间歇过程动态潜结构阶段划分与在线监控

胡 磊[1], 刘 强[1†], 吴永建[1], 范自柱[2]

(1. 东北大学 流程工业综合自动化国家重点实验室, 辽宁 沈阳 110819;

2. 华东交通大学 江西省先进控制与优化重点实验室, 江西 南昌 330000)

摘要: 阶段划分是间歇过程准确建模和有效监控的前提. 针对传统阶段划分方法未考虑间歇过程的动态性造成阶段划分不准确、影响监控精度, 且具有参数选择难、鲁棒性差的局限, 提出一种基于动态潜结构的动态间歇过程阶段划分与在线监控方法. 首先, 对间歇过程三维张量数据沿变量方向展开, 并增加时滞变量构建增广矩阵来提取过程动态关系; 然后, 以增广矩阵作为输入, 定义一种新的基于解释方差变化的合并代价函数, 衡量不同子序列之间的动态潜结构相似度; 利用上述动态潜结构相似度的衡量标准, 提出基于自底向上启发式搜索策略的动态间歇过程阶段划分方法; 最后, 对划分得到的各阶段分别建立基于动态主元分析的子阶段模型和统计指标来实现在线监控. 采用青霉素补料分批发酵过程数据开展实验研究, 结果表明了所提方法的有效性和优越性.

关键词: 间歇过程; 过程监控; 动态主元分析; 相似性度量

引用格式: 胡磊, 刘强, 吴永建, 等. 间歇过程动态潜结构阶段划分与在线监控. 控制理论与应用, 2022, 39(2): 307 – 316

DOI: 10.7641/CTA.2021.00817

Dynamic latent structure based phase partition and online monitoring for batch processes

HU Lei[1], LIU Qiang[1†], WU Yong-jian[1], FAN Zi-zhu[2]

(1. State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang Liaoning 110819, China;

2. Key Laboratory of Advanced Control & Optimization of Jiangxi Province, East China Jiaotong University, Nanchang Jiangxi 330000, China)

Abstract: The phase partition is essential for statistical process modeling and monitoring of multi-phase batch processes. Traditional methods do not take into account the dynamics that make the phase partition and monitoring be inappropriate. Also it is difficult to determine model parameters and obtain fair performance when the underlying data structure is complex. A dynamic latent structure based phase partition and online monitoring method is thus proposed for dynamic batch processes. First, three-dimensional data are unfolded in variable-wise direction and original variables are augmented in order to extract the dynamic relations. After that, a new cost function based on relative variations of explained variance is defined in this paper to measure the difference of dynamic latent structure between sub-sequences. Using the defined cost function, a bottom-up heuristic algorithm is employed to find out the sub-optimal stage division. Finally, dynamic principal component analysis based sub-models and statistical indices for each phase are derived for online monitoring of dynamic batch processes. The application results on penicillin batch fermentation demonstrate the effectiveness and superiority of the proposed method.

Key words: batch processes; process monitoring; dynamic principal component analysis; similarity measure

Citation: HU Lei, LIU Qiang, WU Yongjian, et al. Dynamic latent structure based phase partition and online monitoring for batch processes. *Control Theory & Applications*, 2022, 39(2): 307 – 316

收稿日期: 2020−11−18; 录用日期: 2021−04−23.

†通信作者. E-mail: liuq@mail.neu.edu.cn; Tel.: +86 24-83677701.

本文责任编委: 周东华.

国家重点研发计划(2020YFB1710003), 国家自然科学基金项目(U20A20189, 61991401, 62161160338, 61673095), 兴辽英才计划(XLYC1907049, XLYC18080), 教育部基本科研业务费项目(N180802004)资助.

Supported by the National Key Research and Development Project (2020YFB1710003), National Natural Science Foundation of China (U20A20189, 61991401, 62161160338, 61673095), LiaoNing Revitalization Talents Program (XLYC1907049, XLYC1808001) and the Fundamental Research Funds for the Central Universities (N180802004).

1 引言

生产小批量、高附加值产品的间歇过程已成为现代工业的重要组成部分. 间歇过程运行状态的有效监控是保障其安全、高质量运行的关键[1–2]. 工业过程往往是少量的潜在变化驱动. 近年来, 以主元分析和偏最小二次为代表的潜结构投影方法成为间歇过程建模和监控的主要方向. 潜结构是指驱动过程的潜在变化不是原始的物理变量, 而是原始物理空间的投影; 这类方法所建立的潜结构有别于传统的输入输出关系, 是利用过程正常运行数据提取的原始变量间隐含的相关关系. 比如Nomikos 和MacGregor提出多向主元分析 (multiway principal component analysis, MPCA)[1]和多向偏最小二乘(multiway partial least squares, MPLS)[2]方法, 将间歇过程运行产生的三维数据(批次×时间×变量)沿批次方向展开成二维数据后, 采用主元分析和偏最小二乘方法进行建模与监控. 然而, 上述方法无法有效处理非线性、数据不等长、多模态和时变等复杂特征[3]; 针对间歇过程数据非线性和不等长, 文[4]提出基于统计模量的过程监控方法; 针对间歇过程多模态特性, 文[5] 提出基于概率密度k近邻的过程监控方法, 文[6]提出基于统计局部保持投影的特征提取与过程监控方法; 针对间歇过程的时变性, 文[7]提出基于滑动窗口支持向量数据描述的在线过程监控方法. 但上述方法未考虑间歇过程的多阶段特性.

多阶段是许多间歇过程的共同特点[8]. 由于不同阶段的控制目标或运行状态的显著差异, 造成过程特性呈现分段性, 不同阶段对应的变量相关性具有明显的差异. 传统的多向主元分析和多向偏最小二乘建模方法将间歇过程整个批次作为整体建立单一模型, 未考虑过程特性在运行周期内可能发生的正常变化, 造成利用单一模型进行过程监控易发生误报和漏报.

为了处理多阶段特性, 自然的想法是建立多个局部线性的统计模型来分别刻画各阶段的过程特性. 比如, Kosanovich等结合过程知识将聚合反应过程分成两个时段并分别建立MPCA子模型[9]. Louwerse和Smilde将整个运行周期按照预设定的阶段数平均划分为若干区间, 分别建立一系列从批次运行开始到各个区间的右端点的MPCA模型[10]. 上述阶段划分主要是依赖于人工经验或者只是简单等分为几个区间, 划分精度不高. 同时上述方法的子模型的建立均采用多向统计分析方法, 需要阶段完整的运行轨迹数据, 难以用于在线监控. 针对上述问题, 一种易于实现在线监控的处理方法是在每一个时刻分别建立统计模型, 称为局部模型或时间片模型[11]. 该种方式对过程特性描述精细, 但是未考虑过程特性在多个连续时刻内的相似性, 子模型过多造成不利于过程理解.

针对上述局限, 学者提出数据驱动的阶段划分与在线监控方法[11–18]. 它们的共同点是对于三维数据进行变量方式展开, 将每个时刻的数据单独作为一个样本, 这样易于实现在线监控. 不同点主要在评价划分效果的度量方式和阶段划分的策略. 文[12–14]基于k-means变体的聚类算法, 将时间片单元的载荷矩阵根据相似度聚类来划分不同阶段. Yu等提出了基于混合高斯模型的阶段划分方法[15]. Camacho等提出一种多阶段主元分析(multi-phase principal component analysis, MPPCA)方法[11, 13–16], 采用自顶向下策略以最大化重构预测能力为目标来划分阶段. Zhao 等提出步进有序时段划分 (step-wise sequential phase partition, SSSP)方法[17–18]. 该方法采用基于滑动窗口的策略, 通过从前向后渐近地衡量监控指标的变化来划分阶段. 就阶段划分策略而言, 可以分为聚类和混合高斯模型方法、基于滑动窗口方法、基于自顶向下方法这3类. 基于聚类和混合高斯模型方法由于没有直接考虑阶段的时间连续性, 需要复杂的后处理; 基于自顶向下策略方法, 由于每次寻找最优切分点都需要遍历所有可能的切分点, 具有与序列长度呈平方阶的高时间复杂度; 基于滑动窗口策略的方法只关注局部窗口内的信息, 缺乏全局的视野. 综上, 现有策略无法兼顾时间效率以及鲁棒性.

