目 录

[**A 控制方程** 2](#_Toc532372303)

[**B 度量系数** 5](#_Toc532372304)

[**C 时间推进方法** 7](#_Toc532372305)

[**D 空间离散格式** 11](#_Toc532372306)

[**E 多重网格算法** 16](#_Toc532372307)

[**F 边界条件** 22](#_Toc532372308)

**A 控制方程**

曲线坐标系下的三维非定常可压N-S方程表示为：



式中的Q为守恒变量矢量，F，G，H分别为三个方向的通量。

无粘通量项表示为：



其中逆变速度分别表示为：



粘性通量项表示为：



其中应力项和热通量项表示为：



**B 度量系数**

将N-S方程转到到曲线坐标系下得：



根据链式求导法则，得



式中， 是度量系数，下面给出度量系数的计算方法：





其中J为雅克比矩阵行列式，1/J为单元格的体积。





**C 时间推进方法**

CCFD V3.0采用隐式时间格式，使用AF近似分解方法，ADI交替方向推进方法。



其中，粘性项采用显式处理。









计算坐标下的N-S公式可以变形为：



或者



或者



对上述公式进行近似因子分解（AF）得：



ADI三个方向交替推进步骤为：





这样就将上述公式分为三个方向上的交替计算方法，得



其中



极大的简化了计算复杂度，对左右特征值矩阵求解之后，分解之后的正负特征值形成的三对角矩阵L+D+U,有多种方法进行求解，如Thomas追赶法的直接求解、(L+D)-1D(D+U)的近似迭代求解。

**D 空间离散格式**

**D.1 通量差分方法**

对流项及压力项的空间差分可以表示为：



将上述公式以通量形式表示为：



对上述公式做通量差分（flux-difference splitting）方法变换，





上述表示的使用Roe平均的通量差分方法，其中Roe平均值表示：



其中：





对角矩阵 是矩阵的特征值矩阵。



右边项可以表示为：



其中：



其中的Roe平均值分别表示为：



**D.2 熵修正**

在有激波等强间断处，激波捕捉方法（shock-capturing scheme）对对角化后的三个特征值进行修正，修正方法如下：



如果有，则：



其中 的常值，范围0.01~0.4 ，一般在马赫数大于0.7情况下示意，马赫数越大，该值取值越大。

**D.3 限制器**

几种常用的限制器如下：

1. **van-albada限制器**



其中梯度表示：



为限制器系数，在CCFDV3.0用xkap表示。



1. **min-mod限制器**



minmon算子表示为：



其中:



b表示为：b=(3-k)/(1-k)

