

# **Simulação de Sinais Cerebrais de Espectroscopia por Ressonância Magnética**

**Da Criação à Corrupção (Por Ruído)**

**João Victor Dell Agli Floriano**

**Fernando Fernandes Paiva**

**Curso:** Mestrado

**Período a que se refere:** 02/2025 a 05/2025

**Bolsa de Estudos:** CAPES

**Período de Vigência:** 01/08/2024 a 28/02/2026 (19 meses)

# 1 Resumo

1. Descrever o que é "pencil" **CONCLUIDO**
2. Implementação do MPM sem ruído **CONCLUIDO**
3. Implementação do MPM com ruído **CONCLUIDO**
4. Separação de variáveis (s0, phi, omega, T2) **CONCLUIDO**
5. Testagem do L sem ruído **CONCLUIDO**
6. Testagem do SVD sem ruído **CONCLUIDO**
7. Testagem do L com ruído
8. Testagem do SVD com ruído
9. Testagem do comportamento das variáveis separadas com a introdução de ruído de valores de sigma variados

# 2 Introdução

O termo "lápiz" (do inglês *pencil*), cunhado por Grantmacher [?] no contexto de *pencil* de matrizes, é uma ferramenta que combina matrizes quadradas linearmente a partir de um parâmetro  $\lambda$ , como descrito pela Equação 1.  $l$  é um inteiro não-negativo.

$$L(\lambda) = \sum_{i=0}^l \lambda^i A_i \quad (1)$$

No contexto de *pencil* de funções,  $f(t, \lambda)$  é *pencil* de  $g(t)$  e  $h(t)$  quando segue o formato descrito pela Equação 2.

$$f(t, \lambda) = g(t) + \lambda h(t) \quad (2)$$

O método "lápiz de matrizes", do inglês *Matrix Pencil Method* (MPM) é uma técnica numérica de estimativa de parâmetros de sinais, desenvolvido originalmente por Yingbo Hua e Tapan Sakar [2] como uma alternativa a métodos já existentes como o de Prony [1]. O mesmo consiste em modelar os sinais como uma soma de exponenciais complexas amortecidas, como na Equação 3. Partindo dessa ideia, é então aplicada uma série de

etapas, que inclui a utilização de outros métodos, como Decomposição em Valores Singulares (SVD, do inglês *Singular Value Decomposition*), para estimar os parâmetros dessa função modeladora.

$$y(n) = \sum_{k=1}^M R_k e^{i(\omega_k t + \phi_k) + \alpha_k} \quad (3)$$

devo incluir uma parte sobre SVD?

## 3 Métodos

### 3.1 Implementação do MPM

A implementação do MPM em seu trabalho original é descrita originalmente de duas maneiras: a sem ruído, implementada de maneira mais simplificada; e a que leva em conta a presença de ruído, que utiliza algoritmos mais complexos, como a Decomposição em Valores Singulares (SVD, do inglês *Singular Value Decomposition*).

#### 3.1.1 Caso sem ruído

Assumindo um sinal  $x$  de tamanho  $N$ , para o caso sem ruído, define-se duas matrizes  $(N - L) \times L$ ,  $Y_1$  e  $Y_2$ , descritas pela Equação 4 e Equação 5.

$$Y_2 = \begin{bmatrix} x(1) & x(2) & \dots & x(L) \\ x(2) & x(3) & \dots & x(L+1) \\ \vdots & \vdots & & \vdots \\ x(N-L) & x(N-L+1) & \dots & x(N-1) \end{bmatrix} \quad (4)$$

$$Y_1 = \begin{bmatrix} x(0) & x(1) & \dots & x(L-1) \\ x(1) & x(2) & \dots & x(L) \\ \vdots & \vdots & & \vdots \\ x(N-L-1) & x(N-L) & \dots & x(N-2) \end{bmatrix} \quad (5)$$

Sendo  $L$  o parâmetro de *pencil*, importante para etapas posteriores. É possível escrever  $Y_1$  e  $Y_2$  como a Equação 6 e a Equação 7, sendo  $Z_1$  descrito pela Equação 8,  $Z_2$  descrito pela Equação 9,  $Z_0$  descrito pela Equação 10, e enfim  $R$ , descrito pela Equação 11. Por enquanto, o parâmetro  $M$  pode ser considerado como igual a  $N$ , porém virá a assumir

valores diferentes posteriormente.

