中国科学 F辑: 信息科学 2009 年 第 39 卷 第 5 期: 569～578
www.scichina.com info.scichina.com

《中国科学》杂志社
SCIENCE IN CHINA PRESS

酉空时调制系统中基于球形译码的多符号差分检测算法

李颖[①②*], 魏急波[①], 王欣[①], 于全[②]

① 国防科技大学电子科学与工程学院, 长沙 410073;
② 中国电子设备系统工程公司研究所, 北京 100039
*通信作者 E-mail: liying8800@yahoo.com.cn

收稿日期: 2008-04-12; 接受日期: 2009-01-05
国家重点基础研究发展计划 (批准号: 2009CB320403) 和十一五国防预研项目 (批准号: 1060741001020102) 资助项目

摘要 为酉空时调制系统设计的多符号差分球形译码能以较低复杂度获得最大似然检测性能. 但该算法基于准静态信道假设, 当将它用于快衰落信道时会出现严重的误码平层现象. 文中基于连续衰落信道假设, 将最大似然度量的一种递推形式嵌入自动球形译码中, 构建了多符号差分自动球形译码算法, 并以此为基础提出了降低计算复杂度的多符号差分近似自动球形译码与降低比较操作次数的多符号差分裁减自动球形译码. 与现存的两种典型算法, 即多符号判决反馈检测 (MS-DFD) 和非相干序列检测 (NSD) 相比, 所提算法的性能明显优于 MS-DFD, 接近 NSD 的性能, 但其平均复杂度远低于 NSD, 在大多数情况下明显低于 MS-DFD 的复杂度.

关键词
差分酉空时调制
多符号差分检测
球形译码

0 引言

在多天线(MIMO, multiple input multiple output)系统中, 差分酉空时调制(DUSTM, differential unitary space-time modulation)技术推广了传统差分相移键控的概念, 它不要求接收机对 MIMO 信道进行估计, 适用于不易获得信道状态信息的高速移动环境, 极大降低了系统复杂度, 引起了学者们的广泛关注 [1,2].

高效高性能的非相干检测算法设计是 DUSTM 的关键技术之一. 为了消除传统单符号差分检测 (SSDD, single symbol differential detection)[1,2] 在快衰落信道下的误码平层问题, 人们提出了各种性能优良的多符号差分检测 (MSDD, multiple symbol differential detection) 算法. 这些算法大多基于两种信道模型假设 —— 准静态 (QS, quasi static) 假设 (信道衰落在一个 DUSTM 符号内保持不变, 但在不同的 DUSTM 符号之间变化) 与连续衰落 (CF, continnous fading) 假设 (信道衰落在每个采样点均发生变化). 文献 [3～5] 基于 QS 假设分别设计了判决反馈差分检测、采用 Viterbi 搜索的非相干序列检测与多符号差分球形译码算法. 然而, 当它们应用于快衰落信道时, 会出现严重的性能恶化 [6]. 针对该问题, 文献 [6,7] 分别提出了基于 CF 假设的多符号判决反馈检测 (MS-DFD, multiple symbol-decision feedback detection) 与非相干序列检测. 在这些算法中, 非相干序列检测 [4,6] 最接近理想相干检测的性能, 但其复杂度与截断窗口长度呈指数增长 [7], 这限制了它的应用. 判决反馈差分检测 [3,7] 的复杂度较低, 但其固有的误差传播问题限制了它的性能. 最近提出的 QS 假设下多符号差分球形译码 (MSDSD, multiple symbol differential sphere decoding) 算法 [5] 具有与判决反馈检测相比拟的复杂度, 且达到了最大似然 (ML, maximum likelihood) 检测性能. 据作者所知, 目前没有文献提出在 CF 假设下, 采用球形译码 (SD, sphere decoding) 算法解决 MSDD 问题, 即本文的研究目的.

本文首先推导了一种基于 CF 假设的 ML 度量递推式, 该表达式指示了与球形译码算法相匹配的树

引用格式: 李颖, 魏急波, 王欣, 等. 酉空时调制系统中基于球形译码的多符号差分检测算法. 中国科学 F 辑: 信息科学, 2009, 39(5): 569–578

搜索形式, 然后结合自动球形译码 (ASD, automatic SD)[8] 构造了适用于任意差分酉空时星座的多符号差分自动球形译码 (MSDASD, multiple symbol differential ASD) 算法, 该算法以与观测窗口长度成平方关系的计算复杂度获得了 ML 检测性能, 并提供了基于 SD 算法设计的框架. 在此基础上, 通过忽略对 ML 检测结果影响较小的因素, 提出了降低计算复杂度的多符号差分近似自动球形译码 (MSDAASD, multiple symbol differential approximate ASD). 注意到在目前 DSP(digital signal processing) 技术中, 比较操作与乘加运算同样耗时 (如 TI6000 系列), 因此, 降低比较次数对算法实现也很重要. 我们进一步将统计裁减的概念引入 MSDAASD 中, 提出了减少比较操作次数的多符号差分裁减自动球形译码 (MSDPASD, multiple symbol differential pruning ASD). 分析与仿真表明, MSDAASD 算法非常接近 ML 检测性能; 在相同信噪比范围内, MSDPASD 算法在可接受的性能损失前提下较 MSDAASD 算法节省 30%以上的比较次数.

符号说明: $CN(0,\sigma^2)$ 表示均值为 0, 实虚部统计独立且方差各为 $\sigma^2/2$ 的复 Gauss 分布; $E(\cdot)$ 表示随机变量的均值. $\mathrm{diag}\{\cdot\}$ 表示将大括号内的矩阵 (或向量) 组成分块对角阵; $\det(\cdot)$ 与 $\mathrm{tr}(\cdot)$ 分别表示矩阵的行列式与迹; $(\cdot)^{\mathrm{T}}$ 与 $(\cdot)^{\mathrm{H}}$ 分别表示求矩阵的转置和共轭转置. $\mathrm{Re}(\cdot)$ 和 $\mathrm{Im}(\cdot)$ 分别表示取实部和虚部. $\circ$ 表示 Hadamard 积.

1 信号模型

考虑有 N_T 个发射天线与 N_R 个接收天线的 DUSTM 系统, 设 $l(k)\in\{0,1,\ldots,L-1\}$ 是信息序列, 对应的 N_T 阶信息矩阵 $\boldsymbol{V}(k)=\boldsymbol{V}_{l(k)}$ 取自差分酉空时星座 $V=\{\boldsymbol{V}_l,\ l=0,\ldots,L-1\}$, 其中 $L=2^{RN_T}$, R(比特/符号) 为码率. 第 k 块发送矩阵 $\boldsymbol{S}(k)$ 由对 $\boldsymbol{V}(k)$ 做差分得到

$$\boldsymbol{S}(k)=\boldsymbol{V}(k)\boldsymbol{S}(k-1). \tag{1}$$

式中, 矩阵 $\boldsymbol{S}(k)$ 中元素的列下标表示不同发射天线, 行下标表示从 kN_T 到 kN_T+N_T-1 的发射时刻; 初始值 $\boldsymbol{S}(0)$ 可取任意 N_T 阶酉矩阵. 记第 n_T 发射天线与第 n_R 接收天线之间的信道衰落系数为 $h_{n_Tn_R}(\kappa)$, 服从 $CN(0,1)$ 分布, 它们是空间独立的, 时间自相关函数服从 Clarke 模型 [9], 即 $\varphi_h(i)=E[h_{n_Tn_R}(\kappa)h^*_{n_Tn_R}(\kappa+i)]=J_0(2\pi B_f i)$, 其中 $J_0(\cdot)$ 是零阶 Bessell 函数, B_f 为归一化衰落带宽. 又设加性 Gauss 白噪声采样服从 $CN(0,\sigma_n^2)$ 分布. 依据上述功率归一化条件, 每接收天线上的平均信噪比为 $E_b/N_0=1/(R\sigma_n^2)$.

记 $\boldsymbol{S}(k)$ 的第 $p+1$ 行为 $\boldsymbol{s}_p(k), p=0,\ldots,N_T-1$, $\boldsymbol{H}_p(k)=[h_{n_Tn_R}(kN_T+p)]$ 为信号 $\boldsymbol{s}_p(k)$ 经历的 $N_T\times N_R$ 维信道矩阵, $\boldsymbol{W}(k)$ 为 $N_T\times N_R$ 维加线 Gauss 白噪声矩阵, 则第 k 个接收矩阵的符号表示为 [6,7]

$$\boldsymbol{R}(k)=\boldsymbol{S}_D(k)\boldsymbol{H}(k)+\boldsymbol{W}(k). \tag{2}$$

式中, $\boldsymbol{H}(k)=[\boldsymbol{H}_0^{\mathrm{T}}(k),\ldots,\boldsymbol{H}_{N_T-1}^{\mathrm{T}}(k)]^{\mathrm{T}}$, $\boldsymbol{S}_D(k)=\mathrm{diag}\{\boldsymbol{s}_0(k),\ldots,\boldsymbol{s}_{N_T-1}(k)\}$.

