

加权有向图社区发现的子系统划分

杨晓峰^{1,2}, 谢 巍¹, 张浪文^{1†}

(1. 华南理工大学 自动化科学与工程学院, 广东 广州 510640; 2. 广州市标准化研究院, 广东 广州 510110)

摘要: 提出一种基于加权有向图的社区发现子系统划分方法, 并应用于分布式状态估计设计. 针对一类复杂非线性系统, 构建考虑连接边强度的加权有向图, 引入社区发现算法将复杂非线性系统划分成多个子系统. 同时考虑子系统之间连接边的数量和有向图顶点之间的连接强度, 使得划分得到的子系统内部关联较强, 而子系统之间的耦合强度较弱. 针对划分得到的子系统, 设计基于信息交互的分布式滚动时域估计算法, 并与已有的子系统划分方法对比, 在相同的状态估计设定下, 所提出的子系统划分方法能够有效提高状态估计的性能.

关键词: 子系统划分; 滚动时域估计; 社区发现算法; 加权有向图

引用格式: 杨晓峰, 谢巍, 张浪文. 加权有向图社区发现的子系统划分. 控制理论与应用, 2020, 37(9): 1923 – 1930
DOI: 10.7641/CTA.2020.90632

Weighted directed graph based community detection for subsystem decomposition

YANG Xiao-feng^{1,2}, XIE Wei¹, ZHANG Lang-wen^{1†}

(1. College of Automation Science and Technology, South China University of Technology, Guangzhou Guangdong 510640, China;
2. Guangzhou Institute of Standardization, Guangzhou Guangdong 510110, China)

Abstract: This paper presents a subsystem decomposition method based on community structure detection with weighted directed graph. The configured subsystems are used for the design of distributed state estimation. For a complex nonlinear system, a weighted digraph considering the strength of the connected edge is constructed, and a community structure detection is introduced to divide a complex nonlinear system into several subsystems. The proposed subsystem decomposition method takes both the number of connecting edges between subsystems and the connection strength of the vertex of the digraph into account. To this end, the configured subsystems have stronger connections within a subsystem and weaker couplings among the subsystems. For the decomposed subsystem, a distributed moving horizon estimation method based on information interaction is designed. Compared with the existing methods, the proposed method can effectively improve the performance of state estimation for a same distributed state estimation.

Key words: subsystem decomposition; moving horizon estimation; community detection; weighted directed graph

Citation: YANG Xiaofeng, XIE Wei, ZHANG Langwen. Weighted directed graph based community detection for subsystem decomposition. *Control Theory & Applications*, 2020, 37(9): 1923 – 1930

1 引言

现代化学过程通常由相互连接的操作单元组成, 它们通过能量和信息流紧密地集成在一起. 先进控制算法在应用于这些过程时应满足预定的目标, 达到预期的安全, 可持续性和经济性水平^[1]. 然而, 传统的集中或分散控制架构难以适用于复杂过程系统^[2].

近十年来, 为提高计算效率, 维护灵活性和容错性, 分布式控制架构得到快速发展^[3]. 分布式控制系统设计包括两个关键步骤^[4]: 1) 合理地将整个系统划

分成较小的子系统, 需要保证子系统的能控性、能观性和灵活性等; 2) 针对每个子系统设计局部控制器及其协调算法. 在过去的十年里, 分布式控制算法(尤其是分布式模型预测控制)吸引了非常广泛的研究关注^[5]. 然而, 与其同等重要的子系统划分问题受到的关注相对较少^[6].

近几年来, 子系统划分方法取得了一些成果^[7-9]. 文献[7]提出了一种基于层次聚类的分散控制系统分解方法. 文献[8]提出了一种既考虑耦合强度又考虑结

收稿日期: 2019-08-01; 录用日期: 2020-03-26.

†通信作者. E-mail: aulwzhang@scut.edu.cn; Tel.: +86 13760804711.

本文责任编辑: 李少远.

国家自然科学基金项目(61803161, 61973125), 江门市创新科研团队引进基金项目(2017TD03)资助.

Supported by the National Natural Science Foundation of China (61803161, 61973125) and the Innovative Research Team Project of Jiangmen (2017TD03).

构紧密性的输入-输出配对方法. 另外, 文献[9]研究了基于无权有向图的社区发现子系统划分算法. 在社区发现算法中, 将一个大规模系统看作一个网络, 通过将网络分解为较小的子网络, 建立面向分布式控制或估计的子系统模型.

考虑到不同状态和输出变量之间的结构紧密性, 已有研究尝试将非线性系统分解为较小的单元, 然后设计分布式状态估计^[10-11]. 文献[10]提出了一种面向分布式状态估计的过程网络子系统划分方法. 在文献[11], 分布式状态估计和控制在一个架构内被共同考虑. 上述面向分布式状态估计的子系统划分方法均只考虑过程系统的物理拓扑, 而未考虑不同变量之间的连接强度. 研究发现, 如果不考虑变量间连通性的强弱, 可能导致子系统划分的次优或不适当, 特别是对于连接强度差异较大的系统影响更大^[12].

基于上述分析, 本文提出一种面向分布式状态估计的子系统划分方法, 构建了加权有向图, 既考虑结构的紧密性, 又考虑系统状态/输出之间的连接强度. 基于加权有向图, 利用社区发现算法将所有系统变量划分为较小的群组, 使得划分得到的子系统内部关联较强, 而子系统之间的耦合强度较弱.

2 问题描述

考虑如下复杂非线性系统的子系统划分问题:

$$\dot{x}(t) = f(x(t)), \quad (1a)$$

$$y(t) = h(x(t)), \quad (1b)$$

其中: $x \in \mathbb{R}^{n_x}$ 表示状态向量, $y \in \mathbb{R}^{n_y}$ 表示输出向量, f 和 h 分别表示非线性系统状态和输出方程.

本研究的目标是将系统(1)划分成多个子系统, 得到第 i 个子系统的模型, 以设计分布式状态估计器

$$\dot{x}_{(i)}(t) = f_{(i)}(x_{(i)}(t), X_{(i)}(t)), \quad (2a)$$

$$y_{(i)}(t) = h_{(i)}(x_{(i)}(t)), \quad (2b)$$

其中: $x_{(i)} \in \mathbb{R}^{n_{x(i)}}$ 表示第 i 个子系统的状态向量, $y_{(i)} \in \mathbb{R}^{n_{y(i)}}$ 表示第 i 个子系统的测量输出, $X_{(i)}$ 表示所有与子系统 i 直接关联的子系统状态, $i = 1, \dots, p$, p 是划分得到子系统的个数. 在本设计中, 假设 p 是已知的且数量少于测量输出的个数, 即 $p \leq n_y$.

