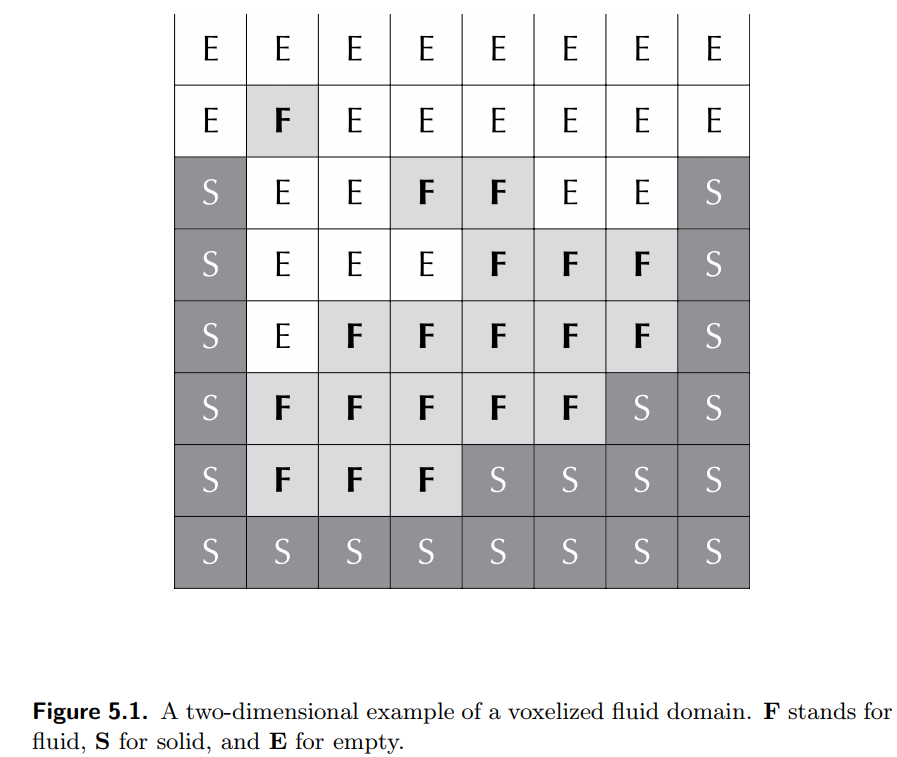
在本章中,我们将研究流体模拟的核心,使流体不可压缩并同时执行边界条件:实现我们在第二章前面提到的.

函数从间接速度场减去压强梯度,

使结果满足流体内部的不可压缩性,

并满足固体壁边界条件

同时还要考虑自由表面边界条件，即那里的压强为零：

在开始离散化这些方程和数值求解方法之前,我们需要对流域进行离散建模.我们将从“体素化”模型开始本章,其中每个体素分别标记为流体，固体或空.流动体素包含流体，固体体素包含固体,而空体素（如果有）根本不包含任何东西-在自由表面液体模拟中，它们代表未建模的空气.固体和流体体素之间的体素将成为我们施加固体边界条件的位置,而流体和空体素之间的体素面是我们的自由表面.有关说明，请参见图（5.1）。我们将放弃对确定离散标注的完整讨论,在本章的后面，我们将使用水平集切换到更精确的几何模型,但这并不是一个困难的命题：检查每个体素的中心是否在实体内部或流体内部，或者都不是. 

考虑到这个模型,我们将首先写下压力更新的离散化:我们如何估计MAC网格上的压强梯度(假设我们知道压强)?之后,我们将研究在MAC网格上定义离散散度,然后将两者放在一起,提出一个线性方程组来求解以找到压力.我们将介绍该系统以及解决该问题的一些有效方法.

5.1 离散压强梯度

正如我们之前简要讨论过的,MAC网格存在的理由是,交错使精确中心差变得稳健.例如,在我们需要从速度的分量减去压力梯度的分量的情况下,在分量两侧各完美地排列着一个压力值,只是在等待它们之间的差值.您可能需要参考图2.1来了解其工作原理.

以下是二维压强更新公式:

在三维空间中,

这些压力更新适用于在两侧具有有效流体压力并且实际上与某些流体接壤的每个速度分量。 编写正确的压力求解器最棘手的部分之一就是跟踪哪些速度和压力样本处于“活动状态”，因此最好记住这一点，或者在编程时参考您附近的图表。 对于体素化流体模型，含流体的体素需要解决的压力未知（此时有效），固体体素没有有效压力，空体素具有已经固定为零的有效压力。 我们将用方程式（5.1）或（5.2）更新的活动速度分量是至少与一个流体体素，零个或一个空体素以及无固体体素相邻的那些。

再次关注空体素：由于自由曲面边界条件指定压力为零，因此我们将空体素中的压力设置为零。 如果您对技术术语感兴趣，这称为Dirichlet边界条件：Dirichlet表示我们直接在边界处指定数量的值.这通常是在数值求解中要处理的最简单的边界条件.

