

用户下线时抑制串音的自更新算法

高 瑶,李有明,洪顺利,金 明

(宁波大学信息科学与工程学院,浙江宁波 315211)

摘 要: 串音是影响 VDSL2 系统传输性能的主要因素.已有的串音抑制算法大都局限于串音用户不变的环境.在实际系统中,用户随时可能下线和上线,导致串音环境发生改变.本文重点研究了有用户下线时的情形,利用矩阵初等变换和矩阵分块,提出了一种抑制串音干扰的自更新置零算法.新算法不仅减少了计算复杂度,同时能保持置零算法的性能.基于实测数据的仿真结果验证了新算法的有效性.

关键词: 第二代甚高速数字用户线;串音抑制;置零算法;用户下线;自更新实现

中图分类号: TN914.3 **文献标识码:** A **文章编号:** 0372-2112 (2012) 07-1368-04

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.3969/j.issn.0372-2112.2012.07.014

Crosstalk Cancellation Self-Updating Algorithm in Changing Interference Environment when Old Users Getting Offline

GAO Yao, LI You-ming, HONG Shun-li, JIN Ming

(Institute of Communication Technology, Ningbo University, Ningbo, Zhejiang 315211, China)

Abstract: Crosstalk is the major factor affecting the transmission performance of the VDSL2 system. Most of the existing crosstalk cancellation algorithms are used when the number of the users remains unchanged. In actual access environment, the old users often are offline or the new users are on-line, causing changes of crosstalk environment. In this paper, by using matrix elementary transformations and matrix partition, we propose a novel method to update the Zero-forcing cancellation method as the old users get offline. The new algorithm dramatically reduces the computational complexity of the original Zero-forcing algorithm while maintaining the same performance. Simulation results based on measured data demonstrate the effectiveness of the algorithm.

Key words: VDSL2; crosstalk cancellation; zero-focusing algorithm; users offline; self-updating implementation

1 引言

随着现代社会对信息的需求越来越大,特别是流媒体业务的兴起,迫使人们去寻找一种更高效的技术,以提供更高的传输速率.甚高速数字用户线技术(VDSL)是解决最后一公里接入瓶颈的有效技术,2006年推出的第二代VDSL(VDSL2),利用光纤到路边(FTTC)或光纤到楼宇(FTTB),在300m内能实现双向对称的100Mbps的高速数据传送,因此成为DSL接入技术的热点研究和应用方向^[1,2].

串音是影响VDSL2传输速率的主要因素.以电话线为基础的接入网中,多条双绞线被绑在同一条电缆中,当用户通过双绞线传递信号时,就会对周围用户产生电磁干扰,形成串音(crosstalk).串音按其传输特性可

分为近端串音(NEXT)和远端串音(FEXT),在上下行传输方向上,采用频分双工可以有效的避免NEXT;在上行传输中,需要在局端(Central office:CO)使用接收滤波器来消除FEXT^[3].而对于VDSL2系统,由于传输距离很短,FEXT通常比噪声强10dB左右,因此如何快速有效的抑制串音干扰是保证VDSL2高速传输的关键.

置零(Zero-forcing:ZF)算法是一种典型的串音抑制算法,但已有的研究仅仅讨论了接入网中用户不变的情况,而在实际系统中,经常会遇到用户上线及下线的情况.这时要及时抑制新干扰环境下的串音干扰,重复使用ZF算法会产生巨大的计算量^[4~6].在有旧用户离线的环境下,利用矩阵初等变换以及块分解,本文提出了一种ZF算法的自更新实现方法.新算法可以及时更新ZF滤波器,使运算量和算法复杂度大大降低,同时自更

新算法保持了原算法的性能.

