

Introdução ao Processamento de Sinais Digitais

Notas de aula sobre filtros recursivos

Estas notas de aula correspondem a uma parte da matéria do curso que não está no livro “Discrete Fourier Analysis and Wavelets: Applications to Signal and Image Processing” de S. Allen Broughton e Kurt M. Bryan. Mais informações sobre filtros recursivos 1D podem ser obtidas nos livros “Discrete-Time Signal Processing” de Alan. V. Oppenheim e Ronald W. Schaffer e “Digital Filters” de R. W. Hamming, e sobre filtros recursivos 2D nos livros “Multidimensional Signal, Image and Video Processing and Coding” de John W. Woods e “Two-Dimensional Signal and Image Processing” de Jae S. Lim.

1 Introdução

No capítulo 4 do livro vimos alguns exemplos de filtros da forma

$$y = x * h$$

onde $h : I \rightarrow \mathbb{C}$ era definida em um conjunto finito $I \subset \mathbb{Z}$, sendo que a expressão acima era equivalente a

$$y(n) = \sum_{i \in I} h(i)x(n-i).$$

Tais filtros são conhecidos como *FIR* (Finite Impulse Response).

Se $i \geq 0, \forall i \in I$ o filtro é chamado de *causal*, pois a saída (presente) $y(n)$ só depende de entradas $x(n-i)$ (do passado). Esta propriedade é relevante por exemplo no processamento em tempo-real de um *stream*, pois a causalidade garante que a saída do instante n pode ser calculada imediatamente quando a entrada $x(n)$ estiver disponível, sem a necessidade de esperar por outros dados. A latência do filtro neste caso será apenas aquela relativa ao cálculo da expressão de convolução, chamada de *latência algorítmica*; dependendo do comprimento da resposta impulsiva h a latência algorítmica pode não ser desprezível.

Filtros FIR não-causais também podem ser processados em tempo-real, desde que se aguarde até a disponibilidade dos dados necessários para o cálculo da equação do filtro, o que introduz uma latência relativa à *transmissão* dos dados. Por exemplo, se uma equação do filtro para $y(n)$ depende do valor futuro $x(n+10)$, bastaria definir o filtro com um atraso de $\Delta = 10$, através da fórmula

$$\hat{y}(n) = y(n - \Delta) = \sum_{i \in I} h(i)x(n - \Delta - i)$$

Observe que se $i + \Delta \geq 0, \forall i \in I$, então o filtro \hat{y} é causal.

Em processamento de imagens o mais usual é termos acesso a todos os dados de entrada simultaneamente, e neste caso a causalidade não é uma propriedade relevante. No entanto, na transmissão e processamento de imagens muito grandes esta propriedade pode se tornar importante, e um filtro será causal se utilizar na expressão de $y(n, m)$ apenas pixels anteriores à entrada $x(n, m)$ no stream; no caso de uma varredura por linhas isso representaria os pixels $x(n-i, m-j)$ onde $i < 0$ ou $i = 0 \wedge j \leq 0$.

Se na equação de convolução $y = x * h$ a resposta impulsiva h possui duração infinita, então o filtro é dito *IIR* (Infinite Impulse Response). Neste caso, a teoria desenvolvida para sinais de tempo infinito e discreto mas com energia finita, correspondentes ao espaço $L^2(\mathbb{Z})$, será bastante útil para caracterizar a operação destes filtros e nos permitir entender as modificações feitas no sinal de entrada a partir de sua representação como soma de sinais senoidais.

2 Filtros recursivos 1D

Conforme veremos nesta seção, é possível construir filtros com resposta impulsiva infinita e que são computáveis em tempo finito. Vejamos inicialmente um exemplo.

Exemplo: Considere a equação

$$y(n) = x(n) + Ky(n-1),$$

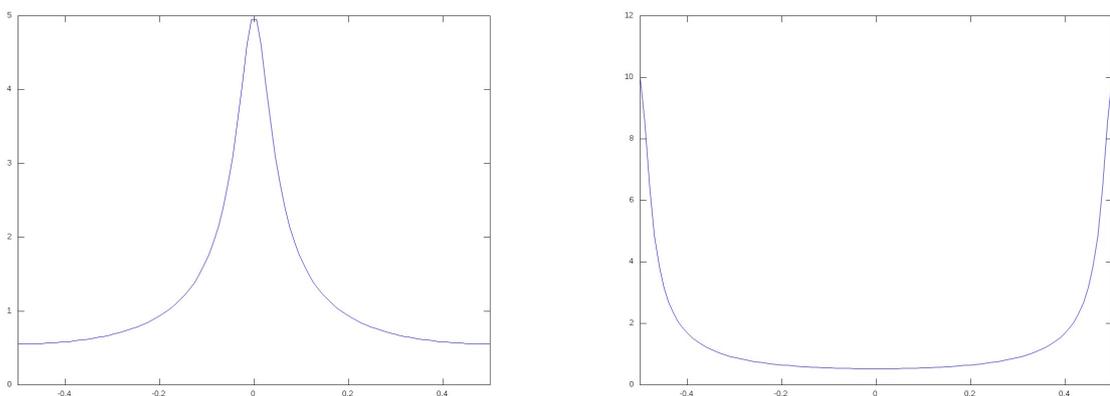
onde K é uma constante. Esta equação possui no lado direito um termo $y(n-1)$, que reutiliza uma saída computada anteriormente. Termos da forma $y(n-i)$ são chamados de termos de *feedback*, e sua utilização depende da definição de condições de contorno. Um exemplo de condição de contorno que torna a expressão acima bem-definida para qualquer $n \geq 0$ é $y(n) = 0, \forall n < 0$; ou seja, $y(0)$ será definido neste caso como $x(0) + K \cdot 0$, e as demais saídas serão computadas normalmente de acordo com a fórmula. Observe que se a entrada é um impulso discreto

$$\delta(n) = \begin{cases} 1, & \text{se } n = 0, \\ 0, & \text{cc} \end{cases}$$

então a saída será a resposta impulsiva do filtro $h(n)$, que neste caso será definida pela sequência $h(0) = 1, h(1) = K, h(2) = K^2, \dots$, ou seja, $h(n) = K^n$ para $n \geq 0$. A transformada z de $h(n)$ é por definição

$$H(z) = \sum_{n=-\infty}^{\infty} h(n)z^{-n} = \sum_{n=0}^{\infty} K^n z^{-n};$$

quando a série acima convergir (ou seja, quando $|Kz^{-1}| < 1$), seu valor será dado por $H(z) = \frac{1}{1-Kz^{-1}}$. Fazendo $z = e^{i2\pi f}$ para $f \in [-\frac{1}{2}, +\frac{1}{2}]$ temos a transformada de Fourier de tempo discreto de $h(n)$, que no caso dos filtros FIR representava a resposta em frequência do filtro (veremos que neste caso isso também é verdade). Os gráficos abaixo ilustram estas funções quando $K = 0.8$ e $K = -0.9$. Podemos ver que $K = 0.8$ produz um filtro passa-baixas, enquanto $K = -0.9$ produz um filtro passa-altas.



