基于最近克罗内克积分解的子带自适应滤波器：算法与应用

叶建宏、赵海泉，IEEE高级会员

arXiv:2601.10078v1 [eess.AS] 15 Jan 2026

摘要——近期，基于最近克罗内克积分解（NKP）的归一化最小均方（NLMS-NKP）算法显示出比传统NLMS算法更优的收敛性能。然而，在处理高度相关的输入信号时，其收敛速度会显著下降。为解决这一问题，我们提出一种I型基于NKP的归一化子带自适应滤波器（NSAF）算法，即NSAF-NKP-I。不过，该算法的计算开销远高于NLMS-NKP算法。值得注意的是，我们改进的II型基于NKP的NSAF（NSAF-NKP-II）算法在实现相同性能的收敛表现的同时，显著降低了计算复杂度。此外，为增强算法抵抗脉冲噪声干扰的鲁棒性，我们开发了两种鲁棒变体：基于最大相关熵准则的鲁棒NSAF-NKP（RNSAF-NKP-MCC）算法，以及基于对数准则的鲁棒NSAF-NKP（RNSAF-NKP-LC）算法。此外，我们还对所提算法的计算复杂度、步长范围和理论稳态性能进行了详细分析。为提升NSAF-NKP-II算法在复杂非线性环境中的实用性，我们进一步设计了两种非线性实现：基于三角函数连接网络的NKP-NSAF（TFLN-NSAF-NKP）算法，以及基于Volterra 级数展开的NKP-NSAF（Volterra-NKP-NSAF）算法。在主动噪声控制（ANC）系统中，我们进一步提出滤波-x NSAF-NKP-II（NKP-FxNSAF）算法。我们在回声消除、稀疏系统辨识、非线性处理和ANC场景中开展了仿真实验，以验证所提算法优于现有最先进的同类算法。

索引词—主动噪声控制、回声消除、脉冲噪声、最近克罗内克积、非线性系统辨识、子带自适应滤波器。

I. 引言

ADAPTIVE 滤波算法是在线算法中典型的一类，广泛应用于许多实时系统的设计，例如回声消除 [1], 语音增强 [2], 波束形成 [3], 和主动噪声控制（ANC） [4]–[6]。最小均方（LMS） [7] 和归一化LMS（NLMS）算法 [8], [9] 是两种经典算法，因结构简单、易于实现而受到研究人员的广泛关注。然而，现有研究表明，NLMS算法在处理高度相关的输入信号时，收敛性能较差近几十年来，研究人员针对这一问题开展了大量富有见地的研究。作为NLMS算法在时间轴上的扩展，仿射投影（AP）算法[10] 在输入信号高度相关的情况下能够比NLMS算法更快地收敛。尽管如此，随着投影阶数增加，AP算法的收敛性能会有所提升，但计算复杂度也相应增加。递归最小二乘（RLS）算法 [11]–[12] 基于最小二乘原理推导而来，相较于AP算法具有更快的收敛速度，但由于其固有的矩阵求逆运算，计算开销更大。为满足强相关输入信号处理算法在收敛速度与计算复杂度之间取得平衡的需求，Lee和Gan提出了归一化子带自适应滤波器（NSAF）算法 [13]。该技术利用分析滤波器组，将高度相关的输入信号分解为近似白化的子带输入信号。因此，与传统的NLMS算法相比，NSAF算法在保持计算复杂度相近的同时，收敛速度更快。

在许多涉及回声环境的应用中，需要估计的系统脉冲响应（IR）（例如房间脉冲响应（RIR）或水声信道响应）通常很长。为了准确地对该响应建模，必须相应地增加自适应FIR滤波器的长度，这会显著提高算法的计算复杂度。此外，较长的IR通常伴随着输入信号自相关矩阵更大的特征值扩展，导致滤波器收敛速度变慢。近期，研究人员提出了一种有趣的最近克罗内克积（NKP）分解技术 [14]–[17]。该技术将辨识单个长系统的问题转换为辨识两个较短系统的问题，从而有效缩短等效滤波器的长度。由于自适应算法的收敛速度与自适应滤波器的长度成反比 [18], , 因此NKP分解技术能够提升自适应算法的收敛性能，尤其适用于高度稀疏的系统 [19]–[23]。迄今为止，这种NKP分解技术已成功应用于不同的应用框架，例如系统辨识 [24], 主动噪声控制 [25],非线性滤波 [19], 语音去混响 [20], 以及波束形成 [21], [22]。在主动噪声控制系统中，缩短子滤波器长度所带来的计算效率提升

本研究得到国家自然科学基金委员会（NSF）项目（批准号：62171388、61871461 和 61571374）的资助。（通讯作者：赵海泉）。叶建宏和赵海泉就职于西南交通大学电气工程学院，地址：中国成都 611756。电子邮箱：yjh zcl@163.com（J. Ye），hqzhao swjtu@126.com（H. Zhao）。

提高了在低功耗设备上实现滤波-x结构的可行性。对于助听器应用，利用NKP结构对实时反馈路径建模，可以有效降低系统延迟。具体而言，基于NKP的归一化最小均方（NLMS-NKP）算法 [23] 通过略微增加计算复杂度，提升了NLMS算法在稀疏系统辨识中的收敛速度。在 [26], 中，基于NKP的递归最小二乘（RLS-NKP）算法的性能略逊于RLS算法，但大幅降低了计算复杂度。此外，近期研究了基于NKP的AP算法在多通道ANC场景中的应用，并由此提出基于NKP的多通道滤波-x AP（NKP-MFxAP）算法 [25], 。与传统滤波-x AP（FxAP）算法相比，该算法能够更有效地抑制不需要的噪声。遗憾的是，NKP-MFxAP和RLS-NKP算法中的分解后短长度自适应滤波器仍涉及矩阵运算，因此其计算复杂度显著高于NLMS-NKP算法。

此外，上述算法的稳定收敛条件取决于加性噪声服从高斯分布这一事实。然而，当噪声为非高斯噪声时（例如，α-稳定噪声），其脉冲响应会呈现尖锐脉冲或偶发突发 [27]。在这种影响下，自适应滤波器的性能将急剧下降，甚至发散。为提高算法的鲁棒性，诸如M估计 [28], 最小均值p-幂误差（MPE）准则 [29]–[32],以及基于信息论学习方案的相关熵准则 [33],[34] 等方法已被证明是可行的鲁棒策略。在 [35], 脉冲噪声场景下，基于NKP的广义最大相关熵准则（NKP-GMCC）和基于NKP的广义双曲正割函数（NKP-GHSAF）算法能够实现稳定收敛。

上述所有方法实现令人满意性能的前提是自适应控制过程为线性的。然而，现实环境中的复杂系统通常表现出强烈的非线性特征 [36], [37]。为解决这一局限，研究人员开发了基于函数连接网络（FLN） [38], [39], 、沃尔泰拉级数 [40], [41], 、核方法 [42], [43], 和样条自适应滤波器 [44], [45] 的算法。在 [19], Bhattacharjee等人提出了基于三角函数FLN的NKP NLMS（TFLN-NKP-NLMS）和基于沃尔泰拉的NKP NLMS（Volterra-NKP-NLMS）算法，这些算法在非线性系统辨识任务中表现出良好的性能。据我们所知，无论在线性还是非线性场景下，尚未有关于子带NKP分解算法的研究报道。因此，基于这一原因，我们围绕子带NKP处理开展了以下研究。

1) 本文基于NKP分解技术提出了一种I型基于NKP的NSAF（NSAF-NKP-I）算法，其收敛速度比传统NSAF算法更快。然而，这种性能提升是以显著增加计算开销为代价的。为缓解这一局限，我们进一步通过修改子带NKP模型结构，提出了一种II型基于NKP的NSAF（NSAF-NKP-II）算法。值得注意的是，NSAF-NKP-II算法只需显著减少的计算资源，即可达到与其I型对应算法相当的收敛性能。

2) 为提高 NSAF-NKP-II算法对脉冲干扰的鲁棒性，本文提出了两种 NSAF-NKP-II框架的鲁棒变体：基于最大相关熵准则（MCC）的鲁棒 NSAF-NKP-II（RNSAF-NKP-MCC）算法，以及基于对数准则的鲁棒 NSAF-NKP-II（RNSAF-NKP-LC）算法。

3) 我们建立了 NSAF-NKP-II算法的理论稳态模型和理论步长范围，分析了所提出的 NSAF-NKP-I、NSAF-NKP-II、RNSAF-NKP-MCC 和 RNSAF-NKP-LC算法的计算复杂度，并将其与 NSAF、NLMS-NKP、APA-NKP 和 RLS-NKP算法进行了比较。

4) 为增强所提出的 NSAF-NKP-II算法在复杂非线性环境（例如非对称扬声器失真和对称软削波非线性）中的实用性，同时弥补现有非线性子带NKP分解方法的不足，我们提出了两种创新的非线性子带NKP结构。具体而言，我们基于三角函数FLN（TFLN）提出了基于TFLN的NSAF-NKP-II（TFLN-NKP-NSAF）算法。此外，我们还基于 Volterra 级数展开，开发了基于沃尔泰拉的NSAF-NKP-II（Volterra-NKP-NSAF）算法。

5) 为了验证 NSAF-NKP-II 算法在 ANC 场景中的有效性，我们基于 NKP分解技术构建了一个前馈子带ANC系统，并提出了滤波-x NSAF-NKP-II（NKP-FxNSAF）算法。

符号说明：$||.||_2$ 表示欧几里得范数，$||.||_F$ 表示 Frobenius 范数，$(.)^{\mathrm{T}}$ 表示转置运算，E{.}表示期望运算，vec(.) 表示向量化运算，$\otimes$ 表示克罗内克积。

II. 克罗内克积分解

设 $\boldsymbol{m}_0$ 表示待辨识未知系统的实值脉冲响应，其大小为 $D = D_1 \times D_2$，其中 $D_1 > D_2$。接下来，我们可以将未知脉冲响应 $\boldsymbol{m}_0$ 分解为 $\boldsymbol{m}_0 = [\boldsymbol{t}_1^{\mathrm{T}}, \boldsymbol{t}_2^{\mathrm{T}}, ..., \boldsymbol{t}_{D_2}^{\mathrm{T}}]^{\mathrm{T}}$，其中 $\{\boldsymbol{t}_i\}_{i=1}^{D_2}$ 表示长度为 D_1 的短脉冲响应，并假设它们之间具有强线性相关性 [46]。根据 [14], $\boldsymbol{m}_0$ 可以近似表示为 $\boldsymbol{m}_2 \otimes \boldsymbol{m}_1 = \mathrm{vec}(\boldsymbol{m}_1 \boldsymbol{m}_2^{\mathrm{T}})$，其中 $\boldsymbol{m}_1$ 和 $\boldsymbol{m}_2$ 是两个较短的脉冲响应，其大小分别为 $D_1 \times 1$ 和 $D_2 \times 1$。为了评估 $\boldsymbol{m}_0$ 与 $\boldsymbol{m}_2 \otimes \boldsymbol{m}_1$ 之间的近似精度，我们引入如下归一化失配度量：

$$\omega \triangleq ||\boldsymbol{m}_0 - \boldsymbol{m}_2 \otimes \boldsymbol{m}_1||_2 / ||\boldsymbol{m}_0||_2. \tag{1}$$

接下来，$\boldsymbol{m}_0$ 可以重新整理为矩阵形式：$\boldsymbol{M}_0 = [\boldsymbol{t}_1, \boldsymbol{t}_2, ..., \boldsymbol{t}_{D_2}]$，其维度为 $D_1 \times D_2$。因此，(1) 可写为

$$\omega \triangleq ||\boldsymbol{M}_0 - \boldsymbol{m}_1 \boldsymbol{m}_2^{\mathrm{T}}||_F / ||\boldsymbol{M}_0||_F. \tag{2}$$

在这种形式下，我们的目标变为找到合适的 $\boldsymbol{m}_1$ 和$\boldsymbol{m}_2$ ，使 ω最小。有趣的是，最小化 $\omega(\boldsymbol{m}_1,\boldsymbol{m}_2)$的 Frobenius 范数，等价于找出与 $\boldsymbol{M}_0$ [47]最近的秩-1 矩阵。奇异值分解（SVD）为这一最优近似提供了数学基础 [48]。因此，我们有

$$\boldsymbol{M}_0=\boldsymbol{H}_1\boldsymbol{\Sigma}\boldsymbol{H}_2^{\mathrm{T}}=\sum_{i=1}^{D_2}\rho_i\boldsymbol{h}_{1,i}\boldsymbol{h}_{2,i}^{\mathrm{T}}. \tag{3}$$

此处， $\boldsymbol{\Sigma}$ 表示一个 $D_1\times D_2$ 矩形对角矩阵，其中包含按降序排列的 $\boldsymbol{M}_0$ 的奇异值，即 $\rho_1\geq\rho_2\geq\ldots\geq\rho_{D_2}\geq 0$；大小为 $D_1\times D_1$ 的矩阵 $\boldsymbol{H}_1$和大小为 $D_2\times D_2$ 的矩阵 $\boldsymbol{H}_2$ 分别由 $\boldsymbol{M}_0$的左、右奇异向量构成，且 $\boldsymbol{H}_1$ 和 $\boldsymbol{H}_2$ 的第i列分别为$\boldsymbol{h}_{1,\,i}$ 和$\boldsymbol{h}_{2,\,i}$。据此， $\boldsymbol{m}_1$ 和 $\boldsymbol{m}_2$ 的最优脉冲响应分别为$\breve{\boldsymbol{m}}_1=\sqrt{\rho_1}\boldsymbol{h}_{1,\,1}$ 和$\breve{\boldsymbol{m}}_2=\sqrt{\rho_1}\boldsymbol{h}_{2,\,1}$。然而，在更一般的情况下， $\{\boldsymbol{t}_i\}_{i=1}^{D_2}$ 可能并非强线性相关。在此类情形下， $\boldsymbol{m}_0$ 可近似表示为

$$\begin{aligned}\boldsymbol{m}_0&\approx\sum_{p=1}^{P}\boldsymbol{m}_{2,p}\otimes\boldsymbol{m}_{1,p}=\mathrm{vec}\left(\sum_{p=1}^{P}\boldsymbol{m}_{1,p}\boldsymbol{m}_{2,p}^{\mathrm{T}}\right)\\&=\mathrm{vec}\left(\boldsymbol{M}_1\boldsymbol{M}_2^{\mathrm{T}}\right)\end{aligned} \tag{4}$$

其中 $P\leq D_2$， $\boldsymbol{m}_{1,\,p}$ 和$\boldsymbol{m}_{2,\,p}$ 分别表示长度为 D_1 和 D_2的权重向量，而大小为$D_1\times P$的 $\boldsymbol{M}_1$ 和大小为 $D_2\times P$的 $\boldsymbol{M}_2$ 则构造如下:

$$\boldsymbol{M}_1=\left[\boldsymbol{m}_{1,1}\ \boldsymbol{m}_{1,2}\ \ldots\ \boldsymbol{m}_{1,P}\right], \tag{5}$$

$$\boldsymbol{M}_2=\left[\boldsymbol{m}_{2,1}\ \boldsymbol{m}_{2,2}\ \ldots\ \boldsymbol{m}_{2,P}\right]. \tag{6}$$

