

文章编号: 1001-2486(2008)06-0063-05

基于 ONPC 体系结构的软输出球形解码器的 VLSI 实现*

方兴, 陈书明

(国防科技大学 计算机学院, 湖南 长沙 410073)

摘要: 由于球形检测算法的高计算复杂度, 能提供软输出信息给后续纠错解码器的球形解码器的 VLSI 实现是一项挑战。提出了一种基于 ONPC 体系结构的深度优先搜索的软输出球形解码器的 VLSI 解决方案。对于采用 4×4 天线配置和 64-QAM 调制的 MIMO 系统, 在 17.7dB 的信噪比下, 所提出解决方案在 0.13- μm CMOS 工艺下能提供 14Mbps 的吞吐率, 面积开销约为 4.1mm^2 , 并且 BER 性能优于幸存路径数为 256 的基于宽度优先搜索的 K-best 算法。

关键词: 多输入多输出; 球形解码; 空分复用; VLSI

中图分类号: TN492 **文献标识码:** A

VLSI Implementation of Soft-output Sphere Detector Based on ONPC Architecture

FANG Xing, CHEN Shu-ming

(College of Computer, National Univ. of Defense Technology, Changsha 410073, China)

Abstract: The efficient high-speed VLSI implementations of sphere detector providing soft information for a subsequent error correcting decoder are not trivial due to their high computational complexity. Therefore, a VLSI solution for soft-output sphere detector based on depth-first tree search and ONPC architecture is presented. In a 0.13- μm CMOS process, the proposed detector solution can achieve 14 Mbps at a signal-to-noise ratio of 17.7dB for 4×4 MIMO system with 64-QAM using only approximately 4.1mm^2 , and realize a better BER performance than K-best algorithm with 256 survivor paths based on depth-first search.

Key words: multiple-input multiple-output (MIMO); sphere detector; spatial multiplexing; VLSI

球形解码器是一种高效的极大似然率 MIMO 检测算法^[1], 可以分为软输出球形解码器和硬输出球形解码器。由于无线通信系统采用的纠错码, 如 Turbo 码、LDPC 码和卷积码等, 通常需要软输入信息来进行纠错译码, 因此无线通信系统更倾向于采用软输出球形解码器, 代价是更高的计算复杂性^[2]。

两种主要的球形解码算法被广泛研究: 基于深度优先搜索的解码器^[3-5]和采用著名 K-best 算法的基于宽度优先搜索的解码器^[2,6-9]。对于 4×4 16-QAM 调制的 MIMO 系统, 在高信噪比的情况下, 基于 ONPC(One-Node-Per-Cycle) 体系结构和深度优先搜索的硬输出球形解码器^[3]获得了比对应宽度优先搜索球形解码器^[6-7]高得多的吞吐率和更好的 BER 性能。然而, 对于软输出解码器, 由于可以通过深度流水来获得较高的频率, 最近的研究倾向于基于宽度优先搜索的球形解码器^[8]。而据我们了解, 目前没有文献针对基于深度优先搜索的软输出球形解码器进行研究。

本文提出的基于 ONPC 体系结构的软输出球形解码器的解决方案较好地解决了幸存路径排序和搜索半径的动态调整问题。为了验证所提出的解决方案, 针对采用 64-QAM 调制的 4×4 MIMO 系统设计了解输出深度优先搜索球形解码器。

1 球形解码算法

本文只考虑采用空分复用的 MIMO 系统, 如图 1 所示。在采用 M_T 个发送天线和 M_R 个接收天线的

* 收稿日期: 2008-07-04

基金项目: 国家“863”计划资助项目(2007AA01Z108)

作者简介: 方兴(1980-), 男, 博士生。

MIMO 系统中, M_R 的接收信号向量由如下公式给出: $y = Hs + n$ 。这里, H 表示 $M_R \times M_T$ 信道矩阵, s 是 M_T 维传输向量, n 是 M_R 维的复高斯加性白噪声, 平均值为 0, 方差为 $N_0/2$ 。本文假设接收端对信道矩阵已知, 采用 Rayleigh 信道模型来对信道矩阵进行仿真。

