

Detecção de Baixa Complexidade para Constelações Multidimensionais com Diversidade Completa Baseadas na Abordagem Combinatorial

Juliana C. Inácio, Bruno F. da Silva, Bartolomeu F. Uchôa-Filho e Didier Le Ruyet

Resumo—Uma abordagem combinatorial foi recentemente proposta para o projeto de constelações de sinais multidimensionais para canais com desvanecimento. Mesmo sendo flexível e garantindo diversidade de espaço de sinais completa, a técnica adota a detecção ótima exponencialmente complexa de máxima verossimilhança (ML). Neste trabalho, propomos dois algoritmos de detecção de baixa complexidade, um sub-ótimo, baseado em listas, e o outro ótimo, baseado numa simplificação do ML. Ambos os algoritmos apresentam um bom compromisso entre desempenho e complexidade. Resultados de simulação mostram que o desempenho ótimo ou próximo do ótimo pode ser alcançado com redução de até 98% da complexidade de detecção.

Palavras-Chave—Detecção em lista, diversidade de espaço de sinais, diversidade completa.

Abstract—A combinatorial approach was recently proposed for the design of multidimensional signal constellations for fading channels. Albeit flexible and guaranteeing full signal space diversity, the technique adopts the exponentially complex optimal maximum likelihood (ML) detection. In this work, we propose two low complexity detection algorithms, a sub-optimal one, based on lists, and an optimal one, based on a simplification of ML. Both algorithms have a good trade-off between performance and complexity. Simulation results show that optimal or close to optimal performance can be achieved with up to 98% reduction in detection complexity.

Keywords— Full diversity, list detection, signal-space-diversity.

I. INTRODUÇÃO

Diversidade de espaço de sinais (SSD) é uma diversidade intrínseca em constelações multidimensionais projetadas para melhorar o desempenho de erro em canais com desvanecimento sem o uso explícito de codificação de canal [1]. SSD é normalmente obtida por rotação [1, 2], pré-codificação [3], ou ambas operações [4, 5] em uma constelação de sinal multidimensional. Diversidade completa é obtida quando essas operações fazem com que as projeções de todos os pontos da constelação em cada dimensão sejam todas distintas [1]. Isso permite que o detector recupere o ponto transmitido, mesmo no pior cenário, em que todas, exceto uma das dimensões, são afetadas por um desvanecimento profundo.

Juliana C. Inácio e Bartolomeu F. Uchôa-Filho, GPqCom/LCS/EEL, Universidade Federal de Santa Catarina (UFSC), Florianópolis-SC, e-mail: julianacamiloinacio@gmail.com e uchoa@eel.ufsc.br; Bruno F. da Silva, Instituto-Federal Sul-rio-grandense (IFSul), Saporanga-RS, e-mail: brunosilva@ifsul.edu.br; Didier Le Ruyet, CEDRIC/LAETITIA, Conservatoire National des Arts et Métiers (CNAM), Paris, França, e-mail: didier.le_ruyet@cnam.fr. Este trabalho foi parcialmente financiado pela CAPES, programa PNPd (88887.357873/2019-00), e pelo CNPq.

Em um trabalho recente [6], constelações multidimensionais com diversidade completa foram projetadas de uma maneira alternativa, chamada de *abordagem combinatorial*, na qual uma grade fixa de coordenadas rotuladas é usada em cada dimensão. A constelação é obtida através de um conjunto de permutações dos rótulos apropriadamente projetado. Essa abordagem apresenta flexibilidade de projeto e um bom compromisso entre desempenho e complexidade.

Para obter o ganho de SSD, a detecção do ponto M -dimensional deve considerar conjuntamente as componentes de cada dimensão. Na literatura, inclusive em [1–6], a maioria dos trabalhos usa a detecção de máxima verossimilhança (ML), cuja complexidade aumenta exponencialmente com o número de bits transmitidos. Portanto, para grandes constelações, o uso desta técnica torna-se inviável.

Em [7], os autores propuseram um método diferente de explorar SSD denominado modulação conjunta, baseado nos códigos de distância máxima separável (MDS). E para superar a restrição de complexidade do ML, os autores também propuseram uma decodificação sub-ótima de seleção de canais de diversidade (DCS). No entanto, para constelações com diversidade completa, a ordem de complexidade da decodificação DCS é equivalente àquela da detecção ML.

Métodos de detecção baseados em descorrelação, tais como erro quadrático médio mínimo (MMSE), podem ser adotados em esquemas SSD [8], ao custo de uma perda significativa de diversidade [9]. Um desmapeador de esfera foi proposto em [10], com o qual um bom desempenho é alcançado quando usado com constelações rotacionadas apropriadamente projetadas para reduzir a complexidade de detecção. Em [11], para constelações rotacionadas associadas a códigos convolucionais, os autores propuseram dois métodos de decodificação iterativa baseados em aproximação Gaussiana para obter um bom compromisso entre desempenho e complexidade.

