

Equalização Cooperativa Cega de Canais Seletivos em Frequência Usando Filtros de Partículas Distribuídos

Claudio J. Bordin Jr. e Marcelo G. S. Bruno

Resumo—Introduz-se neste artigo um novo algoritmo para equalização cega de canais de comunicação seletivos em frequência usando filtros de partículas distribuídos. No cenário considerado, dois ou mais receptores distribuídos em uma rede física de nós remotos observam um mesmo sinal, independentemente distorcido de nó para nó, cooperando entre si através de trocas de mensagem de modo a equalizá-lo em presença de ruído aditivo. O novo algoritmo proposto difere de soluções anteriores pelo uso de aproximações paramétricas que permitem reduzir consideravelmente o custo de comunicação entre nós com custo computacional equivalente. Verifica-se através de simulações numéricas que o método cooperativo proposto apresenta desempenho consideravelmente superior àquele obtido pelos receptores quando operam isoladamente.

Palavras-Chave—Equalização Cega, Processamento Distribuído de Sinais, Filtros de Partículas, Grafos Probabilísticos.

Abstract—We introduce in this paper a novel particle-filter-based distributed algorithm for blind equalization of frequency-selective communication channels. In the considered scenario, multiple receivers distributed over different nodes in a network observe independently distorted versions of the same broadcast signal, and cooperate to recover it in the presence of additive noise. The new algorithm differs from previous methods by employing parametric approximations that lead to a significant reduction in the associated inter-node communication burden at an equivalent computational cost. We verify via numerical simulations that the proposed cooperative method greatly outperforms the non-cooperative isolated equalizers that are based only on local measurements available at each receiver in the network.

Keywords—Blind Equalization, Distributed Signal Processing, Particle Filters, Probabilistic Graphs.

I. INTRODUÇÃO

Em vários casos práticos de interesse, diferentes observações ou medidas relacionadas a um mesmo conjunto de parâmetros ou sinais ocultos podem estar disponíveis em diferentes nós remotos de uma rede física de múltiplos sensores ou receptores distribuídos no espaço. Nas arquiteturas ditas de *processamento distribuído* [1], [2], os diferentes receptores ou sensores nos vários nós da rede processam diretamente apenas as suas respectivas observações locais, mas cooperam entre si através de trocas de mensagens de forma a produzir uma estimativa global dos parâmetros ou

sinal ocultos, sem a necessidade de um processador central que execute fusão de dados.

Trabalhos anteriores na literatura, p.ex. [1], [2], introduziram algoritmos distribuídos que aproximam a estimativa *linear* de mínimo erro quadrático médio de um vetor de parâmetros ou sinal oculto observado por múltiplos receptores remotos. Os algoritmos em [1] e [2] não são adequados, entretanto, para a solução do problema de equalização distribuída de sinais digitais transmitidos através de canais seletivos em frequência, uma vez que, devido à não-gaussianidade dos sinais transmitidos e a variações aleatórias nos parâmetros dos canais, a estimativa ótima da sequência transmitida, que minimiza a taxa de erro de bit, em geral não coincide [3] e pode na realidade ser muito diferente das estimativas lineares de mínimos quadrados geradas por filtros de Kalman [1] ou filtros adaptativos [2] convencionais.

Em [4], preencheu-se essa lacuna da literatura através da introdução de novos algoritmos baseados em filtros de partículas [5] para equalização distribuída “cega” (i.e. não-assistida) de canais de *broadcast* digital. Trabalhos anteriores, p.ex., [6], [7], [8], incluíam propostas de filtros de partículas distribuídos, mas no contexto do problema de rastreamento de alvos, ao invés de comunicações digitais. Adicionalmente, os algoritmos em [6], [7], [8], supõem todos, como em [1], que os parâmetros do modelo de observação dos dados ocultos são perfeitamente conhecidos pelos filtros, enquanto, neste trabalho, supõe-se que os parâmetros dos canais de comunicação são desconhecidos e aleatórios.

Baseado na teoria de filtros de partículas [5], é possível garantir, em cada receptor, a convergência assintótica no número de partículas dos algoritmos em [4] para a estimativa de máximo a posteriori (MAP) [3] da sequência transmitida dados os sinais recebidos em *todos* os nós da rede, minimizando-se assim a taxa *global* de erro de bit. Os algoritmos em [4] têm o inconveniente, entretanto, de exigirem um alto custo de comunicação entre os receptores remotos, limitando assim sua aplicabilidade em cenários reais e reduzindo suas potenciais vantagens em relação a esquemas de processamento centralizado.

Neste artigo, procura-se contornar as limitações dos algoritmos em [4], introduzindo-se aproximações subótimas que permitem reduzir significativamente os requisitos de comunicação entre nós, às custas de uma degradação moderada de desempenho. O método proposto baseia-se em aproximações paramétricas particulares das mensagens transmitidas entre nós para cada partícula. A aproximação proposta assemelha-se a

Claudio J. Bordin Jr., Universidade Federal do ABC, Santo André - SP, e-mail: claudio.bordin@ufabc.edu.br. Marcelo G. S. Bruno, Instituto Tecnológico de Aeronáutica, São José dos Campos - SP, e-mail: bruno@ita.br. O trabalho do primeiro autor foi parcialmente financiado pela FAPESP.

técnicas usadas em [8], estendendo-as porém para um cenário mais complexo onde os parâmetros do modelo de observação do sinal são desconhecidos.

O restante deste artigo é dividido na seguinte forma: na Seção II, introduz-se o modelo de sinal, descrevendo-se na Seção III o algoritmo de equalização cega centralizado baseado em filtros de partículas proposto anteriormente em [10]. Na Seção IV, os algoritmos de equalização distribuídos de [4] são brevemente revistos. O novo algoritmo aproximado de equalização distribuída por filtros de partículas é então apresentado na Seção V, sendo o seu desempenho avaliado por meio de simulações na Seção VI. Finalmente, apresentam-se conclusões gerais e discussões na Seção VII.