更为困难的压力边界条件是在实心壁上，尽管在处理体素化几何近似时仍然不太麻烦，在这种情况下，交错的速度分量在位置和法线上都与实心表面精确对齐。 我们可以超越上面的压力梯度更新一步，直接将与流体相邻的固体体素面上的流体速度分量设置为等于那里的固体速度。 我们将在压力梯度更新的一部分过程中进行此操作。 我们甚至可以通过在概念上包括固体体素内部压力的幻影值，将其视为压力梯度更新的扩展。例如,如果处的速度分量介于处的固体体素和处的流体体素之间,然后设置

兼容压强梯度更新

虚拟压强在满足

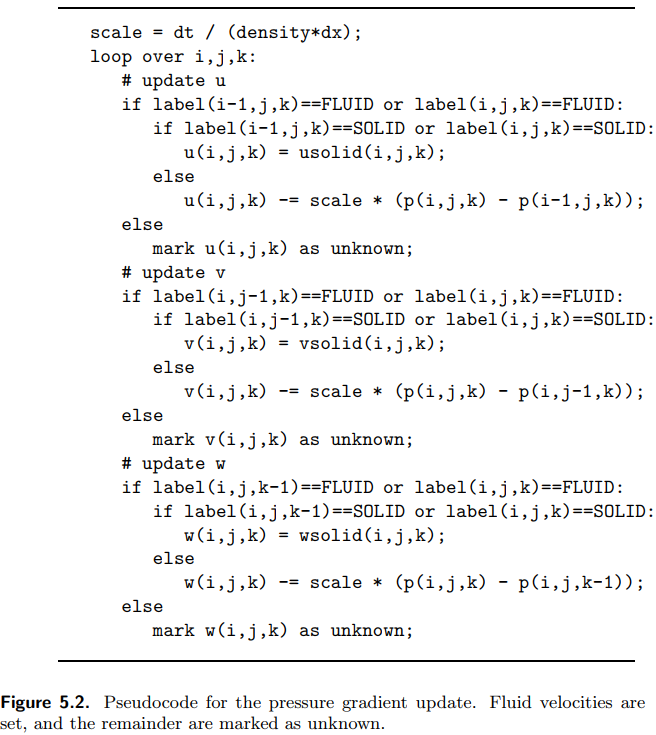
这对于编程而言并不重要,但是这是一个值得考虑的概念:我们可以将流体量,压力扩展到固体中,以使我们在流体内部使用的常用方程式也可以处理边界条件.

实际上,只需稍作重写,该固体边界处的压强条件就是

我们可以将其视为有限差分近似

回到连续方程,如果我们同样将压力梯度更新代入固体边界条件,我们得到的结果更普遍:

当求解压强时,固体边界条件等于指定压强的正态导数,而不是压强值. 在PDE的研究中,这在技术上称为**Neumann**边界条件.



使用与第2章公式（2.14）-（2.17）相同的存储约定，可以将压力更新转换为类似于图5.2的代码。我们使用直接设置壁速的技巧，而不是将其作为压力更新来计算。诸如usolid（i，j，k）之类的术语可以用简单的表达式代替，而不是实际存储在数组中。我们还着重指出了流体速度未知的地方：我们将在本章的后面部分讨论这些速度.

边界条件可能很复杂，并且是出现错误时的常见原因。 您可能需要花些时间慢慢看一下本节，在您面前画一张MAC网格图（如图2.1所示），查看固体，流体和空气池的不同配置，直到您对所有这些都充满信心为止。

5.2 离散散度

现在开始本章的简单部分!在连续情况下,我们希望流体不可压缩:.在网格上,我们将以有限的差值近似此条件,并要求每个流体网格单元处估计的散度在时为零.

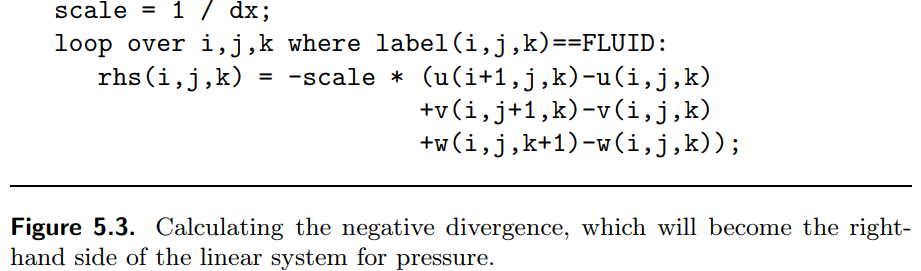
请记住,二维的散度是

并且在三维空间中

利用明显的中心差异(再次看一下MAC网格)我们将流体网格单元中的二维散度近似为

在三维空间中为

根据较早使用的存储约定,以与压力相同的方式存储散度,可以按照图5.3的伪代码实现.事实证明（见下文），我们实际上对散度的负值更感兴趣，因此我们将其存储在矢量中，称为rhs（代表线性系统的右侧）.