就建模方法而言, 现有的间歇过程阶段划分方法主要是依据变量间的静态关系变化. 间歇过程由于含有储能环节和存在反馈控制作用使得过程变量具有强自相关和时序互相关关系. 不同阶段内过程变量间隐含相关关系的变化主要体现在动态潜结构的变化, 而传统的基于静态潜结构变化的阶段划分方法难以发现这种变化, 导致阶段划分不准确. 为了结合过程动态关系进行阶段划分, 需要定义动态潜结构相似度来度量数据内部结构的相似性. 度量数据内部结构相似性的常用方法是主元分析相似性因子以及变体[12–14], 通过计算主元方向之间余弦相似度来比较两组数据结构的差异. 但该方法没有考虑主元的方差信息, 也难以确定不同主元方向的权重. 为了处理动态, 主要有两种方法: 一种是建立显式的时序模型[19–21], 但难以评估动态潜结构相似性; 另一种是静态方法的动态扩展[22–23], 对数据进行时间扩展来提取过程动态特性, 可继承静态潜结构方法易于度量数据潜结构相似度的优势.

本文将基于动态潜结构解释方差变化的合并代价与自底向上策略相结合, 提出一种基于动态潜结构的动态间歇过程阶段划分与在线监控方法. 主要贡献是: 1) 定义一种新的基于解释方差变化的合并代价函数, 用于定量评估不同子序列对应过程数据相关性结构的差异; 2) 提出基于动态潜结构的动态间歇过程阶段划分与在线监控方法. 其中, 阶段划分采用文[24]中的与序列长度具有线性相关的时间复杂度且在典型数

据集上取得优良性能的自底向上策略求解.

以下各节的组织结构如下: 第2节提出基于动态潜结构的动态间歇过程阶段划分和在线监控方法; 第3节在青霉素发酵过程上开展实验验证研究; 第4节给出结论并展望未来研究方向.

2 基于动态潜结构的间歇过程阶段划分和在线监控

2.1 动态间歇过程阶段划分

间歇过程阶段划分的本质是时间序列分割. 为了衡量子序列间数据潜结构的相似性, 第2.1.1节定义一种新的基于解释方差变化的合并代价函数; 第2.1.2节以增广矩阵作为输入, 利用上述代价函数衡量子序列间的动态潜结构相似度, 采用自底向上的贪心策略迭代合并子序列得到阶段划分结果.

2.1.1 基于解释方差变化的合并代价函数

设有多变量时间序列$\mathcal{T}=\{\boldsymbol{x}_k\in\mathbb{R}^m|1\leqslant k\leqslant N\}$, k为采样时间序号, m为变量数, N为样本数. 时间序列分割是将整个时间序列分为C个互不重叠的子序列集合$S_{\mathcal{T}}^C=\{S_c(a_c,b_c)|1\leqslant c\leqslant C\}$, 满足$a_1=1$, $b_C=N$, $a_c=b_{c-1}+1$. 其中, $S_c(a_c,b_c)=\{a_c\leqslant k\leqslant b_c\}$为子序列, a_c和b_c分别表示第c个子序列的起始时刻和终止时刻.

如何定义评估划分效果的代价函数是时间序列分割的关键. 针对多变量时间序列分割, Abonyi等提出一种基于主元分析的T^2统计量和Q统计量构造的代价函数[25], 如式(1)所示, 用于衡量单个子序列自身内部均值、方差以及相关结构的差异. 进一步, 基于式(1)定义了衡量整体划分质量的全局代价函数, 如式(2)所示:

$$\mathrm{cost}_{D^2}(S_c(a_c,b_c))=\frac{1}{(b_c-a_c+1)}\sum_{k=a_c}^{b_c}D_{c,k}^2, \quad (1)$$

$$G_{D^2}(S_{\mathcal{T}}^C)=\sum_{c=1}^{C}((b_c-a_c+1)/N)*\mathrm{cost}_{D^2}(S_c(a_c,b_c)), \quad (2)$$

其中: D^2可指T^2统计量$T_{c,k}^2$或Q统计量$Q_{c,k}$, $T_{c,k}^2$和$Q_{c,k}$分别按式(3)和式(4)计算, $\widehat{\boldsymbol{x}}_k$为对$\boldsymbol{x}_k$利用主元分析投影得到的数据, $\boldsymbol{U}_c$为对$S_c(a_c,b_c)$对应的数据矩阵$\boldsymbol{X_c}=[\boldsymbol{x}_{a_c}\ \boldsymbol{x}_{a_c+1}\ \cdots\ \boldsymbol{x}_{b_c}]$主元分解得到的载荷矩阵. 需要注意的, 区别于Hotelling's T^2统计量, 式(3)定义的T^2是数据在主元空间内到超平面中心的欧式距离.

$$T_{c,k}^2=\widehat{\boldsymbol{x}}_k^{\mathrm{T}}\widehat{\boldsymbol{x}}_k=\boldsymbol{x}_k^{\mathrm{T}}(\boldsymbol{U}_c\boldsymbol{U}_c^{\mathrm{T}})\boldsymbol{x}_k, \quad (3)$$

$$Q_{c,k}=(\boldsymbol{x}_k-\widehat{\boldsymbol{x}}_k)^{\mathrm{T}}(\boldsymbol{x}_k-\widehat{\boldsymbol{x}}_k)=\boldsymbol{x}_k^{\mathrm{T}}(\boldsymbol{I}-\boldsymbol{U}_c\boldsymbol{U}_c^{\mathrm{T}})\boldsymbol{x}_k. \quad (4)$$

将式(3)–(4)代入式(1)得到式(5), 代入式(2)得到式(6). 可以看出, $\mathrm{cost}_{T^2}(S_c(a_c,b_c))$和$\mathrm{cost}_Q(S_c(a_c,b_c))$分别是在子序列对应数据主元分解后的解释方差和未解释方差, 而全局代价函数$G_{T^2}(S_{\mathcal{T}}^C)$和$G_Q(S_{\mathcal{T}}^C)$分别定义了在分段子序列上利用主元分析得到的平均解释方差和平均非解释方差. 序列分割目标是寻找序列分割$S_{\mathcal{T}}^C$使得基于T^2统计量的全局代价函数$G_{T^2}(S_{\mathcal{T}}^C)$最大或者基于$Q$统计量的全局代价函数$G_Q(S_{\mathcal{T}}^C)$最小. 潜在假设是阶段划分越合理, 同一个阶段内部的过程特性就越相似, 那么在同样的主元数和阶段数的条件下, 主元分析就能够提取越多的数据变化信息, 或者说丢失越小的数据变化信息.

$$\begin{cases}\mathrm{cost}_{T^2}(S_c(a_c,b_c))=\dfrac{1}{(b_c-a_c+1)}\displaystyle\sum_{k=a_c}^{b_c}\|\widehat{\boldsymbol{x}}_k\|^2,\\ \mathrm{cost}_Q(S_c(a_c,b_c))=\dfrac{1}{(b_c-a_c+1)}\displaystyle\sum_{k=a_c}^{b_c}\|\boldsymbol{x}_k-\widehat{\boldsymbol{x}}_k\|^2,\end{cases} \quad (5)$$

$$\begin{cases}G_{T^2}(S_{\mathcal{T}}^C)=(1/N)*\displaystyle\sum_{c=1}^{C}\sum_{k=a_c}^{b_c}\|\widehat{\boldsymbol{x}}_k\|^2,\\ G_Q(S_{\mathcal{T}}^C)=(1/N)*\displaystyle\sum_{c=1}^{C}\sum_{k=a_c}^{b_c}\|\boldsymbol{x}_k-\widehat{\boldsymbol{x}}_k\|.\end{cases} \quad (6)$$

利用定义的全局代价函数, 可以将分段问题转化成优化问题, 求最优解的时间复杂度高[26], 通常通过启发式算法求解. 常用的启发式算法有滑动窗口法、自顶向下法和自底向上法等. 其中, 自底向上策略是时间序列分割广泛应用的方法, 具有时间效率高和鲁棒性强等优点[24], 故本文采用自底向上的分割策略求解. 它的基本流程是首先建立原序列的最佳可能逼近, 计算所有相邻子序列之间的合并代价, 然后每一步寻找合并代价最低对应子序列进行合并, 循环执行直到达到目标阶段数停止.

