HW3 单参数曲线拟合

一、 算法分析与实现

由于本次作业要求对平面 2 维曲线的采样点进行拟合，而已知平面 2 维曲线的本征维度为 1, 故作业中采用单参数方程组$\begin{cases} x = x(t) \\ y = y(t) \end{cases}$对各采样点进行曲线拟合，其中$x(t), y(t)$共享幂基函数$1, t, t^2, \dots, t^m, t\epsilon[0,1]$，误差度量采用平方误差$E = \sum_{i=1}^{n} \left\| \begin{pmatrix} x(t_i) \\ y(t_i) \end{pmatrix} - \begin{pmatrix} x_i \\ y_i \end{pmatrix} \right\|^2$，通过最小二乘法计算得到拟合函数$x(t), y(t)$的加权参数值。

本次作业采用了 4 种点列的参数化方法：

1. Equidistant/Uniform parameterization

   Equidistant/均匀参数化是最简单的一种参数化方法，它根据采样点个数将t_i均匀分布在区间$[0,1]$上，公式如下：

   $$t_1 = 0,\ t_i = \frac{i}{n}, 2 \le i \le n-1,\ t_n = 1$$

2. Chordal parameterization

   鉴于曲线采样点的分布往往不均匀、无规律性，因此 Chordal 参数化根据相邻采样点$D_k(x_k, y_k), D_{k-1}(x_{k-1}, y_{k-1})$之间的距离(chord-length)分配各$t_i$的值，公式如下：

   $$t_1 = 0,\ t_i = \frac{\sum_{k=1}^{i} \|D_k - D_{k-1}\|}{\sum_{k=1}^{n} \|D_k - D_{k-1}\|}, 2 \le i \le n-1,\ t_n = 1$$

3. Centripetal parameterization

   Centripetal 向心参数化可看作 Chordal 的改进版本，它对 chord-length 用$e = \frac{1}{2}$次幂作了进一步修正，公式如下：

$$t_1 = 0,\ t_i = \frac{\sum_{k=1}^{i} \|D_k - D_{k-1}\|^e}{\sum_{k=1}^{n} \|D_k - D_{k-1}\|^e}, 2 \le i \le n-1,\ t_n = 1$$

4. Foley parameterization

Foley-Nielson 参数化是 Thomas A. Foley 等人于 1989 年在论文 Knot Selection for Parametric Spline Interpolation 中提出的参数化算法，它使用 Nielson 尺度来测量相邻采样点之间的距离(Neilson distance)，从而对曲线的旋转、平移和缩放等具有仿射不变性，其公式如下：

$$t_1 = 0,\ t_i = \frac{\sum_{k=1}^{i} Neilson(D_k, D_{k-1})}{\sum_{k=1}^{n} Neilson(D_k, D_{k-1})}, 2 \le i \le n-1,\ t_n = 1$$

其中，

$Neilson(D_2, D_1) = \|D_2 - D_1\| \left(1 + \frac{3}{2} \frac{\hat{\alpha}_2 \|D_2 - D_1\|}{\|D_2 - D_1\| + \|D_3 - D_2\|}\right)$,

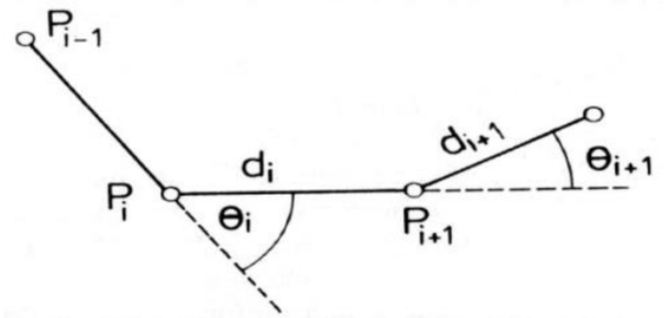
$Neilson(D_k, D_{k-1}) = \|D_k - D_{k-1}\| \left(1 + \frac{3}{2} \frac{\hat{\alpha}_{k-1} \|D_{k-1} - D_{k-2}\|}{\|D_{k-1} - D_{k-2}\| + \|D_k - D_{k-1}\|} + \frac{3}{2} \frac{\hat{\alpha}_k \|D_k - D_{k-1}\|}{\|D_k - D_{k-1}\| + \|D_{k+1} - D_k\|}\right), k = 3, \dots, n-1$,

