道系统的物理特性及运行原理，设计状态、动作、奖励和终止条件如下。

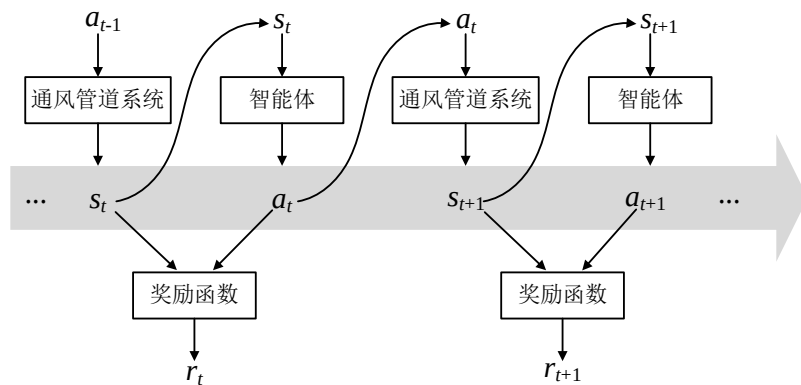


图 4-5 智能体与通风管道系统交互过程

4.3.1.1 状态特征

状态表述了通风管道系统各支路环境特征。状态空间涵盖了所有系统支路状态可能性的集合。若智能体决策时所涉及的状态空间较小，可用 Q-learning

中的表格法来记录状态信息。然而，实际问题中通风管道系统具有巨大且连续的状态空间。通过研究，通风管道系统的状态特征的选择应遵循以下原则：

- a. 状态特征可以描述通风管道支路的主要特征和变化。
- b. 状态特征的选择应与该支路的设定目标相关，否则会导致系统状态特征冗余。
- c. 对不同支路风平衡问题的状态可以用一个共同特征集表示。

为了充分利用深度学习从原始输入提取特征，设计一种通风管道系统的状态表达法，即利用当前分支管道风量信息作为输入。这里的风量信息包括当前风量与目标风量间的差值和目标风量状态。以风量值间的差值作为输入不仅可以反映当前支路风量的变化，且解决了不同目标风量需要不同的调节策略这一难题，可满足多风阀间的协同控制。

由于风压和管道间空气耦合作用影响，面对不同风量设定值，风阀阻尼器不同角度的变化引起的震荡大小也不一样。即当某一支路风量设定值偏大时，阻尼器进行较小的角度调节便会引起较大的风量震荡，当某一支路风量设定偏小时，阻尼器进行较大的角度调节才能引起较小的风量震荡。故针对不同的风量设定值来制定风量等级，以此来降低模型训练复杂度。状态空间可表示为：

$$S_t = [\Delta F, F_s]^T \quad (4-30)$$

式中， ΔF 为当前风量与目标风量间的差值， F_s 为当前目标风量等级。当系统进行风阀调节阶段时，计算出当前的风量差值，并根据设定风值分配对应的风量等级。系统进入新的状态，更新当前风量差值。

状态 ΔF 的正负可以反映出当前状态下阻尼器需要调整的方向。当 ΔF 为正时，当前风量超过目标风量，阻尼器需增大调节角度，当 ΔF 为负时，当前风量未达到目标风量，阻尼器需减小阻尼器角度。此外， ΔF 还可以反映出调节角度的大小。当 ΔF 的绝对值较大时应当大幅度调整阻尼器角度，绝对值较小时应当较大幅度调节。

由于在不同的风量状态下，阻尼器角度变化的幅度可能引起不同程度的风量震荡。以两个不同目标风量值为例：若系统的两条支路风量设定值分别为 $100 \text{ m}^3/\text{h}$ 、 $500 \text{ m}^3/\text{h}$ ，此时两支路的实际流量与设定值偏差都为 $50 \text{ m}^3/\text{h}$ 。设定值为 $100 \text{ m}^3/\text{h}$ 的支路中阻尼器可能进行 5° 的调节才能收敛于设定值，而设定值为 $500 \text{ m}^3/\text{h}$ 的支路中阻尼器可能只需进行 1° 的调节便能收敛于设计值。因此在模

型训练时，对于风量设定值小的支路，风阀阻尼器的调整应该谨慎些，而对于风量设定值大的支路，风阀阻尼器的调整要大胆些。通过对实际环境需求风量研究和针对系统的实验测试，将风量设定值制定为五个区域，(0-100]，(100-200]，(200-300]，(300-400]，(400-500]为别对应 1，2，3，4，5 等级。

4.3.1.2 动作空间

在风平衡控制中，动作是根据当前支路风量状态做出的风阀阻尼器角度调节。动作空间由各种不同的调节角度组成。为了实现快速且准确的风量调节，智能体需要在不同的状态下为不同的风量等级选择合适的角度调整。所指定的动作空间应遵循以下标准：

- a. 角度调整应该是多样性的，在不同的状态下选择最合适的调节角度，以此来缩短风平衡收敛时间。
- b. 动作集合内的元素不能过多，由于管道系统间的耦合效应，过多的动作元素会使模型训练出错误的动作策略。

风阀阻尼器角度空间可表示为：

$$a = [\theta_1, \theta_2, \dots, \theta_n]^T \quad (4-31)$$

式中集合的元素表示风阀阻尼器的角度增量。为实现快速且准确的风平衡状态，需要风阀阻尼器根据当前状态做不同角度的调整。与设定风值相差较大时，应当大范围调整角度；与设定风值接近时，应当小范围调整角度。通过对系统进行测试，将动作空间设定为(-10, -5, -2, -1, 0, 1, 2, 5, 10)。

在实际通风管道系统中，风阀阻尼器的角度只能在有限空间内调节，为避免得到的输出动作使风阀运行故障，对其动作调节范围设置有效空间：

$$\theta_i = \begin{cases} \theta_i + a & \theta^{\min} \leq \theta_i + a \leq \theta^{\max} \\ \theta^{\min} & \theta_i + a < \theta^{\min} \\ \theta^{\max} & \theta^{\max} < \theta_i + a \end{cases} \quad (4-32)$$

公式(4-32)表示当控制动作使风阀角度超过其工作范围后，该智能体输出的控制动作无效。通过第二章对智能风阀的介绍，风阀阻尼器的活动角度为 0° - 90° ，故将 $\theta^{\min}$ 设置为 0° ， $\theta^{\max}$ 设置为 90° 。

4.3.1.3 奖励函数

奖励函数定义了风阀在系统中的目标和行为的优劣程度。通风管道系统中智能体的目标是通过最大化累积的奖励来学习采取最优的角度调节策略，因此，奖励函数直接影响智能体的学习和决策过程。一个合适的奖励函数能够引导风阀朝着期望的行为方向进行角度调节，同时避免不良的角度调节。不恰当的奖励函数可能导致智能体学到错误的策略。在制定奖励函数时，需考虑以下几个方面：

- a. 清晰明确的目标：奖励函数应该对预期的目标行为给予高度正向的奖励，而对违反目标的行为给予较大的负向奖励，这样可以准确地衡量每个动作质量。
- b. 稀疏性和稳定性：奖励信号应该尽可能简洁明了，避免过度稀疏或过于频繁地奖励信号。稀疏的奖励信号可能导致智能体无法学习到有效的策略，而过于频繁的信号则可能造成学习不稳定。
- c. 奖励函数地可调节性：奖励函数应该具有一定的可调节性，便于根据实际需求进行调整和优化。

在通风管道系统中，将 t 时刻风量误差值与 $t-1$ 时刻风量误差值的变化作为 t 时刻的奖励函数，即动作前的风量误差的绝对值减去动作后风量误差的绝对值。奖励函数可表达为：

$$r = (|F_{t-1} - F^*| - |F_t - F^*|) \quad (4-33)$$

式中， F_t 为 t 时刻支路风量， F^* 为该支路目标风量。只有当误差值（实际风量与目标风量的差值）变小奖励为正向奖励，否则为负向奖励。且每次调节后越是接近目标值，得到的正向奖励越大，反之，越是偏离目标风值，得到的负向奖励越大。该种奖励函数表达式使控制策略只需考虑向误差值越来越小的方向调整，提高模型的泛化性。

4.3.1.4 终止条件

在进行控制策略的学习过程中，需合理地设置终止条件来判定回合结束以便进行训练。终止条件的设置对控制策略的学习效果和效率其重要作用。在强化学习中，通常以智能体与环境交互回合最大步长作为边界。当智能体完成控制目标后，根据终止条件的设置提前结束回合。

在设置终止条件时，最大步长 T 应考虑控制问题的复杂程度。当面对复杂问题时，执行一个完整的任务可能需要较长的时间。在这种情况下，设置较大的最大步长 T 可以给智能体足够的时间去探索和学习。对于易完成的任务，过大的步长设置会降低智能体学习效率，当交互步长大于设置步长后，应结束该回合：

$$\text{reset} = \begin{cases} \text{true} & t > T \\ \text{flase} & t < T \end{cases} \quad (4-34)$$

式中， t 表示当前步数， T 为回合允许最大步长。

当前的控制效果也是策略学习过程中每回合结束的重要依据。当智能体提前完成控制目标后，与系统的继续交互无法再为策略更新提供有益信息，此时可提前终止当前回合，以提高控制策略的训练效率。在通风管道系统中，以风量误差来判断是否达到控制目标。风量误差计算公式为：

$$e_t = \frac{|F_t - F^*|}{F^*} \quad (4-35)$$

当风量误差 e 小于设定误差时，结束该回合。因此，本文设计带有任务导向的终止条件 $IsDone$ 。 $IsDone$ 设置为：

$$IsDone = \begin{cases} 1 & e_t < E \\ 0 & \text{其他} \end{cases} \quad (4-36)$$

式中， E 为设定误差。ASHRAE 提出的风量控制误差为 10% 范围内，为提高模型训练准确率，将设定误差设置为 3%。

智能体结束当前回合的条件设置为：

$$\text{reset} = \begin{cases} \text{true} & (t > T) \text{ 或 } (IsDone = 1) \\ \text{flase} & \text{其他} \end{cases} \quad (4-37)$$