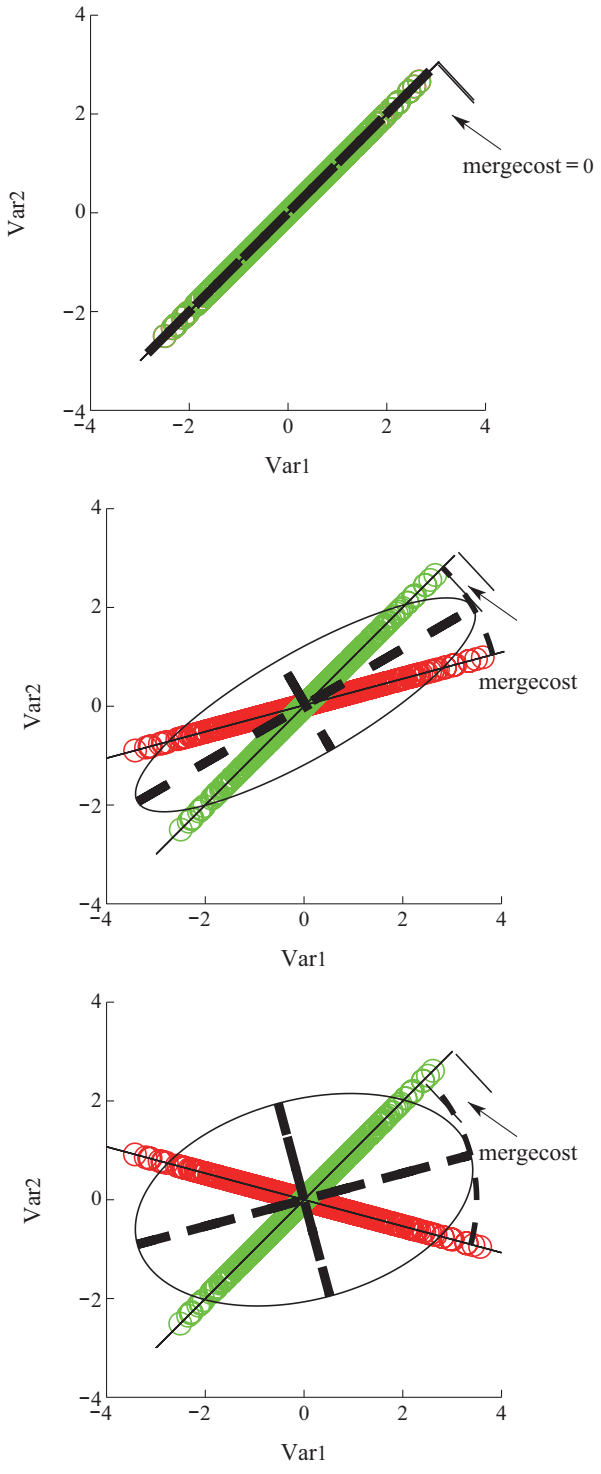
在自底向上策略中, 需要定义合并代价函数来衡量相邻子序列对应的数据间的潜结构相似性. 为此, 在式(1)的基础上, 本文定义一种新的基于解释方差变化的合并代价函数$\mathrm{mergecost}_{D^2}(\cdot)$, 如式(7)所示. 相比文[25]将基于(1)定义的合并后的新子序列的代价$\mathrm{cost}_{D^2}(S(a_c,b_{c+1}))$作为合并代价函数, 式(7)定义的合并代价是合并子序列前后的全局代价函数改变量, 因此在自底向上方法每一次寻找合并代价最低的子序列合并时, 全局代价函数变化最慢.

$$\begin{aligned}&\mathrm{mergecost}_{D^2}(c)=\\&((b_{c+1}-a_c+1)/N)*\mathrm{cost}_{D^2}(S(a_c,b_{c+1}))-\\&((b_{c+1}-a_{c+1}+1)/N)*\mathrm{cost}_{D^2}(S_{c+1}(a_{c+1},b_{c+1}))-\\&((b_c-a_c+1)/N)*\mathrm{cost}_{D^2}(S_c(a_c,b_c)),\end{aligned} \quad (7)$$

其中: $\mathrm{mergecost}_{D^2}(c)$为合并子序列$S_c(a_c,b_c)$和$S_{c+1}(a_{c+1},b_{c+1})$的合并代价, $S(a_c,b_{c+1})$为由$S_c(a_c,b_c)$和$S_{c+1}(a_{c+1},b_{c+1})$合并得到的新子序列.

下面通过例子说明利用基于解释方差变化的合并代价函数来衡量数据潜结构差异的合理性. 假设有一

组经过标准化后的二维数据, 只用一个主元就可以提取全部的方差信息, 用绿色的虚心圆点标识. 将该组数据绕原点顺时针分别旋转0°, 30°, 60°, 90°得到的新数据, 用红色的虚心圆点标识, 这个旋转过程模拟了两组数据的分布差异逐渐增大. 对两组数据单独以及合并组成的新数据利用主元分析建模, 主元个数均选择为1, 如图1所示. 当旋转角度增加时, 也就是数据分布相差越大, 同样的主元数下能够提取过程变化信息就越少, 基于T^2统计量的合并代价越小, 或者说基于Q统计量的合并代价越大.



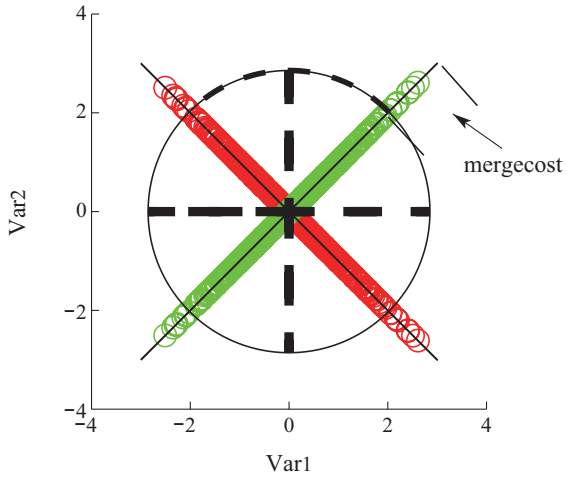


图 1 基于解释方差变化合并代价函数的直观解释

Fig. 1 An intuitive interpretation of explained variance variation based merging cost function

2.1.2 基于动态潜结构的间歇过程阶段划分

第2.1.1节的多变量时间序列分割主要针对连续过程, 而间歇过程具有三维张量数据形式$\underline{\boldsymbol{X}}$ ($I \times J \times K$), 其中I为批次数, J为过程变量数, K为采样样本数. 由于间歇过程的阶段划分潜在假设不同批次的数据在同一时刻过程特性一致, 因此它仍可采用第2.1.1 节中的方法. 区别是数据在每个时刻的样本数量, 连续过程对应的是单个样本, 而间歇过程在每个时刻对应是多个批次的样本组成的时间片矩阵$\boldsymbol{X}_k$ ($I \times J$), $k = 1, 2, \cdots, K$. 另外, 相比于连续过程, 间歇过程采用批次方式标准化的预处理方式, 即对每一个时刻数据标准化, 给采用第2.2.1节提出的相关代价函数进行阶段划分带来5条有益性质.