$Neilson(D_n, D_{n-1}) = \|D_n - D_{n-1}\| \left(1 + \frac{3}{2} \frac{\hat{\alpha}_{n-1} \|D_{n-1} - D_{n-2}\|}{\|D_{n-1} - D_{n-2}\| + \|D_n - D_{n-1}\|}\right)$,

$\hat{\alpha}_k = \min\left(\alpha_k, \frac{\pi}{2}\right), k = 2,3, \dots, n-1$,

$\alpha_k = \pi - \arccos\left(\frac{\|D_{k+1} - D_k\|^2 + \|D_k - D_{k-1}\|^2 - \|D_{k+1} - D_{k-1}\|^2}{2\|D_{k+1} - D_k\| \|D_k - D_{k-1}\|}\right), k = 2,3, \dots, n-1.$

分析图示如下（图中的$\angle\theta_i$即$\angle\alpha_i$，点P_i即点D_i）：



显然，$\angle(\pi - \theta_i)$是在$\Delta P_{i+1} P_i P_{i-1}$中应用余弦定理求得。

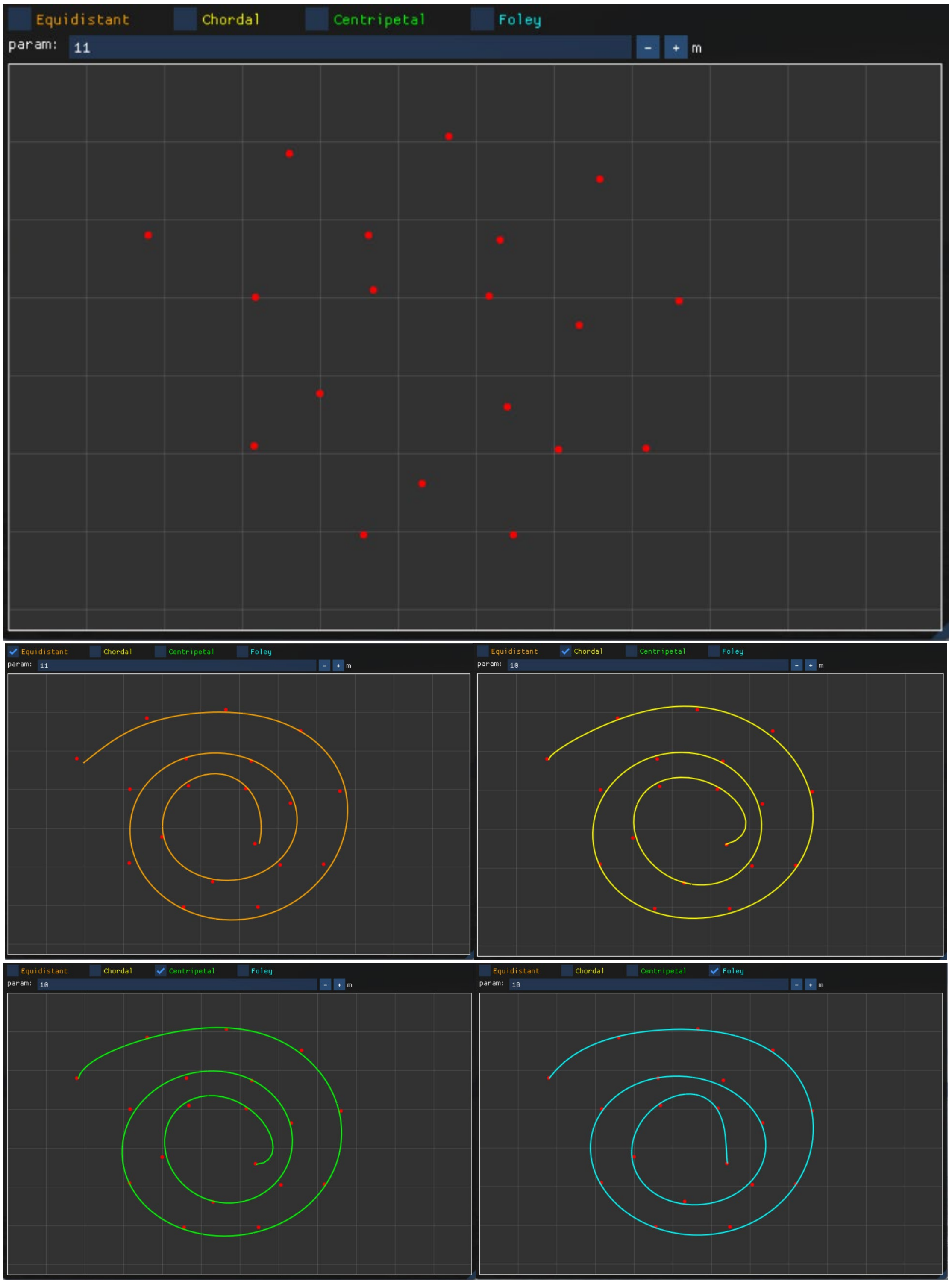
据此对以上 4 种算法作代码实现，截取如下：

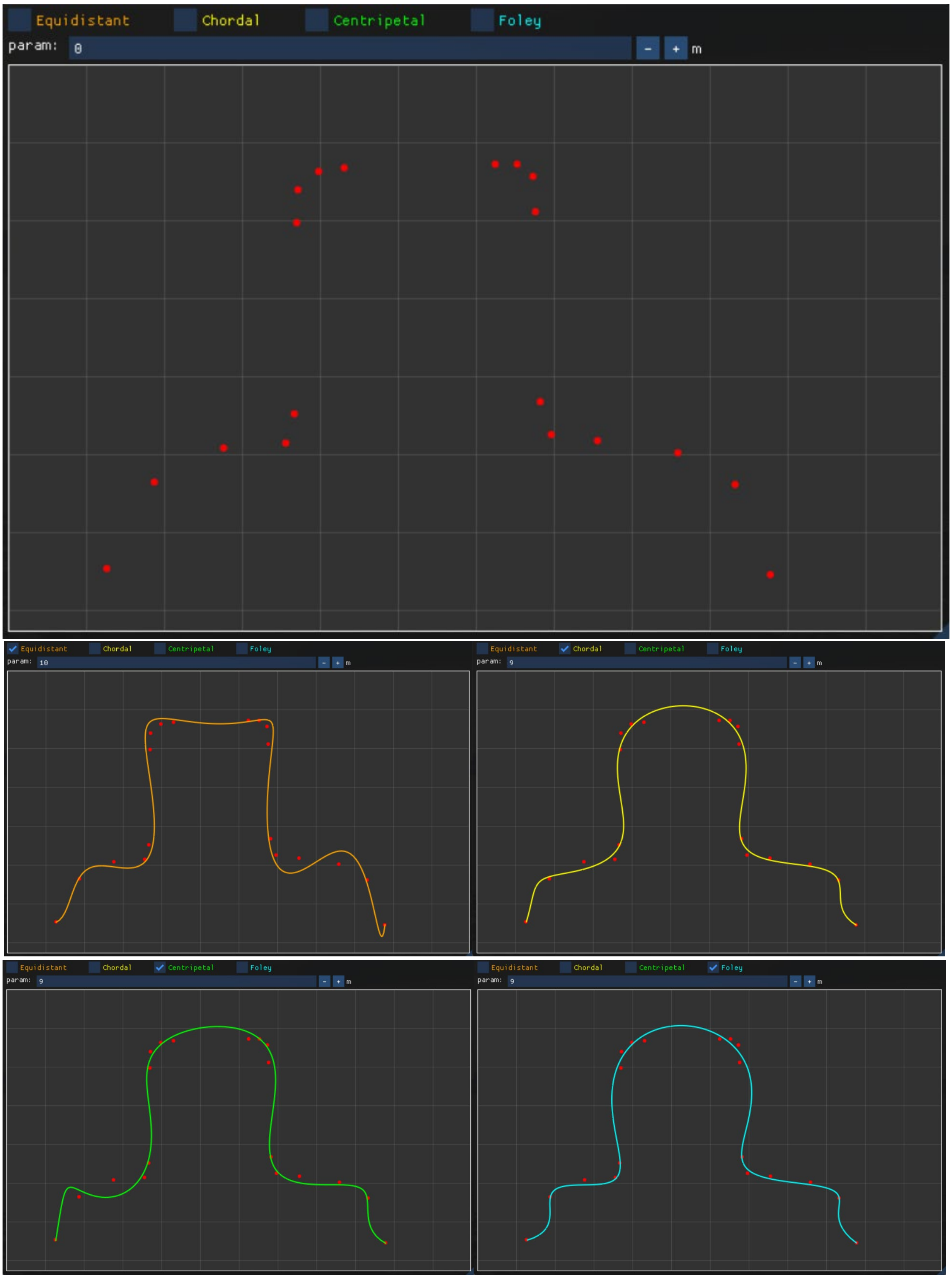
