

引用格式: 何羚, 郑怀洲, 阎啸, 等. 面向航天测控的高效自适应滤波算法及其加速实现 [J]. 电子科技大学学报, 2026, 55(1): 85-92.  
HE L, ZHENG H Z, YAN X, et al. An efficient adaptive filtering algorithm and its accelerated implementation for aerospace TT&C[J]. Journal of University of Electronic Science and Technology of China, 2026, 55(1): 85-92.

# 面向航天测控的高效自适应滤波算法 及其加速实现



何 羚<sup>1,2</sup>, 郑怀洲<sup>1</sup>, 阎 啸<sup>1,2\*</sup>, 王 茜<sup>1</sup>

(1. 电子科技大学航空航天学院, 成都 611731; 2. 飞行器集群智能感知与协同控制四川省重点实验室, 成都 611731)

**摘要:** 在航天测控中, 日益复杂的空间电磁环境对远距离星-地传输链路的可靠性和有效性形成威胁。当链路中存在多个特定来向的干扰时, 运用传统的自适应滤波功率倒置准则无法有效滤除压制式干扰以外的其他干扰信号。该文面向智能天线应用及地面测控站抗干扰接收场景, 提出一种多空域约束的功率倒置准则及自适应滤波算法硬件加速实现方案。在波束成形中, 通过对功率倒置准则添加零陷方向约束, 实现功率域和空间域的非期望信号抑制; 在实现时采用矩阵 LDL 分解完成低复杂度矩阵求逆, 同时通过定点-浮点转换提升计算精度, 进一步利用脉动阵列计算单元实现高效矩阵运算。搭建了测控信号接收系统, 将上述自适应滤波算法部署在硬件上进行实验验证, 结果表明: 所提方法对指定来向的带内干扰和任意来向的压制式带内干扰均能产生深零陷, 且对测控信号无失真影响; 计算得到的自适应滤波最优权值向量相对误差在  $10^{-6} \sim 10^{-7}$  量级。

**关键词:** 测控链路; 多空域约束的功率倒置准则; 矩阵求逆; LDL 分解; 脉动阵列

中图分类号: TN971

文献标志码: A

DOI: 10.12178/1001-0548.2024256

## An efficient adaptive filtering algorithm and its accelerated implementation for aerospace TT&C

HE Ling<sup>1,2</sup>, ZHENG Huaizhou<sup>1</sup>, YAN Xiao<sup>1,2\*</sup>, and WANG Qian<sup>1</sup>

(1. School of Aeronautics and Astronautics, University of Electronic Science and Technology of China, Chengdu 611731, China;

2. Aircraft Swarm Intelligent Sensing and Cooperative Control Key Laboratory of Sichuan Province, Chengdu 611731, China)

**Abstract:** In aerospace telemetry, tracking, and control (TT&C) systems, the increasingly complex space electromagnetic environment threatens the reliability and effectiveness of the satellite-earth communication links. When there are multiple interference signals arriving from specific directions in the communication link, the conventional beamforming power inversion criterion fails to effectively suppress interference signals other than suppressive jamming. Focusing on smart antenna applications and anti-interference reception scenarios in TT&C ground stations, this paper proposes a multi-spatial-domain constrained power inversion criterion and a hardware-accelerated implementation scheme for adaptive filtering algorithms. In beamforming, nulling direction constraints are incorporated into the power inversion criterion to eliminate undesired signals in both power and spatial domains. In implementation, LDL decomposition is adopted to achieve low-complexity matrix inversion; while fixed-point to floating-point conversion is applied to improve computational precision. Furthermore, systolic array units are leveraged to enable highly efficient matrix operations. For verification purpose, a TT&C receiver is constructed and the overall adaptive filtering algorithm mentioned above is deployed on hardware. Experimental results demonstrate that the proposed method generates deep nulls for both in-band interference from specified directions and suppressive in-band interference from arbitrary directions, introducing no distortion to TT&C signals; the relative error of the optimal weight vector for adaptive filtering is on the order of  $10^{-6} \sim 10^{-7}$  in magnitude.

**Key words:** TT&C link; multi-spatial-domain constrained power inversion criterion; matrix inversion; matrix LDL decomposition; systolic array

收稿日期: 2024-09-18

基金项目: 四川省自然科学基金面上项目 (2023NSFSC0494)

作者简介: 何羚, 副教授, 主要从事空间信息获取与处理方面的研究。

\*通信作者 E-mail: yanxiao@uestc.edu.cn

航天测控是对卫星等航天器的各个阶段进行跟踪、测量和控制的重要支撑技术。由于空间信道的开放性和长距离无线电传输的功率损耗, 航天测控链路容易受到地域、天气等客观因素的影响<sup>[1]</sup>。在大规模低轨卫星星座迅速崛起的当前, 囿于空间业务可用频率资源和轨道资源限制, 数目庞大的低轨卫星不可避免地居于中高轨航天器的星-地测控链路上, 对同频段的中高轨航天器地面站形成多个干扰<sup>[2]</sup>, 严重时可致断链。在这种典型场景下, 地面站可应用智能天线构建灵活有效的空域抗干扰结构。典型抗干扰手段有如将自适应信号处理与阵列天线技术相结合的空域自适应滤波<sup>[3]</sup>, 当目标信号和干扰信号的空间来向不同时, 可通过调整阵元的加权参数实现天线波束的自适应赋形, 在干扰来向上形成零陷, 达到消除或抑制干扰的目的。常用基于功率倒置准则<sup>[4]</sup>的波束成形算法, 基本思想是通过最小化输出功率来抑制强干扰, 同时在有用信号方向保持较为平坦的响应。但这种算法仅对具有大干信比的干扰有效, 而对其他非期望信号抑制能力不足。

