大家好，我们组今天为大家带来的展示是加速收敛中的steffensen方法

回顾上节课所学到的Aitken’s △2 method，构造序列各项是按照下面的顺序构造的，从第四项开始用右边的定义的这个序列来代替原序列进行收敛。由上节课所学，我们知道这是一个线性收敛序列，事实上，只要把Aitken方法应用于根据不动点迭代所得到的线性收敛序列，我们还能够将收敛速度加快到二次。这个方法称为Steffensen方法。

下面我们看一下Steffensen方法具体是怎么构造序列的。Steffensn方法构造的序列前面四项与Aitken’s方法中前面四项相同，不同的是在这一步假设p0比p2更好地逼近p，从而将不动点迭代应用于p0而不是p2。每一行的第一项都是由在Aitken方法中得出的公式产生的，其他项则是将不动点迭代算法应用在前一项上。这样更新后的pn+1在下一步中就是g（p0），pn+2就是g（p1），依次类推后面每一项，不断迭代直到达到需要的精确度。

和Aitken对比我们可以看出steffensen每次都是更新了两项代入公式，而Aitken只更新一项

这个过程用算法描述则更为详细直观

在算法中不断重复中间这个过程，p1，p2由应用在前一项上的不动点迭代产生，p由公式产生……

需要注意的是，可能存在分母为0的情况，这种情况下终止序列并选取上一次迭代的最后一项，p2作为近似解

下面我们来看一个steffensen实际应用的例子。在这个式子中，我们通过...得到不动点方法g（x）=...。我们选取1.5作为初始值p0，可以看出，只要经过2轮迭代运算，精确度就可以打打小数点后9位，这个精度和newton迭代法相似，而我们已知newton迭代是二次收敛的，可以猜想steffensen方法也是二次收敛的。

下面这个定理证明了steffensen是二次收敛的。假设x=g（x）有解p且g’(p)≠1，如果存在δ＞0，...

steffensen的一般表达式

Steffensen方法的主要优点是它具有牛顿法那样的二次收敛性。但是，牛顿方法的公式需要评估函数的导数，而斯特芬森的方法仅需要自身。当难以求导的时候，这一点很重要。

缺点：函数复杂时代价大，

与大多数其他迭代根查找算法相似，Steffensen方法的主要缺点在于起始值的选择。如果的值与实际解还不够“接近”，则该方法可能会失败，并且值的序列可能会在两个极端之间翻转，或者发散到无穷大（可能两者都有！）。