

Detecção de Atividade para Canal de Acesso Aleatório Massivo Gaussiano Utilizando Sensoriamento Compressivo

Henrique Pickler da Silva, Gustavo Kasper Facenda e Danilo Silva

Resumo— Este trabalho investiga métodos eficientes de transmissão para o canal de acesso aleatório gaussiano com um número massivo de usuários. Em um trabalho recente, um método *grant-based* com resultados promissores comparado a métodos *grantless* foi proposto. Este artigo é baseado nesse método e foca em aprimorar o estágio de detecção de atividade. O sistema proposto utiliza da interpretação do acesso aleatório como um problema de sensoriamento compressivo para obter uma detecção eficiente. Os resultados de simulação mostram que, para um número suficientemente grande de usuários ativos, o esquema proposto tem um desempenho significativamente superior ao estado-da-arte.

Palavras-Chave— Acesso aleatório, Acesso massivo, Canal gaussiano, Sensoriamento compressivo.

Abstract— This paper investigates efficient methods for the random access Gaussian channel with a massive number of users. A recent work has proposed a grant-based method that has shown promising results compared to grantless methods. This paper is based on that approach and focuses on improving the activity detection stage. The proposed system takes advantage of the compressive sensing interpretation of the problem, leveraging it to achieve an efficient detection. Simulation results show that the proposed scheme is able to significantly outperform the state-of-art for a large enough number of active users.

Keywords— Random access, Massive access, Gaussian channel, Compressive sensing.

I. INTRODUÇÃO

Nos recentes anos, o interesse por Internet das Coisas vem crescendo e tomando mais importância na área de telecomunicações. Um caso específico de nosso interesse é a comunicação massiva de máquinas (*massive machine-type communication*), caracterizada por um canal com uma grande quantidade de usuários que transmitem infrequentemente pequenos pacotes com baixa potência.

Em [1], os autores propõem um modelo de canal para transmissão não-coordenada e resultados de alcançabilidade para o modelo proposto. Neste artigo, mostrou-se que métodos existentes na época, como ALOHA e *Treat Interference as Noise*, eram ineficientes para uma grande quantidade de usuários, em comparação ao resultado proposto. Em busca de atingir o resultado teórico apresentado com uma complexidade realizável, vários artigos propuseram métodos *grantless*, isto é,

Henrique Pickler da Silva, Departamento de Engenharia Elétrica, Universidade Federal de Santa Catarina, Florianópolis-SC, e-mail: henrypcklers@gmail.com; Gustavo Kasper Facenda, Electrical and Computer Engineering, University of Toronto, Toronto, e-mail: gustavo.k.facenda@gmail.com; Danilo Silva, Departamento de Engenharia Elétrica, Universidade Federal de Santa Catarina, Florianópolis-SC, e-mail: danilo.silva@ufsc.br.

em que os usuários transmitem seus dados sem coordenação, com resultados progressivamente melhores [2], [3], [4], [5].

Em [6], os autores argumentam que o foco dado apenas para métodos *grantless* não é completamente justificado, e que métodos *grant-based*, isto é, em que os usuários transmitem seus dados em recursos alocados pela estação base, de maneira que a transmissão de dados pode ser coordenada, podem atingir resultados tão bons quanto, se não melhores, dado que estes sejam cuidadosamente projetados levando em conta o modelo e o número de usuários, como é feito com os métodos *grantless*. No artigo, os autores propõem um modelo de canal que generaliza o modelo em [1] e permite comparação teórica entre os dois tipos de transmissão, e apresentam um esquema que atinge bons resultados.

Mais recentemente, em [7], os autores propuseram um método *grantless* que supera [6] e, no momento, representa o estado-da-arte deste problema.

Neste artigo, utilizamos o mesmo modelo apresentado em [6]. O modelo divide a transmissão em três estágios: Sinalização de atividade, *feedback* e transmissão de dados. Desta forma, o acesso não-coordenado ocorre apenas no primeiro estágio do sistema, e no último estágio é possível realizar acesso coordenado. No entanto, o método utilizado na fase de sinalização de atividade tem algumas limitações, como uma degradação artificial do canal e é baseado em códigos algébricos, não necessariamente ótimos para o canal de múltiplo acesso Gaussiano.

Em busca de melhorar a eficiência do sistema, neste artigo, propomos a utilização de métodos de sensoriamento compressivo para a etapa de sinalização de atividade. A relação entre o problema de acesso aleatório e sensoriamento compressivo já havia sido observada em [1] e é a base de artigos como [4], [3], no entanto, há limitações de complexidade quanto ao seu uso quando consideramos uma mensagem completa. No entanto, ao utilizar sensoriamento compressivo apenas no primeiro estágio, a codificação e decodificação se tornam realizáveis.