由于本文子系统的划分是面向分布式状态估计设计, 对于划分的子系统, 需要进行能观性的判断. 为了检验非线性系统(1)的能观性, 可以判断该系统的能观矩阵是否满秩^[13], 能观矩阵通过如下方式构建:

$$Q(x) = \frac{d}{dx} \begin{bmatrix} h(x) \\ L_f h(x) \\ \vdots \\ L_f^{n_x-1} h(x) \end{bmatrix}, \quad (3)$$

其中: $L_f h$ 表示方程 h 对方程 f 的Lie导数, 其定义为

$L_f h(x) = \frac{\partial h}{\partial x} f(x)$; $L_f^r h$ 表示 h 对 f 的 r 阶Lie导数, 定义为 $L_f^r h(x) = L_f L_f^{r-1} h(x)$. 如果对于 $x \in X$, 有 $\text{rank}(Q(x)) = n_x$, 那么, 系统(1)在范围 X 内是局部可观的^[13], 否则系统(1)是不可观的.

社区发现算法是一种将大规模复杂网络划分成多个子网络的有效工具^[14], 通过由文献[15]定义的模块度来评价一个网络划分的好坏. 考虑一个带有 N 个节点的网络, 其不加权的邻接矩阵 A 是一个 $N \times N$ 矩阵, 定义元素 A_{ij} 为

$$A_{ij} = \begin{cases} 1, & \text{如果从顶点 } j \text{ 到顶点 } i \text{ 存在一条连接,} \\ 0, & \text{其他.} \end{cases} \quad (4)$$

通常, 不考虑节点自身的回路, 也就是说邻接矩阵的主对角线上的元素为0. 用模块度来评价划分好坏, 模块度 Q 的范围为0到1, 社区发现算法通过最大化模块度来找到最优群组(子系统)结构, 使得划分的子系统内部的连接数最多, 而群组之间的连接数最少. 一般模块度 Q 在0.3到0.7之间时被认为是较好的划分方法, 否则表示该系统不适合再进行划分.

然而, 实际应用的子系统划分中, 仅考虑划分得到的子系统内部连接边数最多是不够的, 还需要考虑连接边的连接权重. 本文将对系统构造带权重的有向图, 并利用模块度最大化方法对系统进行划分, 以保证各个划分子系统之间的连接权重最小.

3 基于加权有向图的社区发现子系统划分

本节将提出一种系统化的子系统划分方法, 考虑将非系统(1)划分成多个子系统, 并设计滚动时域估计算法对子系统状态进行分布式状态估计.

本文方法的主要结构如图1所示. 首先, 将非线性系统(1)用加权有向图进行描述, 其中系统状态和测量输出变量看成网络中的节点, 这些节点通过加权连接边进行连接, 连接边上的权重反映了节点之间的连接强度. 然后, 基于加权有向图, 利用社区发现算法找到最大模块度, 将系统划分成多个子系统群组. 最后, 通过检测划分子系统的能观性, 获得子系统划分结果.

3.1 加权连接边的定义

本文将利用加权有向图对系统(1)进行描述, 具体来说, 将所有的状态和测量输出变量当做有向图的节点, 这些节点通过有向边进行连接. 令 f_i 表示向量方程 f 的第 i ($i = 1, \dots, n_x$) 行, h_j 表示向量方程 h 的第 j ($j = 1, \dots, n_y$) 行, x_i 表示状态向量 x 的第 i ($i = 1, \dots, n_x$) 行, y_j ($j = 1, \dots, n_y$) 表示输出变量 y 的第 j 行.

在已有的无权重有向图中, 不同变量(节点)之间的连接强度未被考虑. 为获得权重, 对非线性系统在工作点 x_s 求一阶偏导得到有向图中连接边的敏感度

$$S(x_i, x_l) = \frac{\partial f_l(x)}{\partial x_i} \Big|_{x=x_s}, \quad (5a)$$

$$S(x_l, y_j) = \frac{\partial h_j(x)}{\partial x_l} \Big|_{x=x_s}, \quad (5b)$$

其中: $S(x_i, x_l)$ 表示从状态 x_i 到另一个状态 x_j 的敏感度, $S(x_l, y_j)$ 表示从状态 x_l 到输出 y_j 的敏感度.

那么, 可以根据敏感度构造敏感度矩阵

$$S = \begin{bmatrix} \bar{A} & 0 \\ \bar{C} & 0 \end{bmatrix}_{n_x+n_y}, \quad (6)$$

其中 \bar{A} 和 \bar{C} 通过对非线性系统(1)在工作点 x_s 求雅可比矩阵得到, 即

$$\bar{A} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_{n_x}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_{n_x}}{\partial x_1} & \dots & \frac{\partial f_{n_x}}{\partial x_{n_x}} \end{bmatrix}_{x_s}, \quad \bar{C} = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \dots & \frac{\partial h_1}{\partial x_{n_y}} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_{n_y}}{\partial x_1} & \dots & \frac{\partial h_{n_y}}{\partial x_{n_y}} \end{bmatrix}_{x_s}.$$

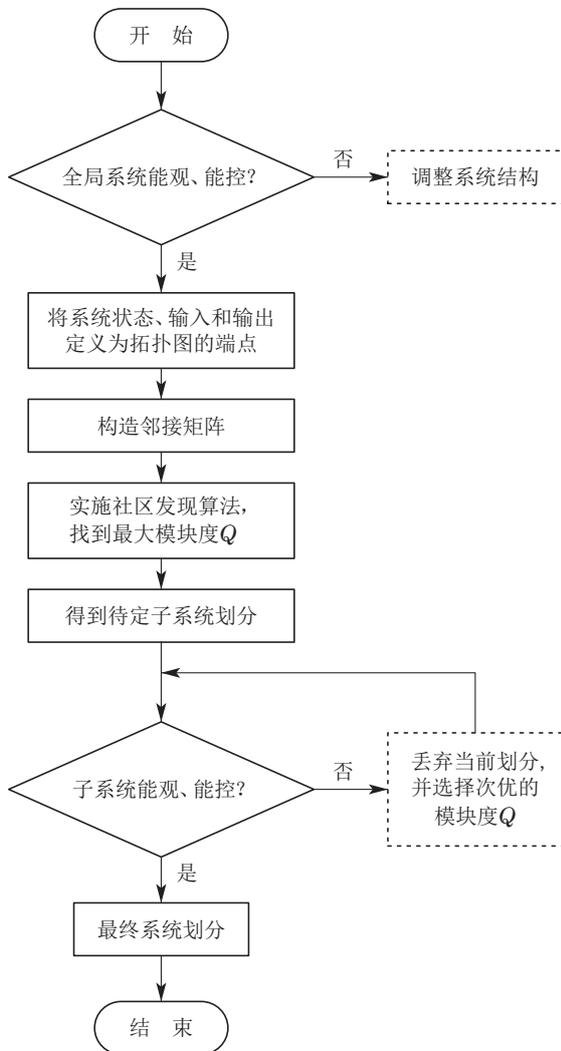


图 1 本文子系统划分结构

Fig. 1 Flowchart of the proposed subsystem decomposition

定义如下连接边的权重:

状态变量 x_i 到另一个状态变量 x_l 连接边的权重定义为 $w(x_i, x_l)$, $i = 1, \dots, n_x$

$$w(x_i, x_l) = \begin{cases} \frac{1}{|S(x_i, x_l)|^\alpha}, & \frac{\partial f_l(x)}{\partial x_i} \Big|_{x_s} \neq 0, \\ \infty, & \text{其他.} \end{cases} \quad (7)$$

状态变量 x_l 到测量输出变量 y_j 连接边的权重定义为 $w(x_l, y_j)$, $l = 1, \dots, n_x, j = 1, \dots, n_y$

$$w(x_l, y_j) = \begin{cases} \frac{1}{|S(x_l, y_j)|^\alpha}, & \frac{\partial h_j(x)}{\partial x_l} \Big|_{x_s} \neq 0, \\ \infty, & \text{其他,} \end{cases} \quad (8)$$

其中 α 是一个介于 $[0, 1]$ 的参数, 用于调节连接边的权重在子系统划分过程中的重要性, 当 $\alpha = 0$ 时, 所有连接边的权重为 1, 即等效于已有的无权重有向图; 当 α 增加时, 连接边的权重将在子系统划分中更多地被考虑. 当 $S(x_l, y_j) = 0$ 或 $S(x_i, x_l) = 0$ 时, 连接边的权重为无穷, 即两个节点之间没有直接连接.

由于不同的 α 值将影响子系统的划分结果, 如何选择合适的参数 α 对子系统划分相当重要. 总体而言, 当有向图中所有连接边的敏感度差异很小时, 在子系统划分时应重点考虑连接边的数量, 此时选择较小的 α 即可. 相反, 当所有连接边的敏感度差异很大时, 应更多地考虑连接边的权重对子系统划分的影响^[16].

3.2 最短路径

为实施社区发现算法, 需要构造加权邻接矩阵, 因此首先应找到有向图中的一个节点到另一个节点的最短路径. 令 \mathcal{P} 表示一个状态变量到另一个状态变量或输出变量的路径, 在这条路径 \mathcal{P} 中的所有连接边表示为 $e \in \mathcal{P}$. 路径 $L(\mathcal{P})$ 的长度可以计算得到

$$L(\mathcal{P}) = \sum_{e \in \mathcal{P}} w(e). \quad (9)$$

对于一条路径, 从节点 x_i ($i = 1, \dots, n_x$) 到另一个节点 x_l ($l = 1, \dots, n_x, l \neq i$), 考虑路径 \mathcal{P}_{il} 经过 $N(\mathcal{P}_{il})$ 个状态变量 $x_1^{(i,l)}, x_2^{(i,l)}, \dots, x_{N(\mathcal{P}_{il})}^{(i,l)}$. 那么, 从 x_i 到 x_l 路径 $L_{il}(\mathcal{P}_{il})$ 的长度为

$$L_{il}(\mathcal{P}_{il}) = w(x_i, x_1^{(i,l)}) + \dots + w(x_{N(\mathcal{P}_{il})}^{(i,l)}, x_l). \quad (10)$$

状态变量 x_i 到另一个状态变量 x_l 的所有路径集合表示为 \mathcal{P}_{il} , 其最短路径通过求解如下优化问题得到

$$\underline{d}(x_i, x_l) = \min_{\mathcal{P}_{il} \in \mathcal{P}_{il}} L_{il}(\mathcal{P}_{il}). \quad (11)$$

本文未考虑节点自身的连接, 即节点到自身的最短路径为 $\underline{d}(x_i, x_i) = \infty, i = 1, \dots, n_x$. 类似地, 从状态向量 x_l 到测量输出向量 y_j 的路径长度 $L_{lj}(\mathcal{P}_{lj})$ 为

$$L_{lj}(\mathcal{P}_{lj}) = w(x_l, x_1^{(l,j)}) + \dots + w(x_{N(\mathcal{P}_{lj})}^{(l,j)}, y_j). \quad (12)$$

相应地, 从状态向量 x_l 到测量输出向量 y_j 的最短路径 $\underline{d}(x_l, y_j)$ 为

$$\underline{d}(x_l, y_j) = \min_{P_{l,j} \in \mathcal{P}_{l,j}} L_{l,j}(P_{l,j}), \quad (13)$$

其中: $l = 1, \dots, n_x, j = 1, \dots, n_y, P_{l,j}, \mathcal{P}_{l,j}$ 表示从状态 x_l 到输出 y_j 所有路径和其中一条路径. 本文利用Dijkstra's算法^[17]找到任意两个节点的最短路径, 其计算复杂度为 $O(E \log V)$, 其中 V 和 E 分别是节点的数量和连接的边数.

3.3 构建邻接矩阵

本节将构建非线性系统(1)的邻接矩阵. 对于系统(1)所形成的有向图, 总共有 n_a (其中 $n_a = n_x + n_y$) 个顶点. 令 c_a 为状态和测量输出向量, 定义为 $c_a = [x_1, \dots, x_{n_x}, y_1, \dots, y_{n_y}]$.

邻接矩阵 $A_w \in \mathbb{R}^{n_a \times n_a}$ 包含了在 c_a 中的所有节点之间的最短距离 $\underline{d}(x_i, x_l)$ 和 $\underline{d}(x_l, y_j)$, 其中: $i = 1, \dots, n_x, j = 1, \dots, n_y$,

$$\bar{A}_w = \begin{bmatrix} \bar{A}_{w,11} & 0_{n_x \times n_y} \\ \bar{A}_{w,12} & 0_{n_y \times n_y} \end{bmatrix}_{n_a \times n_a}, \quad (14)$$

其中 $A_{w,11}$ 和 $A_{w,12}$ 根据如下规则构建:

$$\bar{A}_{w,11} = \begin{bmatrix} \frac{1}{\underline{d}(x_1, x_1)} & \dots & \frac{1}{\underline{d}(x_1, x_{n_x})} \\ \vdots & \ddots & \vdots \\ 1 & & 1 \\ \frac{1}{\underline{d}(x_{n_x}, x_1)} & \dots & \frac{1}{\underline{d}(x_{n_x}, x_{n_x})} \end{bmatrix},$$

$$\bar{A}_{w,12} = \begin{bmatrix} \frac{1}{\underline{d}(x_1, y_1)} & \dots & \frac{1}{\underline{d}(x_1, y_{n_y})} \\ \vdots & \ddots & \vdots \\ 1 & & 1 \\ \frac{1}{\underline{d}(x_{n_x}, y_1)} & \dots & \frac{1}{\underline{d}(x_{n_x}, y_{n_y})} \end{bmatrix}.$$

