同濟大學
TONGJI UNIVERSITY

硕士学位论文

(专业学位)

强化学习算法在空调系统运行优化中的应用研究

姓　　名：王尉同

学　　号：1530990

所在院系：机械与能源工程学院

专业学位名称：工程

专业领域：动力工程

指导教师：潘毅群 教授

二〇一八年三月

同濟大學
TONGJI UNIVERSITY

A dissertation submitted to

Tongji University in conformity with the requirements for

the degree of Master of Engineering

The Application of Reinforcement Learning in HVAC System Operation Optimization

Candidate: Wang Weitong

Student Number: 1530990

School/Department: School of Mechanical Engineering

Discipline: Engineering

Major: Power Engineering

Supervisor: Prof. Pan Yiqun

March, 2018

强化学习算法在空调系统运行优化中的应用研究

王尉同

同济大学

学位论文版权使用授权书

本人完全了解同济大学关于收集、保存、使用学位论文的规定，同意如下各项内容：按照学校要求提交学位论文的印刷本和电子版本；学校有权保存学位论文的印刷本和电子版，并采用影印、缩印、扫描、数字化或其它手段保存论文；学校有权提供目录检索以及提供本学位论文全文或者部分的阅览服务；学校有权按有关规定向国家有关部门或者机构送交论文的复印件和电子版；在不以赢利为目的的前提下，学校可以适当复制论文的部分或全部内容用于学术活动。

学位论文作者签名：

年　　月　　日

同济大学学位论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，进行研究工作所取得的成果。除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人创作的、已公开发表或者没有公开发表的作品的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。本学位论文原创性声明的法律责任由本人承担。

学位论文作者签名：

年　　月　　日

摘要

通过应用全局控制器对空调系统进行运行优化是建筑节能的重要组成部分。建筑运行优化的基本目标是在满足被控区域舒适度要求的前提下尽可能的降低能耗或者能耗消费。为了适应室外不断变化的天气情况和室内负荷变化，如何动态调整空调系统的设定值或运行策略从而提高能效一直是建筑运行优化的研究重点。目前空调系统全局优化控制的方法主要分为基于规则和基于模型两大类，然而基于规则的控制方法需要依赖工程人员经验进行参数设定，基于模型的控制方法的控制效果则严重依赖模型精度且模型建立与校验过程繁琐。因此探索一种新的、有效的全局优化控制方法极具研究价值。

基于强化学习的控制是一种数据驱动的控制方法，具有免模型、目标导向的特点。本文提出将强化学习算法应用在空调系统运行优化中。

首先本文对目前的强化学习算法进行梳理，选择拟合 Q 迭代算法作为主要应用算法。针对强化学习算法特点与空调系统运行优化的实际需求，将强化学习控制器在空调系统运行优化中的应用过程分为准备阶段、初始阶段、探索阶段和运行阶段，并描述了各阶段完整的算法流程。之后，本文通过具体案例对强化学习控制器的控制效果进行验证分析，应用案例包括：（1）强化学习控制器在变风量空调系统中的应用；（2）强化学习控制器在空气源热泵联合电辅热系统中的应用；（3）强化学习控制器在严寒地区太阳能-地源热泵系统中的应用。仿真结果显示，本文提出的基于强化学习算法的空调系统运行方法在满足建筑负荷需求的同时，可以有效降低建筑运行费用。与模型预测控制方法相比，强化学习控制器响应速度更快。此外，该方法具备对先验知识依赖程度低、自适应性强的特点，具备一定的实用性，其应用有助于实现空调系统精细化运行的目标。

最后，本文基于 Python 语言开发了应用工具，用户可以通过编写配置文件和简单脚本的方式使用工具构建用于空调系统的强化学习控制器。

关键词：空调系统，运行优化，强化学习，数据驱动，免模型控制

ABSTRACT

Applying supervisory control to optimize HVAC system operation is an essential part of building energy conservation. The primary goal of building operation optimization is to reduce building energy consumption or operation cost, while maintain acceptable comfort for building occupants. In order to adapt to dynamic indoor disturbances and outdoor weather condition, effective HVAC system setpoints adjustment and operation strategies have been research focus. At present, the rule-based controller and the model-based controller are most widely applied in HVAC system. However, the performance of the rule-based controller depends on parameter setting while the performance of the model-based controller relies on the accurancy of the model. Besides, the model-based controller requires a great deal of effort for model development and calibration. Thus, a new effective globle control method of HVAC system is worthwhile to be explored.

Reinforcement learning controller (RLC) is a type of data-driven controller. Its main features are model-free and goal-oriented. In this study, HVAC system operation optimization based on reinforcement learning controller is proposed.

First, the algorithms for reinforcement leaning are researched and fitted Q-iteration algorithm is selected as the main algorithm. Considering the features of the algorithm and the requirement of HVAC system operation, the application of RLC is divided into four phases, including: preliminary phase, initial phase, exploration phase and operation phase. Detailed process of each phase is described. Then case study is conducted to validate the performance of the reinforcement learning controller. Cases analyzed include: (1) the application of RLC in variable air volume (VAV) system for public building; (2) the application of RLC in air-source heat pump with auxillary heating system for small residential building; (3) the application of RLC in solar-ground coupled heat pump system for buildings in servere cold region. The simulation results show that the proposed RLC method for HVAC system operation can reduce energy cost and at the same time meet the building cooling/heating demand. Compared with model predictive control, RLC also saves the computational time for decision making. In addition, the proposed RLC method requires only little prior knowledge and is capable to adapt to the environment. The

proposed method is practicle and its application could enhance HVAC system operation.

Finaly, a tool is developed using Python programming language. User can utilize this tool to develop reinforcement learning controllers for HVAC system by filling in the configuration file and writing simple scripts.

KEY WORDS:

HVAC system, operation optimization, reinforcement learning, data-driven, model-free control

目录

第 1 章 绪论.....1
1.1 课题背景.....1
1.2 国内外研究现状.....3
1.2.1 空调系统控制优化研究现状.....3
1.2.2 强化学习及其在空调系统中的应用研究现状.....8
1.3 研究目的、意义与主要研究内容.....10
1.3.1 研究目的及意义.....10
1.3.2 主要研究内容.....11
第 2 章 强化学习理论简述.....14
2.1 强化学习问题.....14
2.2 贝尔曼方程.....16
2.3 动态规划.....17
2.3.1 策略迭代.....18
2.3.2 值迭代.....19
2.4 时序差分学习.....19
2.4.1 Q-Learning.....19
2.4.2 SARSA.....20
2.4.3 探索策略.....21
2.5 值函数近似.....22
2.5.1 参数函数近似.....22
2.5.2 非参数函数近似.....23
2.6 批量强化学习算法.....24
2.7 本章小结.....26
第 3 章 强化学习算法在空调系统运行优化中的应用.....27
3.1 已有研究应用情况分析.....27
3.2 本文提出的应用方式.....28
3.3 关于基于强化学习算法的空调系统运行方法的若干问题.....28
3.4 基于强化学习算法的空调系统运行方法流程.....30
3.5 本章小结.....32
第 4 章 应用案例分析.....33
4.1 强化学习控制器在变风量空调系统中的应用.....33
4.1.1 背景介绍.....33
4.1.2 准备阶段.....34
4.1.3 模拟仿真环境.....36

4.1.4 控制效果对比.....38
4.1.5 多区域送风量控制.....41
4.1.6 讨论与分析.....42
4.2 强化学习控制器在空气源热泵联合电辅热系统中的应用.....43
4.2.1 背景介绍.....43
4.2.2 准备阶段.....44
4.2.3 模拟仿真环境.....46
4.2.4 控制效果对比.....47
4.2.5 讨论与分析.....52
4.3 强化学习控制器在严寒地区太阳能-地源热泵系统中的应用.....53
4.3.1 背景介绍.....53
4.3.2 准备阶段.....55
4.3.3 模拟仿真环境.....57
4.3.4 控制效果对比.....59
4.3.5 讨论与分析.....62
4.4 应用特点分析与总结.....63
4.5 本章小结.....64
第 5 章 工具开发.....65
5.1 编程环境简介.....65
5.2 工具设计与运行流程简介.....65
5.3 本章小结.....70
第 6 章 结论与展望.....72
6.1 结论.....72
6.2 进一步工作的方向.....73
致谢.....74
参考文献.....75
附录 部分程序代码.....80
个人简历、在读期间发表的学术论文与研究成果.....86

第1章 绪论

1.1 课题背景

能源是人类社会生存发展的重要物质基础，攸关国计民生和国家战略竞争力[1]。

随着全球经济的迅速发展和世界人口数量的持续上涨，世界能源需求不断增长，据统计，目前世界能源消费年均增长率为 1.8%左右，在当前的全球一次能源消费结构中，石油占比 33.6%，煤炭占比 29.6%，天然气占比 23.8%，水电占比 6.5%，核能占比 5.2%，可再生能源占比 1.3%[2]。在全球变暖，化石能源有限，可再生能源尚未取得突破性进展以及核污染风险尚不能完全避免的背景下，当今世界能源形势仍然严峻，不容乐观。

在我国的能源消费结构中，建筑、工业和交通运输部门是三大“耗能大户”。全球来看，建筑能耗占比约为一次能源消耗的 40%，此外全球超过 30%的二氧化碳排放来自建筑工业。在许多发达国家，建筑能耗占比比重要高于工业能耗和交通能耗。例如，在 2004 年，美国，英国和欧盟的建筑能耗在一次能源消费中的比重分别为 40%，39%和 37%[3]。随着我国经济的发展和居民生活水平的不断提高，中国的建筑工业持续稳定增长。据统计，截止 2015 年，中国的总建筑面积达 573 亿平方米，其中城市居住面积达到 219 亿平方米，农村居住面积达到 238 亿平方米，公共建筑面积达到 116 亿平方米。建筑工业的快速增长伴随着建筑能耗的不断攀升。1996 年，中国的建筑能耗占社会总能耗的比重为 24.1%，在 2001 年达到 27.5%，预计在 2020 年将增长至 35%[4]。

为应对气候变化、实现可持续发展战略，如何不断提高建筑物能效水平已成为当今世界共同面临的问题，建筑节能已成为建设资源节约型社会的关键所在。近年来，与建筑节能相关的标准，规范愈加完善，相关的技术的发展受到越来越多的重视。在我国，空调系统能耗在建筑总能耗中占比约为 1/3，针对空调系统的节能降耗是建筑节能的重要组成部分。目前，基本的建筑空调系统节能方法包括三个方面，即降低冷/热负荷，使用高效的设备和技术，系统设计优化与控制优化。其中系统的控制优化效果与建筑运行能耗紧密相关。据统计，建筑的生命周期通常都是几十年以上，运行阶段能耗在全生命期内占比最高，约为 75.6%，此阶段节能潜力巨大[5]。空调系统全局控制器的作用是针对室外不断变化的天气情况和室内负荷变化情况，动态调整系统中的设定值或者运行规则，从而在满足

被控区域内人员舒适度需求的前提下尽可能地降低空调系统能耗或者运行费用。因此关于空调系统全局控制器的研究一直是建筑节能领域的研究热点。探索如何帮助建筑用户实现建筑能源系统由粗放型管理转变为精细型、科学化管理极具研究价值。

目前，在空调系统中应用的全局控制器主要包括基于规则的控制与基于模型的控制两大类。然而基于规则的控制方法需要依赖工程人员经验进行参数设定，基于模型的控制方法的控制效果严重依赖模型精度且模型建立与校验过程繁琐。因此有必要探索新的、有效的全局优化控制方法。

在现代建筑中，建筑智能化系统应用广泛，楼宇自控系统（Building Automation System，BAS）和建筑能源管理系统（Building Energy Management System，BEMS）是其中重要的组成部分。楼宇自控系统的主要功能是对建筑中的空调系统、照明系统、给排水系统等各类机电系统中设备的运行情况进行监控。建筑能源管理系统对各系统设备能耗情况进行监控记录。通过将二者结合起来，可以保证系统设备的运行状态以及建筑物中的温度、湿度、CO_2浓度等环境参数处于最佳状态，并且有效降低建筑物能耗及运行费用。对于大型公共建筑的运营管理，BA 系统与 BEM 系统起着非常重要的作用，因此，自上世纪 90 年代以来，BA 系统与 BEM 系统发展迅速，并在国内得到广泛应用。BA 系统与 BEM 系统的普及应用不仅为复杂控制策略的实施提供了硬件平台，同时也使得楼宇在运行过程中积累了大量的数据，为分析优化提供了数据基础。因此，为充分挖掘建筑智能化系统的潜力，应探索如何更有效的将信息系统与物理系统结合起来，实现利用信息系统对物理系统进行监测，对反馈回来的监测情况进行分析，进而指导物理系统运行的良性循环。所以在建筑智能化的背景下，利用数据驱动的方式指导空调系统运行、提高建筑运行能效具备广阔的应用前景。

近年来，人工智能领域取得突破性进展。其中机器学习领域相关技术已被广泛应用于互联网，金融，交通等众多领域。机器学习领域的先驱 Arthur Samuel[6]对将机器学习定义为：在不直接针对问题进行编程的情况下，赋予计算机学习能力的一个研究领域。Tom Mitchell[7]将机器学习定义为计算机程序如何随着经验积累自动提高性能。通过应用机器学习方法和数据库相关技术可以实现如图 1.1 所示的从数据到智慧行为的转化。在空调系统节能领域，目前应用较多的为机器学习中的监督学习和无监督学习。监督学习主要学习数据输入输出间的映射关系，用以解决回归和分类任务，具体应用场景包括建筑能耗预测[8]，建筑负荷预测[9]，设备性能预测[10]，系统故障检测与诊断[11]等。无监督学习进行输入数据模式识别，具体应用场景包括对建筑用能模式识别[12]和建筑能耗、设备运行异常值检测[13]等。

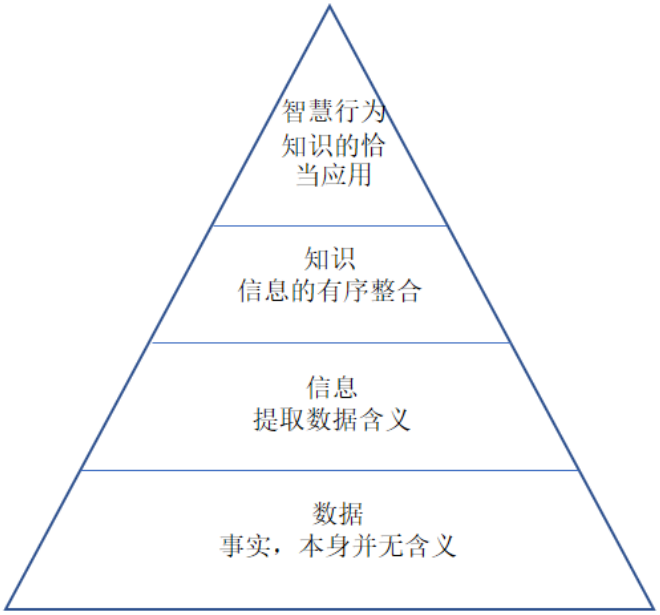

图 1.1 数据到智慧行为的转化过程

强化学习作为机器学习的另一个重要分支，是一种以环境反馈作为输入的、特殊的、适应环境的机器学习方法[14]，它的主要思想是与环境交互和试错，利用评价性的反馈信号实现全链路决策的优化。2017 年，AlphaGo Zero 在围棋比赛中以总比分 100:0 击败 AlphaGo，其主要依赖的关键技术就包括强化学习。基于强化学习的控制方法与人类通过学习获取经验的方式非常相似，展示出了求解复杂控制问题的有效性。随着强化学习在算法和理论上的研究深入，特别是强化学习的数学基础研究取得突破性进展之后，强化学习算法在实际的工程优化和控制中得到了广泛的应用。与基于模型的控制方法的特点相反，基于强化学习的免模型控制方法是一种数据驱动的控制方法，通过不断尝试，利用反馈的信息来更新控制策略，降低了对先验知识的依赖。

目前关于空调系统运行优化的研究和关于强化学习的研究很多，少部分专家与学者已经开展了在二者交叉领域的研究，本文将在前人研究的基础上，针对如何将强化学习应用在空调系统全局控制优化中进行进一步探索，从而提高全局控制器的自适应性，降低其对先验知识的依赖，并通过应用案例对提出的控制方法进行分析验证。

1.2 国内外研究现状

1.2.1 空调系统控制优化研究现状

Wang[15]等人将空调系统的控制方法进行了分类，总体来说目前空调系统的控制器可以分为局部控制器（Local controller）和全局控制器两大类（Supervisory controller）。

局部控制器为底层的控制器，主要根据局部组件特性来考虑组件的控制，通过调节部件使得空调系统能够正常运行。局部控制器可以进一步分类成序列控制（Sequencing control）和过程控制（Process control）。序列控制定义控制空调设备的启停的顺序及其相关条件[16]，亦被称作启停控制。序列控制包括主机启停控制、冷却塔启停控制、水泵启停控制以及风机启停控制等。过程控制是通过测量状态量或者同时测量状态量和扰动量的返回值来调整控制变量，空调系统中典型的过程控制器主要是 P 控制器、PI 控制器、PID 控制器等。相关的研究包括暖通空调系统中的对象系统辨识[17]，以及控制器参数自动整定[18]等。

全局控制器是上层的控制器，是从系统角度出发，充分考虑影响系统控制目标实现的各种因素来对局部控制器进行调整，从而最小化或最大化价值函数。在空调系统控制领域，全局控制器的基本目标为在满足被控区域舒适度要求的前提下尽可能的降低能耗或者能耗消费，全局控制器通过适应室外不断变化的天气情况和室内负荷变化情况，动态调整系统中的设定值或者运行规则从而提高建筑能效[19]。如图 1.2 所示，目前应用在空调系统中的全局控制方法可以分为基于模型的全局控制方法(model-based supervisory control method)，混合全局控制方法(hybrid supervisory control method)，基于性能映射的全局控制方法(performance map-based supervisory control method)和免模型的全局控制方法(model-free supervisory control method)。

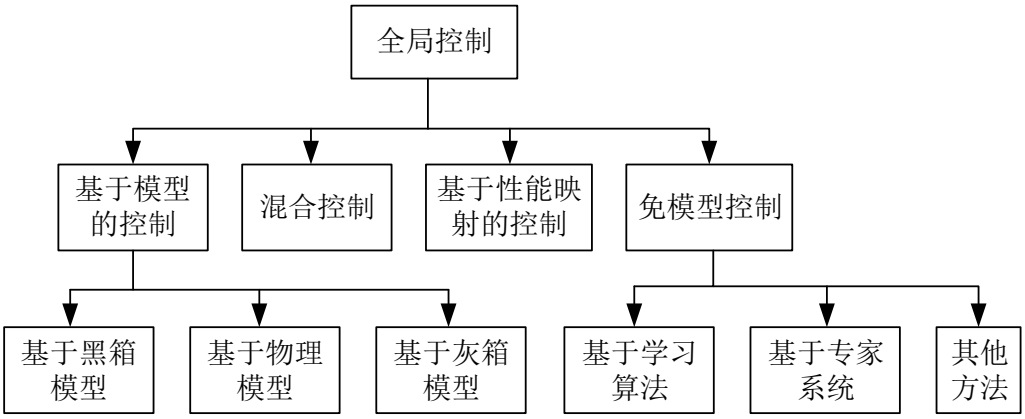

图 1.2 空调系统中的全局控制方法

在基于模型的全局优化控制方法中，用来实现全局控制器的主要工具包括建筑模型、设备模型、系统模型以及优化算法。模型的主要用途是预测系统能耗或

能耗费用、室内环境变化以及修改了控制设定后的系统响应。在每个采样时刻，通过将优化算法应用在搭建好的模型中，求取此工况下能耗最低或者运行费用最低的控制参数。通过这种方式获得的控制策略可以针对室内外变化的情况进行快速响应。按照模型种类的不同可以将控制方法分为基于纯物理模型（physical model-based）的控制方法、基于灰箱模型(gray-box model-based)的控制方法和基于黑箱模型控制(black-box model-based)方法。

基于纯物理模型的控制方法首先对系统和控制过程进行描述，通过关于系统或控制过程的先验知识来确定用于预测系统性能的模型。模型的形式为基于能量守恒定律、质量守恒定律、动量守恒定律等构建的数学方程组。House[20]等人描述了通过构建质量守恒和能量守恒方程的空调系统最优化控制的系统方法，方法充分考虑了空调系统部件间的耦合关系，多区域空调系统和相关的变量。通过把连续控制变量在时间维度上进行离散从而将无限维度的控制优化问题转变为有限维度的控制优化问题，并应用非线性规划技术进行求解。Zaheer-Uddin[21]等人针对建筑最优控制和次优控制开展了相关研究，通过物理模型和优化算法求取最优控制策略和次优控制策略。仿真结果显示控制策略中的多个控制变量可以同时被优化，从而提高系统响应和运行效率，多级优化技术是获取空调系统全局控制策略的有效工具。Wang[22]等人提出了变风量空调的全局控制方法，通过使用简化的物理模型预测系统响应和系统性能，通过遗传算法求解多控制变量的最优解。该方法是空调领域首次应用遗传算法对系统控制进行优化。仿真结果显示这种在线全局控制方法可以充分考虑系统的特点和系统变量间的相互关系，从而提高整个系统的能效。Lebrun[23]等人展示了国际能源署研究项目 Annex 17 的应用实例。Annex 17 主要目标是开发用于建筑环境控制的全局控制策略以及通过模拟仿真的方式对建筑能源管理控制系统中的控制策略进行测试和评价。Sud[24]和 Wang[25]等人开发了多个建筑空调系统仿真平台，可以通过使用仿真平台中已经构建好的详细的和简化的物理模型对多种全局控制策略进行能效评价和经济性分析，基于仿真平台的应用案例显示，与应用局部控制器相比，应用全局控制策略可以降低系统能耗或者系统运行费用。并且此类仿真平台可以在现场测试前帮助测试和评价不同的控制策略，从而确定最优控制策略将其应用在真实的建筑空调系统中。Zhang[26]等人将基于物理模型的全局控制应用在建筑可再生能源系统中，使用进化算法求取最优控制策略和近最优控制策略。仿真结果显示与已有的基于规则的控制方法相比系统运行能效提高显著。总体来说，将物理模型应用在全局控制系统中，由于物理模型背后有先验理论知识作为支撑，可以保证模型在较宽范围内的工况保持较为精确地模拟真实情况，然而物理模型的建立需要对部件和系统进行详细分析，模型结构复杂，往往模型中的物理参数众多又不便测量。此外，复

杂的模型结构往往导致计算开销较大，计算速度缓慢。往往需要对模型进行简化，然而如何对模型进行有效的简化且保持足够的计算精度是应用基于物理模型的全局控制的另一大难题。

基于黑箱模型的控制方法使用不需要任何先验知识的黑箱模型模拟建筑和空调系统，模型参数不存在实际的物理意义。目前基于黑箱模型的控制方法主要有基于人工神经网络的（Artificial Neural Networks, ANNs）全局控制方法和基于经验关系（empirical-relationship）的全局控制方法两种。将 ANNs 应用在建筑空调系统开始于上世纪 90 年代初期。Curtiss[27]等人开发了基于 ANNs 的全局控制器用于最小化建筑空调系统能耗，其中控制器包括两个神经网络：训练网络和预测网络。训练网络用来学习控制变量和非控制变量与空调系统总能耗之间的关系。通过将训练网络的权重传递给预测网络对预测网络进行更新。预测网络搜寻控制变量的最佳设定值来最小化系统运行费用。Bradford[28]将上述方法应用在冷水机组控制优化中，网络结构设定为两个隐藏层，分别包含三个和两个神经元，通过历史数据进行验证，运行费用显著降低。关于将 ANNs 应用在空调系统全局控制中的研究有很多，主要通过神经网络强大的记忆特性和非线性逼近能力去学习建筑和空调系统的响应特性和能耗特性。此外除了 ANNs，有众多研究人员将多项式回归模型和经验模型应用在全局控制中，例如 Braun[29]使用二次多项式获取冷机水系统的最优控制策略和近最优控制策略。其中最优控制策略可以通过基于空调系统部件的模型和非线性优化方法获取，冷机，冷却塔，水泵和风机的能耗关系均采用二次多项式来表达。近最优控制策略采用搭建系统与控制变量之间的二次多项式，从而简化模型结构。Lu[30]等人充分考虑了部件间的耦合关系和控制变量间的耦合关系，其中冷机能耗通过经验模型预测，水泵和风机模型通过相对流量比函数关系式进行预测。使用改进的遗传算法对控制变量进行寻优，并通过小型案例对方法的有效性进行了验证。相比 ANNs，基于经验关系的模型结构简单，计算开销相比较小，但预测精度可能有所降低。总体来说将黑箱模型应用在全局控制中可以不再需要建筑和空调系统相关的先验知识，模型建立相对简单，然而，大多数模型表现并不稳定，因为黑箱模型通常只在其训练数据覆盖的工况下的预测是准确的，模型的外延性较差。当训练数据较少时，使用优化算法对控制变量进行寻优时目标函数的返回值无法得到保证。因此，只有当数据积累量较大，且覆盖工况较全时才适合应用基于黑箱模型的控制方法。

灰色模型将物理模型和黑色模型进行了一定程度的结合，常见的灰色模型形式是通过物理关系确定模型的基本结构，简化描述系统和被控过程的数学形式，模型参数具备一定的物理意义，通过实际监测数据确定模型参数。Platt[31]等人使用 RC 模型对被控区域进行建模用于建筑空调系统控制，文章利用实测数据和遗

传算法确定模型中的热阻和热容值。Berardino[32]等人建立了冷机能耗与冷冻水温度间的函数关系用于建筑需求响应。模型参数通过实测数据和最小二乘法计算得出。灰箱模型兼具物理模型和黑箱模型的优点，与物理模型相比提供更高的精确度，与黑箱模型相比提供更好的泛化能力。但同时也由于其既需要分析物理机理，又需要数据来确定模型参数导致其开发难度最高，因此基于灰色模型的全局控制方法实际工程中应用并不多。

以上提到的控制优化实例都是非预测的全局优化控制，即只关心当前控制步长的优化情况，并不关心对未来的影响。近年来，模型预测控制（Model Predictive Control, MPC）被成功应用在化工、制造等领域。在暖通空调领域的应用也引起了广泛的关注。模型预测控制的基本思想是利用模型来预测系统未来的行为和响应，在每个时间步长，通过对未来一段时间窗内在考虑系统的限制性的前提条件下进行滚动优化得到最优的控制策略。Ma[33]等人将模型预测控制应用在蓄冷空调中，控制器根据建筑负荷的预测情况和天气预测情况来对冷水机组和蓄能水箱进行调节。Moroşan[34]等人将分布式模型预测控制应用在建筑环境控制中，由于求解问题的计算复杂度随系统规模成指数型增长，对于大型公共建筑的末端控制计算开销很大，作者提出每个控制器负责自己的控制区域，在每个控制步长，控制器交换自身行为信息，有效的降低了计算开销并提高了模型预测控制可应对的空调系统规模。Oldewurtel[35]等人将模型预测控制应用在房间集成控制中，模型预测控制在考虑天气因素不确定的情况下对建筑环境进行控制，并对此类随机模型预测控制进行了分析。Dong[36]等人将实时模型预测控制应用在空调制冷和采暖系统中，通过对未来的室内扰动和室外天气状况进行预测，可以显著提高建筑能效。Killian[37]等人，Široký[38]等人和 Siddharth[39]等人针对将 MPC 实际应用在建筑运行中遇到的挑战进行了阐述与总结。

总体来说基于模型的全局控制方法研究发展较为成熟，众多理论研究与工程实验已验证了其有效性。但是其应用也存在一些问题，首先，模型建立与模型校验较为耗时耗力，阻碍了其在实际工程项目中的应用。其次，求解得出的控制策略好坏严重依赖模型质量，一旦模型与建筑或空调系统真实情况存在偏差，控制策略质量将无法得到保障。此外模型预测控制中的滚动优化计算开销较大，对硬件设备要求较高。

基于性能映射的全局控制方法是通过对系统进行详细仿真，模拟所有可能出现的工况从而得到性能映射，之后将性能映射应用在空调系统最优化控制中。Sun[40]等人对空调搭建详细的仿真模型，使用序列二次规划获取性能映射。基于性能映射的控制方法降低了现场的计算开销，但只适用于小型建筑空调系统，此外若生成性能映射的仿真模型无法准确刻画系统真实情况将会导致性能映射的

指导意义严重下降。

混合全局控制方法是将多种不同种类的模型进行混合或者将基于模型的全局控制与免模型全局控制结合起来获取全局控制策略。Braun[41]使用混合全局控制方法获取了建筑蓄冷系统的近最优控制策略，作者建立了蓄冷装置的半经验模型，冷机的经验模型，并在仿真的条件下进行了验证，与相关的最优化控制方法相比，可以将运行费用降低 2%。总体来说，通过此方式获取的控制策略可以提供较好的控制效果。但目前相关研究大部分是在仿真情境下进行验证，实际应用相对较少。

基于免模型的全局控制方法不需要建立建筑空调系统的数学模型而直接获得控制策略。目前免模型全局控制方法主要包括基于专家系统的控制方法和基于强化学习的控制方法两种。基于专家系统的控制方法一般来说是基于规则的，具体形式基本为条件判断语句，通过人工构建规则使系统可以针对不同工况进行调整进而提高空调系统能效。Ling[42]等人设计了基于专家系统的全局控制器用于建筑空调系统。实验结果显示在不舒适程度没有很大提升的情况下运行费用显著降低。Coninck[43]等人将基于规则的控制器应用在家用热水制备的需求侧管理中，可以达到净能量节省 3.4%。基于专家系统的控制方法具有结构简单，稳定性强等特点，因此在实际空调控制系统中应用较为广泛。但其参数设定严重依赖于工程人员先验工程经验，而不是通过优化算法求得，动态调整性不强。关于基于强化学习的全局控制方法将在 1.2.2 节进行详细展开介绍。