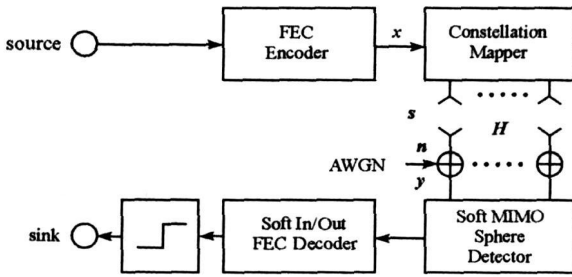


图1 MIMO系统模型

Fig. 1 Error correction coded MIMO system model

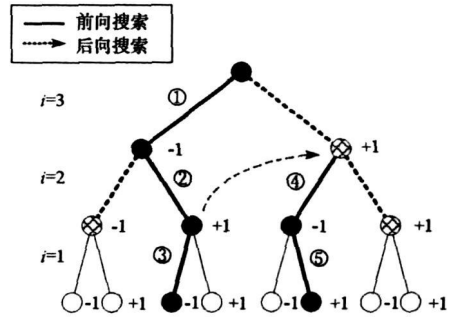


图2 深度优先搜索示意图

Fig. 2 Depth-first tree search example

根据最大似然原则, 传输信号的硬输出估计由如下公式给出: $\hat{s} = \underset{s \in \Omega^{M_T}}{\operatorname{argmin}} \|y - Hs\|^2$ 。球形解码器通过将搜索范围限制在半径为 r 的超球内以减少需要考虑的候选向量符号数。相应的不等式称为球形约束: $d(s) = \|y - Hs\|^2 < r^2$ 。在采用标准的QR分解时, 信道矩阵 H 可以分解成 $H = QR$, 其中 $QQ^H = I$, 并且 R 是上三角矩阵, $d(s)$ 的计算可以简化为: $d(s) = \sum_{i=M_T}^1 \left| \hat{y}_i - \sum_{j=i}^{M_T} R_{ij}s_j \right|^2$, 并且 $\hat{y} = Q^H y$ 。解码器可以等效为深度为 M_T , 分支数为 Ω 的树搜索过程。图2给出了BPSK调制的 2×2 MIMO系统树搜索的示意图。

采用 Jacobian 对数算法, 发送向量 s 的第 i 位的对数似然率通过如下的公式进行计算:

$$L(x_i | y) \approx \frac{1}{N_0} \left(\max_{x_i = +1} \{-d(s)\} - \max_{x_i = -1} \{-d(s)\} \right) \quad (1)$$

在每一搜索层, 部分欧氏距的计算公式为: $T_i = T_{i+1} + |e_i|^2$ 。这里, $|e_i|^2 = |b_{i+1} - R_{ii}s_i|^2$, $b_{i+1} = \hat{y}_i - \sum_{j=i+1}^{M_T} R_{ij}s_j$ 。在每一搜索层, 搜索器都选择具有最小部分欧氏距的节点进行扩展, 部分欧氏距大于半径约束的平方 ($T_i > r^2$) 的节点进行剪枝。

2 软输出解码器设计

根据第1节的描述, 一个节点的所有子节点的部分欧氏距将并行计算, 然后通过一个排序过程来选择具有最小部分欧氏距的节点继续。当抵达叶节点时, 需要另外一个排序过程来选择具有最小路径开销的幸存路径, 从而保证剪枝的有效性。然而, 由于排序是一个串行的操作, 并且包含大量的数据移动, 因此会带来严重的路径延迟和功耗问题^[8]。这样, 就面临如下两个问题: (1) 当幸存路径的数目 M 比较大时, 幸存路径的选择和排序将带来非常大的延迟和严重的面积开销; (2) 由于顺序性, 所有子节点的严格排序也会引起较大的延迟和非常大的面积开销。

