



北京理工大学
Beijing Institute of Technology

本科生实验报告

课程名称: 高级数字信号处理

实验名称: Levinson 求功率谱

任课教师:	田黎育		实验教师:	田黎育			
实验日期:	2025 年 11 月 24 日		实验地点:				
实验类型:	<input checked="" type="checkbox"/> 原理验证		<input type="checkbox"/> 综合设计	<input type="checkbox"/> 自主创新			
学生姓名:	闫子易	班级:	13212302	学号:	1120230621		
学院:	集成电路与电子学院		专业:	电子科学与技术			
组号:		同组同学:					
成绩:							



集成电路与电子学院
SCHOOL OF INTEGRATED CIRCUITS
AND ELECTRONICS

一、实验目的

- ① 用 MATLAB 实现算法，32 点输入（前用）
- ② 画 δ^2 的变化趋势（输入不同 SNR，多组 δ^2 曲线）

二、实验原理

Levinson 递推法在 AR 模型自相关法的基础上进行了优化，通过自相关函数的递推求解 AR 模型的系数，并估计功率谱，避免了直接求解方程的计算复杂度。

Levinson 递推法采用的前向预测可表示为

$$\hat{x}[n] = - \sum_{i=1}^P a_i x[n-i]$$

则前向预测误差为

$$e[n] = x[n] - \hat{x}[n] = x[n] + \sum_{i=1}^P a_i x[n-i]$$

而 AR 模型的差分方程为

$$x(n) = - \sum_{i=1}^P a_i x(n-i) + \omega(n)$$

对比可知，若两个模型的 a_i 相等，则预测误差就是白噪声。利用模型的等效性，可以逐步求解 Yule-Walker 方程。

由 AR 模型自相关的原理可知，自相关矩阵具有对称性，平行主对角线的元素共轭（矩阵元素为实数则相等）。已知 P 阶 AR 模型的参数 $a_{P,1}, a_{P,2}, \dots, a_{P,P}$ ，若要递推求解 $P+1$ 阶的参数，可利用自相关矩阵的对称性，得到 P 阶 Y-W 方程

$$\begin{bmatrix} R(0) & R(1) & \dots & R(P) \\ R(1) & R(0) & \dots & R(P-1) \\ \vdots & \vdots & \ddots & \vdots \\ R(P) & R(P-1) & \dots & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_{P,1} \\ \vdots \\ a_{P,P} \end{bmatrix} = \begin{bmatrix} \sigma_P^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

其中 $R(m)$ 即为自相关函数 $R_{xx}(m)$ 。

对该方程扩展右列、扩展下行，得到方程①

$$\begin{bmatrix} R(0) & R(1) & \dots & R(P) & R(P+1) \\ R(1) & R(0) & \dots & R(P-1) & R(P) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ R(P) & R(P-1) & \dots & R(0) & R(1) \\ R(P+1) & R(P) & \dots & R(1) & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_{P,1} \\ \vdots \\ a_{P,P} \\ 0 \end{bmatrix} = \begin{bmatrix} \sigma_P^2 \\ 0 \\ \vdots \\ 0 \\ \Delta_{P+1} \end{bmatrix}$$

其中 $\Delta_{P+1} = R(P+1) + \sum_{i=1}^P a_{P,i} R(P+1-i)$ 。

再对方程①作行反转、列反转，得到方程②

$$\begin{bmatrix} R(0) & R(1) & \dots & R(P) & R(P+1) \\ R(1) & R(0) & \dots & R(P-1) & R(P) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ R(P) & R(P-1) & \dots & R(0) & R(1) \\ R(P+1) & R(P) & \dots & R(1) & R(0) \end{bmatrix} \begin{bmatrix} 0 \\ a_{P,P} \\ \vdots \\ a_{P,1} \\ 1 \end{bmatrix} = \begin{bmatrix} \Delta_{P+1} \\ 0 \\ \vdots \\ 0 \\ \sigma_P^2 \end{bmatrix}$$

令反射系数 $K_{P+1} = -\frac{\Delta_{P+1}}{\sigma_P^2}$ ，令 $\langle 2 \rangle \times K_{P+1} + \langle 1 \rangle$ ，得

$$\begin{bmatrix} R(0) & \dots & R(P) & R(P+1) \\ \vdots & \ddots & \vdots & \vdots \\ R(P) & \dots & R(0) & R(1) \\ R(P+1) & \dots & R(1) & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_{P,1} + K_{P+1} a_{P,P} \\ \vdots \\ a_{P,P} + K_{P+1} a_{P,1} \\ K_{P+1} \end{bmatrix} = \begin{bmatrix} \sigma_P^2 + K_{P+1} \Delta_{P+1} \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

而 $P+1$ 阶 Yule-Walker 方程为

$$\begin{bmatrix} R(0) & \dots & R(P) & R(P+1) \\ \vdots & \ddots & \vdots & \vdots \\ R(P) & \dots & R(0) & R(1) \\ R(P+1) & \dots & R(1) & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_{P+1,1} \\ \vdots \\ a_{P+1,P+1} \end{bmatrix} = \begin{bmatrix} \sigma_{P+1}^2 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

对比可得

$$a_{P+1,P+1} = K_{P+1} = -\frac{\Delta_{P+1}}{\sigma_P^2} = -\frac{R(P+1) + \sum_{i=1}^P a_{P,i} R(P+1-i)}{\sigma_P^2}$$

$$a_{P+1,k} = a_{P,k} + K_{P+1} a_{P,P+1-k}$$

$$\sigma_{P+1}^2 = \sigma_P^2 + K_{P+1} \Delta_{P+1} = (1 - K_{P+1}^2) \sigma_P^2$$