在空域滤波的实现效率方面, 从利于硬件加速的角度, 为避免阵列信号处理中的矩阵求逆计算, 常用自适应波束成形算法, 包括最小均方 (least mean square, LMS) 算法<sup>[5]</sup>、递归最小二乘 (recursive least squares, RLS) 算法<sup>[6]</sup>、采样矩阵求逆算法 (sample matrix inversion, SMI) 算法等。文献 [7] 在传统 LMS 算法基础上结合高斯函数和多尺度小波变换提出一种变步长算法, 收敛速度更快、稳态误差更小, 且具有更高的消噪能力; 文献 [8] 采用基于 QR 分解的最小二乘 (QR-decomposition recursive least square, QR-RLS) 算法实现抗干扰, 并研究了其脉动阵列实现方法, 大大加快了算法的收敛速度; 文献 [9] 通过矩阵降维, 提出在 RLS 迭代计算模块中采用状态机加流水线的数据结构, 将迭代计算速度提升了 71%。但 LMS 和 RLS 均通过多次迭代的方式逐步逼近最优权值向量, 需要收敛时间; 如果迭代之后算法仍未收敛, 则达不到抗干扰效果。SMI 算法具有开环结构, 所得的最优权值矩阵向量仅由采样信号的自相关矩阵统计特性决定, 不涉及迭代过程, 故不存在收敛问题, 计算速度天然优于 LMS 和 RLS 算法。

基于以上考虑, 本文利用现场可编程门阵列 (field programmable gate array, FPGA) 的并行计算特性<sup>[10]</sup>, 提出了一种高效自适应滤波及其硬件加

速实现方案。首先提出一种多空域约束的功率倒置约束准则, 能够同时在功率域上抑制强干扰、在空域上抑制任意来向的干扰; 借鉴 SMI 的非迭代思想, 通过矩阵 LDL 分解完成逆矩阵求解, 降低算法复杂度; 基于 FPGA 部署实现时, 通过脉动阵列结构完成高效矩阵运算, 同时自制定点-浮点转换单元, 保证了最优加权向量的计算精度。

## 1 高效自适应滤波算法原理

### 1.1 场景定义

某中高轨卫星在地面站上采用智能天线接收目标卫星的测控信号, 同一测控频段有多颗低轨卫星组网运行, 场景如图 1 所示。为简化讨论, 图中的智能天线阵形以线阵为例。天线阵列首先将接收到的混合信号 (包含目标测控信号与同频段的多个干扰信号) 经射频前端下变频为数字中频或数字基带信号, 然后采用自适应滤波算法得到最优权值向量, 用以对数字信号进行加权合成。再进行信号解扩、解调及解析, 恢复出遥测信息。

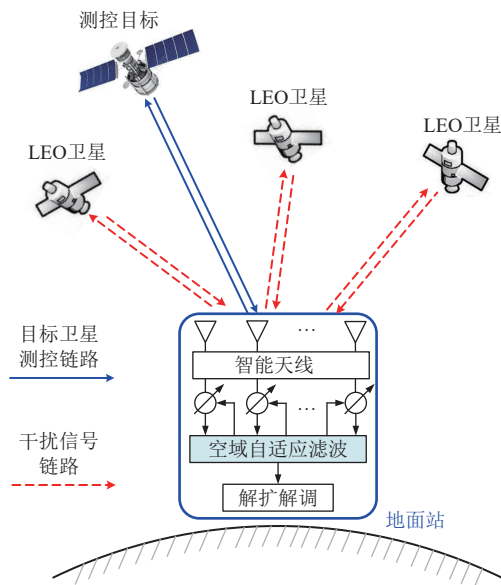


图 1 航天测控地面站抗干扰接收示意图

设航天测控地面站的智能天线有  $M$  个阵元, 其中第 1 个为参考阵元; 可观测  $K$  ( $K \leq M-1$ ) 个空间信号, 其中第  $k$  个信号为  $s_k(t)$  ( $k=1, 2, \dots, K$ )。则第  $m$  个阵元 ( $m=1, 2, \dots, M$ ) 接收到的信号为:

$$x_m(t) = x(t - \tau_{mk}) = s_k(t) e^{-j\omega_k \tau_{mk}} \quad (1)$$

式中,  $x(t)$  为阵列参考阵元接收到的信号;  $\tau_{mk}$  为第  $k$  个信号到达阵元  $m$  时相对于参考阵元的传播延迟有  $\tau_{mk} = [x_m(t) \sin \theta_k] / c$ , 其中  $\theta_k$  为第  $k$  个信号的到来向,  $c$  为光速;  $\omega_k$  为第  $k$  个信号载频  $f_k$  的角频率,

有  $\omega_k = 2\pi f_k$ 。

记空间相位为  $\varphi_{mk} = \omega_k \tau_{mk}$ , 则  $M \times K$  维导向矩阵  $\mathbf{A}$  的第  $k$  个空间导向矢量为:

$$\mathbf{a}_k = \begin{bmatrix} e^{-j\omega_k \tau_{1k}} & e^{-j\omega_k \tau_{2k}} & \dots & e^{-j\omega_k \tau_{Mk}} \end{bmatrix}^T \quad (2)$$

