



重慶理工大學

化工原理课程设计

学 院 化学化工学院

专 业 化学工程与工艺

班 级 122150103

姓 名 刘抗非

学 号 12115990136

本科生课程设计（论文）任务书

设计（论文）题目： SO₂ 吸收填料塔设计

学院： 化学化工学院 专业： 化学工程与工艺

班级： 化学工程与工艺三班 学生： 刘抗非

指导教师（含职称）： 李新利 专业负责人： 郝江瑜

1. 设计（论文）的主要任务及目标

按不同学号给定的任务参数如下表：

按不同学号给定的吸收填料塔参数表

SO ₂ 吸收率	混合气体处理量 (m ³ /h)			操作条件
	900 m ³ /h	1000 m ³ /h	1100 m ³ /h	
96 %	4-6	25-27	16-18	常压
98 %	19-21	1-3	34-36	绝压 120kPa
96 %	10-12	31-33	7-9	常压
98 %	28-30	13-15	22-24	绝压 120kPa
	5.5%	6%	6.5%	
	进塔气体中SO ₂ 的摩尔分数			

依据学号查得个人任务如下：

设计一个SO₂ 吸收填料塔，在常温，绝压 120kPa 操作条件下，处理气体流量为 1000m³/h，SO₂ 含量为 6.0% 的废气，要求SO₂ 吸收率大于等于 98%，使用聚丙烯散装鲍尔环作为填料。

2. 设计（论文）的基本要求和内容

1. 选择合适的填料规格；2. 计算所需填料塔的尺寸和高度；3. 确定吸收液（H₂O）的循环量；4. 分析操作条件对吸收效率的影响；5. 提供详细的设计图纸和流程图；6. 编写设计说明书，包含计算过程和结果分析。

3. 主要参考文献

[1] 刘海洋. \LaTeX 入门^[1] [M]. 北京: 电子工业出版社, 2013.

[2] MITTELBACH F, GOOSSENS M, BRAAMS J, et al. The \LaTeX Companion[M].
2nd ed. Reading, Massachusetts : Addison-Wesley, 2004.

4. 进度安排

	设计（论文）各阶段名称	起止日期
1	文献调研与方案设计	2024-09-07 至 2024-09-08
2	填料塔尺寸计算	2024-09-09 至 2024-09-11
3	设计图纸绘制与说明书编写	2024-09-12 至 2024-09-20

SO₂ 吸收填料塔设计

刘抗非 化学工程与工艺三班 12115990136 指导教师：李新利

摘要

本设计旨在开发一个处理量为 **1000m³/h** 的SO₂ 吸收填料塔，进料气体中SO₂ 含量为 **6.0%**，其余为空气。在常温、绝压 **120kPa** 操作条件下，要求SO₂ 吸收率达到 **98%** 以上。设计中选用聚丙烯散装鲍尔环作为填料。通过计算填料塔的尺寸和高度，选择合适的填料类型和规格，确定吸收液的循环量和浓度，分析操作条件对吸收效率的影响，并提供详细的设计图纸和流程图。本项目的创新点包括：1. 采用高效填料提高吸收效率；2. 优化填料塔设计以减少能耗；3. 提供详细的设计说明书，包含计算过程和结果分析。

关键词：SO₂ 吸收 填料塔 聚丙烯鲍尔环 设计优化 吸收效率

目 录

第 1 章	吸收塔填料塔的设计方案	1
第 1.1 节	吸收流程的选择	1
1.1.1	单剂和多剂的选择	1
1.1.2	单塔和多塔的选择	1
1.1.3	逆流与并流的选择	1
第 1.2 节	塔内填料的选择	2
第 1.3 节	吸收剂的选择	4
第 1.4 节	操作条件的选择	4
第 1.5 节	工艺流程图	5
第 2 章	吸收填料塔工艺参数计算	7
第 2.1 节	基础物性数据	7
2.1.1	气相物性数据	7
2.1.2	液相物性数据	7
2.1.3	气液相平衡数据	8
第 2.2 节	物料衡算	8
第 2.3 节	填料塔工艺尺寸计算	9
2.3.1	填料塔塔径的计算	9
2.3.2	填料塔塔径的校核	10
2.3.3	传质单元数的计算	12
2.3.4	膜吸收系数的计算	12
2.3.5	总传质系数的计算	13
2.3.6	填料层高度的计算	14
2.3.7	填料层压降的计算	14
2.3.8	塔附属高度的计算	15
第 2.4 节	吸收塔工艺尺寸总结	16

第 3 章	吸收塔内部附件的选择	19
第 3.1 节	液体分布器的选择	19
第 3.2 节	液体再分布器的选择	20
第 3.3 节	填料支承装置的选择	20
第 3.4 节	填料压紧装置的选择	21
3.4.1	除沫装置	22
第 4 章	吸收塔外部附件的选择	23
第 4.1 节	吸收塔主要接管的尺寸计算	23
4.1.1	气体入口管道尺寸计算	23
4.1.2	液体入口管道尺寸计算	23
第 4.2 节	离心泵的计算与选择	24
第 4.3 节	风机的选取	24
第 5 章	吸收填料塔成本、安全优化	25
第 5.1 节	变量分析	25
5.1.1	变量趋势分析	25
5.1.2	相关性分析方法	26
5.1.3	相关性分析结果	26
第 5.2 节	多目标优化	26
5.2.1	基本假设	26
5.2.2	决策变量	27
5.2.3	优化目标	27
5.2.4	NSGA-II 算法	28
5.2.5	优化结果	28
结论		31
参考文献		33
符号说明		35
附录 A	工艺参数处理代码	37
第 A.1 节	计算工艺参数	37

第 A.2 节	定义优化问题	54
第 A.3 节	选择关键变量	57
第 A.4 节	实现目标优化	62
附录 B	支撑材料文件列表	69

前言

二氧化硫（ SO_2 ）作为一种主要的大气污染物，不仅会导致酸雨的形成，还会对人体健康造成严重危害，诱发呼吸系统疾病、视觉系统疾病等。如今，随着工业化进程的不断加快， SO_2 排放量逐年递增，控制和治理二氧化硫成为环境保护和可持续发展的重要课题。

在化工领域中处理废气的方式多种多样，吸收是多种方式中行之有效的一种，其基本思想是利用混合气体各组分在液体溶剂中溶解度的不同来达到分离混合气体各组分的目的。如何高效地利用吸收操作去除工业废气中的 SO_2 ，是亟待解决的问题之一。

一般来说，完整的吸收过程应包括吸收和解吸两部分。在化工生产过程中，原料气的净化、气体产品的精制、治理有害气体、保护环境等方面都要用到吸收将气体混合物中的各个组分加以分离，其目的是：①回收或捕获气体混合物中的有用物质，以制取有价值的产品；②除去混合气体中的有害成分，使气体净化，以便进一步加工处理；或除去工业放空尾气中的有害物质，以免污染大气。

考虑到 SO_2 具有腐蚀性，若采用板式塔结构 SO_2 会对塔板造成严重的腐蚀影响吸收塔的正常工作，故此次设计应选择填料塔设计。此外，填料塔结构简单、操作弹性大、压降较低、能耗较小、传质效率高、分离效率高，更适合大规模工业应用，治理废气。

二氧化硫吸收填料塔，以水为溶剂，经济合理，净化度高，污染小。由于水和二氧化硫反应生成硫酸，同时也具有相当的利用价值。

第 1 章 吸收塔填料塔的设计方案

第 1.1 节 吸收流程的选择

工业上使用的吸收流程多种多样，可以从不同的角度进行分类。根据所用的吸收剂种类，有仅用一种吸收剂的一步吸收流程和使用两种吸收剂的两步吸收流程；根据所用塔设备数量，可分为单塔吸收流程和多塔吸收流程；根据塔内气液两相的流向，可分为逆流吸收流程、并流吸收流程等基本流程。此外，还有用于特定条件下的部分溶剂循环流程。吸收用塔设备要求使用较小直径的塔设备完成规定的处理量，塔板或填料层阻力小，具有良好的传质性能，操作弹性大，结构简单，造价低，便于安装、操作和维修等。

吸收过程一般具有液气比大的特点，因此更适用填料塔。其优点是阻力小，效率高，有利于过程节能。因此，对于吸收过程来说，填料塔应用较为广泛。填料塔的工艺设计内容是在明确了装置的处理量、操作温度及操作压力及相应的相平衡关系的条件下，完成填料塔工艺尺寸、其他塔内件设计及其他塔外件设计。

1.1.1 单剂和多剂的选择

一般来说，对于组分较为简单、分离要求不是很高的情况，可以优先考虑单剂吸收。而对于需要去除多种杂质或要求较高纯度的情况，则可能需要采用多剂吸收。在实际工程中，需要根据具体情况进行技术经济比较后做出选择。

1.1.2 单塔和多塔的选择

单塔吸收流程是吸收过程中最常用的流程，如过程无特别需要，则一般采用单塔吸收流程。若过程的分离要求较高，使用单塔操作时，所需要的塔体过高，或采用两步吸收流程时，则需要采用多塔流程（通常是双塔吸收流程）。

1.1.3 逆流与并流的选择

吸收塔或再生塔内气液相可以逆流操作也可以并流操作。由于逆流操作具有传质推动力大，分离效率高（具有多个理论级的分离能力）的显著优点而广泛应用。工程上，如无特别需要，一般均采用逆流吸收流程。本设计采用单塔逆流操作。

第 1.2 节 塔内填料的选择

散堆填料简述：目前散堆填料主要有环形填料、鞍形填料、环鞍形填料及球形填料。所用的材质有陶瓷、塑料、石墨、玻璃及金属等。

- ① **拉西环填料：**拉西环填料于 1914 年由拉西（F. Rashching）发明，为外径与高度相等的圆环。拉西环填料的气液分布较差，传质效率低，阻力大，通量小，目前工业上已较少应用。
- ② **鲍尔环填料：**鲍尔环是对拉西环的改进，在拉西环的侧壁上开出两排长方形的窗孔，被切开的环壁的一侧仍与壁面相连，另一侧向环内弯曲，形成内伸的舌叶。鲍尔环由于环壁开孔，大大提高了环内空间及环内表面的利用率，气流阻力小，液体分布均匀。与拉西环相比，鲍尔环的气体通量可增加 50% 以上，传质效率提高 30% 左右，是一种应用较广的填料。
- ③ **阶梯环填料：**阶梯环结构与鲍尔环填料相似，环壁上开有长方形小孔，环内有两层交错 45° 的十字形叶片，环的一端成喇叭口形状的翻边。这样的结构使得阶梯环填料的性能在鲍尔环的基础上又有提高，其生产能力可提高约 10%，压降则可降低 25%。阶梯环一般由塑料和金属制成，性能优于其他侧壁上开孔的填料，应用广泛。
- ④ **矩鞍填料：**矩鞍填料堆积时不会套叠，液体分布较均匀。一般采用瓷质材料制成，其性能优于拉西环。目前，国内绝大多数应用瓷拉西环的场合，均已被瓷矩鞍填料所取代。
- ⑤ **金属环矩鞍填料：**环矩鞍填料兼顾环形和鞍形结构特点而设计出的一种新型填料，一般以金属材质制成，故又称为金属环矩鞍填料。其综合性能优于鲍尔环和阶梯环，在散装填料中应用较多。

填料性能的优劣：通常根据效率、通量及压降三要素衡量。在相同的操作条件下，填料的比表面积越大，气液分布越均匀，表面的润湿性能越好，则传质效率越高；填料的空隙率越大，结构越开敞，则通量越大，压降亦越低。按上述选择标准，同时考虑任务书要求，本设计将使用聚丙烯散装鲍尔环作为填料。选用聚丙烯材质的鲍尔环具有以下优点：

- ① **耐腐蚀性：**适用于 SO_2 吸收过程中的酸性环境；
- ② **质量较轻：**便于安装和更换；

③ **成本较低**：有利于节省资金；

④ **表面光滑**：有利于液体分布。

填料的规格的选择：工业上工业塔常用散装填料主要有 DN16、DN25、DN38、DN50、DN76 等规格。同类填料，尺寸越小，分离效率越高，但阻力增加，通量减小，填料费用也增加很多，因此，需要在分离效率和操作压力降之间找到合适的平衡点。而大尺寸的填料应用于小直径塔中，又会产生液体分布不良及严重的壁流，使塔的分离效率降低。应选择能够减少放大效应的填料，以保证塔内气液两相的均匀分布，选择结构易于拆装和清洁的填料，以减少维护成本和停机时间。

填料采用散装（乱堆）形式，因为散装形式能使气液相对充分接触，而且填料时省时省工。这种方式有利于提高传质效率，同时也简化了安装过程。

参考化工设备机械基础（第二版）413 页，填料公称直径与塔的公称直径的关系，选择 DN25 聚丙烯散装鲍尔环。查化工原理课程设计 148 页表 6-4 可得 $\Phi_F = 550m^{-1}$ ，则根据表格选定，填料直径 $d = 25mm$ ，比表面积 $a_t = 175m^2/m^3$ ， $\phi_F = 550m^{-1}$

表 1-1 聚丙烯散装鲍尔环参数表

产品名称			塑料鲍尔环		
材质			PP/RPP/PVC/CPVC/PVDF/PTFE, PTFE		
使用寿命			> 3 years		
尺寸 (mm)	比表面积 (m^2/m^3)	空隙率 (%)	堆积个数 (pieces/ m^3)	堆积重量 (kg/m^3)	干填料因子 (m^{-1})
16 × 16 × 1.0	188	91	170000	85	275
25 × 25 × 1.2	175	90	53500	69	239
38 × 38 × 1.4	115	89	15800	69	220
50 × 50 × 1.5	93	90	6500	52	127
76 × 76 × 2.6	73.2	92	1927	48	94
100 × 100 × 3.0	52.8	94	1000	48	82

第 1.3 节 吸收剂的选择

吸收剂的性能对吸收操作效果至关重要。选择吸收剂时应重点考虑以下几个方面:

- ① **溶解度**: 吸收剂对溶质组分的溶解度应大, 以提高吸收速率并减少吸收剂用量。
- ② **选择性**: 吸收剂应对目标溶质组分有良好的吸收能力, 而对混合气体中的其他组分不吸收或吸收甚微, 以实现有效分离。
- ③ **挥发度**: 在操作温度下, 吸收剂的蒸气压应低, 以减少吸收和再生过程中的吸收剂挥发损失。
- ④ **粘度**: 在操作温度下, 吸收剂的粘度越低越好, 这有助于提高其在塔内的流动性, 进而提高传质速率和传热速率。
- ⑤ **其他性质**: 理想的吸收剂还应尽可能满足以下要求: 无毒性、无腐蚀性、不易燃易爆、不发泡、冰点低、价格低廉, 易于获得、化学性质稳定

由于在常温 20°C 下, 操作压力 (绝压, 下同) 为 120kPa 下, 水对 SO_2 有较大的溶解度, 有较好的化学稳定性, 有较低的黏度, 且廉价、易得、无毒、不易燃烧, 所以可选择 20°C 清水作为吸收剂。

第 1.4 节 操作条件的选择

- ① **操作温度的确定**: 由吸收过程的气液平衡关系可知, 对于物理吸收过程, 较低的操作温度有利于提高溶质的溶解度, 但过低温度可能会增加操作费用和能耗。同时较高的操作温度虽然会加快化学反应的速率, 但也会降低传质推动力。
- ② **操作压力的确定**: 由吸收过程的汽液平衡关系可知, 提高操作压力可以提高气体在液体中的溶解度, 增加吸收的推动力。但随着操作压力的升高, 对设备的加工制造要求提高, 且能耗增加, 因此需结合具体工艺条件综合考虑, 以确定操作压力。

