一元三次方程

Hanford

2016年04月06日

目　录

第 1 章　一元三次方程

1　卡丹公式

历史上第一个被明确记载的一元三次方程求根公式就是卡丹公式。本节将说明卡丹公式的计算方法，并对其进行验证。

1.1　计算方法

一元三次方程 $ax^3+bx^2+cx+d=0\left(|a|\neq 0\right)$ 可化为

$$\left(x+\frac{b}{3a}\right)^3+\frac{3ac-b^2}{3a^2}\left(x+\frac{b}{3a}\right)+\frac{27a^2d-9abc+2b^3}{27a^3}=0 \tag{1}$$

令

$$\begin{cases} y=x+\dfrac{b}{3a} \\ p=\dfrac{3ac-b^2}{3a^2} \\ q=\dfrac{27a^2d-9abc+2b^3}{27a^3} \end{cases} \tag{2}$$

则方程(1)可转化为下式

$$y^3+py+q=0 \tag{3}$$

求解上式中 y 的计算步骤如下：

$$P=-\frac{p}{3} \tag{4}$$

$$Q=-\frac{q}{2} \tag{5}$$

$$\Delta=\left(\frac{q}{2}\right)^2+\left(\frac{p}{3}\right)^3=Q^2-P^3 \tag{6}$$

$$D=\sqrt{\Delta}=\sqrt{Q^2-P^3} \tag{7}$$

$$u=\sqrt[3]{-\frac{q}{2}+\sqrt{\Delta}}=\sqrt[3]{Q+D} \tag{8}$$

$$v=\sqrt[3]{-\frac{q}{2}-\sqrt{\Delta}}=\sqrt[3]{Q-D} \tag{9}$$

$$\omega=-\frac{1}{2}+\frac{\sqrt{3}}{2}i \Rightarrow \omega^n=\begin{cases}1 & n=0,3,6,\cdots,3m,\cdots \\ \omega & n=1,4,7,\cdots,3m+1,\cdots \\ \bar{\omega}=-\frac{1}{2}-\frac{\sqrt{3}}{2}i & n=2,5,8,\cdots,3m+2,\cdots\end{cases} \tag{10}$$

方程(3)的三个根为：

$$\begin{cases}y_1=\omega^n u+v \\ y_2=\omega^{n+1}u+\omega^2 v \\ y_3=\omega^{n+2}u+\omega v\end{cases} \quad \text{或} \quad \begin{cases}y_1=u+\omega^{n+3}v \\ y_2=\omega u+\omega^{n+2}v \\ y_3=\omega^2 u+\omega^{n+1}v\end{cases} \tag{11}$$

上式中，n 有三种取值 0,1,2，且满足下式

$$\omega^n uv=-p/3=P \tag{12}$$

方程(1)的三个根为：

$$\begin{cases}x_1=y_1-b/(3a) \\ x_2=y_2-b/(3a) \\ x_3=y_3-b/(3a)\end{cases} \tag{13}$$

1.2　验证

假定方程(3)的三个根为$\alpha+\beta,\omega\alpha+\omega^2\beta,\omega^2\alpha+\omega\beta$，则其等价于下式

$$\left[y-(\alpha+\beta)\right]\left[y-\left(\omega\alpha+\omega^2\beta\right)\right]\left[y-\left(\omega^2\alpha+\omega\beta\right)\right]=0 \tag{14}$$

展开上式，可得：

$$y^3-3\alpha\beta y-\left(\alpha^3+\beta^3\right)=0 \tag{15}$$

公式(11)中

$$\begin{cases}\alpha=\omega^n u\\ \beta=v\end{cases} \quad \text{或} \quad \begin{cases}\alpha=u\\ \beta=\omega^n v\end{cases} \tag{16}$$