2 上行传输模型^[2]

假设在一根铅包电缆中有 M 根双绞线,现有 N 个用户同时在线,采用 DMT 调制将信道分成 K 个子载频,则在第 k 个子载频上的输出信号为:

$$\mathbf{y}_k = \mathbf{H}_{N,k} \mathbf{x}_k + \mathbf{n}_k; k = 1, 2, \dots, K \quad (1)$$

$\mathbf{H}_{N,k}$ 表示第 k 个子载频上 $N \times N$ 的信道传输矩阵. h_{ij} 是 $\mathbf{H}_{N,k}$ 第 i 行第 j 列元素,表示第 j 个用户对第 i 个用户的串音, h_{ii} 表示第 i 个衰落信道的脉冲响应. $\mathbf{x}_k = (x_k^1, x_k^2, \dots, x_k^N)^T$ 表示第 k 个子载频上的发送符号向量, $\mathbf{y}_k = (y_k^1, y_k^2, \dots, y_k^N)^T$ 表示第 k 个子载频上的接收符号向量, $\mathbf{n}_k = (n_k^1, n_k^2, \dots, n_k^N)^T$ 表示第 k 个子载频上的加性高斯白噪声,则第 n 个用户在第 k 个子载频上的噪声功率为 $\sigma_k^n = \epsilon \{ |n_k^n|^2 \}$, $s_k^n = \epsilon \{ |x_k^n|^2 \}$ 表示第 k 个子载频上第 n 个用户的发送功率.

信号经过接收滤波器 \mathbf{W}_k 进行串音抑制后,令 $\mathbf{C}_{N,k} = \mathbf{W}_{N,k} \mathbf{H}_{N,k}$,则用户 n 在第 k 个子载频上所能承载的传输比特数为:

$$b_k^n = \min \left(b_{\max}, \left[\log_2 \left(1 + \frac{s_k^n |c_{nn}|^2}{\Gamma \left(s_k^n \sum_{i=1, i \neq n}^N c_{ni} + \sigma_k^n \sum_{i=1}^N |w_{ni}|^2 \right)} \right) \right] \right) \quad (2)$$

其中 b_{\max} 为实际比特加载能力, c_{ni} 和 w_{ni} 分别表示矩阵 $\mathbf{C}_{N,k}$ 和矩阵 $\mathbf{W}_{N,k}$ 的第 n 行 i 列元素. 在本文中 b_{\max} 取 15, Γ 为信噪比差额. 令符号速率为 f_s , 则每个用户的数据传输速率为:

$$R_n = f_s \sum_k b_k^n \quad (3)$$

3 用户下线时置零算法的自更新实现

假设在原 N 个在线用户中有 l 个用户同时下线,我们要讨论如何利用 N 个用户时抑制串音干扰的置零算法快速实现有 l 个用户离线时的置零算法.

N 个用户在线时,采用置零算法可得第 k 个子载频上抑制串音的滤波器为:

$$\mathbf{W}_{N,k} = \mathbf{H}_{N,k}^{-1} \quad (4)$$

当有 l 个用户离线时,实际在线用户数变为 $N-l$,串音环境发生变化,此时的滤波器应该变为:

$$\mathbf{W}_{N-l,k} = \mathbf{H}_{N-l,k}^{-1} \quad (5)$$

关于 $\mathbf{W}_{N-l,k}$ 与 $\mathbf{H}_{N,k}$ 、 $\mathbf{W}_{N,k}$ 的关系有如下引理刻划.

引理 1 存在一种初等行变换 \mathbf{T}_R 和列变换 \mathbf{T}_C , 使得 $\mathbf{H}_{N,k}$ 变成 $\mathbf{H}'_{N,k}$, 并且 $\mathbf{H}_{N,k}$ 与 $\mathbf{H}_{N-l,k}$ 及 \mathbf{T}_R 与 \mathbf{T}_C 间有如下关系:

$$\mathbf{H}'_{N,k} = \mathbf{T}_R \mathbf{H}_{N,k} \mathbf{T}_C = \begin{pmatrix} \mathbf{H}_{N-l,k} & * \\ * & * \end{pmatrix} \quad (6)$$

$$\mathbf{T}_R \mathbf{T}_C = \mathbf{I} \quad (7)$$

其中 \mathbf{I} 是 N 阶单位矩阵.