由此可知，递推过程中 σ_P^2 逐渐减小。

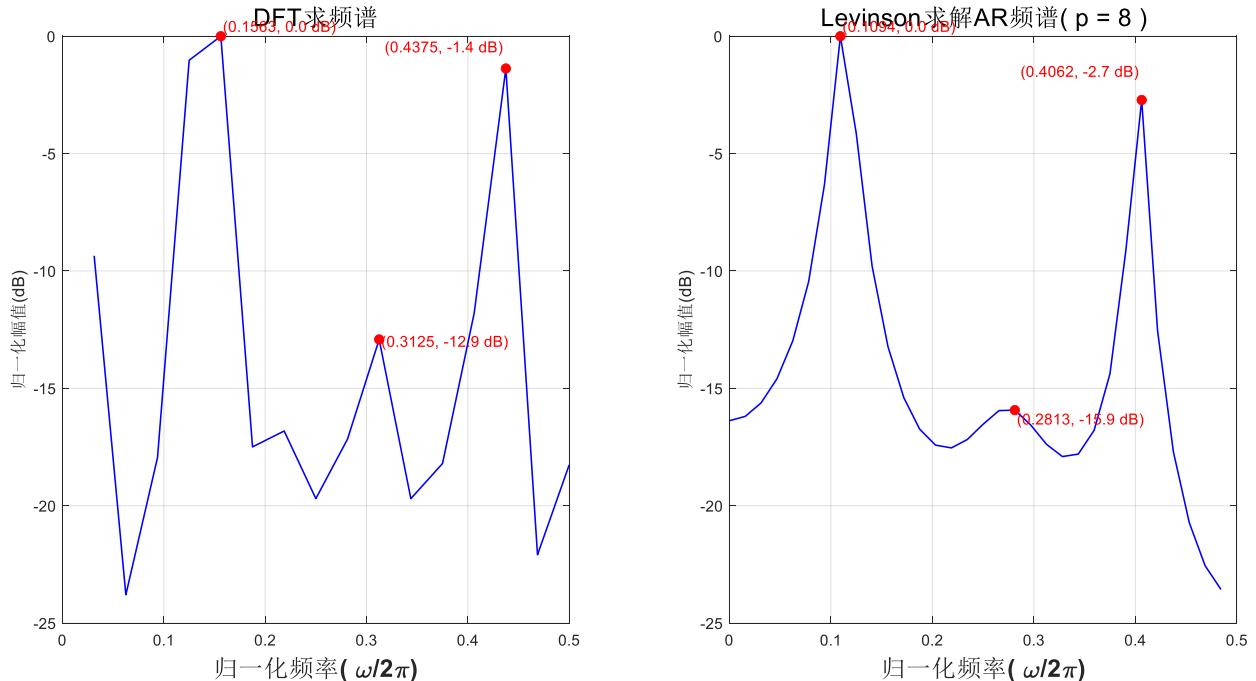
对于 1 阶模型，有 $a_{1,1} = \frac{R(1)}{R(0)}$, $\sigma_1^2 = R(0)(1 - a_{1,1}^2)$ ，由此可实现递推求解。

三、运行代码及实验内容

序列长度 $N = 32$ ，信号频率分别为 $f_1 = 100\text{Hz}$ ， $f_2 = 120\text{Hz}$ ， $f_3 = 405\text{Hz}$ 采样率 $fs = 1000\text{Hz}$ ，信噪比 $\text{SNR}=10$ ，各频率分量幅值分别为 $A_1 = A_2 = A_3 = 1$ 。对序列使用 32 点 DFT 和 Levinson 模型递推求功率谱。

我用红点标注出了两个曲线中最高的 3 个点，也就是如果频谱能分辨出的话理论上三个信号频率的所在位置。

先设 $p=8$



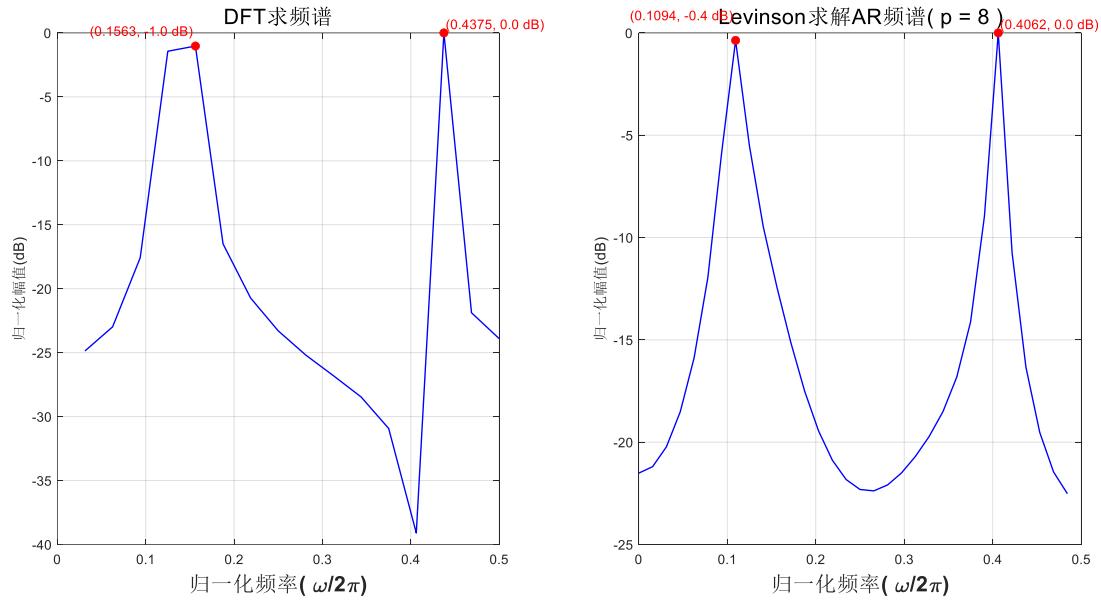
可以看到， $p=8$ 时，DFT 和 Levinson 法都可以区分 405Hz 和两外两个信号，但频率相近的 100Hz 和 120Hz 的信号两种方法都区分不开。