1.2.2 强化学习及其在空调系统中的应用研究现状

关于强化学习的研究具有悠久的历史，如今，强化学习已经成为人工智能、机器学习和自动控制等领域的研究热点。

强化学习领域的奠基人 Sutton[44]在其《强化学习介绍》一书中给出了强化学习的定义，认为强化学习是一种以“试错”和“回报”为突出特征的交互式机器学习方法。它期望最大化从外部环境中获得的累积奖赏，目的是学习环境状态到动作的映射。

Bellman 等人于上世纪 50 年代提出了著名的动态规划方法用于解决最优控制问题[45]。动态规划是一个十分有效且重要的强化学习方法。从 50 年代起。强化学习技术不断发展，突出了其自主学习的特点，区别于监督学习和非监督学习，成为了独立的机器学习方法。

高阳[14]等对强化学习与其他技术进行了简单的对比。得出以下结论：第一，不同于监督学习技术，强化学习不需要事先提供训练例，因此强化学习是一种在

线学习技术。第二，强化学习也不同于规划技术，主要区别在于规划需要构造复杂的状态图，而强化学习只需要记忆其所处的环境状态和当前策略知识。第三，强化学习也不同于自适应控制技术，虽然两者都有共同的奖赏函数形式，但在自适应控制中，尽管不能事先确定系统的动态模型，但系统模型必须可以从统计数据中估计，而且系统的动态模型必须是固定的。相反在强化学习技术中，并没有这些限制。

庄超[46]指出强化学习在实际应用中依然存在很多的难题，随着状态数量的增加，学习的难度也会呈指数级增加，即“维数灾”问题。因此传统中的利用表格存储值函数的形式已经不再适用，并总结了目前处理“维数灾”问题常见方法。

Mnih[47]等人提出了深度强化学习(deep reinforcement learning)算法，该算法可以处理高维度的输入数据，从高维数据中提取有效的数据表征方式。基于深度强化学习的控制方法与人类通过学习获取经验的方式非常相似。大量游戏测试显示，深度强化学习算法可以达到与专业游戏玩家相近的水准。

DeepMind[48]成功地将强化学习算法运用在围棋比赛中，开发了 AlphaGo Zero 并击败了 AlphaGo，机器在复杂任务上取得了超过人类能力的表现，使得强化学习成为目前机器学习领域的前沿研究方向之一。

随着强化学习在算法和理论上的研究深入，强化学习技术的应用得到了广泛的关注。如今，强化学习技术被越来越多的用于在线学习控制、作业调度、路径规划、电梯调度，智能推荐以及很多其他领域。基于强化学习的免模型控制方法是一种数据驱动的控制方法，通过不断尝试，利用反馈的信息来更新控制策略，降低了对先验知识的依赖。因此吸引了部分研究人员对将其应用在建筑空调系统控制优化上进行了初步的尝试。

Liu[49]等人提出将基于模型的最优化控制与免模型的最优化控制相结合的混合控制方法，将强化学习算法应用在空调蓄能装置运行优化中并通过模拟仿真与实验的方式进行了测试，作者指出使用的经典 Q-Learning 算法存在对于高维度空间学习效率低下的问题。

Dalamagkidis[50]等人将经典强化学习算法用于建筑空调系统中，控制器可以综合考虑建筑人员舒适度的需求、室内空气质量和空调系统能耗，作者使用 Matlab/Simulink 作为仿真环境对经典的强化学习控制方法与传统的启停控制和模糊 PD 控制进行了对比，结果表明，在使用几年的仿真数据训练之后的强化学习控制器的控制表现与其他控制器控制效果大致相等或表现更加出色。

Barrett[51]等人提出了基于强化学习的空调系统自主控制方法，方法通过学习空调设备特性和人员习惯，实现被控区域智能温控。

Li[52]等人提出了多尺度强化学习方法，并将其应用于建筑空调系统中，用来

加速最优控制策略求解过程，通过 Matlab 与 EnergyPlus 联合仿真的方式测试方法效果，与经典 Q-learning 算法相比，在运行初期收敛速度得到了提升。

Yang[53]等人将强化学习控制器应用在低烟建筑中，通过使用 Matlab/Simulink 进行了仿真验证，与基于规则的控制策略相比，空调系统控制效果提升明显。但作者在进行模拟仿真时并未考虑建筑的热惰性。

Urieli[54]等人和 Ruelens[55]等人分别将有模型强化学习算法和免模型强化学习算法应用于热泵空调系统，从而提升热泵运行效率，仿真结果显示与基准策略相比节能显著。

Costanzo[56]等人将强化学习控制器应用在建筑需求响应中，并通过实验进行了验证，定量分析表示强化学习控制策略可以达到 90%的全局最优解。

Wei[57]等人提出了基于深度强化学习的空调系统控制方法，并验证了深度强化学习控制器的可扩展性，但深度强化学习控制器存在训练所需的时间过长的问题。

Vázquez-Canteli[58]等人将批量强化学习算法应用于热泵和储能水箱全局控制中，使用区域建筑能耗模拟软件 CitySim 对算法进行了仿真验证，结果显示控制器可以在满足室内舒适度需求的同时降低系统能耗。

Cheng[59]等人使用 Q-Learning 算法对建筑遮阳和采光进行全局优化控制，作者开发了闭环基于满意度的控制系统，用户可以通过人机交互界面反馈舒适度感受，并在实验室条件下进行了测试，结果显示该控制器可以提升采光舒适度的同时有效地降低能耗，但目前该方法只适用于被控区域只有单个人员的场景。

总体来说，目前相关研究主要集中在探索强化学习算法在不同的空调系统运行优化场景中的表现，其应用方式缺乏系统性的梳理，对基于强化学习的全局控制器的开发流程仍需深入。其次，如何进一步提升强化学习控制器的可靠性，降低控制器学习所需时间仍然是目前主要的研究方向，此外，针对具体的建筑空调系统的应用场景的适用性和应用效果仍需进一步分析讨论与总结。

1.3 研究目的、意义与主要研究内容

1.3.1 研究目的及意义

本课题的主要目的是探索将强化学习技术应用于建筑空调系统运行优化中的完整方法，从而降低对先验知识的依赖，完善目前空调系统控制方法体系中的基于免模型的控制方法。此外，开发构建用于空调系统运行优化的强化学习控制

器所需的软件工具，为所提出的方法在建筑能源管理系统中的应用提供基础。

本课题的理论意义在于对目前已有的较为成熟的强化学习算法进行梳理，提出强化学习在空调系统运行优化中的完整流程，其次，通过应用案例对基于强化学习的控制策略进行验证，与其他控制方法的控制效果进行量化对比分析，总结强化学习控制器的特点及适用场景，为其在空调系统运行优化中的实际应用提供依据。

实际应用价值：

1. 本文提出的基于强化学习的建筑运行优化方法属于数据驱动的方法，无需对建筑和空调系统进行建模，降低了对先验知识的依赖，可应用在建筑能源管理系统中，提高建筑空调系统运行能效。
2. 搭建的强化学习控制器开发工具可以嵌入到建筑能源管理系统中，帮助工程人员快速构建用于建筑空调系统运行优化的强化学习控制器。

1.3.2 主要研究内容

为了探索完善空调控制系统中的免模型控制方法，降低用于建筑空调系统运行优化的全局控制器对先验知识的依赖，本文首先对目前的强化学习理论进行梳理，选取拟合 Q 迭代算法作为基于强化学习的控制策略的主要算法。接下来，对强化学习算法在空调系统运行优化中的应用进行探讨分析，确定基于强化学习的建筑空调系统控制策略的通用流程。然后，通过应用实例对基于强化学习的控制策略进行验证，分析总结了其具体应用特点及适用场景。最终利用 Python 语言编程，开发出构建空调系统强化学习控制器所需工具，可以嵌入在建筑能源管理与控制系统中，指导建筑空调系统运行优化。

本文的主要内容包括以下几个部分：

1. 第二章：对目前的强化学习相关理论进行梳理，包括如何定义强化学习问题以及基于模型的强化学习算法和免模型的强化学习算法，本文重点关注免模型的强化学习算法，详细介绍了其算法代表 Q-Learning 算法以及其相关改进算法拟合 Q 迭代算法，作为基于强化学习的空调控制策略的理论基础。
2. 第三章：分析讨论目前已有的基于强化学习的空调系统控制策略，提出本课题的基于强化学习的空调控制策略的应用理念，即将空调系统免模型控制方法中基于规则的控制方法与基于强化学习的控制方法相结合。并分析了基于强化学习的空调控制策略在实际应用中的存在的若干问题，最后提出了完整的应用流程。

3. 第四章：通过应用案例对本文提出的基于强化学习的空调控制策略进行了验证，具体案例包括：强化学习控制器在变风量空调系统中的应用，强化学习控制器在空气源热泵联合电辅热系统中的应用，强化学习控制器在寒冷地区太阳能-地源热泵系统中的应用。与基准策略对比，量化分析了强化学习控制器的控制效果。此外，通过案例分析，本文总结了基于强化学习的空调控制策略的特点，并基于此分析了在建筑空调系统运行优化中的适用场景。
4. 第五章：使用 Python 语言开发了用于构建建筑空调系统运行优化所需的强化学习控制器的软件工具。介绍了工具设计的思路，总结了软件工具的使用流程，并通过应用案例对工具的详细使用流程进行了展示。

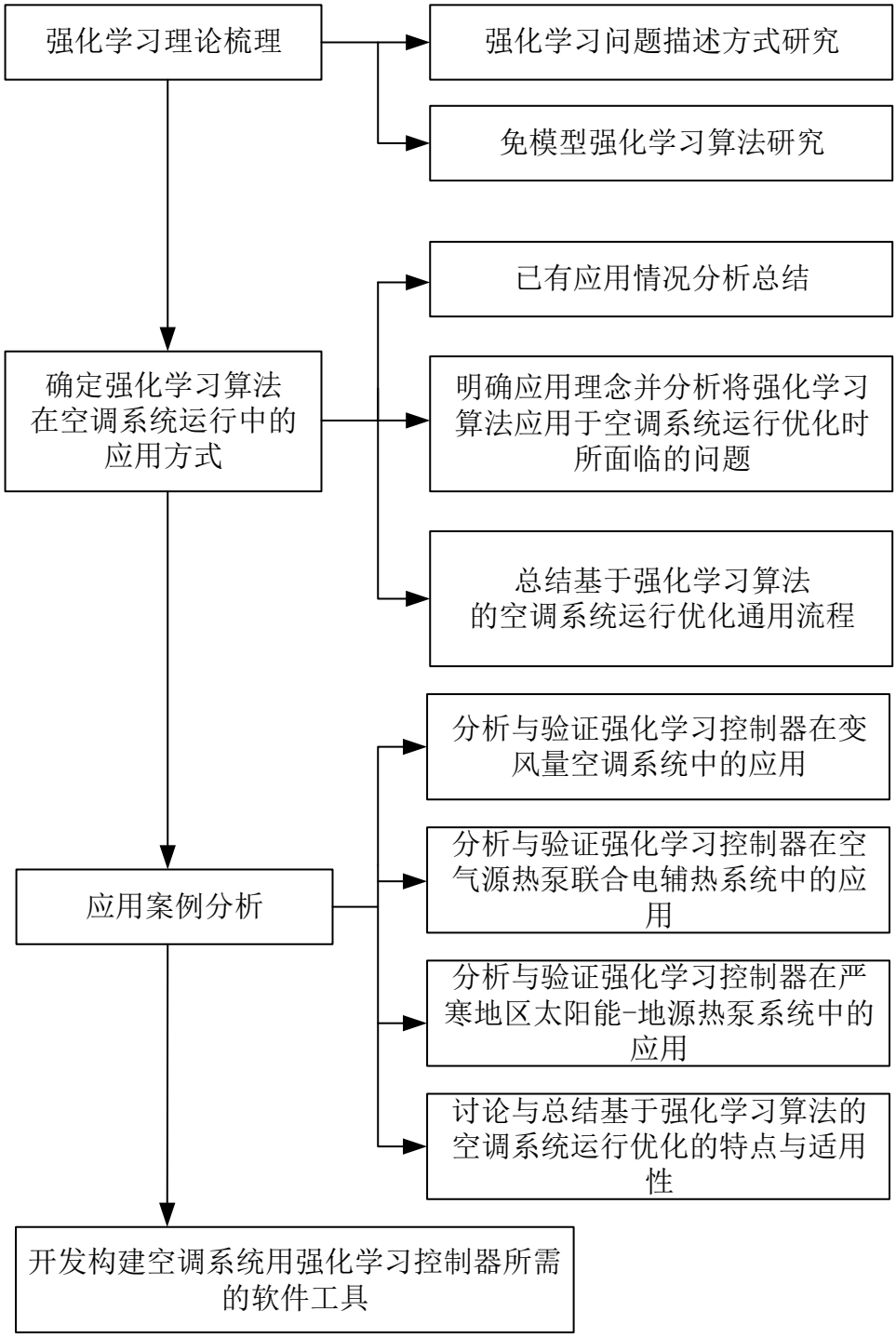

图 1.3 研究路线图

本课题研究路线图如图 1.3 所示。

第2章 强化学习理论简述

空调系统运行优化问题可以被视为空调系统控制变量的全链路优化问题，强化学习方法是一种以环境反馈作为输入的、特殊的、适应环境的机器学习方法，可以用于求解全链路决策优化问题。本章将对强化学习相关理论进行梳理，首先介绍如何使用马尔科夫决策过程描述强化学习任务，此外阐述了目前解决强化学习问题的主要算法及各算法的适用场景，并为求解空调系统运行优化问题确定合适的强化学习算法。

2.1 强化学习问题

目前，强化学习通常包括两方面含义，一方面指强化学习问题，即需要搜索智能系统的行为空间，从而发现系统最优的行为；另一方面是指求解强化学习问题的一类技术，即采用统计技术和动态规划方法来估计在某一环境下的行为效用函数值，从而通过行为效用函数来确定最有行为的技术[60]。

图 2.1 为强化学习问题的一个简单图示。具体来说，强化学习问题可以理解为智能体通过与环境交互，利用环境反馈回来的信息来达到具体的目标，通常智能体是做决策的载体，所有智能体之外的部分都被称作环境。智能体与环境的交互是指，在每个环境状态下智能体选择一个动作，环境响应相应的状态和评价性信号（又称奖赏）。强化学习任务的目的是不断提高智能体的决策能力，使得整个决策链路的累积奖赏最大化。

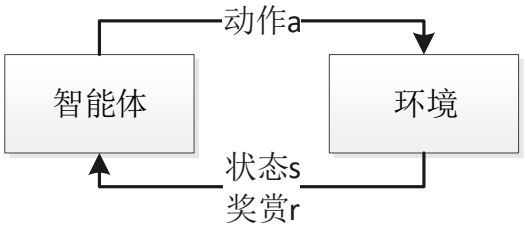

图 2.1 强化学习图示

强化学习问题的数学描述为马尔科夫决策过程（Markov Decision Process, MDP），包括 S 状态空间，A 动作空间，P 转移函数和 R 奖赏函数：

状态空间 S 是状态的集合，$s_t \in S$表示智能体在 t 时刻获取的环境状态量。

动作空间 A 是所有可能动作的集合，$a_t \in A$表示智能体在 t 时刻所采取的动作。

转移函数 P 的形式如式(2-1)所示：

$$s_{t+1} = f(s_t, a_t, w_t) \tag{2-1}$$

转移函数描述的是智能体在 t 时刻环境状态为 s_t时执行动作 a_t，并在随机扰动为 w_t 的情况下，状态转为 s_{t+1} 的概率。随机扰动服从概率分布 $w_t \sim p_w(\cdot\,|s_t)$，$p_w(w_t|s_t)$为出现 w_t的概率。

奖赏函数 R 描述的是在每次环境状态从 s_t转为 s_{t+1} 之后，智能体根据式(2-2)得到的奖赏值。

$$r_t = g(s_t, a_t, w_t) \tag{2-2}$$

马尔科夫决策过程的重要特点是马尔科夫性质，马尔科夫性质是指奖赏 r_t 与环境的下一时刻的状态 s_{t+1} 仅与当前的状态 s_t和当前采取的动作 a_t有关，之前时刻的状态及选取的动作无关。马尔科夫性质的数学描述可以总结为式(2-3)与式(2-4)。

$$p(s_{t+1}|s_t, u_t, s_{t-1}, u_{t-1}, \dots, s_1, u_1) = p_w(s_{t+1}|s_t, u_t) \tag{2-3}$$

$$p(r_t|s_t, u_t, s_{t-1}, u_{t-1}, \dots, s_1, u_1) = p_w(r_t|s_t, u_t) \tag{2-4}$$

马尔科夫性质意味着当前时刻的状态包括了用来预测未来状态即奖赏的所有有效信息，是强化学习算法的重要假设基础。然而在实际工程应用中，一方面由于传感器精度限制环境的精确状态难以获取，另一方面由于应用传感器监测数据的成本限制导致部分维度信息缺失，使得此强化学习问题从严格意义上来讲不具备马尔科夫性质，即具有不完全信息的马尔科夫决策过程（Partially Observable Markov Process, POMDP）。目前已有强化学习算法如 Q-Learning 算法可以求解具有不完全信息的马尔科夫决策过程问题。

在强化学习任务中，学习的目标就是找到能使长期累积奖赏最大化的策略，数学形式如式(2-5)所示，即为从状态到控制动作映射关系的策略函数，通过执行策略函数即可最大化累积奖赏。

$$a = \pi(s) \tag{2-5}$$

长期累积奖赏有多种计算方式，常用的计算方式如式(2-6)所示的“T 步累积奖赏”和式(2-7)所示的“ γ 折扣累积奖赏”。

$$J_T = E(\frac{1}{T}\sum_{t=1}^{T} r_t) \tag{2-6}$$

$$J_\gamma = E(\sum_{t=0}^{+\infty} \gamma^t r_t) \tag{2-7}$$

式中，r_t表示第 t 个步长的奖赏值，E 表示对所有随机变量求期望。

由于空调系统运行优化问题中存在很多不确定性，例如天气情况的预测精度随着预测时刻距离当前时刻的时长的增加而下降，此外不确定性在时间维度上还会产生累积，若按照“T 步累积奖赏”的定义，将未来一段时间范围内的每个步长的奖赏设为同等重要性不尽合理，故对于空调系统运行优化问题，宜采用“γ 折扣累积奖赏”作为长期累积奖赏。

2.2 贝尔曼方程

由 2.1 节可知，强化学习的目标是搜寻最优控制策略函数 $a=\pi(s)$，通过执行此策略函数，可以最大化长期累积奖赏。结合式(2-2)，式(2-5)和式(2-7)可将长期累积奖赏整理为式(2-8)：

$$J_{\gamma}^{\pi}(s_0)=E(\sum_{t=0}^{+\infty}\gamma^t g(s_t,\pi(s_t),w_t)) \tag{2-8}$$

式中，随机扰动服从概率分布 $w_t\sim p_w(\cdot\,|s_t)$，$p_w(w_t|s_t)$为出现 w_t的概率。γ 为折扣因子，取值在 0 到 1 之间。

强化学习中的评估策略带来的累积奖赏分为值函数(state value function)和状态-动作值函数(state-action value function)两种。

令函数 $Q_{\gamma}^{\pi}(s,a)$ 表示从环境状态 s 开始，执行动作 a 之后再按照控制策略 π 执行动作所获得的γ 折扣累积奖赏，此函数即为状态-动作值函数，状态-动作值又称 Q 值。状态-动作值函数的数学描述见公式(2-9)：

$$Q_{\gamma}^{\pi}(s,a)=E_{\pi}(\sum_{t=0}^{+\infty}\gamma^t r_t\mid s_0=s,a_0=a) \tag{2-9}$$

进一步展开可写为式(2-10)的形式：

$$Q_{\gamma}^{\pi}(s_0,a_0)=E_{\pi}(g(s_0,a_0,w_0)+\gamma Q_{\gamma}^{\pi}\big(f(s_0,a_0,w_0),\pi\big(f(s_0,a_0,w_0)\big)\big)) \tag{2-10}$$

上式即为使用状态-动作值函数的贝尔曼等式，可以看出使用任何策略均可获得一个状态-动作值函数，满足式(2-11)要求的状态-动作值函数即为最优状态-动作值函数。

$$Q^*(s,a)=\max_{\pi}Q_{\gamma}^{\pi}(s,a) \tag{2-11}$$

接下来从状态-动作值函数角度出发，如式(2-12)所示，最优控制策略为在某一环境状态 s 下挑选出可以使状态-动作值最大的动作。

$$\pi^*(s)=\arg\max_{a}Q_{\gamma}^{\pi^*}(s,a) \tag{2-12}$$

最优控制策略 π^* 满足贝尔曼方程(2-13)：

$$Q_{\gamma}^{\pi^*}(s,a) = E(g(s,a,w) + \gamma \max_{a} Q_{\gamma}^{\pi^*}(f(s,\pi^*(s),w),a)) \tag{2-13}$$

令函数 $V^{\pi}(s)$ 表示从环境状态 s 出发，通过执行控制策略 π 所获得的累积奖赏，此函数即为状态值函数，状态值又简称 V 值，状态值函数的数学描述见公式(2-14)：

$$V_{\gamma}^{\pi}(s) = E_{\pi}(\sum_{t=0}^{+\infty} \gamma^t g(s_t,\pi(s_t),w_t) \mid s_0 = s) \tag{2-14}$$

进一步展开可写为式(2-15)的形式：

$$V_{\gamma}^{\pi}(s_0,a_0) = E_{\pi}(g(s_0,a_0,w_0) + \gamma V_{\gamma}^{\pi}(f(s_0,a_0,w_0)\,)) \tag{2-15}$$

上式即为使用状态值函数的贝尔曼等式，同样，使用任何策略均可获得一个状态值函数，满足式(2-16)要求的状态值函数即为最优状态值函数。

$$V^*(s) = \max_{\pi} V_{\gamma}^{\pi}(s) \tag{2-16}$$

通过状态值函数可以按照式(2-17)获取最优控制策略 π^*，式中 s' 为环境经历一个控制步长后到达的下一个环境状态。

$$\pi^*(s) = \arg\max_{a}(g(s,a,w) + V_{\gamma}^{\pi^*}(s')) \tag{2-17}$$

使用状态值函数的最优贝尔曼等式可以写为式(2-18)。

$$V_{\gamma}^{\pi^*}(s) = \max_{a} E\,(g(s,a,w) + \gamma V_{\gamma}^{\pi^*}\big(f(s,\pi^*(s),w)\big)) \tag{2-18}$$

由上述介绍可知，贝尔曼方程形式非常简洁，但却是强化学习算法的基础。从公式上看，当前状态的 V 值和下一状态的 V 值以及当前步长的奖赏有关。同样当前状态-动作对的 Q 值和下一状态-动作对的 Q 值以及当前步长的奖赏有关。意味着 Q 值和 V 值是可以通过迭代来进行计算的，当求得最优值函数时即获得了最优控制策略，本章后续小节将介绍获取到最优值函数的强化学习方法。此外，与状态值函数相比，通过动作-状态值函数做决策时不需要转移函数与奖赏函数，因为 Q 值同时依赖环境所处的状态和可能选取的动作。但同时由于状态-动作对的组合数要多于仅考虑状态的组合数，使得动作-状态值函数需要更多的存储空间。

接下来本文将对强化学习算法中的有模型学习(model-based learning)方法与免模型学习(model-free learning)方法做简要介绍。

2.3 动态规划

对于一个强化学习任务，如果此任务对应的马尔科夫决策过程四元组

(S, A, P, R)均为已知，则称此情形为模型已知。即智能体可以模拟出环境的情况，在此条件下的学习被称为有模型学习。动态规划是在模型已知的情况下解决强化学习问题的一类算法。总体来说，动态规划方法可以被分为策略迭代，值迭代和策略搜索三大类，其中策略搜索直接通过优化算法搜寻最优控制策略，前两者应用上一节介绍的通过迭代更新值函数的方式获取最优控制策略，本文接下来对前两种方法进行简要介绍。

2.3.1 策略迭代

策略迭代通过产生一系列的控制策略，每一个策略都对前一个策略进行改进，初始策略通常为随机初始的策略，算法通过不断迭代，重复进行策略评估和策略改进过程，直到控制策略不再改变为止。

表 2.1 为策略迭代的算法描述。

表 2.1 策略迭代算法描述

1. 初始化：

   对于$\forall s \in S$，随机初始$V(s) \in R, \pi(s) \in A(s)$

2. 策略评估：

   重复以下过程：

   设定$\Delta \leftarrow 0$

   对于每个$s \in S$：

   $v \leftarrow V(s)$

   $V(s) \leftarrow \sum_{s',r} p\left(s', r \middle| s, \pi(s)\right)\left(r + \gamma V\left(s'\right)\right)$

   $\Delta \leftarrow \max(\Delta, |v - V(s)|)$

   直至$\Delta < tol$（tol为收敛误差）

3. 策略改进：

   对于每个$s \in S$：

   $a \leftarrow \pi(s)$

   $\pi(s) \leftarrow \underset{a}{\operatorname{argmax}} \sum_{s',r} p\left(s', r \middle| s, a\right)\left(r + \gamma V\left(s'\right)\right)$

   如果$a \neq \pi(s)$，则返回执行第二步策略评估

   π即为得到的最优策略

此方法在进行策略评估的时候需要已知环境模型，在 2.4.2 节中将介绍免模型的策略迭代方法 SARSA。

2.3.2 值迭代

策略的改进本质上与值函数的改进是一致的，因此可将策略的改进视为值函数的改善，于是便可得到如表 2.2 所示的值迭代算法。与策略迭代不同在于，策略迭代是使用贝尔曼方程来更新值函数，得到当前策略的值函数，从而进行后续的策略改进，而值迭代是使用贝尔曼最优方程来更新值函数，从而直接获得最优控制策略的值函数。

表 2.2 值迭代算法描述

对于$\forall s \in S$，随机初始$V(s) \in R$

重复以下过程：

设定$\Delta \leftarrow 0$

对于每个$s \in S$：

$v \leftarrow V(s)$

$V(s) \leftarrow \max_a \sum_{s',r} p\left(s',r \middle| s,a\right)\left(r + \gamma V\left(s'\right)\right)$

$\Delta \leftarrow \max(\Delta, |v - V(s)|)$

直至$\Delta < tol$（tol为收敛误差）

策略$\pi(s) = \operatorname{argmax}_a \sum_{s',r} p\left(s',r \middle| s,a\right)\left(r + \gamma V\left(s'\right)\right)$即为最优策略

同样上述方法同样需要已知环境模型。在 2.4.1 节中将介绍免模型的值迭代方法 Q-Learning。

2.4 时序差分学习

在强化学习任务中，环境的转移概率和奖赏函数往往是未知的或者很难获得。如果学习算法不需要对环境建模，则为免模型学习算法，免模型学习算法是本文关注的重点，时序差分学习是免模型学习方法中的一种。本节将介绍两种典型的时序差分学习方法：Q-Learning 与 SARSA。

2.4.1 Q-Learning

Q-Learning 算法是一种可以求解具有不完全信息的马尔科夫决策问题的强化学习方法。该算法的目标是获取状态-动作对所对应的值函数，即 Q 值函数，用 $Q(s,a)$ 表示，代表状态 s 下执行动作 a 系统将获得的累积奖赏值。表格型 Q-Learning 是指算法将有限个状态-动作对的 Q 值存储在如图 2.2 所示的一张表中格。

$\mathcal{S}$	$\mathcal{A}(s)$			
	1	2	3	4
1	$Q(1,1)$	$Q(1,2)$	$Q(1,3)$	$Q(1,4)$
2	$Q(2,1)$	$Q(2,2)$	$Q(2,3)$	$Q(2,4)$
...	...	...	...	...
12	$Q(12,1)$	$Q(12,2)$	$Q(12,3)$	$Q(12,4)$

图 2.2 存储 Q 值的表格形式

初始阶段表格中的 Q 值均为随机初始值，Q 值更新过程中，智能体不再通过对环境建模，而是与环境不断交互，收集(s,a,r,s')元组形式的样本，使用式(2-19)对表格中的 Q 值进行更新。

$$\begin{gathered}\Delta Q(s,a)=\alpha\left(r+\gamma\max_{a'\in A(s')}Q(s',a')-Q(s,a)\right)\\ Q(s,a)\leftarrow Q(s,a)+\Delta Q(s,a)\end{gathered}\tag{2-19}$$

式中，$A(s)$ 为在状态 s 可以采取的动作集合，s' 为在状态 s 下执行动作 a 后到达的下一状态，奖赏折扣因子 $\gamma\in[0,1]$，表示当前动作对未来奖赏的影响程度，当 $\gamma=0$ 时，智能体只考虑立即奖赏，而 $\gamma=1$ 则表示智能体赋予长期奖赏和立即奖赏同样的重要性。学习率 $\alpha\in(0,1)$为 Q 值的更新速度，较大的学习率可以提高算法的收敛速度，较小的学习率可以提升算法的稳定性。

时序差分项 $\Delta Q(s,a)$ 表示的是“实际值”$r+\gamma\max_{a'\in A(s')}Q(s',a')$ 与“估计值”$Q(s,a)$ 的差值，然而“实际值”也包括下一状态-动作对的“估计值”$Q(s',a')$，Q-Learning 算法通过不断收集样本更新 Q 值使“估计值”越来越接近真实值。在特定条件下，当所有的状态-动作对样本都被收集到时，使用 Q-Learning 算法可以获得最优状态-动作函数[61]。此外，探索与利用是算法中的另一重要的部分，目的是确保所有的状态-动作对样本都可以被收集到。关于探索与利用的内容将在 2.4.3 节展开详细介绍。