Uma observação sobre a nomenclatura: embora teoricamente seja possível definir um filtro a partir de uma equação de convolução com um sinal $h \in L^2(\mathbb{Z})$ qualquer, o termo *filtro IIR* é frequentemente usado como sinônimo para *filtro recursivo*, porque na prática só interessa usar filtros

de resposta impulsiva infinita quando eles possuem uma representação recursiva finita; no entanto, nem toda equação recursiva produz uma resposta ao impulso necessariamente de duração infinita. Por exemplo, a equação $y(n) = x(n) + x(n-1) - y(n-1)$ com a condição de contorno $y(n) = 0$ para $n < 0$ produz como resposta ao impulso a sequência $h = (1, 0, 0, 0, \dots)$, ou seja, a entrada é igual à saída, e possui duração finita. Este filtro na realidade é um FIR, e uma representação equivalente para ele é $y(n) = x(n)$, como veremos mais à frente.

Nosso primeiro objetivo será estabelecer que as equações recursivas acima de fato produzem filtros lineares que podem ser expressos através de uma equação de convolução $y = x * h$, onde h é uma resposta impulsiva (usualmente) infinita. Tal equação de convolução, embora não muito útil do ponto de vista computacional, fornece uma interpretação do filtro bastante útil, pois permite aproveitar a teoria desenvolvida para sequências infinitas em $L^2(\mathbb{Z})$ e assim entender a ação do filtro a partir da representação espectral (DTFT) dos sinais envolvidos, através do teorema da convolução ($Y=XH$). Em seguida vamos ver como determinar as respostas em frequência destes filtros diretamente a partir da equação do filtro, sem necessariamente computar a resposta impulsiva h como fizemos no exemplo acima.

2.1 Propriedades dos filtros recursivos

Considere a equação geral de um filtro recursivo, dada por

$$y(n) = \sum_{i \in I} a_i x(n-i) - \sum_{j \in J} b_j y(n-j),$$

onde $I, J \subset \mathbb{Z}$ são conjuntos finitos. O sinal negativo na frente da segunda somatória é apenas uma convenção, cuja finalidade ficará clara em breve.

A primeira e mais importante propriedade dos filtros recursivos é a **linearidade**, o que pode não ser imediatamente óbvio devido aos termos de feedback na equação. No entanto, considere que $y(n)$ é a saída correspondente à entrada $x(n)$ e que $z(n)$ é a saída correspondente à entrada $w(n)$ usando a mesma equação do filtro, ou seja $z(n) = \sum_{i \in I} a_i w(n-i) - \sum_{j \in J} b_j z(n-j)$, e considere dois escalares α e β quaisquer. Então, somando as duas equações acima multiplicadas por α e β respectivamente, teremos

$$\begin{aligned} \alpha y(n) + \beta z(n) &= \alpha \left[\sum_{i \in I} a_i x(n-i) - \sum_{j \in J} b_j y(n-j) \right] + \beta \left[\sum_{i \in I} a_i w(n-i) - \sum_{j \in J} b_j z(n-j) \right] \\ &= \sum_{i \in I} a_i [\alpha x(n-i) + \beta w(n-i)] - \sum_{j \in J} b_j [\alpha y(n-j) + \beta z(n-j)]. \end{aligned}$$

Definindo $v(n) = \alpha y(n) + \beta z(n)$ podemos re-escrever a equação acima como

$$v(n) = \sum_{i \in I} a_i [\alpha x(n-i) + \beta w(n-i)] - \sum_{j \in J} b_j v(n-j),$$

o que equivale a dizer que a saída do filtro para a entrada $\alpha x + \beta w$ é o sinal v definido por $v = \alpha y + \beta z$. Em outras palavras, se denotarmos por $\mathcal{F}(x)$ a saída do filtro a uma entrada qualquer x , podemos expressar a propriedade acima como

$$\mathcal{F}(\alpha x + \beta w) = \alpha \mathcal{F}(x) + \beta \mathcal{F}(w).$$

Outra propriedade importante é a chamada **invariância por translação no tempo**: em termos informais isso significa que se a entrada é atrasada ou adiantada em Δ amostras, então a saída

será atrasada ou adiantada correspondentemente. A verificação é simples: seja y a saída do filtro correspondente a x , e defina $w(n) = x(n - \Delta)$ (ou seja, w é x atrasado em Δ amostras, pois $w(\Delta) = x(0)$). Pela equação do filtro, teremos $y(n) = \sum_{i \in I} a_i x(n - i) - \sum_{j \in J} b_j y(n - j)$; substituindo n por $n - \Delta$ teremos

$$\begin{aligned} y(n - \Delta) &= \sum_{i \in I} a_i x(n - \Delta - i) - \sum_{j \in J} b_j y(n - \Delta - j) \\ &= \sum_{i \in I} a_i w(n - i) - \sum_{j \in J} b_j y(n - \Delta - j). \end{aligned}$$

Definindo $z(n) = y(n - \Delta)$ e substituindo na expressão acima, teremos $z(n) = \sum_{i \in I} a_i w(n - i) - \sum_{j \in J} b_j z(n - j)$, que é a mesma equação do filtro relacionando a entrada w com a saída z . Em outras palavras, $[\mathcal{F}(x)](n - \Delta) = y(n - \Delta) = z(n)$ para todo $n \in \mathbb{Z}$, ou seja, $\mathcal{F}(w) = z$. Quando não houver ambiguidade, vamos adotar a notação simplificada $\mathcal{F}(x(n))$ ao invés de $[\mathcal{F}(x)](n)$.