II. DESCRIÇÃO DO PROBLEMA

Seja $\{b_n\}$ uma sequência binária, independente e identicamente distribuída (i.i.d)¹ e seja $\{x_n \in \{\pm 1\}\}$ uma sequência correspondente de símbolos obtidos a partir de $\{b_n\}$ por codificação diferencial [9]. A sequência $\{x_n\}$ dá origem no r -ésimo nó de uma rede de R receptores às observações $y_{r,0:n} \triangleq \{y_{r,0}, \dots, y_{r,n}\}$, obtidas como a saída de um canal FIR seletivo em frequência

$$y_{r,n} = \mathbf{h}_r^H \mathbf{x}_n + v_{r,n}, \quad (1)$$

onde $\mathbf{h}_r \in \mathbb{C}^{L \times 1}$ é um vetor que coleciona os coeficientes (assumidos invariantes no tempo) da resposta do canal ao pulso unitário, $\mathbf{x}_n \triangleq [x_n \dots x_{n-L+1}]^T$ e $v_{r,n}$ representa um processo ruído branco complexo gaussiano com média zero e variância σ_r^2 .

Os parâmetros \mathbf{h}_r e σ_r^2 , $1 \leq r \leq R$ são modelados como variáveis aleatórias, independentes para $r \neq s$ e distribuídos *a priori* de acordo com

$$p(\sigma_r^2) = \mathcal{IG}(\sigma_r^2 | \alpha; \beta) \triangleq \frac{\beta^\alpha (\sigma_r^2)^{-(\alpha+1)}}{\Gamma(\alpha)} \exp\left(-\frac{\beta}{\sigma_r^2}\right), \quad \sigma_r^2 > 0, \quad (2)$$

$$p(\mathbf{h}_r | \sigma_r^2) = \mathcal{N}_L(\mathbf{h}_r | 0; I \sigma_r^2 / \epsilon^2), \quad (3)$$

onde \mathcal{N}_L e \mathcal{IG} denotam, respectivamente, uma função normal complexa L -variada e uma função gama inversa, $\Gamma(\cdot)$ representa a função gama e $\{\alpha, \beta, \epsilon\}$ são hiperparâmetros do modelo.

Dado o modelo de sinal, tem-se como objetivo obter um algoritmo recursivo para o cálculo das estimativas MAP suavizadas

$$\hat{b}_{n-d} = \arg \max_{b_{n-d}} p(b_{n-d} | y_{1:R,0:n}), \quad (4)$$

sendo $d \geq 0$ e $y_{1:R,0:n} \triangleq \{y_{1,0:n} \dots y_{R,0:n}\}$.

III. EQUALIZAÇÃO CEGA POR FILTROS DE PARTÍCULAS

Filtros de partículas permitem que a função massa de probabilidade (f.m.p) *a posteriori* do n -ésimo bit transmitido

¹Neste artigo, usa-se também a notação $b_{n_1:n_2}$ para denotar a sequência $\{b_n, n_1 \leq n \leq n_2\}$.

b_n seja aproximada² por [5]

$$p(b_n | y_{1:R,0:n+d}) \approx \sum_{q=1}^Q w_n^{(q)} \mathcal{I}\{b_n = b_n^{(q)}\}, \quad (5)$$

em que $\mathcal{I}\{\cdot\}$ denota a função indicador, Q é o número de partículas $b_n^{(q)}$, amostradas aleatoriamente segundo a função de importância $\pi(\cdot)$ e $w_n^{(q)}$ são os respectivos pesos de importância. Usando o fato de que cada sequência distinta de bits $b_{-L:n-1}$ define unicamente após codificação uma correspondente sequência de símbolos transmitidos $x_{0:n-1}$, segue que a chamada função importância ótima [10]

$$\pi(b_n | b_{-L:n-1}, y_{1:R,0:n}) = p(\mathbf{x}_n | \mathbf{x}_{0:n-1}, y_{1:R,0:n}),$$

podendo ser calculada, por sua vez, como

$$p(\mathbf{x}_n | \mathbf{x}_{0:n-1}, y_{1:R,0:n}) = \frac{p(\mathbf{x}_n, \mathbf{x}_{0:n-1}, y_{1:R,0:n})}{\sum_{\mathbf{x}_n} p(\mathbf{x}_n, \mathbf{x}_{0:n-1}, y_{1:R,0:n})}. \quad (6)$$

Os pesos de importância correspondentes podem ser atualizados então pela recursão [10]

$$\begin{aligned} w_n^{(q)} &\propto w_{n-1}^{(q)} \sum_{\mathbf{x}_n} p(\mathbf{x}_n, y_{1:R,n} | \mathbf{x}_{0:n-1}, y_{1:R,0:n-1}) \\ &= w_{n-1}^{(q)} \sum_{\mathbf{x}_n} \frac{p(\mathbf{x}_n, \mathbf{x}_{0:n-1}, y_{1:R,0:n})}{p(\mathbf{x}_{0:n-1}, y_{1:R,0:n-1})}. \end{aligned} \quad (7)$$

Dada a independência *a priori* dos parâmetros desconhecidos dos diversos canais de comunicação para cada receptor, pode-se representar o problema de inferência distribuída em questão pelo grafo fator sem ciclos [11] da Fig. 1. Aplicando-se então o algoritmo soma-produto [11] a esse grafo fator, mostra-se que

$$p(\mathbf{x}_{0:n}, y_{1:R,0:n}) \propto \prod_{r=1}^R p(\mathbf{x}_{0:n}, y_{r,0:n}). \quad (8)$$