DFT 法出现这个现象很好解释，因为 DFT 的频率分辨率由采样率与变换点数决定：

$$\Delta f = \frac{f_s}{N}$$

在代码里 $f_s = 1000$, $N_{\text{fft}} = 32$ ，所以 $\Delta f = 1000/32 = 31.25\text{ Hz}$ 。而两个信号频率差值位 $120-100=20\text{Hz}$ ，明显小于 DFT 的频率分辨率，所以 DFT 法无法在这个信号参数下分辨出两个信号，即使 SNR 很大没有噪声。

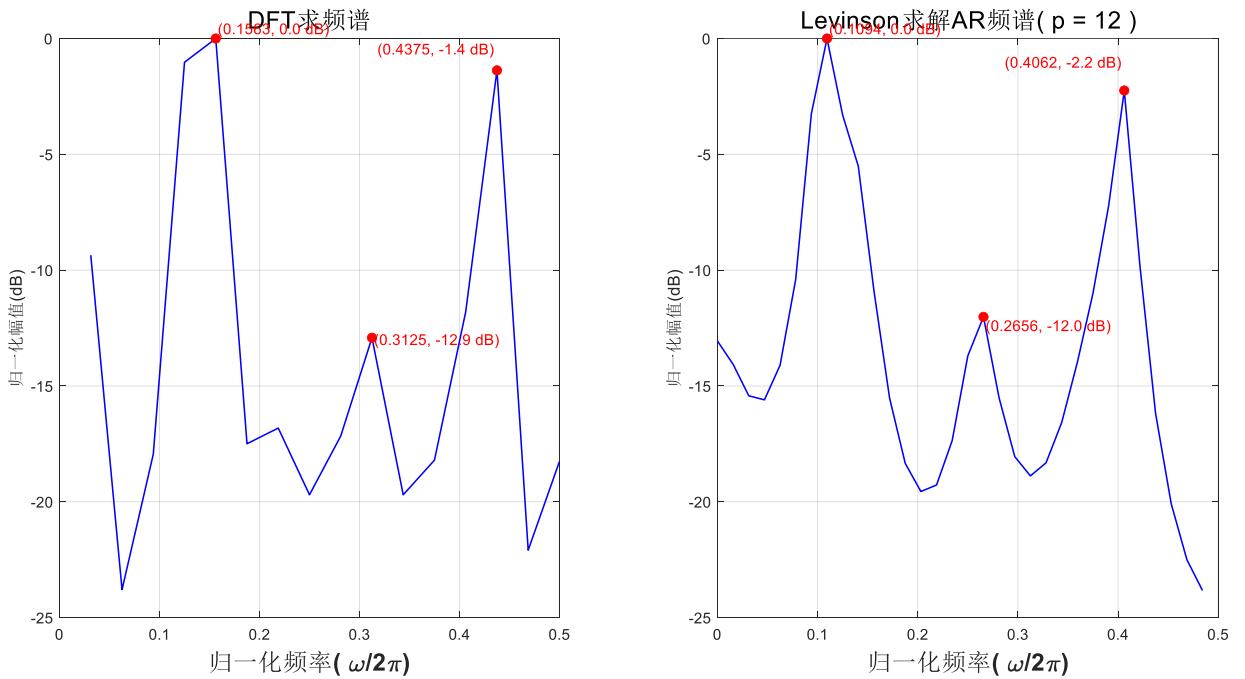
那相同信号条件下 Levinson 递推法能分辨出两条谱线吗？我们继续实验。



先加大 SNR 到 1000dB，发现两种方法还是无法区分开两条谱线，说明 SNR 在此时的影响已经不大了。

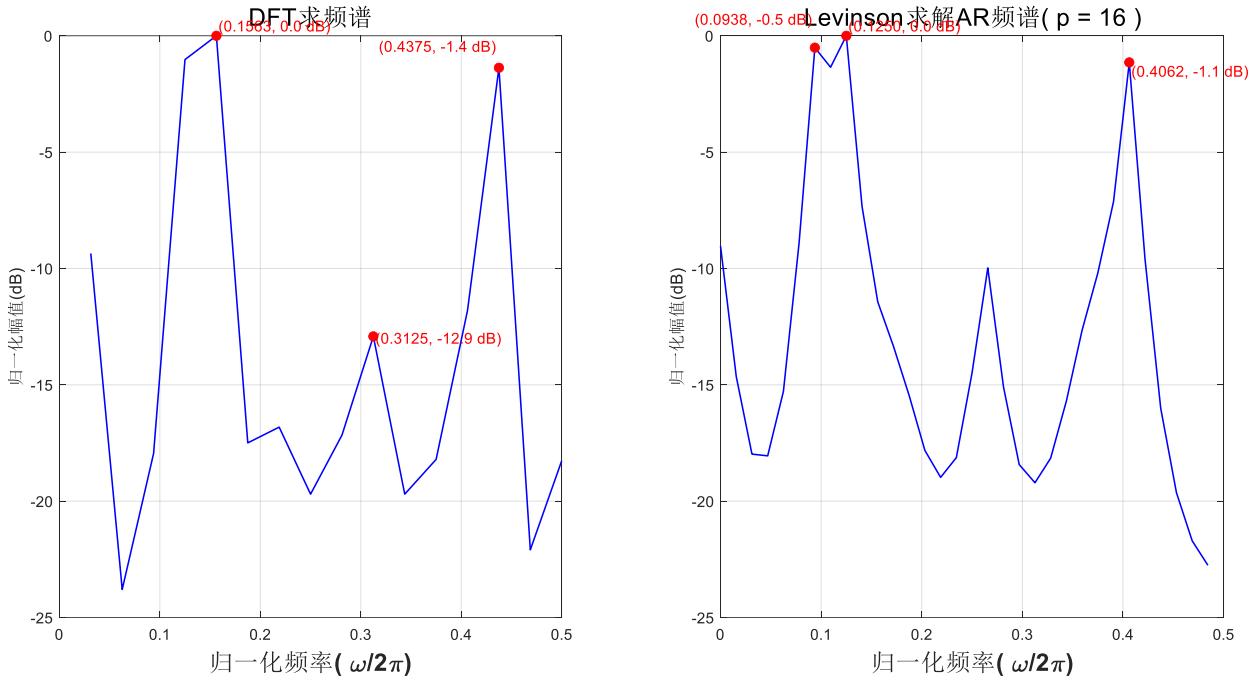
于是我们重新设置 SNR=10dB，增大阶数 p，看 Levinson 法能否分辨出来。