加权邻接矩阵 A_w 通过如下方式构建:

a) 对 \bar{A}_w 的第 (i, j) 个元素 ($\bar{A}_{w,ij}$) 进行归一化

$$A_{w,ij} = \frac{\bar{A}_{w,ij} - \bar{A}_{w,ij}^{\min}}{\bar{A}_{w,ij}^{\max} - \bar{A}_{w,ij}^{\min}}, \quad (15)$$

其中: $i, j = 1, \dots, n_a$,

$$\bar{A}_{w,ij}^{\min} = \min\{\bar{A}_{w,ij} | \bar{A}_{w,ij} \neq 0\},$$

$$\bar{A}_{w,ij}^{\max} = \max\{\bar{A}_{w,ij} | \bar{A}_{w,ij} \neq 0\};$$

b) 如果 $\bar{A}_{w,ij} = 0$, 保留 $A_{w,ij} = 0$, 其中 $A_{w,ij}$ 是邻接矩阵 A_w 的第 (i, j) 个元素.

那么, 所构建的加权邻接矩阵 A_w 将被用于社区发现算法, 以对系统(1)进行划分. 上述定义的加权邻接矩阵(15)是无权重邻接矩阵^[14]的扩展, 其主要区别是:

a) 加权邻接矩阵考虑了两个节点之间连接的权重, 而文献[14]只考虑了两个节点之间的连接度; b) 本文方法中, 考虑了两个节点之间的最短路径, 因此, 邻接矩阵中的0元素表征了两个节点之间没有连接.

3.4 寻找最大模块度

模块度为正值时, 表明子系统内部的连接边比预期多, 且模块度越大, 群组结构越好. 因此, 子系统划分的目标是在所有可能的群组结构候选项上找到模块度的最大值. 本文考虑加权有向图, 有向图中边的总权重之和为 $m_w = \sum_{i=1}^N \sum_{j=1}^N A_{w,ij}$, 基于加权邻接矩阵 A_w 定义加权有向图的模块度

$$Q_w = \frac{1}{m_w} \sum_{i=1}^N \sum_{j=1}^N (A_{w,ij} - \frac{k_{w,i}^{\text{in}} k_{w,j}^{\text{out}}}{m_w}) \delta(c_i, c_j), \quad (16)$$

其中: c_i 表示节点 i 被分配到的群组, 有 $\delta(c_i, c_j) = 1$, 如果顶点 i, j 属于同一个群组, 即 $c_i = c_j$; 否则 $\delta(c_i, c_j) = 0$. $k_{w,i}^{\text{in}}$ 表示节点的内度, 即所有进入节点 i 的连接边权重之和, 由 $k_{w,i}^{\text{in}} = \sum_{j=1}^N A_{w,ij}$ 计算得到. $k_{w,j}^{\text{out}}$ (由

$k_{w,j}^{\text{out}} = \sum_{i=1}^N A_{w,ij}$ 计算得到) 表示从节点 j 外度, 即所有离开节点 j 的连接边之和.

模块度 Q_w 既考虑了灵敏度, 又考虑了边的数量. 子系统划分问题等价于通过寻找最大化的模块度 Q_w 进行社区结构发现. 首先, 需要进行初步的群组配置, 群组结构的初始化基于如下假设: i) 每个群组至少有一个测量输出; ii) 应将直接影响 y_i 的系统状态分配给属于 y_i 同一个群组. 初始化群组结构步骤如下:

步骤1 定义关于复杂系统的群组结构 $s_w(0) = [c_{x_1}(0) \dots c_{x_{n_x}}(0) c_{y_1}(0) \dots c_{y_{n_y}}(0)]$, 其中 $c_{\eta}(0)$ 表示每个节点 $x_i (i = 1, \dots, n_x)$ 或 $y_j (j = 1, \dots, n_y)$ 的群体标签. 应用快速二分法(如文献[18])寻找最大模块度 Q_w 时, 在 s_w 中的第 i 个变量被分配到第 i 个群体, $i = 1, \dots, n_a$, 即 $\bar{s}_w(0) = [1 \dots n_a]$.

步骤2 对于每个状态变量 $x_i, i = 1, \dots, n_x$, 找到所有和状态变量 x_i 关联的测量输出, 在 $\bar{s}_w(0)$ 中将相应的元素标为相同编号.

步骤3 对于每个测量输出 $y_j, j = 1, \dots, n_y$, 找到所有和输出方程 $h_j(\cdot)$ 关联的状态变量, 并将这些状态标量标为和 y_j 相同的编号.

步骤4 所得到的分组 $\bar{s}_w(0)$ 即可作为实施社区发现算法的初始群组.

得到的初始群组结构将用于寻找子系统分解中模块度的最大值. 在所提出的方法中, 子系统(即式(2)中的 p) 的数量应预先设定. 在本文中, 快速二分法用于最大化模块度^[18], 其步骤如下:

步骤1 对每个节点 i 实施如下算法:

步骤1.1 对于节点 i 的每个邻居节点 j , 通过将节点 i 从当前群组移动到节点 j 所在的群组, 计算模块度的变化值 ΔQ_w .

步骤 1.2 找到上一步中最大的 $\Delta Q_w > 0$, 将顶点 i 放入到相应的群组中(即将节点 i 编入最大 ΔQ_w 相应的分组), 得到新的分组.

步骤 2 令 $n_c(k+1)$ 作为节点集结后的群组数量, 如果 $n_c(k+1) < p$, 回到步骤 1. 否则, 找到最大化的模块度 Q_w .

