

4

Estimação de Canal e Detecção em OFDM com Enlace Fechado

Neste capítulo analisamos que opções temos com o uso de um enlace fechado, isto é, na situação onde existe um canal reverso entre os usuários forte e fraco. Pode-se aproveitar da melhor posição logística do usuário forte para otimizar a direção fraco-forte, onde a comunicação é mais delicada. Consideramos por simplicidade que a comunicação reversa é feita de forma imediata e sem erros. O tópico de como melhor utilizar o enlace fechado em transmissões OFDM não foi tão explorado na literatura. Veremos três opções para o seu uso, que estão ilustradas na Fig. 4.1. Na primeira, estudada em (Wan00) e descrita na Seção 4.1, a comunicação reversa consiste na transmissão do canal estimado, $\hat{\mathbf{q}}$. Dessa forma, o transmissor fraco pode *pré-equalizar* o próximo bloco a ser transmitido. A seguir, analisamos a opção de *alocação dinâmica de potência* (Gol02) no usuário fraco. Nesse cenário, o usuário forte calcula qual a melhor opção possível do uso da potência nos sub-canais de dados e transmite esses valores em um vetor \mathbf{e} . O usuário fraco utiliza esse perfil de potência em sua próxima transmissão.

Por fim, na Seção 4.3, voltamos a atenção para uma opção bem mais interessante, a de *alocação dinâmica de pilotos*. Foi mostrado em artigo recente (Pan08) que, das três alternativas, é essa que traz os ganhos mais importantes. Nessa situação, o usuário forte calcula qual a melhor disposição de pilotos, ou seja, acha um vetor \mathbf{p} , e o envia para o usuário fraco. Veremos que, ao contrário do sistema de enlace aberto descrito no Capítulo 3, a melhor opção possível para \mathbf{p} é na imensa maioria das vezes não-uniforme. Nesse ponto, analisamos na Seção 4.3.1 a solução anteriormente proposta na literatura e na seção seguinte propomos uma melhor. Sugerimos também na Seção 4.3.3 um algoritmo sub-ótimo mas de baixa complexidade que nos permitirá diminuir bastante o custo computacional da solução proposta sem grandes perdas em desempenho. São essas as principais contribuições deste trabalho.

Fechando esse capítulo, a Seção 4.4 mostra diversos experimentos que exemplificam o potencial de cada solução para melhorar a probabilidade de erro de bit das transmissões OFDM. Nessa parte, demonstramos com clareza

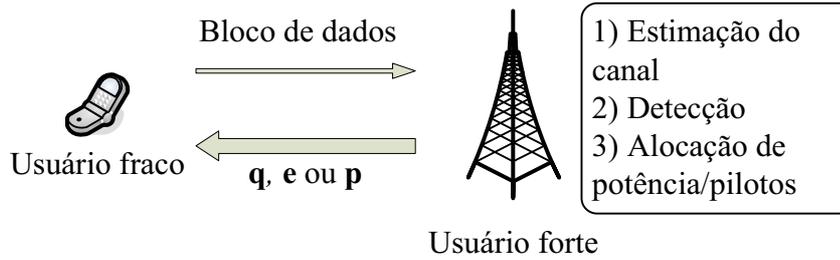


Figura 4.1: Enlace fechado.

primeiro a superioridade dos esquemas de alocações de piloto em relação aos outros dois, e em seguida a melhora conseguida com a solução proposta em relação à solução anterior.

4.1 Pré-Equalização

A pré-equalização, mostrada em diagrama de blocos na Fig. 4.2, funciona da seguinte forma: o usuário forte estima o canal mas não o utiliza para a detecção do mesmo bloco. O vetor $\hat{\mathbf{q}}_{n-1}$ é enviado ao usuário fraco com o uso da comunicação reversa, para a seguir ser usado para *pré-distorcer* o bloco de símbolos a ser enviado no instante n . Caso o canal na frequência seja correlacionado em instantes consecutivos, o que é de se esperar que aconteça, não há necessidade de equalização no recebimento dos símbolos. O bloco transmitido é

$$\mathbf{X}_{\text{pre},n} = G \text{diag} \{ \mathbf{x}_n \} (\text{diag} \{ \hat{\mathbf{q}}_{n-1} \})^{-1}, \quad (4-1)$$

e o bloco recebido é

$$\begin{aligned} \mathbf{r}_{\text{pre},n} &= G^{-1} G \mathbf{X}_{\text{pre},n} \mathbf{q}_n + \mathbf{z}_n \\ &= \text{diag} \{ \mathbf{x}_n \} (\text{diag} \{ \hat{\mathbf{q}}_{n-1} \})^{-1} \mathbf{q}_n + \mathbf{z}_n. \end{aligned} \quad (4-2)$$

Em (4-1), o ganho G é utilizado para tornar justa a comparação entre os sistemas de pré- e pós-equalização. O sistema de pós-equalização, visto no Capítulo 3, dispunha de uma constelação com $E_s = 1$ para cada subcanal, somando K unidades de energia para a transmissão de um bloco. A multiplicação por $\text{diag} \{ \hat{\mathbf{q}}_{n-1} \}^{-1}$ no sistema de pré-equalização faz com que as potências de cada tom sejam diferentes, e usamos G para o controle de potência. Esse ganho deve satisfazer

$$\mathcal{E} \left[\left\| G \text{diag} \{ \hat{\mathbf{q}}_{n-1} \}^{-1} \mathbf{x}_n \right\|^2 \right] = \mathcal{E} \left[\left\| \mathbf{x}_n \right\|^2 \right], \quad (4-3)$$

o que resulta em

$$G = \frac{\sqrt{K}}{\left\| \text{diag} \{ \hat{\mathbf{q}}_{n-1} \}^{-1} \right\|_{\mathcal{F}}}. \quad (4-4)$$

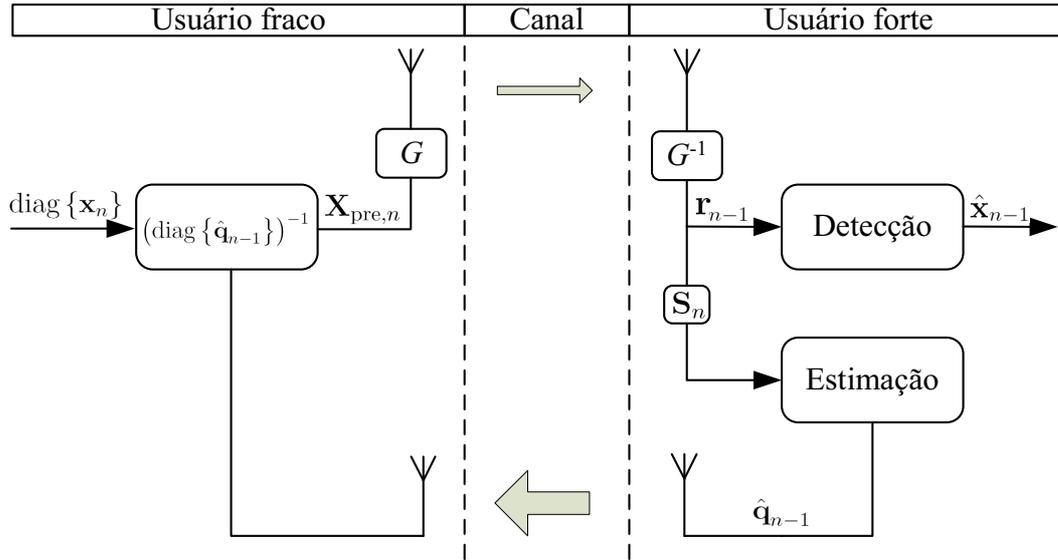


Figura 4.2: Diagrama de bloco para a pré-equalização.

O processo de estimação do canal consiste em alguma das opções vistas no Capítulo 3. Usaremos pilotos sempre uniformemente espaçados para a análise. Em (4-2), idealmente teríamos $(\text{diag} \{\hat{\mathbf{q}}_{n-1}\})^{-1} \mathbf{q}_n = \mathbf{1}_{K \times 1}$, ou seja, a estimação do canal no instante $n-1$ é perfeita e não há mudança no canal entre instantes consecutivos – $\mathbf{1}_{K \times J}$ indica um vetor de 1's de tamanho indicado pelo subscrito. O processo de detecção por mínima distância é

$$\hat{x}_n^k = \arg \min_{\tilde{x} \in \mathcal{X}} |r_n^k - \tilde{x}|^2. \quad (4-5)$$

Não nos deteremos tanto nessa seção a não ser para alguns comentários. O efeito da pré-equalização é o de tornar o sistema mais igualitário em relação aos sub-canais. Existirá o erro de estimação, obviamente, ainda mais acentuado pelo fato de que o canal não é constante de um instante a outro, mas o fato de o receptor não necessitar equalizar o sinal recebido em (4-5) faz com que o ruído total na detecção de todos os sub-canais seja mais equilibrado. Em sistemas com pós-equalização, como aquele visto no Capítulo 3, os diferentes ganhos nos sub-canais fazem com que o ruído de detecção seja aumentado ou diminuído. Para os de pré-equalização, é possível mostrar que

$$\text{SNR}_n^k = \frac{E_b |q_n^k|^2}{\sigma |\hat{q}_{n-1}^k|^2 + \sigma_{\text{dist},n}^k + \sigma_{\text{inter},n}^k}. \quad (4-6)$$

Cabe ressaltar que o RD é um pouco diferente para o caso de pré-equalização. Isso é consequência do fato de que $\hat{\mathbf{q}}$ é calculado no instante anterior ao \mathbf{q} real. O RD é dado por

$$\sigma_{\text{dist},n}^k = (\boldsymbol{\lambda}')^T \mathcal{R}_{q,q,t} \boldsymbol{\lambda}', \quad (4-7)$$

onde nesse caso o vetor $\boldsymbol{\lambda}' = [-1 \quad \boldsymbol{\lambda}^T]^T$ tem tamanho $M+1$. Obviamente

a matriz de correlação em (4-7) tem o mesmo tamanho, ou seja, possui os valores da função $\mathcal{R}_{q,t}[n]$ do instante 0 ao instante M organizados em matriz do mesmo jeito que (2-30). É interessante notar que existe RD mesmo quando não há filtragem temporal, ou seja, mesmo se utilizarmos o ML simples para a estimação o termo do RD é diferente de zero. O RI é o mesmo do caso da pós-equalização.

Considere em (4-6) que utiliza-se mais pilotos do que o necessário e que a ordem do filtro de recepção é grande (M é grande). Esses dois fatos fazem com que ambas as parcelas do RD e do RI sejam pequenas. Podemos então aproximar a SNR por

$$\text{SNR}_n^k \approx \frac{E_b |q_n^k|^2}{\sigma |q_{n-1}^k|^2} \approx \frac{E_b}{\sigma}, \quad (4-8)$$

onde consideramos que a variação temporal do canal é pequena e portanto $|q_n^k| \approx |q_{n-1}^k|$. Em (4-8), percebe-se que a SNR não é seletiva em frequência. É esse o efeito da pré-equalização: não há uma diminuição da BER total do sistema, mas ela é apenas distribuída de forma mais igualitária para todos os blocos entre todos os sub-canais.

