

空间句法简明教程

Space Syntax: a very short introduction

目录

绪论.....	6
名词解释.....	8
第一章 空间建模.....	9
第一节 凸空间.....	9
第二节 Link 与 Unlink.....	14
第三节 空间关系的拓扑学呈现.....	17
第四节 拓扑深度的加总.....	20
1) 在 DepthMapX 中的操作.....	21
2) 空间深度的建筑学意义 (1).....	23
3) 对 Total Depth 的进一步处理.....	25
第五节 尽端空间的专业化.....	30
本章小结.....	33
第二章 视线分析.....	34
第一节 网格设定.....	35
第二节 视线分析的软件操作.....	39
第三节 Visibility Relationships.....	42
1) 深度的定义.....	43
2) 视线整合度.....	47
3) Visual Clustering Coefficient.....	54
本章小结.....	57
第三章 轴线建模.....	58
第一节 用 DepthMapX 自动生成轴线图.....	59
第二节 用轴线图研究城市.....	61
第三节 手绘轴线图的方法.....	66
第四节 局部模型的建立.....	71
本章小结.....	73
第四章 轴线模型的计算与分析.....	74
第一节 验证轴线模型的正确性.....	74
第二节 在 DepthMapX 中处理连接关系.....	77

第三节 Integration Core	80
第四节 Synergy	84
第五节 选择度	87
第六节 Ctrl+D.....	89
本章小结	90
第五章 线段模型的计算与分析	91
第一节 线段模型的生成	92
第二节 角度分析的软件操作	93
第三节 角度基础上的整合度和选择度	95
第四节 再谈整合度	99
第五节 街网密度的问题	101
第六节 通过 Properties 对话框查看计算结果	104
本章小结	105
结语	107
参考文献	109

绪论

空间句法是一种由建筑师开发出来的描述和分析空间的数学方法。

回顾全部的建筑历史，我们会发现，西方人的建筑观念，是有特色的。建筑空间的专业化，是西方人的特色。

我们中国是院子，院子里面，几间上房，几间厢房，这更多的是血缘身份的直接映射。阿拉伯人是帐篷，有了房子以后也是帐篷的那种感觉，所有人都挤在一个空旷的房间，每个人占据墙角的一块。印度人由于炎热的气候的影响，基本上是室外活动，建筑物更多的是用来看的。总之，只有西方人，是针对建筑空间，进行了功能定义的，从而使得某个建筑空间的使用，具有了单一性、排他性，从而实现了建筑空间的专业化。

伴随着这样的特色，也给西方人带来了不少问题。

西方的建筑师面临的是这么个问题：你在图上标好，这间是餐厅，这间是厨房，这间是什么什么，到了将来建成使用的时候，会按照图上标的来么？经常出现的情况是，建成使用的情况，与建筑师在图纸上估计的，是不一致的。小到住宅，大到城市设计，都面临这个问题。想的跟实际的对不上。

西方人的思维，是逻辑中心主义的。他在图纸上标好的某个房间的使用功能，是逻辑思维过后才决定的。当实际使用的情况与图上标的不相符的时候，证明建筑师预测错了的时候，他要寻找背后的根源。他们得到的结论，也是空间句法的整个理论的基石，是这么一句：

空间本身不重要，重要的是空间之间的关系。

其实这个话在政治课上也听过，马克思说，“人是社会关系的总和”，套用到空间句法这里，可以说，“空间是空间关系的总和”。

重申一下，在以功能为核心关切的话语环境中，对于某个具体的空间，发挥什么功能，并不是由这个空间的本身决定的，而是由这个空间与其他空间之间的关系决定的。这个“关系”是最本质的。在功能主义的话语环境中，建筑空间没有本质，建筑师可以把握的，仅仅是空间之间的关系。

离开了“空间 --- 功能”的话语环境，如果你要探讨的是建筑的艺术品质方面的问题，那么空间句法的应用，就不那么直接了，因为判断标准是不同的。

就像同一个苹果，分别给两个人尝尝，一个评价说“非常好吃”，另一个可能就不会认可。“苹果好吃”这样的判断句，其**逻辑有效性**是成问题的。对于建筑的艺术品质方面的探讨，就经常被这个逻辑有效性的问题所纠缠。

空间句法是数学方法的集合，空间句法是依赖数学方法对空间关系进行抽象和建模分析的，空间句法也是依赖数学方法来寻找空间与五彩缤纷的社会活动之间的相关关系的。因为其逻辑基础是有局限性的，所以空间句法探讨的不是空间的艺术品质的问题，空间句法探讨的是与功能有关的那些问题。

一个理论，在实践中如何应用，其应用的范围限于什么之内，是很大的问题。

空间句法已经做成软件，研究人员只要按照既定的工程方法，在软件中正确地描述了空间系统的状态，软件就可以自动地对空间之间的关系进行计算了。在针对空间本身的分析的基础上，还可以对于社会活动进行采样，比如人流量、车流量、小商店的布局等，把这些空

间本体之外的因素，给考虑进来，在各种变量之间进行相关性分析。这样就推动“空间 --- 功能”的互动关系的这个研究领域，往前迈出大大的一步。

因为建筑学的视野当中，许多东西都是不稳定的。人的感觉是不稳定的，人在空间当中的活动是不稳定的，空间承载什么功能，在中长期来看，也是不稳定的。再加上审美的、文化的因素的飘忽不定的影响，加上艺术传统与工程可能性方面的反复角力，使得建筑学当中假定性的东西、虚拟性的东西，与物理上真实的东西，生活中真实的东西，都缠绕在一起。如果所有变量都具有不稳定性，那这个话题就没法谈了。

空间句法的好处是，数学建模的方法是稳定的，数学计算的结果具有唯一性，这就使得我们对于空间关系的考察，具有稳定性。以空间分析的结果，作为某种稳定的参照物，我们就可以对各种其他的不那么稳定的研究对象，进行较为系统全面的考察了。

在空间句法的领域内，国内外已经形成了大量的研究成果，可是直接去读西方人写的原文，往往存在困难，这个话题是怎么来的，解决问题的思路是怎么一步步走下来的，得到的结论又该如何深刻把握，某一项已经完成的研究又揭示了哪些更为广阔的可能性，这些事儿，往往使得中国的读者，汪洋大海一样的，心里总是没底。西方人在空间句法这个领域所展现出来的怀疑的精神，理性的立场，中立的态度，与国内建筑学的专业教育所反复灌输的那些东西，也存在着深刻的裂痕。想要进入空间句法这个领域的读者，往往要花很大的时间精力，跟自己较劲。

本文的目的就在于，搭建一个较为快速的知识通道，花尽量少的篇幅，帮助读者尽快地进入到空间句法的这个话语体系中来，对空间句法有个整体上的把握。如果读者认同空间句法的这个方法论，那么本文的目的就在于，使得空间句法的这个体系，更快地发挥工具的作用，读者有了这个工具，再去研究自己感兴趣的问题。如果读者不认同空间句法的这个系统，那么进来把空间句法的局限性了解清楚，之后就没有必要再浪费时间。

本教程的篇章结构大致如下：

名词解释

第一章 空间建模

空间句法当中，对于空间的理解和概括，发展出了几种不同的数学模型。有一种是“凸空间”，通过这种规则，可以把实际的建筑空间，转译成为由凸空间组成的系统。还有一种是“轴线图”，可以按照一套既定的规则，用直线去概括空间，将空间转译成为由一些直线组成的系统。在轴线图的基础上，通过 DepthMapX 这个软件，还可以自动生成“线段模型”。

三种模型对应的英文是：

- 凸空间： Convex Map
- 轴线图： Axial Map
- 线段模型： Segment Map

第一节 凸空间

凸空间的空间，都是二维平面，是以实际的空间状态为底本，降了一维。

假设一个空间内部，任意两点之间可以互相看得到，那么这就是一个凸空间。所谓任意两点之间互相看得到，对应的英文是“all see all”，参见下图：

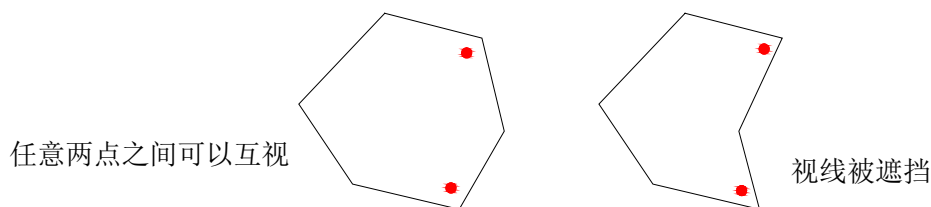


图 1.1-1 凸空间的定义

不是凸空间的，可以将其劈开几瓣，变成由凸空间组成的系统。三角形是边数最少的凸空间。将来运算的时候，只要保证参与运算的所有元素，都是凸空间，那么在数学上就算定义清楚了。

空间是空间关系的总和，在凸空间的建模方法下面，考察的就是凸空间之间的关系。参见下图：

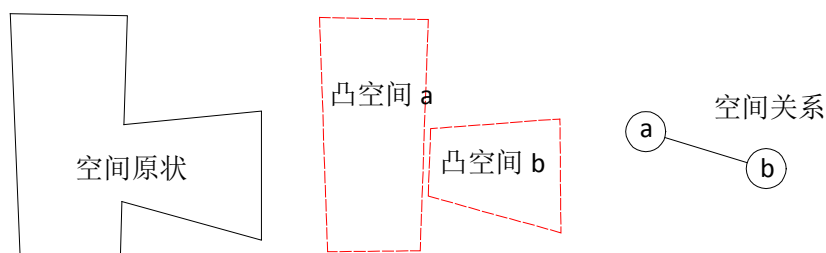


图 1.1-2 凸空间之间的关系

习惯上，我们把参与运算的凸空间，称为元素。假如一个空间系统，是由二十个凸空间组成的，我们就说，这个空间系统中有二十个元素。

不是凸空间的，要将其划分成由凸空间组成的系统。因为三角形是边数最少的凸空间，在计算机科学当中，就可以将平面空间简化成三角网，当两个三角形共边的时候，表示两者有连接关系，互相直接可以走得通，不需要经过别的元素中转。这种算法早已成熟。

建筑学当中，较多时候处理的是矩形空间，矩形本身就是凸空间，已经满足要求，没有必要弄成三角网了。

假设有一个建筑物，要进行空间句法分析，首先可以将其平面图转换为由凸空间组成的空间系统，并进行空间关系的设定。如下图所示：

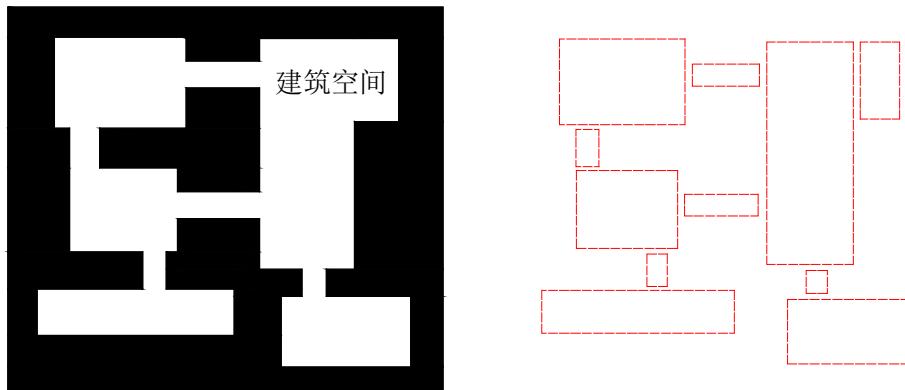


图 1.1-3 将建筑空间转译为凸空间的组合

上面右边这个图，并未定义凸空间之间的连接关系。连接关系的定义，是在 DepthMapX 当中手工进行的。

一般的操作步骤是：在 AutoCAD 中，将凸空间用闭合的多边形做好，通过 dxf 格式的文件，导入到 DepthMapX 当中，直接识别成 Convex Map 就是了，在此基础上，再手工设定连接关系。这个操作步骤，有的时候会出问题，手工点击凸空间的时候，就识别不上，选不中就无法操作。遇到这样的问题，可以尝试选择的顺序变一下，原来是先点击 a 物体，后点击 b 物体，现在反过来试试。如果还不行的话，回到 CAD 当中，有问题的那个多边形删掉，重新画一次，再另存为 dxf 格式，再重新导入，可能就解决了。

在 CAD 当中，导出 dxf 文件的时候，一定要清理干净，不要出现 z 轴方向上重叠的线段、长度为零的线段啥的，只保留闭合的多边形，干干净净地导出来。

DepthMapX 这个软件，软件本身体积很小，功能简单，都是按照既定套路操作。难点不在于软件，而是在于对算法的把握，难在对于计算结果的分析 and 诠释。请读者放宽心，不必对软件产生忧虑。

DepthMapX 这个软件，发展历程从上世纪九十年代就开始了，功能模块逐渐丰富起来，老的代码保留较多，尚未用现代编程工具彻底重写过，这就造成软件架构较为杂乱，使用起来稳定性一般。现在该软件的官方程序员是 Tasos Varoudis，源代码也开放了，在代码托管平台 Github 上可以查到。源代码是 C++ 写的，所有功能模块，都是通过函数库的形式提供的。目前该软件已经发布了 Mac 和 Linux 的版本，进步还是很明显的。

本文写作的时候，使用的是 Win8.1 系统，DepthMapX 的详细版本号是 DepthMapXnet-0.30-windows，软件并不需要安装，双击文件夹中的 exe 文件，就可以用了。

打开 DepthMapX 后，从窗口左上角的那个 File 那里，左键单击一下，展开下拉菜单，通过“File→New”，创建一个新的工作空间。接下去，点击“Edit→Import”，可以将已经准备好的这个 dxf 文件导入进来。如下图所示：

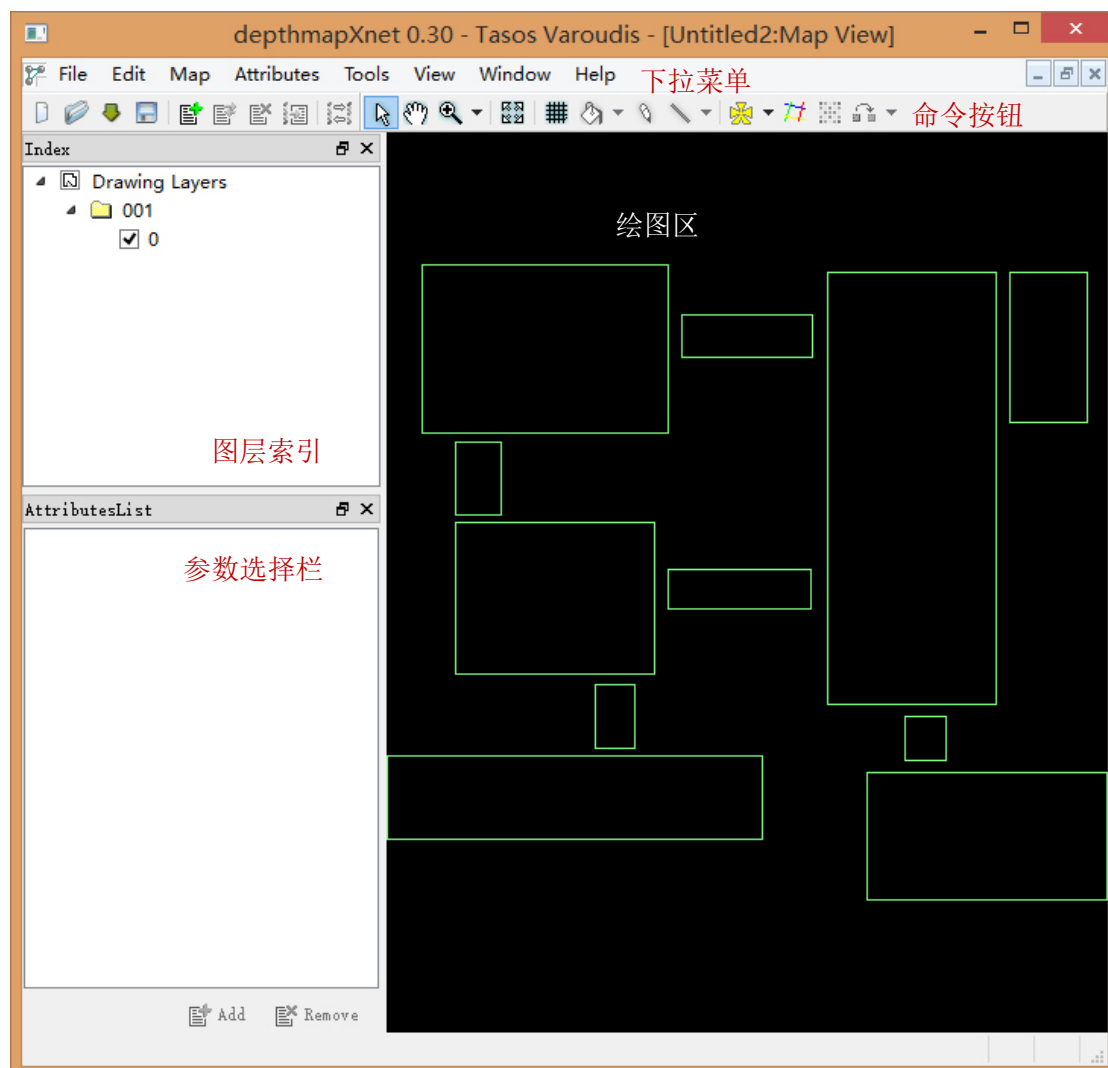


图 1.1-4 导入 dxf 文件

与 Windows 平台下的其他程序一样，当下拉菜单展开的时候，可以同时看到那些命令的快捷键。如下图所示：

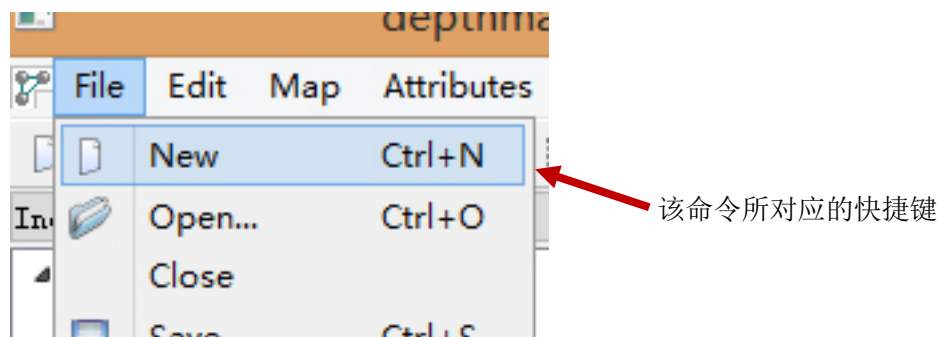


图 1.1-5 快捷键

上文介绍的这个导入 dxf 文件的步骤，也可以通过 Ctrl+N(新建工作区)，然后 Ctrl+I(导入)的快捷方式来实现。

刚导入进来的 dxf 文件，还不能直接用，需要将其中的内容指定成 DepthMapX 认得的元素格式。从窗口顶部的下拉菜单那一行，找到 Map，左键单击展开，找到 Convert Drawing Map 这个命令，左键单击弹出对话框。Convert 这个单词的意思，就是转换。在对话框中，问你将来想操作哪个类型的元素，本节主题是凸空间，我们就在弹出的对话框的 New Map Type 那里，展开下拉列表，选中 Convex Map 就行了。如下图所示：