阵列天线射频前端在对接收信号下变频的同时进行量化, 式 (1) 可表示为离散时间序列:

$$\mathbf{x}(n) = \mathbf{A} \mathbf{s}(n) + \mathbf{v}(n) \quad (3)$$

进一步展开为:

$$\begin{bmatrix} x_1(n) \\ x_2(n) \\ \vdots \\ x_M(n) \end{bmatrix} = \begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \dots & \mathbf{a}_K \end{bmatrix} \begin{bmatrix} s_1(n) \\ s_2(n) \\ \vdots \\ s_K(n) \end{bmatrix} + \begin{bmatrix} v_1(n) \\ v_2(n) \\ \vdots \\ v_M(n) \end{bmatrix} \quad (4)$$

式中, 离散序号  $n$  通常被称为快拍 (snapshot);  $\mathbf{x}(n) \in \mathbb{C}^{M \times 1}$  表示所有阵元在第  $n$  个快拍的离散观测向量;  $\mathbf{A} \in \mathbb{C}^{M \times K}$  为导向矩阵;  $\mathbf{s}(n) \in \mathbb{C}^{K \times 1}$  为第  $n$  个快拍的信号向量;  $\mathbf{v}(n) \in \mathbb{C}^{M \times 1}$  为噪声向量。

设对第  $m$  个阵元接收信号的加权值为  $w_m$ , 则空域滤波器的输出信号  $y(n)$  可写为:

$$y(n) = \sum_{m=1}^M w_m x_m(n) \quad (5)$$

定义权值向量  $\mathbf{w} = [w_1 \ w_2 \ \dots \ w_M]^T \in \mathbb{C}^{M \times 1}$ , 可将式 (5) 表示为矩阵形式:

$$y(n) = \mathbf{w}^H \mathbf{x}(n) \quad (6)$$

## 1.2 多空域约束的功率倒置准则

传统的功率倒置准则规定约束条件为  $\mathbf{w}^H \mathbf{u} = 1$ , 其中  $\mathbf{w} = [w_1 \ w_2 \ \dots \ w_M]^T \in \mathbb{C}^{M \times 1}$ ,  $\mathbf{u} = [1 \ 0 \ \dots \ 0]^T$ , 实质上约束第 1 个阵元的加权值  $w_1$  始终为 1, 同时使系统输出功率最小。这就导致只能在功率域抑制强干扰。

多空域约束的功率倒置准则包含多个空域约束条件。假设有  $P$  个不期望的信号来向, 记为  $\theta_i (i=1, 2, \dots, P)$ , 将新增的空域约束表示为  $\mathbf{w}^H \mathbf{H} = \mathbf{V}$ , 其中  $\mathbf{H} \in \mathbb{C}^{M \times P}$  为空域约束矩阵,  $\mathbf{V} \in \mathbb{C}^{1 \times P}$  为空域约束响应向量。有:

$$\mathbf{H} = \begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \dots & \mathbf{a}_P \end{bmatrix} \\ \mathbf{V} = \begin{bmatrix} 0 & 0 & \dots & 0 \end{bmatrix}$$

为使滤波器输出功率最小, 取最优权值准则为:

$$\min_{\mathbf{w}} \mathbf{w}^H \mathbf{R} \mathbf{w} \quad \text{st.} \begin{cases} \mathbf{w}^H \mathbf{u} = 1 \\ \mathbf{w}^H \mathbf{H} = \mathbf{V} \end{cases} \quad (7)$$

式中,  $\mathbf{R} \in \mathbb{C}^{M \times M}$  为接收信号采样矩阵。

构造代价函数:

$$J(\mathbf{w}) = \mathbf{w}^H \mathbf{R} \mathbf{w} + \lambda_1 (\mathbf{w}^H \mathbf{u} - 1) + \lambda_2 (\mathbf{w}^H \mathbf{H} - \mathbf{V}) \quad (8)$$

当其梯度为零时, 解得最优权值向量为:

$$\mathbf{w}_{\text{opt}} = \mathbf{R}^{-1} \mathbf{C} (\mathbf{C}^H \mathbf{R}^{-1} \mathbf{C})^{-1} \mathbf{f} \quad (9)$$

式中,  $\mathbf{C} = \begin{bmatrix} \mathbf{u} & \mathbf{H} \end{bmatrix}$  为总约束矩阵;  $\mathbf{f} = \begin{bmatrix} 1 & \mathbf{V} \end{bmatrix}^T$  为总约束响应向量。

## 1.3 基于 LDL 分解的矩阵求逆算法

由式 (9) 可知, 求解最优权值向量必须进行矩阵求逆操作。线性代数的矩阵求逆方式有伴随矩阵法、初等变换法、特定系数法<sup>[11]</sup>等, 随着矩阵维度的增大, 所需的存储单元和计算单元呈指数律增加, 不利于硬件实现。本文借鉴 SMI 算法的开环思想, 在硬件实现中先通过矩阵分解, 将初始的目标矩阵变换为易于求逆的矩阵乘积形式: 即首先对这些分解所得的矩阵分别求逆, 再将多个逆矩阵相乘, 即可得到目标矩阵的逆矩阵。常见的矩阵分解方法包括 QR 分解、LU 分解、Cholesky 分解、LDL 分解<sup>[12]</sup>等。为获得低复杂度的求解方案, 对几种矩阵分解算法逐一分析如下 (Cholesky 分解涉及开方运算, 易累积误差, 本文不作讨论)。

