有限元算法说明

这里处理的方程组为

|  |  |
| --- | --- |
|  | (0) |

# 关于边界条件的处理

(0)🡪

🡪

注意到有以下向量恒等式：

🡪

(散度定理) 🡪

🡪

|  |  |
| --- | --- |
|  | (1-1) |

这里的表示面元的外法向。

若选取一组空间基函数，将待求函数展开为：

则(1-1)式可以改写为线性方程组：

🡪

|  |  |
| --- | --- |
|  | (1-2) |

如果矩阵具有以下形式：

其中为标量函数，为不依赖于空间坐标的常数矩阵。则可以将矩阵简化为：

其中

于是矩阵可以用直积来构造：

所以多分量方程组系数矩阵可以按如下步骤构造：

1. 构造单分量方程的系数矩阵；
2. 通过直积得到多分量方程组的系数矩阵。

问题：当Ndomains比较小时，MP是一个稠密的矩阵，此时用sparse存储MG并不划算，矩阵运算效率较低。Python有Block Sparse Matrix(BSR格式)，但MATLAB没有。。。

## 常数第一类边界条件

其中为不依赖于空间的常数。处理这种边界条件很简单，由于方程是线性的，函数也满足原来的方程，并且在边界上恒等于零。于是我们可以直接求解函数，并要求上述基函数在边界上为零。注意，此时。变形后的方程：

所以(1-2)式应该改写为：

其中为的展开系数，为的加权展开系数：

这里选取的基函数在边界上都是零，而在方向的边界附近并不趋于零，所以可能并不能很好地逼近。为的展开系数：

与有相同的问题。故这种边界处理方法的结果可能并不好。

## 第一类边界

如果第一类边界条件不是常数：

或者说方程不是线性的。对于这种边界条件，我们可以引入一组边界基。设前面引入的基函数在边界上均为零：

另外引入一组在边界上非零的边界基，将待求函数重新展开为：

通过条件

可求出系数 (例如取个点联立线性方程组，或者计算重叠积分)，方程(1-2)修改为：

|  |  |
| --- | --- |
|  | (1-3) |

其中

## 第二类边界条件

对于这种边界条件，1.1节中的基需要包含边界基，然后可直接适用方程(1-2)。边界效应体现在中。

## 关于第一类边界条件的物理含义(proposed by 陈劲夫)

气体原子是不能穿过气室壁的，在热平衡条件下，也不应该吸附在气室壁上。所以应该有：

另一方面，如果我们还希望指定第一类边界条件：

那么将导致边界条件超出定解的要求，方程不一定有解。实际上，边界上的总原子密度

并不是已知的，我们希望给定的边界条件仅仅是而已。所以第一类边界条件应该改写为：

|  |  |
| --- | --- |
|  | (1-6) |

满足的方程直接由(0)式给出：

|  |  |
| --- | --- |
|  | (1-7) |

实际上，我们的矩阵总是满足：

所以(1-7)式简化为：

|  |  |
| --- | --- |
|  | (1-8) |

如果我们给定的初态刚好就是稳态解，那么边界条件(1-6)就重新简化为原来的第一类边界条件：

另外，从方程(1-8)可以看出，激光并不会改变总原子密度的分布。

# 基的选取

采用直积形式的基：

中的函数是在给定格点为1，其余格点为零的线性函数：

中的函数是在给定区间上的Lobatto函数：

当均来自时，为nodal基，它在与格点相邻的8个区域内都是非零的；

当中有两个来自时，为edge基，它在与边相邻的4个区域内都是非零的；

当中有三个来自时，为surface基，它在与面相邻的2个区域内都是非零的；

当全部来自时，为domain基，它仅在1个区域内有非零值。

边界基仅包含nodal、edge和surface基。

## 基的遍历方法

对于每一个domain，使用一个三重循环来遍历所有在这个domain中有非零值的基：

for ix=0:K+1

for iy=0:K+1

for iz=0:K+1

countPoint= count(ix<2, iy<2, iz<2);

if countPoint == 3

% nodal basis

elseif countPoint == 2

% edge basis

elseif countPoint == 1

% surface basis

else

% domain basis

end

end

end

end

ix=0和ix=1对应的两个基属于，其中ix=0的基在处取得最大值；ix=1的基在处取得最大值。ix>1对应的基为，属于。iy，iz与此类似。

## 基的内积

设

则有：

也就是说为了计算系数矩阵，实际上我们只需要知道就足够了。矩阵由于涉及未知的权函数，需要使用数值积分。

下面来计算在区间上积分的值。为了精确起见，这里重新对记号进行说明：