2.4.2 SARSA

SARSA 是另一个重要的时序差分算法，同样 SARSA 也是每执行一步更新一次 Q 值，不同的在于如式(2-20)所示时序差分项中的下一状态的 Q 值中的动作采用实际发生的动作，而不是选择时 Q 值最大的动作。由于算法需要(s,a,r,s',a')元组形式的样本，因此算法被命名为 SARSA。

$$\Delta Q(s,a)=\alpha\big(r+\gamma Q(s',a')-Q(s,a)\big)\tag{2-20}$$

与 Q-Learning 算法相比，SARSA 算法特点相对保守，与 SARSA 算法匹配的探索策略应更加贪婪。

2.4.3 探索策略

强化学习中的一个重要的部分就是探索与利用的平衡，在学习过程中，智能体通过不断尝试获取经验，经历一段时间的学习之后，智能体得到具有一定性能的策略，为了测试是否还有改进策略的可能，需要对已有策略建议之外的部分进行探索，使所有可能的状态-动作对样本都被收集到，然而探索过程具有牺牲性能的风险，若仅利用已有知识则很难进一步改进策略。探索与利用是对立的，强化其中一方也就必然弱化了另一方，即所谓的“探索-利用窘境”(Exploration-Exploitation dilemma)。若要达到强化学习累积奖赏最大的目标，必须合理分配探索与利用的比例，接下来将介绍两种常用的探索策略：ϵ-贪心探索策略与 Softmax 探索策略。

ϵ-贪心探索策略[44]可能是目前应用最为广泛的探索策略，即智能体在每个决策时刻以 ϵ 的概率随机选择一个动作，以 $1-\epsilon$ 的概率选择 Q 值最高的动作。其数学描述见式(2-21)。

$$a=\begin{cases}\arg\max\limits_{a} Q(s,a), & if\ rand()>1-\epsilon \\ \text{uniform random choice } a\ in\ A(s), & if\ rand()\le\epsilon\end{cases} \tag{2-21}$$

其中，式中 rand()表示在[0,1]中生成的随机数。

在ϵ-贪心探索策略中，ϵ 的值定义了智能体探索的比例。ϵ 的值可以为固定的值，也可以随着搜集到的样本的增加而减小，一种典型的形式即为 $\epsilon=\max(\epsilon_0-d\times\Delta\epsilon,\ \epsilon_{min})$。$\epsilon$-贪心探索策略的优点是形式简单，在大多数强化学习算法中均表现良好，因此被广泛使用。但 ϵ-贪心探索策略在探索时不考虑每个备选动作的质量，即对于不同 Q 值的动作被选取的概率是相同的，与此相对，Softmax 探索策略则考虑了备选动作的质量。

Softmax 探索策略[62]是基于当前已知的备选动作的质量对探索与利用进行分配，其基本理念是，若采取各动作带来的累积奖赏相当，则选取各动作的概率相当，若采用某些动作带来的收益明显高于其他动作，则采用这些动作的概率也应明显更高。

Softmax 探索策略的动作选取是基于 Boltzmann 分布：

$$p(a|s)=\frac{e^{\frac{Q(s,a)}{\tau}}}{\sum_{a'\in A(s)} e^{\frac{Q(s,a')}{\tau}}} \tag{2-22}$$

式中，参数 $\tau>0$，被称为“温度”，决定了探索与利用的分配比例，$p(a|s)$ 代表在状态 s 下选择每个动作的概率。从式中可以看出，各动作被选取的概率大小取决于其 Q 值的相对大小。τ 越小则 Q 值相对较大的动作被选取的概率就越高，

当 τ 趋于 0 时，则 Softmax 探索策略趋于仅利用。当 τ 趋于正无穷时，则 Softmax 探索策略趋于仅探索。在应用过程中，τ 值可以为固定的值，也可以随着搜集到的样本的增加而减小。

2.5 值函数近似

以上章节介绍的算法均为针对离散状态空间的马尔科夫决策过程，但实际工程中的许多优化决策问题都具有连续的状态空间。一个简单的思路即将连续的状态空间进行离散，之后再应用上述算法求解，但一方面如何将状态空间进行有效的离散是个难题，另一方面针对大规模的状态空间，例如表格型 Q-Learning 算法会面临维数灾难的问题[60]。为了突破以上限制，有必要将泛化方法用于值函数估计。目前常用的泛化方法可以分为参数函数近似(Parametric Function Approximation)方法和非参数函数近似(Non-parametric Function Approximation)方法两种。

2.5.1 参数函数近似

参数函数近似方法是指需事先定义用于进行函数逼近的模型结构的一类方法，在模型结构固定的前提下，通过训练数据调整模型参数。

人工神经网络（Artificial Neural Networks, ANNs）是一种典型的参数函数近似方法。人工神经网络是由人工神经元组成的运算模型。随着其监督学习方法的广泛研究和应用，将人工神经网络的函数逼近能力用于强化学习的值函数逼近的应用显著增加。在将人工神经网络用于值函数近似时，首先需先定义神经网络结构，包括神经网络的输入与输出层，隐藏层数，神经元的激活函数，各层的连接方式等。然后通过学习算法（如反向传播算法）更新神经网络中的权重，使神经网络的最终输出值与期望输出值尽可能的接近。关于神经网络的详细介绍可参考[62]。

目前应用于强化学习值函数近似的神经网络结构主要包括图 2.3 与图 2.4 所示两种结构。二者的主要区别是是否将动作作为神经网络的输入变量。对于图 2.3 中描述的多输出的神经网络，按照式(2-23)计算训练数据的标签值。

$$label(s_t) = \begin{cases} Q(s_t, a) + \Delta Q(s_t, a), & if\ a = a_t \\ Q(s_t, a), & if\ a \neq a_t \end{cases} \tag{2-23}$$

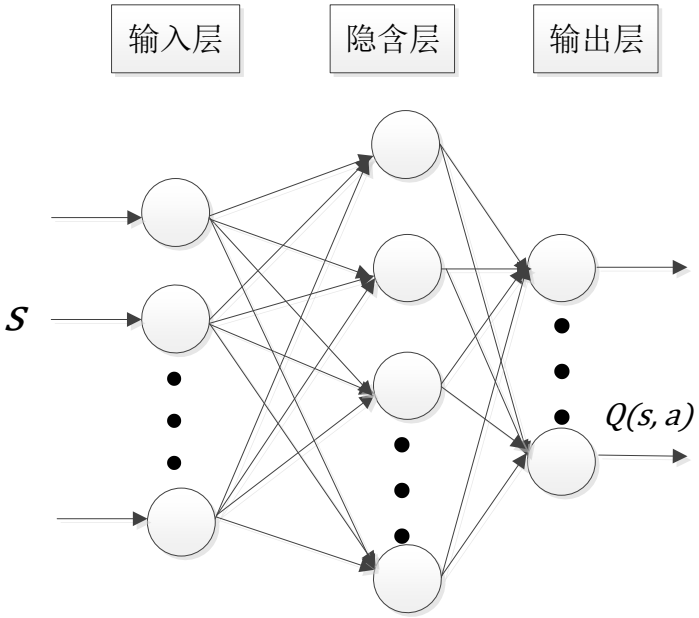

图 2.3 用于 Q 值估计的神经网络结构 A

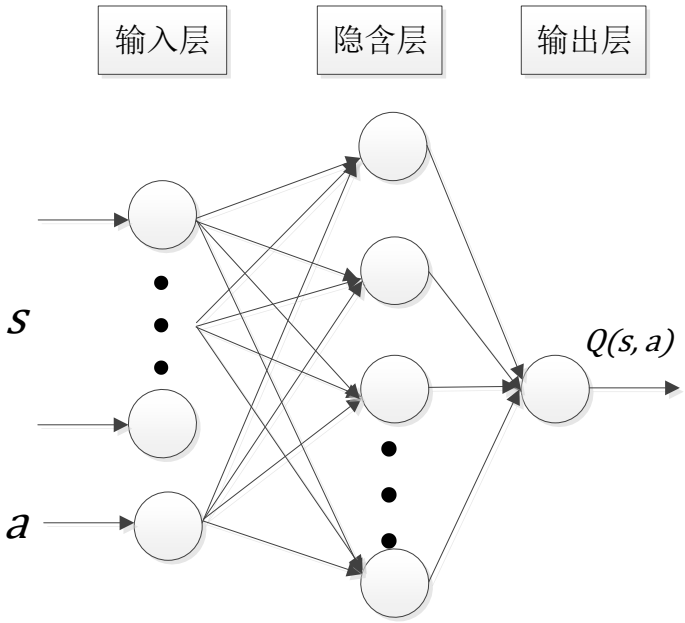

图 2.4 用于 Q 值估计的神经网络结构 B

2.5.2 非参数函数近似

与参数函数近似方法不同，使用非参数函数近似方法无需人为设定确定模型结构，模型的结构与参数均可以通过训练数据确定和调整。

回归树(Regression Tree)是一种典型的非参数函数近似方法。回归树通过学习基于“如果…则…否则”条件形式的决策规则实现目标值的预测，其形式是多

个决策规则的组合，具体结构是由内部节点和页节点组成的树状结构，其中内部节点为条件判断规则，叶节点为数值。在预测时，输入值经过树状结构向前传播直到到达叶节点到达叶节点，此叶节点的数值即为预测值。目前最为常用的构建回归树的方法是 CART 算法[63]，主要思路是通过用平方误差最小化准则进行特征筛选，生成二叉树，使模型输出值逼近期望输出值。回归树方法的主要特点是算法相对简单，计算速度快，此外，回归树方法无需事先假设模型结构，其树状结构下的各节点都是通过训练数据获得。但回归树算法亦存在稳定性低，方差高的缺点，较小的数据扰动很容易造成回归树的结构和参数发生较大的变化。

针对回归树算法的缺点，可以使用集成学习的方式来降低单棵回归树中的高方差而提高了整体的预测能力。集成学习是众多机器学习方法中的一种，其基本思想是通过构建多个不同的预测模型，每个预测模型被称作基学习器，然后以各模型预测结果做某种组合作为最终的输出。计算学习理论的研究结果证明只要基学习器的预测效果比随机猜测稍微好些，集成学习就可以达到相当好的预测精度[64]。常见的以回归树为基学习器的集成学习方法包括随机森林(random forest)[65]、极端随机树(extremely randomized trees)[66]和梯度渐进回归树(gradient boosted regression trees)[67]等。Ernst[68]等人探索了上述集成学习方法在值函数估计上的表现，研究发现极端随机树的表现最好，适合用于 Q 值函数的估计，在使用极端随机树时需要设定树的数目和分裂节点最少所需样本数来控制模型规模，关于极端随机树算法的原理以及算法流程可以参考[66]。

总体来说，若对强化学习问题规模可以进行合理的估计，可以选用人工神经网络作为值函数逼近器，否则可以选用非参数函数近似方法极端随机树作为值函数逼近器。

2.6 批量强化学习算法

在 2.4 节中介绍的 Q-Learning 算法与 SARSA 算法均为在线算法，即每采集到一个样本，算法立刻使用此样本对 Q 值函数更新一次。以此方式更新 Q 值函数存在以下两个问题，一方面后序样本的更新可能会一定程度上覆盖掉之前样本的对 Q 值的更新，尤其是当使用如神经网络等泛化方法用于 Q 值逼近时，后序样本的对神经网络的更新有可能会破坏掉先前样本对神经网络权重的调整。另一方面，在在线算法中，每个样本仅被利用一次，学习效率不高。

为了解决上述问题，可以采用批量强化学习[69](Batch Reinforcement Learning)算法。批量强化学习算法采用历史样本的集合对值函数进行更新，批量强化学习的基本原理是经验回放[70]，所谓经验即为四元组 (s, a, r, s')形式的样本，反复使

用经验的方法即为经验回放。经验回放的前提是智能体所处环境的变化规律应保持不变或变化很小，空调系统运行优化显然符合环境的变化很小这一前提，因此可以应用批量强化学习方法。与在线算法相比，批量强化学习算法的数据利用效率更高，且更稳定。

拟合 Q 迭代(fitted Q-iteration)算法是由 Ernst[68]等人提出的一种常用的批量强化学习方法，拟合 Q 迭代算法的基本思路是首先构建经验的集合 D：

$$D = \{(s_n, a_n, r_n, s'_n)\}_{n=1}^{\#D} \tag{2-24}$$

算法不断迭代建立用于 Q 值函数估计的训练数据，训练数据中的输入数据为 D 中的所有状态-动作对(s_n, a_n)，输出数据按照式(2-25)计算：

$$Q_M(s_n, a_n) = \left(r_n + \gamma_{\max_{a'_n \in A(s'_n)}} \hat{Q}_{M-1}(s'_n, a'_n)\right) \tag{2-25}$$

式中 $\hat{Q}_{M-1}$ 表示算法在上一次迭代得到的值函数逼近器，公式表示每个状态-动作对的 Q 值是通过状态-动作对所对应的奖赏和上一次迭代得到的值函数逼近器计算的下一状态的最优 Q 值估计值求和得到。

在 2.5 节中介绍的两种值函数近似方法神经网络与极端随机树均可用于构建拟合 Q 迭代算法中的值函数逼近器，但二者存在不同之处，首先，使用神经网络需先定义网络结构等，而极端随机树则不需要定义模型结构。另一个不同之处在于经验回放的方式，在每次迭代更新函数逼近器时，由于神经网络算法是通过误差反向传播计算权重的更新值，因此每次更新时神经网络可以只使用经验集合中的部分经验更新权重，被选取的数据称为微批(mini-batch)数据。然而，每一次迭代极端随机树通过训练数据重新确定模型结构，因此必须使用经验集合中的全部经验，导致其计算开销随时间的推进会显著增加。

表 2.3 为算法详细的流程描述。

表 2.3 拟合 Q 迭代算法流程

1. 初始化 Q 值函数逼近器
2. 设定 Q 值函数逼近器更新条件
3. 从环境中获取当前状态量 s
4. 重复以下过程：
   a) 计算状态为 s 下不同动作的 Q 值。
   b) 按照探索策略选择动作 a，执行动作 a 直至下一决策时刻。
   c) 获取当前时刻状态量 s' 以及此控制时间步长内奖赏 r，将(s, a, r, s')存入经验集合。

（续表 2.3）

d) 判断是否需要更新 Q 值函数逼近器

i. 使用经验集合构建用于更新 Q 值函数逼近器的数据集合 $D = \{(s_n, a_n, r_n, s'_n)\}_{n=1}^{\#D}$。

ii. 按照式(2-24)计算D中每个状态-动作对对应的 Q_n值。

iii. 使用数据集合$\{(s_n, a_n, Q_n)\}_{n=1}^{\#D}$更新 Q 值函数逼近器。

e) 将 s' 赋值给 s 。

本课题选取拟合 Q 迭代算法作为解决建筑空调系统运行优化问题的强化学习算法。

2.7 本章小结

本章对强化学习相关理论进行了梳理。首先介绍了如何使用马尔科夫决策过程定义强化学习问题以及强化学习的重要基础贝尔曼等式，接下来对强化学习算法中的有模型学习方法动态规划与免模型学习方法时序差分方法做简要介绍。其中免模型学习方法无需对环境建模，通过与环境交互的方式得到最优控制策略，是本文关注的重点。然而时序差分方法如 Q-Learning 这类表格型的在线学习方法存在数据利用效率低、不稳定等问题，批量强化学习算法中的拟合 Q 迭代算法可以通过值函数近似与经验回放的方式提高数据利用效率，增强算法稳定性。本章详细描述了拟合 Q 迭代算法的计算流程，并最终选择拟合 Q 迭代算法作为解决建筑空调系统运行优化问题的强化学习算法。

第3章 强化学习算法在空调系统运行优化中的应用

本章主要关注如何将强化学习技术用于建筑空调系统运行优化中，在降低对先验知识的依赖的同时提高系统运行能效。本章首先对目前已有研究的应用方法进行讨论分析，在此基础上提出本课题的应用理念与基于强化学习的空调系统运行优化算法流程。

3.1 已有研究应用情况分析

在本课题领域研究初期，研究人员使用完全基于强化学习(Pure Reinforcement Learning)的方式优化空调系统运行，然而完全基于强化学习算法的空调系统运行方法存在策略学习所需时间过长以及可靠性较差等问题，因此不适合将其应用到实际工程中。针对上述问题，目前，有研究人员提出了基于仿真强化学习[49] (simulated reinforcement learning)的空调系统控制方法来克服上述缺点，该方法如图 3.1 所示包括仿真学习阶段和应用学习阶段，在仿真学习阶段需先对建筑空调系统进行建模，智能体通过与模型交互改进策略，仿真模型用于指引智能体进入正确的“区域”，而应用学习阶段通过实际运行数据消除模型与真实情况的偏差带来的影响，从而获取最优策略。

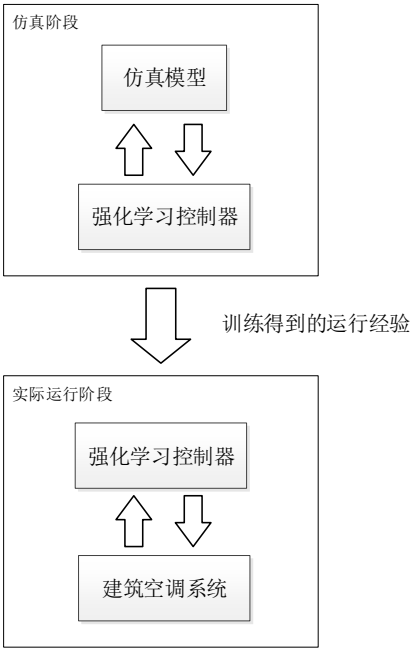

图 3.1 基于仿真强化学习的空调系统控制方法

然而，此种应用方式并未充分利用强化学习算法免模型的特性，与基于模型的控制方法相比不具备任何优势，其控制效果依然依赖模型的质量。总结来说以上两种方式均存在实用性不强的问题。因此，需探索更为实用且有效的应用方式。

3.2 本文提出的应用方式

本文提出的强化学习算法在空调系统运行优化中的应用方式如图 3.2 所示，无须建立建筑空调系统的仿真模型，而是采用将规则与强化学习算法相结合的方式，通过前者将工程经验转化成条件判断语句形式的规则，确定各种工况条件下的系统控制变量的合理区间，通过后者收集分析空调监测系统反馈回来的运行数据，不断改进控制策略，确定各种工况条件下的系统控制变量的最优设定值。

此应用方式的目的在于，一方面规则是工程人员的工程经验最简单的表达方法，通过引入规则降低强化学习控制器所需的探索次数，保证学习型控制器的可靠性，另一方面通过强化学习算法增强基于规则的控制方法的动态调整性。

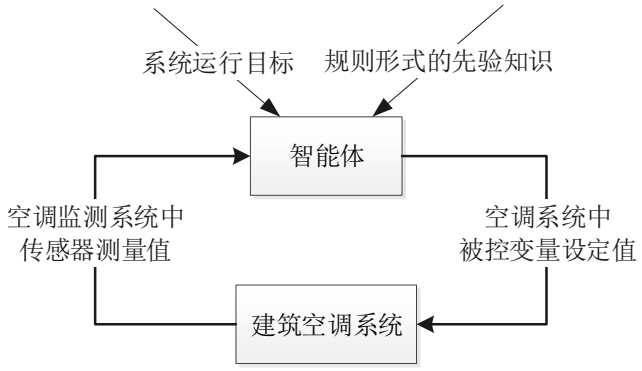

图 3.2 基于规则辅助强化学习的空调系统运行优化

3.3 关于基于强化学习算法的空调系统运行方法的若干问题

本小节将对 3.2 节中提出的应用方式做进一步细化分析，目的在于提高强化学习控制器在空调系统运行优化中的实用性，为确定基于强化学习的空调系统运行优化算法流程提供基础。主要内容包括探讨如何合理地引入规则、将空调系统运行优化问题描述为强化学习问题和总结应用强化学习算法时的注意事项。接下来，本文将其拆分成以下四个问题并针对每个问题进行讨论分析。

问题 1：如何将基于规则的控制方法与基于强化学习的控制方法有效的结合起来？

总体来说，规则可以在两个阶段介入到基于强化学习的空调系统运行优化当

中。首先在空调系统运行初始阶段，可以按照基于规则的控制方法收集运行数据，用于初始强化学习控制器。合理的初始对于强化学习控制器至关重要，因为本文在 2.4.3 提到探索与利用是强化学习算法的重要组成部分，可知合理的初始可以使控制器快速得到较为合理的控制策略，并在此基础上进行改进。

在初始强化学习控制器后，部署应用阶段时，可以通过确立规则对探索空间进行缩减，从而确保智能体在探索过程中避免损害空调系统设备，同时尽可能的满足建筑被控区域舒适度的要求。

问题 2：如何将空调系统运行优化问题合理地描述为强化学习问题，进而利用强化学习算法改进空调系统控制策略？

在 2.1 节中，本文介绍了通过马尔科夫决策过程定义强化学习问题，当应用免模型强化学习算法解决强化学习问题时，则仅需定义马尔科夫决策过程中的状态、动作与奖赏，因此将空调系统运行优化问题描述为强化学习问题，仅需合理地确定空调系统运行优化问题中的状态、动作和奖赏。

奖赏为空调系统运行优化任务的短期目标，因此通常设定为空调系统能效或者当前控制步长的能耗或者运行费用与舒适度的加权等形式。

状态和动作的共同点在于二者皆可以影响控制目标的实现情况，其中动作为影响因素中的可控变量，即为空调系统运行优化中的被控对象。针对空调系统运行优化问题，动作通常为局部控制器的设定值，例如风量设定值，水流量设定值，温度设定值等。

除去被控变量，其他可测量的影响因素均应该被包含在状态中，当状态覆盖的影响因素越全面，则智能体能接受到的有效信息越完整，所学习到的策略越接近最优策略。然而维度过多意味着待探索的状态-动作空间过大，会导致学习到最优控制策略所需的时长过长，因此在状态的定义时应尽可能地使用较少的维度表征全面的信息。针对建筑空调系统运行优化问题，通常需要完整考虑室外天气状况，室内环境状态和空调系统运行情况。

问题 3：将强化学习算法用于解决空调系统运行优化问题时，算法的参数设置有哪些注意事项？

在上一章中，本文比较了各算法的特点，最终选取拟合 Q 迭代算法作为用于空调系统优化的基础算法。在拟合 Q 迭代算法中主要需要定义折扣因子，值函数逼近器的选择和重新训练值函数逼近器的条件。

折扣因子反映的是当前动作对未来奖赏的影响程度。对于仅需要考虑将每一个控制步长内的奖赏最大化的情景，需将折扣因子设为 0，当对空调系统中的设定值进行序列优化时，则需将折扣因子设为大于 0 且小于 1 的实数，当前动作对未来奖赏影响程度越大，则折扣因子的值应设置的越大，折扣因子在强化学习控

制器中的作用相当于模型预测控制中的时间窗。实际应用中需根据特定的空调系统运行优化的问题合理地设置。

值函数逼近器通常可以选择神经网络或者极端随机树，极端随机树无需定义模型结构，但经验回放必须使用经验集合中的全部数据，计算开销较大，考虑到空调控制系统负责计算控制策略的上位机目前的计算能力限制，因此建议对问题规模较小的空调系统运行优化问题建议采用极端随机树作为函数逼近器，否则采用神经网络作为函数逼近器。在使用神经网络时，首先神经网络规模“宜大不宜小”，避免出现规模过小导致的无法存储所有有效信息，其次在构建微批数据 M 更新神经网络中的权重时，微批数据 M 应由两部分组成：所有新收集到的数据和历史数据中随机抽取的部分数据，从而确保所有的数据均能被有效的利用起来。

理想状况下，经验集合每收集到一个新的样本数据，函数逼近器应更新一次。然而受限于空调控制系统中计算资源的计算能力限制，只能降低值函数逼近器的更新频率，通常会采用定期更新的方式，即每新收集 N 个样本，则更新一次。此外，当空调系统运行过程中出现异常情况，例如当被控区域温度违背了舒适度要求，则表明当前的强化学习控制器缺乏指导性，应立即更新值函数逼近器。因此，在实际应用中应将上述二者更新条件结合起来作为重新训练值函数逼近器的条件。

问题 4：在基于强化学习的空调系统运行优化中，是否在值函数估计达到收敛条件之后再停止探索，改为完全利用的方式选择动作？

不应在值函数估计达到收敛条件之后再停止探索。因为一方面在空调系统运行中存在例如室内人行为，室外天气变化等难以预测或者难以测量的随机因素，另一方面，随着运行时间的加长，空调系统老化，换热器结垢，机组部件损耗，空调系统维保等原因，空调系统实际运行性能发生改变。因此空调系统运行优化问题具有信息不完全、不确定的特点，可能会导致值函数的收敛无法得到保证。此外过多的探索可能会导致舒适度的下降和系统能效的降低，因此本文建议人为设定探索时期的长度，在探索时期之后空调系统的强化学习控制器采用完全利用的方式选择动作。

3.4 基于强化学习算法的空调系统运行方法流程

在充分考虑 3.3 节中的四个问题后，本文将基于强化学习算法的空调系统运行方法的应用按时间顺序分为准备阶段、初始阶段、探索阶段和运行阶段四个阶段。

在准备阶段，首先明确待解决的空调系统运行优化问题中的运行目标与被控变量，将其建模为马尔科夫决策过程，之后引入规则，包括确定初始阶段使用的以基于规则的控制方法，定义规则缩减强化学习控制器的探索空间。最后确定强化学习算法中的相关参数。

在初始阶段，采用基于规则的控制方法收集空调系统运行数据，并初始化强化学习控制器。

在探索阶段，针对每个决策时刻，通过空调监测系统获取当前的状态量，并按照设定好的探索策略确定被控变量的值，并记录此控制步长的控制效果。同时利用上述收集到的数据与拟合 Q 迭代算法不断更新值函数逼近器。

在运行阶段，对于每个决策时刻，通过空调监测系统获取当前的状态量，采用完全利用的方式（即选取 Q 值最高的动作）确定被控变量的值，记录此控制步长的控制效果。并继续利用上述收集到的数据与拟合 Q 迭代算法不断更新值函数逼近器。

表 3.1 为基于强化学习算法的空调系统运行方法的完整流程。

表 3.1 基于强化学习算法的空调系统运行流程

准备阶段：

1) 将空调系统运行优化问题建模为马尔科夫决策过程。
2) 设定初始阶段使用的以基于规则的控制方法，定义规则缩减探索空间。
3) 确定探索策略以及探索阶段时长，选择值函数逼近器，设置折扣因子和值函数逼近器更新条件。

初始阶段：

1) 以基于规则的控制方法运行空调系统，收集初始样本。

探索阶段：

1) 使用初始阶段收集到的数据初始化值函数逼近器。
2) 获取当前状态量 s。
3) 重复以下过程：
   a) 计算状态为 s 下不同动作的 Q 值。
   b) 按照探索策略选择动作 a，执行动作 a 直至下一决策时刻。
   c) 获取当前时刻状态量 s' 以及此控制时间步长内奖赏 r，将 (s,a,r,s') 存入经验集合。
   d) 判断是否达到值函数逼近器更新条件，若是则使用经验集合中的数据更新值函数逼近器，若否则执行下一步。
   e) 将 s' 赋值给 s。

（续表 3.1）

运行阶段：

1) 获取当前状态量 $\boldsymbol{s}$。
2) 重复以下过程：
   a) 计算状态为 $\boldsymbol{s}$ 下不同动作的 Q 值。
   b) 选取 Q 值最高的动作 $\boldsymbol{a}$，执行动作 $\boldsymbol{a}$ 直至下一决策时刻。
   c) 获取当前时刻状态量 $\boldsymbol{s}'$ 以及此控制时间步长内奖赏 $\boldsymbol{r}$，将 $\left(\boldsymbol{s},\boldsymbol{a},\boldsymbol{r},\boldsymbol{s}'\right)$ 存入经验集合。
   d) 判断是否达到值函数逼近器更新条件，若是则使用经验集合中的数据更新值函数逼近器，若否则执行下一步。
   e) 将 $\boldsymbol{s}'$ 赋值给 $\boldsymbol{s}$。

3.5 本章小结

本章主要探讨了强化学习算法在空调系统运行优化中的应用方式。首先对目前已有研究的应用方法进行讨论分析并总结了其在应用时存在的主要问题。接下来提出了将基于规则的控制方法与基于强化学习的控制方法相结合的应用方式，分析了本文提出的应用方式在具体实现时面临的若干问题，最后提出了较为通用的基于强化学习算法的空调系统运行方法。

第4章 应用案例分析

本章将通过应用案例展示强化学习算法在空调系统运行优化中的应用过程，并对应用情况进行讨论分析。具体案例包括：（1）强化学习控制器在变风量空调系统中的应用；（2）强化学习控制器在空气源热泵联合电辅热系统中的应用；（3）强化学习控制器在严寒地区太阳能-地源热泵系统中的应用。其中前两个案例使用强化学习控制器解决典型的建筑环境控制问题，案例一侧重分析强化学习控制器的可扩展性，案例二关注使用强化学习控制器解决小规模空调系统运行优化问题，并与最优控制策略的控制效果进行对比，案例三主要关注在空调系统设备难以建模的情况下，通过使用强化学习控制器提高空调系统能效。

4.1 强化学习控制器在变风量空调系统中的应用

4.1.1 背景介绍

本节选取的研究对象为一单层办公建筑，总空调面积为 475 m^2，空调系统为变风量空调系统，强化学习控制器负责调节被控区域送风量，对被控区域送风量设定值进行序列优化，旨在实现在满足室内舒适度需求的同时尽可能的降低空调系统运行费用的目标。