当 $p=12$ 时



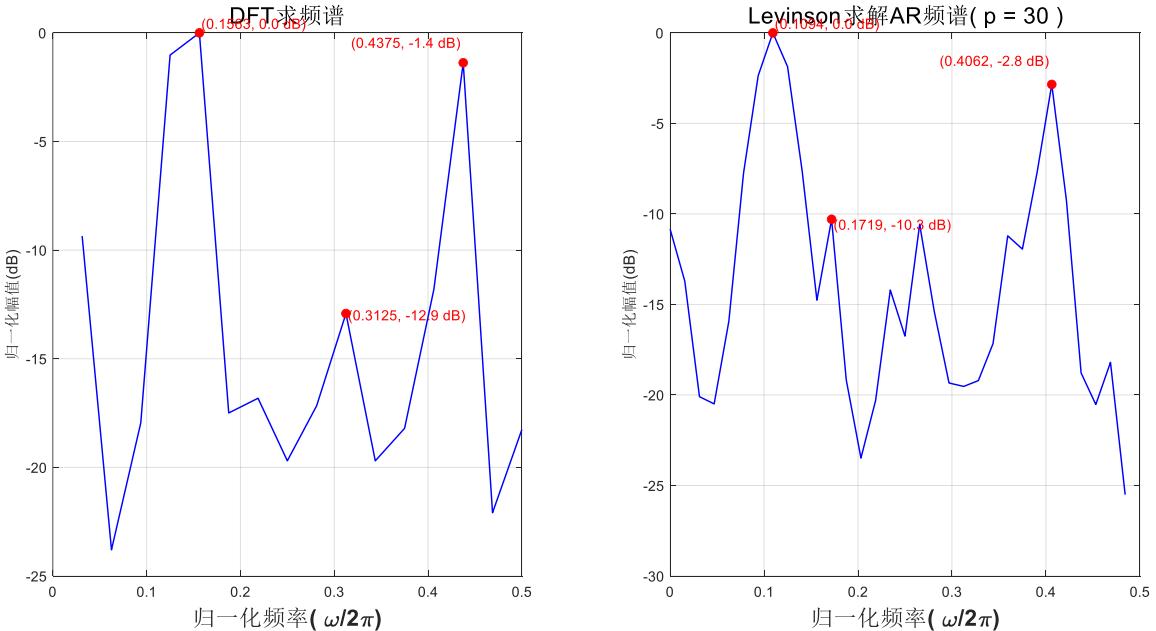
Levinson 法依然无法分辨出两条谱线。所以我们接着加大 p。

当 $p=16$ 时，也就是 $N/2$ 时



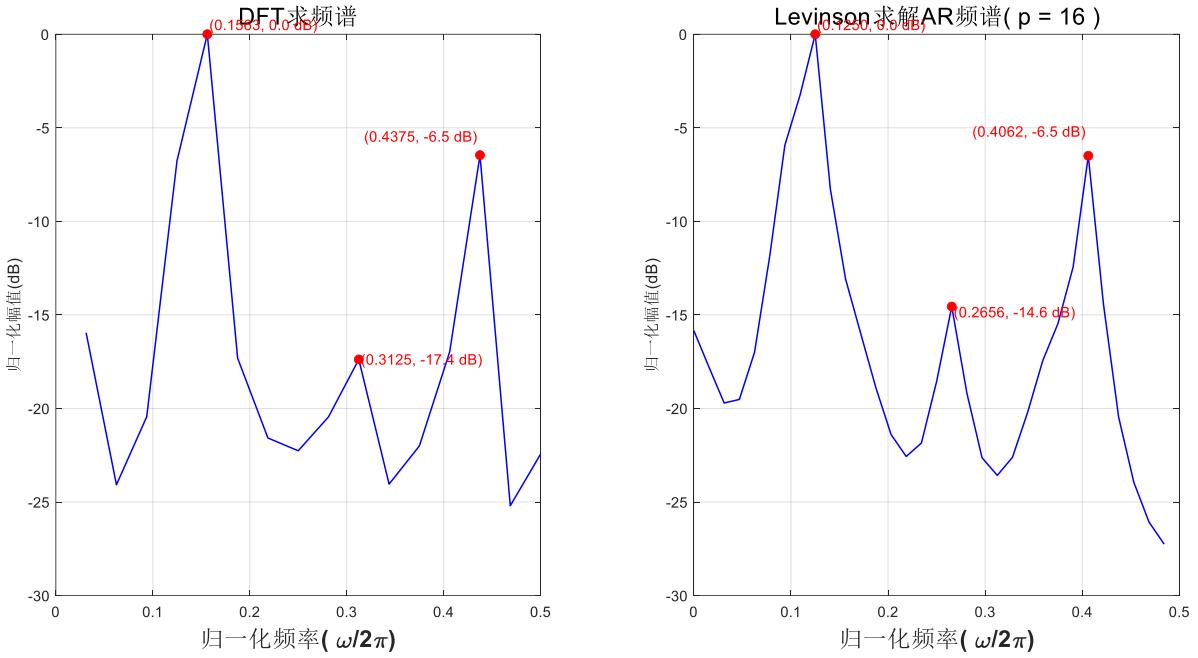
可以看到，此时 DFT 法依然无法分辨，但 Levinson 法分辨出了两条谱线，归一化后的频率显示两个信号的频率分别为 93.8Hz 和 1280Hz，这与原信号的 100Hz 和 120Hz 已经非常相近。

继续增大 p 到 30，



但此时已经出现了过拟合状况，Levinson 不能分辨出两条谱线。

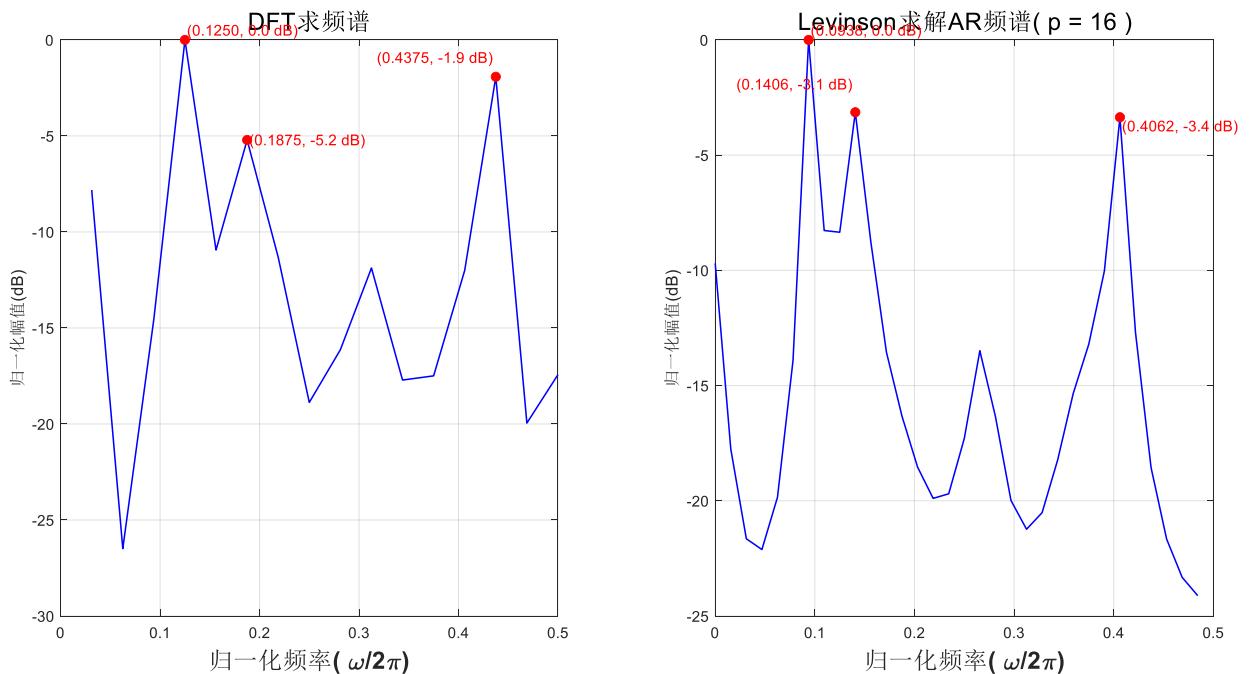
再让 $p=16$, 但设前两个信号频率相近的信号幅度 A1、A2 分别为 1 和 2。



这是 Levinson 法的频谱只显示了幅度为 2 的 120Hz 的信号, 而没有另一个幅度为 1 的 100Hz 信号, 功率谱上产生了“掩蔽效应”, 掩盖邻近的小幅度频率分量, 无法分辨出两条谱线。

