ECC算法

# 平行线相交

以前我们认为平行线不相交，但是“平行线，永不相交”只是假设，并没有数学证明。我们也可以假设平行线在无穷远处相交了，就如下图所示。

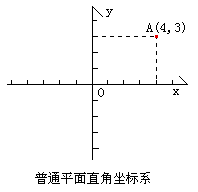


记为无穷远点，它具有如下一些特性：

1. 直线L上的无穷远点只能有一个。 （从定义可直接得出）
2. 平面上一组相互平行的直线有公共的无穷远点。 （从定义可直接得出）
3. 平面上任何相交的两直线L1,L2有不同的无穷远点。 （否则L1和L2有公共的无穷远点P ，则L1和L2有两个交点A、P，故假设错误。）
4. 平面上全体无穷远点构成一条**无穷远直线**。（自己想象一下这条直线吧）
5. 平面上全体无穷远点与全体平常点构成**射影平面**。

# 射影平面坐标系

射影平面坐标系是对普通平面直角坐标系（就是我们初中学到的那个笛卡儿平面直角坐标系）的扩展。我们知道普通平面直角坐标系没有为无穷远点设计坐标，不能表示无穷远点。为了表示无穷远点，产生了射影平面坐标系，当然射影平面坐标系同样能很好的表示旧有的平常点（数学也是“向下兼容”的）。



假设平面直角坐标系上有点A（x,y），做如下变换：

令x=X/Z ，y=Y/Z（Z≠0）；则在射影平面坐标系中，A点可以表示为（X:Y:Z）。 同样的，新坐标系中直线的方程可以表示为aX+bY+cZ=0。

有了直线方程，思考一下无穷远点该怎么表示？我们知道无穷远点是两条平行直线的交点。假设两条平行线的方程如下：

aX+bY+c1Z =0； aX+bY+c2Z =0  (c1≠c2)；

联立两方程，得到c1Z- c2Z=0.

∵c1≠c2

∴Z=0

∴aX+bY=0；

所以无穷远点就是这种形式（X：Y：0）表示。注意，平常点Z≠0，无穷远点Z=0，因此无穷远直线对应的方程是Z=0。

# 椭圆曲线

椭圆曲线的定义： 一条椭圆曲线是在射影平面上满足方程

**y2+a1xy+a3y = x3+a2x2+a4x+a6**

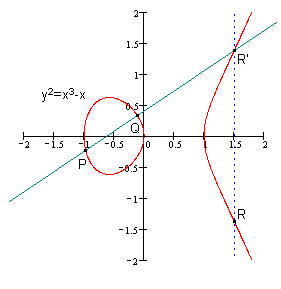
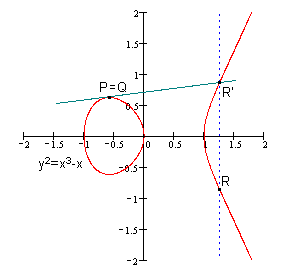
且曲线必须光滑，并存在一个无穷远点。

由椭圆曲线的定义可以知道，椭圆曲线是光滑的，所以椭圆曲线上的平常点都有切线。而切线最重要的一个参数就是斜率k。

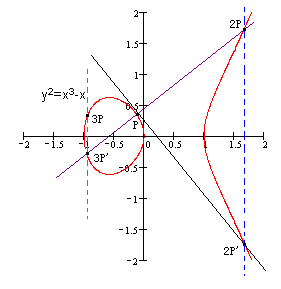
  例3.1：求椭圆曲线方程y2+a1xy+a3y=x3+a2x2+a4x+a6上，平常点A(x,y)的切线的斜率k。   
   解：令F(x,y)= y2+a1xy+a3y-x3-a2x2-a4x-a6   
   求偏导数 ：  
   Fx(x,y)= a1y-3x2-2a2x-a4   
   Fy(x,y)= 2y+a1x +a3   
   则导数为：f'(x)=- Fx(x,y)/ Fy(x,y)=-( a1y-3x2-2a2x-a4)/(2y+a1x +a3)   
                   = (3x2+2a2x+a4-a1y) /(2y+a1x +a3)   
   所以k=(3x2+2a2x+a4-a1y) /(2y+a1x +a3)

# 椭圆曲线上的加法

加法运算法则：任意取椭圆曲线上两点P、Q （若P、Q两点重合，则做P点的切线）做直线交于椭圆曲线的另一点R’，过R’做y轴的平行线交于R。我们规定P+Q=R。

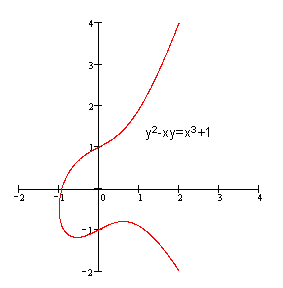
k个相同的点P相加，我们记作kP。如下图：P+P+P = 2P+P = 3P。



例4.1：已知椭圆曲线方程y2+a1xy+a3y=x3+a2x2+a4x+a6，平常点坐标P(x1,y1)，Q(x2,y2)，求P+Q的和R(x4,y4)的坐标。