由此, 在本次 SO_2 吸收设计中, 操作温度为常温即设定为 20°C , 操作压力为 120kPa 。

第 1.5 节 工艺流程图

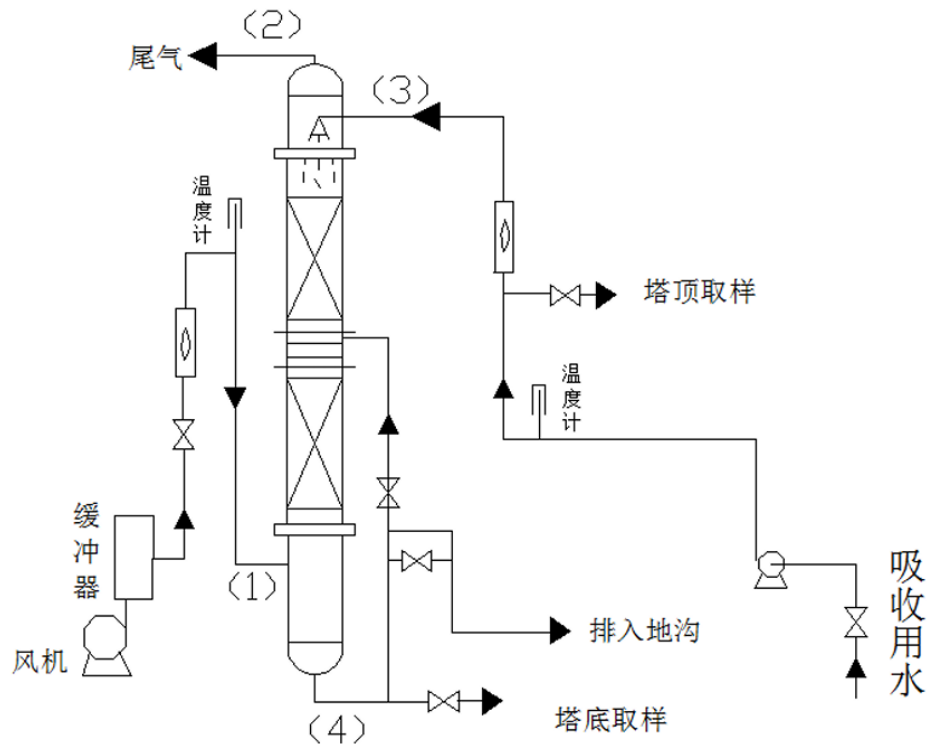


图 1-1 吸收塔工艺流程示意图

第 2 章 吸收填料塔工艺参数计算

按上述基本逻辑，进行吸收填料塔工艺参数的设计^[2]，具体步骤如下。

第 2.1 节 基础物性数据

2.1.1 气相物性数据

对于气相，重要的物性包括混合气体的平均摩尔质量、混合气体的平均密度和混合气体的黏度。而混合气体的黏度由于SO₂含量较低，混合气体黏度可以近似取为空气的黏度。已知SO₂的摩尔质量 M_{SO_2} 为 64.06 g/mol，空气的摩尔质量 M_{Air} 为 28.95 g/mol，理想气体常数 $R = 8.314 \text{ m}^3 \cdot \text{kPa}/(\text{kmol} \cdot \text{K})$ ，则所需数据按理想气体状态方程计算如下：

混合气体的平均摩尔质量：

$$\overline{M_V} = 0.060 \times 64.06 + (1 - 0.060) \times 28.95 = 31.0566 \text{ g/mol} \quad (2-1)$$

混合气体的平均密度为：

$$\rho_V = \frac{P\overline{M_V}}{RT} = \frac{120 \times 31.0566}{8.314 \times (273.15 + 20)} = 1.4291 \text{ kg/m}^3 \quad (2-2)$$

混合气体黏度可近似取为空气黏度。查化工原理上册教材附录 338 页，得 20°C 时空气的黏度：

$$\mu_V = \mu_{\text{Air}} = 1.81 \times 10^{-5} \text{ Pa} \cdot \text{s} = 0.060 \text{ kg}/(\text{m} \cdot \text{h}) \quad (2-3)$$

查化工原理课程设计得 20°C 时SO₂在空气中的扩散系数：

$$D_V = 0.108 \text{ cm}^2/\text{s} = 0.039 \text{ m}^2/\text{h} \quad (2-4)$$

2.1.2 液相物性数据

对于低浓度吸收过程，溶液的物性数据可近似取水的物性数据。重要的液相物性包括密度、黏度、表面张力和SO₂在水中的扩散系数。由化工原理实验教材 127 页水的重要物理性质表得 20°C 时水的有关物性数据如下：

水的密度：

$$\rho_L = 998.2 \text{ kg/m}^3 \quad (2-5)$$

水的黏度：

$$\mu_L = 1.005 \text{ mPa} \cdot \text{s} = 0.001005 \text{ Pa} \cdot \text{s} = 3.618 \text{ kg}/(\text{m} \cdot \text{h}) \quad (2-6)$$

水的表面张力：

$$\sigma_L = 7.28 \times 10^{-2} = 0.0728 \text{ N}/\text{m} \quad (2-7)$$

SO₂ 在水中的扩散系数：

$$D_L = 1.47 \times 10^{-9} \text{ m}^2/\text{s} = 5.292 \times 10^{-6} \text{ m}^2/\text{h} \quad (2-8)$$

2.1.3 气液相平衡数据

在吸收塔设计中，气液相平衡数据是确定传质驱动力的关键。对于SO₂ 在水中的溶解，我们需要考虑亨利系数、相平衡常数和溶解度系数。查教材 82 页表 2-1 得，常压下 20°C 时SO₂ 在水中的相关数据如下：

亨利系数：

$$E = 0.355 \times 10^4 = 3.55 \times 10^3 \text{ kPa} \quad (2-9)$$

相平衡常数：

$$m = \frac{E}{P} = 29.5833 \quad (2-10)$$

溶解度系数：

$$H = \frac{\rho_L}{EM_{\text{SO}_2}} = 0.0156039 \text{ kmol}/(\text{kPa} \cdot \text{m}^3) \quad (2-11)$$

其中， P 为系统压力， ρ_L 为液相密度， M_{SO_2} 为SO₂ 的摩尔质量。

第 2.2 节 物料衡算

根据设计要求得知，进口气体的体积流量 $V' = 1000 \text{ m}^3/\text{h}$ ，二氧化硫的摩尔分数为 $y_1 = 0.060$ ，二氧化硫吸收率 $\eta = 98\%$ 。

进塔气相摩尔比：

$$Y_1 = \frac{y_1}{1 - y_1} = 0.0638298 \quad (2-12)$$

出塔气相摩尔比：

$$Y_2 = Y_1(1 - \eta) = 0.0012766 \quad (2-13)$$

进塔惰性气相流量:

$$V = \frac{V'}{22.4} \times (1 - y_1) \times \frac{273}{273 + 20} = 39.0998 \text{ kmol/h} \quad (2-14)$$

该吸收过程属于低浓度吸收，平衡关系为直线，因此最小液气比为:

$$\left(\frac{L}{V}\right)_{\min} = \frac{Y_1 - Y_2}{Y_1/m - X_2} = 28.9917 \quad (2-15)$$

式中: $X_2 = 0$, L 为液相流量, V 为气相流量。

一般情况下, 吸收剂用量为吸收剂最小用量的 1.1 ~ 2.0 倍, 这里取操作液气比为 1.4 倍最小液气比。

$$\frac{L}{V} = 1.4 \left(\frac{L}{V}\right)_{\min} = 40.5883 \quad (2-16)$$

吸收剂用量:

$$L = 40.5883 \times 39.0998 = 1587 \text{ kmol/h} \quad (2-17)$$

物料衡算:

$$V(Y_1 - Y_2) = L(X_1 - X_2) \quad (2-18)$$

得 $X_1 = 0.00154116$

第 2.3 节 填料塔工艺尺寸计算

相关塔径计算相关逻辑流程图如下:

2.3.1 填料塔塔径的计算

采用 Eckert 通用关联图计算泛点气速。液相质量流量可近似按纯水流量计算, 即液相质量流量为:

$$W_L = 1587 \times 18.02 = 28597.7 \text{ kg/h} \quad (2-19)$$

气相质量流量为:

$$W_V = 1000 \times 1.4291 = 1529.1 \text{ kg/h} \quad (2-20)$$

Eckert 通用关联图的横纵坐标值分别为:

$$X_{Eckert} = \frac{W_L}{W_V} \left(\frac{\rho_V}{\rho_L}\right)^{0.5} = 0.731989 \quad (2-21)$$

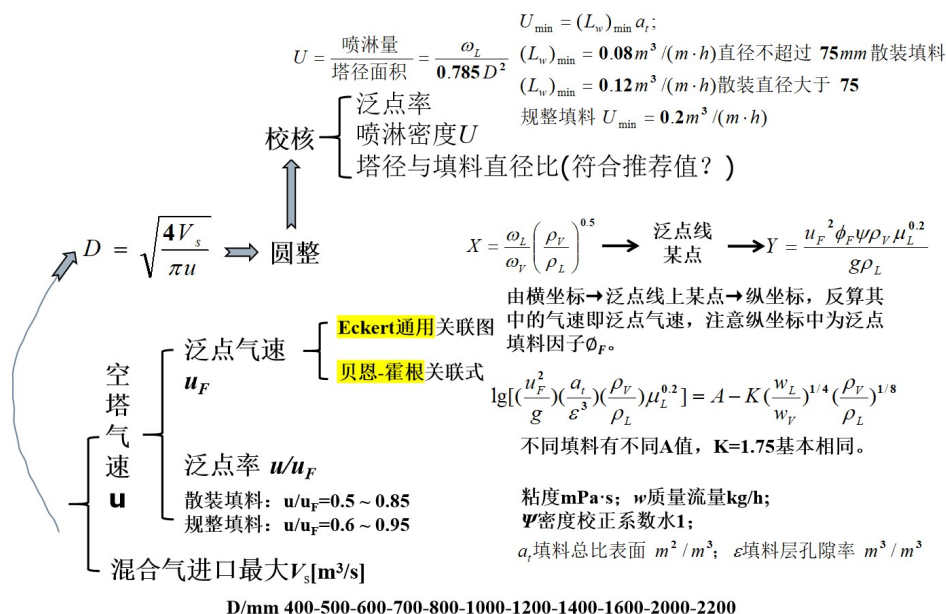


图 2-1 泛点气速-塔径相关逻辑示意图

查 Eckert 通用关联图得:

$$Y_{Eckert} = \frac{u_F^2 \phi \rho_V}{g \rho_L \mu_L^{0.2}} = 0.034 \quad (2-22)$$

式中: u_F 为泛点气速, m/s ; ρ_V 为气相密度, kg/m^3 ; ρ_L 为液相密度, kg/m^3 ; μ_L 为液体黏度, $mPa \cdot s$; ϕ 为实验填料因子, g 为重力加速度, $9.81 m/s^2$; ψ 为水的密度与液体密度之比, 此处为 1。

由化工原理课程设计教材 148 页表 6-4 得 $\phi = 184 m^{-1}$, 则:

$$u_F = \sqrt{\frac{Y_{Eckert} g \rho_L}{\phi \rho_V \mu_L^{0.2}}} = \sqrt{\frac{0.034 g \rho_L}{\phi \rho_V \mu_L^{0.2}}} = 0.928879 m/s \quad (2-23)$$

取 $u = 0.7 u_F = 0.440215 m/s$, 有:

$$D = \sqrt{\frac{4V_s}{\pi u}} = \sqrt{\frac{4 \times 1000/3900}{3.14 \times 0.440215}} = 0.896337 m = 896.337 mm \quad (2-24)$$

圆整塔径, 取 $D = 0.9m$, 也即 $D = 900mm$ 。

2.3.2 填料塔塔径的校核

泛点率校核: 泛点率是塔内实际操作气速与泛点气速的比值, 一般要求在 50% ~ 80% 之间。

$$u = (1000/3900) / \left(\frac{\pi}{4} \times 0.9^2 \right) = 0.436639 m/s \quad (2-25)$$

$$u/u_F = 0.436639/0.928879 = 0.694313 \text{ (在允许范围 } 0.50\text{-}0.85 \text{ 之内)} \quad (2-26)$$

填料规格校核:

当塔径与填料外径之比（即径比）过小时，塔的性能会变得不稳定。因为径比过小时，靠近塔壁的填料层的空隙率会增大且不均匀，塔通过能力虽然会有所提升，但由于气流短路，塔的效率反而会下降。所以需要对填料规格进行校核，以下是常用填料的径比下限。

表 2-1 填料种类及其推荐值

填料种类	D/d 推荐值
拉西环	$D/d \geq 20 \sim 30$
鞍环	$D/d \geq 15$
鲍尔环	$D/d \geq 10 \sim 15$
阶梯环	$D/d > 8$
环矩鞍	$D/d > 8$

进行校核有:

$$D/d = 900/38 = 23.6842 > 10 \quad (2-27)$$

液体喷淋密度校核: 最小润湿速率是指塔的截面上，单位长度的填料周边的最小液体流体体积。直径不超过 75 mm 的散装填料，最小润湿速率为:

$$(L_W)_{\min} = 0.08 \text{ m}^3 / (\text{m} \cdot \text{h}) \quad (2-28)$$

由化工原理教材 325 页的聚丙烯鲍尔环特性数据表得

$$a_t = 175 \text{ m}^2 / \text{m}^3 \quad (2-29)$$

$$U_{\min} = (L_W)_{\min} a_t = 14 \text{ m}^3 / (\text{m}^2 \cdot \text{h}) \quad (2-30)$$

$$U = \frac{28597.7/998.2}{\frac{\pi}{4} \times 0.9^2} = 45.0338 \text{ m}^3 / (\text{m}^2 \cdot \text{h}) > U_{\min} \quad (2-31)$$

故满足最小喷淋密度的要求。经以上校核可知，填料塔直径 $D = 900 \text{ mm}$ 合理。

2.3.3 传质单元数的计算

脱析因数 S 计算如下：

$$S = \frac{mV}{L} = \frac{29.5833}{40.5883} = 0.728863 \quad (2-32)$$

气相总传质单元数 N_{OG} 的计算为：

$$\begin{aligned} N_{OG} &= \frac{1}{1-S} \ln \left[(1-S) \frac{Y_1 - Y_2^*}{Y_2 - Y_2^*} + S \right] \\ &= \frac{1}{1-0.728863} \ln \left[(1-0.728863) \frac{0.0582-0}{0.002328-0} + 0.728863 \right] \\ &= 9.80781 \end{aligned} \quad (2-33)$$

2.3.4 膜吸收系数的计算

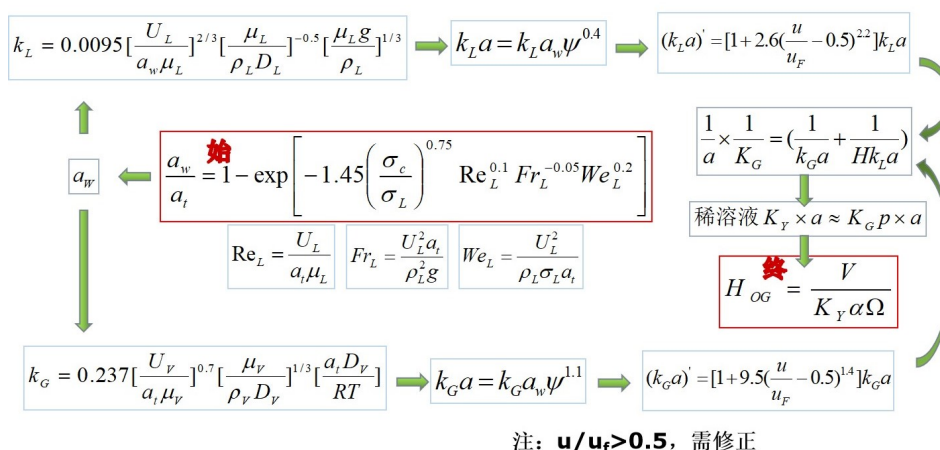


图 2-2 由膜吸收系数至传质单元高度的流程示意图

气相总传质单元高度采用修正的 Onda 关联式：

$$\frac{a_w}{a_t} = 1 - e^{-1.45 \left(\frac{\sigma_c}{\sigma_w} \right)^{0.75} \left(\frac{U_L}{a_t \mu_L} \right)^{0.1} \left(\frac{U_L \mu_L}{\rho_L g} \right)^{-0.05} \left(\frac{U_L^2}{\rho_L \sigma_L a_t} \right)^{-0.2}} \quad (2-34)$$

由化工原理课程设计教材中 175 页表 6-6 得 $\sigma_c = 427680 \text{ kg/h}^2$

液体质量通量：

$$U_L = \frac{28597.7}{\frac{\pi}{4} \times 0.9^2} = 44952.7 \text{ kg}/(\text{m}^2 \cdot \text{h}) \quad (2-35)$$

代入 Onda 关联式：

$$\frac{a_w}{a_t} = 0.443873 \quad (2-36)$$

所以:

$$a_w = 0.443873 \times 175 = 77.6778 \text{ m}^2/\text{m}^3 \quad (2-37)$$

气体质量通量:

$$U_V = \frac{1529.1}{\frac{\pi}{4} \times 0.9^2} = 2403.59 \text{ kg}/(\text{m}^2 \cdot \text{h}) \quad (2-38)$$