该式表示当智能体与系统交互达到设置最大步长或满足目标风量后结束该回合。

4.3.2 单支路动态目标训练机制

在模型预训练阶段，Double DQN 算法通过对估计网络和目标网络进行训练。两个网络的初始权值完全相同，为随机值。目标网络的权值定期从估计网络拷

贝。在每一回合中，智能体在通风管道系统随机起始状态开始的情况下通过控制策略达到设定风量目标。

对于多支路通风管道系统，不同系统其分支数量不同，每条分支所需风量也会不同，且所需风量也会随区域需求动态变化，这提高了多支路系统的控制策略要求，尤其对于目标风量复杂的通风系统。预训练阶段智能体所学到的控制策略将被用作在线学习阶段实际风量控制中，因此，提高智能体在预训练阶段的训练效率，可以有效缓解在线学习阶段的参数敏感问题。

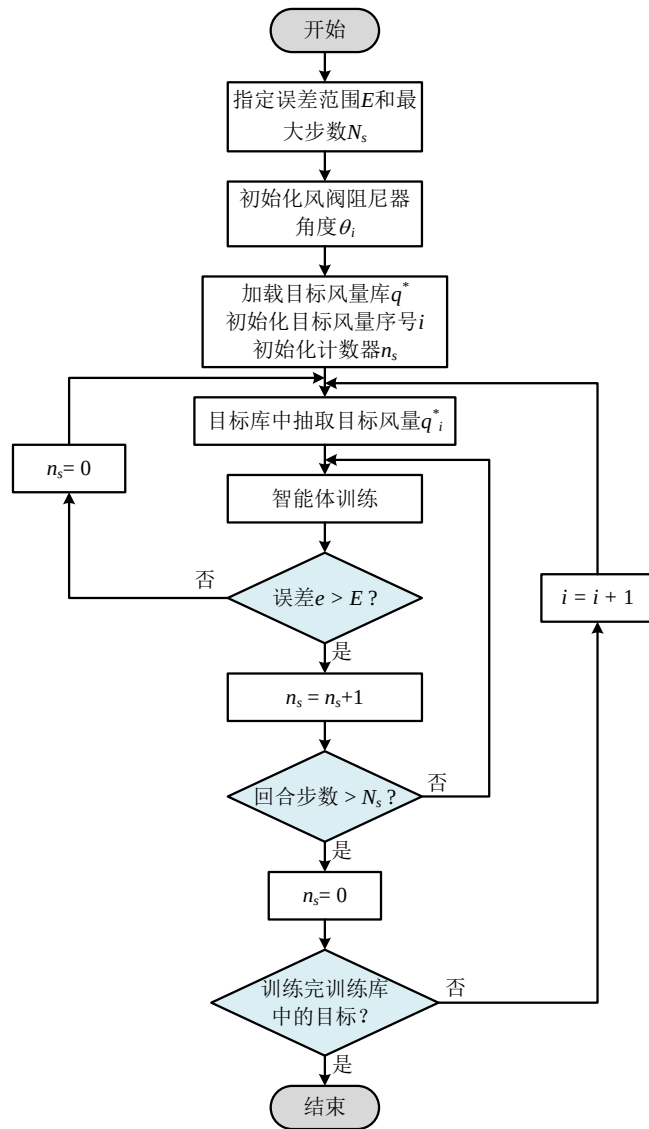


图 4-6 单支路动态目标训练机制流程图

本文提出了一种单支路动态目标训练机制。该训练机制在智能体训练过程中，以一条支路为训练对象，每一回合中更换不同的目标风量，再将训练完成的策略应用至所有支路中，以提高智能体的训练效率，提高智能体在应对不同

拓扑结构和不同目标风量时的泛化能力，使智能体可以从任意的系统状态开始，完成不同的设定标准。

具体而言，在智能体训练之前，建立一个旨在影响预训练效率的目标风量库。库中的目标风量应多样性和合理性，避免目标风量集中于某一值附近，且库中风量范围对应实际环境中所需风量范围。在训练中，逐步随机挑选目标风量库中的值进行训练。

在训练初始时随机选择风阀阻尼器角度作为开始状态，在训练过程中随机更换目标风量作为收敛目标。考虑到实际调节过程中支路间的耦合作用，其他支路风阀初始角度固定为 45° 。为保证智能体能够朝目标方向快速、准确收敛，设置误差范围和最大步数 N_s 。

所提出的单支路动态目标训练机制的具体流程图如图 4-6 所示。设置误差范围 $E=3\%$ 和最大步数 $N_s=30$ 。初始风阀阻尼器角度 θ_i 从 $[0 - 90^\circ]$ 中随机抽选。加载目标风量库 q^*_i ，初始化目标风量序号 i 。初始化计数器 n_s 记录智能体调节次数，用于判断智能体能否在最大步数内满足目标风量 q^* 的调节。当测得误差 e 小于设定误差 E 即满足收敛条件。此时，智能体更换下一目标风量，目标风量序号 i 加 1。否则，继续当前目标训练。

4.4 实验设计与步骤

本章节采用第二章介绍的通风管道实验平台对所提出的基于强化学习的风平衡控制方法(DRL-AB)进行预训练和在线学习。智能体与实验平台进行实时交互，观测系统状态由智能体给出风阀控制动作。由所提出的单支路动态目标训练机制对模型进行预训练。强化学习模型由 PyCharm 搭建，在配有 Inter(R)Core i7-10700、64GB RAM 和 NVIDIA GeForce RTX3090 台式计算机上进行。

预训练过程：

由于在实际环境中不同的系统会有不同数量的通风出口，给风平衡控制方法带来了极大的限制性，控制方法难以部署至任意通风管道系统中。考虑各区域的需求风量变化大，为提高控制策略对不同管道结构系统的泛化能力，提出单支路动态目标训练机制对模型进行训练。

算法 3

算法 3: Double DQN 算法

//初始化 //

Initialize Q-network randomly with weights θ Initialize target network Q^- with weights $\theta^- = \theta$ Initialize memory buffer B with a size of C , greedy probability ε , size of mini-batch samples n , discount rate γ , learning rate α and update period K of target network

// 预训练过程 //

For episode = 1, ..., M do Initialize the tarin state s_0 For $t = 1, \dots, T$ do Get ventilation duct system state s_t Select a random action a_t with a probability ε Otherwise select $a_t = \arg \max_{a'} Q(a')$ Execute action a_t in ventilation duct system and observe reward r_t , next state s_{t+1}

If air balancing

Break

 Store transition (s_t, a_t, r_t, s_{t+1}) in B Sample random mini-batch n of transitions (s_j, a_j, r_j, s_{j+1}) from B

$$\text{Set } Q = \begin{cases} r_j & \text{for terminal } s_{j+1} \\ r_j + \hat{Q}(s_{j+1}, \arg \max_a Q(s_{j+1}, a; \theta_t), \theta^-) & \text{for non-terminal } s_{j+1} \end{cases}$$
 Perform a gradient on $(Q_j - \hat{Q}(s, a; \theta))^2$

End for

End for

// 在线学习过程 //

Initialize the tarin state to s_0 For i in I

$$\text{Set } Q = \begin{cases} r_i & \text{for terminal } s_{i+1} \\ r_i + \hat{Q}(s_{i+1}, \arg \max_a Q(s_{i+1}, a; \theta_t), \theta^-) & \text{for non-terminal } s_{i+1} \end{cases}$$
Select $a_t = \arg \max_{a'} Q(a')$

If air balancing

Break

End for

由单支路动态目标训练机制选取系统初始状态和训练目标，模型具体训练过程见算法 3。由于实际通风管道系统中存在不可预测的噪声和干扰，这将会影响风量控制过程中的精确性。因此，在训练过程中，将风平衡完成信号设置为风量误差 3%。

为了使控制方法在测试中获得最优控制性能，需要在训练过程中确定 Double DQN 模型的最优超参数。其中，折扣因子 γ 表示智能体在选择动作时对未来奖励的重视程度，其值满足 $0 \leq \gamma \leq 1$ ， γ 越接近 1 表示智能体更重视长期回报，反之表示更重视即时奖励。初始探索率 $\epsilon_{\max}$ 会随着时间按照 ϵ_{decay} 的衰减率减小至 ϵ_{decay} ，将探索转为利用。学习率 α 指在训练神经网络时，每次更新参数权重的调整速率。经验回放池用于存储训练过程中智能体与环境交互产生的经验 (s_t, a_t, r_t, s_{t+1}) ，当存放数据较多时，导致收敛速度降低，当经验池存放数据较少时，不能有效降低数据间相关性。批量大小为每次训练迭代中处理样本数量，批量大小会影响训练速率和梯度估计的稳定。目标网络更新频率帮助减少过度估计，当设置过大时会导致训练不稳定，反之延缓训练进程。

表 4-1 Double DQN 训练超参数表

Hyper-parameter	Candidate value range	Selected value
γ	{0.9, 0.95, 0.99}	0.99
$\epsilon_{\min}$	{0.01, 0.05, 0.09}	0.01
$\epsilon_{\max}$	{0.9, 0.95, 1}	1
ϵ_{decay}	{0.0001, 0.001, 0.002}	0.001
α	{0.005, 0.001, 0.01}	0.001
Experience replay pool	{ $10^3, 10^4, 10^5$ }	10^4
Batch size	{64, 128, 256}	128
Maximum steps	{150, 200, 250}	200
Target network update frequency	{3, 5, 8}	5
Neural network layers	{3, 4, 5}	5
Number of neural units in hidden layer	{64, 128, 256}	[128, 64, 64]
Activation function	-	Relu
Optimizer	-	Adam

通过验证集交叉验证，以最终的奖励值大小和达到最大奖励值的最小回合步数为选择依据，选取合适超参数。表 4-1 列出来用于 Double DQN 训练的详细超参数配置。本文采用 PyTorch 实现 Q 网络和目标网络的搭建，使用 Adam 优

化器^[67]对神经网络的参数进行优化，共三层隐藏层，隐藏层设置为全连接层，每层均采用 Relu 激活函数。

在线学习过程：

将预训练的模型网络结构与参数复制到在线学习模型中作为初始的网络结构和参数，用以解决在线学习阶段网络参数随机初始化带来的参数敏感问题，以提高风平衡收敛速度。根据需要部署风平衡方法的实际通风管道系统，选择需要进行控制的支路风阀应用训练完成的控制策略。在线学习过程中，多支路通风管道系统的实际运行状态和智能体进行交互对网络参数进行自适应调整。在线学习过程直至所有目标支路达到目标风量时结束，在线学习过程流程图如图 4-7 所示。