解：（1）先求点-R(x3,y3)   
   因为P,Q,-R三点共线，故设共线方程为y=kx+b,其中   
   若P≠Q(P,Q两点不重合) 则   
       直线斜率k=(y1-y2)/(x1-x2)   
   若P=Q(P,Q两点重合) 则直线为椭圆曲线的切线，故由例3.1可知：   
       k=(3x2+2a2x+a4 -a1y) /(2y+a1x+a3)   
  
   因此P,Q,-R三点的坐标值就是方程组：   
       y2+a1xy+a3y=x3+a2x2+a4x+a6    -----------------[1]   
       y=(kx+b)                     -----------------[2]   
   的解。   
  
   将[2]，代入[1] 有   
      (kx+b)2+a1x(kx+b)+a3(kx+b) =x3+a2x2+a4x+a6    --------[3]   
   对[3]化为一般方程，根据三次方程根与系数关系（当三次项系数为1时；-x1x2x3 等于常数项系数， x1x2+x2x3+x3x1等于一次项系数，-(x1+x2+x3)等于二次项系数。）   
   所以-(x1+x2+x3)=a2-ka1-k2   
   x3=k2+ka1-a2-x1-x2;---------------------求出点-R的横坐标 （这里，注意原文有误）  
   因为k=(y1-y3)/(x1-x3) 故   
   y3=y1-k(x1-x3);-------------------------------求出点-R的纵坐标   
  
  （2）利用-R求R   
   显然有 x4=x3=k2+ka1-a2-x1-x2;------------求出点R的横坐标   
   而y3 y4 为 x=x4时 方程y2+a1xy+a3y=x3+a2x2+a4x+a6的解 （注意，这里不能直接取y4=- y3 ，因为椭圆曲线不一定关于x轴对称）  
   化为一般方程y2+(a1x+a3)y-(x3+a2x2+a4x+a6)=0 , 根据二次方程根与系数关系得：   
       -(a1x+a3)=y3+y4   
   故y4=-y3-(a1x+a3)=k(x1-x4)-y1-(a1x4+a3); ---------------求出点R的纵坐标   
   即：   
       x4=k2+ka1+a2+x1+x2;   
       y4=k(x1-x4)-y1-a1x4-a3;

本节的最后，提醒大家注意一点，以前提供的图像可能会给大家产生一种错觉，即椭圆曲线是关于x轴对称的。事实上，椭圆曲线并不一定关于x轴对称。如下图的y2-xy=x3+1



# 密码学中的椭圆曲线

前面学到的椭圆曲线是连续的，并不适合用于加密；因为：

1. 实数域上的椭圆曲线是连续的，有无限个点，密码学要求有限点。
2. 实数域上的椭圆曲线的运算有误差，不精确。密码学要求精确。

所以，我们必须把椭圆曲线变成离散的点，把椭圆曲线定义在有限域上。

有限域Fp具有如下一些特性：

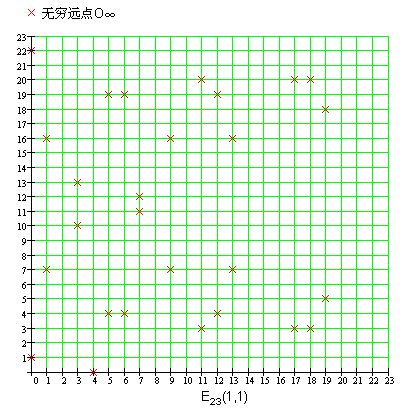
1. Fp中只有p（p为素数）个元素0,1,2 …… p-2,p-1；
2. Fp 的加法（a+b）法则是 a+b≡c (mod p)；即，(a+b)÷p的余数 和c÷p的余数相同。
3. Fp 的乘法(a×b)法则是  a×b≡c (mod p)；
4. Fp 的除法(a÷b)法则是  a/b≡c (mod p)；即 a×b-1≡c  (mod p)；（b-1也是一个0到p-1之间的整数，但满足b×b-1≡1 (mod p)；具体求法可以参考初等数论，或[另一篇文章](https://www.pediy.com/kssd/pediy06/pediy50391.htm)）。
5. Fp 的单位元是1，零元是 0。