计算气侧吸收系数:

$$\begin{aligned} k_G &= 0.237 \left(\frac{U_V}{a_t \mu_V} \right)^{0.7} \left(\frac{\mu_V}{\rho_V D_V} \right)^{1/3} \left(\frac{a_t D_V}{RT} \right) \\ &= 0.0288785 \text{ kmol}/(\text{m}^2 \cdot \text{h} \cdot \text{kPa}) \end{aligned} \quad (2-39)$$

计算液膜吸收系数:

$$\begin{aligned} k_L &= 0.0095 \left(\frac{U_L}{a_w \mu_L} \right)^{2/3} \left(\frac{\mu_L}{\rho_L D_L} \right)^{-1/2} \left(\frac{\mu_L g}{\rho_L} \right)^{1/3} \\ &= 0.826184 \text{ m}/\text{h} \end{aligned} \quad (2-40)$$

2.3.5 总传质系数的计算

体积吸收系数:

由化工原理课程设计教材 175 页表 6-7 得参数 $\psi = 1.45$, 计算 $k_G a$ 和 $k_L a$ 如下:

$$\begin{aligned} k_G a &= k_G a_w \psi^{1.1} \\ &= 0.0288785 \times 91.0681 \times 1.45^{1.1} \\ &= 3.3758 \text{ kmol}/(\text{m}^3 \cdot \text{h} \cdot \text{kPa}) \end{aligned} \quad (2-41)$$

$$\begin{aligned} k_L a &= k_L a_w \psi^{0.4} \\ &= 0.826184 \times 91.0681 \times 1.45^{0.4} \\ &= 74.4596 \text{ kmol}/(\text{m}^3 \cdot \text{h} \cdot \text{kPa}) \end{aligned} \quad (2-42)$$

修正体积吸收系数:

由于 $u/u_F = 0.694313 > 0.5$, 使用化工原理课程设计教材 152 页修正公式 (6-12) 与 (6-13) 进行修正:

$$\begin{aligned} k'_G a &= \left[1 + 9.5 \left(\frac{u}{u_F} - 0.5 \right)^{1.4} \right] k_G a \\ &= 6.61176 \text{ kmol}/(\text{m}^3 \cdot \text{h} \cdot \text{kPa}) \end{aligned} \quad (2-43)$$

$$k'_L a = \left[1 + 2.6 \left(\frac{u}{u_F} - 0.5 \right)^{2.2} \right] k_L a \quad (2-44)$$

$$= 79.727 \text{ kmol}/(\text{m}^3 \cdot \text{h} \cdot \text{kPa})$$

总体积传质系数:

$$K_G a = \frac{1}{\frac{1}{k'_G a} + \frac{1}{H k'_L a}} = 1.04705 \text{ kmol}/(\text{m}^3 \cdot \text{h} \cdot \text{kPa}) \quad (2-45)$$

2.3.6 填料层高度的计算

计算填料层高度 H_{OG} 和总高度 Z :

$$H_{OG} = \frac{V}{K_G a p \Omega} = 0.489162 \text{ m} \quad (2-46)$$

$$Z = H_{OG} \cdot N_{OG} = 4.79761 \text{ m} \quad (2-47)$$

$$Z' = 1.25Z = 5.99701 \text{ m} \quad (2-48)$$

设计取填料层高度为 $Z' = 4.1 \text{ m} < h_{\max} = 6 \text{ m}$, 故不进行分段。

2.3.7 填料层压降的计算

填料层压降直接影响到填料塔的能效和操作性能。压降的大小决定了塔内气体的流动状态, 过高的压降会增加泵送能耗, 降低塔的处理能力和操作灵活性。而过低的压降可能意味着填料层的传质效率不足, 影响塔的分离效率。

在设计阶段, 通过计算填料层压降, 可以选择合适的填料类型和尺寸, 以及确定塔的操作参数, 确保塔在经济和技术上的可行性。在操作阶段, 监控和调整填料层压降有助于维持塔内气液两相的理想流动状态, 避免出现液泛等不稳定现象, 从而保证塔的稳定运行和提高产品质量。因此, 填料层压降的计算是填料吸收塔设计和优化中的一个基本步骤, 对于实现高效、节能和安全的化工生产具有重要意义。

采用埃克特通用关联图计算填料层压降, Eckert 通用关联图的横坐标值为:

$$X_{Eckert} = \frac{W_L}{W_V} \left(\frac{\rho_V}{\rho_L} \right)^{0.5} = 0.731989 \quad (2-49)$$

查化工原理课程设计 154 页表 6-11^[3] 可得 $\phi_p = 232 \text{ m}^{-1}$, 则纵坐标为:

$$Y_{Eckert} = \frac{u_F^2 \phi_p \varphi \rho_V}{g \rho_L \mu_L^{0.2}} = 0.00893255 \quad (2-50)$$

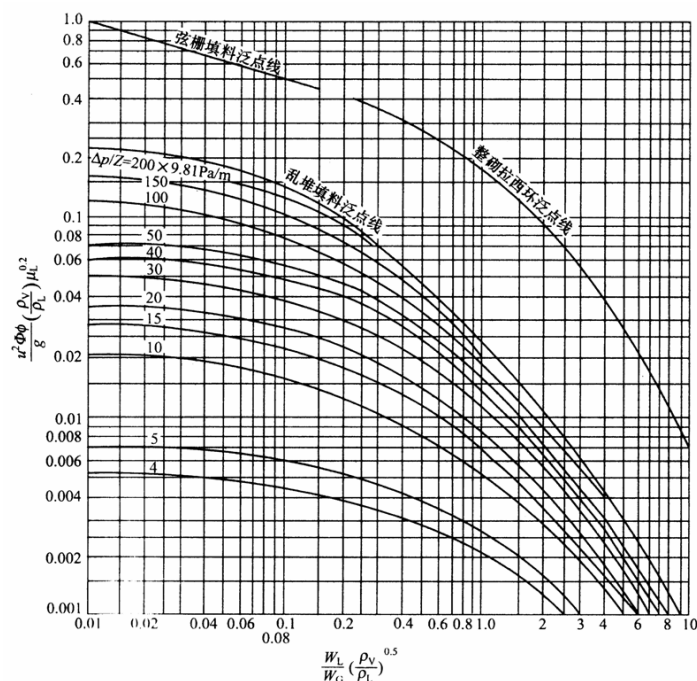


图 2-3 Eckert 通用关联图

式中： u_F 为泛点气速，m/s； ρ_V 为气相密度，kg/m³； ρ_L 为液相密度，kg/m³； μ_L 为液体黏度，mPa·s； ϕ_p 为填料层压降的实验填料因子， g 为重力加速度，9.81 m/s²； φ 为水的密度与液体密度之比，此处为 1。

查化工原理课程设计 148 页图 6-2，得单位填料层压降：

$$\Delta p/Z = 9 \times 9.81 = 88.29 \text{ Pa/m} \quad (2-51)$$

则填料层压降：

$$\Delta p = 88.29 \times 5.7 \text{ Pa} = 503.253 \text{ Pa} \quad (2-52)$$

2.3.8 塔附属高度的计算

塔的附属高度包括上部空间、液体分布器空间和底部空间。根据设计，塔的上部空间取 1.0 米，底部空间取 1.1 米，液体分布器高度为 0.2 米，因此塔的附属高度为：

$$H_{\text{附属}} = 1.0 + 1.1 + 0.2 = 2.3 \text{ m} \quad (2-53)$$

第 2.4 节 吸收塔工艺尺寸总结

吸收塔工艺计算的目的是确保设备能够在实际操作中高效、稳定地进行传质操作，并满足设计要求。通过严密的气液传质模型和物料衡算，我们能够设计出适合特定工艺条件的吸收塔。

首先，在物性数据的获取与应用中，通过对气液相物理性质如密度、黏度、扩散系数等的准确估算，保证了后续计算的可靠性。特别是亨利系数等气液相平衡数据的获取，为传质驱动力的计算提供了坚实的基础。

在物料衡算中，基于已知的气体流量、二氧化硫的浓度及吸收率，计算出进出塔的气相摩尔比、惰性气体流量及最小液气比，并确定了实际操作中的液气比。通过这些数据，可以进一步推导出吸收剂的用量，确保吸收操作能够实现高效的二氧化硫去除。

随后，在填料塔的尺寸设计中，利用 Eckert 通用关联图和相关实验参数，计算出泛点气速、塔径及填料层高度，并进行了校核。通过对泛点率、填料规格及液体喷淋密度的核查，确保塔的设计既能满足工艺需求，又不会出现气流短路或液泛等不良现象。此外，对填料层压降的计算也表明塔内流体流动阻力在合理范围内，保证了塔的能耗在可接受的范围。

在传质计算中，应用膜吸收系数、Onda 关联式以及修正体积吸收系数，确保了传质效率的准确估计。通过计算气侧和液侧的传质单元数及传质单元高度，得出合适的填料层高度，保证气液传质能够在塔内有效进行。

最后，通过附属高度的设计，包括塔的上部空间、液体分布器及底部空间的合理布置，确保塔的整体结构稳定性和操作便利性。在附属设备的设计中，液体分布器和再分布器的合理配置，能够有效改善塔内流体的分布，减少局部液体聚集或气流短路的情况，从而进一步提高塔的效率。

总而言之，吸收塔的工艺计算不仅包括物料衡算、传质单元和填料层高度等核心内容，还涉及到结构设计、流体力学参数和操作稳定性的校核。通过这一系列精确的计算与校核，吸收塔的设计可以确保其在实际工业操作中具有高效的传质性能和良好的操作稳定性，从而实现高效的二氧化硫吸收和废气处理目标。

表 2-2 参数汇总表

描述信息	参数符号	参数值	单位
摩尔质量	M_V	31.0566	g/mol
气体密度	ρ_V	1.4291	kg/m ³
气体扩散系数	D_V	1.08×10^{-5}	m ² /s
气体粘度	μ_V	1.81×10^{-5}	Pa · s
亨利常数	E	3550	kPa
气体分子质量比	m	29.5833	-
液体密度	ρ_L	998.2	kg/m ³
液体粘度	μ_L	1.005×10^{-3}	Pa · s
表面张力	σ_L	0.0728	N/m
液体扩散系数	D_L	1.47×10^{-9}	m ² /s
溶解度系数	H	0.0156039	kmol/(kPa · m ³)
进料气体组分	Y_1	0.0638298	-
出料气体组分	Y_2	0.0012766	-
气体流量	V	39.0998	kmol/h
液气比	$(\frac{L}{V})_{min}$	28.9917	-
最小液气比	$(\frac{L}{V})_{min_ratio}$	1.4	-
实际液气比	$\frac{L}{V}$	40.5883	-
液体流量	L	1587	kmol/h
进料液体组分	X_1	0.00154116	-
液体质量流量	W_L	28597.7	kg/h
气体质量流量	W_V	1529.1	kg/h
泛点气速 Eckert 坐标 X 值	X_{Eckert, ϕ_F}	0.731989	-
泛点气速 Eckert 坐标 Y 值	Y_{Eckert, ϕ_F}	0.034	-
泛点气速	u_F	0.928879	m/s
设计气速	u_{design}	0.440215	m/s
未圆整塔径	D	0.896337	m
圆整后塔径	$D_{rounded}$	0.9	-
实际操作气速	u_{actual}	0.436639	m/s
泛点率	$flooding\ ratio$	0.994313	-
圆整后塔径与填料直径比	$\frac{D_{rounded}}{d}$	23.6842	-

——接下页

描述信息	参数符号	参数值	单位
最小喷淋密度	U_{min}	14	$\text{m}^3/(\text{m}^2 \cdot \text{h})$
实际喷淋密度	U	45.0338	$\text{m}^3/(\text{m}^2 \cdot \text{h})$
脱吸因数	S	0.728863	m^2
气相总传质单元数	N_{OG}	9.80781	-
液相质量通量	U_L	44952.7	$\text{kg}/(\text{m}^2 \cdot \text{h})$
有效比表面积比	$\frac{a_w}{a_t}$	0.443873	-
有效比表面积	a_w	77.6778	m^2/m^3
气相质量通量	U_V	2403.59	$\text{kg}/(\text{m}^2 \cdot \text{h})$
气膜吸收系数	k_G	0.0288785	$\text{kmol}/(\text{m}^2 \cdot \text{h} \cdot \text{kPa})$
液膜吸收系数	k_L	0.826184	m/h
气膜体积吸收系数	$k_G a$	3.3758	$\text{kmol}/(\text{m}^2 \cdot \text{h} \cdot \text{kPa})$
液膜体积吸收系数	$k_L a$	74.4596	m/h
修正后的气膜体积吸收系数	$k'_G a$	6.61176	$\text{kmol}/(\text{m}^2 \cdot \text{h} \cdot \text{kPa})$
修正后的液膜体积吸收系数	$k'_L a$	79.727	m/h
总传质系数	$K_G a$	1.04705	$\text{kmol}/(\text{m}^2 \cdot \text{h} \cdot \text{kPa})$
填料层高度	H_{OG}	0.489162	m
总高度	Z	4.79761	m
设计高度	Z'	5.99701	m
填料层压降 Eckert 坐标 X 值	X_{Eckert, ϕ_p}	0.731989	-
填料层压降 Eckert 坐标 Y 值	Y_{Eckert, ϕ_p}	0.00893255	-

第3章 吸收塔内部附件的选择

填料塔内件是填料塔的重要组成部分，它们的设计和选择对填料塔的性能有着直接的影响。填料塔内件的设计需要考虑多种因素，包括塔的直径、操作压力、处理的物料性质、填料的类型和特性等。设计时还需要确保内件能够提供足够的强度和刚度，以承受操作中的压力波动、机械震动和温度变化。此外，内件的设计应促进气液的均匀分布，以提高传质效率和塔的整体性能。

第3.1节 液体分布器的选择

液体分布器是填料塔中的关键内件，它的主要作用是将液体均匀地分布在填料层的顶部，或者在塔内的某一高度上进行再分布。液体分布器的性能直接影响填料的传质效率和操作弹性。合理的液体分布可以减少填料塔内的放大效应和端效应，提高塔的整体效率，减少塔的尺寸和造价。

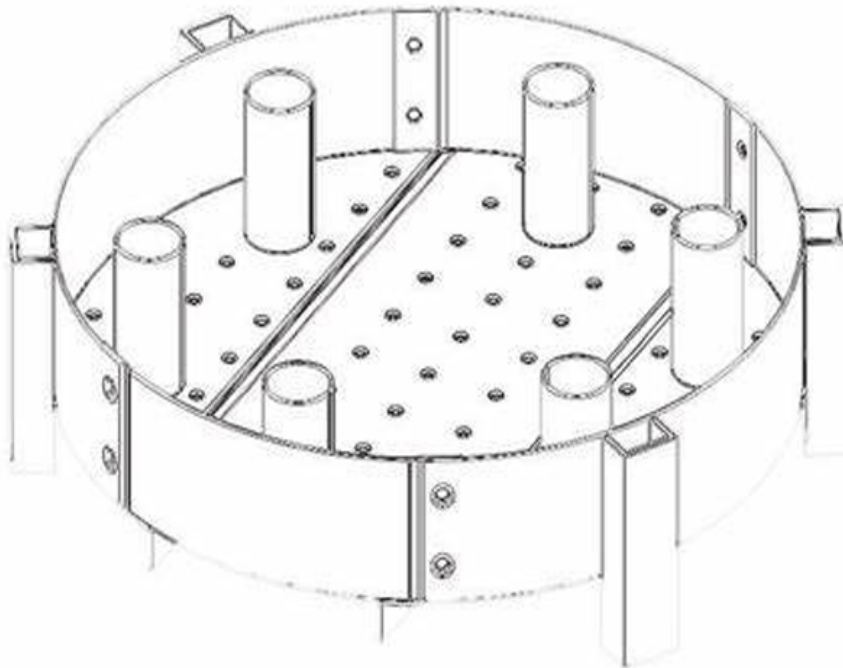
液体分布器的工作原理是通过其设计的结构，将进入塔内的液体以均匀的方式分配到填料层上，确保液体能够在填料表面形成均匀的膜状流动，从而实现有效的气液接触和传质。设计要求包括液体分布均匀、操作弹性大、结构紧凑、气体流通面积大、具备液体收集、分布及气体分布功能等。