性质 1 对于任意子序列, 基于T^2统计量的代价函数和基于Q统计量的代价函数之和为常数; 证明见附录A;

性质 2 基于T^2统计量的合并代价函数和基于Q统计量的合并代价函数之和为0, 即$\text{mergecost}_{T^2}(c) + \text{mergecost}_Q(c) = 0$. 由性质1可推导出;

性质 3 基于T^2统计量的全局代价函数与基于Q统计量的全局代价函数之和为常数, 与阶段划分结果无关. 由性质1可推导出;

性质 4 对于阶段划分结果, 选择基于T^2统计量的相关代价函数与基于Q统计量的相关代价函数等价; 由性质2可推导出;

性质 5 在合并子序列过程中, 全局代价函数单调变化, 证明见附录B.

为了提取间歇过程变量间的动态关系, 采用时滞增广技术[22]在时间片数据$\boldsymbol{X}_k$基础上构成增广时间片矩阵$\boldsymbol{X}_{k,d}$, 如式(8)所示. 利用增广矩阵按照式(7)定义合并代价函数来计算动态潜结构相似度, 用于衡量子

序列动态内部结构的差异.

$$\boldsymbol{X}_{k,d} = [\boldsymbol{X}_{k+d} \ \ \boldsymbol{X}_{k+d-1} \ \cdots \ \boldsymbol{X}_k]. \tag{8}$$

基于上述动态潜结构相似度的衡量标准, 本文提出一种基于动态潜结构的间歇过程阶段划分方法(DLSPP, dynamic latent structure based phase partition). 主要思路是对于间歇过程预处理得到的增广数据, 采用自底向上贪心策略寻找动态潜结构相似度最大的相邻子序列合并, 循环执行直到达到目标阶段数停止. 具体地, 由数据预处理和基于自底向上策略的阶段划分两部分组成. 数据预处理包含两个步骤. 第1步, 利用文[27]所述方法进行批次方式标准化和沿变量方向展开成一系列的时间片矩阵数据, $k = 1, 2, \cdots, K$. 第2步, 添加滞后变量构成增广矩阵时间片矩阵数据$\boldsymbol{X}_{k,d}$. 基于自底向上策略的阶段划分算法实现步骤如下:

1) 创建初始子序列集合$S_{\mathcal{T}}^{K} = \{S_c(a_c, b_c) = \{c\} | 1 \leqslant c \leqslant K\}$, 即将每一个时刻作为一个子序列. 根据确定的主元数目A 和第2.1.1节相关代价函数定义, 计算每一个子序列的代价函数$\text{cost}_Q(S_c(a_c, b_c))$和相邻子序列间的合并代价函数$\text{mergecost}_Q(c)$;

2) 判断目前子序列数目是否等于目标的阶段数目C, 如果是, 则执行第5)步; 否则, 执行第3)步;

3) 寻找合并代价$\text{mergecost}_Q(c)$最小对应两个子序列, 记为index, 合并对应S_{index}和$S_{\text{index}+1}$的子序列;

4) 删除$S_{\text{index}+1}$, 更新合并后新的子序列的代价函数和相邻时间片的合并代价函数; 然后返回到第2)步;

5) 后处理步骤: 针对实际间歇过程同一阶段内物理化学反应过程通常要保持一段时间, 对各阶段的数据长度带来物理约束, 对于不满足最小阶段长度的子序列, 将它分配到与它相邻且合并代价最低的子序列之中.

DLSPP算法有4个待定的模型参数, 具体的确定方法说明如下:

阶段划分主元数(A): 由于间歇过程各个阶段内通常是少量潜变量驱动的, 主元数一般选择较小值. 具体确定方式是设置主元数从1开始依次增加, 并记录相应的全局代价函数随阶段数变化的曲线; 针对每条曲线, 利用手肘法确定最佳阶段数量并记录相应的阶段划分结果; 结合过程知识选择为最合理的阶段划分结果所对应的主元数.

滞后阶数(d): 采用并行分析和得分交叉相关性方法选择, 详见文[22].

目标阶段数目(C): 本文采用Hallac等提出的方法确定阶段数目[28], 选择全局代价函数与分段数量C变化曲线肘点作为最佳的阶段数. 当分段数量增加能够使不同结构的数据分开后, 全局代价函数的变化应该由迅速趋于平缓.

最小阶段长度(L): 应结合实际过程关于阶段长度的先验知识来确定. 当没有先验知识时, 默认设置为1.

2.2　子模型统计建模与在线监控

利用阶段划分的结果$S_{\mathcal{T}}^{C} = \{S_c(a_c, b_c) | 1 \leqslant c \leqslant C\}$, 对每个阶段分别利用动态主元分析方法进行统计建模. 对由增广时间片矩阵合并得到的子矩阵$\boldsymbol{X}_d^c = [\boldsymbol{X}_{a_c,d} \ \ \boldsymbol{X}_{a_c+1,d} \ \cdots \ \boldsymbol{X}_{b_c,d}]$, 利用动态主元分析方法建模如式(9)所示. 各子模型的主元数采用累计方差贡献率方法选择[29].

$$\begin{cases} \boldsymbol{T}_d^c = \boldsymbol{X}_d^c \boldsymbol{P}_d^c, \\ \widehat{\boldsymbol{X}}_d^c = \boldsymbol{T}_d^c (\boldsymbol{P}_d^c)^{\mathrm{T}} = \boldsymbol{X}_d^c (\boldsymbol{P}_d^c)^{\mathrm{T}} \boldsymbol{P}_d^c, \\ \boldsymbol{E}_d^c = \boldsymbol{X}_d^c - \widehat{\boldsymbol{X}}_d^c, \end{cases} \tag{9}$$

其中: $c = 1, 2, \cdots, C$为阶段序号, $\boldsymbol{P}_d^c \in \mathbb{R}^{J(d+1) \times R_c}$和$\boldsymbol{T}_d^c \in \mathbb{R}^{I(b_c - a_c + 1) \times R_c}$分别为子矩阵$\boldsymbol{X}_d^c$对应的动态主元分析的载荷矩阵和得分矩阵, $\widehat{\boldsymbol{X}}_d^c \in \mathbb{R}^{I(b_c - a_c + 1) \times J(d+1)}$和$\boldsymbol{E}_d^c \in \mathbb{R}^{I(b_c - a_c + 1) \times J(d+1)}$分别为对应的主元投影矩阵和残差矩阵, R_c表示第c个阶段动态主元分析子模型的主元数.

对于某时刻采集得到的单个样本, Hotelling's T^2统计量和Q统计量可计算如式(10)所示:

$$\begin{cases} T^2 = \boldsymbol{t}^{\mathrm{T}} (\boldsymbol{\Lambda}_d^c)^{-1} \boldsymbol{t}, \\ Q = \boldsymbol{e}^{\mathrm{T}} \boldsymbol{e}, \end{cases} \tag{10}$$

其中: $\boldsymbol{t} = \boldsymbol{P}_d^c \boldsymbol{x}_{k,d} \in \mathbb{R}^{R_c \times 1}$为得分向量, $\boldsymbol{e} = \boldsymbol{x}_{k,d} - (\boldsymbol{P}_d^c)^{\mathrm{T}} \boldsymbol{t} \in \mathbb{R}^{J(d+1) \times 1}$为残差向量, $\boldsymbol{\Lambda}_d^c$为子矩阵$\boldsymbol{X}_d^c$ 对应的由主元特征值组成的对角矩阵.