### 1.3.1 矩阵 QR 分解

矩阵 QR 分解将给定的矩阵  $\mathbf{A}$  分解为  $\mathbf{Q}$  和  $\mathbf{R}$  两个矩阵的乘积, 即  $\mathbf{A} = \mathbf{Q} \mathbf{R}$ 。其中  $\mathbf{Q}$  为正交矩阵, 即  $\mathbf{Q}$  的逆矩阵等于其转置矩阵 ( $\mathbf{Q}^{-1} = \mathbf{Q}^T$ );  $\mathbf{R}$  为上三角矩阵, 所有非零元素都位于主对角线及其以上的位置。

实现矩阵 QR 分解的主要途径是 Gram-Schmidt 正交化法<sup>[13]</sup>。Gram-Schmidt 正交化通过一系列正交化步骤, 将目标矩阵的列向量转换为一组正交向量, 然后用这些向量形成矩阵  $\mathbf{Q}$ 。在这个过程中, 矩阵  $\mathbf{A}$  的每一列都被表示为正交向量的线性组合, 相应的系数形成上三角矩阵  $\mathbf{R}$ 。

将  $n \times n$  阶矩阵  $\mathbf{A}$  分解为  $\mathbf{A} = \mathbf{Q} \mathbf{R}$  的公式如下:

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \dots & \mathbf{a}_n \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1 & \mathbf{q}_2 & \dots & \mathbf{q}_n \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ & r_{22} & \dots & r_{2n} \\ & & \ddots & \vdots \\ & & & r_{nn} \end{bmatrix} \quad (10)$$

式中,  $\mathbf{a}_i$  是  $\mathbf{A}$  的第  $i$  个列向量;  $\mathbf{q}_i$  为正交矩阵  $\mathbf{Q}$  的列向量。递推求解矩阵  $\mathbf{Q}$  和  $\mathbf{R}$  的步骤如下:

- 1) 计算  $r_{11}$  和  $\mathbf{q}_1$ :  $r_{11} = \|\mathbf{a}_1\|_2$ ,  $\mathbf{q}_1 = \mathbf{a}_1 / r_{11}$
- 2) 计算矩阵  $\mathbf{R}$  第  $j$  列的上三角非对角元素:

$$r_{ij} = \mathbf{q}_i^T \mathbf{a}_j, \quad i = 1, 2, \dots, j-1$$

3) 正交化:  $\mathbf{q}_j = \mathbf{a}_j - r_{1j}\mathbf{q}_1 - r_{2j}\mathbf{q}_2 - \dots - r_{(j-1)j}\mathbf{q}_{j-1}$

4) 计算矩阵  $\mathbf{R}$  的对角线元素:  $r_{jj} = \|\mathbf{q}_j\|_2$

5) 单位化:  $\mathbf{q}_j = \mathbf{q}_j / r_{jj}$

6) 重复步骤②~步骤⑤, 直到遍历完所有变量

### 1.3.2 矩阵 LU 分解

矩阵 LU 分解将给定的  $n \times n$  阶矩阵  $\mathbf{A}$  分解为下三角矩阵  $\mathbf{L}$  和上三角矩阵  $\mathbf{U}$  的乘积, 即  $\mathbf{A} = \mathbf{LU}$ , 前提是矩阵  $\mathbf{A}$  的任意阶顺序主子式不为零。

LU 分解公式见式 (11), 求解公式见式 (12):

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n1} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} 1 & & & \\ l_{21} & 1 & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ & u_{22} & \cdots & u_{2n} \\ & & \ddots & \vdots \\ & & & u_{nn} \end{bmatrix} \quad (11)$$

$$\begin{cases} u_{1j} = a_{1j} & j = 1, 2, \dots, n \\ l_{i1} = a_{i1} / u_{11} & i = 1, 2, \dots, n \\ u_{ij} = a_{ij} - \sum_{k=1}^{i-1} l_{ik} u_{kj} & i = 2, 3, \dots, n; j = i, i+1, \dots, n \\ l_{ij} = \frac{a_{ij} - \sum_{k=1}^{j-1} l_{ik} u_{kj}}{u_{jj}} & i = j+1, j+2, \dots, n; j = 2, 3, \dots, n \end{cases} \quad (12)$$

### 1.3.3 矩阵 LDL 分解

矩阵 LDL 分解将给定的  $n \times n$  阶对称正定矩阵  $\mathbf{A}$  分解为 1 个下三角矩阵  $\mathbf{L}$ 、1 个对角矩阵  $\mathbf{D}$  及下三角矩阵共轭转置矩阵  $\mathbf{L}^H$  的乘积, 即  $\mathbf{A} = \mathbf{LDL}^H$ 。

LDL 分解公式见式 (13), 求解公式见式 (14)。

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n1} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} 1 & & & \\ l_{21} & 1 & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & 1 \end{bmatrix} \begin{bmatrix} d_{11} & & & \\ & d_{n2} & & \\ & & \ddots & \\ & & & d_{nn} \end{bmatrix} \begin{bmatrix} 1 & l_{21}^* & \cdots & l_{n1}^* \\ & 1 & \cdots & l_{n1}^* \\ & & \ddots & \vdots \\ & & & 1 \end{bmatrix} \quad (13)$$

$$\begin{cases} d_{11} = a_{11} \\ d_{ij} = a_{ij} - \sum_{k=1}^{j-1} l_{jk} l_{jk}^* d_{kk} & j = 2, 3, \dots, n \\ l_{i1} = a_{i1} / d_{11} \\ a_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk}^* d_{kk} \\ l_{ij} = \frac{a_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk}^* d_{kk}}{d_{ij}} & i = j+1, j+2, \dots, n \\ j = 2, 3, \dots, n \end{cases} \quad (14)$$