同时，并不是所有的椭圆曲线都适合加密。y2=x3+ax+b是一类可以用来加密的椭圆曲线，也是最为简单的一类。下面我们就把y2=x3+ax+b 这条曲线定义在Fp上：

选择两个满足下列条件的小于p(p为素数)的非负整数a、b   
      4a3+27b2≠0　(mod p)   
则满足下列方程的所有点(x,y)，再加上 无穷远点O∞ ，构成一条椭圆曲线。   
     y2=x3+ax+b  (mod p)

其中 x,y属于0到p-1间的整数，并将这条椭圆曲线记为Ep(a,b)。

y2=x3+x+1  (mod 23)的图像如下图所示：



Fp上的椭圆曲线同样有加法，且具有如下特性：

1. 无穷远点 O∞是零元，有O∞+ O∞= O∞，O∞+P=P
2. P(x,y)的负元是 (x,-y)，有P+(-P)= O∞
3. P(x1,y1),Q(x2,y2)的和R(x3,y3) 有如下关系：   
        x3≡k2-x1-x2(mod p)   
        y3≡k(x1-x3)-y1(mod p)   
        其中若P=Q 则 k=(3x2+a)/2y1  若P≠Q，则k=(y2-y1)/(x2-x1)

例5.1 已知E23(1,1)上两点P(3,10)，Q(9,7)，求1)-P，2)P+Q，3) 2P。   
   解 1)  –P的值为(3,-10)   
      2)  k=(7-10)/(9-3)=-1/2，2的乘法逆元为12 因为2\*12≡1 (mod 23)   
          k≡-1\*12 (mod 23) 故 k=11。   
          x=112-3-9=109≡17 (mod 23);   
          y=11[3-(-6)]-10=89≡20 (mod 23)   
          故P+Q的坐标为(17,20)   
      3)  k=[3(32)+1]/(2\*10)=1/4≡6 (mod 23)   
          x=62-3-3=30≡20 (mod 23)   
          y=6(3-7)-10=-34≡12 (mod 23)   
          故2P的坐标为(7,12)

最后，我们讲一下椭圆曲线上的点的阶。

如果椭圆曲线上一点P，存在最小的正整数n，使得数乘nP=O∞，则将n称为P的 **阶**，若n不存在，我们说P是无限阶的。

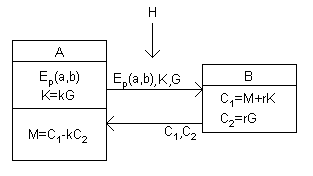
事实上，在有限域上定义的椭圆曲线上所有的点的阶n都是存在的（证明，请参考近世代数方面的书）

# 椭圆曲线上简单的加密/解密

公开密钥算法总是要基于一个数学上的难题。比如RSA 依据的是：给定两个素数p、q 很容易相乘得到n，而对n进行因式分解却相对困难。那椭圆曲线上有什么难题呢？   
  
考虑如下等式：   
   K=kG  [其中 K,G为Ep(a,b)上的点，k为小于n（n是点G的阶）的整数，表示k个G做加法]   
   不难发现，给定k和G，根据加法法则，计算K很容易；但给定K和G，求k就相对困难了（首先是因为有限域上没有减法运算，其次正向穷举时运算量会比较大）。   
   这就是椭圆曲线加密算法采用的难题。我们把点G称为基点（base point），k（k<n，n为基点G的阶）称为私有密钥（privte key），K称为公开密钥（public key)。

  现在我们描述一个利用椭圆曲线进行加密通信的过程：   
  
   1、用户A选定一条椭圆曲线Ep(a,b)，并取椭圆曲线上一点，作为基点G。   
   2、用户A选择一个私有密钥k，并生成公开密钥K=kG。   
   3、用户A将Ep(a,b)和点K，G传给用户B。   
   4、用户B接到信息后 ，将待传输的明文编码到Ep(a,b)上一点M（编码方法很多，这里不作讨论），并产生一个随机整数r（r<n）。   
   5、用户B计算点C1=M+rK；C2=rG。   
   6、用户B将C1、C2传给用户A。   
   7、用户A接到信息后，计算C1-kC2，结果就是点M。因为   
          C1-kC2=M+rK-k(rG)=M+rK-r(kG)=M   
      再对点M进行解码就可以得到明文。

  在这个加密通信中，如果有一个偷窥者H ，他只能看到Ep(a,b)、K、G、C1、C2 而通过K、G 求k 或通过C2、G求r 都是相对困难的。因此，H无法得到A、B间传送的明文信息。



# 参考文献

1. <https://www.pediy.com/kssd/pediy06/pediy6014.htm>
2. <https://www.cnblogs.com/Kalafinaian/p/7392505.html>

# 附图

