

阵列式集成精馏系统分布式容错控制器设计

李欣¹, 薄翠梅^{1,4†}, 张贺¹, 李俊¹, 乔旭^{2,4}, 高福荣³

(1. 南京工业大学 电气工程与控制科学学院, 南京 211816; 2. 南京工业大学 材料化学工程
国家重点实验室, 南京 211816; 3. 香港科技大学 化学及生物工程学系, 香港 999077;
4. 工业互联网+危化品安全生产应急管理部重点实验室, 南京 211816)

摘要: 针对现有化工装备规模大、集成度高, 潜在安全风险高的难题, 提出一种阵列式集成精馏系统分布式容错控制方法. 该方法通过簇单元之间网络拓扑结构, 实现各单元协作与信息共享, 并应用动态事件触发机制避免系统通讯资源浪费问题. 此外, 针对执行器故障提出虚拟执行器故障补偿机制, 给出系统一致性稳定的充分条件. 利用仿真工具箱求解阵列式集成精馏系统容错控制器增益, 结合传统分布式控制与分散控制理论, 对系统控制方法进行仿真与对比论证, 证明了所提出分布式容错控制算法的有效性, 其可以显著提高系统通讯资源节省率.

关键词: 阵列式精馏过程; 多智能体; 稳定性分析; 分布式容错控制; 事件触发控制

中图分类号: TP273 文献标志码: A

DOI: 10.13195/j.kzyjc.2025.0231

引用格式: 李欣, 薄翠梅, 张贺, 等. 阵列式集成精馏系统分布式容错控制器设计 [J]. 控制与决策, 2025, 40(10): 2969-2977.

Distributed fault-tolerant controller design for array-integrated distillation system

LI Xin¹, BO Cui-mei^{1,4†}, ZHANG He¹, LI Jun¹, QIAO Xu^{2,4}, GAO Fu-rong³

(1. School of Electrical Engineering and Control Science, Nanjing University of Technology, Nanjing 211816, China; 2. State Key Laboratory of Chemical Engineering of Materials, Nanjing University of Technology, Nanjing 211816, China; 3. Department of Chemical and Biological Engineering, The Hong Kong University of Science and Technology, Hong Kong 999077, China; 4. Key Laboratory of Industrial Internet + Safety Production of Hazardous Chemical of Ministry of Emergency Management, Nanjing 211816, China)

Abstract: In response to the challenges posed by the large scale, high integration, and high potential safety risks of existing chemical equipment, a distributed fault-tolerant control method for array-integrated distillation systems is proposed. This method leverages the network topology among cluster units to enable collaboration and information sharing among the units. Moreover, a dynamic event-triggered mechanism is applied to avoid the waste of system communication resources. Additionally, the virtual actuator fault compensation mechanism is proposed to mitigate the impact of actuator faults, and sufficient conditions under actuator faults for the consistency and stability of system are demonstrated. The fault-tolerant controller gain of the array-integrated distillation system is solved using a simulation toolbox. Finally, by integrating traditional distributed control and decentralized control theories, simulations and comparative analyses of the system control method are conducted. The results prove that the proposed distributed fault-tolerant control algorithm is effective, and can significantly enhance the saving rate of system communication resources.

Keywords: array-type distillation process; multi-agent; stability analysis; distributed fault-tolerant control; event-triggered control

0 引言

多智能体系统 (multi-agent systems, MAS) 是由多个自主或半自主的智能体组成的复杂系统, 这些

智能体通过分布式控制、相互通信和协同合作来共同完成任务. 近年来, 多智能体系统在无人机^[1]、智能电网^[2]、机器人^[3]等领域得到了广泛应用. MAS 由

收稿日期: 2025-03-05; 录用日期: 2025-05-22.

基金项目: 国家自然科学基金项目 (62333010, 62203214, 62173178); 江苏省重点研发计划重大科技示范项目 (SBE2021710043); 江苏省高等学校自然科学研究项目 (22KJB120003).

†通信作者. E-mail: Lj_bcm@163.com.

于其自主性、交互性、协作性、鲁棒性,显著提升了系统的稳定性、容错性和执行效率.然而,尽管MAS在上述领域取得了显著进展,但其在化工领域的应用仍相对稀缺.

目前,传统化学工业装备仍具有体积大、材料量大、能量密度高的特点,导致化工装置总体呈现出“高塔林立、釜罐成群”的固有业态,使得过程安全风险难以根本消除.近年来,因安全误操作、控制系统故障等引发的安全事故屡见不鲜^[4].针对化学工艺复杂化、装置大型化、本质安全化水平不高等现象,乔旭等^[5]从本质安全角度提出了范厘米阵列式集成精馏技术,大幅降低了微化工技术在化工行业应用的局限性.阵列式集成单元技术平稳高效运行的关键在于各单元及其装置整体精准检测与控制,保证单元的物料均匀性、状态一致性以及在少数单元发生故障时的控制精度和鲁棒性.针对分布式控制过程故障问题,文献[6]提出一种具有不确定执行器故障的随机多智能体系统自适应固定时间一致性控制算法,并成功应用于多电机系统.文献[7-8]提出一种分布式自适应容错共识控制技术.文献[9]针对不同系统执行器与传感器故障问题,解决了系统状态一致性问题.

本文借鉴多智能体在无人机等领域成果应用案

例,将多智能体一致性控制引入所提出的阵列式精馏集成装置对象,一个精馏簇单元视为一个智能体,通过多智能体拓扑结构实现若干个精馏簇单元之间的协作与信息共享,深入研究阵列式集成精馏系统分布式容错控制,旨在解决装置多单元物料均匀性、状态一致性问题.针对控制系统存在执行器故障的问题,进一步提出容错故障补偿机制,设计执行器故障函数,基于仿真工具箱给出系统快速确定容错控制器增益的矩阵不等式算法,推导其在执行器进料故障下多智能体系统实现状态一致性的充分条件,解决精馏单元中由于执行器进料故障引起的各单元状态变量不一致问题.

1 问题描述

1.1 阵列式集成精馏系统

本文引入一种阵列式集成精馏模型,如图1所示.该模型由若干个精馏簇单元集成^[10],各个簇单元在物理空间上呈相互独立、均匀分布,无质量传递及径向物料流动现象,每个精馏单元都具有相同的分离性能.虽然单元结构相同,但生产中由于各精馏单元进料波动或初始参数不一致,造成产物浓度不一致,无法满足产品质量要求.为提升产品质量,本文考虑分布式控制策略协同调控各精馏单元,以确保产物浓度一致.