Em um contexto diferente, a decodificação em lista foi proposta em [12, 13]. Nesta abordagem, uma lista de palavras código candidatas é obtida na saída do decodificador, na qual é esperada que esteja a palavra código transmitida. A abordagem da decodificação em lista tem sido extensivamente estudada na literatura e é amplamente utilizada em esquemas com códigos polares [14], por exemplo.

Neste trabalho, propomos dois algoritmos de detecção de baixa complexidade para constelações projetadas a partir da abordagem combinatorial em [6]. O primeiro algoritmo é baseado na detecção em listas e produz desempenho próximo ao desempenho ótimo, com uma redução de complexidade

significativa, conforme mostrado através de simulações computacionais. O segundo algoritmo é uma versão simplificada da detecção ML, e apresenta o mesmo desempenho ótimo com redução de complexidade de detecção.

Esta proposta foi motivada pelos resultados promissores e pela flexibilidade de projeto apresentados em [6], além do fato de que na literatura existem poucas propostas de redução da complexidade de detecção de constelações com diversidade completa sem associação com códigos corretores de erros.

O restante do trabalho está organizado como segue. Na Seção II, o modelo do sistema é descrito. Os algoritmos de detecção propostos são apresentados na Seção III. Os resultados numéricos são apresentados na Seção IV. Por fim, a Seção V apresenta a conclusão do trabalho.

II. MODELO DO SISTEMA

De acordo com a abordagem combinatorial recentemente proposta em [6], para construir uma constelação M -dimensional de N pontos com diversidade completa, primeiro é necessário preparar uma grade unidimensional fixa de N coordenadas que representam as projeções distintas dos sinais na primeira dimensão da constelação. Em [6], consideraram-se grades com coordenadas dispostas de maneira uniforme ou Gaussiana (coordenadas mais próximas umas das outras no centro da grade, e mais separadas nas extremidades). Porém, neste trabalho, a grade uniforme [6, Seção IV.A] foi escolhida devido a sua regularidade e simplicidade, necessárias para os algoritmos de detecção aqui propostos. A grade uniforme \mathbf{u} pode ser descrita como

$$\mathbf{u} = \left[-\frac{(N-1)\epsilon}{2}, -\frac{(N-3)\epsilon}{2}, \dots, \frac{(N-3)\epsilon}{2}, \frac{(N-1)\epsilon}{2} \right], \\ = [u^{(1)}, u^{(2)}, \dots, u^{(N-1)}, u^{(N)}]. \quad (1)$$

em que a constante ϵ é escolhida de tal maneira que a grade tenha energia média unitária, a saber $\epsilon = (12/(N^2 - 1))^{1/2}$.

Na formação da primeira dimensão da constelação, as coordenadas da grade são rotuladas sequencialmente, por exemplo, de 1 a N . O rotulamento das demais dimensões é obtido através de $M-1$ permutações distintas dos rótulos da primeira dimensão. A constelação M -dimensional produzida possui diversidade completa. As permutações podem ser obtidas através de diferentes algoritmos de acordo com [6, Seção V]. Neste trabalho, escolhemos utilizar a grade uniforme com permutações algébricas (UGAP) [6] para obter a constelação com diversidade completa \mathcal{C} .

Na transmissão, é realizado um mapeamento de cada sequência de $\log_2 N$ bits de informação em um ponto M -dimensional da constelação $\mathcal{C} = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}\}$. Note que cada ponto $\mathbf{x}^{(i)}$ de \mathcal{C} é um vetor real cujos elementos são formados por coordenadas da grade uniforme, com as restrições impostas pela construção UGAP em [6], que garantem bom desempenho. Após a transmissão, o vetor recebido no destino é descrito pelo modelo de canal real apresentado em [4]:

$$\mathbf{y} = \text{diag}(\mathbf{h})\mathbf{x} + \mathbf{w}, \quad (2)$$

em que $\mathbf{x} = [x_1, \dots, x_M] \in \mathbb{R}^{M \times 1}$ e $\mathbf{x} \in \mathcal{C}$ é o ponto transmitido, $\mathbf{y} = [y_1, \dots, y_M] \in \mathbb{R}^{M \times 1}$ é o ponto recebido,

$\text{diag}(\mathbf{h})$ é a matriz diagonal cujos elementos da diagonal são obtidos a partir de $\mathbf{h} = [h_1, \dots, h_M] \in \mathbb{R}^{M \times 1}$ que é o vetor de coeficientes reais independentes e identicamente distribuídos (i.i.d.) de desvanecimento Rayleigh (os quais são obtidos tomando-se o módulo de uma variável aleatória Gaussiana de média zero com variância 1/2) e $\mathbf{w} \in \mathbb{R}^{M \times 1}$ é o vetor de ruído Gaussiano, com média zero e matriz de covariância $\frac{N_0}{2} \mathbf{I}_M$, em que \mathbf{I}_M denota a matriz identidade M -dimensional.