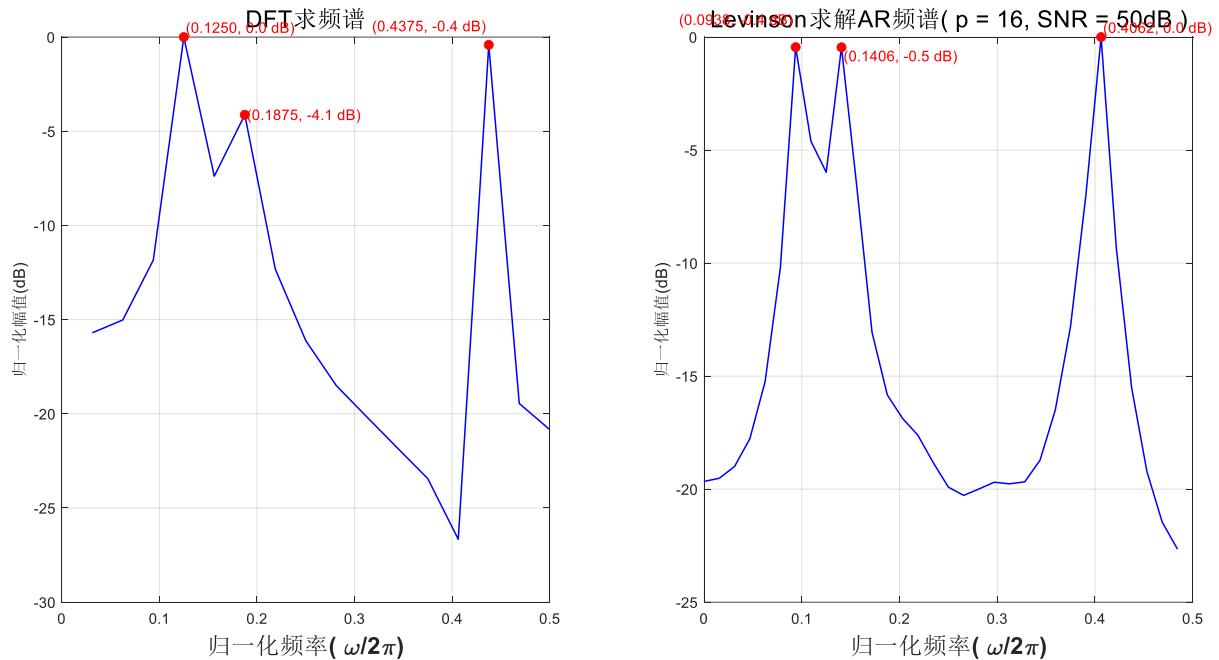
由此, 我们可以得出结论, Levinson 通常能超越 DFT 的格点限制, 能够分辨邻近频率, 估计效果较好, 其分辨率取决于模型阶数 p 、样本长度 N 、SNR 以及谱峰的相对幅度与相位。

再把 A_1 、 A_2 设为 1, 讲两个临近信号的频率差加大, 变为 $f_1 = 100Hz$, $f_2 = 140Hz$, $f_3 = 405Hz$, 信噪比 $SNR=10$, 阶数 $p=16$ 。



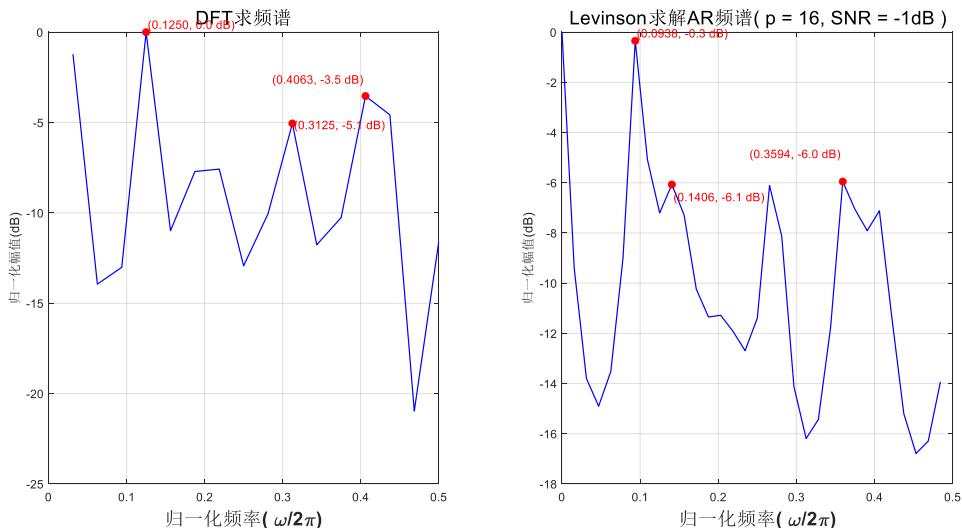
可以看到，此时 DFT 可以分辨出两条谱线，但是和真实频率差距很大，说明其受到点数制约，无法精确地分辨出邻近频率分辨率的信号。而 Levinson 法则轻松地分辨出了三个信号的谱线。