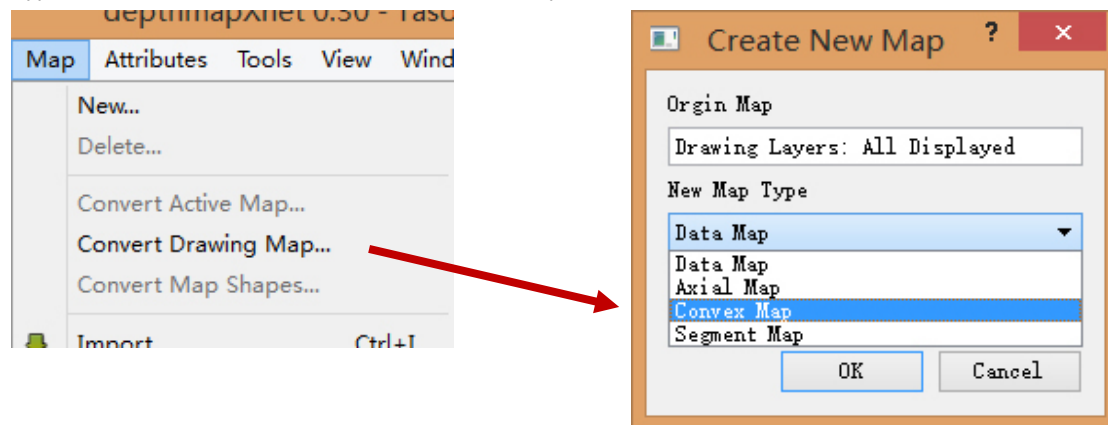


图 1.1-6 Convert to Convex Map

上图中的这个对话框的其他选项，在后面的章节中，会逐步涉及到，到时再详细介绍。

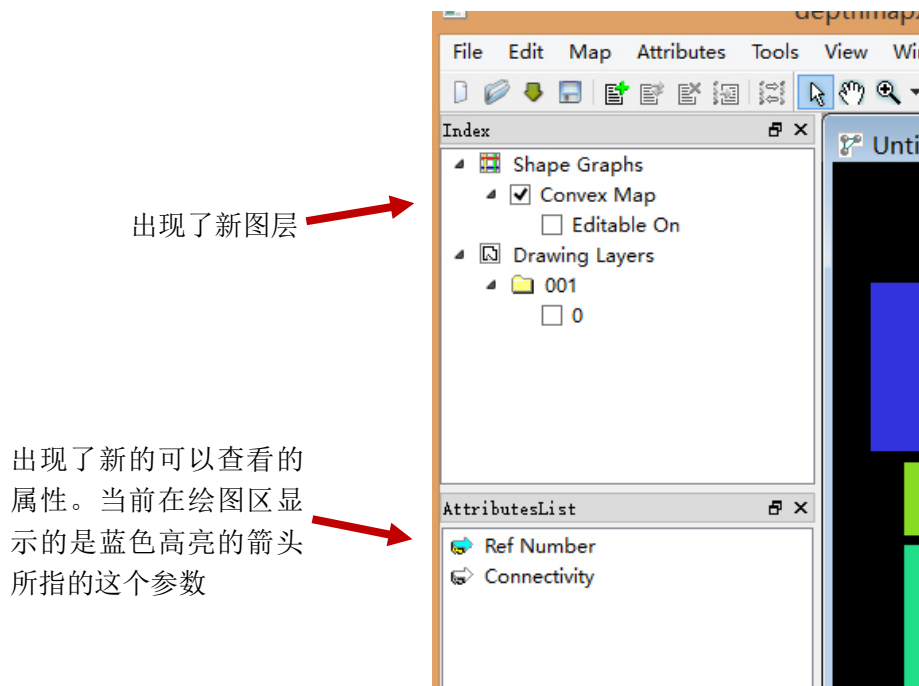
对话框设置好以后，点击 OK，这时绘图区的状态就发生了变化了，原来的绿色线框变成了色块了。这是因为，DepthMapX 对于每一个单独的元素，都要单独赋予名称，赋予某个 Reference Number，将来计算结果反馈回来，就可以按照这个序列号进行对应，将计算结果对应到每个元素头上。在绘图区显示出来的时候，就按照数值高低，赋予每个元素特定的颜色。

DepthMapX 默认的显示规则是，如果是彩色色块，表示这个元素头上有数值；如果是灰色的色块，表示它处于 No Value 的状态。

空间句法采用的是先对元素进行分类，然后按类赋予颜色的显示策略。也就是说，每个元素头上不是记着个数值么，把全系统中的最高值与最低值找到，然后在这个数值区间内，等分成十个数值段，具体的某个元素，落在哪个数值段内，就与本段内其他元素，接受相同的颜色。越是数值低的，颜色上越偏冷，越是数值高的，颜色上越偏暖。红色代表最高的数值段。

类似这样的显示策略，在 ArcGIS、Ansys 等软件中也有，是常用的方法。当然，等分成十个数值段，红色代表最高值，这都是软件的默认的设置，读者将来也可以手动调整。

Convert 成了 Convex Map 以后，我们来观察一下窗口左侧的图层索引和参数选择栏那里的变化，如下图所示：



出现了新图层

出现了新的可以查看的属性。当前在绘图区显示的是蓝色高亮的箭头所指的这个参数

图 1.1-7 界面的变化

图层的概念，在 CAD、PS 中都会涉及到，读者应该不会感到陌生，在 DepthMapX 中工作习惯也差不多相同，具体干什么事儿之前，一定要先确定好图层的状态。

上图中图层索引那个地方，Convex Map 前面那个方框要是取消勾选，这个图层就不显示了。Convex Map 前面的方框处于勾选状态，那么绘图区就会显示这个图层。下面有个 Editable On 的选项，默认不勾选，如果勾选上了，可以在绘图区进行某些编辑操作。Editable On 这个地方没勾选的状态下，只能看不能编辑。

在窗口左下角的参数选择栏，可以通过左键单击，改变蓝色高亮的箭头所指的具体内容。具体执行了某个计算的命令以后，得到的结果，会出现在参数选择栏中，而蓝色高亮箭头所指的那个参数，就对应于当前在绘图区可以看到的内容。

再看 DepthMapX 程序窗口的右下角，显示了当前这个系统的一些信息。如下图所示：



系统中的元素总数

图幅的长宽尺寸

鼠标指针当前所处的坐标

图 1.1-8 窗口右下角显示的系统信息

Convert 这一步做完了以后，就算是顺利地把素材导入进来了。下面一节，介绍 link 和 unlink 命令，通过这两个命令，可以设定空间之间的连接关系。这个 link 和 unlink，不但在凸空间的操作中会涉及到，在轴线模型和线段模型之间，也会用到。

第二节 Link 与 Unlink

在 CAD 中，按照上节中所介绍的思路，将空间的实际状态，转译成由凸空间组成的系统以后，导入到 DepthMapX 当中，Convert 成 Convex Map 之后，下一步，就是手工设定空间的连接关系了。

在空间句法的建模当中，主要关心两个问题：

- ┌ 空间是否存在的问题；
- └ 空间之间的连接关系是怎样的？

做历史研究的时候，史料有限，在还原空间的历史面貌的时候，就是个大问题。要确定空间是否存在，就比较困难。就算是以现存的建筑空间为研究对象，也有不少困难。空间本身的存在状态，要落实清楚。

然后就是关心空间之间的连接问题。两个空间要么是直接连接，directly linked，要么是通过其他空间过渡，indirectly linked。在空间建模的时候，连接关系的设定，也要花费不少精力。

接着上节的内容，导入 dxf 文件，并转换为 Convex Map 之后，到 DepthMapX 窗口顶部的命令按钮那一行，找到 link 命令。如下图所示：

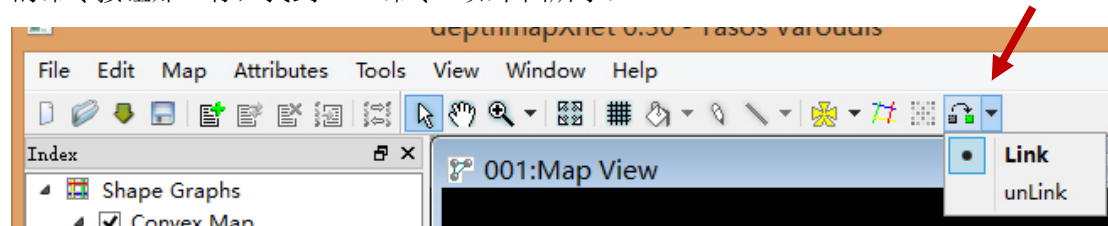


图 1.2-1 Link 命令

通过点击图标右侧的黑色三角形，展开下拉列表，设定这个图标的新的默认值。读者可以多点击几次，切换 Link 和 Unlink 的状态，多试试。左键单击这个 Link 命令以后，绘图区会变成暗色调，提示操作人员，现在进入了连接关系的设定状态。左键单击任意一个元素，它会变成黄色高亮的色块。在 DepthMapX 当中，只要某个元素是黄色高亮的状态，就表示它已经被选中，不论是凸空间的模型，还是将来的轴线模型、线段模型，都是这样。DepthMapX 也支持通过框选的方式，同时选中多个物体。在进行 Link 操作的时候，不支持多选。

在 Link 操作的时候，先左键单击选中一个元素，再选中另一个元素，这时两个元素之间就出现了一条绿色的线，连接了两个凸空间的重心，这表示两者的连接关系已经确立。如下图所示：

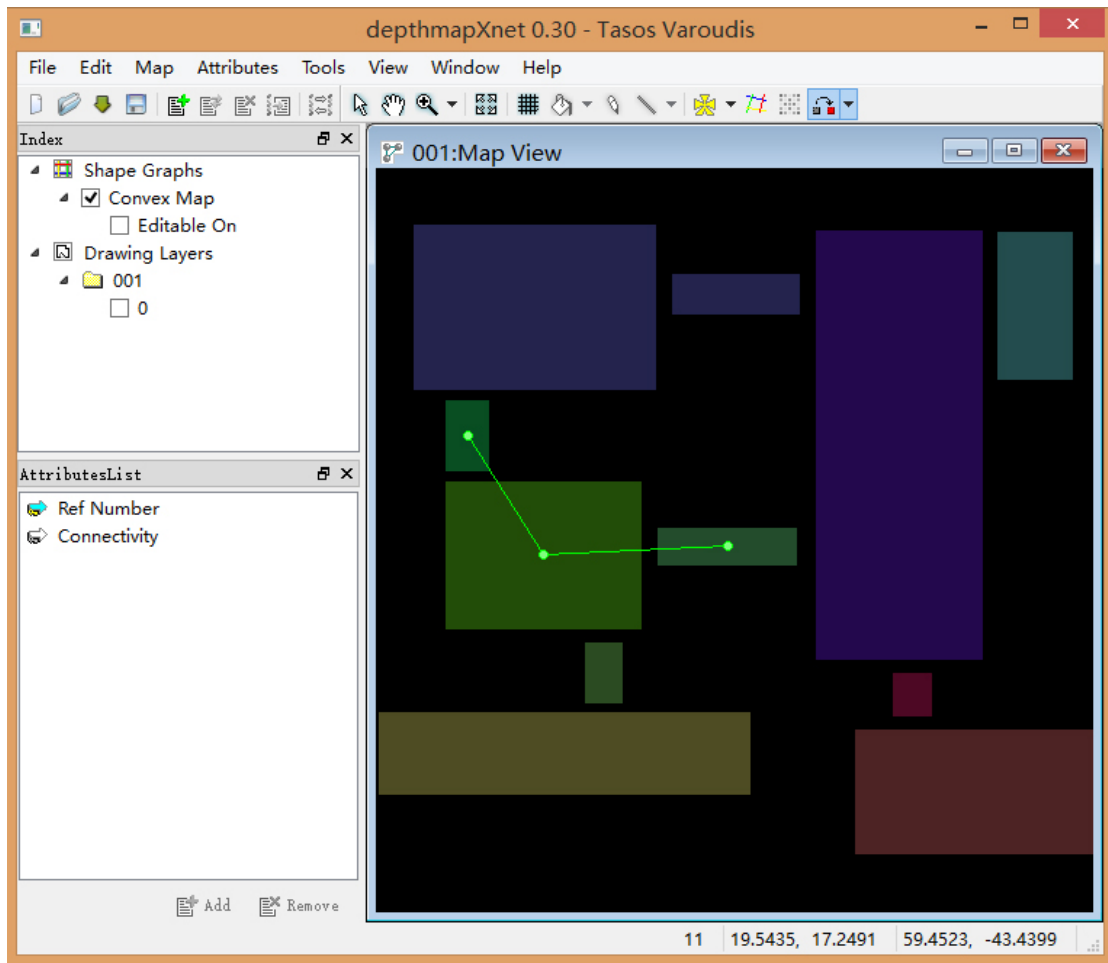


图 1.2-2 Link 操作

Link 命令操作完成以后，要退出编辑状态，可以通过命令按钮那一行的左边的 Select 按钮完成。左键单击一下箭头形状的 Select 命令，就退出编辑状态了。如下图所示：



图 1.2-3 Select 命令

本节所涉及到的这个假想的建筑空间，全部连接关系都设定好以后的状态，如下图所示：

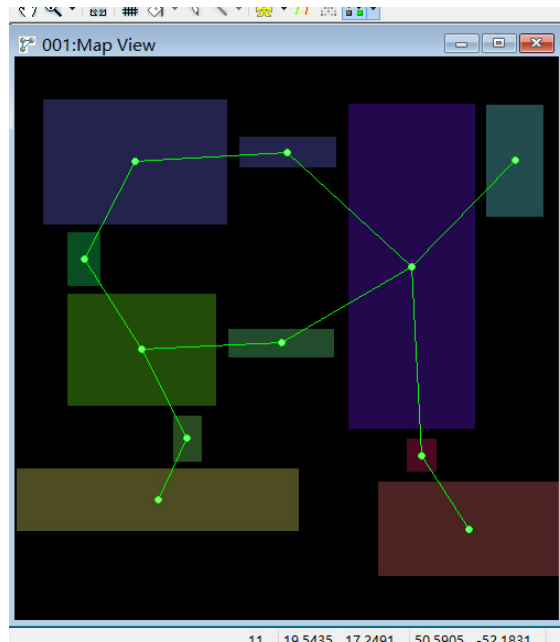


图 1.2--4 连接关系的设定

在实际的研究案例中，一定会遇到多层建筑的情况，那么，上下层之间的空间链接，应该如何处理呢？一种较为粗糙的处理办法是，把所有平面图都并列在一起，将来用 Link 命令将交通核那部分连接起来就行了。如下图所示：

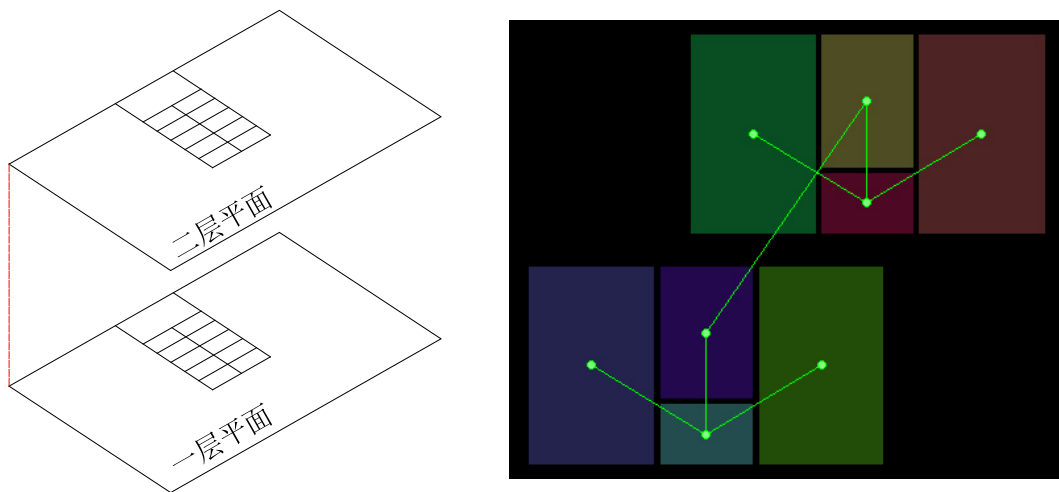


图 1.2--5 多层空间的连接关系

上面这种处理方法，并不令人完全满意。假设一种空间连接是上下楼的那种连接关系，另一种是相同标高上的连接关系，在人的感觉上，这两种连接关系不能等同。上图这样的处理办法，是默认所有的 Link 都是匀质的，这与我们的主观感受不相符。

