

Experiência 5: Cancelamento de Eco e RLS

Nesta experiência vamos comparar o funcionamento do algoritmo NLMS com o do algoritmo RLS, e usá-los para resolver um problema de cancelamento de eco com dados reais. Primeiramente, vamos comparar o RLS com o NLMS com sinais artificiais, depois verificaremos o funcionamento com sinais de voz.

1. Crie uma simulação com os parâmetros usando tanto o NLMS quanto o RLS
 - (a) $x[n] = N(0, \sigma_x^2)$ iid;
 - (b) $v[n] = N(0, \sigma_v^2)$, iid e independente de $x[n]$;
 - (c) $\mathbf{h} = \text{randn}(100, 1)$; $\mathbf{h} = \mathbf{h} / \text{norm}(\mathbf{h})$; Use cem coeficientes para o vetor W .
2. Escolha valores de μ e λ para que os filtros funcionem adequadamente e tenham o mesmo EMSE em regime, e desenhe as curvas de aprendizado (EMSE) do filtro.
3. Repita o exercício anterior, mas agora defina o sinal de entrada $x[n]$ como sendo a saída de um filtro $G(z) = \sqrt{1 - a^2} / (1 - az^{-1})$, com $a = 0,9$, e cuja entrada é um ruído branco $u[n]$ iid com distribuição $N(0,1)$.
4. Use agora como entradas os sinais dos arquivos `conversa1.wav` (sinal $v[n]$) e `conversa2.wav` (sinal $x[n]$). Esses são dois sinais de voz que se alternam, de maneira a simular uma conversa. Para $h[n]$ use as variáveis do arquivo `respimp.mat` (Matlab, Python) ou `respimp.jld` (Julia). Filtre $x[n]$ pelo filtro $h[n]$, gerando o eco $y[n]$ e some-o a $v[n]$ para gerar o sinal $d[n]$.

Compare o funcionamento dos dois algoritmos. Cuidado que desta vez vai ser um pouco mais difícil ajustar os parâmetros dos algoritmos: em particular, o valor de λ pode precisar ser bem próximo de 1 (tente valores da ordem de 0,999 ou 0,9999), e o valor da regularização ϵ do NLMS não pode ser muito pequeno.

É interessante testar o efeito de modificar a resposta ao impulso no meio da simulação. Para experimentar isso da maneira mais simples possível, gere o sinal

de eco $y[n]$ como descrito acima, depois concatene-o com ele mesmo com sinal trocado, simulando uma mudança instantânea da resposta ao impulso de $h[n]$ para $-h[n]$ no meio da simulação.

Como saída, compare o sinal de erro $e[n]$ com o sinal $v[n]$, e a curva do MSD instantâneo a cada iteração.

Uma outra medida muito usada para cancelamento de eco é chamada “Echo Return Loss Enhancement”, que é dado por

$$\text{ERLE}[n] = 10 \log_{10} \left\{ \frac{\text{E}\{y^2[n]\}}{\text{E}\{e^2[n]\}} \right\}.$$

Para melhorar os resultados é útil usar-se um Detector de Dupla Conversa (*Double-Talk Detector*, DTD), ou algum outro mecanismo para diminuir a velocidade de adaptação quando o sinal $v[n]$ for grande. Dois mecanismos para se fazer isso são descritos a seguir

1. DTD de Geigel [1]: Este é um DTD simples e de baixo custo, que parte da observação de que $d[n]$ tende a ser grande comparado às últimas amostras de $x[n]$ quando há dupla conversa. O critério é então adaptar os coeficientes do filtro apenas quando

$$d_g[n] = \frac{|d[n]|}{\max\{|x[n]|, |x[n-1]|, \dots, |x[n-M+1]|\}} \leq D_T,$$

em que D_T é um limiar convenientemente ajustado.

2. Uma outra opção também de baixo custo é o algoritmo de passo variável PVSS [2], que consiste em aplicar um fator variável ao passo do NLMS a cada iteração, como a seguir:

$$\begin{aligned} \hat{\sigma}_d^2[n+1] &= \lambda_d \hat{\sigma}_d^2[n] + (1 - \lambda_d) d^2[n], \\ \hat{\sigma}_y^2[n+1] &= \lambda_y \hat{\sigma}_y^2[n] + (1 - \lambda_y) y^2[n], \\ \hat{\sigma}_e^2[n+1] &= \lambda_e \hat{\sigma}_e^2[n] + (1 - \lambda_e) e^2[n], \\ \alpha[n] &= \left| 1 - \frac{\sqrt{|\hat{\sigma}_d^2[n+1] - \hat{\sigma}_y^2[n+1]|}}{\zeta + \sqrt{\hat{\sigma}_e^2[n+1]}} \right|, \end{aligned}$$

em que λ_d pode ser escolhido em torno de 0,99, e a variável de regularização $\zeta \approx 0,002$. O passo do NLMS é escolhido como

$$\mu[n] = \alpha[n] \frac{\mu_0}{\epsilon + \|\Phi[n]\|^2}.$$

A mesma ideia pode ser aplicada ao RLS, escolhendo-se o fator de esquecimento do RLS como

$$\lambda[n] = 1 - \alpha[n](1 - \lambda_0).$$

Os dois algoritmos podem ser usados simultaneamente.

A literatura sobre cancelamento de ruído é vasta, estes são apenas alguns exemplos de algoritmos.

Referências

- [1] T. Gänslér e J. Benesty. The fast normalized cross-correlation double-talk detector. *Signal Processing*, 86:1124–1139, 2006.
- [2] C. Paleologu, J. Benesty, S. L. Grant, e C. Osterwise. Variable step-size NLMS algorithms designed for echo cancellation. In *2009 Conference Record of the Forty-Third Asilomar Conference on Signals, Systems and Computers*, pp. 633–637. IEEE, nov. 2009.