请注意,们只会评估标记为流体的网格单元的发散度.例如,我们的流体模拟与固体是否在改变体积无关.

解释离散散度的另一种方法是通过直接估计进入或离开网格单元的总流体速率.请记住,这(在精确的连续统设置中)只是围绕网格单元面的速度的法线分量的积分:

这是每个网格单元面上的积分之和.我们将速度的法线分量存储在每个面的中心，并且可以将其视为对整个面上的平均法线速度的良好（技术上为二阶精度）估计。因此，我们可以通过将速度的法线分量乘以脸部面积来轻松估算积分。请注意此处的符号-在上述积分中，法线始终是指向外部的，而存储在网格上的速度分量始终是相同的方向（例如，如图2.1所示）。重新缩放后，将得出完全相同的中心差公式。如果您有兴趣，我会让您自己解决。这种数值技术可以直接估算网格单元表面周围的量的积分，而不用查看微分方程的公式，这种方法称为**有限体积**[finite volume]法，本章稍后将在处理不规则实体边界时对此进行详细说明。

最后,我们可以解释为什么MAC网格如此有用.如果我们使用规则网格,且所有速度分量都存储在网格点,那么散度计算将很困难.例如,如果我们使用中心差分公式,

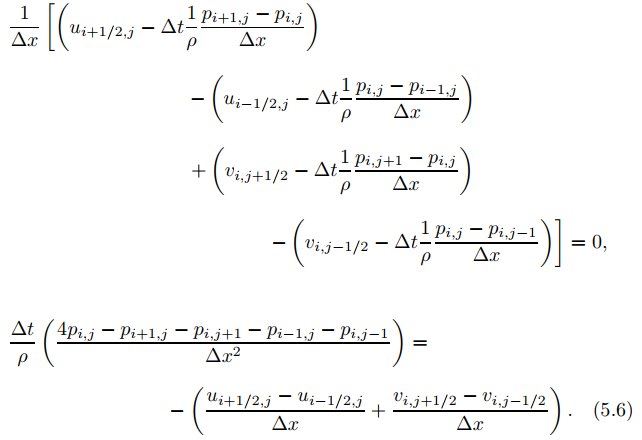
那么我们恰好有我们在第2章中提到的零空间问题.某些高度发散的速度场,例如，将得出零散度.因此，压强求解对校正它们无能为力,因此在仿真过程中,速度场中的高频振荡可能会持续存在甚至不稳定.在仍然使用并置网格的情况下，有两种可能的解决方法来解决此问题。第一种是使用偏斜的单边差分近似值，尽管这样做有效，但确实会给模拟带来一个独特的偏见，这可能会非常明显。第二种是在进行压力求解之前滤除高频发散模（即，平滑速度场，或至少是发散分量），以明确消除它们，但这很容易引入更多不必要的数值平滑。即使进行了适当的滤波，当涉及到并置速度时，线性求解（如下）甚至存在一些棘手的问题，因此我们坚持使用MAC网格。

5.3 压强方程

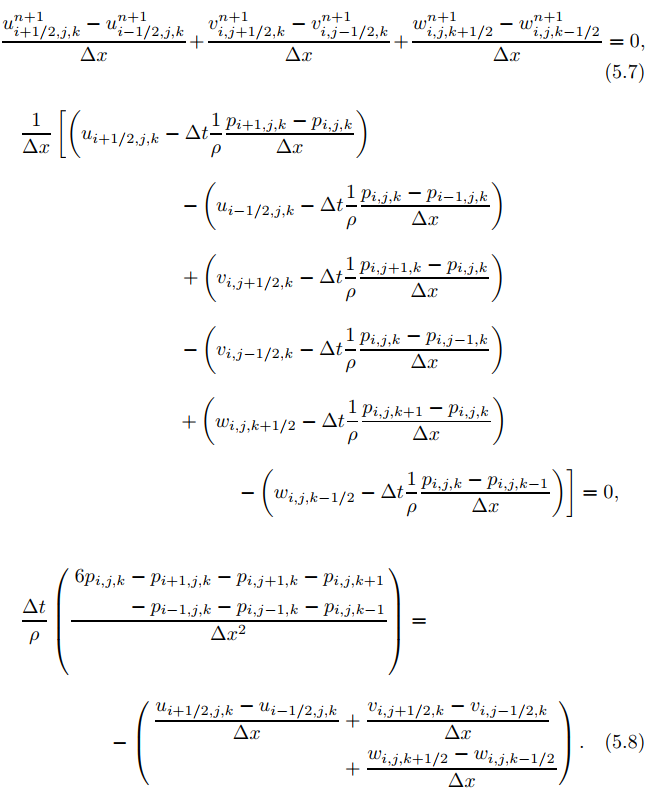
现在,我们需要解决不可压缩性的两个因素:如何使用压强梯度更新速度以及如何估算散度.