1. **spekreijse vankat限制器**



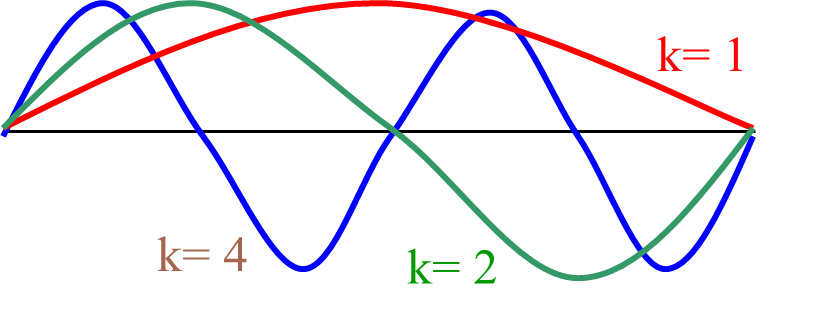
限制器算子表示为：

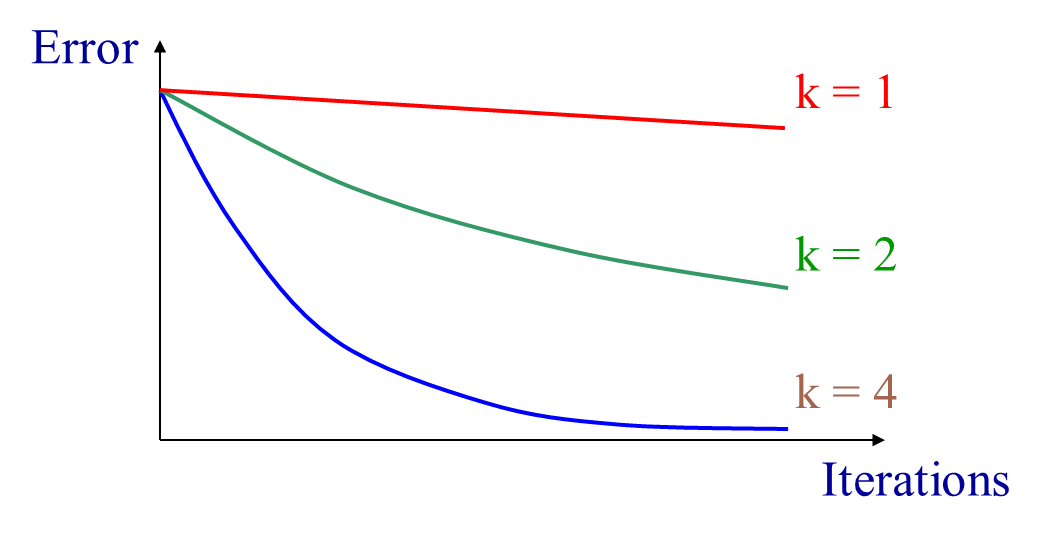


**E 多重网格算法**

Multigrid多重网格算法，是一种加快迭代收敛速度的方法，可以分为几何多重网格和代数多重网格方法，CCFDV3.0使用的是几何多重网格算法。

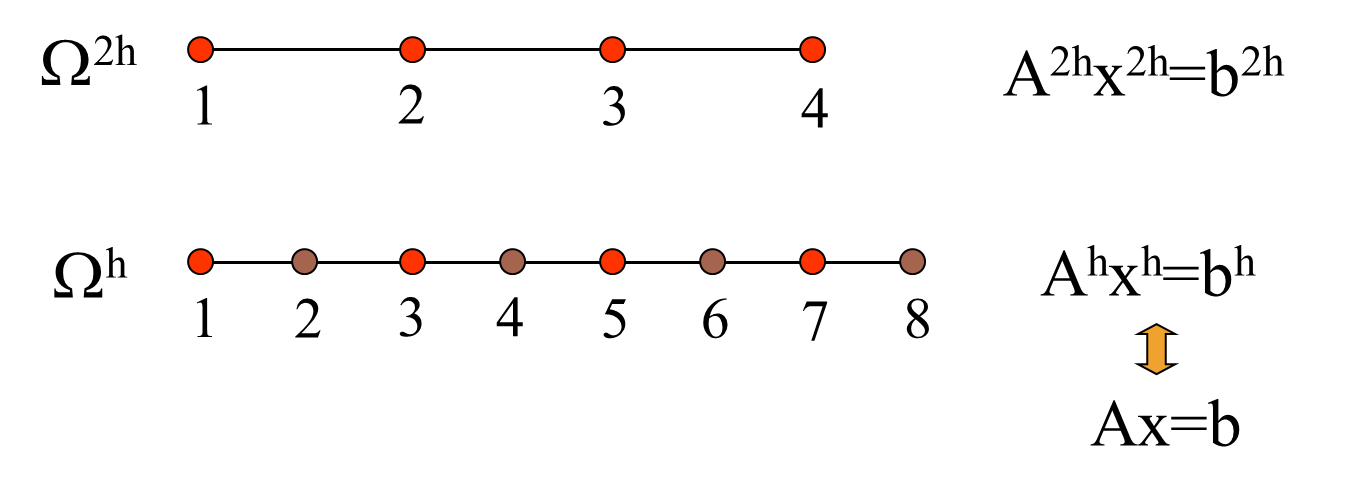
总的来说，残差包括两种一种为高频误差，一种为低频误差，大多数迭代法在细网格上可以很快消除误差的高频分量，但低频分量相当顽固。可以在粗网格上消除这些低频分量。