4.2

Alocação de Potência

Proposto por Goldfeld *et al.* (Gol02), a alocação dinâmica de potência visa distribuir de forma mais inteligente as $(K - K_p)$ unidades de potência disponíveis para os sub-canais de dado do usuário fraco. Manteremos os pilotos com $E_s = 1$, e portanto há um total de K_p unidades de potência para a sua transmissão.

Dado um canal estimado, $\hat{\mathbf{q}}_{n-1}$, o usuário forte calcula qual a melhor utilização da potência em um vetor \mathbf{e}_n e o envia para o usuário fraco para utilização no próximo bloco a ser transmitido. O problema sugerido por Goldfeld é

$$\mathbf{e}_n = \arg \min_{\mathbf{e}} \left[1 - \prod_{k \in \{d\}}^{K-1} \left(1 - P_{n-1}^k(\epsilon | \hat{q}_{n-1}^k) \right) \right], \quad (4-9)$$

ou seja, a minimização é a da probabilidade de erro de *vetor*. Define-se erro de vetor quando um ou mais símbolos transmitidos em um bloco são decodificados de forma incorreta. Os termos do produto são as probabilidades de bits decodificados corretamente para cada sub-canal de dado. Lembrar que o vetor de potência entra no cálculo de cada termo, isto é, a SNR a ser utilizada em (4-9) é dada por $\text{SNR}_{n-1}^k = |e^k \hat{q}_{n-1}^k|^2 / \sigma_{\text{tot},n-1}^k$. A probabilidade de erro de vetor é conseguida subtraindo o produto de 1. É argumentado em (Lov07) que (4-9) equivale achar um vetor \mathbf{e}_n que maximiza a SNR mínima entre os

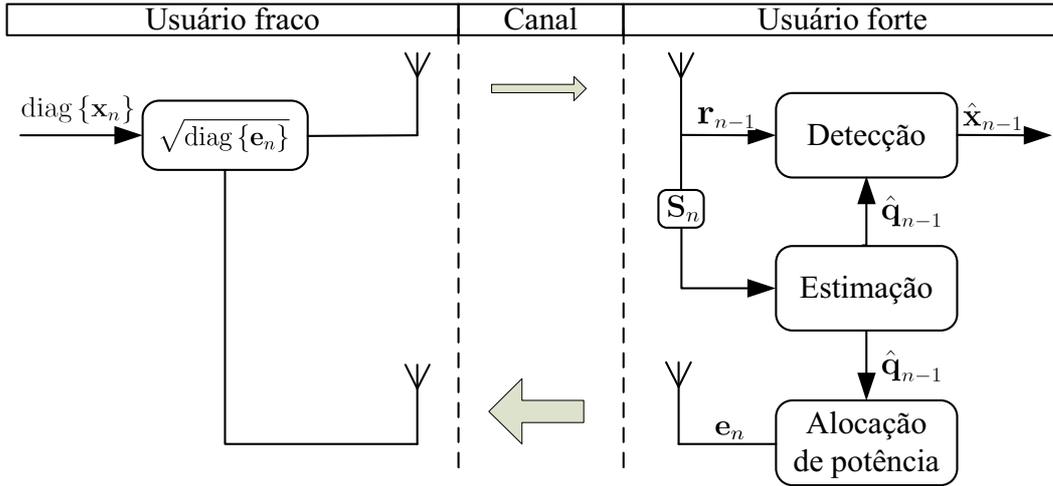


Figura 4.3: Diagrama de blocos para a alocação dinâmica de potência.

sub-canais de dados.

Resolver (4-9) é complicadíssimo. Segundo (Gol02), é necessário resolver $(K - K_p)$ equações transcendentais para a solução ótima. Por esse motivo, é sugerido uma solução sub-ótima mas de fácil computação, dada por

$$e_n^k = (K - K_p) \frac{1}{(\hat{q}_{n-1}^k)^2} \left[\sum_{k \in \{d\}} (\hat{q}_{n-1}^k)^{-2} \right]^{-1}. \quad (4-10)$$

É essa proposta que levaremos em conta no restante desse texto.

O bloco transmitido para a alocação de potência seria

$$\mathbf{X}_{ap,n} = \sqrt{\text{diag}\{e_n\}} \text{diag}\{\mathbf{x}_n\}, \quad (4-11)$$

e o bloco recebido é

$$\begin{aligned} \mathbf{r}_{ap,n} &= \mathbf{X}_{ap,n} \mathbf{q}_n + \mathbf{z}_n; \\ &= \sqrt{\text{diag}\{e_n\}} \text{diag}\{\mathbf{x}_n\} \mathbf{q}_n + \mathbf{z}_n. \end{aligned} \quad (4-12)$$

O diagrama de blocos para a transmissão é mostrado na Fig. 4.3.

É argumentado em (Pan08) que a estratégia de alocação de potência não traz ganhos significativos para o sistema. Confirmaremos esse fato na Seção 4.4.

4.3

Alocação de pilotos

Vimos no Capítulo 3 que o perfil de pilotos que melhor estima o canal é aquele uniformemente espaçado. Conseguimos melhores resultados ainda caso utilizemos estimadores mais sofisticados, como o MMSE e o filtro de Wiener.

Para o caso do loop aberto é essa a melhor opção possível. Porém a qualidade da estimação do canal é apenas parte do processo de detecção, e com o enlace fechado temos uma opção mais interessante.

4.3.1

Solução de Panah et al.

Panah *et al.* propuseram em (Pan08) um esquema de alocação de pilotos que leva em conta não apenas o MSE de estimação resultante, mas também a qualidade dos sub-canais que carregam pilotos. Ao mesmo tempo que é desejável manter os pilotos uniformemente espaçados para o menor MSE possível, é também interessante que os pilotos sejam transmitidos nos piores canais. Afinal, os sub-canais piloto não carregam dados, e portanto não contribuem para a probabilidade de erro final do sistema.

Os dois interesses são, na imensa maioria dos casos, conflitantes, e um compromisso deve ser conseguido entre eles. A referência (Pan08) propõe selecionar a localização dos pilotos de acordo com a SNR média do sistema. Define-se

$$\overline{\text{SNR}}_{n-1}(\mathbf{p}) = \frac{\hat{\mathbf{q}}_{n-1}^{\mathcal{H}} (\mathbf{I}_K - \mathbf{D}) \hat{\mathbf{q}}_{n-1}}{\sigma(K - K_p) + \sigma_{\text{inter},n-1} + \sigma_{\text{dist},n-1}} \quad (4-13)$$

como o SNR média (média das SNR's dos tons que carregam dado) conseguida com a alocação \mathbf{p} . A matriz $(\mathbf{I}_K - \mathbf{D})$ seleciona os canais de dados. O estimador estudado no artigo é o ML simples, e portanto

$$\sigma_{\text{inter},n} = \sigma \left(\text{tr} \{ \mathbf{F}_L \mathbf{U}_n^{-1} \mathbf{F}_L^{\mathcal{H}} \} - L \right) \quad (4-14)$$

e $\sigma_{\text{dist},n} = 0$. Iremos também experimentar essa solução com estimadores mais confiáveis na seção de experimentos.

O esquema proposto funcionaria da seguinte forma: na primeira transmissão não se tem nenhuma informação sobre o canal, e portanto a melhor coisa a se fazer é estimá-lo da melhor forma possível. Isso é conseguido com uma distribuição uniforme dos pilotos, como visto anteriormente. A partir do momento em que dispomos de um $\hat{\mathbf{q}}$, o problema a ser resolvido então é simplesmente achar um \mathbf{p}_n que maximiza a expressão em (4-13), ou seja,

$$\begin{aligned} \mathbf{p}_n &= \arg \max_{\mathbf{p} \in \mathcal{P}} \overline{\text{SNR}}_{n-1}(\mathbf{p}) \\ \text{tal que } & \sum_{k=0}^{K-1} p^k = K_p. \end{aligned} \quad (4-15)$$

O vetor \mathbf{p}_n é então usado para a transmissão do bloco seguinte e processo se repete. A Fig. 4.4 mostra o processo graficamente. Ganhos consideráveis são registrados em (Pan08).

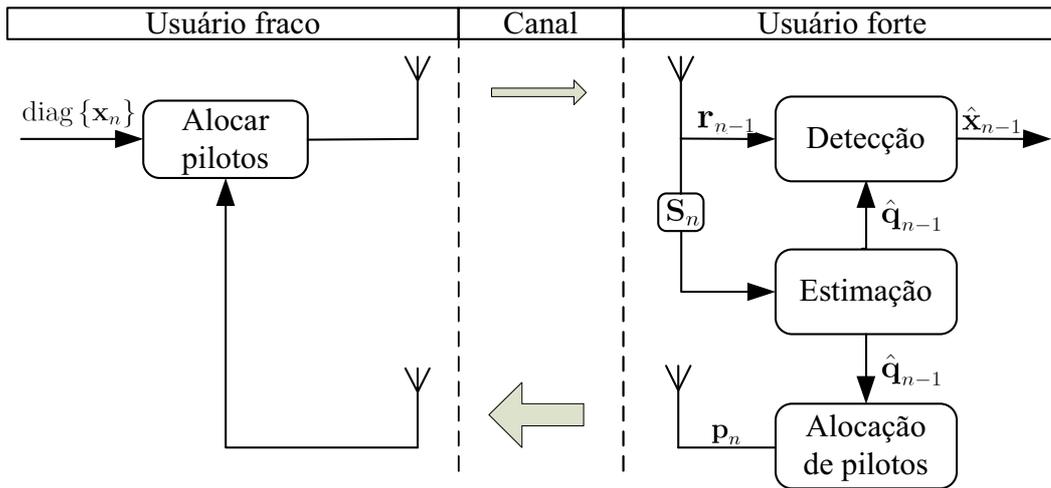


Figura 4.4: Diagrama de blocos para a alocação dinâmica de pilotos.

Observe como a função-objetivo em (4-15) leva em conta ambos os interesses conflitantes: de um lado, não se quer “desperdiçar” bons canais com a transmissão de pilotos, o que se reflete na maximização do numerador de (4-13); e, do outro, se deseja que os pilotos estejam dispostos de forma suficientemente ordenada para uma estimação satisfatória do canal. O vetor \mathbf{p}_n resultante muito provavelmente não será uniforme, pois a estimação do canal não é um fim em si. Caso a função última do sistema fosse a estimação do canal, então com certeza a melhor escolha para os pilotos seria aquela uniformemente espaçada. Almeja-se, porém, uma estimação apenas *boa o suficiente* para uma detecção eficiente do dados, esse sim o objetivo da transmissão. O vetor \mathbf{p}_n ótimo tampouco terá os pilotos alocados somente nos piores canais, pois isso muito provavelmente produziria uma estimação do canal muito ruim ($\text{tr}\{\mathbf{F}_L \mathbf{U}_n^{-1} \mathbf{F}_L^H\}$ muito grande), o que por sua vez levaria a uma taxa de erro elevada. A solução da Eq. (4-15) leva em conta ambos os lados, e é nisso que consiste a contribuição de (Pan08). Além disso, deve-se notar que a transmissão reversa para essa proposta é bem mais simples e confiável do que para as duas anteriores: o vetor transmitido do usuário forte para o fraco é binário (ver Fig. 4.4), enquanto que para a pré-equalização e alocação de potência é necessário o envio de vetor de números reais. Isso incorre em erros inevitáveis de quantização e em um processo de comunicação bem mais complexo.