设观测窗口长度为 N, 注意到 MSDD 对每个观测窗口独立处理, 为方便起见, 省略每个观测窗口的时间起点标号. 记某个观测窗口的接收信号为 $\bar{\boldsymbol{R}}=[\boldsymbol{R}^{\mathrm{T}}(0),\ldots,\boldsymbol{R}^{\mathrm{T}}(N-1)]^{\mathrm{T}}$, 又记 $\bar{\boldsymbol{S}}_D=\mathrm{diag}\{\boldsymbol{S}_D(0),\ldots,\boldsymbol{S}_D(N-1)\}$, $\bar{\boldsymbol{H}}=[\boldsymbol{H}^{\mathrm{T}}(0),\ldots,\boldsymbol{H}^{\mathrm{T}}(N-1)]^{\mathrm{T}}$, $\bar{\boldsymbol{W}}=[\boldsymbol{W}^{\mathrm{T}}(0),\ldots,\boldsymbol{W}^{\mathrm{T}}(N-1)]^{\mathrm{T}}$, 得到该观测窗口的接收信号为

$$\bar{\boldsymbol{R}}=\bar{\boldsymbol{S}}_D\bar{\boldsymbol{H}}+\bar{\boldsymbol{W}}. \tag{3}$$

记 $\bar{\boldsymbol{S}}=[\boldsymbol{S}^{\mathrm{T}}(0),\ldots,\boldsymbol{S}^{\mathrm{T}}(N-1)]^{\mathrm{T}}$, 由文献 [6,7] 知, $\bar{\boldsymbol{R}}$ 的条件概率密度是

$$p(\bar{\boldsymbol{R}}|\bar{\boldsymbol{S}})=\frac{1}{\pi^{NN_TN_R}\det^{N_R}(\boldsymbol{B})}\exp\{-\mathrm{tr}(\bar{\boldsymbol{R}}^{\mathrm{H}}\boldsymbol{B}^{-1}\bar{\boldsymbol{R}})\}. \tag{4}$$

式中, $\boldsymbol{B}=\sigma_n^2\boldsymbol{I}_{N_TN}+\boldsymbol{C}_h\circ(\bar{\boldsymbol{S}}\bar{\boldsymbol{S}}^{\mathrm{H}})$ 是条件 $\bar{\boldsymbol{S}}$ 下 $\bar{\boldsymbol{R}}$ 的协方差矩阵, $\boldsymbol{C}_h$ 是信道衰落过程的 N_TN 阶自相关矩阵, 可表示为

$$\boldsymbol{C}_h(m)=\begin{bmatrix}\varphi_h(N_Tm) & \cdots & \varphi_h(N_Tm+N_T-1)\\ \vdots & & \vdots\\ \varphi_h(N_Tm-N_T+1) & \cdots & \varphi_h(N_Tm)\end{bmatrix},\quad \boldsymbol{C}_h=\begin{bmatrix}\boldsymbol{C}_h(0) & \cdots & \boldsymbol{C}_h(N-1)\\ \vdots & & \vdots\\ \boldsymbol{C}_h(-N+1) & \cdots & \boldsymbol{C}_h(0)\end{bmatrix}.$$

通过最大化 (4) 式得到最大似然检测结果为

$$\begin{aligned}\hat{\tilde{\boldsymbol{S}}}^{\mathrm{ML}} &= \arg\min_{\tilde{\boldsymbol{S}}}\{\mathrm{tr}(\bar{\boldsymbol{R}}^{\mathrm{H}}\tilde{\boldsymbol{B}}^{-1}\bar{\boldsymbol{R}}) + N_R \ln\det(\tilde{\boldsymbol{B}})\} \\ &= \arg\min_{\tilde{\boldsymbol{S}}}\{\tilde{d}_1 + N_R\tilde{d}_2\} = \arg\min_{\tilde{\boldsymbol{S}}}\tilde{d}. \end{aligned} \tag{5}$$

式中, $\tilde{d}_1 = \mathrm{tr}(\bar{\boldsymbol{R}}^{\mathrm{H}}\tilde{\boldsymbol{B}}^{-1}\bar{\boldsymbol{R}})$, $\tilde{d}_2 = \ln\det(\tilde{\boldsymbol{B}})$, $\tilde{d} = \tilde{d}_1 + N_R\tilde{d}_2$, $\tilde{\boldsymbol{B}} = \sigma_n^2\boldsymbol{I}_{N_TN} + \boldsymbol{C}_h \circ (\tilde{\tilde{\boldsymbol{S}}}\tilde{\tilde{S}}^{\mathrm{H}})$, $\tilde{\tilde{\boldsymbol{S}}} = [\tilde{\boldsymbol{S}}^{\mathrm{T}}(0), \ldots, \tilde{\boldsymbol{S}}^{\mathrm{T}}(N-1)]^{\mathrm{T}}$ 为实验矩阵序列. 在以下分析中, 我们称 $\tilde{d}$ 为 ML 度量.

2 连续衰落信道下的 MSDSD 算法

2.1 MSDASD 算法

为了利用球形译码解决 CF 假设下的多符号差分检测问题, 我们首先推导一种 ML 度量的递推形式, 然后利用最近提出的 ASD 算法结构 [8] 构建 MSDASD 算法. 我们首先引入一些符号:

$$\bar{\boldsymbol{R}}(n) = [\boldsymbol{R}^{\mathrm{T}}(0), \ldots, \boldsymbol{R}^{\mathrm{T}}(n)]^{\mathrm{T}},$$

$$\tilde{\tilde{\boldsymbol{S}}}(n) = [\tilde{\boldsymbol{S}}^{\mathrm{T}}(0), \ldots, \tilde{\boldsymbol{S}}^{\mathrm{T}}(n)]^{\mathrm{T}},$$

$$\bar{\boldsymbol{C}}_h(n) = [\boldsymbol{C}_h^{\mathrm{T}}(n), \ldots, \boldsymbol{C}_h^{\mathrm{T}}(1)]^{\mathrm{T}},$$

$$\bar{\boldsymbol{C}}_{h,n} = \begin{bmatrix} \boldsymbol{C}_h(0) & \cdots & \boldsymbol{C}_h(n) \\ \vdots & & \vdots \\ \boldsymbol{C}_h(-n) & \cdots & \boldsymbol{C}_h(0) \end{bmatrix},$$

$$\tilde{\boldsymbol{B}}_n = (\bar{\boldsymbol{C}}_{h,n} + \sigma_n^2\boldsymbol{I}_{N_T(n+1)}) \circ (\tilde{\tilde{\boldsymbol{S}}}(n)\tilde{\tilde{\boldsymbol{S}}}^{\mathrm{H}}(n)),$$

$$\tilde{\boldsymbol{B}} = \tilde{\boldsymbol{B}}_{N-1}.$$

利用上述符号, 并记 $\bar{\boldsymbol{R}}(n)$ 的第 n_R 列为 $\bar{\boldsymbol{R}}_{n_R}(k)$, 则

$$\begin{aligned}\tilde{d}_{1,n} &= \mathrm{tr}(\bar{\boldsymbol{R}}^{\mathrm{H}}(n)\tilde{\boldsymbol{B}}_n^{-1}\bar{\boldsymbol{R}}(n)) \\ &= \sum_{n_R=1}^{N_R}\bar{\boldsymbol{R}}_{n_R}^{H}(n)\tilde{\boldsymbol{B}}_n^{-1}\bar{\boldsymbol{R}}_{n_R}(n), \end{aligned} \tag{6a}$$

$$\tilde{d}_{2,n} = \ln\ \det(\tilde{\boldsymbol{B}}_n), \tag{6b}$$

$$\tilde{d}_n = \tilde{d}_{1,n} + N_R\tilde{d}_{2,n}. \tag{6c}$$

显然, $\tilde{d} = \tilde{d}_{N-1}$, $\tilde{d}_1 = \tilde{d}_{1,N-1}$, $\tilde{d}_2 = \tilde{d}_{2,N-1}$. 以下分别推导度量 $\tilde{d}_{1,n}$ 与 $\tilde{d}_{2.n}$ 的递推计算式. 首先将 $\tilde{\boldsymbol{B}}_n$ 表示成分块矩阵形式

$$\begin{aligned}\tilde{\boldsymbol{B}}_n &= (\bar{\boldsymbol{C}}_{h,n} + \sigma_n^2\boldsymbol{I}_{N_T(n+1)}) \circ (\tilde{\tilde{\boldsymbol{S}}}(n)\tilde{\tilde{\boldsymbol{S}}}^{\mathrm{H}}(n)) \\ &= \begin{bmatrix} \bar{\boldsymbol{C}}_{h,n-1} + \sigma_n^2\boldsymbol{I}_{N_Tn} & \bar{\boldsymbol{C}}_h(n) \\ \bar{\boldsymbol{C}}_h^{\mathrm{H}}(n) & \boldsymbol{C}_h(0) + \sigma_n^2\boldsymbol{I}_{N_T} \end{bmatrix} \\ &\quad \circ \left(\begin{bmatrix} \tilde{\tilde{\boldsymbol{S}}}(n-1) \\ \tilde{\boldsymbol{S}}(n) \end{bmatrix} \begin{bmatrix} \tilde{\tilde{\boldsymbol{S}}}^{\mathrm{H}}(n-1) & \tilde{\boldsymbol{S}}^{\mathrm{H}}(n) \end{bmatrix} \right) \\ &= \begin{bmatrix} (\bar{\boldsymbol{C}}_{h,n-1} + \sigma_n^2\boldsymbol{I}_{N_Tn}) \circ (\tilde{\tilde{\boldsymbol{S}}}(n-1)\tilde{\tilde{\boldsymbol{S}}}^{\mathrm{H}}(n-1)) & \\ \bar{\boldsymbol{C}}_h^{H}(n) \circ (\tilde{\boldsymbol{S}}(n)\tilde{\tilde{\boldsymbol{S}}}^{\mathrm{H}}(n-1)) & \\ \bar{\boldsymbol{C}}_h(n) \circ (\tilde{\tilde{\boldsymbol{S}}}(n-1)\tilde{\boldsymbol{S}}^{\mathrm{H}}(n)) & \\ (\boldsymbol{C}_h(0) + \sigma_n^2\boldsymbol{I}_{N_T}) \circ \boldsymbol{I}_{N_T} & \end{bmatrix} \\ &\triangleq \begin{bmatrix} \tilde{\boldsymbol{B}}_{n-1} & \tilde{\boldsymbol{B}}_n^{12} \\ (\tilde{\boldsymbol{B}}_n^{12})^{\mathrm{H}} & (\varphi(0) + \sigma_n^2)\boldsymbol{I}_{N_T} \end{bmatrix}. \end{aligned} \tag{7}$$