Por outro lado, usando os modelos *a priori* dos parâmetros do canal introduzidos na Seção II, pode-se mostrar após exaustivas manipulações algébricas que (vide [9] e Apêndice I),

$$\begin{aligned} p(\mathbf{x}_{0:n}, y_{r,0:n}) &= \int_{\mathbb{R}^+} \int_{\mathbb{C}^L} p(\mathbf{x}_{0:n}, y_{r,0:n}, \mathbf{h}_r, \sigma_r^2) d\mathbf{h}_r d\sigma_r^2 \\ &\propto [\beta_{r,n}]^{-\alpha_n} \prod_{i=0}^n \gamma_i^{-1}, \end{aligned}$$

onde

$$\gamma_n^{-1} \triangleq 1 + \mathbf{x}_n^H \Sigma_{n-1} \mathbf{x}_n$$

e Σ_n , $\beta_{r,n}$ e α_n são calculados recursivamente pelas expressões

$$\alpha_n = \alpha_{n-1} + 1, \quad (9)$$

$$\beta_{r,n} = \beta_{r,n-1} + \gamma_n^{-1} \|e_{r,n}\|^2, \quad (10)$$

$$\Sigma_n = \Sigma_{n-1} - \gamma_n^{-1} \Sigma_{n-1} \mathbf{x}_n \mathbf{x}_n^H \Sigma_{n-1}, \quad (11)$$

com $e_{r,n} \triangleq y_{r,n} - \bar{\mathbf{h}}_{r,n-1}^H \mathbf{x}_n$ e $\bar{\mathbf{h}}_{r,n}$ calculado sequencialmente por

$$\bar{\mathbf{h}}_{r,n} = \bar{\mathbf{h}}_{r,n-1} + \gamma_n^{-1} \Sigma_{n-1} \mathbf{x}_n e_{r,n}^*.$$

As recursões (9)-(11) são inicializadas fazendo $\alpha_{-1} = \alpha$, $\beta_{r,-1} = \beta$, $\bar{\mathbf{h}}_{r,-1} = 0$, and $\Sigma_{-1} = I \epsilon^{-2}$. Em um algoritmo

²Neste artigo, usa-se a letra minúscula p para denotar: (i) a função densidade de probabilidade de uma variável ou vetor aleatório contínuo e (ii) a função massa de probabilidade de uma variável aleatória discreta.

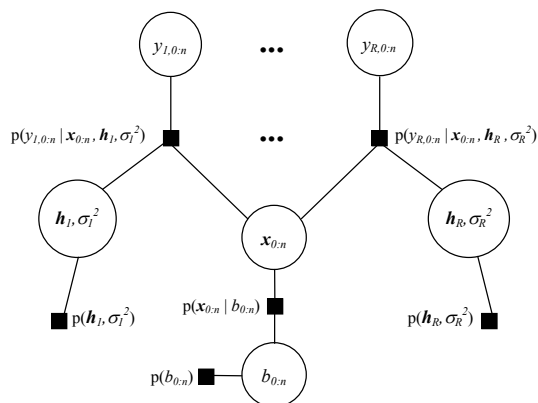


Fig. 1. Representação em grafo fator do problema de inferência distribuída tratado, com parâmetros dos canais aleatórios e desconhecidos.

centralizado, os R receptores localizados nos nós remotos da rede retransmitem as observações $y_{1:R,0:n}$ para o centro de fusão, que então calcula (8) usando (9)-(11).

IV. EQUALIZAÇÃO DISTRIBUÍDA ÓTIMA

Substituindo (8) em (6), pode-se reescrever a função de importância ótima como

$$p(\mathbf{x}_n | \mathbf{x}_{0:n-1}^{(q)}, y_{1:R,0:n}) = \frac{\prod_{r=1}^R \lambda_{r,n}^{(q)}(\mathbf{x}_n)}{\sum_{\mathbf{x}_n} \prod_{r=1}^R \lambda_{r',n}^{(q)}(\mathbf{x}_n)}, \quad (12)$$

onde $\lambda_{r,n}^{(q)}(\mathbf{x}_n) \triangleq p(\mathbf{x}_n, \mathbf{x}_{0:n-1}^{(q)}, y_{r,0:n})$. Analogamente, substituindo (8) em (7), vem que a regra de atualização dos pesos se reduz a

$$w_n^{(q)} \propto w_{n-1}^{(q)} \sum_{\mathbf{x}_n} \prod_{r=1}^R \frac{\lambda_{r,n}^{(q)}(\mathbf{x}_n)}{\lambda_{r,n-1}^{(q)}(\mathbf{x}_{n-1}^{(q)})}. \quad (13)$$

Em uma implementação distribuída imediata das Equações (12) e (13) [4], o r -ésimo receptor dispõe de um filtro de partículas local que, usando apenas observações locais, calcula $\lambda_{r,n}^{(q)}(\mathbf{x}_n)$, $q = 1, \dots, Q$, para todos os possíveis vetores \mathbf{x}_n (neste artigo, apenas dois para cada trajetória de partículas q , correspondendo respectivamente a $b_n = 1$ e $b_n = -1$). Os $2Q$ números reais resultantes são então transmitidos para os demais receptores que calculam (12) e atualizam os pesos de importância de acordo com (13). O algoritmo distribuído descrito neste parágrafo é denominado em [4] e, doravante neste artigo, algoritmo DcPF-II.

Como o algoritmo DcPF-II garante a mesma função de importância em todos os receptores, o uso de amostragem *sincronizada* [6], i.e. amostragem na mesma ordem a partir de geradores de números pseudoaleatórios com a mesma semente, resultará em conjuntos idênticos de partículas e, portanto, estimativas idênticas do sinal transmitido em todos os nós, sem a necessidade do uso de algoritmos iterativos de consenso [12],[13]. Quando o número de partículas Q tende a infinito, pode-se mostrar pela teoria de métodos Monte Carlo sequenciais [5] que a estimativa gerada pelo algoritmo DcPF-II, excluindo-se possíveis erros numéricos e incongruências entre o modelo de sinal e os dados reais, tem convergência assintótica garantida a cada instante n para a estimativa MAP centralizada do n -ésimo bit transmitido.