现在验算方程(15)的系数

$$\begin{aligned}-3\alpha\beta&=-3\omega^n uv=-3\omega^n\sqrt[3]{Q+\sqrt{\Delta}}\sqrt[3]{Q-\sqrt{\Delta}}=-3\omega^n\sqrt[3]{Q^2-\Delta}\\&=-3\omega^n\sqrt[3]{Q^2-\left(Q^2-P^3\right)}=-3\omega^n\sqrt[3]{P^3}=\omega^n\sqrt[3]{\left(-3P\right)^3}=\omega^n\sqrt[3]{\left(p\right)^3}=p\end{aligned} \tag{17}$$

$$\begin{aligned}-\left(\alpha^3+\beta^3\right)&=\begin{cases}-\left(\omega^n u\right)^3-v^3\\ -u^3-\left(\omega^n v\right)^3\end{cases}=-u^3-v^3\\&=-\left(\sqrt[3]{-\frac{q}{2}+\sqrt{\Delta}}\right)^3-\left(\sqrt[3]{-\frac{q}{2}-\sqrt{\Delta}}\right)^3=q\end{aligned} \tag{18}$$

可见：公式(11)确实是方程(3)的三个根。u,v满足下式：

$$\begin{cases}\omega^n uv=-p/3=P\\ u^3+v^3=-q=2Q\end{cases} \tag{19}$$

注意：公式(17)中$\sqrt[3]{\left(p\right)^3}$应有三个结果，但是它只能返回其中一个。$\omega^n$的作用就是调整开立方结果（复数）的辐角，使其增加$120°n$。

2　求根公式一

将上一节的内容整理一下，即可得到一元三次方程$ax^3+bx^2+cx+d=0$的求根公式。

根据公式(2)可得

$$P=-\frac{p}{3}=\frac{b^2-3ac}{9a^2} \tag{20}$$

$$Q=-\frac{q}{2}=\frac{9abc-27a^2d-2b^3}{54a^3} \tag{21}$$

根据公式(7)可知

$$D=\sqrt{\Delta}=\sqrt{Q^2-P^3}=\sqrt{\left(\frac{9abc-27a^2d-2b^3}{54a^3}\right)^2+\left(\frac{3ac-b^2}{9a^2}\right)^3}$$
$$=\frac{\sqrt{3\left(4ac^3-b^2c^2-18abcd+27a^2d^2+4b^3d\right)}}{18a^2} \tag{22}$$

$$u=\begin{cases}\sqrt[3]{Q+D} & 当\left|Q+D\right|\geq\left|Q-D\right| \\ \sqrt[3]{Q-D} & 当\left|Q+D\right|<\left|Q-D\right|\end{cases} \tag{23}$$

$$v=\begin{cases}\dfrac{P}{u}=\dfrac{b^2-3ac}{9a^2u} & 当\left|u\right|\neq 0 \\ 0 & 当\left|u\right|=0\end{cases} \tag{24}$$

$$\omega=-\frac{1}{2}+\frac{\sqrt{3}}{2}i\Rightarrow\omega^n=\begin{cases}1 & n=0,3,6,\cdots,3m,\cdots \\ \omega & n=1,4,7,\cdots,3m+1,\cdots \\ \overline{\omega}=-\dfrac{1}{2}-\dfrac{\sqrt{3}}{2}i & n=2,5,8,\cdots,3m+2,\cdots\end{cases} \tag{25}$$

一元三次方程的三个根为

$$\begin{cases}x_1=u+v-\dfrac{b}{3a} \\ x_2=\omega u+\omega^2v-\dfrac{b}{3a} \\ x_3=\omega^2u+\omega v-\dfrac{b}{3a}\end{cases} \tag{26}$$

上式写成通解就是

$$x_n=\omega^{n-1}u+\omega^{4-n}v-\frac{b}{3a}\qquad\left(n=1,2,3\right) \tag{27}$$

注意：公式中的u和v，一个是$\sqrt[3]{Q+D}$另一个是$\sqrt[3]{Q-D}$且满足下式：

$$\begin{cases}uv=P \\ u^3+v^3=2Q\end{cases} \tag{28}$$

复数开立方有三个结果，也就是说$\left(u,v\right)$共有 9 组，其中只有 3 组符合

$uv=P$。公式(26)中的$\left(u,v\right)$、$\left(\omega u,\omega^2 v\right)$、$\left(\omega^2 u,\omega v\right)$就是符合要求的 3 组。