根据分块矩阵求逆公式 [10], 矩阵 $\tilde{\boldsymbol{B}}_n$ 的逆可表示为

$$\tilde{\boldsymbol{B}}_n^{-1} = \begin{bmatrix} \tilde{\boldsymbol{B}}_{n-1}^{-1} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} + \Delta\tilde{\boldsymbol{B}}_n. \tag{8}$$

式中, $\boldsymbol{T}_{\mathrm{en}} = (\varphi(0) + \sigma_n^2)\boldsymbol{I}_{N_T} - (\tilde{\boldsymbol{B}}_n^{12})^{\mathrm{H}}\tilde{\boldsymbol{B}}_{n-1}^{-1}\tilde{\boldsymbol{B}}_n^{12}$, 记

$$\Delta\tilde{\boldsymbol{B}}_n = \begin{bmatrix} \tilde{\boldsymbol{B}}_{n-1}^{-1}\tilde{\boldsymbol{B}}_n^{12} \\ -\boldsymbol{I}_{N_T} \end{bmatrix} \boldsymbol{T}_{\mathrm{en}}^{-1} \begin{bmatrix} (\tilde{\boldsymbol{B}}_n^{12})^{\mathrm{H}}\tilde{\boldsymbol{B}}_{n-1}^{-1} & -\boldsymbol{I}_{N_T} \end{bmatrix}.$$

将 (8) 式代入 (6a) 式得

$$\begin{aligned}\tilde{d}_{1,n} &= \sum_{n_R=1}^{N_R} \begin{bmatrix} \bar{\boldsymbol{R}}_{n_R}^{\mathrm{H}}(n-1) & \boldsymbol{R}_{n_R}^{\mathrm{H}}(n) \end{bmatrix} \\ &\quad \begin{bmatrix} \tilde{\boldsymbol{B}}_{n-1}^{-1} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \bar{\boldsymbol{R}}_{n_R}(n-1) \\ \boldsymbol{R}_{n_R}(n) \end{bmatrix} \\ &\quad + \sum_{n_R=1}^{N_R}\bar{\boldsymbol{R}}_{n_R}^{\mathrm{H}}(n)\Delta\tilde{\boldsymbol{B}}_n\bar{\boldsymbol{R}}_{n_R}(n) \\ &= \sum_{n_R=1}^{N_R}\bar{\boldsymbol{R}}_{n_R}^{\mathrm{H}}(n-1)\tilde{\boldsymbol{B}}_{n-1}^{-1}\bar{\boldsymbol{R}}_{n_R}(n-1) \\ &\quad + \sum_{n_R=1}^{N_R}\bar{\boldsymbol{R}}_{n_R}^{\mathrm{H}}(n)\Delta\tilde{\boldsymbol{B}}_n\bar{\boldsymbol{R}}_{n_R}(n) \\ &= \tilde{d}_{1,n-1} + \sum_{n_R=1}^{N_R}\bar{\boldsymbol{R}}_{n_R}^{\mathrm{H}}(n)\Delta\tilde{\boldsymbol{B}}_n\bar{\boldsymbol{R}}_{n_R}(n). \end{aligned} \tag{9a}$$

上式建立了 $\tilde{d}_{1,n}$ 的递推式. 将矩阵行列式的公式 [10] 应用到公式 (6b) 中, 得到 $\tilde{d}_{2,n}$ 的递推式

$$\tilde{d}_{2,n} = \ln\det\tilde{\boldsymbol{B}}_n = \ln(\det\tilde{\boldsymbol{B}}_{n-1}\det\boldsymbol{T}_{\mathrm{en}})$$

$$= \tilde{d}_{2,n-1} + \ln\det \boldsymbol{T}_{\mathrm{en}}. \tag{9b}$$

将 (9) 式代入 (6c) 式中即建立了 $\tilde{d}_n$ 的递推式. 注意到 $\tilde{S}(0)$ 的选取不影响 $\tilde{d}_0$ 的取值, 不失一般性, 可假设 $\tilde{\boldsymbol{S}}(0) = I_{N_T}$. 由 (9) 式看到, 在第 n 步递推中, 实验矩阵符号仅与 $\{\tilde{\boldsymbol{S}}(1), \ldots, \tilde{\boldsymbol{S}}(n)\}$ 有关, 这将 ML 解的搜索过程构建成树, 恰好符合球形译码 (SD) 算法所应用的模型.

在传统 SD 算法中, 通常首先根据前几维实验符号的取值确定当前维实验符号的取值范围, 然后按照某种策略 (F-P 或 S-E) 在所确定的范围内枚举当前维的实验符号 [11]. 但从上述分析看到, 在 DUSTM 多符号差分检测的度量计算中, 每一维上的实验符号是矩阵形式, 度量为矩阵的标量函数, 很难像传统 SD 算法一样通过不等式计算实验符号的取值范围. 注意到 ASD 算法不需要设置初始半径, 避免了每一维上实验符号取值范围的选定, 将其应用到这种情形中是非常自然的.

将 ML 度量递推式 (9) 嵌入 ASD 算法中, 便得到 MSDASD 算法. 以 $L = 2$, $N = 4$ 为例, 图 1 给出了 MSDASD 搜索 ML 解过程的示例. 与 ASD 算法 [8] 类似, MSDASD 将搜索 $\tilde{\tilde{\boldsymbol{V}}}$ 使 $\tilde{d}$ 最小的过程建立成一棵由 $\boldsymbol{C}_h$, σ_n^2, V 与 $\bar{\boldsymbol{R}}$ 唯一确定的深度为 N 的加权 L 元树 (从根至叶子节点的层次编号依次记为 $n = 0, \ldots, N-1$), 树中每个节点对应由根节点到该节点的唯一分支所确定的实验信息矩阵序列 $\tilde{\tilde{\boldsymbol{V}}}(n) = [\tilde{\boldsymbol{V}}^{\mathrm{T}}(1), \ldots, \tilde{\boldsymbol{V}}^{\mathrm{T}}(n)]^{\mathrm{T}}$、实验发送矩阵序列 $\tilde{\tilde{\boldsymbol{S}}}(n) = [\boldsymbol{I}_{N_T}, \tilde{\boldsymbol{S}}^{\mathrm{T}}(1), \ldots, \tilde{\boldsymbol{S}}^{\mathrm{T}}(n)]^{\mathrm{T}}$ 以及该节点的权重 $\tilde{d}_n$, 我们用 $\tilde{\tilde{\boldsymbol{V}}}(n)$ 来标识节点. 显然, 根节点对应的实验信息矩阵序列 $\tilde{\tilde{\boldsymbol{V}}}(0)$ 为空, 实验发送矩阵序列为 $\tilde{\tilde{\boldsymbol{S}}}(0) = \boldsymbol{I}_{N_T}$. 如图 1 所示, 搜索过程首先从根节点出发, 将扩张得到的节点存入边界节点列表 [1)]中, 然后选择列表内具有最小权重的节点进行扩张, 并将该节点从列表中删除、插入新的边界节点. 依此类推, 直到要扩张的节点为叶子节点时终止搜索, 该叶子节点便是 ML 解. 图 1 所示的搜索过程经历了 4 次节点扩张后找到了 ML 解. 将上述过程总结为算法伪码表述如下:

1) 初始化边界节点列表 N_b, N_b 用堆来存储, 并保证根节点 n_r 的权值总是最小的. 初始化根节点 n_r, 计算 $\tilde{d}_{1,0}, \tilde{d}_{2,0}, \tilde{\boldsymbol{B}}_0^{-1}, \tilde{d}_0$, 并预置实验发送矩阵 $\tilde{\boldsymbol{S}}(0) = \boldsymbol{I}_{N_T}$.

2) While n_r 不是叶子节点.

3) 调用函数 expandnode, 扩张节点 n_r, 得到它的 L 个孩子节点 $\{n_{r,1}, \ldots, n_{r,L}\}$.

4) 将 n_r 从 N_b 中删除, 并将节点 $\{n_{r,1}, \ldots, n_{r,L}\}$ 插入 N_b.

5) End while.

6) 将根节点 n_r 对应的实验信息矩阵序列 $\tilde{\tilde{\boldsymbol{V}}}$ 作为检测结果输出.