$$Y_2 = Z_1 R Z_0 Z_2 \quad (6)$$

$$Y_1 = Z_1 R Z_2 \quad (7)$$

$$Z_1 = \begin{bmatrix} 1 & 1 & \dots & 1 \\ z_1 & z_2 & \dots & z_M \\ \vdots & \vdots & & \vdots \\ z_1^{N-L-1} & z_2^{N-L-1} & \dots & z_M^{N-L-1} \end{bmatrix} \quad (8)$$

$$Z_2 = \begin{bmatrix} 1 & z_1 & \dots & z_1^{L-1} \\ 1 & z_2 & \dots & z_2^{L-1} \\ \vdots & \vdots & & \vdots \\ 1 & z_M & \dots & z_M^{L-1} \end{bmatrix} \quad (9)$$

$$Z_0 = \text{diag}(z_1, z_2, \dots, z_M) \quad (10)$$

$$R = \text{diag}(R_1, R_2, \dots, R_M) \quad (11)$$

Considerando agora o *pencil*, descrito pela Equação 12: Reescrevendo-o a partir da Equação 6 e da Equação 7, obtém-se a Equação 13, que, reorganizada, rende a Equação 14.

$$Y_2 - \lambda Y_1 \quad (12)$$

$$Y_2 - \lambda Y_1 = Z_1 R Z_0 Z_2 - \lambda Z_1 R Z_2 \quad (13)$$

$$Y_2 - \lambda Y_1 = Z_1 R (Z_0 - \lambda I) Z_2 \quad (14)$$

Escolhendo  $\lambda = z_i$  de maneira intencional, a matriz  $Z_0 - \lambda I$  é zero, fazendo com que o *pencil* seja igual a zero, transformando o problema em um problema de autovalores generalizados. Encontrando os autovalores generalizados do par  $\{Y_1, Y_2\}$ , encontra-se os polos  $z_i$ . É possível também encontrar  $z_i$  como resultado do problema de autovalores comuns da matriz  $Y_1^+ Y_2$ , como na Equação 15.  $Y_1^+$ , a matriz pseudoinversa de Moore-Penrose de  $Y_1$ , é definida pela Equação 16, no qual  $H$  denota o conjugado transposto.

$$Y_1^+ Y_2 - \lambda I = 0 \quad (15)$$

$$Y_1^+ = (Y_1^H Y_1)^{-1} Y_1^H \quad (16)$$

Encontrados os polos  $z_i$ , basta encontrar os resíduos  $R_i$  a partir de um problema de mínimos quadrados, descritos pela Equação 17.

$$\begin{bmatrix} y(0) \\ y(1) \\ \vdots \\ y(N-1) \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ z_1 & z_2 & \dots & z_M \\ \vdots & \vdots & \dots & \vdots \\ z_1^{N-1} & z_2^{N-1} & \dots & z_M^{N-1} \end{bmatrix} \cdot \begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_M \end{bmatrix} \quad (17)$$

### 3.1.2 Caso com ruído

No caso com ruído, é construída a matriz generalizada  $Y$ , descrita pela Equação 18, da qual é possível extrair  $Y_1$  e  $Y_2$  excluindo a última e a primeira coluna, respectivamente.

$$Y = \begin{bmatrix} y(0) & y(1) & y(2) & \dots & y(L) \\ y(1) & y(2) & y(3) & \dots & y(L+1) \\ \vdots & \vdots & \vdots & & \vdots \\ y(N-L-1) & y(N-L) & y(N-L+1) & \dots & y(N-1) \end{bmatrix} \quad (18)$$

Como citado anteriormente, o parâmetro  $L$  viria a ser importante para filtragem do ruído, sendo escolhido entre  $N/3$  e  $N/2$ , intervalo no qual a variância dos parâmetros  $z_i$ , por conta do ruído, apresentou-se como mínima [2].

Após a construção, aplica-se uma decomposição em Valores Singulares (SVD) nessa matriz.

$$Y = U \Sigma V^H \quad (19)$$

A qual a matriz  $U$ , a matriz de vetores singulares à esquerda, contém os autovetores da matriz  $Y Y^H$ .  $V$ , chamada de matriz de vetores singulares à direita, contém os autovetores da matriz  $Y^H Y$ , e  $\Sigma$ , a matriz diagonal de valores singulares, contém os valores singulares de  $Y Y^H$  e  $Y^H Y$ .