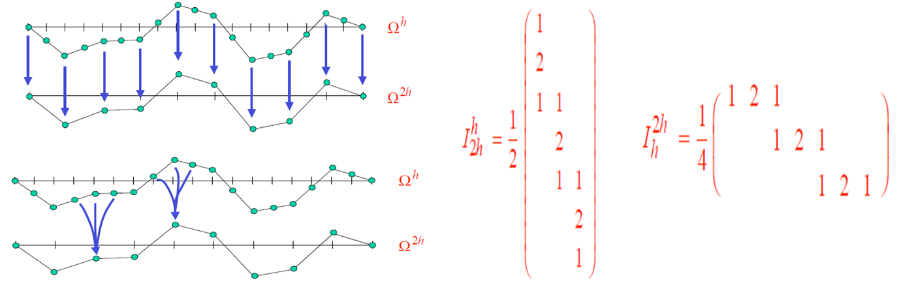
根据上图所示，k=4的高频误差，可以很快的收敛，k=1的低频误差收敛速度很慢，但是这种低频误差在粗网格上，则可以较快的收敛，这也是多重网格算法的基本思路。

多重网格算法的基础分为限制（restrict）操作，即细网格限制粗化到粗网格上，提升（prolonged）操作，即粗网格提升插值到细网格。

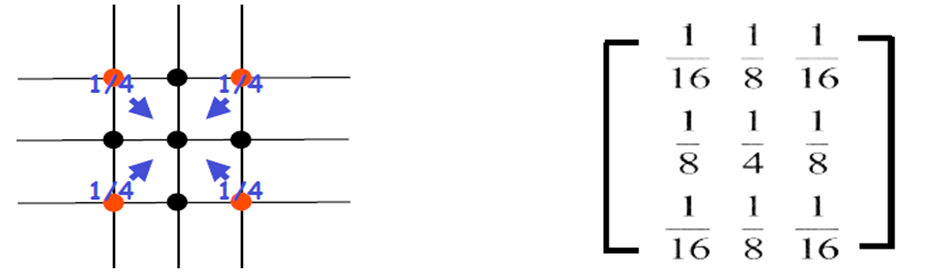


限制算子，



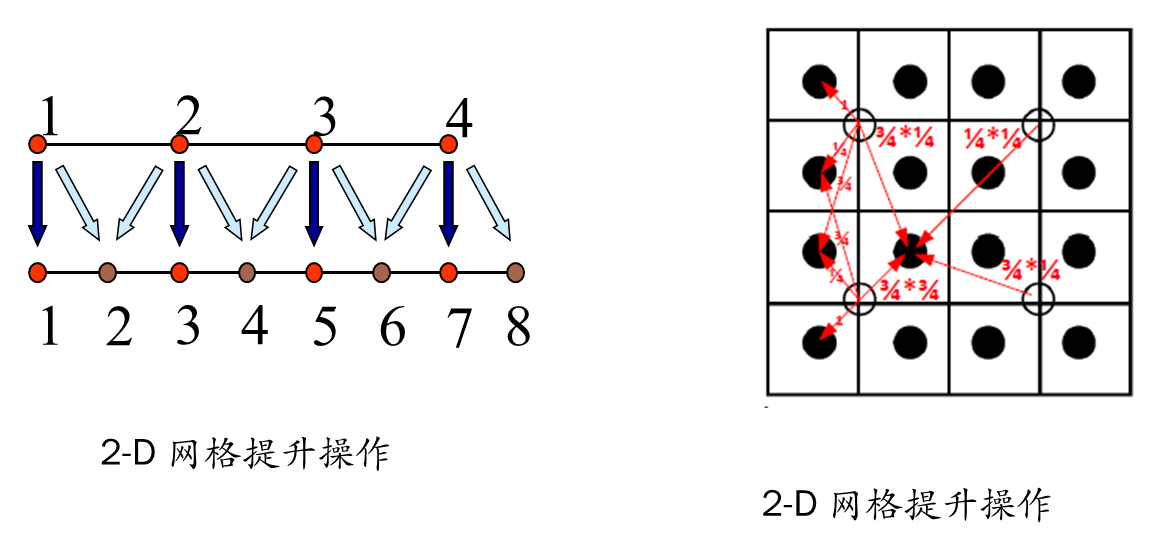


在三维网格下，一般使用体积加权平均方法，向下限制操作，二维的表示如下：



提升算子， 。