2.1 幸存路径策略

为了解决第一个问题, 我们采用了如下策略: 所有具有相同父节点的叶节点中, 最多只有一个叶节点被保留为幸存路径, 以计算 LLR。基于这一策略, 幸存路径的排序可以由如下的体系结构高效实现, 我们称这种体系结构为部分移位寄存器链, 如图3所示。部分移位寄存器链中寄存器的初始值设为无穷大, 在抵达叶节点时, 采用如下策略来对其中的寄存器进行更新:

$$u_i = \begin{cases} u_{i-1}, & u_i > d_0(s) \ \&\& \ u_{i-1} > d_0(s) \\ d_0(s), & u_i > d_0(s) \ \&\& \ u_{i-1} \leq d_0(s) \\ u_i, & d_0(s) \geq u_i \end{cases}$$

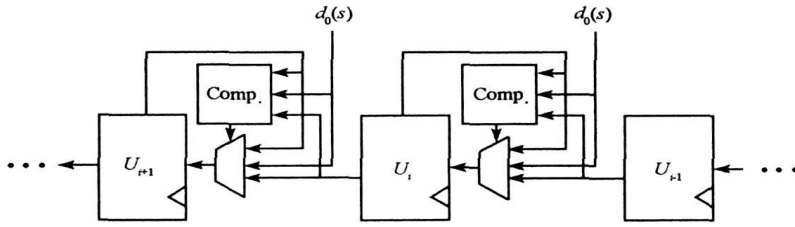


图 3 部分移位寄存器链

Fig. 3 Partial Shift Register List

这里, $d_0(s)$ 是具有相同父节点的叶节点的部分欧氏距的最小值。由于部分移位寄存器链保持了幸存路径的顺序, 因此当前半径缩减可以通过下列的办法高效实现:

$$r = \begin{cases} u_{M-1}, & u_{M-1} \leq d_0(s) \\ d_0(s), & u_{M-1} > d_0(s) \end{cases} \&\& u_{M-2} \leq d_0(s) \\ \begin{cases} u_{M-2}, & d_0(s) < u_{M-2} \end{cases}$$

这里, u_{M-1} 和 u_{M-2} 是更新前部分移位寄存器链的前两个寄存器的值。通过这一策略, 幸存路径排序和半径调整所引入的延迟相当于两个比较器和两个多路选择器, 大大小于排序策略所带来的延迟。

2.2 改进的 PSK 遍历策略

PSK 遍历策略的基本思想是: 对于 QAM 调制, 所有调制点位于 P_Q 个不同的以原点为中心的圆上, 在同样的圆上, 期望子节点与具有最小的部分欧氏距增量 $|b_{i+1} - R_{ii}|^2$ 的点相对应^[3, 8, 10]。当 R_{ii} 被假设为正实数时, 期望子节点通过如下的公式获得: $s_i^{(0)} = \underset{s_i \in \Omega}{\operatorname{argmin}} | \operatorname{arc}(b_{i+1}) - \operatorname{arc}(s_i) |$, 其中, $\operatorname{arc}(\cdot)$ 表示复数的相位。

然而, 对于复数星阵图来说, 如果 R_{ii} 是复数, 那么复数星阵图将旋转一个角度, 因此可能导致上述公式所获得的点并非最近点。为了解决这一问题, 将公式修正为: $s_i^{(0)} = \underset{s_i \in \Omega}{\operatorname{argmin}} | \operatorname{arc}(b_{i+1} R_{ii}^*) - \operatorname{arc}(s_i) |$ 。

3 VLSI 实现

3.1 体系结构

由于本文所提出的解决方案基于文献[3]中所提出的 ONPC 体系结构, 因此从简要回顾 ONPC 体系结构开始我们的讨论。最初的 ONPC 体系结构是针对硬输出球形解码器提出的, 主要思想是: 每拍访问一个新的节点, 并且不重复访问相同的节点。两个主要部件如下:

- (1) 度量计算部件(MCU)。MCU 负责树搜索的前向迭代过程, 通过迫零选择算法或直接 QAM 遍历算法确定 SE 遍历的起始点, 如果球形约束满足, 那么搜索继续前进到树的下一层; 否则前向搜索中止, 并返回到前一层。
- (2) 度量遍历部件(MEU)。MEU 和 MCU 并行工作, 负责

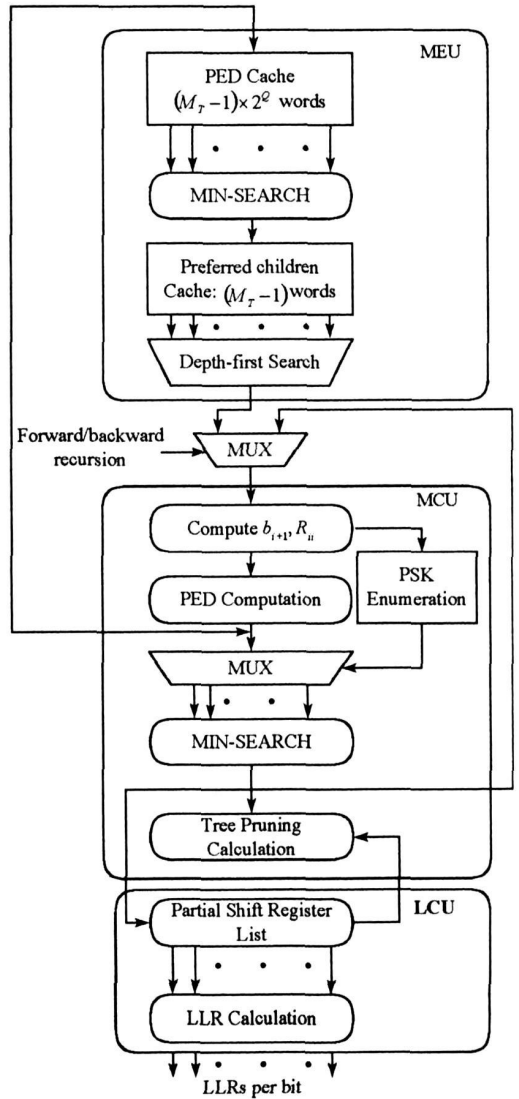


图 4 软输出球形解码器的模块图

Fig. 4 Block diagram of the soft-output sphere detector

后向迭代。MEU 保存介于根节点和当前 MCU 访问节点之间的期望子节点的列表。当 MCU 的前向搜索中止时,MEU 以深度优先搜索的方式在下一拍提供一个新的父节点给 MCU。当期望子节点的列表为空时,搜索中止。

软输出球形解码器体系结构增加了 LLR 计算部件,并对 MCU 体系结构进行了相应的改变,如图 4 所示。(1)MCU 体系结构。为了进一步减小关键路径的延迟,在我们的实现中,PSK 遍历和 PED 计算是通过并行 PSK 功能部件进行的,PSK 遍历部件负责从每个子集的 PED 中选择最小的 PED。(2)LLR 计算部件(LCU)。LCU 部件维护幸存路径列表,如图 3 所示。当抵达叶节点时,幸存路径列表在下一个周期被更新,并且幸存路径列表中的头两项被反馈给 MCU 以确定当前的搜索半径。当搜索结束时,根据式(1)计算每个比特的 LLR。这一过程可以通过流水的方式进行,并且不会位于关键路径之上。