O símbolo resultante da detecção ML é dado por

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}' \in \mathcal{C}} \|\mathbf{y} - \text{diag}(\mathbf{h})\mathbf{x}'\|^2. \quad (3)$$

Tal detecção apresenta alta complexidade quando o número de pontos da constelação aumenta.

III. ALGORITMOS DE BAIXA COMPLEXIDADE DE DETECÇÃO PROPOSTOS

Nesta seção, são apresentados dois algoritmos de detecção de baixa complexidade para constelações baseadas na abordagem combinatorial com a grade uniforme [6]. Para ambos os algoritmos é necessário o conhecimento instantâneo do estado do canal no receptor. O primeiro algoritmo é uma adaptação do método de decodificação em lista proposto em [12, 13]. A metodologia de detecção desse algoritmo considera todas as M dimensões do ponto da constelação conjuntamente e simultaneamente, porém, para apenas uma lista de pontos candidatos.

O segundo algoritmo é uma versão simplificada da detecção de máxima verossimilhança. Diferentemente da metodologia proposta no primeiro algoritmo, a detecção é realizada a cada dimensão, ou seja, alguns pontos terão apenas uma dimensão testada (e serão descartados), alguns pontos poderão testar até M dimensões, enquanto outros pontos serão descartados sem a realização de testes.

A. Algoritmo de Detecção em Lista Proposto

Para a detecção em lista, a primeira etapa é criar M listas básicas de pontos candidatos, cada uma associada com uma dimensão. Seja \mathbf{y} o vetor real recebido. Então, para $d = 1, \dots, M$, a d -ésima lista básica, denotada por \mathcal{L}_d , consiste dos L pontos da constelação cujas componentes na dimensão d sejam as mais próximas de y_d em termos de distância Euclidiana. Observe que esta lista pode ser criada sem realizar nenhum cálculo de distância, arredondando-se¹ primeiro y_d para o ponto mais próximo da grade, e então selecionando os $L-i$ (resp., i) pontos da grade à esquerda (resp., à direita) de y_d , em que i deve ser o mais próximo possível de $L/2$ (dependendo de y_d estar no centro da grade ou próximo de uma de suas bordas). A criação das listas básicas está ilustrada na Fig. 1 para uma constelação tridimensional de $N = 8$ pontos, em que o tamanho da lista é $L = 3$. As três listas básicas são: $\mathcal{L}_1 = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \mathbf{x}^{(3)}\}$, $\mathcal{L}_2 = \{\mathbf{x}^{(1)}, \mathbf{x}^{(3)}, \mathbf{x}^{(5)}\}$ and $\mathcal{L}_3 = \{\mathbf{x}^{(7)}, \mathbf{x}^{(2)}, \mathbf{x}^{(6)}\}$.

O algoritmo de detecção em lista proposto neste trabalho é chamado algoritmo LD e está descrito no Algoritmo 1. A

¹É necessário que seja feita uma equalização y_d/h_d para que a operação de arredondamento seja bem sucedida.

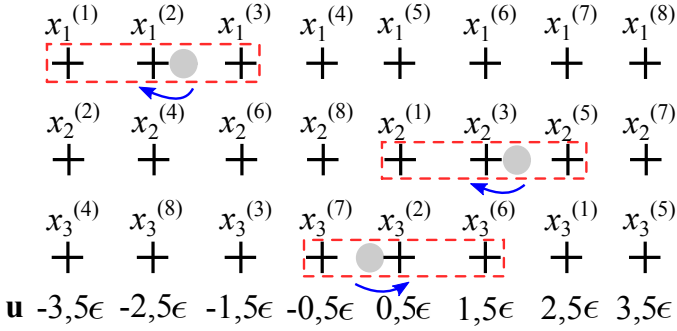


Fig. 1. Exemplo da construção das listas básicas para a constelação UGAP com $M = 3$ dimensões e $N = 8$ pontos, projetada de acordo com [6]. O tamanho da lista é $L = 3$. Note que a grade u é idêntica em todas as dimensões. Legenda de símbolos: símbolo de adição significa coordenada da grade, círculo cheio representa y , as setas representam a operação de arredondamento e os retângulos tracejados representam as listas básicas.

partir das listas básicas, $\mathcal{L}_1, \dots, \mathcal{L}_M$, o algoritmo LD constrói as listas \mathcal{K}_i , para $i = 0, \dots, M$. A lista \mathcal{K}_i contém os pontos da constelação que aparecem em exatamente i listas básicas (independentemente de quais sejam tais listas). Para o nosso exemplo na Fig. 1, temos $\mathcal{K}_0 = \{\mathbf{x}^{(4)}, \mathbf{x}^{(8)}\}$, $\mathcal{K}_1 = \{\mathbf{x}^{(5)}, \mathbf{x}^{(6)}, \mathbf{x}^{(7)}\}$, $\mathcal{K}_2 = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \mathbf{x}^{(3)}\}$ e $\mathcal{K}_3 = \emptyset$.