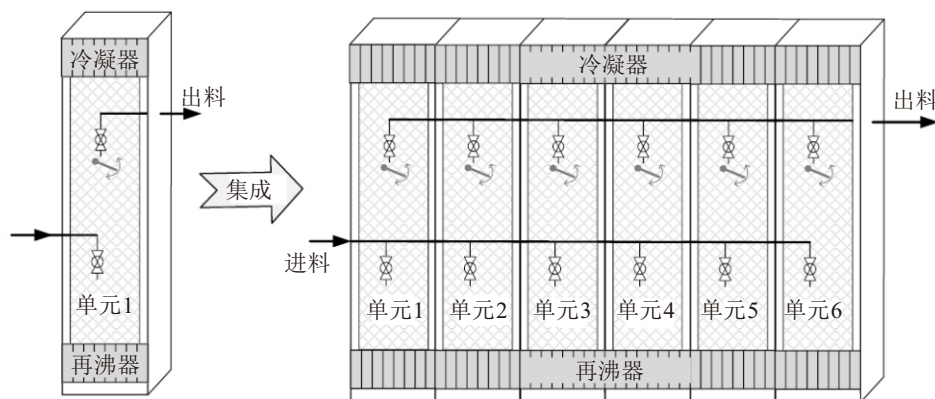


图1 阵列式集成精馏系统模型

在生产过程中,为实现其他单元的协同运行,需要选定一个基准簇作为系统参考精馏单元,通过参考其单元状态数值指标调控其他单元,指导其他单元生产,使得整个系统协同一致.因此,将提供参考系的精馏簇单元定义为基准簇(reference cluster),需与基准簇保持同步的其他精馏簇单元簇定义为协同簇(synergistic cluster).假设系统有1个基准簇与 N 个协同簇组成,基准簇状态方程可写为

$$\begin{aligned} \dot{x}_0(t) &= A_L x_0(t), \\ y_0(t) &= C x_0(t); \end{aligned} \quad (1)$$

协同簇的状态方程^[11-12]可写为

$$\begin{aligned} \dot{x}_i(t) &= A x_i(t) + B u_i(t), \\ y_i(t) &= C x_i(t). \end{aligned} \quad (2)$$

其中:对于 $i \in 1, 2, \dots, N$, A 、 B 、 C 、 A_L 为维度合适的状态矩阵; $x_0(t)$ 、 $x_i(t)$ 分别为 t 时刻下基准簇和第 i 个协同簇的状态.

注1 考虑到协同簇开环系统状态可能不稳定,为了确保基准簇能够提供稳定的参考信号,根据矩阵 A 的正实部特征值特性对基准簇进行局部控制,初步配置基准簇极点,使得系统输出满足动态需求,

通过线性矩阵求解使得基准簇状态矩阵转变为符合要求的 Hurwitz 矩阵, 形成新矩阵 $A_L = A - BK_c$.

1.2 分布式控制协议模型

传统精馏过程控制通过独立调节系统回流量和液相蒸汽量等变量状态, 实现流出组分浓度一致. 但传统分散控制在多智能体系统中存在信息交互慢、容错性差等问题, 因此本文提出考虑拓扑结构, 在阵列式精馏系统中每个簇还可以通过信号流形式与相邻的簇单元或其他单元进行信息交流. 这种信号流的方向根据拓扑网络结构确定, 每个单元对应一个节点, 节点之间的通信链路则对应图中的边. 通过这种图论模型, 每个精馏簇单元能够自主地感知自身状态, 并与其他单元进行信息交互和协调, 从而实现整个系统的协同控制.

在本文中, 分布式控制采用主从结构, 使用图 $\mathcal{G} = \mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{A})$ 表示网络拓扑. 其中: \mathcal{V} 为网络中所有非空节点的集合, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ 为节点边的集合^[13], \mathcal{A} 为描述协同簇节点与边之间关系的矩阵, 也可称为邻接矩阵. $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N \times N}$, 有向图中, i 节点指向 j 节点, 表示节点 i 有信息流向节点 j , 并且如果 $(j, i) \in \mathcal{E}$, 则 $a_{ij} > 0$, 否则 $a_{ij} = 0$. 这里不允许存在自边 (i, i) , 因此 $a_{ii} = 0$. 基准簇从顶点 i 的入度矩阵定义为 $D^m = \text{diag}(d_i^m)$, 此外从顶点 i 到顶点 j 的路径是一个有序的顶点序列, 使得每个直接的顶点对都是一条边. 因此有向图的拉普拉斯算子为 $L_G = D^m - \mathcal{A}$. 阵列式集成精馏系统的通讯拓扑如图 2 所示, 从 RC0 到 SC1 的箭头表示 RC0 向 SC1 传输信息.

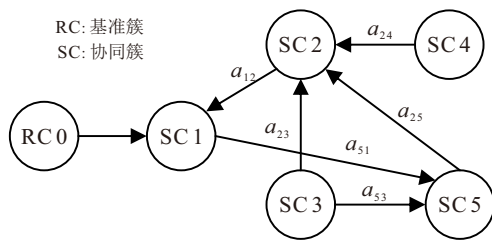


图2 多智能体系统拓扑结构

根据文献 [14-15] 的描述, 常规分布式控制协议如下:

$$u_i(t) = K \left(\sum_{j \in N_i} a_{ij}(x_i - x_j) + d_i(x_i - x_0) \right), \quad (3)$$

其中对于所有 $i, j \in \mathcal{V}$, K 为系统控制器增益. 该协议通过簇单元之间持续通信更新控制器增益, 但实际上所有智能体共享网络带宽有限, 持续传输状态信息会导致网络拥塞与控制指令频繁切换等问题.

此外, 常规控制协议通常只考虑到执行器正常工作状况, 忽略了执行器故障下系统是否能维持预

期控制性能. 由于阵列式集成精馏系统内部存在多个执行器, 结构较为复杂, 常规控制器难以实现故障快速响应与容错控制, 本文提出一项执行器故障下基于事件触发的分布式容错控制协议以弥补常规分布式控制器的不足.

2 执行器故障下分布式系统容错控制

2.1 执行器故障矩阵设计

本研究针对各个簇单元存在执行器阀门故障. 假设每个簇单元存在一个执行器有效矩阵, 此处用对角矩阵 Γ_i 表示 $B_i^f = B\Gamma_i$, B_i^f 为协同簇 i 带有执行器故障的系统矩阵, $\Gamma_i = (\Pi_i^1, \Pi_i^2, \dots, \Pi_i^n)$, $\Pi_i^n \in [0, 1]$ 表示协同簇 i 第 n 个输入的有效性, 当 $\Pi_i^n \in (0, 1)$ 时表示第 i 个执行器部分失效; 当执行器处于正常状态时, $\Pi_i^n = 1$; 相反当执行器完全故障时, $\Pi_i^n = 0$. 此处假设 Π_i^n 值是可估计的且规定当系统某一执行器完全故障失效, 仍有冗余执行器作为间接控制执行器替补维持系统正常运作.