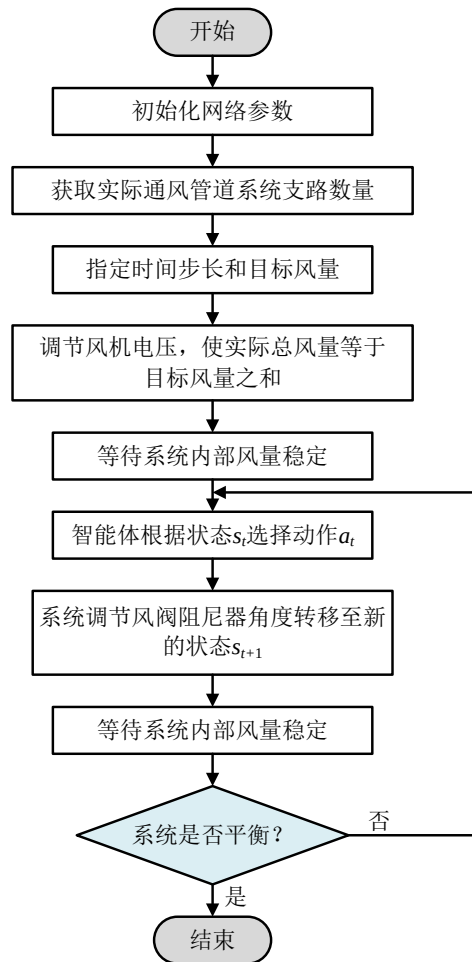


图 4-7 在线学习过程流程图

所提出的 DRL-AB 方法实现过程总结如下：

步骤 1：设定各支路所需风量，初始化所有风阀阻尼器角度。

步骤 2：通过模糊 PID 控制对风机给定电压进行调整，重复该步骤，直至所有支路实际风量之和等于设计风流量之和。

步骤 3：由算法 3 中的在线控制过程，调整各个支路风阀阻尼器角度，重复该步骤，直至各个支路风量误差达到设定标准。

步骤 4：保持阻尼器角度和风机电压恒定，实现风平衡并记录实验结果。

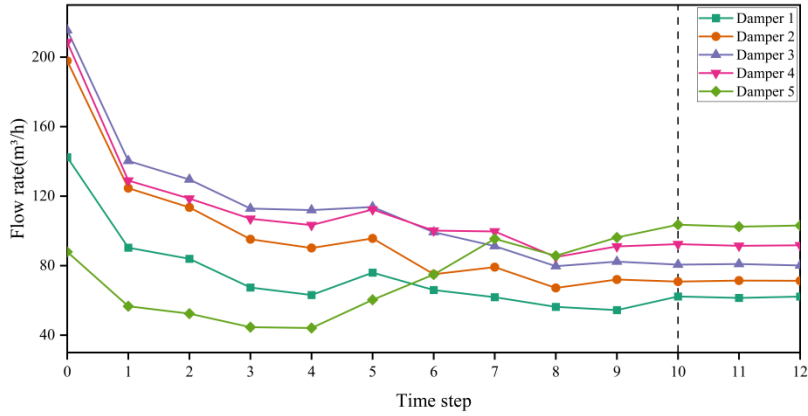
4.5 实验验证

表 4-2 5 组风量实验结果

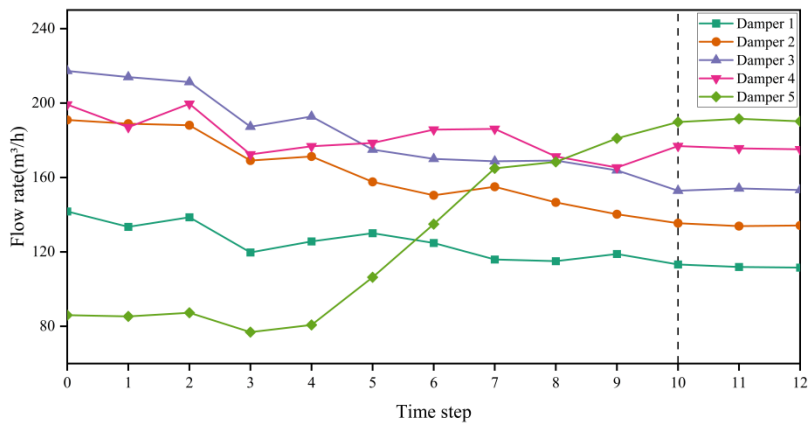
No	Terminal	Desired flowrate(m ³ /h)	Measured Angle (°)	Measured flowrate(m ³ /h)	Error(%)
1	1	60.0	51	61.72	2.87
	2	70.0	54	71.32	1.89
	3	80.0	57	80.12	0.15
	4	90.0	53	91.76	1.96
	5	100.0	27	102.52	2.52
2	1	110.0	52	111.57	1.43
	2	130.0	55	133.23	2.48
	3	150.0	55	153.29	2.19
	4	170.0	50	174.56	2.68
	5	190.0	26	190.23	0.12
3	1	210.0	47	205.65	2.07
	2	230.0	51	235.67	2.47
	3	250.0	51	257.19	2.88
	4	270.0	48	274.32	1.60
	5	290.0	23	286.12	1.03
4	1	310.0	46	315.98	1.93
	2	330.0	52	323.19	2.06
	3	350.0	52	358.28	2.36
	4	370.0	50	365.93	1.10
	5	390.0	27	398.65	2.22
5	1	410.0	45	415.19	1.27
	2	430.0	52	430.56	0.13
	3	450.0	52	455.48	1.22
	4	470.0	50	482.67	2.69
	5	490.0	28	498.95	1.83

为验证本文提出的基于强化学习的风平衡控制方法的收敛效果，采用 3.4 节改造实验平台进行实验验证。在本章所有实验中，时间步长设置为 20 秒，用以消除各支路之间因气流变化引起的气流震荡，为系统达到稳态提供充足的时间。在所有实验开始前，风机的初始电压调为一致，各个支路的初始风阀阻尼器角度均设置为 45° ，以保证收敛行为的任何差异归因于所使用风平衡方法的有效性，而不是初始条件的变化。

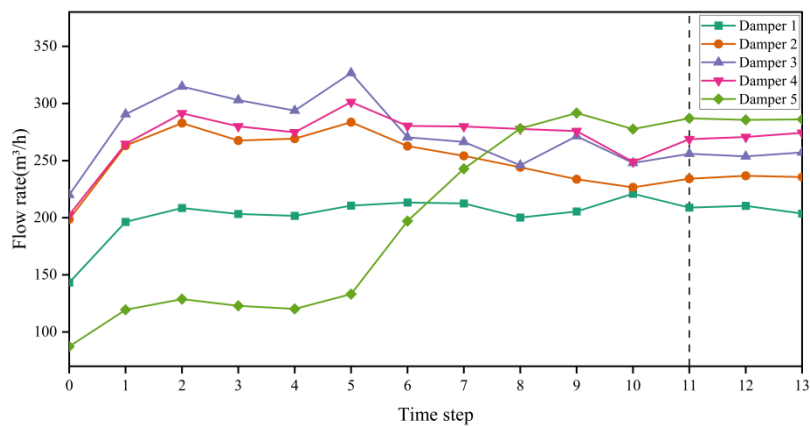
为验证 DRL-AB 方法大范围目标风量的收敛性能，进行 5 组实验来测试 5 组风量等级设定下的收敛情况，如表 4-2 所示。每组有 5 个支路管道，目标风量分别设为该风量等级下的不同风量。由图 4-8 可知，5 组实验分别在 10、10、11、9、12 步长内收敛到目标风量，且误差均在 3% 以内，符合 ASHRAE 标准 (10%)。



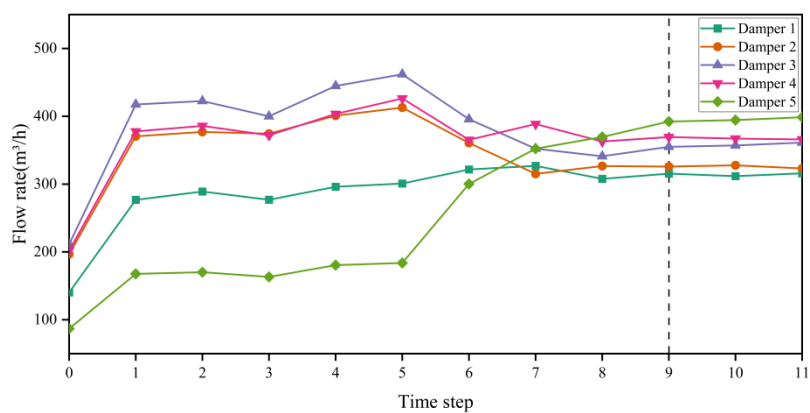
(a) 风量等级 1 控制结果



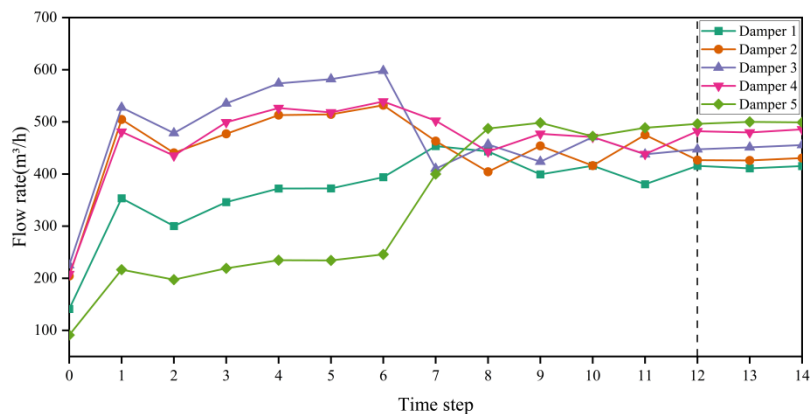
(b) 风量等级 2 控制结果



(c) 风量等级 3 控制



(d) 风量等级 4 控制结果



(e) 风量等级 5 控制结果

图 4-8 风量控制效果图

在实际系统上的实验结果表明, 经过训练后的智能体在风量调节过程中具有较好的控制性能, 这主要是因为: (1) 智能体根据目标风量等级做出不同的控制策略, 由实际系统误差输出控制动作, 保证了风量控制性能; (2) 智能体