Utilizamos uma matriz de sensoriamento gaussiana e implementamos um sistema de sensoriamento compressivo [8], [9], [10] utilizando algoritmos como ISTA (*Iterative soft thresholding algorithm*), AMP [11] (*Approximate message passing*) e VAMP [12] (*Vector approximate message passing*), este último utilizado no nosso método final, devido ao seu melhor desempenho.

Também apresentamos um otimizador, semelhante a [6], a fim de obter bons parâmetros de projeto para o sistema proposto. Os resultados utilizando estes parâmetros são com-

parados à literatura e mostramos que é possível alcançar resultados superiores aos de [7] utilizando o sistema proposto, para um número de usuários ativos suficientemente grande.

II. PRELIMINARES

Nesta seção, revisamos a abordagem apresentada em [6], na qual o nosso método se baseia.

A. Modelo do Canal

Seja $K_{\text{tot}} \rightarrow \infty$ o número total de usuários. A cada sessão de transmissão, a qual consiste de N usos de canal, um subconjunto \mathcal{A} de usuários estão ativos, com $|\mathcal{A}| = K_a$. Sem perda de generalidade, assumimos que $\mathcal{A} = \{1, 2, \dots, K_a\}$. Em cada sessão, cada usuário $i \in \mathcal{A}$ deseja transmitir sua mensagem $w_i \in \{1, \dots, 2^k\}$, composta de k bits, para a estação base. Assumimos que não existe coordenação *a priori*. Uma consequência imediata disto é que, no início de cada sessão, todos os usuários devem transmitir utilizando o mesmo codificador, possivelmente randomizado.¹

Cada sessão é dividida em três estágios. Primeiramente, há um estágio de *uplink*, em que cada usuário i transmite um sinal $\mathbf{x}_i \in \mathbb{R}^{N_1}$. A estação base recebe o sinal

$$\mathbf{y} = \sum_{i=1}^{K_a} \mathbf{x}_i + \mathbf{z} \quad (1)$$

onde $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$ e $\|\mathbf{x}_i\|^2 \leq N_1 P_1$.

Em seguida, há um estágio de *downlink*, em que a estação base gera um sinal $\mathbf{x}^{[f]} \in \mathbb{R}^{N_f}$, utilizando a informação decodificada de \mathbf{y} , e o transmite por difusão para todos os usuários. Cada usuário recebe um sinal de *feedback* dado por

$$\mathbf{y}_i^{[f]} = \mathbf{x}^{[f]} + \mathbf{z}_i \quad (2)$$

onde $\mathbf{z}_i \sim \mathcal{N}(0, \mathbf{I})$ e $\|\mathbf{x}^{[f]}\|^2 \leq N_f P_f$.

Finalmente, no terceiro estágio, cada usuário i transmite um sinal $\mathbf{x}_i^{[2]} \in \mathbb{R}^{N_2}$, gerado utilizando informações obtidas de $\mathbf{y}_i^{[f]}$ e \mathbf{x}_i . Novamente, a estação base recebe

$$\mathbf{y}^{[2]} = \sum_{i=1}^{K_a} \mathbf{x}_i^{[2]} + \mathbf{z}^{[2]} \quad (3)$$

onde $\mathbf{z}^{[2]} \sim \mathcal{N}(0, \mathbf{I})$ e $\|\mathbf{x}_i^{[2]}\|^2 \leq N_2 P_2$.

Ao fim dos três estágios de transmissão, a estação base deve gerar uma lista $\mathcal{L} \subseteq \{1, \dots, 2^k\}$ que contém a estimativa da estação base de quais mensagens foram transmitidas. A probabilidade de erro é definida como

$$\epsilon = \frac{1}{K_a} \sum_{i=1}^{K_a} \Pr[w_i \notin \mathcal{L}]. \quad (4)$$

Note que esta é uma definição de probabilidade de erro média *por usuário*, em que um erro ocorre para um usuário i se a sua mensagem w_i não estiver na lista \mathcal{L} . Note também que esta probabilidade apenas considera se a mensagem é recuperada ou não, isto é, não se exige que decodificador identifique o usuário que transmitiu tal mensagem.

¹Essa propriedade permite considerar esquemas como o ALOHA, em que a mesma mensagem pode ser transmitida em *slots* distintos escolhidos aleatoriamente, correspondendo, portanto, a codificações diferentes quando se leva em conta o sinal produzido em N usos de canal.