2.2 基于事件触发的分布式控制协议设计

执行器故障下, 协同簇表达式可写为

$$\begin{aligned} \dot{x}_i^f(t) &= Ax_i^f(t) + B_i^f u_i^f(t), \quad i = 1, 2, \dots, N, \\ y_i^f(t) &= Cx_i^f(t). \end{aligned} \quad (4)$$

其中: x_i^f 表示第 i 个簇故障下的系统状态, y_i^f 表示第 i 个簇故障下的系统输出状态, u_i^f 表示第 i 个簇故障下的系统状态.

1) 动态事件触发机制.

为了减少故障下系统控制指令传输频繁问题, 节约系统通讯资源, 引入一项动态事件触发机制. 设协同簇事件触发状态与系统观测器误差为 $\gamma_i = \tilde{x}_i - \hat{x}_i$, 动态事件触发机制描述如下所示:

$$\begin{aligned} t_{k+1}^i &= \inf\{t | t > t_k^i, h_i(t) \geq 0\}, \\ h_i(t) &= -\omega_i(t) + \theta_i(\gamma_i^T(t)T_1\gamma_i(t) - \varphi_i^T(t)T_2\varphi_i(t)), \\ \dot{\omega}_i(t) &= -\beta\omega_i(t) + \alpha_i(\gamma_i^T(t)T_1\gamma_i(t) - \varphi_i^T(t)T_2\varphi_i(t)). \end{aligned} \quad (5)$$

其中: $h_i(t)$ 为第 i 个协同簇对应动态事件触发函数; θ_i 为系统未知变化参数, 规定 $\theta_i \geq \frac{1 - \alpha_i}{\beta_i}$; ω 为辅助动态变量, 且 $\omega_i(0) > 0$; T_1, T_2 为正定误差矩阵; α, β 为正常数. 该通信机制对不满足要求的系统采样数据进行筛选, 规定事件触发函数 $h_i(t) \geq 0$ 时, 系统进行通信交互; 否则, 数据被筛选丢包.

2) 分布式控制器设计.

考虑系统存在执行器故障条件下, 为实现系统的各个协同簇与系统基准簇的产品质量输出一致性, 基于事件触发的阵列式精馏系统分布式控制协议可

以描述为

$$\begin{aligned} u_i^f(t) &= K\varphi_i^f(t), \\ \varphi_i^f(t) &= \sum_{j \in N_i} a_{ij}(\tilde{x}_j(t_k^i) - \tilde{x}_i^f(t_k^i)) + \\ &\quad d_i(x_0(t) - \tilde{x}_i^f(t_k^i)). \end{aligned} \quad (6)$$

其中: $i, j \in \mathcal{V}$; K 为待设计控制器增益; t_k^i ($k \in N^+$)为第 i 个协同簇事件触发时刻, 由动态事件触发机制 (5) 产生; $\tilde{x}_i(t_k^i)$ 为事件触发时刻下第 i 个协同簇状态更新.

3) 系统观测器设计.

为观测系统每个簇单元的状态, 建立故障下分布式状态观测器方程, 每个观测器被视作单一的监测器实时监测每个精馏单元的状态, 观测器表达式为

$$\begin{aligned} \dot{\hat{x}}_i^f(t) &= A_L \hat{x}_i^f(t) + B u_i^f(t) + G(y_i(t) - \hat{y}_i^f(t)), \\ \hat{y}_i^f(t) &= C \hat{x}_i^f(t). \end{aligned} \quad (7)$$

其中: G 为观测误差增益矩阵, \hat{x}_i^f 为故障下由观测器生成的第 i 个协同簇系统状态, \hat{y}_i 为系统网络第 i 个观测器的输出估计.

此外, 当系统启动动态事件触发机制 (5) 时, 协同簇将根据观测器状态 (7) 对系统进行状态更新, 更新状态表达式可写为

$$\begin{aligned} \tilde{x}_i(t_k^i) &= \hat{x}_i(t_k^i), \\ \dot{\tilde{x}}_i(t) &= A \tilde{x}_i(t), \quad t \in (t_k^i, t_{k+1}^i), \end{aligned} \quad (8)$$

其中 \hat{x}_i 是由观测器生成的系统状态. 观测器表达式如式 (7) 所示, 当系统满足事件触发函数条件时, 将进行式 (8) 中第 i 个协同簇状态的更新.

2.3 虚拟容错补偿机制 (VAFC)

在故障条件下, 需借助虚拟执行器故障补偿机制对故障单元进行及时补偿. 当第 i 个簇检测到系统故障时, VAFC 机制通过误差 Δx_i 对系统故障部分进行计算并将值返回控制器. 设 VAFC 处于激活状态下, 系统控制器输入及补偿值状态可表示为

$$\begin{aligned} u_i^f &= R_i^f u_i + K_i^{va} \Delta x_i, \quad R_i^f = (B_i^f)^T B, \\ \Delta \dot{x}_i &= (A - B_i^f K_i^{va}) x_i + (B - B_i^f R_i^f) u_i^f. \end{aligned} \quad (9)$$

其中: K_i^{va} 为虚拟故障补偿增益, R_i^{va} 为考虑故障影响的系统控制输入变换矩阵. 此处 Δx_i 表示虚拟故障机制的内部状态, 即正常系统与执行器故障下系统补偿值. 对于虚拟故障补偿增益, 若矩阵 B 满足条件 $\text{Im}(B) = \text{Im}(B_i^f)$, 则易得 $B - B_i^f R_i^f = 0$. 通过设计的系统补偿值更新方程, 重新确定执行器故障下系统的控制器内部参数.

当 VAFC 被激活时, 系统中基准簇与协同簇的一致性误差为 $z_i^f = x_i^f + \Delta x_i - x_0$, 协同簇实际状

态与观测估计器间误差定义为 $e_i^f = \hat{x}_i^f - \Delta x_i - x_i^f$, 事件触发状态与系统观测器误差为 $\gamma_i(t) = \tilde{x}_i^f - \hat{x}_i^f$, 因此协同簇闭环系统状态可表示为

$$\begin{aligned} \dot{x}(t) &= (I_N \otimes A)x(t) - \\ &\quad (L_G \otimes B^f K)(z(t) + e(t) + \gamma(t)). \end{aligned} \quad (10)$$

其中: $B^f = \text{diag}(B_1^f, \dots, B_N^f)$, $z(t) = [z_1^T(t), \dots, z_N^T(t)]^T$. 此外其他误差向量 $e(t)$ 、 $\gamma(t)$ 、 $\varphi^f(t)$ 与 $z(t)$ 定义类似.