A referência (Pan08) propõe um processo de busca exaustiva para achar o melhor \mathbf{p} em (4-15). Isso quer dizer que calcula-se (4-13) para *todas* as opções possíveis. São $\binom{K}{K_p}$ possibilidades, o que implica que a busca exaustiva só poderá ser feita para sistemas pequenos. Se, como no experimento da Seção 3.4, os parâmetros do sistema forem aqueles mostrados na Tabela 3.2, o número de possibilidades é $\binom{64}{16}$, que é da ordem de 10^{14} .

A solução de Panah *et al.* pode melhorar em alguns pontos, e é isso o assunto da próxima seção. Eles não levam em conta que maximização da SNR *média* do canal não é a melhor métrica para a seleção dos pilotos. O artigo também peca por não explorar estimadores de canal mais sofisticados – o estimador usado é o ML apresentado na Seção 3.1.1 – e em não propor algoritmos de menor complexidade para a solução de (4-15).

4.3.2 Solução proposta

São duas as principais mudanças da solução proposta. Em primeiro lugar, usamos estimadores mais sofisticados, como o ML seguido de filtro de Wiener. Esse conjunto explora de forma eficiente tanto a correlação em frequência quanto a temporal do canal, e isso levará a uma estimação muito melhor do que a do ML simples. Para o cálculo do filtro de Wiener, consideramos sempre pilotos uniformemente espaçados – mais sobre esse ponto a seguir. Resultados potencialmente melhores seriam conseguidos com o estimador MMSE. Porém, foi visto na Seção 3.4 que o MMSE não traz ganhos tão grandes para o MSE e principalmente para a probabilidade de erro de bit quando comparado com o ML, e dessa forma usaremos apenas o ML. A segunda mudança é que levamos em conta a probabilidade de erro precisa do sistema para a escolha dos pilotos. A seguir descrevemos o método proposto assim como uma análise detalhada dos potenciais ganhos.

Definimos a SNR do sub-canal k no instante n como

$$\text{SNR}_{n-1}^k \approx \frac{|\hat{q}_{n-1}^k|^2}{\sigma_{\text{tot},n-1}^k}, \quad (4-16)$$

em que $\sigma_{\text{tot},n-1}^k = \sigma + \sigma_{\text{inter},n-1}^k + \sigma_{\text{dist},n-1}^k$. Se usarmos o estimador ML seguido de um filtro de Wiener, as variáveis $\sigma_{\text{inter},n-1}^k$ e $\sigma_{\text{dist},n-1}^k$ são calculadas por (3-68) e (3-62), respectivamente (ver a terceira linha da Tabela 3.1). Idealmente disporíamos do valor real do canal para o cálculo da SNR em (4-16), mas o receptor não tem acesso a essa informação. Usamos a estimativa do canal, \hat{q}_{n-1}^k , como uma aproximação. O esquema funcionaria da mesma forma que a solução de Panah: na primeira transmissão não há conhecimento algum do canal, e portanto pilotos uniformemente espaçados fornecem a melhor estimativa possível. A partir do ponto em que dispomos de $\hat{\mathbf{q}}_{n-1}$, temos condições de alocar pilotos para a próxima transmissão a partir do cálculo da função-objetivo proposta.

A função-objetivo levará em conta a probabilidade de erro precisa do sistema. Seja a probabilidade de erro de bit para o bloco n e sub-canal k dado q_n^k , já mencionada em (3-72) e repetida aqui por conveniência,

$$P_{n-1}^k(\epsilon | \kappa) = \mathcal{Q}\left(\sqrt{\text{SNR}_{n-1}^k}\right).$$

Lembramos que $\kappa = |q_n^k|$. A função-objetivo proposta é a média das probabilidades de erro dos canais que carregam dados, ou seja,

$$\begin{aligned} \hat{\mathbf{p}}_n = \arg \min_{\mathbf{p} \in \mathcal{P}} \left\{ \frac{1}{K - K_p} \sum_{k \in \{d\}} P_{n-1}^k(\epsilon | \kappa) \right\} \\ \text{tal que } \sum_{k=0}^{K-1} p^k = K_p. \end{aligned} \quad (4-17)$$

Assim, é garantido que a seleção das portadoras que carregam pilotos será aquela que minimiza a probabilidade de erro média de bit do sistema. Em (4-17), o somatório é apenas nos sub-canais que contém dado – isso é expresso por $k \in \{d\}$, ou, de forma equivalente, $p^k = 0$.

O problema a ser resolvido leva em conta ambos os fatores da solução de Panah, porém de forma bem mais precisa. Uma determinada alocação \mathbf{p} será boa se bons canais não forem desperdiçados com pilotos – os pilotos tendem a ocupar os piores canais; e se o pilotos estiverem dispostos de forma suficientemente ordenada para a estimação. A palavra *suficientemente* é a chave aqui. Por exemplo, caso o ruído seja muito pequeno, ou seja, $E_b/\sigma \rightarrow \infty$, *qualquer* \mathbf{p} estimará o canal de forma razoável, mesmo aquele em que os pilotos estão em sub-canais adjacentes. Pode-se nesse caso alocar os pilotos nos K_p piores canais e ainda assim ter uma boa estimação.

Um cuidado deve ser tomado no cálculo da probabilidade de erro em (3-72). Seja a recepção por mínima distância expressa na Eq. (3-60), que repetimos aqui por conveniência,

$$\Omega = q_n^k (x_n^k - \tilde{x}) + z_n^k - \tilde{x} \sum_{i=0}^{M-1} \lambda'[i] q_{n-i}^k - \tilde{x} \sum_{i=0}^{M-1} \lambda[i] \mathbf{f}^k(\mathbf{T}\mathbf{w}_p)_{n-i}.$$

Mencionamos no Capítulo 3 que, para que o cálculo preciso da probabilidade de erro seja dado por (3-72), é necessário que duas condições sejam cumpridas. Em primeiro lugar, o ruído total deve ser gaussiano de média zero, o que, como já citado, é exatamente o caso. Segundo, o ruído deve ser estatisticamente independente das flutuações da energia do sinal de interesse. Isso não é respeitado, pois a componente do sinal e o ruído de distorção (RD) têm correlação. Como estamos interessados nesse ponto em algo mais preciso do que a noção intuitiva de probabilidade de erro dada no Capítulo 3, esse fato deve ser contornado. Isso pode ser conseguido caso calculemos apropriadamente os coeficiente do filtro de Wiener. Pois uma vez que, como também já mencionado, q_n^k e o RD são VAs conjuntamente gaussianas, decorrelação entre elas garante

a independência estatística. Assim, a condição necessária e suficiente para a independência estatística entre a componente do sinal e o RD é

$$\mathcal{E} \left[(q_n^k)^* \sum_{i=0}^{M-1} \lambda' [i] q_{n-i}^k \right] = 0, \quad (4-18)$$

onde λ' é dado por (3-46). A condição (4-18) resulta em

$$(\lambda')^T [\mathcal{R}_{qq,t}]_{\text{col } 1} = 0, \quad (4-19)$$

onde $\mathcal{R}_{qq,t}$ é dado por (3-65). De forma equivalente, obtemos

$$\lambda^T [\mathcal{R}_{qq,t}]_{\text{col } 1} - 1 = 0. \quad (4-20)$$

Podemos então incorporar essa restrição à solução de Wiener da seguinte forma: re-escrevemos (3-37) como

$$\hat{\lambda}_n = \arg \min_{\lambda} \mathcal{E} \left[\|\mathbf{h}_n - \mathbb{Y}\lambda\|^2 \right], \quad (4-21)$$

tal que $\lambda^T [\mathcal{R}_{hh,t}]_{\text{col } 1} - 1 = 0.$

Re-escrevendo (4-21) com a introdução do multiplicador de Lagrange β , obtemos

$$\hat{\lambda}_n = \arg \min_{\lambda} \left\{ \mathcal{E} \left[\|\mathbf{h}_n - \mathbb{Y}\lambda\|^2 \right] + \beta \left(\lambda^T [\mathcal{R}_{hh,t}]_{\text{col } 1} - 1 \right) \right\}. \quad (4-22)$$

Calculado o gradiente $\nabla \lambda$, igualamos a zero, resolvemos a equação resultante e conseguimos

$$\lambda = \frac{\mathcal{R}_{\mathbb{Y}\mathbb{Y}}^{-1} [\mathcal{R}_{hh,t}]_{\text{col } 1}}{[\mathcal{R}_{hh,t}]_{\text{col } 1}^H \mathcal{R}_{\mathbb{Y}\mathbb{Y}}^{-1} [\mathcal{R}_{hh,t}]_{\text{col } 1}}, \quad (4-23)$$

onde $\mathcal{R}_{\mathbb{Y}\mathbb{Y}}$ é dado por (3-43) para o ML com a alocação de pilotos invariante no tempo. Portanto, o cálculo do filtro de Wiener original deve conter uma normalização pela constante do denominador de (4-23). Com essa condição satisfeita, todas as VAs da recepção são conjuntamente gaussianas, de média zero e descorrelacionadas entre si. Aí sim podemos calcular precisamente a probabilidade de erro para cada sub-canal e a probabilidade de erro média do sistema *dado* \mathbf{q} em (4-17). É de se notar que a constante que normaliza (4-23) é geralmente próxima a 1, e portanto o filtro resultante não é tão diferente assim do filtro original sem restrições.

O cálculo do filtro de Wiener leva em conta em $\mathcal{R}_{\mathbb{Y}\mathbb{Y}}$ uma disposição uniforme para os pilotos. Obviamente essa não é a melhor filtragem possível, pois, como já falado, o vetor \mathbf{p}_n resultante de (4-17) será na imensa maioria dos casos não-uniforme. Assumimos pilotos uniformes para os cálculos dos coeficientes de Wiener por simplicidade. Caso levássemos em conta a variação temporal de \mathbf{p}_n , o filtro resultante seria também variante no tempo, o que

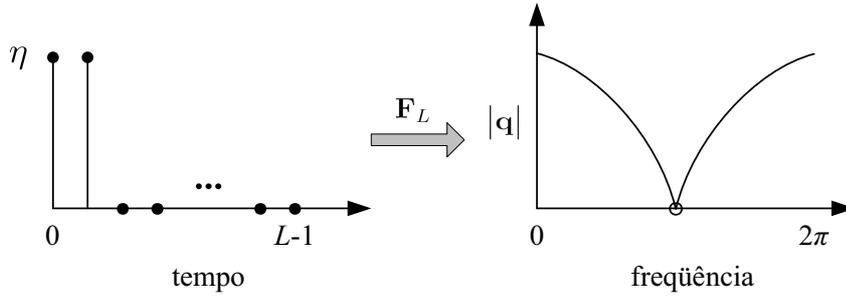


Figura 4.5: Degrau no tempo com dois instantes tem apenas um nulo espectral.

