**步骤（画流程图）**

**总体**流程：（每小点为一步，顺延；如果出现迭代，以iter steps x—x表达）

程序开始：读取所有输入；

1. 计算模型里的每像素距离，以得到通用的距离值。如果有多个可行区域和相对应的distance信息，则需要全部计算后，取平均。
2. 2D结构向3D结构转变。细分步骤如下：
   1. 整体2D矩阵变为3D张量（此时未包括横梁区域）
   2. 对每个横梁，把横梁信息并入3D张量里。

注：如同计算Pixel\_dis，人手输入的横梁信息，同样需要“前续检查”，以及同样可能并非墙到墙。

1. **冲突检测及管道排列算法1**
   1. 后续更多的算法可以插入此部分，得到多种结果。
2. 输出报告
   1. 第一版
      1. 输出图纸：3D全图，加一些关键处的截面图（包含了结构，空间和排好的管线）；关键处的截面图是为了标识碰撞，对碰撞区域圈出来
      2. 一些简单的语句，不需要复杂，如：哪条管线穿过横梁
      3. 同一个模型，5个左右的case（对别对应不同的rules集），包括从
         1. 完全没有冲突的
         2. 有冲突到排不下的
      4. 围着一圈的走廊，2个case
3. 后续人工选择
4. 除了给定的PDF中的结构，还需要一些功能提供用户自己选择或添加可行和非可行区域，然后再iter steps a—d，更新输出报告
5. 提供一些功能是用户可以选择需要那些rules，不需要那些rules，输出对应的报告iter steps c—d。

**冲突检测及管道排列算法1**流程：（每小点为一步，顺延；如果出现迭代，以iter steps x—x表达）

1. 首先考虑GraFlow，对其里面的每一条管道，iter steps 1）– 3）

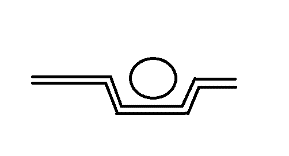
* GraFlow起点一定是最高的；
* 所有的管都是能排高就排高

1. 检测该管道是否符合要求：
   1. 15度要求

此检查需要分2部分：首先认定此管道均运行于横梁底；如果不能满足，则检测是否可以在起始处穿横梁；由此，确定管道的起点flag：起点flag： 0/1表示起点应于横梁底或横梁上（可以穿横梁）。

* 1. 30m要求：不能直线或者15°斜线超过30米，需要穿插垂直管道（具体问协兴）。如检查完可行区域后不能满足要求就照排，返回提示就行
  2. 其他要求？

1. 管道排布于3D空间：
   1. 读取可行空域
   2. 对从“起点🡪转折点🡪终点”中的每一条线段：iter steps c) – d)
   3. 计算管道所占的三维cube空间结构
   4. 计算与墙壁间距离（rules），与天花间距离（rules），与其他管道间距离（rules），与其他管道群间距离（rules），按照“起点🡪转折点”的下一个转折方向，将管道贴边放置。注：如果碰到分层需要小管避大管，如下图所示

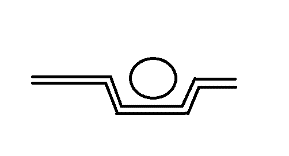


* 1. 输出管道实际三维空间（包括点到点，和cube结构信息）
  2. 更新可行空域（StrucWorkArea的更新）

1. 联合检查：
   * 1. 假设有管线N条，至此已全部排放完毕，对1🡪N条：iter steps b) – d)
     2. 从更新可行空域去掉此条的信息，并将此新的可行空域定义为可行空域2
     3. 于可行空域2重新排放管线
     4. 检查是否一致，如不一致，更新结果。
2. 其次考虑MVAC，对更新可行空间去掉所有水管的信息，在新的可行空间中对其里面的每一条风管，iter steps 1） – 3）