对被控区域进行送风量调节属于典型的建筑环境控制问题，针对建筑环境控制问题，Killian[37]对比了目前已有的三种控制方法：PID 控制，PID 带室外温度补偿控制(PIDc)和模型预测控制(MPC)，并使用图 4.1 所示的场景对三种方法进行了对比。PID 控制器在测量到室内温度与设定值间存在偏差时，通过调整被控变量（如图中供热量）从而消除偏差的方法，当 t_0时刻室外环境温度快速下降时，由于建筑存在热惰性，室内温度逐步下降，PID 控制器根据下降的情况调整供热量，然而随下降幅度越来越大，而供热功率又存在最大值限制，所以导致在运行过程中室内温度在较长时间违背了舒适度条件的要求。PIDc 控制器在 PID 控制方法的基础上又考虑了室外温度大幅变化的情况，然而由于控制过程的大滞后特性，运行过程依然存在室内温度在一段时间违背了舒适度条件的要求的情况。MPC 可以利用预测得到的室外天气未来的变化情况，并结合建筑空调系统模型以滚动优化确定当前的最优控制策略。从基本思想上看预测控制优于 PID 控制。然而精确的模型是 MPC 控制效果能得以保证的必要条件，对于部分建筑与空调

系统，常常难以获取精确又计算效率高的仿真模型。因此，本案例主要关注在不使用模型的前提下，利用可测量的数据、预测得到室外天气未来的变化情况和强化学习算法实现有效的建筑环境控制。

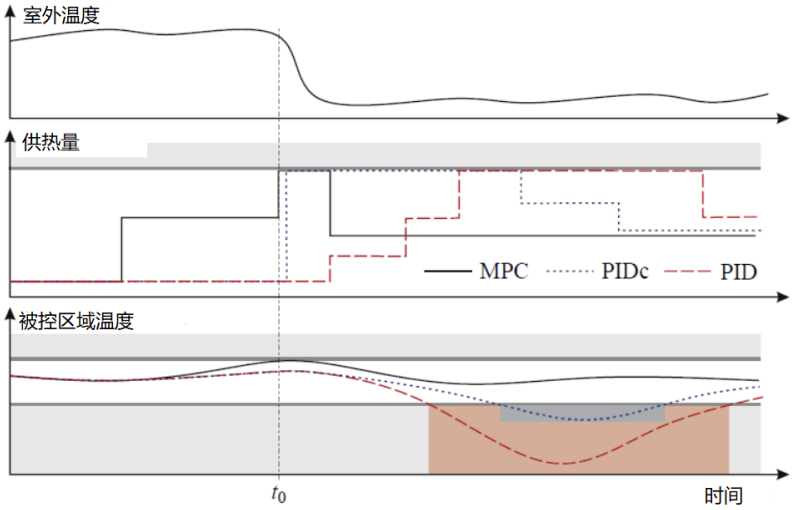

图 4.1 PID, PIDc 与 MPC 的对比[37]

本节首先探讨强化学习控制器负责调节单区域送风量调节的情况，包括准备阶段工作的介绍，模拟仿真环境描述和与其他控制策略进行效果对比，接下来探讨强化学习控制器负责调节多区域送风量调节的情况，最后对强化学习控制器在变风量空调系统中的应用进行分析总结。

4.1.2 准备阶段

根据本文 3.4 节中总结的基于强化学习的空调系统运行优化算法流程，在准备阶段，第一步应先建立此运行优化问题的马尔科夫决策过程模型，首先分析本案例中的影响空调系统运行费用的影响因素如图 4.2 所示。在此基础上先定义动作，即被控变量，在本案例中为被控区域送风量的设定值。确定最小送风量为室内人员所需新风量，最大送风量确定为满足室内 6 次换气次数，将动作区间离散为四个水平$\{a_1, a_2, a_3, a_4\}$，设置控制时间步长为 0.2h（12 分钟）。接下来定义状态，本案例选取时间，室外天气情况，被控区域温度作为状态。其中时间用于反映如室内热扰和分时电价等与时间相关信息。室外天气状况选取当前时刻室外温度，当前室外太阳辐射和预测的未来一小时的温度变化值。被控区域温度应包含室内空气温度和壁面温度两部分，其中壁面温度用于反映建筑蓄热情况，然而在实际工程中通常无法测量。因此，本案例采用过去五个时间步长的室内空气温度的均值与当前室内空气温度的差值来近似反映建筑蓄热情况，通过此方式可以在状态量仅增加了一个维度的情况下，为控制器提供更完整的信息。

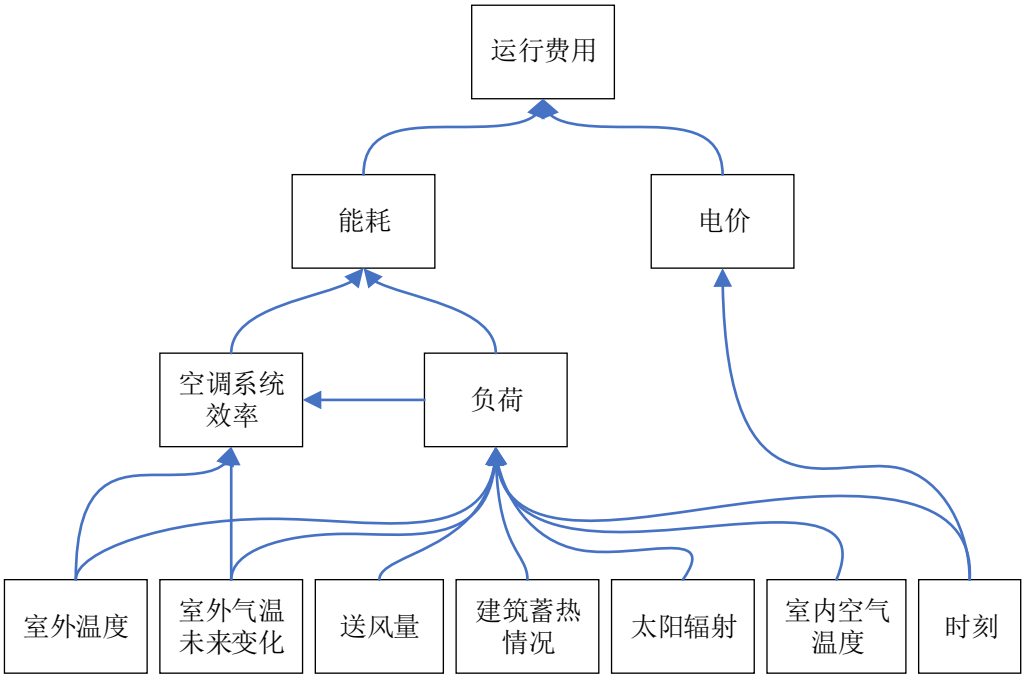

图 4.2 变风量空调系统运行费用影响因素

最后定义奖赏函数，奖赏函数如式(4-1)所示分为两个部分，

$$r_{t-1} = -cost(s_{t-1}, a_{t-1}) + penalty(T_t)$$

$$penalty(T_t) = \begin{cases} -\beta(T_t - T_{lower\ bound})^2 + b, if\ T_t < T_{lower\ bound} \\ -\beta\left(T_t - T_{upper\ bound}\right)^2 + b, if\ T_t > T_{upper\ bound} \\ 0, if\ T_{lower\ bound} \le T_t \le T_{upper\ bound} \end{cases} \tag{4-1}$$

一部分为此控制时间步长内的能耗消费，其中能耗为冷机，冷冻水泵，冷却水泵，冷却塔，风机能耗之和，电价信息如表 4.1 所示，

表 4.1 分时电价信息（工商业电价）

时段	电价（元/kW·h）	时间
谷电	0.279	22:00-6:00
平电	0.719	6:00-8:00、11:00-13:00、15:00-18:00、21:00-22:00
峰电	1.172	8:00-11:00、13:00-15:00、18:00-21:00

另一部分为室内温度超出舒适度范围时的惩罚项，按照 GB50736-2012《民用建筑供暖通风与空气调节设计规范》[71]，夏季室内温度应保持在 24-28℃之间，故取温度上限为 28℃，下限为 24℃。

准备阶段第二步为设定初始阶段使用的基于规则的控制方法以及定义规则对探索空间进行缩减，本案例中初始阶段采用类似于启停控制的运行方式，当被控区域温度超过 27.5℃时，保持最大送风量送风，当被控区域温度低于 24.5℃时，则保持最小送风量送风。同时设置初始阶段时长为系统运行前五天，在初始阶段

结束后，强化学习控制器将在此基础上探索改进控制策略的可能。

关于探索空间缩减，本案例定义两条规则确保系统运行满足舒适度约束，当室内温度大于 27.5℃时的状态，则下个控制步长按照最大送风量送风，避免温度进一步上升，当室内温度小于 24.5℃时，则下个控制步长按照最小送风量送风，降低能耗开销。上述规则用于探索阶段与运行阶段，旨在加速强化学习控制器的学习过程，避免无意义的探索，同时也提高了系统运行的可靠性。

准备阶段第三步的工作包括确定探索策略以及探索阶段时长，选择值函数逼近器，设置折扣因子和值函数逼近器更新条件。

首先探索策略选择 ϵ-贪心探索选择动作，ϵ设置为$\epsilon = \max(\epsilon_0 - d \times \Delta\epsilon, 0.1)$，其中$\epsilon_0$为 0.5，$\Delta\epsilon$为 0.02，$d$为探索阶段运行天数，通过采用上述方式可使在探索初期进行较多的探索，采集不同的状态-动作对的样本。设置第一年运行为探索阶段，即采用ϵ-贪心探索选择动作。从第二年起空调系统即转为运行阶段。

本案例选用神经网络用作 Q 值函数逼近器，本案例采用的神经网络结构如图 2.4 所示，包括三个隐藏层，每个隐藏层的神经元数分别为 32,64,32。前两个隐藏层采用修正线性单元(ReLU)作为激活函数，最后一个隐藏层采用线性单元(Linear)作为激活函数。设定微批数据的大小为 500，微批数据首先包括截止更新时新收集到的数据样本，其他数据为从经验集合中的历史数据随机抽取产生。

最后设置折扣因子 γ 为 0.5，值函数逼近器更新条件为每天更新一次，此外当出现室内温度违背舒适度约束条件的情况时，则意味着强化学习控制器的控制效果仍不理想，因此每当室内温度违背舒适度约束条件时，则立刻更新值函数逼近器。

4.1.3 模拟仿真环境

本案例通过模拟仿真测试强化学习控制器的表现，采用的主要仿真软件为 TRNSYS，其全称为 Transient System Simulation Program，即瞬时系统模拟程序。TRNSYS 软件最初版本源于美国 Wisconsin-Madison 大学 Solar Energy 实验室（SEL），后来经过欧洲的一些研究所共同完善，成为成熟的建筑空调系统模拟仿真软件。

TRNSYS 在建立仿真模型的过程中采用了模块化的思想。在 TRNSYS 中每个模块代表一个设备或者过程，在建立系统模型过程时，使用者只需梳理每个部件的输入与输出，将部件正确的连接起来从而组成完整的系统，并合理的设置各部件参数，不需要过度关注每个部件内部的实现。此外，TRNSYS 软件作为动态仿真软件，可以很精确的模拟系统在部分负荷运行下的特性，比其它软件如

eQUEST 等更具优势。另外，TRNSYS 软件可以与很多专业软件如 EnergyPlus，Matlab 等来进行联合仿真，扩展其模拟仿真能力。虽然 TRNSYS 软件有很多优点，它也一定的局限性，TRNSYS 主要应用于研究分析空调系统的运行情况，而不是侧重于对建筑的热特性进行研究分析，它对建筑的模拟功能较弱，只能设定比较简单的建筑模型，不能像 EnergyPlus 等软件建立详细的建筑模型。但是首先 TRNSYS 中的建筑模型能反应建筑热响应的基本机理，其次本案例主要探讨分析系统的运行特性，因此本文选用 TRNSYS 模拟软件。

使用 Python 语言编写强化学习控制器，TRNSYS 负责建筑及空调系统仿真，TRNSYS 与 MATLAB 通过 UDP 协议，用 Internet socket 的方法进行联合仿真。图 4.3 为联合仿真示意图。

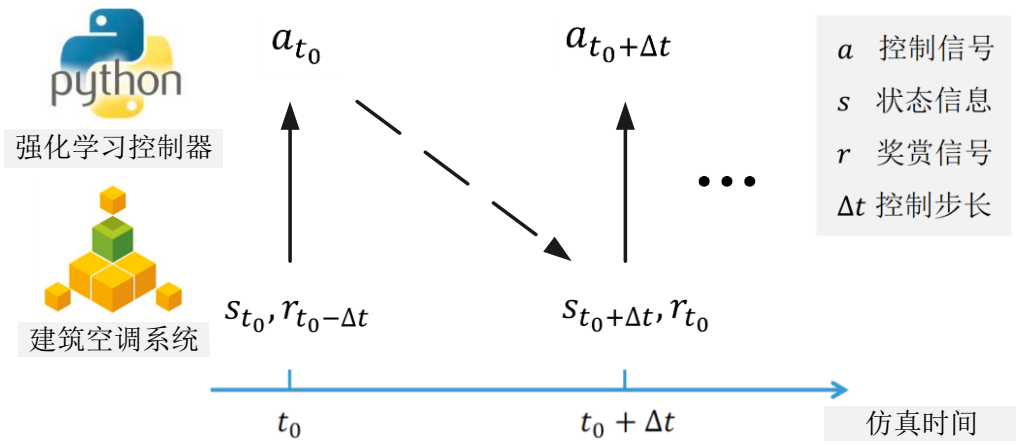

图 4.3 联合仿真示意图

围护结构热工参数的设置参照 GB50189－2005《公共建筑节能设计标准》[72]，具体的热工参数包括：外墙为 0.888 W/m^2·K，屋顶为 0.638 W/m^2·K，外窗为 2.73 W/m^2·K。内热源部分设置人员密度为 0.25 p/m^2，设备密度为 20 W/m^2，照明密度为 11 W/m^2，人员、照明和设备时间表设置如图 4.4 所示。建筑运行时间为制冷季 6.1 - 9.20 间每天 8:00 - 18:00。设定室内送风温度为 15℃，人均新风量为 30 m^3/(h·p)。建筑依靠额定 COP 为 5 的离心式制冷机组提供冷量，冷冻水泵为变频水泵，冷却水泵为定频水泵。在 TRNSYS 中建立好建筑空调系统仿真模型，如图 4.5 所示。

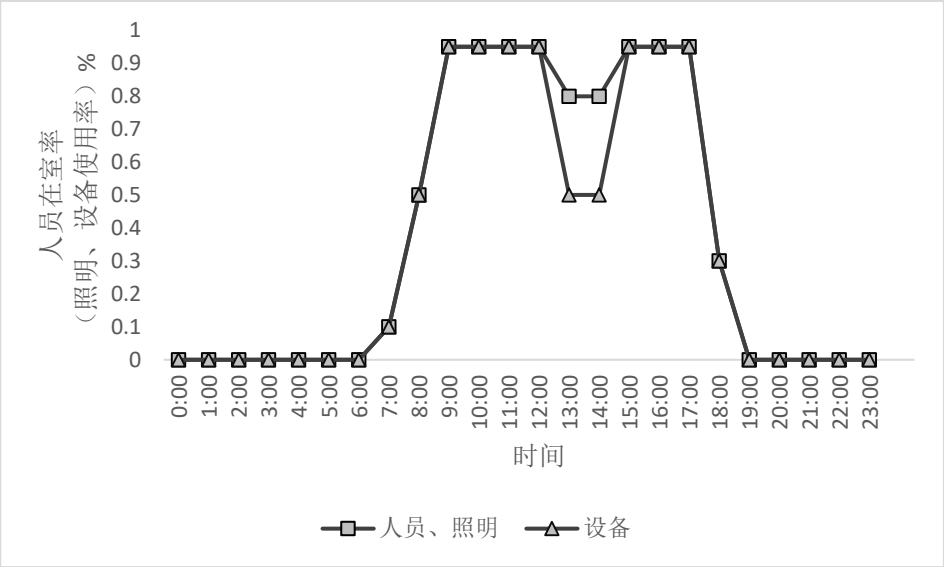

图 4.4 人员、照明和设备时间表

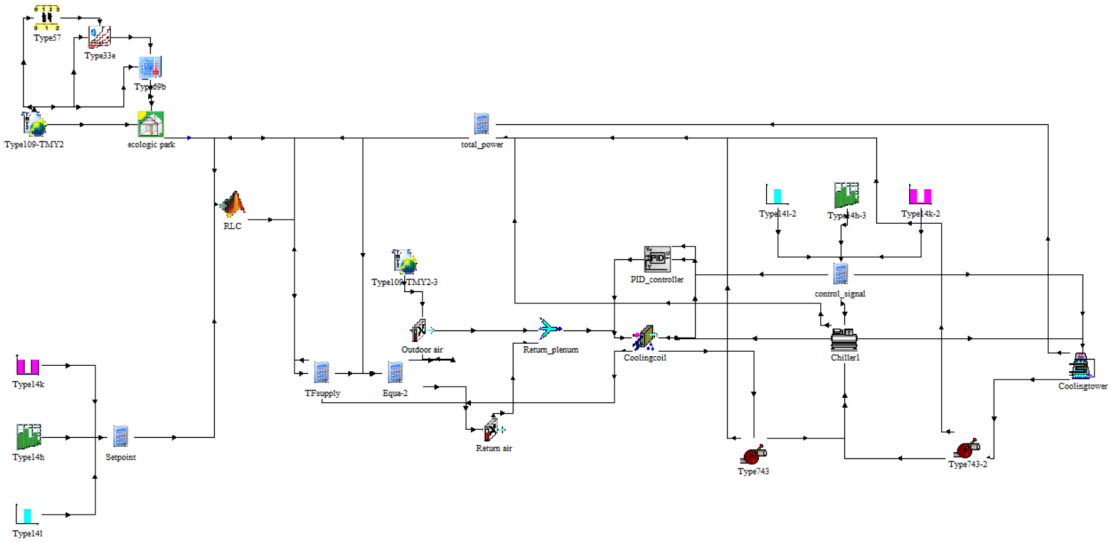

图 4.5 TRNSYS 中建筑空调系统仿真模型

本案例用上海市典型气象年数据模拟气象参数，温度预测值采用真实温度加入 $N(0, 0.2)$ 正态分布随机误差计算得出。

4.1.4 控制效果对比

在分析系统运行效果时，首先关注运行过程中室内环境情况。图 4.6 为节选了部分运行阶段室内温度变化情况，可以看出在室外温度不断变化的过程中，强化学习控制器可以保证在绝大多数时间内满足舒适度的需求。

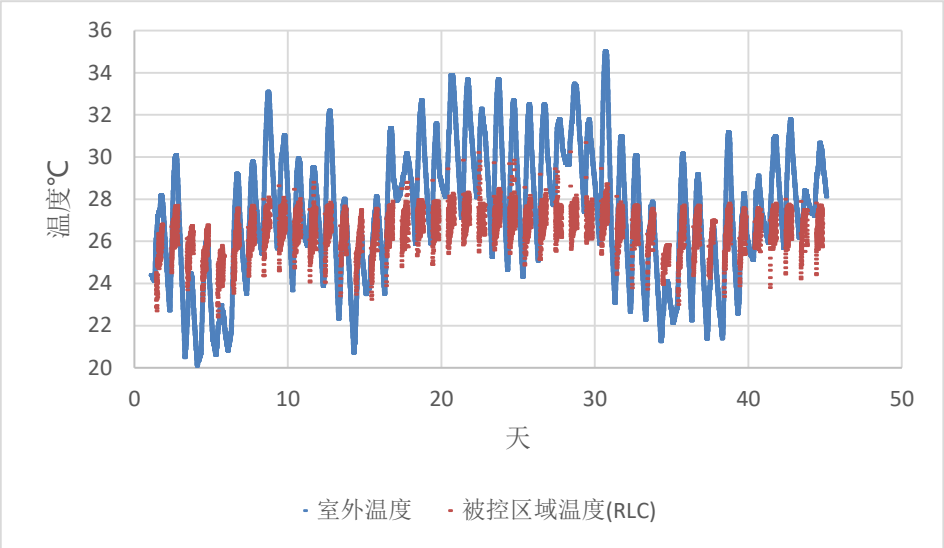

图 4.6 运行阶段被控区域温度变化情况

接下来，为了量化强化学习控制器(RLC)的控制效果，基准控制策略选取类似于启停控制的基于规则的控制策略(RBC)和固定温度设定值的 PID 控制器作为对比对象，RBC 具体设置为当被控区域温度超过 27.5℃时，保持最大送风量送风，当被控区域温度低于 24.5℃时，则保持最小送风量送风，PID 控制器的温度设定值为 26℃，通过对送风量的调整维持室内温度的稳定。

图 4.7 为采用上述两种基准策略时被控区域温度变化情况，可以看出在绝大多数时间内，被控区域温度满足舒适度的要求，此外，采用 PID 控制器时，被控区域温度可保持在设定值 26℃附近小幅波动，采用 RBC 控制策略运行时，被控区域在 24 至 28℃间波动，同时处于 24-26℃的时间较长一些。

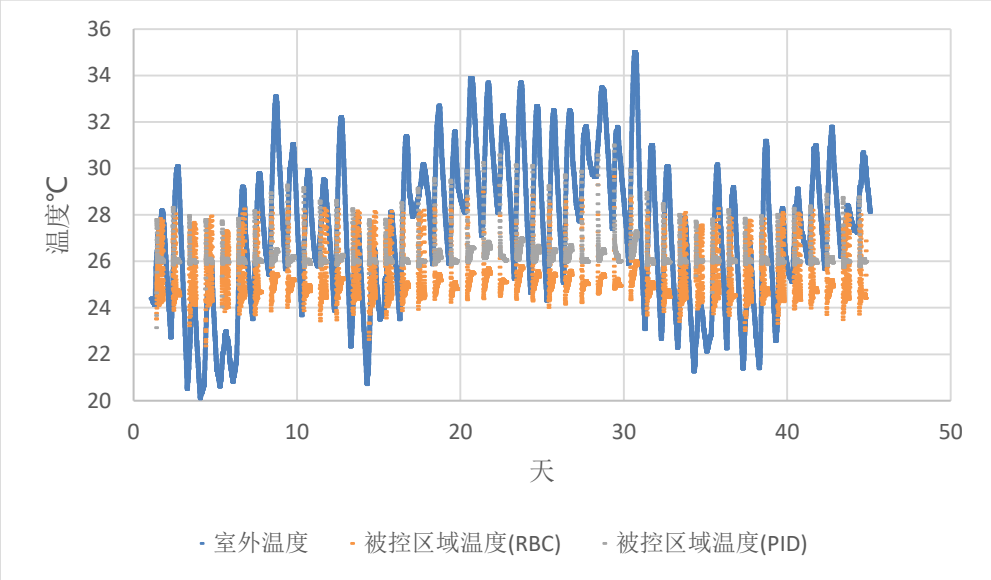

图 4.7 采用基准策略时被控区域温度变化

对采用上述三种控制策略的空调系统年运行费用和被控区域舒适度情况进行统计对比，结果如图 4.8 和图 4.9 所示，

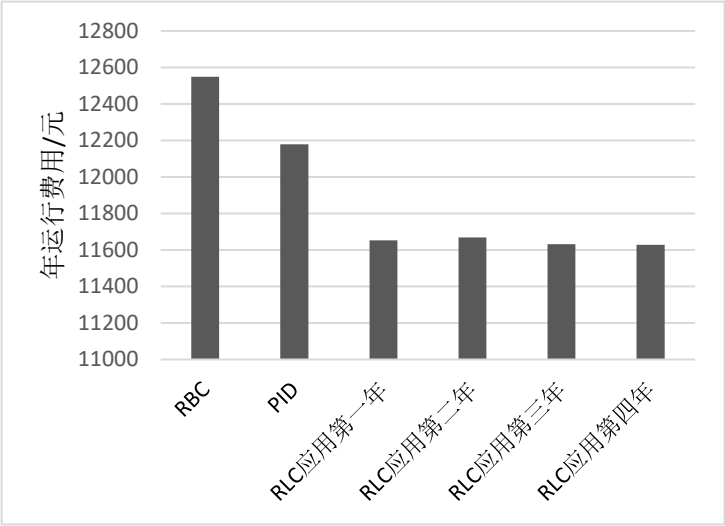

图 4.8 年运行费用对比

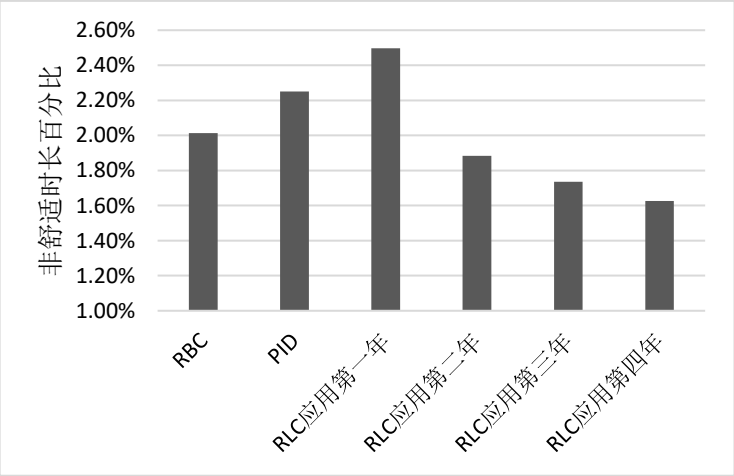

图 4.9 舒适度情况对比

从结果对比可以看出，在探索阶段，由于强化学习控制器（RLC）通过不断尝试更新策略，导致非舒适时长百分比有所增加。但在运行阶段，利用已经学习到的经验，强化学习控制器的表现均优于基准策略。与 RBC 和 PID 控制相对比，在小幅提升室内舒适度水平的同时，分别将运行费用降低超过 7%和 4.6%。并且随着运行时间的推进，强化学习控制器仍可以不断改进控制策略。最后，从能耗角度对运行策略进行比较，从图 4.10 中可以看出，应用强化学习控制器可以降低空调系统机组制冷能耗和输配能耗，在应用第四年，与 RBC 和 PID 控制相对比，可将总能耗分别降低 7.7%和 4.7%。

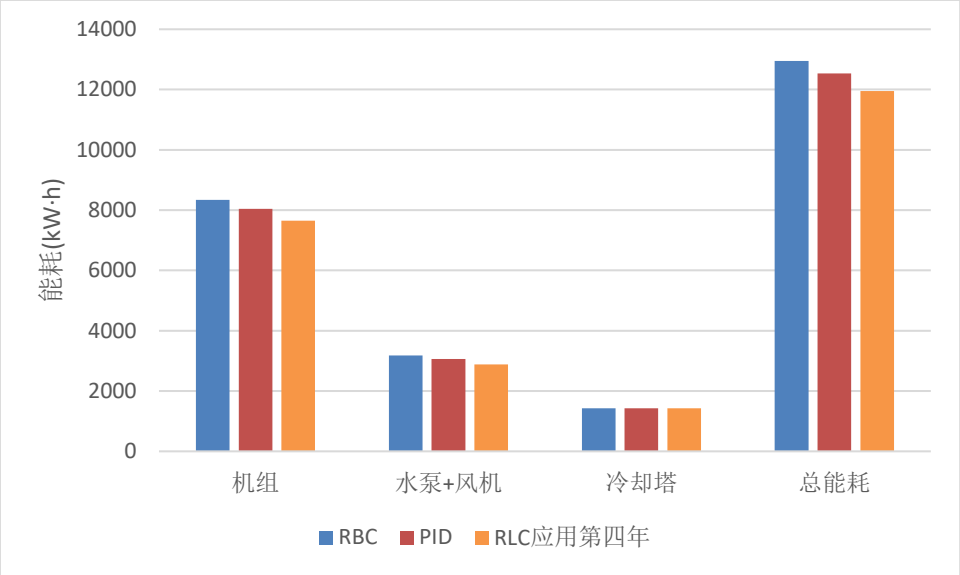

图 4.10 分项年能耗对比

4.1.5 多区域送风量控制

本小节将探讨强化学习控制器在多区域送风量控制中的表现，模型分区主要是依据房间的功能、气流分配形式和朝向来确定。由于本案例的研究对象面积较小，故不对其进行内外分区。本案例将单区域模拟中的被控区域转换为如下图 4.11 所示设定的分区形式。

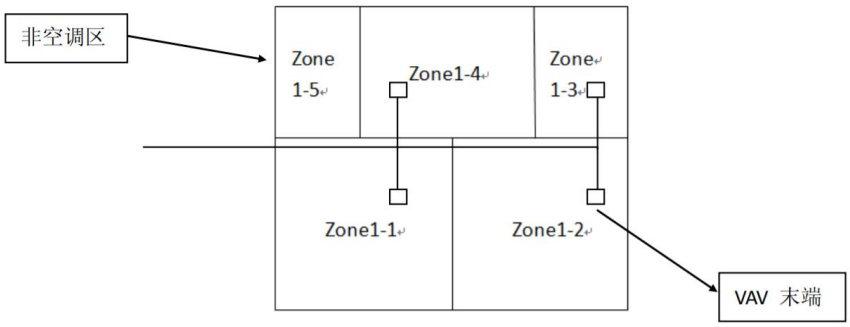

图 4.11 建筑分区情况

在应用强化学习控制器进行多区域送风量控制时，准备阶段的准备工作与 4.1.2 节中描述的大致相同，不同点在于修改了状态、动作和奖赏的定义，状态中的被控区域温度改为每个被控区域的室内空气温度，动作为每个 VAV 末端的送风量，奖赏改为系统能耗消费与每个被控区域的舒适度惩罚项的和。此外，由于状态-动作空间的扩大，因此延长了探索阶段的时长至两年，其中第一年的探索策略保持不变，第二年则采取 ϵ 值为 0.05 的 ϵ-贪心探索策略。