因此，式（2）的近似问题可等价地转化为最小化

$$\omega\triangleq||\boldsymbol{M}_0-\boldsymbol{M}_1\boldsymbol{M}_2^{\mathrm{T}}||_F/||\boldsymbol{M}_0||_F, \tag{7}$$

由此可得最优解:

$$\begin{aligned}\breve{\boldsymbol{M}}_1=&\left[\breve{\boldsymbol{m}}_{1,1}\ \breve{\boldsymbol{m}}_{1,2}\ \ldots\ \breve{\boldsymbol{m}}_{1,P}\right]\\=&\left[\sqrt{\rho_1}\boldsymbol{h}_{1,1},\sqrt{\rho_2}\boldsymbol{h}_{1,2},\ldots,\sqrt{\rho_P}\boldsymbol{h}_{1,P}\right],\end{aligned} \tag{8}$$

$$\begin{aligned}\breve{\boldsymbol{M}}_2=&\left[\breve{\boldsymbol{m}}_{2,1}\ \breve{\boldsymbol{m}}_{2,2}\ \ldots\ \breve{\boldsymbol{m}}_{2,P}\right]\\=&\left[\sqrt{\rho_1}\boldsymbol{h}_{2,1},\sqrt{\rho_2}\boldsymbol{h}_{2,2},\ldots,\sqrt{\rho_P}\boldsymbol{h}_{2,P}\right].\end{aligned} \tag{9}$$

根据式（8）和式（9）， $\boldsymbol{m}_0$ 的最优近似为

$$\breve{\boldsymbol{m}}_0=\sum_{p=1}^{P}\breve{\boldsymbol{m}}_{2,p}\otimes\breve{\boldsymbol{m}}_{1,p}=\sum_{p=1}^{P}\rho_p\boldsymbol{h}_{2,p}\otimes\boldsymbol{h}_{1,p}. \tag{10}$$

III. 所提出的 NSAF-NKP 算法

A. NSAF-NKP-I 算法的推导

在线性系统辨识场景中，时刻 r 的输入向量 $\boldsymbol{x}_r=[x_r,x_{r-1},\ldots,x_{r-D+1}]^{\mathrm{T}}$ 与期望信号 d_r 具有如下关系:

$$d_r=\boldsymbol{x}_r^{\mathrm{T}}\boldsymbol{m}_0+v_r, \tag{11}$$

其中， v_r 表示加性噪声，它在统计上与输入向量 $\boldsymbol{x}_r$相互独立。

图 1 描述了具有 N 个子带的 NSAF-NKP-I 算法结构。“合成器”用于将短子滤波器 $\hat{}\boldsymbol{m}_{r,2}$ 和 $\hat{}\boldsymbol{m}_{r,1}$ 组合成最终的自适应滤波器 $\hat{}\boldsymbol{m}_r$。借助脉冲响应表示为 $\{\boldsymbol{f}_j\}_{j=1}^{N}$ 且长度为 L的分析滤波器，信号d_r 可分解为子带期望信号 $d_{r,j}$。此外，我们引入一个大小为 $1\times L$的期望向量 $\boldsymbol{d}_r$，其定义为

$$\hat{\boldsymbol{d}}_r\triangleq[d_r,d_{r-1},\ldots,d_{r-L+1}]. \tag{12}$$

我们将由子带期望信号$\{d_{r,j}\}_{j=1}^{N}$ 构成的集合称为子带期望向量 $\boldsymbol{d}_r$，即

$$\boldsymbol{d}_r\triangleq[d_{r,1},d_{r,2},\ldots,d_{r,N}]=\hat{\boldsymbol{d}}_r\mathbf{F}, \tag{13}$$

其中，F $\triangleq[\boldsymbol{f}_1,\boldsymbol{f}_2,\ldots,\boldsymbol{f}_N]$ 表示大小为 $L\times N$的分析滤波器矩阵。

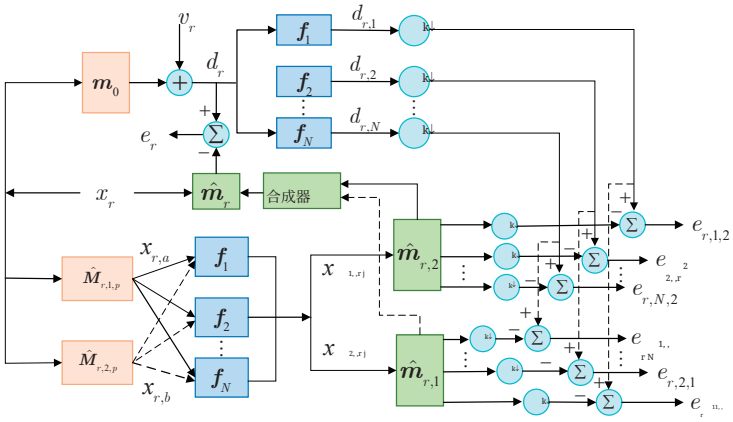


图1. NSAF-NKP-I 的结构。

采用与式（10）类似的分解方法，自适应滤波器 $\hat{}\boldsymbol{m}_r$ 可分解为

$$\hat{\boldsymbol{m}}_r=\sum_{p=1}^{P}\hat{\boldsymbol{m}}_{r,2,p}\otimes\hat{\boldsymbol{m}}_{r,1,p}, \tag{14}$$

其中，$\hat{}\boldsymbol{m}_{r,1,p}\in\mathbb{R}^{D_1\times 1}$ 和 $\hat{}\boldsymbol{m}_{r,2,p}\in\mathbb{R}^{D_2\times 1}$ 分别表示子滤波器矩阵 $\widetilde{\boldsymbol{M}}_{r,1}\triangleq[\hat{}\boldsymbol{m}_{r,1,1}、\hat{}\boldsymbol{m}_{r,1,2}、\ldots、\hat{}\boldsymbol{m}_{r,1,P}]\in\mathbb{R}^{D_1\times P}$ 和 $\widetilde{\boldsymbol{M}}_{r,2}\triangleq[\hat{}\boldsymbol{m}_{r,2,1}、\hat{}\boldsymbol{m}_{r,2,2}、\ldots、\hat{}\boldsymbol{m}_{r,2,P}]\in\mathbb{R}^{D_2\times P}$ 的第 p 列。此外，我们定义以下关系 [49]:

$$\begin{aligned}\hat{\boldsymbol{m}}_{r,2,p}\otimes\hat{\boldsymbol{m}}_{r,1,p}&\triangleq\left[\hat{\boldsymbol{m}}_{r,2,p}\otimes\boldsymbol{I}_{D_1}\right]\hat{\boldsymbol{m}}_{r,1,p}\\&\triangleq\left[\boldsymbol{I}_{D_2}\otimes\hat{\boldsymbol{m}}_{r,1,p}\right]\hat{\boldsymbol{m}}_{r,2,p},\end{aligned} \tag{15}$$

其中， $\boldsymbol{I}_{D_1}$ 和 $\boldsymbol{I}_{D_2}$ 是大小分别为 $D_1\times D_1$和 $D_2\times D_2$的单位矩阵。

将式（15）代入式（14），可得

$$\hat{\boldsymbol{m}}_r=\sum_{p=1}^{P}\hat{\boldsymbol{M}}_{r,2,p}\hat{\boldsymbol{m}}_{r,1,p}=\sum_{p=1}^{P}\hat{\boldsymbol{M}}_{r,1,p}\hat{\boldsymbol{m}}_{r,2,p}, \tag{16}$$

其中，$\hat{}\boldsymbol{M}_{r,1,p}\triangleq\left[\boldsymbol{I}_{D_2}\otimes\hat{}\boldsymbol{m}_{r,1,p}\right]$和 $\hat{}\boldsymbol{M}_{r,2,p}\triangleq\left[\hat{}\boldsymbol{m}_{r,2,p}\otimes\boldsymbol{I}_{D_1}\right]$分别是大小为 $D_1D_2\times D_2$和 $D_1D_2\times D_1$的矩形矩阵。

误差信号 e_r 的计算公式为

$$e_r = d_r - \hat{\boldsymbol{m}}_r^{\mathrm{T}} \boldsymbol{x}_r. \tag{17}$$

将 (16) 代入 (17)，可得子滤波器的误差信号为

$$\begin{aligned} e_r = e_{r,1} &= d_r - \sum_{p=1}^{P} \hat{\boldsymbol{m}}_{r,1,p}^{\mathrm{T}} \hat{\boldsymbol{M}}_{r,2,p}^{\mathrm{T}} \boldsymbol{x}_r \\ &= d_r - \sum_{p=1}^{P} \hat{\boldsymbol{m}}_{r,1,p}^{\mathrm{T}} \boldsymbol{x}_{r,2,p} = d_r - \hat{\boldsymbol{m}}_{r,1}^{\mathrm{T}} \boldsymbol{x}_{r,b}, \end{aligned} \tag{18}$$

$$\begin{aligned} e_r = e_{r,2} &= d_r - \sum_{p=1}^{P} \hat{\boldsymbol{m}}_{r,2,p}^{\mathrm{T}} \hat{\boldsymbol{M}}_{r,1,p}^{\mathrm{T}} \boldsymbol{x}_r \\ &= d_r - \sum_{p=1}^{P} \hat{\boldsymbol{m}}_{r,2,p}^{\mathrm{T}} \boldsymbol{x}_{r,1,p} = d_r - \hat{\boldsymbol{m}}_{r,2}^{\mathrm{T}} \boldsymbol{x}_{r,a}, \end{aligned} \tag{19}$$

其中

$$\boldsymbol{x}_{r,2,p} = \hat{\boldsymbol{M}}_{r,2,p}^{\mathrm{T}} \boldsymbol{x}_r, \tag{20}$$

$$\boldsymbol{x}_{r,b} = \left[\boldsymbol{x}_{r,2,1}^{\mathrm{T}}, \boldsymbol{x}_{r,2,2}^{\mathrm{T}}, ..., \boldsymbol{x}_{r,2,P}^{\mathrm{T}}\right]^{\mathrm{T}}, \tag{21}$$

$$\hat{\boldsymbol{m}}_{r,1} = [\hat{\boldsymbol{m}}_{r,1,1}^{\mathrm{T}}, \hat{\boldsymbol{m}}_{r,1,2}^{\mathrm{T}}, ..., \hat{\boldsymbol{m}}_{r,1,P}^{\mathrm{T}}]^{\mathrm{T}}, \tag{22}$$

$$\boldsymbol{x}_{r,1,p} = \hat{\boldsymbol{M}}_{r,1,p}^{\mathrm{T}} \boldsymbol{x}_r, \tag{23}$$

$$\boldsymbol{x}_{r,a} = \left[\boldsymbol{x}_{r,1,1}^{\mathrm{T}}, \boldsymbol{x}_{r,1,2}^{\mathrm{T}}, ..., \boldsymbol{x}_{r,1,P}^{\mathrm{T}}\right]^{\mathrm{T}}, \tag{24}$$

$$\hat{\boldsymbol{m}}_{r,2} = [\hat{\boldsymbol{m}}_{r,2,1}^{\mathrm{T}}, \hat{\boldsymbol{m}}_{r,2,2}^{\mathrm{T}}, ..., \hat{\boldsymbol{m}}_{r,2,P}^{\mathrm{T}}]^{\mathrm{T}}. \tag{25}$$

注1. $\hat{\boldsymbol{m}}_{r,1,p}$ 和 $\hat{\boldsymbol{m}}_{r,2,p}$ 的初始值分别设为 $\hat{\boldsymbol{m}}_{0,1,p} = [\lambda\ \boldsymbol{0}_{D_1-1}^{\mathrm{T}}]^{\mathrm{T}}$ 和 $\hat{\boldsymbol{m}}_{0,2,p} = [\lambda\ \boldsymbol{0}_{D_2-1}^{\mathrm{T}}]^{\mathrm{T}}$，其中 $0 < \lambda \leq 1$。据我们所知，所有现有的基于 NKP 分解的自适应滤波算法均采用上述初始值设置方法（即原始方法）。此外，我们提出了一种设置初始值的新方法，即

$$\hat{\boldsymbol{M}}_{0,2} = \begin{pmatrix} \lambda & 0 & \cdots & 0 \\ 0 & \lambda & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda \end{pmatrix}_{D_2 \times P}.$$

我们将这种初始值设置方法称为 YIM 方法，其有效性已在图 8(e) 中得到验证。不过，这个问题并非本文的重点。为与其他基于 NKP 的自适应滤波算法保持一致，我们仍选择使用原始方法。

接下来，定义输入矩形矩阵 $\boldsymbol{X}_{r,a}$，其大小为 $PD_2 \times L$；以及 $\boldsymbol{X}_{r,b}$，其大小为 $PD_1 \times L$：

$$\boldsymbol{X}_{r,a} \triangleq [\boldsymbol{x}_{r,a}, \boldsymbol{x}_{r-1,a}, ..., \boldsymbol{x}_{r-L+1,a}], \tag{26}$$

$$\boldsymbol{X}_{r,b} \triangleq [\boldsymbol{x}_{r,b}, \boldsymbol{x}_{r-1,b}, ..., \boldsymbol{x}_{r-L+1,b}]. \tag{27}$$

利用分析滤波器 $\{\boldsymbol{f}_j\}_{j=1}^{N}$，子带输入向量 $\boldsymbol{x}_{r,j,1} \triangleq [\boldsymbol{x}_{r,j,1,1}^{\mathrm{T}}, \boldsymbol{x}_{r,j,1,2}^{\mathrm{T}}, ..., \boldsymbol{x}_{r,j,1,P}^{\mathrm{T}}]^{\mathrm{T}}$ 以及，子滤波器 $\hat{\boldsymbol{m}}_{r,2}$，$[\boldsymbol{x}_{r,j,2,1}^{\mathrm{T}}, \boldsymbol{x}_{r,j,2,2}^{\mathrm{T}}, ..., \boldsymbol{x}_{r,j,2,P}^{\mathrm{T}}]^{\mathrm{T}}$ 和 $\hat{\boldsymbol{m}}_{r,1}$ 可以分别收集到子带输入矩形矩阵 $\boldsymbol{X}_{r,a}$ 和 $\boldsymbol{X}_{r,b}$中，即，

$$\bar{\boldsymbol{X}}_{r,a} \triangleq [\boldsymbol{x}_{r,1,1}, \boldsymbol{x}_{r,2,1}, ..., \boldsymbol{x}_{r,N,1}] = \boldsymbol{X}_{r,a}\mathbf{F}, \tag{28}$$

$$\bar{\boldsymbol{X}}_{r,b} \triangleq [\boldsymbol{x}_{r,1,2}, \boldsymbol{x}_{r,2,2}, ..., \boldsymbol{x}_{r,N,2}] = \boldsymbol{X}_{r,b}\mathbf{F}, \tag{29}$$

其中，$\bar{\boldsymbol{X}}_{r,a}$和 $\bar{\boldsymbol{X}}_{r,b}$的大小分别为 $PD_2 \times N$和 $PD_1 \times N$。

考虑克罗内克积分解方法，式（18）和（19）中误差信号的子带形式可写为

$$e_{r,j} = d_{r,j} - \hat{\boldsymbol{m}}_{r,1}^{\mathrm{T}} \boldsymbol{x}_{r,j,2} = d_{r,j} - \hat{\boldsymbol{m}}_{r,2}^{\mathrm{T}} \boldsymbol{x}_{r,j,1}. \tag{30}$$