As duas propriedades acima permitem enunciar o seguinte resultado:

Teorema: Seja \mathcal{F} o filtro recursivo definido pela equação

$$y(n) = \sum_{i \in I} a_i x(n - i) - \sum_{j \in J} b_j y(n - j)$$

e h a resposta deste filtro ao impulso discreto $\delta(n) = \begin{cases} 1, & \text{se } n = 0, \\ 0, & \text{cc} \end{cases}$.

Então

$$y = x * h$$

e conseqüentemente as DTFTs satisfazem $Y(f) = X(f)H(f)$ e as transformadas z satisfazem $Y(z) = X(z)H(z)$.

Prova: Pela definição da resposta impulsiva, temos $\mathcal{F}(\delta) = h$, e pela propriedade da invariância por translação, temos que $\mathcal{F}(\delta(n - k)) = h(n - k)$, $\forall n \in \mathbb{Z}$. O sinal de entrada pode ser escrito alternativamente como $x(n) = \sum_{k=-\infty}^{\infty} x(k)\delta(n - k)$, pois esta soma se reduz a uma única parcela quando $\delta(n - k) = 1$, ou seja, quando $k = n$. Aplicando a linearidade na equação anterior, temos

$$\begin{aligned} y(n) &= \mathcal{F}(x(n)) \\ &= \mathcal{F}\left(\sum_{k=-\infty}^{\infty} x(k)\delta(n - k)\right) \\ &= \sum_{k=-\infty}^{\infty} x(k)\mathcal{F}(\delta(n - k)) \\ &= \sum_{k=-\infty}^{\infty} x(k)h(n - k), \end{aligned}$$

ou seja, $y = x * h$. As demais afirmações são consequência dos teoremas 4.5.1 e 4.5.2 do livro.

2.2 Resposta em frequência dos filtros recursivos

Nosso próximo objetivo é escrever a resposta em frequência diretamente a partir da equação do filtro recursivo. A propriedade da transformada z a seguir será imensamente útil.

Propriedade da translação no tempo: Seja $w(n) = x(n - \Delta)$ e $X(z)$ a transformada z de $x(n)$. Então a transformada z de $w(n)$ será

$$W(z) = z^{-\Delta}X(z).$$

Verificação: $W(z) = \sum_{n=-\infty}^{\infty} w(n)z^{-n} = \sum_{n=-\infty}^{\infty} x(n - \Delta)z^{-n}$. Fazendo $m = n - \Delta$ teremos

$$W(z) = \sum_{m=-\infty}^{\infty} x(m)z^{-(m+\Delta)} = z^{-\Delta} \sum_{m=-\infty}^{\infty} x(m)z^{-m} = z^{-\Delta}X(z).$$

Voltando à equação do filtro recursivo $y(n) = \sum_{i \in I} a_i x(n - i) - \sum_{j \in J} b_j y(n - j)$, podemos aplicar a definição da transformada z aos dois lados da expressão, utilizando a propriedade da linearidade da transformada e a propriedade da translação no tempo:

$$Y(z) = \sum_{i \in I} a_i z^{-i} X(z) - \sum_{j \in J} b_j z^{-j} Y(z).$$

Agrupando os termos com $Y(z)$ do lado esquerdo teremos

$$\left[1 + \sum_{j \in J} b_j z^{-j} \right] Y(z) = \left[\sum_{i \in I} a_i z^{-i} \right] X(z)$$

ou equivalentemente, sempre que $X(z) \neq 0$ e $\left[1 + \sum_{j \in J} b_j z^{-j} \right] \neq 0$, e já usando o fato de que $Y(z) = X(z)H(z)$,

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{i \in I} a_i z^{-i}}{1 + \sum_{j \in J} b_j z^{-j}}.$$

Esta expressão é chamada de **função de transferência**, e sua restrição aos valores $z = e^{i2\pi f}$ é a **resposta em frequência** do filtro recursivo. A forma do denominador com o sinal positivo na frente da somatória é a razão da convenção sobre o sinal negativo na equação do filtro.

Retomando o exemplo do filtro cuja equação é $y(n) = x(n) + x(n - 1) - y(n - 1)$, com a condição de contorno $y(n) = 0$ para $n < 0$, temos que sua função de transferência é

$$H(z) = \frac{1 + z^{-1}}{1 + z^{-1}},$$

que é equivalente a $H(z) = 1$, $\forall z \neq 1$ (sendo que o limite da expressão acima para $z \rightarrow 1$ também é 1); esta última função de transferência corresponde exatamente ao filtro cuja equação é $y(n) = x(n)$.

Exemplo O exemplo a seguir visa mostrar que o conjunto de índices J poderia conter índices negativos¹, correspondentes a termos de feedback do futuro, sem levar a uma contradição ou a uma impossibilidade de cálculo. Considere a equação

$$y(n) = x(n) + \frac{1}{2}y(n-1) + \frac{1}{2}y(n+1).$$

Deve ser fácil ver que apenas definir uma condição de contorno como $y(n) = 0, \forall n < 0$ não resolve o problema de como calcular $y(0)$ sem conhecer $y(1)$. Para ver como isso seria possível, vamos calcular a função de transferência deste filtro:

$$\begin{aligned} H(z) = \frac{Y(z)}{X(z)} &= \frac{1}{1 - \frac{1}{2}z^{-1} - \frac{1}{2}z^1} \\ &= \frac{1}{z^1 \left(z^{-1} - \frac{1}{2}z^{-2} - \frac{1}{2} \right)} \\ &= \frac{z^{-1}}{-\frac{1}{2} + z^{-1} - \frac{1}{2}z^{-2}}. \end{aligned}$$

Esta última é a função de transferência que corresponde ao filtro

$$-\frac{1}{2}\hat{y}(n) = x(n-1) - \hat{y}(n-1) - \frac{1}{2}\hat{y}(n-2),$$

ou equivalentemente

$$\hat{y}(n) = -2x(n-1) + 2\hat{y}(n-1) - \hat{y}(n-2).$$

Observe que esta expressão é análoga à expressão original isolando o termo $y(n+1)$:

$$y(n+1) = -2x(n) + 2y(n) - y(n-1),$$

e se torna idêntica fazendo $m = n+1$, de onde se pode concluir que $\hat{y}(n) = y(n)$, ou seja, o filtro original é equivalente a \hat{y} , cuja fórmula é computável a partir do conhecimento da entrada e das saídas anteriores.