1) *Métricas de Desempenho*: As métricas de desempenho consideradas são a razão

$$\frac{E_b}{N_0} = \frac{P_1 N_1 + P_f N_f / K_a + P_2 N_2}{2k} \quad (5)$$

o número total de usos de canal $N = N_1 + N_f + N_2$ e a probabilidade de erro (4).

Note que, uma vez fixados N e erro total máximo $\epsilon_{\text{max}} \in \mathbb{R}$ e exigindo-se $\epsilon \leq \epsilon_{\text{max}}$, a métrica de interesse se reduz a E_b/N_0 . Além disso, conforme argumentado em [6], ao fixarmos estes parâmetros, fixamos o uso de recursos de canal e a latência da comunicação, permitindo comparar de forma justa esquemas *grantless* e *grant-based*.

B. Sistema de Transmissão

1) *Primeiro Estágio*: No primeiro estágio do sistema, os K_a usuários ativos transmitem, de maneira não-ortogonal, uma coluna escolhida aleatoriamente de uma matriz $\mathbf{A} \in \mathbb{R}^{N_1 \times M}$. A escolha da coluna é feita usando os $k_1 = \lfloor \log_2(M) \rfloor$ primeiros bits de mensagem (e, possivelmente, mais alguma aleatoriedade extra, se M não for uma potência de 2).

Sob essas hipóteses, o canal (1) pode ser expresso na forma

$$\mathbf{y} = \mathbf{A}\mathbf{b} + \mathbf{z} \quad (6)$$

onde $\mathbf{b} \in \mathbb{N}^M$ é um vetor com $\|\mathbf{b}\|_1 = K_a$ que identifica quais colunas de \mathbf{A} foram escolhidas para transmissão pelos usuários. Mais precisamente, o elemento b_j representa o número de usuários que escolheram a j -ésima coluna de \mathbf{A} para transmissão.

De posse de \mathbf{y} , e conhecendo a matriz \mathbf{A} , a estação base produz uma estimativa do vetor \mathbf{b} , denotada por $\hat{\mathbf{b}} \in \{0, 1\}^M$, para a qual exige-se $\|\hat{\mathbf{b}}\|_1 = K_a$.

Não revisaremos aqui os métodos de construção da matriz \mathbf{A} e de detecção propostos em [6], visto que nosso sistema utiliza métodos diferentes para ambos.

2) *Segundo Estágio*: No segundo estágio, a estação base distribui os recursos para os usuários detectados no primeiro estágio, a fim de possibilitar uma transmissão ortogonal no último estágio. Com o objetivo de informar aos usuários qual recurso foi destinado a cada usuário, a estação base gera uma mensagem composta de $k_f = \lceil \log_2 \binom{M}{K_a} \rceil$ bits, e então codifica esta mensagem utilizando um código para canal AWGN ponto-a-ponto de comprimento N_f , com potência P_f . Este sinal é transmitido pelo canal (2) a todos os usuários, os quais por sua vez tentam decodificá-lo.

3) *Terceiro Estágio*: No terceiro estágio, cada usuário utiliza sua estimativa de $\mathbf{x}^{[f]}$ para escolher em que recurso irá transmitir. Visto que o usuário transmitiu k_1 bits no primeiro estágio, a mensagem agora contém os $k_2 = k - k_1$ bits restantes. Cada usuário codifica esta mensagem utilizando um código para canal AWGN ponto-a-ponto com comprimento $n_{c,2} = N_2/K_a$ e potência P_2 .

C. Análise da Probabilidade de Erro

Abaixo descrevemos os possíveis eventos de erro e respectivas expressões para suas probabilidades. Note que, como a probabilidade de erro é definida por usuário, as probabilidades

abaixo são calculadas para um usuário i que transmite a j -ésima coluna da matriz \mathbf{A} no primeiro estágio. Uma dedução detalhada pode ser encontrada em [6].

- **Erro de colisão:** Ocorre quando outro usuário $i' \neq i$ escolhe a mesma j -ésima coluna para transmitir. Supondo (por simplicidade) $M = 2^{k_1}$, a probabilidade deste evento é dada por

$$\epsilon_c = 1 - \left(1 - \frac{1}{M}\right)^{K_a - 1}. \quad (7)$$

- **Erro de sensoriamento** (detecção de atividade): Ocorre quando não há erro de colisão na coluna j mas ainda assim a estação base falha em detectá-la. A probabilidade deste evento é dada por

$$\epsilon_s = P\{\{\hat{\mathbf{b}}_j \neq 1\} \cap \{\mathbf{b}_j = 1\}\} \quad (8)$$

e depende da matriz \mathbf{A} e do algoritmo de detecção utilizado pela estação base.