处理的办法还是有的，对空间的连接关系赋予不同的权重，类似这样的话题，在空间句法的圈子中，也一直在探讨着。读者先理解和掌握这种最简单的办法，暂时就够了。

全系统的连接关系都设定好以后，就通过 Select 命令，退出编辑状态，就可以进行下一步的分析计算了。

第三节 空间关系的拓扑学呈现

重申一下空间句法的基本结论：空间本身不重要，重要的是空间之间的关系。

空间句法用了拓扑学的方法，来研究空间之间的关系。还是前面的那个假想的建筑空间，我们把凸空间用圆圈表示，空间之间的连接关系用直线表示，可以得到：

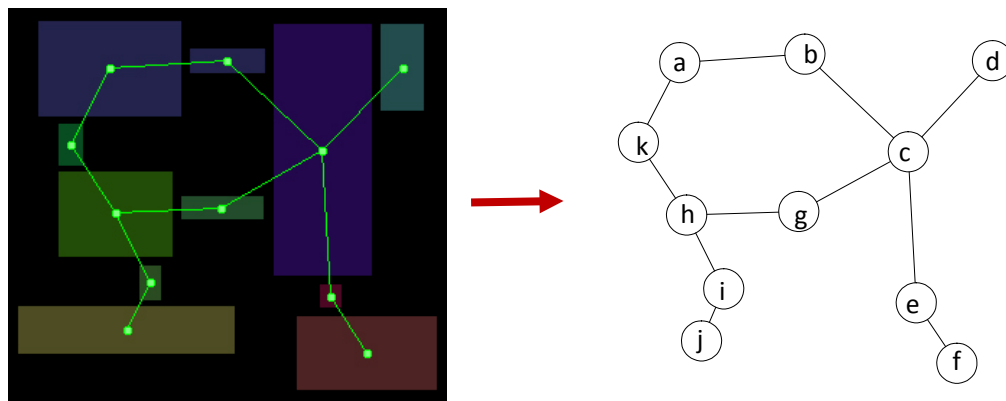


图 1.3--1 空间关系的拓扑学表达方式

虽然从空间句法的视角来看，空间的本体并无意义，但是我们依然要为每个空间命名。者并不是建筑学意义上的命名，这件是厨房，这间是餐厅，这间是什么什么，而是为了将来计算结果反馈回来，要分别记在每个元素头上。DepthMapX 当中对于空间的命名，就是操作数据库的时候，为每个元素赋予流水号的那种感觉。

上面右边这张图，每个圆圈的位置，都放在凸空间的重心处了，算是保留了一些空间的信息。为了称呼方便，把上面右图的这种表达方式，称作“带 GPS 坐标”的状态。

因为实际的建筑空间，被抽象成凸空间，又抽象成了圆圈，而我们要研究的是空间之间的关系，那么，可以对上面右图进行更进一步的处理，将 GPS 的信息，可以丢掉不管，拓扑图可以重新绘制成许多种形式。重绘之后，与实际的建筑空间仍然对的上，因为空间之间的连接关系没有改变。参见下图：

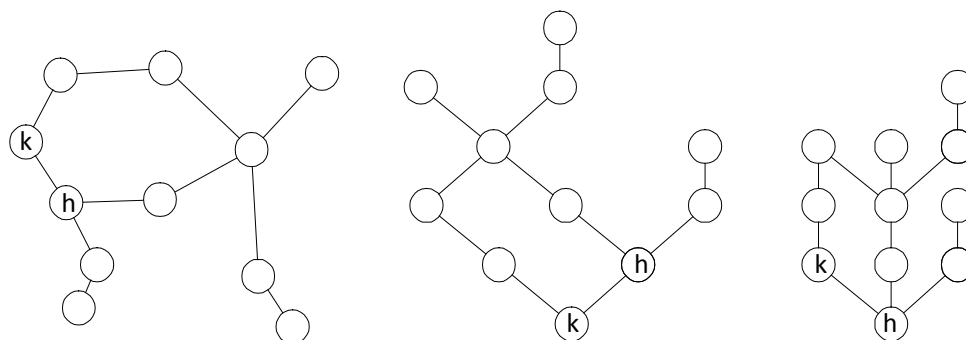


图 1.3--2 拓扑图的调整

把空间系统的 GPS 坐标的信息丢掉，只考察其拓扑学意义上的连接关系的话，可以把拓扑图重绘成许多种状态，称为“空间重映射”，英文是“Relationship Remapping”。我们可以从系统当中任意选取一个元素，放在最下面，从它开始，按照连接关系一步一步往上画，这样，空间关系会变得较为简单直观。如上面右边两个图那样。

习惯上，把空间重映射之后，最下边的这个元素，称为中心（也有的英文文献里面，把最下面的这个空间称为“root”，本文中就不采用 root 的叫法了）。上图中，中间那个图，可以称为“以 k 为中心进行空间重映射”，右边那个图，可以称为“以 h 为中心进行空间重映射”。

在许多情况下，并不需要将全系统都纳入研究范围，而只关心距离中心空间一定范围以内的情况，那么做了空间重映射以后，就比较简单直观了。

在上图中，h 和 k 的关系，叫做 h 和 k 相距一个拓扑步数。因为空间关系是相互的，所以从 h 出发到 k 是一个拓扑步数，从 k 出发到 h 也是一个拓扑步数。这个拓扑步数，英文单词是 step，也有人习惯用 depth 的，说 h 与 k 相距一个 depth。

现在如果只关心局部的空间结构，比如，只关心以 h 为中心，两个拓扑步数之内的情况，那么就如下图所示，将红线以上的部分全部丢掉，不参与计算就是了。当面对元素数量很多的大系统的时候，把不必要的那些东西给裁掉，是很自然的。如下图所示：

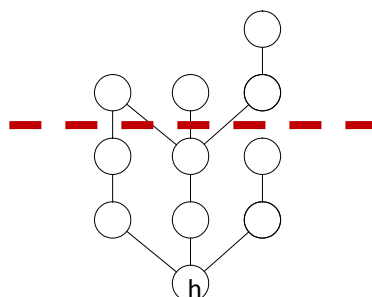


图 1.3--3 拓扑图的调整

日常生活的经验告诉我们，有的小街小巷，分布着一些小卖铺，只服务于附近的一些居民，而城市当中的某些繁华的街道，具有全局性的影响力，能够吸引全市的人来这里消费。小商铺在城市当中星罗棋布，数量众多，而大型的商业街区，全市也没有几处。有了拓扑学的这套考虑问题的办法以后，我们从微观的一些局部系统的状态，一步步放大范围，直到宏观的、全系统的状态，就都能够清晰地加以界定和研究了。

以某个空间为中心，进行空间重映射以后，我们把红线慢慢往上抬，红线以上的都裁掉不要了，只考虑红线以下的部分，可以截取出不同尺度的拓扑结构的片段，把这些截取出来的局部，都罗列在一起，进行一下横向比较，就会很有看点。

在空间句法的讨论中，经常可以听到“半径 3”、“半径 n”的说法，意思就是多少多少步数以外的，不参与计算。半径 n 表示不限定步数，全系统都参与进来。

再啰嗦一下，空间重映射的这个方法，比较要紧。针对同样一个空间系统，以不同的元素为中心，分别进行空间重映射的话，得到的拓扑图是不同的。也就是说，从不同的元素眼中看出去，看到的是不同的空间结构。这已经暗示我们，空间之间是有差别的，其差别不是来源于空间本体，而是来源于空间关系。

下面摆一张图上来，这个图片来自第五届空间句法大会的论文集(Thaler, 2005, p. 330)，用了空间重映射的方法，去考察希腊的一个古城。

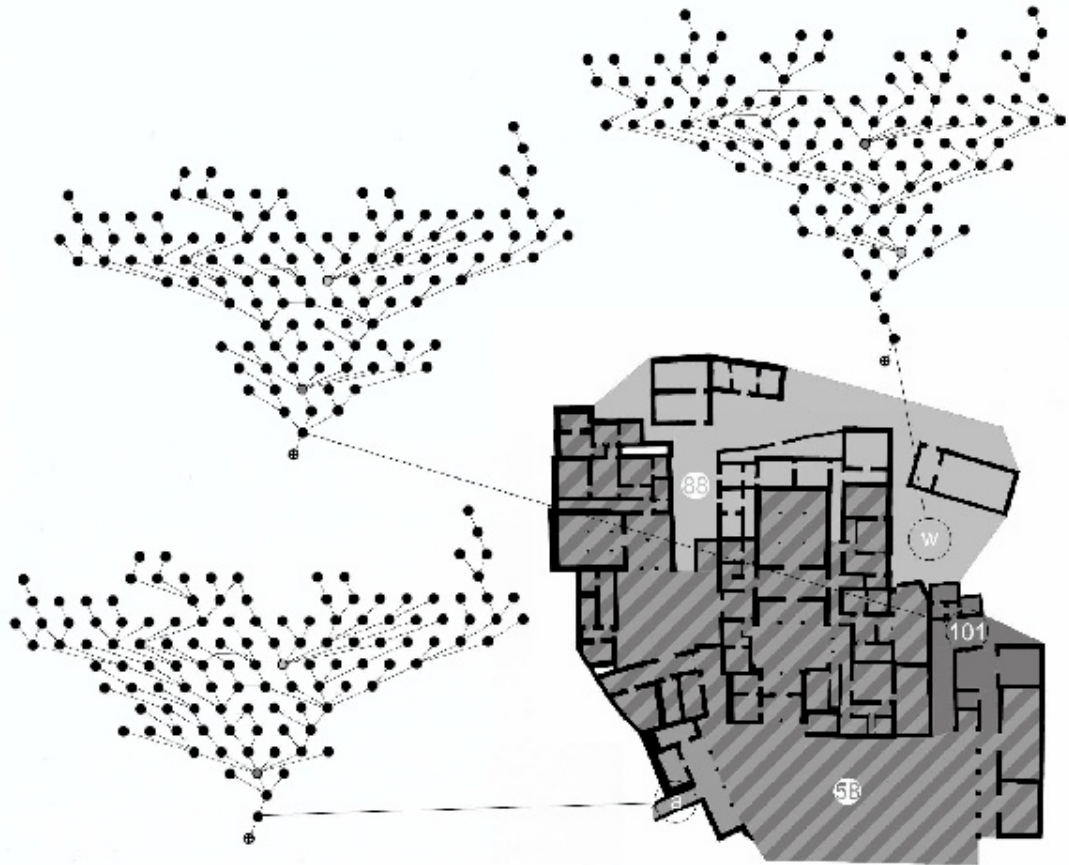


图 1.3--4 案例研究

上面这张图，就是考察 Pylos 古城的几个主入口。分别以几个主入口为中心，进行空间重映射，得到的不同的拓扑结构。在此基础上，还可以继续深入。

本文中，只是希望展示一下拓扑图的感觉，无意进行真正意义上的学术考察。读者先熟悉熟悉这种感觉，就行了。

在后面的章节中，会逐步展开，介绍一些算法，按照这些算法，对拓扑图进行了计算以后，计算结果反馈回来，记在中心空间的头上，再通过数值比较的方式，可以更加深刻地考察空间之间的差别。

通过空间句法这种比较数学化的方式去理解和考察建筑，会带来许多新的知识。空间句法这一套理论体系，影响越来越大，这里边的术语、行话越来越通用，读者也许会慢慢发现，建筑学当中越来越多的东西，是来源于数学的。这是在用数学改造建筑学呀。

第四节 拓扑深度的加总

空间句法这里边的绝大多数算法，都是在弄“全系统加总”。首先从数学上定义好，两个元素之间，具有了怎样怎样的关系，就参与计算；然后，按照这种算法的安排，把全系统所有的可能情况都算一遍；最后，进行全系统加总，反馈回来的数值，跟谁对应的，就记在谁的头上。绝大多数算法都是按照这个套路写的。

前面的章节中提到，拓扑步数和拓扑深度是同一个概念，两个元素相距一个拓扑步数，也就意味着两个元素之间有一个拓扑深度。接下来就可以介绍一下“全局拓扑深度”的概念，英文是 **Total Topological Depth**。这个全局拓扑深度，意思是说，在空间重映射以后，从中心空间出发，到系统当中任意一个其他元素，最短的拓扑步数是几步？从中心空间出发，穷尽了系统中所有的其他元素以后，穷尽了所有可能性以后，把每种情况的拓扑步数都加在一起，这个数值反馈回来，记在中心空间的头上，就是这个中心空间的全局拓扑深度了。

为了说明全系统加总的这种思路，不妨再从“选择度”的定义上来看一下。选择度，英文是 **Choice**。选择度的定义是，出现在最短拓扑路径上的次数。也就是说，在空间重映射之后，除去中心空间之外，系统中任意一个元素到任意另一个元素，最短的拓扑路径是怎样的？中心空间是否出现在这个最短路径上？穷尽了所有可能的情况以后，中心空间出现在最短拓扑路径上的次数，一共是几次？这个值反馈回来，就是中心空间的选择度的值。

为什么空间句法对于建筑学的现象，所给出的解读，是靠得住的，就是因为这个“全系统加总”，就是因为这个“穷尽了所有可能性”。数学上穷尽了所有的可能性，在真实的生活当中，所遇到的情况，肯定是被考虑进来了的。

在空间句法之前，在建筑学的范围内，要做什么实证研究，只能用采样的办法，比如随机选取三千位市民，记录他们的出行轨迹，在一定的样本的基础上，再尝试着发现一些规律。这样的办法不行，因为这样的数学模型，做不到“穷尽所有的可能性”，有很多隐藏的因果关系，无法在采样而来的数据中，得到揭示。

西方人的思维，是逻辑中心主义的，他们做空间句法的研究，就是要保证，这个逻辑链条延伸出去，一定是环环相扣，每一环都彻底靠得住的。逻辑有效性受到质疑的，一定要找到出问题的根源。从 **Bill Hillier** 写的那个黑皮子的《空间是机器》就能体会到，围绕着相同的问题，反反复复地论证，为了保证逻辑上能成立，假设了各种情况，假设之上还可以再加假设，细节之下还可以再分，看得我们眼睛都蓝了。

我们中国人的思维，是那种诗性的、概括性的思维，找到一点儿靠得住的，就希望弄成放之四海而皆准的，对于自己认为对的东西，感情极其真诚，态度极其热烈。后来，法国的那个解构主义哲学家，德里达，来中国演讲，直接下结论说，“中国没有哲学”，说得我们瞠目结舌。

在空间句法的基础上，要重新检视我们过去所认为的对的东西，要重新验证那些早已默认为的合理性，这是要跟自己较劲的。

1) 在 DepthMapX 中的操作

还是上文中的那个假想的建筑空间，通过 dxf 格式，导入到 DepthMapX 中，用 Convert 命令，将其转换成 Convex Map 以后，就可以计算全局深度了。

在 DepthMapX 窗口最上部，下拉菜单那一行，找到 Tools，然后通过 Tools→菜单中的 Axial/Convex/Pesh→左键单击 Run Graph Analysis，弹出对话框。如下图所示：

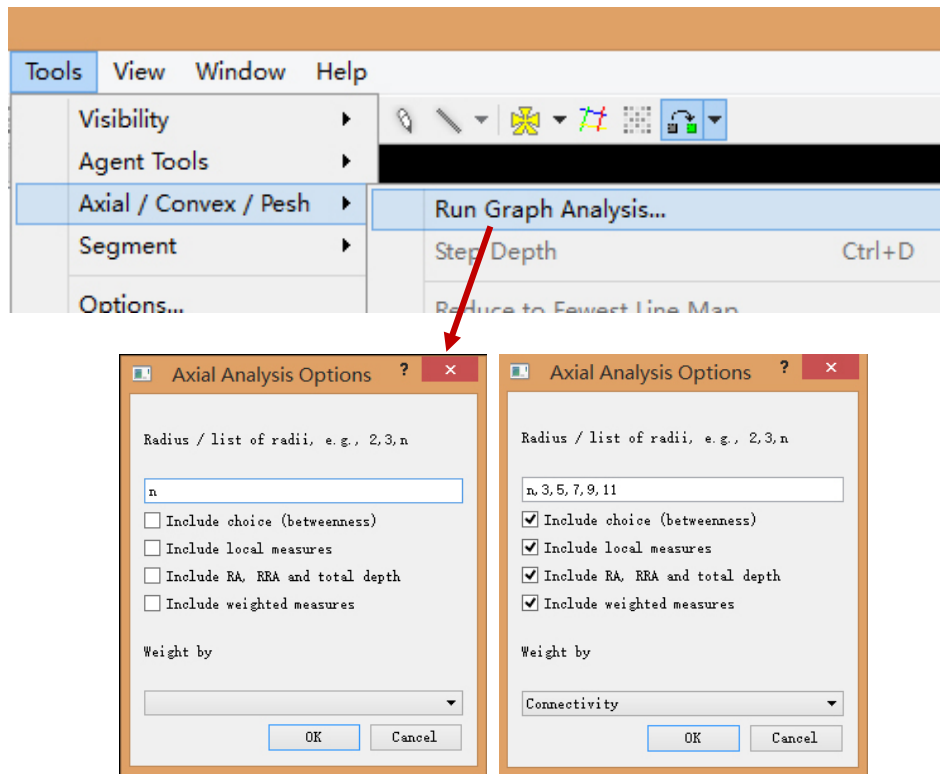


图 1.4-1 凸空间分析的参数设定

弹出的这个对话框，要调一下参数，而几乎每个案例做到这里，设置都差不多。对于初学者来说，这个对话框的设置，直接记牢了就行了。调完参数的情况，如上面右图所示。

对话框中，最上面填数字的地方，意思是设定拓扑半径的。输入的时候，数字之间用英文输入法的逗号隔开，将来计算的时候，以某一个元素为中心，进行空间重映射以后，分别计算全系统、半径 3、半径 5.....的值，计算结果反馈回来，记在中心空间的头上。要把全系统中的每一个元素，都作为中心空间计算一遍，每个元素头上都有对应的数值，才能结束。计算结果其实是一个数据表，在软件中也可以通过数据表的方式进行查看，在后面的章节中会介绍到。