为建立子带域中自适应滤波器 $\hat{\boldsymbol{m}}_{r,1}$ 和 $\hat{\boldsymbol{m}}_{r,2}$ 的权重递推关系，代价函数可表示为

$$J_{\hat{\boldsymbol{m}}_{r,2}}(\hat{\boldsymbol{m}}_{r,1}) = \frac{1}{2} \sum_{j=1}^{N} \frac{e_{r,j}^2}{||\boldsymbol{x}_{r,j,2}||_2^2}, \tag{31}$$

$$J_{\hat{\boldsymbol{m}}_{r,1}}(\hat{\boldsymbol{m}}_{r,2}) = \frac{1}{2} \sum_{j=1}^{N} \frac{e_{r,j}^2}{||\boldsymbol{x}_{r,j,1}||_2^2}. \tag{32}$$

利用随机梯度下降规则，子滤波器 $\hat{\boldsymbol{m}}_{r,1}$ 和 $\hat{\boldsymbol{m}}_{r,2}$ 的权重更新可写为

$$\hat{\boldsymbol{m}}_{r+k,1} = \hat{\boldsymbol{m}}_{r,1} + \mu_1 \sum_{j=1}^{N} \frac{\boldsymbol{x}_{r,j,2} e_{r,j}}{||\boldsymbol{x}_{r,j,2}||_2^2 + \delta}, \tag{33}$$

$$\hat{\boldsymbol{m}}_{r+k,2} = \hat{\boldsymbol{m}}_{r,2} + \mu_2 \sum_{j=1}^{N} \frac{\boldsymbol{x}_{r,j,1} e_{r,j}}{||\boldsymbol{x}_{r,j,1}||_2^2 + \delta}, \tag{34}$$

其中，$\mu_1 > 0$ 和 $\mu_2 > 0$ 为步长，k 为决定迭代间隔的正整数，δ 为避免除零的小正数。

到目前为止，式（14）、（28）、（29）、（30）、（33）和（34）构成了所提出的 NSAF-NKP-I 算法。

B. NSAF-NKP-II算法的推导

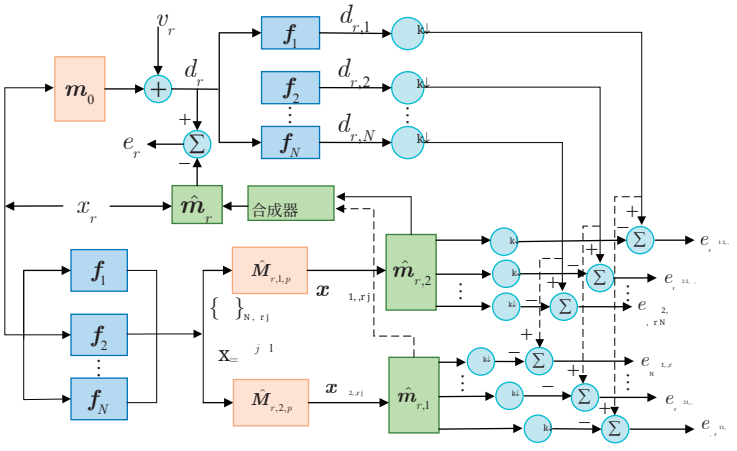


图2. NSAF-NKP-II 的结构。

借助分析滤波器，可以将输入向量 $\boldsymbol{x}_r$ 分解为子带输入向量 $\{\boldsymbol{x}_{r,j}\}_{j=1}^{N}$。然后，我们引入大小为 $D\times L$ 的输入矩形矩阵 $\boldsymbol{X}_r$，如下所示：

$$\boldsymbol{X}_r \triangleq [\boldsymbol{x}_r, \boldsymbol{x}_{r-1}, ..., \boldsymbol{x}_{r-L+1}]. \tag{35}$$

子带输入矩形矩阵 $\bar{\boldsymbol{X}}_r \in \mathbb{R}^{D\times N}$ 通过收集子带输入向量 $\{\boldsymbol{x}_{r,j}\}_{j=1}^{N}$获得，即

$$\bar{\boldsymbol{X}}_r \triangleq [\boldsymbol{x}_{r,1}, \boldsymbol{x}_{r,2}, ..., \boldsymbol{x}_{r,N}] = \boldsymbol{X}_r\mathbf{F}. \tag{36}$$

图 2描述了NSAF-NKP-II算法的结构。与NSAF-NKP-I算法的推导类似，第 p个子带输入向量 $\boldsymbol{x}_{r,\ j,\ 2,\ p}$ 和 $\boldsymbol{x}_{r,\ j,\ 1,\ p}$ 对于子滤波器 $\hat{}\boldsymbol{m}_{r,\ 1}$ 和 $\hat{}\boldsymbol{m}_{r,\ 2}$ 可通过以下方式计算

$$\boldsymbol{x}_{r,j,2,p} = \hat{\boldsymbol{M}}_{r,2,p}^{\mathrm{T}}\boldsymbol{x}_{r,j}, \tag{37}$$

$$\boldsymbol{x}_{r,j,1,p} = \hat{\boldsymbol{M}}_{r,1,p}^{\mathrm{T}}\boldsymbol{x}_{r,j}. \tag{38}$$

公式 (14)、(30)、(33)、(34)、(37) 和 (38)构成所提出的NSAF-NKP-II算法。

注 2. 必须强调的是，NSAF-NKP-I 算法和NSAF-NKP-II算法的区别仅在于其模型架构。分析和仿真表明，这种结构差异不会影响算法性能；然而，如第 IV 节的图 4和表 II所示，NSAF-NKP-II算法的计算复杂度显著低于NSAF-NKP-I 算法。

C. 性能分析

为建立 NKP-NSAF-II 算法的稳态均方偏差（MSD）模型，我们将滤波器的子带误差信号 $\hat{}\boldsymbol{m}_r$ 定义为

$$e_{r,j} = \boldsymbol{x}_{r,j}^{\mathrm{T}}\widetilde{\boldsymbol{m}}_r + v_{r,j} = \zeta_{r,j} + v_{r,j}, \tag{39}$$

其中，$\widetilde{\boldsymbol{m}}_r \triangleq \boldsymbol{m}_0 - \hat{}\boldsymbol{m}_r$ 表示权重误差向量，$v_{r,\ j}$ 是子带噪声向量，而 $\zeta_{r,\ j}$ 表示超额误差。与式（10）类似，长未知系统 $\boldsymbol{m}_0$ 可表示为

$$\boldsymbol{m}_0 = \sum_{p=1}^{P}\boldsymbol{m}_{2,p}^{o}\otimes\boldsymbol{m}_{1,p}^{o}. \tag{40}$$

接下来，我们将分解后的权重误差向量定义为

$$\widetilde{\boldsymbol{m}}_{r,2,p} = \boldsymbol{m}_{2,p}^{o} - \hat{\boldsymbol{m}}_{r,2,p}, \tag{41}$$

$$\widetilde{\boldsymbol{m}}_{r,1,p} = \boldsymbol{m}_{1,p}^{o} - \hat{\boldsymbol{m}}_{r,1,p}. \tag{42}$$

根据（14）和（40），超额误差信号可近似表示为

$$\begin{aligned}\zeta_{r,j} &= \boldsymbol{x}_{r,j}^{\mathrm{T}}\Big[\sum_{p=1}^{P}\boldsymbol{m}_{2,p}^{o}\otimes\boldsymbol{m}_{1,p}^{o} - \sum_{p=1}^{P}\hat{\boldsymbol{m}}_{r,2,p}\otimes\hat{\boldsymbol{m}}_{r,1,p}\Big]\\ &\approx \boldsymbol{x}_{r,j}^{\mathrm{T}}\Big[\sum_{p=1}^{P}\widetilde{\boldsymbol{m}}_{r,2,p}\otimes\hat{\boldsymbol{m}}_{r,1,p} + \sum_{p=1}^{P}\hat{\boldsymbol{m}}_{r,2,p}\otimes\widetilde{\boldsymbol{m}}_{r,1,p}\Big]\\ &= \zeta_{r,j,1} + \zeta_{r,j,2}.\end{aligned} \tag{43}$$

在（43）中，我们通过省略项 $\boldsymbol{x}_{r,j}^{\mathrm{T}}\sum_{p=1}^{P}\widetilde{\boldsymbol{m}}_{r,2,p}\otimes\widetilde{\boldsymbol{m}}_{r,1,p}$ 进行近似，因为该项包含权重误差向量的高阶项，而这些高阶项在稳态阶段会变得无关紧要。此外，我们汇集了 $\widetilde{\boldsymbol{m}}_{r,2} \triangleq [\widetilde{\boldsymbol{m}}_{r,2,1}^{\mathrm{T}}, ..., \widetilde{\boldsymbol{m}}_{r,2,P}^{\mathrm{T}}]^{\mathrm{T}}$ 和 $\widetilde{\boldsymbol{m}}_{r,1} \triangleq [\widetilde{\boldsymbol{m}}_{r,1,1}^{\mathrm{T}}, ..., \widetilde{\boldsymbol{m}}_{r,1,P}^{\mathrm{T}}]^{\mathrm{T}}$。据此，项 $\zeta_{r,j,1}$ 和 $\zeta_{r,j,2}$ 可重新表示为

$$\zeta_{r,j,1} = \sum_{p=1}^{P}\boldsymbol{x}_{r,j}^{\mathrm{T}}\Big[\widetilde{\boldsymbol{m}}_{r,2,p}\otimes\hat{\boldsymbol{m}}_{r,1,p}\Big] = \boldsymbol{x}_{r,j,1}^{\mathrm{T}}\widetilde{\boldsymbol{m}}_{r,2}, \tag{44}$$

$$\zeta_{r,j,2} = \sum_{p=1}^{P}\boldsymbol{x}_{r,j}^{\mathrm{T}}\Big[\hat{\boldsymbol{m}}_{r,2,p}\otimes\widetilde{\boldsymbol{m}}_{r,1,p}\Big] = \boldsymbol{x}_{r,j,2}^{\mathrm{T}}\widetilde{\boldsymbol{m}}_{r,1}. \tag{45}$$

为便于后续推导，作出如下假设：

假设 1：权重误差向量 $\widetilde{\boldsymbol{m}}_{r,\ 1}$ 与 $\widetilde{\boldsymbol{m}}_{r,\ 2}$ 相互独立，且均与输入信号不相关。

假设 2：子滤波器的子带误差信号 $e_{r,\ j,\ 1}$ 和 $e_{r,\ j,\ 2}$ 分别与输入向量 $\boldsymbol{x}_{r,\ j,\ 2}$ 和 $\boldsymbol{x}_{r,\ j,\ 1}$渐近不相关。零均值加性噪声 v_r 与其他信号不相关。该假设常用于自适应滤波算法的性能分析。

假设 3：当输入信号为零均值白噪声时，超额误差信号 $\zeta_{r,\ j,\ 1}$ 和 $\zeta_{r,\ j,\ 2}$ 均值为零且相互独立，该假设用于消除交叉项 $\mathrm{E}[\zeta_{r,j,1}\zeta_{r,j,2}]$。

根据 (41) 和 (42)，我们进一步得到

$$\widetilde{\boldsymbol{m}}_{r+k,1} = \widetilde{\boldsymbol{m}}_{r,1} - \mu_1\sum_{j=1}^{N}\frac{\boldsymbol{x}_{r,j,2}e_{r,j}}{||\boldsymbol{x}_{r,j,2}||_2^2}, \tag{46}$$

$$\widetilde{\boldsymbol{m}}_{r+k,2} = \widetilde{\boldsymbol{m}}_{r,2} - \mu_2\sum_{j=1}^{N}\frac{\boldsymbol{x}_{r,j,1}e_{r,j}}{||\boldsymbol{x}_{r,j,1}||_2^2}. \tag{47}$$

然后，将 (46) 和 (47) 两边分别左乘其各自的转置并取期望，得到

$$\begin{aligned}&\mathrm{E}\big\{||\widetilde{\boldsymbol{m}}_{r+k,1}||_2^2\big\} = \mathrm{E}\big\{||\widetilde{\boldsymbol{m}}_{r,1}||_2^2\big\}\\ &- 2\mu_1\sum_{j=1}^{N}\mathrm{E}\Big\{\frac{e_{r,j}\zeta_{r,j,2}}{||\boldsymbol{x}_{r,j,2}||_2^2}\Big\} + \mu_1^2\sum_{j=1}^{N}\mathrm{E}\Big\{\frac{e_{r,j}^2}{||\boldsymbol{x}_{r,j,2}||_2^2}\Big\},\end{aligned} \tag{48}$$

$$\begin{aligned}&\mathrm{E}\big\{||\widetilde{\boldsymbol{m}}_{r+k,2}||_2^2\big\} = \mathrm{E}\big\{||\widetilde{\boldsymbol{m}}_{r,2}||_2^2\big\}\\ &- 2\mu_2\sum_{j=1}^{N}\mathrm{E}\Big\{\frac{e_{r,j}\zeta_{r,j,1}}{||\boldsymbol{x}_{r,j,1}||_2^2}\Big\} + \mu_2^2\sum_{j=1}^{N}\mathrm{E}\Big\{\frac{e_{r,j}^2}{||\boldsymbol{x}_{r,j,1}||_2^2}\Big\}.\end{aligned} \tag{49}$$

由于子带信号互不相关，并且从一次迭代到下一次迭代，$||\boldsymbol{x}_{r,j,1}||^2$ 和 $||\boldsymbol{x}_{r,j,2}||^2$ 的波动可以假设足够小 [50]。当子滤波器收敛到稳态阶段，即 $r\to\infty$ 时，能量守恒关系 (48) 和 (49) 可近似为

$$2\mathrm{E}\big\{e_{r,j}\zeta_{r,j,2}\big\} = \mu_1\mathrm{E}\big\{e_{r,j}^2\big\}, \tag{50}$$

$$2\mathrm{E}\big\{e_{r,j}\zeta_{r,j,1}\big\} = \mu_2\mathrm{E}\big\{e_{r,j}^2\big\}. \tag{51}$$

基于假设 2 和 3，将 (39) 和 (43) 代入 (48) 和 (49)，可得

$$(2-\mu_1)\mathrm{E}\{\zeta_{r,j,2}^2\}=\mu_1\big(\mathrm{E}\{\zeta_{r,j,1}^2\}+\mathrm{E}\{v_{r,j}^2\}\big),\tag{52}$$

$$(2-\mu_2)\mathrm{E}\{\zeta_{r,j,1}^2\}=\mu_2\big(\mathrm{E}\{\zeta_{r,j,2}^2\}+\mathrm{E}\{v_{r,j}^2\}\big).\tag{53}$$

结合式（52）和式（53），可得

$$\mathrm{E}\{\zeta_{r,j,1}^2\}=\frac{\mu_2\mathrm{E}\{v_{r,j}^2\}}{2-\mu_1-\mu_2},\tag{54}$$

$$\mathrm{E}\{\zeta_{r,j,2}^2\}=\frac{\mu_1\mathrm{E}\{v_{r,j}^2\}}{2-\mu_1-\mu_2}.\tag{55}$$