- **Erro de feedback:** Corresponde a um erro de decodificação em (2). A probabilidade deste evento pode ser limitada superiormente por

$$\epsilon_f = Q\left(\left(C_{\text{AWGN}}(P_f) - \frac{k_f}{N_f}\right) \frac{\sqrt{N_f}}{\sqrt{V_{\text{AWGN}}(P_f)}}\right) \quad (9)$$

onde $C_{\text{AWGN}}(P)$ e $V_{\text{AWGN}}(P)$ denotam, respectivamente, a capacidade e a dispersão de um canal AWGN ponto-a-ponto com razão sinal-ruído P [13]. Sabe-se que (9) é alcançável [13].

- **Erro de transmissão:** Corresponde a um erro de decodificação em (3). De forma similar, a probabilidade deste evento pode ser limitada superiormente por

$$\epsilon_t = Q\left(\left(C_{\text{AWGN}}(P_2 K_a) - \frac{k_2}{n_{c2}}\right) \frac{\sqrt{n_{c2}}}{\sqrt{V_{\text{AWGN}}(P_2 K_a)}}\right). \quad (10)$$

- **Erro de alocação de recursos:** Ocorre quando um usuário $i' \neq i$ decodifica erroneamente o feedback e transmite no recurso destinado ao usuário i . A probabilidade deste evento pode ser limitada superiormente por

$$\epsilon_{f,2} = 1 - \left(1 - \frac{\epsilon_f}{K_a}\right)^{K_a - 1} \leq \frac{K_a - 1}{K_a} \epsilon_f \leq \epsilon_f. \quad (11)$$

Finalmente, a probabilidade de erro por usuário é limitada por $\epsilon \leq \epsilon_s + \epsilon_c + 2\epsilon_f + \epsilon_t$.

III. DETECÇÃO COM SENSORIAMENTO COMPRESSIVO

Analisando (7), observa-se que, para se obter uma probabilidade de colisão baixa, é necessário que M seja significativamente maior que K_a . Desta forma, (6) pode ser visto como um problema de sensoriamento compressivo, uma vez que \mathbf{b} é, por construção, esparso. Com esta interpretação do problema, propomos que a decodificação seja realizada resolvendo-se o problema de otimização

$$\tilde{\mathbf{b}} = \underset{\mathbf{b} \in \mathbb{R}^M}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{A}\mathbf{b}\|_2^2 + \lambda \|\mathbf{b}\|_1 \quad (12)$$

onde λ é um parâmetro que controla a relação entre a esparsidade de $\tilde{\mathbf{b}}$ e a fidelidade de reconstrução.

Esta interpretação também guia a construção da matriz \mathbf{A} (conhecida nesse contexto como matriz de sensoriamento). Na literatura de sensoriamento compressivo, é conhecido que matrizes aleatórias gaussianas com elementos i.i.d. são boas matrizes de sensoriamento [9]. No que segue, assumimos a utilização de uma matriz \mathbf{A} com tal distribuição. Para consistência com a literatura, reescalamos o canal (1) por um fator $1/\sqrt{P_1 N_1}$, assim podemos assumir que os elementos de \mathbf{A} são gerados com variância $1/N_1$.

Como a aplicação pretendida deve funcionar em tempo real, ao invés de buscar resolver (12) de forma exata, focamos no uso de algoritmos iterativos de baixa complexidade, alguns dos quais apresentam desempenho próximo do ótimo. A seguir, apresentamos uma revisão de destes algoritmos, bem como uma comparação entre eles aplicados ao nosso problema.

Antes de prosseguir, é importante ressaltar que, em geral, a estimativa $\tilde{\mathbf{b}}$ obtida em (12) possui componentes quaisquer em \mathbb{R} . Para garantir $\tilde{\mathbf{b}} \in \{0, 1\}^M$, com $\|\mathbf{b}\|_1 = K_a$, atribui-se $\hat{b}_j = 1$ se \tilde{b}_j é um dos K_a maiores elementos de $\tilde{\mathbf{b}}$, caso contrário atribui-se $\hat{b}_j = 0$. Esta mesma finalização é aplicada a todos os algoritmos descritos abaixo.