2.3 Polos e zeros de filtros recursivos

A função de transferência de um filtro pode ser caracterizada equivalentemente pelas raízes (zeros) dos polinômios na variável z^{-1}

$$P(z) = \sum_{i \in I} a_i z^{-i} \quad \text{e} \quad Q(z) = 1 + \sum_{j \in J} b_j z^{-j}.$$

As raízes de $P(z)$ são chamadas de **zeros** do filtro, e as raízes de $Q(z)$ são chamadas de **polos** do filtro. É fácil ver que a função de transferência se anula nos zeros do filtro, ao passo que ela normalmente não está definida nos polos, pois o denominador se anula. Nas vizinhanças de um polo é de se esperar que a função de transferência tenda a $\pm\infty$ (embora isso nem sempre aconteça, como veremos a seguir).

¹observe que o conjunto de índices I pode conter índices negativos, o que já foi visto no início do texto quando falamos de filtros causais e não-causais

Um polinômio de grau N em \mathbb{C} sempre possui N raízes em \mathbb{C} , e quando os coeficientes do polinômio são reais (como é o nosso caso, pois $a_i, b_j \in \mathbb{R}$) então suas raízes são *ou* reais *ou* em pares complexo-conjugados. Isso é fácil de verificar, pois se z satisfaz $\sum_{i \in I} a_i z^{-i} = 0$, então

$$0 = \bar{0} = \overline{\left(\sum_{i \in I} a_i z^{-i} \right)} = \sum_{i \in I} \bar{a}_i \bar{z}^{-i} = \sum_{i \in I} a_i \bar{z}^{-i},$$

ou seja, \bar{z} também será raiz do polinômio.

Mais ainda, todo polinômio em \mathbb{C} pode ser fatorado deixando evidentes suas raízes. No caso da função de transferência isso corresponde à representação

$$H(z) = a_0 \frac{\prod_{i=1}^N (1 - Z_i z^{-1})}{\prod_{j=1}^M (1 - P_j z^{-1})},$$

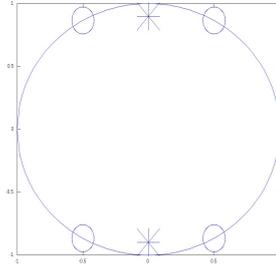
onde z_1, z_2, \dots, z_N são os zeros e P_1, P_2, \dots, P_M são os polos do filtro, e o coeficiente a_0 é o termo independente do polinômio $P(z)$ (o termo independente de $Q(z)$ é 1). Observe que se um polo é idêntico a um zero então os termos correspondentes das produtórias acima se anulam, e esse é o único caso em que a função de transferência pode estar definida exatamente em cima de um polo (para todos os efeitos práticos este polo e o zero coincidentes deixam de existir).

Podemos determinar a resposta em frequência de um filtro recursivo posicionando polos e zeros no plano complexo, criando regiões de ressonância (polos) e anti-ressonância (zeros) próximas a frequências que desejamos atingir. Isso tem relação com o fato de que uma componente senoidal de frequência f , correspondente ao ponto $z = e^{i2\pi f}$ no círculo unitário, responde ao filtro na proporção direta da distância de z a cada um dos zeros, e na proporção inversa da distância de z a cada um dos polos, pois

$$\begin{aligned} |H(z)| &= \left| \frac{a_0 \prod_{i=1}^N (1 - Z_i z^{-1})}{\prod_{j=1}^M (1 - P_j z^{-1})} \right| = |a_0| \frac{\prod_{i=1}^N |1 - Z_i z^{-1}|}{\prod_{j=1}^M |1 - P_j z^{-1}|} \\ &= |a_0| \frac{\prod_{i=1}^N |z^{-1}| |z - Z_i|}{\prod_{j=1}^M |z^{-1}| |z - P_j|} = |a_0| \frac{|z^{-N}| \prod_{i=1}^N |z - Z_i|}{|z^{-M}| \prod_{j=1}^M |z - P_j|} \\ &= |a_0| \frac{\prod_{i=1}^N |z - Z_i|}{\prod_{j=1}^M |z - P_j|} \end{aligned}$$

sendo a última passagem válida sempre que $|z| = 1$, o que é verdade para os pontos $z = e^{i2\pi f}$ de interesse da DTFT.

Exemplo: Vamos projetar um filtro para enfatizar a frequência $f = \frac{1}{4}$ (metade da frequência de Nyquist) usando um par complex-conjugado de polos em $z = Ke^{i\pi/2}$ e $z = Ke^{-i\pi/2}$ (com $|K| < 1$), e eliminar as frequências correspondentes a $f = \frac{1}{6}$ e $f = \frac{1}{3}$ usando zeros em $z = e^{i\pi/3}, e^{-i\pi/3}, e^{i2\pi/3}, e^{-i2\pi/3}$, conforme o diagrama abaixo:



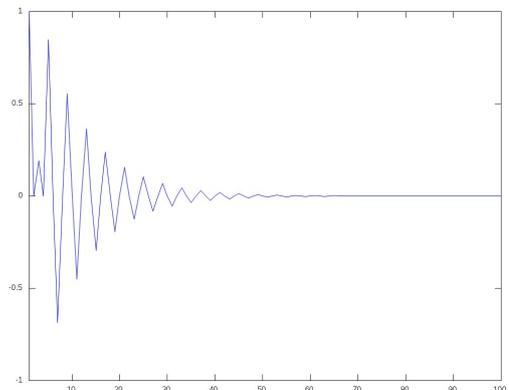
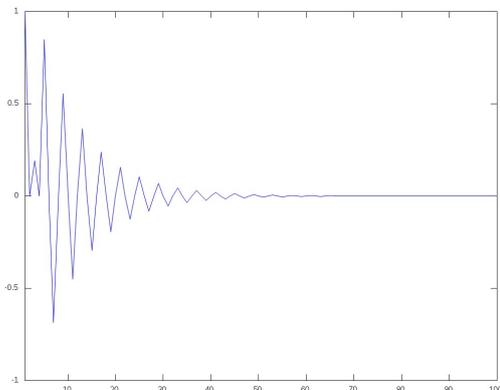
Isso corresponde à função de transferência