对话框中部，有四个方框，可以勾选的。前面三个勾，任何情况下都勾上，意思是，计算过程中的某些中间步骤，要保留的。第四个勾要是打上了，下面 Weight by 的下拉选项，才能继续进行设定。如果第四个勾没有勾选，对话框最下部的下拉列表是灰色的，不能操作。

这个 Weight by，就是以什么什么为权重，是对已有的计算结果的一种修正。有的时候，算法不够周到，可以通过在每个元素头上，乘以一个经验系数的方式，将计算结果修正一下。这个权重的设置，就是一种经验系数的选择。在本例中，因为尚未进行任何计算，数据表还是一片空白，只能选 Connectivity，就选它了。

对话框设置好以后，点击 OK，看看什么情况。

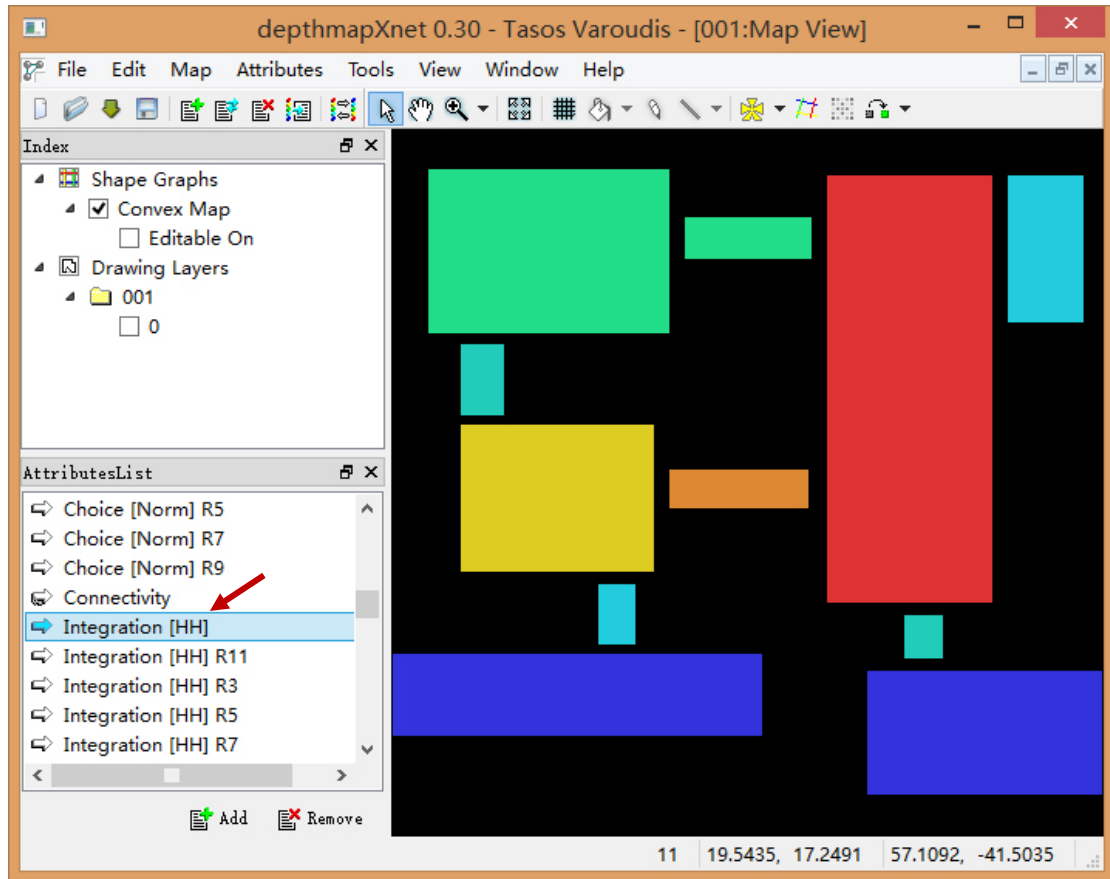


图 1.4-2 查看计算结果

计算完成以后，DepthMapX 程序的窗口，左下角的那个参数选择栏当中，多了不少东西。而且蓝色高亮的箭头，自动跳到 Integration[HH]上面了。我们在这个参数选择栏当中，可以拖动右边的这个上下滑动的滑块，向下翻，找到 Total Depth 这个参数，左键单击一下，在程序中的绘图区，色块的状态就会发生变化。在参数选择栏当中，蓝色高亮的箭头在哪，右边的绘图区的色块，就显示与之相对应的信息。

有的时候会误操作，鼠标中键无意中滚动了一下，色块就不知道飘到哪里去了。为了使图像居中最大化显示，我们可以点击命令按钮的那一行，点击 Recentre View 的那个按钮，色块就找回来了。如下图所示：

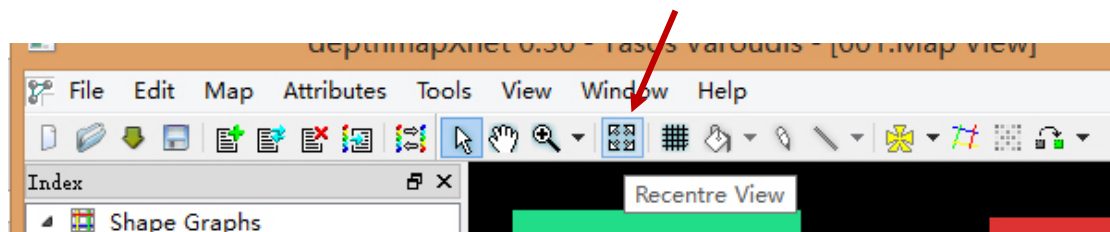


图 1.4-3 居中最大化显示

先从左边的参数选择栏，选中 Total Depth 这个参数，使得绘图区显示 Total Depth 的状态，再把鼠标指针放到随便哪个色块上，稍微停一会儿，鼠标指针的附近会蹦出来一个数字，这个数字就是该色块的 Total Depth 的值。

读者可以先找个蓝色的，冷色调的色块，看看它的 Total Depth 的值是多少，再找个红色的，偏暖的色块，看看它的 Total Depth 的值是多少，回忆一下，这个偏冷和偏暖，当初是怎么定的？

2) 空间深度的建筑学意义 (1)

在上面的案例中，全局深度的结果计算出来以后，我们可以将数学模型和生活现象加以对比。

城市当中的活动，是有成本方面的制约的。比如，一群人要去参加个什么活动，面临的问题是路途遥远，交通不便，总的交通成本太高，花这么大的代价，觉得划不来，就不去了。当成本过高的时候，城市活动就不能发生了。

生活在 18 世纪的德国古典经济学家杜能，讲过一个故事，说，有人驾着马车去运送草料，在运送的途中，马儿也要吃掉一些。如果运送的路途过远的话，半路上马儿就把草料都吃光了。可见超过某个时空极限，经济活动不能发生。

超过了一定的时空极限，城市活动就无法开展。

我们眼中看到的城市，这个物质实体，只是城市的一个外在的表象。而城市的本质，在于生产、消费和交易活动的地理上的集中(赵红军, 2005)。

所有的城市活动，都会伴随着某种成本的付出，包括人们的时间和精力的付出。在地理上集中起来，可以影响人流物流信息流在空间中的分布，把巨量的城市活动压缩在有限的区域内，这样能够极大地节省成本，提高效率。

我们国家优秀的经济学家赵红军，系统性地提出了“交易效率”的研究框架。所谓的交易效率，就是说，一个人花了一块钱去买东西，所得到的实际价值，只有 k ，这个 k 是大于零小于一的一个数。 k 是交易效率， $1-k$ 的部分就是交易成本。这就是把过去经济学上的交易成本，变成了一个比率的概念。如果我们顺着这个思路往下走，那么通过这个 k 值，不但可以衡量经济活动的效率，也可以衡量社会活动的效率。

城市本身是动态演进的，城市所处的自然地理环境，具有约束性。但是人们通过不断改善交通和生活方面的基础设施，不断改善交通工具，不断发展运输技术，可以使得城市的硬件条件大大改善，促使交易效率大大提高。另一方面，人类进入了信息社会，通过信息化的建设，社会开放程度日益提高，城市当中的教育水平、人力资本方面的投入大大提高，这也促使社会的法制化程度越来越高。软件条件的改善，也能大大提高交易效率(赵红军, 2005, p. 55)。参见下图：



图 1.4-4 影响交易效率的因素

从赵红军提出的这个框架来看，城市化的推动因素，可以从硬件和软件两个方面加以综合考虑。

如果社会处于强权肆虐的状态之下，人们为了保护自己的生命财产安全，为城市空间设置一定的“深度”，是必要的。典型的如中国古代的都城，一直有重城体制，大城套小城，一层套一层，层层把关，进出都需要特权。如果社会处于和平稳定的氛围中，是有法治的基础的，人们的生活是有理性的基础与法律的保障的，那么，城市就变成了市民平等竞争、自由追逐财富的地方，那么，城市的空间结构，就要追求效率了，就是要提高这个k值。

如果城市当中的某个街区，处于 Total Depth 的值较低的状态，就意味着，从这个地方出发，到达系统中任意的其他街区，需要克服的距离成本，是比较少的。城市活动如果发生在 Total Depth 的值较低的区域，需要的成本可能较少，而人们得到的实际价值，可能占比更大。这样，我们就从理论上，把城市空间，和生活现实的某些方面，建立起了某种联系，可以用一个因素去解释另一个因素。

本文中的假设是，Total Depth 的值越低的，越有利于提高交易效率。至于实际上是不是这样，还是要结合实际情况去证明的。可以大胆地假设，然后再小心地求证。

并不是因为设计师在图上标了，这个地方是商场，将来这个地方就一定能够吸引到商业活动。而是因为我们掌握了商业活动的一些较为本质的规律，可以合理地做出逻辑判断，将来在某个特定的地方，商业活动会较为活跃，在此基础上，我们才推动工程项目往某个方向发展。

如果我们考察最近十年来城市生活所发生的变化，那么，软件条件的变化是最引人注目的。在信息社会成型之前，软件条件的改善，要依赖于城市空间结构的。你不是要提高人们的教育水平么，那好，你要设立足够多的学校。你不是要提高银行的竞争程度么，那好，你要散布足够多的金融网点。到了现在，类似这样的要求，都可以通过加强互联网服务来满足。城市的物理意义上的空间，已经失去了中心地位。城市活动甚至渐渐地与城市空间结构发生了脱钩。在人类历史上，这是从未发生过的现象。

城市规划到了最后就是个土地利用的问题。我们应该警惕软件条件所引发的深刻变化。

因为土地利用的落实，要靠工程技术来一步步做出来的。工程技术的实施是有周期的，工程的结果是有滞后性的。现在正在进行的工程，是许多年前就已经做出决定的。在某个时间点上，软件条件已经导致了城市发展的趋势性的变化，而工程的实施，还是在按照许多年前定下的方向在走，这可能造成巨大的问题。

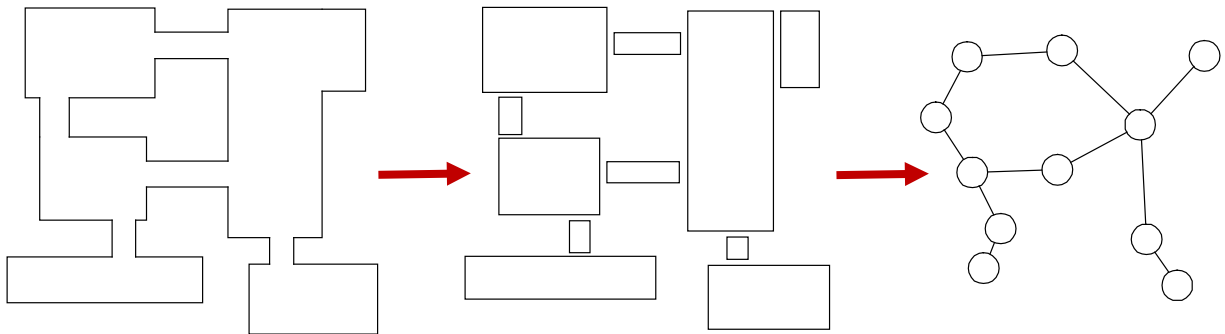
3) 对 Total Depth 的进一步处理

将一个拓扑结构拿来，以其中的某个元素为中心，进行了空间重映射，求出了这个中心空间的 Total Depth 的值，全系统每个元素都做计算一遍，最后形成了一个彩色的特征图，这个结果直接拿来用，还是存在问题的。

拓扑学的这个词，是当初莱布尼兹提出来的。莱布尼兹和牛顿是同时代的人物，都是十七世纪的人物。经过了这么长时间的积淀，在拓扑学的内部，早就走得很远了，不可能局限于拓扑深度加总这么初级的水平。要进一步地做数学处理，把许多干扰性的因素给考虑进来，一步步地剔除掉干扰因素，最后达到某种可靠的状态。本小节的内容，在黑皮子的那个《空间是机器》的 55 页左右，也有，读者可以看一下西方人对这个问题的阐述。

为了使得读者彻底了解现在所要解决的问题，必须要回到开头的地方，重新梳理一下。首先是把建筑空间，用凸空间的方式，给概括一下；然后，把由凸空间所组成的这个系统，要抽象成拓扑图；然后，人为地定义好拓扑深度的算法，并以某个元素为中心，空间重映射，计算一遍；最后，计算得到的值，反馈回来，记在中心空间的头上，直到全系统中的每个元素都这样干了以后，结束。

这个流程下来，就是形成了一张数据表，每个元素头上都记着各自的 Total Depth 的值。在绘图区显示的时候，按照十个数值段等分一下，将来元素显示的时候，就显示这个数值段的颜色。如下图所示：



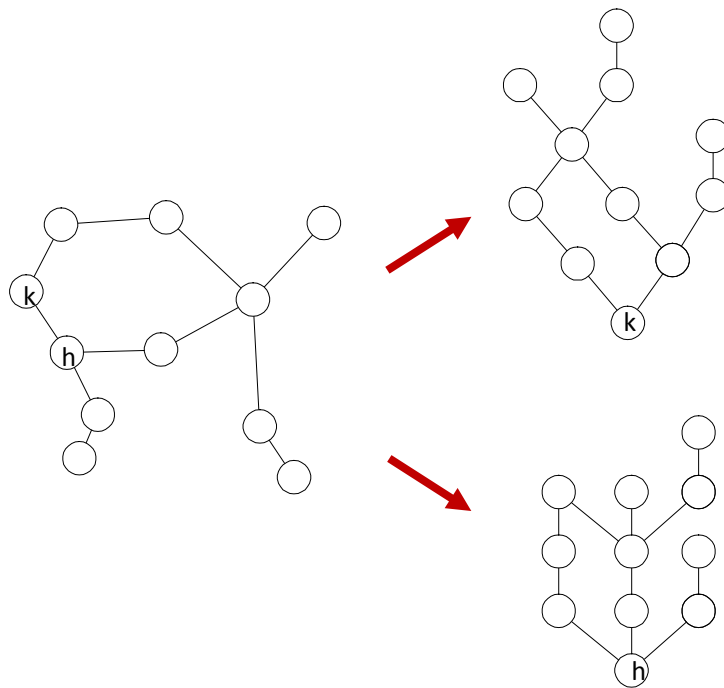


图 1.4-5 Total Depth 的值

通过软件给出的结果，其实手算一下也来得及， $Total\ Depth(k)=27$ ； $Total\ Depth(h)=22$ 。

现在 k 和 h 两个元素，全局深度值是不一样的。这两个元素处于同一个系统中，系统中的元素总数是确定的，全局深度的值出现差异，就是因为这两个元素头上顶着的拓扑结构，是不同的。

本节后面将一步步地介绍，如何将拓扑结构的个性给剔除掉，留下共性的东西，以便两个元素直接进行数值比较。

为了说话方便，我们假设有一个系统，称为 A，里面的元素，从 a_1 开始编号，这个系统中一共有 n 个元素。现在针对 a_1 元素，已经算出了 $Total\ Depth(a_1)$ 。

首先，可以把拓扑结构中元素数目的影响，给剔除掉。参见下面公式：

$$Mean\ Depth(a_1) = \frac{Total\ Depth(a_1)}{n} \quad (\text{公式 1})$$

因为从数学的定义上，我们是考察 a_1 与系统中“所有其他元素”之间的关系，除去它自身，系统中还剩下 n-1 个元素，因而分母应该变成 n-1。

$$Mean\ Depth(a_1) = \frac{Total\ Depth(a_1)}{n-1} \quad (\text{公式 2})$$

这一步好理解么？减去 1 的意思是，不能把自身与自身的关系也算进去。当初 Total Depth 在计算的时候，就考察的是中心空间与其他空间之间的关系。

第二步，要考虑一下对称性的问题。假设两个空间系统，具有相同的元素数目，不同之处在于，一个系统的拓扑关系是严格对称的，另一个是非常不对称的，如下图所示。我们来考察一下两者的 Total Depth 的特征。

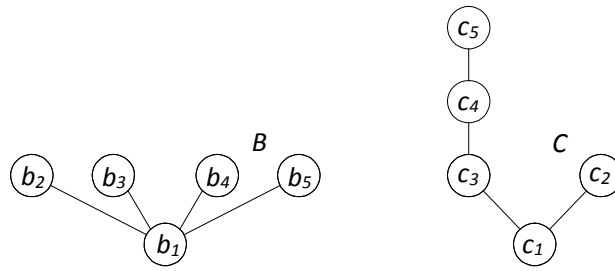


图 1.4-6 对称性差异造成的问题

对于 B 系统来说，在 b_1 的眼中，系统完全对称， $Total\ Depth(b_1) = 1+1+1+1 = 4$ ，把上面那个公式 2 搬过来， $Mean\ Depth(b_1) = 4/(5-1) = 1$ ；对于 C 系统来说，在 c_1 的眼中，系统是非常不对称的， $Total\ Depth(c_1) = 1+1+2+3 = 7$ ， $Mean\ Depth(c_1) = 7/4 = 1.75$ 。可见拓扑结构的对称性问题，对于空间深度的影响，很显著。可以通过某种方式，将对称性的影响因素剔除掉。