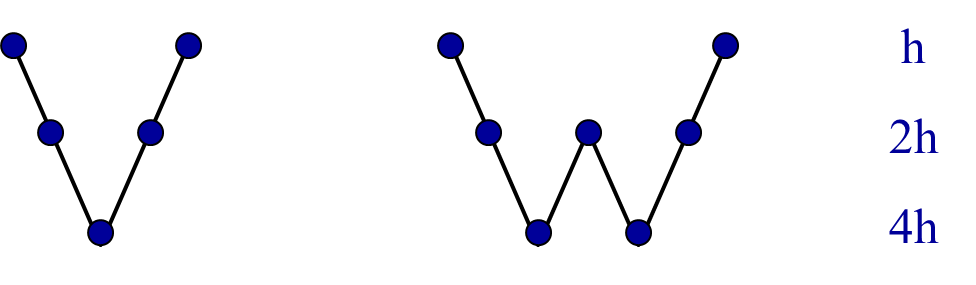
多重网格的限制算子和提升算子需要满足Galerkin条件，



以Ax=b推导：



多重网格的限制提升操作，常用的有V-cycle和W-cycle。



下面详细的介绍下每层的提升和限制在n-s方程下的完整推导过程，根据上面的公式推导知道：



该公式以半离散形式进行计算，右端项（源项）以flux-split（Roe）方法进行计算，左端项多AF多项式近似分解，可以使用多种隐式方法，LU-ADI、LU-SGS等；

将上述公式简化表示，方便下面公式推导：





τ是截断误差

L表示的左端项的时间操作，S表示右端项，Q表示计算项； 表示从细网格到粗网格精确解， 表示从细网格层到N-1层的限制操作；同理， 表示从N-1到细网格层的提升（插值）操作；