由于 $\sigma_{v_r}^2=\frac{1}{N}\sum_{j=1}^{N}\sigma_{v_{r,\ j}}^2$，NSAF-NKP-II算法的稳态超额均方误差（EMSE）可定义为

$$\begin{aligned}\mathrm{EMSE}&\triangleq\frac{1}{N}\sum_{j=1}^{N}\mathrm{E}\{\zeta_{r,j}^2\}=\frac{1}{N}\sum_{j=1}^{N}\Big[\mathrm{E}\{\zeta_{r,j,1}^2\}+\mathrm{E}\{\zeta_{r,j,2}^2\}\Big]\\&=\frac{(\mu_1+\mu_2)\mathrm{E}\{v_r^2\}}{2-\mu_1-\mu_2}.\end{aligned}\tag{56}$$

因此，为确保NSAF-NKP-II算法的稳定性，式（54）和式（55）中的分母应大于零，即

$$0<\mu_1+\mu_2<2.\tag{57}$$

若设置相同的步长，即 $\mu_1=\mu_2$，则可得

$$0<\mu<1,\tag{58}$$

其验证结果见图6（b）。从结果可见，该步长范围小于NSAF算法的步长范围，即 $0<\mu<2$ [13]。

IV. NSAF-NKP-II的稳健变体

为增强 NSAF-NKP-II 算法对脉冲噪声的鲁棒性，我们分别引入缩放函数$\mathcal{X}_{r,\ j,\ 1}$和 $\mathcal{X}_{r,\ j,\ 2}$，针对子带 j 将（33）和（34）推广为

$$\hat{\boldsymbol{m}}_{r+k,1}=\hat{\boldsymbol{m}}_{r,1}+\mu_1\sum_{j=1}^{N}\mathcal{X}_{r,j,1}\frac{\boldsymbol{x}_{r,j,2}e_{r,j,1}}{||\boldsymbol{x}_{r,j,2}||_2^2+\delta},\tag{59}$$

$$\hat{\boldsymbol{m}}_{r+k,2}=\hat{\boldsymbol{m}}_{r,2}+\mu_2\sum_{j=1}^{N}\mathcal{X}_{r,j,2}\frac{\boldsymbol{x}_{r,j,1}e_{r,j,2}}{||\boldsymbol{x}_{r,j,1}||_2^2+\delta}.\tag{60}$$

注3. 在脉冲噪声下，要使(59)和(60)实现稳定收敛，其条件是脉冲噪声出现时，$\mathcal{X}_{r、\ j、\ 1}$以及 $\mathcal{X}_{r、\ j、\ 2}$ 的值较小。为达到这一效果，我们分别通过最大化基于相关熵的代价函数和最小化基于对数的代价函数，设计了 RNSAF-NKP-MCC 和 RNSAF-NKP-LC 算法。

A. RNSAF-NKP-MCC 算法的推导

在信息论学习中，相关熵是衡量两个任意随机变量 $\mathcal{R}$ 和 $\mathcal{Y}$ [33]局部相似性的有效工具，其定义为

$$V(\mathcal{R},\mathcal{Y})=\mathrm{E}\big\{\kappa(\mathcal{R},\mathcal{Y})\big\},\tag{61}$$

其中，κ(.) 表示平移不变梅瑟核。由于具有平滑、对称等诸多优良性质，高斯函数常被用作相关熵的核函数。因此，我们有

$$\kappa(\mathcal{R},\mathcal{Y})=\frac{1}{\sqrt{2\pi}\sigma}\exp\Big(-\frac{(\mathcal{R}-\mathcal{Y})^2}{2\sigma^2}\Big),\tag{62}$$

其中，$\sigma>0$ 表示核带宽。

然后，我们基于 NSAF-NKP-II 框架建立如下鲁棒代价函数:

$$J_{\hat{\boldsymbol{m}}_{r,1,\mathrm{MCC}}}=\frac{1}{2\psi}\sum_{j=1}^{N}\exp\Big(-\psi\frac{\Big[d_{r,j}-\hat{\boldsymbol{m}}_{r,1}^{\mathrm{T}}\boldsymbol{x}_{r,j,2}\Big]^2}{||\boldsymbol{x}_{r,j,2}||_2^2}\Big),\tag{63}$$

$$J_{\hat{\boldsymbol{m}}_{r,2,\mathrm{MCC}}}=\frac{1}{2\psi}\sum_{j=1}^{N}\exp\Big(-\psi\frac{\Big[d_{r,j}-\hat{\boldsymbol{m}}_{r,2}^{\mathrm{T}}\boldsymbol{x}_{r,j,1}\Big]^2}{||\boldsymbol{x}_{r,j,1}||_2^2}\Big),\tag{64}$$

其中，$\psi=\frac{1}{2\sigma^2}$ 表示核参数。

结合梯度上升法，分别对 (63) 和 (64) 两边关于 $\hat{}\boldsymbol{m}_{r,\ 1}$ 和 $\hat{}\boldsymbol{m}_{r,2}$ 求偏导，并将结果设为零，即可得到所提出的 RNSAF-NKP-MCC 算法:

$$\begin{aligned}\hat{\boldsymbol{m}}_{r+k,1}&=\hat{\boldsymbol{m}}_{r,1}\\&+\mu_1\sum_{j=1}^{N}\frac{\boldsymbol{x}_{r,j,2}e_{r,j,1}}{||\boldsymbol{x}_{r,j,2}||_2^2+\delta}\times\underbrace{\exp\Big(-\psi\frac{e_{r,j,1}^2}{||\boldsymbol{x}_{r,j,2}||_2^2}\Big)}_{(a)},\end{aligned}\tag{65}$$

$$\begin{aligned}\hat{\boldsymbol{m}}_{r+k,2}&=\hat{\boldsymbol{m}}_{r,2}\\&+\mu_2\sum_{j=1}^{N}\frac{\boldsymbol{x}_{r,j,1}e_{r,j,2}}{||\boldsymbol{x}_{r,j,1}||_2^2+\delta}\times\underbrace{\exp\Big(-\psi\frac{e_{r,j,2}^2}{||\boldsymbol{x}_{r,j,1}||_2^2}\Big)}_{(b)}.\end{aligned}\tag{66}$$

显然，发生脉冲噪声时，误差信号 $\{e_{r,j,1},e_{r,j,2}\}$ 的幅值会显著增大。然而，为确保选择较小的步长，(65) 和 (66) 中的项 (a) 和 (b) 会变得非常小；事实证明，这能够有效避免脉冲噪声中的离群值对滤波器权重 [51]更新造成的不利影响。

B. RNSAF-NKP-LC 算法的推导

受 [52]，启发，我们将基于对数的代价函数扩展到 NSAF-NKP-II 结构，以建立

$$J_{\hat{\boldsymbol{m}}_{r,1,\mathrm{LC}}}=\frac{1}{2\beta}\sum_{j=1}^{N}\ln\Big(1+\beta\frac{\Big[d_{r,j}-\hat{\boldsymbol{m}}_{r,1}^{\mathrm{T}}\boldsymbol{x}_{r,j,2}\Big]^2}{||\boldsymbol{x}_{r,j,2}||_2^2}\Big),\tag{67}$$

$$J_{\hat{\boldsymbol{m}}_{r,2,\mathrm{LC}}}=\frac{1}{2\beta}\sum_{j=1}^{N}\ln\Big(1+\beta\frac{\Big[d_{r,j}-\hat{\boldsymbol{m}}_{r,2}^{\mathrm{T}}\boldsymbol{x}_{r,j,1}\Big]^2}{||\boldsymbol{x}_{r,j,1}||_2^2}\Big),\tag{68}$$

表 I 每 k 个输入样本中不同算法的计算复杂度。

算法	×/÷
NLMS	$k(3D+2)$
NSAF	$3ND+2N+(D+1)\ LN$
NLMS-NKP	$k(5PD+3PD_1+3PD_2+4)$
RLS-NKP	$k(5PD+5P^2D_1^2+5P^2D_2^2+3PD_1+3PD_2)$
APA-NKP	$k[5PD+(2C+1+C^2)(PD_1+PD_2)\ +2C^2]$
NSAF-NKP-II	$PD+4PND+3NPD_2+3NPD_1+4N+\ (D+1)\ LN$
NSAF-NKP-I	$PD+4DPL+(PD_1+1)LN+(PD_2+1)LN+3PND_1+3PND_2+4N$
RNSAF-NKP-MCC	$PD+4PND+3NPD_2+3NPD_1+4N+(D+1)LN+NP(D_1+D_2)\ +8N$
RNSAF-NKP-LC	$PD+4PND+3NPD_2+3NPD_1+4N+(D+1)LN+NP(D_1+D_2)\ +8N$

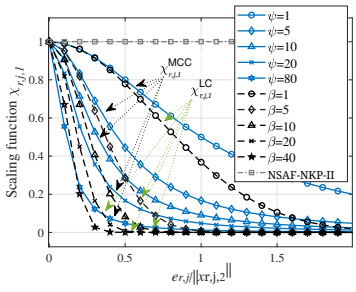


图3. 第 i个子带的尺度函数。

其中， $\beta>0$ 是一个可调参数。

分别对 $J_{\hat{\boldsymbol{m}}_{r,1,\mathrm{LC}}}$ 和 $J_{\hat{\boldsymbol{m}}_{r,2,\mathrm{LC}}}$ 关于 $\hat{}\boldsymbol{m}_{r,1}$ 和 $\hat{}\boldsymbol{m}_{r,2}$求梯度后，建立 RNSAF-NKP-LC 算法如下

$$\hat{\boldsymbol{m}}_{r+k,1}=\hat{\boldsymbol{m}}_{r,1}+\sum_{j=1}^{N}\frac{\mu_1\frac{\boldsymbol{x}_{r,j,2}e_{r,j,1}}{||\boldsymbol{x}_{r,j,2}||_2^2+\delta}}{1+\beta\left(e_{r,j,1}/||\boldsymbol{x}_{r,j,2}||_2\right)^2},\tag{69}$$

$$\hat{\boldsymbol{m}}_{r+k,2}=\hat{\boldsymbol{m}}_{r,2}+\sum_{j=1}^{N}\frac{\mu_2\frac{\boldsymbol{x}_{r,j,1}e_{r,j,2}}{||\boldsymbol{x}_{r,j,1}||_2^2+\delta}}{1+\beta\left(e_{r,j,2}/||\boldsymbol{x}_{r,j,1}||_2\right)^2}.\tag{70}$$

备注4。图3展示了缩放函数$\chi_r^{\mathrm{MCC}}{}_j$,1 和 $\chi_r^{\mathrm{LC}}{}_j$,1 在采用不同的 ψ 和 β时，相对于项 $e_{rj1}/||\boldsymbol{x}_{rj2}||$的形状。显然，发生脉冲干扰时，$e_{rj1}/||\boldsymbol{x}_{rj2}||$ 会呈现较大的幅值。随后，缩放函数 $\chi_r^{\mathrm{MCC}}{}_j$,1 和 $\chi_r^{\mathrm{LC}}{}_j$,1 会将步长限制在较小值，以确保离群值不会对 RNSAF-NKP-MCC 和 RNSAF-NKP-LC 算法的权重更新过程产生不利影响。值得注意的是，较小的步长可以减轻离群值对滤波器权重 [53] 更新造成的负面影响。当然，如果没有发生脉冲干扰，$e_{rj1}/||\boldsymbol{x}_{rj2}||$ 的值非常小，因此，由于 $\chi_r^{\mathrm{MCC}}{}_j$,1 和 $\chi_r^{\mathrm{LC}}{}_j$,1 趋近于 1，RNSAF-NKP-MCC 和 RNSAF-NKP-LC 算法的收敛行为与 NSAF-NKP-II 算法大致相似。例如，如图 9 所示，在高斯噪声场景下，与 NSAF-NKP-II 算法相比，引入相应鲁棒性准则并未损害 RNSAF-NKP-MCC 和 RNSAF-NKP-LC 算法在收敛速度和稳态NMSD方面的性能。

C. 算法复杂度

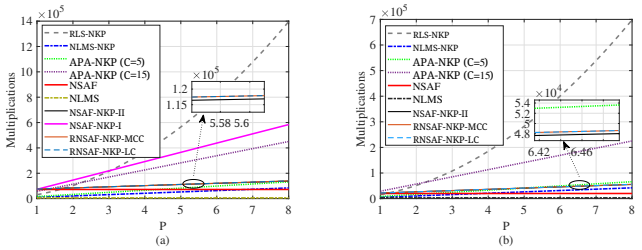


图 4. NLMS、NSAF、NLMS-NKP、RLS-NKP、APA-NKP、NSAF-NKP-I、NSAF-NKP-II、RNSAF-NKP-MCC 和 RNSAF-NKP-LC 算法每 k 个输入样本的总乘法次数。(a) 参数设置为 [$L=33$, $N=4$, D =500, $D_1=25$, $D_2=20$和 $k=4$] 时；(b) 参数设置为 [$L=17$,$N=2$和 $k=2$] 时；算法的其他参数与图3(a)中的相同。

表 I 总结了 NLMS [8], NSAF [13], NLMS-NKP [23],RLS-NKP [26],基于 NKP 的仿射投影算法（APA-NKP） [25],NSAF-NKP-I、NSAF-NKP- II、RNSAF-NKP-MCC 和 RNSAF-NKP-LC 算法在系统辨识中的计算复杂度，统计了每 k 个输入样本所需的乘法运算次数。请注意，NSAF 结构在划分扩展输入向量 $\boldsymbol{x}_r$ 和期望输出信号 d_r时具有固有的计算复杂度，需要 $LN(D+1)$ 次乘法运算。投影阶数 C 决定了 APA-NKP 算法的收敛性能和稳态性能；当 $C=1$ 时，APA-NKP 算法退化为 NLMS-NKP 算法。通常，较高的投影阶数可带来更好的性能，但也会增加计算复杂度。图 4(a) 绘出了在 $N=4$ 条件下各算法每 k 个输入样本所需的总乘法次数。与 NLMS 算法相比，NLMS-NKP 算法需要更多的乘法运算。正如我们将在下文仿真中看到的，所提出的 NSAF-NKP-I 算法比 NLMS-NKP 算法收敛速度更快，但乘法运算负担也更重。所提出的 NSAF-NKP-II 算法在收敛性能上与 NSAF-NKP-I 相当，同时显著降低了计算复杂度。请注意，在大多数情况下，RLS-NKP 和 APA-NKP ($C=15$) 算法的计算复杂度远高于所提出的算法。此外，所提出的 RNSAF-NKP-MCC 和 RNSAF-NKP-LC 算法仅以相对于 NSAF-

表 II 不同算法的时间消耗比较。

算法	归一化								
运行时间	0.1872秒	0.3121秒	5.04s	6.8008s	18.2783秒	51.4564秒	90.2347秒	5.8983秒	6.3114秒	6.3824秒

NKP-II 算法而言极小的计算开销增幅为代价，实现了对脉冲噪声的鲁棒性。图 4(b) 展示了在 $N = 2$ 条件下各算法每 k 个输入样本所需的总乘法次数。我们也可以得出与图 4(a) 中类似的一些结论。