$$\begin{aligned}
 H(z) &= \frac{(1 - e^{i\pi/3}z^{-1})(1 - e^{-i\pi/3}z^{-1})(1 - e^{i2\pi/3}z^{-1})(1 - e^{-i2\pi/3}z^{-1})}{(1 - Ke^{i\pi/2}z^{-1})(1 - Ke^{-i\pi/2}z^{-1})} \\
 &= \frac{\left(1 - \overbrace{(e^{i\pi/3} + e^{-i\pi/3})}^{2 \cos(\pi/3)=1} z^{-1} + z^{-2}\right) \left(1 - \overbrace{(e^{i2\pi/3} + e^{-i2\pi/3})}^{2 \cos(2\pi/3)=-1} z^{-1} + z^{-2}\right)}{1 - K \underbrace{(e^{i\pi/2} + e^{-i\pi/2})}_{2 \cos(\pi/2)=0} z^{-1} + K^2 z^{-2}} \\
 &= \frac{(1 - z^{-1} + z^{-2})(1 + z^{-1} + z^{-2})}{1 + K^2 z^{-2}} \\
 &= \frac{1 + z^{-2} + z^{-4}}{1 + K^2 z^{-2}},
 \end{aligned}$$

o que pode ser realizado pelo filtro cuja equação é

$$y(n) = x(n) + x(n - 2) + x(n - 4) - K^2 y(n - 2).$$

Os gráficos abaixo mostram a resposta impulsiva e a resposta em frequência deste filtro, para $K = 0.9$.



2.4 Decomposição de filtros

Considere um filtro recursivo com N zeros e M polos, sendo N_1 zeros reais $Z_{\mathbb{R}}^1, Z_{\mathbb{R}}^2, \dots, Z_{\mathbb{R}}^{N_1}$ e $N_2 = \frac{N-N_1}{2}$ pares de zeros complexo-conjugados $Z_{\mathbb{C}}^1, \overline{Z_{\mathbb{C}}^1}, \dots, Z_{\mathbb{C}}^{N_2}, \overline{Z_{\mathbb{C}}^{N_2}}$, e analogamente M_1 polos reais $P_{\mathbb{R}}^1, P_{\mathbb{R}}^2, \dots, P_{\mathbb{R}}^{M_1}$ e $M_2 = \frac{M-M_1}{2}$ pares de polos complexo-conjugados $P_{\mathbb{C}}^1, \overline{P_{\mathbb{C}}^1}, \dots, P_{\mathbb{C}}^{M_2}, \overline{P_{\mathbb{C}}^{M_2}}$. A função de transferência será

$$H(z) = a_0 \frac{\prod_{i=1}^{N_1} (1 - Z_{\mathbb{R}}^i z^{-1}) \prod_{i=1}^{N_2} (1 - Z_{\mathbb{C}}^i z^{-1})(1 - \overline{Z_{\mathbb{C}}^i} z^{-1})}{\prod_{j=1}^{M_1} (1 - P_{\mathbb{R}}^j z^{-1}) \prod_{j=1}^{M_2} (1 - P_{\mathbb{C}}^j z^{-1})(1 - \overline{P_{\mathbb{C}}^j} z^{-1})}.$$

Observe que a expressão acima pode ser re-escrita como

$$H(z) = a_0 \prod_{i=1}^{N_1} H_{\mathbb{R}}^i \prod_{i=1}^{N_2} H_{\mathbb{C}}^i \prod_{j=1}^{M_1} \tilde{H}_{\mathbb{R}}^j \prod_{j=1}^{M_2} \tilde{H}_{\mathbb{C}}^j,$$

onde

$$\begin{aligned} H_{\mathbb{R}}^i(z) &= 1 - Z_{\mathbb{R}}^i z^{-1}, & H_{\mathbb{C}}^i(z) &= (1 - Z_{\mathbb{C}}^i z^{-1})(1 - \overline{Z_{\mathbb{C}}^i} z^{-1}), \\ \tilde{H}_{\mathbb{R}}^j(z) &= \frac{1}{1 - P_{\mathbb{R}}^j z^{-1}}, & \tilde{H}_{\mathbb{C}}^j(z) &= \frac{1}{(1 - P_{\mathbb{C}}^j z^{-1})(1 - \overline{P_{\mathbb{C}}^j} z^{-1})}, \end{aligned}$$

que são filtros de 1 zero, 2 zeros, 1 polo e 2 polos, respectivamente. Observando que

$$\begin{aligned} (1 - Z_{\mathbb{C}}^i z^{-1})(1 - \overline{Z_{\mathbb{C}}^i} z^{-1}) &= 1 - (Z_{\mathbb{C}}^i + \overline{Z_{\mathbb{C}}^i})z^{-1} + Z_{\mathbb{C}}^i \overline{Z_{\mathbb{C}}^i} z^{-2} \\ &= 1 - 2 \cos(\arg(Z_{\mathbb{C}}^i))z^{-1} + |Z_{\mathbb{C}}^i|^2 z^{-2} \\ &= 1 - 2 \operatorname{real}(Z_{\mathbb{C}}^i)z^{-1} + |Z_{\mathbb{C}}^i|^2 z^{-2}, \end{aligned}$$

e fazendo a conta análoga para os polos, teremos que os filtros $H_{\mathbb{R}}^i, H_{\mathbb{C}}^i, \tilde{H}_{\mathbb{R}}^j$ e $\tilde{H}_{\mathbb{C}}^j$ correspondem às equações de filtro abaixo:

$$\begin{aligned} H_{\mathbb{R}}^i : \quad y(n) &= x(n) - Z_{\mathbb{R}}^i x(n-1), \\ H_{\mathbb{C}}^i : \quad y(n) &= x(n) - 2 \operatorname{real}(Z_{\mathbb{C}}^i) x(n-1) + |Z_{\mathbb{C}}^i|^2 x(n-2), \\ \tilde{H}_{\mathbb{R}}^j : \quad y(n) &= x(n) - P_{\mathbb{R}}^j y(n-1), \\ \tilde{H}_{\mathbb{C}}^j : \quad y(n) &= x(n) + 2 \operatorname{real}(P_{\mathbb{C}}^j) y(n-1) - |P_{\mathbb{C}}^j|^2 y(n-2). \end{aligned}$$

