

## 基于箕舌线可变步长 LMS 的空频抗干扰算法

郭辰锋<sup>1</sup>, 舒东亮<sup>1</sup>, 路寅<sup>1</sup>, 靳小琴<sup>2</sup>

(1. 天津航海仪器研究所, 天津 300131; 2. 南开大学人工智能学院, 天津 300350)

**摘要:** 针对空频最小均方 (Least mean square, LMS) 算法抗干扰性能与收敛速度不能兼顾的问题, 提出了一种基于箕舌线可变步长 LMS 的空频抗干扰算法, 简称空频基于箕舌线的可变步长 LMS 算法 (Variable step LMS of tongue-like curve function, TLCVSLMS) 算法。在兼顾抗干扰性能与收敛速度的基础上, 空频 TLCVSLMS 算法避免了针对每一个频点人为地选取合适的固定迭代步长因子  $\mu$  的困难, 并根据不同频点的信号功率, 对箕舌线函数的幅度因子与形状因子作更精细的调节。仿真实验表明, 在抗干扰性能接近的情况下, 空频 TLCVSLMS 算法比空频 LMS 算法少迭代至少 400 点, 空频 TLCVSLMS 算法的收敛速度更快, 而在收敛速度相同的情况下, 空频 TLCVSLMS 算法比空频 LMS 算法的抗干扰性能提升至少 3 dB 以上。

**关键词:** 空频抗干扰; 最小均方算法; 箕舌线函数; 收敛速度; 抗干扰性能

**中图分类号:** TN966.2      **文献标志码:** A

## A Space-Frequency Anti-jamming Algorithm Based on Variable Step LMS of Tongue-Like Curve Function

GUO Chenfeng<sup>1</sup>, SHU Dongliang<sup>1</sup>, LU Yin<sup>1</sup>, JIN Xiaoqin<sup>2</sup>

(1. Tianjin Institute of Marine Instruments, Tianjin 300131, China; 2. College of Artificial Intelligence, Nankai University, Tianjin 300350, China)

**Abstract:** To solve the problem that the space-frequency algorithm based on least mean square (LMS) cannot consider the anti-jamming performance and the convergence speed simultaneously, a space-frequency anti-jamming algorithm based on variable step LMS of tongue-like curve function is proposed as space-frequency variable step LMS of tongue-like curve function (TLCVSLMS) algorithm. On the basis of both anti-interference performance and convergence speed, the space frequency TLCVSLMS algorithm avoids the difficulty of artificially selecting a suitable fixed iterative step size factor for each frequency point, and makes more precise adjustments to the amplitude factor and shape factor of the tongue line function based on the signal power of different frequency points. Simulation results show that, when the anti-interference performance is close, the space-frequency TLCVSLMS algorithm has at least 400 fewer iterations than the space-frequency LMS algorithm, and the convergence speed of the space-frequency TLCVSLMS algorithm is faster. When the convergence speed is proximate, the anti-interference performance of the space-frequency TLCVSLMS algorithm is improved at least 3 dB than the space-frequency LMS algorithm.

**Key words:** space-frequency anti-jamming; least mean square (LMS); tongue-like curve function; convergence speed; anti interference performance

## 引 言

卫星信号落地电平约为 $-133$  dBm,极易受到大功率干扰的影响而使导航接收机无法捕获和跟踪卫星信号<sup>[1]</sup>。空频自适应处理(Space frequency adaptive processing, SFAP)算法在空域与频域上同时进行自适应滤波,比单一维度的抗干扰算法性能更好,因此被广泛地应用在卫星导航抗干扰领域中<sup>[2]</sup>。

空频抗干扰算法的最优准则是线性约束最小方差准则(Linearly constraint minimum variance, LCMV)<sup>[3]</sup>,即阵列输出功率最小为最优,但是为了避免阵列输出功率最小为零,在已知卫星信号来向的条件下,可以利用该来向对滤波权值进行约束,使阵列始终输出在卫星来向上的信号,同时也可以保证卫星信号不被削弱。而在未知卫星来向的条件下,一般采用广义旁瓣相消器(General sidelobe canceller, GSC)的结构使阵列始终输出参考阵元的信号,这种方法被称为功率倒置(Power inverse, PI)算法<sup>[4]</sup>。PI算法只在干扰来向上形成零陷,并不会同时在卫星来向上形成固定增益,因此PI算法又被称为调零算法。

PI算法的最优滤波权值可以通过最小均方(Least mean square, LMS)算法迭代计算得出<sup>[5]</sup>,简称空频LMS算法。但是,LMS算法的性能受到迭代步长因子 $\mu$ 的制约,大步长下LMS算法的收敛速度快,但是收敛后稳态失调大,卫导接收机会出现掉星无法重捕的情况;小步长下LMS算法收敛后稳态失调小,但是收敛速度却慢,卫导接收机会出现掉星后经过较长的时间才能重捕的情况,即收敛速度快与稳态失调小这两个性能不可同时兼得<sup>[6-7]</sup>。

本文通过引入变步长LMS算法以解决这一问题。文献[8]提出了一种基于变步长频域块LMS(Frequency block least mean square, FBLMS)的空频抗干扰算法,该算法对同一阵元后的不同频点的信号进行FBLMS自适应滤波处理,继而引入变步长FBLMS算法(即Shi算法<sup>[9]</sup>)进行改进,然后在多个阵元上重复相同的步骤以完成空频滤波处理,其特点是在计算误差时需要将频域滤波后的信号逆快速傅里叶变换(Inverse fast Fourier transform, IFFT)到时域上再与时域上的期望信号作差,再将误差快速傅里叶变换(Fast Fourier transform, FFT)到频域上更新频域BLMS算法的滤波权值。本文采用更直观的思路,即对同一频点上的不同阵元的信号进行空域LMS自适应滤波处理<sup>[6]</sup>,继而引入变步长LMS算法进行改进,然后在多个频点上重复相同的步骤以完成全部空频滤波处理,因此在计算误差时某一频点上滤波后的信号可以直接与该频点的期望信号作差。

目前,典型的变步长LMS算法有基于Sigmoid函数的可变步长LMS算法(Variable step LMS of sigmoid function, SVSLMS)<sup>[10]</sup>及其改进型<sup>[11-13]</sup>、基于双曲正切函数的可变步长LMS算法(Variable step LMS of hyperbolic tangent function, HTVSLMS)<sup>[14]</sup>和基于箕舌线的可变步长LMS算法(Variable step LMS of tongue-like curve function, TLCVSLMS)<sup>[15-16]</sup>,以及基于凸组合的LMS(Combination of LMS, CLMS)算法<sup>[17]</sup>。因为TLCVSLMS算法相较于SVSLMS算法及其改进型与HTVSLMS算法在性能接近的前提下避免了指数运算,计算量更小<sup>[18]</sup>,因此本文选择利用TLCVSLMS算法改进空频LMS算法,简称空频TLCVSLMS算法。