3.5 分布式滚动时域估计设计

为对子系统的划分结果进行分析, 本文将对所得到的子系统应用于分布式滚动时域估计算法, 并说明考虑加权有向图的必要性. 本文提出分布式滚动时域估计算法, 局部估计器通过共享的通信网络进行信息交互. 子系统 i 的局部估计器为求解如下优化问题:

$$\min_{\hat{x}_{(i)}(t_k|t_k)} \sum_{d=k-N}^{k-1} |\omega_{(i)}(t_d)|_{Q_i}^2 + \sum_{d=k-N}^k |v_{(i)}(t_d)|_{R_i}^2, \quad (17a)$$

$$\text{s.t. } \dot{\hat{x}}_{(i)}(\eta|t_k) = f_{(i)}(\hat{x}_{(i)}(\eta|t_k), \bar{X}_{(i)}) + \omega_{(i)}(t_d), \quad \eta \in [t_d, t_{d+1}], d = k - N, \dots, k - 1, \quad (17b)$$

$$v_{(i)}(t_d) = y_{(i)}(t_d) - h_{(i)}(\hat{x}_{(i)}(t_d)), \quad d = k - N, \dots, k - 1, \quad (17c)$$

$$\hat{x}_{(i)}(t_d) \in X_i, d = k - N, \dots, k, \quad (17d)$$

$$v_{(i)}(t_d) \in V_i, d = k - N, \dots, k, \quad (17e)$$

$$\omega_{(i)}(t_d) \in W_i, d = k - N, \dots, k - 1, \quad (17f)$$

其中: N 表示估计窗口, $Q_i > 0$ 和 $R_i > 0$ 分别表示过程扰动 $w_{(i)}$ 和测量噪声 $v_{(i)}$ 的权重矩阵; $\hat{x}_{(i)}(t_d|t_k)$, $d = k - N, \dots, k$ 表示在采样时刻 t_k 对子系统 i 状态的估计, $w_{(i)}$ 和 $v_{(i)}$ 分别表示扰动和测量噪声的估计值, $\bar{X}_{(i)}(t_d)$ 是 $X_{(i)}(t)$ 的估计值, $t \in [t_d, t_{d+1}]$, $d = k - N, \dots, k - 1$.

优化问题(17)的目标是最小化子系统 i 的局部代价函数, $\{\hat{x}_{(i)}(t_d|t_k)\}_{d=k-N}^k$ 是优化变量. 同时, 分布式滚动时域估计问题受到子系统模型, 过程扰动和测量噪声的约束限制. 在采样时刻 $t = t_k$, 在线求解优化问题(17)得到子系统 i 的最优的状态估计序列, 即 $\hat{x}_{(i)}(t_{k-N}|t_k), \dots, \hat{x}_{(i)}(t_k|t_k)$. 子系统 i 的最优状态估计为 $\hat{x}_{(i)}(t_k) = \hat{x}_{(i)}(t_k|t_k)$. 在每个采样周期, 求解子系统 i 的估计状态, 本文提出如下非线性系统的分布式滚动时域估计算法:

步骤 1 (初始化)在 $t_0 = 0$, 初始化子系统 i 的估计状态, 即 $\hat{x}_{(i)}(0)$, $i = 1, \dots, p$.

步骤 2 (观测器设计)对于 $t_k > 0$, 重复如下步骤:

步骤 2.1 每个局部估计器 i 获得子系统 i 的测量输出, 即 $y_{(i)}(t_k)$.

步骤 2.2 如果 $\text{mod}(k, n) = 0$, 每个局部估计器 i 获得其他关联子系统的状态估计序列, 即 $\bar{X}_{(i)}(t_d)$, $d = k - N, \dots, k - 1$, 其中 $\bar{X}_{(i)}(t_d)$ 表示其他关联子系统最近的估计结果, 即 $\hat{X}_{(i)}(t_{d-\tau})$, $t_{d-\tau}$, $\tau = \text{mod}(d, n)$.

步骤 2.3 根据关联子系统的状态估计及子系统 i 的局部测量状态, 每个估计器 i 求解优化问题(17), 以获得子系统 i 的状态估计, 即 $\hat{x}_{(i)}(t_k)$. 那么, 所有子系统的状态结果估计为

$$\hat{x}(t_k) = [\hat{x}_{(1)}^T(t_k) \ \dots \ \hat{x}_{(p)}^T(t_k)]^T.$$

步骤 2.4 在 $k = k + 1$ 时刻, 回到步骤 2.1.

4 实例应用

本节考虑将所提出的子系统划分及分布式滚动时域估计算法应用于反应-分离过程系统^[19], 该过程包括两个连续搅拌反应釜和一个快速分离釜(见图 2). 第一个连续搅拌反应釜包含反应物 J , 其给定蒸汽为 F_{10} , 然后产产品 K . 然后, 产品 K 在第 2 个反应釜中转换成最终产品 L .

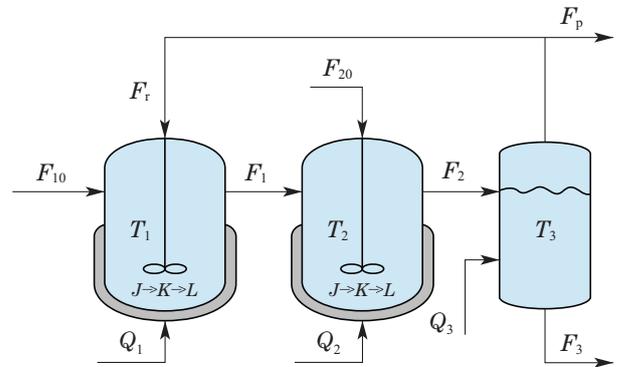


图 2 反应-分离过程系统

Fig. 2 Reactor-separator process

该反应过程的动态模型如下^[19]:

$$\dot{x}_{A1} = \frac{F_{10}}{V_1}(x_{A10} - x_{A1}) + \frac{F_r}{V_1}(x_{Ar} - x_{A1}) - k_1 e^{-\frac{E_1}{RT_1}} x_{A1}, \quad (18a)$$

$$\dot{x}_{B1} = \frac{F_{10}}{V_1}(x_{B10} - x_{B1}) + \frac{F_r}{V_1}(x_{Br} - x_{B1}) + k_1 e^{-\frac{E_1}{RT_1}} x_{A1} - k_2 e^{-\frac{E_2}{RT_1}} x_{B1}, \quad (18b)$$

$$\dot{T}_1 = \frac{F_{10}}{V_1}(T_{10} - T_1) + \frac{F_r}{V_1}(T_3 - T_1) - \frac{\Delta H_1}{C_p} k_1 \times e^{-\frac{E_1}{RT_1}} x_{A1} - \frac{\Delta H_2}{C_p} k_2 e^{-\frac{E_2}{RT_1}} x_{B1} + \frac{Q_1}{\rho C_p V_1}, \quad (18c)$$

$$\dot{x}_{A2} = \frac{F_1}{V_2}(x_{A1} - x_{A2}) + \frac{F_{20}}{V_2}(x_{A20} - x_{A2}) - k_1 e^{-\frac{E_1}{RT_2}} x_{A2}, \quad (18d)$$

$$\dot{x}_{B2} = \frac{F_1}{V_2}(x_{B1} - x_{B2}) + \frac{F_{20}}{V_2}(x_{B20} - x_{B2}) + k_1 e^{-\frac{E_1}{RT_2}} x_{A2} - k_2 e^{-\frac{E_2}{RT_2}} x_{B2}, \quad (18e)$$