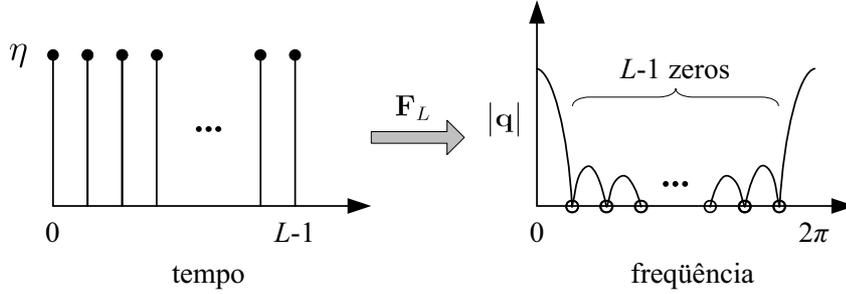


Figura 4.6: Degrau no tempo com L instantes tem $L - 1$ nulos espectrais.

aumentaria bastante a complexidade computacional do receptor – a cada novo bloco recebido um novo filtro teria que ser calculado.

Corolário 1 *Evitando canais nulos:* *Se o algoritmo de alocação de pilotos proposto for utilizado, sempre que ocorrer um sub-canal nulo no módulo na resposta de frequência do canal, esse sub-canal será ocupado por um piloto.*

Prova. Para a análise seguinte, consideramos que L e K são sempre pares. Uma condição necessária, obviamente, é que o canal exista, ou seja, $\mathbf{h} \neq \mathbf{0}_{L \times 1}$. Nulos no módulo da resposta de frequência de canal, que chamaremos de *nulos espectrais*, ocorrem quando há elementos de \mathbf{h} iguais uns aos outros. Considere como exemplo a seguinte resposta ao impulso,

$$\mathbf{h} = [\eta \quad \eta \quad 0 \quad \cdots \quad 0]^T, \quad (4-24)$$

onde η é um complexo qualquer. Lembramos que os taps da resposta ao impulso são decorrelacionados. A resposta em frequência terá o perfil mostrado na Fig. 4.5, ou seja, com um nulo espectral. Caso um canal de dado seja alocado no nulo, a probabilidade de erro de bit da transmissão neste sub-canal seria a maior possível – já que não existe sinal no receptor, este decidiria por sorteio os bits e a probabilidade de erro para cada bit seria $1/2$.

Considere agora a situação com maior número de zeros espectrais, aquela em que $\mathbf{h} = \eta [1 \quad \cdots \quad 1]^T$. A resposta na frequência nesse caso é mostrada na Fig. 4.6. Mesmo nessa situação, há apenas $L - 1$ nulos espectrais em $|\mathbf{q}|$.

Lembramos que devem existir pelo menos $K_p \geq L$ pilotos para a estimação. Caso algum canal de dado seja alocado em qualquer nulo espectral, sua probabilidade de erro será a maior possível. Como queremos minimizar a soma das probabilidades de erro dos canais de dado, nenhum canal de dado pode ter a maior probabilidade de erro possível. Já que temos mais pilotos do que nulos espectrais, $L - 1$ pilotos serão alocados nos nulos e os pilotos restantes serão distribuídos de acordo com as necessidades do sistema. ■

Canais nulos acontecem com probabilidade tendendo a zero. Porém canais *quase-nulos* (chamados também de desvanecimentos profundos) acontecem com frequência considerável e devem ser evitados a todo custo. Se verificamos que, na ocorrência de canais nulos, *sempre* um piloto será alocado no nulo, é de se esperar que pilotos serão alocados em canais quase-nulos *com grande probabilidade*. Quanto mais o ganho de um sub-canal se aproxima de zero, maior a probabilidade de que um piloto seja alocado nele. Em outras palavras: mostramos que canais nulos sempre carregarão pilotos. Argumentamos agora que, quando o módulo de um sub-canal *tende a zero*, a probabilidade que um piloto seja alocado nesse sub-canal *tende a 1*. Isso é nada mais do que consequência da função-objetivo expressa em (4-17). Observar que a função-objetivo de Panah não necessariamente se comporta da mesma forma. A maximização da SNR média dá margem para dados em canais nulos ou quase-nulos desde que haja canais com ganho bom o suficiente para compensar o zero.

Vale nesse ponto comentar sobre as diferenças entre as funções-objetivo expressas em (4-15) e (4-17). Enquanto a estratégia de Panah escolhe o perfil de pilotos que produz a maior média de SNR, a que propomos seleciona a menor probabilidade de erro média entre os tons. As duas abordagens estão ilustradas na Fig. 4.7. A abordagem proposta é superior. Pode-se interpretar geometricamente a diferença entre as duas situações com a ajuda das figuras. A abordagem de Panah não necessariamente seleciona o perfil de pilotos com menor probabilidade de erro total. Um exemplo é mostrado na Fig. 4.7(a). Essa solução quer “empurrar” a média das SNRs dos sub-canais selecionados para carregar dados o mais à direita possível, mas pelo do fato da função $Q(\cdot)$ ser altamente não linear em seu argumento (ver (3-73)), os valores menores de SNR serão dominantes para o cálculo de erro. Selecionar o perfil de pilotos que maximiza a mínima SNR entre os canais seria uma estratégia melhor, apesar de não ótima. O esquema proposto, mostrado na Fig. 4.7(b), não incorre no mesmo erro.

Uma consequência interessante do Corolário 1 é que, já que canais com ganho pequeno são deliberadamente evitados, mexe-se na densidade de probabilidade de $\kappa = |q|$ dos canais de dado. A fdp não mais será dada por

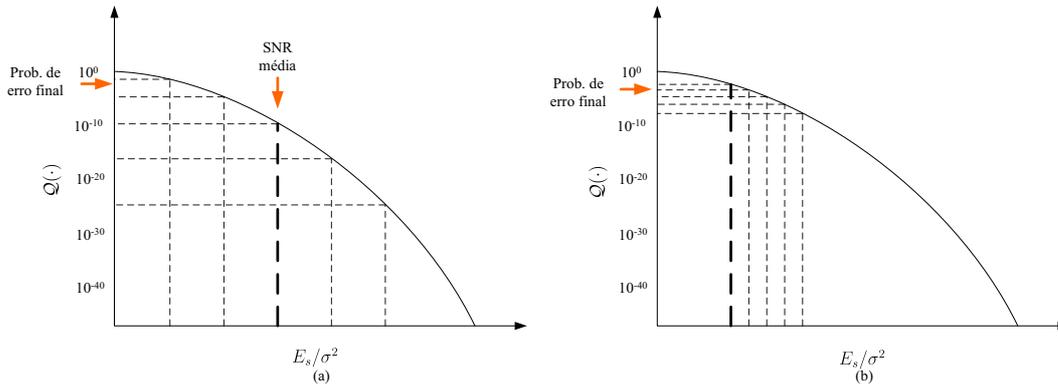


Figura 4.7: SNR vs. probabilidade de erro.

(3-76). Sua nova forma é muito difícil de calcular, já que a escolha dos canais de dado em (4-17) depende de diversos fatores – na Seção 4.4 veremos resultados experimentais que mostram a mudança em $p(\kappa)$. Essa mudança é fundamental para a BER resultante do sistema. Vimos no Capítulo 3 que, em sistemas com enlace aberto, tudo o que podemos fazer para melhorar o desempenho do erro era melhorar a estimação do canal. A melhora da estimativa torna a descida da função $Q(\cdot)$ mais rápida em (3-75), que repetimos aqui para facilitar a leitura:

$$P(\epsilon) = \int_0^{+\infty} P_n^k(\epsilon | \kappa) p(\kappa) d\kappa.$$

O estreitamento de $Q(\cdot)$ resulta em uma menor área a ser integrada e em uma menor probabilidade de erro. Vemos agora que, em sistemas com enlace fechado, temos mais uma opção. A alocação dinâmica de pilotos “empurra” $p(\kappa)$ para à direita, formando assim $p'(\kappa)$ e fazendo com que o produto $Q(\cdot) \times p'(\kappa)$ tenha menor área. A Fig. 4.8 ilustra a situação. Vimos que o esquema proposto seleciona os pilotos levando em conta dois fatores. Primeiro, deseja-se que os pilotos estejam nos piores canais possíveis. Segundo, é preciso que os pilotos estejam dispostos de forma suficientemente ordenada para uma estimação satisfatória do canal. Esses dois fatores são na imensa maioria dos casos conflitantes entre si. Chegamos, pois, a uma interpretação matemática desse compromisso: pilotos nos piores canais significam dados nos melhores canais, que pode ser traduzido como empurrar $p(\kappa)$ o mais à direita possível; e pilotos suficientemente ordenados significam melhor estimação, que pode ser traduzido como o estreitamento de $Q(\cdot)$. O esquema proposto chega à melhor relação possível entre os dois fatores concorrentes.

Pelo fato de que $p'(\kappa)$ ser extremamente difícil de calcular, não podemos partir de (3-75) chegar a uma expressão final para a BER como fizemos no Capítulo 3 (ver (3-77)). Entretanto, veremos adiante através de experimentos que os ganhos conseguidos com o esquema proposto são bem significativos.

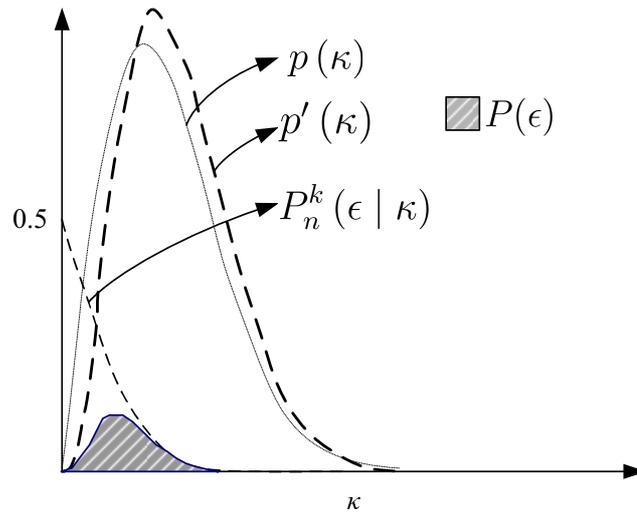


Figura 4.8: Probabilidade de erro para enlace fechado.

A necessidade de tornar $\mathcal{Q}(\cdot)$ mais íngreme pode ser aliviada com o uso de estimadores mais confiáveis para o canal. A estimação recursiva do filtro de Wiener nos dá margem para relaxar o desempenho da estimação no decorrer do tempo. Dado um canal estimado com precisão, de preferência no primeiro símbolo OFDM, seria possível para o usuário forte gerar uma alocação de pilotos no tempo e na frequência (um *lattice*). Para canais com pouca variação temporal haveria nesse caso um transitório: as alocações de piloto nos primeiros instantes seriam mais voltadas para a estimação e, com o passar do tempo, os pilotos migrariam para os piores canais e a alocação estabilizaria. Ou seja, se há boas estimativas em instantes passados ($\text{tr}\{\mathbf{F}_L \mathbf{U}_{n-1}^{-1} \mathbf{F}_L^H\}$, $\text{tr}\{\mathbf{F}_L \mathbf{U}_{n-2}^{-1} \mathbf{F}_L^H\}$, \dots pequenos), pode-se relaxar o desempenho relativo à estimação da alocação atual ($\text{tr}\{\mathbf{F}_L \mathbf{U}_n^{-1} \mathbf{F}_L^H\}$ pode ser grande).

