

结论先给出：

这两段代码本质上都来自中心差分法，但第二段在索引和力学平衡式上是错误的，物理意义不一致。

下面严格从公式—离散—代码三层对比说明。

一、第一段代码在做什么（标准中心差分法，正确）

1. 连续方程

$$m\ddot{u} + c\dot{u} + ku = F(t)$$

2. 中心差分离散

$$\dot{u}_i \approx \frac{u_{i+1} - u_{i-1}}{2\Delta t}, \quad \ddot{u}_i \approx \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta t^2}$$

代入并整理为显式格式：

$$\left(\frac{m}{\Delta t^2} + \frac{c}{2\Delta t}\right)u_{i+1} = F_i - \left(\frac{m}{\Delta t^2} - \frac{c}{2\Delta t}\right)u_{i-1} - \left(k - \frac{2m}{\Delta t^2}\right)u_i$$

3. 第一段代码逐项对应

```
eff_k = m / (dt**2) + c / (2*dt)      # u_{i+1} 的等效刚度
term_a = m / (dt**2) - c / (2*dt)      # 乘 u_{i-1}
term_b = k - 2*m / (dt**2)              # 乘 u_i
```

这是系数预处理阶段，完全正确。

在正确用法中，后续应当是：

```
u_next = (F[i] - term_a*u_prev - term_b*u_curr) / eff_k
```

二、第二段代码在做什么（形式错误）

第二段代码：

```
for i in range(1, N + 1):
    term1 = F[i - 1]
    term2 = (m / (dt ** 2) - c / (2 * dt)) * u_vals[i]
    term3 = (k - 2 * m / (dt ** 2)) * u_vals[i + 1]
```

问题逐条说明

✖ 1. 力项时间层错位

```
term1 = F[i - 1]
```

中心差分平衡方程使用的是 F_i , 不是 F_{i-1} 。

✖ 2. 变量角色错位

```
term2 = (m/(dt**2)-c/(2*dt)) * u_vals[i]
```

这个系数 应当乘 u_{i-1} , 而不是乘 u_i 。

✖ 3. 最严重错误：使用了未来位移

```
term3 = (k - 2*m/(dt**2)) * u_vals[i + 1]
```

但中心差分法是 显式算法：

- u_{i+1} 是未知量
- 不能出现在右端

你这里等于写成：

$$\text{右端} = \dots + (k - 2m/\Delta t^2)u_{i+1}$$

这在数学上已经破坏显式格式，在物理上也没有任何意义。

三、两段代码的本质区别总结（核心）

对比项	第一段	第二段
算法来源	标准中心差分法	非一致公式
是否显式	是	被破坏
是否使用未来量	否	是（致命）
力项时间层	正确 F_i	错误 F_{i-1}
位移索引	u_{i-1}, u_i	混乱
数值稳定性	可判定	无法判定

四、正确的中心差分代码模板（给你一个对照）

```
u = np.zeros(N+1)
u[0] = u0
u_prev = u0 - dt*v0 + 0.5*a0*dt**2

eff_k = m/dt**2 + c/(2*dt)
term_a = m/dt**2 - c/(2*dt)
term_b = k - 2*m/dt**2

for i in range(N):
    u_next = (F[i] - term_a*u_prev - term_b*u[i]) / eff_k
    u_prev = u[i]
    u[i+1] = u_next
```

五、一句话定性判断

第一段是标准、正确、可用于工程分析的中心差分系数定义；

第二段在索引、物理意义和数值结构上全部错误，不能使用。

如果你愿意，下一步我可以帮你把第二段改写成严格正确的中心差分实现，逐行对齐你现在的代码风格。