Encontradas as submatrizes, o número  $M$  de polos a serem calculados é escolhido, sendo eles os  $M$  valores singulares de  $\Sigma$  que mais contribuem para a matriz final. A sugestão

de exclusão segue a Equação 20, sendo  $\sigma_c$  o valor singular analisado,  $\sigma_{max}$  o maior valor singular e  $p$  a ordem do número de algarismos significativos. A partir disso, é possível filtrar os autovalores, separando os mais significativos de ruído.

$$\frac{\sigma_c}{\sigma_{max}} > 10^{-p} \quad (20)$$

Encontrado o valor singular limite de filtragem, a matriz  $Y'$  é construída a partir de  $Y$  com menos valores singulares, e a partir dela,  $Y'_1$  e  $Y'_2$  são derivadas. Realizada essa etapa, basta calcular os autovalores generalizados do par  $\{Y'_2, Y'_1\}$ , ou o autovalor da matriz  $Y'^+_1 Y'_2$ , e os resíduos  $R_i$ , seguindo o procedimento descrito anteriormente, concluindo o processo do MPM. O sinal pode ser reconstruído com os polos  $z_i$  e resíduos  $R_i$  pela Equação 21.

$$y(kT_s) \approx \sum_{i=1}^M R_i z_i^k, \quad k = 0, \dots, N - 1 \quad (21)$$

O algoritmo do MPM pode ser resumido, então, pelas seguintes etapas:

1. Calcular a matriz  $Y$ .
2. Calcular a SVD de  $Y$ .
3. Filtrar os valores singulares de acordo com o critério de seleção da Equação 20.
4. Reconstruir a matriz filtrada  $Y'$ .
5. Obter  $Y'_1$  e  $Y'_2$  a partir de  $Y'$ .
6. Calcular os autovalores generalizados do par  $\{Y'_2, Y'_1\}$  ou calcular os autovalores da matriz  $Y'^+_1 Y'_2$ .
7. Encontrar os resíduos por meio do problema de mínimos quadrados descrita pela Equação 17.

Essas etapas foram traduzidas em um algoritmo na linguagem *python*, implementado em uma biblioteca própria customizada. Para sua implementação, foram usadas funções já prontas da biblioteca *numpy*, como a função para o cálculo da pseudoinversa de Moore-Penrose, cálculo de autovalores e de resolução de mínimos quadrados. É importante ressaltar que, por se tratar de um algoritmo com características numéricas em múltiplas etapas, algumas adaptações e aproximações foram feitas de maneira a viabilizar seu funcionamento. Foi necessário aproximar valores de saída dos cálculos dos polos  $z_i$  e resíduos  $R_i$  para zero

em casos os quais seus valores calculados eram significativamente baixos. Essa aproximação foi feita a partir de um valor limite, definido a critérios do usuário, para o qual valores abaixo ou igual ao valor limite eram truncados. Além disso, para as funções externas, foi necessário se atentar ao seus parâmetros próprios de filtro, usados em etapas intermediárias de cálculos que utilizam o SVD. Nesse último caso, foi definido como filtro de corte o valor de  $10^{-7}$ .

### 3.2 Separação de variáveis

Com o intuito de melhorar o controle e entendimento dos parâmetros calculados pelo MPM, foi feita uma correspondência entre os polos  $z_i$  e resíduos  $R_i$  com os parâmetros originais que compõe o sinal, nesse caso,  $S_{0,i}$ ,  $\phi_i$ ,  $\omega_i$  e  $T_{2,i}$ . Considerando que um sinal de Espectroscopia por Ressonância Magnética (do inglês *Magnetic Resonance Spectroscopy*, MRS) pode ser representado pela Equação 22, sua versão discreta pode ser escrita assumindo  $t = kT_s$ , sendo  $T_s$  o período de *sampling* do sinal, e  $k$  um inteiro representando o passo, como na Equação 23.

$$S = \sum_{i=1}^M S_{0,i} e^{j\omega_i t - \frac{t}{T_2}} e^{j\phi_i} \quad (22)$$

$$S = \sum_{i=1}^M S_{0,i} e^{j\phi_i} (e^{j\omega_i T_s - \frac{T_s}{T_2}})^k \quad (23)$$

Considerando que a aproximação feita por meio da Equação 21 é uma representação suficientemente correta do sinal, é possível estabelecer uma equivalência entre os parâmetros originais e calculados, a partir da Equação 24.

$$\sum_{i=1}^M S_{0,i} e^{j\phi_i} (e^{j\omega_i T_s - \frac{T_s}{T_2}})^k = \sum_{i=1}^M R_i z_i^k \quad (24)$$