4.3.3 Busca Iterativa

O problema do esquema proposto é a complexidade computacional quando K ou K_p crescem. Aliás, só é possível fazer a busca exaustiva para valores pequenos de ambos. Se, como no sistema simulado na Seção 3.4 do Capítulo 3, $K = 64$ e $K_p = 16$, então $|\mathcal{P}|$ é da ordem de 10^{14} , o que é claramente impraticável tanto para simulações quanto para sistemas reais.

O que propomos nessa seção é um algoritmo de menor complexidade mas que, como veremos, mantém boa parte da qualidade da busca exaustiva. A esse esquema, nos referimos por *busca iterativa* (BITE). Ele consiste basicamente em buscas independentes para cada piloto e está detalhado no Algoritmo 1. A função-objetivo utilizada é a mesma usada em (4-17), mas o método de busca nesse caso é mais simples.

Algoritmo 1: Busca Iterativa

Entrada: $\hat{\mathbf{q}}_{n-1}$, \mathbf{p}_{n-1} , K , K_p
Saída: \mathbf{p}_{otm}

```

1  $\mathbf{p}_{\text{otm}} \leftarrow \mathbf{p}_{n-1}$ ;
2  $\mathcal{G}_{\text{in}} \leftarrow 1$ ;
3 ind  $\leftarrow$  índices  $k$  de  $\mathbf{p}_{n-1}$  que contém pilotos;
4 repita
5      $\mathcal{G}_{\text{out}} \leftarrow \mathcal{G}_{\text{in}}$ ;
6     para  $i \leftarrow 0$  até  $K_p - 1$  faça
7          $\mathbf{p} \leftarrow \mathbf{p}_{\text{otm}}$ ;
8          $\mathbf{p}[\text{ind}[i]] \leftarrow 0$ ;
9          $\mathcal{G}_{\text{in}} \leftarrow 1$ ;
10        para  $j \leftarrow 0$  até  $K - 1$  faça
11            se  $\mathbf{p}[j] = 1$  então Voltar para linha 10;
12            senão  $\mathbf{p}[j] \leftarrow 1$ ;
13             $\mathcal{G}(\mathbf{p}) \leftarrow (K - K_p)^{-1} \sum_{k \in \{d\}} P(\epsilon | \hat{q}_{n-1}^k)$ ; (ver (4-17))
14            se  $\mathcal{G}(\mathbf{p}) < \mathcal{G}_{\text{in}}$  então
15                 $\mathcal{G}_{\text{in}} \leftarrow \mathcal{G}(\mathbf{p})$ ;
16                 $\mathbf{p}_{\text{otm}} \leftarrow \mathbf{p}$ ;
17                ind $[i] \leftarrow j$ ;
18 até  $\mathcal{G}_{\text{in}} < \mathcal{G}_{\text{out}}$ 
    
```

O que se propõe é realizar a busca dos pilotos uma por vez. Um piloto será deslocado, enquanto que os outros ficarão estáticos. Considere o sistema já citado no Capítulo 3, com $K = 16$ e $K_p = 4$ e com pilotos uniformemente espaçados. Os pilotos estão nas sub-portadoras 2, 6, 10 e 14, e o vetor **ind** na linha 3 do algoritmo guardará esses índices. O processo começaria da seguinte forma: retira-se o piloto do tom 2 (linha 8) para a seguir deslocá-lo para todos os outros sub-canais possíveis no laço que compreende as linhas 10-17. Nomearemos o piloto dessa primeira rodada de a . Obviamente dois pilotos não podem ocupar o mesmo sub-canal, e a linha 11 do Algoritmo evita isso. Para cada uma das possíveis posições forma-se um vetor \mathbf{p} . São $K - K_p + 1$ posições possíveis para o piloto a e todas elas serão experimentadas para o cálculo da função-objetivo, denotada no Algoritmo por $\text{OBJ}(\mathbf{p})$. Ao fim da primeira rodada, o algoritmo guarda o melhor \mathbf{p} no vetor \mathbf{p}^{otm} e começa a segunda rodada. Dessa vez mexe-se na posição do piloto que começou no sub-canal 6, que chamaremos de b . O piloto b também deve ser experimentado para todas as posições possíveis assim como o foi a , só que dessa vez b deve levar em conta a primeira rodada do algoritmo e é bastante provável que a não esteja mais no sub-canal 2. O processo segue daí em diante para os pilotos restantes.

Deve-se repetir o processo descrito no parágrafo acima até que não haja

melhora na função-objetivo desejada. Se dois laços consecutivos retornarem o mesmo \mathbf{p} , então não há mais melhoras possíveis a serem feitas e o BITE pode terminar.

Ilustramos agora o funcionamento do BITE com um exemplo prático. Considere um sistema OFDM de $K = 20$ sub-canais e $K_p = L = 2$. O canal \mathbf{h} foi gerado com o perfil de potência de (2-26). Para esse sistema é possível plotar uma superfície correspondente à função-objetivo em (4-17) se ordenarmos as coordenadas x e y de acordo com os índices da localização dos pilotos. A Fig. 4.9 mostra um grid referente à posição dos pilotos e um código de cor para demonstrar em dB o valor da função-objetivo. Por exemplo, uma possível alocação uniforme para esse sistema é aquela onde os pilotos estão localizados no 5º e 15º sub-canais. Essa alocação tem um número associado a ela correspondente ao cálculo de (4-17). Obviamente que as alocações (5, 15) e (15, 5) são equivalentes, e por isso apenas o triângulo inferior direito do grid da Fig. 4.9 é de interesse.

Considera-se que o BITE parte da alocação uniforme (15, 5). Isso corresponde ao ponto preto mais abaixo na Fig. 4.9. O BITE começa as suas tarefas mantendo o piloto da posição 15 fixo e experimentando o outro piloto em todas as opções possíveis (lembramos que dois pilotos não podem ocupar o mesmo sub-canal). Essa busca resulta nos pilotos localizados em (15, 10), como apontado pela seta número 1. A seguir, mantém-se o piloto do tom 10 parado e mexe-se no piloto do tom 15. A seta número 2 aponta o deslocamento resultante. O ponto final encontrado corresponde ao mínimo global da função-objetivo para esse caso. Há um mínimo local, que se encontra na posição (11, 5). Obviamente, não é garantido que o BITE alcance sempre o mínimo global. Para o caso do exemplo acima, se tivéssemos movido primeiramente o piloto do sub-canal 15, chegaríamos ao mínimo local.

Apesar de o BITE ser um algoritmo de busca local, não é garantido que o vetor \mathbf{p} resultante caia em um mínimo local da função-objetivo. Isso pode ser facilmente entendido por que o BITE não “enxerga” na diagonal.

Veremos na seção de experimentos desse capítulo que o algoritmo de busca iterativo tem desempenho muito bom, mesmo para casos mais complicados do que o do exemplo citado. O interessante é saber *por que* isso acontece. Não se tem uma resposta definitiva para essa pergunta, mas especulamos que isso ocorre por que a função-objetivo em (4-17) é bem comportada. Esse fato, por sua vez, pode ser explicado pelo fato de que canais adjacentes são correlacionados. Não comentaremos mais sobre o assunto, que merece estudo detalhado.

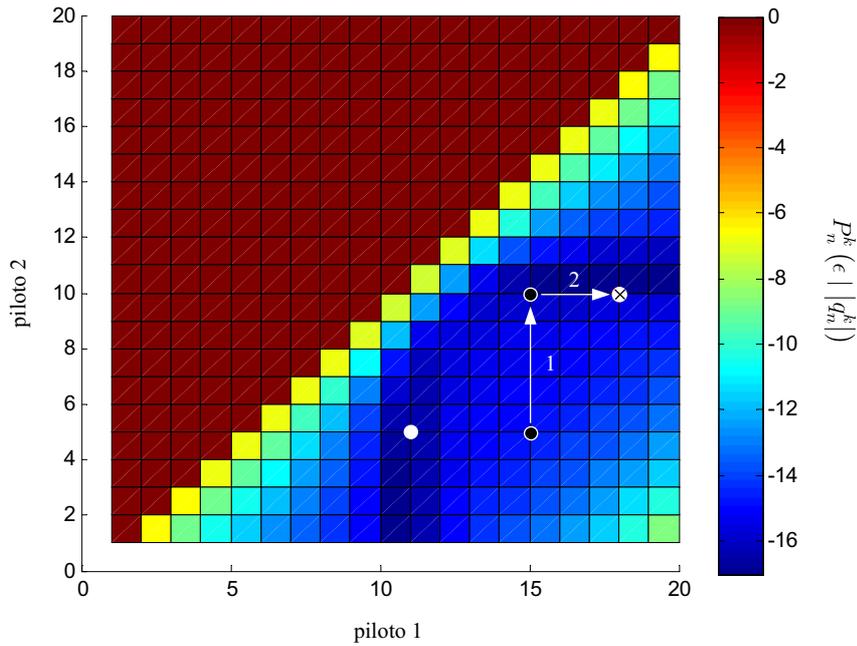


Figura 4.9: Exemplo do funcionamento do BITE.

Tabela 4.1: Parâmetros para a simulação: comparação entre os esquemas de uso do enlace reverso

Comprimento do canal no tempo	L	4
Número de sub-canais	K	16
Número de pilotos	K_p	4
Espalhamento Doppler	f_m	5×10^{-3}
Blocos transmitidos	N	40.000

4.4 Experimentos

Experimento 1

O primeiro experimento compara o desempenho das quatro opções apresentadas para o uso do enlace reverso. São elas: pré-equalização, alocação de potência, alocação de pilotos sugerida por Panah e a alocação de pilotos proposta. O sistema OFDM simulado tem $K = 16$, $K_p = 4$ e $L = 4$. Usamos a busca exaustiva para ambas propostas de alocação de pilotos. Cada sub-canal é modulado com uma constelação QPSK. Usamos $f_m = 5 \times 10^{-3}$ e foram realizadas 40.000 transmissões de blocos de dados. A Tabela 4.1 resume esses parâmetros. Para todos as propostas usaremos o estimador ML simples. Não estamos interessados nesse ponto em averiguar as melhoras proporcionadas pelo uso do filtro de Wiener, e sim em ilustrar a qualidade intrínseca de cada solução. A fim de manter a comparação justa, o filtro ML simples foi escolhido.