假设有个系统 D，现在以 d_1 为中心，进行了空间重映射，并已经求出了 $Mean\ Depth(d_1)$ ，可以通过下面公式这样，处理一下，把不对称的程度给剔除掉。如下面公式所示：

$$Relativized\ Asymmetry(d_1) = \frac{Mean\ Depth(d_1) - 1}{\frac{\text{该系统的元素总数}}{2} - 1} \quad (\text{公式 3})$$

上面这个公式，意思是， d_1 头上顶着的这一个拓扑结构，很可能不是完全对称的。我们假设一个新的系统，其元素数目与 D 系统的元素数目相同，而其拓扑结构是完全对称的。这样，用 d_1 头上顶着的这个空间结构，与完全对称的结构，进行比较，就知道 d_1 头上顶着的这个系统，不对称的情况有多么严重了。

也就是说，在 $Mean\ Depth$ 的基础上，不论实际情况如何，再强制性地除以系统中元素数目的一半，来剔除因不对称而造成的影响。asymmetry 这个单词的意思是，不对称。习惯上把这个 Relativized Asymmetry 简称为 RA。

D 系统中的每个元素，都可以算它的 RA 值。

上式中，为什么分子分母都要减去 1，这个不太好理解。还是用这个逻辑来解释吧：空间关系考察的是中心空间与所有其他空间之间的关系，通过减去 1 的操作，把中心空间自身从参与运算的集合当中剔除出去。

把上面的公式 3，推向极限，如果以某个元素为中心，进行空间重映射以后，发现这个元素的 RA 值为零，那么在这个中心空间的眼中，拓扑结构完全对称，说对了吗？

在 RA 的基础上，还可以继续。

主要是还存在这么个问题：前面公式 1、2、3 中，分母上都是与系统的元素数目有关的东西。拓扑结构除了有元素数目上的差别，还有连接关系的差别。如下图所示，两个系统具有相同的元素数目，可是中心空间的 Total Depth 还是存在差异的。

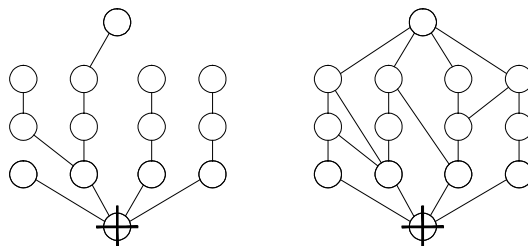


图 1.4-7 连接关系的差异性造成的问题

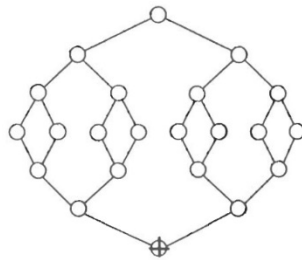
为了把连接关系造成的影响剔除掉，我们可以找一个有特色的拓扑结构，作为一个具有稳定性的样品，让所有的其他的结构与之进行比较。

理想的拓扑结构是这样的：在这个结构中，不论以哪个元素为中心，进行空间重映射，所得到的拓扑结构都是一样的。也即，这个理想的拓扑结构中，所有元素的 **Total Depth** 都是一样的，所有元素的 **Mean Depth** 都是一样的，所有元素的 **RA** 都是一样的。这是一个非常匀质的空间结构。

限制条件还没完。这个理想的拓扑结构，它的构成法则必须非常简单，这样，我们往里面加任意多个新的元素，所得到的新系统，仍然能够保证，以任意一个元素为中心，进行空间重映射以后，所得到的拓扑结构是相同的。

什么样的拓扑结构是合适的呢？

最后找到这么个结构：



它被称作钻石形（**Diamond-Shaped**）拓扑结构。这个玩意的 **RA**，可以被记作 **RA of Diamond**，计算公式如下：

$$\text{RA of Diamond} = \frac{n\{\log_2\left(\frac{n}{3}\right) - 1\} + 1}{\frac{(n-1)(n-2)}{2}} \quad (\text{公式 4})$$

上式当中的 n 表示钻石型结构当中包含的元素数量。仔细观察一下公式 4，就会发现，这实际上是个单变量的公式。也即，只要符合钻石形拓扑结构的构造方法，任意元素数目的空间结构，其 **RA** 的数值，只与元素数量有关。

这样就好了，我们现在研究 **D** 系统，针对其中的元素 d_1 ，已经求出其 **RA(d₁)** 了。现在找个钻石形拓扑结构，设定其元素数目与 **D** 系统的元素数目相等，代入公式 4，我们就知道此时的 **RA of Diamond** 了。

重申一遍。**RA of Diamond** 的值，只与元素数量有关。现在人为设定钻石形结构的元素数目与 **D** 系统的元素数目相等，就确定了当前的 **RA of Diamond** 的具体数值了。两者相比一下，可得：

$$\text{Relativized RA}(d_1) = \frac{\text{RA}(d_1)}{\text{RA of Diamond}} \quad , \quad \text{其中，两系统元素数目相等} \quad (\text{公式 5})$$

上面公式的意思是，以某个元素为中心，进行空间重映射以后，所得到的拓扑结构，与同系统中，别的元素头上顶着的拓扑结构，是不同的。而拓扑结构里面，连接关系的不同，会造成一定的问题。我们寻找一个连接关系特别匀质、稳定的拓扑结构，即钻石形的拓扑结构，作为标准，让所有其它的结构来跟它相比较，就知道别的元素的连接关系，与标准情况偏差多大了。

我们再回顾一下上面这些数学处理的步骤，在 **D** 系统中，针对 d_1 和 d_2 ，我们分别求出了 **Total Depth(d₁)**、**Mean Depth(d₁)**、**RA (d₁)** 和 **RRA (d₁)**，以及 d_2 的对应的参数。

如果一开始， $Total\ Depth(d_1)$ 就大于 $Total\ Depth(d_2)$ ，那么不用具体计算，我们也知道 $RRA(d_1)$ 一定是大于 $RRA(d_2)$ 的，因为计算过程中，除以的数值都是相同的。

按照建筑学当中，对于空间深度的理解， $Total\ Depth$ 比较小的元素，其可达性是较高的。而 $Total\ Depth$ 相对较大的元素，可达性不高。按照基本假设来说是这样，不需要剔除影响因素也知道。可是这个话就越说越绕，听起来也费劲。我们给这个 RRA 取个倒数，命名为整合度，这样，整合度越高的，可达性越高，这样就清爽了。

$$Integration(d_1) = \frac{1}{RRA(d_1)} \quad (\text{公式 6})$$

啰嗦了这么多，就一个意思：整合度是一个关于 $\frac{1}{Total\ Depth}$ 的函数。

整合度这个概念，是针对每个具体的元素而来的。全局深度越高的空间，整合度的值越低。全局深度的值越高的，就是穷尽了所有可能性以后，从别的空间到这个中心空间，所花费的步数较多。全局深度越高的，可达性越差。那么好了，我们给它取个倒数，全局深度值越高的，整合度的值越小。全局深度值越小的，整合度就越高。这样，整合度的数值变化，与可达性的概念，变化方向就一致了。

也就是说，整合度高的空间，可达性高。

也就是说，整合度衡量了一个空间吸引到达交通的潜力。

如果看到这里，您还没有吃夹生饭，那么恭喜您，空间句法您入门了！

第五节 尽端空间的专业化

在本文的绪论当中提到过一下，西方人的建筑观念，是有特色的。建筑空间的专业化，是西方人的特色。

当一个空间，被专门用于某项功能的时候，可以减少外界的干扰，使得人们做事情更加专注，效率更高。

专业化这件事儿，好处虽然很大，但造成的问题也不少。因为专业化也就排除了用作其他用途的可能性。太过专业化的空间，因为承载的功能单一，可能造成空间布局缺乏灵活性，缺乏弹性，很容易造成浪费。

为了保证不受打扰，专门的空间只用作单独的功能，西方人在创造尽端空间的方向上，走了很远。

西方人的建筑空间，可以分为两类，一类是尽端空间，一类是通过性的空间。

尽端空间是私密的，通过性的空间是公共的；

尽端空间分别属于不同的人，尽端空间被完全地控制着，而通过性的空间属于所有的人，通过性的空间不被控制；

从一个尽端空间去往另一个尽端空间，一定要从公共空间过渡；

上面所述的这些特点，可以用一个示意性的空间结构来表示，并将这个空间结构弄到空间句法中来算一下，用空间句法的话语体系对它加以诠释。假想的空间结构如下：

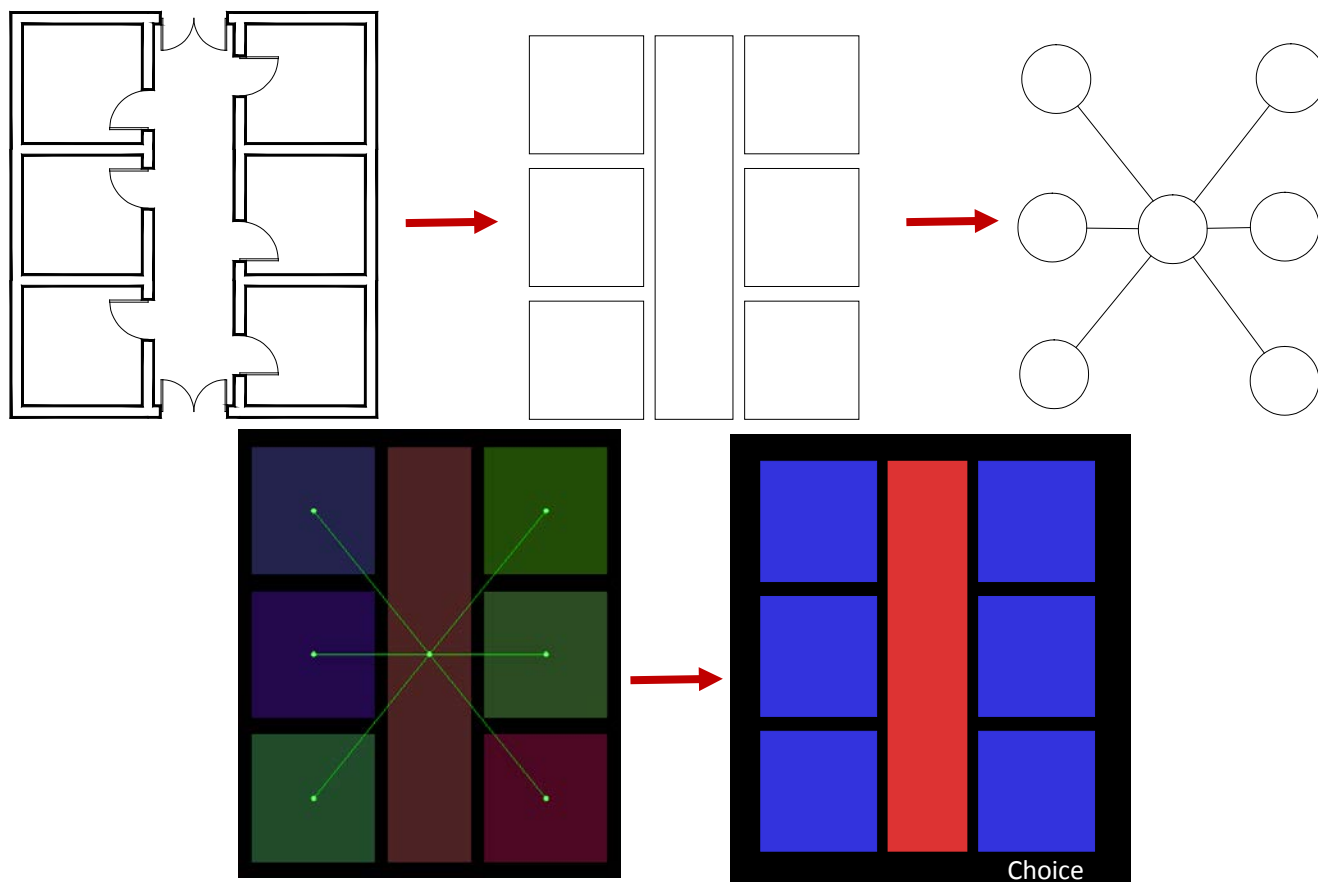


图 1.5-1 示意空间

上图中，拓扑结构算了一下，截图出来的是选择度（Choice）的情况。尽端空间的 Choice 值为零，没有穿越交通。中间的公共空间吸收了全部的 Choice 值。

一条走廊两侧，挂着无数的尽端式的房间，这种空间概念，在西方人的电影作品中也被经常探讨。参见下图：



图 1.5-2 通过性的空间

为了避免被打扰，尽端空间与通过性的空间之间，这个分隔手段，常常是采用彻底隔绝的生硬办法。

如上图这样的，用不透明的门彻底隔离的，太过生硬，凸显出使用尽端空间的人的那种绝对性的支配权力，对于客人来说，这种空间安排显得不够友好。

人与人的关系，有的时候不那么绝对，就不需要空间上的彻底隔离了。许多办公室的门，就换成了透明玻璃的。用了玻璃门的这种方式以后，虽然这个尽端空间，仍然从物理上隔离出来了，但是给人造成的心理冲击没那么大，人们视线相通，或多或少保留了交流的渠道，对于工作空间来说，会更加合适一些。

在办公场所的设计上，为了促进人与人之间的互动，更进一步的办法，就是把物理隔离的手段都取消，变成开敞空间。取消了空间的物理隔离，或多或少，会损害到空间的专业化水平。空间开放到什么程度，就能取得平衡，既保证人们能专心工作，又保证人们有足够的交流，这个还是需要多多体验一下，再做决定。

黑皮子的那个《空间是机器》的第七章，就围绕这个话题，展示了一些实证研究的成果。读者可以参考一下。

最近十几年的发展方向是，在开敞的办公空间中，地坪做出较大的高差，来创造新的可能性。如下图所示：



图 1.5-3 带高差的工作空间

如上图这样的空间设计，工作人员面向同一个方向，有一种心理暗示，促使人们更加齐心协力地朝着团队的共同目标迈进。带高差的这种空间设计，对于团队成员之间的信息交流，具有直接的促进作用。如果上图这个空间，其他状态不变，只是取消了地坪的高差，所有人都呆在相同的标高上，那么，这个空间会失色不少。

DepthMapX 目前，还不能把 z 轴的高差考虑进来。将来软件进化了以后，如果能够带标高进来计算，也许能够揭示一些更加深刻的规律。

本节是以办公空间为主题，把考虑问题的方法展示一下。

西方人在尽端空间的专业化方向上，走得太远了，干什么事儿，都要专门设置一个尽端的房间，门上钉个牌子，再强调一下该房间的功能。这种观念太强大了，后来，建筑师尝试着取消尽端空间的时候，比如密斯设计的那些流动空间，还在建筑历史当中砸起不小的浪花。

除了办公空间以外，设计工作中还会涉及到大量其他的功能空间。如果设计专业性的厨房，或者设计医院，这时候尽端空间和通过性空间的区分，就无法软化。因为服务于非常专业化的尽端空间，就造成，通过性的空间本身，也被专业化了。一个专业化的通过空间，与另一个专业化的通过空间，两者可能互相排斥。这就造成了交通流线的切割。例如，从厨房往外上菜的这条流线，和回收用过的餐具的这条流线，互相排斥。为了保证卫生，不能允许脏盘子回收走相同的通道，处理的办法，只能是流线互相独立，从物理上进行隔离了。

通过性的空间如果设置不合理，造成功能流线出问题，这设计就通不过了。没有空间句法的情况下，功能流线的情况，也能探讨清楚。有了空间句法以后，我们就多了一个工具，分析问题的时候，可以更加彻底一些。

本章小结

第二章 视线分析

对于某种深度的考察，在空间句法中占有重要的地位。本章分析视线的深度。

本文前面章节介绍的凸空间的那种拓扑关系的思路，暂时放一放。等视线这部分结束了，到了轴线模型的章节，拓扑关系的事儿还会再拎出来折腾。

本章的重点是视线的深度问题。视线深度的定义，参见下图。在一条笔直的街道上，a 可以直接看到 b，这种情况称为 a 距离 b，一个视线深度。而从 c 不能直接看到 e，必须先走到 d 点，这种情况称为 c 距离 e，两个视线深度。

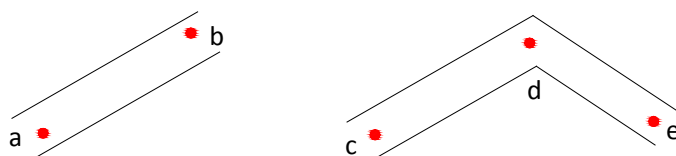


图 2-1 视线深度的定义

定义了视线深度以后，还可以更进一步，设置视线所及的最远距离。比如，我们已经在软件中设置好，最远能看 150 米，而 a 与 b 的距离，已经达到两百米了，那么，软件就认为它俩相距两个视线深度。

在上述思想的基础上，我们可以对平面进行一下加工，把空间变成可以计算的状态。如下图所示：

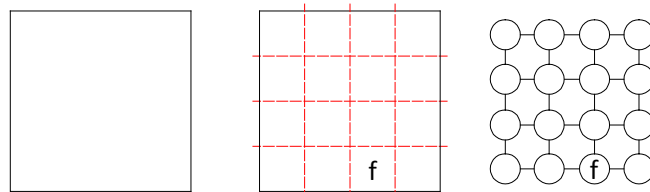


图 2-2 用无限细分的方格网，去代替平面图

上面这个图的意思是，最左侧，是原始的建筑平面，用无限细分的方格网，把这个平面切开，就变成了一个由方格作为元素，带有拓扑连接关系的这么个空间结构。因为方格本身也是凸空间，这就表示，前面章节中介绍的基本思考方法，在这里也成立。当方格网划分得足够细密时，就可以将其中的某个元素，理解为“点”了。视线分析就是处理点与点的关系的。