3 主要定理

假设 1 矩阵 (A_L, B) 和 (A_L, C) 是可控和可观测的, 由矩阵对 (A_L, B^f) 表示的故障过程也是可观测的.

假设 2 通信拓扑是固定有向的, 增广图 G 包含一个生成树, 基准簇是其根节点.

定义 1 实现协同簇与基准簇的一致性共识, 可以将系统一致性共识表示为

$$\lim_{k \rightarrow +\infty} \|x_i(t) - x_0(t)\| = 0, \quad \forall i \in \{i, \dots, N\}. \quad (11)$$

定理 1 假设 1 与假设 2 成立的条件下, 确定分布式容错控制器 (6) 与补偿机制 (9) 增益, 确保故障下闭环系统 (10) 满足求解算法 1 (如表 1 所示), 从而实现基准簇与协同簇产物浓度的一致性控制.

根据定理 1, 针对进料执行器故障设计总体控制框架如图 3 所示.

4 实验结果与分析

4.1 阵列式集成精馏系统描述

实验仿真中, 设定阵列式精馏系统共 6 个簇, 内部具备 1 个基准簇和 5 个协同簇, 每个精馏簇单元同构, 第 i 个簇表达式系统状态空间方程如下所示:

$$\begin{aligned} &[\Delta \dot{x}_i^D \quad \Delta \dot{x}_i^t \quad \Delta \dot{x}_i^B \quad \Delta \dot{x}_i^{30}]^T = \\ &\begin{bmatrix} 0 & 1 & 0 & 0 \\ -0.6303 & 1.6070 & -0.3926 & 0.4068 \\ 0 & 0 & 0 & 1 \\ 3.9724 & -4.0295 & 4.5448 & -0.5198 \end{bmatrix} \times \\ &\begin{bmatrix} \Delta x_i^D \\ \Delta x_i^t \\ \Delta x_i^B \\ \Delta x_i^{30} \end{bmatrix} + \begin{bmatrix} 0.18656 & -0.0003 \\ 0.0374 & 0.0729 \\ -0.2633 & 0.1563 \\ -0.2776 & -0.2219 \end{bmatrix} \begin{bmatrix} \Delta L_i^D \\ \Delta V_i^m \end{bmatrix}, \\ &\begin{bmatrix} y_i^D \\ y_i^B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x_i^D \\ \Delta x_i^t \\ \Delta x_i^B \\ \Delta x_i^{30} \end{bmatrix}. \end{aligned} \quad (12)$$

此处选择第 i 个协同簇的顶部蒸汽增量 ΔV_i^m 和

表1 阵列式精馏系统分布式容错控制算法

算法1 根据式(4)选择控制器增益.

step 1: 获取进料故障下系统矩阵 B_i^f .

step 2: 根据故障矩阵 B_i^f 求解故障补偿增益 K_i^{va} .

step 3: 根据不等式 $\Omega < 0, Q \geq 0$ 求解误差矩阵 T_1, T_2 .

$$\Omega = \begin{bmatrix} \Omega_1 & \Omega_{12} & \Omega_{13} & \Omega_{14} \\ * & \Omega_2 & \Omega_{23} & \Omega_{24} \\ * & * & \Omega_3 & \Omega_{34} \\ * & * & * & \Omega_4 \end{bmatrix} < 0, Q = \begin{bmatrix} Q_1 & 0 & Q_{13} \\ * & Q_2 & Q_{23} \\ * & * & Q_3 \end{bmatrix} \geq 0,$$

$$\Omega_{11} = Q_1(A_L - L_G B K) + (A_L - L_G B K)^T Q_1^T - Q_{13}(B - B^f R^{va}) K L_G - (Q_{13}(B - B^f R^{va}) K L_G)^T + L_G^2 T_2,$$

$$\Omega_{12} = -Q_1 L_G B K - Q_{23}(B - B^f R^{va}) K L_G - (Q_{23}(B - B^f R^{va}) K L_G)^T + L_G^2 T_2,$$

$$\Omega_{13} = -(Q_3(B - B^f R^{va}) K L_G)^T + Q_{13}(B - B^f R^{va}) K L_G - (Q_{13}(A_L - B^f K^{va}))^T + Q_{13}(A_L - B^f K^{va}),$$

$$\Omega_{14} = -(Q_1 L_G B K)^T + Q_{13}(B - B^f R^{va}) K L_G + L_G^2 T_2;$$

$$\Omega_{22} = (Q_2(A_L - G C) + (A_L - G C)^T Q_2) + Q_{23}(B - B^f R^{va}) K L_G + (Q_{23}(B - B^f R^{va}))^T + L_G^2 T_2,$$

$$\Omega_{23} = Q_2 L_G C - (Q_3(B - B^f R^{va}) K L_G)^T - Q_{23}(A_L - B^f K^{va}) + (Q_{23}(A_L + L_G C))^T,$$

$$\Omega_{24} = Q_{23}(B - B^f R^{va}) K L_G + L_G^2 T_2;$$

$$\Omega_{33} = -Q_3(A_L - B^f K^{va}) + (Q_3(A_L - B^f K^{va}))^T - Q_3 G C - (Q_3 G C)^T,$$

$$\Omega_{34} = Q_3(B - B^f R^{va}) K L_G - Q_{13} B K L_G;$$

$$\Omega_{44} = L_G^2 T_2 - T_1.$$

step 4: 如果以上不等式成立, 则返回VAFC增益 K_i^{va} 、控制器增益 K 、观测器增益 G ; 否则, 宣布无不可行解.