A. Iterative Soft Thresholding Algorithm (ISTA)

Este algoritmo é inicializado com $\mathbf{b}_0 = \mathbf{0}$ e calcula, iterativamente para $t = 0, 1, \dots, T - 1$,

$$\mathbf{v}_t = \mathbf{y} - \mathbf{A}\mathbf{b}_t \quad (13)$$

$$\mathbf{r}_t = \mathbf{b}_t + \beta \mathbf{A}^T \mathbf{v}_t \quad (14)$$

$$\mathbf{b}_{t+1} = \eta_t(\mathbf{r}_t) \quad (15)$$

onde $\eta_t(\cdot) : \mathbb{R} \rightarrow \mathbb{R}$ é uma função de limiar suave, ou *denoiser*, definida por

$$\eta_t(r) = \operatorname{sign}(r) \max(|r| - \lambda_t, 0) \quad (16)$$

e estendida a vetores componente-a-componente, e β e λ_t são parâmetros de projeto. Tipicamente, escolhe-se $\lambda_t = \alpha \tau_t$, onde

$$\tau_t = \frac{1}{n} \|\mathbf{v}_t\|^2 \quad (17)$$

para algum $\alpha > 0$. A estimativa final é obtida como $\tilde{\mathbf{b}} = \mathbf{b}_T$.

Uma dificuldade prática no uso desse algoritmo é a necessidade de escolher α e β manualmente.

B. Approximate Message Passing (AMP)

Este algoritmo é similar ao ISTA, exceto que se fixa $\beta = 1$ e (13) é substituída por

$$\mathbf{v}_t = \mathbf{y} - \mathbf{A}\mathbf{b}_t + \frac{1}{\delta} \alpha_t \mathbf{v}_{t-1} \quad (18)$$

onde $\mathbf{v}_{-1} = \mathbf{0}$, $\delta = N_1/M$,

$$\alpha_{t+1} = \langle \eta'_t(\mathbf{r}_t) \rangle \quad (19)$$

$\eta'_t(r) = \partial \eta_t(r) / \partial r$ e $\langle \mathbf{x} \rangle = \sum_{j=1}^M x_j$, $\mathbf{x} \in \mathbb{R}^M$ [11].

O último termo em (18) é conhecido como termo de Onsager e pode ser interpretado como a aplicação de *momentum* [11]. A importância deste termo está em desacoplar as iterações, de tal forma que \mathbf{r}_t se comporta como uma versão de \mathbf{b} corrompida por ruído gaussiano, i.e., $\mathbf{r}_t \approx \mathbf{b} + \mathcal{N}(\mathbf{0}, \tau_t^2 \mathbf{I})$

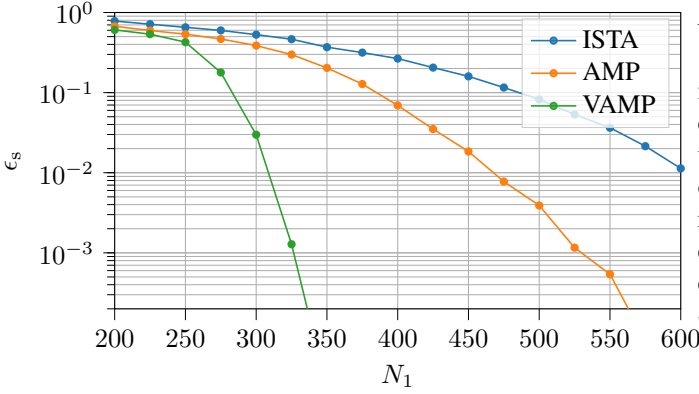


Fig. 1. Probabilidade de erro ϵ_s em função de N_1 para os métodos discutidos. ($M = 4096$, $P_1 = 0.2$, $K_a = 50$)

onde $\tau_t = \frac{1}{n} \|\mathbf{v}_t\|^2$. Em particular, se \mathbf{b} é um vetor i.i.d. com distribuição $P(\mathbf{b})$ conhecida, tal modelo permite determinar analiticamente a escolha ótima de $\eta_t(\cdot)$ como

$$\eta_t(\mathbf{r}_t) = E[\mathbf{b} \mid \mathbf{b} + \tau_t \mathbf{W} = \mathbf{r}_t] \quad (20)$$

onde $\mathbf{W} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Caso $P(\mathbf{b})$ seja desconhecida, (16) é considerada uma boa escolha [14].

Em geral, o desempenho do AMP é superior ao ISTA, além de convergir mais rapidamente. No entanto, o AMP é conhecido por se tornar instável e divergir com matrizes não-gaussianas ou levemente mal-condicionadas.