函数: expandnode

(当对节点 n_r 进行扩张时, 已确定的量有实验发送矩阵序列 $\{\tilde{\boldsymbol{S}}(0), \ldots, \tilde{\boldsymbol{S}}(n-1)\}$、实验信息矩阵序列 $\{\tilde{\boldsymbol{V}}(1), \ldots, \tilde{\boldsymbol{V}}(n-1)\}$. 其中, n 为 n_r 对应的实验发送矩阵序列长度.)

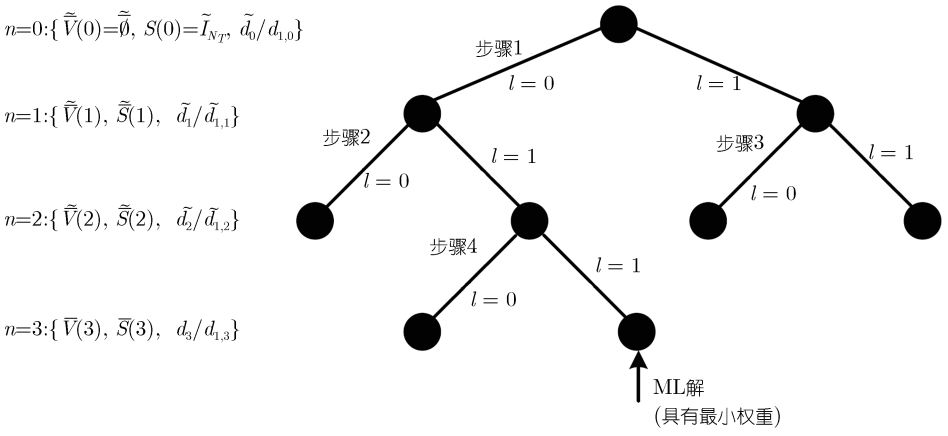


图 1 MSDASD/MSDAASD 算法流程示例 (L=2, N=4)

1) 在 ASD 算法中定义了 "边界节点列表" 的概念, 它界定了树中已遍历部分和未遍历部分之间的边界 [8]

7) For $i = 0 : L-1$.

8) 选择当前维实验信息矩阵为 $\tilde{\boldsymbol{V}}(n) = \boldsymbol{V}_l$, 将新生成的序列 $\{\tilde{\boldsymbol{V}}(1), \ldots, \tilde{\boldsymbol{V}}(n)\}$ 作为 n_r 的第 l 个孩子节点 $n_{r,l}$, 并利用上一维实验发送矩阵 $\tilde{\boldsymbol{S}}(n-1)$ 生成当前维实验发送矩阵 $\tilde{\boldsymbol{S}}(n) = \tilde{\boldsymbol{V}}(n)\tilde{\boldsymbol{S}}(n-1)$.

9) 计算节点 $n_{r,l}$ 的更新量 $\tilde{\boldsymbol{B}}_n^{12}, \boldsymbol{T}_{\mathrm{en}}, \Delta\tilde{\boldsymbol{B}}_n$, 递推得到 $\tilde{d}_{1,n}, \tilde{d}_{2,n}, \tilde{\boldsymbol{B}}_n^{-1}, \tilde{d}_n$ 并存储.

10) End for.

关于 ASD 算法的具体描述请参见文献 [8]. 在函数 expandnode 中, 直接利用实验信息矩阵生成实验发送矩阵, 因此该算法不要求星座必须是群结构, 但群星座可将上述伪码第 8) 行中的矩阵乘法简化为模加运算. 显然, 该算法可以达到 ML 检测性能. 以上算法描述是对每个观测窗口内接收数据的处理, 为了消除差分编码引入的相位模糊 (体现在假设 $\tilde{\boldsymbol{S}}(0) = I_{N_T}$), 相邻两观测窗口必须重叠一个符号.

2.2 MSDAASD 算法

注意到在文献 [5] 中关于准静态假设的 ML 度量中不包含 (5) 式中的 "ln det" 项, 而准静态假设可看作是连续衰落信道假设的近似, 因此, 当衰落带宽较小时, 我们有理由猜想, 该项对 ML 度量 $\tilde{d}$ 的动态范围影响是很小的 [12], 我们称 $\tilde{d}_1$ 为近似 ML(AML, approximate ML) 度量. 将关于 $\tilde{d}_1$ 的递推式 (9a) 嵌入 ASD 算法中, 即将 MSDASD 算法伪码中计算 ML 度量替换为计算 AML 度量, 便得到 MSDAASD 算法. 该算法与 MSDASD 算法流程上的不同仅在于将图 1 中节点度量 $\tilde{d}_n$ 替换为 $\tilde{d}_{1,n}$. 我们将在 3.2 小节中给出 AML 度量对性能影响的分析.

2.3 MSDPASD 算法

MSDAASD 算法的搜索过程需实时维护一个以堆这种数据结构来存储的边界节点列表. 随着酉星座集的增大 (相当于加权 L 元树中非叶子节点的孩子节点数增多), 需维护的堆中的元素也迅速增多, 从而引入了大量的比较操作. 事实上, 很多边界节点在后续搜索过程中被扩张的概率很小, 但它们必须参与由所有边界节点构成堆的相关操作, 这些操作对误符号率性能的贡献并不大 [13]. 受文献 [13] 中裁减自动球形译码 (PASD) 算法的启发, 本节我们提出 MSDPASD 算法以进一步减少 MSDAASD 中的比较操作. 参照图 1, 在 MSDAASD 算法中, 加权 L 元树第 n $(1 \leqslant n \leqslant N-1)$ 层节点 $\tilde{\tilde{\boldsymbol{V}}}(n)$ 对应的权重为 $\tilde{d}_{1,n} = \sum_{n_R=1}^{N_R} \bar{\boldsymbol{R}}_{n_R}^{\mathrm{H}}(n)\tilde{\boldsymbol{B}}_n^{-1}\bar{\boldsymbol{R}}_{n_R}(n)$, 设该层对应的裁减阈值为 th_n, 则节点 $\tilde{\tilde{\boldsymbol{V}}}(n)$ 经裁减后仍保留的概率是 $\Pr\{\tilde{d}_{1,n} \leqslant \mathrm{th}_n\}$. 为保证误符号率性能, 必须使第 n 层对应正确信息/发送矩阵序列的节点 $(\tilde{\tilde{\boldsymbol{V}}}(n) = \bar{\boldsymbol{V}}(n) = [\boldsymbol{V}^{\mathrm{T}}(1), \ldots, \boldsymbol{V}^{\mathrm{T}}(n)]^{\mathrm{T}}$, $\tilde{\tilde{\boldsymbol{S}}}(n) = \bar{\boldsymbol{S}}(n) = [\boldsymbol{S}^{\mathrm{T}}(0), \ldots, \boldsymbol{S}^{\mathrm{T}}(n)]^{\mathrm{T}})$ 被保留的概率足够大, 我们预设这个概率为 P_0, 即选择阈值 th_n, 满足

$$\Pr\left\{d_{1,n} = \sum_{n_R=1}^{N_R} \bar{\boldsymbol{R}}_{n_R}^{\mathrm{H}}(n)\boldsymbol{B}_n^{-1}\bar{\boldsymbol{R}}_{n_R}(n) \leqslant \mathrm{th}_n\right\} = P_0,$$

$$\boldsymbol{B}_n = \sigma_n^2 \boldsymbol{I}_{N_T(n+1)} + \bar{\boldsymbol{C}}_{h,n} \circ (\bar{\boldsymbol{S}}(n)\bar{\boldsymbol{S}}^{\mathrm{H}}(n)). \tag{10}$$

为此, 首先确定第 n 层的正确节点 $\bar{\boldsymbol{V}}(n)$ 对应度量 $d_{1,n} = \sum_{n_R=1}^{N_R} \bar{\boldsymbol{R}}_{n_R}^{\mathrm{H}}(n)\boldsymbol{B}_n^{-1}\bar{\boldsymbol{R}}_{n_R}(n)$ 的概率分布函数. 由 (4) 式推知, 在发送 $\bar{\boldsymbol{V}}(n)\bar{\boldsymbol{V}}(n)$ 的条件下, $\bar{\boldsymbol{R}}_{n_R}(n)$ 服从 $CN(0, \boldsymbol{B}_n)$ 分布, 从而二次型 $2\bar{\boldsymbol{R}}_{n_R}^{\mathrm{H}}(n)\boldsymbol{B}_n^{-1}\bar{\boldsymbol{R}}_{n_R}(n)$ 服从 $\chi^2_{2(n+1)N_T}$ 分布 [14]. 又 $\bar{\boldsymbol{R}}_1(n), \ldots, \bar{\boldsymbol{R}}_{N_R}(n)$ 相互统计独立, 因此 $2d_{1,n} = 2\sum_{n_R=1}^{N_R} \bar{\boldsymbol{R}}_{n_R}^{\mathrm{H}}(n)\boldsymbol{B}_n^{-1}\bar{\boldsymbol{R}}_{n_R}(n)$ 服从 $\chi^2_{2(n+1)N_TN_R}$ 分布, 代入条件 (10) 中, 得到 $\mathrm{th}_n = (1/2) \cdot \chi^{2-1}_{2(n+1)N_TN_R}(P_0)$. 事实上, 在上述分析中忽略了裁减中低层节点对高层节点的制约关系. 即如果位于较低层的节点被裁掉, 那么即使在以它为根的子树中包含权重小于阈值的节点, 也会被后续搜索过程忽略. 如果这个被裁节点恰为 ML 解对应叶子节点的先驱节点, 那么本次搜索将不会得到 ML 解. 为了减弱这种制约关系, 可以适度增大低层节点的裁减阈值 [13], 这里我们采用下式选取裁减阈值

$$\mathrm{th}_n = \max(1.5, n)(1/2)\chi^{2-1}_{2(n+1)N_TN_R}(P_0), \quad n = 1, 2, \ldots, N-1. \tag{11}$$

我们定义裁减操作为, 只保留权重小于相应阈值的边界节点, 如果所有边界节点的权重都大于阈值, 那么只保留其中最小权重对应的节点. 将 MSDASD 算法伪码中 ML 度量替换为 AML 度量, 并在第 3) 和 4) 行之间插入上述裁减操作, 便得到 MSDPASD 算法. 最后说明一点, 虽然我们没有证明这种阈值分配形式是最优的, 但仿真发现 (参见第 4 节), 在阈值比较合理的范围内, MSDPASD 算法的性能 (包括误符号率与比较次数) 对 th_n 的取值并不敏感. 因此, 从

实用意义上说, 没有必要追求最优的选取方式.