此外，我们在表 II 中进行了严格的数值仿真，以测量实际执行时间。所有实验均在 MATLAB 中完成，所用笔记本电脑配备 AMD Ryzen 9 4000HS with Radeon Graphics 3.0 GHz 处理器和 16 GB 内存。输入信号包含 30000 个样本。表 II 中的运行时间是将 50 次蒙特卡罗实验所需的时间求和后取平均值得到的。显然，NSAF-NKP-II算法的运行时间略长于 NLMS-NKP 算法，因为前者需要使用分析滤波器将输入信号和期望信号分解为子带信号。NSAF-NKP-II 算法的运行时间略短于 RLS-NKP算法。同时，其运行时间仅为 APA-NKP（$C = 5$）算法的 32.2%，也仅为 NSAF-NKP-I 算法的 6.5%，这是因为 APA-NKP 算法涉及矩阵乘法和求逆。此外，所提出的 RNSAF-NKP-LC 和 RNSAF-NKP-MCC 算法的运行时间分别比 NSAF-NKP-II 算法长约 0.4131 秒和 0.4841 秒，从而实现了对脉冲噪声的鲁棒性。

V. 基于 NKP 的非线性子带算法

非线性干扰在实际应用场景中普遍存在。为提高NSAF-NKP-II算法对这类干扰的鲁棒性，我们提出了NSAF-NKP-II算法的两种鲁棒变体：TFLN-NKP-NSAF 和 Volterra-NKP-NSAF 算法。这些鲁棒算法利用函数展开模块（FEBs），通过非线性变换将输入向量 $\boldsymbol{x}_r = [x_r, x_{r-1}, ..., x_{r-B+1}]^{\mathrm{T}}$ 投影到更高维的特征空间中。图5描述了非线性子带 NKP 算法的实现过程。关键在于，TFLN-NKP-NSAF 与 Volterra-NKP-NSAF 的区别仅在于二者采用了不同的非线性投影机制，从而具备不同的非线性建模能力。例如，FEB 模块利用三角函数展开进行设计，通过该方法得到的高维输入向量 $\boldsymbol{g}_r$ 可数学表示为:

$$\begin{aligned}\boldsymbol{g}_r = \Big[&\boldsymbol{x}_r^{\mathrm{T}}, \sin(\pi x_r), \cos(\pi x_r), ..., \\ &\cos(A\pi x_r), \sin(\pi x_{r-1}), \cos(\pi x_{r-1}) \\ &, ..., \cos(A\pi x_{r-1}), ..., \cos(A\pi x_{r-B+1})\Big]^{\mathrm{T}},\end{aligned} \tag{71}$$

其中，A 表示展开阶数，$\boldsymbol{g}_r$的长度为 $D = (2A + 1)B$。类似地，FEB 模块的设计采用了 Volterra 级数展开，因此我们得到

$$\begin{aligned}\boldsymbol{g}_r = \big[&\boldsymbol{x}_r^{\mathrm{T}}, x_r^2, x_{r-1}^2, ..., x_{r-B+1}^2, \\ &x_r x_{r-1}, ..., x_r x_{r-B+1}, x_{r-1} x_{r-2}, ..., \\ &x_{r-1} x_{r-B+1}, ..., x_{r-B+2} x_{r-B+1}\big]^{\mathrm{T}},\end{aligned} \tag{72}$$

长度为 $D = (B^2 + 3B)/2$。

为使论文的表述更加简洁，这里直接将 NSAF-NKP-II 学习规则扩展到非线性子带 NKP 场景中，TFLN-NKP-NSAF 和 Volterra-NKP-NSAF 算法的实现见表 IV。

注 5. 在非线性系统辨识中，所提出的 TFLN-NKP-NSAF 和 Volterra-NKP-NSAF 算法的计算复杂度与线性 NSAF-NKP-II 算法具有相同的形式，即 $PD + 4PND + 3NP(D_1 + D_2) + 4N + (D+1)LN$。唯一的区别在于输入向量 $\boldsymbol{x}_r$ 的长度，其长度已从D 变为 B。因此，对于 TFLN-NKP-NSAF 算法，D 的长度为 $(2A+1)B$。对于 Volterra-NKP-NSAF 算法，D 的长度为 $(B^2 + 3B)/2$。

注释 6. 为了突出所提出的子带 NKP 算法与现有 NKP 算法之间的差异，我们给出了表 III，总结了相关方法和所提出方法的主要特征。显然，在相关输入条件下，基于子带结构的所提算法比 NLMS-NKP、NKP-GMCC 和 NKP-GHSAF 算法收敛速度更快。尽管 RLS-NKP 和 APA-NKP 算法的收敛性能与所提出的 NSAF-NKP-II 算法相近，但这两种算法都涉及矩阵求逆运算，因此计算成本更高。在非高斯场景中，由于 RNSAF-NKP-LC 和 RNSAF-NKP-MCC 算法继承了子带结构的固有特性，其性能优于 NKP-GMCC 和 NKP-GHSAF 算法。在非线性系统辨识场景中，所提出的 Volterra-NKP-NSAF 和 TFLN-NKP-NSAF 算法也比 TFLN-NKP-NLMS 和 Volterra-NKP-NLMS 算法收敛更快。总体而言，在系统辨识、回声消除、非线性系统和脉冲噪声环境等应用中，所提出的基于子带的 NKP 算法性能优于其他竞争算法。

VI. 仿真

本节通过全面的仿真验证所提算法在稀疏系统辨识中的性能。基于余弦调制原理，采用原型滤波器 [13]设计了子带数为 $N = 4$ 、长度为 $L = 33$ 的分析滤波器 $\{\boldsymbol{f}_i\}_{i=1}^{N}$。图5描述了

表 III 所提出的方法与现有方法的比较。

算法	域	相关输入	矩阵运算逆运算	噪声鲁棒性	非线性	参考
NLMS-NKP	全频带	中等	×	仅高斯	×	[23]
RLS-NKP	全频带	强	✓	仅高斯	×	[26]
APA-NKP	全频带	强	✓	仅高斯	×	[25]
NKP-GMCC	全频带	中等	×	非高斯	×	[35]
NKP-GHSAF	全频带	中等	×	非高斯	×	[35]
TFLN-NKP-NLMS	全频带	中等	×	仅高斯	✓	[19]
Volterra-NKP-NLMS	全频带	中等	×	仅高斯	✓	[19]
NSAF-NKP-I	子带	强	×	仅高斯	×	所提出的
NSAF-NKP-II	子带	强	×	仅高斯	×	所提出的
RNSAF-NKP-LC	子带	强	×	非高斯	×	所提出的
RNSAF-NKP-MCC	子带	强	×	非高斯	×	所提出的
Volterra-NKP-NSAF	子带	强	×	仅高斯	✓	所提出的
TFLN-NKP-NSAF	子带	强	×	仅高斯	✓	所提出的

表 IV RNSAF-NKP-LC 和 RNSAF-NKP-MCC 算法伪代码

初始化：$\hat{\boldsymbol{m}}_0 = [\lambda\ 0\ \dots\ 0]^{\mathrm{T}}$，$p = 1, 2, \dots, P$

$\hat{\boldsymbol{m}}_{0,2,p} = [\lambda\ 0\ \dots\ 0]^{\mathrm{T}}$，$p = 1, 2, \dots, P$

对于每个时刻 r，执行：

/ * 三角函数展开 * /

$\boldsymbol{g}_r = [\boldsymbol{x}_r^{\mathrm{T}}$, 正弦($\pi x_r$)、余弦($\pi x_r$).....余弦($A\pi x_r$)、正弦($\pi x_{r-1}$)、余弦($\pi x_{r-1}$).....余弦($A\pi x_{r-1}$).....,
余弦($A\pi x_{r-B+1}$)$]^{\mathrm{T}}$

/ * Volterra级数展开 * /

$\boldsymbol{g}_r = [\boldsymbol{x}_r^{\mathrm{T}}, x_r^2, x_{r-1}^2, \dots, x_{r-B+1}^2,\ x_r x_{r-1},\ \dots\dots,$
$x_r x_{r-B+1},\ x_{r-1}x_{r-2},\ \dots\dots,\ x_{r-1}x_{r-B+1},\ \dots$
$\dots,\ x_{r-B+2}x_{r-B+2}]^{\mathrm{T}}$ $\hat{\boldsymbol{X}}_r = [\boldsymbol{g}_r, \boldsymbol{g}_{r-1}, \dots, \boldsymbol{g}_{r-L+1}]$

$\hat{\boldsymbol{d}}_r = [d_r, d_{r-1}, \dots, d_{r-L+1}]$

若 $\mathrm{mod}(r, k) == 0$

$\boldsymbol{X}_r = [\boldsymbol{x}_{r,1}, \boldsymbol{x}_{r,2}, \dots, \boldsymbol{x}_{r,N}] = \hat{\boldsymbol{X}}_r\mathrm{F}$

$\boldsymbol{d}_r = [d_{r,1}, d_{r,2}, \dots, d_{r,N}] = \hat{\boldsymbol{d}}_r\mathrm{F}$

对于每个子带 j

$\boldsymbol{x}_{r,j,2} = [\boldsymbol{x}_{r,j,2,1}^{\mathrm{T}}, \boldsymbol{x}_{r,j,2,2}^{\mathrm{T}}, \dots, \boldsymbol{x}_{r,j,2,P}^{\mathrm{T}}]^{\mathrm{T}}$

$e_{r,j,1} = d_{r,j} - \hat{\boldsymbol{m}}_{r,1}^{\mathrm{T}}\boldsymbol{x}_{r,j,2}$

end

$\hat{\boldsymbol{m}}_{r+k,1} = \hat{\boldsymbol{m}}_{r,1} + \mu_1 \sum_{j=1}^{N} \frac{\boldsymbol{x}_{r,j,2} e_{r,j,1}}{\|\boldsymbol{x}_{r,j,2}\|_2^2 + \delta}$

对于每个子带 j

$\boldsymbol{x}_{r,j,1,p} = \hat{\boldsymbol{M}}_{r,1,p}^{\mathrm{T}}\boldsymbol{x}_{r,j}$

$\boldsymbol{x}_{r,j,1} = [\boldsymbol{x}_{r,j,1,1}^{\mathrm{T}}, \boldsymbol{x}_{r,j,1,2}^{\mathrm{T}}, \dots, \boldsymbol{x}_{r,j,1,P}^{\mathrm{T}}]^{\mathrm{T}}$

$e_{r,j,2} = d_{r,j} - \hat{\boldsymbol{m}}_{r,2}^{\mathrm{T}}\boldsymbol{x}_{r,j,1}$

end

$\hat{\boldsymbol{m}}_{r+k,2} = \hat{\boldsymbol{m}}_{r,2} + \mu_2 \sum_{j=1}^{N} \frac{\boldsymbol{x}_{r,j,1} e_{r,j,2}}{\|\boldsymbol{x}_{r,j,1}\|_2^2 + \delta} \sum_{p=1}^{P} \hat{\boldsymbol{m}}_{r,2,p} \otimes \hat{\boldsymbol{m}}_{r,1,p}$

else

$\hat{\boldsymbol{m}}_{r+k} = \hat{\boldsymbol{m}}_r$

end

end

两个长度为$D = 500$的稀疏系统 $\boldsymbol{m}_0$ 的脉冲响应。在仿真实验中，我们分别将图6(a)和(b)中的脉冲响应称为未知系统1（US-1）和未知系统2（US-2）。表V列出了所提算法在不同应用场景下的主要参数设置。除非另有说明，仿真结果均为50次蒙特卡洛试验的平均值。

A. 系统辨识

在系统辨识场景中，输入信号 x_r 由零均值高斯噪声 w_r 通过一个一阶

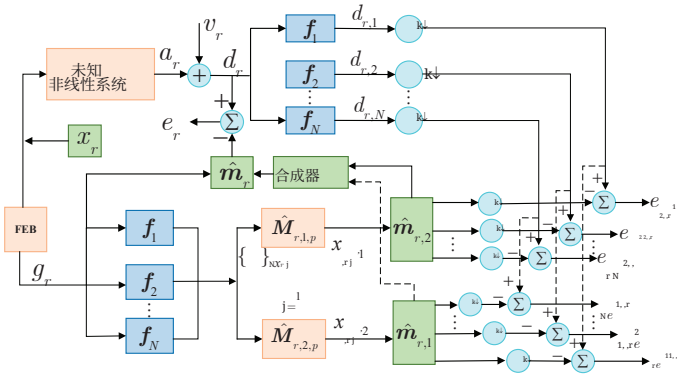


图5。非线性子带NKP算法的结构。

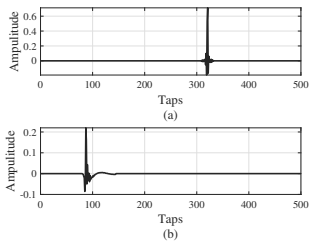


图6。未知系统的脉冲响应 $\boldsymbol{m}_0$，其 $D = 500$。（a）声学回波路径[54] 和（b）网络回波路径选自G.168建议 [55]。

自回归（AR(1)）模型生成，该模型的极点为 0.9。除非另有说明，加性噪声 v_r 为方差为 $\sigma_v^2 = 0.01$的白高斯噪声。我们采用归一化均方偏差（NMSD）、均方误差（MSE）和超额均方误差（EMSE）来评估这些算法的性能，其定义如下：

$$\mathrm{NMSD(dB)} \triangleq 20\log_{10}\mathrm{E}\left\{\frac{\|\boldsymbol{m}_0 - \hat{\boldsymbol{m}}_r\|_2}{\|\boldsymbol{m}_0\|_2}\right\}, \tag{73}$$

$$\mathrm{MSE(dB)} \triangleq 20\log_{10}\mathrm{E}\left\{e_r\right\}, \tag{74}$$

$$\mathrm{EMSE(dB)} \triangleq 20\log_{10}\mathrm{E}\left\{\boldsymbol{x}_r^{\mathrm{T}}\widetilde{\boldsymbol{m}}_r\right\}. \tag{75}$$

Case A) 理论结果验证

表V 不同应用场景下所提算法的仿真参数。

算法	US-1 高斯	US-2 高斯	US-2 α 稳定噪声	AEC Gausian US-1	AEC 高斯 US-2	AEC α-稳定噪声 US-2	非线性 案例1 案例2	ANC
NSAF-NKP-I	D_1=25,D_2=20 μ_1=μ_2=0.03 λ=0.01 ,P=2,k=4	μ_1=μ_2=0.02 D_1=25,D_2=20 λ=0.01 ,P=2,k=4	/	D_1=25,D_2=20 μ_1=μ_2=0.03 λ=0.01 ,P=2,k=4	μ_1=μ_2=0.02 D_1=25,D_2=20 λ=0.01 ,P=2,k=4	/	/	/
NSAF-NKP-II	D_1=25, D_2=20 μ_1=μ_2=0.03 λ=0.01 ,P=2,k=4	D_1=25, D_2=20 μ_1=μ_2=0.02 ,P=2,k=4	/	D_1=25, D_2=20 μ_1=μ_2=0.03 λ=0.01 ,P=2,k=4	,P=2,k=4	/	D_1=10, D_2=5 μ_1=μ_2=0.01 λ=0.1 ,P=2,k=4	/
RNSAF-NKP-LC	/	μ_1=μ_2=0.02,β=1 D_1=25, D_2=20 λ=0.01 ,P=2,k=4	μ_1=μ_2=0.02,β=1 D_1=25, D_2=20 λ=0.01 ,P=2,k=4	/	/	μ_1=μ_2=0.08,β=20 D_1=25, D_2=20 λ=0.01 ,P=2,k=4	/	μ_1=μ_2=0.01,k=4 β=0.2, D_1=D_2=10 λ=0.01 ,P=2
RNSAF-NKP-MCC	/	μ_1=μ_2=0.02,ψ=1 D_1=25, D_2=20 λ=0.01 ,P=2,k=4	D_1=25, D_2=20 ,P=2,k=4	/	/	μ_1=μ_2=0.08,ψ=20 D_1=25, D_2=20 λ=0.01 ,P=2,k=4	/	μ_1=μ_2=0.01,k=4 ψ=0.2, D_1=D_2=10 λ=0.01 ,P=2
TFLN-NKP-NSAF	/	/	/	/	/	/	μ_1=μ_2=0.01, A=2, P=2, k=4 B=10, D_1=10, D_2=5, λ=0.1	/
Volterra-NKP-NSAF	/	/	/	/	/	/	B=10, D_1=13, D_2=5, λ=0.1 μ_1=μ_2=0.01,P=2,k=4	/
NKP-FxNSAF	/	/	/	/	/	/	/	D_1=D_2=10,P=2,k=4 μ_1=μ_2=0.01,λ=0.01