将 SNR 改为 50dB



可以看到旁瓣明显减少，三个主峰更加明显了，估计频率的结果也更接近真实值了。

再将 SNR 改为-1dB



在这种高噪声弱信号的环境下，DFT 已经无法分辨出两条谱线，但 Levinson 法依然能分辨，尽管两个峰确实一高一低有些变形，但估计的频率确实是准确的。这是因为在噪声主导时，AR 方法通过参数化降低估计自由度，方差更小，因此对弱信号的检测更有利。

```
% ----- 定义要绘制的 SNR 列表（长度自适应） -----
SNR_list = [-1,0, 5, 10, 15, 20]; % 你可以任意修改为行向量，长度会自动适配

% ----- 为每个 SNR 计算并绘制 AR 频谱（替换原 subplot(122) 部分） -----
figure; % 新建图窗用于 AR 频谱比较
colors = lines(length(SNR_list)); % 自动生成足够的颜色
hold on;
legend_entries = cell(1, length(SNR_list));

for iS = 1:length(SNR_list)
    curSNR = SNR_list(iS);
    % 重新生成带噪信号（保持原始无噪信号 signal 不变）
    signal_noisy_cur = awgn(signal, curSNR);

    % Levinson 求解 AR 系数与最小预测误差功率
    [ar_coeff_cur, min_pred_error_power_cur] = Levinson(signal_noisy_cur, p);

    % 计算频率响应与功率谱（使用与原来相同的点数 signal_len）
    [freq_response_cur, freq_vec_cur] = freqz(sqrt(min_pred_error_power_cur),