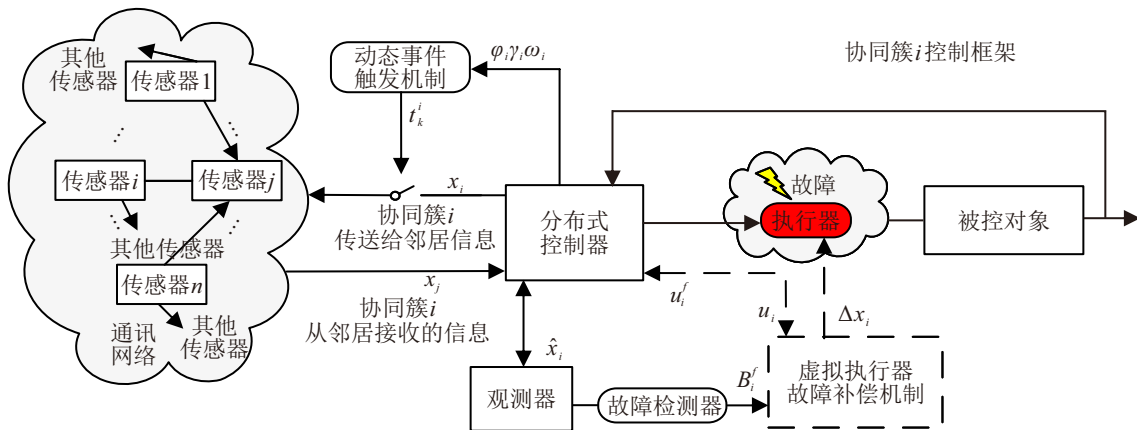


图3 阵列式精馏系统控制结构

单元内部回流增量 ΔL_i^D 作为控制输入, $u_i = [\Delta V_i^m \ \Delta L_i^D]^T$; $x_i = [\Delta x_i^D \ \Delta x_i^t \ \Delta x_i^B \ \Delta x_i^{30}]^T$ 表示第 i 簇单元 4 个状态变量, 分别为单元顶部组分浓度增量、提馏段浓度增量、底部输出浓度增量、第 30 块塔板浓度增量; 在仿真过程中令单元顶部组分浓度 y_i^D 与单元底部组分浓度 y_i^B 作为系统的输出变量, $y_i = [y_i^D \ y_i^B]^T$; 给定系统基准簇初始状态为 $x_0 = [0.13 \ -0.24 \ 0.32 \ 0.15]^T$, 初始控制输入状态为 $u_0 = [0.9 \ 30]^T$.

4.2 多智能体容错控制器参数设计

考虑系统通讯拓扑结构, 在进行一致性控制之前需要对基准簇进行单独反馈控制设计, 配置极点设计为 $pole = [-1, -0.9, -2, -3]$, 使得基准簇状

态实现稳定. 基准簇系统反馈控制增益计算如下:

$$K_c = \begin{bmatrix} 728.72 & -3007.49 & -169.25 & 1154.14 \\ 307.69 & -1344.87 & 160.38 & -581.84 \end{bmatrix}. \tag{13}$$

获取基准簇的反馈增益后, 令 $K = B^T Q_1$, 并将其代入容错控制算法 1, 计算系统矩阵所得分布式控制器增益及观测器增益分别为

$$K = \begin{bmatrix} 0.9865 & -0.0961 & 0.3919 & 0.2412 \\ -3.662 & 1.436 & -1.6459 & -0.7108 \end{bmatrix}, \tag{14}$$

$$G = \begin{bmatrix} -4.9628 & 6.6631 & 23.3833 & -15.6655 \\ -1.1688 & 1.3194 & 4.9628 & -2.9020 \end{bmatrix}^T. \tag{15}$$

仿真故障设置如下: 当 $5\text{ h} < t < 15\text{ h}$ 时, 协同簇 1 发生故障, 故障矩阵为 $\Gamma_1 = \text{diag}(0.6, 0)$; 当 $15\text{ h} < t < 25\text{ h}$ 时, 协同簇 1 仍故障, 协同簇 2 发生故障, 故障矩阵为 $\Gamma_2 = \text{diag}(0.5, 1)$; 当 $25\text{ h} < t < 35\text{ h}$ 时, 协同簇 1 和协同簇 2 恢复正常状态, 协同簇 3 和协同簇 5 发生故障, 其故障矩阵分别为 $\Gamma_3 = \text{diag}(1, 0.8)$, $\Gamma_5 = \text{diag}(0.6, 0.2)$; 当 $35\text{ h} < t < 45\text{ h}$ 时, 协同簇 4 发生故障, $\Gamma_4 = \text{diag}(0.6, 0.8)$, 其余协同簇处于正常的工作状态; 当 $t > 45\text{ h}$ 时, 所有协同簇都处于正常的工作状态。

在事件触发机制变量初始设计中, 系统事件触发机制函数 $h_i(t)$ 中动态变量 $\omega(0) = 10$, $\theta = 0.5$, $\alpha = \beta = 1$. 在仿真过程中, 获取系统相关增益及每个簇单元故障矩阵, 根据工具箱计算给出不等式并返回对应的触发误差矩阵 T_1 和 T_2 .

4.3 执行器故障下系统输出一致性仿真测试

当智能体执行器发生故障时, 系统产品出料浓度可能出现不同程度的波动, 其波动幅度与故障程度密切相关. 当系统发生执行器故障时, 在故障补偿机制作用下, 系统输出仿真如图 4 所示. 可以观察到, 虽然出现了小幅度波动, 但在该机制的作用下系统单元顶部出料输出浓度 y_1 始终保持在 99% 以上, 输出变量均与一致跟随基准簇变化。

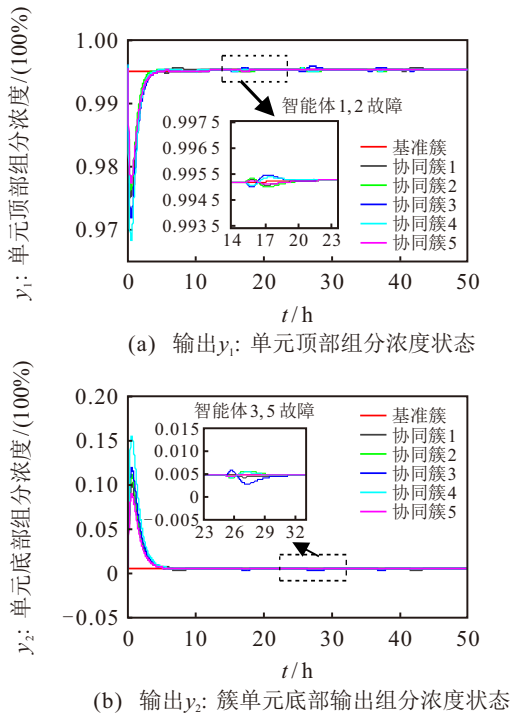


图4 分布式容错控制下系统输出状态

图 5 展示了阵列式集成精馏系统在故障下系统实际状态与观测器间的误差响应曲线. 如图 5(a) 所示, 以协同簇 1 为例, 当系统在 5 h 和 15 h 时检测到