```
for (int i = 0; i < coords_size - 1; i++) {
    chordalDistance[i] = CalDistance(coords[i + 1], coords[i]);
    centripetalDistance[i] = sqrt(chordalDistance[i]);

    sumChordalDistance += chordalDistance[i];
    sumCentripetalDistance += centripetalDistance[i];
}

for (int i = 0; i < coords_size - 1; i++) {
    if (i < coords_size - 2) {
        alpha[i] = acos((pow(chordalDistance[i], 2) + pow(chordalDistance[i + 1], 2) - pow(CalDistance(coords[i + 2], coords[i]), 2)) / (2.0 * chordalDistance[i] * chordalDistance[i + 1])); // Law of cosines
        alpha[i] = min(M_PI - alpha[i], M_PI_2);
    }

    if (i == 0) {
        foleyDistance[i] = chordalDistance[i] * (1.0 + 1.5 * alpha[i] * chordalDistance[i] / (chordalDistance[i] + chordalDistance[i + 1]));
    }
    else if (i == coords_size - 2) {
        foleyDistance[i] = chordalDistance[i] * (1.0 + 1.5 * alpha[i - 1] * chordalDistance[i - 1] / (chordalDistance[i - 1] + chordalDistance[i]));
    }
    else {
        foleyDistance[i] = chordalDistance[i] * (1.0 + 1.5 * alpha[i - 1] * chordalDistance[i - 1] / (chordalDistance[i - 1] + chordalDistance[i]) + 1.5 * alpha[i] * chordalDistance[i] / (chordalDistance[i] + chordalDistance[i + 1]));
    }

    sumFoleyDistance += foleyDistance[i];
}
```