液体分布器的类型多样，可以根据其流体动力特性、形状、液体离开的方式、分布的次数以及组合方式进行分类。例如，按分布器流体动力分有重力型和压力型，按形状分有管式、槽式、盘式等，按液体离开形式分有孔流型和溢流型。液体分布器的选择依据主要包括分布质量、操作弹性、处理量、气体阻力、对水平度等多个方面。

为了确保液体分布器的正常运行和填料塔的高效操作，需要定期检查分布器的工作状态，包括液体分布的均匀性和压降。在维护时，应注意检查是否存在堵塞或磨损，并及时进行清理或更换。优化建议可能包括调整液体分布器的设计以适应特定的操作条件，或者采用新型抗堵塞的设计来提高塔内的操作弹性和减少维护工作量。

槽式液体分布器具有较大的操作弹性和极好的抗污堵性，特别适合于大气液负荷，应用范围非常广泛。所以本次SO₂设计中选用槽式液体分布器。

图 3-1 槽式液体分布器示意图



第 3.2 节 液体再分布器的选择

再分布器采用升气管式设计，适合大直径塔体。除塔顶液体的分布之外，填料层中的液体的再分布是填料塔中的一个重要问题。往往会发现，在离填料顶面一定距离处，喷淋的液体便开始向塔壁偏流，然后雁塔壁下流，塔中心处填料得不到好的湿润，形成所谓“干椎体”的不正常现象，减少了气液两相的有效接触面积。因此每隔一定距离必须设置液体再分布器，以克服此种现象。升气管式再分布器适用于直径 0.9 m 以上的塔，而且可以分段卸下填料，更换填料方便，所以本设计选用升气管式再分布器

第 3.3 节 填料支承装置的选择

填料支承装置的作用是支承填料以及填料层内液体的质量，同时保证气液两相顺利通过。支承若设计不当，填料塔的液泛可能首先发生在支承板上。为使气体能顺利

通过，对于普通填料塔，支承件上的流体通过的自由截面积为填料面的 50% 以上，且应大于填料的空隙率。此外，应考虑到装上填料后要将支承板上的截面堵去一些，所以设计时应取尽可能大的自由截面。自由截面太小，在操作中会产生拦液现象。增加压强降，降低效率，甚至形成液泛。由于填料支承装置本身对塔内气液的流动状态也会产生影响，因此作为填料支承装置，除考虑其对流体流动的影响外，一般情况下填料支承装置应满足如下要求：

足够的强度和刚度，以支持填料及所持液体的质量（持液量），并考虑填料空隙中的持液量，以及可能加于系统的压力波动，机械震动，温度波动等因素。

足够的开孔率（一般要大于填料的空隙率），以防止首先在支撑处发生液泛；为使气体能顺利通过，对于普通填料塔，支承件上的流体通过的自由截面积为填料面的 50% 以上，且应大于填料的空隙率。此外，应考虑到装上填料后要将支承板上的截面堵去一些，所以设计时应取尽可能大的自由截面。自由截面太小，在操作中会产生拦液现象。增加压强降，降低效率，甚至形成液泛。

结构上应有利于气液相的均匀分布，同时不至于产生较大的阻力（一般阻力不大于 20Pa）。本设计运用的瓷质鲍尔环，孔隙率相对较大，升气管式支撑板能更好的克服支撑板的强度和自由截面之间的矛盾，耗能更好的适应高空隙率填料的要求，本设计选用升气管式支撑板。

第 3.4 节 填料压紧装置的选择

填料支撑是填料塔内部的一个重要组成部分，它的主要作用是支撑填料层，确保填料在塔内均匀分布，防止填料在操作过程中发生下沉或移动，从而维持填料层的结构稳定性和传质效率。填料支撑还能够承受塔内的压力和机械负荷，保护塔体结构不受损害。

填料支撑的类型包括栅板式支撑、压板式支撑和驼峰支撑等。栅板式支撑通常由金属材料制成，具有一定的空隙率，可以放置在塔的支承圈上。压板式支撑则是制成栅板形状的填料压紧器，依靠自身重力将填料压紧。驼峰支撑则是通过冲压、切割等方式制成的立体结构，具有良好的力学性能和流体力学性能。

驼峰型支撑装置具有高刚性与载荷能力、优良流体分布性、节省材料，降低塔体重量，提升操作弹性和处理能力等优点。所以本次 SO₂ 吸收设计中选用驼峰型支撑装置。

压紧装置的作用是在填料塔操作过程中，通过施加压力来确保填料层的紧密性和稳定性。压紧装置通常安装在填料层的上方，可以是简单的重力式压紧，也可以是带有弹簧或其他机械装置的压紧系统。压紧装置的设计要能够适应不同操作条件下的压力变化，确保填料层在高压降或负荷波动时不会松动或破损。

为了确保填料支撑和压紧装置的正常运行，需要定期检查其结构的完整性和压紧效果。在维护时，应检查支撑结构是否有变形、腐蚀或损坏，并及时进行修复或更换。优化建议可能包括使用高质量的材料来提高支撑和压紧装置的耐久性，以及根据操作条件调整压紧力度，以达到最佳的传质效果和塔的运行效率。

填料压紧网板的设计要求有足够的空隙率，通常大于 70%，可以减少流体阻力并促进气液的均匀分布和防止填料从间隙中漏出。所以本次 SO₂ 吸收设计采用填料压紧网板。

3.4.1 除沫装置

除沫装置在化工、石油等行业的塔器设备中起着至关重要的作用。它的主要功能是从气体流中去除夹带的液滴，以提高塔内的传质效率，减少物料损失，保护下游设备如压缩机不受液沫侵蚀，延长设备寿命，并减少环境污染。在气体吸收过程中，除沫装置还能保证气体的纯度，使后续过程能正常运行。

除沫装置的类型包括丝网除沫器、折流板除沫器和旋流板除沫器等。丝网除沫器利用丝网的比表面积大、空隙率高的特点，通过气体与丝网的碰撞和惯性作用来分离液滴。折流板除沫器和旋流板除沫器则利用气流在板上的旋转运动产生的离心力来分离液滴。这些装置通常安装在塔的顶部或其他需要气液分离的位置。

除沫装置的设计要点包括高效率的除沫、小的压力降、抗堵塞能力以及结构的简单性。优化方法可能包括改进丝网的材料和结构，以提高除沫效率和抗堵塞能力，或者优化旋流板的设计以提高分离效率和降低阻力。

抽屉式丝网除沫器是一种经济型除沫装置，它采用标准的丝网除沫元件，可以在塔体外更换，便于操作和维护。这种除沫器的特点是阻力低，可反复清洗，经济性高，且结构简单，投资少。此外，这种除沫器还具有较高的除沫效率和较小的压力降，有利于提高设备的生产效率。所以本次 SO₂ 吸收设计选用抽屉式丝网除沫器。

第 4 章 吸收塔外部附件的选择

第 4.1 节 吸收塔主要接管的尺寸计算

在吸收塔的设计中，接管的尺寸对气体和液体的传输效率有着直接影响。接管的尺寸取决于流体的流量、流速和系统压力损失等因素。

4.1.1 气体入口管道尺寸计算

根据表中数据，气体流量为 $V = 39.0998 \text{ kmol/h}$ ，实际操作气速为 $u_{actual} = 0.436639 \text{ m/s}$ ，气体密度为 $\rho_V = 1.5291 \text{ kg/m}^3$ 。计算气体体积流量：

$$V_{\text{体积}} = \frac{V \times M_V}{\rho_V} = \frac{39.0998 \times 31.0566}{1.5291} \approx 795.5 \text{ m}^3/\text{h} \quad (4-1)$$

转换为 m^3/s ：

$$V_{\text{体积},s} = \frac{795.5}{3600} \approx 0.221 \text{ m}^3/\text{s} \quad (4-2)$$

根据流量公式，气体入口管道直径 $D_{\text{气体进口}}$ 计算如下：

$$D_{\text{气体进口}} = \sqrt{\frac{4V_{\text{体积},s}}{\pi u_{actual}}} = \sqrt{\frac{4 \times 0.221}{\pi \times 0.436639}} \approx 0.448 \text{ m} \quad (4-3)$$

4.1.2 液体入口管道尺寸计算

根据表中数据，液体流量为 $L = 1587 \text{ kmol/h}$ ，液体密度为 $\rho_L = 998.2 \text{ kg/m}^3$ 。计算液体体积流量：

$$L_{\text{体积}} = \frac{L \times M_V}{\rho_L} = \frac{1587 \times 31.0566}{998.2} \approx 49.37 \text{ m}^3/\text{h} \quad (4-4)$$

转换为 m^3/s ：

$$L_{\text{体积},s} = \frac{49.37}{3600} \approx 0.0137 \text{ m}^3/\text{s} \quad (4-5)$$

假设操作液体流速为 $u_L = 1 \text{ m/s}$ （从表中未给出，使用通常的工业应用速度），则液体入口管道直径 $D_{\text{液体进口}}$ 计算如下：

$$D_{\text{液体进口}} = \sqrt{\frac{4L_{\text{体积},s}}{\pi u_L}} = \sqrt{\frac{4 \times 0.0137}{\pi \times 1}} \approx 0.132 \text{ m} \quad (4-6)$$

第 4.2 节 离心泵的计算与选择

离心泵的选型主要基于吸收塔液体的流量和所需扬程。根据表中数据，液体流量为 $L = 28597.7 \text{ kg/h}$ ，液体密度为 $\rho_L = 998.2 \text{ kg/m}^3$ 。计算液体体积流量：

$$L_{\text{体积}} = \frac{L}{\rho_L} = \frac{28597.7}{998.2} \approx 28.64 \text{ m}^3/\text{h} \quad (4-7)$$

扬程计算假设所需扬程为 $H = 30 \text{ m}$ ，泵的效率假设为 $\eta_p = 0.75$ 。离心泵的功率 P 可通过以下公式计算：

$$P = \frac{\rho_L g L_{\text{体积}} H}{\eta_p} = \frac{998.2 \times 9.81 \times 28.64 \times 30}{0.75} \approx 107374 \text{ W} \approx 107.374 \text{ kW} \quad (4-8)$$

第 4.3 节 风机的选取

根据表中数据，气体体积流量为 $0.221 \text{ m}^3/\text{s}$ 。风机的选取需要确保能够处理该流量。假设风机的效率为 $\eta_f = 0.90$ ，系统的总压降为 $\Delta P = 2000 \text{ Pa}$ （从表中未给出，使用估计值）。风机的功率 P_f 计算如下：

$$P_f = \frac{V_{\text{体积},s} \Delta P}{\eta_f} = \frac{0.221 \times 2000}{0.90} \approx 490.222 \text{ W} \approx 0.490 \text{ kW} \quad (4-9)$$

第 5 章 吸收填料塔成本、安全优化

现考虑吸收填料塔设计的优化，因吸收塔内部、外部附件的选择取决于吸收塔工艺参数，当吸收填料塔任务目标及外界条件确定时，则整个填料塔对象各项参数确定。也就是说，选定的气体的流量、气体中 SO_2 的摩尔分数、 SO_2 的吸收率、操作温度、操作压力、填料总比表面积、填料直径、单个填料吸收塔高度限制、实际液气比取最小液气比的比率一定时，填料塔的工艺参数即可确定，所需成本价格、安全因素即确定。

第 5.1 节 变量分析

5.1.1 变量趋势分析

为了更直观地了解各个变量对成本因素和安全因素的影响，我们绘制了变量趋势图。图5-1展示了 9 个变量 (即对象初始化的 9 个决策变量) 分别对成本因素和安全因素的影响趋势。

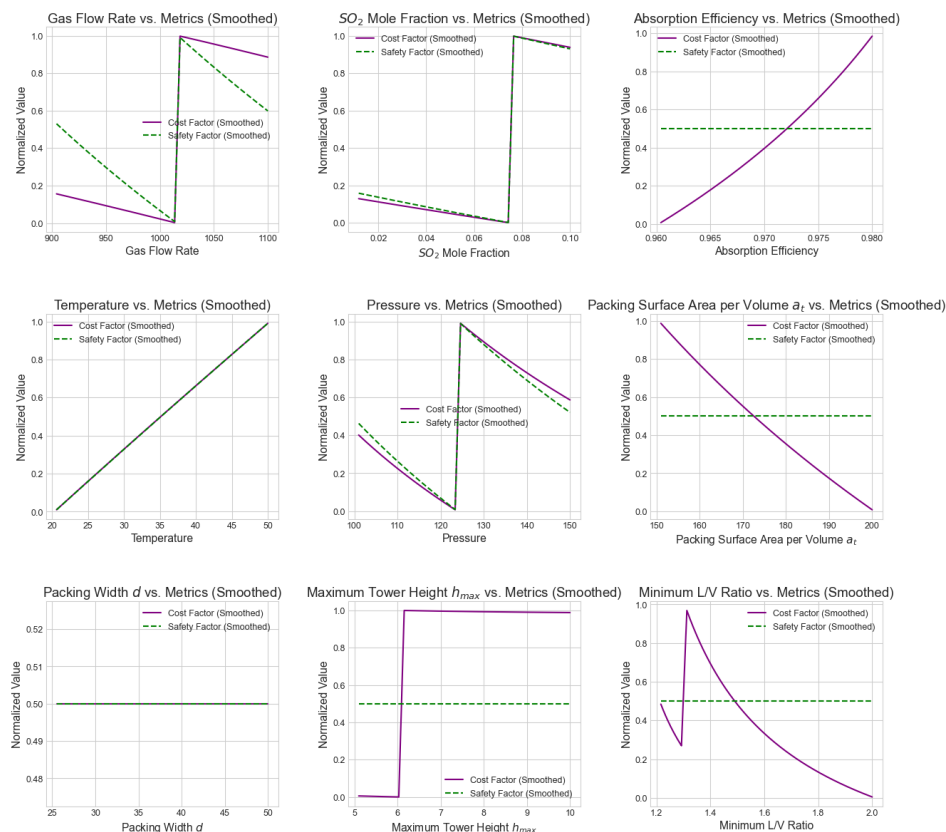


图 5-1 决策变量对成本因素和安全因素的影响趋势

从图中可以观察到：

- 气体流量的增加会导致成本因素和安全因素同时上升。
- 气相中二氧化硫含量的增加会导致成本因素下降，但安全因素也会降低。
- 二氧化硫吸收率的提高会增加成本因素，但同时也会提高安全因素。

这些趋势分析为我们后续的多目标优化提供了重要的参考。

5.1.2 相关性分析方法

为了确定哪些变量对成本因素和安全因素影响最大，我们进行了相关性分析。具体步骤如下：

- ① 对每个变量，在其可能的取值范围内取 200 个均匀分布的点。
- ② 固定其他变量，仅改变当前变量的值。
- ③ 计算每个点对应的成本因素和安全因素。
- ④ 使用 Pearson 相关系数计算变量与成本因素、安全因素的相关性。

相关性系数的计算公式如下：

$$r = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (5-1)$$

其中， x_i 和 y_i 分别表示变量和因素（成本或安全）的值， \bar{x} 和 \bar{y} 表示它们的平均值。

5.1.3 相关性分析结果

通过相关性分析，我们得到了每个变量与成本因素和安全因素的相关系数。结果如表 5-1 所示。

根据相关性分析结果，可选择相关性最高的几个变量作为关键变量进行后续优化。

第 5.2 节 多目标优化

5.2.1 基本假设

- ① 吸收填料塔的成本因素仅与塔径和塔高有关。
- ② 填料的成本与填料塔的面积成正比。
- ③ 不考虑塔内附件和塔外附件对成本、安全的影响，因填料塔工艺参数及填料确定内外附属部件可由此确定，所需成本也随即确定。

表 5-1 变量与成本因素和安全因素的相关系数

变量	成本因素相关性	安全因素相关性
气体流量	0.8036	0.5080
气相中二氧化硫含量	0.7276	0.7137
二氧化硫吸收率	0.9951	-0.0000
操作温度	1.0000	1.0000
操作压力	0.6063	0.5222
填料总比表面积	-0.9981	0.0000
填料直径	nan	-0.0000
单个填料吸收塔最大高度	-0.7055	0.0000
取最小液气比的比例常数	-0.6191	-0.0000

5.2.2 决策变量

现对如下九个变量关于成本因素、安全因素进行相关性分析和趋势性分析。

- ① 气体流量, V
- ② 气相中二氧化硫含量, Y_1
- ③ 二氧化硫吸收率, η
- ④ 操作温度, T
- ⑤ 操作压力, P
- ⑥ 填料总比表面积, a_t
- ⑦ 填料直径, d
- ⑧ 单个填料吸收塔最大高度, h_{max}
- ⑨ 取最小液气比的比例常数, $(\frac{L}{V})_{min ratio}$