回想一下,我们希望最终速度在流体内散度为零.为了找到实现此目的的压强，我们只需简单地带入压强更新公式——将2D方程（5.1）和3D方程（5.2）分别代入散度方程(5,4)和(5.5).这为每一个流体网格单元提供了的线性方程(请记住,我们仅评估包含流体的网格单元的散度),压强为未知数.尽管我们可能会提及固体或空气中的压力,但没有方程式（但是我们先验地知道这些压力是指流体细胞的压力）.

代入2D网格点后,

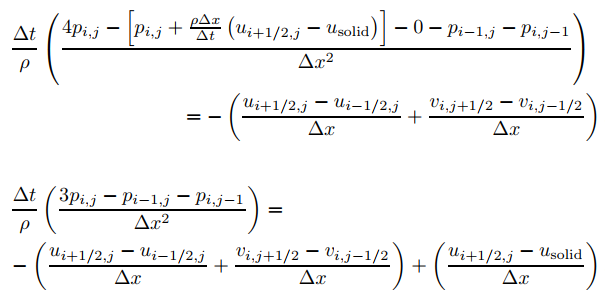


代入3D网格点后,

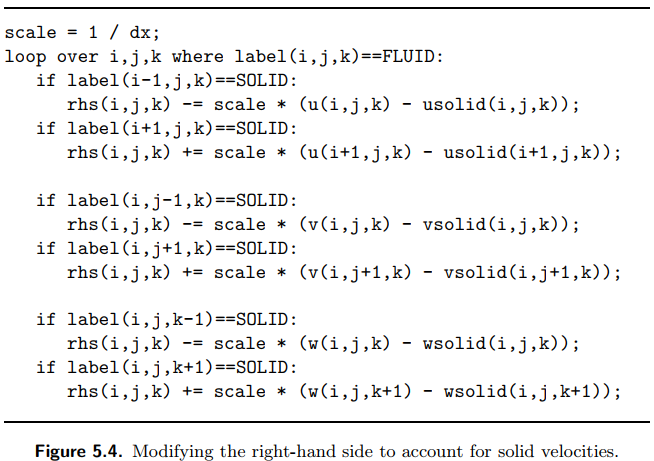


注意公式(5.6)和(5.8)是泊松问题.

如果流体网格单元位于边界处,请记住,边界面上的新速度涉及流体外部的压强,我们必须通过边界条件来定义这些压强:我们需要在此处使用它.例如,如果网格单元是空气单元,则我们将方程(5.6)中的替换为零.如果网格像元是固体单元，则将替换为根据边界条件计算的值,如公式(5.3)所示. 假设和是流体单元,这将使方程简化为以下公式:



我们可以观察到有关此示例的一些总体信息，以及如何使我们在代码中实现它。 首先，对于气室边界条件，我们只需从方程中删除该p的提及即可。 其次，对于实体单元边界条件，我们删除了对p的提及，但也将pi，j前面的系数减小了一个-换句话说，pi，j前面的系数等于非实体的数量 网格单元邻居（在三个维度上都是相同的）。 第三，我们用一个涉及流体和固体速度之差的项来增加在右侧测得的负散度。 可以在代码中将其实现为修改rhs的附加循环，如图5.4所示。



5.3.1 矩阵向量形式

现在,我们为未知的压强定义了一个大型的线性方程组.我们可以从概念上将其视为一个大系数矩阵A,乘以一个包含所有未知压强的向量p,该乘积等于一个包含每个流体网格单元中的负散度的向量b（在固体壁边界处进行适当的修改）:

到目前为止，在实现中我们已经讨论了，当然，和在逻辑上存储在二维或三维网格结构中，因为每个元素对应于一个网格单元.

我们不需要将A直接存储为矩阵。 注意，A的每一行对应一个方程，即一个流体单元。 例如，如果网格单元（i，j，k）是流体，那么将有一行矩阵可以用索引（i，j，k）标记。 该行中的条目是该方程式中所有压力未知数的系数：几乎所有这些都是零，除了可能对应于pi，j，k及其六个邻居pi±1，j，k，pi的七个条目外 ，j±1，k和pi，j，k±1。 （当然，在二维中最多有四个邻居。）我们只有非零（i，j，k）及其流体单元邻居。 存储所有零当然是没有意义的：这是一个稀疏矩阵。