Essa equivalência demonstra que enquanto os polos  $z_i$  representam  $e^{j\omega_i T_s} e^{-\frac{T_s}{T_2}}$ , os resíduos representam a multiplicação  $S_{0,i} e^{j\phi_i}$ . Considerando a representação polar de um número complexo, na Equação 25, é possível visualizar uma maneira simples de cálculo dos parâmetros originais a partir de  $z_i$  e  $R_i$ , concluindo que  $e^{-\frac{T_s}{T_2}}$  corresponde ao módulo de  $z$ ,  $\omega_i T_s$  ao argumento de  $z$ ,  $S_{0,i}$  ao módulo de  $R$ , e  $\phi_i$  ao argumento de  $R$ . A Tabela 1 reúne as equivalências resultantes.

$$z = Ae^{j\theta} \quad (25)$$

$S_0$	$\phi$	$\omega$	$T_2$
$ R $	$arg(R)$	$-\frac{1}{T_s} arg(z)$	$-\frac{T_s}{\log( z )}$

Tabela 1: Relação entre os parâmetros calculados e originais.

Com o cálculo dos parâmetros individuais bem estabelecidos, foi implementada uma função na biblioteca própria para realização dessa tarefa. Foi feita uma adaptação dos cálculos dos argumentos de  $R$  e  $z$ , visto que a função *arctan2* da biblioteca *numpy* retornava valores no intervalo  $[-\pi, \pi)$ , sendo que o desejado era o intervalo  $[0, 2\pi)$ . Para isso, foi somado  $2\pi$  para valores negativos da função, de maneira a adequá-la às necessidades apresentadas.

### 3.3 Sinais de Controle

#### 3.3.1 Sinal de controle completo

Com o objetivo de caracterizar o comportamento do algoritmo com relação às suas variáveis  $L$  e  $p$ , foi gerado um sinal de controle de comportamento previsível e variáveis geradoras conhecidas. O sinal se trata de uma simulação de MRS elaborada a partir de um algoritmo próprio, gerada a partir de parâmetros já caracterizados de metabólitos conhecidos, e fase  $\phi = 0$ . O mesmo foi gerado com um campo de  $B_0 = 3T$ , começando em  $t_0 = 0.0$  s e terminando em  $t_n = 1.0$  s, com um passo de  $dt = \frac{1}{2048} \approx 0.0005$  s, resultando em um sinal com 2048 pontos, de maneira a se adequar ao padrão de medidas clínicas. O sinal pode ser conferido pela Figura 1.



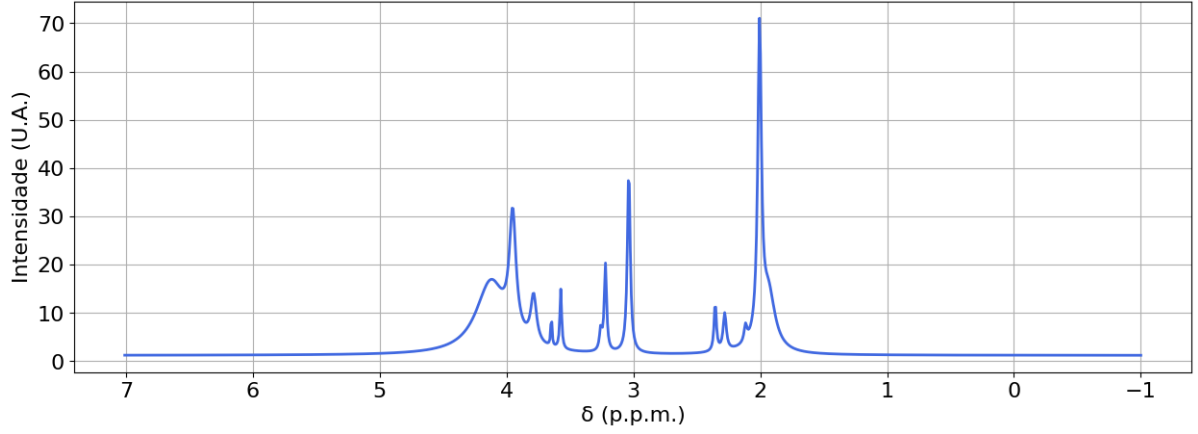


Figura 1: Sinal de controle gerado para caracterização das variáveis.