5.2.3 优化目标

本章节优化目标为:

- ① 最小化吸收填料塔的成本因素, *Cost Factor*
- ② 最大化吸收填料塔的安全因素, *Safety Factor*

这两个目标通常是相互矛盾的, 因为提高安全性通常会增加成本。因此, 我们需要在这两个目标之间寻找一个平衡点。

基于前面的分析, 我们将吸收填料塔的优化问题定义为一个多目标优化问题:

$$\begin{aligned}
& \text{minimize } f_1(\mathbf{x}) \quad (\text{成本因素}) \\
& \text{maximize } f_2(\mathbf{x}) \quad (\text{安全因素}) \\
& \text{subject to } \mathbf{x}_l \leq \mathbf{x} \leq \mathbf{x}_u
\end{aligned} \tag{5-2}$$

其中， \mathbf{x} 是决策变量向量， \mathbf{x}_l 和 \mathbf{x}_u 分别是决策变量的下界和上界。

5.2.4 NSGA-II 算法

为了解决这个多目标优化问题，我们选择使用 NSGA-II（Non-dominated Sorting Genetic Algorithm II）算法。NSGA-II 是一种高效的多目标优化算法，它具有以下特点：

- 使用快速非支配排序方法
- 采用拥挤度距离来保持解的多样性
- 精英策略，确保最优解不会在进化过程中丢失

NSGA-II 的主要步骤如下：

- ① 初始化种群
- ② 对种群进行非支配排序
- ③ 计算拥挤度距离
- ④ 选择、交叉和变异操作生成子代
- ⑤ 合并父代和子代，进行精英选择
- ⑥ 重复步骤 2-5，直到达到终止条件

5.2.5 优化结果

通过运行 NSGA-II 算法，我们得到了一系列非支配解，即帕累托前沿。图5-2展示了优化结果的帕累托前沿。

从帕累托前沿可以看出，成本因素和安全因素之间存在明显的权衡关系。决策者可以根据具体需求，从帕累托前沿上选择合适的解。

表5-2列出了帕累托前沿上的一些代表性解。

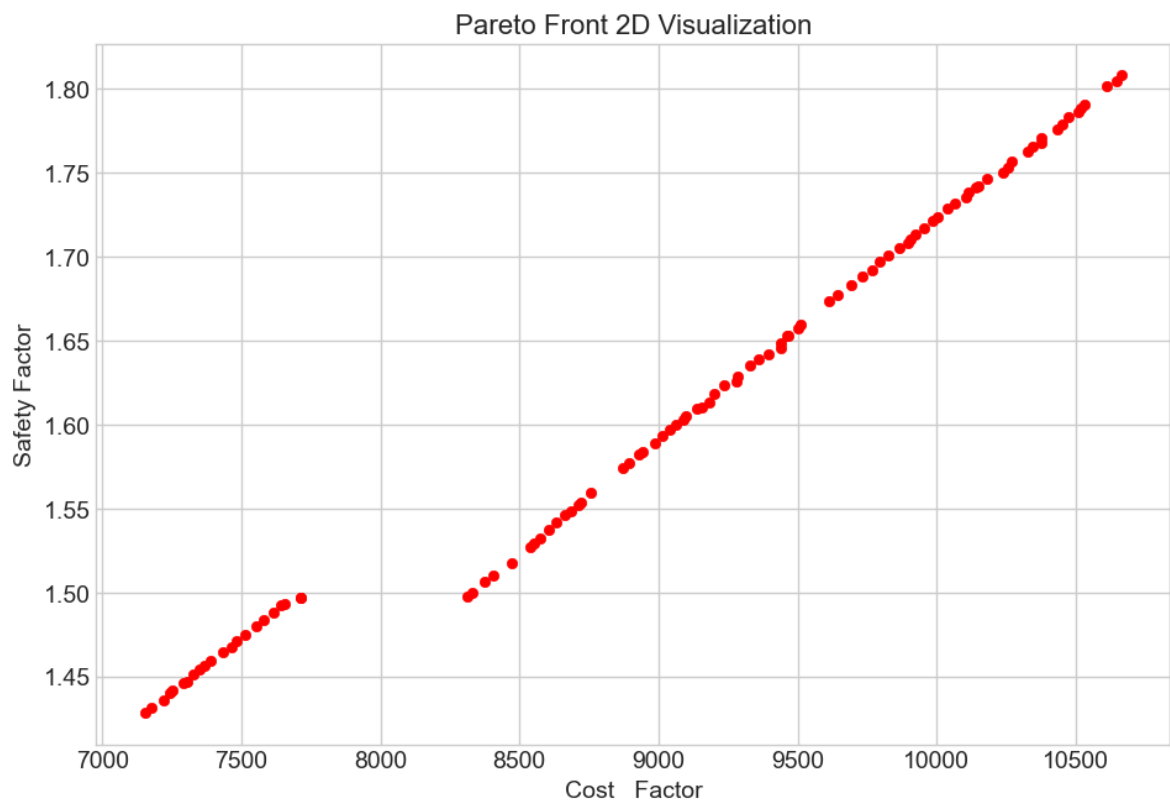


图 5-2 优化结果的帕累托前沿

表 5-2 帕累托前沿上的代表性解

解	气体流量	SO ₂ 含量	吸收率	温度	压力	填料比表面积	填料宽度	最大塔高	最小 ($\frac{L}{V}$) 比率
1	49.77	300.00	0.96	900.05	0.01	109.72	42.91	9.68	2.00
2	49.73	295.97	0.96	900.03	0.01	109.72	42.77	9.79	2.00
3	49.79	300.00	0.96	900.00	0.07	149.89	41.70	9.79	2.00
4	50.00	300.00	0.96	900.03	0.01	100.03	42.58	9.97	2.00
5	49.77	300.00	0.96	900.08	0.02	143.88	43.84	10.00	2.00
6	49.70	300.00	0.96	900.00	0.02	118.19	39.48	9.99	2.00
7	49.99	299.99	0.96	900.02	0.01	143.49	34.46	9.69	2.00
8	49.97	299.97	0.96	900.01	0.03	118.54	33.68	8.79	2.00
9	49.97	299.97	0.96	900.01	0.01	111.59	33.69	8.77	2.00
10	49.99	300.00	0.96	900.02	0.01	114.97	34.44	9.51	2.00

表 5-3 优化结果的代表性目标值

优化结果	成本因素	安全因素
1	7107	1.43
2	10658	1.81
3	8255	1.50
4	7627	1.50
5	8786	1.57
6	10001	1.73
7	8851	1.58
8	9911	1.72
9	10459	1.79
10	10250	1.76

通过对吸收填料塔的多目标优化研究，我们得出以下结论：

- ① 气体流量、SO₂ 含量和吸收率是影响成本因素和安全因素最显著的变量。
- ② 成本因素和安全因素之间存在明显的权衡关系，无法同时实现两者的最优。
- ③ NSGA-II 算法能够有效地找到吸收填料塔设计的帕累托最优解集。

基于以上结论，我们提出以下建议：

- ① 在实际设计中，应根据具体需求在成本和安全之间寻找平衡点。
- ② 可以考虑使用本研究中的优化方法来辅助吸收填料塔的设计决策。
- ③ 在条件允许的情况下，可以进一步考虑其他因素（如环境影响）进行多目标优化。

结论

本设计针对SO₂吸收填料塔进行了全面的工艺设计和计算，旨在实现高效的SO₂气体净化和分离。通过详细分析吸收剂的选择、吸收操作的流程、填料的类型和规格、以及塔内件的设计，我们确定了以 20 °C 清水作为吸收剂，采用逆流操作流程，并选择了 DN25 聚丙烯散装鲍尔环作为填料。在工艺设计中，我们计算了液相和气相的物性数据，进行了物料衡算，并确定了操作温度和压力。

在填料塔工艺尺寸的计算中，我们通过泛点率校核、填料规格校核和液体喷淋密度核算，确保了塔径和填料层高度的合理性。此外，我们还计算了传质单元数和传质单元高度，以确保填料塔的传质效率。最终，我们设计了一个直径为 1000mm，填料层高度为 5.7m 的填料吸收塔，并计算了填料层的压降，确保了操作的经济性和安全性。

在塔内件的设计中，我们选择了槽式液体分布器、驼峰型支撑装置和抽屉式丝网除沫器，以保证气液的均匀分布和高效除沫。通过这些设计，我们预期能够实现SO₂吸收率大于等于 98% 的分离要求，同时保证塔的稳定运行和高效传质。

综上所述，本设计不仅满足了SO₂吸收填料塔的基本工艺要求，而且在提高传质效率、降低能耗和维护成本方面做出了优化。通过本次设计，我们为化工、环保、石油和天然气处理等领域的气体净化和组分分离提供了一个高效、经济和可靠的解决方案。

参考文献

- [1] 刘海洋. \LaTeX 入门[M]. 北京: 电子工业出版社, 2013.
- [2] JOSEPH C, LEBNEBISO J, ADEJOH Z A. Development of a computer software for design of packed absorption column[Z]. 2023.
- [3] 喻健良. 化工设备机械基础[M]. 大连: 大连理工大学出版社, 2014.
- [4] 夏清, 贾绍义. 化工原理. 上册[M]. 第 2 版. 天津: 天津大学出版社, 2012.
- [5] 柴诚敬, 贾绍义. 化工原理课程设计[M]. 北京: 高等教育出版社, 2015.
- [6] 夏清, 贾绍义. 化工原理. 下册[M]. 第 2 版. 天津: 天津大学出版社, 2012.
- [7] 郑育英. 化工原理课程设计[M]. 北京: 化学工业出版社, 2022.
- [8] 陈英南, 刘玉兰. 常用化工单元设备的设计[M]. 第 2 版. 上海: 华东理工大学出版社, 2017.
- [9] 中国国家标准化管理委员会. GB/T 7714-2005: 文后参考文献著录规则[S]. 北京: 中国标准出版社, 2005.

符号说明

M_V	摩尔质量, g/mol
ρ_V	气体密度, kg/m ³
D_V	气体扩散系数, m ² /s
μ_V	气体粘度, Pa · s
E	亨利常数, kPa
m	气体分子质量比, 量纲一
ρ_L	液体密度, kg/m ³
μ_L	液体粘度, Pa · s
σ_L	表面张力, N/m
D_L	液体扩散系数, m ² /s
H	溶解度系数, kmol/(kPa · m ³)
Y_1	进料气体组分, 量纲一
Y_2	出料气体组分, 量纲一
V	气体流量, kmol/h
$(\frac{L}{V})_{min}$	液气比, 量纲一
$(\frac{L}{V})_{min_ratio}$	最小液气比, 量纲一
$\frac{L}{V}$	实际液气比, 量纲一
L	液体流量, kmol/h
X_1	进料液体组分, 量纲一
W_L	液体质量流量, kg/h
W_V	气体质量流量, kg/h
X_{Eckert, ϕ_F}	泛点气速 Eckert 坐标 X 值, 量纲一
Y_{Eckert, ϕ_F}	泛点气速 Eckert 坐标 Y 值, 量纲一
u_F	泛点气速, m/s
u_{design}	设计气速, m/s
D	未圆整塔径, m
$D_{rounded}$	圆整后塔径, 量纲一
u_{actual}	实际操作气速, m/s

$flooding\ ratio$	泛点率，量纲一
$\frac{D_{rounded}}{d}$	圆整后塔径与填料直径比，量纲一
U_{min}	最小喷淋密度， $m^3/(m^2 \cdot h)$
U	实际喷淋密度， $m^3/(m^2 \cdot h)$
S	脱吸因数， m^2
N_{OG}	气相总传质单元数，量纲一
U_L	液相质量通量， $kg/(m^2 \cdot h)$
$\frac{a_w}{a_t}$	有效比表面积比，量纲一
a_w	有效比表面积， m^2/m^3
U_V	气相质量通量， $kg/(m^2 \cdot h)$
k_G	气膜吸收系数， $kmol/(m^2 \cdot h \cdot kPa)$
k_L	液膜吸收系数， m/h
$k_G a$	气膜体积吸收系数， $kmol/(m^2 \cdot h \cdot kPa)$
$k_L a$	液膜体积吸收系数， m/h
$k'_G a$	修正后的气膜体积吸收系数， $kmol/(m^2 \cdot h \cdot kPa)$
$k'_L a$	修正后的液膜体积吸收系数， m/h
$K_G a$	总传质系数， $kmol/(m^2 \cdot h \cdot kPa)$
H_{OG}	填料层高度， m
Z	总高度， m
Z'	设计高度， m
X_{Eckert, ϕ_p}	填料层压降 Eckert 坐标 X 值，量纲一
Y_{Eckert, ϕ_p}	填料层压降 Eckert 坐标 Y 值，量纲一

附录 A 工艺参数处理代码

第 A.1 节 计算工艺参数

代码 A-1 吸收填料塔工艺参数处理代码

```

1 import logging
2 import numpy as np
3 from tabulate import tabulate
4
5 # 配置日志格式和输出级别
6 logging.basicConfig(level=logging.INFO, format='%(message)s')
7 logger = logging.getLogger()
8
9 class AbsorptionColumn:
10     # 单位均为注释中的第一个列出的,后面的为换算公式
11     def __init__(self, gas_flow_rate, so2_mole_fraction, absorption_efficiency,
12                 temperature,
13                 pressure, a_t, d, h_max, L_over_V_min_ratio):
14         # 输入参数
15         self.V_prime = gas_flow_rate # 进口气体体积流量,  $m^3/h$ 
16         self.y1 = so2_mole_fraction # 二氧化硫摩尔分数
17         self.eta = absorption_efficiency # 二氧化硫吸收率
18         self.T = temperature # 温度,  $^{\circ}C$ 
19         self.p = pressure # 压力,  $kPa$ 
20         self.a_t = a_t # 填料总比表面积,  $m^2/m^3$ 
21         self.d = d # 填料宽度  $mm$ 
22         self.h_max = h_max # 最大塔高  $m$ 
23         self.L_over_V_min_ratio = L_over_V_min_ratio # 最小液气比, 量纲 $I$ 
24
25         # 常量参数
26         self.p0 = 1.03125 * 1e2 # 标准大气压,  $kPa$ 

```

```

26     self.R = 8.314                                # 理想气体常数,  $m^3 \cdot$ 
            $kPa/(kmol \cdot K)$ 
27     self.M_air = 28.95                            # 空气的摩尔质量,  $g/$ 
            $mol = kg/kmol$ 
28     self.M_so2 = 64.06                            # SO2的摩尔质量,  $g/mol$ 
            $= kg/kmol$ 
29     self.M_L = 18.02                             # 水的摩尔质量,  $g/mol$ 
            $= kg/kmol$ 
30     self.E = 3.55e3                               # 亨利系数,  $kPa$ 
31     self.rho_L = 998.2                            # 水的密度,  $kg/m^3$ 
32     self.mu_L = 1.005e-3                          # 水的黏度,  $Pa \cdot s =$ 
            $10^3 \text{ mPa} \cdot s = 3600 \text{ kg}/(m \cdot h)$ , 后面计算 $a_w/a_t$ 时转换成  $kg/(m \cdot h)$ 
33     self.mu_V = 1.81e-5                           # 空气黏度,  $Pa \cdot s =$ 
            $10^3 \text{ mPa} \cdot s = 3600 \text{ kg}/(m \cdot h)$ , 后面计算 $a_w/a_t$ 时转换成  $kg/(m \cdot h)$ 
34     self.D_L = 1.47e-9                             # SO2 在水中的扩散系数
            $, m^2/s = 3600 \text{ m}^2/h$ 
35     self.D_V = 1.08e-5                             # SO2在空气中的扩散系
           数,  $m^2/s = 3600 \text{ m}^2/h$ 
36     self.sigma_c = 0.033                           # 材料材质的临界表面张
           力,  $N/m = kg/s^2 = 3600^2 \text{ kg}/h^2$ 
37     self.sigma_L = 0.0728                          # 水的表面张力,  $N/m =$ 
            $kg/s^2 = 3600^2 \text{ kg}/h^2$ 
38     self.g = 9.81                                 # 重力加速度,  $m/s^2 =$ 
            $3600^2 \text{ m}/h^2$ 
39     self.phi_F = 550                              # 散装填料的泛点填料因
           子平均值, 计算泛点气速的系数,  $m^{-1}$ 
40     self.phi_P = 232                              # 散装填料的压降填料因
           子平均值, 计算填料塔压降Eckert图坐标的系数,  $m^{-1}$ 
41     self.varphi = 1                               # 计算泛点气速的系数,
            $l$ 
42     self.psi = 1.45                               # 体积吸收系数修正系数
43
44     # 计算所得参数
45     self.m = None                                  # 相平衡常数
46     self.M_V = None                                # 混合气体平均摩尔质量