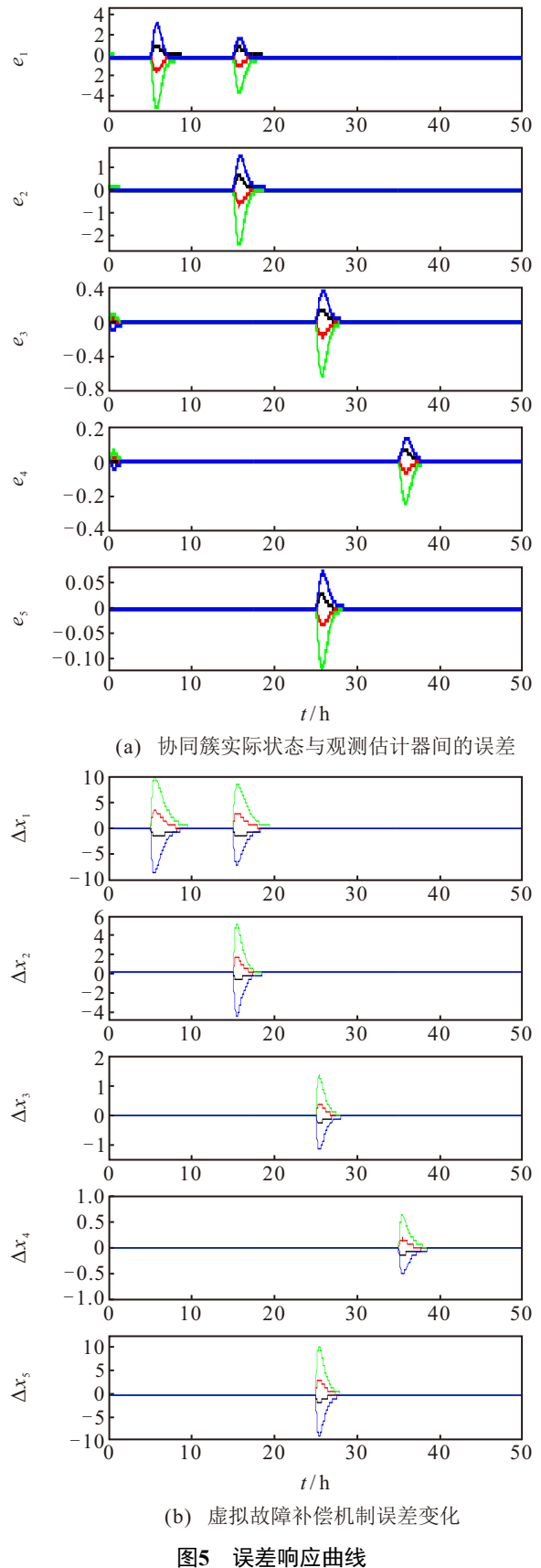


图5 误差响应曲线

簇 1 出现故障, VAFC 机制会向系统执行器发送一个误差补偿值, 如图 5(b) 所示. 此后当其他协同簇发生故障时, 依次通过故障检测器评估系统故障状态, 进行系统补偿. 这表明即使在故障情况下, 通过引入补偿值, 系统的容错性与鲁棒性也会显著增强, 确保

了各单元稳定持续运行.

4.4 改进分布式容错控制算法对比测试分析

为了验证该容错控制算法的优势,从控制性能与系统信息传输率两方面阐述所提出事件触发机制的优势.聚焦协同簇1发生故障状况,模拟系统从启动到15 h内,研究以下3种情况下每个协同簇状态响应状况.

情况1 分散控制算法模拟测试分析.

根据文献 [16] 提出的分散式控制协议,在仿真过程中进行控制性能对比,系统状态响应如图6所示.由于该协议缺乏内部通信拓扑结构,采用无状态互联项分散式控制方式.具体而言,当协同簇1发生故障时,其邻居单元无法感知该故障,导致其状态保持不变,而协同簇1的状态会因此发生变化.该状况下系统输出结果图7所示.尽管在故障情况下系统表现出较小的波动,表明每个簇单元基本能够实现一致的稳定性,但系统超调较大,单元输出浓度最低

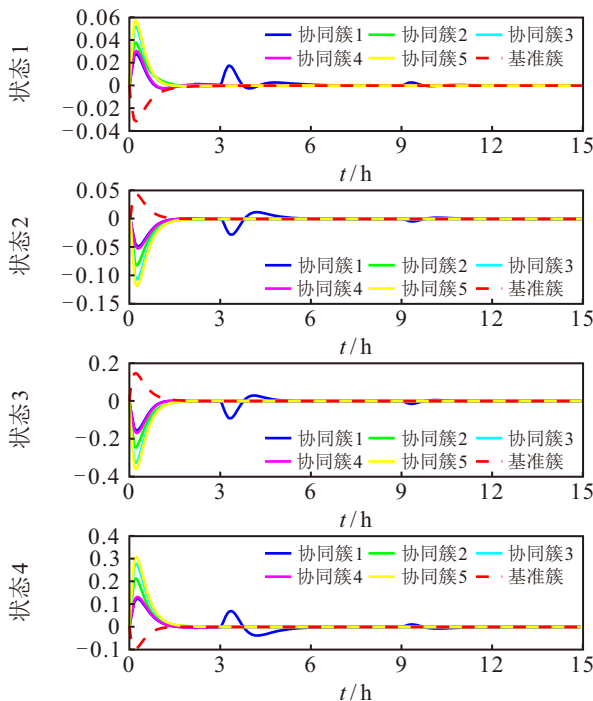


图6 协同簇1故障下跟随变化状况(情况1)

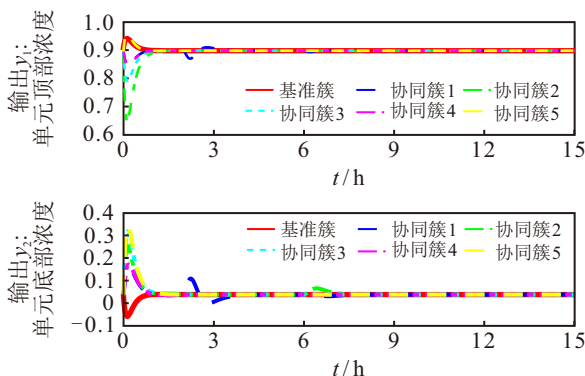


图7 协同簇1故障下系统输出变化状况

至65%.因此该方法下虽然系统能够实现稳定,但系统动态性能和容错性不佳.如果故障持续发生,则各单元状态逐渐偏离一致,导致输出浓度大幅降低.

情况2 传统分布式控制算法模拟测试分析.

文献 [17] 采用传统基于事件触发的分布式控制协议,该协议通过设计事件触发条件减少通讯负担,但故障下故障单元无法将状态信息准确传递给其他协同,致使每个单元接收到信息开始发生偏差.由于无容错机制,各个状态发生信息缺失将进一步放大系统状态差异.采用该方法控制协议下系统状态变化如图8所示.在故障下,每个协同簇单元状态波动很大,当系统状态偏离稳定区域达到一定程度,且无法通过现有控制协议使得系统稳定时,系统的Lyapunov函数将不再满足稳定性条件,系统面临失稳风险.对比图6与图8可发现,无触发机制的情况1总体控制性能仍优于无容错机制设计的分布式事件触发控制.