证明请见附录一

对 $\mathbf{W}_{N,k}$ 采用与引理 1 中相同的初等行列变换,记变换后的矩阵为 $\mathbf{W}'_{N,k}$, 即:

$$\mathbf{W}'_{N,k} = \mathbf{T}_R \mathbf{W}_{N,k} \mathbf{T}_C \quad (8)$$

则由式(7)容易推得:

$$\mathbf{H}'_{N,k} \mathbf{W}'_{N,k} = \mathbf{I} \quad (9)$$

引理 2 对 $\mathbf{H}'_{N,k}$ 、 $\mathbf{W}'_{N,k}$ 采用相同的矩阵分块法有:

$$\mathbf{H}'_{N,k} = \begin{pmatrix} \mathbf{H}_{N-l,k} & \mathbf{U} \\ \mathbf{V} & \mathbf{C} \end{pmatrix}, \mathbf{W}'_{N,k} = \begin{pmatrix} \mathbf{D}_{11} & \mathbf{D}_{12} \\ \mathbf{D}_{21} & \mathbf{D}_{22} \end{pmatrix} \quad (10)$$

则有:

$$\mathbf{H}_{N-l,k}^{-1} = \mathbf{D}_{11} + \mathbf{D}_{11} \mathbf{U} (\mathbf{I}_{22} - \mathbf{D}_{21} \mathbf{U})^{-1} \mathbf{D}_{21} \quad (11)$$

其中 \mathbf{I}_{22} 是 $l \times l$ 的单位矩阵.

证明请见附录二

根据引理 1 和引理 2 可推得, l 个用户离线时置零算法的自更新实现可表示为:

$$\mathbf{W}_{N-l,k} = \mathbf{D}_{11} + \mathbf{D}_{11} \mathbf{U} (\mathbf{I}_{22} - \mathbf{D}_{21} \mathbf{U})^{-1} \mathbf{D}_{21} \quad (12)$$

由上面的推导过程可以看出,自更新算法获得的用户下线时的串音滤波器与 ZF 滤波器相同,由此推断自更新算法保持了置零算法的性能.第 5 节的仿真结果也验证了这一推断的合理性.

4 计算复杂度分析

本文讨论的算法复杂度用乘除次数表示,当 n 个用户同时下线后,根据式(5),继续采用置零算法共需要 $4(N-n)^3$ 次乘除运算;根据式(6)、(8)、(10)、(12)可以统计出自更新算法共需要 $4(2(N-n)^2n + 2(N-n) \cdot n^2 + n^3)$ 次乘除运算.为了使结果更加具体,表 1 分别列出了 100 个用户在线,1 个、5 个、10 个用户同时下线时,一个子载频上两种算法的计算复杂度,表中 p-OL 表示有 p 个用户下线.

比较上面的计算复杂度可知,减少用户数 $n < N/3$ 时,自更新算法比继续采用置零算法的计算复杂度少,特别是当 N 较大, n 相对 N 较小时,计算复杂度减少非常大.而在实际系统中,多个用户同时下线概率很低,因此自更新算法能够快速实现用户下线时的置零算法.

表 1 置零算法和自更新算法计算复杂度比较

算法	置零	自更新
1-OL/次	3881196	79204
5-OL/次	3429500	380500
10-OL/次	2916000	724000

5 仿真分析

本文仿真采用 France Telecom(法国电信部)提供的实测数据,频带划分采用 VDSL2 定义的第三种方案.该数据提供的用户线长度分别有 75m、150m、300m、

600m,仿真的用户分布拓扑结构如图1所示,仿真信道初始时,用户1、2、3、5、6、7在线.在仿真中,上行传输频段3.75~5.2MHz、8.5~12MHz和18.1~30MHz的发送功率谱选为-50dBm,噪声功率谱选为-140dBm,子载频间距为8.625KHz,符号速率为7KHz.