通过模拟仿真测试强化学习控制器的表现，并与基准策略进行对比，结果如图 4.12 所示：

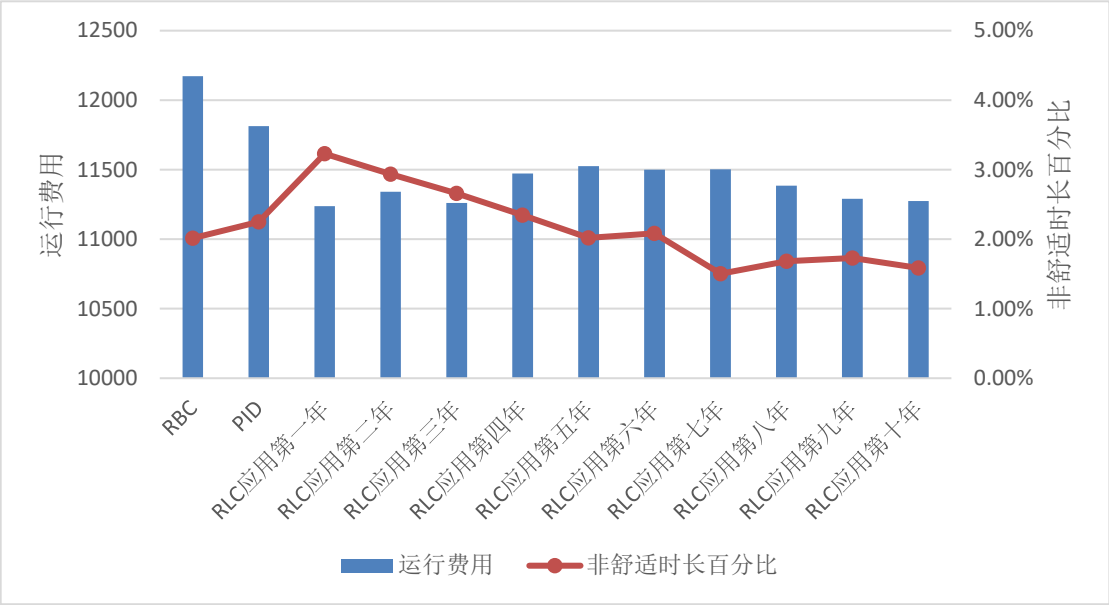

图 4.12 多区域送风量调节不同控制策略表现对比

从图中可以看出，在两个基准策略中，PID 控制器在舒适度表现差不多的情况下相比 RBC 可以将运行费用降低 3%，因此本案例将 PID 控制器作为对比对象，与 PID 控制器相比，RLC 控制器在前三年将运行费用降低 4%左右，但非舒适时长相比增加，在 RLC 控制器运行第四年至第七年间，在舒适度方面的表现与 PID 控制基本相同，运行费用降低 2.6%左右，在 RLC 控制器运行第八年至第十年间，在舒适度方面的表现优于 PID 控制，同时运行费用降低 3.6%至 4.5%之间。

4.1.6 讨论与分析

在本案例中，我们主要关注变风量空调系统的运行优化。在求解使用变风量空调系统的建筑环境最优化控制问题时，需充分考虑建筑动态热响应特性、空调系统性能特性、室内热扰以及室外天气情况等信息，从而确定每个控制步长内被控区域的送风量设定值。本案例将基于强化学习的空调系统运行策略应用在变风量空调系统中，将上述相关信息及控制效果反馈信号提供给强化学习控制器，利用控制器的探索学习能力，总结系统运行“经验”，不断改进控制策略。在本案例中，对比图 4.6 和图 4.7 可以直观地看出，在运行阶段，强化学习控制器将被控区域温度控制在舒适度区域温度较高的水平，降低系统制冷能耗以及输配能耗，从而降低运行费用。

针对单区域送风量调节，经过一年的探索学习，强化学习控制器的表现均优于基准策略。针对多区域送风量调节，由于状态-动作空间的明显增加，导致经过三年的探索学习，强化学习控制器的表现才超过基准策略。在第八年左右开始强化学习控制器的表现才较明显的优于基准策略。即使随着运行时间的推进，强化学习控制器均可以不断改进控制策略。但运行问题规模的增加带来了探索成本的上升，因此，在小规模运行问题中，强化学习控制器更具优势，更具实际应用价值。本文的接下来两个案例关注在小规模空调系统运行优化问题中的应用。

4.2 强化学习控制器在空气源热泵联合电辅热系统中的应用

4.2.1 背景介绍

本案例主要关注强化学习控制器在冬季小型住宅建筑空气源热泵联合电辅热采暖系统中的应用。

空气源热泵通过从空气中吸收热量，可以利用一份的电能生产出多份的热量，而得到越来越多的应用。在中国的中南部，对于非锅炉形式的供暖，采用热泵实现建筑采暖具有重要价值。然而在使用空气源热泵进行冬季供暖时，热泵本身会随着环境温度的降低，COP 会相应下降，导致制热能力下降，为防止极端天气出现，室外温度过低时，热泵提供制热量无法满足室内热负荷的情况，有必要使用电辅热增加系统可靠性，同时电辅热的投入使用，不仅可以确保供热能力以及化霜的时候热量衰减的补偿，还可以提高热泵产品的性价比，通过合理设计电辅热的功率来降低主机的容量，从而降低系统初投资。基于以上特点，空气源热泵联合电辅热系统被广泛应用于住宅建筑中。

与空气源热泵相比，电辅热的采暖热效率要低很多，因此在系统运行中要尽可能的使用热泵采暖，避免电辅热的开启，因此通常只有当室内温度过低时才开启电加热，用以辅助采暖，使室内温度快速回到舒适度区间。同时对于住宅建筑，由于工作的原因，人员并非一直在室内，在室内没有人员的情况下（本案例假定 8:00-18:00 为用户外出工作时间，因此在此时间段内没有舒适度条件约束。），可以关闭系统，在人员返回建筑之前开启系统将室内温度重新加热至室内舒适度区间，通过此方式运行可以降低室内热量的散失，然而上述运行方式是否节能取决于运行策略的好坏，图 4.13 展示了两种不同的运行情况，对于第一种情况，控制器可以及时将室内温度加热回舒适度区间，然而对于第二种情况，热泵重新启动的时间较晚，室内温度未及时被加热回舒适度区间，导致电加热启动，对于这

种情况，系统能耗往往要高于使用固定室温的方式运行方式。因此，美国能源部刊出文章[73]建议使用固定室温的方式运行，即使住宅内并无人员时也保持运行维持室内温度不变，然而文中也提到，在目前已有一些初创科技类企业如 Nest[74]和 BuildingIQ[75]等探索开发智能温控器，从而实现更有效的系统运行方式。

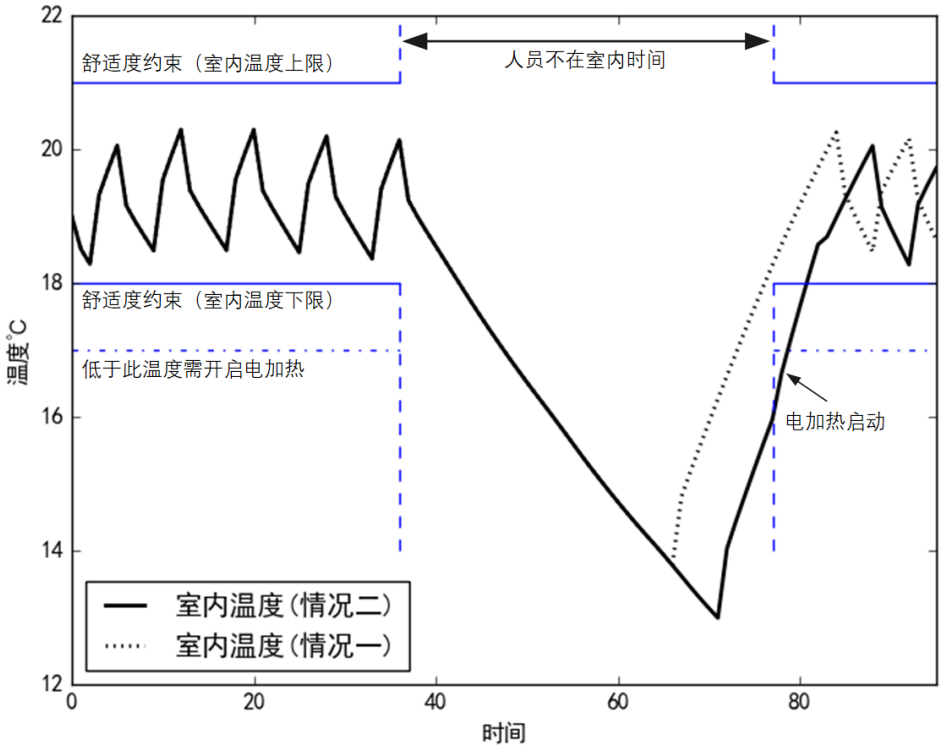

图 4.13 不同控制策略的运行情况

为实现系统高效运行，需对整个制热量序列进行优化。然而获取最优控制策略面临着两个挑战，首先空气源热泵的能效随室外温度的变化而变化，需考虑天气变化对系统能效的影响，更关键的在于，对于每个住宅建筑，建筑保温性能会有差异，同时建筑热性能难以测量，难以估计，因此难以应用基于模型的控制方法。基于以上背景，本案例主要探讨分析如何利用强化学习控制器的免模型特性，对此类建筑空调系统的制热量设定值进行序列优化。

4.2.2 准备阶段

在准备阶段，需先建立此运行优化问题的马尔科夫决策过程模型，本案例中的可控变量包括热泵制热量和电加热器制热量，在运行中，由于电加热器能效低，应避免启动，因此系统运行中应重点关注热泵制热量的设定，电加热器可设置为当室内温度低于一定阈值（本案例中设置为 17℃）的时候开启电加热器使室温

快速达到舒适度要求的范围，达到一定阈值（本案例中设置为 18℃）时则立即关闭电加热器，因此选取热泵制热量的设定值作为强化学习控制器的被控变量，制热量最小值为 0，即关闭状态，最大值取额定制热量，将动作区间等距离离散为十个水平$\{a_1, a_2, \dots, a_9, a_{10}\}$，设置控制时间步长为十五分钟。

接下来定义状态，本案例同样为建筑环境控制，因此状态量选取与案例 4.1 中介绍的相同，选取时刻信息，室外天气情况（包括当前时刻室外温度，当前室外太阳辐射和预测的未来一小时的温度变化值，其中温度预测值采用真实温度加入 $N(0, 0.2)$ 正态分布随机误差计算得出。），被控区域温度（包括室内空气温度和壁面温度，壁面温度信息采用过去四个时间步长的室内空气温度的平均值与室内空气温度的差值来近似）作为状态。确定上述变量作为状态的原因见 4.1.2 节。

最后定义奖赏函数，奖赏函数如式(4-2)所示分为两个部分，

$$r_{t-1} = -cost\left(P_{hp,t-1} \times \Delta t + P_{aux,t-1} \times \Delta t\right) + penalty(T_t)$$

$$penalty(T_t) = \begin{cases} -\beta(T_t - T_{lower\ bound})^2 + b, if\ T_t < T_{lower\ bound} \\ -\beta\left(T_t - T_{upper\ bound}\right)^2 + b, if\ T_t > T_{upper\ bound} \\ 0, if\ T_{lower\ bound} \le T_t \le T_{upper\ bound} \end{cases} \tag{4-2}$$

一部分为此控制时间步长内的能耗消费，其中能耗为空气源热泵机组与电加热器能耗之和，电价信息如表 4.2 所示，

表 4.2 分时电价信息（居民电价）

时段	电价（元/kW·h）	时间
谷电	0.307	22:00-6:00
峰电	0.617	6:00-22:00

另一部分为室内温度超出舒适度范围时的惩罚项，按照 GB50736-2012《民用建筑供暖通风与空气调节设计规范》[71]，冬季住宅建筑室内温度应保持在 18-21℃之间，故取温度上限为 21℃，下限为 18℃。

准备阶段第二步工作是设定初始阶段使用的基于规则的控制方法以及定义规则对探索空间进行缩减，在本案例中初始阶段采用类似于启停控制的方式，当被控区域温度低于 18.5℃时，保持设定热泵系统按照最大制热量运行，当被控区域温度高于 20.5℃时，则关闭热泵系统，同时，每天 8:00-14:00 系统关闭。设置初始阶段时长为系统运行前五天。在初始阶段结束后，强化学习控制器将在此基础上进行探索学习，提升空调系统运行水平。

关于探索空间缩减相关规则如下，当室内温度低于 18℃时，则下个控制步长热泵系统按照最大制热量运行，避免温度进一步降低，当室内温度高于 21℃时，则下个控制步长关闭热泵系统，降低能耗开销，同时，每天 8:00-14:00 系统关闭。上述规则用于探索阶段与运行阶段，上述规则与电加热器启停规则的应用

可使强化学习控制器避免过多不必要的探索，同时也为满足人员对室内环境的舒适度要求增加了保障，提高了系统运行的可靠性。

最后描述准备阶段最后的工作包括确定探索策略以及探索阶段时长，选择值函数逼近器，设置折扣因子和值函数逼近器更新条件。

首先探索策略选择 ϵ-贪心探索选择动作，ϵ设置为$\epsilon = \max(\epsilon_0 - d \times \Delta\epsilon, 0.1)$，其中$\epsilon_0$为 0.5，$\Delta\epsilon$为 0.02，$d$为探索阶段运行天数，通过采用上述方式可使在探索初期进行较多的探索，采集不同的状态-动作对的样本。设置系统运行第一年为探索阶段，即采用ϵ-贪心探索选择动作。探索阶段结束后即转为运行阶段。

本案例选择极端随机树作为控制器的值函数逼近器，设定极端随机树参数：树的数目为 50，分裂节点最少样本数为 3。在使用极端随机树作为值函数逼近器时，值函数逼近器的更新应使用经验集合中的所有样本。

最后设置折扣因子 γ 为 0.7，值函数逼近器更新条件为每天更新一次，此外每当室内温度违背舒适度约束条件则立刻更新。

4.2.3 模拟仿真环境

在本案例中，需要模拟建筑的热响应，以及空气源热泵在不同工况下的性能。首先描述用于模拟仿真的建筑模型，本案例的建筑模型参考美国能源部开发的 GridLAB-D[76]仿真软件中关于小型住宅建筑的仿真模型，该模型可以模拟建筑的热惰性，其数学描述如式(4-3)至(4-6)所示：

$$C_a \frac{dT_z}{dt} = H_w(T_w - T_z) + U_a(T_{out} - T_z) + Q_z \tag{4-3}$$

$$C_w \frac{dT_w}{dt} = Q_w + H_w(T_z - T_w) \tag{4-4}$$

$$Q_z = \alpha Q_i + \beta Q_s + Q_{hp} + Q_{aux} \tag{4-5}$$

$$Q_w = (1 - \alpha)Q_i + (1 - \beta)Q_s \tag{4-6}$$

式中，T_z为被控区域空气温度，C_a为空气比热容，T_w为蓄热体表面温度（反应建筑内的蓄热情况），T_{out}为室外环境温度，H_w为建筑内蓄热体与被控区域内空气间的传热系数，U_a为室外环境与被控区域间的传热系数，Q_z为被控区域的太阳辐射得热，内扰得热（包括人员、设备等）和系统加热量之和，C_w表示建筑内蓄热体（如墙体、家具等）的比热容，Q_w表示建筑内蓄热体得热来自内扰得热和太阳辐射的部分。尽管上述模型不如 EnergyPlus、DeST 等专业模拟软件中建筑模型完善，但模型计算速度快，可以表征建筑热响应物理机理，因此本案例采用此模型作为建筑仿真模型

对于空气源热泵，其加热热水所需的输入功率与输出热量和热泵的性能系数

相关，可表示为：

$$P_{hp}(t) = \frac{\dot{Q}_{hp}(t)}{COP(t)} \tag{4-7}$$

其中 COP 的值与热泵的输出热水温度、外界环境温度相关，丁强[77]等人采用经验模型计算空气源热泵 COP，其形式如下：

$$\begin{aligned} COP &= F(T_{amb}, T_{ws}) \\ &= COP_0 + \alpha_1 T_{amb} + \alpha_2 T_{ws} + \alpha_3 T_{amb}^2 + \alpha_4 T_{ws}^2 + \alpha_5 T_{amb} T_{ws} \end{aligned} \tag{4-8}$$

α_1、α_2、α_3、α_4、α_5为与系统结构有关参数，可通过空气源热泵厂家样本数据或者实际运行数据拟合得到，图 4.14 为空气源热泵供暖系统的性能拟合情况。

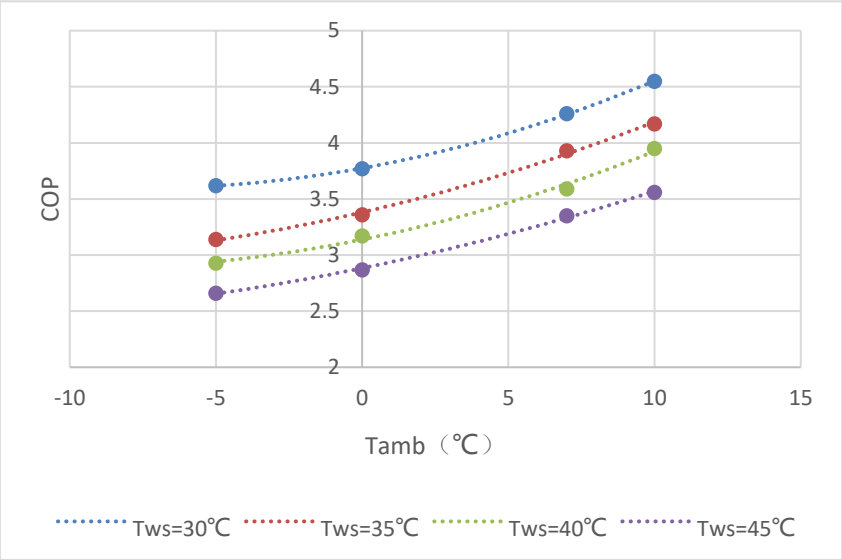

图 4.14 热泵性能拟合曲线

本案例使用 Python 语言编写住宅建筑模型、空气源热泵模型和强化学习控制器，气象数据来自上海典型气象年数据。通过数值仿真的方式验证强化学习控制器的运行效果。

4.2.4 控制效果对比

首先关注运行过程中室内环境情况。图 4.15 表示的是强化学习控制器运行第三十五天至第三十九天室内温度变化情况，图中虚线表示依次为每天的 8:00 和 18:00 时刻。从图中可以看出，首先在每天 18:00 之前，强化学习控制器会重新将室内温度提升至舒适度区间，避免电加热器的开启。其次在有人员在室内的时候可以将温度维持在稍高于 18℃的情况，满足舒适度要求的同时可降低建筑热损失，从而降低能耗。

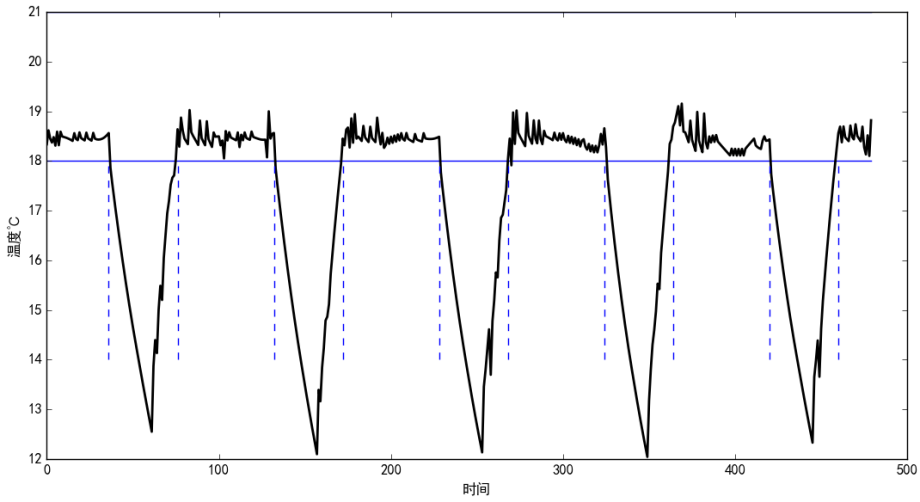

图 4.15 采用强化学习控制器时室内温度变化情况

图 4.16 为每日不舒适时长百分比变化情况，在运行初期，由于强化学习控制器缺乏运行经验，不舒适时长百分比较高，在强化学习控制器运行三十天后只有在少许极端天气下，会出现少量室内温度超过舒适度区间的情况，舒适度要求基本得以保证。

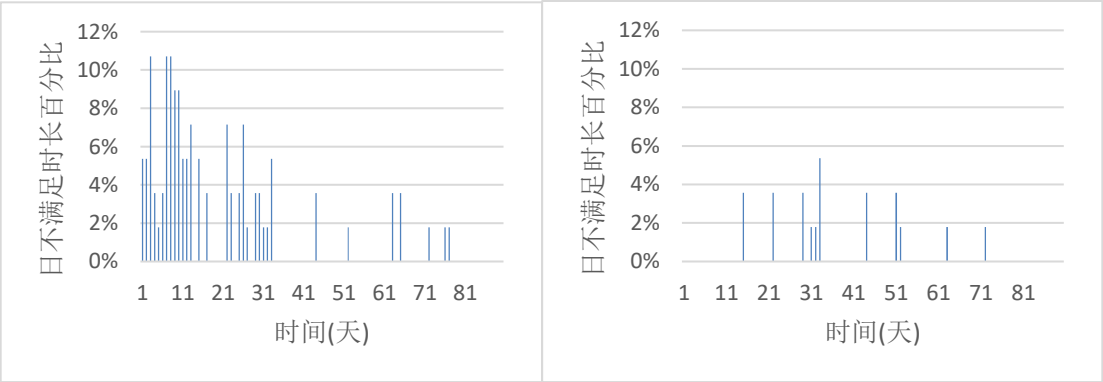

图 4.16 每日不舒适时长百分比变化情况（左图为运行第一年，右图为运行第二年）

接下来关注从运行费用角度定量评价强化学习控制器的运行效果，本案例选取两个控制策略作为对比策略，两个策略均将热泵机组的制热量作为控制变量。电加热的开启与关闭条件与强化学习控制策略中的一致，第一种控制方法为美国能源部推荐的固定室内温度运行策略，在本案例中设定室内温度为 19℃，使用 PID 控制器调整空气源热泵的制热量设定值，第二种控制方法为模型预测控制方法，其目标函数和相应约束见式(4-9)：

$$\min \int_0^{t_{end}} cost\left(P_{hp}(t) + P_{aux}(t)\right) + penalty(T(t))$$
$$\text{subject to} \quad s_{t+1} = f(s_t, a_t, w_t) \tag{4-9}$$
$$0 \le P_{hp} \le P_{max}$$

设置控制步长为 30 分钟，时间窗设置大小为三个小时，参考刘羽岱关于模型预测控制的工作[78]，对时间窗内的控制变量进行滚动优化。由于本案例中的模型预测控制方法假设房间模型与空气源热泵模型完全已知，即模型预测与实际系统运行完全吻合，不存在任何偏差，模型预测控制计算得出的控制策略可以被视为最优控制策略。

图 4.17 表示采用上述基准策略运行时部分时间内室内温度的变化情况。其中采用 PID 控制器时，室内温度保持在设定值 19℃附近小幅波动，采用模型预测控制方法负责系统运行时，由于假设模型与真实情况不存在偏差，因此几乎实现最优化控制，在人员不在室内时，关闭系统降低能耗，在每天 18:00 之前，控制器会重新将室内温度精确地提升至舒适度区间，避免电加热器的开启，当人员在室内时，室内温度始终保持在略高于 18℃的水平。

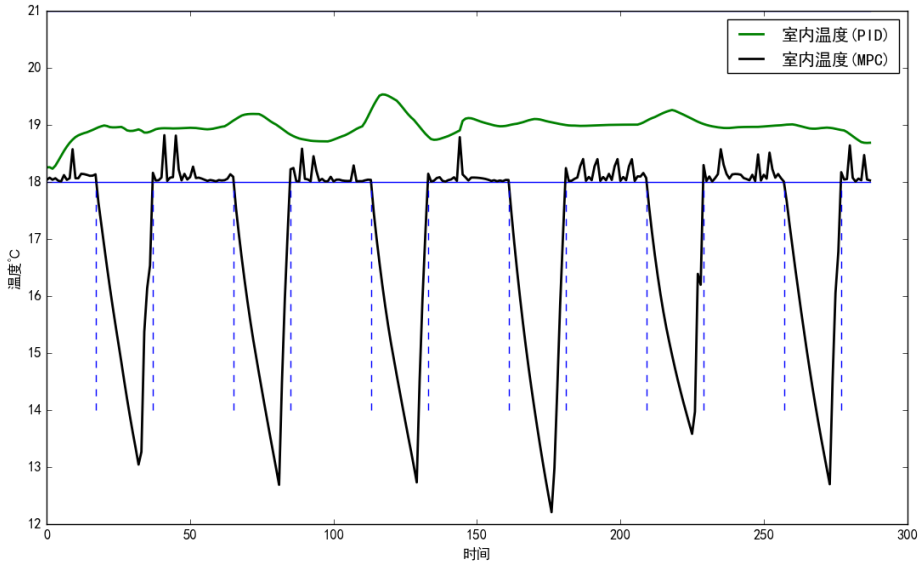

图 4.17 采用基准策略运行时室内温度变化情况

为了更好地与上述两种基准策略作对比，量化强化学习控制器的控制效果，本案例定义了 Score 指标，计算方法如公式(4-10)所示，式中 cost 代表不同运行策略下的供热季累积运行费用，Score 的值越接近 1，则意味着强化学习控制策略越接近最优控制策略。图 4.18 为 Score 指标随运行天数的变化，表 4.3 为不同运行策略下年运行费用的对比情况，可以看出在运行初期，强化学习控制器缺乏

“运行经验”，运行费用甚至高于使用固定温度控制策略的运行费用。随着强化学习控制器不断探索，“运行经验”的增加，Score 值逐步变大，截止第一个供热季结束，Score 值为 0.55，与固定温度控制策略的运行费用相比降低了 8.3%。在第二个供热季，强化学习控制器进入运行阶段，完全利用已有“运行经验”做决策，经计算，第二个供热季结束时，Score 值为 0.73，与固定温度控制策略的运行费用相比降低了 10.8%。

$$\text{Score} = \frac{cost_{RLC} - cost_{PID}}{cost_{MPC} - cost_{PID}} \tag{4-10}$$

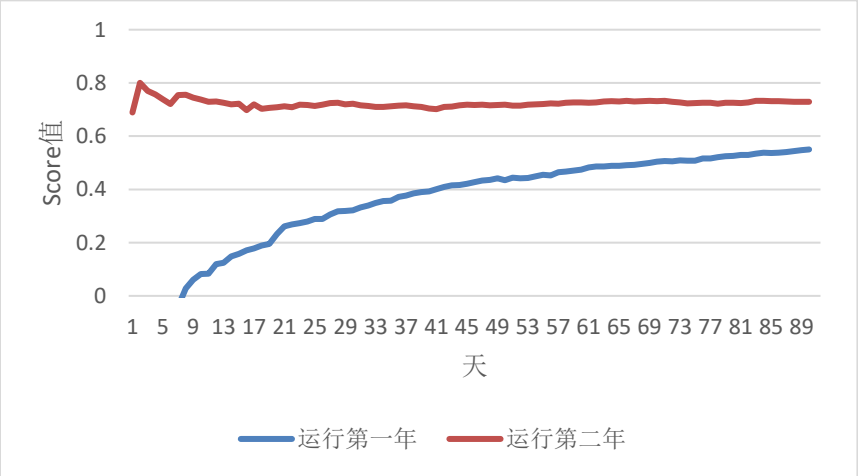

图 4.18 Score 指标随运行天数的变化

表 4.3 年运行费用对比

	运行费用（元）	相比 PID 运行费用节省(元）	相比 PID 运行费用节省百分比	相比 MPC 增加运行费用（元）	相比 MPC 运行费用增加百分比
RLC 第一年	1566.4	140.9	8.3%	115.2	7.9%
RLC 第二年	1523.7	183.6	10.8%	72.5	5.0%

接下来验证强化学习控制器在不同保温性能建筑中的自适应性，将U_a修改为 125W/℃，提高建筑的保温性能，同样通过模拟仿真计算各控制策略的控制效果，图 4.19 为修改参数后被控区域的不舒适时长百分比变化情况，其基本规律与U_a为 270W/℃的情况基本相同，在强化学习控制器运行一个月的时长后，只有在极少数情况下，会出现少量室内温度超过舒适度区间的情况，舒适度要求基本得以保证。

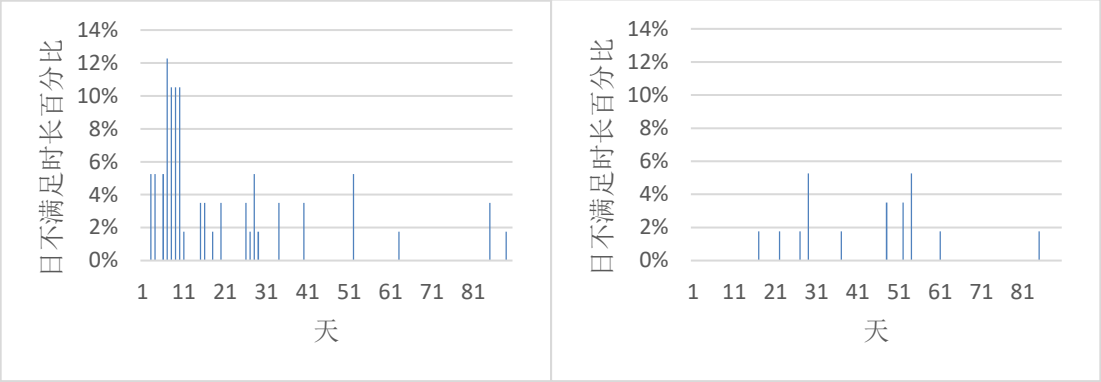

图 4.19 每日不舒适时长百分比变化情况(U_a=125W/℃情况，左图为运行第一年，右图为运行第二年)