### 3.3.2 Sinal de controle de pico único

Além de averiguar a influência dos parâmetros tradicionais no processo do algoritmo, foi também verificada a necessidade de avaliar o comportamento dos parâmetros resultantes do processo,  $S_0$ ,  $\phi$ ,  $\omega$  e  $T_2$ , com relação à qualidade do sinal, traduzida pela sua relação sinal-ruído (do inglês, *Signal-to-Noise Ratio*, SNR). Para prosseguir, foi escolhido um novo sinal de controle: a simulação de MRS de um único metabólito, o N-Acetylaspartato, ou NAA, escolhido por ser o maior pico dentre os metabólitos comumente encontrados no cérebro. Essa escolha foi motivada pela simplificação apresentada, visto que um único pico facilitaria o processo de investigação de possíveis fenômenos associados à distorções causada pela presença de ruído. A simulação foi feita com as mesmas condições de contorno apresentadas na Subsubseção 3.3.1, podendo ser conferida pela Figura 2.

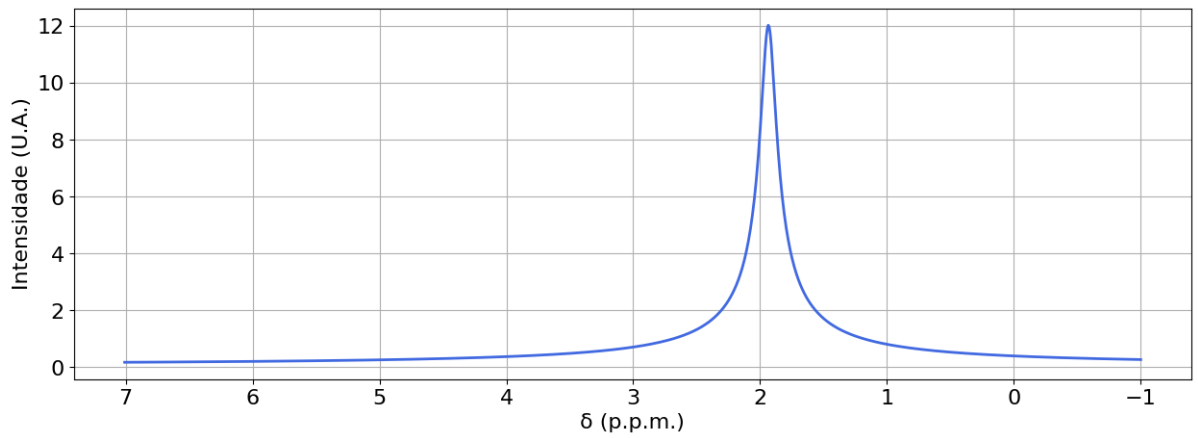


Figura 2: Sinal de controle de pico único gerado para estudo dos parâmetros resultantes.

## 4 Resultados

### 4.1 Implementação do MPM

### 4.2 Separação de variáveis

### 4.3 Testagem de $L$ sem ruído.

Foi testado para o algoritmo implementado como as variáveis resultantes do processo se comportavam com relação à variação de  $L$ , no sinal anteriormente apresentado, livre de ruído. Para isso, o algoritmo foi rodado com  $L$  variando de  $L_0 = 20\%$  do  $N$  a  $L_N = 80\%$  do  $N$ , limites além de  $N/3$  e  $N/2$  previamente estabelecidos, com o objetivo de verificar se havia alguma flutuação significativa nos valores das variáveis resultantes. Para cada  $L$ , variado nesse intervalo com um passo de  $5\%$  do  $N$  foram feitas 10 médias. O parâmetro  $p$  foi fixado em 15, e a comparação foi feita por meio da Raiz do Erro Quadrático Médio (do inglês *Root-mean-squared Error*, RMSE), podendo ser conferida pela Figura 3.