空频TLCVSLMS算法的主要思想如下:在信号的每一个频点上,由于迭代初期误差较大,收敛至稳态后误差较小,因此在利用误差信号表征迭代阶段的基础上,通过建立步长因子 $\mu$ 与误差之间的箕舌线函数关系,使得LMS算法在迭代初期采用较大的步长从而快速收敛,而在收敛到稳态时采用较小的步长以保持较小的稳态失调。显然,空频TLCVSLMS算法可以避免针对每一个频点人为地选取合适的固定步长因子 $\mu$ 的困难。同时,根据不同频点的信号功率,空频TLCVSLMS算法对箕舌线函数的幅度因子与形状因子作更精细的调节,以增强算法的稳定性。仿真实验表明,在抗干扰性能接近的情况

下,空频 TLCVSLMS 算法比空频 LMS 算法少迭代至少 400 点,空频 TLCVSLMS 算法的收敛速度更快,而在收敛速度相同的情况下,空频 TLCVSLMS 算法比空频 LMS 算法的抗干扰性能提升至少 3 dB 以上,空频 TLCVSLMS 算法比空频 LMS 算法更有效地兼顾了收敛速度与抗干扰性能。

### 1 空频 LMS 算法

SFAP(简称空频滤波)的流程简述如下:首先对输入的宽带信号进行 FFT,从而将宽带信号分解成不同的频谱分量,然后对每一个频谱分量进行空域滤波处理,所有频谱分量上的单拍次空域滤波结构如图 1 所示。

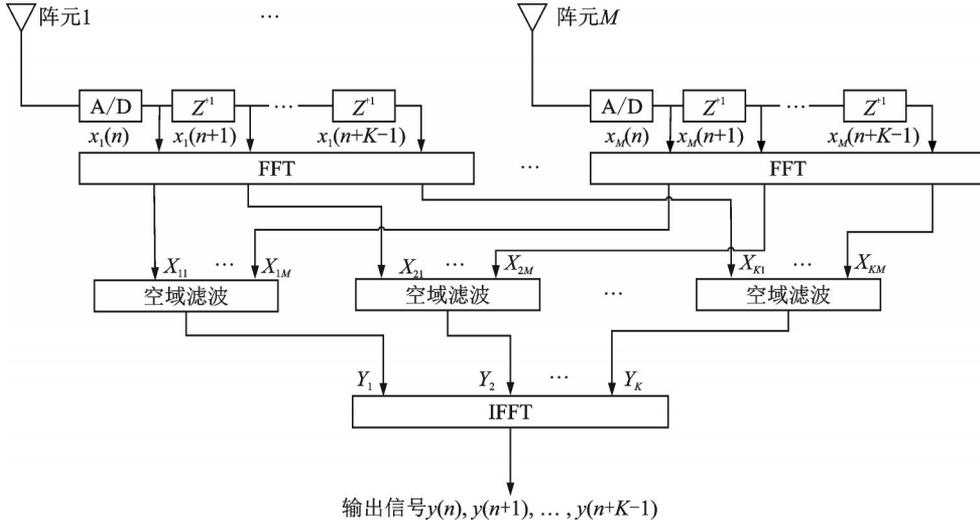


图 1 所有频谱分量上的单拍次空域滤波结构示意图

Fig.1 Schematic diagram of single shot spatial filtering structure on all spectrums

假设图 1 中所示的阵列天线由  $M$  个阵元组成,每个阵元后有  $K$  个延迟节点,则第  $n$  个时刻任意第  $m$  个阵元后的  $K$  点信号可以表示为

$$\mathbf{x}_m = [x_m(n), x_m(n+1), \dots, x_m(n+K-1)]^T \tag{1}$$

式中  $\mathbf{x}$  表示时域数据样本。对时域信号做  $K$  点 FFT 变换,可以得到频域信号为

$$\mathbf{X}_m = [X_m(f_1), X_m(f_2), \dots, X_m(f_k)]^T \tag{2}$$

式中  $\mathbf{X}$  为频域数据(以下同)。SFAP 算法是单独对每一个子频带信号进行滤波权值计算,从而将宽带信号滤波转变为窄带信号滤波,所以将同一个频带的数据放到一起组成子频带信号模型,则第  $k$  个子频带的空域信号为

$$\mathbf{X}'_k = [X_1(f_k), X_2(f_k), \dots, X_M(f_k)]^T \tag{3}$$

为了简便,省略频率符号  $f_k$ ,将频率  $k$  移至右下角标,则式(3)变为

$$\mathbf{X}'_k = [X_{k1}, X_{k2}, \dots, X_{kM}]^T \tag{4}$$

则频域内第  $k$  个子频带的滤波结果为<sup>[6]</sup>

$$Y_k = \mathbf{w}_{k, \text{opt}}^H \mathbf{X}'_k \tag{5}$$

式中  $\mathbf{w}_{k, \text{opt}}$  为第  $k$  个频谱分量上的最优滤波权值,有

$$\mathbf{w}_{k, \text{opt}} = [\omega_{k1, \text{opt}}, \omega_{k2, \text{opt}}, \dots, \omega_{kM, \text{opt}}]^T \tag{6}$$

对全部  $K$  个子频带滤波后,结果为

$$\mathbf{Y} = [Y_1, Y_2, \dots, Y_K]^T \tag{7}$$

对  $\mathbf{Y}$  进行 IFFT,可得最后的时域输出结果为

$$\mathbf{y} = [y(n), y(n+1), \dots, y(n+K-1)]^T \quad (8)$$

利用LMS算法第 $k$ 个频谱分量上的最优滤波权值 $\mathbf{w}_{k, \text{opt}}$ 。选择第1个阵元作为参考阵元,即

$$\tau_{k1, \text{opt}} = 1 \quad (9)$$

然后对其他阵元输入信号的最优滤波权值进行迭代求解。上文及图1所述的是所有频谱分量上的单拍次空域滤波,如要迭代求解最优权值,则需要进行多拍次空域滤波。假设总共有 $L$ 拍数据,则每个频点上有 $J$ 拍数据,有 $J=L/K$ ,于是第 $j+1$ 拍数据与第 $j$ 拍数据之间的滤波权值迭代公式为<sup>[6]</sup>

$$\mathbf{w}'_k(j+1) = \mathbf{w}'_k(j) + \mu \mathbf{X}_k'(j) e^*(j) \quad (10)$$

式中:“\*”表示共轭; $\mathbf{w}'_k(j)$ 为非参考阵元的输入信号的滤波权值,有

$$\mathbf{w}'_k(j) = [\tau_{k2}(j), \tau_{k3}(j), \dots, \tau_{kM}(j)]^T \quad (11)$$