基于预训练阶段学习到的通风系统物理规律，在线控制阶段根据系统反馈调整网络参数，使智能体更适用于实际系统风量控制。

为验证本文所提 DRL-AB 方法的平衡收敛速度和精准度，通过上述实验平台，与分布式协同风平衡控制方法(DCC)和风机自适应的分布式协同风平衡方法(DCAC-AB)进行比较。测试的目标风量分别设定为不同风量等级下风量，用于检验风量控制性能。实验结果如表 4-3、4-4 和 4-5 所示，表 4-6 记录了三种方法的实验对比结果。图 4-9 是各支路风量的平衡收敛图。表 4-3、4-4 和 4-5 中， q_{int} 为系统各支路初始风量， q^* 为各支路目标风量， q_f 为系统平衡后的风量， θ 为各支路风阀阻尼器初始角度， θ_f 为系统平衡后各支路风阀阻尼器角度。

表 4-3 DCC 实验结果

Terminal	$q_{int}(m^3/h)$	$q^*(m^3/h)$	$q_f(m^3/h)$	$\theta(^{\circ})$	$\theta_f(^{\circ})$
1	143.45	90.00	90.33	45.0	49.3
2	199.67	350.00	340.33	45.0	21.6
3	220.76	420.00	423.77	45.0	20.8
4	208.78	220.00	203.55	45.0	43.5
5	91.34	150.00	152.66	45.0	30.4

表 4-4 DCAC-AB 实验结果

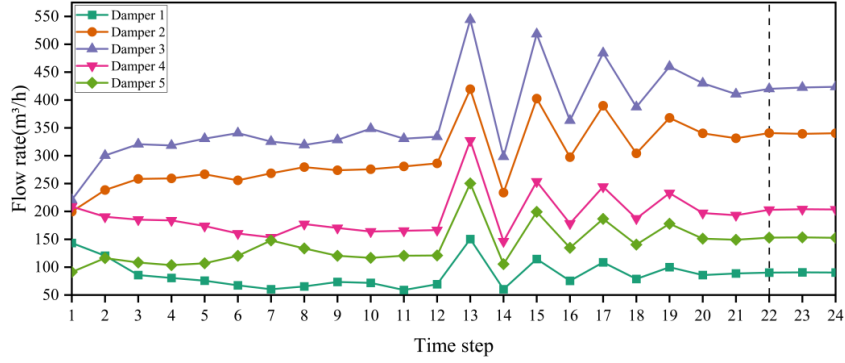
Terminal	$q_{int}(m^3/h)$	$q^*(m^3/h)$	$q_f(m^3/h)$	$\theta(^{\circ})$	$\theta_f(^{\circ})$
1	141.89	90.00	86.65	45.0	50.8
2	200.58	350.00	343.71	45.0	22.1
3	223.62	420.00	415.96	45.0	22.2
4	205.68	220.00	206.64	45.0	44.3
5	90.90	150.00	149.26	45.0	31.7

表 4-5 DRL-AB 实验结果

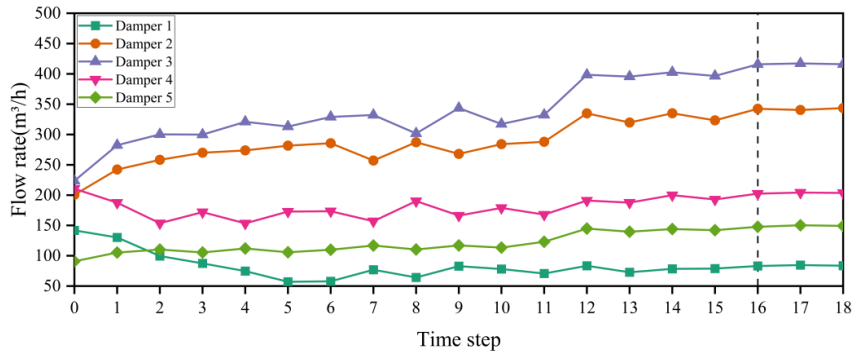
Terminal	$q_{int}(m^3/h)$	$q^*(m^3/h)$	$q_f(m^3/h)$	$\theta(^{\circ})$	$\theta_f(^{\circ})$
1	144.40	90.00	92.49	45.0	60.0
2	197.31	350.00	358.34	45.0	39.0
3	215.00	420.00	426.28	45.0	36.0
4	198.62	220.00	215.29	45.0	52.0
5	86.55	150.00	146.72	45.0	41.0

如图 4-9 所示，三种方法下最终都完成风量平衡收敛，各支路末端风量与目标风量之间的误差均在 $\pm 10\%$ 以内，符合 ASHRAE 标准。由表 4-6 可得，DRL-AB 方法中系统所有支路误差均控制 3%以内，小于其它两种方法。由图 4-9 (a)可以得到，DCC 控制方法在第 22 个步长处，各支路风量都达到平衡状态，

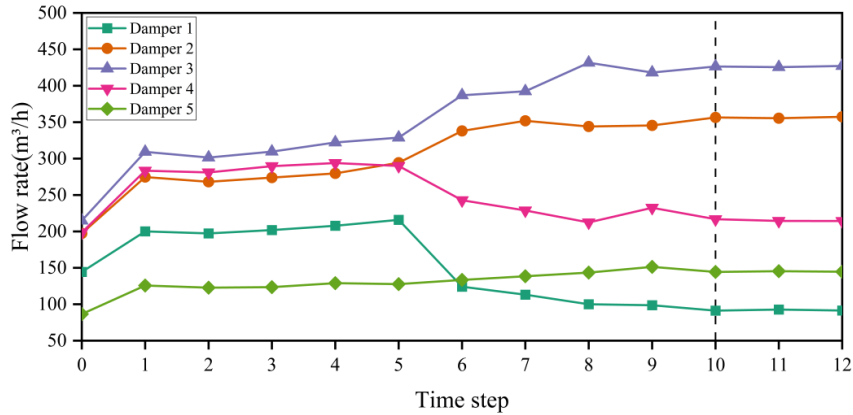
耗时 7.33 min；图 4-9 (b)中 DCAC-AB 控制方法在 16 个步长处，各支路风量达到平衡状态，耗时 5.33 min；图 4-9 (c)中 DRL-AB 控制方法在 10 个步长处，各支路风量达到平衡状态，耗时 3.33 min。



(a) DCC 方法各支路风量



(b) DCAC-AB 方法各支路风量



(c) DRL-AB 方法各支路风量

图 4-9 实验结果对比图

通过三种控制方法的实验结果对比，在面对大范围目标风量调整时，所提出的 DRL-AB 方法在控制精度上和收敛速度优于其它两种方法。这主要是因为 DRL-AB 方法建立深度神经网络拟合控制策略，学习通风管道系统内部非线性、强耦合性的物理规律，能根据系统状态快速做出高精度调整。

表 4-6 DCC 法、DCAC-AB 法与 DRL-AB 法的对比结果

控制方法	相对误差		调节时间 (min)
	最大误差(%)	平均误差(%)	
DCC	6.25	2.66	7.33
DCAC-AB	7.48	2.64	5.33
DRL-AB	2.77	2.20	3.33

为进一步验证 DRL-AB 方法的收敛速率，随机生成 15 组 50 m³/h - 500 m³/h 的目标风量，对比三种方法达到风平衡的调节时间。在每次测试中，三种方法的初始状态保持相同，平衡时间对比图如图 4-10 所示。与 DCC 方法和 DCAC-AB 方法对比，本文所提出的 DRL-AB 方法在收敛速率上有明显提升。

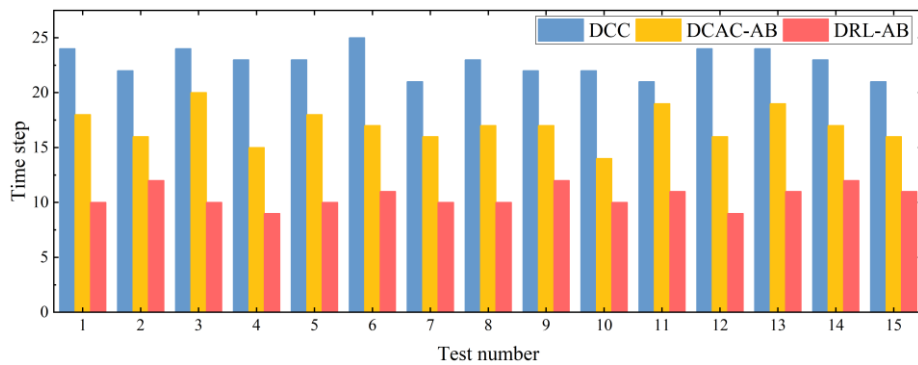


图 4-10 平衡时间对比图

表 4-7 控制误差对比结果

控制方法	平均误差(%)
PID	8.25
Model-based	6.80
SVM	4.60
DPH-AB	5.00
GOAB	4.30
ANN-based	3.80
DRL-AB	2.20

为进一步验证 DRL-AB 控制方法的精准性，选择传统控制方法、基于模型的方法和数据驱动的控制方法进行比较。分别与 PID 风平衡方法、基于模型的风平衡控制方法(Model-based)^[68]、基于支持向量机的风平衡控制方法(SVM)^[69]、数据物理混合驱动的风平衡控制方法(DPH-AB)^[70]、基于梯度的自适应风平衡控制方法(GOAB)^[71]、基于神经网络的风平衡控制方法(ANN-based)^[72]进行误差精准度对比，对比结果如表 4-7 所示，DRL-AB 方法在对风量控制中，平均误差较其他方法最小，是一种精准的风平衡控制方法。

本文还在不同支路数量、不同拓扑结构的通风系统（如图 4-11 所示）上对所提 DRL-AB 方法的泛化能力进行了测试。智能体以预训练的网络结构与参数为初始，通过与系统交互自适应调整网络参数，根据风量等级和风量误差输出控制动作，实现风平衡控制要求。