输入信号为零均值高斯噪声。如图7(a)所示，我们利用理论模型 (56) 估算 NSAF-NKP-II算法的稳态均方误差。显然，当 $\mu = 0.2$ 和 $\mu = 0.5$时，理论稳态均方误差模型能够准确预测该算法的仿真结果。然而，当 $\mu = 0.7$ 时，理论稳态均方误差值与仿真值不够吻合。因此，研究人员值得进一步探究更精确的理论稳态模型。如图7(b)所示，步长 $\mu = 1$ 超出了 (58) 中规定的范围，导致 NSAF-NKP-II算法发散。

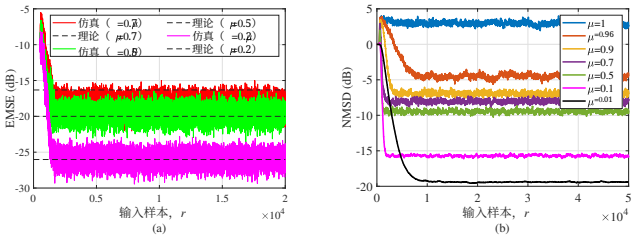


图 7. (a) 理论与仿真的均方偏差随.....变化；(b) 不同步长下 NSAF-NKP-II 算法的归一化均方偏差曲线。

Case B) 参数的影响

图 8 分析了不同参数（例如，N、 L、 k、 P、 λ、 ψ 和 β）对所提出的子带 NKP 算法在辨识 US-2 时性能的影响。在图 8(a) 中，我们分析了当子维度 D_1 和 D_2 与实际系统维度 D 不匹配时（即 $D_1 \times D_2 \neq D$），对 NSAF-NKP-II 算法性能的影响。显然，采用 $D_1 \times D_2 = 400$ 和 $D_1 \times D_2 = 200$ 的 NSAF-NKP-II 算法与采用 $D_1 \times D_2 = 500$ 的 NSAF-NKP-II 算法性能基本相同。然而， $D_1 \times D_2 = 100$ 的情况无法充分描述待辨识系统的特性，从而导致 NSAF-NKP-II 算法的稳态均方误差性能下降。值得注意的是，在系统辨识场景中，我们通常设置 $D_1 \times D_2 = D$ ，以确保自适应滤波器具有足够的自由度，从而准确地对待辨识系统建模。

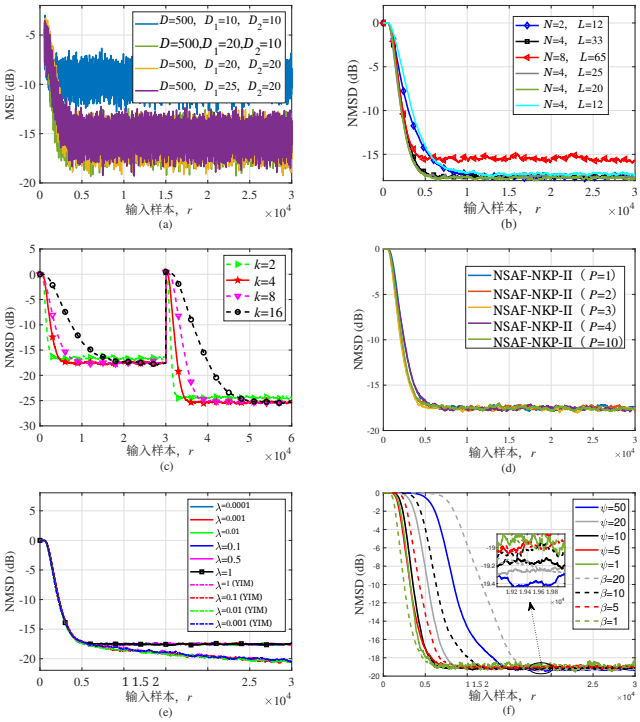
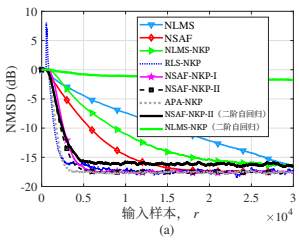


图 8. NSAF-NKP-II算法在不同参数下的NMSD曲线。

图 8(b) 考察了不同参数 N和 L 对 NSAF-NKP-II算法性能的影响。随着子带数量 N 增加，滤波器对相关输入进行去相关的能力增强，从而加快 NSAF-NKP-II算法的收敛速度。然而，在本实验中，如果子带数量超过 $N = 4$ ，算法的收敛速度基本不再变化，但稳态性能会变差。因此，在后续示例中，我们设定 $N = 4$ 。此外，当 $N = 4$时，略小的 L（例如 $L = 25$ 和 $L = 20$）不会对 NSAF-NKP-II算法的性能产生不利影响。不过，L 不能太小，否则会减慢算法的收敛速度。

图 8(c) 描述了 NSAF-NKP-II算法在不同 k下的归一化均方偏差性能。显然，更新间隔 k越小，NSAF-NKP-II算法的收敛速度提升越显著，但计算复杂度也会相应增加。值得注意的是，当 k 的数量小于 N时，NSAF-NKP-II算法的收敛速度提升变得不明显，稳态归一化均方偏差也会增大。因此，在后续示例中，我们设定 $k = 4$ 。此外，我们通过在输入样本中间引入突变（从US-2 到 US-1），考察算法在动态系统条件下的跟踪效果。数值结果证实，NSAF-NKP-II算法具有强跟踪能力，能够在时变环境中快速适应突发的参数变化。



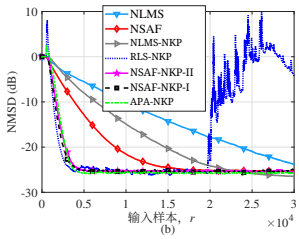


图10。(a) 使用算法识别US-2时的NMSD曲线，$[P = 2, \mu_1 = \mu_2 = 0.02]$ $[\lambda_1 = 0.998$, 对于NLMS-NKP算法; $\lambda_2 = 0.998$, $P = 2]$ $[P = 2, C = 5, \mu_1 =$ 对于RLS-NKP算法，$\mu_2 = 0.008]$ $[\mu = 0.02]$ 对于APA-NKP；对于NLMS和NSAF算法；(b) 使用算法识别US-1时的NMSD曲线，$[\mu = 0.03]$ $[\mu_1 = \mu_2 = 0.009]$ 对于NLMS-NKP算法；对于APA-NKP； $[\mu = 0.03]$ 对于NLMS和NSAF算法，算法的其他参数与图10(a)中的相同。

图 8(d) 和 (e) 分析了参数 P 和初始值 λ 对 NSAF-NKP-II算法性能的影响。在当前实验场景中，所提出的 NSAF-NKP-II 算法不受参数 P 和 λ 的影响。此外，基于 YIM 方法的 NSAF-NKP-II 算法比原始方法具有更低的稳态误差。不过，为了与其他 NKP 自适应滤波算法保持一致，我们仍选择使用原始方法。

图 8(f) 展示了 RNSAF-NKP-LC 和 RNSAF-NKP-MCC 算法在不同 β和 ψ 参数值下的 NMSD 性能。结果表明， β 和 ψ的值越大，RNSAF-NKP-LC 和 RNSAF-NKP-MCC 算法的稳态误差就越低。不过，这些值不能过大，因为我们还需要考虑基于梯度的自适应方法的收敛速度。

在图9中，我们考察了 US-2 条件下 NSAF-NKP-II、RNSAF-NKP-MCC 和 RNSAF-NKP-LC算法在高斯噪声中的性能。显然，与 NSAF-NKP-II算法相比，即使引入相应的鲁棒性准则，鲁棒子带 NKP 算法在收敛速度和稳态归一化均方偏差方面的性能在高斯噪声场景下也不会降低。

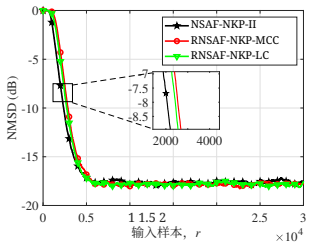


图9. 高斯噪声下 NSAF-NKP-II、RNSAF-NKP-MCC 和 RNSAF-NKP-LC算法的NMSD曲线。

Case C) 高斯噪声场景

图10(a)比较了用于辨识US-2系统的NLMS[8]，NSAF [13]，NLMS-NKP [23],RLS-NKP [26]，APA- NKP [25]，NSAF-NKP-I和NSAF-NKP-II算法的NMSD曲线。在该图中，由方程 x_r =生成的二阶自回归（AR(2)）输入信号 $1.5x_{r-1} - 0.6x_{r-2} + w_r$。值得注意的是，此前尚未在系统辨识背景下研究过基于仿射投影的NKP算法。为确保比较公平，我们将 [25]中的NKP-MFxAP算法扩展到系统辨识场景，从而得到NKP-APA算法。比较分析表明，与NLMS算法相比，NSAF算法凭借其子带自适应滤波结构展现出更优的收敛性能。值得注意的是，与NLMS算法相比，NLMS-NKP算法在初始收敛特性方面更为优越。所提出的NSAF-NKP-I算法进一步提升了相较于NLMS-NKP方法的收敛速度，但代价是乘法运算的计算负荷增加。所幸，所提出的NSAF-NKP-II算法在大幅降低计算复杂度的同时，实现了与NSAF-NKP-I相当的收敛性能。尽管RLS-NKP和APA-NKP算法的收敛速度略快于所提出的算法，但这一优势伴随着更高的计算负荷。当输入信号的相关性进一步增强，即输入为二阶自回归信号时，NLMS-NKP算法的性能会急剧恶化。然而，所提出的NSAF-NKP-II算法的性能与输入为一阶自回归信号时观察到的情况基本一致。此外，定阶RLS-NKP算法在有限精度实现中容易出现数值不稳定性，图10(b)中观察到的发散行为便是证据。图10(b)比较了图10(a)所示算法在US-1场景下的性能。显然，我们可以得出与图10(a)中所述结论类似的一些结论。

Case D) α-稳定噪声情形

脉冲噪声 v_r 由 α-稳定过程建模，其特征函数为 [56]

$$v_r = \exp(-\gamma|r|^{\varpi}), \tag{76}$$

其中，特征指数 $\varpi \in (0, 2]$ 控制噪声的脉冲性， $\gamma > 0$ 决定噪声离散度（本文中为 $\gamma = 1/60$ ），而当 $\varpi = 2$ 时，噪声 v_r 表示高斯噪声。

图11比较了NSAF-NKP-II、基于MCC的NLMS-NKP（NLMS-NKP-MCC）和NKP-GMCC的NMSD曲线

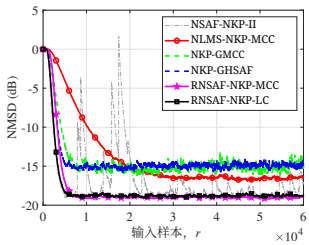


图11. α $[\alpha = 2$, -稳定噪声下，不同算法用于US-2辨识的NMSD曲线。所有算法均具有 $\mu_1 = 0.02$ 和 $\mu_2, \lambda = 0.02$。NKP-GHSAF算法为0.0002, $\mu = 0.003$] [α $[P = 2]$ ；NKP-GMCC算法为；NKP分解为； α-稳定噪声为 $[\varpi = 1.5]$ 。 $\lambda = 0.99$, $\mu = 0.0053$]

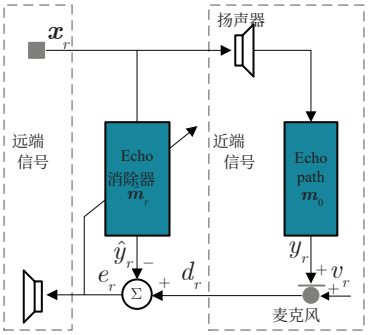


图12. 自适应回声消除器的结构。

[35],NKP-GHSAF [35],RNSAF-NKP-MCC以及RNSAF-NKP-LC算法在US-2辨识中的表现。值得注意的是，此前尚无研究探讨NLMS-NKP-MCC算法。为确保比较公平，我们将MCC技术扩展应用于现有的NLMS-NKP框架。当然，非鲁棒的NSAF-NKP-II算法在脉冲噪声下表现出不稳定的收敛行为。NLMS-NKP-MCC算法在脉冲噪声环境中能够稳定收敛，这是因为MCC准则会在出现脉冲噪声时将步长减小到极小值。通过采用GMCC和GHSF准则作为鲁棒代价函数，NKP-GMCC和NKP-GHSAF算法能够在脉冲噪声环境中稳定收敛；此外，它们的收敛速度比NLMS-NKP-MCC算法更快，但稳态失调相对较大。有趣的是，所提出的RNSAF-NKP-MCC和RNSAF-NKP-LC算法在实现最快收敛速度的同时，稳态误差也最低。

B. 回声消除

在回声消除场景中，我们将添加另一个指标来评估算法在这些场景中的性能，即回声回波损耗增强（ERLE），其定义为 [38]

$$\mathrm{ERLE(dB)} \triangleq 10\log_{10}\left(\frac{\mathrm{E}\{d_r^2\}}{\mathrm{E}\{e_r^2\}}\right), \tag{77}$$

其中，期望值通过时间平均进行估计。

Case A）高斯噪声场景

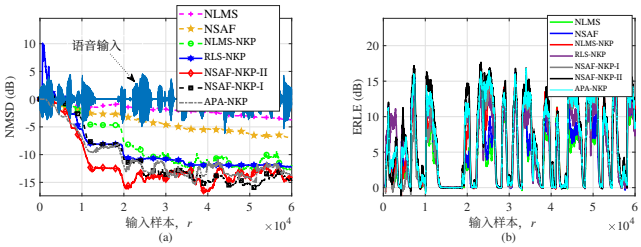


图 13. 在 US-2 的辨识中，NLMS、NSAF、NLMS-NKP、RLS-NKP、APA-NKP、NSAF-NKP-I 和 NSAF-NKP-II算法的 NMSD 和回声回波损耗增强曲线。各算法的参数与图10中的相同。