$\mathbf{X}_k''(j)$ 为非参考阵元的输入信号,有

$$\mathbf{X}_k''(j) = [X_{k2}(j), X_{k3}(j), \dots, X_{kM}(j)]^T \quad (12)$$

式(9)迭代收敛后得到 $\mathbf{w}'_{k, \text{opt}}$ ,则 $\mathbf{w}_{k, \text{opt}}$ 为<sup>[6]</sup>

$$\mathbf{w}_{k, \text{opt}} = [1, -\mathbf{w}'_{k, \text{opt}}]^T \quad (13)$$

将式(13)代入式(5)后得到最终滤波后的数据。

式(9)中 $\mu$ 为常数,称为步长因子,为了保证LMS算法收敛, $\mu$ 满足<sup>[6]</sup>

$$0 < \mu < \frac{2}{\text{tr}(\mathbf{R}_k(j))} \quad (14)$$

式中: $\mathbf{R}_k(j)$ 为第 $j$ 拍第 $k$ 个频谱分量的所有非参考阵元信号的自相关矩阵; $\text{tr}(\mathbf{R}_k(j))$ 为 $\mathbf{R}_k(j)$ 的迹,有

$$\mathbf{R}_k(j) = \mathbf{X}_k''(j) \mathbf{X}_k''^H(j) \quad (15)$$

式(10)中 $e(j)$ 为第 $j$ 拍误差信号,其表达式为<sup>[6]</sup>

$$e(j) = d(j) - \mathbf{w}'_k^H(j) \mathbf{X}_k''(j) \quad (16)$$

式中 $d(j)$ 为第 $j$ 拍数据的期望信号,是参考阵元的输出信号,即

$$d(j) = X_{k1}(j) \quad (17)$$

在进行迭代前,没有先验知识的情况下,滤波权值初始值置为零,即

$$\mathbf{w}'_k(1) = \underbrace{[0, 0, \dots, 0]}_{(M-1)\uparrow}^T \quad (18)$$

在实际应用中,为了减少输入信号对LMS算法性能的影响,方便步长因子的选择,常常将步长因子归一化最小均方(Normalized least mean squares, NLMS)<sup>[19-21]</sup>,此时, $\mu$ 由常数变为 $\mu(j)$ ,即

$$\mu(j) = \mu_0 / (\gamma + \mathbf{X}_k^H(j) \mathbf{X}_k(j)) \quad (19)$$

式中 $\gamma$ 为很小的实数,是为了防止输入信号功率很小时步长因子 $\mu(j)$ 过大而引入的,对算法性能没有影响。

对于NLMS算法, $\mu_0$ 大则迭代滤波的收敛速度快,但是稳态误差大, $\mu_0$ 小则迭代速度慢,稳态误差小。算法的收敛条件变为

$$0 < \mu_0 < 2 \quad (20)$$

## 2 空频TLCVSLMS算法

固定步长LMS算法中,收敛速度与稳态失调是一对矛盾,可以通过引入可变步长LMS算法予以缓解,即对于式(18)中的 $\mu_0$ ,引入TLCVSLMS, $\mu_0$ 变为 $\mu_0(j)$ ,有<sup>[15]</sup>

$$\mu_0(j) = \beta \left( 1 - \frac{1}{\alpha \|e(j)\|^2 + 1} \right) \quad (21)$$

式中:  $\|\cdot\|$ 表示二范数; $\beta$ 和 $\alpha$ 为常数,分别称为幅度因子与形状因子。 $\beta$ 决定了 $\mu_0(j)$ 的最大值, $\beta$ 越大,误差 $e(j)$ 可以对应到更大的步长 $\mu_0(j)$ 上,TLCVSLMS算法的收敛速度越快。而在 $\beta$ 值较大时,较小的误差 $e(j)$ 也会对应较大的步长 $\mu_0(j)$ ,因此算法稳态误差也会增大。同时,过大的 $\beta$ 也会导致算法发散,即 $\beta$ 需满足收敛条件

$$0 < \beta < 2 \tag{22}$$

$\alpha$ 决定了箕舌线函数的形状, $\alpha$ 越大,箕舌线函数开口越小,此时在误差 $e(j)$ 较大时更有可能对应到较大的步长 $\mu_0(j)$ ,TLCVSLMS算法的收敛速度也就越快,但是过小的开口下误差 $e(j)$ 较小时也会对应较大的步长 $\mu_0(j)$ ,算法的稳态误差也会增大。幅度因子 $\beta$ 和形状因子 $\alpha$ 与误差信号 $e(j)$ 关系如图2、3所示<sup>[15]</sup>。

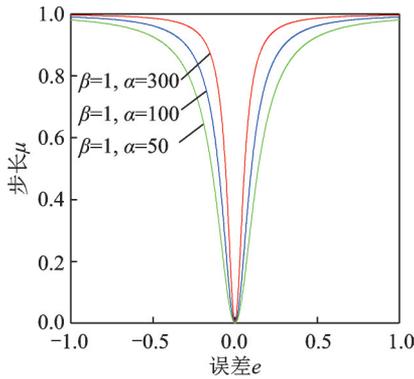


图2  $\beta$ 不变时 $\alpha$ 的影响<sup>[15]</sup>

Fig.2 Impact of  $\alpha$  with constant  $\beta$ <sup>[15]</sup>

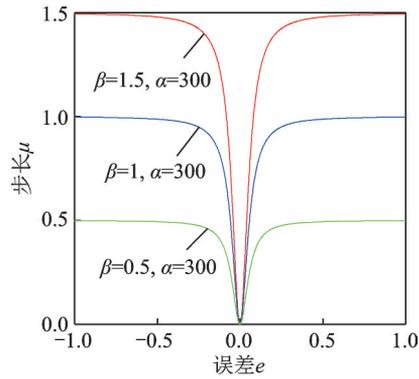


图3  $\alpha$ 不变时 $\beta$ 的影响<sup>[15]</sup>

Fig.3 Impact of  $\beta$  with constant  $\alpha$ <sup>[15]</sup>

将TLCVSLMS应用到空频抗干扰中时,对于大功率的频谱分量,选择小的 $\beta$ 与 $\alpha$ 以减小滤波器的稳定误差,否则即便是已收敛至稳态的滤波权值 $w'_{k,opt}$ 也无法产生足够深的零陷,所以可以牺牲一些收敛速度。而对于小功率的频谱分量,选择大的 $\beta$ 与 $\alpha$ 加快滤波器的收敛速度。

以每频谱分量上的平均功率作为阈值,当前频谱分量上的信号功率大于功率阈值时,选择较小的 $\beta$ 值和 $\alpha$ 值,而当前频谱分量上的信号功率小于功率阈值时,则选择较大的 $\beta$ 值和 $\alpha$ 值,即有

$$P_k = \frac{1}{J} \sum_{j=1}^J \text{tr}(R_k(j)) \tag{23}$$

$$\bar{P} = \frac{1}{K} \sum_{k=1}^K P_k \tag{24}$$

$$\begin{cases} \beta_k = \beta_1, \alpha_k = \alpha_1 & P_k < \bar{P} \\ \beta_k = \beta_2, \alpha_k = \alpha_2 & P_k > \bar{P} \end{cases} \tag{25}$$

式中: $0 < \beta_2 < \beta_1 < 2, 0 < \alpha_2 < \alpha_1$ 。

由于需要多拍数据LMS算法计算的滤波权值才会收敛,实际应用时可以设置每 $L$ 拍(每频点拍数 $J$ 大于权值收敛所需要的拍数)更新一次最终的最佳滤波权值,即虽然 $w'_k$ 每拍更新一次,但 $w_{k,opt}$ 是每 $L$ 拍更新一次,因此空频TLCVSLMS抗干扰算法具备块自适应滤波的特点。综上,对于每 $L$ 拍训练数据,本文所提的空频TLCVSLMS抗干扰算法流程如图4所示。