Algoritmo 1 Algoritmo de Detecção em Lista

- 1: **procedimento** LD($y, M, m, \mathbf{h}, \mathcal{L}_1, \dots, \mathcal{L}_M$)
- 2: A partir das listas básicas, construa as listas \mathcal{K}_i , $0 \leq i \leq M$
- 3: $\mathcal{U} := \mathcal{K}_M \cup \mathcal{K}_{M-1} \cup \dots \cup \mathcal{K}_{M-m+1}$
- 4: **se** $\mathcal{U} \neq \emptyset$ **então**
- 5: $\hat{\mathbf{x}} = \arg \min_{\mathbf{x}' \in \mathcal{U}} \|\mathbf{y} - \text{diag}(\mathbf{h})\mathbf{x}'\|^2$
- 6: **senão**
- 7: $\mathcal{L}_{1:M} := \mathcal{L}_1 \cup \mathcal{L}_2 \cup \dots \cup \mathcal{L}_M$
- 8: $\hat{\mathbf{x}} = \arg \min_{\mathbf{x}' \in \mathcal{L}_{1:M}} \|\mathbf{y} - \text{diag}(\mathbf{h})\mathbf{x}'\|^2$
- 9: **fim se**
- 10: **retorna** $\hat{\mathbf{x}}$
- 11: **fim procedimento**

O algoritmo LD está baseado na hipótese de que os pontos da constelação que aparecem em \mathcal{K}_i são mais prováveis de ser o ponto ML do que os pontos da constelação em $\mathcal{K}_{i'}$, se $i > i'$. Na sequência, temos a construção da lista \mathcal{U} , resultante da união de m listas \mathcal{K}_i , em que m é escolhido para obter um bom compromisso entre complexidade e desempenho. Na Fig. 1, para $m = 2$, $\mathcal{U} = \mathcal{K}_3 \cup \mathcal{K}_2 = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \mathbf{x}^{(3)}\}$. Logo, apenas os pontos candidatos contidos em \mathcal{U} seriam testados com relação à distância Euclidiana quadrática do ponto recebido. Caso $\mathcal{U} = \emptyset$, o algoritmo LD constrói a lista $\mathcal{L}_{1:M}$, resultante da união de todas as listas básicas, em que $L \leq |\mathcal{L}_{1:M}| \leq ML \leq N$. Note que $|\mathcal{L}_{1:M}| \geq |\mathcal{U}|$. Portanto, para $m \geq 2$, abre-se a oportunidade para que uma lista menor seja considerada (Passo 6 do Algoritmo 1). O valor de L também deve ser escolhido para garantir um bom compromisso entre complexidade e desempenho. Note que, a medida que N aumenta, a redução de complexidade do algoritmo LD torna-se mais significativa.

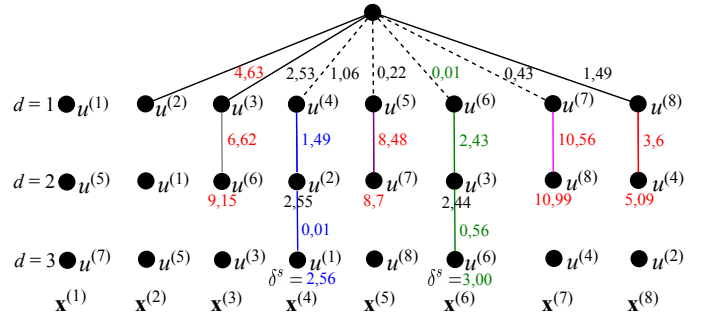


Fig. 2. Exemplo do funcionamento do algoritmo ML simplificado para a constelação UGAP com $M = 3$ e $N = 8$, projetada de acordo com [6], $\mathbf{y} = [0,75 -2,07 -0,5]$ e $\mathbf{h} = [1,29 0,77 0,39]$. Legenda de símbolos: círculos pretos são as componentes dos pontos tridimensionais, os ramos representam os cálculos de distâncias Euclidianas quadráticas realizados e os números em vermelho representam as distâncias parciais que quando somadas são superiores a δ^s em verde e azul.