A. Função de Importância Localmente Ótima

O custo de comunicação associado ao algoritmo DcPF-II pode ser reduzido, *sem perda* de otimalidade assintótica no número de partículas, se substituirmos a função de importância globalmente ótima em (6) pela função de importância alternativa [4]

$$\pi(\mathbf{x}_n | \mathbf{x}_{0:n-1}^{(q)}, y_{1:R,0:n}) = \sum_{r=1}^R \mathcal{I}\{r=g(q)\} p(\mathbf{x}_n | \mathbf{x}_{0:n-1}^{(q)}, y_{r,0:n}) \quad (14)$$

onde $\mathcal{I}(\cdot)$ é novamente a função indicador e $g(\cdot)$ é uma função que associa de forma unívoca uma dada partícula q a um certo receptor r na rede. Matematicamente, (14) pode ser interpretada como uma *mistura* de R funções de importância *localmente ótimas* ponderadas pelos fatores $\mathcal{I}\{r=g(q)\}$, cuja soma sobre os índices r é igual a um para todo q . Substituindo (8) e (14) na regra de atualização de pesos [5]

$$w_n^{(q)} \propto w_{n-1}^{(q)} \frac{p(\mathbf{x}_{0:n}^{(q)}, y_{1:R,0:n})}{p(\mathbf{x}_{0:n-1}^{(q)}, y_{1:R,0:n-1}) \pi(\mathbf{x}_n | \mathbf{x}_{0:n-1}^{(q)}, y_{1:R,0:n})},$$

segue que

$$w_n^{(q)} \propto w_{n-1}^{(q)} \mu_{g(q),n}^{(q)} \prod_{r=1, r \neq g(q)}^R \Phi_{r,n}^{(q)} \quad (15)$$

onde

$$\Phi_{r,n}^{(q)} \triangleq \frac{p(\mathbf{x}_{0:n}^{(q)}, y_{r,0:n})}{p(\mathbf{x}_{0:n-1}^{(q)}, y_{r,0:n-1})}, \quad \mu_{r,n}^{(q)} \triangleq p(y_{r,n} | \mathbf{x}_{0:n-1}^{(q)}, y_{r,0:n-1}).$$

Em uma implementação distribuída de (14), ao invés de amostrar todas as partículas $q = 1, \dots, Q$, o r -ésimo receptor amostra apenas Q/R partículas do conjunto $\{q | g(q) = r\}$, usando a função de importância $p(\mathbf{x}_n | \mathbf{x}_{0:n-1}^{(q)}, y_{r,0:n})$, que depende apenas das observações locais $y_{r,0:n}$. As partículas sorteadas (Q/R bits) são então transmitidas para os demais nós da rede. Após receber as partículas amostradas pelos outros receptores, o nó r calcula e retransmite para a rede, por sua vez, as grandezas $p(x_{0:n}^{(l)}, y_{r,0:n})$, $\{l | g(l) \neq r\}$ ($Q - Q/R$ números reais). Finalmente, o r -ésimo nó calcula e transmite para a rede os pesos de importância ótimos obtidos segundo (15) para todo $\{q | g(q) = r\}$ (ou seja, Q/R números reais). O custo global de comunicação, i.e., Q números reais (mais Q/R bits) por nó por símbolo transmitido é igual, portanto, a aproximadamente metade dos $2Q$ números reais por nó por símbolo exigidos pelo algoritmo anterior (DcPF-II).

No restante deste artigo, segue-se a terminologia de [4], sendo as implementações distribuídas das equações (14) e (15) referidas como o algoritmo DcPF-III.

V. FILTRO DE PARTÍCULAS DISTRIBUÍDO APROXIMADO (ADPF)

Apesar de sua otimalidade assintótica, os algoritmos DcPF-II e DcPF-III descritos na Seção IV apresentam um custo de comunicação entre nós elevado que dificulta seu uso em problemas reais de engenharia. O custo de comunicação

pode ser significativamente reduzido, porém, se adotarmos simplificações adicionais que permitam eliminar a dependência dos coeficientes $\lambda_{r,n}^{(q)}$ em (12) e (13) no índice sobre-escrito q .

Para esse fim, observe primeiro que, explorando-se a independência *a priori* dos parâmetros dos canais, pode-se mostrar que

$$p(\mathbf{h}_r, \sigma_r^2 | \mathbf{x}_{0:n}, y_{1:R,0:n-1}) = p(\mathbf{h}_r, \sigma_r^2 | \mathbf{x}_{0:n-1}, y_{1:R,0:n-1}), \quad (16)$$

$$p(\mathbf{h}_r, \sigma_r^2 | \mathbf{x}_{0:n-1}, y_{1:R,0:n-1}) = p(\mathbf{h}_r, \sigma_r^2 | \mathbf{x}_{0:n-1}, y_{r,0:n-1}). \quad (17)$$

Por outro lado, pelos resultados do Apêndice A, tem-se que

$$\begin{aligned} p(\mathbf{h}_r | \sigma_r^2, \mathbf{x}_{0:n-1}, y_{r,0:n-1}) &= \mathcal{N}(\mathbf{h}_r | \bar{\mathbf{h}}_{r,n-1}^{(q)}, \sigma_r^2 \Sigma_{n-1}^{(q)}) \\ p(\sigma_r^2 | \mathbf{x}_{0:n-1}, y_{r,0:n-1}) &= \mathcal{IG}(\sigma_r^2 | \alpha_{n-1}, \beta_{r,n-1}^{(q)}), \end{aligned}$$

sendo $\Sigma_n^{(q)}$, $\bar{\mathbf{h}}_{r,n}^{(q)}$, α_n e $\beta_{r,n}^{(q)}$ obtidos através das recursões em (9)-(11).