### 1.3.4 不同矩阵分解方法的复杂度对比

工程应用中, 自适应滤波算法的性能优劣很大程度上取决于矩阵分解算法所需的时间和空间资源<sup>[14]</sup>。以下从算法的操作次数多少和存储空间大小两方面, 分析不同矩阵分解算法的复杂度。

#### 1) 时间复杂度

以采用 Gram-Schmidt 正交化方法为例, 分析 QR 分解。由式 (10) 可知对  $n \times n$  阶矩阵而言, 在第  $j$  次循环中, 计算矩阵  $\mathbf{R}$  上三角非对角元素, 即与前  $j-1$  个已正交化的向量做点乘, 需要  $n$  次乘法和  $n-1$  次加法; 在正交化过程中, 需要进行  $j-1$  次向量减法, 每次向量减法需要  $n$  次乘法和  $n-1$  次减法; 在计算对角线元素, 即计算向量的模长过程中, 需要  $n$  次乘法、 $n-1$  次加法和 1 次开方操作; 最后进行单位化, 需要  $n$  次除法, 因此总计需要  $(4n-1)(j-1)+3n$  次操作数。遍历的循环次数为  $n$ , 故总操作次数为:

$$\sum_{j=1}^n [(4n-1)(j-1)+3n] = (4n-1)n(n-1)/2 + 3n^2 \Rightarrow O(2n^3) \quad (15)$$

LU 分解时, 由式 (11) 和式 (12) 可知对  $n \times n$  阶矩阵而言, 计算  $l_{ij}$  需  $j-1$  次乘法、 $j$  次加减法以及 1 次除法; 计算  $u_{ij}$  需  $i-1$  次乘法、 $i$  次加减法。则总操作次数为:

$$\sum_{j=1}^n \sum_{i=j+1}^n (2j) + \sum_{i=1}^n \sum_{j=i}^n (2i-1) \Rightarrow O\left(\frac{2}{3}n^3\right) \quad (16)$$

类似地, 由式 (13) 和式 (14) 可知,  $n \times n$  阶矩阵 LDL 分解中计算  $l_{ij}$  需  $2(j-1)$  次乘法、 $j$  次加减法及 1 次除法; 计算  $d_{jj}$  需  $2(j-1)$  次乘法、 $j$  次加法, 总操作次数为:

$$\sum_{j=1}^n \sum_{i=j+1}^n [2(j-1)+j+1] + \sum_{j=1}^n [2(j-1)+j] \Rightarrow O\left(\frac{1}{2}n^3\right) \quad (17)$$

#### 2) 空间复杂度

对  $n \times n$  阶矩阵  $\mathbf{A}$ , QR 分解和 LU 分解均需要

大小为 $2n^2$ 的存储空间来计算矩阵; 而 LDL 分解将正定矩阵  $A$  分解为下三角矩阵  $L$ 、对角矩阵  $D$  及其下三角矩阵的共轭转置  $L^H$  的乘积, 存储空间为  $n^2 + n$ 。

由表 1 可见, LDL 分解在时间、空间复杂度两方面均优于其他分解方法。

表 1 不同矩阵分解算法的复杂度对比

矩阵分解算法	时间复杂度	空间复杂度
QR分解	$O(2n^3)$	$2n^2$
LU分解	$O(2n^3/3)$	$2n^2$
LDL分解	$O(n^3/2)$	$n^2 + n$

## 2 硬件加速的自适应滤波算法实现

为高效实现本文提出的自适应滤波算法, 将算法整体部署到可并行计算的 FPGA 平台上。图 2 为基于 FPGA 平台实现的自适应滤波算法硬件加速部署示意图。

图中, 自适应滤波算法由定点-浮点转换、脉动阵列计算、矩阵求逆 (LDL 分解)、最优权值向量计算及其加权等顶层模块组成。自适应滤波完成后, 加权合成的 1 路信号进入后续解扩解调电路, 通过对信号的捕获、跟踪、同步, 获得原始 TT&C 信息。

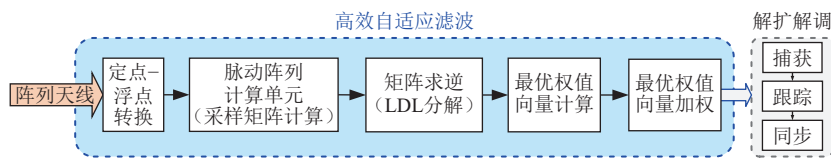


图 2 基于 FPGA 平台的自适应滤波硬件加速框图

### 2.1 定点-浮点转换模块

为确保权值计算精度, 首先进行阵列天线输入信号的数据预处理, 即定点-浮点转换。相应的模块结构如图 3 所示。

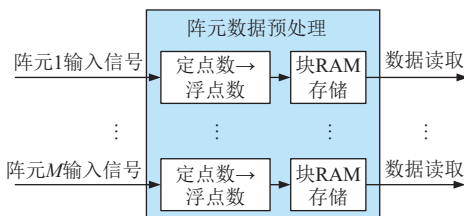


图 3 定点-浮点转换模块结构

通过每个阵元射频前端的下变频及数字化器件得到数字中频或数字基带信号。以 AD9361 为例, 每阵元输出为 12 bit 定点数表示的零中频数字基带信号。将每个 12 bit 定点数转换为 32 bit 单精度浮点数, 利用 FPGA 的片上块 RAM 为每个阵元数据开辟相应的缓存区。共存储  $N$  个快拍的阵元数据。