```

```

        ,  $g/mol = kg/kmol$ 
47     self.rho_V = None                                # 混合气体的平均密度,
         $kg/m^3$ 
48     self.D_rounded = None                            # 圆整后的直径  $m$ 
49     self.D_over_d = None                            # 填料径比
50     self.u_actual = None                            # 实际操作气速  $m/s$ 
51     self.u_F = None                                  # 泛点气速
52     self.flooding_ratio = None                      # 泛点率
53     self.X_Eckert_phi_F = None                      # Eckert 图的横坐标,
        用于泛点气速, 计算时用到参数  $\Phi_F$ 
54     self.Y_Eckert_phi_F = None                      # Eckert 图的纵坐标,
        用于泛点气速, 计算时用到参数  $\Phi_F$ 
55     self.X_Eckert_phi_P = None                      # Eckert 图的横坐标,
        用于填料层压降, 计算时用到参数  $\Phi_P$ 
56     self.Y_Eckert_phi_P = None                      # Eckert 图的纵坐标,
        用于填料层压降, 计算时用到参数  $\Phi_P$ 
57
58     # 结果存储字典
59     self.results = {}
60
61     # 计算基础物性参数
62     self.compute_gas_properties()
63     self.compute_liquid_properties()
64
65     def compute_gas_properties(self):
66         # 计算气体的摩尔质量、密度和相平衡常数
67         self.M_V = self.y1 * self.M_so2 + (1 - self.y1) * self.M_air
68         self.rho_V = (self.p * self.M_V) / (self.R * (273.15 + self.T))
69         self.m = self.E / self.p
70
71         # 存储结果
72         self.results['M_V'] = self.M_V
73         self.results['rho_V'] = self.rho_V
74         self.results['D_V'] = self.D_V
75         self.results['mu_V'] = self.mu_V

```

```

76     self.results['E'] = self.E
77     self.results['m'] = self.m
78
79     def compute_liquid_properties(self):
80         # 计算液相的相关参数
81         self.H = self.rho_L / (self.E * self.M_L) #  $\text{kmol}/(\text{kPa} \cdot \text{m}^3)$ , self.M_L
            是混合液体的摩尔质量,低浓度,近似为水的摩尔质量
82
83         # 存储结果
84         self.results['rho_L'] = self.rho_L
85         self.results['mu_L'] = self.mu_L
86         self.results['sigma_L'] = self.sigma_L
87         self.results['D_L'] = self.D_L
88         self.results['H'] = self.H
89
90     def material_balance(self):
91         # 计算物料平衡,得到吸收塔中气液相流量的关系
92         self.Y1 = self.y1 / (1 - self.y1)
93         self.Y2 = self.Y1 * (1 - self.eta)
94         self.V = (self.V_prime / 22.4) * (1 - self.y1) * (273 / (273 + self.T)
95             )
96         self.L_over_V_min = (self.Y1 - self.Y2) / (self.Y1 / self.m - 0)
97         self.L_over_V_min_ratio = self.L_over_V_min_ratio
98         self.L_over_V = self.L_over_V_min_ratio * self.L_over_V_min
99         self.L = self.L_over_V * self.V
100         self.X1 = self.V * (self.Y1 - self.Y2) / self.L
101
102         # 存储结果
103         self.results.update({
104             "Y1": self.Y1,
105             "Y2": self.Y2,
106             "V": self.V,
107             "L_over_V_min": self.L_over_V_min,
108             "L_over_V_min_ratio": self.L_over_V_min_ratio,
109             "L_over_V": self.L_over_V,

```

```

1109         "L": self.L,
1110         "X1": self.X1
1111     })
1112
1113     return self.results
1114
1115     def calculate_tower_diameter(self):
1116         # 根据物料平衡结果计算塔径
1117         self.L = self.results["L"]
1118         self.W_L = self.L * self.M_L
1119         self.W_V = self.V_prime * self.rho_V
1120
1121         self.X_Eckert_phi_F = (self.W_L / self.W_V) * (self.rho_V / self.rho_L
1122             ) ** 0.5
1123         self.Y_Eckert_phi_F = 0.034
1124
1125         # 泛点气速
1126         self.u_F = np.sqrt((self.Y_Eckert_phi_F * self.g * self.rho_L) /
1127             (self.phi_F * self.varphi * self.rho_V * (self.mu_L
1128                 * 1000) ** 0.2)) # m/s
1129
1130         # 设计操作气速
1131         self.u_design = 0.7 * self.u_F
1132
1133         # 塔径计算
1134         self.V_s = self.V_prime / 3600
1135         self.D = np.sqrt((4 * self.V_s) / (np.pi * self.u_design))
1136
1137         # 存储结果
1138         self.results.update({
1139             "W_L": self.W_L,
1140             "W_V": self.W_V,
1141             "X_Eckert_phi_F": self.X_Eckert_phi_F,
1142             "Y_Eckert_phi_F": self.Y_Eckert_phi_F,
1143             "u_F": self.u_F,

```

```

142         "u_design": self.u_design,
143         "D": self.D
144     })
145
146     return self.u_F, self.D
147
148     def round_tower_diameter(self, D):
149         # 圆整塔径到小数点后第一位
150         self.D_rounded = np.ceil(D * 10) / 10
151
152         self.results.update({
153             "D_rounded": self.D_rounded
154         })
155
156         return self.D_rounded
157
158     def tower_diameter_check(self, D_rounded, d, a_t):
159         # 塔径校核
160         self.V_s = self.V_prime / 3600
161         self.u_actual = self.V_s / (np.pi * D_rounded**2 / 4)
162         self.u_F = self.results["u_F"]
163
164         # 泛点率参数
165         self.flooding_ratio = self.u_actual / self.u_F
166
167         # 填料规格校核参数
168         self.D_over_d = D_rounded * 1000 / d # 转换为mm
169
170         self.L_W_min = 0.08 # 最小润湿速率,  $m^3/(m \cdot h) = m^2/h$ 
171         self.U_min = self.L_W_min * a_t # 最小喷淋密度,  $m^3/(m^2 \cdot h) = m/h$ 
172         self.L = self.results["L"]
173         self.V_L = self.L * self.M_L / self.rho_L
174         self.A = np.pi * D_rounded**2 / 4

```

```

175
176     # 液体喷淋密度
177     self.U = self.V_L / self.A          # 转换为  $kg/(m^2 \cdot h)$ 
178
179     # 填料规格校核 + 泛点率校核 + 喷淋密度校核
180     if 0.5 <= self.flooding_ratio <= 0.85 and self.D_over_d > 10 and self.
181         U > self.U_min:
182         # 存储结果
183         self.results.update({
184             "u_actual": self.u_actual,
185             "flooding_ratio": self.flooding_ratio,
186             "D_over_d": self.D_over_d,
187             "U_min": self.U_min,
188             "U": self.U
189         })
190
191     return self.flooding_ratio, self.D_over_d, self.U
192
193 def calculate_mass_transfer_units(self):
194     # 脱吸因数
195     self.S = self.m / self.results["L_over_V"] # 使用计算得到的  $S = mV / L$ 
196
197     self.Y1 = self.results["Y1"]
198     self.Y2 = self.results["Y2"]
199     self.Y2_star = 0 # 稀溶液, 平衡摩尔分数近似为0
200
201     # 计算气相总传质单元数  $N_{OG}$ 
202     if (1 - self.S) != 0 and ((1 - self.S) * (self.Y1 - self.Y2_star) / (
203         self.Y2 - self.Y2_star)) + self.S > 0:
204         self.N_OG = (1 / (1 - self.S)) * np.log(((1 - self.S) * (self.Y1 -

```

```

205     # 存储结果
206     self.results.update({
207         "S": self.S,
208         "N_OG": self.N_OG
209     })
210
211     return self.N_OG
212
213 def calculate_membrane_absorption_coefficient(self, D_rounded, a_t):
214     # 膜吸收系数的计算
215     self.L = self.results["L"]
216     self.W_L = self.L * self.M_L
217     self.A = (np.pi * D_rounded**2) / 4
218     self.U_L = self.W_L / self.A # 转换为  $kg/(m^2 \cdot h)$ 
219
220     # 检查输入参数是否合理, 计算 $a_w/a_t$ 和其他传质参数
221     if a_t > 0 and self.mu_L > 0 and self.rho_L > 0 and self.sigma_L > 0
222         and self.U_L > 0:
223         # 计算公式
224         self.a_w_over_a_t = 1 - np.exp( (-1.45 * (self.sigma_c * 3600 *
225             3600 / (self.sigma_L * 3600 * 3600))**0.75 ) * \
226             ( (self.U_L / (a_t * self.mu_L *
227                 3600))**0.1 ) * \
228             ( (self.U_L**2 * self.mu_L * 3600 / (
229                 self.rho_L * self.g * 3600 *
230                 3600))** -0.05 ) * \
231             ( (self.U_L**2 / (self.rho_L * self.
232                 sigma_L * 3600 * 3600 * a_t))
                **0.2 ) )
233
234     else:
235         self.a_w_over_a_t = np.nan # 设置为无效值以避免错误
236
237     self.a_w = self.a_w_over_a_t * a_t
238     self.W_V = self.V_prime * self.rho_V
239     self.U_V = self.W_V / self.A # 转换为  $kg/(m^2 \cdot h)$ 

```



```

233
234 # 气膜吸收系数的公式  $k_G$ 
235 self.k_G = 0.237 * ( ( (self.U_V / (a_t * self.mu_V * 3600))**0.7 ) *
236 \
237 \
238 \
239 \
240 \
241 \
242 \
243 \
244 \
245 \
246 \
247 \
248 \
249 \
250 \
251 \
252 \
253 \
254 \
255 \
256 \
257 \
258 \
259 \
260 \
261 \
262 \
263 \
264 \
265 \
266 \
267 \
268 \
269 \
270 \
271 \
272 \
273 \
274 \
275 \
276 \
277 \
278 \
279 \
280 \
281 \
282 \
283 \
284 \
285 \
286 \
287 \
288 \
289 \
290 \
291 \
292 \
293 \
294 \
295 \
296 \
297 \
298 \
299 \
300 \
301 \
302 \
303 \
304 \
305 \
306 \
307 \
308 \
309 \
310 \
311 \
312 \
313 \
314 \
315 \
316 \
317 \
318 \
319 \
320 \
321 \
322 \
323 \
324 \
325 \
326 \
327 \
328 \
329 \
330 \
331 \
332 \
333 \
334 \
335 \
336 \
337 \
338 \
339 \
340 \
341 \
342 \
343 \
344 \
345 \
346 \
347 \
348 \
349 \
350 \
351 \
352 \
353 \
354 \
355 \
356 \
357 \
358 \
359 \
360 \
361 \
362 \
363 \
364 \
365 \
366 \
367 \
368 \
369 \
370 \
371 \
372 \
373 \
374 \
375 \
376 \
377 \
378 \
379 \
380 \
381 \
382 \
383 \
384 \
385 \
386 \
387 \
388 \
389 \
390 \
391 \
392 \
393 \
394 \
395 \
396 \
397 \
398 \
399 \
400 \
401 \
402 \
403 \
404 \
405 \
406 \
407 \
408 \
409 \
410 \
411 \
412 \
413 \
414 \
415 \
416 \
417 \
418 \
419 \
420 \
421 \
422 \
423 \
424 \
425 \
426 \
427 \
428 \
429 \
430 \
431 \
432 \
433 \
434 \
435 \
436 \
437 \
438 \
439 \
440 \
441 \
442 \
443 \
444 \
445 \
446 \
447 \
448 \
449 \
450 \
451 \
452 \
453 \
454 \
455 \
456 \
457 \
458 \
459 \
460 \
461 \
462 \
463 \
464 \
465 \
466 \
467 \
468 \
469 \
470 \
471 \
472 \
473 \
474 \
475 \
476 \
477 \
478 \
479 \
480 \
481 \
482 \
483 \
484 \
485 \
486 \
487 \
488 \
489 \
490 \
491 \
492 \
493 \
494 \
495 \
496 \
497 \
498 \
499 \
500 \
501 \
502 \
503 \
504 \
505 \
506 \
507 \
508 \
509 \
510 \
511 \
512 \
513 \
514 \
515 \
516 \
517 \
518 \
519 \
520 \
521 \
522 \
523 \
524 \
525 \
526 \
527 \
528 \
529 \
530 \
531 \
532 \
533 \
534 \
535 \
536 \
537 \
538 \
539 \
540 \
541 \
542 \
543 \
544 \
545 \
546 \
547 \
548 \
549 \
550 \
551 \
552 \
553 \
554 \
555 \
556 \
557 \
558 \
559 \
560 \
561 \
562 \
563 \
564 \
565 \
566 \
567 \
568 \
569 \
570 \
571 \
572 \
573 \
574 \
575 \
576 \
577 \
578 \
579 \
580 \
581 \
582 \
583 \
584 \
585 \
586 \
587 \
588 \
589 \
590 \
591 \
592 \
593 \
594 \
595 \
596 \
597 \
598 \
599 \
600 \
601 \
602 \
603 \
604 \
605 \
606 \
607 \
608 \
609 \
610 \
611 \
612 \
613 \
614 \
615 \
616 \
617 \
618 \
619 \
620 \
621 \
622 \
623 \
624 \
625 \
626 \
627 \
628 \
629 \
630 \
631 \
632 \
633 \
634 \
635 \
636 \
637 \
638 \
639 \
640 \
641 \
642 \
643 \
644 \
645 \
646 \
647 \
648 \
649 \
650 \
651 \
652 \
653 \
654 \
655 \
656 \
657 \
658 \
659 \
660 \
661 \
662 \
663 \
664 \
665 \
666 \
667 \
668 \
669 \
670 \
671 \
672 \
673 \
674 \
675 \
676 \
677 \
678 \
679 \
680 \
681 \
682 \
683 \
684 \
685 \
686 \
687 \
688 \
689 \
690 \
691 \
692 \
693 \
694 \
695 \
696 \
697 \
698 \
699 \
700 \
701 \
702 \
703 \
704 \
705 \
706 \
707 \
708 \
709 \
710 \
711 \
712 \
713 \
714 \
715 \
716 \
717 \
718 \
719 \
720 \
721 \
722 \
723 \
724 \
725 \
726 \
727 \
728 \
729 \
730 \
731 \
732 \
733 \
734 \
735 \
736 \
737 \
738 \
739 \
740 \
741 \
742 \
743 \
744 \
745 \
746 \
747 \
748 \
749 \
750 \
751 \
752 \
753 \
754 \
755 \
756 \
757 \
758 \
759 \
760 \
761 \
762 \
763 \
764 \
765 \
766 \
767 \
768 \
769 \
770 \
771 \
772 \
773 \
774 \
775 \
776 \
777 \
778 \
779 \
780 \
781 \
782 \
783 \
784 \
785 \
786 \
787 \
788 \
789 \
790 \
791 \
792 \
793 \
794 \
795 \
796 \
797 \
798 \
799 \
800 \
801 \
802 \
803 \
804 \
805 \
806 \
807 \
808 \
809 \
810 \
811 \
812 \
813 \
814 \
815 \
816 \
817 \
818 \
819 \
820 \
821 \
822 \
823 \
824 \
825 \
826 \
827 \
828 \
829 \
830 \
831 \
832 \
833 \
834 \
835 \
836 \
837 \
838 \
839 \
840 \
841 \
842 \
843 \
844 \
845 \
846 \
847 \
848 \
849 \
850 \
851 \
852 \
853 \
854 \
855 \
856 \
857 \
858 \
859 \
860 \
861 \
862 \
863 \
864 \
865 \
866 \
867 \
868 \
869 \
870 \
871 \
872 \
873 \
874 \
875 \
876 \
877 \
878 \
879 \
880 \
881 \
882 \
883 \
884 \
885 \
886 \
887 \
888 \
889 \
890 \
891 \
892 \
893 \
894 \
895 \
896 \
897 \
898 \
899 \
900 \
901 \
902 \
903 \
904 \
905 \
906 \
907 \
908 \
909 \
910 \
911 \
912 \
913 \
914 \
915 \
916 \
917 \
918 \
919 \
920 \
921 \
922 \
923 \
924 \
925 \
926 \
927 \
928 \
929 \
930 \
931 \
932 \
933 \
934 \
935 \
936 \
937 \
938 \
939 \
940 \
941 \
942 \
943 \
944 \
945 \
946 \
947 \
948 \
949 \
950 \
951 \
952 \
953 \
954 \
955 \
956 \
957 \
958 \
959 \
960 \
961 \
962 \
963 \
964 \
965 \
966 \
967 \
968 \
969 \
970 \
971 \
972 \
973 \
974 \
975 \
976 \
977 \
978 \
979 \
980 \
981 \
982 \
983 \
984 \
985 \
986 \
987 \
988 \
989 \
990 \
991 \
992 \
993 \
994 \
995 \
996 \
997 \
998 \
999 \
1000 \