As Figs. 4.10 e 4.11 mostram os resultados em MSE na frequência e em BER para 8 valores de E_b/σ distintos. Ambas também incluem um sistema de enlace aberto, a fim de termos um parâmetro de comparação com os sistemas estudados no Capítulo 3. A figura que mostra a probabilidade de erro, claramente, é mais importante. A primeira coisa que chama atenção é a superioridade do esquema proposto em relação aos demais para E_b/σ de valores médios e altos. O esquema alocação de potência e de Panah *et al.* não mostram ganhos perceptíveis em relação ao sistema de enlace aberto. A pré-equalização, como previsto, tem desempenho muito próximo do enlace aberto também. Apenas em E_b/σ mais elevados a pré-equalização piora um pouco, muito provavelmente devido ao ruído de distorção introduzido. A solução proposta tem desempenho significativamente melhor que todas as outras. Por exemplo, para uma BER de 10^{-2} , o esquema proposto economiza 5 dB de potência em relação às outras propostas.

Isso para E_b/σ altos, que fique claro. Para valores baixos, nenhuma solução obtém ganhos perceptíveis em BER em relação ao sistema de enlace aberto. Para o esquema proposto, isso ocorre provavelmente porque a estimativa do canal em qualquer instante é muito ruim. Observar que o vetor $\hat{\mathbf{q}}$ necessário para o cálculo da função-objetivo em (4-17) é a referência para a alocação de pilotos na transmissão seguinte. Caso $\hat{\mathbf{q}}$ seja uma estimação muito ruim do canal verdadeiro, não é garantido que os pilotos sejam alocados de forma a gerar ganhos. De forma geral, pode-se concluir que, para valores de E_b/σ baixos e para esse sistema específico, não há sentido em utilizar o enlace reverso.

É curioso analisar a Fig. 4.10 também, que plota o MSE de estimação da resposta de frequência do canal. De acordo com essa métrica, o método proposto tem o pior desempenho possível. Entretanto, como já discutido, o objetivo do sistema é detectar corretamente os dados, não estimar o canal da melhor forma possível. A solução proposta quer apenas uma estimação *boa o suficiente* para a detecção dos dados, e por isso o desempenho em MSE pode ser relaxado.

Experimento 2

Queremos agora quantificar a melhora no sistema com o uso do filtro de Wiener. Continuamos com estimador ML, agora funcionando em conjunto com um filtro de Wiener com $M = 50$. Todos os outros parâmetros da simulação são os mesmos daqueles já mostrado na Tabela 4.1.

O desempenho em MSE e BER são mostrados respectivamente nas Figs. 4.12 e 4.13. Comparamos o desempenho do enlace aberto, do algoritmo de

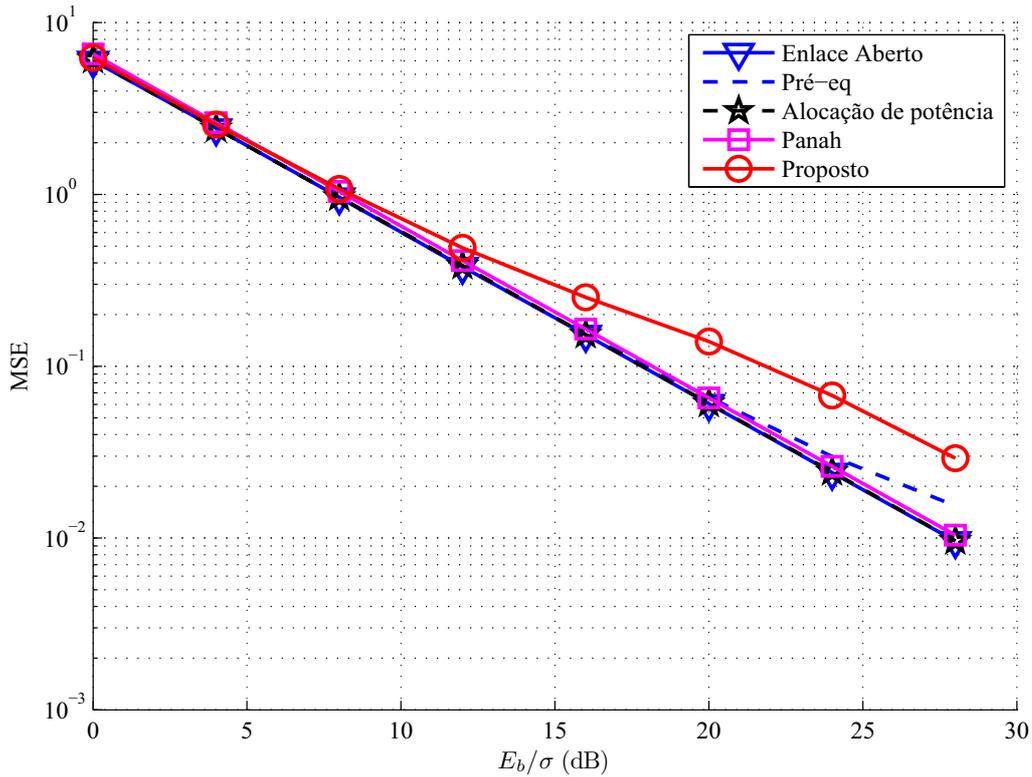


Figura 4.10: Erro médio quadrático de estimação do canal: comparação entre os diferentes esquemas de uso do enlace reverso. Estimador ML simples.

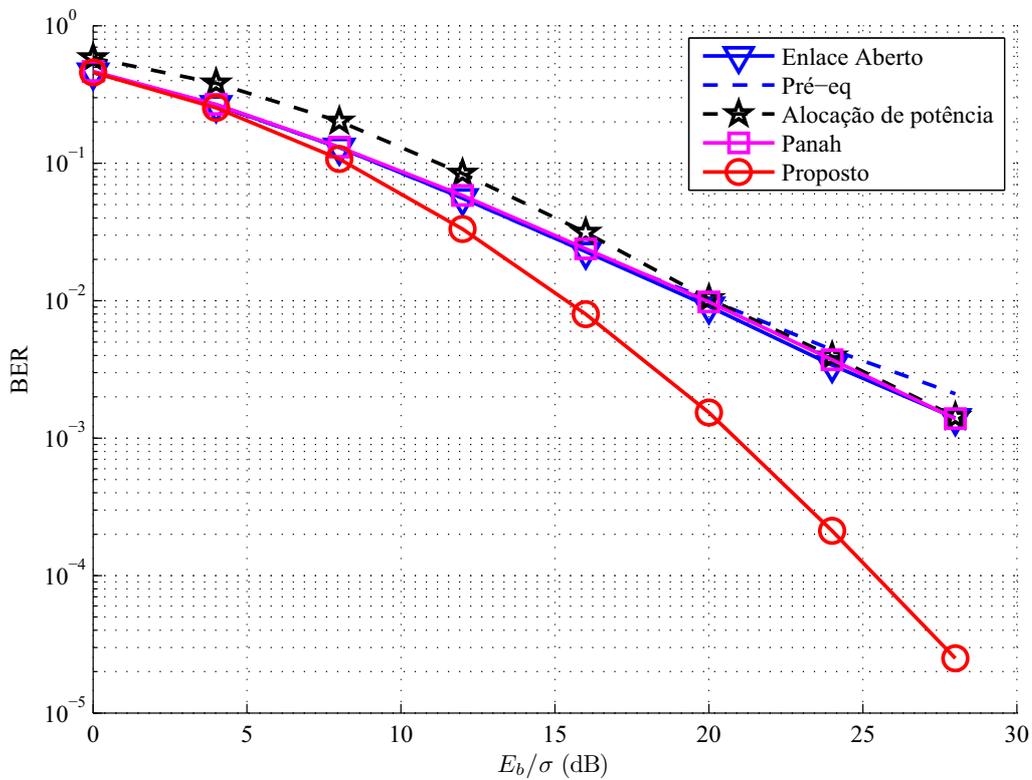


Figura 4.11: Probabilidade de erro de bit: comparação entre os diferentes esquemas de uso do enlace reverso. Estimador ML simples.

Panah e do algoritmo proposto nos casos com e sem filtro de Wiener. O desempenho dos sistemas de enlace aberto já havia sido ilustrado na Seção 3.4, e aqui confirmamos sua baixa eficácia em diminuir a probabilidade de erro de bit do sistema. Quanto à solução de Panah, observamos uma melhora tímida em BER do estimador ML mais Wiener em relação ao estimador ML simples. O algoritmo proposto possui melhora considerável com o uso do filtro de Wiener. Para uma BER de 10^{-2} , economiza-se cerca de 3,5 dB de potência. Mostramos também os resultados do BITE, que são excelentes. Em todos os pontos experimentados o desempenho do BITE é parelho com o da busca exaustiva.

Analizamos agora um outro aspecto interessante: ilustramos experimentalmente o que já havia sido comentado na Seção 4.3.2. Foi dito que, como consequência do Corolário 1, canais com alto desvanecimento (ou ganho baixo) eram explicitamente evitados na solução proposta. A Fig. 4.14 mostra o histograma dos módulos dos canais que carregam dados para as 40.000 transmissões de bloco da simulação para $E_b/\sigma = 20$ dB – ou seja, as curvas plotadas na Fig. 4.14 aproximam as fdp's em κ . A Fig. 4.14-A plota $p(\kappa)$ para transmissões com enlace aberto. Como nesse caso não se tem controle sobre a qualidade dos canais de dado, a fdp obtida no experimento (linha pontilhada) coincide com a fdp teórica (linha cheia), dada por (3-76). A solução de Panah com ML e Wiener mexe na fdp, como visto na Fig. 4.14-B, formando assim $p'(\kappa)$. Pelo fato de o problema proposto em 4-15 querer maximizar a SNR média entre os sub-canais de dados, o que se observa é que a parte mais central da fdp é que sofre deslocamento. Isso não influenciará tanto na probabilidade de erro final do sistema.

A solução proposta tem comportamento bastante diferente. Consideramos aqui, de novo, o estimador ML mais Wiener. Como indicado no Corolário 1, canais com ganho pequeno são deliberadamente evitados. Isso faz com que $p(\kappa)$ seja empurrado à direita, formando $p'(\kappa)$. Diferentemente da solução de Panah, a fdp é deslocada não apenas só da metade pra frente, e sim desde o começo. Uma $p'(\kappa)$ mais à direita fará com que a área a ser integrada para a probabilidade de erro seja menor (ver Seção 3.3).

A Fig. 4.15 mostra uma visão aproximada (um *close*) das fdp's da Fig. 4.14, dessa vez mostradas todas na mesma figura. Enquanto que a solução de Panah segue fielmente a $p(\kappa)$ original, a fdp resultante do Proposto evita os canais com ganho pequeno de forma consistente. Em quase absolutamente nenhuma transmissão de dado o canal que o carrega tem módulo menor do que 1/10.

Encerrando esse experimento, mostramos outro ângulo interessante: o do

histórico da alocação de pilotos no tempo (dessa vez para $E_b/\sigma = 16$ dB). As Figs. 4.16, 4.17, 4.18 e 4.19 mostram a evolução do canal e a maneira como os pilotos são alocados para o enlace aberto, Panah com ML, Panah com ML e Wiener e Proposto com ML e Wiener. O código de cores indica a qualidade do sub-canal. No eixo x , temos tempo e no eixo y os sub-canais. A localização dos pilotos é dada pelos retângulos pretos.

A alocação uniforme, mostrada na Fig. 4.16, é previsível. Panah com ML e Panah com ML e Wiener têm comportamento parecido, a não ser por pequenas diferenças. O resultado mais interessante é o do Proposto. De novo percebe-se que essa solução evita os canais piores mantendo uma relativa ordenação na alocação dos pilotos.