为了能够清晰地展现LMS算法<sup>[7]</sup>、TLCVSLMS算法<sup>[15]</sup>、空频LMS抗干扰算法与空频TLCVSLMS抗干扰算法之间的区别与联系,下面将详细地归纳总结这4种算法的主要步骤,其中LMS算法与基于箕舌线的变步长LMS算法如下:

(1) 初始化第一组滤波权值: $w(1) = \underbrace{[0, 0, \dots, 0]}_{M\uparrow}^T$ ,并获得相应的时域滤波数据 $x(1) =$

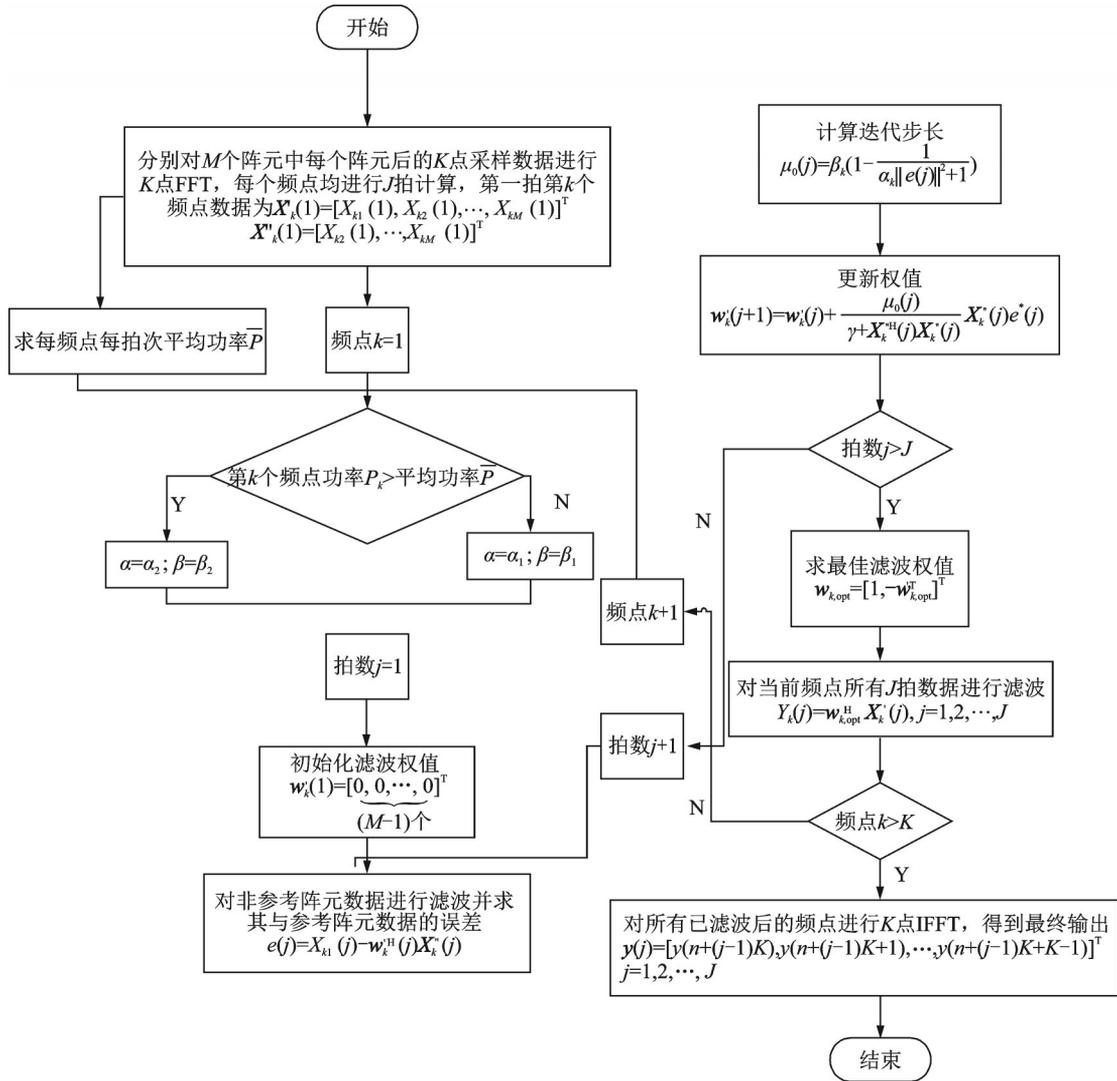


图4 空频 TLCVSLMS 抗干扰算法流程图

Fig.4 Flow chart of space-frequency TLCVSLMS anti-jamming algorithm

$[x(1), x(2), \dots, x(M)]^T$ , 其中  $M$  为滤波器阶数;

(2) 进行滤波: 则滤波输出的结果为  $y(1) = \mathbf{w}(1)^H \mathbf{x}(1)$ ;

(3) 计算滤波输出与期望信号之间的误差:  $e(1) = d(1) - y(1)$ , 其中期望信号  $d$  需已知;

(4) 更新权值: 对于 LMS 算法<sup>[7]</sup>, 其权值更新公式为  $\mathbf{w}(2) = \mathbf{w}(1) + \mu e(1) \mathbf{x}(1)$ , 步长  $\mu$  固定, 而对于 TLCVSLMS 算法<sup>[15]</sup>, 步长  $\mu$  与误差之间建立了算舌线关系, 其权值更新公式为  $\mu(1) = \beta(1 - \frac{1}{\alpha \|e(1)\|^2 + 1})$ ,  $\mathbf{w}(2) = \mathbf{w}(1) + \mu(1)e(1)\mathbf{x}(1)$ ;

(5) 重复步骤(1~5), 直至进行  $J$  次迭代。

空频 LMS 抗干扰算法与空频 TLCVSLMS 抗干扰算法的主要步骤如下:

(1) 初始化第  $k$  频点第一拍滤波权值:  $\mathbf{w}_k(1) = [0, 0, \dots, 0]^T$ , 获取第  $k$  个频点第一拍不同阵元频域  $(M-1)$  个

数据  $\mathbf{X}'_k(1)=[X_{k1}(1), X_{k2}(1), \dots, X_{kM}(1)]^T$ , 则需要更新权值的数据为  $\mathbf{X}''_k(1)=[X_{k2}(1), X_{k3}(1), \dots, X_{kM}(1)]^T$ , 其中  $M$  为阵元个数;

(2) 进行滤波: 则滤波输出的结果为  $Y'_k = \mathbf{w}'_k{}^H(1)\mathbf{X}'_k(1)$ ;

(3) 计算滤波输出与期望信号之间的误差:  $e(1) = X_{k1}(1) - Y'_k(1)$ , 此处为第一个阵元作为参考阵元, 其信号作为期望信号;