B. Algoritmo de Detecção ML Simplificado Proposto

Para a detecção ML simplificada proposta, o primeiro passo é ordenar os ganhos de canal h_d do mais confiável para o menos confiável, ou seja, do maior para o menor, em que $d \in \{1, \dots, M\}$. Em seguida, encontra-se o ponto da constelação cuja primeira coordenada é a mais próxima de y_1 :

$$\mathbf{x}^{(s)} = \arg \min_{\mathbf{x}^{(i)} \in \mathcal{C}} |y_1 - h_1 x_1^{(i)}|^2. \quad (4)$$

O ponto $\mathbf{x}^{(s)}$ passa a ser uma referência para a segunda etapa do algoritmo proposto (Algoritmo 2).

Para os próximos passos do algoritmo, é conveniente introduzir a notação genérica, $\delta_d^{(i)}$, para $d = 1, \dots, M$:

$$\delta_d^{(i)} = |y_d - h_d x_d^{(i)}|^2. \quad (5)$$

Explorando-se a regularidade da grade uniforme, a busca em (4) pode ser substituída pela operação de arredondamento, como na Seção III-A, o que implica que não há custo de complexidade nessa etapa.

O funcionamento do algoritmo ML simplificado é ilustrado através da árvore na Fig. 2, para $M = 3$ dimensões e $N = 8$ pontos. A árvore é formada por um nó único no nível zero (superior) e oito nós em cada um dos seus três níveis seguintes (representando as três dimensões). Os oito nós em cada nível representam as coordenadas dos pontos da constelação na dimensão correspondente. A presença de um ramo unindo nós entre os níveis $d-1$ e d indica que o cálculo da distância entre a coordenada d do ponto correspondente, $x_d^{(i)}$, e y_d , i.e., $\delta_d^{(i)}$, foi realizado. A ausência do ramo indica que este cálculo não foi necessário. O valor de $\delta_d^{(i)}$ é indicado à direita do ramo, enquanto que o valor da distância acumulada é indicado abaixo do nó.

Na primeira etapa do algoritmo, a operação de arredondamento envolve somente a primeira dimensão (nível $d = 1$ da árvore). O ponto mais próximo de y em relação à primeira dimensão é $\mathbf{x}^{(s)} = \mathbf{x}^{(6)}$. Em seguida, a distância acumulada total entre os pontos $\mathbf{x}^{(s)}$ e y é obtida (ramos verdes): $\delta^s = \sum_{d=1}^M \delta_d^{(s)} = 3,00$.

Na segunda etapa do algoritmo ML simplificado, mais especificamente no Passo 8 do Algoritmo 2, obtemos o conjunto

\mathcal{S} como o conjunto dos pontos $\mathbf{x}^{(j)}$ cujas distâncias $\delta_1^{(j)}$ em relação a y_1 são inferiores à distância δ^s . Estes são os potenciais candidatos para o ponto de máxima verossimilhança. Note que nesta etapa podemos utilizar $\pm\delta^s$ como limitante inferior e superior para escolher os pontos $\mathbf{x}^{(j)}$ e formar o conjunto \mathcal{S} sem a realização de nenhum cálculo. Isso implica que não há custo de complexidade nessa etapa. Além disso, devido ao ordenamento das coordenadas na primeira dimensão, alguns pontos, como o ponto $\mathbf{x}^{(1)}$, podem ser eliminados sem qualquer necessidade de cálculo de distância. Nesse exemplo, $\mathcal{S} = \{\mathbf{x}^{(3)}, \mathbf{x}^{(4)}, \mathbf{x}^{(5)}, \mathbf{x}^{(7)}, \mathbf{x}^{(8)}\}$.

Na última etapa, que inicia no Passo 11 do Algoritmo 2, é realizada a escolha pelo ponto $\mathbf{x}^{(j)} \in \mathcal{S}$ que seja vizinho à esquerda ou à direita do ponto $\mathbf{x}^{(s)}$, alternadamente. Neste trabalho, foi arbitrado iniciar sempre pelo vizinho à esquerda de $\mathbf{x}^{(s)}$. Além disso, começa-se pelo ponto $\mathbf{x}^{(j)}$ vizinho de $\mathbf{x}^{(s)}$ pois este possui maiores chances de ser o ponto ML, baseando-se no mesmo princípio da decodificação de esfera, em que o ponto do reticulado mais próximo do centro da esfera provavelmente é o ponto ML. Então, inicia-se uma sequência de testes, dimensão por dimensão. Ou seja, primeiro testa-se se a soma das distâncias das duas primeiras dimensões do ponto $\mathbf{x}^{(j)}$ em relação ao ponto recebido é inferior a δ^s . Se sim, os testes continuam, adicionando-se a contribuição $\delta_d^{(j)}$ na dimensão d à distância acumulada e comparando-se este valor a δ^s , até que todas as M dimensões tenham sido consideradas. Durante esse processo, podemos ter um novo ponto $\mathbf{x}^{(s)}$ de referência. Neste caso, redefinimos o conjunto \mathcal{S} e refazemos a sequência de testes até que o conjunto \mathcal{S} esteja vazio e o ponto $\mathbf{x}^{(s)}$ seja o ponto ML estimado. Se as somas das distâncias parciais $\delta_d^{(j)}$ (ou a soma das M distâncias) for superior a δ^s , independentemente em qual teste isso ocorra, o ponto $\mathbf{x}^{(j)}$ é eliminado do conjunto \mathcal{S} .