```

```
262     self.k_Ga = self.k_G * self.a_w * self.psi**1.1
263     self.k_La = self.k_L * self.a_w * self.psi**0.4
264
265     self.results.update({
266         "k_Ga": self.k_Ga,
267         "k_La": self.k_La
268     })
269
270     return self.k_Ga, self.k_La
271
272 def correct_volume_absorption_coefficient(self):
273     # 修正体积吸收系数
274     self.k_Ga = self.results["k_Ga"]
275     self.k_La = self.results["k_La"]
276
277     self.u_over_uF = self.u_actual / self.u_F
278
279     self.k_prime_Ga = (1 + 9.5 * (self.u_over_uF - 0.5)**1.4) * self.k_Ga
280     self.k_prime_La = (1 + 2.6 * (self.u_over_uF - 0.5)**2.2) * self.k_La
281
282     # 存储结果
283     self.results.update({
284         "k_prime_Ga": self.k_prime_Ga,
285         "k_prime_La": self.k_prime_La
286     })
287
288     return self.k_prime_Ga, self.k_prime_La
289
290 def calculate_overall_absorption_coefficient(self):
291     # 计算总传质系数
292     self.k_prime_Ga = self.results.get("k_prime_Ga", self.results.get("k_Ga")) # 如果 'k_prime_Ga' 不存在, 使用 'k_Ga'
293     self.k_prime_La = self.results.get("k_prime_La", self.results.get("k_La")) # 如果 'k_prime_La' 不存在, 使用 'k_La'
294     self.H = self.results["H"]
```

```

295     self.K_Ga = 1 / (1 / self.k_prime_Ga + 1 / (self.H * self.k_prime_La))
296
297     # 存储结果
298     self.results["K_Ga"] = self.K_Ga
299     return self.K_Ga
300
301     def calculate_packing_height(self, D_rounded):
302         # 计算填料层高度  $H_{OG}$ 
303         self.K_Ga = self.results["K_Ga"] # 总传质系数
304         self.p = self.p # 总压力
305         self.V = self.results["V"] # 气体流量
306         self.Ω = np.pi * (D_rounded / 2)**2 #  $D_{rounded}$  圆整后
            的直径
307         self.H_OG = self.V / (self.K_Ga * self.p * self.Ω) #  $\Omega$  是公式中的系数
308         self.N_OG = self.results["N_OG"] # 传质单元数
309
310         # 计算总高度  $Z$ 
311         self.Z = self.H_OG * self.N_OG
312
313         # 计算设计高度  $Z_{prime}$ , 增加25%余量
314         self.Z_prime = 1.25 * self.Z
315
316         # 存储结果
317         self.results.update({
318             "H_OG": self.H_OG,
319             "Z": self.Z,
320             "Z_prime": self.Z_prime
321         })
322
323         return self.H_OG, self.Z, self.Z_prime
324
325     def calculate_packing_delta_pressure(self):
326         # 计算Eckert降压的适用坐标, 以便查填料塔降压
327         self.L = self.results["L"]
328         self.W_L = self.L * self.M_L

```

```

329     self.W_V = self.V_prime * self.rho_V
330
331     # 计算X_Eckert_phi_P
332     self.X_Eckert_phi_P = (self.W_L / self.W_V) * (self.rho_V / self.rho_L
333         ) ** 0.5
334
335     # 计算Y_Eckert_phi_P
336     self.Y_Eckert_phi_P = (self.u_actual**2 * self.phi_P * self.varphi) /
337         self.g * \
338             (self.rho_V / self.rho_L) * (self.mu_L * 3600)
339         **0.2
340
341     # 存储结果
342     self.results.update({
343         "X_Eckert_phi_P": self.X_Eckert_phi_P,
344         "Y_Eckert_phi_P": self.Y_Eckert_phi_P
345     })
346
347     return self.X_Eckert_phi_P, self.Y_Eckert_phi_P
348
349 def display_results(self):
350     # 使用表格展示计算结果
351     logger.info("计算结果汇总:")
352     table_data = []
353     descriptions = {
354         'M_V': '摩尔质量',
355         'rho_V': '气体密度',
356         'D_V': '气体扩散系数',
357         'mu_V': '气体粘度',
358         'E': '亨利常数',
359         'm': '气体分子质量比',
360         'rho_L': '液体密度',
361         'mu_L': '液体粘度',
362         'sigma_L': '表面张力',
363         'D_L': '液体扩散系数',

```

361 'H': '溶解度系数',
 362 'Y1': '进料气体组分',
 363 'Y2': '出料气体组分',
 364 'V': '气体流量',
 365 'L_{over_V_min}': '液气比',
 366 'L_{over_V_min_ratio}': '最小液气比',
 367 'L_{over_V}': '实际液气比',
 368 'L': '液体流量',
 369 'X1': '进料液体组分',
 370 'W_L': '液体质量流量',
 371 'W_V': '气体质量流量',
 372 'X_{Eckert_phi_F}': '泛点气速Eckert坐标X值',
 373 'Y_{Eckert_phi_F}': '泛点气速Eckert坐标Y值',
 374 'u_F': '泛点气速',
 375 'u_{design}': '设计气速',
 376 'D': '未圆整塔径',
 377 'D_{rounded}': '圆整后塔径',
 378 'u_{actual}': '实际操作气速',
 379 'flooding_ratio': '泛点率',
 380 'D_{over_d}': '圆整后塔径与填料直径比',
 381 'U_{min}': '最小喷淋密度',
 382 'U': '实际喷淋密度',
 383 'S': '脱吸因数',
 384 'N_{OG}': '气相总传质单元数',
 385 'U_L': '液相质量通量',
 386 'a_{w_{over_a_t}}': '有效比表面积比',
 387 'a_w': '有效比表面积',
 388 'U_V': '气相质量通量',
 389 'k_G': '气膜吸收系数',
 390 'k_L': '液膜吸收系数',
 391 'k_{Ga}': '气膜体积吸收系数',
 392 'k_{La}': '液膜体积吸收系数',
 393 'k_{prime_Ga}': '修正后的气膜体积吸收系数',
 394 'k_{prime_La}': '修正后的液膜体积吸收系数',
 395 'K_{Ga}': '总传质系数',

```

396      'H_OG': '填料层高度',
397      'Z': '总高度',
398      'Z_prime': '设计高度',
399      'X_Eckert_phi_P': '填料层压降Eckert坐标X值',
400      'Y_Eckert_phi_P': '填料层压降Eckert坐标Y值'
401  }
402  units = {
403      'M_V': 'g/mol',
404      'rho_V': 'kg/m³',
405      'mu_V': 'Pa • s',
406      'D_V': 'm²/s',
407      'E': 'kPa',
408      'm': '-',
409      'rho_L': 'kg/m³',
410      'mu_L': 'Pa • s',
411      'sigma_L': 'N/m',
412      'D_L': 'm²/s',
413      'H': 'kmol/(kPa • m³)',
414      'Y1': '-',
415      'Y2': '-',
416      'V': 'kmol/h',
417      'L_over_V_min': '-',
418      'L_over_V_min_ratio': '-',
419      'L_over_V': '-',
420      'L': 'kmol/h',
421      'X1': '-',
422      'W_L': 'kg/h',
423      'W_V': 'kg/h',
424      'X_Eckert_phi_F': '-',
425      'Y_Eckert_phi_F': '-',
426      'u_F': 'm/s',
427      'u_design': 'm/s',
428      'D': 'm',
429      'D_rouned': 'm',
430      'u_actual': 'm/s',

```

```

431         'flooding_ratio': '-',
432         'D_over_d ': '-',
433         'U_min': 'm³/(m² • h)',
434         'U': 'm³/(m² • h)',
435         'S': '-',
436         'N_OG': '-',
437         'U_L': 'kg/(m² • h)',
438         'a_w_over_a_t': '-',
439         'a_w': 'm²/m³',
440         'U_V': 'kg/(m² • h)',
441         'k_G': 'kmol/(m² • h • kPa)',
442         'k_L': 'm/h',
443         'k_Ga': 'kmol/(m² • h • kPa)',
444         'k_La': 'm/h',
445         'k_prime_Ga': 'kmol/(m² • h • kPa)',
446         'k_prime_La': 'm/h',
447         'K_Ga': 'kmol/(m² • h • kPa)',
448         'H_OG': 'm',
449         'Z': 'm',
450         'Z_prime': 'm',
451         'X_Eckert_phi_P': '-',
452         'Y_Eckert_phi_P': '-'
453     }
454
455     for key, value in self.results.items():
456         desc = descriptions.get(key, '-')
457         unit = units.get(key, '-')
458         table_data.append([desc, key, value, unit])
459
460     logger.info("\n" + tabulate(table_data, headers=['描述信息', '参数符号', '参数值', '单位'], tablefmt='grid'))
461
462 def main():
463     # 任务参数
464     gas_flow_rate = 1000                # 进口气体体积流量, m³/h

```

```

465     so2_mole_fraction = 0.060      # 二氧化硫摩尔分数
466     absorption_efficiency = 0.98    # 二氧化硫吸收率
467     temperature = 20                # 操作温度, °C
468     pressure = 120                  # 操作压力, kPa
469     a_t = 175                       # 填料总比表面积,  $m^2/m^3$ 
470     d = 38                          # 填料宽度 mm
471     h_max = 6                       # 最大塔高 m
472     L_over_V_min_ratio = 1.40       # 最小液气比 量纲1
473
474     # 创建吸收塔对象
475     absorption_column = AbsorptionColumn(gas_flow_rate, so2_mole_fraction,
476                                           absorption_efficiency, temperature,
477                                           pressure, a_t, d, h_max,
478                                           L_over_V_min_ratio)
479
480     # 执行物料平衡计算
481     absorption_column.material_balance()
482
483     # 计算塔径
484     u_F, D = absorption_column.calculate_tower_diameter()
485
486     # 圆整塔径
487     D_rounded = absorption_column.round_tower_diameter(D)
488
489     # 塔径校核
490     flooding_ratio, D_over_d, U = absorption_column.tower_diameter_check(
491         D_rounded, absorption_column.d, absorption_column.a_t)
492
493     # 计算传质单元数
494     _ = absorption_column.calculate_mass_transfer_units()
495
496     # 计算膜吸收系数
497     a_w, k_G, k_L = absorption_column.
498         calculate_membrane_absorption_coefficient(D_rounded, absorption_column
499             .a_t)

```



```

495
496 # 计算体积吸收系数
497 k_Ga, k_La = absorption_column.calculate_volume_absorption_coefficient()
498
499 # 修正体积吸收系数
500 k_prime_Ga, k_prime_La = absorption_column.
    correct_volume_absorption_coefficient()
501
502 # 计算总传质系数
503 _ = absorption_column.calculate_overall_absorption_coefficient()
504
505 # 计算填料层高度
506 H_OG, Z, Z_prime = absorption_column.calculate_packing_height(D_rounded)
507
508 # 计算Eckert坐标值以查压降
509 X_Eckert_phi_P, Y_Eckert_phi_P = absorption_column.
    calculate_packing_delta_pressure()
510
511 # 展示所有计算结果
512 absorption_column.display_results()
513
514 # 校核塔径设计合理性输出结果
515 if 0.5 <= absorption_column.results['flooding_ratio'] <= 0.85 and
    absorption_column.results['D_over_d'] > 10 \
516     and absorption_column.results['U'] > absorption_column.results['
        U_min']:
517     logger.info("塔径校核通过,塔径设计合理")
518 else:
519     logger.info("塔径校核不通过,需要调整其他参数")
520
521 # 填料层是否需要分段
522 if absorption_column.results['Z_prime'] < absorption_column.h_max:
523     logger.info(f"设计填料层高度为 {absorption_column.results['Z_prime
        ']:.6f} m, 小于等于最大高度 {absorption_column.h_max} m, 故不进行
        分段。")

```

```
524     else:
525         logger.info(f"设计填料层高度为 {absorption_column.results['Z_prime']:.6f} m, 超过最大高度 {absorption_column.h_max} m, 需要进行分段。")
526
527         absorption_column_results = absorption_column.results
528
529     return absorption_column, absorption_column_results
530
531 if __name__ == "__main__":
532
533     # 调用 main 函数，获取实例化对象和计算结果汇总字典
534     absorption_column, absorption_column_results = main()
```

第 A.2 节 定义优化问题

代码 A-2 吸收塔问题类定义了三个优目标：经济最低，安全最大，符合最大

```
1 import numpy as np
2 from absorption_column import AbsorptionColumn
3
4 # 创建吸收塔问题的类
5 class AbsorptionColumnProblem:
6
7     def __init__(self):
8         pass
9
10    def evaluate(self, x):
11        gas_flow_rate, so2_mole_fraction, absorption_efficiency, temperature,
12        pressure, a_t, d, h_max, L_over_V_min_ratio = x
13
14        # 创建吸收塔对象
15        absorption_column = AbsorptionColumn(gas_flow_rate, so2_mole_fraction,
16        absorption_efficiency, temperature,
```

```
15         pressure, a_t, d, h_max,  
16             L_over_V_min_ratio)  
17  
18     # 执行物料平衡计算  
19     absorption_column.material_balance()  
20  
21     # 计算塔径  
22     u_F, D = absorption_column.calculate_tower_diameter()  
23  
24     # 圆整塔径  
25     D_rounded = absorption_column.round_tower_diameter(D)  
26  
27     # 塔径校核  
28     flooding_ratio, D_over_d, U = absorption_column.tower_diameter_check(  
29         D_rounded, absorption_column.d, absorption_column.a_t)  
30  
31     # 计算传质单元数  
32     _ = absorption_column.calculate_mass_transfer_units()  
33  
34     # 计算膜吸收系数  
35     a_w, k_G, k_L = absorption_column.  
36         calculate_membrane_absorption_coefficient(D_rounded,  
37             absorption_column.a_t)  
38  
39     # 计算体积吸收系数  
40     k_Ga, k_La = absorption_column.calculate_volume_absorption_coefficient  
41         ()  
42  
43     # 修正体积吸收系数  
44     k_prime_Ga, k_prime_La = absorption_column.  
45         correct_volume_absorption_coefficient()  
46  
47     # 计算总传质系数  
48     _ = absorption_column.calculate_overall_absorption_coefficient()
```

```

44     # 计算填料层高度
45     H_OG, Z, Z_prime = absorption_column.calculate_packing_height(
46         D_rounded)
47
48     # 计算Eckert坐标值以查压降
49     X_Eckert_phi_P, Y_Eckert_phi_P = absorption_column.
50         calculate_packing_delta_pressure()
51
52     # 分段处理
53     num_sections = np.ceil(Z_prime / h_max)
54     effective_height = Z_prime / num_sections
55
56     # 计算塔的有效体积（塔内径减去壁厚后的体积）
57     width = 0.08
58     internal_radius = (D_rounded / 2) - width # 塔皮的厚度 width
59     effective_volume = np.pi * internal_radius**2 * effective_height
60
61     # 填料个数基于塔的体积来计算
62     packing_cost_factor_per_unit = d * 0.5
63     number_of_packings = effective_volume / (d * h_max) * num_sections #
64         基于体积的填料个数
65     packing_cost_factor = packing_cost_factor_per_unit *
66         number_of_packings
67
68     # 计算塔的总成本，包括分段成本
69     base_cost = effective_volume * 10000 + packing_cost_factor * 1500
70
71     # 每分一段的额外成本
72     section_cost_per_unit = 2000
73     section_cost = (num_sections - 1) * section_cost_per_unit # 减1是因为
74         第一个段不需要额外成本
75     total_cost = base_cost + section_cost
76
77     # 安全系数计算
78     safety_factor = 1 / (flooding_ratio + 1e-5)