在N层上，那么使用LU-ADI方法计算该等式，得到一个计算解，可以表示为：



表示的是在该层的残差；同理，递归到N-1层，可以表示为：



或者



上述等式有个隐藏的关键就是，我们将上一层计算的q值，限制下来，作为N-1层的初始计算解，所以，在修正的时候公式表达为：



所以插值为：



在N-1层计算的时候，也不是精确计算，同样是近似解，那么在N-1层一样是有残差的，那么上述公式进一步表示为：



同理在递归到N-2层时，



如果我们合并表示成，包含源项和误差项公式，











Multigrid算法的提升和限制操作，必须满足Galerkin条件，R表示向下限制操作，P表示向上提升（插值）操作即：





递归到N-2层可以表示为：



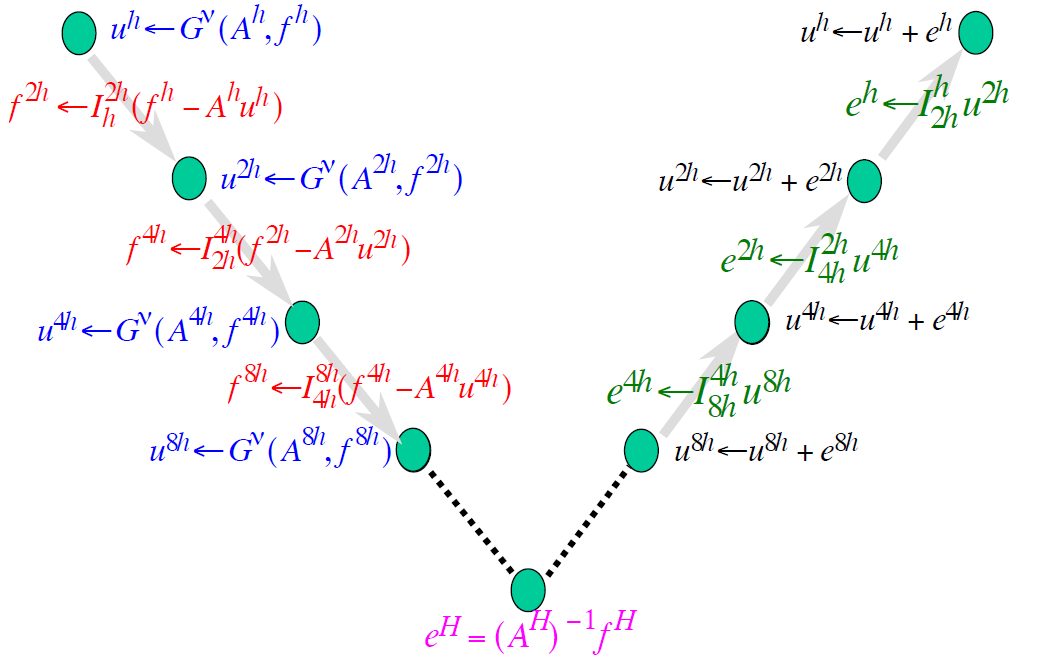
可得：



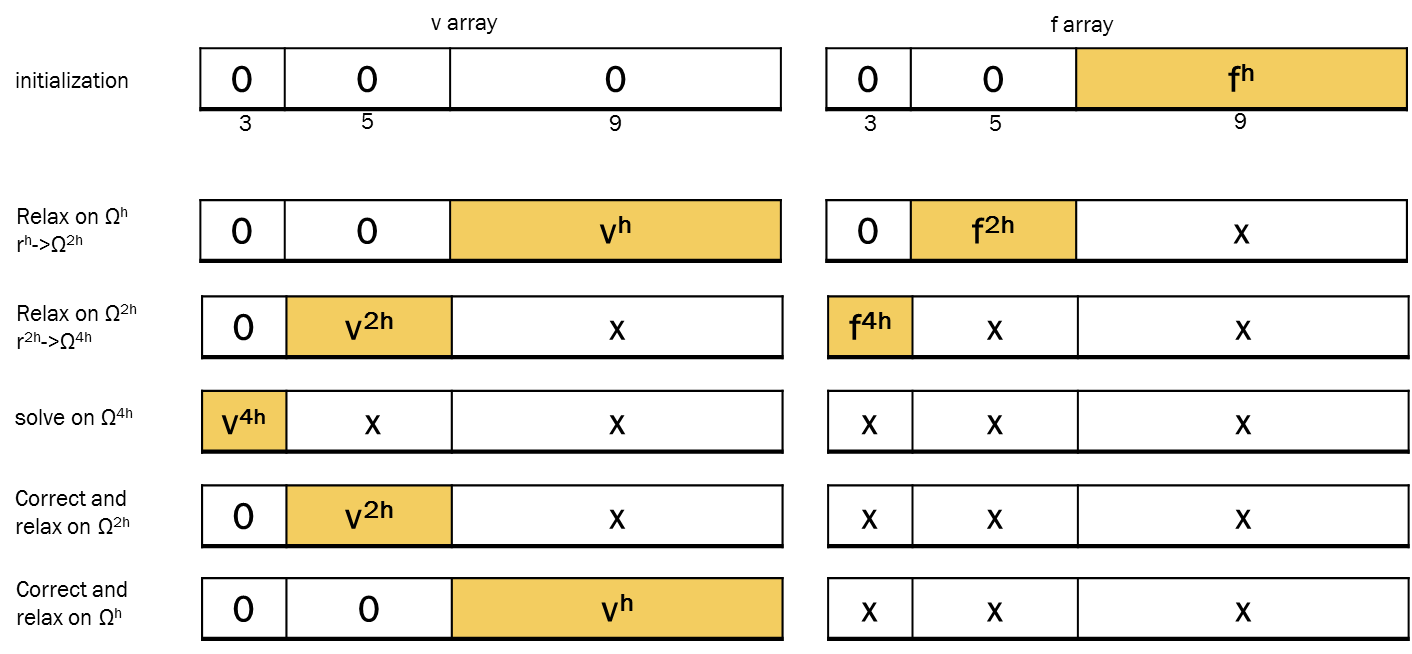
所以残差、截断误差可以表示为：



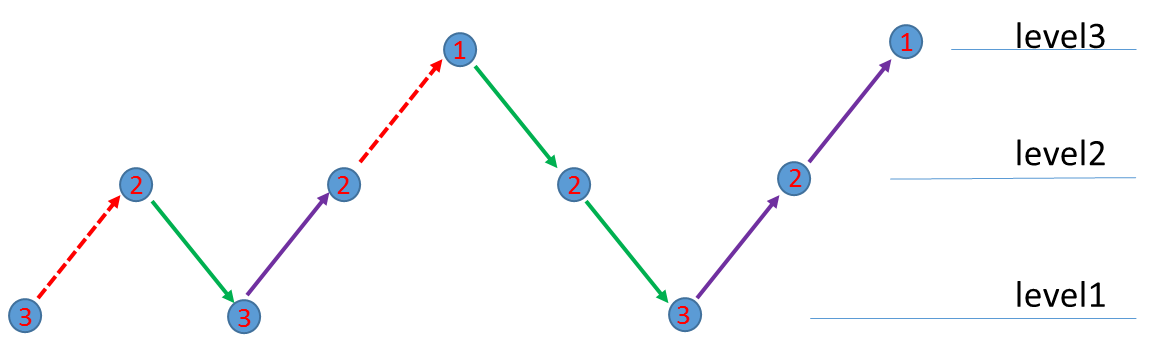
多重网格的V-cycle的流程图表示：



在编程中的V-cycle数组上使用情况。



在计算N-S方程时，起始的流场值是人为的给定一个随机值，是否可以给定一个更加合理的值，同时为了发挥多重网格算法在粗网格快速消除低频误差的特性？引入Mesh Sequence方法。其基本原理就是先在粗网格上进行流场计算，又因为粗网格上网格规模已经成几何倍数的减少，所以计算时间开销很小，可以很快收敛到一个合理的值，再将该值插值到更细一层的网格作为初始值进行计算。比单纯使用多重网格算法，这种操作可以加快细网格上的收敛速度。