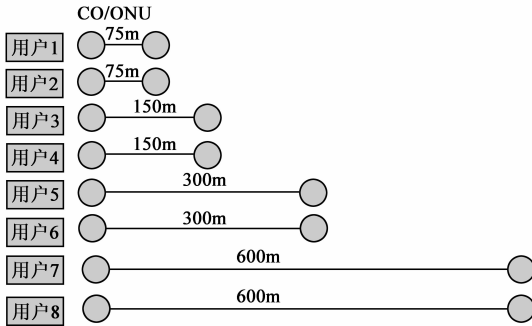


图1 信道拓扑结构

表2 用户下线后各用户的数据速率

用户	置零算法		自更新算法	
	用户5 下线	用户2、6 下线	用户5 下线	用户2、6 下线
用户1速率/Mbps	162.435	162.435	162.435	162.435
用户2速率/Mbps	162.435	—	162.435	—
用户3速率/Mbps	162.435	162.435	162.435	162.435
用户5速率/Mbps	—	142.072	—	142.072
用户6速率/Mbps	139.125	—	139.125	—
用户7速率/Mbps	33.46	33.46	33.46	33.46

表2列出了用户5下线和用户2、6同时下线时,自更新算法和置零算法的数据速率.从表2得到的仿真结果可以得出:自更新算法不会导致置零算法性能损失.

6 结论

本文提出了一种用户下线时能够快速实现置零算法的自更新算法.对于初始时用置零算法求出的串音滤波器,用户下线后,自更新算法根据原信道传输矩阵和滤波矩阵,利用矩阵初等变换和块分解,可以快速更新ZF滤波器.通过分析自更新算法的实现原理以及比较自更新算法和置零算法的计算复杂度表明,自更新算法在保持置零算法性能的同时计算复杂度也得到很大降低.

附录1

引理1 存在一种初等行变换 T_R 和列变换 T_C ,使得 $H_{N,k}$ 变成 $H'_{N,k}$,并且 $H_{N,k}$ 与 $H_{N-l,k}$ 及 T_R 与 T_C 间有如下关系:

$$H'_{N,k} = T_R H_{N,k} T_C = \begin{pmatrix} H_{N-l,k} & * \\ * & * \end{pmatrix} \quad (13)$$

$$T_R T_C = I \quad (14)$$

其中 I 是 N 阶单位矩阵.

证明: 现假设检测到 l 个下线用户依次是第 n_1, n_2, \dots, n_l 个用户,对 $H_{N,k}$ 采用初等变换,依次将第 n_p ($p = l, \dots, 1$) 行移到第 $N - (l - p)$ 行,第 n_p ($p = l, \dots, 1$) 列移到第 $N - (l - p)$ 列,其他行和列相对位置不变,记变换后的信道传输矩阵为 $H'_{N,k}$,则有:

$$H_{N-l,k} = H'_{N,k} (1:N-l, 1:N-l) \quad (15)$$

$$H'_{N,k} = T_R H_{N,k} T_C \quad (16)$$

其中 $T_R = \prod_{p=1}^l \prod_{i=N-(l-p)}^{n_p} I_i$, $T_C = \prod_{p=l}^1 \prod_{i=n_p}^{N-(l-p)} I_i$, N 阶矩阵 I_i ($n_1 \leq i \leq N-1$) 第 i 行第 $i+1$ 列元素为1,第 $i+1$ 行第 i 列元素为1,其他行对角线元素为1,其余元素都为0.

容易验证 $(I_i)^{-1} = I_i$,从而可推得:

$$T_R T_C = I \quad (17)$$

证毕

附录2

引理2 对 $H'_{N,k}$ 、 $W'_{N,k}$ 采用相同的矩阵分块法有:

$$H'_{N,k} = \begin{pmatrix} H_{N-l,k} & U \\ V & C \end{pmatrix}, W'_{N,k} = \begin{pmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{pmatrix} \quad (18)$$

则有:

$$W_{N-l,k} = D_{11} + D_{11} U (I_{22} - D_{21} U)^{-1} D_{21} \quad (19)$$

其中 I_{22} 是 $l \times l$ 的单位矩阵.