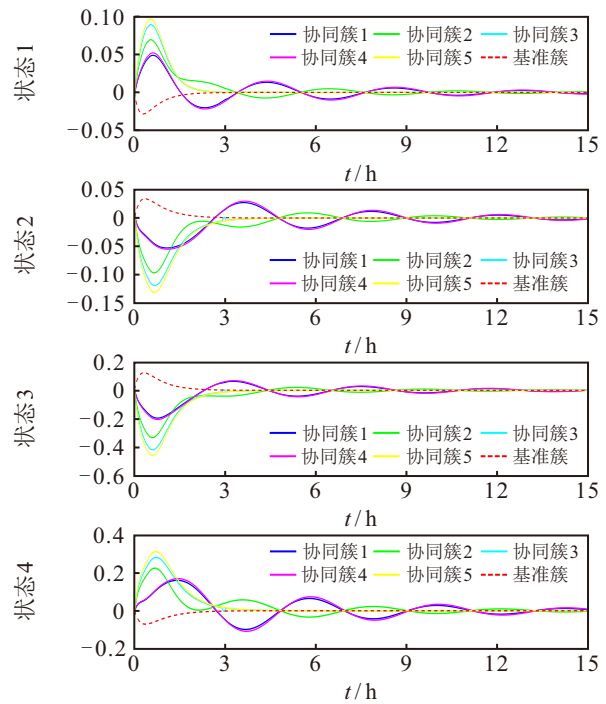


图8 协同簇1故障下跟随变化状况(情况2)

情况3 含容错补偿分布式控制算法模拟测试分析.

采用本文提出的动态事件触发分布式容错机制进行模拟仿真,结果如图9所示.可知,系统在发生执行器故障下状态仍保持一致稳定.当系统发生故障时,协同簇实际状态与观测器之间的误差超出设定阈值,触发系统故障补偿及动态事件触发机制,经过观测器更新系统状态,采样器受到动态事件函数触发进行信息传输,系统传输的VAFC故障补偿值来源于 Δx_i ,矩阵 R_i^{va} 重构控制器参数与权重,调整

控制逻辑, 并通过 Matlab 中 CVX 工具箱优化求解返回更新误差矩阵 T_1 和 T_2 ; 同时, 系统动态触发机制发生更新后的补偿参数返回控制器内部传输给被控对象, 从而对执行器故障进行补偿, 维持塔内物料及动态平衡. 因此, 系统处于之前的稳定一致状态不变, 从而避免了状态波动造成生产中断.

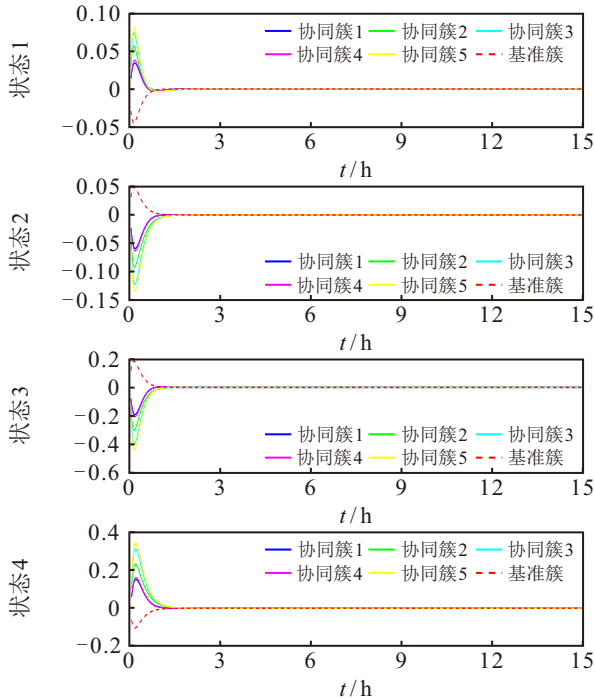


图9 协同簇 1 故障下每个智能体单元状态变化状况

根据以上 3 种控制方法, 采用分散输出反馈控制方案在故障状态下的系统输出浓度如图 7 所示. 对比图 7 与图 9 运行状态, 情况 1、情况 3 下的系统在故障发生后都能够实现稳定运行, 但明显可以发现情况 3 下系统在故障补偿速度和控制精度上均优于情况 1 所示系统. 与常规分布式输出反馈控制方案相比, 采用补偿机制的分散式反馈控制方案能够在一定程度上抑制故障影响.

从系统容错性和可靠性的角度看, 对于存在潜在故障风险阵列式精馏系统而言, 分布式控制策略在应对故障时表现更优, 是比分散式控制策略更合适的选择. 从上述讨论可知, 当系统发生故障时, 采用所提出基于动态事件触发机制的容错控制器, 相较于采用传统事件触发机制控制器, 能够在执行器故障下保证系统运行状态更加平稳, 维持系统质量输出的一致性.

以上仿真实验中, 协同簇 1 故障下总步长设为 1500, 图 10 与图 11 分别统计了情况 2 和情况 3 下系统每个智能体事件触发次数. 给出不同控制方法下系统数据传输特性如表 2 所示. 可以发现, 系统故障下, 情况 2 资源节省近 18.8%. 但由于系统波动明

显, 针对控制出料不一致, 误差不断变大, 触发机制易出现失效. 情况 3 下每个单元发生触发采样机制的次数明显比情况 1 少, 且情况 3 系统输出平稳, 计算可得节省资源率近 40.06%. 实验结果表明引入事件触发机制有效减少了单元间信息流交换频率, 显著提升了系统资源利用率, 避免系统各单元频繁控制指令切换; 同时将本文分布式容错控制算法应用在阵列式集成精馏系统中能够更好地实现一致性稳定, 提高系统容错性能.