Empregando uma representação Monte Carlo para $p(\mathbf{x}_{0:n-1} | y_{1:R,0:n-1})$ composta pelo conjunto de amostras $\{\mathbf{x}_{0:n-1}^{(q)}\}$ com pesos $\{w_{n-1}^{(q)}\}$ e usando (17), é imediato verificar que $p(\mathbf{h}_r, \sigma_r^2 | y_{1:R,0:n-1})$ pode ser aproximada pela soma ponderada

$$\begin{aligned} p(\mathbf{h}_r, \sigma_r^2 | y_{1:R,0:n-1}) &\approx \sum_{q=1}^Q w_{n-1}^{(q)} \mathcal{IG}(\sigma_r^2 | \alpha_{n-1}, \beta_{r,n-1}^{(q)}) \\ &\quad \times \mathcal{N}(\mathbf{h}_r | \bar{\mathbf{h}}_{r,n-1}^{(q)}, \sigma_r^2 \Sigma_{n-1}^{(q)}). \quad (18) \end{aligned}$$

Para reduzir o custo de comunicação associado ao algoritmo distribuído, propomos substituir $p(\mathbf{h}_r, \sigma_r^2 | \mathbf{x}_{0:n-1}, y_{r,0:n-1})$ pela aproximação paramétrica $\tilde{p}(\mathbf{h}_r, \sigma_r^2 | \mathbf{x}_{0:n-1}, y_{r,0:n-1}) = \mathcal{N}(\mathbf{h}_r | \tilde{\mathbf{h}}_{r,n-1}, \sigma_r^2 \tilde{\Sigma}_{r,n-1}) \mathcal{IG}(\sigma_r^2 | \tilde{\alpha}_{r,n-1}, \tilde{\beta}_{r,n-1})$, onde os parâmetros (\cdot) são *invariantes* com $\mathbf{x}_{0:n-1}$ e escolhidos de modo que a função densidade de probabilidade (f.d.p) conjunta aproximada tenha média e variâncias associadas que coincidem com aquelas obtidas a partir da soma ponderada³ (18).

Explorando independências condicionais mostradas na Fig. 1 e fazendo cálculos semelhantes aos descritos no Apêndice I, obtém-se a função de verossimilhança aproximada

$$\begin{aligned} \tilde{p}(y_{r,n} | \mathbf{x}_{0:n}, y_{r,0:n-1}) &= \iint p(y_{r,n} | \mathbf{h}_r, \sigma_r^2, \mathbf{x}_{0:n}, y_{r,0:n-1}) \\ &\quad \times \tilde{p}(\mathbf{h}_r, \sigma_r^2 | \mathbf{x}_{0:n}, y_{r,0:n-1}) d\mathbf{h}_r d\sigma_r^2 \quad (19) \\ &= \frac{\tilde{\gamma}_n^{-1} \tilde{\beta}_{r,n-1}^{\tilde{\alpha}_{r,n-1}}}{(\tilde{\beta}_{r,n-1} + \tilde{\gamma}_n^{-1} \|\mathbf{y}_{r,n} - \tilde{\mathbf{h}}_{r,n-1}^H \mathbf{x}_n\|^2)^{(\tilde{\alpha}_{r,n-1}+1)}}, \quad (20) \end{aligned}$$

onde $\tilde{\gamma}_n \triangleq 1 + \mathbf{x}_n^H \tilde{\Sigma}_{r,n-1} \mathbf{x}_n$. Por outro lado, pela aplicação iterativa da Lei de Bayes, vem que

$$\tilde{\lambda}_{r,n}(\mathbf{x}_n) \triangleq \tilde{p}(\mathbf{x}_{0:n}, y_{r,0:n}) \propto \prod_{i=0}^n \tilde{p}(y_{r,i} | \mathbf{x}_{0:i}, y_{r,0:i-1}). \quad (21)$$

³Note que (18) pode ser informalmente interpretada como a “média” sobre todas as possíveis realizações de $\mathbf{x}_{0:n-1}$ da f.d.p $p(\mathbf{h}_r, \sigma_r^2 | \mathbf{x}_{0:n-1}, y_{r,0:n-1})$.

Escolhem-se a seguir os parâmetros $\tilde{\alpha}_{r,n-1}$ e $\tilde{\beta}_{r,n-1}$ de modo que a média e variância associadas à densidade marginal

$$p(\sigma_r^2 | y_{1:R,0:n-1}) \approx \sum_{q=1}^Q w_{n-1}^{(q)} \mathcal{IG}(\sigma_r^2 | \alpha_{n-1}, \beta_{r,n-1}^{(q)}), \quad (22)$$

coincidam com aquelas de $\mathcal{IG}(\sigma_r^2 | \tilde{\alpha}_{n-1}, \tilde{\beta}_{r,n-1})$. Após alguns cálculos algébricos, segue que

$$\begin{aligned} \tilde{\alpha}_{r,n-1} &= 2 + \tilde{\mathbb{E}}_{n-1}^2[\sigma_r^2] / \tilde{\mathbb{V}}\text{AR}_{n-1}[\sigma_r^2] \\ \tilde{\beta}_{r,n-1} &= (\tilde{\alpha}_{r,n-1} - 1) \tilde{\mathbb{E}}_{n-1}[\sigma_r^2], \end{aligned}$$

onde $\tilde{\mathbb{E}}_{n-1}[\sigma_r^2]$ e $\tilde{\mathbb{V}}\text{AR}_{n-1}[\sigma_r^2]$ são dados por

$$\tilde{\mathbb{E}}_{n-1}[\sigma_r^2] = \frac{\sum_{q=1}^Q w_{n-1}^{(q)} \beta_{r,n-1}^{(q)}}{\alpha_{n-1} - 1}, \quad (23)$$

$$\tilde{\mathbb{V}}\text{AR}_{n-1}[\sigma_r^2] = \frac{\sum_{q=1}^Q [w_{n-1}^{(q)} (\beta_{r,n-1}^{(q)})^2]}{(\alpha_{n-1} - 1)(\alpha_{n-1} - 2)} - \tilde{\mathbb{E}}_{n-1}^2[\sigma_r^2]. \quad (24)$$