Esta decomposição da função de transferência de um filtro recursivo qualquer, aliada ao teorema da convolução, nos permite concluir que um filtro recursivo sempre pode ser decomposto como uma cadeia em série de filtros mais simples, contando com apenas 1 ou 2 zeros ou 1 ou 2 polos. No domínio do tempo isso corresponde a uma sequência de convoluções do sinal original por cada uma das respostas impulsivas $h_{\mathbb{R}}^i, h_{\mathbb{C}}^i, \tilde{h}_{\mathbb{R}}^j$ e $\tilde{h}_{\mathbb{C}}^j$ dos filtros acima. Observe mais uma vez que, se $Z^i = P^j$, então os filtros H^i e \tilde{H}^j anulam o efeito um do outro.

3 Filtros recursivos 2D

Para imagens em $L^2(\mathbb{Z} \times \mathbb{Z})$ temos as mesmas propriedades que vimos na extensão da teoria de \mathbb{C}^N para $L^2(\mathbb{Z})$, sendo as demonstrações extremamente parecidas, porém envolvendo duas somatórias

ao invés de uma. Supondo que $\sum_{r,s=-\infty}^{\infty} |a_{r,s}|^2 < \infty$ e que $\sum_{r,s=-\infty}^{\infty} |b_{r,s}|^2 < \infty$, temos que o produto interno

$$(A, B) = \sum_{r,s=-\infty}^{\infty} a_{r,s} \bar{b}_{r,s}$$

está bem-definido, assim como a transformada de Fourier em $L^2(\mathbb{Z} \times \mathbb{Z})$, dada por

$$\hat{A}(f_1, f_2) = \sum_{r,s=-\infty}^{\infty} a_{r,s} e^{-i2\pi(rf_1 + sf_2)}, \text{ onde } f_1, f_2 \in \left[-\frac{1}{2}, +\frac{1}{2}\right],$$

cuja inversa é

$$a_{r,s} = \int_{-\frac{1}{2}}^{\frac{1}{2}} \int_{-\frac{1}{2}}^{\frac{1}{2}} \hat{A}(f_1, f_2) e^{i2\pi(rf_1 + sf_2)} df_1 df_2.$$

A convolução de duas imagens em $L^2(\mathbb{Z} \times \mathbb{Z})$ é definida como

$$(A * B)_{r,s} = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} a_{k,l} b_{r-k,s-l},$$

e se $C = A * B$, então $\hat{C}(f_1, f_2) = \hat{A}(f_1, f_2) \hat{B}(f_1, f_2)$.

A transformada z em 2D é

$$\hat{A}(z_1, z_2) = \sum_{r,s=-\infty}^{\infty} a_{r,s} z_1^{-r} z_2^{-s},$$

que coincide com a transformada de Fourier quando $z_1 = e^{i2\pi f_1}$ e $z_2 = e^{i2\pi f_2}$. Também é análoga ao caso unidimensional a demonstração de que se $C = A * B$ então $\hat{C}(z_1, z_2) = \hat{A}(z_1, z_2) \hat{B}(z_1, z_2)$, $\forall z_1, z_2$.

Vamos estudar então equações de filtros da forma

$$y(n, m) = \sum_{(i,j) \in I} a_{i,j} x(n-i, m-j) - \sum_{(k,l) \in J} b_{k,l} y(n-k, m-l),$$

onde $I, J \subset \mathbb{Z} \times \mathbb{Z}$ são conjuntos finitos. A verificação de que tais filtros são lineares e invariantes por translação espacial é análoga ao caso unidimensional.

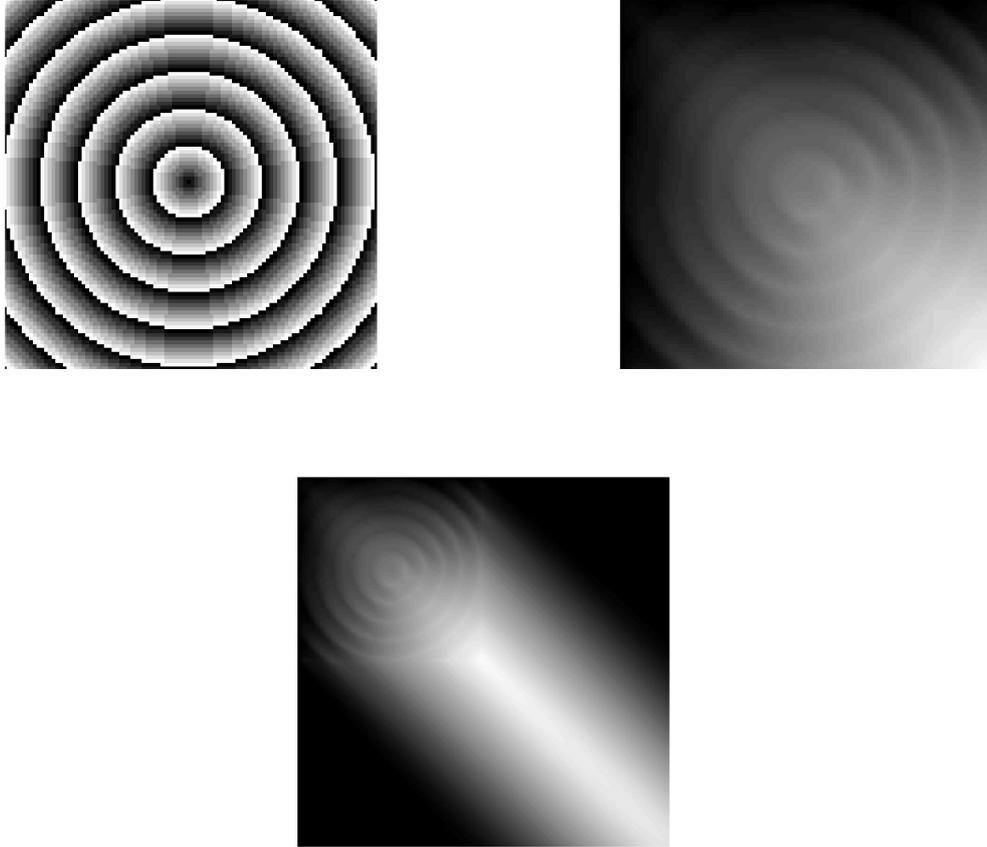
Exemplo: Considere a equação dada por

$$y(n, m) = x(n, m) + \frac{1}{2}y(n-1, m) + \frac{1}{2}y(n, m-1).$$

Dada uma entrada impulsiva da forma $\delta(n, m) = \begin{cases} 1, & \text{se } n = m = 0 \\ 0, & \text{cc} \end{cases}$ e as condições de contorno $y(n, m) = 0$, $\forall n < 0$ ou $m < 0$, teremos como saída do filtro a matriz