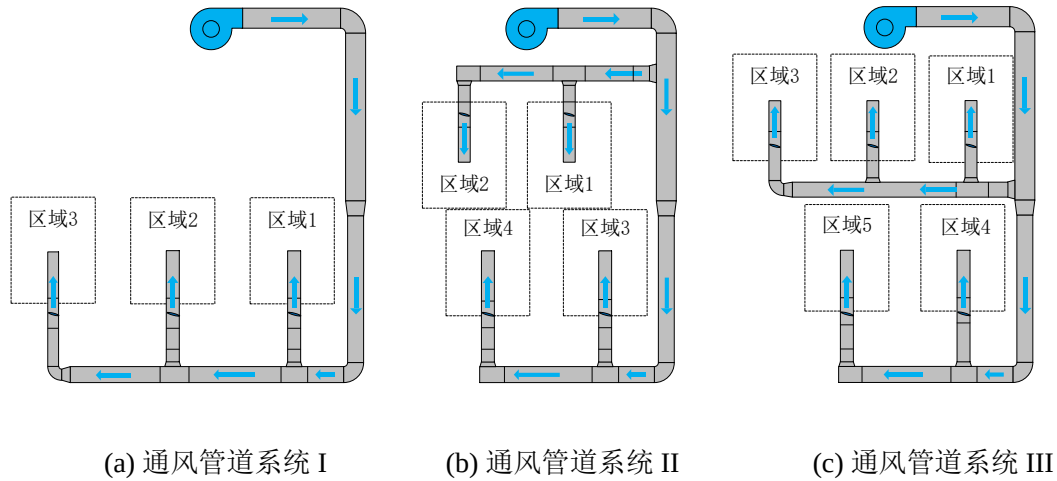


图 4-11 不同通风管道系统的拓扑结构图

因不同区域数量的通风管道系统稳定时间不同，其调节步数也不同。表 4-8 记录了平衡时间。由表 4-8 可知，风量控制误差均在 3%以内，满足 ASHRAE 标准。其中，最大调节步数为 13 步，具备较快的平衡收敛速度。因此，所提 DRL-AB 方法可实现不同支路数量、不同拓扑结构通风管道系统下风量的快速、精准控制，具备优秀的泛化性和鲁棒性。

表 4-8 不同通风管道系统调节结果

通风管道系统	目标风量 $q_{int}(m^3/h)$	平均误差(%)	调节步数
I	[120, 150, 180]	1.9	11
	[100, 210, 220]	2.4	9
	[150, 200, 240]	1.8	9
II	[130, 80, 90, 160]	1.7	10
	[190, 200, 210, 250]	2.0	8
	[230, 310, 330, 90]	2.1	13
III	[210, 120, 150, 90, 310]	2.3	10
	[80, 160, 330, 410, 230]	2.6	12
	[80, 90, 140, 210, 380]	2.1	10

4.6 本章小结

为解决目前风平衡方法中存在的精准度低、平衡速度慢和通用性差的问题,本章提出了基于强化学习的风平衡控制方法。首先,本章介绍了强化学习的基本知识,包括马尔可夫决策过程与深度学习算法。然后,基于对深度强化学习算法的分析,设计了具有马尔可夫性质的风量控制过程,以减小风量误差为导向设计了通风管道系统的状态、动作、奖励函数以及终止条件。设计预训练和在线学习框架。为提高训练效率、提高方法泛化能力,设计了单支路动态目标训练机制。设计了在线学习智能体以预训练后的网络结构与参数为初始,由实际情况反馈调整控制策略网络,保证了方法的控制性能,提高方法通用性。

由实验平台对 DRL-AB 方法进行性能验证,通过大范围目标风量设定测试和对比实验,验证了所提出方法平衡收敛的快速性和精准性。通过不同支路数量、不同拓扑结构的通风管道系统,验证了 DRL-AB 方法的泛化能力。实验结果表明 DRL-AB 方法是一种快速、精准且通用性强的大范围风平衡控制方法,可应用于实际环境中。

第 5 章 风平衡控制系统 Web 端部署

5.1 系统需求分析

风平衡控制技术是面向楼宇建筑环境应用中的，网页端的部署是控制方法的落地的重要一环。为方便根据不同环境选择合适的风平衡控制方法，本章将风机自适应的分布式协同风平衡控制方法和基于强化学习的风平衡控制方法进行 Web 端部署，使用户能够在可视化界面中，实时观测到通风管道系统中风机和风阀的运行状态和各支路的风量信息，方便用户根据环境选择不同的风平衡控制方法。

5.2 Web 端总体设计

5.2.1 开发环境介绍

Web 端开发在配有 Inter (R) Core i7-10700、64GB RAM 和 NVIDIA GeForce RTX3090 台式计算机上实现。采用 IDEA (IntelliJ IDEA)软件^[73-74]进行云服务器的开发设计，获取并解析通风管道系统内部数据。该软件是 Java 编程语言的集成开发环境，可满足各种工程中的需求。本文所借助的开发平台为 HX (HbuilderX)软件^[75]，利用其完成了 Web 网页的设计，实现系统在线控制、状态展示和历史数据查询等功能。

5.2.2 远程监控平台架构

远程监控平台由云服务器、数据库、客户端组成。云服务器端主要用于接收并解析通风管道系统发送的信息，对客户端的请求进行反馈^[76]；数据库用于存储用户数据与通风管道系统的工作数据；客户端通过 Web 网页展示通风管道系统的工作数据，支持风平衡控制模式指令下发，风阀、风机单独指令控制。整体架构如图 5-1 所示。

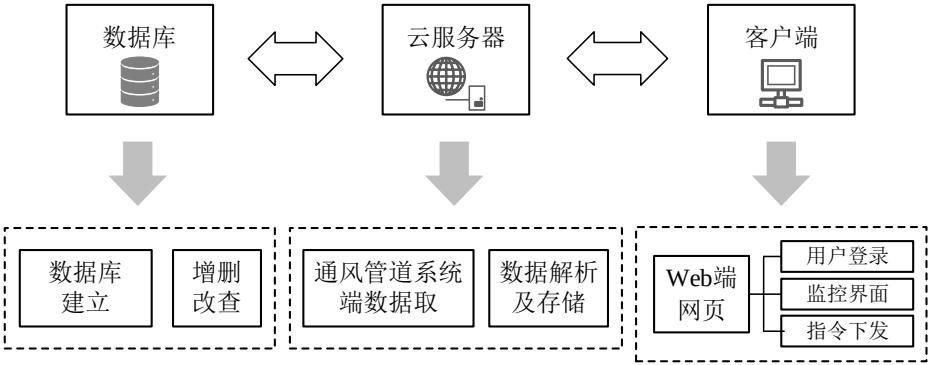


图 5-1 远程监控平台架构图

5.3 客户端设计

客户端为用户提供监控界面，可以实时观测通风管道系统状态和在线对其进行模式控制等^[77]。由 Vue.js 技术框架、Element-UI 组件库和 Webpack 工具，完成对客户端页面的设计以及打包管理与部署。

5.3.1 登录界面

为了增强网页的安全性和保护用户的隐私，本文开发设计用户管理员登陆界面以进行身份验证，登录界面如图 5-2 所示。



图 5-2 登录界面

在用户完成登录认证后，后台服务器生成加密后的 Token，并伴随着 HTTP Header 传送至云服务器。服务器通过数据库中的用户信息进行筛选辨认，以确保其真实性。同时，浏览器以 cookie 形式储存 Token，以供后续使用。

5.3.2 系统简介界面

本文设计了系统简介界面，如图 5-3 所示。在该界面中，用户可以了解实验平台的平面图及工作原理。界面中提供实验平台各分支实图，供用户对平台结构深入了解。



图 5-3 系统简介界面

5.3.3 平台监控界面

本文设计了平台监控界面，实现对通风管道平台各组件工作数据的监控，如图 5-4 所示。

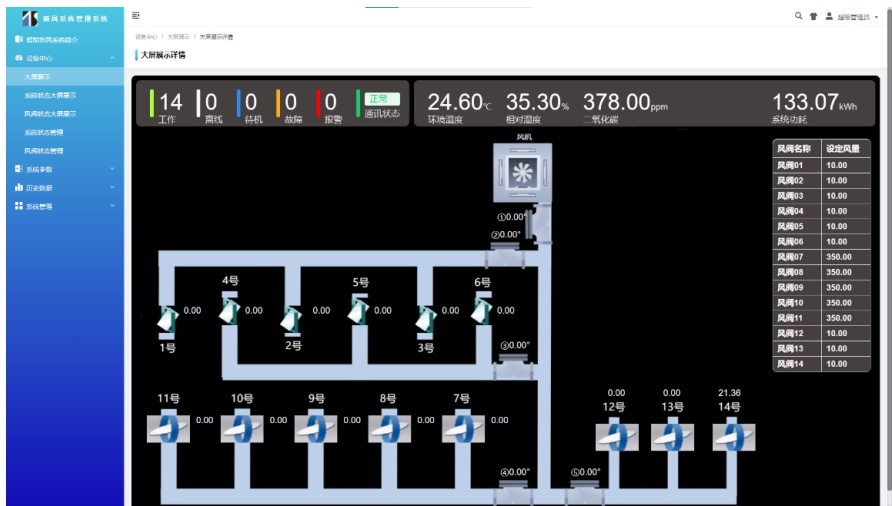


图 5-4 平台监控界面

该界面中，用户可观测到各风阀的连接状态、当前管道内温湿度、管道内二氧化碳浓度和当前系统功耗。组态动画界面中可实时显示当前各风阀和比例风阀的阻尼器开口角度，同时也可观测到各支路末端的目标风量值。

5.3.4 远程控制界面

本文设计了远程控制界面，实现通风管道平台风平衡控制模式的选择以及系统调节参数、各分支比例控制阀角度和风机电压的控制，如图 5-5 所示。



图 5-5 远程控制界面

在模式选择中，将本文所设计的风机自适应的分布式协同风平衡控制方法和基于强化学习的风平衡控制方法进行部署，用户可通过远程控制界面选择通风管道平台风平衡工作模式，如图 5-6 所示，每条支路风量信息可由图 5-10 观测。可根据需要调节风机初始控制参数，如图 5-7 所示。可根据实际需求对各分支比例风阀进行角度调节，如图 5-8 所示。可自定义风机给定电压大小，如图 5-9 所示。