### 2.2 脉动阵列矩阵计算模块

自适应滤波最优权值计算过程中所需的矩阵运算占用大量时钟周期, 随着阵列维度的增大, 耗时呈指数增加。本文采用脉动阵列结构, 如图 4 所示, 其中“PE”表示脉动单元。参与运算的矩阵元素以流水线形式流入 PE 组成的脉动阵列。

通过流水线和阵列并行计算, 将矩阵运算分解成多个独立的脉动矩阵计算任务, 由此减少了计算

过程中的时间延迟, 极大提高了整个自适应滤波算法实现的实时性和计算效率。

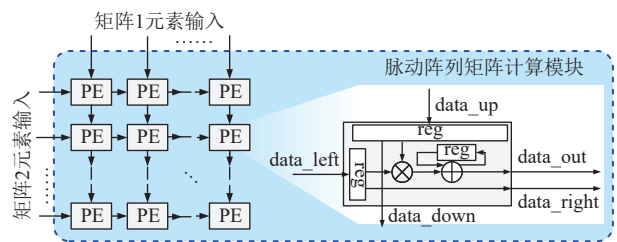


图 4 脉动阵列矩阵运算模块

### 2.3 矩阵求逆 (LDL 分解) 模块

矩阵求逆模块框图如图 5 所示。该模块按照式 (13) 完成 LDL 矩阵分解后, 由于对  $L$ 、 $D$  矩阵的求逆过程相互独立, 故可以同时进行。

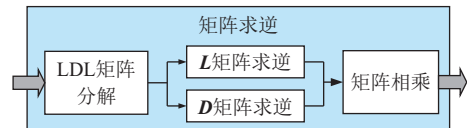


图 5 矩阵求逆模块框图

通过 LDL 分解计算出逆矩阵  $L^{-1}$  和  $D^{-1}$  之后,  $R$  的逆矩阵可由  $R^{-1} = (L^H)^{-1} D^{-1} L^{-1}$  得到。

### 2.4 最优权值向量计算与加权模块

该模块用于按照前述多空域约束的功率倒置准则计算最优权值向量  $w_{opt} = R^{-1} C (C^H R^{-1} C)^{-1} f$ 。分 3 步进行计算, 相应的模块结果如图 6 所示。

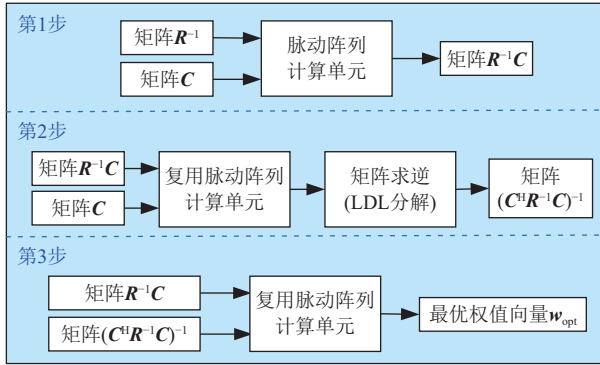


图 6 最优权值向量计算模块框图

之后，按照式 (6) 对阵列天线的观测向量  $\mathbf{x}(n)$  用最优权值向量  $\mathbf{w}_{\text{opt}}$  进行加权，得到 1 路合成信号，最后传入信号解扩解调部分。

### 3 实验验证

为验证本文所提算法及硬件加速的有效性，搭建了 10 元线阵的测控信号接收机系统进行实验。接收机基于 Xilinx 的 KCU105 开发板进行二次开发，其中 FPGA 芯片型号为 XCKU040。

实验验证所用的测控信号参数设置如表 2 所示。为不失一般性，信号采用航天测控中的 S 频段扩频 TT&C 体制，其中直序扩频 (direct sequence spread spectrum, DSSS) 伪码使用平衡 GOLD 码，码长为 1 023，码片宽度为  $1 \mu\text{s}$ 。考虑中轨卫星高度，将测控信号落地功率电平设置为  $-130 \text{ dBm}$ 。

表 2 测控信号参数设置

设置项	参数
信号体制	DSSS-BPSK
载波频率/GHz	2.36
伪码码型	平衡GOLD码
伪码长度	1 023
扩频码速率/Mcps	1.023
信息速率/kbps	1
信号入射角/ (°)	10
到达天线口面的功率/dBm	-130
信噪比/dB	-20

为了模拟测控链路的多干扰场景，本文设置了窄带、宽带 2 个不同来向的带内干扰信号。依照测控领域通常研究设定，将窄带干扰、宽带干扰的带宽分别设为测控信号带宽的 10%、50%；为了验证本文所提自适应滤波算法的有效性，设置任意未知来向的宽带干扰具有更大干信比，而将窄带干扰的来向作为先验条件来约束功率倒置准则，从而可以根据滤波结果进行分析：是否通过功率域的倒置准

则抑制了宽带干扰，以及是否通过增加的空域约束抑制了窄带干扰。相关参数的具体设置情况如表 3 所示。

表 3 带内干扰信号参数设置

干扰形式	窄带干扰	宽带干扰
干扰来向/ (°)	50 (已知)	-30
干扰带宽/kHz	200	1 023
干扰频率/GHz	2.361	2.361
干信比/dB	30	40
干扰波形	高斯白噪声	高斯白噪声