$n \backslash m$	0	1	2	3	4	...
0	1	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$...
1	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{3}{8}$	$\frac{4}{16}$	$\frac{5}{32}$...
2	$\frac{1}{4}$	$\frac{3}{8}$	$\frac{3}{8}$	$\frac{5}{16}$	$\frac{15}{64}$...
3	$\frac{1}{8}$	$\frac{4}{16}$	$\frac{5}{16}$	$\frac{5}{16}$	$\frac{35}{128}$...
...

Podemos observar o efeito deste filtro na seguinte sequência de imagens, que correspondem a uma entrada arbitrária, à imagem processada pelo filtro acima, e à extensão da imagem processada para além das bordas da figura original, considerando a extensão da imagem original com zeros (pixels pretos).



Aqui a transformada z também satisfaz a **propriedade da translação (espacial)**: se $y(n, m) = x(n - \Delta, m - \Gamma)$ então

$$\begin{aligned} Y(z_1, z_2) &= \sum_{r,s} y(r, s) z_1^{-r} z_2^{-s} &= \sum_{r,s} x(r - \Delta, s - \Gamma) z_1^{-r} z_2^{-s} \\ &= \sum_{m,n} x(m, n) z_1^{-(m+\Delta)} z_2^{-(n+\Gamma)} &= z_1^{-\Delta} z_2^{-\Gamma} \sum_{m,n} x(m, n) z_1^{-m} z_2^{-n} \\ &= z_1^{-\Delta} z_2^{-\Gamma} X(z_1, z_2), \end{aligned}$$

onde a segunda linha é obtida substituindo-se $m = r - \Delta$ e $n = s - \Gamma$. Com esta propriedade podemos determinar a função de transferência do filtro recursivo com equação $y(n, m) = \sum_{(i,j) \in I} a_{i,j} x(n - i, m - j) - \sum_{(k,l) \in J} b_{k,l} y(n - k, m - l)$ de forma análoga ao que foi feito no caso unidimensional:

$$Y(z_1, z_2) = \sum_{(i,j) \in I} a_{i,j} z_1^{-i} z_2^{-j} X(z_1, z_2) - \sum_{(k,l) \in J} b_{k,l} z_1^{-k} z_2^{-l} Y(z_1, z_2),$$

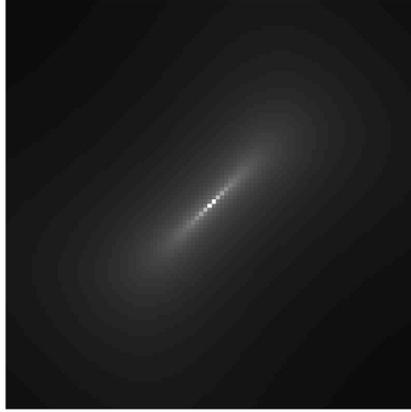
de onde

$$H(z_1, z_2) = \frac{Y(z_1, z_2)}{X(z_1, z_2)} = \frac{\sum_{(i,j) \in I} a_{i,j} z_1^{-i} z_2^{-j}}{1 + \sum_{(k,l) \in J} b_{k,l} z_1^{-k} z_2^{-l}}.$$

No caso do exemplo anterior temos a função de transferência

$$H(z_1, z_2) = \frac{1}{1 - \frac{1}{2}z_1^{-1} - \frac{1}{2}z_2^{-1}},$$

e podemos visualizar o logaritmo da resposta em magnitude deste filtro no gráfico abaixo:



Outra propriedade interessante da transformada z em 2D é a da **mudança linear de variáveis**, que corresponde a operações de mudança de escala e rotações de uma imagem no plano. Especificamente, sejam $\alpha, \beta, \gamma, \delta \in \mathbb{Z}$, e considere a imagem y definida por

$$y(n, m) = x(\alpha n + \beta m, \gamma n + \delta m).$$

Vamos supor que x é 0 nos pontos que não foram utilizados em y , ou seja, em pontos que não têm a forma $(\alpha n + \beta m, \gamma n + \delta m)$. As transformadas z de x e y são relacionadas pela equação

$$X(z_1, z_2) = Y(z_1^\alpha z_2^\gamma, z_1^\beta z_2^\delta),$$

o que pode ser verificado como segue:

$$\begin{aligned} Y(z_1^\alpha z_2^\gamma, z_1^\beta z_2^\delta) &= \sum_{r,s} y(r, s) (z_1^\alpha z_2^\gamma)^{-r} (z_1^\beta z_2^\delta)^{-s} \\ &= \sum_{r,s} x(\alpha r + \beta s, \gamma r + \delta s) z_1^{-(\alpha r + \beta s)} z_2^{-(\gamma r + \delta s)} \\ &= \sum_{n,m} x(n, m) z_1^{-n} z_2^{-m} \\ &= X(z_1, z_2) \end{aligned}$$

onde o penúltimo passo corresponde a definir $(n, m) = (\alpha r + \beta s, \gamma r + \delta s)$, e lembrar da hipótese de que $x(n, m) = 0$ nos outros pontos.

O desenho de filtros recursivos usando o posicionamento manual de polos e zeros, de forma independente para z_1 e para z_2 , é possível, bem como é possível a composição de filtros complexos a partir da combinação de filtros simples, obtendo os coeficientes da equação do filtro a partir dos termos da função de transferência. No entanto, devemos chamar a atenção para diferenças importantes que ocorrem quando passamos de 1D para 2D.