证明 根据式(9)有:

$$\begin{pmatrix} H_{N-l,k} & U \\ V & C \end{pmatrix} \begin{pmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{pmatrix} = \begin{pmatrix} I_{11} & 0 \\ 0 & I_{22} \end{pmatrix} \quad (20)$$

其中 I_{11} 是 $(N-l) \times (N-l)$ 的单位矩阵.

由式(20)得:

$$H_{N-l,k} D_{11} + U D_{21} = I_{11} \quad (21)$$

即:

$$H_{N-l,k} (D_{11} + H_{N-l,k}^{-1} U D_{21}) = I_{11} \quad (22)$$

由式(22)得:

$$H_{N-l,k}^{-1} = D_{11} + H_{N-l,k}^{-1} U D_{21} \quad (23)$$

从而:

$$H_{N-l,k}^{-1} = D_{11} (I_{11} - U D_{21})^{-1} \quad (24)$$

根据 $(A + UCV)^{-1} = A^{-1} - A^{-1} U (C^{-1} + VA^{-1}U)^{-1} VA^{-1}$ 有:

$$(I_{11} - U D_{21})^{-1} = I_{11} + U (I_{22} - D_{21} U)^{-1} D_{21} \quad (25)$$

由式(25)可得:

$$H_{N-l,k}^{-1} = D_{11} + D_{11} U (I_{22} - D_{21} U)^{-1} D_{21} \quad (26)$$

证毕

参考文献

- [1] Issam Wahibi, Meryem Ouzzif, Jerome Le Masson, Samir

- Saoudi. Crosstalk cancellation in upstream coordinated DSL using an iterative MMSE receive [A]. IEEE International Conference on Communications [C]. Dresden, 2009. 1 – 5.
- [2] 李有明, 沈微, 张敬艳, 徐铁峰. 上行传输中的迭代串音抑制算法[J]. 电子学报, 2010, 38(8): 1941 – 1944.
LI You-ming, Shen Wei, ZHANG Jing-yan, XU Tie-feng. Iterative crosstalk cancellation algorithm for upstream transmission [J]. Acta Electronica Sinica, 2010, 38(8): 1941 – 1944. (in Chinese)
- [3] Amir Leshem, Li Youming. A low complexity linear precoding technique for multichannel VDSL [J]. IEEE Transactions on Signal Processing, 2007, 55(11): 5527 – 5534.
- [4] S M Zafaruddin, Piyush Agrawal. Krylov algorithms for FEXT cancellation in upstream MIMO VDSL [A]. National Conference on Communications [C]. Chennai, 2010. 1 – 5.
- [5] John Homer, Mandar Gujrathi, Raphael Cendrillon, I Vaughan L Clarkson, Marc Moonen. Adaptive NLMS partial crosstalk cancellation in digital subscriber lines [A]. Fortieth Asilomar Conference on Signal, System and Computers [C]. Pacific Grove, CA, 2006. 1385 – 1389.

- [6] William W Hager. Updating the inverse of a matrix [J]. SIAM Review, 1989, 31(02): 221 – 239.

作者简介



高 瑶 男, 1988 年 1 月出生于湖北荆州, 2010 年毕业于中国矿业大学数学系. 现为宁波大学硕士研究生, 主要从事 DSL 中的串音抑制, 最优资源分配等的研究.



李有明(通信作者) 男, 1963 年 9 月出生于陕西扶风. 宁波大学通信技术研究所教授, 主要从事无线和有线宽带通信技术研究, 内容包括干扰抑制、动态频谱分配、MIMO-OFDM 技术、认知无线电技术等.

E-mail: liyouming@nbu.edu.cn