测控信号和 2 种带内干扰同时送入阵列天线中，接收到的混合信号频谱如图 7 蓝色信号所示。由于测控信号湮没于噪声之下，仅可观察到干扰信号。自适应滤波输出的频谱见图中红色信号，此时窄带干扰和宽带干扰都得以滤除，低于噪声的测控信号可借助扩频增益实现解译。

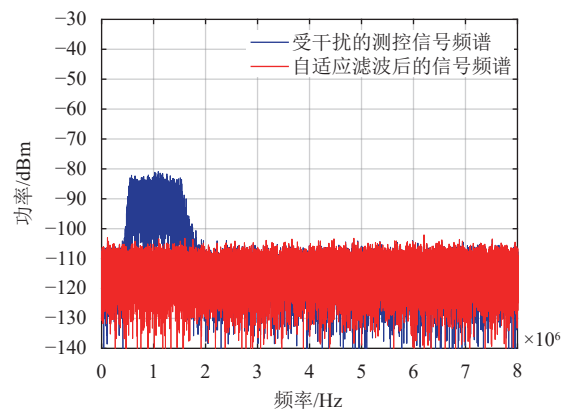


图 7 阵列天线接收信号频谱图

将 FPGA 计算出的最优权值向量与 Matlab 解得的最优权值向量进行对应元素的相对误差计算，结果如图 8 所示。可见，FPGA 计算出的各阵元最优权值向量相对误差均在  $10^{-6} \sim 10^{-7}$  量级。相比文献 [8] 基于 QR-RLS 的自适应滤波方法输出最优权值向量的相对误差 ( $10^{-3}$  左右)，本文的计算精度提升了 3 个数量级。

由最优权值向量进行阵元加权，合成后的波束方向图如图 9 所示。可以看到，算法对于有空域约束的窄带干扰方向形成了超过 80 dB 的深零陷，对压制式宽带干扰亦有 46.7 dB 的零陷。

为验证自适应滤波对有用信号未产生失真影响，对滤波后的测控信号继续进行解扩解调，结果如图 10 所示。由图可知，经过 2 000 次相关积分计算后，载波环和码环得到收敛，此时可以提取出原

始的基带信息。说明本文所提自适应滤波算法对抑制功率域、空域干扰同时有效。

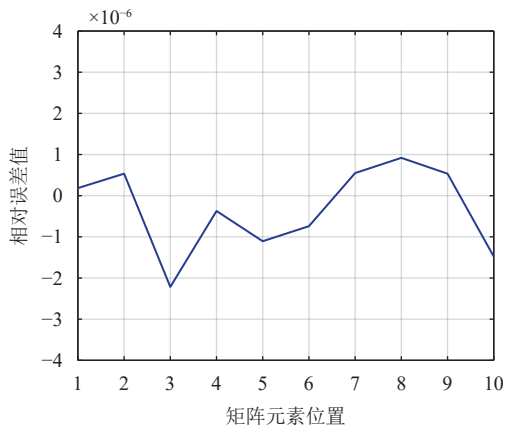


图 8 最优权值向量的相对误差曲线

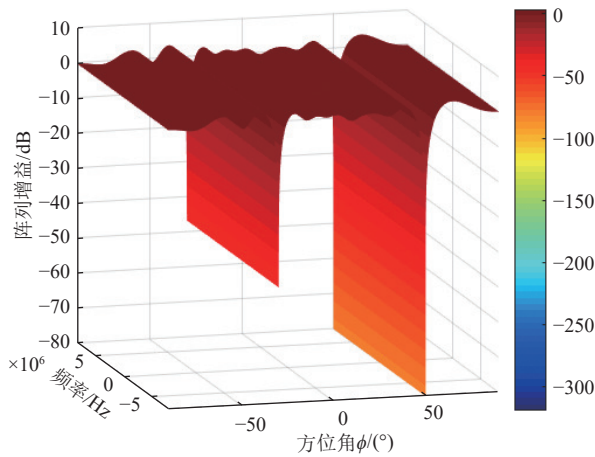


图 9 由最优权值合成的波束方向图

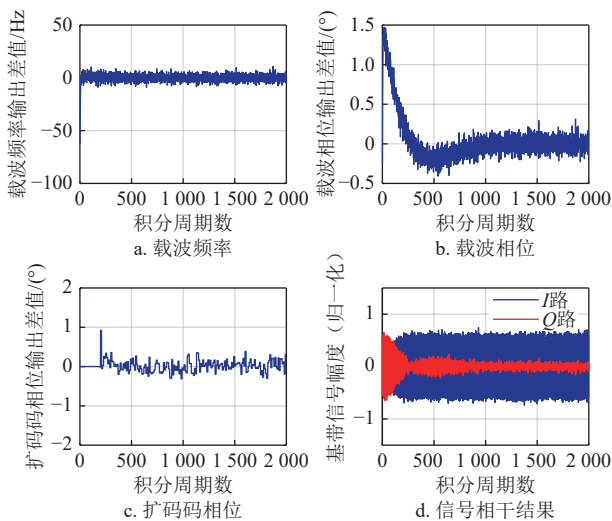


图 10 自适应滤波后的信号解调结果

将本文所提自适应滤波算法部署到 KCU105 开发平台上进行矩阵求逆计算, 接收阵列仍采用 10

元线阵, 系统时钟频率为 250 MHz。表 4 和表 5 分别给出了基于 QR 分解、LU 分解、LDL 分解的矩阵求逆模块所需时长以及硬件资源消耗情况对比。

表 4 不同算法计算权值耗时对比

算法	计算速度/ $\mu\text{s}$	相对误差
QR分解	876	$10^{-6}\sim 10^{-7}$
LU分解	308	$10^{-6}\sim 10^{-7}$
LDL分解	212	$10^{-6}\sim 10^{-7}$