(4) 更新权值: 对于空频 LMS 抗干扰算法, 其权值更新公式为  $\mathbf{w}'_k(2) = \mathbf{w}'_k(1) + \mu\mathbf{X}''_k(1)e^*(1)$ , 而对于本文所提的空频 TLCVSLMS 抗干扰算法, 权值更新公式为  $\mu_0(1) = \beta_k \left( 1 - \frac{1}{\alpha_k \|e(1)\|^2 + 1} \right)$ ,  $\mathbf{w}_k(2) =$

$$\mathbf{w}_k(1) + \frac{\mu_0(1)}{\gamma + \mathbf{X}_k{}^H(1)\mathbf{X}_k(1)} \mathbf{X}''_k(1)e^*(1);$$

(5) 重复步骤(1~4), 直至进行  $J$  次迭代;

(6) 获取最终滤波权值:  $\mathbf{w}_{k, \text{opt}} = [1, -\mathbf{w}'_{k, \text{opt}}{}^T]^T$ , 其效果是将非参考阵元的信号滤波后与参考阵元信号对消, 使整体阵列输出信号功率尽可能得小, 从而起到消除大功率干扰的作用;

(7) 使用最终滤波权值对相同频点下的每一拍数据再次进行滤波:  $Y_k(j) = \mathbf{w}_{k, \text{opt}}^H \mathbf{X}'_k(j)$ , 其中  $j=1, 2, \dots, J$ ;

(8) 对每一段滤波输出后的频域信号进行  $K$  点 IFFT, 得到时域信号  $\mathbf{y}(j) = [y(n+(j-1)K), y(n+(j-1)K+1), \dots, y(n+(j-1)K+K-1)]^T$ , 其中  $j=1, 2, \dots, J$ 。

综上, 4种算法的区别在于: LMS 算法与 TLCVSLMS 算法是在时域上进行滤波, 而空频 LMS 抗干扰算法与空频 TLCVSLMS 抗干扰算法则是在每一个频点上对不同阵元的数据(即空域)进行滤波; LMS 算法与空频 LMS 抗干扰算法迭代时步长固定, 而 TLCVSLMS 算法与空频 TLCVSLMS 抗干扰算法迭代时步长随误差的改变而改变。联系在于: 空频 LMS 抗干扰算法与空频 TLCVSLMS 抗干扰算法可以看作是空频 PI 算法分别使用 LMS 算法与 TLCVSLMS 算法的实现, 而 TLCVSLMS 算法是针对 LMS 算法的变步长改进, 因此空频 TLCVSLMS 抗干扰算法也是针对空频 LMS 抗干扰算法的变步长改进。

### 3 性能分析

#### 3.1 仿真条件

通过仿真实验验证空频 TLCVSLMS 算法相较于空频 LMS 算法的性能改进。仿真条件设置如下: (1) 阵列天线为四阵元圆阵, 以圆心为原点, 任取一个阵元作为第 1 个阵元, 顺时针排序其他阵元编号。以第 1 个阵元与圆心的连线作为  $X$  轴建立平面直角坐标系。(2) 阵列天线接收信号高频信号通过下变频与 AD 采样转换为数字中频信号。仿真中频信号的中心频率为 18 MHz, 采样率为 68 MHz, 信噪比为 -20 dB。取当前数据 0.01 s 后的所有采样点作为训练数据迭代计算滤波权值, 此时拍数  $L$  为 680 000 次。(3) 空频滤波中, 做 64 点 FFT 将时域信号变换到频域, 各个频点的拍次  $J$  为 10 625。最后, 绘制相同频谱分量的空域方向图, 通过观察抗干扰处理后方向图中的零陷深度来判断抗干扰性能优劣, 即在相同干扰强度的条件下, 零陷越深, 抗干扰性能越好。统计每频谱分量上的平均收敛点数作为算法收敛速度。在空频 TLCVSLMS 算法和空频 LMS 算法中, 常数  $\gamma = 0.0001$ 。

#### 3.2 单干扰场景

在(俯仰角, 方位角)为  $(10^\circ, 60^\circ)$  的方向上设置一个线性调频信号作为宽带干扰, 干信比为 90 dB, 带宽为 20 MHz。设置空频 TLCVSLMS 算法中的参数为  $\beta_1 = 1.2, \beta_2 = 1, \alpha_1 = 300, \alpha_2 = 50$ , 其抗干扰结果如图 5 所示, 收敛速度如图 6 所示。从图 5, 6 的结果中可以看出, 空频 TLCVSLMS 算法在 23 点左右收敛到最优权值, 同时在(俯仰角, 方位角)为  $(10^\circ, 60^\circ)$  的方向上产生了一 90.96 dB 的一个零陷, 准确

地对准了干扰来向,因此可以有效地抗击在该方向上的大功率干扰。

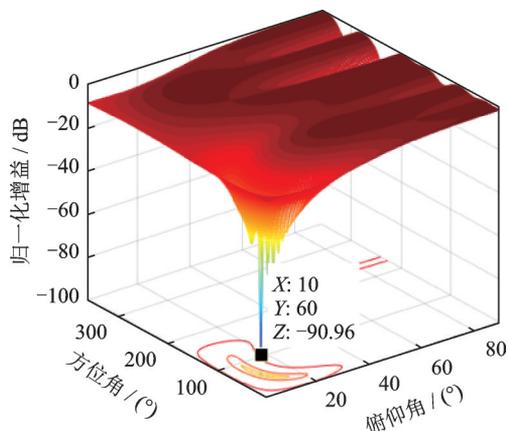


图5 空频 TLCVSLMS算法抗单干扰方向图

Fig.5 Anti single interference pattern of space-frequency TLCVSLMS

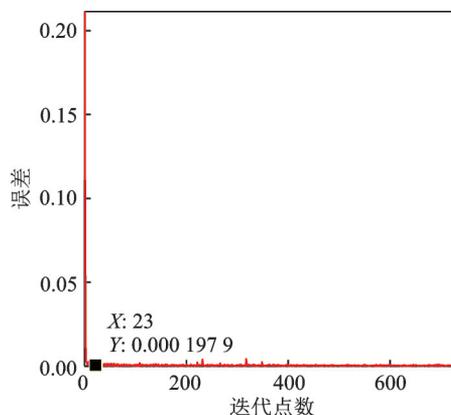


图6 空频 TLCVSLMS算法抗单干扰收敛速度

Fig.6 Anti single interference convergence of space-frequency TLCVSLMS

调整空频 LMS 算法与空频 TLCVSLMS 算法的抗干扰性能接近,此时空频 LMS 算法中  $\mu_0 = 0.003$ ,比较两者的收敛速度。空频 LMS 算法的抗干扰结果如图 7 所示,收敛速度如图 8 所示。

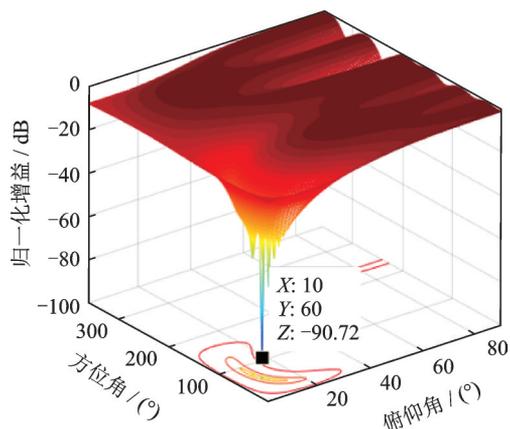


图7 抗单干扰性能接近时空频 LMS 算法方向图

Fig.7 Pattern of space-frequency LMS algorithm with close anti single interference performance