Experimento 3

Nesse experimento voltamos ao sistema da Seção 3.4, cujos parâmetros são dados na Tabela 3.2. Aqui $|\mathcal{P}|$ é da ordem de 10^{14} , o que torna as soluções que dependem de buscas exaustivas impraticáveis. Aplicamos a esse sistema, pois, somente o BITE e comparamos os resultados com os de um sistema de enlace aberto. Os resultados para MSE e BER são mostrados respectivamente nas Figs. 4.20 e 4.21 e os ganhos são bem consideráveis. Por exemplo, para uma BER de 10^{-2} , o BITE economiza cerca de 8 dB de potência em relação ao sistema de enlace aberto com ML e Wiener. Mais uma vez, para E_b/σ muito baixos, o BITE não consegue ganhos significativos.

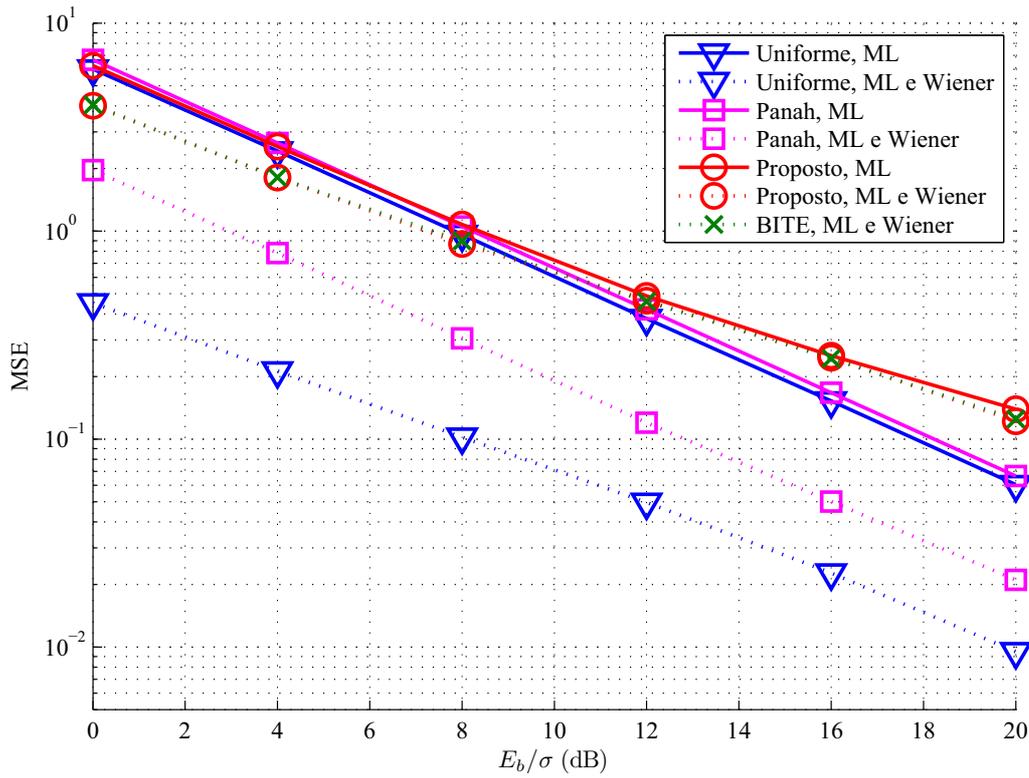


Figura 4.12: Erro médio quadrático: comparação entre as situações de estimador ML simples e ML mais filtro de Wiener.

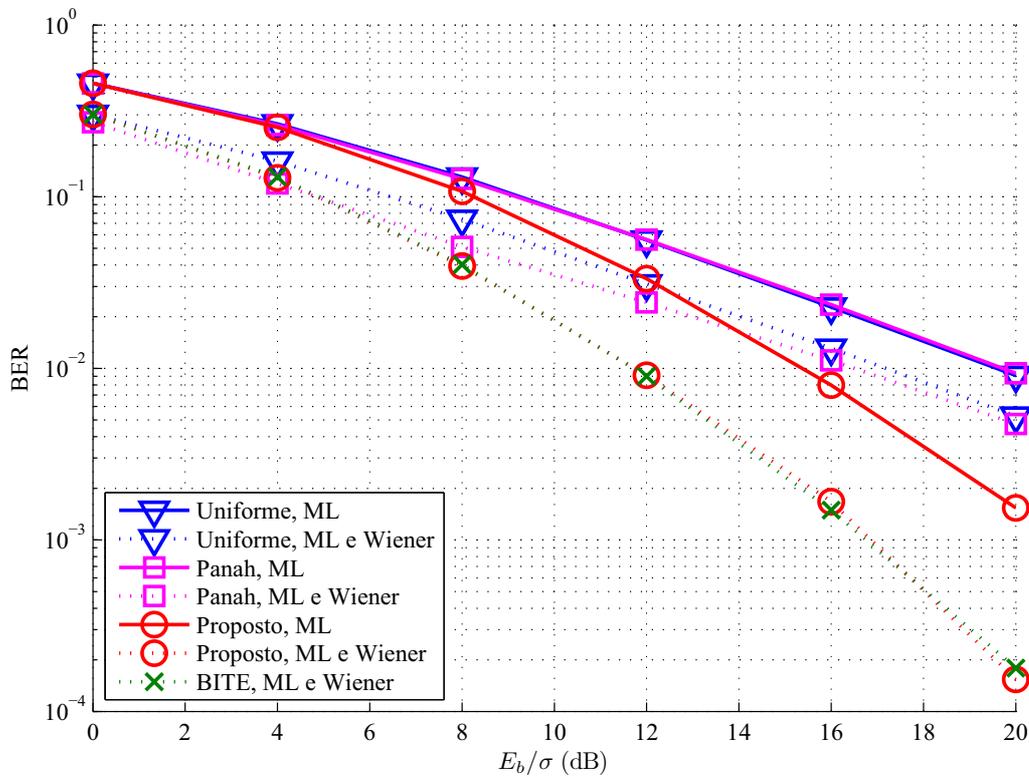


Figura 4.13: Probabilidade de erro de bit: comparação entre as situações de estimador ML simples e ML mais filtro de Wiener.

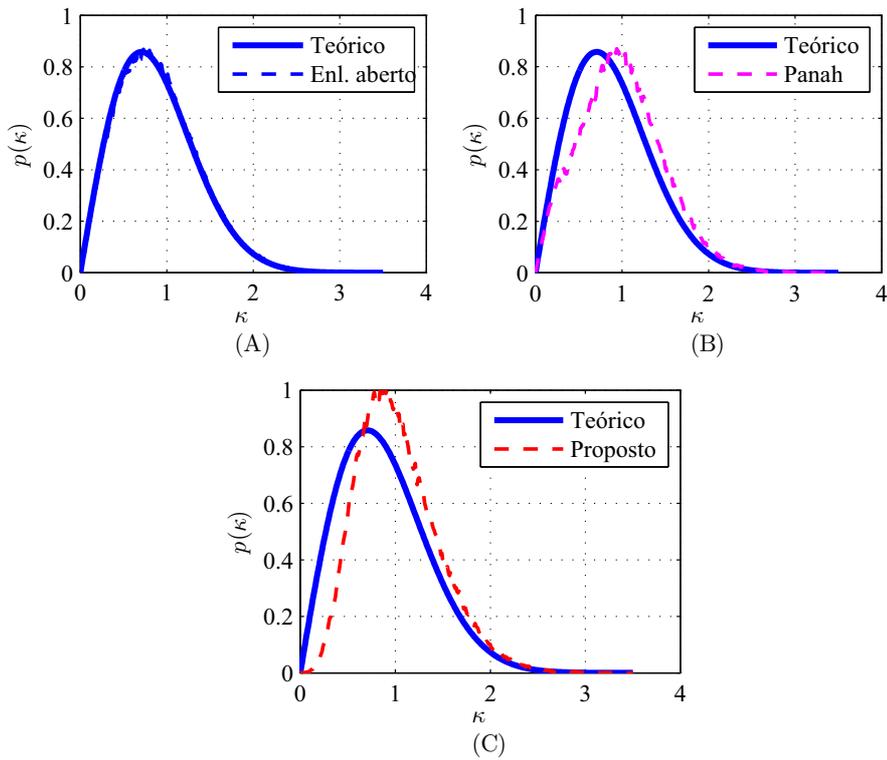


Figura 4.14: Histograma dos módulos dos canais que carregam dados para (A) enlace aberto, (B) Panah, ML e Wiener e (C) Proposto, ML e Wiener.

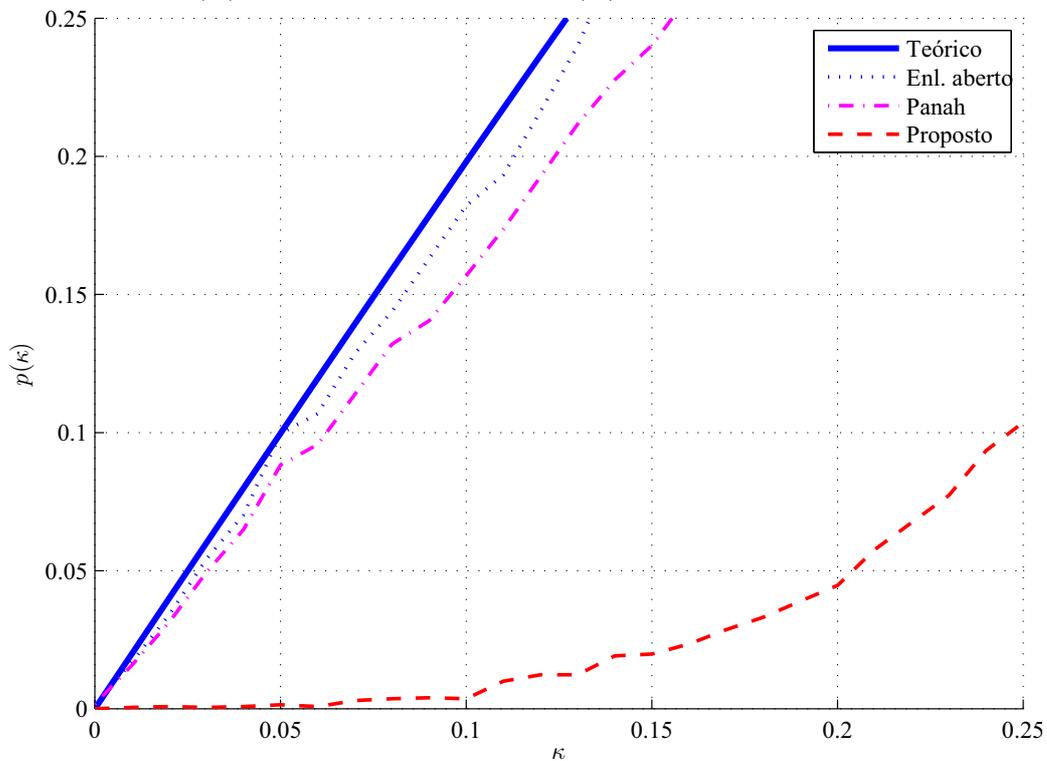


Figura 4.15: Visão aproximada do histograma.