$$\dot{T}_2 = \frac{F_1}{V_2}(T_1 - T_2) + \frac{F_{20}}{V_2}(T_{20} - T_2) - \frac{\Delta H_1}{C_p} k_1 \times e^{-\frac{E_1}{RT_2}} x_{A2} - \frac{\Delta H_2}{C_p} k_2 e^{-\frac{E_2}{RT_2}} x_{B2} + \frac{Q_2}{\rho C_p V_2}, \quad (18f)$$

$$\dot{x}_{A3} = \frac{F_2}{V_3}(x_{A2} - x_{A3}) - \frac{F_r + F_p}{V_3}(x_{Ar} - x_{A3}), \quad (18g)$$

$$\dot{x}_{B3} = \frac{F_2}{V_3}(x_{B2} - x_{B3}) - \frac{F_r + F_p}{V_3}(x_{Br} - x_{B3}), \quad (18h)$$

$$\dot{T}_3 = \frac{F_2}{V_3}(T_2 - T_3) + \frac{Q_3}{\rho C_p V_3} + \frac{F_r + F_p}{\rho c_p V_3} (x_{Ar} \Delta H_{v1} + x_{Br} \Delta H_{v2} + x_{Cr} \Delta H_{v3}), \quad (18i)$$

其中: x_{Ai}, x_{Bi} 表示产品 J, K 在容器 $i(i = 1, 2, 3)$ 中的质量分数; x_{C3} 表示产品 L 在反应釜3中的质量分数; T_i 容器 $i(i = 1, 2, 3)$ 中液体的温度; T_{10}, T_{20} 表示输入到容器1和2的蒸汽温度; F_1, F_2 表示物料从容器1和2流出的流速; F_{10}, F_{20} 表示物料从容器1和2流出的稳态流速; F_r, F_p 分别表示产品的循环和废弃流速; V_i 表示容器 $i(i = 1, 2, 3)$ 的体积; E_1 和 E_2 表示反应过程1和2的催化剂; k_1 和 k_2 表示反应过程1和2的指前因子值; ΔH_1 和 ΔH_2 表示反应过程1和2的热量; $\alpha_A, \alpha_B, \alpha_C$ 表示物质 J, K, L 的相对挥发率; Q_1, Q_2, Q_3 分别表示容器 $i(i = 1, 2, 3)$ 的热量输入; C_p 热容量; R 表示气体常数; ρ 表示溶液浓度.

相关模型参数参见表1.

表1 模型参数

Table 1 Model parameters

变量	值	变量	值
T_{10}, T_{20}	300 K	α_A	3.5
F_{10}, F_{20}	5.04 m ³ /h	α_B	1
F_r	50.4 m ³ /h	α_C	0.5
E_1	50 kJ/mol	ΔH_1	-240 kJ/mol
E_2	60 kJ/mol	ΔH_2	-280 kJ/mol
k_1	9.972 × 10 ⁶ h ⁻¹	R	8.314
k_2	9.36 × 10 ⁶ h ⁻¹	x_{A10}	1
ΔH_{v1}	-3.53 × 10 ⁴ kJ/kmol	x_{B10}	1
ΔH_{v2}	-1.57 × 10 ⁴ kJ/kmol	x_{A20}	0
ΔH_{v3}	-4.068 × 10 ⁴ kJ/kmol	x_{B20}	0
F_p	0.504 m ³ /h	C_p	4.2 × 10 ³
ρ	1000 kg/m ³	Q_1	2900 MJ/h
Q_2	1000 MJ/h	Q_3	2900 MJ/h
V_1	1 m ³	V_2	0.5 m ³
V_3	1 m ³		

该反应-分离过程的其中一个稳定工作点为

$$\begin{aligned} \bar{x}_{A1} &= 0.1763, \bar{x}_{B1} = 0.6731, \bar{T}_1 = 480.3165 \text{ K}, \\ \bar{x}_{A2} &= 0.1965, \bar{x}_{B2} = 0.6536, \bar{T}_2 = 472.7863 \text{ K}, \\ \bar{x}_{A3} &= 0.0651, \bar{x}_{B3} = 0.6703, \bar{T}_3 = 474.8877 \text{ K}. \end{aligned}$$

x_{Ar}, x_{Br}, x_{Cr} 由如下计算得到:

$$x_{Ar} = \frac{\alpha_A x_{A3}}{\alpha_A x_{A3} + \alpha_B x_{B3} + \alpha_C x_{C3}}, \quad (19a)$$

$$x_{Br} = \frac{\alpha_B x_{B3}}{\alpha_A x_{A3} + \alpha_B x_{B3} + \alpha_C x_{C3}}, \quad (19b)$$

$$x_{Cr} = \frac{\alpha_C x_{C3}}{\alpha_A x_{A3} + \alpha_B x_{B3} + \alpha_C x_{C3}}. \quad (19c)$$

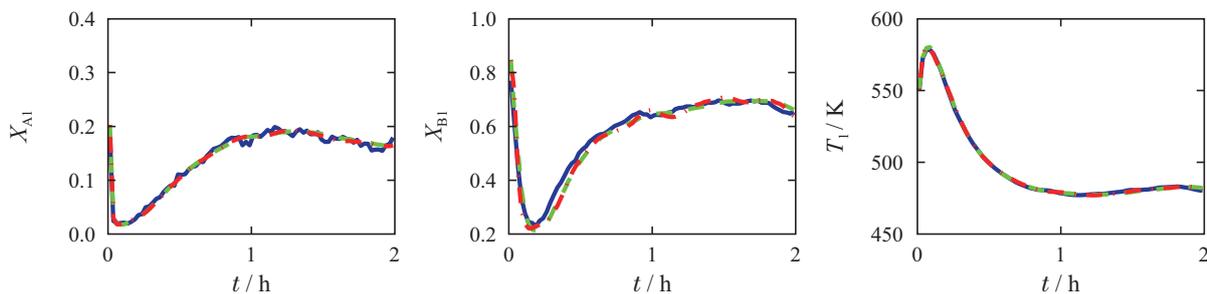
为实现利用测量输出 $T_i(i = 1, 2, 3)$ 对系统状态 $x_{Ai}, x_{Bi}, T_i(i = 1, 2, 3)$ 的估计, 首先对子系统进行划分, 并和已有的子系统划分方法(文献[14]) 进行比较, 得到如表2所示的划分结果.