Infelizmente, não é tão simples achar os parâmetros de $\mathcal{N}(\mathbf{h}_r | \tilde{\mathbf{h}}_{r,n-1}, \sigma_r^2 \tilde{\Sigma}_{r,n-1})$ de modo a obter um casamento exato com os momentos de primeira e segunda ordens de $p(\mathbf{h}_r | \sigma_r^2, y_{1:R,0:n-1})$. Para contornar essa dificuldade, propõe-se uma simplificação adicional, em que $\tilde{\mathbf{h}}_{r,n-1}$ e $\sigma_r^2 \tilde{\Sigma}_{r,n-1}$ são calculados de forma a coincidir, respectivamente, com o vetor média e com a matriz de covariância associadas a uma aproximação de $p(\mathbf{h}_r | y_{1:R,0:n-1})$ obtida substituindo-se a densidade gama inversa em (18) pela função Delta de Dirac $\delta(\sigma_r^2 - \tilde{\mathbb{E}}_{n-1}[\sigma_r^2])$. Isto equivale a ignorar a incerteza sobre σ_r^2 , fazendo esse parâmetro igual à sua estimativa Monte Carlo no instante $n-1$ (variável no tempo). O procedimento descrito anteriormente implica que

$$\begin{aligned} \tilde{\mathbf{h}}_{r,n-1} &= \sum_{q=1}^Q w_{n-1}^{(q)} \bar{\mathbf{h}}_{r,n-1}^{(q)}, \quad \tilde{\Sigma}_{r,n-1} = \sum_{q=1}^Q \left\{ w_{n-1}^{(q)} \times \right. \\ &\quad \left. \times \left[\Sigma_{n-1}^{(q)} + \frac{\bar{\mathbf{h}}_{r,n-1}^{(q)} \bar{\mathbf{h}}_{r,n-1}^{(q)H}}{\tilde{\mathbb{E}}_{n-1}[\sigma_r^2]} \right] \right\} - \frac{\tilde{\mathbf{h}}_{r,n-1} \tilde{\mathbf{h}}_{r,n-1}^H}{\tilde{\mathbb{E}}_{n-1}[\sigma_r^2]}. \quad (25) \end{aligned}$$

Na sequência, descreve-se o algoritmo ADPF proposto. Inicialmente, o r -ésimo receptor calcula (21) no instante n para todos os 2^L possíveis valores de \mathbf{x}_n e transmite os valores calculados para a rede⁴. Após receber os dados transmitidos pelos demais receptores, o r -ésimo receptor calcula a função de importância aproximada

$$\tilde{p}(\mathbf{x}_n | \mathbf{x}_{0:n-1}, y_{1:R,0:n}) = \frac{\prod_{s=1}^R \bar{\lambda}_{s,n}^{(q)}(\mathbf{x}_n)}{\sum_{\mathbf{x}_n} \prod_{s'=1}^R \bar{\lambda}_{s',n}^{(q)}(\mathbf{x}_n)}, \quad (26)$$

e obtém os respectivos pesos usando a expressão

$$w_n^{(q)} \propto w_{n-1}^{(q)} \sum_{\mathbf{x}_n} \prod_{s=1}^R \frac{\bar{\lambda}_{s,n}^{(q)}(\mathbf{x}_n)}{\bar{\lambda}_{s,n-1}^{(q)}(\mathbf{x}_{n-1})} \quad (27)$$

onde $\bar{\lambda}_{s,n}^{(q)}(\mathbf{x}_n) \triangleq p(\mathbf{x}_n, \mathbf{x}_{0:n-1}, y_{r,0:n})$ se $r = s$, (i.e. a quantidade exata), ou $\bar{\lambda}_{s,n}^{(q)}(\mathbf{x}_n) \triangleq \tilde{\lambda}_{s,n}(\mathbf{x}_n)$, se $r \neq s$. Em implementações práticas de (25), usam-se adicionalmente

⁴Note que, para L grande, pode-se reduzir o custo de comunicação para $O(L^2)$ sem aproximações adicionais transmitindo-se $\tilde{\mathbf{h}}_{r,n-1}$, $\tilde{\Sigma}_{r,n-1}$, $\tilde{\alpha}_{r,n-1}$, $\tilde{\beta}_{r,n-1}$ e $y_{r,n}$, e calculando-se (21) no receptor remoto.

técnicas de *pivotagem* [10] para eliminar a ambiguidade de fase no vetor $\bar{\mathbf{h}}_{r,n-1}^{(q)}$.

Note que (26) e (27) são versões modificadas, respectivamente, das Equações (12) e (13) na Seção IV. A diferença, porém, é que, ao contrário do que acontecia em (12) e (13), o termo $\bar{\lambda}_{s,n}^{(q)}(\cdot)$ em (26) e (27) torna-se *independente* de q para $s \neq r$, permitindo assim a redução nos requisitos de comunicação entre nós. Observe, entretanto, que a faixa de valores admissíveis para \mathbf{x}_n em (26), correspondendo respectivamente a um bit transmitido $b_n = +1$ ou $b_n = -1$, varia de acordo com a trajetória anterior de amostras $\mathbf{x}_{0:n-1}^{(q)}$. Assim, ainda que os termos $\bar{\lambda}_{s,n}^{(q)}(\cdot) = \tilde{\lambda}_{s,n}(\cdot)$ independam de q para $s \neq r$, eles têm que ser calculados em pontos \mathbf{x}_n diferentes para diferentes valores de q . Analogamente, o conjunto de vetores \mathbf{x}_n válidos sobre o qual é feita a soma em (27) também varia com q .

VI. RESULTADOS EXPERIMENTAIS

O desempenho do algoritmo ADPF foi avaliado através de simulações Monte Carlo com duzentas realizações independentes. Para cada realização, foi calculada a taxa de erro de bit (BER) média como função da relação sinal-ruído E_B/N_0 , definida a seguir. Em cada experimento, foi transmitida uma sequência aleatória i.i.d de 300 bits, sendo os primeiros 150 bits descartados no cálculo da BER para assegurar que a convergência temporal do algoritmo fosse atingida.