对于图12所示的回声消除系统， $\boldsymbol{x}_r$表示语音输入信号。回声路径 $\boldsymbol{m}_0$ 由 US-1 和 US-2 系统构成。将语音输入信号 $\boldsymbol{x}_r$ 通过回声路径后，即可得到回声信号 y_r。使用相同的输入信号，回声消除器通过自适应滤波器 $\hat{\boldsymbol{m}}_r$识别回声路径，其输出为 $\hat{y}_r$，对应于回声信号 y_r的估计值。随后，从期望信号 d_r中减去 $\hat{y}_r$ ，即可得到质量更高且不受回声干扰的语音信号 e_r。

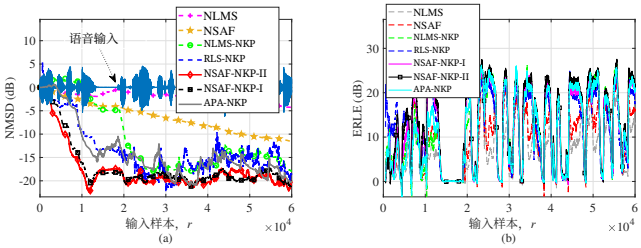


图14. 在识别US-1时，不同算法的归一化均方偏差和回声回波损耗增强曲线。算法的参数与图10中的相同。

图 13 比较了各算法在 US-2 辨识中的性能。图 13(a) 中，NSAF算法的收敛速度快于NLMS算法。采用NKP分解技术后，与NSAF算法相比，NLMS-NKP算法表现出更好的收敛特性，且稳态失调更小。RLS-NKP算法通过利用输入信号的自相关矩阵更新权重，相比NLMS-NKP算法实现了更高的稳态精度。利用历史输入数据的APA-NKP算法，其稳态性能略优于RLS-NKP算法。所提出的NSAF-NKP-I算法与NLMS-NKP算法的性能相近。然而，RLS-NKP、APA-NKP和NSAF-NKP-I算法的优势是以增加计算复杂度为代价的。所提出的NSAF-NKP-II算法有效解决了这一局限（即有效降低了计算复杂度），同时实现了更优的收敛性能。图 13(b) 中，我们采用ERLE评估指标进一步验证所提出算法的有效性，得出了类似结论。图 14 比较了 NLMS、NSAF、NLMS-NKP、RLS-NKP、

APA-NKP、NSAF-NKP-I 和 NSAF-NKP-II 算法在 US-1 辨识中的性能。我们也可以得出与图 13 中所述类似的结论。

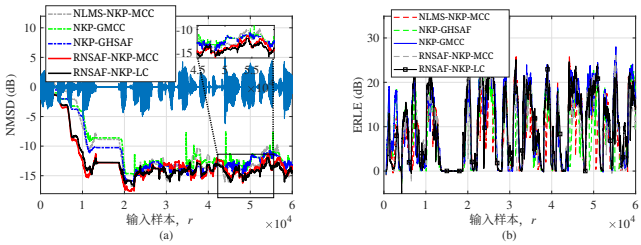


图 15. 在 α-稳定噪声下进行 US-2 辨识时的 NMSD 和 ERLE 曲线。 $[\mu_1 = \mu_2 = 0.08]$ 表示 $[\alpha = 2,$ NLMS-NKP-MCC 算法; $\lambda = 0.99,\ \mu = 0.01]$ 表示 NKP-GHSAF 算法; $[\alpha = 2,\ \lambda = 0.0006,$ 表示 $\mu = 0.007]$ NKP-GMCC 算法。各算法的参数与图 11 中的相同。

Case B) α-稳定噪声场景

图 15 对比了所提出的鲁棒算法与其他算法在 US-2 辨识中的表现。相比之下，所提出的 RNSAF-NKP-MCC 和 RNSAF-NKP-LC 算法收敛最快，且稳态失调量最低。

C. 复杂非线性场景

在非线性场景中，输入信号 x_r 由取值范围为 $[-0.5, 0.5]$的均匀分布随机序列生成，噪声 v_r 为方差为 $\sigma_v^2 = 0.001$的白高斯噪声。

Case A）非对称扬声器失真 [57]非线性模型建立如下

$$a_r = \phi\Big[\frac{1}{1+e^{-\vartheta a_{r,1}}} - \frac{1}{2}\Big], \tag{78}$$

其中 $a_{r,\ 1} = 1.5x_r - 0.3x_r^2$、 $\phi = 2$ 和

$$\begin{cases} \vartheta = 4 \text{ if } a_{r,1} > 0 \\ \vartheta = \dfrac{1}{2} \text{ if } a_{r,1} \le 0. \end{cases}$$

图 16(a) 对比了在非对称扬声器失真条件下，NSAF-NKP-II、TFLN-NSAF [39], TFLN-NKP-NLMS [19], TFLN-NKP-NSAF、Volterra-NKP-NSAF、Volterra-NKP-NLMS [19], 和 Volterra-NSAF [38] 算法的均方误差曲线。遗憾的是，所提出的线性 NSAF-NKP-II 算法在遇到这种非对称扬声器失真时，稳态精度表现不佳。相比之下，基于 TFLN 和 Volterra 结构的非线性算法在收敛性能和稳态性能方面均能取得更好的均方误差表现。对于采用 TFLN 结构的算法，所提出的 TFLN-NKP-NSAF 算法比 TFLN-NSAF 和 TFLN-NKP-NSAF 算法收敛更快、稳态精度更高。此外，对于采用 Volterra 结构的算法，也可以得出与基于 TFLN 的算法类似的结论。

Case B）对称软削波非线性 [58]

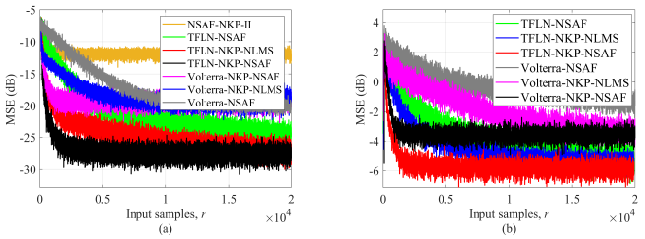


图16. 所提出的算法与竞争算法的比较。均方误差曲线是对200次独立试验的结果取平均后得到的。$[A = 2,\ B = 10,\ D = 50,\ D_1 = 10,\ D_2 = 5,\ B = 10,\ D = 65,\ D_1 = 13,\ D_2 = 5,$ 适用于基于TFLN的算法; $\mu_1 = \mu_2 = 0.01,\ P = 2]$ 适用于基于Volterra的算法。

非线性模型如下

$$a_r = \begin{cases} \dfrac{2}{3\tau}x_r, & 0 \le |x_r| \le \tau \\ \text{sign}[x_r]\dfrac{3 - [2 - x_r/\tau]^2}{3}, & \tau \le |x_r| \le 2\tau \\ \text{sign}[x_r], & 2\tau \le |x_r| \le 1 \end{cases} \tag{79}$$

其中，在我们的仿真中， $\tau \in (0, 0.5]$ ($\tau = 0.3$ 。

图16(b)比较了对称软削波非线性环境下所提出的算法与竞争算法。如图所示，所提出的TFLN-NKP-NSAF和Volterra-NKP-NLMS算法的性能优于其竞争算法。

Case C) 非线性算法的乘法复杂度

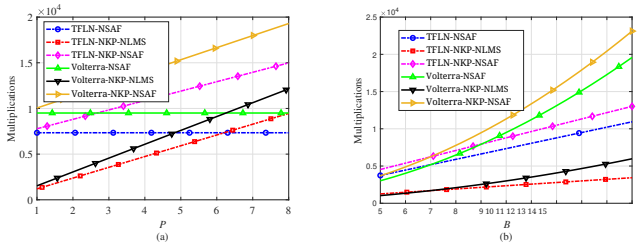


图17. (a) 乘法运算的计算复杂度与参数 P 的关系；(b) 乘法运算的计算复杂度与参数 B 的关系。 $N = 4$ 和 $k = 4$ 对应图17(a)； D_1=D_2=$\sqrt{D}$ 和 $P = 2$ 对应图17(b)。算法的其他参数与图16中的相同。

根据备注5，图17(a)比较了非线性算法的乘法运算复杂度与参数 P 的关系。所提出的 Volterra-NKP-NSAF 和 TFLN-NKP-NSAF 算法所需的乘法运算次数多于其对应算法。此外，与基于 Volterra 的方法相比，基于 TFLN 结构的算法不仅计算成本更低，而且稳态失调也更低。显然，从图17(b)中可以得出与图17(a)类似的结论。

D. 主动噪声控制

本节介绍滤波-x NSAF-NKP-II（NKP-FxNSAF）、滤波-x RNSAF-NKP-MCC（NKP-FxNSAF-MCC）和滤波-x RNSAF-NKP-LC（NKP-FxNSAF-LC）算法。图18展示了

ANC场景下的子带 NKP 分解模型，其中 $P(z)$ 和 $S(z)$ 分别表示主路径和次级路径，$\hat{S}(z)$ 表示 $S(z)$ 的估计值。与图 2 所示的系统辨识模型相比，图18所示的 ANC 模型只有两处不同。首先，分析滤波器 $\{\boldsymbol{f}_j\}_{j=1}^{N}$ 的输入信号是滤波后的输入信号 x_r'。其次，在噪声抵消点，只有 e_r 是可测信号，而 d_r 和 y_r 是不可测的。因此，将误差向量 $\boldsymbol{e}_r=[e_r,e_{r-1},...,e_{r-L+1}]$ 通过分析滤波器处理后，可得到子带误差信号 $\{e_{r,j}\}_{j=1}^{N}$，即

$$\boldsymbol{e}_{r,s}\triangleq[e_{r,1},e_{r,2},...,e_{r,N}]=\boldsymbol{e}_r^{\mathrm{T}}\boldsymbol{F}, \tag{80}$$

其中 $\boldsymbol{e}_{r,\ s}$ 表示子带误差向量。此外，算法性能指标平均噪声降低量（ANR）定义为 [38]

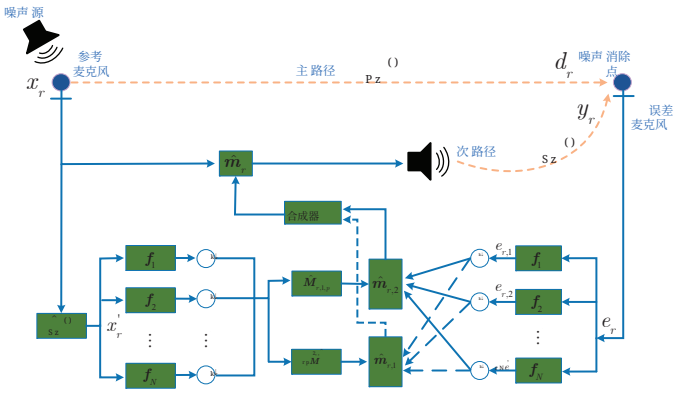


图18. ANC场景下子带 NKP 分解模型的结构。

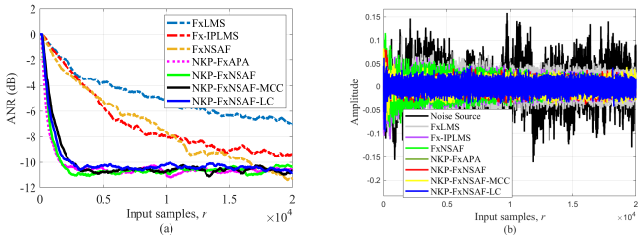


图19. (a) 多种算法的 ANR 曲线，以及 (b) 粉红噪声的降噪结果。 $\mu=0.01$ 表示 FxLMS 算法； $\mu=0.1$、$\alpha_1=-0.5$ $\epsilon_1=10^{-6}$ $\mu=0.01$ 和 表示 Fx-IPLMS 算法；表示 FxNSAF 算法； $C=5$、$\mu_1=\mu_2=0.004$ 表示 NKP-FxAPA 算法。

$$\mathrm{ANR}(r)\triangleq 10\log\left\{\frac{S_e^2(r)}{S_d^2(r)}\right\}, \tag{81}$$

其中， $S_e(r)=\eta S_e(r-1)+(1-\eta)|e_r|$ 和 $S_d(r)=\eta S_d(r-1)+(1-\eta)|d_r|$ 分别表示残差和期望信号的平均幅值，且 $\eta=0.999$。

图19(a)比较了滤波-x 最小均方（FxLMS） [59], 滤波-x 改进比例最小均方（FxIPLMS） [60],滤波-x 归一化子带自适应滤波（FxN- SAF）、基于 NKP 分解的 FxAP（NKP-FxAP） [25], 滤波-x NSAF-NKP-II（NKP-FxNSAF）、滤波-x NSAF-NKP-MCC（NKP-FxNSAF-MCC）和滤波-x NSAF-NKP-LC（NKP-FxNSAF-LC）算法。主路径和次级路径分别采用有限冲激响应（FIR）滤波器建模，其传递函数为 $P(z)=z^{-3}-0.3z^{-4}+0.2z^{-5}$和 $S(z)=z^{-2}+0.5z^{-5}$ [38]。显然，与 FxLMS、Fx-IPLMS 和 FxNSAF 算法相比，所提出的算法表现出更好的收敛性能，同时达到与 NKP-FxAPA 算法一致的性能。如图19(b)所示，我们可以得出与图19(a)相同的结论。

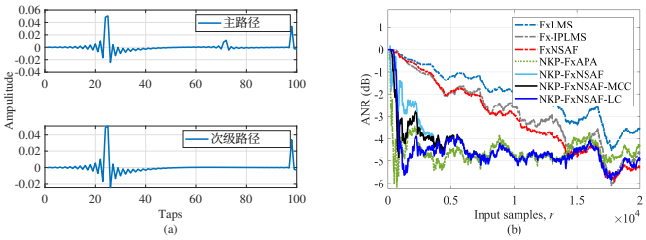


图20。（a）主动噪声控制系统的时域脉冲响应；（b）若干算法的 ANR 曲线。各算法的参数与图19中的相同。

图20（a）展示了主路径和次级路径的时域脉冲响应，这些响应使用镜像法 [61]生成。研究表明，镜像法在模拟房间脉冲响应方面有效且易于使用，因此已广泛应用于音频信号处理领域。所用噪声源信号是从牵引变电所录制的真实噪声 [62]。

图20（b）比较了各算法在牵引变电所噪声下的 ANR 曲线。值得注意的是，我们将[25] 中的多通道NKP分解 FxAP 算法应用于单通道 ANC场景，从而得到 NKP-FxAP算法。显然，与其他算法相比，所提出的 NKP-FxNSAF、NKP-FxNSAF-MCC 和 NKP-FxNSAF-LC 算法在主动噪声控制系统中的收敛速度更快。

VII. 结论

本文提出了一种新颖的框架，将最近克罗内克积分解技术拓展至子带自适应滤波域。所开发的 NSAF-NKP-I 算法在稀疏系统辨识任务中的收敛速度快于传统 NSAF 实现，但代价是计算复杂度有所增加。为降低 NSAF-NKP-I 算法的计算复杂度，我们随后开发了 NSAF-NKP-II算法。为解决 NSAF-NKP-II 在脉冲噪声干扰下稳定性下降的问题，我们提出了 RNSAF-NKP-MCC 和 RNSAF-NKP-LC 两种变体，它们在脉冲噪声场景中表现出稳健的收敛性。此外，我们还将所提出的基于子带 NKP 的算法与现有竞争算法的计算复杂度进行了比较。为确保 NSAF-NKP-II 算法在复杂非线性环境中的可靠性，我们进一步提出了一类非线性