Inicialmente observe que as transformadas z dos sinais envolvidos são agora funções de duas variáveis complexas. Em particular, a visualização de zeros e polos ou de regiões de convergência é mais complicada, pois $(z_1, z_2) \in \mathbf{C}^2$. No caso das regiões de convergência, dado que elas dependem apenas das magnitudes $|z_1|$ e $|z_2|$, é frequente representarmos estas regiões no plano $|z_1| \times |z_2|$, chamado de plano de magnitude. O conjunto de todos os pontos $z_1 = e^{i2\pi f_1}$ e $z_2 = e^{i2\pi f_2}$ que correspondem às funções básicas da decomposição de Fourier correspondem ao chamado bi-círculo unitário em \mathbf{C}^2 , que não deve ser confundido com o círculo unitário em \mathbf{C}^2 ; em particular, todos os pontos do bi-círculo satisfazem $|z_1| = |z_2| = 1$, e portanto correspondem todos ao mesmo ponto $(1, 1)$ no plano de magnitude.

Uma das diferenças mais fundamentais no estudo dos polos e zeros em 2D é o fato de que eles normalmente não são pontos isolados, como no caso unidimensional. Isso pode ser facilmente compreendido através de um exemplo.

Exemplo: Considere a função de transferência dada por

$$H(z_1, z_2) = 1 + 2z_1^{-1} + 2z_2^{-1} + z_1^{-1}z_2^{-1}.$$

Observe que

$$\begin{aligned} H(z_1, z_2) = 0 &\iff 1 + 2z_2^{-1} + z_1^{-1}(2 + z_2^{-1}) = 0 \\ &\iff z_1^{-1} = -\frac{(1+2z_2^{-1})}{2+z_2^{-1}} \\ &\iff z_1^{-1} = -\frac{z_2}{z_2} \frac{(1+2z_2^{-1})}{2+z_2^{-1}} \\ &\iff z_1^{-1} = -\frac{(1+2z_2)}{2+z_2}, \end{aligned}$$

ou seja, para cada valor de $z_2 \neq 2$ temos um valor de z_1 associado tal que $X(z_1, z_2) = 0$. Este conjunto é denominado um **locus de zeros**.

Para usar outro exemplo, considere

$$H(z_1, z_2) = \frac{1 + z_1}{1 + z_1 z_2}.$$

Neste caso temos como locus de zeros o conjunto $\{(z_1, z_2) \mid z_1 = 1\}$ e como **locus de polos** o conjunto $\{(z_1, z_2) \mid z_1 \neq 0 \text{ e } z_2 = -\frac{1}{z_1}\}$. Este exemplo mostra que um locus de zeros pode interceptar um locus de polos (estes dois conjuntos se interceptam no ponto $(1,1)$) sem que haja cancelamento de polos e zeros (pois as expressões $1 + z_1$ e $1 + z_1 z_2$ não são idênticas).

Outra diferença fundamental relacionada ao exemplo anterior é que normalmente não é possível fatorar um polinômio em duas variáveis usando apenas termos simples da forma $(1 - Rz^{-1})$. Para ver isso basta tomar a expressão $f(z_1, z_2) = 1 - z_1^{-1}z_2^{-1}$ e observar que não existem A e B tais que $g(z_1, z_2) = (1 - Az_1^{-1})(1 - Bz_2^{-1})$ se torne idêntica a $f(z_1, z_2)$ (A e B teriam que satisfazer $AB = -1$ e $A = B = 0$).

3.1 Projeto de filtros recursivos

Frequentemente o problema do projeto de filtros recursivos (1D ou 2D) corresponde a encontrar coeficientes $a_{i,j}$ para $(i, j) \in I$ e $b_{k,l}$ para $(k, l) \in J$ de tal maneira que a resposta em frequência do filtro seja semelhante a uma dada resposta ideal, que corresponde a um processamento específico que

desejamos fazer. Lembremos que, no caso do projeto de filtros FIR, podíamos discretizar a resposta ideal e obter os coeficientes do filtro FIR usando a inversa da transformada de Fourier. A vantagem do uso de filtros recursivos é que usualmente podemos atingir uma resposta suficientemente próxima da resposta ideal usando uma quantidade muito menor de coeficientes do filtro, o que leva a uma vantagem computacional.

O problema fica configurado assim: dada uma resposta em frequência ideal $H_I(f_1, f_2)$ (para $f_1, f_2 \in [-\frac{1}{2}, +\frac{1}{2}]$), desejamos encontrar coeficientes $a_{i,j}$ e $b_{k,l}$ de tal forma que a resposta do filtro correspondente

$$H(f_1, f_2) = \frac{\sum_{(i,j) \in I} a_{i,j} e^{-i2\pi(i f_1 + j f_2)}}{1 + \sum_{(k,l) \in J} b_{k,l} e^{-i2\pi(k f_1 + l f_2)}}$$

seja a mais próxima possível da resposta ideal. Alguns critérios para especificar o que significa a expressão “a mais próxima possível” são:

Projeto levando em consideração magnitude e fase: neste caso o objetivo do projeto é minimizar a expressão

$$\|H - H_I\|^2 = \int_{-\frac{1}{2}}^{\frac{1}{2}} \int_{-\frac{1}{2}}^{\frac{1}{2}} |H(f_1, f_2) - H_I(f_1, f_2)| df_1 df_2.$$

Projeto levando em consideração apenas a magnitude: quando as alterações de fase não são importantes para a aplicação, podemos adotar como critério de minimização a função

$$\||H| - |H_I|\|^2 = \int_{-\frac{1}{2}}^{\frac{1}{2}} \int_{-\frac{1}{2}}^{\frac{1}{2}} \||H(f_1, f_2)| - |H_I(f_1, f_2)|\| df_1 df_2,$$

o que pode levar a um erro de projeto menor do que o anterior (às custas de não controlarmos o que acontece com as fases).

Aproximação no domínio do tempo: podemos considerar a minimização do ponto de vista das respostas impulsivas h e h_I , minimizando a expressão

$$\|h - h_I\|^2 = \sum_{n=-\infty}^{\infty} |h(n) - h_I(n)|^2.$$

Detalhes de como estes e outros métodos de projeto de filtros recursivos são implementados podem ser encontrados no livro do Woods.