Dando continuidade ao exemplo na Fig. 2, após a obtenção do conjunto \mathcal{S} , o primeiro ponto $\mathbf{x}^{(j)}$ a ser testado é $\mathbf{x}^{(5)}$. Como a soma $0,22 + 8,48 = 8,7 > 3,00$, $\mathbf{x}^{(5)}$ é eliminado do conjunto \mathcal{S} . O mesmo ocorre em relação ao ponto $\mathbf{x}^{(7)}$. Na sequência, testa-se o ponto $\mathbf{x}^{(4)}$, e a soma de suas distâncias parciais é $2,56 < \delta^s = 3,00$. Portanto, temos a definição de um novo ponto $\mathbf{x}^{(s)} = \mathbf{x}^{(4)}$. Em seguida, redefinimos $\mathcal{S} = \{\mathbf{x}^{(3)}, \mathbf{x}^{(8)}\}$. Como a soma das distâncias parciais de $\mathbf{x}^{(8)}$ e $\mathbf{x}^{(3)}$ é maior do que $\delta^s = 2,56$, ambos os pontos são eliminados do conjunto \mathcal{S} , o qual se torna um conjunto vazio. Portanto, encerra-se o processo de detecção, em que $\hat{\mathbf{x}} = \mathbf{x}^{(s)} = \mathbf{x}^{(4)}$, e o ponto ML é encontrado.

Claramente, o Algoritmo 2 produz como decisão final o ponto de máxima verossimilhança, visto que os pontos da constelação com distâncias parciais acumuladas maiores do que δ^s são eliminados, e os pontos não eliminados são confrontados com o ponto $\mathbf{x}^{(s)}$ ao longo ou ao final do algoritmo, até que não reste nenhum ponto concorrente (ao ponto $\mathbf{x}^{(s)}$). Sendo assim, o Algoritmo 2 é ótimo. Por outro lado, sua complexidade é consideravelmente inferior àquela do algoritmo ML, como será visto na próxima seção.

IV. RESULTADOS NUMÉRICOS

Nas simulações de Monte Carlo, foi adotado um canal real e independente com desvanecimento Rayleigh de 4 taps. A

Algoritmo 2 Algoritmo ML Simplificado

```

1: procedimento ML SIMPLIFICADO( $y, M, \mathbf{h}$ )
2:   Ordene as dimensões de acordo com  $h_d$ , do maior para
   o menor, onde  $d \in \{1, \dots, M\}$ .
3:   Defina  $\mathcal{S} = \mathcal{C}$ 
4:   Escolha o ponto  $\mathbf{x}^{(s)}$ , que é o ponto mais próximo de
    $y$  na dimensão  $d = 1$ , através da operação de arredonda-
   mento.
5:    $\delta^s = \sum_{d=1}^M |y_d - h_d x_d^{(s)}|^2$ 
6:    $\mathcal{A} = \mathcal{S} \setminus \mathbf{x}^{(s)}$ 
7:   Defina  $\mathcal{S} = \{\mathbf{x}^{(j)} : \mathbf{x}^{(j)} \in \mathcal{A}, \delta_1^{(j)} < \delta^s\}$ 
8:   se  $\mathcal{S} = \emptyset$  então
9:      $\hat{\mathbf{x}} = \mathbf{x}^{(s)}$ 
10:  senão
11:    Escolha o ponto  $\mathbf{x}^{(j)} \in \mathcal{S}$  vizinho à esquerda ou à
    direita (escolha alternadamente) de  $\mathbf{x}^{(s)}$ .
12:    para  $i = 2 : M$  faça
13:      se  $\delta_1^{(j)} + \delta_2^{(j)} + \dots + \delta_i^{(j)} \geq \delta^s$  então
14:        Elimine  $\mathbf{x}^{(j)}$  de  $\mathcal{S}$ 
15:      retorne para Passo 8.
16:    senão
17:      se  $i = M$  então
18:         $\mathbf{x}^{(s)} = \mathbf{x}^{(j)}$ 
19:      retorne para Passo 8.
20:    fim se
21:  fim se
22:  fim para
23: fim se
24: retorna  $\hat{\mathbf{x}}$  ▷ Ponto que minimiza  $\delta^s$ 
25: fim procedimento

```

constelação UGAP em [6] foi utilizada para $M = 4$ dimensões reais e $N = 256$ pontos, o que nos dá 8 bits por ponto da constelação, e a energia média de um ponto da constelação é $E_c = M$. A constelação UGAP com detecção ML de [6] será o nosso esquema de referência. Para o algoritmo LD, $m = 3$ foi adotado.