Para fins de comparação, foram testados nas mesmas condições os algoritmos DcPF-II e DcPF-III descritos na Seção IV. O sistema de comunicação simulado tem $R = 2$ receptores e todos os filtros operam com $Q = 300$ partículas. Em todos os algoritmos, emprega-se reamostragem residual (sincronizada) [6] em todas as iterações. Os canais simulados \mathbf{h}_r tem $L = 3$ coeficientes (ordem do canal conhecida) e são obtidos amostrando-se independentemente, para cada receptor e em cada realização Monte Carlo, de uma densidade gaussiana complexa circularmente simétrica com média zero e matriz de covariância $\Lambda = \text{diag}(2, 1, 0.5)$. Os vetores \mathbf{h}_r amostrados são então normalizados de modo que $\|\mathbf{h}_r\|^2 = 1$. A variância σ_r^2 do ruído do r -ésimo é escolhida, por sua vez, de forma que $E_B/N_0 = \|\mathbf{h}_r\|^2/\sigma_r^2$ assuma o valor desejado. Note que esse procedimento de simulação resulta, propositadamente, em parâmetros \mathbf{h}_r e σ_r^2 com distribuições *a priori* diferentes das supostas na Seção II, permitindo assim avaliar-se a robustez do algoritmo a incongruências entre o modelo de sinal e os dados. Finalmente, os hiperparâmetros do modelo de sinal foram empiricamente escolhidos iguais a $\alpha = 3$, $\beta = 0.1$ e $\epsilon = 1$.

Os resultados obtidos nas simulações são mostrados na Fig. 2. Para efeitos de comparação, mostra-se na mesma figura a BER média obtida por dois receptores que operam independentemente e não cooperam entre si para estimar o sinal transmitido. A BER média obtida pelo o algoritmo BCJR centralizado [11] operando com *conhecimento perfeito* dos parâmetros dos canais também é mostrada no mesmo gráfico, constituindo um limitante inferior para as curvas de probabilidade de erro. Como se pode observar, os algoritmos cooperativos em rede (DcPF-II, DcPF-III e ADPF) têm desempenhos muito superiores àquele dos receptores operando

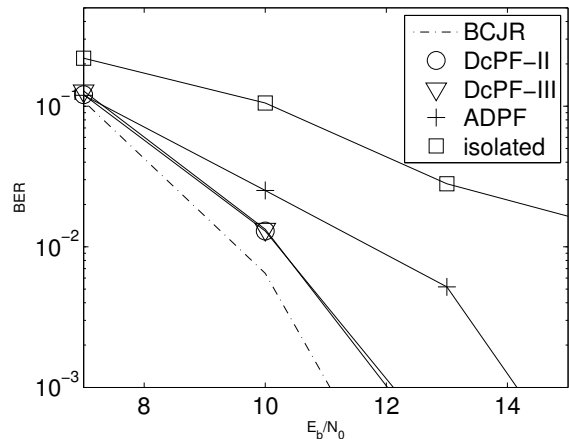


Fig. 2. Taxa de erro de bit (BER) média estimada em 200 experimentos independentes.

isoladamente. A degradação de desempenho do algoritmo distribuído aproximado (ADPF, subótimo) aumenta na medida em que diminui a variância do ruído do canal, sendo igual a aproximadamente 2 dB para uma taxa de erro de bit de 10^{-2} .

VII. CONCLUSÕES

Introduziu-se em [4] uma solução para o problema de equalização cega distribuída de canais de comunicação digital seletivos em frequência em sistemas *broadcast* (i.e. “um para muitos”). Os algoritmos DcPF-II e DcPF-III em [4] têm convergência garantida, caso o número de partículas tenda ao infinito, para a estimativa MAP centralizada ótima dos dados transmitidos. Entretanto, tanto o algoritmo DcPF-II como o algoritmo DcPF-III exigem uma capacidade de comunicação muito elevada entre os diversos receptores localizados em nós remotos da rede distribuída. Neste artigo, contornou-se essa limitação com a introdução de um novo algoritmo aproximado para equalização cega distribuída (ADPF) que, no exemplo simulado, permitiu reduzir o custo de comunicação de $2Q = 600$ números reais por nó por bit para $2^L = 8$ números reais por nó por bit. Por outro lado, o custo computacional do algoritmo ADPF proposto tem ordem de grandeza semelhante ao dos algoritmos DcPF-II e DcPF-III ótimos, i.e., aproximadamente equivalente ao custo computacional de Q filtros de Kalman por nó por bit transmitido. O estudo teórico da convergência do algoritmo ADPF, porém, ainda permanece um problema em aberto. Em simulações numéricas com duzentas realizações Monte Carlo independentes, o algoritmo ADPF apresentou boas propriedades empíricas de convergência, com uma degradação de desempenho em relação aos algoritmos ótimos (DcPF-II e DcPF-III) da ordem de 2 dB em relações sinal-ruído na faixa média a alta.

REFERÊNCIAS

- [1] A. Ribeiro, G. B. Giannakis, e S. I. Roulmeliotis, “SOI-KF: Distributed Kalman filtering with low-cost communications using the sign of innovations,” *IEEE Trans. on Sig. Proc.*, vol.54, n.12, pp. 4782-4795, 2006.

- [2] C. G. Lopes e A. H. Sayed, "Incremental adaptive strategies over distributed networks," *IEEE Trans. on Sig. Proc.*, vol.55, n.8, pp. 4064-4077, 2007.
- [3] S. M. Kay, *Fundamentals of Statistical Signal Processing, Vol.I: Estimation Theory*, Prentice-Hall Signal Processing Series, A. V. Oppenheim (editor), Prentice-Hall Inc., 1993.
- [4] C. J. Bordin Jr. e M. G. S. Bruno, "Cooperative blind equalization of frequency-selective channels in sensor networks using decentralized particle filtering," in *42nd Asilomar Conf. on Sign., Syst. and Comp.*, 2008.
- [5] A. Doucet, S. J. Godsill, e C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Stat. Comput.*, vol.10, pp. 197-208, 2000.
- [6] M. J. Coates, "Distributed particle filters for sensor networks," in *Third Intl. Symp. on Inf. Process. in Sensor Networks*, pp. 99-107, 2004.
- [7] X. Xeng e Y.-H. Hu, "Distributed particle filters for wireless sensor network target tracking," *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Process.*, vol.IV, pp.845-848, 2005.
- [8] M. Vemula, M. F. Bugallo, e P. M. Djuric, "Target tracking by fusion of random measures," *Signal, Image and Video Process.*, vol.1, n.2, pp.149-161, 2007.
- [9] C. J. Bordin Jr. e M. G. S. Bruno, "Bayesian blind equalization of time-varying frequency-selective channels subject to unknown variance noise," *Proc. IEEE Intl. Conf. on Acoustics, Speech, and Sig. Process.*, pp. 3449-3452, 2008.
- [10] C. J. Bordin Jr. e M. G. S. Bruno, "Particle filters for joint blind equalization and decoding in frequency-selective channels," *IEEE Trans. on Sig. Proc.*, vol.56, n.6, pp.2395-2405, 2008.
- [11] F. R. Kschischang, B. J. Frey, e H.-A. Loelinger, "Factor graphs and the Sum-Product algorithm," *IEEE Trans. Inf. Th.*, vol.47, n.2, pp.498-519, 2001.
- [12] I. D. Schiozas, A. Ribeiro, e G. B. Giannakis, "Consensus in Ad Hoc WSNs with noisy links, Part I: Distributed estimation of deterministic signals," *IEEE Trans. on Sig. Proc.*, vol.56, n.1, pp.342-356, 2008.
- [13] I. D. Schiozas, A. Ribeiro, S. I. Roumeliotis, e G. B. Giannakis, "Consensus in Ad Hoc WSNs with noisy links, Part II: Distributed estimation and smoothing of random signals," *IEEE Trans. on Sig. Proc.*, vol.56, n.4, pp.1650-1666, 2008.