表 5 XCKU040 资源占用情况

算法	LUT数	FF数	BRAM数
QR分解	24 486	34 881	57.5
LU分解	18 564	28 313	52
LDL分解	12 353	23 216	29.50

可见, 在保证计算精度的前提下 (相对误差均为  $10^{-6}\sim 10^{-7}$  量级), 本文所采用的 LDL 分解方法速度优势明显, 且占用资源最少。

## 4 结束语

本文面向航天测控中星-地链路上多种干扰共存场景, 研究解决从功率域和空域同时抑制干扰的方法及其高效实现途径。针对自适应波束成形问题, 提出了一种多空域约束的功率倒置准则, 不仅能在功率域对强干扰进行抑制, 同时可以在空域对任意来向的干扰形成零陷; 针对抗干扰算法的高实时性需求, 提出了一种适宜于硬件加速的自适应滤波实现方案, 从算法复杂度、计算速度、资源占用等层面进行效率提升, 同时保持了很高的权值计算精度。实验结果表明了本文所提算法的有效性。

后续研究拟面向大规模面阵以及共形阵天线, 研究可能的采样矩阵降维<sup>[15]</sup>或降秩方法, 在此基础上进一步讨论高效自适应滤波的实现可行性。

## 参考文献

[1] 夏姝姝. 卫星通信系统的空域抗干扰技术研究[D]. 成都: 电子科技大学, 2023.  
XIA S S. Research on anti-jamming technology in space of satellite communication system[D]. Chengdu: University of Electronic Science and Technology of China, 2023.

[2] 王海旺, 黄晨光, 邹诚, 等. 低轨卫星通信系统波束成形算法[J]. 天地一体化信息网络, 2022, 3(2): 3-11.  
WANG H W, HUANG C G, ZOU C, et al. Beamforming algorithm of LEO satellite communication system[J]. Space-Integrated-Ground Information Networks, 2022, 3(2): 3-11.

- [3] 何子述, 夏威, 等. 现代数字信号处理及其应用[M]. 北京: 清华大学出版社, 2009.  
HE Z S, XIA W, et al. Modern digital signal processing and its application[M]. Beijing: Tsinghua University Press, 2009.
- [4] COMPTON R T. The power-inversion adaptive array: Concept and performance[J]. IEEE Transactions on Aerospace and Electronic Systems, 1979, AES-15(6): 803-814.
- [5] WIDROW B, MANTEY P E, GRIFFITHS L J, et al. Adaptive antenna systems[J]. Proceedings of the IEEE, 1967, 55(12): 2143-2159.
- [6] HAYKIN S. Adaptive filter theory[M]. 3rd. [S.l.]: Prentice-Hall, Inc., 1996.
- [7] 陆友晴. 智能天线中自适应波束形成算法研究[D]. 昆明: 云南师范大学, 2023.  
LU Y Q. Research on adaptive beamforming algorithm in smart antenna[D]. Kunming: Yunnan Normal University, 2023.
- [8] 董海洋. 基于 QR-RLS 算法的阵列抗干扰研究与 FPGA 实现[D]. 成都: 电子科技大学, 2023.  
DONG H Y. Research on array anti-jamming based on QR-RLS algorithm and FPGA implementation[D]. Chengdu: University of Electronic Science and Technology of China, 2023.
- [9] 周围. 北斗导航自适应抗干扰算法研究及 FPGA 实现[D]. 成都: 电子科技大学, 2020.  
ZHOU W. Research on BeiDou navigation adaptive anti-jamming algorithm and FPGA implementation[D]. Chengdu: University of Electronic Science and Technology of China, 2020.
- [10] ROSADO A, IAKYMCHUK T, BATALLER M, et al. Hardware-efficient matrix inversion algorithm for complex adaptive systems[C]//Proceedings of the 19th IEEE International Conference on Electronics, Circuits, and Systems. New York: IEEE, 2012: 41-44.
- [11] ZHANG X W, ZUO L, LI M, et al. High-throughput FPGA implementation of matrix inversion for control systems[J]. IEEE Transactions on Industrial Electronics, 2020, 68(7): 6205-6216.
- [12] LI W, LEI Z Y, WU G, et al. A matrix inversion method based on LDLT decomposition and its application in STAP[C]//Proceedings of the 7th International Conference on Signal and Image Processing. New York: IEEE, 2022: 756-762.
- [13] DESAI P, ASLAN S, SANIIE J. FPGA implementation of Gram-Schmidt QR decomposition using high level synthesis[C]//Proceedings of the IEEE International Conference on Electro Information Technology. New York: IEEE, 2017: 482-487.
- [14] 未履伦. 阵列信号自适应抗干扰算法研究[D]. 成都: 电子科技大学, 2020.  
WEI L L. Research on adaptive anti-jamming algorithm of array signal[D]. Chengdu: University of Electronic Science and Technology of China, 2020.
- [15] 和洁, 冯大政, 向聪, 等. 机载 MIMO 雷达降维空时自适应处理算法[J]. 电子科技大学学报, 2012, 41(1): 31-35.  
HE J, FENG D Z, XIANG C, et al. Reduced-dimension STAP for airborne MIMO radars[J]. Journal of University of Electronic Science and Technology of China, 2012, 41(1): 31-35.

责任编辑 叶 芳