子带最近克罗内克积算法，即 TFLN-NKP-NSAF 和 Volterra-NKP-NSAF。涵盖回声消除、主动噪声控制、系统辨识和复杂非线性场景的全面仿真一致验证了所提出算法的优越性能。对于所提出的基于子带 NKP 的自适应滤波器，一个潜在的研究方向是在瞬态阶段和稳态阶段自适应调整子滤波器的抽头长度，以进一步优化计算效率和收敛性能。

参考文献

[1] G. Guo、Y. Yu、R. C. de Lamare、Z. Zheng、L. Lu 和 Q. Cai，“用于声学回声消除的近端归一化子带自适应滤波”，IEEE/ACM 音频、语音与语言处理汇刊，第 29 卷，页 2174–2188，2021年。[2] J. Ye、Y. Yu、Y. Zakharov、Z. Liu 和 H. He，“基于卡尔曼机制的自适应子带前向盲源分离算法”，IEEE仪器与测量汇刊，2024年。[3] M. Z. A. Bhotto 和 I. V. Baji´c，“基于无迹卡尔曼滤波的恒模盲自适应波束形成”，IEEE 信号处理快报，第 22 卷，第 4 期，页 474–478，2015年。[4] Z. Luo、D. Shi 和 W.-S. Gan，“一种基于深度学习的主动噪声控制混合SFANC-FXNLMS算法”，IEEE 信号处理快报，第 29 卷，页 1102–1106，2022年。[5] J. Chen 和 J. Yang，“一种用于窄带主动噪声控制的分布式FXLMS算法及其收敛分析”，Journal of Sound and Vibration，第 532 卷，文章号 116986，2022年。[6] A. Kar、G. Singh、P. K. Shill、S. Pradhan、Vasundhara 和 M. G. Christensen，“一种改进的增益归一化步长自适应算法，用于主动噪声控制中的在线次级路径建模”，载于 ICASSP 2025 - 2025 IEEE 国际声学、语音与信号处理会议（ICASSP），2025年，页 1–5。[7] T. T. Silva、F. Igreja、P. Lara、L. Tarrataca、A. Kar 和 D. B. Haddad，“关于LMS自适应权重的偏度”，IEEE 电路与系统汇刊 II：快报，第 68 卷，3026，2021 8，3022 no. pp. – 。[8] D. T. Slock，“关于LMS算法和归一化LMS算法的收敛行为”，IEEE 信号处理汇刊，第 41 卷，第 9 期，页 2811–2825，1993年。[9] A. Kar 和 M. Swamy，“线性自适应滤波器抽头长度优化算法的收敛与稳态分析”，AEU- 国际电子与通信杂志，第 70 卷，第 9 期，页 1114–1121，2016。[10] M. Ferrer、M. de Diego、G. Pi˜nero 和 A. Gonzalez，“用于主动噪声控制的声学传感器网络仿射投影算法”，IEEE/ACM 音频、语音与语言处理汇刊，第 29 卷，页 448–461，2021年。[11] L. Ljung 和 T. S¨oderstr¨om，《递归辨识理论与实践》。MIT出版社，1983。[12] M. Towliat、Z. Guo、L. J. Cimini、X.-G. Xia 和 A. Song，“用于时变系统辨识的多层递归最小二乘法”，IEEE 信号处理汇刊，第 70 卷，页 2280–2292，2022年。[13] K.-A. Lee 和 W.-S. Gan，“利用约束子带更新改进NLMS算法的收敛性”，IEEE 信号处理快报，第 11 卷，第 9 期，页 736–739，2004。[14] C. Paleologu、J. Benesty 和 S. Ciochin˘a，“基于克罗内克积分解的线性系统辨识”，IEEE/ACM 音频、语音与语言处理汇刊，第 26 卷，第 10 期，页 1793–1808，2018。[15] J. Benesty、C. Paleologu、L.-M. Dogariu 和 S. Ciochin˘a，“线性与双线性系统辨识：统一研究”，Electronics，第 10 卷，第 15 期，文章号 1790，2021年。[16] L.-M. Dogariu、J. Benesty、C. Paleologu 和 S. Ciochin˘a，“通过克罗内克积分解辨识房间声学脉冲响应”，IEEE/ACM 音频、语音与语言处理汇刊，第 30 卷，页 2828–2841，2022年。[17] X. Wang、G. Huang、J. Benesty、J. Chen 和 I. Cohen，“基于克罗内克积分解的到达时间差估计”，IEEE 信号处理快报，第 28 卷，页 51–55，2020年。[18] M. Rupp 和 S. Schwarz，“一种张量LMS算法”，载于 2015 IEEE 国际声学、语音与信号处理会议（ICASSP）。IEEE，2015年，页 3347–3351。

[19]S. S. Bhattacharjee 和 N. V. George，“基于最近克罗内克积分解的参数线性非线性滤波器”，IEEE/ACM 音频、语音与语言处理汇刊，第 29 卷，页 2111–2122，2021年。[20] G. Huang、J. Benesty、I. Cohen 和 J. Chen，“用于基于自适应加权预测误差的语音去混响的克罗内克积多通道线性滤波”，IEEE/ACM 音频、语音与语言处理汇刊，第 30 卷，页 1277–1289，2022年。[21] X. Wang、J. Benesty、J. Chen、G. Huang 和 I. Cohen，“通过克罗内克积分解实现立方体麦克风阵列波束形成”，IEEE/ACM 音频、语音与语言处理汇刊，第 29 卷，页 1774–1784，2021年。[22] S. Vadhvana、S. K. Yadav、S. S. Bhattacharjee 和 N. V. George，“一种基于低秩近似、用于快速自适应波束形成的改进约束最小均方算法”，IEEE 电路与系统汇刊 II：快报，第 69 卷，第 8 期，页 3605–3609，2022年。[23] S. S. Bhattacharjee 和 N. V. George，“基于最近克罗内克积分解的归一化最小均方算法”，载于 ICASSP 2020-2020 IEEE 国际声学、语音与信号处理会议（ICASSP）。IEEE，2020年，页 476–480。[24] J. Benesty、C. Paleologu 和 S. Ciochin˘a，“利用维纳滤波器辨识双线性形式”，IEEE 信号处理快报，第 24 卷，第 5 期，页 653–657，2017年。[25] L. Li、S. Wang、S. S. Bhattacharjee、J. R. Jensen 和 M. G. Christensen，“基于最近克罗内克积分解的主动噪声控制多通道滤波-x仿射投影算法”，机械系统与信号处理，第 224 卷，文章号 112055，2025年。[26] C. Elisei-Iliescu、C. Paleologu、J. Benesty 和 S. Ciochina，“一种基于最近克罗内克积分解的递归最小二乘算法”，载于 ICASSP 2019-2019 IEEE 国际声学、语音与信号处理会议（ICASSP）。IEEE，2019年，页 4843–4847。[27] K. Kumar、M. L. S. Karthik 和 N. V. George，“一种基于逻辑距离度量的新型稀疏感知鲁棒自适应滤波器系列”，IEEE 信号处理汇刊，第 70 卷，页 6128–6141，2022年。[28] Y. Yu、H. He、B. Chen、J. Li、Y. Zhang 和 L. Lu，“基于M估计的归一化子带自适应滤波算法：性能分析与改进”，IEEE/ACM 音频、语音与语言处理汇刊，第 28 卷，页 225–239，2019年。[29] M. Shao 和 C. L. Nikias，“分数低阶矩信号处理：稳定过程及其应用”，IEEE 会刊，第 81 卷，第 7 期，页 986–1010，1993年。[30] J. Ye、Y. Yu、Q. Cai、T. Yu 和 B. Chen，“基于P范数的子带自适应滤波算法：性能分析与改进”，电路、系统与信号处理，第 43 卷，1239，2024 2，1208 no. pp. – 。[31] J. Ni 和 F. Li，“可变正则化参数符号子带自适应滤波器”，电子快报，第 46 卷，第 24 期，页 1605–1607，2010年。[32] J. Ye、H. Zhao 和 Y. Peng，“用于脉冲噪声和含噪输入的基于P范数的分数阶鲁棒子带自适应滤波算法”，IEEE 信号处理快报，页 1–5，2025年。[33] B. Chen 、 L. Xing 、 J. Liang 、 N. Zheng 和 J. C. Principe，“ 最大相关熵准则下自适应滤波的稳态均方误差分析”，IEEE 信号处理快报，第 21 卷，第 7 期，页 880–884，2014年。[34] F. Huang、J. Zhang 和 S. Zhang，“基于最大维索里亚准则的鲁棒自适应滤波算法”，IEEE 电路与系统汇刊 II：快报，第 64 卷，第 10 期，页 1252–1256，2017年。[35] S. S. Bhattacharjee、K. Kumar 和 N. V. George，“基于最近克罗内克积分解的广义最大相关熵与广义双曲正割鲁棒自适应滤波器”，IEEE 信号处理快报，第 27 卷，页 1525–1529，2020年。[36] Q. Zhu，《非线性系统》。MDPI 多学科数字出版研究所，2023年。[37] S. Chen 和 S. A. Billings，“用于非线性动态系统建模与辨识的神经网络”，国际控制杂志，第 56 卷，第 2 期，页 319–346，1992年。[38] J. Ye、Y. Yu、B. Chen、Z. Zheng 和 J. Chen，“函数连接神经网络上的最优子带自适应滤波器：算法与应用”，IEEE 电路与系统汇刊 I：常规论文，页 1 14，2024 – 。[39] S. Zhang、W. X. Zheng 和 H. Han，“无延迟多采样子带函数连接神经网络的设计及其在主动噪声控制中的应用”，信号处理，第 202 卷，文章号 108757，2023年。[40] Y. Yu、L. Lu、Z. Zheng 和 X. Yang，“用于非线性主动噪声控制的插值个体加权子带沃尔泰拉滤波器

电路与系统 II：快报，第 70 卷第 2 期，第 816–820 页，2022年。[41] S. Zhang、Z. Zhou、W. X. Zheng 和 X. Tang，“频域 Volterra 核自适应：公式与算法”，《信号处理》，第 216 卷，文章编号 109295，2024年。[42] L. Shi、J. Tan、J. Wang、Q. Li、L. Lu 和 B. Chen，“用于非线性时间序列预测的鲁棒核自适应滤波”，《信号处理》，第 210 卷，文章编号 109090，2023年。[43] V. C. Gogineni、R. Sambangi、D. Alex、S. Mula 和 S. Werner，“基于随机傅里叶特征的核自适应滤波器的算法与架构设计”，《IEEE 电路与系统汇刊 I：常规论文》，第 70 卷第 2 期，第 833–845 页，2022年。[44] M. Scarpiniti、D. Comminiello、G. Scarano、R. Parisi 和 A. Uncini，“样条自适应滤波器的稳态性能”，《IEEE 信号处理汇刊》，第 64 卷第 4 期，第 816–828 页，2016年。[45] H. Zhao、Y. Gao 和 R. Zhu，“最小均值 p 次幂 Hammerstein 样条自适应滤波算法：公式与分析”，《IEEE 航空航天与电子系统汇刊》，第 60 卷第 5 期，第 6275–6283 页，2024年。[46] C. Elisei-Iliescu、C. Paleologu、J. Benesty、C. Stanciu、C. Anghel 和 S. Ciochin˘a，“低秩系统辨识的递归最小二乘算法”，《IEEE/ACM 音频、语音与语言处理汇刊》，第 27 卷第 5 期，第 903–918 页，2019年。[47] C. F. Van Loan，“无处不在的克罗内克积”，《计算与应用数学杂志》，第 123 卷第 1-2 期，第 85–100 页，2000年。[48] G. H. Golub 和 C. F. Van Loan，《矩阵计算》。约翰斯·霍普金斯大学出版社，2013年。[49] D. A. Harville，“从统计学家的视角看矩阵代数”，1998年。[50] K.-A. Lee、W.-S. Gan 和 S.-M. Kuo，“归一化子带自适应滤波器的均方性能分析”，载于《2006 年第四十届阿西洛马会议：信号、系统与计算机》，2006年，第 248–252 页。[51] I. Song、P. Park 和 R. W. Newcomb，“一种针对脉冲测量噪声、带步长缩放器的归一化最小均方算法”，《IEEE 电路与系统汇刊 II：快报》，第 60 卷第 7 期，第 442–445 页，2013年。[52] M. O. Sayin、N. D. Vanli 和 S. S. Kozat，“一种基于对数代价函数的新型自适应滤波算法族”，《IEEE 信号处理汇刊》，第 62 卷第 17 期，第 4411–4424 页，2014年。[53] I. Song、P. Park 和 R. W. Newcomb，“一种针对脉冲测量噪声、带步长缩放器的归一化最小均方算法”，《IEEE 电路与系统汇刊 II：快报》，第 60 卷第 7 期，第 442–445 页，2013 年。[54] H. Zhao 、 Y. Yu 、 S. Gao 、 X. Zeng 和 Z. He，“用于声学回声消除、带独立激活因子的记忆比例化仿射投影算法”，《IEEE/ACM 音频、语音与语言处理汇刊》，第 22 卷第 6 期，第 1047–1055 页，2014年。[55] “数字网络回声消除器”，ITU-T 建议 G.168，2002年。[56] S. P. Talebi、S. Werner 和 D. P. Mandic，“对 α-稳定信号进行分布式自适应滤波”，《IEEE 信号处理快报》，第 25 卷第 10 期，第 1450–1454 页，2018年。[57] D. Comminiello、M. Scarpiniti、L. A. Azpicueta-Ruiz、J. Arenas-Garcia 和 A. Uncini，“用于非线性声学回声消除的函数链接自适应滤波器”，《IEEE 音频、语音与语言处理汇刊》，第 21 卷第 7 期，第 1502–1512 页，2013年。[58] D. Comminiello、M. Scarpiniti、L. A. Azpicueta-Ruiz、J. Arenas-Garc´ıa 和 A. Uncini，“基于稀疏函数链接表示的非线性声学回声消除”，《IEEE/ACM 音频、语音与语言处理汇刊》，第 22 卷第 7 期，第 1172–1183 页，2014年。[59] S. Elliott，《主动控制的信号处理》。爱思唯尔，2000年。[60] J. Arenas-Garc´ıa、M. de Diego、L. A. Azpicueta-Ruiz、M. Ferrer 和 A. Gonzalez，“声学中的比例自适应滤波器组合：主动噪声控制应用”，载于《2011 年第 19 届欧洲信号处理会议》。IEEE，2011年，第 1269–1273 页。[61] J. B. Allen 和 D. A. Berkley，“高效模拟小房间声学的镜像法”，《美国声学学会杂志》，第 65 卷第 4 期，第 943–950 页，1979年。[62] Y. Zhu、H. Zhao、X. Zeng 和 B. Chen，“用于主动噪声控制的鲁棒广义最大相关熵准则算法”，《IEEE/ACM 音频、语音与语言处理汇刊》，第 28 卷，第 1282–1292 页，2020年。