APÊNDICE I

CÁLCULO DE $p(\mathbf{x}_{0:n}, y_{r,0:n})$

Para calcular $p(\mathbf{x}_{0:n}, y_{r,0:n})$, observe primeiro que

$$p(\mathbf{x}_{0:n}, y_{r,0:n}) = p(y_{r,n} | \mathbf{x}_{0:n}, y_{r,0:n-1}) \times p(\mathbf{x}_n | \mathbf{x}_{0:n-1}, y_{r,0:n-1}) p(\mathbf{x}_{0:n-1}, y_{r,0:n-1}). \quad (28)$$

O termo $p(\mathbf{x}_n | \mathbf{x}_{0:n-1}, y_{r,0:n-1}) = p(\mathbf{x}_n | \mathbf{x}_{0:n-1})$ no lado direito de (28) assume o mesmo valor para todo \mathbf{x}_n válido, uma vez que a sequência $\{b_n\}$ é assumida i.i.d. Por outro lado, o primeiro termo à direita em (28) é dado por

$$\begin{aligned} p(y_{r,n} | \mathbf{x}_{0:n}, y_{r,0:n-1}) &= \\ &= \int \int p(y_{r,n}, \mathbf{h}_r, \sigma_r^2 | \mathbf{x}_{0:n}, y_{r,0:n-1}) d\mathbf{h}_r d\sigma_r^2. \\ &= \int \int p(y_{r,n} | \mathbf{h}_r, \sigma_r^2, \mathbf{x}_{0:n}, y_{r,0:n-1}) p(\mathbf{h}_r | \sigma_r^2, \mathbf{x}_{0:n}, y_{r,0:n-1}) \\ &\quad \times p(\sigma_r^2 | \mathbf{x}_{0:n}, y_{r,0:n-1}) d\mathbf{h}_r d\sigma_r^2. \end{aligned} \quad (29)$$

Dados os modelos *a priori* para os parâmetros do canal e lembrando que

$$p(y_{r,n} | \mathbf{h}_r, \sigma_r^2, \mathbf{x}_{0:n}, y_{r,0:n-1}) = \mathcal{N}(y_{r,n} | \mathbf{h}_r^H \mathbf{x}_n, \sigma_r^2), \quad (30)$$

mostra-se, após cálculos algébricos tediosos [9], que

$$p(\mathbf{h}_r | \sigma_r^2, \mathbf{x}_{0:n}, y_{r,0:n-1}) = \mathcal{N}(\mathbf{h}_r | \bar{\mathbf{h}}_{r,n-1}, \sigma_r^2 \Sigma_{r,n-1}), \quad (31)$$

$$p(\sigma_r^2 | \mathbf{x}_{0:n}, y_{r,0:n-1}) = \mathcal{IG}(\sigma_r^2 | \alpha_{n-1}, \beta_{r,n-1}), \quad (32)$$

onde $\bar{\mathbf{h}}_{r,n}$, $\Sigma_{r,n}$, α_n e $\beta_{r,n}$ são calculados pelas recursões (9)-(11). Substituindo (30)-(32) em (29) e lembrando que

$$\mathcal{N}(y_{r,n} | \mathbf{h}_r^H \mathbf{x}_n, \sigma_r^2) \mathcal{N}(\mathbf{h}_r | \bar{\mathbf{h}}_{r,n-1}, \sigma_r^2 \Sigma_{r,n-1}) = \mathcal{N}(y_{r,n} | \bar{\mathbf{h}}_{r,n}^H \mathbf{x}_n, \sigma_r^2 \gamma_n) \mathcal{N}(\mathbf{h}_r | \bar{\mathbf{h}}_{r,n}, \sigma_r^2 \Sigma_{r,n}), \quad (33)$$

segue que a integral em \mathbf{h}_r pode ser calculada analiticamente. Para integrar em relação a σ_r^2 , basta reconhecer que $\mathcal{N}(y_{r,n} | \bar{\mathbf{h}}_{r,n}^H \mathbf{x}_n, \sigma_r^2 \gamma_n) \mathcal{IG}(\sigma_r^2 | \alpha_{n-1}, \beta_{r,n-1})$ é, a menos de uma constante multiplicativa, uma função gama inversa em σ^2 , restando apenas identificar os termos de normalização necessários. Procedendo-se dessa forma, obtém-se [9]

$$p(y_{r,n} | \mathbf{x}_{0:n}, y_{r,0:n-1}) \propto \frac{1}{\gamma_n} \cdot \frac{\beta_{r,n-1}^{\alpha_{n-1}}}{\beta_{r,n}^{\alpha_n}}, \quad (34)$$

que, substituído em (28), leva à conclusão de que

$$p(\mathbf{x}_{0:n}, y_{r,0:n}) \propto \beta_{r,n}^{-\alpha_n} \prod_{i=0}^n \gamma_i^{-1}. \quad (35)$$