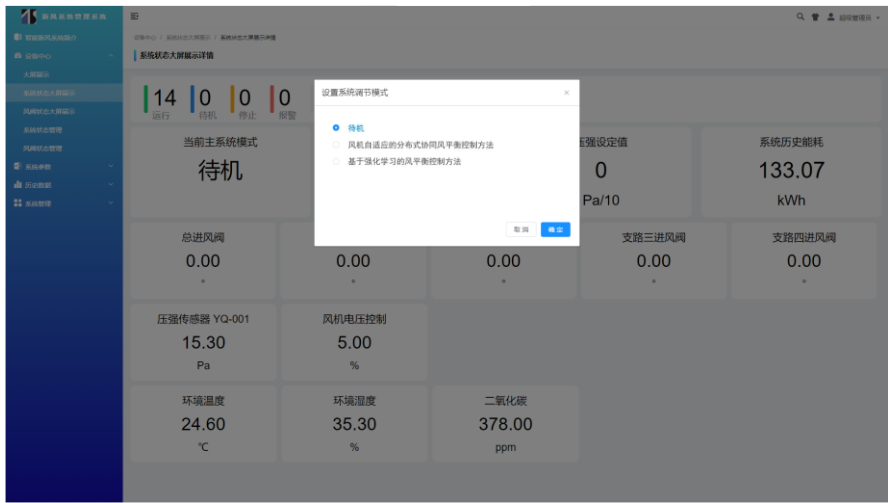


图 5-6 工作模式选择界面

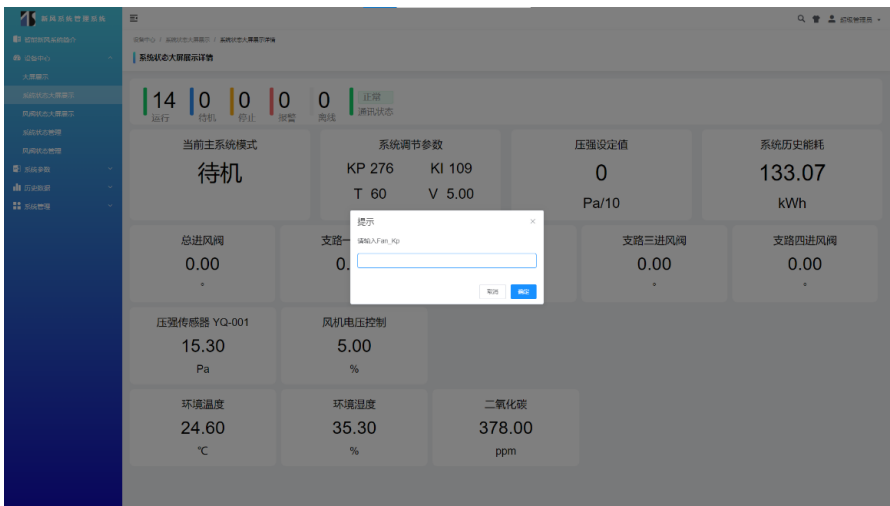


图 5-7 系统控制参数调节界面

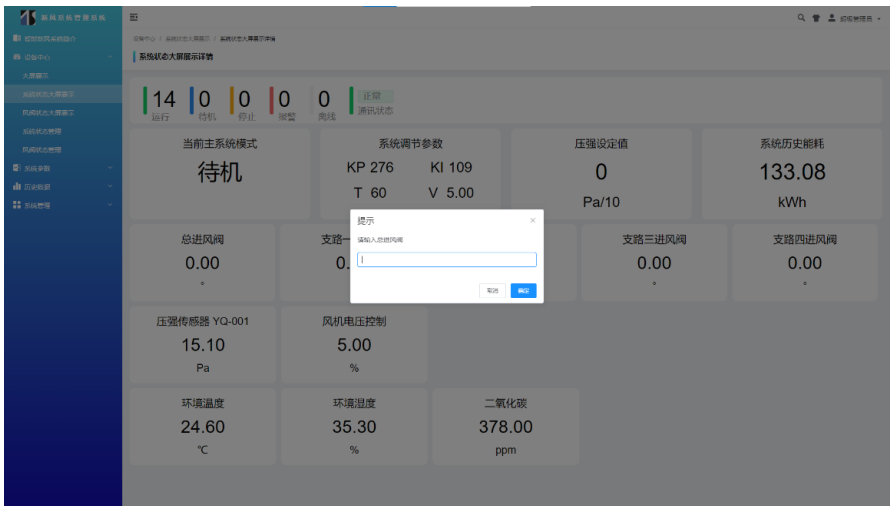


图 5-8 分支比例风阀控制界面

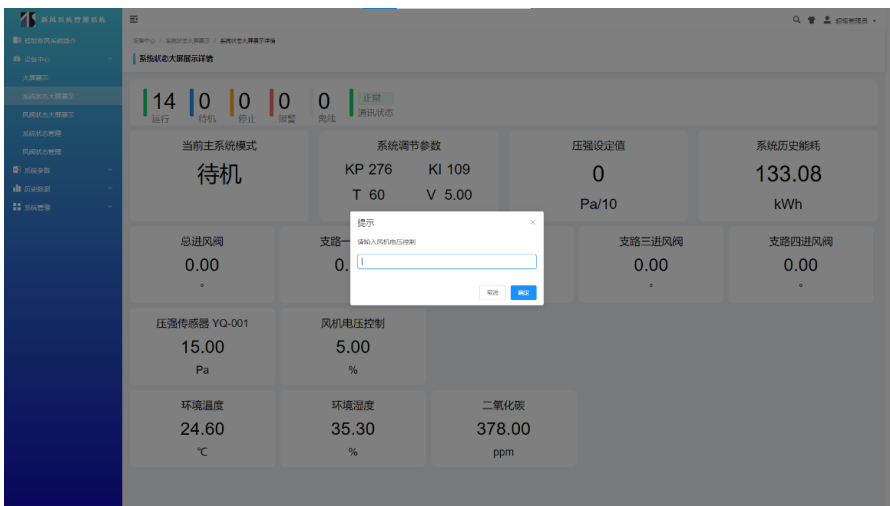


图 5-9 风机电压控制界面

5.3.5 风阀信息界面

为更加方便用户观测各风阀信息，本文设计了风阀信息界面，如图 5-10 所示。在该界面中，用户可观测各风阀当前角度状态、各支路实时风量与设定目标风量。

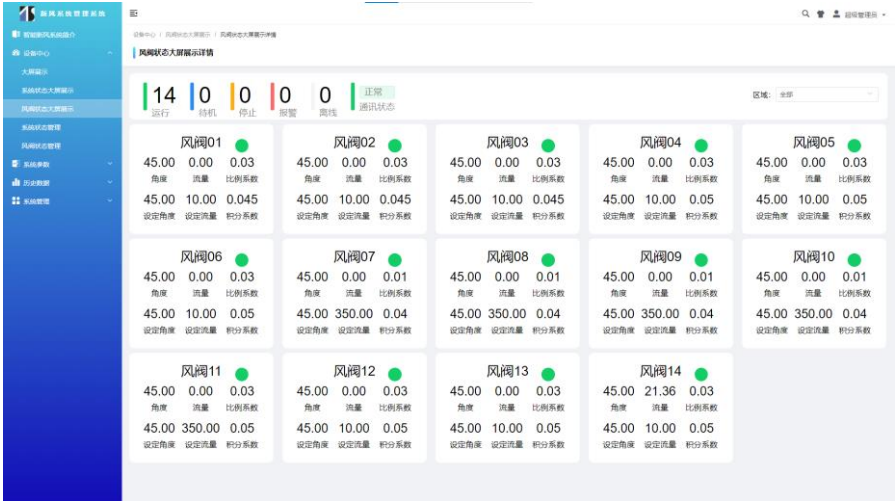


图 5-10 风阀信息界面

5.3.6 风阀状态控制界面

本文设计了风阀控制界面，用户可根据需要对单个或多个风阀进行控制，如图 5-11 所示。

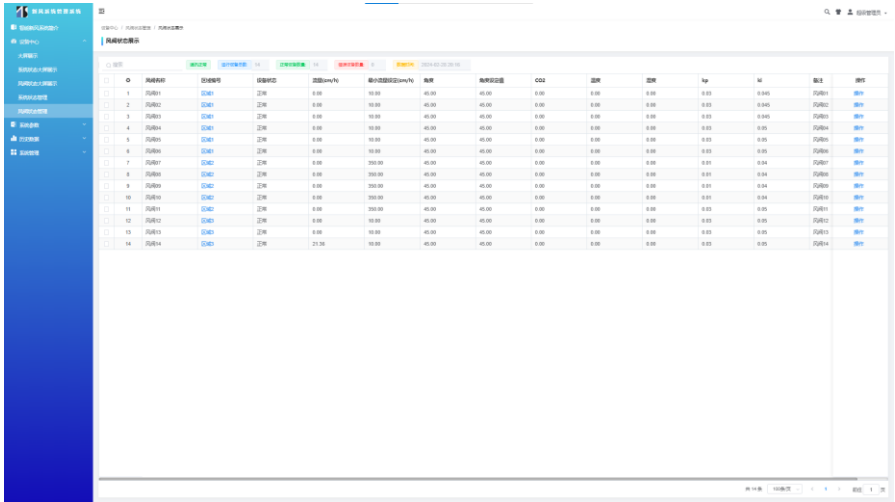


图 5-11 风阀状态控制界面

5.3.7 历史数据查询界面

本文设计了历史数据查询界面，将通风管道系统的工作数据保存至数据库中，便于用户查询，如图 5-12 所示。

系统历史数据查询界面													
序号	设备名称	数据时间	主控制模式	数据时间	电压	电压功率	I	M	Tap	温度	风阀开度	风速/m/s	风压/Pa
1	风阀控制	2023-03-20 17:17	手动	17:17	0	0	0	0	0	0	0	0	0
2	风阀控制	2023-03-20 17:18	手动	17:18	0	0	0	0	0	0	0	0	0
3	风阀控制	2023-03-20 17:19	手动	17:19	0	0	0	0	0	0	0	0	0
4	风阀控制	2023-03-20 17:20	手动	17:20	0	0	0	0	0	0	0	0	0
5	风阀控制	2023-03-20 17:21	手动	17:21	0	0	0	0	0	0	0	0	0
6	风阀控制	2023-03-20 17:22	手动	17:22	0	0	0	0	0	0	0	0	0
7	风阀控制	2023-03-20 17:23	手动	17:23	0	0	0	0	0	0	0	0	0
8	风阀控制	2023-03-20 17:24	手动	17:24	0	0	0	0	0	0	0	0	0
9	风阀控制	2023-03-20 17:25	手动	17:25	0	0	0	0	0	0	0	0	0
10	风阀控制	2023-03-20 17:26	手动	17:26	0	0	0	0	0	0	0	0	0
11	风阀控制	2023-03-20 17:27	手动	17:27	0	0	0	0	0	0	0	0	0
12	风阀控制	2023-03-20 17:28	手动	17:28	0	0	0	0	0	0	0	0	0
13	风阀控制	2023-03-20 17:29	手动	17:29	0	0	0	0	0	0	0	0	0
14	风阀控制	2023-03-20 17:30	手动	17:30	0	0	0	0	0	0	0	0	0
15	风阀控制	2023-03-20 17:31	手动	17:31	0	0	0	0	0	0	0	0	0
16	风阀控制	2023-03-20 17:32	手动	17:32	0	0	0	0	0	0	0	0	0
17	风阀控制	2023-03-20 17:33	手动	17:33	0	0	0	0	0	0	0	0	0
18	风阀控制	2023-03-20 17:34	手动	17:34	0	0	0	0	0	0	0	0	0
19	风阀控制	2023-03-20 17:35	手动	17:35	0	0	0	0	0	0	0	0	0
20	风阀控制	2023-03-20 17:36	手动	17:36	0	0	0	0	0	0	0	0	0
21	风阀控制	2023-03-20 17:37	手动	17:37	0	0	0	0	0	0	0	0	0
22	风阀控制	2023-03-20 17:38	手动	17:38	0	0	0	0	0	0	0	0	0
23	风阀控制	2023-03-20 17:39	手动	17:39	0	0	0	0	0	0	0	0	0
24	风阀控制	2023-03-20 17:40	手动	17:40	0	0	0	0	0	0	0	0	0
25	风阀控制	2023-03-20 17:41	手动	17:41	0	0	0	0	0	0	0	0	0
26	风阀控制	2023-03-20 17:42	手动	17:42	0	0	0	0	0	0	0	0	0
27	风阀控制	2023-03-20 17:43	手动	17:43	0	0	0	0	0	0	0	0	0
28	风阀控制	2023-03-20 17:44	手动	17:44	0	0	0	0	0	0	0	0	0
29	风阀控制	2023-03-20 17:45	手动	17:45	0	0	0	0	0	0	0	0	0
30	风阀控制	2023-03-20 17:46	手动	17:46	0	0	0	0	0	0	0	0	0
31	风阀控制	2023-03-20 17:47	手动	17:47	0	0	0	0	0	0	0	0	0
32	风阀控制	2023-03-20 17:48	手动	17:48	0	0	0	0	0	0	0	0	0
33	风阀控制	2023-03-20 17:49	手动	17:49	0	0	0	0	0	0	0	0	0
34	风阀控制	2023-03-20 17:50	手动	17:50	0	0	0	0	0	0	0	0	0
35	风阀控制	2023-03-20 17:51	手动	17:51	0	0	0	0	0	0	0	0	0
36	风阀控制	2023-03-20 17:52	手动	17:52	0	0	0	0	0	0	0	0	0

图 5-12 历史数据查询界面

5.4 本章小结

本章设计了 Web 端远程监控平台，主要包括云服务器、数据库和客户端。在客户端设计中，用户通过登录界面进入通风管道系统远程监控界面，可远程实现对平台的风平衡模式选择、风阀控制、数据监控和历史数据查询等功能，方便用户对通风管道平台状态的观测和控制。

第 6 章 总结与展望

6.1 总结

本文对通风管道系统中多支路风平衡控制方法展开研究，分别提出了风机自适应的分布式协同风平衡控制方法和基于强化学习的风平衡控制方法，实现了通风管道系统中风量的准确控制，保证了系统的节能运行。本文具体工作总结如下：

(1) 开展了风机自适应的分布式协同风平衡控制方法研究。首先，将共识算法引入多支路管道系统中，通过相邻风阀间进行风量信息交换，使各支路的风量与目标风量比值达成一致，实现多风阀的分布式协同控制。然后，针对通风管道系统风量控制的时变特性，提出采用风机自适应控制算法，利用模糊规则表，实现风量的快速收敛。最后，设计风机自适应的分布式协同风平衡控制方法并进行实验测试。通过与传统分布式控制方法的实验对比，得出该控制方法的快速有效性，可用于实际多支路通风管道系统风量控制问题。

(2) 开展了基于强化学习的风平衡控制方法研究。首先，根据通风系统管道内部风量-压强的高度非线性和风量强耦合的物理特性，设计了具有马尔可夫性质的风量控制过程，设置了以减小风量误差为导向的强化学习状态、动作和奖励函数。随后，为提高方法的泛化性，减少模型训练时间成本，设计了单支路动态目标训练机制。将模型先进行预训练，再将训练后的网络结构和参数为初始，进行在线学习，以保证实际系统的控制效率。最后，设计了基于强化学习的风平衡控制方法步骤，并通过实验平台验证了所提出的基于强化学习的风平衡控制方法对大范围风量的准确、快速控制，验证了该方法的通用性，可在实际应用中推广。

6.2 展望

通风管道系统中的风平衡控制方法对于节约能源、维护身体健康有着重要作用。然而该方向的研究较为缓慢，多智能体控制方法相对困难，更加节能的方法还需深入研究。随着强化学习和深度学习的发展，未来可以从下方向进行

研究：

（1）本文采用深度强化学习算法仅对风阀进行控制调节，未曾将风机调节加入到强化学习的控制策略中，因此在节能方向还有大幅度上升空间。随着多智能体强化学习算法的发展，研究如何在风机最低能耗的条件下实现多支路间的风平衡控制具有重大意义。

（2）本文采用深度神经网络来拟合强化学习中的估计网络和目标网络。近些年随着深度学习的发展，注意力机制和神经网络等新型网络结构陆续出现。研究网络结构的设计对强化学习模型的训练效率和控制性能提升有广阔前景。