```

```

74
75     return total_cost, safety_factor, absorption_column

```

第 A.3 节 选择关键变量

代码 A-3 控制变量分析，敏感度分析，相关性分析

```

1  import numpy as np
2  import matplotlib.pyplot as plt
3  from absorption_problem import AbsorptionColumnProblem
4
5  # 设置字体为 Times New Roman
6  plt.rcParams['font.family'] = 'Times New Roman'
7  plt.rcParams['axes.unicode_minus'] = False
8
9  # 使用 seaborn 风格美化
10 plt.style.use('seaborn-v0_8-whitegrid')
11
12 # 固定的其他变量值
13 fixed_values = np.array([1000, 0.05, 0.98, 20, 120, 175, 38, 7.5, 1.40])
14
15 # 变量名称（英文）
16 variable_names = [
17     'Gas Flow Rate', '$SO_{2}$ Mole Fraction', 'Absorption Efficiency',
18     'Temperature', 'Pressure', 'Packing Surface Area per Volume $a_{t}$',
19     'Packing Width $d$', 'Maximum Tower Height $h_{max}$', 'Minimum L/V Ratio'
20 ]
21
22 # x 轴变量的取值范围
23 x_ranges = [
24     np.linspace(900, 1100, 200),    # Gas Flow Rate
25     np.linspace(0.01, 0.1, 200),    # SO2 Mole Fraction
26     np.linspace(0.96, 0.98, 200),   # Absorption Efficiency
27     np.linspace(20, 50, 200),        # Temperature

```

```

28     np.linspace(100, 150, 200),      # Pressure
29     np.linspace(150, 200, 200),      # Packing Surface Area per Volume  $a_t$ 
30     np.linspace(25, 50, 200),        # Packing Width  $d$ 
31     np.linspace(5, 10, 200),         # Maximum Tower Height  $h_{max}$ 
32     np.linspace(1.2, 2.0, 200)       # Minimum L/V Ratio
33 ]
34
35 problem = AbsorptionColumnProblem()
36
37 # 归一化函数, 使用 Min-Max 归一化将值缩放到 [0, 1]
38 def normalize(values):
39     min_val = np.min(values)
40     max_val = np.max(values)
41     if max_val - min_val == 0:
42         return np.zeros_like(values) + 0.5
43     return (values - min_val) / (max_val - min_val)
44
45 # 移动平均函数
46 def moving_average(data, window_size):
47     """ 计算移动平均 """
48     return np.convolve(data, np.ones(window_size)/window_size, mode='valid')
49
50 # 参数敏感性分析函数
51 def sensitivity_analysis(problem, fixed_values, variable_names, perturbation
52                          =0.01):
53     sensitivity_results = {}
54
55     for i, name in enumerate(variable_names):
56         original_value = fixed_values[i]
57         perturbed_values = [original_value * (1 - perturbation),
58                             original_value * (1 + perturbation)]
59         costs, safety_factors = [], []
60
61         for value in perturbed_values:
62             x = fixed_values.copy()

```

```
61         x[i] = value
62         cost_factor, safety_factor, _ = problem.evaluate(x)
63         costs.append(cost_factor)
64         safety_factors.append(safety_factor)
65
66         # 计算标准差作为敏感性度量
67         sensitivity_results[name] = {
68             'cost_sensitivity': np.std(costs),
69             'safety_factor_sensitivity': np.std(safety_factors)
70         }
71
72     return sensitivity_results
73
74 # 相关性分析函数
75 def correlation_analysis(problem, x_ranges, fixed_values):
76     results = {'cost_factor': [], 'safety_factor': []}
77     variable_values = []
78
79     for i, var_range in enumerate(x_ranges):
80         costs, safety_factors = [], []
81         for value in var_range:
82             x = fixed_values.copy()
83             x[i] = value
84             cost_factor, safety_factor, _ = problem.evaluate(x)
85             costs.append(cost_factor)
86             safety_factors.append(safety_factor)
87
88         # 记录每个变量的结果
89         results['cost_factor'].append(costs)
90         results['safety_factor'].append(safety_factors)
91         variable_values.append(var_range)
92
93     return variable_values, results
94
95 # 在计算相关系数之前，检查变量数据
```

```
96 def safe_corrcoef(x, y):
97     if np.std(x) == 0 or np.std(y) == 0:
98         return np.nan
99     else:
100         return np.corrcoef(x, y)[0, 1]
101
102 # 相关性分析时使用 safe_corrcoef 函数
103 def get_correlation_coefficients(variable_values, results, variable_names):
104     correlations = {}
105     for i, name in enumerate(variable_names):
106         cost_corr = safe_corrcoef(variable_values[i], results['cost_factor'][i
107                                     ])
108         safety_corr = safe_corrcoef(variable_values[i], results['safety_factor
109                                     '][i])
110         correlations[name] = {
111             'cost_correlation': cost_corr,
112             'safety_correlation': safety_corr
113         }
114     return correlations
115
116 # 绘制9张图，控制每个变量，其他变量固定
117 fig, axes = plt.subplots(3, 3, figsize=(18, 18), dpi=75) # 调整图的大小以增加
118 # 子图之间的空间
119 window_size = 5 # 设置移动平均窗口大小
120
121 for i in range(9):
122     ax = axes[i // 3, i % 3]
123     variable_values = x_ranges[i]
124
125     costs = []
126     safety_factors = []
127
128     for value in variable_values:
129         x = fixed_values.copy()
130         x[i] = value
```



```
128     cost_factor, safety_factor, _ = problem.evaluate(x)
129     costs.append(cost_factor)
130     safety_factors.append(safety_factor)
131
132     # 归一化数据
133     normalized_costs = normalize(np.array(costs))
134     normalized_safety_factors = normalize(np.array(safety_factors))
135
136     # 平滑处理
137     smoothed_costs = moving_average(normalized_costs, window_size)
138     smoothed_safety_factors = moving_average(normalized_safety_factors,
139                                             window_size)
140
141     # 重新计算变量值的范围，因为移动平均会减少数据点数
142     smoothed_variable_values = variable_values[window_size-1:]
143
144     # 绘制平滑后的归一化曲线
145     ax.plot(smoothed_variable_values, smoothed_costs, label='Cost Factor (
146             Smoothed)', color='purple', lw=2)
147     ax.plot(smoothed_variable_values, smoothed_safety_factors, label='Safety
148             Factor (Smoothed)', color='green', lw=2, linestyle='--')
149
150     ax.set_title(f'{variable_names[i]} vs. Metrics (Smoothed)', fontsize=18)
151     ax.set_xlabel(variable_names[i], fontsize=15)
152     ax.set_ylabel('Normalized Value', fontsize=15)
153     ax.tick_params(axis='both', which='major', labelsize=12)
154
155     # 添加图例
156     ax.legend(fontsize=12)
157
158     # 调整布局以避免重叠，并调整子图之间的间距
159     plt.subplots_adjust(wspace=0.3, hspace=0.4)
160
161     # 显示图表
162     plt.show()
```

```

160
161 # 运行相关性分析
162 variable_values, results = correlation_analysis(problem, x_ranges,
        fixed_values)
163
164 # 打印相关性系数
165 correlation_coefficients = get_correlation_coefficients(variable_values,
        results, variable_names)
166 for param, correlations in correlation_coefficients.items():
167     print(f"{param}:")
168     print(f"  Cost    Factor Correlation: {correlations['cost_correlation']:.4 f
        }")
169     print(f"  Safety Factor Correlation: {correlations['safety_correlation
        ']:.4 f}")

```

第 A.4 节 实现目标优化

代码 A-4 提取关键变量用遗传算法，梯度下降实现最优化

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3 from pymoo.algorithms.moo.nsga2 import NSGA2
4 from pymoo.core.problem import ElementwiseProblem
5 from pymoo.operators.sampling.rnd import FloatRandomSampling
6 from pymoo.operators.crossover.sbx import SimulatedBinaryCrossover
7 from pymoo.operators.mutation.pm import PolynomialMutation
8 from pymoo.optimize import minimize
9 from pymoo.termination import get_termination
10 from absorption_problem import AbsorptionColumnProblem
11
12 # 设置字体为 Times New Roman
13 plt.rcParams['font.family'] = 'Times New Roman'
14 plt.rcParams['axes.unicode_minus'] = False
15

```

```
16 # 使用 seaborn 风格美化
17 plt.style.use('seaborn-v0_8-whitegrid')
18
19 # 固定的其他变量值
20 fixed_values = np.array([1000, 0.05, 0.98, 20, 120, 175, 38, 7.5, 1.40])
21
22 # 变量名称（英文）
23 variable_names = [
24     'Gas Flow Rate', '$SO_{2}$ Mole Fraction', 'Absorption Efficiency',
25     'Temperature', 'Pressure', 'Packing Surface Area per Volume $a_{t}$',
26     'Packing Width $d$', 'Maximum Tower Height $h_{max}$', 'Minimum L/V Ratio'
27 ]
28
29 # 初始化吸收塔问题
30 problem_instance = AbsorptionColumnProblem()
31
32 # 相关性分析函数
33 def correlation_analysis(problem, x_ranges, fixed_values):
34     results = {'cost_factor': [], 'safety_factor': []}
35     variable_values = []
36
37     for i, var_range in enumerate(x_ranges):
38         cost_factors, safety_factors = [], []
39         for value in var_range:
40             x = fixed_values.copy()
41             x[i] = value
42             cost_factor, safety_factor, _ = problem.evaluate(x)
43             cost_factors.append(cost_factor)
44             safety_factors.append(safety_factor)
45
46         # 记录每个变量的结果
47         results['cost_factor'].append(cost_factors)
48         results['safety_factor'].append(safety_factors)
49         variable_values.append(var_range)
50
```

```

51     return variable_values, results
52
53 # 安全相关系数计算函数
54 def safe_corrcoef(x, y):
55     if np.std(x) == 0 or np.std(y) == 0:
56         return np.nan
57     else:
58         return np.corrcoef(x, y)[0, 1]
59
60 # 获取相关性系数
61 def get_correlation_coefficients(variable_values, results, variable_names):
62     correlations = {}
63     for i, name in enumerate(variable_names):
64         cost_factor_corr = safe_corrcoef(variable_values[i], results['
65             cost_factor'][i])
66         safety_corr = safe_corrcoef(variable_values[i], results['safety_factor
67             '][i])
68         correlations[name] = {
69             'cost_factor_correlation': cost_factor_corr,
70             'safety_correlation': safety_corr
71         }
72     return correlations
73
74 # x 轴变量的取值范围
75 x_ranges = [
76     np.linspace(900, 1100, 200),    # Gas Flow Rate
77     np.linspace(0.01, 0.1, 200),    # SO2 Mole Fraction
78     np.linspace(0.96, 0.98, 200),   # Absorption Efficiency
79     np.linspace(20, 50, 200),        # Temperature
80     np.linspace(100, 150, 200),      # Pressure
81     np.linspace(150, 200, 200),      # Packing Surface Area per Volume a_t
82     np.linspace(25, 50, 200),        # Packing Width d
83     np.linspace(5, 10, 200),         # Maximum Tower Height h_max
84     np.linspace(1.2, 2.0, 200)      # Minimum L/V Ratio
85 ]

```

```
84
85 # 运行相关性分析
86 variable_values, results = correlation_analysis(problem_instance, x_ranges,
87         fixed_values)
88
89 # 获取相关性系数
90 correlation_coefficients = get_correlation_coefficients(variable_values,
91         results, variable_names)
92
93 # 根据相关性排序，并选择相关性最高的  $n$  个变量，这里设置为 9，即全部考虑
94 n = 9
95 sorted_params = sorted(correlation_coefficients.items(), key=lambda item: abs(
96         item[1]['cost_factor_correlation']), reverse=True)
97 top_correlated_params = [param[0] for param in sorted_params[:n]]
98
99 # 打印相关性最高的四个变量
100 print("选择的关键变量:", top_correlated_params)
101
102 # 获取关键变量的索引
103 variable_indices = [variable_names.index(param) for param in
104         top_correlated_params]
105
106 # 变量的上下限定义（针对所有可能的变量）
107 bounds_dict = {
108     'Gas Flow Rate': (900, 1100),
109     '$SO_{2}$ Mole Fraction': (0.01, 0.1),
110     'Absorption Efficiency': (0.96, 0.98),
111     'Temperature': (20, 50),
112     'Pressure': (100, 150),
113     'Packing Surface Area per Volume $a_{t}$': (150, 300),
114     'Packing Width $d$': (25, 50),
115     'Maximum Tower Height $h_{max}$': (5, 10),
116     'Minimum L/V Ratio': (1.2, 2.0)
117 }
```

```

115 # 获取关键变量的索引并根据这些索引提取对应的上下限
116 variable_indices = [variable_names.index(param) for param in
    top_correlated_params]
117 bounds = [bounds_dict[variable_names[i]] for i in variable_indices]
118
119 # 定义问题类，用于多目标优化
120 class AbsorptionColumnMultiObjectiveProblem(ElementwiseProblem):
121     def __init__(self, fixed_values, variable_indices, bounds):
122         # 设置变量范围 (bounds)，对应变量的上下限
123         super().__init__(n_var=len(variable_indices), n_obj=2, n_constr=0, xl=
            np.array([b[0] for b in bounds]), xu=np.array([b[1] for b in
            bounds]))
124         self.fixed_values = fixed_values
125         self.variable_indices = variable_indices
126         self.problem = problem_instance
127
128     def _evaluate(self, x, out, *args, **kwargs):
129         # 更新优化变量到全局变量空间
130         full_x = self.fixed_values.copy()
131         for i, idx in enumerate(self.variable_indices):
132             full_x[idx] = x[i]
133
134         # 获取各个性能指标
135         cost_factor, safety_factor, _ = self.problem.evaluate(full_x)
136
137         # 成本最小化，安全因子最大化（因此取负）
138         out["F"] = [cost_factor, -safety_factor]
139
140 # 创建问题实例，并传入动态生成的bounds
141 problem = AbsorptionColumnMultiObjectiveProblem(fixed_values, variable_indices
    , bounds)
142
143 # 继续使用NSGA-II算法进行优化
144 algorithm = NSGA2(
145     pop_size=100,

```

```
146     sampling=FloatRandomSampling(),
147     crossover=SimulatedBinaryCrossover(prob=0.9, eta=15),
148     mutation=PolynomialMutation(eta=20),
149     eliminate_duplicates=True
150 )
151
152 # 使用明确的终止条件
153 termination = get_termination("n_gen", 100)
154
155 # 进行优化
156 res = minimize(problem,
157               algorithm,
158               termination, # 明确的终止条件
159               seed=1,
160               verbose=True)
161
162 # 输出帕累托前沿解集及其目标值
163 print("帕累托前沿解集:")
164 for solution in res.X:
165     print(solution)
166
167 print("对应的目标值（成本，安全因子）:")
168 for objective in res.F:
169     print(objective)
170
171 # 可视化帕累托前沿 - 二维图
172 plt.figure(figsize=(10, 7))
173 plt.scatter(res.F[:, 0], -res.F[:, 1], c='r', marker='o')
174
175 # 设置轴标签
176 plt.xlabel('Cost Factor', fontsize=14)
177 plt.ylabel('Safety Factor', fontsize=14)
178 plt.title('Pareto Front 2D Visualization', fontsize=16)
179 plt.show()
```


附录 B 支撑材料文件列表

名称	修改日期	类型	大小
code	2024/9/10 15:52	文件夹	
figure	2024/9/10 16:22	文件夹	
src	2024/9/12 21:59	文件夹	
task	2024/9/12 17:41	文件夹	
clean_latex_auxiliary_files.py	2024/9/3 21:48	PY 文件	1 KB
contents.py	2024/9/12 12:58	PY 文件	3 KB
contents.txt	2024/9/12 21:57	TXT 文件	48 KB
cqutthesis.cls	2024/9/12 13:59	CLS 文件	37 KB
cqutthesis.dtx	2024/9/12 13:59	DTX 文件	99 KB
cqutthesis.ins	2024/9/10 17:07	INS 文件	3 KB
file_list.py	2024/9/10 22:57	PY 文件	1 KB
file_list.txt	2024/9/11 11:41	TXT 文件	1 KB
main.lodf	2024/9/12 21:56	LODF 文件	0 KB
main.pdf	2024/9/12 21:56	Microsoft Edge ...	408 KB
main.tex	2024/9/12 22:00	TEX 文件	3 KB
mycfg.sty	2023/5/21 13:50	STY 文件	1 KB
psheader.txt	2024/9/12 12:52	TXT 文件	1 KB
thesisbib.bib	2024/9/12 21:53	BibTeX 源文件	3 KB