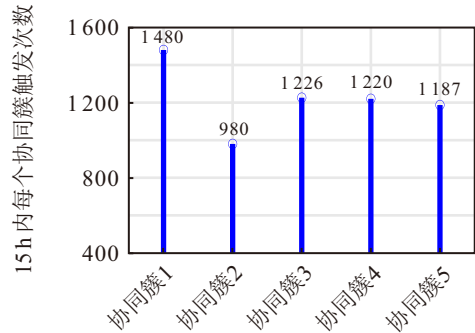


图10 协同簇 1 故障下智能体单元事件触发次数 (情况 2)

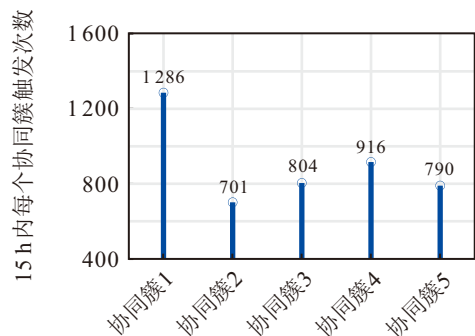


图11 协同簇 1 故障下智能体单元事件触发次数 (情况 3)

表2 不同控制机制下系统数据传输特性

控制方法	分散控制	分布式 ETC 控制	分布式 DETC 容错控制
平均数据传输量	1500	1218	899
平均传输时间/s	1	1.81	3.59
资源节省率/%	0	18.8	40.06

5 结论

针对阵列式集成精馏系统状态不一致及执行器故障问题, 本文提出分布式容错控制方法, 设计了基于动态事件触发机制的容错补偿策略, 实现多智能体系统稳定运行. 理论分析表明, 在满足所提出不等式条件下, 通过仿真工具箱能够实现快速求解容错控制器控制增益, 同时给出系统二次范数下一致性收敛的充分条件. 通过不同控制方法对阵列式精馏系统进行对比实验, 结果表明动态事件触发机制能够显著提高系统通讯利用率, 通讯资源节省 40.06%,

从而减少单元控制指令切换频率.此外,通过虚拟故障补偿机制对产生故障的精馏单元进行故障值补偿,避免了因故障而引起的生产中断,实现了阵列式集成精馏系统各单元产物浓度一致.

参考文献 (References)

- [1] 何云凤, 史贤俊, 卢建华, 等. 基于同步 DMPC 的无人机编队跟踪与避障控制[J]. 控制与决策, 2024, 39(11): 3557-3566.
(He Y F, Shi X J, Lu J H, et al. UAV formation tracking and obstacle avoidance control based on synchronous DMPC[J]. Control and Decision, 2024, 39(11): 3557-3566.)
- [2] 邬晶, 李依宁, 李少远. 分布式电网 CPS 系统数据攻击下的状态估计[J]. 控制与决策, 2016, 31(2): 331-336.
(Wu J, Li Y N, Li S Y. State estimation for distributed cyber-physical power systems under data attacks[J]. Control and Decision, 2016, 31(2): 331-336.)
- [3] Bai Y, Lv Y Q, Zhang J T. Smart mobile robot fleet management based on hierarchical multi-agent deep Q network towards intelligent manufacturing[J]. *Engineering Applications of Artificial Intelligence*, 2023, 124: 106534.
- [4] Wang B, Wu C, Reniers G, et al. The future of hazardous chemical safety in China: Opportunities, problems, challenges and tasks[J]. *Science of The Total Environment*, 2018, 643: 1-11.
- [5] 乔旭, 张竹修. 化工本征安全技术发展路径的思考与探索[J]. 化工进展, 2023, 42(7): 3319-3324.
(Qiao X, Zhang Z X. Consideration and exploration of the development path for inherent safety of chemical engineering[J]. Chemical Industry and Engineering Progress, 2023, 42(7): 3319-3324.)
- [6] Chen S, Chen B, Shi F. Distributed fault-tolerant consensus protocol for fuzzy multi-agent systems[J]. *Circuits, Systems, and Signal Processing*, 2019, 38(2): 611-624.
- [7] Liu C, Jiang B, Zhang K, et al. Distributed fault-tolerant consensus tracking control of multi-agent systems under fixed and switching topologies[J]. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2021, 68(4): 1646-1658.
- [8] Liu C, Yu Z Y, Patton R J. Decentralized fault estimation and distributed fault-tolerant tracking control co-design for sensor faulty multi-agent systems with bidirectional couplings[J]. *International Journal of Control, Automation and Systems*, 2023, 21(3): 810-819.
- [9] Ye Z Y, Cheng Y H, Yu Z Q, et al. Distributed adaptive fault-tolerant control for leaderless/leader-follower multi-agent systems against actuator and sensor faults[J]. *Electronics*, 2023, 12(13): 2924.
- [10] 乔旭, 张竹修, 陈献, 等. 一种簇式精馏塔 [P]. ZL202110144285.6, 2021-06-15.
(Qiao X, Zhang Z X, Chen X, et al. A cluster distillation tower [P]. ZL202110144285.6, 2021-06-15.)
- [11] Shi J T, Chen C, Shen M Q, et al. Actuator fault tolerant region analysis for multi-agent systems using regional pole assignment protocol[J]. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 2023, 70(7): 2450-2454.
- [12] Xu L X, Zhao L N, Ma H J, et al. Event-triggered cooperative output regulation of heterogeneous multi-agent systems with adaptive fault-tolerant control[J]. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 2022, 69(3): 1149-1153.
- [13] Ju Y M, Tian X, Wei G L. Fault tolerant consensus control of multi-agent systems under dynamic event-triggered mechanisms[J]. *ISA Transactions*, 2022, 127: 178-187.
- [14] Wang X H, Ji H B. Leader-follower consensus for a class of nonlinear multi-agent systems[J]. *International Journal of Control, Automation and Systems*, 2012, 10(1): 27-35.
- [15] Wang X, Yang G H. Adaptive reliable coordination control for linear agent networks with intermittent communication constraints[J]. *IEEE Transactions on Control of Network Systems*, 2018, 5(3): 1120-1131.
- [16] Khodabandeh S, Kharrati H, Hashemzadeh F. Control for leader-follower consensus of multi-agent systems with actuator faults using decentralized robust fault-tolerant control[J]. *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, 2021, 45(2): 529-541.
- [17] Yi X L, Liu K, Dimarogonas D V, et al. Dynamic event-triggered and self-triggered control for multi-agent systems[J]. *IEEE Transactions on Automatic Control*, 2019, 64(8): 3300-3307.

作者简介

李欣 (2001-), 女, 硕士生, 研究方向为化工精馏过程及容错控制, E-mail: lx202261106042@njtech.edu.cn;

薄翠梅 (1973-), 女, 教授, 博士, 研究方向为故障诊断及工业过程控制, E-mail: Lj_bcm@163.com;

张贺 (1987-), 男, 副教授, 博士, 研究方向为非线性理论及网络系统安全控制、产品质量精益管控, E-mail: zhanghe@njtech.edu.cn;

李俊 (1972-), 男, 教授, 博士, 研究方向为电力电子技术及其应用、新能源及其控制技术, E-mail: j262402@163.com;

乔旭 (1962-), 男, 教授, 博士, 研究方向为反应精馏分离及耦合集成系统、废弃物资源化与精准治理, E-mail: qct@njtech.edu.cn;

高福荣 (1962-), 男, 教授, 博士, 研究方向为塑料注射成型过程、间歇过程建模及控制, E-mail: kefgao@ust.hk.