图 4.20 为U_a=125W/℃时，Score 指标随运行天数的变化情况，表 4.4 为U_a=125W/℃时，不同运行策略下年运行费用的对比情况。截止第一个供热季结束，Score 值为 0.52，与固定温度控制策略的运行费用相比降低了 5.7%。在第二个供热季，强化学习控制器进入运行阶段，完全利用已有“运行经验”做决策，经计算，第二个供热季结束时，Score 值为 0.81，与固定温度控制策略的运行费用相比降低了 8.8%。

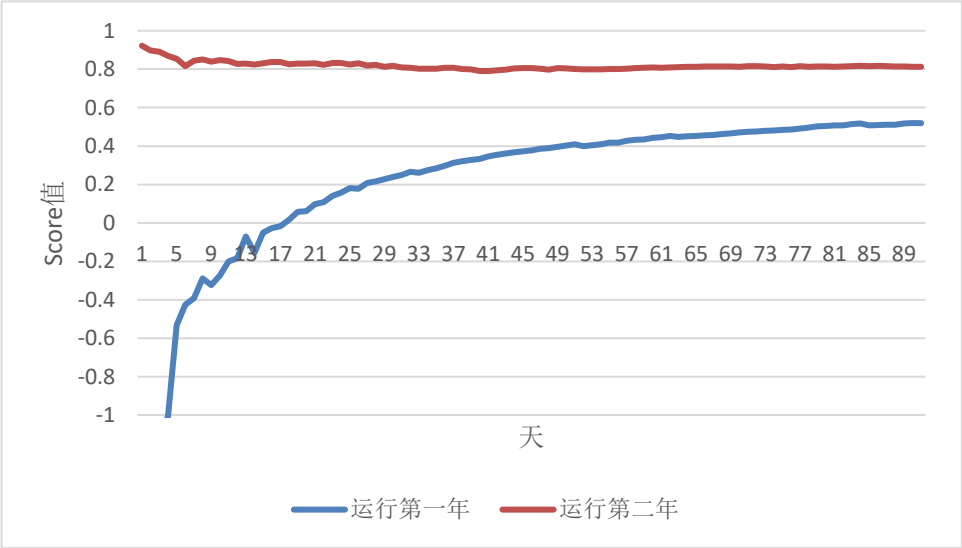

图 4.20 Score 指标随运行天数的变化(U_a=125W/℃情况)

表 4.4 年运行费用对比(U_a=125W/℃情况)

	运行费用（元）	相比 PID 运行费用节省(元）	相比 PID 运行费用节省百分比	相比 MPC 增加运行费用（元）	相比 MPC 运行费用增加百分比
RLC 第一年	745.5	45.0	5.7%	41.9	6.0%
RLC 第二年	721.2	69.3	8.8%	17.6	2.5%

4.2.5 讨论与分析

在本案例中，我们将基于强化学习的空调系统运行策略应用在小型住宅建筑空气源热泵联合电辅热系统中，在运行过程中，将时刻信息、室外天气情况和室内环境状况信息及控制效果反馈信号提供给强化学习控制器，利用控制器的探索学习能力，不断提高控制器的控制效果。通过模拟仿真的方式测试其在不同保温性能建筑的运行效果。与固定室内温度运行策略相比，在运行阶段，当人员在室内时，强化学习控制器通过调整空气源热泵制热量的设定值将被控区域温度控制在舒适度区域内室内温度相对较低的水平，当人员不在室内时，强化学习控制器会关闭系统并在人员回到室内前重新启动系统将室内温度加热回舒适度要求区间，在以上两种情况下降低了建筑热损失与空调系统采暖能耗，从而降低运行费用。

从舒适度角度来看，在强化学习控制器运行初期，由于控制器缺乏运行经验需进行较多探索，违背舒适度约束情况发生率较高，然而基于规则的电辅热运行方式可以有效减少不舒适时长百分比，增强了系统的可靠性，在运行三十天后，只有在少许极端天气下，会出现少量室内温度超过舒适度区间的情况，绝大多数时间均可以满足室内舒适度要求。接下来对比分析系统运行费用，首先与固定室内温度运行策略相比，基于强化学习的空调系统运行策略可以将运行费用降低5.7%到 10.8%。与最优控制策略相比控制效果存在一定差距，但是最优控制策略需要完全已知环境模型，在实际应用中模型仿真与真实情况间的偏差很难避免，同时对于小型住宅建筑很少有工程人员对建筑和空调系统进行建模用来精确地模拟仿真，然而本文提出的基于强化学习算法的系统运行策略无需建立精确地仿真模型，通过定义规则和不断探索获取接近最优的控制策略，牺牲了部分性能换取了更高的应用性。

在本案例的应用场景中，强化学习控制器可以主动学习建筑热特性和空气源热泵系统的特性，对空气源热泵制热量设定值进行序列优化，从而降低全天累积运行费用最低。与模型预测控制相比，一方面降低了对先验知识的依赖，具有较强的自适应性，另一方面强化学习控制器无需滚动优化，仅需计算各状态-动作对的 Q 值，因此响应速度更快。对于不同的室内外环境条件，强化学习控制器可以调整相应的动作，具有较强的自调整性。此外，随着运行时间的推进，控制器可以充分利用建筑运行产生的数据，不断的强化自身能力，其数据驱动的方式具备较高的应用价值，易于在小型住宅建筑中应用与推广。

接下来的案例主要关注对于空调系统侧较复杂且难与建模的情况下，如何利用强化学习控制器应对负荷侧和室外环境不断变化的情况，提升建筑整体能效，

实现空调系统精细化运行。

4.3 强化学习控制器在严寒地区太阳能-地源热泵系统中的应用

4.3.1 背景介绍

本案例主要探讨分析强化学习控制器在严寒地区太阳能-地源热泵系统中的应用，地源热泵是以地源能作为冬季采暖供热的低温热源和夏季制冷的冷源的热泵系统，由于地源热泵具有全年土壤温度波动小、不需要除霜、土壤有较好的蓄能作用，不会对大气环境造成热污染及噪声污染的特点，符合可持续发展战略，因此该项技术目前在国际上正日益受到人们的关注。但对于严寒地区，热负荷远大于冷负荷，单独使用土壤源热泵系统时，系统连续运行会导致土壤温度持续降低，热泵系统效率下降，同时也会存在无法满足建筑负荷的风险。针对冷热负荷差别比较大的情况，在技术经济分析合理的前提下，可采用复合式地源热泵系统，对土壤进行热量补偿。太阳能-地源热泵系统即为一种复合式地源热泵系统。太阳能土壤源热泵的形式有多种选择。本案例选取高翠[79]提出的应用在哈尔滨地区的太阳能-土壤源热泵系统作为研究对象，该系统可以同时提供采暖与生活热水所需能量，其系统图如图 4.21 所示。

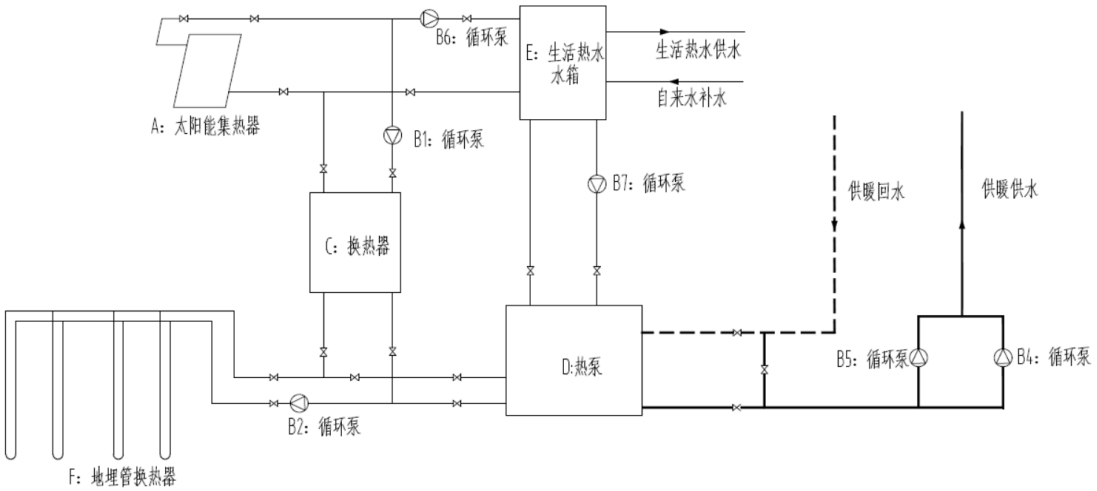

图 4.21 太阳能-土壤源热泵系统图

该系统运行模式分为以下二种：在夏季和过渡季使用太阳能进行蓄热、加热生活热水；在冬季使用太阳能预热生活热水，启动土壤源热泵用于建筑采暖。在夏季和过渡季时将太阳能集热器收集到的一部分太阳能通过地埋管换热器储存到土壤中，一部分太阳能用来加热生活热水，当太阳能强度较低时，生活热水负

荷难以得到满足，此时开启辅助电加热器加热生活热水。冬季运行时，自来水补水的温度比较低且冬季的太阳辐射强度低，因此将收集到的全部太阳能用来预热生活热水，使用土壤源热泵供热。此外，在冬季太阳能集热器内的水温不高，热水供应会加大辅助加热量的消耗，可以考虑用一部分采暖热水进一步预热生活用水，再使用辅助热源加热。系统的太阳能与地热能的能量流向图如图 4.22 所示。

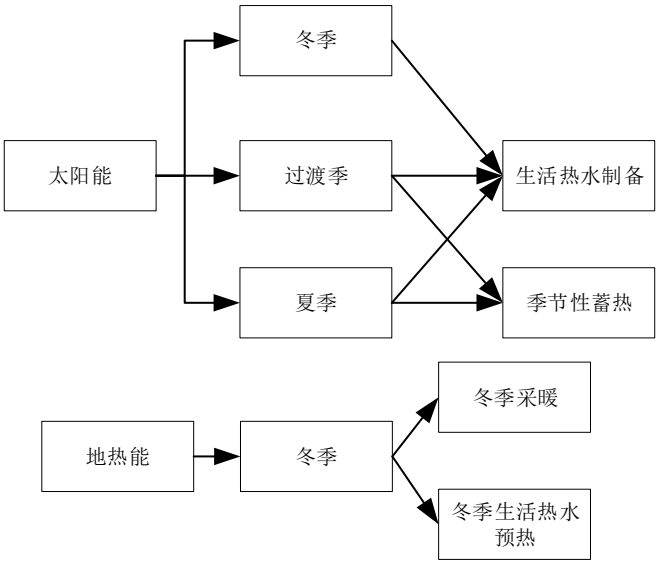

图 4.22 系统可再生能源能量流向图

对于复合式地源热泵系统，系统运行策略一直是研究热点，针对太阳能-土壤源热泵系统，首先系统能效受气候影响很大，太阳辐射具有不连续性和不稳定性的特点，除去太阳辐射强度，太阳能集热器效率还受环境温度，入水温度，循环水流量等多种因素影响，其中集热效率与循环水流量间的关系如图 4.23 所示，当流量超过一定水平时，集热器循环水流量的增加对效率提高的影响逐步降低，然而循环泵的能耗与水流量成三次方关系，因此有必要对动态调整太阳能集热器循环泵的流量设定值使净集热量（太阳能集热量与循环泵能耗的差值）最大化。此外，而地埋管地源热泵的地源侧管路复杂多变且管程较长，地源水在流动过程中所产生总阻力较大，循环水泵的功率也相对较大，一方面，在冬季供暖期部分负荷下降低地源侧水流量可以显著降低水泵能耗，然而热泵机组 COP 随地源侧水流量的降低而降低，因此需将地埋管源侧水泵与热泵机组整体考虑，研究地源水的变流量运行策略从而最大化系统供热能效，另一方面在夏季与过渡季进行季节性蓄热时，地源侧水流量的大小也会影响土壤蓄热情况与循环泵能耗，因此也需对地源侧水流量进行动态调整，使季节性蓄热的净蓄热量最大化。

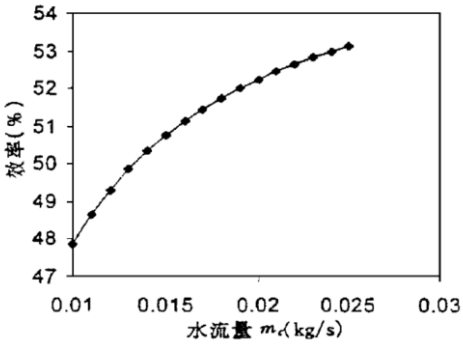

图 4.23 太阳能集热器集热效率与循环水流量间的关系[80]

基于以上背景，本案例探讨通过强化学习控制器负责 B1，B2，B6 水泵的运行优化。旨在实现在不显示地定义太阳能集热器、地埋管换热器、热泵机组，水泵等模型的前提下，通过数据驱动的方式提高系统运行效率，实现寒冷地区太阳能-地源热泵系统的精细化运行。

4.3.2 准备阶段

为实现 B1，B2，B6 水泵的运行优化，本案例决定设定三个强化学习控制器实现系统运行优化，分别为生活热水制备 RLC、季节性蓄热 RLC 和冬季采暖 RLC。接下来分别介绍关于这三个强化学习控制器准备阶段的工作。

首先建立空调系统运行问题的马尔科夫决策过程，对于生活热水制备，影响太阳能集热器运行阶段集热效率的因素主要包括太阳辐射强度、集热器入水温度、室外环境温度和集热器循环水流量。其中循环水流量可通过调整循环泵 B6 频率改变，因此将循环泵 B6 作为被控对象，设置其循环水流量为动作，设置当前时刻太阳辐射强度、室外环境温度和集热器入口温度为状态，奖赏设定为当前控制步长的集热功率与循环泵功率的差值，其中集热功率采用太阳能集热器进出口温差与流量和比热容的乘积计算。

对于过渡季与夏季阶段的季节性蓄热，影响蓄热效果的可测量变量主要包括太阳辐射强度、室外环境温度、集热器侧循环水流量和地埋管侧循环水流量，其中太阳辐射强度、室外环境温度、集热器侧循环水流量影响太阳能集热器的集热效率，集热器侧循环水流量和地埋管侧循环水流量影响换热器 C 的换热效率，地埋管侧循环水流量影响地埋管与土壤的换热情况。在上述影响因素中，集热器侧循环水流量和地埋管侧循环水流量可分别通过调整循环泵 B1 与 B2 频率改变，因此将 B1 与 B2 循环泵作为被控对象，设置其循环水流量为动作，设置当前时刻太阳辐射强度、室外环境温度为状态，奖赏设定为控制步长内的净蓄热功率，

即蓄热功率与 B1、B2 两个循环泵功率的差值，其中蓄热功率可由地埋管进出口温差与流量和比热容的乘积计算得出。

对于冬季使用地源热泵进行采暖时，为了提高系统效率，可将地埋管水泵变频与热泵机组压缩机变频联合起来，从而达到满足建筑热负荷的同时降低能耗的目标。目前已有方法如杨前明[81]等人提出的 VFC 与 EEV 的联合控制方法可以根据设定热负荷与热泵机组的当前制热量的偏差值调整热泵机组的压缩机频率，因此，本案例中的强化学习控制器的目标是学习在不同的负荷状态下如何调整地埋管源侧流量从而实现系统的最优运行。因此可设定状态为负荷率（热负荷/热泵机组额定制热量），动作为地埋管源侧流量即 B2 循环泵流量，奖赏为系统功率（B2 循环泵功率与热泵机组功率的和）与发生制热量未满足负荷情况时的惩罚项的和。

接下来定义规则缩减探索空间。首先确定各循环泵运行时合理的流量区间，当管内流量减小时，管内流体流动速度的变化会影响流体的流态，为了保证良好的换热情况，管内流体流态应保持为湍流。此外，在水泵变速过程中，水泵综合效率（即水泵效率与变频器效率的综合）是下降的。随着水泵转速的降低，变频器的效率随之降低，尤其在转速小于其额定转速的 60%时，变频器效率显著下降，导致水泵的综合效率明显下降，当转速降到一定程度，系统不再节能。因此为避免流量过小引起综合效率过低以及出现层流的情况，需要针对变频水泵设定最小流量比[82]。本案例中，根据经验设定的各水泵最小流量比设为 0.5。因此设定 B6 与 B1 循环泵的流量区间为 0.6kg/s 到 1.2kg/s，B2 循环泵的流量区间为 0.8kg/s 到 1.6kg/s。应用强化学习控制器时，需要对动作空间进行离散，所以对于过渡季与夏季阶段的生活热水制备将动作空间离散为{0kg/s, 0.6kg/s, 0.8kg/s, 1.0kg/s, 1.2kg/s}，同时当太阳辐射强度低于 250W/m^2 时，则关闭循环泵，即设定动作为 0kg/s，对于季节性蓄热需同时控制 B1 与 B2 循环泵流量，此外蓄热过程中地埋管的流量无需过大，因此将动作空间离散为{(0kg/s, 0kg/s), (0.6kg/s, 0.8kg/s), (0.6kg/s, 1.0kg/s), (0.6kg/s, 1.2kg/s), (0.8kg/s, 0.8kg/s), (0.8kg/s, 1.0kg/s), (0.8kg/s, 1.2kg/s), (1.0kg/s, 0.8kg/s), (1.0kg/s, 1.0kg/s), (1.0kg/s, 1.2kg/s), (1.2kg/s, 0.8kg/s), (1.2kg/s, 1.0kg/s), (1.2kg/s, 1.2kg/s)}，其中每个元组中前者为 B1 循环泵流量，后者为 B2 循环泵流量。同样当太阳辐射强度低于 250W/m^2 时，则关闭循环泵，即设定动作为(0kg/s, 0kg/s)。对于热泵机组冬季采暖，设定 B2 循环泵的动作空间为{0.8kg/s, 0.9kg/s, 1.0kg/s, 1.1kg/s, 1.2kg/s, 1.3kg/s, 1.4kg/s, 1.5kg/s, 1.6kg/s}。此外，为防止当热负荷较高时，地埋管侧水流量较低导致系统供热量无法满足情况的发生，相应的增加两条规则，当负荷率大于 70%时，动作空间缩减为{1.4kg/s, 1.5kg/s, 1.6kg/s}，当负荷率大于 90%时，B2 循环泵的流量设定为最大流量运行，

即 1.6kg/s。使用上述规则剔除了明显不合理的状态-动作组合，缩减了状态-动作空间，使强化学习控制器在需要探索的区域进行探索学习，一方面降低了获取近优控制策略所需时间，另一方面，提高了系统运行的可靠性。

设定初始阶段使用的以基于规则的控制方法，本案例的初始运行规则采用简单的固定值运行策略，对于生活热水制备，设定 B6 循环泵流量为 1.0kg/s，对于季节性蓄热设定 B1 与 B2 循环泵流量均为 1.0kg/s，对于冬季热泵供暖，设定流量值为 1.4kg/s，在系统运行前五天，按照上述固定值运行策略运行。上述设定值均为根据经验设定，设定的值往往无法保证是最优的，强化学习控制器将在此基础上探索改进运行策略的可能。

最后确定探索策略和强化学习算法相关参数，探索策略选择与案例 4.1 中相同的 ϵ-贪心探索策略，ϵ设置为$\epsilon = \max(\epsilon_0 - d \times \Delta\epsilon,\ 0.1)$，其中$\epsilon_0$为 0.5，$\Delta\epsilon$为 0.02，$d$为探索阶段运行天数，设置第一年运行为探索阶段，即采用ϵ-贪心探索选择动作。从第二年起空调系统即转为运行阶段。

本案例中，由于三个强化学习控制器的目标均只考虑将当前控制步长的奖赏最大化，因此设置折扣因子 γ 均为 0。另外，本案例选择极端随机树作为三个控制器的值函数逼近器，设定极端随机树参数：树的数目为 50，分裂节点最少样本数为 3。

在使用极端随机树作为值函数逼近器时，值函数逼近器的更新应使用经验集合中的所有样本。本案例中三个强化学习控制器的值函数更新条件为每天更新一次，此外在运行过程中，当强化学习控制器出现明显不合理的行为时，需要立即更新值函数逼近器，从而尽量避免在相似工况下出现类似不合理的行为，针对本案例来说，当生活热水制备时净集热量小于 0 时，则立刻更新生活热水制备 RLC 的值函数逼近器，当季节性蓄热时净蓄热量小于 0 时，则立刻更新季节性蓄热 RLC 的值函数逼近器，当冬季供暖制热量小于建筑热负荷时，则立刻更新冬季采暖 RLC 的值函数逼近器。

4.3.3 模拟仿真环境

本案例使用模拟仿真的方式对上述基于强化学习算法的空调系统运行策略进行验证，其中仿真模型在 TRNSYS 中建立，TRNSYS 中各部件模型及模型参数均按照作者在其文章[79]中的描述选取和设置，因此本文不再赘述，但文章作者选取的 Type1b 平板型太阳能集热器只能模拟定流量的工况，Type668 水-水热泵也无法准确模拟地埋管侧变流量工况，因此需要对上述两个模型进行修改。

本案例仿真模型中的太阳能集热器参考 Duffie 与 Beckman[83]提出的模拟方

法，集热器的有效得热 Q_u 按下式(4-11)计算：

$$Q_u = A_c \times F_R[(\alpha\tau) \times I_t - U_L(T_{ci} - T_a)] = mc_p(T_{co} - T_{ci}) \tag{4-11}$$

式中A_c为集热器面积，$\alpha\tau$表示太阳光辐射的吸收率，I_t为太阳辐射强度，U_L表示集热器总热损系数(collector overall heat loss coefficient)。T_{ci}为集热器入口温度，T_a表示室外环境温度，T_{co} 为集热器出口温度，F_R为集热器热转移因子(collector heat removal factor)，其值的大小与集热器面积，集热器总热损失系数和质量流量相关，其计算公式见式(4-12)：

$$F_R = \frac{m \times c_p}{A_c \times U_L}[1 - \exp\left(-\frac{A_c \times U_L \times F'}{m \times c_p}\right)] \tag{4-12}$$

式中 F' 为集热器效率因子(collector efficiency factor)，F'和U_L参数由 Reindl[84]和 Bujedo[85]提供。

本案例仿真模型中的水水热泵应能模拟在源侧变流量情况下 COP 的变化，针对地埋管地源热泵变流量系统，杨少刚[86]提出了多元多项式回归模型，可准确模拟热泵性能，在多元多项式回归模型中，热泵机组 *COP* 与热泵机组当前的制热量 CAP、冷凝器侧流量 M_c、地源侧流量 M_e、蒸发器侧回水温度 T_{ei} 和冷凝器侧出水温度 T_{co}，模型形式如式(4-13)所示：

$$\begin{aligned} COP=&\beta_0+\beta_1 CAP+\beta_2 CAP^2+\beta_3 M_e+\beta_4 M_e^2+\beta_5 M_c+\beta_6 M_c^2+ \\ &\beta_7 T_{ei}+\beta_8 T_{ei}^2+\beta_9 T_{co}+\beta_{10} T_{co}^2+\beta_{11} CAP\cdot M_e+\beta_{12} CAP\cdot M_c+ \\ &\beta_{13} CAP\cdot T_{co}+\beta_{14} CAP\cdot T_{ei}+\beta_{15} M_c\cdot M_e+\beta_{16} M_e\cdot T_{ei}+\beta_{17} M_e\cdot T_{co} \\ &+\beta_{18} M_c\cdot T_{ei}+\beta_{19} M_c\cdot T_{co}+\beta_{20} T_{ei}\cdot T_{co} \end{aligned} \tag{4-13}$$

采用机组样本数据或者运行数据做多元非线性回归，可得到各拟合参数。本案例使用的样本数据来自于型号为 EXLSR250.1 的热泵机组提供的样本文件。

本案例使用 MATLAB 语言编写上述两个模型，通过 TRNSYS 组件 Type155 将编写好的模型接入到 TRNSYS 中，使用 Python 语言编写强化学习控制器，通过 UDP 协议，用 Internet socket 的方法实现联合仿真。搭建好的太阳能-地源热泵系统及其强化学习控制器的 TRNSYS 仿真模型见图 4.24。模型中设置土壤的密度为 1975kg/m^3，导热系数为 2.44W/m·℃，比热为 2120J/kg·℃，设置地埋管换热器为 12 根 80m 深的地埋管与土壤进行换热，设置太阳能集热器的面积为 70 m^2。关于模型仿真部分不是本案例的研究重点，更多关于模型中各部件的参数设定情况可参考文章[79]的第三章。

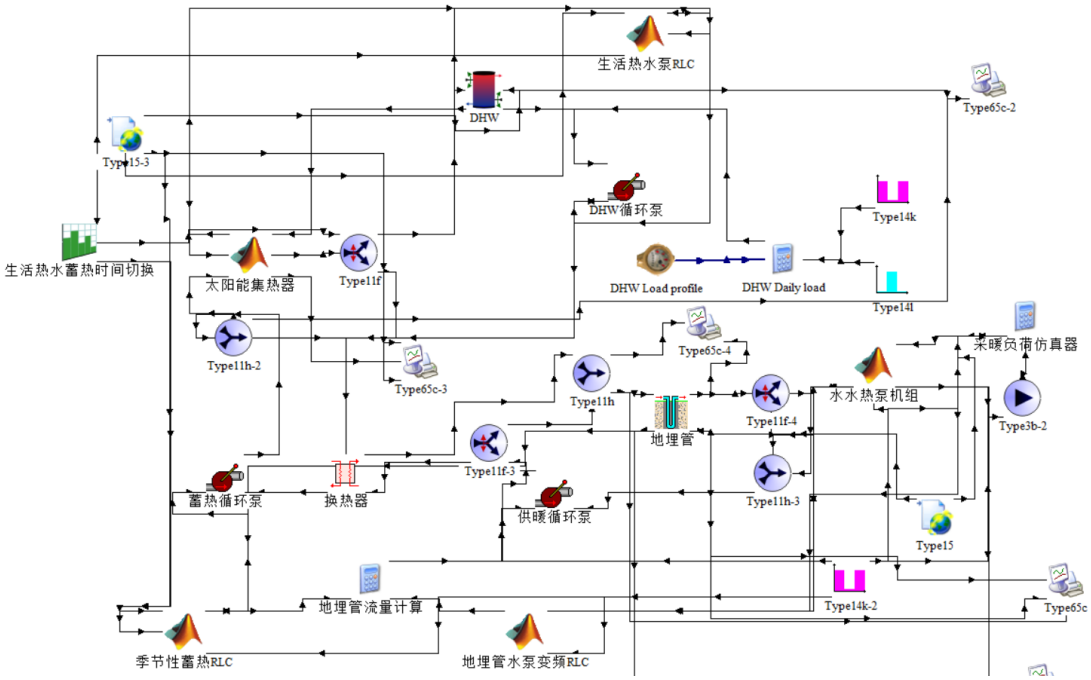

图 4.24 太阳能-地源热泵系统的 TRNSYS 仿真模型

4.3.4 控制效果对比

为了定量分析本案例设计的强化学习控制器的运行效果，首先需设定基准控制策略作为对比。

目前针对强迫循环式太阳能集热器的循环泵控制通常采用温差控制法，测量太阳能集热器的出口温度 T_{so}与水箱出口温度或集热器入口温度T_{si}，计算二者的差值$\Delta T_s(\Delta T_s = T_{so} - T_{si})$，当 ΔT_s的值大于ΔT_{on}则启动循环泵，直至ΔT_s的值小于ΔT_{off}时关闭循环泵。相关文献[84]建议设置ΔT_{on}为 10℃，设置ΔT_{off}为 1.5℃。循环泵通常采用定流量的方式运行，Persson[87]等人提供了确定流量的经验公式，公式如式(4-14)所示：

$$m_c = (0.14\frac{A_c}{10}) \tag{4-14}$$

据此可以算出本案例的集热器循环流量 m_c 应为 0.98kg/s。本案例中，针对生活热水制备与季节性采暖，B1，B2 与 B6 均按照上述方式启停，且流量均设置为 0.98kg/s。

对于冬季热泵机组供暖，目前大多数实际应用是采用定频即固定流量运行的方式，同时也有关于变流量运行的研究和应用，因此本案例设定了如表 4.5 所示的两个基准策略。

表 4.5 冬季热泵机组采暖工况下 B2 循环泵运行基准策略

1. 定流量	循环流量保持 1.6kg/s 运行
2. 变流量	当负荷率大于 75%： 循环流量设为 1.6kg/s 当负荷率大于 50%且小于等于 75%时： 循环流量设为 1.4kg/s 当负荷率大于 25%且小于等于 50%时： 循环流量设为 1.2kg/s 当负荷率小于等于 25%时： 循环流量设为 1.0kg/s

通过模拟仿真的方式模拟了太阳能-地源热泵系统在使用上述基准策略和强化学习控制器(RLC)的表现，首先对比生活热水净集热量，图 4.25 为生活热水制备净集热量的对比，