* 风管，在不与重力流管道冲突的情况下，能排多高就排多高，排不了就放下面。

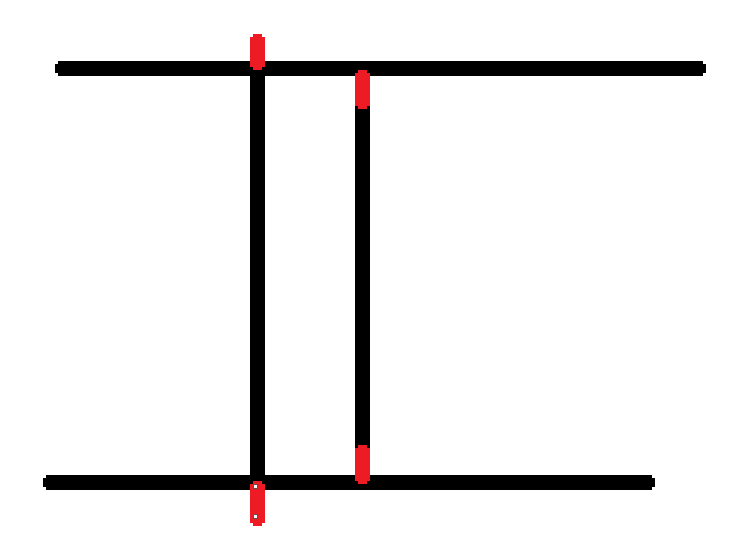
1. 检测该管道是否符合要求：
2. 检测风管的形状，面积不能变，宽和高中，短边比长边，不能小于1：6。如果风管形状不满足此条件，改变风管的形状大小，使其满足条件，如果仍无法满足条件，就照排，最后提示并返回冲突发生的区域
3. 管道排布于3D空间：
4. 读取可行空域（排完水管后的可行空域）
5. 对从“起点🡪转折点🡪终点”中的每一条线段：iter steps c) – d)
6. 计算管道所占的三维cube空间结构
7. 计算管道所需安装空间（rules），墙壁间距离（rules），与天花间距离（rules），与其他管道间距离（rules），与其他管道群间距离（rules），按照“起点🡪转折点”的下一个转折方向，将管道贴边放置。注：如果碰到分层需要小管避大管，如下图所示



1. 输出管道实际三维空间（包括点到点，和cube结构信息）
2. 更新可行空域
3. 联合检查：
   * 1. 假设有管线N条，至此已全部排放完毕，对1🡪N条：iter steps b) – d)
     2. 从更新可行空域去掉此条的信息，并将此新的可行空域定义为可行空域2
     3. 于可行空域2重新排放管线
     4. 检查是否一致，如不一致，更新结果。

按照流程图，需要写的程序包括：

1. 计算Pixel\_dis
   1. 输入：Stru\_WorkArea，enSegment（实际长度），(x\_seg1, y\_seg1)，(x\_seg2, y\_seg2)
   2. 输出：LenPixel
   3. @少武
   4. 输入输出限制：
      1. 在前续检查中，对Segment进行了限制，如下：
         1. 如果可行区域沿xy轴，则Segment至少有x或y轴的取值要一致。
         2. 反之，可行区域在平面图上展示时是斜的，则在计算时，需要沿斜线进行计算。
         3. 根据两点坐标构成一条线，计算线的解析式，结合走廊的解析式，看这个线是否和走廊是否成90度
   5. 如何计算？
2. 在b部分中，我们已经通过调整两点的坐标使得两点组成的线跟墙成正交状态。所以我们只需要延长或者缩小线段的长度来计算走廊间的像素数量。
3. 在Segment上进行两点连线得到一个线段，检测这条线段是否因为太长穿过了走廊还是因为太短没接触到走廊，对线段进行缩短和延长，使得线段刚好接触到走廊，如下图，红色为缩短和延长的部分。



1. 根据公式得到LenPixel



1. 2D结构向3D结构转换
2. 输入：StructWorkArea，LenPixel，High\_info（高度信息组合），Beam
   1. 输出：Stru\_WorkArea\_3D
   2. 计算
3. 计算可行域最大高度h\_max:

h\_max = h\_floor - thick\_floor - h\_false - thick\_light

1. 计算StructWorkArea3D的Z轴方向像素个数Num\_Pixels\_Z：

Num\_Pixels\_Z = h\_max / Len\_Pixel

据此将StructWorkArea由2维拓展到3维，大小为( 2D\_X, 2D\_Y, Num\_Pixels\_Z)，即在Z轴方向（第三维度）增加Num\_Pixels\_Z -1 个(2D\_X, 2D\_Y)大小的2D矩阵。新增元素用255填充，命名为StructWorkArea3D\_temp。