3.2 BER 性能和吞吐率

通过 Matlab 模拟来考察所提出的解决方案的 BER 性能。系统配置采用 Turbo 编码的 64-QAM 调制的 4×4 MIMO 系统。Turbo 码的码率为 $1/3$, 帧长为 256, 生成多项式为 $g_0(D) = 1 + D^2 + D^3$ 和 $g_1(D) = 1 + D + D^3$ 。Turbo 码解码器的迭代次数固定为 5, 球形解码器的幸存路径数设为 16。采用文献[8, 10]中关于 SNR 的定义:

$$\frac{E_b}{N_0} \Big|_{dB} = \frac{E_s}{N_0} \Big|_{dB} + 10 \lg \frac{M_r}{RM_t q}$$

为了进行比较,我们也实现了幸存路径数为 256 的 K-best 软输出球形解码器和预设搜索半径为 2σ 的穷尽搜索解码器^[9]。三者的 BER 性能比较如图 5 所示。不同 SNR 下软输出球形解码器的吞吐率如图 6 所示。所提出的解决方案的 BER 性能大大优于幸存路径数为 256 的 K-best 算法,接近预设搜索半径的穷尽搜索解码器。

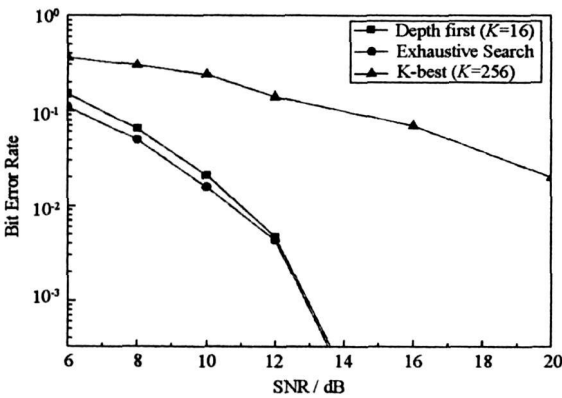


图 5 BER 性能比较
Fig. 5 Comparison of BER performance

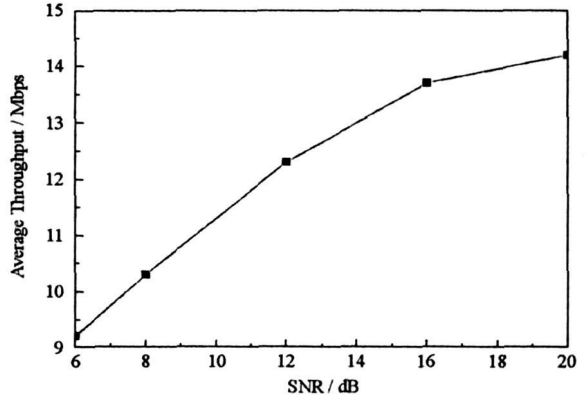


图 6 不同 SNR 下球形解码器吞吐率
Fig. 6 The average throughput and chip micrograph

3.3 ASIC 实现开销

为了验证硬件实现的有效性,我们在 0.13-μm CMOS 工艺下采用标准单元库实现了所提出的解决方案,路径度量采用 8bit 来表示,矩阵 R 和 s 中的每一项分别采用 15bit 和 16bit 来表示,实现开销如表 1 所示。

表 1 球形解码器实现开销

Tab. 1 Implementation metrics of the proposed detector

V_{DD}	f_{clk}	面积	逻辑门
1.3 V	87MHz	4.1mm ²	486k

表 2 给出了所提出的解决方案与其他典型解决方案的比较。与基于宽度优先搜索的软输出球形解码器^[8]相比,由于需要保存介于根节点和当前访问节点之间的所有期望子节点的列表,本文所提出解决方案具有较高的面积开销,但是在较高的 BER 性能下获得了更高的平均吞吐率。与文献[3]所提出的