C. Vector Approximate Message Passing (VAMP)

Este algoritmo possui várias semelhanças com o AMP, em particular o seu baixo custo computacional por iteração, mas consegue convergir sob uma classe muito mais ampla de matrizes de sensoriamento. As principais diferenças estão no cálculo de \mathbf{r}_t e τ_t , realizados de forma recursiva, e na necessidade de efetuar uma SVD (*Singular Value Decomposition*) da matriz \mathbf{A} . A descrição completa do algoritmo é extensa e não será reproduzida aqui, podendo ser encontrada em [12].

D. Aplicação para Detecção de Atividade

No contexto da detecção de atividade, a distribuição de \mathbf{b} é conhecida, o que, em princípio, nos permitiria determinar o *denoiser* ótimo em (20). Uma dificuldade, no entanto, é o fato de que \mathbf{b} não é um vetor i.i.d. dado que $\|\mathbf{b}\|_1 = K_a$. Uma forma de contornar esse problema é aproximando b_j por uma variável aleatória de Bernoulli com $P[b_j = 1] = K_a/M$. Sob essa hipótese, temos o seguinte resultado.

Proposition 1: Se $b \in \{0, 1\}$ com $P[b = 1] = p$, então a função $n_t(\cdot) : \mathbb{R} \rightarrow \mathbb{R}$ definida em (16) é dada por

$$n_t(r) = \text{sigmoid} \left(\frac{r - 0.5}{\tau_t^2} - \log \left(\frac{1 - p}{p} \right) \right) \quad (21)$$

onde $\text{sigmoid}(x) = 1/(1 + e^{-x})$.

E. Comparação

Na Fig. 1 é apresentada uma comparação entre os algoritmos descritos. Utilizou-se $\beta = 0.55$, λ foi otimizado para cada ponto, o mesmo para o parâmetro α . Para o VAMP utilizou-se um *denoiser* ótimo². A figura nos indica que com o VAMP pode-se realizar a decodificação utilizando menos recursos do canal. Por isso, no sistema proposto utilizou-se esse algoritmo, que também possui convergência mais rápida e maior estabilidade ao custo de ser necessário armazenar a SVD da matriz \mathbf{A} na estação base.

IV. OTIMIZAÇÃO

Nesta seção, descrevemos o método de otimização utilizado. Lembre que, como mencionado na Seção II, desejamos fixar³ K_a , k , N e ϵ_{max} e minimizar E_b/N_0 . Em especial, perceba que N_1 , N_f e $n_{c,2}$, i.e., o número de usos de canal em cada fase, são parâmetros de projeto, bem como M , que determina quantos *bits* de informação transmitimos na primeira etapa. Finalmente, P_1 , P_2 e P_f podem ser projetados. Para simplicidade, similar a [6], assumimos $P \triangleq P_2 = P_f/K_a$, i.e., a potência por usuário se mantém constante nas duas fases finais.

Note que tal otimização é complexa por alguns motivos. Primeiramente, a alteração de um parâmetro de projeto, como M , influencia em diversas probabilidades de erro, como ϵ_s , ϵ_c , ϵ_f e ϵ_t , de maneiras opostas. Além disso, não há uma expressão simples que nos dê ϵ_s logo é necessário realizar uma simulação para avaliar essa taxa de erro. Finalmente, os parâmetros, exceto P e P_1 , são inteiros, logo, de forma geral, seria necessário testar todos os possíveis valores realizáveis a fim de encontrar o ótimo.

A fim de obter uma otimização realizável, optamos por uma solução heurística. Dados M e N_1 , simulamos ϵ_s , que depende de P_1 , N_1 , M e K_a , para suficientes valores de P_1 . Após realizar uma transformação logarítmica em P_1 e ϵ_s , esses dados são utilizados para treinar um modelo linear nos fornecendo a aproximação $\epsilon_{rs} \approx \epsilon_s$.

Para calcular as outras probabilidades de erro, utilizamos as expressões analíticas apresentadas no artigo [6] que foram revisadas nas seções anteriores e também relaxamos a restrição de que $n_{c,2}$ e N_f são inteiros (perceba que as expressões analíticas aceitam valores reais) e por fim utilizamos a substituição $N_f = N - N_1 - n_{c,2}K_a$.

Em seguida, obtemos o valor de $n_{c,2}$ que minimiza $\epsilon_{ft}(P) = 2\epsilon_f + \epsilon_t$. Assim o problema é reduzido a uma minimização de $E_b/N_0 = (P_1N_1 + PN_f + Pn_{c,2}K_a)/2k$ com restrição $\epsilon(P, P_1) \leq \epsilon_{max}$, onde $\epsilon(P, P_1) = \epsilon_{rs}(P_1) + \epsilon_c + \epsilon_{ft}(P)$.