（3）本文所设计的基于强化学习的控制方法需在实际环境中与系统交互进行训练，训练会占用设备的运行时间。研究准确的离线训练方式可以有效降低时间成本和能源消耗，具有重要意义。

（4）本文在强化学习模型训练期间，智能体需要环境进行大量实时交互获取控制策略。交互初期，智能体对环境的探索是随机的，研究如何根据工程师控制经验来引导智能体学习方向，对于提升强化学习模型训练效率有着重要研究价值。

参考文献

- [1] Ahmad M W, Mourshed M, Mundow D, et al. Building energy metering and environmental monitoring—A state-of-the-art review and directions for future research[J]. *Energy and Buildings*, 2016, 120: 85-102.
- [2] Toub M, Reddy C R, Robinett III R D, et al. Integration and optimal control of microcsp with building hvac systems: Review and future directions[J]. *Energies*, 2021, 14(3): 730.
- [3] 中国建筑能耗与碳排放研究报告(2022 年)[J]. *建筑*, 2023, (02): 57-69.
- [4] 习近平. 高举中国特色社会主义伟大旗帜为全面建设社会主义现代化国家而团结奋斗——在中国共产党第二十次全国代表大会上的报告[J]. *创造*, 2022, 30(11): 6-29.
- [5] 在经济发展中促进绿色转型在绿色转型中实现更大发展[N]. *人民日报*, 2024-05-07(012). DOI: 10.28655/n.cnki.nrmrb. 2024. 004574.
- [6] Okafor V C, Ezeanya N C, Nwandikom G I, et al. Energy saving potential, environmental and economic importance of evaporative cooling system: A review[J]. *European Journal of Advances in Engineering and Technology*, 2019, 6(3): 34-45.
- [7] Valenzuela P E, Ebadat A, Everitt N, et al. Closed-loop identification for model predictive control of HVAC systems: From input design to controller synthesis[J]. *IEEE Transactions on Control Systems Technology*, 2019, 28(5): 1681-1695.
- [8] Cui C, Zhang X, Cai W, et al. A gradient-based adaptive balancing method for dedicated outdoor air system[J]. *Building and Environment*, 2019, 151: 15-29.
- [9] Gao G, Li J, Wen Y. DeepComfort: Energy-efficient thermal comfort control in buildings via reinforcement learning[J]. *IEEE Internet of Things Journal*, 2020, 7(9): 8472-8484.
- [10] Less B D, Dutton S M, Walker I S, et al. Energy savings with outdoor temperature-based smart ventilation control strategies in advanced California homes[J]. *Energy and Buildings*, 2019, 194: 317-327.

- [11]Cui C, Zhang X, Cai W. An energy-saving oriented air balancing method for demand controlled ventilation systems with branch and black-box model[J]. Applied Energy, 2020, 264: 114734.
- [12]Cao G, Awbi H, Yao R, et al. A review of the performance of different ventilation and airflow distribution systems in buildings[J]. Building and Environment, 2014, 73: 171-186.
- [13]Sundell J, Levin H, Nazaroff W W, et al. Ventilation rates and health: Multidisciplinary review of the scientific literature[J]. Indoor air, 2011, 21(3): 191-204.
- [14]Hati A S, Kumar P. An adaptive neural fuzzy interface structure optimisation for prediction of energy consumption and airflow of a ventilation system[J]. Applied Energy, 2023, 337: 120879.
- [15]He D, Teng X, Chen Y, et al. Energy saving in metro ventilation system based on multi-factor analysis and air characteristics of piston vent[J]. Applied Energy, 2022, 307: 118295.
- [16]Chua K J, Chou S K, Yang W M, et al. Achieving better energy-efficient air conditioning—a review of technologies and strategies[J]. Applied Energy, 2013, 104: 87-104.
- [17]Kusiak A, Li M, Zhang Z. A data-driven approach for steam load prediction in buildings[J]. Applied Energy, 2010, 87(3): 925-933.
- [18]Peng Y, Rysanek A, Nagy Z, et al. Using machine learning techniques for occupancyprediction-based cooling control in office buildings[J]. Applied energy, 2018, 211: 1343-1358.
- [19]Huang Y, Khajepour A, Ding H, et al. An energy-saving set-point optimizer with a sliding mode controller for automotive air-conditioning/refrigeration systems[J]. Applied Energy, 2017, 188: 576-585.
- [20]Chaudhuri T, Soh Y C, Li H, et al. A feedforward neural network based indoor-climate control framework for thermal comfort and energy saving in buildings[J]. Applied energy, 2019, 248: 44-53.