在上图中，对于 f 来说，拓扑关系上，与之有联系的，有三个元素，所以 Topological Connectivity (f) = 3; 从 f 向外看，全系统的所有其他元素，都能看见，所以 Visual Connectivity (f) = 15, 对吗？

将空间结构转译成无限细分的方格网以后，DepthMapX 软件当中，有三种深度定义，都可以进行计算：

- Point Visual Depth
- Point Metric Depth
- Point Angular Depth

这三种深度定义，是查看 DepthMapX 源代码的时候，看到引用函数的时候，三个东西并列着，才知道的。好像没有文献提过这些事儿。

其实弄成细分的方格网以后，可以干很多事儿，可以测量点与点之间的距离、点与点的角度啥的，除了这个，还可以测量每个点看到的面积，还可以把这个元素所对应的空间的高度弄进来，作为某种权重，在继续往下发展。算法是可以人工地去设计的。

第一节 网格设定

DepthMapX 这个软件，不认曲线。下面左图是在 CAD 当中画的，都是标准圆弧，导入到 DepthMapX 当中，发现曲线不闭合。将来要让软件自动去识别空间边界，不闭合的不行。处理的办法是，在 CAD 当中，用直线段去拼这个圆弧。保证所有边界，都是用直线闭合起来的，就没问题。

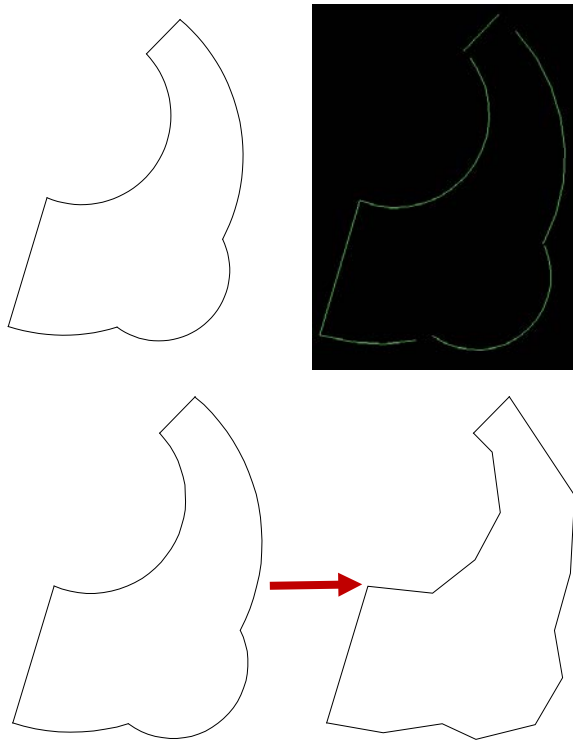


图 2.1-1 用直线段去近似地表示曲线

假设我们的研究对象已经有了，把它的平面图，在 CAD 中，将曲线全部取直，平面边界全部闭合好，清理干净，重新通过 dxf 格式的文件输出出来，再通过 Import 命令导入到 DepthMapX 里面，就可以进行视线分析了。本文中，再弄一个假想的平面，作为示例。空间的原始状态，如下面左图所示，在 CAD 中清理完的状态，如下面右图所示。

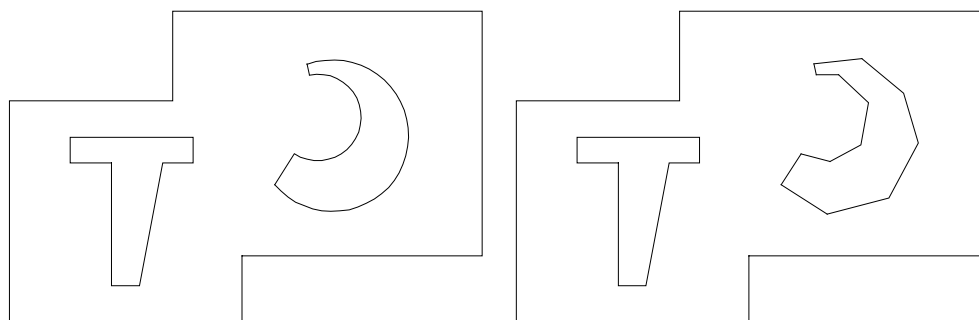


图 2.1-2 假想空间

第一步是设定网格，英文是 Set Grid，可以在 DepthMapX 窗口上部的命令按钮那一行找到对应的图标，也可以通过下拉菜单 Tools → Visibility → Set Grid，一步一步在菜单中找到这个命令。如下图所示。



图 2.1-3 Set Grid

dxf 格式的文件导入进来后，什么都不做，直接 Set Grid。点击了这个 Set Grid 的命令以后，弹出对话框。如下图所示。对话框里面要设置的这个 Spacing，就是设定网格的具体尺度的。一般人流不是 500-550 毫米宽么，这个尺度，对于建筑设计来说，够用了，本教程中就用 550 吧。读者自己设定网格的时候，可以根据自己的感觉来设置，只要方格网足够细密就行。有的论文中，研究城市广场的，方格网的间距设置成两米的，也有；有的研究住宅的，为了使得将来输出的图像较为细致，方格网设置成一百的也有。读者可以灵活掌握。

当初从 CAD 导出平面图的时候，一般一个单位就代表一毫米，在 DepthMapX 中是无单位的状态，本教程中就直接输入 550 了。如下图所示：

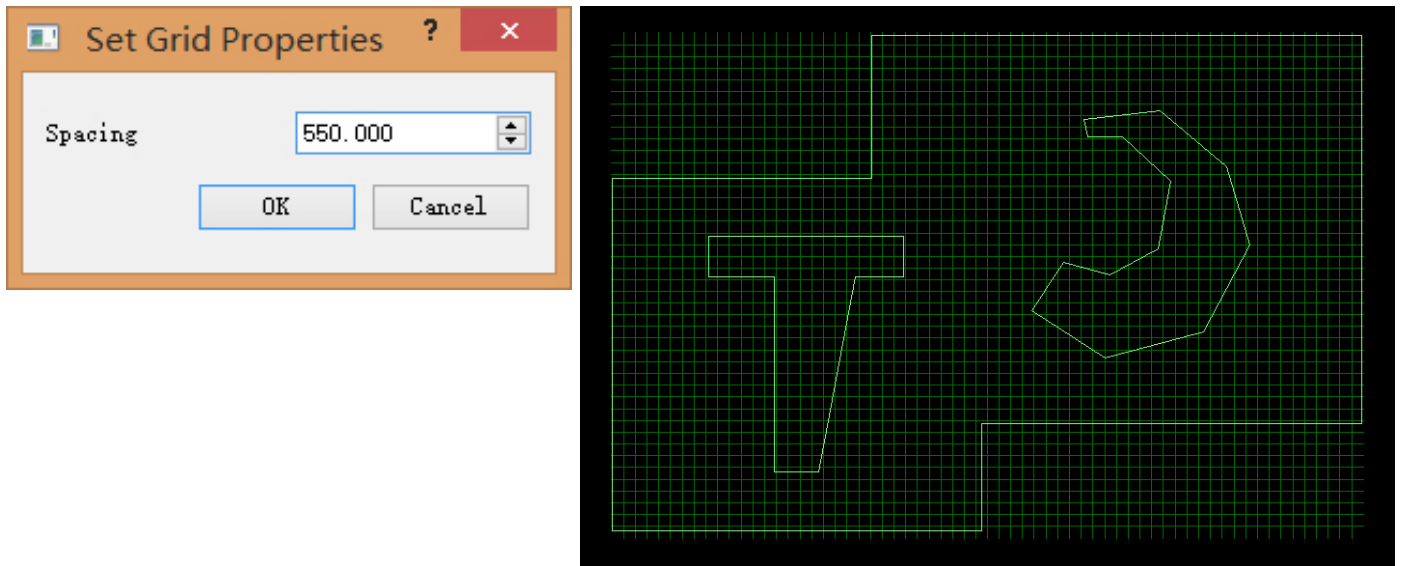


图 2.1-4 网格尺度的设定

设置网格尺度的这个对话框，OK 一下，右侧的绘图区，就会发生相应的变化。软件会自动用方格网将平面图全部覆盖住。

读者可以注意一下 DepthMapX 软件窗口的左侧，图层索引的那个地方，已经出现了新的图层，将来有了什么计算结果，会保存在这个图层上。

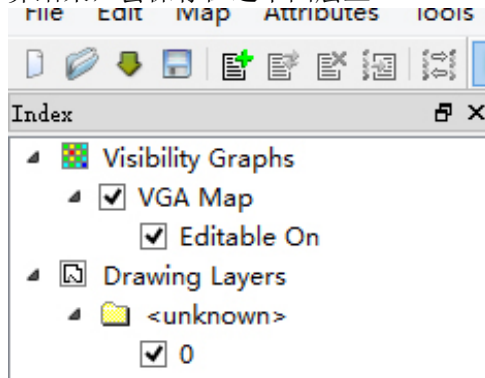


图 2.1-5 产生新的图层

如果设置了某个网格尺度，觉得不满意，重新 Set Grid 的话，这个图层上过去存着的计算结果，都会丢掉。可以把当前图层关掉，回到刚导入进来的那个 Drawing Layers 那个图层，重新设定网格，就会产生新的图层。就像操作 PhotoShop 一样，图层状态先检查一下，再去操作其他的功能。

接下去，用油漆桶命令，找到参与计算的元素，英文是 Fill 命令。如下图所示。Fill 命令，下面还分为 Standard Fill 和 Context Fill，本教程中，仅介绍 Standard Fill 命令，因为 Context Fill 是什么意思，至今也没搞明白。

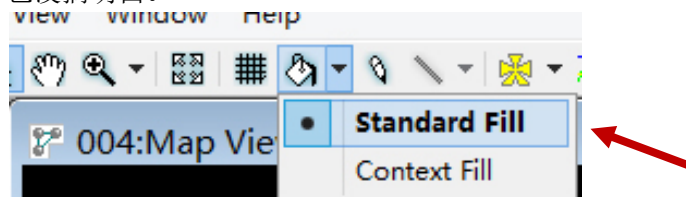


图 2.1-6 油漆桶工具

Fill 命令，用的时候，先点击图标右侧那个黑色三角形，打开下拉列表，选中 Standard Fill 的状态，然后再点击油漆桶图标本身，这种状态下，把鼠标移动到 DepthMapX 的绘图区，鼠标指针会变成油漆桶。

在我们的平面空间的内部，随便找一个元素，左键单击一下，软件就会自动把所有参与计算的元素识别出来。空间的边界会变成绿色的，参见下图。

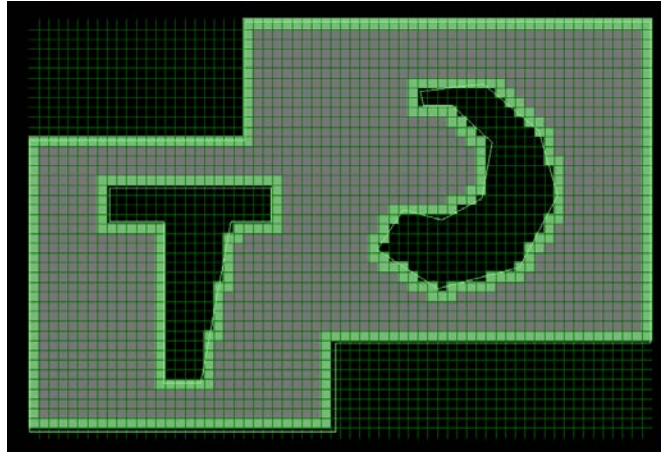


图 2.1-7 找到参与计算的元素

上面这步做完了以后，通过 Select 命令，也就是命令图标那一行，靠左边的那个箭头状的图标，点击一下，退出 Fill 命令。免得 Fill 命令一直开着，鼠标不注意，正在绘图区乱点，容易造成误操作。

到了这一步，网格设置就算完成了。后面的操作，就是设置些对话框，让软件去计算点与点之间的关系。

这里不妨重申一下中心思想。数学上不是有微积分么，所谓的微分，就是用某个假想的任意小的间隔，将一个物体切片。切开以后，研究这个非常非常薄的薄片，是什么状态。将来，把这些薄片再重新加总起来，就是积分。本节介绍的，用无限细分的方格网，去切这个空间平面，这就是数学上微分的思路。后面的章节中，将要介绍的用直线段去概括空间，就是数学上的积分的思路。

当然，微分积分，这都是类比的说法，为了方便读者理解，这样提一句。

本节内容到这就结束了。下节开始，介绍一些软件操作的知识，再介绍一些视线分析的算法。

第二节 视线分析的软件操作

所谓的视线分析，就是用无限细分的方格网，将空间结构转译成小方格组成的空间系统以后，考察元素与元素之间的某种数学关系。

接着前面一节的步骤，继续操作软件。首先要制作一下 Visibility Graph。在 DepthMapX 窗口的最上部，找到 Tools 下拉菜单，通过 Tools → Visibility → Make Visibility Graph，左键单击一下，弹出对话框。如下图所示：

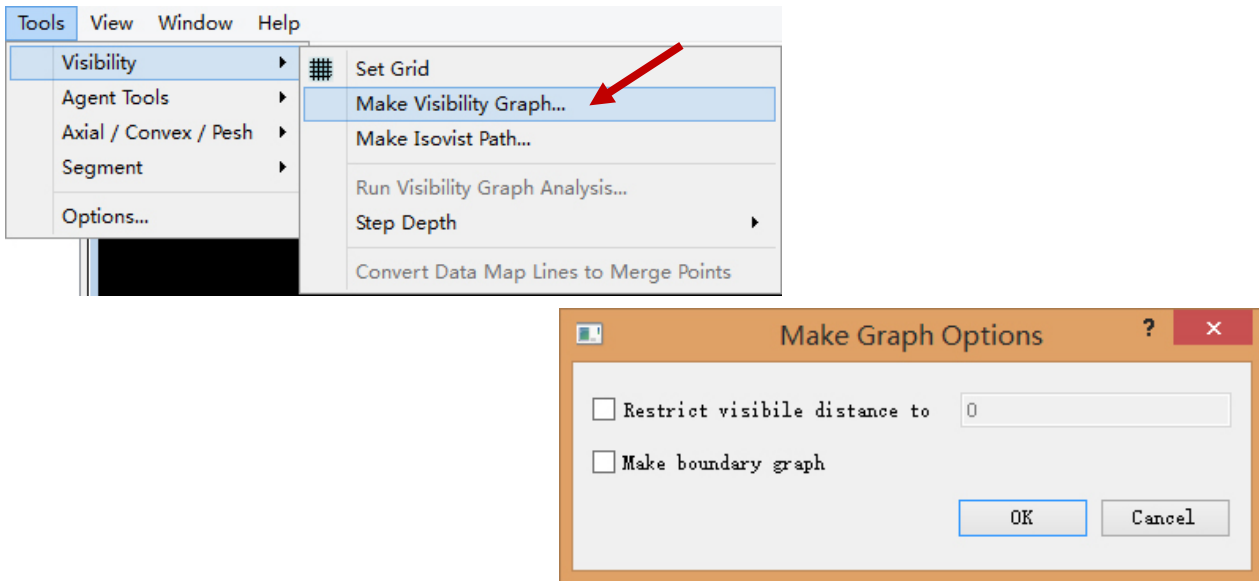


图 2.2-1 Make Visibility Graph

弹出来的这个对话框，有两个方框可以勾选。上面那个选项，是设置视线所及的最远范围的。默认情况是无限远。如果前面的勾打上，后面可以输入数值。当初 CAD 底图，导入进来的时候，DepthMapX 不认单位。当初一个单位代表一毫米，在这里就输入毫米的数值。例如，现在要限制视线所及的最远范围，是五米，那么就在这个对话框中输入 5000。后面几个零，要数对，多数几次。

到了具体的研究项目中，具体数值设为多少，可以现场去测量一下。现在那种手持的激光测距仪，几百块就能买到，方便实惠。

影响视线所及的最远范围的，一方面是光照条件，另一方面是俯视和仰视。在操作实际的研究项目的时候，先去体验一下，再回来确定这个 Visible distance 的值。

当然，DepthMapX 的算法较为简单，无法计算透明性的空间边界，无法计算柔性的空间隔离的状态，无法计算带高差的空间的互视问题，只能计算不透明的完全遮挡的平面边界。有经验的读者，可以在软件给出的计算结果的基础上，根据自己的经验去修正。

上面对话框中，下面那个打勾的地方，字面直译是“制作边界图”，从来没用过，也不知道是什么意思。

本文中，对话框中的两个选项，都不勾选，只是保持默认状态，OK 一下。

这个时候检查 DepthMapX 窗口的左下角，参数选择栏那个地方，我们会发现，新出现了一些选项。读者可以左键单击，分别选中看看。蓝色高亮箭头所指的那个参数，就是绘图区当前正在显示的内容。默认的当前状态是 Connectivity。

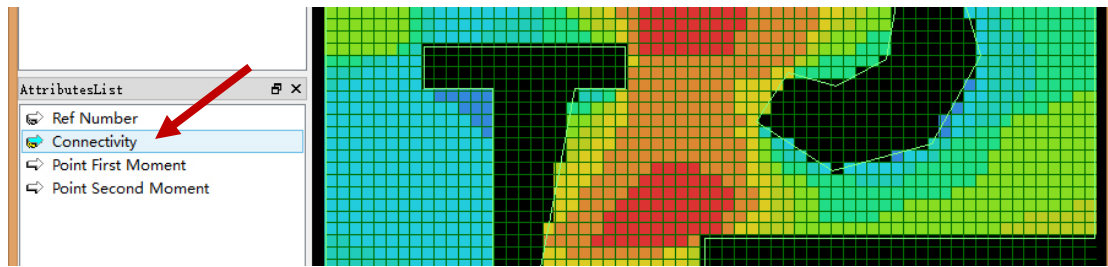


图 2.2-2 查看计算结果

上图所示的这个基于网格的 Connectivity，意思是，从某个元素向外看，一共看到几个其他的元素，这个加总的数值，反馈回来，就记在该元素的头上。系统中的每个元素都这样做了以后，每个元素头上都有了 Connectivity 的数值了。然后按照 Connectivity 的值，从高到低，分为十个数值段，某个具体的元素落在哪个数值段，就显示相对应的颜色，生成了上面这张图。