各阶段分别计算对应上述两个统计量的控制限. Hotelling's T^2统计量的控制限利用式(11)计算[30], Q统计量的控制限利用式(12)计算[31].

$$T_{c;\alpha}^2 = \frac{R_c({N_c}^2 - 1)}{N_c(N_c - R_c)} F_{R_c, N_c - R_c; \alpha}, \tag{11}$$

$$Q_{c;\alpha} = g_c \chi_{h_c;\alpha}^2, \tag{12}$$

其中: $N_c = I(b_c - a_c + 1)$为第c个子模型建模样本个数, α为显著性水平, $F_{R_c, N_c - R_c; \alpha}$是自由度为$R_c$, $N_c - R_c$的F分布的上α分位点, $\chi_{h_c;\alpha}^2$是自由度为h_c的卡方分布的上α分位点, $g_c = \dfrac{v_c}{2m_c}$, $h_c = \dfrac{2m_c^2}{v_c}$, m_c和v_c分别为第c个子模型建模数据Q值的均值和方差.

为了同时监控两个统计指标对应的子空间的异常变化, 通常按文[32]的方式将上述两个指标合并为一个综合指标来监控过程运行.

当获得运行批次当前时刻k实时数据后, 按下述步骤实现动态间歇过程的在线监控:

1) 利用训练数据中相应时刻的均值和方差进行标准化, 得到的结果记为$\boldsymbol{x}_k$. 利用前d 个时刻的数据, 构成增广数据向量$\boldsymbol{x}_{k,d} = [\boldsymbol{x}_{k+d} \ \ \boldsymbol{x}_{k+d-1} \ \cdots \ \boldsymbol{x}_k]$;

2) 利用阶段划分的结果, 判断当前时刻所处的阶段序号c;

3) 利用阶段c对应的负载矩阵$\boldsymbol{P}_d^c$, 计算对应的得分向量$\boldsymbol{t}$和残差向量$\boldsymbol{e}$;

4) 按式(11)和式(12)计算新数据$\boldsymbol{x}_{k,d}$对应的Hotelling's T^2和Q统计量;

5) 判断Hotelling's T^2和Q统计量是否超出相应的控制限. 若两者均未超出, 判定过程当前运行正常; 否则, 判定当前发生异常.

3 青霉素发酵过程仿真案例研究

3.1 过程描述与故障描述

本节利用Cinar等研发的Pensim 2.0青霉素生产仿真软件 (http://www.chbe.iit.edu/~cinar/)[33–34] 开展仿真验证研究. 青霉素补料分批发酵是一个典型的多阶段动态间歇过程, 工艺流程如图2所示. 整个生产周期包含了两个操作时段: 细胞培养阶段(大约持续45 h)与青霉素补料发酵阶段(大约为期355 h).

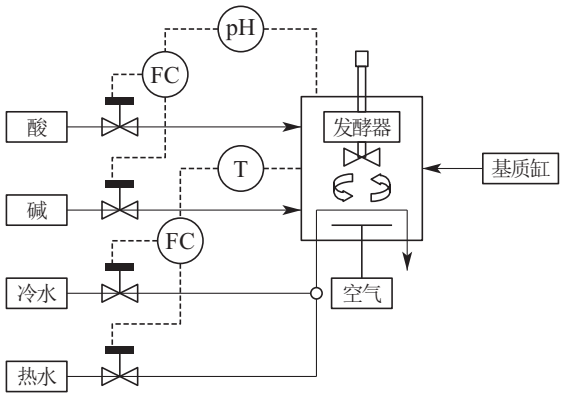


图 2 青霉素发酵过程工艺流程图

Fig. 2 Flow sheet of the penicillin fermentation process

建模变量选择如表1所示. 各批次反应周期设定为400 h, 采样周期为1 h. 设定初始培养基容量在100～120 L之间做高斯随机波动来模拟批次间的正常变化, 产生100个批次的正常运行数据作为建模数据; 测试数据包括与建模数据同分布的100个批次的正常运行数据, 以及2个批次的2种不同类型的故障数据, 故障描述详见第3.2.1节. 用于实验的硬件计算机配置为: Intel i5 CPU, 4 GB处理器.

表 1 建模变量定义

Table 1 Definition of modeling variables

序号	变量名称	序号	变量名称
1	通风率	6	培养基容量
2	搅拌功率	7	二氧化碳浓度
3	基质补料速度	8	pH值
4	基质补料温度	9	产生热量
5	溶解氧浓度	10	冷却水流量

3.2 实验验证与对比分析

3.2.1 实验验证

利用第2.1.2节的动态间歇过程阶段划分方法, 确定滞后阶数为2, 阶段划分主元数为1, 最小阶段长度为1, 全局代价函数相对变化量随阶段数目的变化情况如图3所示. 利用第2.1.2节阶段数确定方法选为10, 将整个过程分为10个阶段, 如图4所示. 具体地, 1～14 h, 15～46 h, 47～286 h, 287～292 h, 293～299 h, 300～322 h, 323～331 h, 332～336 h, 337～362 h, 363～367 h, 368～385 h, 386～391 h, 392～400 h.

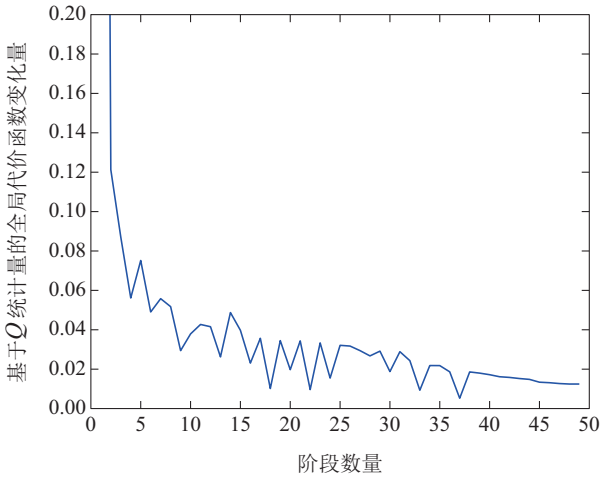


图 3 基于Q统计量的全局代价函数随阶段数变化曲线

Fig. 3 Relative variation of Q statistics based global cost function

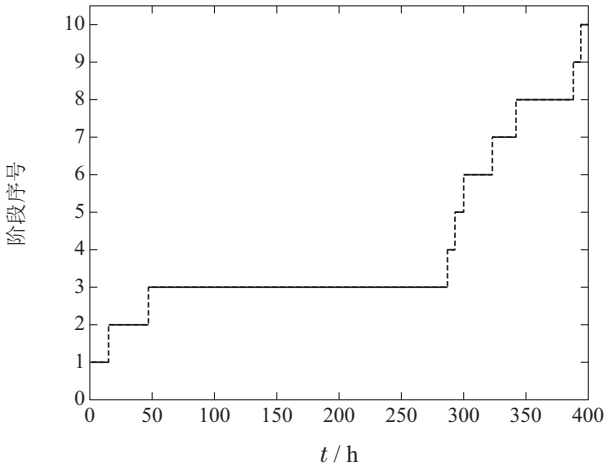


图 4 阶段划分结果的示意图

Fig. 4 Illustration of phase partition results

分析如下: 1～14 h对应了反应迟滞期, 菌体在此时期对新的环境进行适应; 15～46 h对应菌体指数生长期, 此时间段内菌体的浓度随时间指数增加; 47～292 h对应了青霉素的合成期, 这是一个平稳的时期; 而292～400 h对应了菌体衰亡期, 可以注意到整个菌体衰亡期有很多“短”阶段, 这表明此期间过程动态特性变化迅速, 需要多个阶段去近似. 综上, 阶段划分结

果与实际的物理过程存在对应关系, 并且进行了更精细的划分, 表明了本文所提阶段划分方法的有效性.