[1, ar_coeff_cur], signal_len);
    power_spectrum_ar_cur = abs(freq_response_cur).^2;
    power_spectrum_ar_cur = power_spectrum_ar_cur /
max(power_spectrum_ar_cur); % 归一化
    power_spectrum_ar_db_cur = 10*log10(power_spectrum_ar_cur);

    % 归一化频率轴
    norm_freq_ar_cur = freq_vec_cur/(2*pi);

    % 绘制曲线
    plot(norm_freq_ar_cur, power_spectrum_ar_db_cur, 'Color', colors(iS,:),
'LineWidth', 1.2);
    legend_entries{iS} = sprintf('SNR=%d dB', curSNR);
end
```

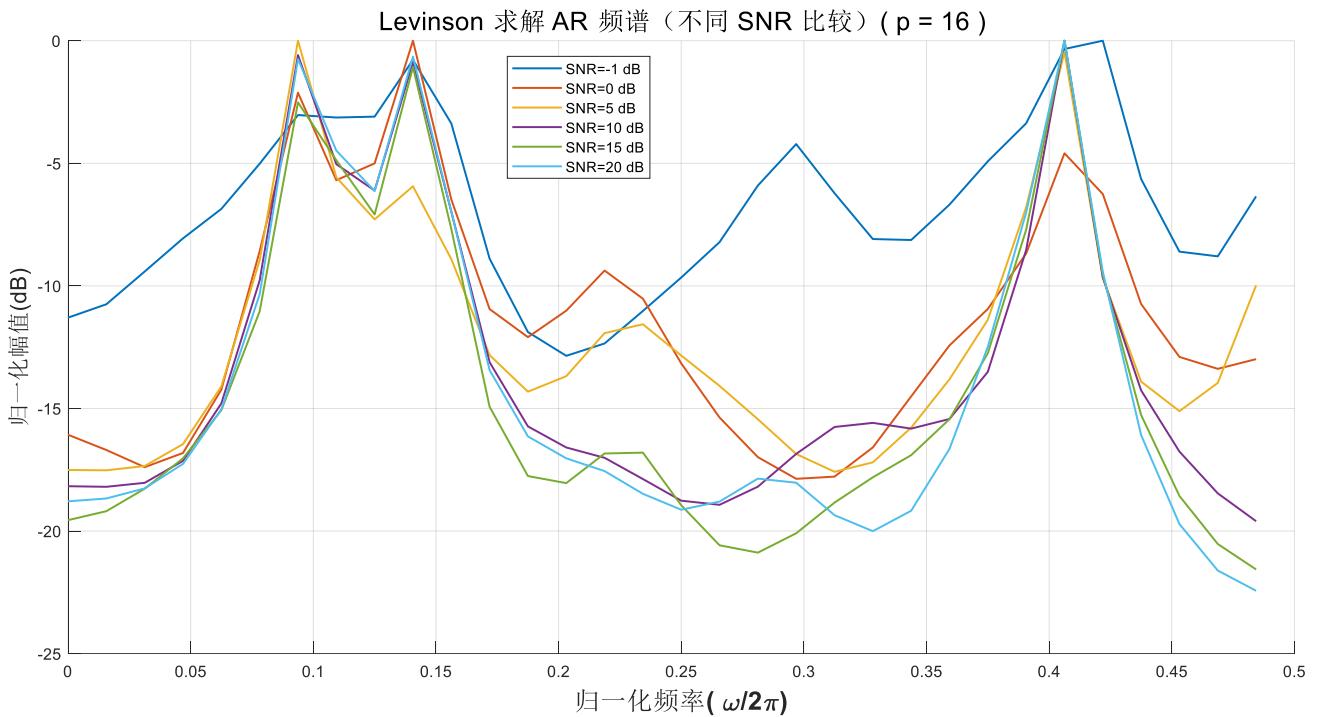
```

% 图形美化
xlabel('归一化频率(\omega/2\pi)', 'FontWeight', 'bold', 'FontSize', 16);
ylabel('归一化幅值(dB)', 'FontSize', 14);
title_str = sprintf('Levinson 求解 AR 频谱 (不同 SNR 比较) ( p = %d )', p);
title(title_str, 'FontSize', 16);

grid on;
legend(legend_entries, 'Location', 'best');
hold off;

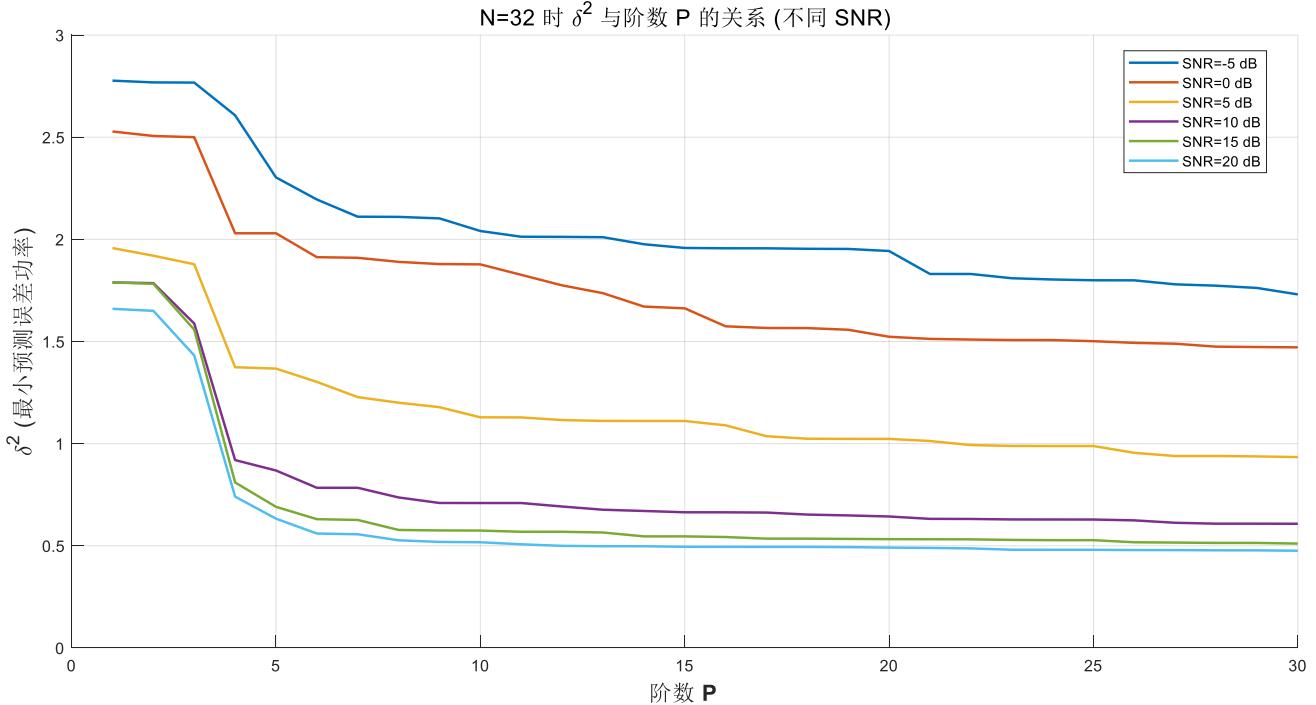
```