后续篇幅有Mesh Sequence算法的测试结果说明。下图为三层的Mesh Sequence算法示意图，从最粗网格计算，值插值提升到更细网格层，再进行2层的V-cycle计算，得到的结果提升插值到原始细网格上，进行3层的V-cycle计算。



Mesh Sequence methods

**F 边界条件**

CCFD V3.0支持4种边界类型，1-1（cut）边界、对称边界、wall边界和远场边界。分别说明如下：

1. **远场边界,** farfield

远场边界一般通过引入与边界垂直的一维无粘流动的Riemann不变量来处理。对于一维非定常等熵流动，流场存在以下两族特征线:



速度在 方向的投影满足下述条件，其中 为逆变速度。





其中 表示沿着特征线 不变值， 表示沿着特征线 不变值，可以得到边界上的法向速度和声速，分别为：





ref表示为无穷远流场值。

可以计算出边界处，三个方向上的物面速度，压力值及密度值，分别为：



1. **对称边界,** symmetryplane

对称边界条件比较简单，只要使对称面两侧的速度分量符号相反，密度与压力等值即可。



三个方向的速度分别表示为，其中 为逆变速度。



1. **无粘壁面,** wallinviscid

无粘壁面边界的密度和压力使用相邻内点网格数据：



同时满足固壁无穿透条件：



1. **粘性壁面,** wallviscid

粘性壁面边界条件的物面压力可通过法向动量方程简化形式计算为：



壁面附近温度由壁温条件决定，即



壁面速度满足无穿透和无滑移条件：



壁面的压力和密度计算如下：