为了验证在线监控效果, 引入两种不同类型的故障, 如表2所示. 本文所提方法的监控效果如图5–6所示. 从图5可以看出, 在故障1引入后Q统计量快速检测到了故障, 并且在第220 h时故障消失后Q统计量快速恢复到正常的控制限之下. 引入故障2的监控效果如图6所示, 在第155 h左右两个监控指标均超过控制限, 较早地监测到缓变故障的发生. 值得注意的是, Hotelling's T^2统计量和Q统计量均出现了先上升超出控制限、下降再上升的现象. 结合过程知识分析如下: 早期故障发生造成pH值波动, 从而Hotelling's T^2统计量和Q统计量均上升超出控制限; 此后由于闭环反馈控制作用, pH值较快恢复正常, 监控量也随之下降. 但原始故障变化伴随着物质能量的流动和控制作用传播和发展, 造成溶解氧浓度和发酵罐温度等变量随之发生异常, 从而统计量再次上升超过控制限. 结合上述两个案例的分析, 监控结果与实际故障情况一致.

表 2　补料分批青霉素发酵过程故障描述

Table 2　The fault description in the fed-batch penicillin fermentation process

序号	故障描述	故障引入时刻/h	故障结束时刻/h
1	搅拌器速率阶跃性增长10%	100	220
2	基质补料速度以0.1%的斜率增加	150	400

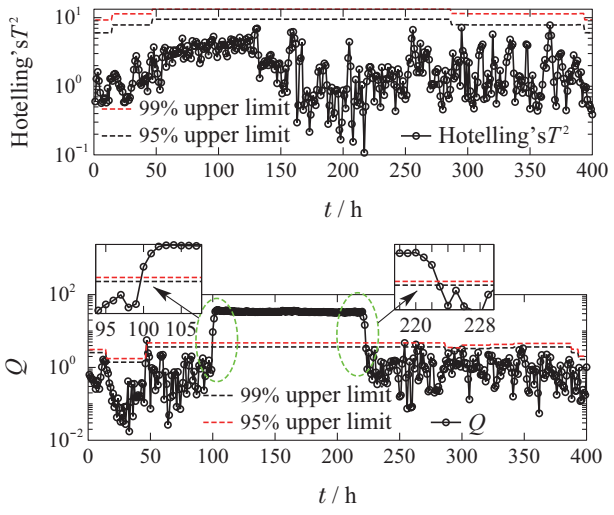


图 5　故障1的监控效果

Fig. 5　Monitoring results of fault 1

3.2.2　比较分析

本节将提出的算法DLSPP与其他多阶段自动划分方法, 包括SSSP[13]方法以及MPPCA[15]方法, 从监控性能、解搜索能力和时间效率这3个方面进行比较. 为了体现不同方法之间的对比公平性, 其它方法也进行动态扩展. 同时, 进行大量实验选择各方法的最佳参数.

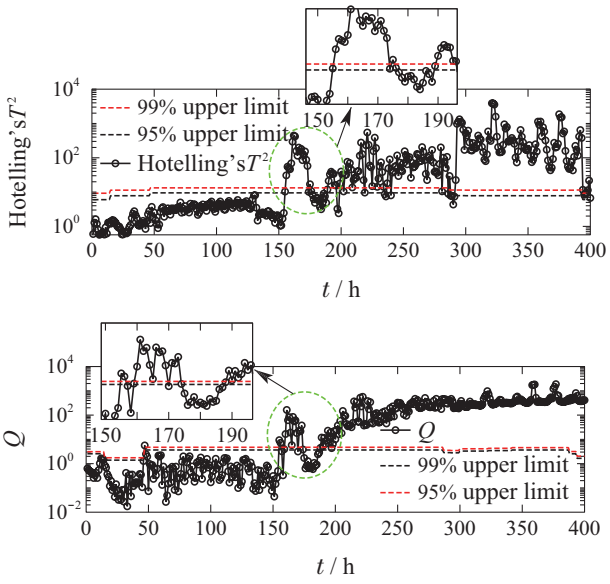


图 6　故障2的监控效果

Fig. 6　Monitoring results of fault 2

为了反映分段合理性, 利用阶段划分结果建立相应的监控模型, 并与其它阶段划分方法就误报率进行比较. 对于正常运行时的测试数据, 设置显著性水平为95%和99%, 各子模型的主元数设置为相同数值, 是由各阶段按照累积方差贡献率90%原则确定的最常出现的主元数, 误报率的比较结果见表3所示. 表3每一列分别给出DLSPP、SSSP 和MPPCA三种方法对于不同统计指标的误报率. 可以看出, 对于基于Hotelling's T^2指标的监控而言, 所提DLSPP方法的误报率更低; 尽管基于Q指标的监控DLSPP方法的误报率略高, 但基于综合指标的监控DLSPP方法误报率更低, 表明Hotelling's T^2指标相对Q指标而言是主导因素. 综上, 基于DLSPP划分结果的误报率更低, 表明本文所提方法的优越性.

表 3　所提方法和其它方法误报率比较

Table 3　Compared false alarm rate between the proposed method and other methods

方法	指标					
	T^2		Q		综合指标	
	95%, 99%		95%, 99%		95%, 99%	
DLSPP	3.82%	1.75%	6.19%	3.85%	6.70%	4.04%
SSPP	4.32%	1.89%	5.59%	3.44%	6.83%	4.22%
MPPCA	4.00%	1.97%	6.20%	3.70%	6.78%	4.05%

本文选择全局代价函数值作为评价解搜索能力的标准. 相同的阶段数和主元数目情况下, 阶段划分越

合理, 能够提取到的过程变化方差信息越多, 那么基于T^2统计量的全局代价函数越大或者基于Q 统计量的全局代价函数越小. 所提方法与其它方法划分结果对应的基于Q统计量的全局代价函数随阶段数目的变化如图7所示.

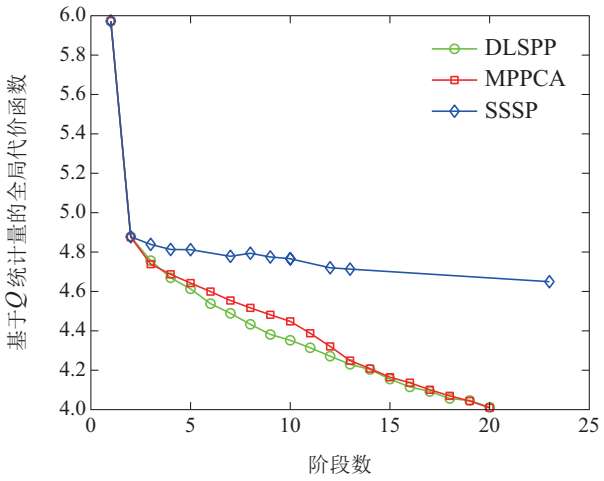


图 7 基于Q统计量全局代价函数随阶段数变化曲线

Fig. 7 Q statistics based global cost with number of phases

可以看出, 相同阶段数DLSPP方法的全局代价函数较小、SSSP最大. 需要说明的是, 由于在MPPCA 和本文所提DLSPP方法中, 阶段数均为输入参数, 是算法的停止条件, 因此可以设置为连续的自然数. 然而, 基于滑动窗口策略的SSSP方法, 通过对阶段内过程特性变化的容忍系数来确定该阶段是否结束, 造成SSSP的阶段数随容忍因子变化, 无法保证在连续自然数中取值. 故图中SSSP曲线的点数少于其他两种方法.