上图中，参数选择栏那里，有两个参数，Point First Movement 和 Point Second Movement，查了一下代码，没有找到具体的算法，就解释不了了。

接下去继续操作软件，在 DepthMapX 窗口上部的下拉菜单那里，通过 Tools → Visibility → Run Visibility Graph Analysis, 弹出对话框。如下图所示：

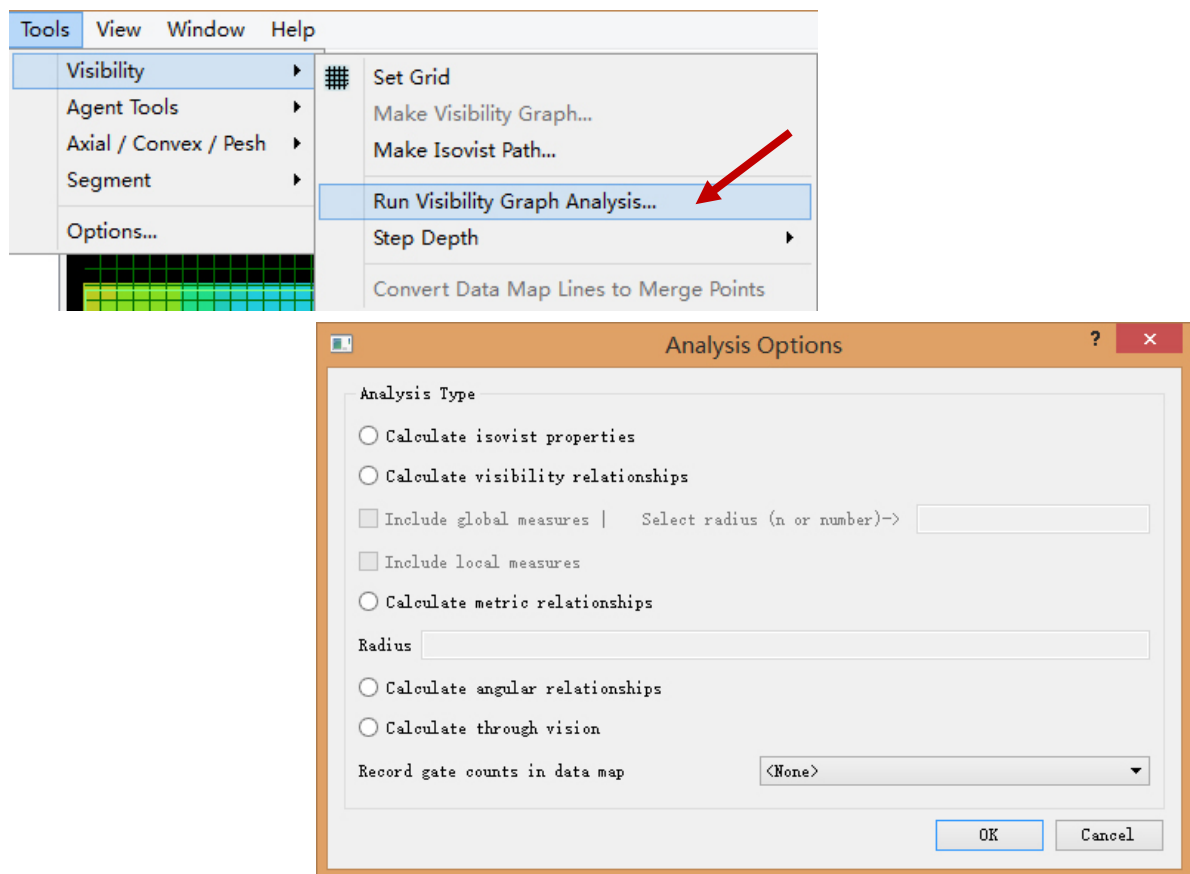


图 2.2-3 视线分析

这个对话框，有五个选项。前面的圆圈，左键单击一下，变成黑色实心的，表示这是当前选中的操作。不能多选，每次只能操作一个。

首先选中第一行， Calculate Isovist properties 这个选项， OK 一下。

计算结束后，参数选择栏那里，出现了一堆新东西，包括 Isovist Area、 Isovist Compactness、 Isovist Drift Angle、 Isovist Drift Magnitude、 Isovist Max Radial、 Isovist Min Radial 、 Isovist Occlusivity、 Isovist Perimeter 这几个选项。

Isovist 这个词，是个新造的词，大概可以翻译成“等视域图”，意思是，从一个元素向外看，能够看到多大一片地方，这个面积的值，反馈回来，记在该元素的头上。在最后给出的计算结果中，相同颜色的两个元素，视域面积应该是差不多的。参见下图。

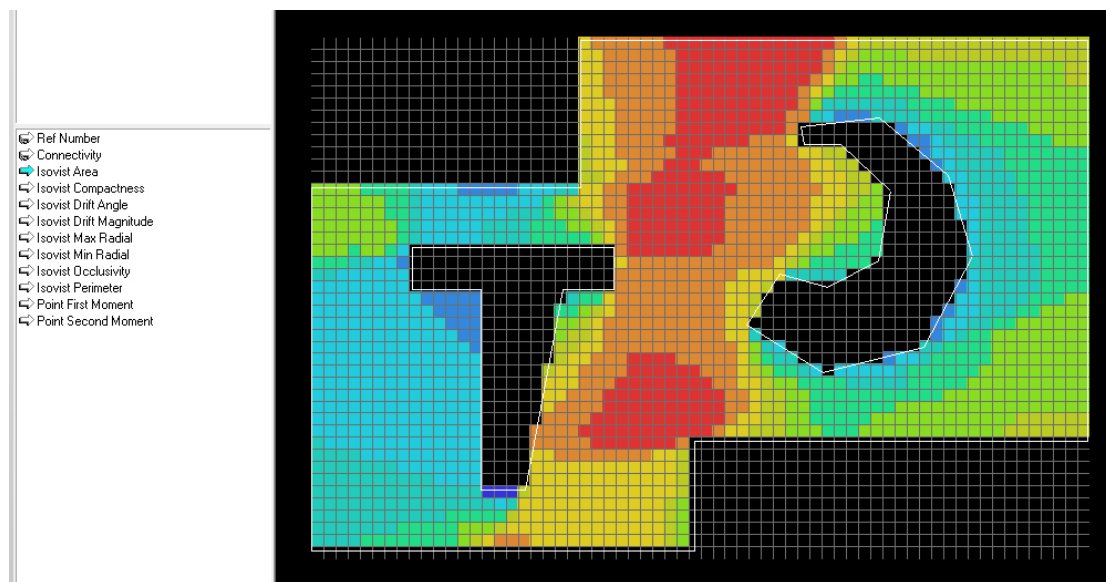


图 2.2-4 Isovist Area

前面介绍的 Connectivity，是进入视线范围内的元素的数目，而这个 Isovist Area，是视域面积，两者所说的应该是一码事儿。下面把 Connectivity 的结果也摆上来，两者可以对比一下。

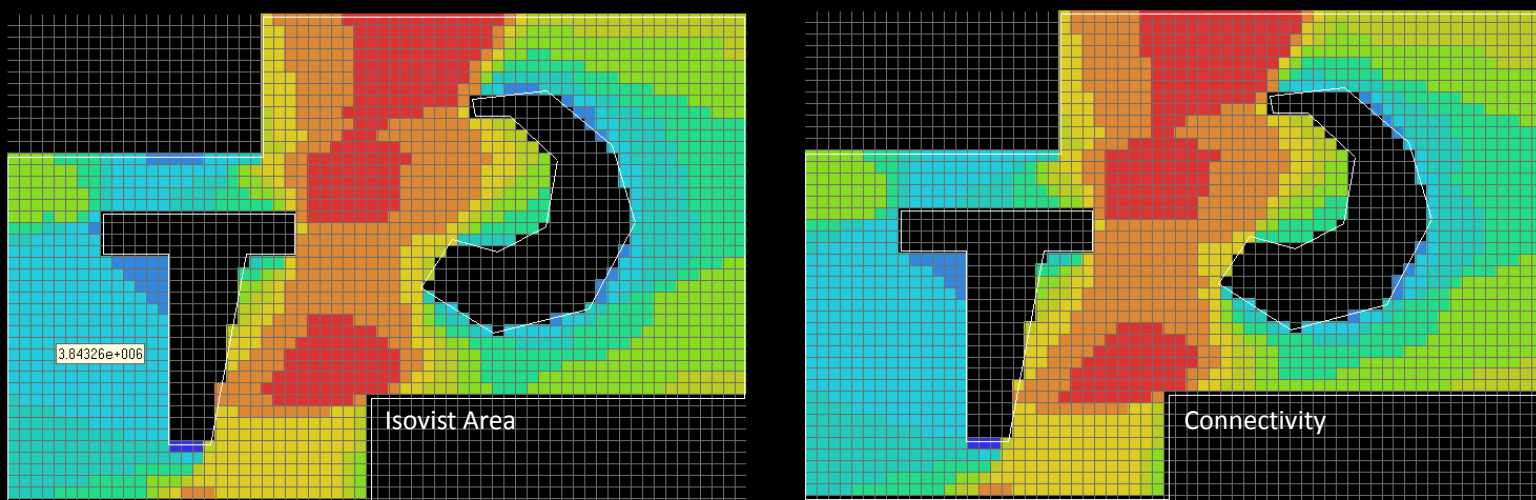


图 2.2-5 对比 Isovist Area 和 Connectivity

当设置的视线的最远范围以后，Isovist Area 和 Connectivity 的情况，就可能出现较大差别，因为设置了半径以后，可能有些元素就不参与计算了。

与 Isovist 有关的其他那些计算结果，文献里面没提过，源代码也看不懂，本文就不涉及了。

这个视域范围的事儿，在空间句法官方发布的犀牛插件里面，也有。有兴趣的读者可以到犀牛里面玩玩。

有了空间分析的结果以后，到底应该怎么解释，还是先要从算法开始。撇开算法，图像无法解释。

下面的章节中，继续介绍对话框中剩下的四个选项。

第三节 Visibility Relationships

还是上一节提到的那个对话框，现在选中第二个选项。

从 DepthMapX 软件窗口的最上部，下拉菜单那里，找到 Tools → Visibility → Run Visibility Graph Analysis，左键单击一下，弹出对话框，在对话框中，选中 Calculate Visibility Relationships 这个选项。如下图所示：

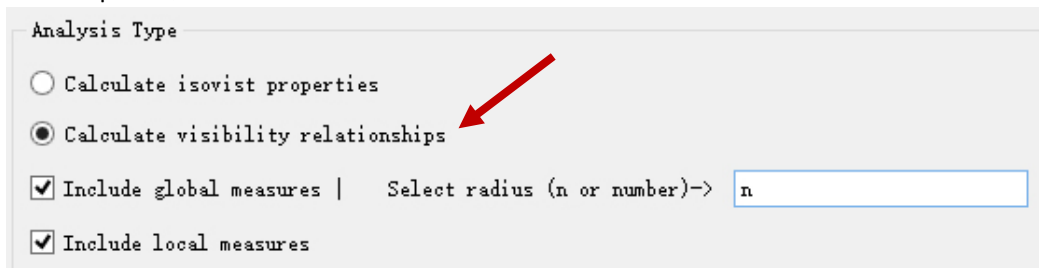


图 2.3-1 设定 Visibility Relationships

在这个选项中，有两个下一级的方框可以勾选，如果上面那个 Global measures 勾上的话，将来在参数选择栏会出现 Visual Integration、Visual Mean Depth、Visual Node Count 和 Visual Relativized Entropy 这些计算结果；如果勾上下面那个 Local measures 的话，会出现 Visual Clustering Coefficient、Visual Control、Visual Controllability 这几个计算结果。

但是，这个功能模块有问题，有的时候计算结果不显示。如果读者检查参数选择栏那个地方，看不到这些应有的计算结果，直接退回去用低版本的 DepthMap 10.14 就是了。本文写作的时候，用的版本是 DepthMapXnet-0.30-windows，如果不显示计算结果，就不用这个版本了。不要在调试软件上面浪费时间。

在 2001 年举行的第三届空间句法大会上，老的 DepthMap 程序员 Alasdair Turner 发表了一篇论文(Turner, 2001)，解释了上面提到的这些计算结果，是怎么弄出来的。本节后面的内容，就是在那篇文章的基础上，改写了一下，搬过来的。

1) 深度的定义

在 DepthMapX 中，元素之间的距离，也就是深度，有三种定义的方法，分别是 Visual Depth、Metric Depth 和 Angular Depth。在这些定义的基础上，设计了视线分析的那些算法。

我们先点击 Select 命令图标，进入元素选择的状态，可以通过左键单击选中单个的元素，也可以按住鼠标左键不放，拖动，进行框选。选择集会变成黄色高亮的显示状态。如下图所示：

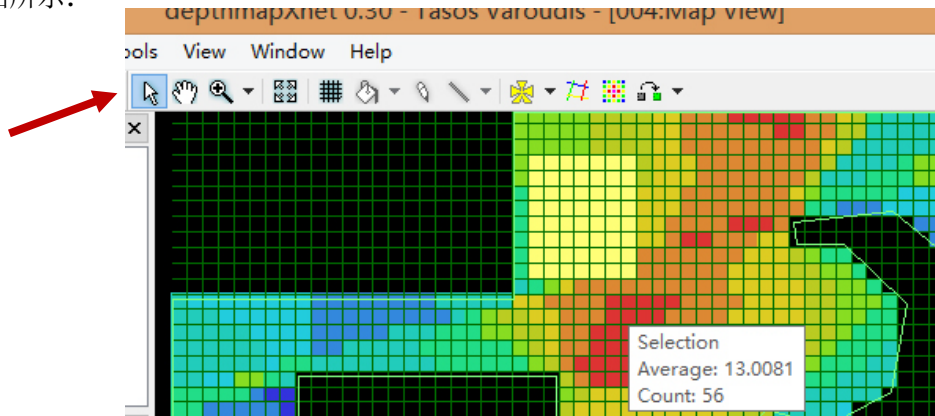


图 2.3-2 找到选择集

当选择集分别位于不同的地方时，可以通过按住 shift 再单击，按住 shift 再拖动，这样的方法，来进行多选。例如，研究商业中心的时候，希望选中前门的一片区域，再选中后门的一片区域，同时参与计算，就用这个 shift 键就行了。有了选择集以后，鼠标附近会跳出一个白色窗口，会显示这个选择集的情况，比如一共选中了几个元素，平均值是多少。这个平均值，意思是，当前，在参数选择栏那里，选中的是那个计算结果，就读取这个计算结果里面的数值，算个平均数。这个平均值的事儿可以先放放，后文还有介绍。

有了选择集以后，就可以考察选择集与全系统其他元素直接的深度关系了。从 DepthMapX 软件窗口的上部，下拉菜单那里，选择 Tools → Visibility → Step Depth，下一级菜单有三种深度关系的选项。参见下图：

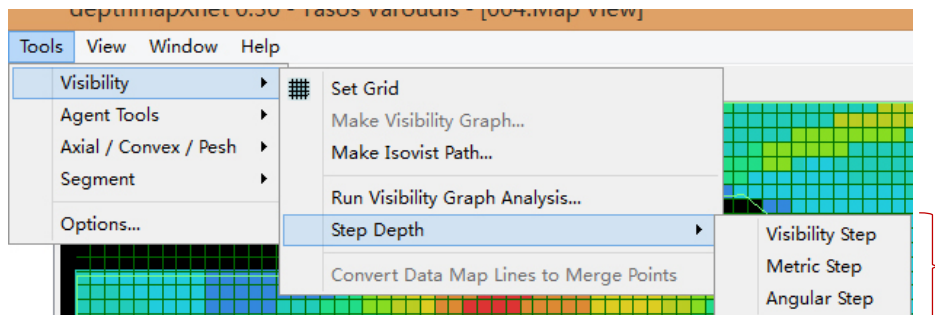


图 2.3-3 三种深度关系

没有选择集的情况下，无法点击后面那三个选项。

首先左键单击这个 Visibility Step 的选项，计算结果自动在参数选择栏那里处于当前状态了。参见下图：

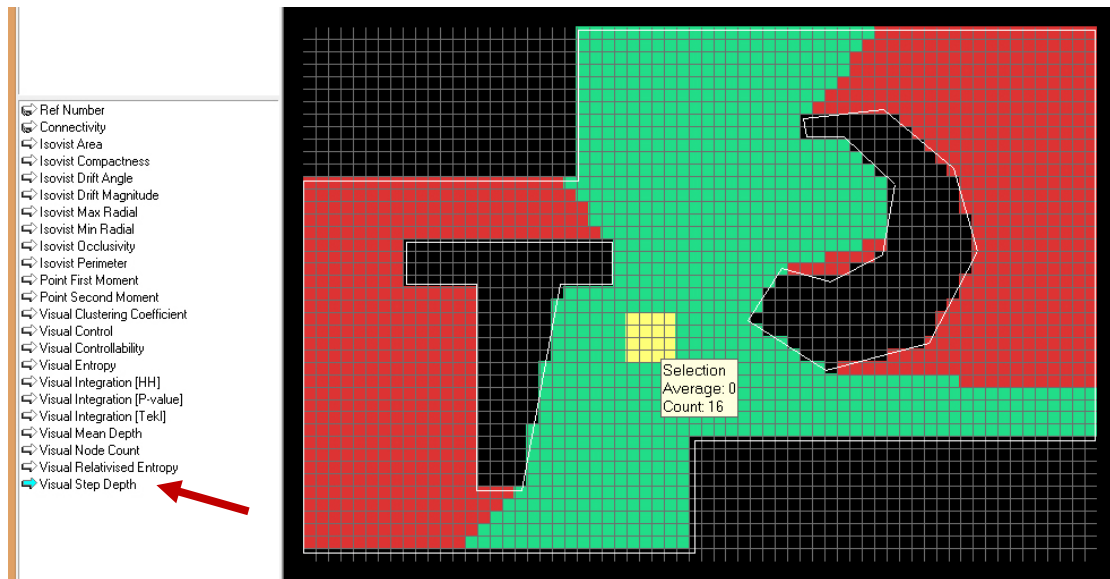


图 2.3-4 Visual Step Depth 的情况

上图的意思是，如果从选择集往外看，能直接看到的元素，就算是一步深度的，然后，在一步深度的区域内，活动活动，往外看，能直接看到的，算是两步深度的。依次类推，直到穷尽了全系统会后，按照每个元素头上记着的数值，进行分级显示。越偏暖色的数值越高。