两种硬输出球形解码器实现 ASIC iv 和 ASIC ㊟相比, 本文所提出的解决方案的面积开销增加, 而吞吐率却有明显的下降。这是由如下三个方面的原因造成的: (1) 与硬输出球形解码器相比, 软输出球形解码器不仅增加了计算 LLR 的硬件开销, 而且由于需要保留多个幸存路径以计算 LLR, 导致了搜索半径收敛速度和吞吐率的下降; (2) 所采用的调制方式不同, 与 16-QAM 的调制方式相比, 64-QAM 调制方式下, 每一个搜索步的候选路径数增加了 4 倍, 带来了硬件复杂性的增加; (3) ASIC ㊟采用了改进的球形解码算法, 以 l^∞ 范数来代替公式 l^2 范数的计算, 虽然获得了面积开销的减小和吞吐率的提高, 但 BER 性能有所下降。

表 2 与其他球形解码器 VLSI 实现的比较

Tab. 2 Comparison of VLSI implementations for MIMO detection

	ASIC iv ^[3]	ASIC ㊟ ^[3]	文献[8]	本文
天线数			4 × 4	
调制	16-QAM	16-QAM	64-QAM	64-QAM
BER 性能	ML	Close to ML	Close to ML	ML
算法	Depth-first	Depth-first	Relaxed K-best	Depth-first
硬/软输出	Hard-output	Hard-output	Soft-output	Soft-output
面积(逻辑门数)	117k	50k	280k	486k
时钟频率	51MHz	71MHz	270MHz	87MHz
吞吐率	73Mbps @ SNR= 20dB	169Mbps @ SNR= 20dB	8.57Mbps @ SNR= 17.7dB	14Mbps @ SNR= 17.7dB

4 总结

本文提出了一种面向 VLSI 实现的基于 ONPC 体系结构和深度优先搜索的软输出球形解码器解决方案, 对于采用 64-QAM 调制的 4 × 4 MIMO 系统, 在 0.13- μm 的 CMOS 工艺和 17.7dB 的信噪比下, 解码器可达到 14Mbps 的吞吐率, 优于基于宽度优先搜索的软输出球形解码器, 并且 BER 性能优于基于宽度优先搜索的 K-best 算法。

参考文献:

- [1] Hassibi B, Vikalo H. On the Sphere-decoding Algorithm iv. Expected Complexity[J]. IEEE Transactions Signal Processing, 2005, 53: 2806-2818.
- [2] Widdup B, Woodward G, Knagge G. A Highly-parallel VLSI Architecture for a List Sphere Detector[C]//Proceedings of the IEEE International Conference on Communications, 2004: 2720-2725.
- [3] Bug A, et al. VLSI Implementation of MIMO Detection Using the Sphere Decoding Algorithm[J]. IEEE Journal of Solid-state Circuits, 2005, 40(7): 1566-1577.
- [4] Amiri K, Cavallaro J R. FPGA Implementation of Dynamic Threshold Sphere Detection for MIMO Systems[C]//Proc. 2006 Asilomar Conference on Signals, Systems and Computers, 2006: 94-98.
- [5] Garrett D, et al. Silicon Complexity for Maximum Likelihood MIMO Detection Using Spherical Decoding[J]. IEEE Journal of Solid-state Circuits, 2004, 39: 1544-1552.
- [6] Wong K, et al. A VLSI Architecture of a K-best Lattice Decoding Algorithm for MIMO Channels[C]//Proc. IEEE ISCAS'02, 2002, 3: 273-276.
- [7] Guo Z, Nilsson P. VLSI Architecture of the Schnorr-eudner Decoder for MIMO Systems[C]//Proc. IEEE CAS Symposium on Emerging Technologies, 2004: 65-68.
- [8] Chen S, Zhang T, Xin Y. Relaxed K-best MIMO Signal Detector Design and VLSI Implementation[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2007, 15(3): 328-337.
- [9] Radosavljevic P, Cavallaro J R. Soft Sphere Detection with Bound Search for High-throughput MIMO Receivers[C]//Proc. 2006 Asilomar Conference on Signals, Systems and Computers, 2006: 1175-1179.
- [10] Hochwald B M, Brink S T. Achieving Near-capacity on a Multiple-antenna Channel[J]. IEEE Trans. Commun., 2003, 51(3): 389-399.