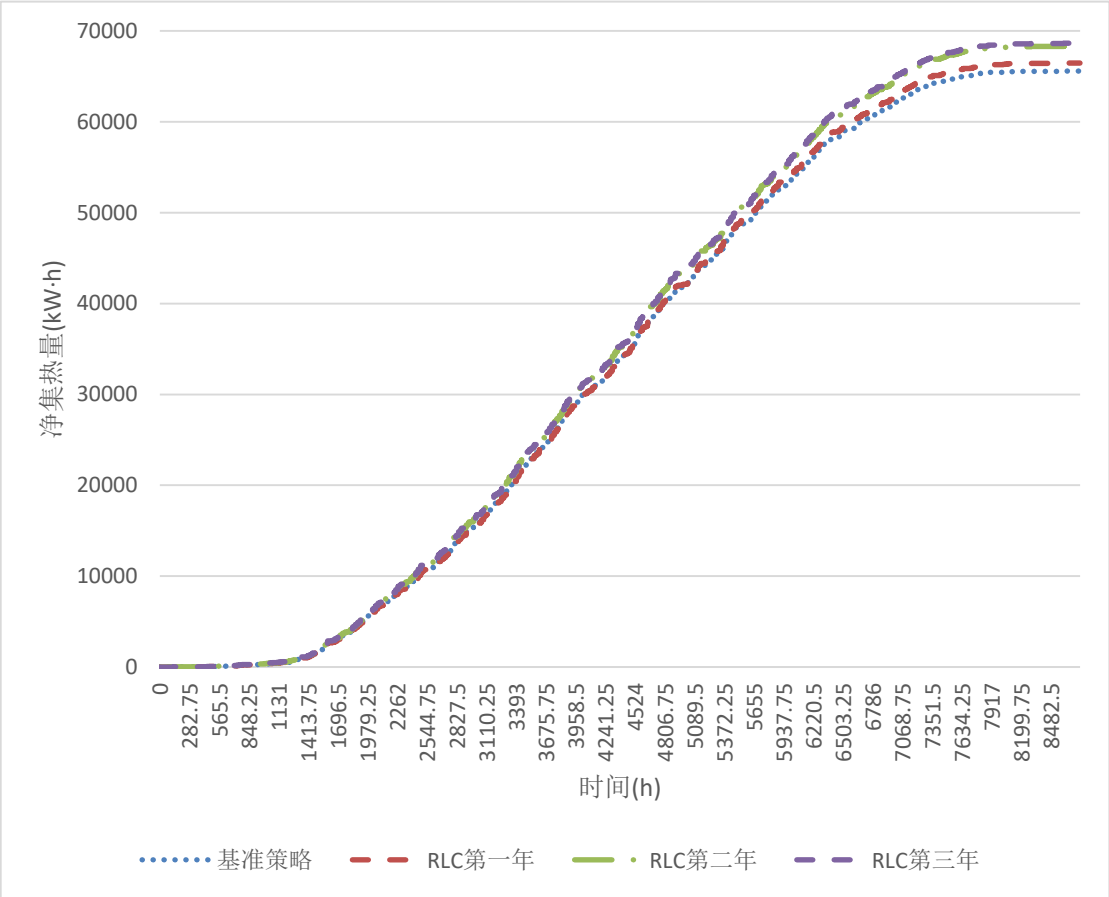

图 4.25 生活热水净集热量对比

在强化学习控制器使用初期，由于强化学习控制器需要不断探索，导致其表现并不出色，在第一年探索阶段结束后，使用强化学习控制器的运行方式年累积净集

热量的提升仅为 1.4%，从第二年开始强化学习控制器转为运行阶段，意味着强化学习控制器转为充分利用已经学习到的“经验”，在第二年与第三年可分别将年累积净集热量提升 4.2%和 4.7%。表 4.6 为系统能量对比情况，可以看出使用强化学习控制器可以降低循环泵能耗的同时提高生活热水集热量，在系统运行第三年，可将循环泵能耗降低 39%同时生活热水集热量提高 3.9%。

表 4.6 生活热水制备运行效果对比

	集热器集热量(kW·h)	水泵能耗(kW·h)	净集热量(kW·h)	相比基准策略净集热量增加(kW·h)	相比基准策略净集热量增加百分比
基准策略	66859	1267	65592		
RLC 第一年	67394	914	66480	888	1.4%
RLC 第二年	69150	811	68339	2747	4.2%
RLC 第三年	69435	777	68658	3066	4.7%

接下来对季节性蓄热效果进行对比，图 4.26 为土壤温度变化情况，可以定性地看出使用强化学习控制器蓄热量更高。

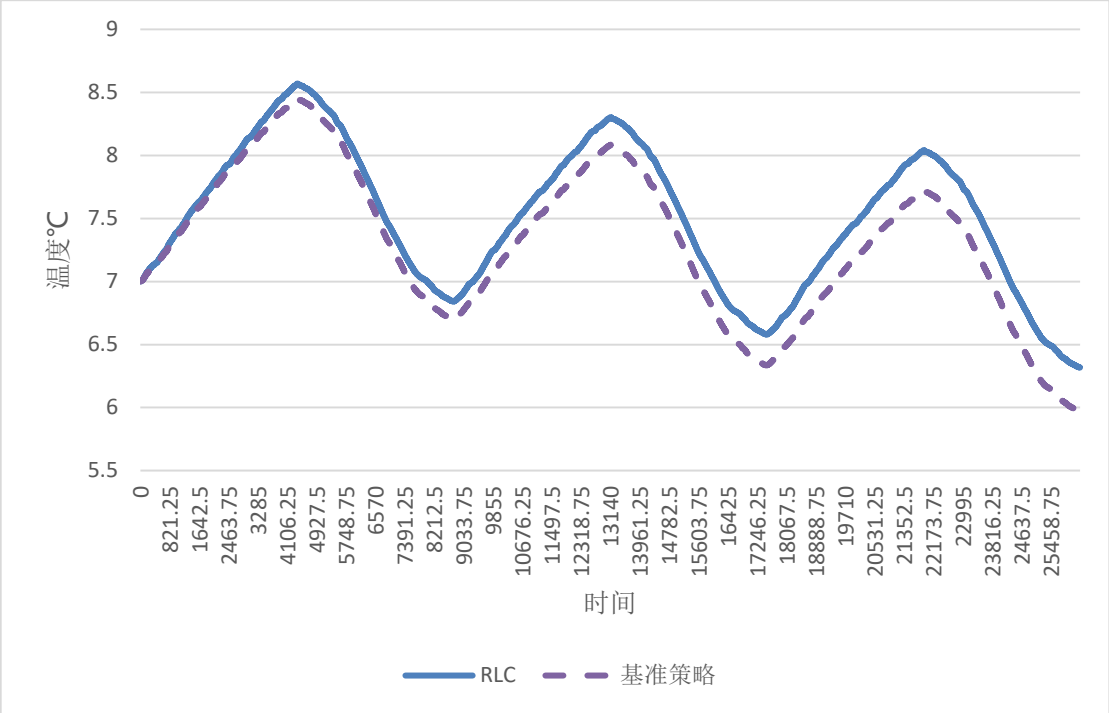

图 4.26 土壤温度变化对比

表 4.7 为季节性蓄热效果的分项统计，在强化学习控制器运行的前三年中，水泵能耗相对基准策略分别降低了 25%、39%和 40%，系统年净蓄热量提高了 1%、4.1%和 4.4%，可以看出使用强化学习控制器的季节性蓄热效率更高，效果

更好。

表 4.7 季节性蓄热集热量对比

	土壤蓄热量(kW·h)	水泵能耗(kW·h)	净蓄热量(kW·h)	相比基准策略净蓄热量增加(kW·h)	相比基准策略净集蓄量增加百分比
基准策略	53890	2720	51170		
RLC 第一年	55081	3415	51666	496	1.0%
RLC 第二年	57015	3768	53248	2078	4.1%
RLC 第三年	57230	3803	53428	2258	4.4%

最后对比冬季土壤源热泵系统运行效果，表 4.8 为冬季采暖系统各控制策略分项能耗统计情况，首先从表中可以看出，使用变流量控制方式虽然热泵机组能耗升高，但由于水泵能耗降低更为显著，因此总能耗会降低。在运行前三年，与定流量基准策略对比，强化学习控制器可将系统总能耗分别降低 7.6%、8.5%和 8.6%，与变流量基准策略对比，强化学习控制器可将系统总能耗分别降低 4.4%、5.3%和 5.4%。

表 4.8 冬季土壤源热泵系统能耗对比

	热泵能耗(kW·h)	水泵能耗(kW·h)	总能耗(kW·h)	与定流量基准策略相比能耗降低(kW·h)	与变流量基准策略相比能耗降低(kW·h)
定流量基准策略	20778	9880	30658		
变流量基准策略	23682	5938	29620		
RLC 第一年	25078	3238	28316	2342	1304
RLC 第二年	25237	2801	28038	2620	1582
RLC 第三年	25245	2780	28025	2633	1595

4.3.5 讨论与分析

本案例将强化学习控制器应用于严寒地区太阳能-地源热泵系统中，并使用 TRNSYS 软件进行了模拟仿真，结果显示，通过强化学习控制器不断的探索学习，在运行阶段与基准策略相比，生活热水制备 RLC、季节性蓄热 RLC 可以综合考虑太阳能集热效率与循环泵能耗间的关系，针对不断变化的工况对循环流量设定值进行动态调整，能够有效提高系统净集热量与净蓄热量，冬季采暖 RLC 可以综合考虑地埋管侧循环泵能耗与热泵主机 COP 间的关系，针对不断变化的工况

对地埋管侧循环流量设定值进行动态调整，能够有效降低热泵机组与地埋管侧循环泵的总能耗。

太阳能-地源热泵系统是属于具有以下特点的一类典型空调系统，即系统内部分部件（如本案例中的太阳能集热器与地埋管换热器）由于机理较为复杂或者热物性参数很难准确获取等原因，难以建立精确且简洁的数学模型，因此此类系统也很难应用基于模型的系统控制策略。针对此类系统，目前通常采用基于规则的控制方式运行，然而使用基于规则的控制方式时，需要确定控制器参数的具体值，通常工程人员仅能估计一个粗糙的范围（例如本例中不同热负荷情况下，冬季地埋管循环流量的确定），因此设定的值往往无法保证是最优的，本文提出的基于强化学习算法的空调系统运行方法无需对空调系统建模，可利用工程人员估计得到合理范围对探索空间削减，通过不断探索积累“运行经验”，因此非常适合具备上述特点的空调系统。

此外，对于空调系统，在不同运行时段，不同环路或子系统往往具有独立的运行优化目标，因此可将大系统拆解成若干子系统（如本案例中分别定义了生活热水制备 RLC、季节性蓄热 RLC 和冬季采暖 RLC），一方面可以更加明确地描述空调系统运行优化问题，另一方面每个子问题的规模相对较小，对应着状态-动作空间较小，应用强化学习控制器通过数据驱动的方式可以更快的学习到良好的控制策略，在本案例中，强化学习控制器在第一年探索阶段的表现就已经超越了基准策略。

综上所述，本案例中强化学习控制器表现出了具有对先验知识依赖程度较低，良好的自适应性和自调整性的特点，适合应用于难以建模的空调系统中，不断强化提升空调系统运行水平。

4.4 应用特点分析与总结

本小节在本章三个应用案例的研究基础上，对基于强化学习算法的空调系统运行方法特点进行分析与总结。

强化学习控制器的优势基础来自于强化学习算法可以通过分析运行数据改进自身控制策略，尽可能的使被控过程累计奖赏最大化。相比 PID 控制器等局部控制器，基于强化学习算法的空调系统控制方法属于全局控制方法，具备目标导向的特点，可以综合考虑影响空调系统运行目标各影响因素，进而对被控变量进行调整，从而优化空调系统运行。此外，强化学习控制器通过数据驱动的方式改进空调系统控制策略，无需建立建筑空调系统仿真模型，具备免模型特性，降低了对先验知识的依赖，适用于建筑（如案例 4.2）和空调系统（如案例 4.3）难以

建立精确且计算效率较高的数学模型的场景，与免模型控制方法中的基于规则的控制方法相比，增强了控制器的动态调整性，控制器的自适应性强，与基于模型的控制方法相比，避免了模型建立、模型校验等繁琐的工作。最后，强化学习控制器可以最大化长期累积奖赏，即可以对空调系统运行中的控制变量进行序列优化，针对控制变量的序列优化，模型预测控制采取的是滚动优化的方式，计算开销较高，强化学习控制器只需计算当前状态下不同动作的 Q 值，因此在每次做决策时，强化学习控制器的响应速度更快。

然而，基于强化学习算法的空调系统运行方法也存在不足之处。首先，在实际应用时，需根据具体的空调系统运行问题合理地设定强化学习算法中的参数（例如折扣因子的设定、Q 值函数逼近器相关参数的设定等），目前相关参数设定尚缺乏指导依据，需做进一步的研究。此外，由于强化学习控制器需要通过探索来不断改进控制策略，所以无法在应用初期立即达到良好的控制效果，参考案例 4.1，对于大规模空调系统运行优化问题，意味着数据维度较高，控制器需探索的状态-动作空间较大，需要较长时间的探索学习其表现才能接近最优控制策略。因此，在小规模运行问题中，强化学习控制器更具优势，更具实际应用价值。对于大多数小规模空调系统运行优化场景，由于建筑运行具有长生命周期的特点，意味着监测的数据均属于“瘦高型”数据，即空调系统监测数据的样本数要远大于数据维度，对于这类“瘦高型”数据，初期探索带来的代价远小于建筑全生命周期中得到的回报，因此，本文提出的基于强化学习算法的空调系统运行优化具备应用与推广价值。

4.5 本章小结

本章通过应用案例展示了强化学习算法在空调系统运行优化中的应用过程，并对应用情况进行讨论分析。首先介绍了强化学习控制器在变风量空调系统中的应用，计算了强化学习控制器在单区域和多区域送风量调节中的运行效果，认为在小规模建筑空调系统运行问题中，强化学习控制器更具优势，更具实际应用价值。接下来探讨分析了强化学习控制器在小型住宅建筑空气源热泵联合电辅热系统中的应用和强化学习控制器在寒冷地区太阳能-地源热泵系统中的应用，与不同的基准策略进行了对比，量化了强化学习控制器的控制效果。在此基础上分析总结了强化学习控制器的特点，认为本文提出的基于强化学习算法的空调系统运行优化具备应用与推广价值。

第5章 工具开发

5.1 编程环境简介

Python[88]是一种解释型计算机程序设计语言，其语法简洁清晰，易于学习。目前，Python 在科学计算和数据挖掘领域应用广泛。与 Matlab 相比，Python 免费易用，众多开源的数据挖掘库和科学计算库均提供了 Python 的调用接口，在数据预处理、数据库连接、机器学习算法、数据可视化等方面均可以找到功能强大的工具。此外，Python 是一种通用编程语言，对于界面设计、网络通信、文件管理等各种需求，Python 同样有着丰富的扩展库，生态环境更加完整，开发者可以用 Python 快速实现完整应用程序所需的各种功能。近年来，越来越多的科研人员使用 Python 语言开展建筑能耗分析和空调系统运行策略方面的研究工作，例如西北太平洋国家实验室(PNNL)使用 Python 语言开发了名为 Volttron 的开源控制系统平台，可以使用 Python 语言在平台上开发建筑空调系统控制算法。因此，在上述背景下，本文选择 Python 语言作为工具开发语言。

Anaconda[89]是 Python 的一个科学计算发行版，内置了数百个 Python 的常用库，其中也包括许多数据处理、数据挖掘的库。同时 Anaconda 自动集成了 MKL(Math Kernel Libaray)库，加速矩阵运算和线性代数运算。总体来说，Anaconda 是目前最好的科学计算的 Python 环境，安装方便，性能较高，因此本文选择 Anaconda 作为工具开发环境。Anaconda 下载与安装可以参考清华大学开源软件镜像站[90]中关于 Anaconda 部分的相关说明。

5.2 工具设计与运行流程简介

本章的主要工作为开发用于构建建筑空调系统运行优化所需的强化学习控制器的软件工具。总体来说，软件工具的运行流程应如图 5.1 所示，为实现图中描述的流程，在工具设计时，应解决以下几个问题。

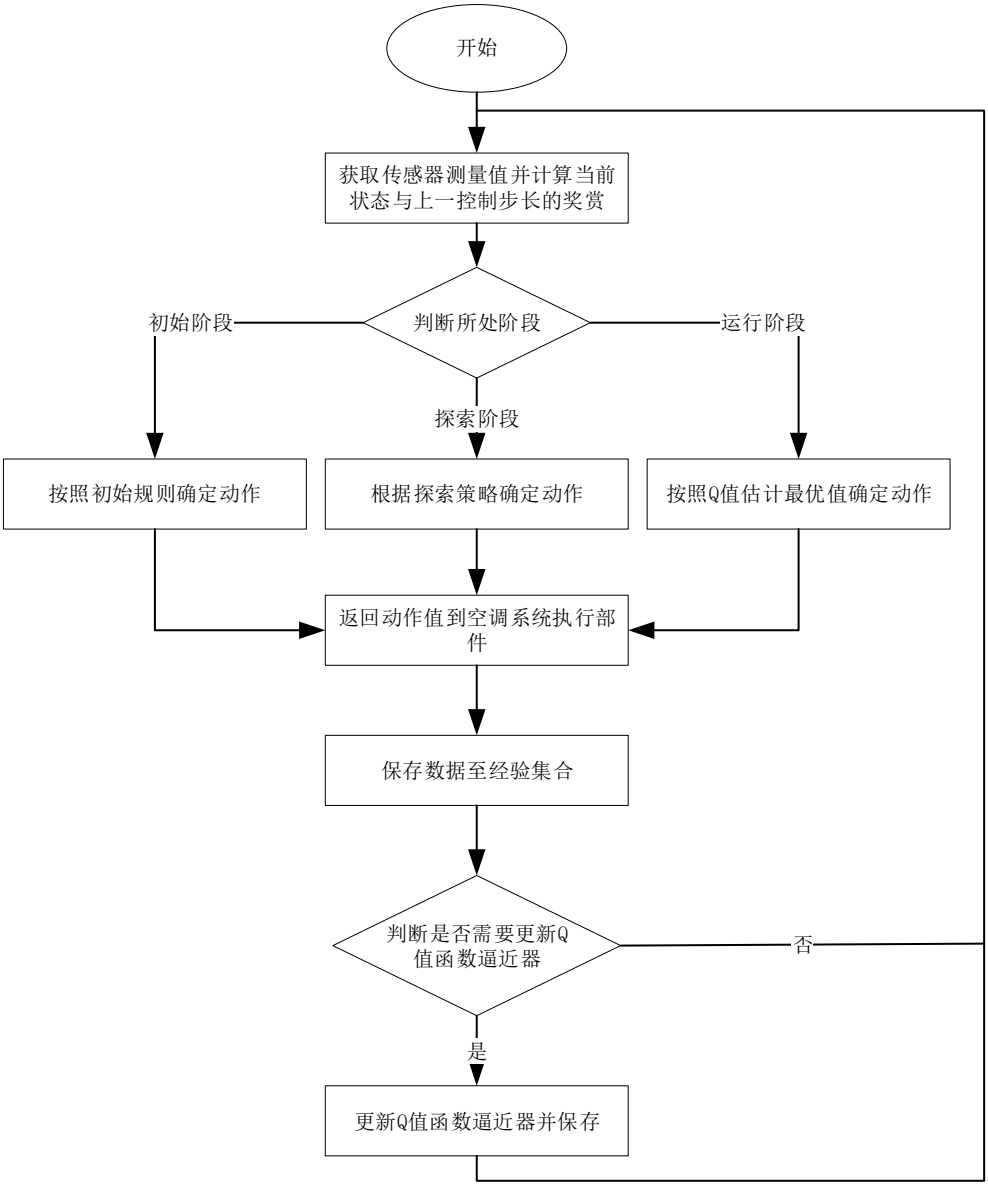

图 5.1 工具运行流程图

首先，应解决数据传送的问题，即将建筑空调系统中的传感器测量值发送至强化学习控制器中，再将强化学习控制器确定的动作值传递至执行部件中。目前，空调系统的监测与控制体系如图 5.2 所示，为实现不同设备间的通信，需设定网络通信层所使用的通信协议，目前常用的网络协议有 TCP/IP 协议、UDP 协议等。本文采用 UDP 协议作为通信协议。通过设置 IP 地址和端口即可实现不同设备间的通信，其中 IP 地址用来确定该设备地址，端口对同一设备的不同进程进行区分，利用同一台上位机中可以运行多个强化学习控制器，每个控制器使用一个专属的端口。

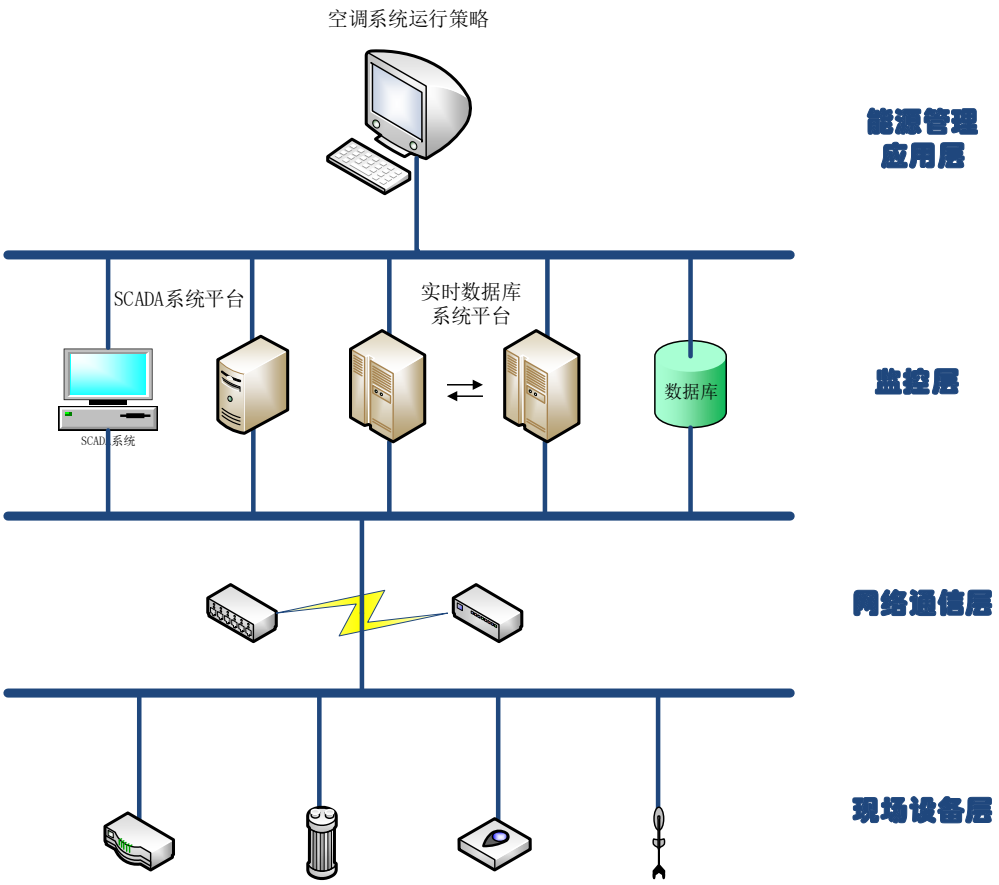

图 5.2 建筑空调系统监测与控制体系

接下来，应解决参数设置的问题，参数包括网络通信相关参数和基于强化学习的空调系统运行优化算法流程中准备阶段涉及的参数，表 5.1 为总结的工具中需要设定的参数。本文将此表存为 csv 格式文件，将此文件称为配置文件，在应用时可使用 Excel 等软件填写配置文件，软件工具通过读取此配置文件获取参数设定。

表 5.1 工具中需设定的参数

项目信息	值	备注
端口		此强化学习控制器的端口号
数据格式		传送至上位机的数据格式如(请使用英文填写，例如时间,室外温度,室内温度：Time,Tout,Tz)
分隔符		使用此符号对传送回来的数据进行分隔，可以获得每个监控变量的值
动作空间		被控变量取值
控制步长		分钟

初始阶段时长		天
探索阶段时长		天
折扣因子		取值 0-1
epsilon_0		注:目前工具仅实现了 ϵ 探索策略,探索策略参数，epsilon_0 表示初始探索比例
delta_epsilon		探索策略参数,表示探索比例每天减少 delta_epsilon
epsilon_min		探索策略参数,表示探索阶段探索比例的最小值
值函数逼近器		ET(极端随机树)或者 ANN(人工神经网络）
树的数目		当值函数逼近器为 ET 时填写，默认为 60
分裂节点最少所需样本数		当值函数逼近器为 ET 时填写，默认为 3
隐藏层节点数		当值函数逼近器为 ANN 时填写，默认为 32,64,32
隐藏层激活函数		当值函数逼近器为 ANN 时填写（可以设置 relu,sigmoid 和 inear)，默认为 relu,relu,relu
输出层激活函数		当值函数逼近器为 ANN 时填写，默认为 linear
固定值函数更新频率		次/天
修改运行设备设定值所在设备的 IP 地址		确定输出地址
修改运行设备设定值程序的端口		确定输出地址

在解决上述问题后，接下来需考虑如何在软件工具中描述空调系统运行优化问题，并将规则的引入加入到软件工具中，其中前者主要是将空调系统运行优化问题建模为马尔科夫决策过程，即定义状态和奖赏（动作空间已经在配置文件中填写），状态与奖赏可以从传感器测量得到的数据计算得出。规则引入主要包括初始阶段空调系统运行规则、用于探索空间缩减的规则和判断是否更新 Q 值函数逼近器的条件，上述规则均为如果…则…否则…形式的条件判断语句。因此，用户可以通过编写简单的 Python 脚本的方式实现上述需求。为此，本文利用 Python 语言面向对象的特性，编写 RLC_Base 基类，将在应用于具体空调系统运行优化问题中不需要修改的部分均在基类中完成，确定了 RLC_Base 基类的子类 RLC 类框架，表 5.2 中为 RLC 类中需要根据具体应用问题编写的方法，在 RLC 类中，可以通过 observation 属性得到最新获取的空调系统反馈回来的监测值。工具中的主要程序代码见附录。

表 5.2 RLC 类中的方法说明

方法名	方法用途	方法返回数据类型
get_state	定义状态	包含状态的列表
get_reward	定义奖赏	数值
init_rules	初始阶段空调系统运行规则	包含设定值的列表
shrinkage_exploration_space_rules	用于探索空间缩减的规则	包含动作的列表
set_condition_for_updating_Q_func	判断是否更新 Q 值函数逼近器	布尔值

在填写好配置文件和编写好 RLC 类中的方法后，即可运行程序，接下来本文将 4.3 节中强化学习控制器在严寒地区太阳能-地源热泵系统中的应用中生活热水制备部分作为简单应用示例，展示工具应用的具体流程。

首先在系统中建立强化学习控制器所在的工作目录，将 config.csv(配置文件)，rlc_base.py(RLC_Base 基类文件)，rlc.py(RLC 子类文件)和 main.py(主程序文件)拷贝至工作目录下。

根据网络通信需要和 4.3.2 节中描述的准备阶段工作内容填写配置文件，本文利用本机 TRNSYS 软件模拟真实环境，因此将负责修改运行设备设定值所在设备的 IP 地址为本机地址 127.0.0.1。TRNSYS 每次传送至控制器的数据为分号分隔的数据，包括时间、太阳辐射强度、室外空气温度、太阳能集热器入口温度、太阳能集热器出口温度、循环水流量和循环泵功率，因此，在数据格式填写为“Time;Rad;Tair;Tci;Tco;m;Ppump”，分隔符设置为“;”，其他设置如图 5.3 所示。

项目信息	值	备注
端口	9999	此强化学习控制
数据格式	Time;Rad;Tair;Tci;Tco;m;Ppump	传送至上位机的
分隔符	;	使用此符号对传
动作空间	0,0.6,0.8,1.0,1.2	被控变量取值,同
控制步长	10	分钟
初始阶段时长	5	天
探索阶段时长	360	天
折扣因子	0	取值0-1
epsilon_0	0.5	注:目前工具仅实
delta_epsilon	0.02	探索策略参数,表
epsilon_min	0.1	探索策略参数,表
值函数逼近器	ET	ET(极端随机树)或
树的数目	50	当值函数逼近器
分裂节点最少所需样本数	3	当值函数逼近器
隐藏层节点数		当值函数逼近器
隐藏层激活函数		当值函数逼近器
输出层激活函数		当值函数逼近器
固定值函数更新频率	1	次/天
负责修改运行设备设定值所在设备的IP地	127.0.0.1	确定输出地址
负责修改运行设备设定值程序的端口	8888	确定输出地址

图 5.3 配置文件中的设定值

接下来定义 RLC 子类，根据准备阶段的描述编写 rlc.py 文件如图 5.4 所示。

```python
class RLC(BASE_RLC):
    '''
    RLC子类，用于定义状态与奖赏，和编写强化学习控制器所需的规则
    '''

    def init_rules(self):
        '''
        设定空调系统初始运行规则
        返回值：为被控变量设定值的列表,列表中的每一项为一个控制变量的设定值
        '''
        if self.observation['Rad'] < 250:
            return [0]
        else:
            return [1.0]

    def get_state(self):
        '''
        计算当前状态值
        返回值：包含状态的列表
        '''
        state = [self.observation['Rad'],self.observation['Tci'],self.observation['Tair']]
        return state

    def get_reward(self):
        '''
        计算当前奖赏值
        返回值：奖赏值
        '''
        Psolar = 4.19 * self.observation['m'] * (self.observation['Tco'] - self.observation['Tci'])
        Ppump = self.observation['Ppump']
        reward = Psolar - Ppump
        return reward

    def shrinkage_exploration_space_rules(self):
        '''
        缩减探索空间
        返回值：列表的列表，其中每个列表为每个控制变量的可取值范围
        '''
        if self.observation['Rad'] < 250:
            return [[0]]

    def set_condition_for_updating_Q_func(self):
        '''
        判断是否有必要立即更新Q值函数逼近器
        返回值：布尔值
        '''
        if self.get_reward() < 0:
            return True
        else:
            return False
```