3 性能分析

3.1 MSDASD 的误符号率

我们首先分析 MSDASD 的误符号率 (即信息矩阵 $\boldsymbol{V}(k)$ 的平均错误概率), 在此基础上, 讨论次最优算法 MSDAASD 的误符号率性能, MSDPASD 的性能将在第 4 节给出仿真结果.

要得到精确的误符号率闭式表达式非常困难, 已有文献表明, 利用截断联合界可以在较高信噪比下得到误符号率的良好近似 [7,15], 其中成对错误概率 (PEP, pairwise error probability) 的计算与主要错误事件的提取是两个关键步骤. 这里, 我们也采用这种方法对误符号率进行分析.

设发送矩阵序列 $\bar{\boldsymbol{S}}$ 对应的 ML 度量为 $d = \mathrm{tr}(\bar{\boldsymbol{R}}^{\mathrm{H}}\boldsymbol{B}^{-1}\bar{\boldsymbol{R}}) + N_R\mathrm{lndet}(\boldsymbol{B})$, 实验矩阵序列 $\tilde{\bar{\boldsymbol{S}}} \neq \bar{\boldsymbol{S}}$ 对应的 ML 度量为 $\tilde{d}$, 则当采用 ML 度量时, 将 $\bar{\boldsymbol{S}}$ 误判成 $\tilde{\bar{\boldsymbol{S}}}$ 的概率 (PEP) 为

$$\begin{aligned}
&\Pr\{\bar{\boldsymbol{S}} \to \tilde{\bar{\boldsymbol{S}}}|\mathrm{ML}\} = \Pr\{\tilde{d} < d\} \\
&= \Pr\left\{\mathrm{tr}(\bar{\boldsymbol{R}}^{\mathrm{H}}(\tilde{\boldsymbol{B}}^{-1} - \boldsymbol{B}^{-1})\bar{\boldsymbol{R}}) < N_R\ln\frac{\det\boldsymbol{B}}{\det\tilde{\boldsymbol{B}}}\right\} \\
&= \Pr\left\{\sum_{n_R=1}^{N_R}\bar{\boldsymbol{R}}_{n_R}^{\mathrm{H}}(\tilde{\boldsymbol{B}}^{-1} - \boldsymbol{B}^{-1})\bar{\boldsymbol{R}}_{n_R}\right. \\
&\qquad \left. < N_R\ln\frac{\det\boldsymbol{B}}{\det\tilde{\boldsymbol{B}}} = c_{\bar{S}\to\tilde{\bar{S}}S}\right\}. \qquad (12)
\end{aligned}$$

由 (4) 式知, $\bar{\boldsymbol{R}}_{n_R} \sim CN(0, \boldsymbol{B})$, 得到 $\bar{\boldsymbol{R}}_{n_R}^{\mathrm{H}}(\tilde{\boldsymbol{B}}^{-1} - \boldsymbol{B}^{-1})\bar{\boldsymbol{R}}_{n_R}$ 的特征函数是 $\prod_{i=1}^{N_TN}(1 + s\lambda_i(\boldsymbol{B}\tilde{\boldsymbol{B}}^{-1} - \boldsymbol{I}_{N_TN}))^{-1}$[7], 0 其中 $\lambda_i(\boldsymbol{A})$ 为矩阵 $\boldsymbol{A}$ 的特征值. 又 $\bar{\boldsymbol{R}}_{n_R}, n_R = 1,\ldots,N_R$ 统计独立, 因此 $\sum_{n_R=1}^{N_R}\bar{\boldsymbol{R}}_{n_R}^{\mathrm{H}}(\tilde{\boldsymbol{B}}^{-1} - \boldsymbol{B}^{-1})\bar{\boldsymbol{R}}_{n_R}$ 的特征函数为

$$\Phi_{\bar{\boldsymbol{S}}\to\tilde{\bar{\boldsymbol{S}}}}(s) = \prod_{i=1}^{N_TN}(1 + s\lambda_i(\boldsymbol{B}\tilde{\boldsymbol{B}}^{-1} - \boldsymbol{I}_{N_TN}))^{-N_R}. \quad (13)$$

利用 (13) 式将 (12) 式进一步表示为

$$\begin{aligned}
&\Pr\{\bar{\boldsymbol{S}} \to \tilde{\bar{\boldsymbol{S}}}|\mathrm{ML}\} \\
&= \frac{1}{2\pi\mathrm{j}}\int_{\sigma-\mathrm{j}\infty}^{\sigma+\mathrm{j}\infty}\frac{\mathrm{e}^{sc_{\bar{S}\to\tilde{\bar{S}}}}\Phi_{\bar{\boldsymbol{S}}\to\tilde{\bar{\boldsymbol{S}}}}(s)}{s}\mathrm{d}s \\
&= \frac{1}{2\pi}\int_{-\infty}^{\infty}\frac{\mathrm{e}^{(\sigma+\mathrm{j}\omega)c_{\bar{S}\to\tilde{\bar{S}}}}\Phi_{\bar{\boldsymbol{S}}\to\tilde{\bar{\boldsymbol{S}}}}(\sigma+j\omega)}{\sigma+\mathrm{j}\omega}\mathrm{d}\omega. \quad (14)
\end{aligned}$$

采用 Gauss-Chebyshev 求积公式计算上式围线积分 [16], 得到

$$\begin{aligned}
&\Pr\{\bar{\boldsymbol{S}} \to \tilde{\bar{\boldsymbol{S}}}|\mathrm{ML}\} \\
&\approx \frac{\mathrm{e}^{\sigma c_{\bar{S}\to\tilde{\bar{S}}}}}{N} \\
&\quad\cdot\sum_{\nu=1}^{\mathrm{N}/2}\{\mathrm{Re}[\mathrm{e}^{\mathrm{j}\sigma c_{\bar{S}\to\tilde{\bar{S}}_{\eta_\nu}}}\Phi_{\bar{\boldsymbol{S}}\to\tilde{\bar{\boldsymbol{S}}}}(\sigma+\mathrm{j}\sigma\eta_\nu)] \quad (15) \\
&\quad+\eta_\nu\mathrm{Im}[\mathrm{e}^{\mathrm{j}\sigma c_{\bar{S}\to\tilde{\bar{S}}_{\eta_\nu}}}\Phi_{\bar{\boldsymbol{S}}\to\tilde{\bar{\boldsymbol{S}}}}(\sigma+\mathrm{j}\sigma\eta_\nu)]\}.
\end{aligned}$$

其中, 节点个数 N 是偶数, 节点 $\eta_\nu = \tan((2\nu-1)\pi/(2N))$. 为了使数值积分快速收敛, 最好选取 σ 为函数 $\mathrm{e}^{sc_{\bar{S}\to\tilde{\bar{S}}}}\Phi_{\bar{\boldsymbol{S}}\to\tilde{\bar{\boldsymbol{S}}}}(s)$ 的鞍点. 然而, 该积分对 σ 的选择并不敏感 [7]. 为方便起见, 我们选择 σ 为 $\Phi_{\bar{\boldsymbol{S}}\to\tilde{\bar{\boldsymbol{S}}}}(s)$ 的所有极点实部中最小正值的一半 [16].

