考虑系统

$$\dot{x}(t) = \sum_{i=1}^{n} A_i x(t - \tau_i)$$

1 稳定性判别代码

表 1: panding.m

函数原型	[W,pzend]=panding(Ai,taui,n)
功能	判断以 A_i 为系数,τ_i 为延时量的系统是否稳定
输入项	Ai 为元胞数组形式;τ_i 为数组形式
输出项	W 不稳定根个数,pzend 为 $P(z)$ 的轨迹
算法	通过离散半圆, 计算 $P(z)$. 用 matlab 的 angle 函数计算幅角. 并将幅角的变化量连续化.

注：

- 此程序最为需要注意的是，在 $P(z)$ 比较接近 0 时, 幅角的变化快, 从而导致结果不准确. 这涉及参考文章中提到的 tolerance δ. 本程序在 0 附近对轨迹进行了加细.
- $Ai\{N+1\}$ 是 A_0;

参考文章 <DELAY-DEPENDENT STABILITY OF RUNGE-KUTTA METHODS FOR LINEAR NEUTRAL SYSTEMS WITH MULTIPLE DELAYS>

2 稳定性判别代码 =v2

表 2: numWv2.m

函数原型	[W,tnew,xnew] = numWv2(Ai,taui,N)
功能	判断以 A_i 为系数,τ_i 为延时量的系统是否稳定
输入项	Ai 为元胞数组形式;τ_i 为数组形式
输出项	W 为不稳定根个数, tnew(i)=θ_i 为 $Im(P(s(\theta_i))) = 0$ xnew(i)= $Im(P(s(\theta_i))), Im(P(s(\theta_i))) = 0$
算法	通过求根程序计算 $Im(P(s(\theta_i))) = 0$, $Im(P(s(\theta_i + a)) - P(s(\theta_i - a)))$ 的符号判断穿越 x 轴的方向 计算轨迹与 x 轴正半轴穿越的总和得到 W

注:

- 对高阶系统，本程序求根时的误差较大，需要具体分析. 高阶系统的 $Im(P(s(\theta_i)))$ 部分图像与 x 轴非常接近求根误差较大.