Uma vez obtido um mínimo local com parâmetro relaxado $n_{c,2}^*$, testamos $n_{c,2} = \lceil n_{c,2}^* \rceil$ e $n_{c,2} = \lfloor n_{c,2}^* \rfloor$, novamente substituindo $N_f = N - N_1 - n_{c,2}K_a$ e escolhemos o arredondamento que proporciona uma menor potência P necessária para satisfazer a probabilidade de erro $\epsilon = \epsilon_{max}$.

²Utilizamos a implementação feita pela biblioteca Vampire: <https://github.com/GAMPTeam/vampire>

³Na prática, K_a é desconhecido e irá variar em torno de um valor esperado. Para lidar com isso, pode-se projetar o sistema para mais usuários ativos que o valor esperado.

TABELA I

PARÂMETROS OTIMIZADOS PARA $K_a = 50, 100, 200, 225, 250, 275, 300$

K_a	N_1	M	n_{c2}	n_f	P	P_1
50	400	4096	546	2300	0.006994	0.103795
100	800	8192	270	2200	0.007878	0.049412
200	1500	16384	132	2100	0.010349	0.026658
225	1600	16384	117	2075	0.011117	0.025425
250	1900	16384	104	2100	0.012206	0.020964
275	1900	32768	94	2250	0.012786	0.025258
300	2000	32768	86	2200	0.013853	0.026041

Finalmente, este processo é repetido para alguns valores de M e N_1 , escolhidos manualmente pelo projetista, e seleciona-se o melhor par de parâmetros como resultado final. Para confirmar que o sistema satisfaz as restrições, são simuladas várias realizações do sistema com os parâmetros finais.

V. RESULTADOS

Nesta seção, apresentamos os resultados da nossa otimização e comparamos a outros trabalhos da área. Similar à literatura, utilizamos $\epsilon_{max} = 0.05$, $N = 30000$ e $k = 100$. Os parâmetros obtidos pela otimização são apresentados na Tabela I.

Em sequência, comparamos nosso resultado em função de K_a com outros trabalhos da literatura. Na Fig. 2, apresentamos o resultado de alcançabilidade com códigos aleatórios (*Random coding*) de [1], método ortogonal (OMA) com coordenação *a priori* (ambos apenas para referência) [1], o método *grant-based* proposto em [6], e o método que atualmente é o melhor desempenho da literatura, utilizando códigos polares [7]. Apesar de o método proposto ser inferior ao *grantless* para um número de usuários pequeno, quando este aumenta ($K_a \geq 200$), o método proposto supera o estado-da-arte consideravelmente.

VI. CONCLUSÕES

O artigo apresentou um método de detecção de atividade para o modelo de canal e sistema propostos em [6], além de propor um *framework* de otimização para os parâmetros do sistema. Através de simulação, os resultados deste sistema são comparados ao desempenho de outros métodos da literatura, e observa-se, para um número de usuários ativos suficientemente grande, um desempenho consideravelmente superior utilizando-se a nossa proposta.

Acreditamos que tais resultados são promissores e motivam futuras pesquisas em métodos *grant-based* para o problema de acesso massivo, em especial em canais complexos com desvanecimento e assincronia.

REFERÊNCIAS

[1] Y. Polyanskiy, "A perspective on massive random-access", in 2017 IEEE International Symposium on Information Theory (ISIT), Aachen, Germany, jun. 2017, p. 2523–2527, doi: 10.1109/ISIT.2017.8006984.
 [2] O. Ordentlich e Y. Polyanskiy, "Low complexity schemes for the random access Gaussian channel", in 2017 IEEE International Symposium on Information Theory (ISIT), jun. 2017, p. 2528–2532, doi: 10.1109/ISIT.2017.8006985.