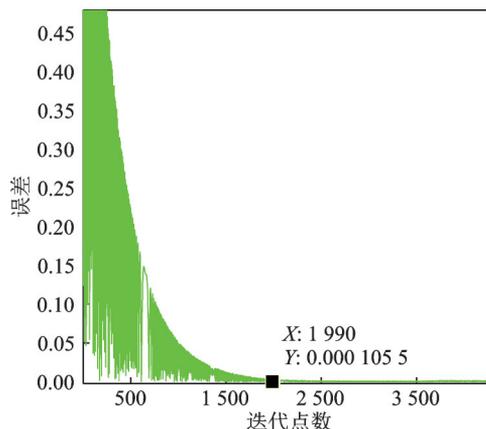


图8 抗单干扰性能接近时空频 LMS 算法收敛速度

Fig.8 Convergence speed of space-frequency LMS algorithm with close anti single interference performance

从图 5 与图 7、图 6 与图 8 的结果比较中可以看到,在抗干扰性能接近的前提下,空频 LMS 算法迭代约 1 990 点才能收敛到最优权值,而空频 TLCVSLMS 算法只需约 23 点,所以在单干扰场景下,空频 TLCVSLMS 算法比空频 LMS 少迭代 1 967 点进入稳态,空频 TLCVSLMS 算法的收敛速度更快。

调整空频 LMS 算法与空频 TLCVSLMS 算法具备相近的收敛速度,此时空频 LMS 算法中  $\mu_0 = 0.5$ ,比较两者的抗干扰性能。空频 LMS 算法的抗干扰结果如图 9 所示,其收敛速度如图 10 所示。

从图 5 与图 6、图 9 与图 10 的结果比较中可以看出,在保证空频 LMS 算法收敛速度与空频 TLCVSLMS 算法的收敛速度接近的前提下,空频 LMS 算法产生了一 63.78 dB 的一个零陷,而此时空频 TLCVSLMS 算法产生了一 90.96 dB 的一个零陷,所以在单干扰场景下,空频 TLCVSLMS 算法的抗干扰性能比空频 LMS 算法好 20 dB 以上,空频 TLCVSLMS 算法具备更优的抗干扰性能。

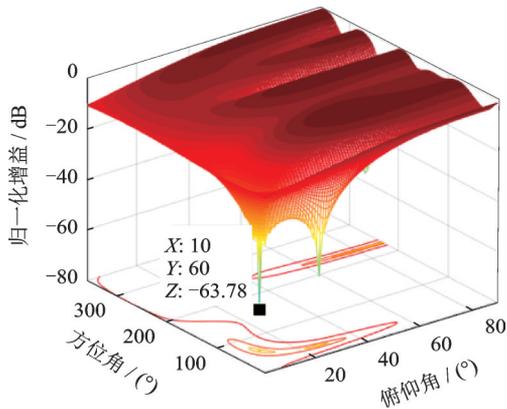


图9 收敛速度一致时空频LMS算法单干扰方向图  
Fig.9 Anti single interference pattern of space-frequency LMS algorithm with consistent convergence speed

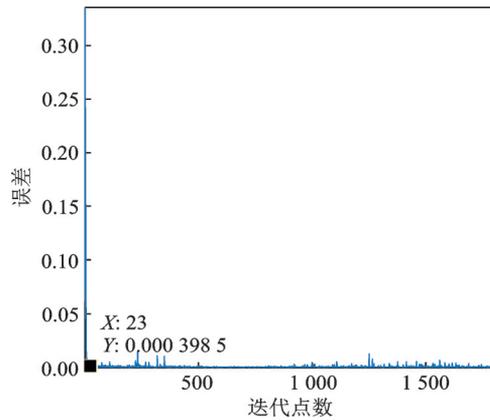


图10 收敛速度一致时空频LMS算法单干扰收敛速度  
Fig.10 Anti single interference convergence speed of space-frequency LMS algorithm with consistent convergence speed

### 3.3 三干扰场景

由于是四阵元圆阵,最多可以抗击3个宽带干扰,所以仿真设置三干扰场景,即在(俯仰角,方位角)分别为(10°,60°)、(10°,160°)和(10°,250°)的方向上设置3个线性调频信号作为宽带干扰,干信比均为90 dB,带宽均为20 MHz。空频TLCVSLMS算法的参数与单干扰场景时一致,其抗干扰结果如图11所示,收敛速度如图12所示。

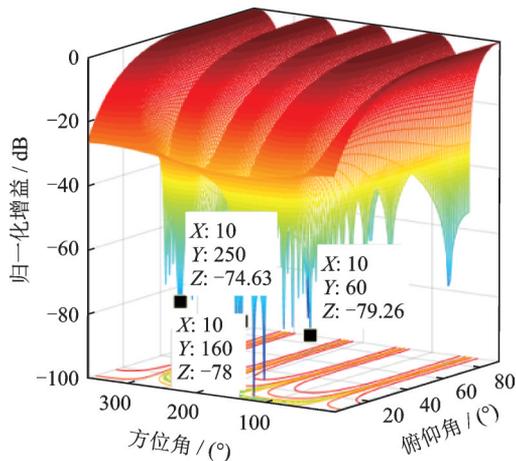


图11 空频TLCVSLMS算法三干扰方向图  
Fig.11 Anti three interference pattern of space-frequency TLCVSLMS algorithm

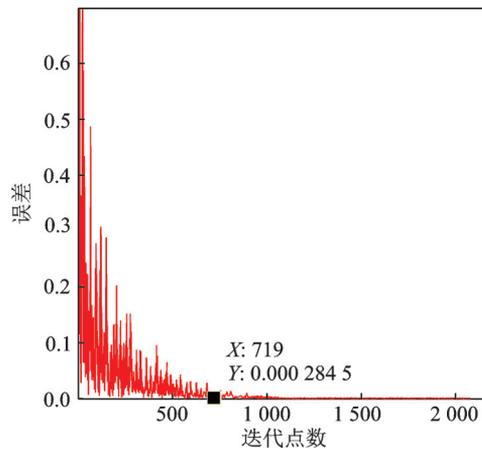


图12 空频TLCVSLMS算法三干扰收敛速度  
Fig.12 Anti three interference convergence speed of space-frequency TLCVSLMS algorithm