设 $w_s(\bar{\boldsymbol{S}}, \tilde{\bar{\boldsymbol{S}}})$ 是 $\bar{\boldsymbol{S}}$ 与其错误判决 $\tilde{\bar{\boldsymbol{S}}}$ 之间不同的矩阵符号个数, 则发送矩阵平均错误概率的联合界可利用 PEP, 通过对 L^{N-1} 个可能发送矩阵序列 $\bar{\boldsymbol{S}}$ 对应的错误概率取平均得到

$$\begin{aligned}
P_S^{\mathrm{ML}} &\leqslant \frac{1}{L^{N-1}(N-1)}\sum_{\bar{\boldsymbol{S}}}\sum_{\tilde{\bar{\boldsymbol{S}}}\neq\bar{\boldsymbol{S}}}w_s(\bar{\boldsymbol{S}}, \tilde{\bar{\boldsymbol{S}}}) \\
&\quad\Pr\{\bar{\boldsymbol{S}} \to \tilde{\bar{\boldsymbol{S}}}|\mathrm{ML}\}. \qquad (16)
\end{aligned}$$

由于发送矩阵 $\boldsymbol{S}(k)$ 出错通常会引起连续两个信息矩阵 $\boldsymbol{V}(k)$ 与 $\boldsymbol{V}(k+1)$ 出错, 因此, 误符号率约是发送矩阵平均错误概率的两倍. 由 (16) 式得到误符号率的联合界为

$$\begin{aligned}
P_V^{\mathrm{ML}} \approx 2P_S^{\mathrm{ML}} &\leqslant \frac{2}{L^{N-1}(N-1)}\sum_{\bar{\boldsymbol{S}}}\sum_{\tilde{\bar{\boldsymbol{S}}}\neq\bar{\boldsymbol{S}}}w_s(\bar{\boldsymbol{S}}, \tilde{\bar{\boldsymbol{S}}}) \\
&\quad\cdot\Pr\{\bar{\boldsymbol{S}} \to \tilde{\bar{\boldsymbol{S}}}|\mathrm{ML}\}. \qquad (17)
\end{aligned}$$

每个发送矩阵序列 $\bar{\boldsymbol{S}}$ 对应错误事件 $\bar{\boldsymbol{S}} \to \tilde{\bar{\boldsymbol{S}}}$ 的个数为 $L^{N-1}-1$, 当 N 与 L 稍大时, (17) 式中求和项数变得非常庞大. 采用提取主要错误事件技术可以简化 (17) 式的计算, 并得到误符号率的一个良好近似 [15]. 文献 [5] 指出, 对于静态信道, 主要错误事件满足的条件是在发送矩阵序列 $\bar{\boldsymbol{S}}$ 中只有发送矩阵 $\boldsymbol{S}(n)(1 \leqslant n \leqslant N-1)$ 错成 $\tilde{\boldsymbol{S}}(n)$, 且 $\tilde{\boldsymbol{S}}(n)$ 使 $\mathrm{Retr}(\tilde{\boldsymbol{S}}^{\mathrm{H}}(n)\boldsymbol{S}(n))$ 最大. 只要衰落带宽不是很大, 可以认为连续衰落信道与静态信道下的主要错误事件是相同的 [17]. 在外求和项中, 对于非群星座, 为了满足只有 $\tilde{\boldsymbol{S}}(n)$ 出错, 且所有 $\tilde{\boldsymbol{V}}(n) \in V$, 我们只对满足形式 $\bar{\boldsymbol{S}}' = [\boldsymbol{I}_{N_T}\ \boldsymbol{V}_l^{\mathrm{T}}\ \boldsymbol{I}_{N_T}\ \boldsymbol{V}_l^{\mathrm{T}}\cdots]^{\mathrm{T}}, 0 \leqslant l \leqslant L-1$ 的发送矩阵序列对应的误符号率求平均也能得到联合

界的良好近似 [18]. 对 (17) 式简化后的联合界表示为

$$P_V^{\mathrm{ML}} \leqslant \frac{2}{L(N-1)} \sum_{\bar{\boldsymbol{S}}'} \sum_{\substack{\tilde{\bar{\boldsymbol{S}}}' \text{为} \bar{\boldsymbol{S}}' \text{的主} \\ \text{要错误事件}}} \Pr\{\bar{\boldsymbol{S}}' \to \tilde{\bar{\boldsymbol{S}}}'|\mathrm{ML}\}$$
$$\triangleq P_{\mathrm{VU}}^{\mathrm{ML}}. \tag{18}$$

上式考虑了当 $\tilde{\bar{\boldsymbol{S}}}'$ 为主要错误事件时, $w_s(\bar{\boldsymbol{S}}', \tilde{\bar{\boldsymbol{S}}}') = 1$. 利用 (18) 式计算的联合界 $P_{\mathrm{VU}}^{\mathrm{ML}}$ 称截断联合界.

3.2 MSDAASD 的误符号率

MSDAASD 算法采用了近似 ML 度量, 即发送矩阵序列 $\bar{\boldsymbol{S}}$ 对应的度量为 $d_1 = \mathrm{tr}(\bar{\boldsymbol{R}}^{\mathrm{H}}\boldsymbol{B}^{-1}\bar{\boldsymbol{R}})$, 实验矩阵序列 $\tilde{\bar{\boldsymbol{S}}} \neq \bar{\boldsymbol{S}}$ 对应的度量为 $\tilde{d}_1 = \mathrm{tr}(\bar{\boldsymbol{R}}^{\mathrm{H}}\tilde{\boldsymbol{B}}^{-1}\bar{\boldsymbol{R}})$, 则采用 AML 度量时, 将 $\bar{\boldsymbol{S}}$ 误判成 $\tilde{\bar{\boldsymbol{S}}}$ 的概率为 (PEP)

$$\Pr\{\bar{\boldsymbol{S}} \to \tilde{\bar{\boldsymbol{S}}}|\mathrm{AML}\} = \Pr\{\tilde{d}_1 < d_1\}$$
$$= \Pr\{\mathrm{tr}(\bar{\boldsymbol{R}}^{\mathrm{H}}(\tilde{\boldsymbol{B}}^{-1} - \boldsymbol{B}^{-1})\bar{\boldsymbol{R}}) < 0\}. \tag{19}$$

对照 3.1 小节中的分析, 只需令 (14) 式中 $c_{\bar{\boldsymbol{S}} \to \tilde{\bar{\boldsymbol{S}}}} = 0$, 便得到采用近似 ML 度量时的 PEP

$$\Pr\{\bar{\boldsymbol{S}} \to \tilde{\bar{\boldsymbol{S}}}|\mathrm{AML}\}$$
$$= \frac{1}{2\pi \mathrm{j}} \int_{\sigma - \mathrm{j}\infty}^{\sigma + \mathrm{j}\infty} \frac{\Phi_{\bar{\boldsymbol{S}} \to \tilde{\bar{\boldsymbol{S}}}}(s)}{s} \mathrm{d}s$$
$$= \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{\Phi_{\bar{\boldsymbol{S}}} \to \tilde{\bar{\boldsymbol{s}}}(\sigma + \mathrm{j}\omega)}{\sigma + \mathrm{j}\omega} \mathrm{d}\omega. \tag{20}$$

同样地, 令 (15) 式中 $c_{\bar{\boldsymbol{S}} \to \tilde{\bar{\boldsymbol{S}}}} = 0$, 得到采用 Gauss-Chebyshev 求积公式计算上式

$$\Pr\{\bar{\boldsymbol{S}} \to \tilde{\bar{\boldsymbol{S}}}|\mathrm{AML}\}$$
$$\approx \frac{1}{N} \sum_{\nu=1}^{N/2} \{\mathrm{Re}[\Phi_{\bar{\boldsymbol{S}} \to \tilde{\bar{\boldsymbol{S}}}}(\sigma + \mathrm{j}\sigma\eta_\nu)] \tag{21}$$
$$+ \eta_\nu \mathrm{Im}[\Phi_{\bar{\boldsymbol{S}} \to \tilde{\bar{\boldsymbol{S}}}}(\sigma + \mathrm{j}\sigma\eta_\nu)]\}.$$

类似 (18) 式, 利用 (21) 式得到采用 AML 度量时的截断联合界 $P_{\mathrm{VU}}^{\mathrm{AML}}$ 为

$$P_V^{\mathrm{AML}} \leqslant \frac{2}{L(N-1)}$$
$$\cdot \sum_{\bar{\boldsymbol{S}}}{}' \sum_{\substack{\tilde{\bar{\boldsymbol{S}}}' \text{为} \bar{\boldsymbol{S}}' \text{的主} \\ \text{要错误事件}}} \Pr\{\bar{\boldsymbol{S}}' \to \tilde{\bar{\boldsymbol{S}}}'|\mathrm{AML}\}$$
$$\triangleq P_{\mathrm{VU}}^{\mathrm{AML}}. \tag{22}$$

最后, 我们用一个示例来说明 AML 度量所造成的性能损失. 当采用典型的 Alamouti's STBC ($N_T = 2$)[19] 作为 DUSTM 星座时, 利用 (18) 式和 (22) 式计算 MSDASD 与 MSDAASD 的截断联合界 $(P_{\mathrm{VU}}^{\mathrm{ML}}, P_{\mathrm{VU}}^{\mathrm{AML}})$ 如图 2(a) 所示, 它们的差值 $(P_{\mathrm{VU}}^{\mathrm{AML}} - P_{\mathrm{VU}}^{\mathrm{ML}})$ 如图 2(b) 所示 (参数设置为 $R = 1, N_R = 1, N = 4, B_f = 0.03$). 由图可见, 在很大信噪比范围内, 两者基本重合, 两者差值比它们本身的值小约 3 个量级, 这说明采用 AML 度量造成的性能损失可以忽略.