1. 计算横梁下可行域高度h\_with\_beam:

h\_with\_beam = h\_floor - h\_false - thick\_light - depth\_beam

1. 计算横梁深度（扣除楼板度）对应像素个数Num\_Pixels\_Beam：

Num\_Pixels\_Beam = （Depth\_beam - thick\_floor）/ Len\_Pixel

根据Num\_Pixels\_Beam和梁的位置信息Beam将StructWorkArea3D\_1中梁所在的区域的像素值改为127，该张量命名为StructWorkArea3D。

1. 重力排水管的特殊要求检查
2. 输入：所有重力流管道的数据：路劲Route，尺寸Size，管线的类型type，
3. 输出：重力流水管是否超过30米，是否满足15°倾斜角要求，返回纠正后的管线路劲，如果无法纠正则做flag提示。
4. 计算：
5. 检测30米要求

* 计算管线段长度是否超过30米，若超过则需要往下垂直转弯200mm（暂定）。
* 根据此规则，在需要下弯处增加一个转弯点，加入到重力流的Route中
* 管线段长度公式为：



1. 检测15度倾角的高度要求

* 计算管线在15度倾角的要求下，所需最小空间高度要求：



* 判断是否超过可行域高度：如果 ，报告冲突；

增加option选项，将高度延长至，使得工作得以继续。

1. 管线排布
2. 输入：重力流管线路径Route，尺寸Size，管线的类型type，管线包裹材料，管线所需空间，Stru\_WorkArea\_3D
3. 输出：包含重力流管线的3D空间矩阵Gra\_Stru\_WorkArea\_3D
4. 计算：
5. 管线测绘

* 各管线的包裹材料、预留空间并入管线的Size中，测算管线在3D Models中的像素位置，改变StructWorkArea3D中的相关元素。
* 排水管用63表示，风管用191表示。
* 先写入所有重力排水管，再写入风管。
* 重力排水管贴着从天花板出发按15度倾角向下排布
* 风管在允许状况下尽量往高处排布

1. 冲突检测与报告

* 设置一个标志位flag，用来标记冲突类型。
* 在Models生成写入元素时，若该位置原值为63（排水管）或191（风管），报告管线碰撞，flag=1；
* 若该位置原值为0，报告结构碰撞，flag=2；
* 若该位置原值为127，则是穿过了横梁。若该管线尺寸小于250mm，则顺利穿过；否则报告横梁碰撞，flag=3。

1. 尝试简单调整

* 当flag=1（管线碰撞）：

1. 若是平行管线部分重叠：

尝试左右平移管线，隔出距离，判断冲突是否消失。

1. 若是管线正交：

若规则允许，在正交处将其中一条增加下拐及上拐接头规避碰撞，检测冲突；

或者尝试将其中一条移到下一图层，判断冲突是否消失。

* 当flag=2（结构碰撞）：

一般管线与墙或结构平行，若出现这种情况，应该是管线中心点过于靠近结构。

尝试将该管线外可行域方向挪动直到结构碰撞消失，检测是否产生其他冲突。

* + - * 当flag=3（穿过横梁）：

规则允许情况下，增加下拐接头从梁下走线，判断梁下空间h\_with\_beam是否满足空间要求，判断是否产生新冲突。

1. Weak & Strong Rules

在3)中原则上尽量遵循所有Rules。

Strong Rules一定要遵守，当无法遵守时在报告中描述；

Weak Rules尽量遵守，但在冲突无法解决时可以不遵守Weak Rules并在报告中描述。

软件相关：

1. 确认要使得程序正确检测需要保证1mm内至少要有多少个像素
2. 需要提供软件的误差范围
3. 提供让用户可以自己选择可行或不可行区域的功能
4. 提供让用户可以自己选择删除或者添加那些rules的功能