从结果可以看出,空频TLCVSLMS算法在719点左右收敛到最优权值,相较于单干扰场景下收敛速度变慢。同时,空频TLCVSLMS算法在(俯仰角,方位角)分别为(10°,60°)、(10°,160°)和(10°,250°)的方向上产生了-79.26 dB、-78 dB、-74.63 dB的3个零陷,均准确地对准了3个干扰的来向,因此可以有效地抗击在该方向上的大功率干扰。与此同时,相较于单干扰场景,三干扰场景下空频TLCVSLMS算法在非干扰来向上产生了更多的伪零陷,这些零陷会影响到卫星信号的接收,出现该现象的原因是此时宽带干扰的数目等于阵列天线的自由度。四阵元天线的自由度为3,即可以独立操控3个方向,随着宽带干扰数目的增大,方向图中的伪零陷数目就会增多,而当宽带干扰数目超过阵列天线

自由度时,方向图中的零陷将不再正确地对准干扰的来向,阵列天线丧失抗干扰能力。

调整空频LMS算法与空频TLCVSLMS算法的抗干扰性能接近,此时空频LMS算法中 $\mu_0 = 0.08$ ,比较两者的收敛速度。空频LMS算法的抗干扰结果如图13所示,收敛速度如图14所示。

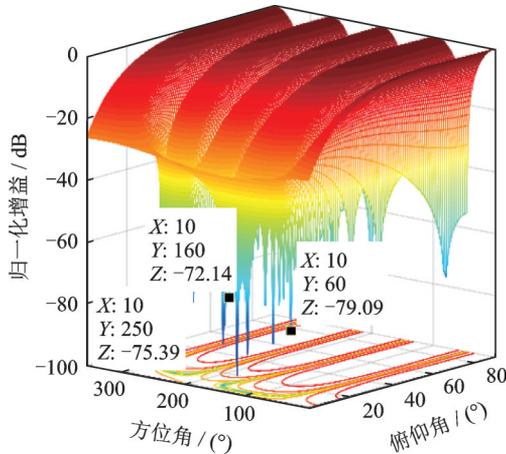


图13 抗三干扰性能接近时空频LMS算法方向图  
Fig.13 Pattern of space-frequency LMS algorithm with close anti three interference performance

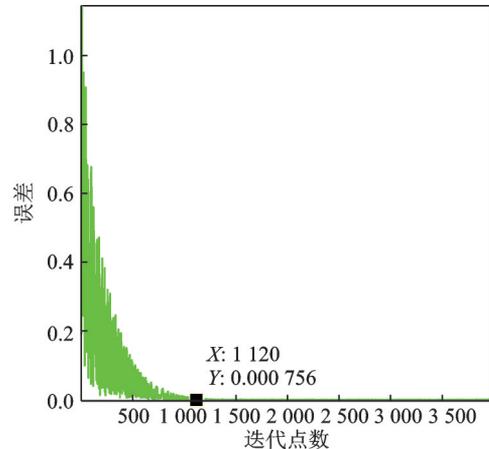


图14 抗三干扰性能接近时空频LMS算法收敛速度  
Fig.14 Convergence speed of space-frequency LMS algorithm with close anti three interference performance

从图11与图12、图13与图14的结果中可以看到,在抗干扰性能接近的前提下,空频LMS算法迭代约1120点收敛到最优权值,而空频TLCVSLMS算法需约719点。因此,在三干扰场景下,空频TLCVSLMS算法比空频LMS算法少迭代400点左右进入稳态,空频TLCVSLMS算法的收敛速度好于空频LMS算法。

调整空频LMS算法与空频TLCVSLMS算法具备相近的收敛速度,此时空频LMS算法中 $\mu_0 = 0.15$ ,比较两者的抗干扰性能。空频LMS算法的抗干扰结果如图15所示,其收敛速度如图16所示。

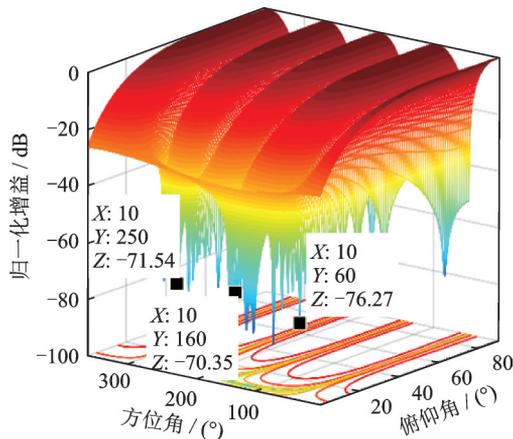


图15 收敛速度一致时空频LMS算法三干扰方向图  
Fig.15 Anti three interference pattern of space-frequency LMS algorithm with consistent convergence speed

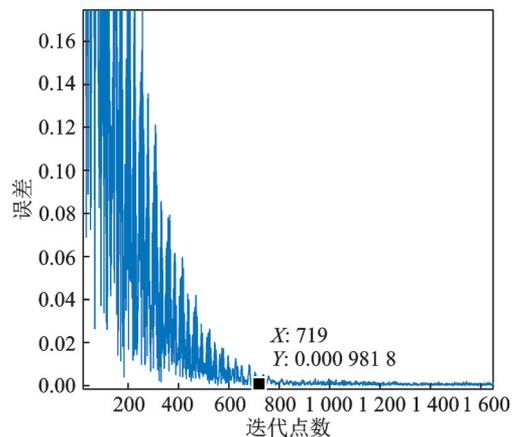


图16 收敛速度一致时空频LMS算法三干扰收敛速度  
Fig.16 Convergence speed of space-frequency LMS algorithm with close anti three interference performance

从图 11 与图 15、图 12 与图 16 的结果比较中可以得出,在空频 LMS 算法收敛速度与空频 TLCVSLMS 算法接近的前提下(两者均在 719 点左右收敛至稳态),空频 LMS 算法同时在(俯仰角,方位角)分别为 $(10^\circ, 60^\circ)$ 、 $(10^\circ, 160^\circ)$ 和 $(10^\circ, 250^\circ)$ 的方向上产生了一76.27 dB、-70.35 dB、-71.54 dB 的 3 个零陷,而空频 TLCVSLMS 算法则产生了一79.26 dB、-78 dB、-74.63 dB 的 3 个零陷。因此,在三干扰场景下,空频 TLCVSLMS 算法的抗干扰性能比空频 LMS 算法至少高 3 dB 左右,空频 TLCVSLMS 算法的抗干扰性能较优于空频 LMS 算法。

#### 4 结束语

本文提出一种空频 TLCVSLMS 算法,以解决空频 LMS 算法的抗干扰性能与收敛速度无法兼顾的问题。仿真实验表明,在抗干扰性能接近的情况下,空频 TLCVSLMS 算法比空频 LMS 算法少迭代至少 400 点,空频 TLCVSLMS 算法的收敛速度更快,而在收敛速度相同的情况下,空频 TLCVSLMS 算法比空频 LMS 算法的抗干扰性能提升至少 3 dB 以上。空频 TLCVSLMS 算法更有效地兼顾了抗干扰性能和收敛速度,这一优点在单干扰场景比在三干扰场景中表现得更为明显。同时由于箕舌线函数相较于其他变步长 LMS 算法所引进的函数而言没有指数运算项,因此空频 TLCVSLMS 算法的计算过程更简单,计算量更小,具有一定的工程应用价值。除此之外,本文所提的空频 TLCVSLMS 还可以结合对角加载算法来进一步提升抗干扰性能的稳健性。