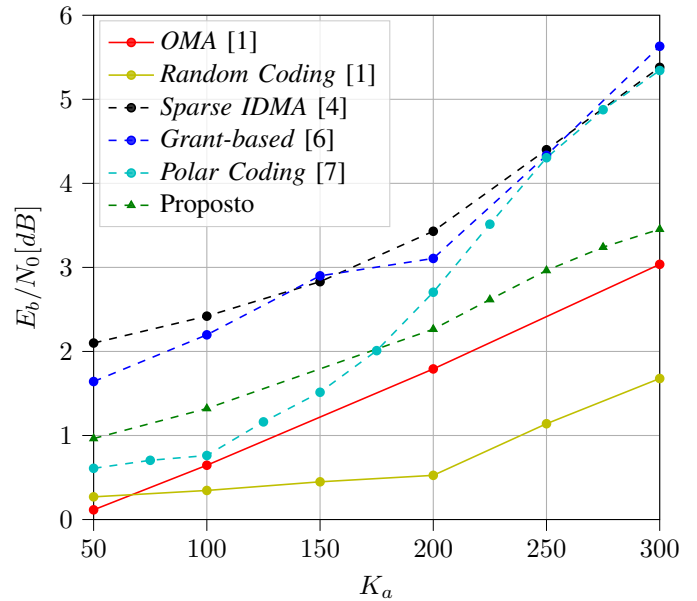


Fig. 2. Relação entre número de usuários e E_b/N_0 para $\epsilon_{max} = 0.05$. Parâmetros utilizados estão na Tabela I. $k = 100$ bits, $N = 30000$ usos de canal

[3] V. K. Amalladinne, J.-F. Chamberland, e K. R. Narayanan, "A Coded Compressed Sensing Scheme for Uncoordinated Multiple Access", arXiv:1809.04745 [cs, eess, math], set. 2018, Acessado: ago. 14, 2019. [Online]. Disponível em: <http://arxiv.org/abs/1809.04745>.
 [4] A. Pradhan, V. Amalladinne, A. Vem, K. R. Narayanan, e J.-F. Chamberland, "A Joint Graph Based Coding Scheme for the Un-sourced Random Access Gaussian Channel", arXiv:1906.05410 [cs, math], jun. 2019, Acessado: set. 04, 2019. [Online]. Disponível em: <http://arxiv.org/abs/1906.05410>.
 [5] A. Vem, K. R. Narayanan, J.-F. Chamberland, e J. Cheng, "A User-Independent Successive Interference Cancellation Based Coding Scheme for the Un-sourced Random Access Gaussian Channel", IEEE Transactions on Communications, vol. 67, nº 12, p. 8258–8272, dez. 2019, doi: 10.1109/TCOMM.2019.2940216.
 [6] G. K. Facenda e D. Silva, "Efficient Scheduling for the Massive Random Access Gaussian Channel", arXiv:1905.06479 [cs, math], dez. 2019, Acessado: maio 15, 2020. [Online]. Disponível em: <http://arxiv.org/abs/1905.06479>.
 [7] A. K. Pradhan, V. K. Amalladinne, K. R. Narayanan, e J.-F. Chamberland, "Polar Coding and Random Spreading for Un-sourced Multiple Access", arXiv:1911.01009 [cs, math], nov. 2019, Acessado: jul. 02, 2020. [Online]. Disponível em: <http://arxiv.org/abs/1911.01009>.
 [8] E. J. Candes, J. Romberg, e T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information", IEEE Trans. Inform. Theory, vol. 52, nº 2, p. 489–509, fev. 2006, doi: 10.1109/TIT.2005.862083.
 [9] E. J. Candes e T. Tao, "Near-Optimal Signal Recovery From Random Projections: Universal Encoding Strategies?", IEEE Trans. Inform. Theory, vol. 52, nº 12, p. 5406–5425, dez. 2006, doi: 10.1109/TIT.2006.885507.
 [10] D. L. Donoho, "Compressed sensing", IEEE Trans. Inform. Theory, vol. 52, nº 4, p. 1289–1306, abr. 2006, doi: 10.1109/TIT.2006.871582.
 [11] D. L. Donoho, A. Maleki, e A. Montanari, "Message Passing Algorithms for Compressed Sensing", PNAS, vol. 106, nº 45, p. 18914–18919, nov. 2009, doi: 10.1073/pnas.0909892106.
 [12] S. Rangan, P. Schniter, e A. K. Fletcher, "Vector Approximate Message Passing", arXiv:1610.03082 [cs, math], jun. 2018, Acessado: out. 23, 2019. [Online]. Disponível em: <http://arxiv.org/abs/1610.03082>.
 [13] Y. Polyanskiy, H. V. Poor, e S. Verdú, "Channel Coding Rate in the Finite Blocklength Regime", IEEE Transactions on Information Theory, vol. 56, nº 5, p. 2307–2359, maio 2010, doi: 10.1109/TIT.2010.2043769.
 [14] A. Montanari, "Graphical Models Concepts in Compressed Sensing", arXiv:1011.4328 [cs, math], mar. 2011, Acessado: jun. 17, 2020. [Online]. Disponível em: <http://arxiv.org/abs/1011.4328>.