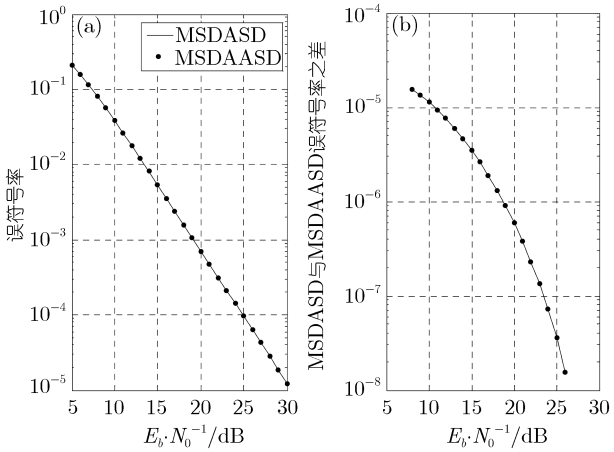


图 2 采用 AML 度量造成的性能损失 (Alamouti's STBC, R=1, N_R=1, N=4, B_f=0.03)

3.3 计算复杂度

除了误符号率性能之外, 复杂度也是评价算法的重要标准. 文中的 3 种算法复杂度是随机变量, 其均值与信噪比有关, 我们将平均每检测一个信息矩阵符号所消耗的平均乘法次数作为复杂度的度量标准. 文献 [8] 已指出, ASD 算法复杂度与在传统 SD 算法中设置初始半径为最优值时对应的复杂度相比拟, 但要得到它的解析表达式却非常困难. 考虑到 ASD 算法复杂度随信噪比增大较快收敛到一个稳定值 [8], 我们仅给出它在大信噪比下可能达到的最小复杂度, 在 4.2 小节中将给出平均复杂度的仿真结果. 由于文中 3 种算法的复杂度处于同一量级, 我们以 MSDAASD 为代表进行分析. 根据第 2 节的分析, 得到当采用 AML 度量时, 扩张搜索树第 $n(= 1, \ldots, N-1)$ 层某节点所需乘法次数约为

$$L[(n^2 + 4n - 2/3)N_T^3 + ((n+1)^2 N_R + n)N_T^2$$
$$+ (n+1)N_R N_T].$$

当信噪比很大时, 算法在搜索过程中扩张节点数

趋近于搜索树的深度 $N-1$[8]. 准确地说, 在大信噪比下, 搜索过程很有可能只扩张了正确叶子节点对应分支上的所有节点, 此时, 算法复杂度达到最低

$$\frac{L}{N-1}\sum_{n=1}^{N-1}[(n^2+4n-2/3)N_T^3 + ((n+1)^2N_R+n)N_T^2+(n+1)N_RN_T]. \quad (23)$$

与本文算法不同, 两种典型的同样基于 CF 假设的 MSDD 算法 ——MS-DFD[6] 和 NSD[7] 的复杂度都是与信噪比无关的常数, 通过分析算法步骤可以直接得到它们的计算复杂度. 表 1 给出了本文所述算法与 MS-DFD 和 NSD 的复杂度与观测窗口 (对 NSD 来说是截断窗口 [7]) 长度关系的比较. 从表中看到, NSD 算法的复杂度随截断窗长度呈指数增长, 致使它与具有相同观测窗口的 ML 多符号差分检测算法的复杂度相比拟. MS-DFD 由于不存在递推关系, 其观测窗口长度决定了需求逆矩阵的大小 (NN_T 阶), 这限制了它的复杂度与观测窗口长度至少呈 3 次方关系. 而本文算法利用递推避免了对大矩阵求逆, 只涉及 N_T 阶矩阵求逆, 从而将主要决定复杂度的操作转化为矩阵乘法, 其复杂度在高信噪比下与观测窗口长度呈平方关系. 另外, 观察 (23) 式可以看出, 接收天线数 N_R 对算法的复杂度影响不大 (呈线性关系增长), 但它决定了接收分集阶数, 因此, 通过增大 N_R 获取性能提高是很经济的.

表 1　连续衰落假设下不同检测算复杂度比较

算法	复杂度
MS-DFD	$O(N^3)$
NSD	$O(N^3\mathrm{e}^{N-2})$
MSDASD/MSDAASD/MSDPASD	$O(N^2)$(高信噪比)

4　仿真结果

我们采用 Alamouti's STBC 作为酉空时星座, 在连续衰落信道下, 从误符号率与平均复杂度两方面仿真文中 3 种算法, 并与 MS-DFD[6], NSD[7] 做比较. 作为性能基准, 我们同时给出标准的单符号差分检测 (SSDD)[1] 与理想相干 (coherent) 检测 (信道估计无误差) 的性能 [20]. 其中, 理想相干检测的误符号率是无差分编码情况下的两倍, 这是因为差分编码会引起相邻两符号间的误差传递.

4.1　误符号率比较

图 3 是在同等仿真环境 ($B_f=0.05, R=1, N_R=1$) 下, MSDASD 与 MSDAASD 算法的误符号率性能, 图中标记表示仿真结果, 曲线表示分析结果. 从图中看到, 虽然信噪比较低时, 由 (18) 与 (22) 式计算出的截断联合界较松, 但随着信噪比的增大, 这个联合界成为误符号率的一个很好的估计. 正如 3.2 小节所述, 在本例中 MSDASD 与 MSDAASD 的性能非常接近. 考虑到 MSDAASD 具有更低的复杂度, 在后续仿真中, 我们只给出 MSDAASD 的仿真结果. 另外, 从图 3 还可以看出, 根据信道衰落变化的快慢, 恰当的选取观测窗口长度, 可以消除误码平层现象.

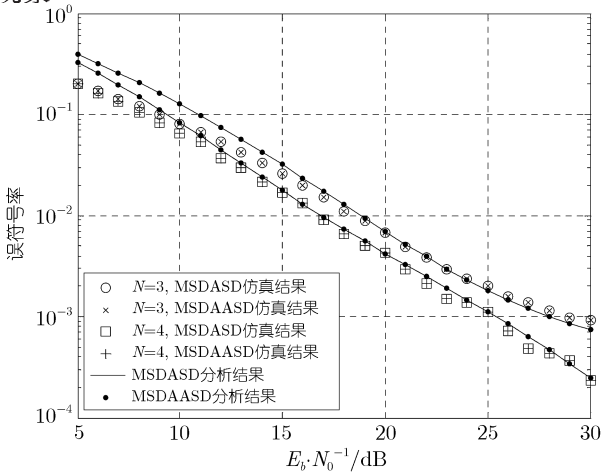


图 3　MSDASD 与 MSDAASD 误符号率性能比较 (B_f=0.05, R=1, N_R=1)

图 4 是在 $B_f=0.03, R=1, N_R=1, N=4$ 时, 各种检测算法在连续衰落信道下的检测性能. 仿真表明, SSDD 产生了严重的误码平层, 而其他 3 种算法在仿真信噪比范围内均显示出良好性能. 在误符号率为 10^{-4} 时, MSDAASD 的性能优于 MS-DFD 约 3 dB, 但劣于 NSD 约 1 dB, 距理想相干检测还有约 2.5 dB 的性能差异.

我们设置系统参数为 $B_f=0.03, R=2, N_R=2, N=4$, 对 MSDPASD 算法选取 3 种阈值, 比较它们的性能. 第 1 种 (阈值 1): 设置参数 $P_0=0.9$, 按照 (11) 式计算阈值 th_n; 第 2 种 (阈值 2): 设置 $p_{0,1}=0.1, p_{0,N-1}=0.0001$, 按照 (24) 式选取阈值 th_n; 第 3 种 (阈值 3): 设置参数 $P_0=0.99$, 按照 (11) 式计算阈值 th_n.

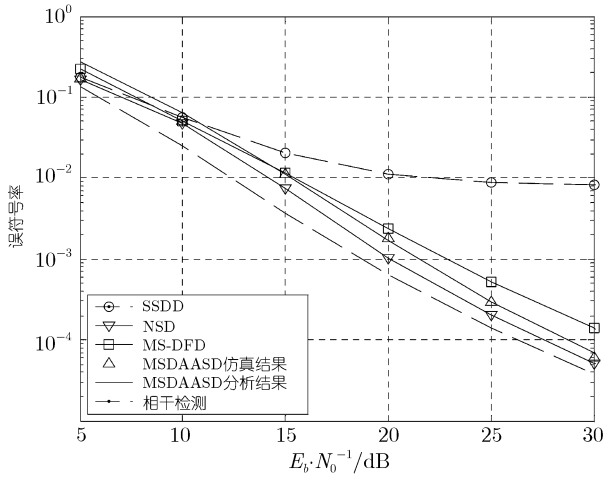


图 4 不同检测器的误符号率性能 (B_f=0.03, R=1, N_R=1, N=4)

$$p_{0,n} = p_{0,1} \cdot \left(\frac{p_{0,N-1}}{p_{0,1}}\right)^{\frac{n-1}{N-2}},$$

$$\mathrm{th}_n = \max(1.5, n)\left(\frac{1}{2}\right)\chi^{2-1}_{2(n+1)N_T N_R}(1-p_{0,n}),$$

$$n = 1, \ldots, N-1. \tag{24}$$

图 5 是这 3 种阈值设置对应的 MSDPASD 算法的误符号率性能, 作为比较, 我们也绘出了 MSDAASD 算法的误符号率. 从图中看到, 第 3 种阈值选择方法对应的误符号率在整个仿真信噪比范围内与 MSDAASD 算法的误符号率基本重合, 在误符号率为 10^{-5} 量级时, 前 2 种阈值选择方法出现了性能损失. 注意到第 2 种方法相对于第 1 种放松了搜索树中较低层节点对应的阈值, 其误符号率损失比第 1 种方法的损失稍小. 在 4.2 小节中, 我们将给出这 3 种阈值选择方法对应的复杂度降低效果.