#### 参考文献:

- [1] 尹继东,冯雪枫.GNSS 抗干扰技术综述[J].电讯技术,2020,60(8):989-998.  
YIN Jidong, FENG Xuefeng. Overview of GNSS anti-interference technology[J]. Telecommunication Engineering, 2020, 60(8): 989-998.
- [2] 尚亚锐,刘雪梅,王泽.基于空频自适应算法的抗干扰天线设计[J].电子设计工程,2020,16(28):142-145.  
SHANG Yarui, LIU Xuemei, WANG Ze. Anti-interference antenna design based on space frequency adaptive algorithm[J]. Electronic Design Engineering, 2020, 16(28): 142-145.
- [3] 田玉坤.卫星导航自适应调零抗干扰算法研究与实现[D].西安:西安电子科技大学,2017.  
TIAN Yukun. Research and realization of adaptive nulling anti-jamming algorithm for satellite navigation[D]. Xi'an: Xidian University, 2017.
- [4] 曹建波.GPS 接收机降维空时联合处理算法研究[J].现代导航,2014(3):162-166.  
CAO Jianbo. Research on algorithm of reduced-dimension joint space-time processing for GPS receiver[J]. Modern Navigation, 2014(3): 162-166.
- [5] 鲁祖坤.卫星导航天线阵列抗干扰关键技术研究[D].长沙:国防科技大学,2018.  
LU Zukun. Research on key technology of anti-jamming of satellite navigation antenna arrays[D]. Changsha: National University of Defense Technology, 2018.
- [6] 周正.导航系统中空频自适应算法的研究和实现[D].成都:电子科技大学,2016.  
ZHOU Zheng. Research and implementation of the space and frequency adaptive processing algorithm for the navigation system [D]. Chengdu: University of Electronic Science and Technology of China, 2016.
- [7] 李宁.LMS 自适应滤波算法的收敛性能研究与应用[D].哈尔滨:哈尔滨工程大学,2009.  
LI Ning. Convergence performance analysis and applications of the adaptive least mean square(LMS) algorithm[D]. Harbin: Harbin Engineering University, 2009.
- [8] 万大伟.GNSS 空频抗干扰技术研究[D].西安:西安理工大学,2016.  
WAN Dawei. Research on anti-interference technology of GNSS space frequency[D]. Xi'an: Xi'an University of Technology, 2016.
- [9] KUN Shi. A frequency domain step-size control method for LMS algorithms[J]. Signal Processing Magazine, 2010, 17(2): 125-128.

- [10] 覃景繁, 欧阳景正. 一种新的变步长自适应滤波算法[J]. 数据采集与处理, 1997, 12(3): 171-194.  
TAN Jingfan, OUYANG Jingzheng. A novel variable step size LMS adaptive filtering algorithm based on sigmoid function[J]. *Journal of Data Acquisition & Processing*, 1997, 12(3): 171-194.
- [11] 高鹰, 谢胜利. 一种变步长 LMS 自适应滤波算法及分析[J]. 电子学报, 2001, 29: 1094-1097.  
GAO Ying, XIE Shengli. A variable step size LMS adaptive filtering algorithm and its analysis[J]. *ACTA Electronica Sinica*, 2001, 29: 1094-1097.
- [12] 牛群, 陈天宁. 一种改进变步长 LMS 算法及其 DSP 功能实现[J]. 电子测量技术, 2018, 41(7): 83-88.  
NIU Qun, CHEN Tianning. An improved variable step size LMS algorithm and its DSP function realization[J]. *Electronic Measurement Technology*, 2018, 41(7): 83-88.
- [13] 陈泳, 田金鹏, 刘燕平. 一种新的变步长 LMS 自适应滤波算法[J]. 电子测量技术, 2015, 38(4): 27-31.  
CHEN Yong, TIAN Jinpeng, LIU Yanping. New variable step size LMS adaptive filtering algorithm[J]. *Electronic Measurement Technology*, 2015, 38(4): 27-31.
- [14] 钟慧湘, 郑莎莎, 冯月萍. 基于双曲正切函数的智能天线变步长 LMS 算法[J]. 吉林大学学报(理学版), 2008, 46(5): 935-939.  
ZHONG Huixiang, ZHENG Shasha, FENG Yueping. A variable step size LMS algorithm in smart antennas based on hyperbolic tangent function[J]. *Journal of Jilin University (Science Edition)*, 2008, 46(5): 935-939.
- [15] 邓江波, 候新国, 吴正国. 基于箕舌线的变步长 LMS 自适应算法[J]. 数据采集与处理, 2004, 19(3): 282-285.  
DENG Jiangbo, HOU Xinguo, WU Zhengguo. Variable step adaptive filtering LMS algorithm based on tongue-like curve[J]. *Journal of Data Acquisition & Processing*, 2004, 19(3): 282-285.
- [16] 韩允, 解传军, 刘宝华, 等. 基于箕舌线函数的变步长归一化最小均方算法[J]. 现代电子技术, 2008, 31(19): 29-32.  
HAN Yun, XIE Chuanjun, LIU Baohua, et al. Modified variable step-size normalized least-mean-square algorithm based on versoria function[J]. *Modern Electronic Technology*, 2008, 31(19): 29-31.
- [17] FAN Xinfeng, TAN Zhiliang, SONG Peijiao, et al. A variable step-size CLMS algorithm and its analysis[J]. *Radio Engineering*, 2021, 29(1): 182-188.
- [18] 胡春娇, 杨顺. 基于箕舌线变步长 LMS 算法的分析与改进[J]. 计算机仿真, 2010, 27(11): 359-362.  
HU Chunjiao, YANG Shun. Analysis and improvement of a variable step LMS algorithm based on correlation tongue-like curve [J]. *Computer Integrated Manufacturing Systems*, 2010, 27(11): 259-362.
- [19] 董李梅. 一种变步长自适应算法在 GPS 空时抗干扰中的应用[J]. 通信技术, 2015, 48(3): 295-301.  
DONG Limei. Application of NLMS algorithm in GPS space-time adaptive processing[J]. *Communications Technology*, 2015, 48(3): 295-301.
- [20] 郑宗生. 复杂环境下稳健自适应估计方法研究[D]. 成都: 西南交通大学, 2020.  
ZHENG Zongsheng. Robust adaptive estimation methods for complex environments[D]. Chengdu: Southwest Jiaotong University, 2020.
- [21] ALI ANUM M M, AI N, TAREQ Y. NLMS is more robust to input-correlation than LMS: A proof[J]. *IEEE Signal Processing Letters*, 2021, 29: 279-283.

## 作者简介:



郭辰锋(1994-), 通信作者, 男, 助理工程师, 研究方向: 阵列信号处理, E-mail: gcf123456gcf@163.com.



舒东亮(1982-), 男, 研究员, 研究方向: 综合导航与卫星导航技术。



路寅(1986-), 女, 高级工程师, 研究方向: 综合导航与卫星导航技术。



靳小琴(1993-), 女, 博士研究生, 研究方向: 卫星导航与组合导航。