3　求根公式二

$$P_6=\left(6a\right)^2 P=4\left(b^2-3ac\right) \tag{29}$$

$$Q_6=\left(6a\right)^3 Q=4\left(9abc-27a^2d-2b^3\right) \tag{30}$$

$$\begin{aligned}D_6&=\left(6a\right)^3 D=\sqrt{Q_6^2-P_6^3}\\&=12a\sqrt{3\left(4ac^3-b^2c^2-18abcd+27a^2d^2+4b^3d\right)}\end{aligned} \tag{31}$$

$$u_6=\left(6a\right)u=\begin{cases}\sqrt[3]{Q_6+D_6} & 当\left|Q_6+D_6\right|\ge\left|Q_6-D_6\right|\\ \sqrt[3]{Q_6-D_6} & 当\left|Q_6+D_6\right|<\left|Q_6-D_6\right|\end{cases} \tag{32}$$

$$v_6=\left(6a\right)v=\begin{cases}\dfrac{P_6}{u_6}=\dfrac{4\left(b^2-3ac\right)}{u_6} & 当\left|u_6\right|\ne 0\\ 0 & 当\left|u_6\right|=0\end{cases} \tag{33}$$

$$x_n=\frac{\omega^{n-1}u_6+\omega^{4-n}v_6-2b}{6a}\qquad\left(n=1,2,3\right) \tag{34}$$

4　VC++代码

求根代码如下

```
#include <math.h>
#include <complex>
/*****************************************************************\
对复数 x 开 n 次方
\*****************************************************************/
std::complex<double> sqrtn(const std::complex<double>&x,double n)
{
	double	r	=	_hypot(x.real(),x.imag());	//模
	if(r > 0.0)
	{//模不为零时，开方
		double	a	=	atan2(x.imag(),x.real());	//辐角
```

```
        n     =     1.0 / n;
        r     =     pow(r,n);
        a    *=     n;
        return std::complex<double>(r * cos(a),r * sin(a));
    }
    return std::complex<double>();    //模为零时，返回零
}
/*****************************************************************\
求解一元三次方程 a*x^3 + b*x^2 + c*x + d = 0
\*****************************************************************/
void CubicEquation(std::complex<double> x[3]
                  ,std::complex<double> a
                  ,std::complex<double> b
                  ,std::complex<double> c
                  ,std::complex<double> d)
{
    a     =     1.0 / a;
    b    *=     a;
    c    *=     a;
    d    *=     a;
    std::complex<double>    u =    ((9.0 * c - 2.0 * b * b) * b - 27.0 * d) / 54.0;
    std::complex<double>    v =    3.0 * ((4.0 * c - b * b) * c * c
                              + ((4.0 * b * b - 18.0 * c) * b + 27.0 * d) * d);
    v     =     sqrtn(v,2.0) / 18.0;
    std::complex<double>       m     =     u + v;
    std::complex<double>       n     =     u - v;
    if(n.real() * n.real() + n.imag() * n.imag()    >
      m.real() * m.real() + m.imag() * m.imag())
    {
        m     =     n;
    }
    a     =     b/-3.0;
    if(m.real() * m.real() + m.imag() * m.imag() > 0.0)
    {
        m           =     sqrtn(m,3.0);
        n           =     (b * b - 3.0 * c) / (9.0 * m);
  std::complex<double>o1(-0.5,+0.866025403784438646763723170752 94);
  std::complex<double>o2(-0.5,-0.866025403784438646763723170752 94);
        x[0]      =     m + n + a;
        x[1]      =     o1 * m + o2 * n + a;
        x[2]      =     o2 * m + o1 * n + a;
    }
    else
    {
        x[0]      =
```

```
            x[1]    =
            x[2]    =   a;
        }
}
```

验证代码如下

```
std::complex<double>    x[3];
std::complex<double>    x1(1.0,0.0);    //随便填
std::complex<double>    x2(2.0,0.0);    //随便填
std::complex<double>    x3(3.0,0.0);    //随便填
std::complex<double>    a (4.5,0.0);    //随便填(不为零即可)
std::complex<double>    b   =   a * (-x1-x2-x3);
std::complex<double>    c   =   a * (x2 * x3 + x1 * x3 + x1 * x2);
std::complex<double>    d   =   a * (-x1 * x2 * x3);
CubicEquation(x,a,b,c,d);
```