Na Fig. 3, os algoritmos propostos LD e ML simplificado são comparados à detecção ML [6]. Como pode ser visto, para diferentes tamanhos de lista básica L , o desempenho do LD muda significativamente. Para $L = 8$, o LD apresenta uma perda de desempenho de aproximadamente 8,3 dB para uma probabilidade de erro de 10^{-2} , enquanto que para $L = 16$ essa perda cai para 4,6 dB para uma probabilidade de erro de 10^{-3} , ambos comparados ao ML [6]. Quando $L = 32$, a perda de desempenho do LD proposto é de apenas 0,33 dB para uma probabilidade de erro de 10^{-4} , como pode ser visto no detalhe ampliado das curvas. Portanto, o valor $L = 32$ será adotado para as simulações a seguir. O desempenho do ML simplificado proposto é idêntico ao desempenho ML [6].

Na Fig. 4 é apresentada uma comparação em termos do número médio de operações com pontos flutuantes (flops). Um flop pode ser definido como uma adição, subtração, multiplicação ou divisão de dois números ponto flutuante. Comparado ao ML [6], o algoritmo ML simplificado apresenta uma complexidade que varia de acordo com a razão sinal-ruído (SNR),

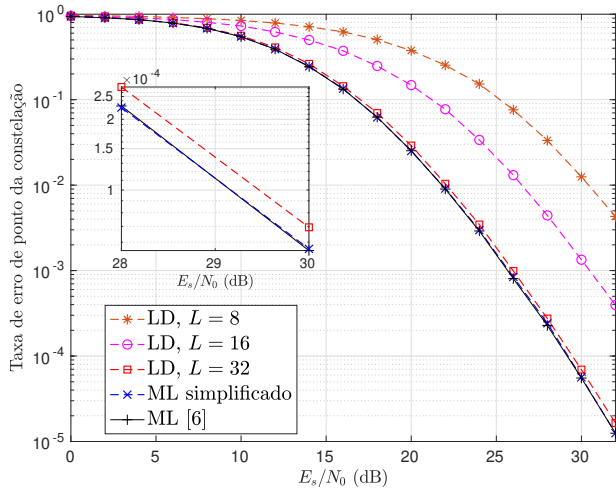


Fig. 3. Taxa de erro de ponto da constelação para a constelação UGAP com $M = 4$ e $N = 256$ com o detector de referência ML [6] e com os detectores propostos ML simplificado e LD para diferentes valores de L .

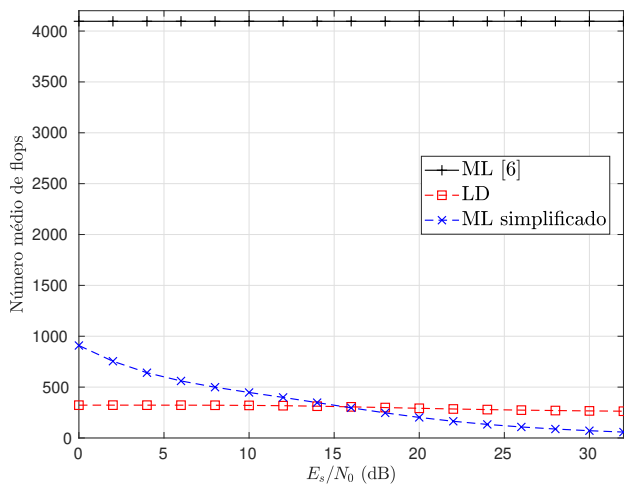


Fig. 4. Número médio de flops para a constelação UGAP com $M = 4$ e $N = 256$ com o detector de referência ML [6] e com os detectores propostos LD, para $L = 32$, e ML simplificado.

com redução do número médio de flops variando entre 77,8% e 98% aproximadamente. Essa variação de complexidade ocorre porque para regiões de alta SNR os pontos são facilmente distinguíveis. Portanto, basta testar alguns poucos pontos da constelação para que o algoritmo ML simplificado encontre o ponto ML e elimine os demais concorrentes do conjunto \mathcal{S} . O algoritmo LD proposto apresenta uma grande redução no número médio de flops, em média 92,7% aproximadamente, ao custo de uma pequena perda de desempenho. Sendo assim, o algoritmo LD proposto (Algoritmo 1) parece apresentar um excelente compromisso entre desempenho e complexidade de detecção para regiões de baixa SNR. Por outro lado, o algoritmo ML simplificado proposto (Algoritmo 2) apresenta desempenho ótimo e redução de complexidade significativa para regiões de média e alta SNR.