二、 实验分析与结论

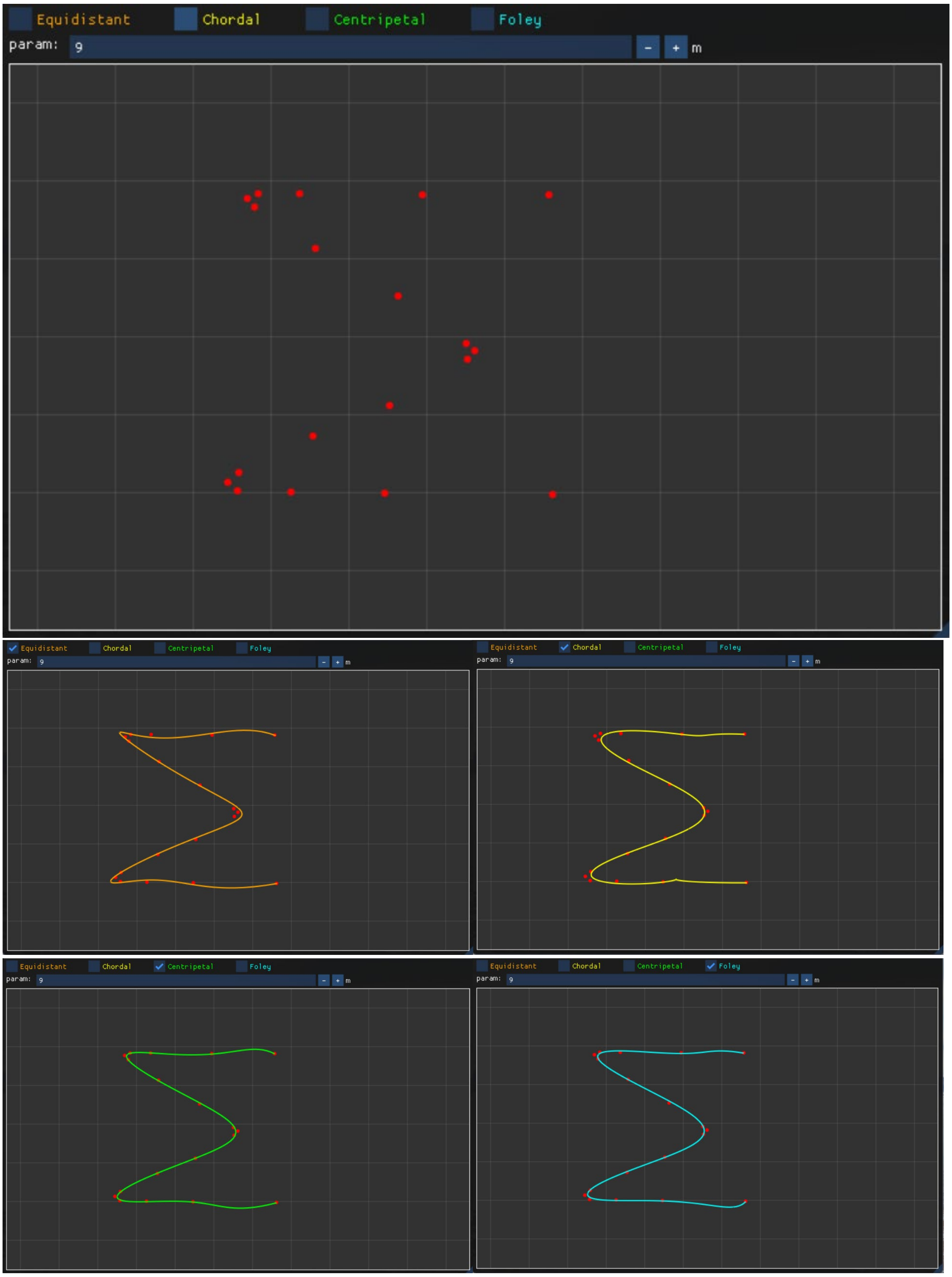
1. 实验结果

以下各图示出了上述 4 种算法对 3 组不同曲线采样点的拟合结果（幂基函数的最高次数 m 都经过一定优化调整）：





Equidistant
Chordal
Centripetal
Foley
param: 0
m
Equidistant
Chordal
Centripetal
Foley
param: 10
m
Equidistant
Chordal
Centripetal
Foley
param: 9
m
Equidistant
Chordal
Centripetal
Foley
param: 9
m
Equidistant
Chordal
Centripetal
Foley
param: 9
m


Equidistant
Chordal
Centripetal
Foley
param:
9
-
+
m

2. 结论

a) 对于采样均匀、规则的曲线，与另外 3 种算法相比，Equidistant 均匀参数化作为一种简单而有效的算法，既节约了计算成本，又节省了存储空间；

b) 对于采样不均匀、不规则且曲率较大的曲线，Chordal, Centripetal 及 Foley 参数化根据相邻采样点之间的距离度量合理分配各t_i的值，使拟合曲线在曲率、采样率较大处保持平滑过渡，效果明显优于均匀参数化算法；

c) 后 3 种算法比较而言，Foley-Nielson 参数化以更大的计算和存储成本为代价，换取了更为平滑、精确的拟合结果。