- [21]Fayazbakhsh M A, Bagheri F, Bahrami M. Gray-box model for energy-efficient selection of set point hysteresis in heating, ventilation, air conditioning, and refrigeration controllers[J]. Energy Conversion and Management, 2015, 103: 459-467.
- [22]Cui C, Cai W, Zhang X. A 3-stage online distributed cooperative control based air balancing method for air duct systems[J]. Energy and Buildings, 2020, 214: 109868.
- [23]Jung W, Jazizadeh F. Human-in-the-loop HVAC operations: A quantitative review on occupancy, comfort, and energy-efficiency dimensions[J]. Applied Energy, 2019, 239: 1471-1508.
- [24]Zhang X, Cui C, Cai W J, et al. Introduction of the Air Balancing Technology[M]. Principle, Design and Optimization of Air Balancing Methods for the Multi-zone Ventilation Systems in Low Carbon Green Buildings. Singapore: Springer Nature Singapore, 2022: 1-14.
- [25]Taylor S T. VAV System Duct Main Design[J]. ASHRAE Journal, 2019, 61(4): 54-59.
- [26]ASHRAE, ASHRAE Duct Fitting Database: Version 6.0, American Society of Heating, Refrigerating Air-Conditioning Engineers, 2014.
- [27]American society of heating, refrigerating and air-conditioning engineers. 2015 ASHRAE Handbook: Heating, ventilating, and air-conditioning applications[M]. ASHRAE, 2015.
- [28]NEBB C. Procedural standards for testing adjusting balancing of environmental systems[J]. National Environmental Balancing Bureau, 2005.
- [29]PEDRANZINI F, COLOMBO L P M, JOPPOLO C M. A non-iterative method for Testing, Adjusting and Balancing (TAB) air ducts systems: Theory, practical procedure and validation[J]. Energy and buildings, 2013, 65: 322-330.
- [30]Tamminen J, Ahonen T, Ahola J, et al. Fan pressure-based testing, adjusting, and balancing of a ventilation system[J]. Energy Efficiency, 2016, 9: 425-433.
- [31]Small M L. Non-iterative technique for balancing an air distribution system[D]. Virginia Tech, 2002.

- [32]Chen H, Cai W, Chen C. Model-based method for testing, adjusting and balancing of HVAC duct system[J]. Energy and Buildings, 2016, 126: 498-507.
- [33]Chen H, Cai W, Chen C. Fan-independent air balancing method based on computation model of air duct system[J]. Building and Environment, 2016, 105: 295-306.
- [34]Jing G, Cai W, Zhai D, et al. A model-based air balancing method of a ventilation system[J]. Energy and Buildings, 2018, 174: 506-512.
- [35]Jing G, Cai W, Chen H, et al. An air balancing method using support vector machine for a ventilation system[J]. Building and Environment, 2018, 143: 487-495.
- [36]Zhang J, Li X, Zhao T, et al. Experimental study on a novel fuzzy control method for static pressure reset based on the maximum damper position feedback[J]. Energy and Buildings, 2015, 108: 215-222.
- [37]Li X, Zhao T, Fan P, et al. Rule-based fuzzy control method for static pressure reset using improved Mamdani model in VAV systems[J]. Journal of Building Engineering, 2019, 22: 192- 199.
- [38]Zheng K, Li M, Zhang T, et al. An Air Balancing Method Using Artificial Neural Networks for the Ventilation System[C]. IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society. IEEE, 2020: 4799-4804.
- [39]Cui C, Zhang X, Cai W, et al. A novel air balancing method for HVAC systems by a full data-driven duct system model[J]. IEEE Transactions on Industrial Electronics, 2020, 68(12): 12595-12606.
- [40]Mu Y, Zhang J, Ma Z, et al. A novel air flowrate control method based on terminal damper opening prediction in multi-zone VAV system[J]. Energy, 2023, 263: 126031.
- [41]Svoboda J A, Dorf R C. Introduction to electric circuits[M]. John Wiley & Sons, 2013.
- [42]Larock B E, Jeppson R W, Watters G Z. Hydraulics of pipeline systems[M]. CRC press, 1999.

- [43]Coelho P M, Pinho C. Considerations about equations for steady state flow in natural gas pipelines[J]. Journal of the Brazilian Society of Mechanical Sciences and Engineering, 2007, 29(3): 262-273.
- [44]ASHRAE. ASHRAE duct fitting database, version 6.00. 01: [S]. ASHRAE Atlanta, 2012.
- [45]Xu L, Zheng J, Xiao N, et al. Mean square consensus of multi-agent systems over fading networks with directed graphs[J]. Automatica, 2018, 95: 503-510.
- [46]Lewis F L, Zhang H, Hengster-Movric K, et al. Cooperative control of multi-agent systems: optimal and adaptive design approaches[M]. Springer Science & Business Media, 2013.
- [47]Zhao L, Jia Y, Yu J. Adaptive finite-time bipartite consensus for second-order multi-agent systems with antagonistic interactions[J]. Systems & Control Letters, 2017, 102: 22-31.
- [48]Olfati-Saber R, Fax J A, Murray R M. Consensus and cooperation in networked multi-agent systems[J]. Proceedings of the IEEE, 2007, 95(1): 215-233.
- [49]Seyboth G S, Dimarogonas D V, Johansson K H. Event-based broadcasting for multi-agent average consensus[J]. Automatica, 2013, 49(1): 245-252.
- [50]Oyedepi M O, Mahmoud M S. Couple-group consensus conditions for general first-order multiagent systems with communication delays[J]. Systems & Control Letters, 2018, 117: 37-44.
- [51]Zhu M, Martínez S. Discrete-time dynamic average consensus[J]. Automatica, 2010, 46(2): 322-329.
- [52]Wei J, van der Schaft A J. Load balancing of dynamical distribution networks with flow constraints and unknown in/outflows[J]. Systems & Control Letters, 2013, 62(11): 1001-1008.
- [53]Geng H, Liang Y, Liu Y, et al. Bias estimation for asynchronous multi-rate multi-sensor fusion with unknown inputs[J]. Information Fusion, 2018, 39: 139-153.
- [54]Ren W, Beard R W, Atkins E M. Information consensus in multivehicle cooperative control[J]. IEEE Control systems magazine, 2007, 27(2): 71-82.

- [55]白志刚. 自动调节系统解析与 PID 整定[M]. 化学工业出版社, 2012.
- [56]刘金琨. 先进 PID 控制 MATLAB 仿真[M]. 电子工业出版社, 2004.
- [57]顾星, 钟鸣, 姚玉峰. 基于模糊自适应 PID 的加样臂位置控制[J]. 工程设计学报, 2012, 19(5): 385-390.
- [58]李国勇, 杨丽娟. 神经·模糊·预测控制及其 MATLAB 实现. 第 3 版[M]. 电子工业出版社, 2013.
- [59]范海亮. 基于 PLC 的模糊 PID 复合控制在热泵干燥控制系统中的应用[D]. 南京: 南京农业大学, 2012.
- [60]Garcia F, Thomas P S. A meta-MDP approach to exploration for lifelong reinforcement learning[J]. Advances in Neural Information Processing Systems, 2019, 32.
- [61]Rummery G A, Niranjan M. On-line Q-learning using connectionist systems[M]. Cambridge, UK: University of Cambridge, Department of Engineering, 1994.
- [62]Sutton R S. Generalization in reinforcement learning: Successful examples using sparse coarse coding[J]. Advances in nNeural Information Processing Systems, 1995, 8.
- [63]Watkins C J C H. Learning from delayed rewards[J]. 1989.
- [64]Mnih V, Kavukcuoglu K, Silver D, et al. Playing atari with deep reinforcement learning[J]. arXiv preprint arXiv:1312.5602, 2013.
- [65]Schaul T, Quan J, Antonoglou I, et al. Prioritized experience replay[J]. arXiv preprint arXiv:1511.05952, 2015.
- [66]Van Hasselt H, Guez A, Silver D. Deep reinforcement learning with double q-learning[C]. Proceedings of the AAAI conference on artificial intelligence. 2016, 30(1).
- [67]Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980, 2014.
- [68]Jing G, Cai W, Zhai D, et al. A model-based air balancing method of a ventilation system[J]. Energy and Buildings, 2018, 174: 506-512.

- [69]Jing G, Cai W, Chen H, et al. An air balancing method using support vector machine for a ventilation system[J]. Building and Environment, 2018, 143: 487-495.
- [70]Li B, Cui C, Zhang X, et al. A data-physical hybrid-driven air balancing method for the ventilation system[J]. IEEE Transactions on Industrial Informatics, 2020, 17(10): 6583-6593.
- [71]Cui C, Zhang X, Cai W, et al. A gradient-based adaptive balancing method for dedicated outdoor air system[J]. Building and Environment, 2019, 151: 15-29.
- [72]Zheng K, Li M, Zhang T, et al. An Air Balancing Method Using Artificial Neural Networks for the Ventilation System[C]//IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society. IEEE, 2020: 4799-4804.
- [73]滕子健. 基于物联网的在线滤油机远程监控平台的开发[D]. 东南大学, 2016.
- [74]叶季衡. 湿式冷却塔热力性能在线监测仪与远程监控平台研制[D]. 华南理工大学, 2014.
- [75]贾云祥. 基于云平台的智能窗远程监控系统的设计[D]. 哈尔滨理工大学, 2021.
- [76]陈泽鹏. 基于物联网的设备监控管理系统的研究与开发[D]. 浙江大学, 2018.
- [77]王灿. 基于云平台的物联网远程监控系统研究[D]. 华中科技大学, 2015.

学术成果

学术论文：

- [1] 以第一作者在安徽工程大学学报发表期刊论文一篇，已录用；
- [2] 以第一作者在 **Energy & Buildings** 发表期刊论文一篇，已录用.

致谢

总觉得来日方长，却不知岁月清浅，时节如流。三年的研究生学习生涯，于我而言是最好的礼物，愿走出校园的我们都会成为更好的自己。

桃李不言，下自成蹊。在三年研究生学习生涯中，我最想感谢的是我的导师程凡永教授，从入学开始，他一直悉心指导我，培养我专业严谨的学术能力，是我得以在一个陌生城市求学三年的坚实后盾。很幸运成为程老师的学生，也祝愿老师在今后的日子里一切顺遂、万事胜意。

春晖寸草，山高海深。从蹒跚学步到求学生涯即将结束，我要感谢生我养我，无条件支持我，为我付出的父母。对于社会而言,他们都是非常平凡的人，但在我心中，他们就是这个世界最伟大的人，他们给予了我生命，陪伴我成长，尊重我的选择，给我无限的包容和关怀，祝愿我的父母永远平安喜乐。

爱隔山海，山海可平。感谢我的女孩李品萱，你的存在照亮了我许多个黑暗角落。虽相隔千里，但心动如初，感谢你的鼓励与陪伴，给了我在迷茫时继续前行的动力。

以梦为马，不负韶华。我还要感谢那个不曾放弃的自己，感谢你一直努力热烈真诚的爱与生活。

凡是过往，皆为序章，毕业快乐，未来可期！