V. CONCLUSÕES

Neste trabalho, foram propostos dois algoritmos de detecção, um baseado em listas e outro baseado na detecção ML, para constelações projetadas a partir da abordagem combinatorial de [6] em canais com desvanecimento Rayleigh. Os resultados das simulações mostraram que o algoritmo LD proposto oferece uma redução de complexidade de 92,7% ao custo de uma perda de desempenho de 0,33 dB comparado ao ML [6]. Já o ML simplificado proposto, para um desempenho ótimo, apresenta uma redução de complexidade que varia de 77,8% a 98%, de acordo com a SNR. Portanto, ambos os algoritmos propostos apresentam um excelente compromisso entre desempenho e complexidade. Esses resultados tornam promissora a aplicação das constelações com diversidade completa em sistemas práticos futuros.

AGRADECIMENTOS

Os autores gostariam de agradecer ao Instituto Federal de Santa Catarina (IFSC) ao apoio dado para a realização deste trabalho, à CAPES, programa PNPd (no. 88887.357873/2019-00), e ao CNPq pelo financiamento parcial deste trabalho.

REFERÊNCIAS

- [1] J. Boutros, E. Viterbo, "Signal Space Diversity: A Power- and Bandwidth-Efficient Diversity Technique for the Rayleigh Fading Channel". *IEEE Trans. Inform. Theory*, vol. 44, no. 4, pp. 1453–1467, Jul. 1998.
- [2] J. Boutros, E. Viterbo, C. Rastello, J. C. Belfiore, "Good Lattices Constellations for Both Rayleigh Fading and Gaussian Channels." *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 502–518, Mar. 1996.
- [3] Z. Liu, Y. Xin, G. B. Giannakis, "Linear Constellation-Precoding for OFDM with Maximum Multipath Diversity and Coding Gains". *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 416–427, Mar. 2003.
- [4] J. C. Inácio, B. F. Uchôa-Filho, D. Le Ruyet, "Exploiting Signal Space Diversity in OFDM with Grouped Subcarriers: Going Beyond Subcarrier Index Modulation". *IEEE Wireless Commun. Lett.*, vol. 7, no. 4, pp. 650–653, Aug. 2018.
- [5] J. C. Inácio, B. F. Uchôa-Filho, D. Le Ruyet, "Diversidade de Espaço de Sinais em OFDM com Subportadoras Agrupadas: Resultados Adicionais". *XXXVI Simp. Brasileiro de Telecom. e Proc. de Sinais, SBRT 2018*, Campina Grande, PB, Brasil, Sept. 2018.
- [6] J. C. Inácio, B. F. Uchôa-Filho, D. Le Ruyet, S. Montejo-Sánchez, "Full Diversity Multidimensional Codebook Design for Fading Channels: The Combinatorial Approach". *IEEE Trans. Commun.*, vol. 68, no. 7, Apr. 2020.
- [7] Y. Shang, D. Wang, X. Xia, "Signal Space Diversity Techniques with Fast Decoding Based on MDS Codes". *IEEE Trans. Commun.*, vol. 58, no. 9, pp. 2525–2536, Sept. 2010.
- [8] G. M. Guvensen, A. O. Yilmaz, "Iterative Decision Feedback Equalization and Decoding for Rotated Multidimensional Constellations in Block Fading Channels". *IEEE 69th Vehic. Tech. Conf., VTC Spring 2009*, Barcelona, Spain, Apr. 2009.
- [9] M. L. McCloud, "Analysis and Design of Short Block OFDM Spreading Matrices for Use on Multipath Fading Channels". *IEEE Trans. Commun.*, vol. 53, no. 4, p. 656–665, Apr. 2005.
- [10] J. X. Yang, et al., "A Low-Complexity 2D Signal Space Diversity Solution for Future Broadcasting Systems". *IEEE Intern. Conf. Commun. (ICC)*, London, UK, Jun. 2015.
- [11] Y. Li, X. Xia, G. Wang, "Simple Iterative Methods to Exploit the Signal-Space Diversity". *IEEE Trans. Commun.*, vol. 53, no. 1, pp. 32–38, Jan. 2005.
- [12] P. Elias, "List decoding for noisy channels". *Res. Lab. Electron., Massachusetts Inst. Technol., Cambridge, MA, USA, Tech. Rep.* pp. 335, 1957.
- [13] J. M. Wozencraft, "List decoding". Ph.D. dissertation, *Dept. Elect. Eng. Comput.*, Massachusetts Inst. Technol., Cambridge, MA, USA, 1958.
- [14] I. Tal and A. Vardy, "List Decoding of Polar Codes". *IEEE Trans. Inf. Theory*, vol. 61, no. 5, pp. 2213–2226, May 2015.