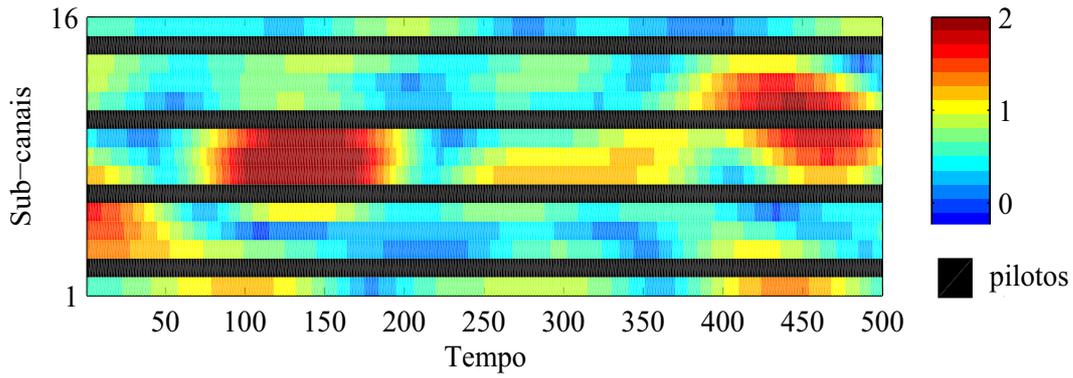


Figura 4.16: Grid para o enlace aberto.

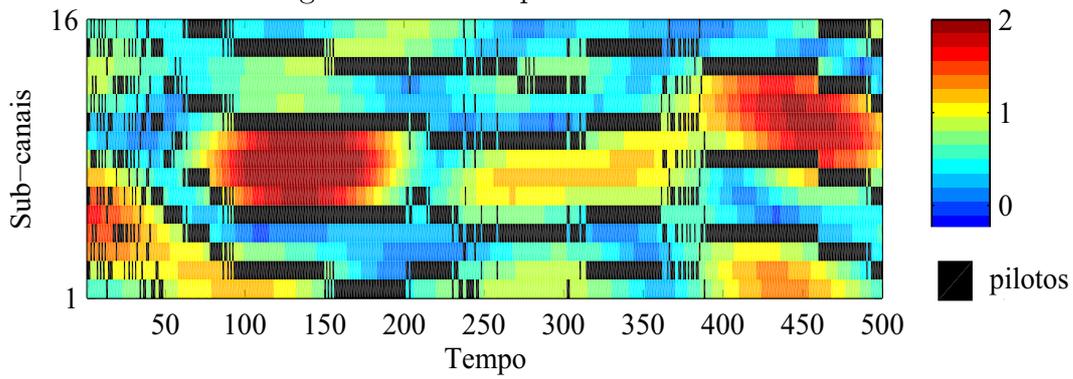


Figura 4.17: Grid para o Panah, ML.

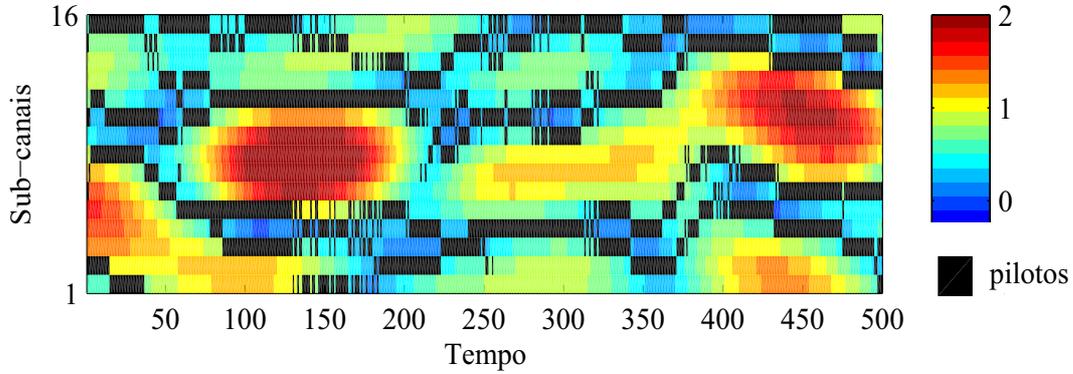


Figura 4.18: Grid para o Panah, ML e Wiener.

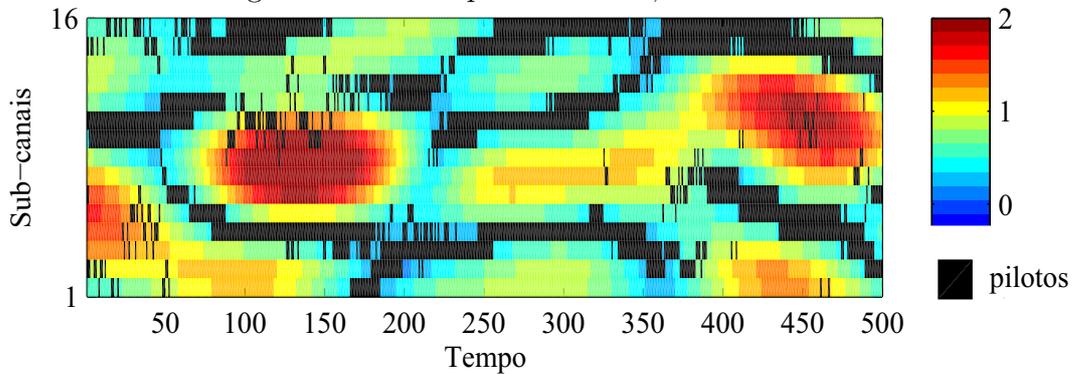


Figura 4.19: Grid para o Proposto, ML e Wiener.

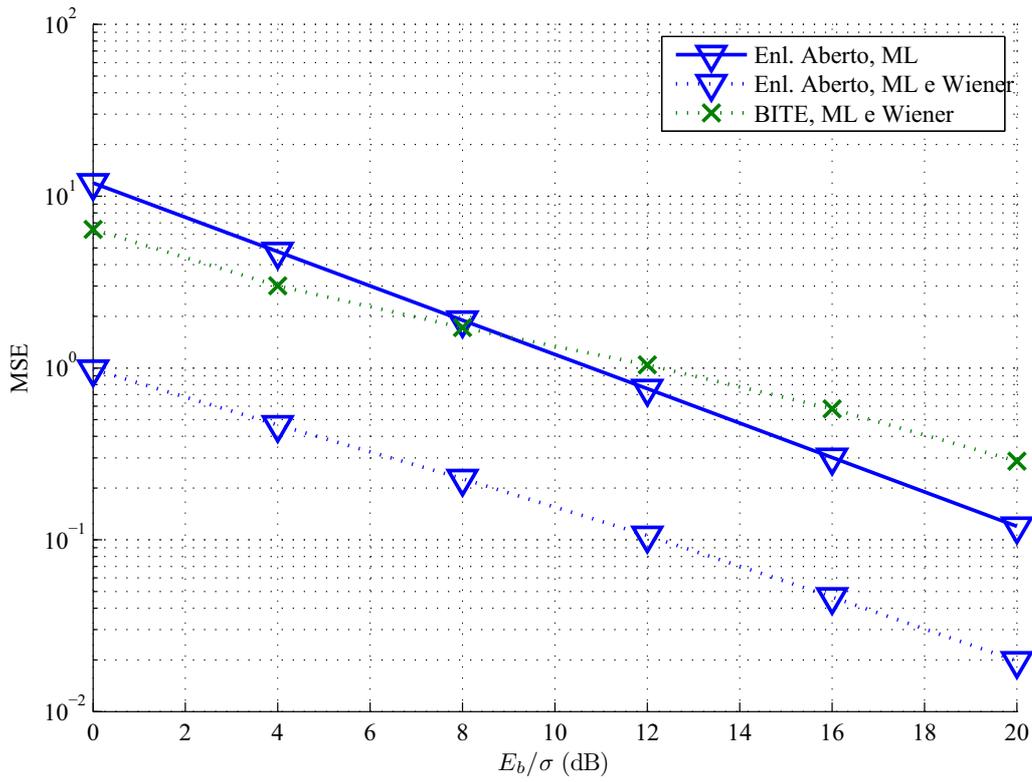


Figura 4.20: Erro médio quadrático: $K = 64$ e $K_p = 16$.

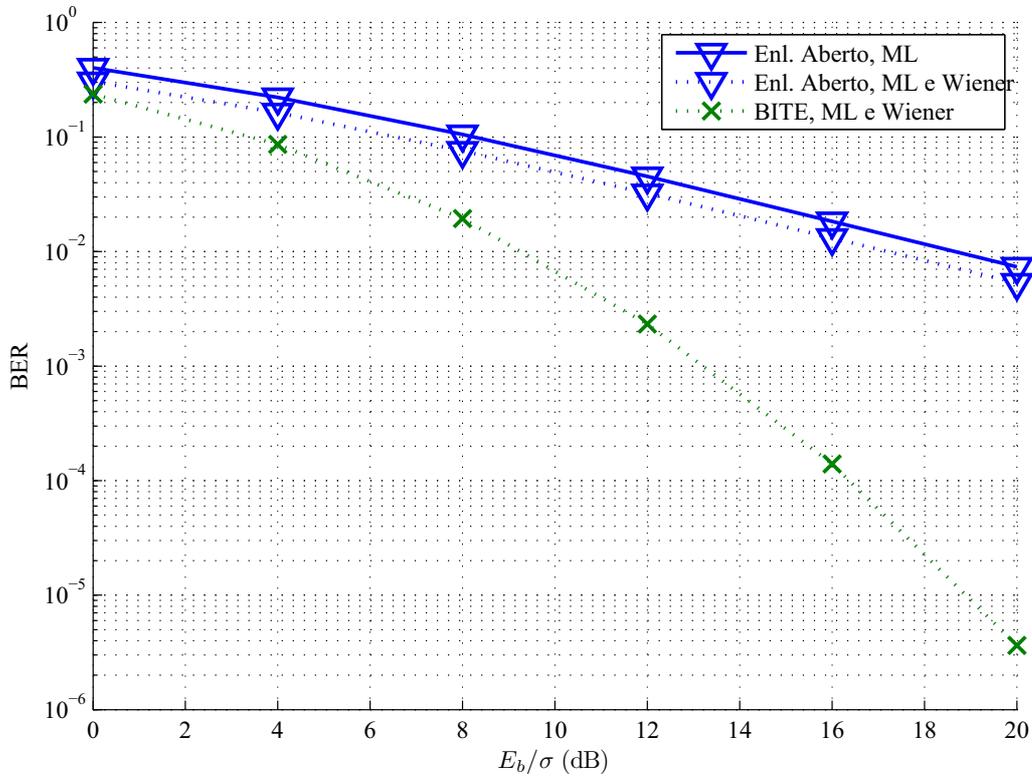


Figura 4.21: Probabilidade de erro de bit: $K = 64$ e $K_p = 16$.