不同阶段划分算法在本案例的运行时间如表4所示. 可以看出, DLSPP时间效率高, 而MPPCA耗时较长. 这也验证了本文所采用的自底向上策略, 相较于MPPCA所采用的的自顶向下策略, 在时间效率上具有明显优势.

表 4 运行时间比较

Table 4 Comparison results of running time

划分方法	DLSPP	SSSP	MPPCA
运行时间/s	15.6094	26.2188	348.2813

4 结论

本文提出基于动态潜结构的动态间歇过程阶段划分与在线监控方法. 研究表明: 1) 阶段划分结果与在线监控效果验证了以解释方差变化的合并代价函数作为相似度度量的合理性; 2) 阶段划分的求解策略采用自底向上启发式算法, 时间效率高; 3) 采用全局代价函数随时间变化曲线作为阶段数选择依据, 参数易确定.

本文假设不同批次的运行周期相同, 实际间歇过程运行中常发生运行周期不一致因而造成数据不等长, 下一步可针对不等长的间歇过程阶段划分与过程监控开展研究.

参考文献:

[1] NOMIKOS P, MACGREGOR J F. Monitoring batch processes using multiway principal components analysis. *AIChE Journal*, 1994, 40(8): 1361 – 1375.

[2] NOMIKOS P, MACGREGOR J F. Multi-way partial least squares in monitoring batch processes. *Chemometrics & Intelligent Laboratory Systems*, 1995, 30(1): 97 – 108.

[3] ZHAO Chunhui, YU Wanke, GAO Furong. Data analytics and condition monitoring methods for nonstationary batch processes – current status and future. *Acta Automatica Sinica*, 2020, 46(10): 2072 – 2091.
(赵春晖, 余万科, 高福荣. 非平稳间歇过程数据解析与状态监控–回顾与展望. 自动化学报, 2020, 46(10): 2072 – 2091.)

[4] HE Q P, WANG J. Statistics pattern analysis: a new process monitoring framework and its application to semiconductor batch processes. *AIChE Journal*, 2011, 57(1): 107 – 121.

[5] GUO J, WANG X, LI Y. kNN based on probability density for fault detection in multimodal processes. *Journal of Chemometrics*, 2018, 32: 1 – 14.

[6] HE F, XU J W. A novel process monitoring and fault detection approach based on statistics locality preserving projections. *Journal of Process Control*, 2016, 37(5): 46 – 57.

[7] XIE Yanhong, SUN Chengao, LI Yuan. Fault monitoring of batch process based on moving window SVDD. *Information and Control*, 2015, 44(5): 531 – 537.
(谢彦红, 孙呈敖, 李元. 基于滑动窗口SVDD的间歇过程故障监测. 信息与控制, 2015, 44(5): 531 – 537.)

[8] UNDEY C, CINAR A. Statistical monitoring of multistage, multiphase batch processes. *IEEE Control Systems*, 2002, 22(5): 40 – 52.

[9] KOSANOVICH K A, DAHL K S, PIOVOSO M J. Improved process understanding using multiway principal component analysis. *Industrial & Engineering Chemistry Research*, 1996, 35(1): 138 – 146.

[10] LOUWERSE D J, SMILDE A K. Multivariate statistical process control of batch processes based on three-way models. *Chemical Engineering Science*, 2000, 55(7): 1225 – 1235.

[11] CAMACHO J, PICO J. Online monitoring of batch processes using multi-phase principal component analysis. *Journal of Process Control*, 2006, 16(10): 1021 – 1035.

[12] LU N Y, GAO F R, WANG F L. Sub-PCA modeling and on-line monitoring strategy for batch processes. *AIChE Journal*, 2004, 50(1): 255 – 259.

[13] ZHAO C H, WANG F L, LU N Y, et al. Stage-based soft-transition multiple PCA modeling and online monitoring strategy for batch processes. *Journal of Process Control*, 2007, 17(9): 728 – 741.

[14] WANG Shu, CHANG Yuqing, YANG Jie, et al. Multiway principle component analysis monitoring and fault variable detection based on sub-stage separation for batch processes. *Control Theory & Applications*, 2011, 28(2): 149 – 156.
(王姝, 常玉清, 杨洁, 等. 时段划分的多向主元分析间歇过程监测及故障变量追溯. 控制理论与应用, 2011, 28(2): 149 – 156.)

[15] YU J, QIN S J. Multiway gaussian mixture model based multiphase batch process monitoring. *Industrial & Engineering Chemistry Research*, 2009, 48(18): 8585 – 8594.

[16] CAMACHO J, PICO J. Multi-phase principal component analysis for batch processes modelling. *Chemometrics & Intelligent Laboratory Systems*, 2006, 81(2): 127 – 136.

[17] ZHAO C H, SUN Y X. Step-wise sequential phase partition (SSPP) algorithm based statistical modeling and online process monitoring. *Chemometrics & Intelligent Laboratory Systems*, 2013, 125(15): 109 – 120.

[18] LI Wenqing, ZHAO Chunhui, SUN Youxian. Sequential unequal-length phase identification and modelling method for fault detection of varying-duration batch processes. *Control Theory & Applications*, 2015, 32(9): 1226 – 1232.
(李文卿, 赵春晖, 孙优贤. 不等长批次过程的有序时段划分、建模及故障检测. 控制理论与应用, 2015, 32(9): 1226 – 1232.)

[19] SHANG C, HUANG B, YANG F. Slow feature analysis for monitoring and diagnosis of control performance. *Journal of Process Control*, 2016, 39: 21 – 34.

[20] DONG Y N, QIN S J. A novel dynamic PCA algorithm for dynamic data modeling and process monitoring. *Journal of Process Control*, 2018, 67: 1 – 11.

[21] ZOU X Y, ZHAO C H. Concurrent assessment of process operating performance with joint static and dynamic analysis. *IEEE Transactions on Industrial Informatics*, 2019, 16(4): 2776 – 2786.

[22] KU W, STORER R H, GEORGAKIS C. Disturbance detection and isolation by dynamic principal component analysis. *Chemometrics & Intelligent Laboratory Systems*, 1995, 30(1): 179 – 196.

[23] CHEN J, LIU K C. On-line batch process monitoring using dynamic PCA and dynamic PLS models. *Chemical Engineering Science*, 2002, 57(1): 63 – 75.

[24] KEOGH E, CHU S, Hart D, et al. An online algorithm for segmenting time-series. *Proceedings 2001 IEEE International Conference on Data Mining*. San Jose, CA, USA: IEEE, 2001, 289 – 296.

[25] ABONYI J, FEIL B, NEMETH S, et al. Modified Gath-Geva clustering for fuzzy segmentation of multivariate time-series. *Fuzzy Sets and Systems*, 2005, 149: 39 – 56.

[26] HIMBERG J, KORPIAHO K, MANNILA H, et al. Time-series segmentation for context recognition in mobile devices. *Proceedings of the 2001 IEEE International Conference on Data Mining*. San Jose, CA, USA: IEEE, 2002: 203 – 210.

[27] WESTERHUIS J A, KOURTI T, MACGREGOR J F. Comparing alternative approaches for multivariate statistical analysis of batch process data. *Journal of Chemometrics*, 1999, 13(3/4): 397 – 413.

[28] HALLAC D, NYSTRUP P, BOYD S. Greedy gaussian segmentation of multivariate time series. *Advances in Data Analysis and Classification*, 2019, 13: 727 – 751.

[29] JACKSON J E. *A User's Guide to Principal Components*. New York: Wiley, 1991.