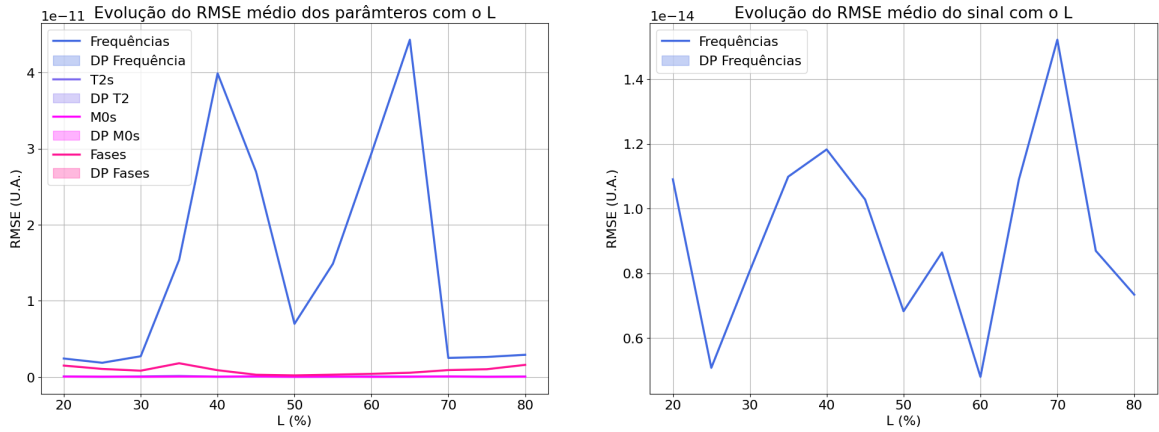


Figura 3: Resultados para a variação de  $L$

Pela Figura 3, é possível concluir que a escolha do  $L$  parece não impactar a capacidade de cálculo das variáveis originais do sinal limpo de ruído, visto que as flutuações de valor do RMSE médio são praticamente zero.

### 4.4 Testagem de SVD sem ruído.

Também foi testado, para o mesmo sinal, como as variáveis resultantes se comportavam com relação ao outro parâmetro do MPM, nesse caso, a ordem da variável de corte do SVD.  $p$

foi variado entre  $p_0 = 3$  e  $p_N = 15$ , com um passo de uma unidade, e  $L = 40\%$ . Para cada  $p$ , foram feitas também 10 médias.

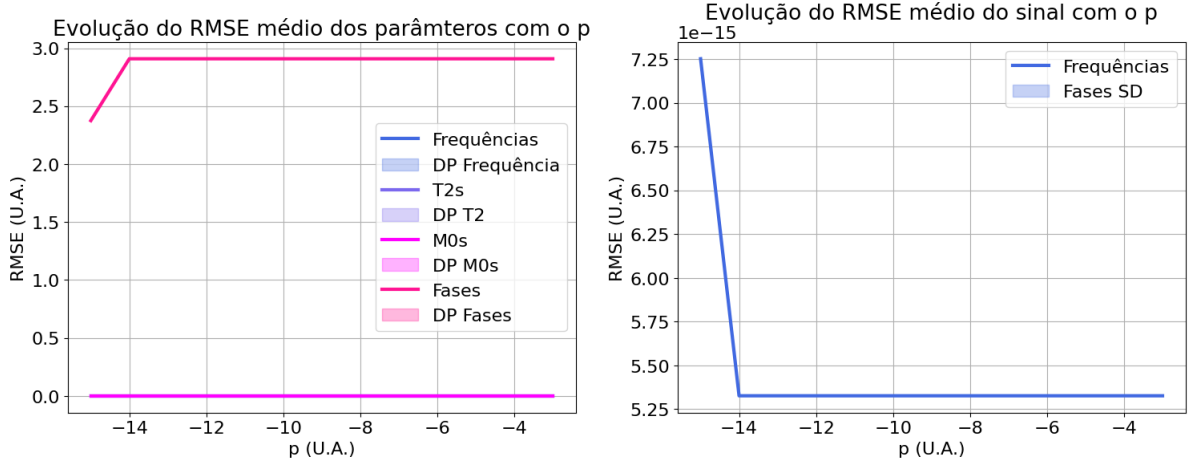


Figura 4: Resultados para a variação de  $p$

A priori, o evidente erro com relação às fases pode chamar a atenção, visto que a fase de entrada do sinal de controle era  $\phi = 0$ . Constatou-se, porém, que sua origem reside em pequenos erros numéricos que se propagam na função  $\arctan2$ , provocando um resultado de saída de aproximadamente  $2\pi$  para fases calculadas, equivalente à fase esperada. Esclarecido esse importante detalhe, é possível concluir que, novamente para o sinal limpo, a  $p$  não demonstra nenhuma influência significativa na capacidade de cálculo das variáveis originais.

## 4.5 Relação entre as variáveis e o ruído

De maneira a caracterizar o comportamento da variáveis com relação ao ruído, a simulação de pico único da Subsubseção 3.3.2 foi corrompida com ruídos de diferentes níveis

## 5 Conclusão

## Referências

- [1] J.F. Hauer, C.J. Demeure, and L.L. Scharf. Initial results in prony analysis of power system response signals. *IEEE Transactions on Power Systems*, 5(1):80–89, 1990.

- [2] T.K. Sarkar and O. Pereira. Using the matrix pencil method to estimate the parameters of a sum of complex exponentials. *IEEE Antennas and Propagation Magazine*, 37(1):48–55, 1995.