表2 子系统划分结果

Table 2 Decompositions for the CSTR process

分解1(文献[14])		
子系统	状态变量	测量输出
1	x_{A1}, x_{B1}, T_1	T_1
2	x_{A2}, x_{B2}, T_2	T_2
3	x_{A3}, x_{B3}, T_3	T_3
分解2(本文方法)		
子系统	状态变量	测量输出
1	x_{A1}, T_1	T_1
2	x_{A2}, T_2	T_2
3	$x_{B1}, x_{B2}, x_{A3}, x_{B3}, T_3$	T_3

在应用分布式估计算法时, 如果进行实时信息交互, 分布式估计器的通信成本较高, 因此希望子估计器之间可以在较少通信交互的同时获得较好的估计效果. 如图3所示, 应用分布式滚动时域估计算法时(通讯间隔 $n = 1$), 基于子系统划分2的估计效果比基于子系统划分1获得了更好的效果, 本文提出的基于加权有向图的子系统划分方法比未考虑连接权重的分解算法具有更小的估计误差.



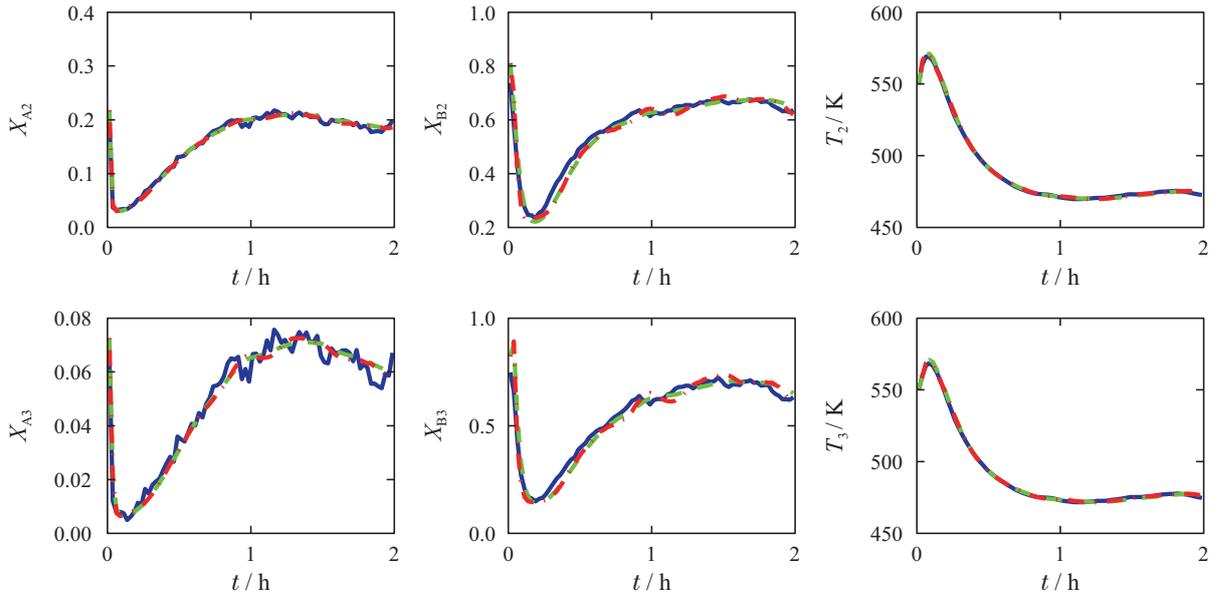


图 3 当 $n = 1$ 时, 实际状态轨迹(蓝色实线)、基于分解1的状态估计(绿色虚线)和基于分解2的状态估计(红色点划线)
Fig. 3 The trajectories for $n = 1$ of the actual states (blue solid lines), estimated states based on Decomposition 1 (green dashed-dot lines) and Decomposition 2 (red dashed lines)

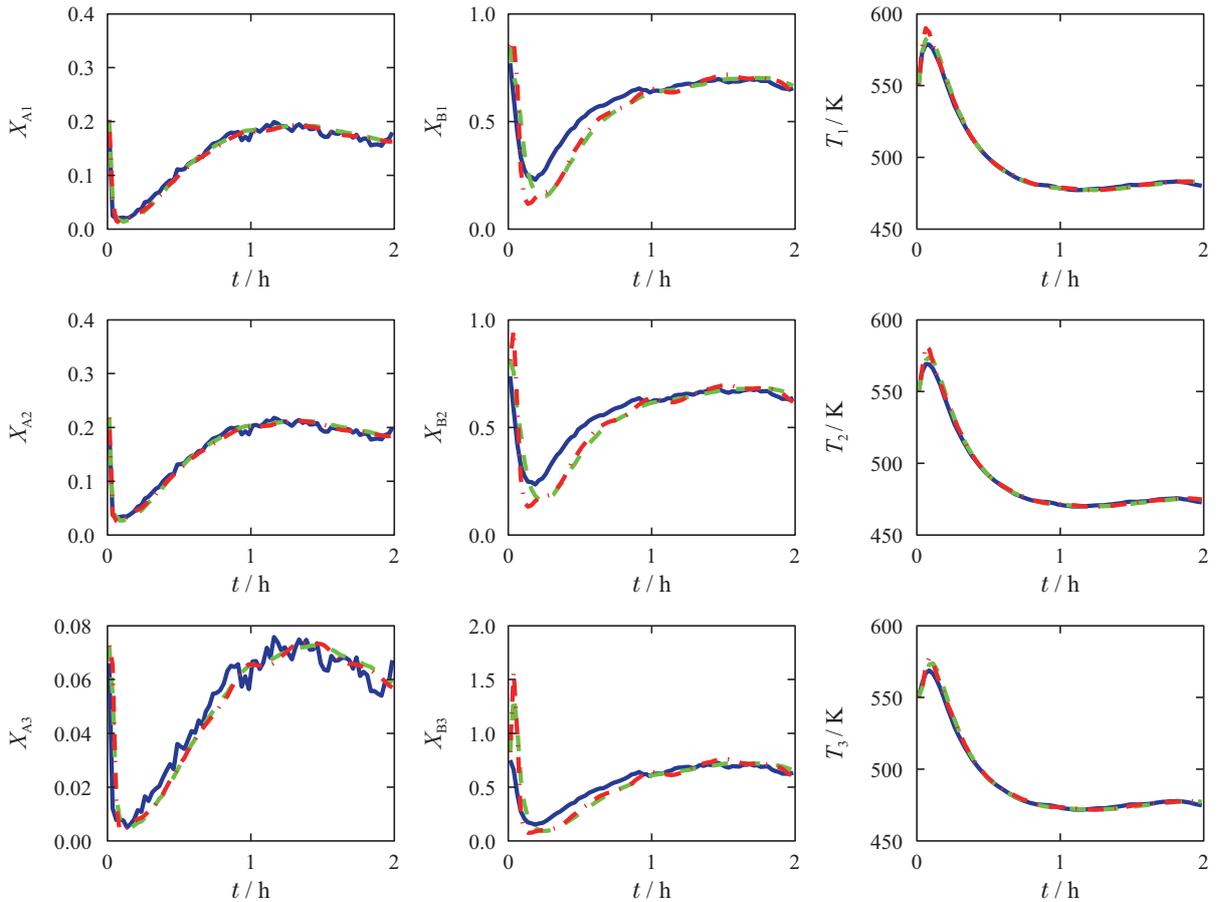


图 4 当 $n = 3$ 时, 实际状态轨迹(蓝色实线)、基于分解1的状态估计(绿色虚线)和基于分解2的状态估计(红色点划线)
Fig. 4 The trajectories for $n = 3$ of the actual states (blue solid lines), estimated states based on Decomposition 1 (green dashed-dot lines) and Decomposition 2 (red dashed lines).