4.2 复杂度比较

图 6 是在与图 4 相同仿真环境下得到的 MSDAASD、MS-DFD 与 NSD3 种算法的平均复杂度, 图中虚线是由 (23) 式计算出的 MSDAASD 可能的最小复杂度, 右边小图是对左下角部分的放大. 图中显示, NSD 算法的复杂度远高于其他 2 种算法. 在低信噪比时, MSDAASD 的平均复杂度与 MS-DFD 的复杂度相比拟; 随着信噪比的增大, 前者越来越多地低于后者; 在高信噪比下, 前者约为后者的一半. 对照图 4 我们可以明显看到 MSDAASD 算法在性能与复杂度折衷上的优势.

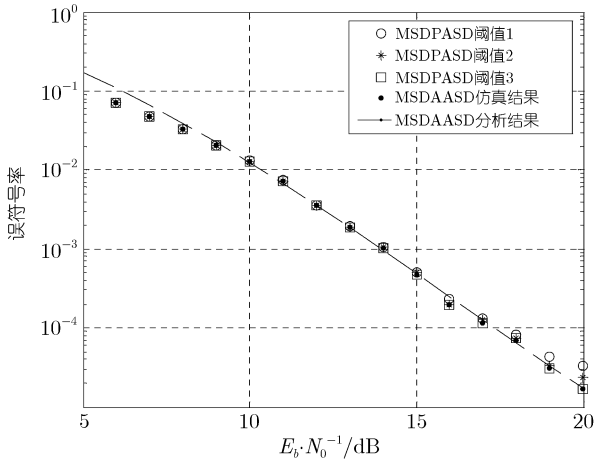


图 5 MSDPASD 的误符号率性能 (B_f=0.03, R=2, N_R=2, N=4)

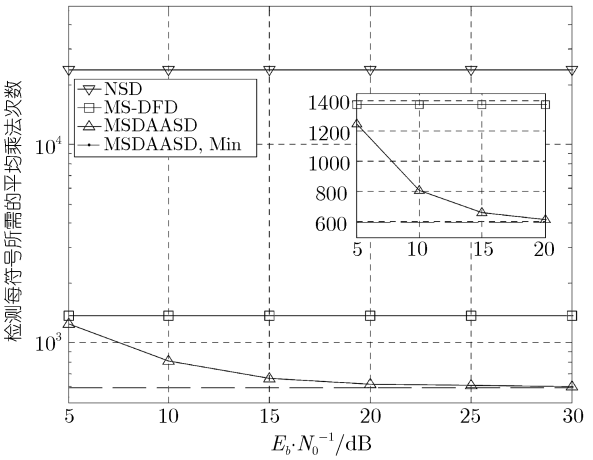


图 6 不同检测算法复杂度比较 (B_f=0.03, R=1, N_R=1, N=4)

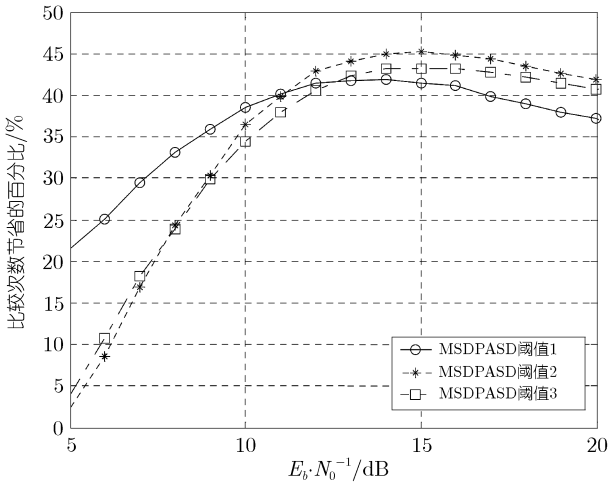


图 7 MSDPASD 的比较次数相对于 MSDAASD 的节省百分比 (B_f=0.03, R=2, N_R=2, N=4)

在MSDAASD与MSDPASD算法复杂度的比较中, 我们主要考虑后者相对于前者平均每检测一个观测窗口所需比较次数下降的百分比. 我们选用与4.1小节中相同的3种阈值选择方法, 在相同仿真环境下, 给出它们对应的比较次数的节省程度如图7所示. 当信噪比在10 dB时, 这3种阈值选取方法均节省了30%以上的比较次数, 同时对照图5可以看到在相同信噪比下它们的误符号率性能损失可以忽略. 在高信噪比时, 3种方法对应的复杂度节省均有所下降, 其中第1种方式最为明显. 这是因为, 当信噪比增大时, MSDPASD算法中的裁减操作频繁的在边界节点列表中查找具有最小权重的节点. 考虑到高信噪比时, MSDPASD算法复杂度本身已经收敛到较低水平, 因此, 从实用角度来看, 图中显示的高信噪比下轻微复杂度升高可以接受. 在实际应用中, 可以根据系统工作时信噪比的大致范围, 选取一个合适的阈值.

5 结论

本文在连续衰落信道假设下, 提出了3种基于球形译码的MSDD算法——MSDASD, MSDAASD与MSDPASD. 其中, MSDASD可以达到ML检测性能; MSDAASD通过舍弃对ML解的选取影响较小的项, 在几乎不损失检测性能的前提下降低了复杂度; MSDPASD通过统计裁减进一步减少了MSDAASD中对性能贡献不大的比较操作, 获得了较好的性能与复杂度的折衷. 与现存的2种典型算法MS-DFD和NSD相比, 所提算法的性能明显优于MS-DFD, 接近NSD的性能, 但复杂度远低于NSD, 在小信噪比下与MS-DFD的复杂度相当, 在较高信噪比下, 明显低于MS-DFD的复杂度. 本文算法适用于任意非对角差分酉空时星座, 然而, 如何利用特定星座的结构特点来进一步降低算法复杂度有待继续研究.

参考文献

1 Hochwald B M, Sweldens W. Differential unitary space-time modulation. IEEE Trans Commun, 2000, 48(12): 2041–2052
2 Hughes B L. Differential space-time modulation. IEEE Trans Inf Theor, 2000, 46(11): 2567–2578
3 Bhukania B, Schniter P. Multiple-symbol detection of differential unitary space-time modulation in fast-fading channels with known correlation. In: Proceedings of CISS-02. NJ: Princeton University Press, 2002. 1–6
4 Chiavaccini E, Vitertta G M. Further results on differential space-time modulation. IEEE Trans Commun, 2003, 51(7): 1093–1101
5 Pauli V, Lampe L. Multiple-symbol differential sphere decoding for unitary space-time modulation. In: Proceedings of IEEE Global Telecommunications Conference. Vol 1–6. NY: IEEE Press, 2005. 1630–1635
6 Ling C, Li K H, Kot A C. On decision-feedback detection of differential space-time modulation in continuous fading. IEEE Trans Commun, 2004, 52(10): 1613–1617
7 Ling C, Li K H, Kot A C. Noncoherent sequence detection of differential space-time modulation. IEEE Trans Inform Theor, 2003, 49(10): 2727–2734
8 Su K. Efficient Maximum Likelihood Detection for Communication over MIMO Channels. Technical Report. University of Cambridge. 2005
9 Mattias Patzald. 陈伟, 译. 移动衰落信道. 北京: 电子工业出版社, 2009. 29–30
10 张尧庭, 方开泰. 多元统计分析引论. 北京: 科学出版社, 1999. 94
11 Damen M O, Gamal H El, Caire G. On maximum-likelihood detection and the search for the closest lattice point. IEEE Trans Inf Theor, 2003, 49(10): 2389–2402
12 Li Y, Wang X, Wei J B. Multiple symbol differential approximate maximum-likelihood detection for unitary space-time modulation in continuous fading. In: Proceedings of ICCS. New York: IEEE Press, 2006. 6–11
13 李颖, 魏急波. 裁减自动球形译码算法与性能分析. 通信学报, 2007, 28(5): 49–54
14 Key S M. 罗鹏飞, 张文明, 译. 统计信号处理基础 – 估计与检测理论. 北京: 电子工业出版社, 2003. 479–480
15 Divsalar D, Simon M K. Multiple-symbol differential detection of MPSK. IEEE Trans Commun, 1990, 38(5): 300–308
16 Biglieri E, Caire G, Tarcco G, et al. Simple method for evaluating error probabilities. Electron Lett, 1996, 32(2): 191–192
17 Ho P, Fung D. Error performance of multiple-symbol differential detection of PSK signals transmitted over correlated Rayleigh fading channels. IEEE Trans Commun, 1992, 40(10): 1566–1569
18 Pauli V, Lampe L. Tree-search multiple-symbol differential decoding for unitary space-time modulation. IEEE Trans Commun, 2007, 55(8): 1567–1576
19 Alamouti S M. A simple transmit diversity technique for wireless communications. IEEE J Sel Area Comm, 1998, 16(8): 1451–1458
20 Vielmon A, Li Y, Barry J R. Performance of Alamouti transmit diversity over time-varying Rayleigh-fading channels. IEEE Trans Wirel Commun, 2004, 3(9): 1369–1373