图 5.4 在 rlc.py 文件中编写的 RLC 子类

最后，开启终端并切换至强化学习控制器所在的工作目录，输入 python main.py 启动主运行程序，即可运行强化学习控制器。在运行过程中，为应对现场断电等突发状况，工具设置了保存功能，将经验集合保存为 batch.csv 文件和将 Q 值函数逼近器参数保存为 Q_Approximator.model 文件，在软件工具遇突发情况突然关闭再打开时，工具可以读取上述文件继续运行。

5.3 本章小结

本章首先介绍了编程语言 Python 的特点及本文使用的运行环境 Anaconda，

在前面章节的基础上，使用 Python 语言开发了用于构建空调系统运行优化所需的强化学习控制器的软件工具，介绍了软件工具的设计思路和运行流程，用户可以通过填写配置文件和编写简单脚本的方式构建强化学习控制器，最后本章通过应用案例对工具的详细使用流程进行了展示。

第6章 结论与展望

6.1 结论

空调系统运行优化是建筑节能的重要组成部分。空调系统运行优化的基本目标是在满足被控区域舒适度要求的前提下尽可能的降低能耗或者能耗消费。为了适应室外不断变化的天气情况和室内负荷变化情况，如何动态调整系统中的设定值或者运行规则从而提高系统能效一直是空调系统运行优化的研究重点。随着空调系统监测技术的成熟，获取空调系统的运行数据不再困难，探索数据驱动的空调系统运行优化具有十分重要的研究意义。本文将强化学习算法应用于空调系统运行优化中，旨在通过数据驱动的方式不断提升空调系统运行能力。

本文首先对目前的强化学习理论进行梳理，选取拟合 Q 迭代算法作为基于强化学习的空调系统运行优化的基础算法。接下来，对强化学习算法在空调系统运行优化中的应用情况进行探讨分析，提出了将基于规则的控制方法与基于强化学习的控制方法相结合的应用理念，通过规则的引入缩减强化学习控制器所需的探索空间，保证学习型控制器的可靠性，在此基础上分析了应用过程面临的具体问题，确定了基于强化学习算法空调系统运行优化的通用算法流程。然后通过应用实例验证了基于强化学习算法的空调系统运行优化方法的有效性，结果表明，与基准策略相比，将强化学习控制器应用于变风量空调系统和空气源热泵联合电辅热系统中，在运行阶段可有效降低运行费用，将强化学习控制器应用于严寒地区太阳能-地源热泵系统中，在运行阶段可有效提高系统净集热量与净蓄热量，降低冬季采暖阶段地源热泵系统总能耗。此外，对强化学习控制器的应用特点进行了分析与总结。最终利用 Python 语言编程，开发出构建空调系统强化学习控制器所需工具，为嵌入在建筑能源管理与控制系统中提供基础，指导建筑空调系统运行优化。

本文通过分析和总结，得出以下结论：

(1) 拟合 Q 迭代算法是一种免模型强化学习方法，具有数据利用效率高，算法稳定性强的特点，适合作为解决建筑空调系统运行优化问题的基础强化学习算法。

(2) 针对强化学习算法的特点，本文将其在空调系统运行优化中的应用过程分为准备阶段、初始阶段、探索阶段和运行阶段，描述了各阶段完整的算法流程，算法具备一定的通用性。通过模拟仿真的方式对本文提出的基于强化学习算法的

空调系统运行优化方法进行了验证，结果显示，本文提出的空调系统运行方法在满足室内用户舒适度需求的同时，可以有效降低建筑能耗或者建筑运行费用。

(3) 本文提出的基于强化学习算法的空调系统运行优化方法具有对先验知识的依赖程度低，自适应性强的特点，此外，与模型预测控制方法相比，控制器在做决策时，无需进行滚动优化，故在响应速度方面具有优势，具有在实际工程中应用与推广价值。

6.2 进一步工作的方向

由于本人知识水平的欠缺，加上能力与时间等因素的限制，使得本文对基于强化学习算法的空调系统运行方法的研究还不完善，在研究中还有不少问题未能解决，下面是本文还没有解决的问题并需要深入研究的问题，在此提出供参考。

(1) 本文提出的基于强化学习算法的空调系统运行方法虽然经过了模拟仿真验证，但是算法运行效果验证所需时间周期较长和本研究时间限制，因此并未在实验室条件或者在实际建筑中进行验证。未来的工作可以将算法应用在实际建筑中，对算法进行实际的验证与完善。

(2) 本文研究重点在于如何应用强化学习算法提升空调系统运行水平，在研究中假设传感器测得的数据即为准确的数据，但在真实的空调系统中监测数据会出现缺失值和异常值的情况，所以在今后的研究中可以增加对数据预处理展开研究，完善基于强化学习算法的空调系统运行流程。

(3) 在实际应用时，需根据待解决的空调系统运行问题合理地设定强化学习算法中的参数（例如折扣因子的设定、Q 值函数逼近器相关参数的设定等），目前相关参数设定主要凭借经验，尚缺乏依据，需做进一步的研究，为具体应用中的参数设置提供指导。

(4) 本文开发的用于构建建筑空调系统运行优化所需的强化学习控制器的软件工具在实际使用时需填写配置文件和编写脚本，使用方式不够简便，今后的工作包括对软件工具的优化，并将工具嵌入在建筑能源管理与控制系统中。

随着智慧建筑的兴起和对建筑节能要求的提高，探索和发展有效的数据驱动的空调系统运行优化方法是当前建筑节能领域的研究热点，笔者认为，将工程人员经验与机器学习能力相结合提升空调系统运行能力是未来发展趋势。为实现建筑空调系统从粗放式运行向精细化运行转变的目标，我们仍需不断努力。

致谢

时光飞逝，转眼间近三年的硕士研究生生活即将结束。在论文即将完成之际，我要向所有帮助过我的老师、同学、朋友和亲人致以最诚挚的谢意。

首先，非常感谢我的导师潘毅群教授对我的帮助，潘老师严谨的治学态度、精益求精的工作作风和积极乐观的生活态度深深的影响了我，能够加入潘门的大家庭是我莫大的荣幸。在选题过程中，潘老师鼓励我去深入自己感兴趣的领域，在课题研究过程中，潘老师给了我悉心的指导，对论文的修改细致入微。此外，潘老师提供给我很多接触实际项目的宝贵机会，拓展了我的视野，提高了我的专业水平。同样非常感谢课题组的黄治钟老师，黄老师渊博的知识储备、缜密的逻辑思维让我由衷地敬仰，与黄老师的交谈总能获益匪浅，在课题研究过程中，黄老师总能一针见血指出存在的问题，并提出宝贵的指导意见。衷心感谢二位老师对我的谆谆教导，这将是我受益终生的宝贵财富。

感谢李玉明老师、王秋涧师兄、朱明亚师姐和刘海静师姐对我学术上的帮助与指导，同样感谢三年来与我朝夕相处，共同学习的同门师兄弟姐妹们：孙勇、林美顺、杨陶、侯丹琳、秦冰月、徐蕾、吕岩、潘媛、杨一昆、谢建彤、佘昭、李诗尧，非常留恋与你们共同进步的快乐时光。感谢我的室友，在生活上对我的包容。感谢一同度过硕士生涯的同窗好友们。感谢我的女朋友在我完成论文日子里对我的生活无微不至的关心，是你的一路陪伴坚定了我的信念。感谢我的家人，尤其是我的父母，是你们让我永远拥有一个温馨的港湾，你们无私的爱是我继续前行的不竭动力。

最后感谢各位老师和专家对本论文的评阅，由于本人学识、精力有限，文中不免有错误和待改进之处，希望各位前辈提出宝贵意见。

王尉同

2018 年 3 月

参考文献

[1] 国家发展改革委、国家能源局，能源发展“十三五”规划：http://www.ndrc.gov.cn/zcfb/zcfbtz/201701/t20170117_835278.html
[2] 王峰丽. 中国新能源发电装备发展概况与趋势[J]. 现代制造, 2010(39):10-13.
[3] Liu Y, Yan H, Lam J C. Thermal comfort and building energy consumption implications – A review[J]. Applied Energy, 2014, 115(4):164-173.
[4] 清华大学建筑节能研究中心. 中国建筑节能年度发展研究报告. 2017[M]. 中国建筑工业出版社, 2017.
[5] 王宇. 基于设计方案的公共建筑全生命周期能耗分析与评价[D]. 清华大学, 2012.
[6] Samuel A L. Some studies in machine learning using the game of checkers[M]// Computers & thought. MIT Press, 1995:206-226.
[7] Mitchell T M, Carbonell J G, Michalski R S. Machine Learning[M]. China Machine Press ;McGraw-Hill Education (Asia), 2003.
[8] Fan C, Xiao F, Wang S. Development of prediction models for next-day building energy consumption and peak power demand using data mining techniques[J]. Applied Energy, 2014, 127(6):1-10.
[9] Fan C, Xiao F, Zhao Y. A short-term building cooling load prediction method using deep learning algorithms[J]. Applied Energy, 2017, 195:222-233.
[10] 徐哲恬. 基于 ANN 的机械通风开式冷却塔运行性能在线预测研究[D]. 同济大学, 2013.
[11] Han H, Gu B, Kang J, et al. Study on a hybrid SVM model for chiller FDD applications[J]. Applied Thermal Engineering, 2011, 31(4):582-592.
[12] Yoon Y R, Shin S H, Moon H J. Analysis of Building Energy Consumption Patterns according to Building Types Using Clustering Methods[J]. Journal of the Korean Society of Living Environmental System, 2017, 24(2):232-237.
[13] Du Z, Fan B, Jin X, et al. Fault detection and diagnosis for buildings and HVAC systems using combined neural networks and subtractive clustering analysis[J]. Building & Environment, 2014, 73(3):1-11.
[14] 高阳, 陈世福, 陆鑫. 强化学习研究综述[J]. 自动化学报, 2004, 30(1):86-100.
[15] Wang S, Ma Z. Supervisory and Optimal Control of Building HVAC Systems: A Review[J]. HVAC & R Research, 2008, 14(1):3-32.
[16] ASHRAE. 2003. 2003 ASHRAE Handbook—HVAC Applications. Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.
[17] 侯丹琳, 潘毅群, 黄治钟. 一种基于粒子群算法的暖通空调过程对象辨识方法[J]. 建筑节能, 2017(3):19-23.
[18] 刘健, 张桂青. 便携式 PID 自整定控制器在 HVAC 系统中的实现[J]. 福州大学学报, 2008, 36(s1):119-122.
[19] 刘羽岱, 潘毅群, 黄治钟. 基于模型预测控制的空调系统运行优化仿真研究[C]// 上海市制冷学会 2013 年学术年会. 2013.
[20] House J M, Smith T F. System approach to optimal control for HVAC and building systems[J].

Ashrae Transactions, 1995, 101(2):647-660.
[21] Zaheer-Uddin M, Zheng G R. Multistage optimal operating strategies for HVAC systems[J]. Ashrae Transactions, 2001, 107:346-352.
[22] Wang S, Jin X. Model-based optimal control of VAV air-conditioning system using genetic algorithm[J]. Building & Environment, 2000, 35(6):471-487.
[23] Lebrun J, Wang S. Evaluation and Emulation of Building Energy Management Systems: Synthesis Report[J]. 1993.
[24] Sud, I. 1984. Control strategies for minimum energy usage. ASHRAE Transactions 90(2A):247–77.
[25] Wang S. Dynamic simulation of a building central chilling system and evaluation of EMCS on-line control strategies[J]. Building & Environment, 1998, 33(1):1-20.
[26] Zhang, Y., V.I. Hanby. Model-Based Control of Renewable Energy Systems in Buildings[J]. HVAC & R Research, 2006, 12(sup1):739-760.
[27] Curtiss P S, Brandemuehl M J, Kreider J F. Energy management in central HVAC plants using neural networks[J]. ASHRAE Transactions, 1994, 100(1):476-493.
[28] Bradford, J.D. 1998. Optimal supervisory control of cooling plants without storage. PhD thesis, University of Colorado.
[29] Braun J E. Methodologies for the Design and Control of Central Cooling Plants[J]. Ph D Mechanical Engineering, 1988.
[30] Lu Lu, Wenjian Cai, Lihua Xie, et al. HVAC system optimization—in-building section[J]. Energy & Buildings, 2005, 37(1):11-22.
[31] Glenn Platt, Geoff Poulton, Josh Wall, et al. Dynamic Zone Modelling for HVAC System Control[J]. International Journal of Modelling Identification & Control, 2010, 9(1/2):5-14.
[32] Berardino J, Nwankpa C. Dynamic Load Modeling of an HVAC Chiller for Demand Response Applications[C]// First IEEE International Conference on Smart Grid Communications. IEEE, 2010:108-113.
[33] Ma Y, Kelman A, Daly A, et al. Predictive Control for Energy Efficient Buildings with Thermal Storage: Modeling, Stimulation, and Experiments[J]. Control Systems IEEE, 2012, 32(1):44-64.
[34] Moroşan P D, Bourdais R, Dumur D, et al. Building temperature regulation using a distributed model predictive control[J]. Energy & Buildings, 2010, 42(9):1445-1452.
[35] Oldewurtel F, Parisio A, Jones C N, et al. Energy efficient building climate control using Stochastic Model Predictive Control and weather predictions[C]// American Control Conference. IEEE, 2010:5100-5105.
[36] Dong B, Lam K P. A real-time model predictive control for building heating and cooling systems based on the occupancy behavior pattern detection and local weather forecasting[J]. Building Simulation, 2014, 7(1):89-106.
[37] Killian M, Kozek M. Ten questions concerning model predictive control for energy efficient buildings[J]. Building & Environment, 2016, 105:403-412.
[38] Široký J, Oldewurtel F, Cigler J, et al. Experimental analysis of model predictive control for an energy efficient building heating system[J]. Applied Energy, 2011, 88(9):3079-3087.
[39] Siddharth Goyal, Herbert A. Ingley, Prabir Barooah. Occupancy-based zone-climate control

for energy-efficient buildings: Complexity vs. performance ☆[J]. Applied Energy, 2013, 106(11):209-221.
[40] Sun J, Reddy A. Optimal control of building HVAC&R systems using complete simulation-based sequential quadratic programming (CSB-SQP)[J]. Building & Environment, 2005, 40(5):657-669.
[41] Braun, J.E. 1990b. Reducing energy costs and peak electrical demand through optimal control of building thermal storage. ASHRAE Transactions 96(2):876–88.
[42] Ling K V, Dexter A L. Expert control of air-conditioning plant ☆[J]. Automatica, 1994, 30(5):761-773.
[43] Coninck R D, Baetens R, Saelens D, et al. Rule-based demand-side management of domestic hot water production with heat pumps in zero energy neighbourhoods[J]. Journal of Building Performance Simulation, 2014, 7(4):271-288.
[44] Sutton R, Barto A. Reinforcement Learning: An Introduction[M]. MIT Press, 2005.
[45] Bellman R. On the Theory of Dynamic Programming[J]. Proceedings of the National Academy of Sciences of the United States of America, 1952, 38(8):716.
[46] 庄超. 基于高斯过程回归的强化学习算法研究[D]. 苏州大学, 2014.
[47] Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning[J]. Nature, 2015, 518(7540):529.
[48] Silver D, Schrittwieser J, Simonyan K, et al. Mastering the game of Go without human knowledge[J]. Nature, 2017, 550(7676):354.
[49] Simeng Liu, Gregor P. Henze. Experimental analysis of simulated reinforcement learning control for active and passive building thermal storage inventory : Part 2: Results and analysis[J]. Energy and Buildings, 2006, 38(2):148-161.
[50] Dalamagkidis K, Kolokotsa D, Kalaitzakis K, et al. Reinforcement learning for energy conservation and comfort in buildings[J]. Building & Environment, 2007, 42(7):2686-2698.
[51] Barrett E, Linder S. Autonomous HVAC Control, A Reinforcement Learning Approach[M]// Machine Learning and Knowledge Discovery in Databases. Springer International Publishing, 2015:3-19.
[52] Li B, Xia L. A multi-grid reinforcement learning method for energy conservation and comfort of HVAC in buildings[C]// IEEE International Conference on Automation Science and Engineering. IEEE, 2015:444-449.
[53] Yang L, Nagy Z, Goffin P, et al. Reinforcement learning for optimal control of low exergy buildings[J]. Applied Energy, 2015, 156:577-586.
[54] Urieli D, Stone P. A learning agent for heat-pump thermostat control[C]// International Conference on Autonomous Agents and Multi-Agent Systems. International Foundation for Autonomous Agents and Multiagent Systems, 2013:1093-1100.
[55] Ruelens F, Iacovella S, Claessens B J, et al. Learning Agent for a Heat-Pump Thermostat With a Set-Back Strategy Using Model-Free Reinforcement Learning[J]. Energies, 2015, 8(8):8300-8318.
[56] Costanzo G T, Iacovella S, Ruelens F, et al. Experimental analysis of data-driven control for a building heating system[J]. Sustainable Energy Grids & Networks, 2015, 6:81-90.
[57] Wei T, Wang Y, Zhu Q. Deep Reinforcement Learning for Building HVAC Control[C]//

Design Automation Conference. ACM, 2017:22.
[58] Vázquez-Canteli J, Kämpf J, Nagy Z. Balancing comfort and energy consumption of a heat pump using batch reinforcement learning with fitted Q-iteration[J]. Energy Procedia, 2017, 122:415-420.
[59] Cheng Z, Zhao Q, Wang F, et al. Satisfaction based Q-learning for integrated lighting and blind control[J]. Energy & Buildings, 2016, 127:43-55.
[60] 陈学松, 杨宜民. 强化学习研究综述[J]. 计算机应用研究, 2010, 27(8):2834-2838.
[61] Watkins C J C H, Dayan P. Q-learning[C]// Machine Learning. 1992:279--292.
[62] 周志华. 机器学习[M]. 清华大学出版社, 2016 年 1 月.
[63] Breiman L, Friedman J, Olshen R, et al. Classification and Regression Trees. Belmont, CA: Wadsworth International Group.[J]. Biometrics, 1984, 40(3):358.
[64] Michael Bowles, 沙赢, 李鹏. Python 机器学习预测分析核心算法[M]. 人民邮电出版社, 2017.
[65] L Breiman. Random Forest[J]. Machine Learning, 2001, 45:5-32.
[66] Geurts P, Ernst D, Wehenkel L. Extremely randomized trees[J]. Machine Learning, 2006, 63(1):3-42.
[67] Jerome H. Friedman. Greedy Function Approximation: A Gradient Boosting Machine[J]. The Annals of Statistics, 2001, 29(5):1189-1232.
[68] Ernst D, Geurts P, Wehenkel L. Tree-Based Batch Mode Reinforcement Learning.[J]. Journal of Machine Learning Research, 2005, 6(2):503-556.
[69] Lange S, Gabel T, Riedmiller M. Batch Reinforcement Learning[M]// Reinforcement Learning. Springer Berlin Heidelberg, 2012:45-73.
[70] 黄小燕. 基于经验回放 Q-Learning 的最优控制算法[J]. 计算机工程与设计, 2017, 38(5):1352-1355.
[71]《民用建筑供暖通风与空气调节设计规范》(GB50736-2012) .2012.
[72]《公共建筑节能设计标准》（GB50189-2005）.2005
[73] Energy Saver: Setting Your Thermostat for Comfort and Savings: https://energy.gov/energysaver/thermostats
[74] Nest Thermostat: https://store.nest.com/product/thermostat/T3007ES
[75] Building Operations: https://buildingiq.com/products/building-operations/
[76] Chassin D P, Schneider K, Gerkensmeyer C. GridLAB-D: An open-source power systems modeling and simulation environment[C]// Transmission and Distribution Conference and Exposition, 2008. T&D. IEEE/PES. IEEE, 2008:1-5.
[77] 丁强, 邢长新, 江爱朋,等. 热泵供暖系统动态操作节能优化[J]. 化工学报, 2015, 66(6):2166-2173.
[78] 刘羽岱. 空调系统联合仿真平台的开发及利用[D]. 同济大学, 2013.
[79] 高翠. 严寒地区太阳能—土壤源热泵供热系统的分析[D]. 哈尔滨工业大学, 2012.
[80] 范貌宏, RobertC.Brown, 范秀英,等. 影响平板式太阳能热水器热效率诸因素的研究[J]. 环境工程学报, 1999(6):122-126.
[81] 杨前明, 隋志蔚, 孔祥强. 热泵空调变频压缩机与电子膨胀阀控制方法探讨[J]. 电子测量与仪器学报, 2008, 22(z2).
[82] 戴霖姗, 刘金祥, 盛建军,等. 基于 TRNSYS 地埋管地源热泵变流量系统性能分析[J].

暖通空调, 2014(10):89-93.
[83] J. A. Duffie, William A. Beckman, W. M. Worek. Solar Engineering of Thermal Processes, 2nd ed.[M]. A wiley-interscience, 1994.
[84] Reindl D, Sun K K, Kang Y T, et al. Experimental verification of a solar hot water heating system with a spiral-jacketed storage tank[J]. Journal of Mechanical Science & Technology, 2008, 22(11):2228-2235.
[85] Bujedo L A, Rodríguez J, Martínez P J. Experimental results of different control strategies in a solar air-conditioning system at part load[J]. Solar Energy, 2011, 85(7):1302-1315.
[86] 杨少刚. 基于 TRNSYS 地埋管地源热泵变流量系统仿真研究[D]. 山东建筑大学, 2016.
[87] Persson T, Rönnelid M. Increasing solar gains by using hot water to heat dishwashers and washing machines[J]. Applied Thermal Engineering, 2007, 27(2–3):646-657.
[88] Python: https://www.python.org/
[89] Anaconda: https://www.anaconda.com
[90] 清华大学开源软件镜像站 Anaconda 镜像使用帮助: https://mirror.tuna.tsinghua.edu.cn/help/anaconda/

附录 部分程序代码

本附录节选了本课题开发的软件工具中关于强化学习控制器的部分代码（BASE_RLC 基类），用户可以通过继承 BASE_RLC 基类搭建适用于具体空调系统运行优化场景的强化学习控制器，类中各部分功能见注释说明。

以下为代码节选：

```
# -*- coding: utf-8 -*-
# 导入所需工具
from sklearn.ensemble import ExtraTreesRegressor
from collections import defaultdict
import numpy as np
import pandas as pd
from sklearn.externals import joblib
import itertools
import socket

class BASE_RLC:
    '''强化学习控制器基类'''
    def __init__(self, item_names, separator, action_space, epsilon_0, delta_epsilon, epsilon_min, decay, approximator, tree_num, min_samples, init_day, explore_day, update_freq, host_port, client_IP, client_port):
        self.names_list = item_names.split(separator) # 数据名称列表
        self.separator = separator #分隔符
        def foo(x):
            return map(float, x.split(','))

        # 动作集合
        self.action_space = map(list, list(itertools.product(*map(foo, action_space.split(separator)))))
        self.batch = defaultdict(list) #经验集合
        self.decay = decay #折扣因子
        self.approximator = approximator #Q 值函数逼近器
        self.tree_num = tree_num #ET 参数：树的数目
        self.min_samples = min_samples #ET 参数：分裂节点所需最少样本数
        self.init_day = init_day # 初始阶段时长
        self.explore_day = explore_day #探索阶段时长
        self.day = 0 #当前运行天数
```

```
        self.update_freq = update_freq #Q 值函数逼近器更新频率
        self.epsilon_0 = epsilon_0 #探索策略参数
        self.delta_epsilon = delta_epsilon #探索策略参数
        self.epsilon_min = epsilon_min #探索策略参数
        self.host_port = int(host_port) #通信参数
        self.client_IP = client_IP #通信参数
        self.client_port = int(client_port) #通信参数

    def init_rules(self):
        '''
        设定空调系统初始运行规则
        返回值：为被控变量设定值的列表,列表中的每一项为一个控制变量的设定值
        '''
        pass

    def get_state(self):
        '''
        计算当前状态值
        返回值：包含状态的列表
        '''
        pass

    def get_reward(self):
        '''
        计算当前奖赏值
        返回值：奖赏值
        '''
        pass

    def shrinkage_exploration_space_rules(self):
        '''
        缩减探索空间
        返回值：列表的列表，其中每个列表为每个控制变量的可取值范围
        '''
        pass

    def set_condition_for_updating_Q_func(self):
        '''
        判断是否有必要立即更新 Q 值函数逼近器
        返回值：布尔值
        '''
```

```
70.             pass
71.
72.         def receive_and_preprocess_raw_data(self, data):
73.             '''
74.             数据获取及预处理
75.             '''
76.             self.observation = dict()
77.             data_list = map(float, data.split(self.separator))
78.             for i in range(len(self.names_list)):
79.                 self.observation[self.names_list[i]] = data_list[i]
80.             print 'receiving data', self.observation
81.
82.         def store_transition(self, s, a, r, s_):
83.             '''
84.             将数据存入经验集合。
85.             '''
86.             self.batch['s'].append(s)
87.             self.batch['a'].append(a)
88.             self.batch['r'].append(r)
89.             self.batch['s_'].append(s_)
90.
91.         def choose_action(self, state, actions):
92.             '''
93.             根据当前状态选择动作。
94.             '''
95.             # 初始阶段按照初始规则选择动作
96.             if self.day <= self.init_day:
97.                 action = self.init_rules()
98.             # 探索阶段按照探索策略选择动作
99.             elif self.day <= self.explore_day:
100.                X_test_s = pd.DataFrame([state] * len(actions))
101.                X_test_a = self.actions
102.                X_test = pd.concat([X_test_s, X_test_a], axis = 1)
103.                actions_value = self.model.predict(X_test)
104.
105.                prob = max(self.epsilon_0 - self.day * self.delta_epsil
on, self.epsilon_min)
106.                if np.random.uniform() > prob:
107.                    action = actions[np.argmax(actions_value)]
108.                else:
109.                    action = actions[np.random.randint(0, len(actions))
]
```

```
        # 运行阶段选取已知的最优动作
        else:
            X_test_s = pd.DataFrame([state] * len(actions))
            X_test_a = self.actions
            X_test = pd.concat([X_test_s, X_test_a], axis = 1)
            actions_value = self.model.predict(X_test)
            action = actions[np.argmax(actions_value)]
        return action

    def build_Q_approximator(self):
        '''
        初始化 Q 值函数逼近器
        '''
        if self.approximator = 'ET':
            self.model = ExtraTreesRegressor(n_estimators = self.tree_num, min_samples_split = self.min_samples)
        else:
            pass

    def learn(self):
        '''
        训练/更新 Q 值函数逼近器
        '''
        s_ = pd.DataFrame(self.batch['s_'])
        Q_next = pd.DataFrame()
        for action in self.action_space:
            action_df = pd,DataFrame([action]*len(s_))
            input_matrix = pd.concat([s_, action_df],axis = 1).values
            try:
                Q_next[str(action)] = self.model.predict(input_matrix)
            except:
                Q_next[str(action)] = np.zeros(len(self.batch['r']))

        Q_next = Q_next.T.min().values.flatten()
        label = np.array(self.batch['r']) + self.decay * Q_next
        self.build_Q_approximator()

        Xtrain_s = pd.DataFrame(self.batch['s'])
        Xtrain_a = pd.DataFrame(self.batch['a'])
```

```
148.            Xtrain = pd.concat([Xtrain_s, Xtrain_a], axis = 1)
149.
150.            self.model.fit(Xtrain, label)
151.            #存储模型
152.            joblib.dump(self.model, 'Q_Approximator.model')
153.            #存储经验集合
154.            s_df = pd.DataFrame(self.batch['s'])
155.            a_df = pd.DataFrame(self.batch['a'])
156.            r_df = pd.DataFrame(self.batch['r'])
157.            s__df = pd.DataFrame(self.batch['s_'])
158.            pd.concat([s_df, a_df, r_df, s__df], axis = 1).to_csv('batc
h.csv', index=False)
159.
160.        def run(self):
161.            '''
162.            主运行流程
163.            '''
164.            sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
165.            sock.bind(('127.0.0.1', self.host_port))
166.            data, addr = sock.recvfrom(1024)
167.
168.            self.receive_and_preprocess_raw_data(data)
169.            s = self.get_state()
170.            while True:
171.                self.day += 0.01  #实际应用中可以从系统中读取时间信息，此处
为了模拟采用通过计算的方式获得时间信息
172.                try:
173.                    s = s_
174.                except:
175.                    pass
176.                shrinkaged_action_space = self.shrinkage_exploration_sp
ace_rules()
177.                if shrinkaged_action_space is not None:
178.                    action_space = map(list, list(itertools.product(shr
inkaged_action_space)))
179.                else:
180.                    action_space = self.action_space
181.                action = self.choose_action(self, s, action_space)
182.                sock.sendto(str(action), (self.client_IP, self.client_p
ort))
183.
184.                data, addr = sock.recvfrom(1024)
```

```
            self.receive_and_preprocess_raw_data(data)
            s_ = self.get_state()
            r = self.get_reward()
            self.store_transition(s, action, r, s_)
            if (np.mod(self.day, 1.0/self.update_freq) == 0) or self.set_condition_for_updating_Q_func():
                self.learn()
                print 'update Q-value function approximator.'
```

个人简历、在读期间发表的学术论文与研究成果

个人简历：

王尉同，男，1993 年 11 月出生。

2015 年 9 月毕业于同济大学 建筑设施智能技术专业 获学士学位。

2015 年 9 月至今 入读同济大学硕士研究生。

已发表论文：

[1] 王尉同, 潘毅群, 黄治钟. 基于强化学习的空调系统运行优化[C]. 2017 年全国暖通空调模拟学术年会论文集. 2017.12

待发表论文：

[1] 王尉同, 潘毅群, 黄治钟. 梯度渐进回归树在建筑能耗短期预测中的应用. 建筑节能（已录用）