考虑不同的通信间隔 $n = 1, 2, 3$, 对分布式滚动时域估计算法进行测试. 如表3和图4(通信间隔 $n = 3$)所示, 随着通信间隔的增加, 基于分解1的状态估计效果受到较大的影响. 而对于基于本文方法的分解2, 由于子系统之间的关联较弱, 因此较大的通信间隔对估计性能的影响并不大. 其现实意义是在保证估计效果的同时, 可以减轻分布式状态估计设计过程中子系统通信的次数, 降低通信成本.

表3 不同通信间隔对估计性能的影响

Table 3 Summary of the estimation performance

n	分解1	分解2	误差降低比例/%
1	23.0728	19.5030	-15.4718
2	24.2589	20.1610	-16.8924
3	25.6276	20.8374	-18.6916

5 总结

本文提出一种基于加权有向图的社区发现子系统划分方法, 并应用于分布式状态估计设计中. 基于耦合关联的强度研究了一种系统化的复杂大系统分解方法, 根据系统的动态模型自动构建系统的加权有向图, 进而获得系统的划分, 得到的子系统内部的关联远强于子系统之间的关联. 这类分解方能够为在复杂系统的分布式状态估计和控制奠定基础, 同时也对研究复杂网络的特性具有重要作用. 未来可扩展到带系统输入的复杂系统划分方法, 及其对分布式控制策略下的性能提升研究, 通过分布式状态估计和控制, 对该子系统划分方法进行系统的评估.

参考文献:

- [1] SCATTOLINI R. Architectures for distributed and hierarchical model predictive control—a review. *Journal of Process Control*, 2009, 19(5): 723 – 731.
- [2] ZHANG Yi, LIU Xiangjie. Robust distributed model predictive control for load frequency control of uncertain power systems. *Control Theory & Applications*, 2016, 33(5): 621 – 630.
(张怡, 刘向杰. 互联电力系统鲁棒分布式模型预测负荷频率控制. 控制理论与应用, 2016, 33(5): 621 – 630.)
- [3] FARINA M, SCATTOLINI R. Distributed predictive control: a non-cooperative algorithm with neighbor-to-neighbor communication for linear systems. *Automatica*, 2012, 48(6): 1088 – 1096.
- [4] ZHANG Langwen, XIE Wei. Multi-step predictive control for input-saturated uncertain systems based on sequential optimization scheme. *Control and Decision*, 2017, 32(8): 1499 – 1504.
(张浪文, 谢巍. 执行器饱和受限不确定系统的次序优化多步预测控制. 控制与决策, 2017, 32(8): 1499 – 1504.)
- [5] ZHANG L, XIE W, LIU J. Robust control of saturating systems with Markovian packet dropouts under distributed MPC. *ISA Transactions*, 2019, 85(2): 49 – 59.
- [6] HE Defeng, DING Baocang, YU Shuyou. Review of fundamental properties and topics of model predictive control for nonlinear systems. *Control Theory & Applications*, 2013, 30(3): 273 – 287.
(何德峰, 丁宝苍, 于树友. 非线性系统模型预测控制若干基本特点与主题回顾. 控制理论与应用, 2013, 30(3): 273 – 287.)
- [7] HEO S, MARVIN W A, DAOUTIDIS P. Automated synthesis of control configurations for process networks based on structural coupling. *Chemical Engineering Science*, 2015, 136(11): 76 – 87.
- [8] YIN X, LIU J. Input-output pairing accounting for both structure and strength in coupling. *AIChE Journal*, 2017, 63(4): 1226 – 1235.
- [9] TANG W, POURKARGAR D B, DAOUTIDIS P. Relative time-averaged gain array (RTAGA) for distributed control-oriented network decomposition. *AIChE Journal*, 2018, 64(5): 1682 – 1690.
- [10] YIN X, LIU J. Subsystem decomposition of process networks for simultaneous distributed state estimation and control. *AIChE Journal*, 2019, 65(3): 904 – 914.
- [11] YIN X, DECARDI-NELSON B, LIU J. Subsystem decomposition and distributed moving horizon estimation of wastewater treatment plants. *Chemical Engineering Research and Design*, 2018, 134(7): 405 – 419.
- [12] DAOUTIDIS P, TANG W, JOGWAR S S. Decomposing complex plants for distributed control: Perspectives from network theory. *Computers & Chemical Engineering*, 2018, 114(7): 43 – 51.
- [13] CICCARELLA G, DALLA MORA M, GERMANI A. A Luenberger-like observer for nonlinear systems. *International Journal of Control*, 1993, 57(3): 537 – 556.
- [14] LEICHT E A, NEWMAN M E J. Community structure in directed networks. *Physical Review Letters*, 2008, 100(11): 118703.
- [15] NEWMAN M E J, GIRVAN M. Finding and evaluating community structure in networks. *Physical Review E*, 2004, 69(2): 026113.
- [16] GOMEZ-PILAR J, POZA J, BACHILLER A, et al. Quantification of graph complexity based on the edge weight distribution balance: application to brain networks. *International Journal of Neural Systems*, 2018, 28(1): 1750032.
- [17] DIJKSTRA E W. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1959, 1(1): 269 – 271.
- [18] BLONDEL V D, GUILLAUME J L, LAMBIOTTE R, et al. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008, DOI: 10.1088/1742-5468/2008/10/p10008.
- [19] ZHANG J, LIU J. Distributed moving horizon state estimation for nonlinear systems with bounded uncertainties. *Journal of Process Control*, 2013, 23(9): 1281 – 1295.

作者简介:

杨晓峰 博士, 从事子系统分解及其分布式控制的研究, E-mail: auyang@mail.scut.edu.cn;

谢巍 教授, 博士, 从事线性参数变化系统的建模与控制的研究, E-mail: weixie@scut.edu.cn;

张浪文 助理研究员, 博士, 从事分布式模型预测控制理论与应用的研究, E-mail: aulwzhang@scut.edu.cn.