[30] NOMIKOS P, MACGREGOR J F. Multivariate SPC charts for batch processes. *Technometrics*, 1995, 37(1): 41 – 59.

[31] TRACY N D, YOUNG J C, Mason R L. Multivariate control charts for individual observations. *Journal of Quality Technology*, 1992, 24(2): 88 – 95.

[32] YUE H H, QIN S J. Reconstruction-based fault identification using a combined index. *Industrial & Engineering Chemistry Research*, 2001, 40(20): 4403 – 4414.

[33] BAJPAI R, REUSS M. A mechanistic model for penicillin production. *Journal of Chemical Technology and Biotechnology*, 1980, 30: 332 – 344.

[34] GULNUR B, UNDEY C, CINAR A. A modular simulation package for fed-batch fermentation: penicillin production. *Computer and Chemical Engineering*, 2002, 26(11): 1553 – 1565.

附录 A

对于任意子序列, 基于T^2统计量的代价函数和基于Q统计量的代价函数之和为常数.

经过间歇过程批次方式的标准化后, 对每个时刻进行去均值和方差归一化后,

$$\begin{cases} \sum\limits_{i=1}^{I} x_{i,j,k} = 0, \\ \dfrac{1}{I}\sum\limits_{i=1}^{I} x_{i,j,k}^2 = 1, \end{cases} \tag{A.1}$$

其中$x_{i,j,k}^2$代表第i个批次第k个时刻第j个传感器的标准化后的值.

利用式(3)和式(4)关于$T_{i,k}^2$和$Q_{i,k}$的定义,得

$$T_{i,k}^2 + Q_{i,k} = \|\boldsymbol{x}_{i,k}\|^2, \tag{A.2}$$

其中$\boldsymbol{x}_{i,k}$为第i个批次第k个时刻标准化后的数据.

利用式(A.1)和(A.2), 可得

$$\sum_{i=1}^{I}(T_{i,k}^2 + Q_{i,k}) = \sum_{i=1}^{I}\|\boldsymbol{x}_{i,k}\|^2 = \sum_{i=1}^{I}\sum_{j=1}^{J}(x_{i,j,k})^2 = \sum_{j=1}^{J}\sum_{i=1}^{I}(x_{i,j,k})^2 = \sum_{j=1}^{J} I = J * I. \tag{A.3}$$

进一步利用式(A.3), 可得

$$\sum_{k=a_c}^{b_c}\sum_{i=1}^{I}(T_{i,k}^2 + Q_{i,k}) = (b_c - a_c + 1) * I * J. \tag{A.4}$$

基于间歇过程数据形式, 代入式(1)定义的代价函数得

$$\begin{aligned} \mathrm{cost}_{T^2}(S_c(a_c, b_c)) &= \frac{1}{(b_c - a_c + 1) * I}\sum_{k=a_c}^{b_c}\sum_{i=1}^{I} T_{i,k}^2, \\ \mathrm{cost}_{Q}(S_c(a_c, b_c)) &= \frac{1}{(b_c - a_c + 1) * I}\sum_{k=a_c}^{b_c}\sum_{i=1}^{I} Q_{i,k}. \end{aligned} \tag{A.5}$$

将式(A.4)代入式(A.5)可得

$$\mathrm{cost}_{T^2} S_c(a_c, b_c) + \mathrm{cost}_{Q} S_c(a_c, b_c) = J. \tag{A.6}$$

故对于任意子序列, 基于T^2统计量的代价函数和基于Q统计量的代价函数之和为常数, 得证.

附录 B

在合并子序列过程中, 全局代价函数单调变化.

由于合并代价函数$\mathrm{mergecost}_{D^2}(c)$恰好是在合并子序列$S_c(a_c, b_c)$和子序列$S_{c+1}(a_{c+1}, b_{c+1})$前后的全局代价函数改变量, 因此只需证明合并代价函数恒小于等于0或恒大于等于0即可. 下面以基于T^2统计量构造的相关代价函数为例证明.

由主元分析性质[29]知

$$\begin{cases} T^2 = \max\limits_{\boldsymbol{P}} (1/N) * \mathrm{Trace}(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}\boldsymbol{P})), \\ \text{s.t. } \boldsymbol{P}^{\mathrm{T}}\boldsymbol{P} = \boldsymbol{I}. \end{cases} \tag{B.1}$$

其中: $\boldsymbol{P}$为数据矩阵$\boldsymbol{X} \in \mathbb{R}^{N\times m}$对应主元分析载荷矩阵, $\boldsymbol{I}$为单位矩阵.

将式(B.1)代入代价函数定义(1), 可得

$$\begin{cases}\text{cost}_{T^2}(S_c(a_c,b_c))=\\ \max\limits_{\boldsymbol{P}}(1/((b_c-a_c+1)*I)*\text{Trace}(\boldsymbol{P}^{\text{T}}\boldsymbol{X}_c^{\text{T}}\boldsymbol{X}_c\boldsymbol{P}),\\ \text{s.t. }\boldsymbol{P}^{\text{T}}\boldsymbol{P}=\boldsymbol{I}.\end{cases}\tag{B.2}$$

由不等式性质,

$$\max_x(f_1(x)+f_2(x))\leqslant\max_x f_1(x)+\max_x f_2(x),$$

同理可得

$$\begin{cases}\max\limits_{\boldsymbol{P}}\text{Trace}(\boldsymbol{P}^{\text{T}}(\boldsymbol{X}_c^{\text{T}}\boldsymbol{X}_c+\boldsymbol{X}_{c+1}^{\text{T}}\boldsymbol{X}_{c+1})P)\leqslant\\ \max\limits_{\boldsymbol{P}}\text{Trace}(\boldsymbol{P}^{\text{T}}\boldsymbol{X}_c^{\text{T}}\boldsymbol{X}_c\boldsymbol{P})+\\ \max\limits_{\boldsymbol{P}}\text{Trace}(\boldsymbol{P}^{\text{T}}\boldsymbol{X}_{c+1}^{\text{T}}\boldsymbol{X}_{c+1}\boldsymbol{P}).\end{cases}\tag{B.3}$$

将式(B.2)代入式(7)的合并代价函数mergecost$(\cdot)$定义, 可得

$$\begin{aligned}&\text{mergecost}_{T^2}(c)=\\ &(1/N)*\text{Trace}(\boldsymbol{P}^{\text{T}}(\boldsymbol{X}_{c+1}^{\text{T}}\boldsymbol{X}_{c+1}+\boldsymbol{X}_c^{\text{T}}\boldsymbol{X}_c)\boldsymbol{P})-\\ &(1/N)*\text{Trace}(\boldsymbol{P}^{\text{T}}(\boldsymbol{X}_{c+1}^{\text{T}}\boldsymbol{X}_{c+1})\boldsymbol{P})-\\ &(1/N)*\text{Trace}(\boldsymbol{P}^{\text{T}}(\boldsymbol{X}_c^{\text{T}}\boldsymbol{X}_c)P).\end{aligned}\tag{B.4}$$

由式(B.3)知, $\text{mergecost}_{T^2}(c)\leqslant 0$, 因此在子序列合并过程中, 基于$T^2$统计量的全局代价函数单调递减. 同理可证, 在子序列合并过程中, 基于Q统计量的全局代价函数单调递增.

作者简介:

胡 磊 硕士研究生, 研究方向为工业过程统计过程监控与故障诊断, E-mail: 1800757@stu.neu.edu.cn;

刘 强 教授, 博士生导师, 研究方向为数据驱动的建模、过程监控与故障诊断, E-mail: liuq@mail.neu.edu.cn;

吴永建 副教授, 研究方向为工业过程建模与控制, E-mail: wyj_neu@163.com;

范自柱 教授, 研究方向为统计机器学习和模式识别, E-mail: zzfan3@163.com.