将各个 SNR 的 Levinson 频谱画在一起，看趋势。



可以明显看到 SNR 越大，谱线越清晰尖锐，临近频率的谱线之间分隔也更大，旁瓣也更少。

接下来, 我们来看 δ^2 的变化趋势(输入不同 SNR, 多组 δ^2 曲线), 我也把不同 SNR 下的 δ^2 曲线画在了一张图里, 方便比较。



在固定样本长度 $N = 32$ 且用 Levinson 求解自回归模型的条件下, 随着模型阶数 P 增大, 最小预测误差功率 δ^2 在所有 SNR 下都呈下降趋势, 但下降并非匀速——在较小的阶数区间(例如从 1 到几阶) δ^2 下降最快, 随后曲线逐步趋于平缓, 说明增加阶数在初期能显著改善拟合精度, 但到了一定阶数后边际收益递减。

不同 SNR 曲线之间的差异也很明显: 信噪比越高, 整体 δ^2 越小, 曲线越靠下且在低阶时下降更陡峭; 信噪比较低时(噪声占优), δ^2 不仅起点高, 而且随着 P 增加的改善幅度受限, 最终会被噪声限制住, 难以继续显著下降。

阶数 P 较小时($P < 5$): δ^2 快速下降是因为此时 AR 阶数不足, 无法充分拟合“多频纯净信号”的频率结构, 预测误差 $e(n)$ 中不仅包含噪声, 还包含未被模型拟合的信号成分, 因此 δ^2 较大。

随着 P 增加, 模型能拟合更多信号的频率特征, 预测误差中“信号成分占比”快速降低, 因此 δ^2 快速下降。

阶数 P 足够大时($P > 5$): δ^2 趋于平稳, 当 P 足够拟合信号的所有频率成分, 此时预测误差 $e(n)$ 几乎只由加性噪声构成, 因此 δ^2 不再随 P 增加而明显减小。

实验收获和体会

这次实验让我对参数化谱估计和传统 DFT 之间的差异有了更直观的认识：在 MATLAB 中实现 Levinson 递推、用 $N = 32$ 、 $f_s = 1000$ 的信号（100、120、405Hz）做对比时，我看到 DFT 因点数限制 ($\Delta f = 1000/32 = 31.25\text{Hz}$) 无法分辨相差 20Hz 的两条谱线，而 AR/Levinson 在合适阶数下能突破这个格点限制；具体地，当 $p = 8$ 或 $p = 12$ 时两法都分不清 100 和 120Hz，但把 p 提高到 16（即 $N/2$ ）后 Levinson 能把两条谱线分开，继续把 p 增到 30 又出现过拟合，说明阶数过大反而有害；另外我也观察到 SNR 的影响：在高 SNR（例如 50dB）下谱峰更尖锐、旁瓣更小，而在低 SNR（如 -1dB）时 AR 方法仍能较好地检测到弱信号，这是因为参数化模型降低了估计自由度；还有一个实际体会是“掩蔽效应”，当两个临近分量幅度不一致（1 与 2）时，Levinson 频谱可能只显示较强的那条，说明幅度差会影响可分辨性。通过画不同 SNR 下的 δ^2 随阶数 P 变化曲线，我也验证了一个常见规律： δ^2 随 P 下降但边际收益递减（在 $P < 5$ 时下降最快，之后趋于平缓），这让我在今后选阶数时会更注重在分辨率与过拟合之间权衡，而不是一味追求更高阶数。