读者可以将计算结果与自己的主观经验对应一下，看看是否相符。

接下来，从 Tools → Visibility → Step Depth 往下走，选中 Metric Step，看看什么情况。这个是考察米制距离的。如下图所示：

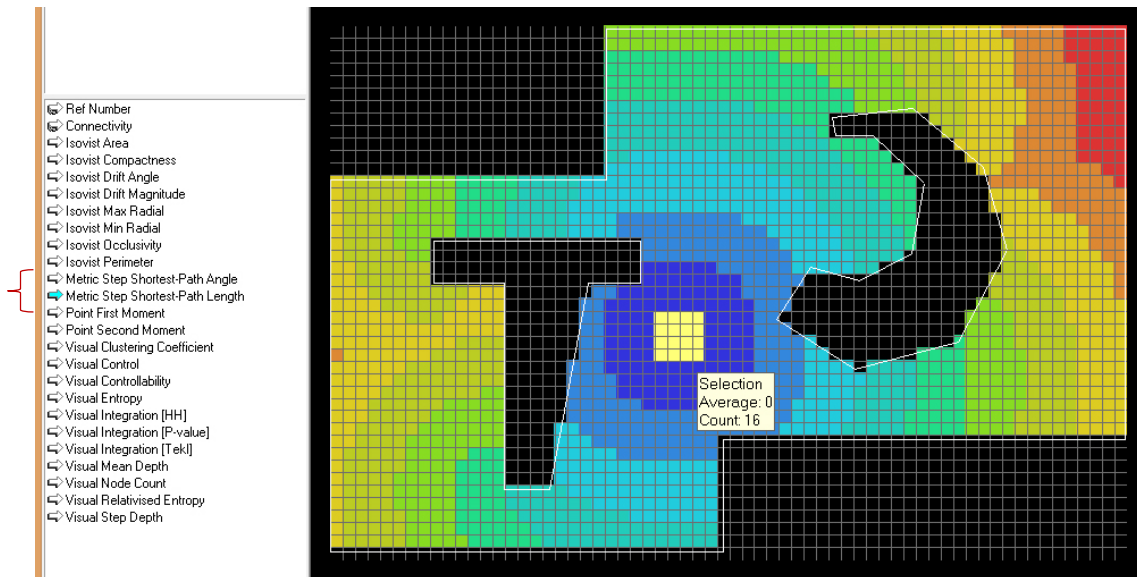


图 2.3-5 Metric Step Depth 的情况

这一步执行以后，有两个计算结果，在上图红色括号所指的那个地方。都是米制的距离，分别加了限制条件，一个是考虑实际路程的，按最短路程来的；另一个是考虑转角的，路径所转过的角度，加总以后最小的这个，参与计算。

具体的算法是，先按照某个规则，确定起点和终点之间的路径，然后，看一下这个路径的米制距离，这个数值反馈回来，记在终点的头上。将来形成一个彩色特征图以后，越是偏

暖的颜色，表示距离选择集越远。

这个路径选择的问题，是十分重要的基础性的问题。本文前面的章节中，不是提到，西方人的建筑空间的专业化，主要是尽端空间的专业化么，而决定尽端空间在城市当中的分布的，却是通过性空间的这个系统决定的。

当通过性的空间，的空间结构，有问题的时候，会造成尽端空间分布上的问题。西方人的思维，是逻辑中心主义的，既然我们既要求尽端空间的专业化，又要求通过性的空间能够发挥直接而又明确的作用，那么好了，弄个带状带状城市上来，看看怎么样。

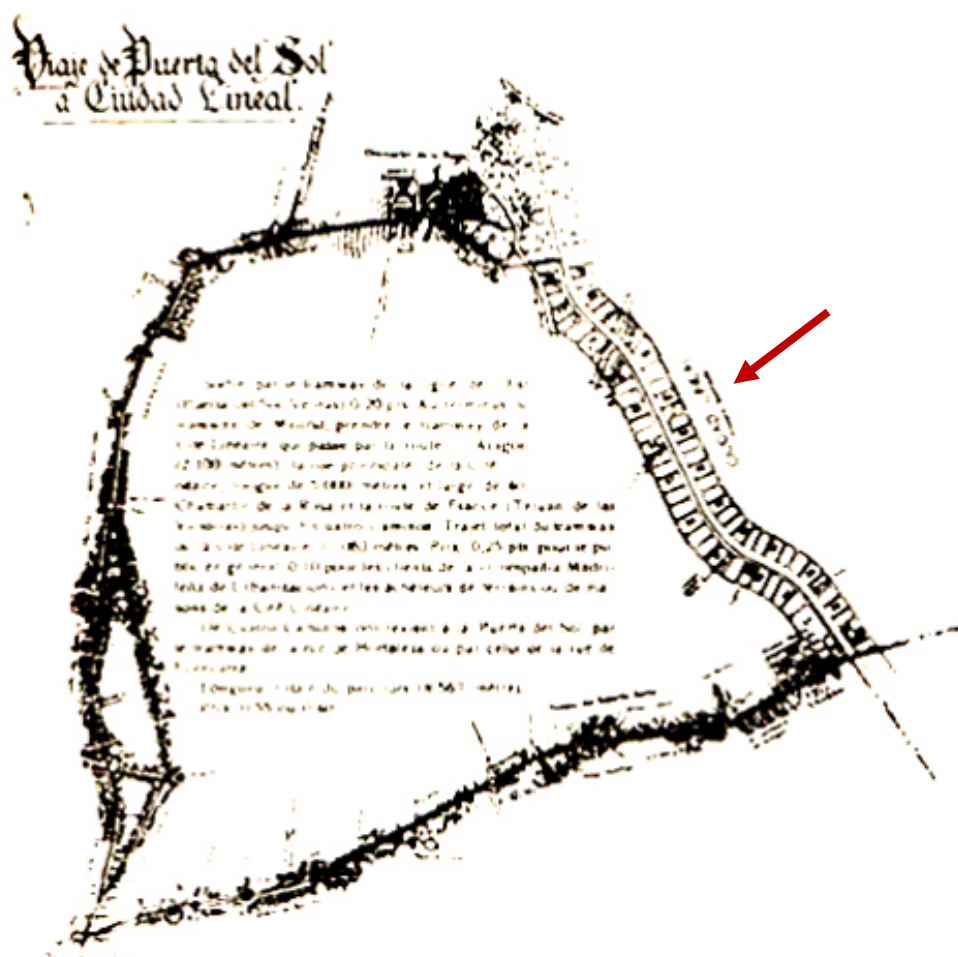


图 2.3-6 带状城市

这个带状城市，是西班牙工程师马塔在 1882 年提出的。这样地设定尽端空间与通过性空间的关系，直接就取消了路径选择的问题。

城市规划到了最后，就是个土地利用的问题。真实的城市生活不可能这么简单的，土地利用也不可能简化到这样的程度。

提到带状城市，只是为了展示一下西方人对于这个路径选择问题的关注。本文的后面章节中还会不断地围绕这个话题进行探讨。

本小节的最后，从下拉菜单的 Tools → Visibility → Step Depth 往下走，选中 Angular Step，看看什么情况。这个是考察转过角度的。如下图所示：

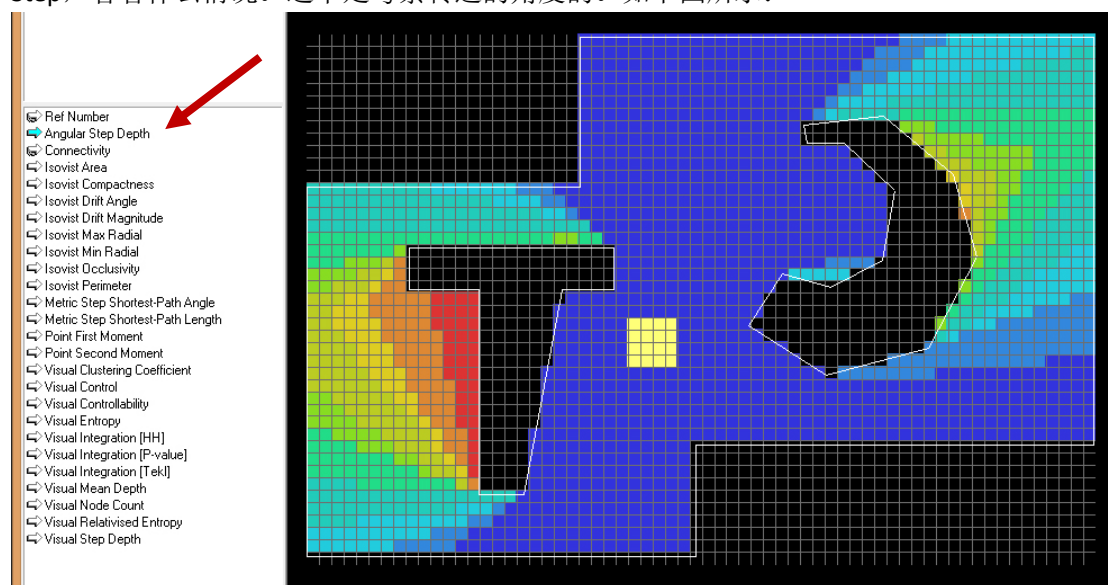


图 2.3-7 Angular Step Depth 的情况

上图的意思，是从选择集出发，到达某一个其他的元素，最小要转过多少角度，这个值记在终点空间的头上。将来形成彩色特征图以后，越是偏暖的，表示从选择集出发，越要转过更大的角度才能到达。

Angular StepDepth 的这个图，与视线转过的角度有关，而与经过的路程无关。现实生活中能碰到的，比如小朋友的捉迷藏的游戏，就是要往 Angular Step 更高的地方躲，对么？

而城市当中，为什么比较讨厌死胡同，一定要强调城市道路两头都走得通，到了这里，就好理解了，因为从视线分析的角度看，死胡同容易造成局部空间结构的 Angular Step Depth 的值快速上升，而一旦这条路两头都能走通，这个值就容易维持在较低水平。从人的认知来说，从人对于空间的使用来说，Angular Step Depth 维持在较低水平，是有利的。

另一方面，这个 Angular Step Depth 的值也不是越低越好。有些商业空间的设计，比如宜家的商场，就很强调视线的猛烈转折。商场里面有前进方向的标识，如果不按指路牌的方向走，就很容易迷路。而这样的空间设计，背后肯定是基于一定的研究的。读者可以在空间句法所提供的工具的基础上，思考一下，为什么宜家的空间设计具有那样的特点？好在哪？



图 2.3-8 宜家卖场的平面图

本小节介绍了三种深度定义，下面回到当初的视线分析的话题，去看看视线整合度等算法。

2) 视线整合度

这个视线整合度，是在 DepthMapX 软件窗口的最上部，下拉菜单那里，找到 Tools → Visibility → Run Visibility Graph Analysis，左键单击一下，弹出对话框，在对话框中，选中 Calculate Visibility Relationships 这个选项。如下图所示：

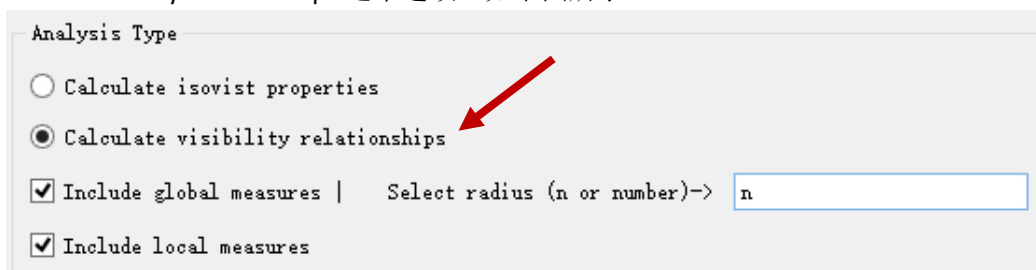


图 2.3-1 设定 Visibility Relationships

在对话框中，Visibility Relationships 下面，有两个方框可以勾选。当上面那个方框，Global measures 被勾选的时候，就会提供 Visual Integration 的计算结果。

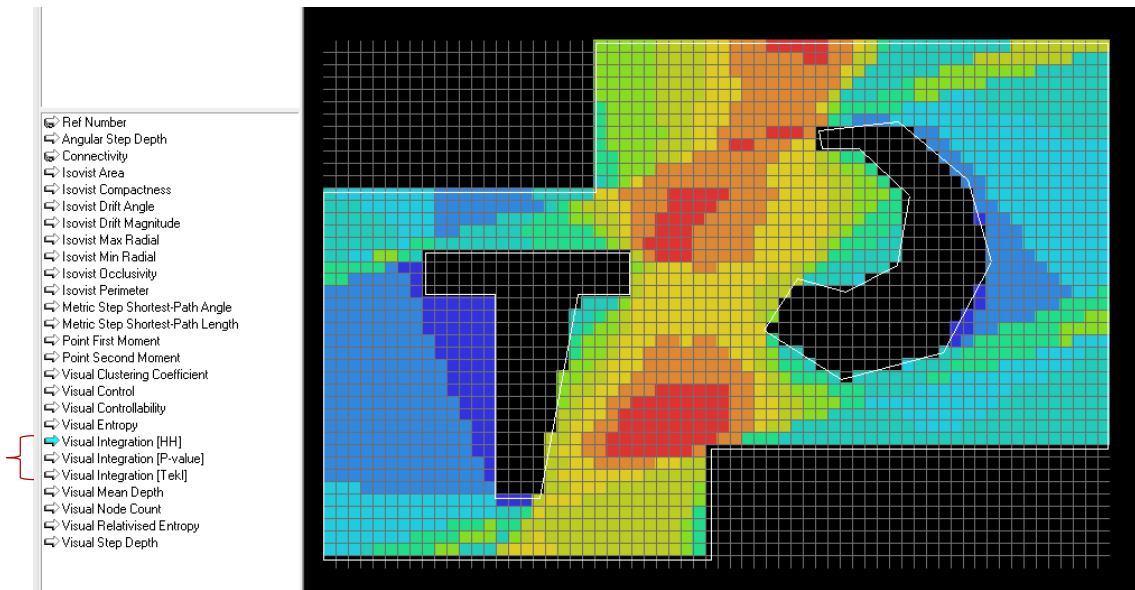


图 2.3-9 Visual Integration 的情况

计算完成后，默认显示的是 Visual Integration[HH]的值。还有 Visual Integration[P-value]和 Visual Integration [Tekl] 两个计算结果。这三个东西的算法，到现在也没有完全搞清楚。在正式的论文中，包括软件说明书啥的，都没有正式给过公式。

在 Depthmap 4: A Researcher's Handbook(Turner, 2004, p. 15)里面，提到：

applies it to VGA graphs, the result is called 'Integration (HH)'. The application of a normalisation designed for completely different circumstances may seem a little naïve, but the reasoning was that at least the measure is clearly defined, and if you say integration for VGA or axial analysis, then you know you are talking about the same quantity. de Arruda Campos and Fong (2003) examine the application of the d-value to VGA graphs in more detail and suggest that instead a normalisation using a different

上面这段，意思是说，在视线分析的时候所提到的 Visual Integration[HH]和拓扑上的那个整合度，算法完全相同。这个是官方的程序员的话，只能信这个了。如果我们自己能读源代码的话，就不需要这么麻烦了，直接看一下算法是怎么写的，就清楚了。靠着读文献，不是根本的解决办法。

前面的章节中，如何一步一步处理拓扑结构，最后得到整合度的值，详细地呈现出来了。

这个 Visual Integration [HH]的算法，是在 Visual Step Depth 的基础上得来的。从一个特定的元素出发，到达另一个元素，视线一共需要转折几次，把这个值反馈回来，记在起点的头上。以任意一个其他元素为终点，穷尽了所有可能性以后，将转折次数进行加总，这个值反馈回来，记在起点的头上，就是起点的 Total Visual Step Depth 的值。有了这个值以后，再一步步求 RA、RRA、最后再求出整合度的值，就是该元素的 Visual Integration[HH]的值。全系统照此都计算一遍，每个元素头上都记着视线整合度的值以后，就可以由软件生成一个彩色特征图，越是偏暖的元素，视线整合度的值越高。

这个Visual Integration[HH]的意思就是说，剔除拓扑学上的所有影响因素以后，全局视线深度最浅。

Visual Integration [HH]的值越高的，表示这个元素只需要较少的转折，就能看到全系统中的其他元素。

Visual Integration [HH]的值越低的，表示从这个元素出发，要看到其他元素，需要更

多的转折。

话也可以反过来说，Visual Integration [HH] 的值越高的，就是从全系统任意位置开始，只需要较少的转折就能够看到。Visual Integration [HH] 的值越高，越容易吸引到视线的关注，越容易被看到。Visual Integration [HH] 的值越低的，越难以吸引到注视的目光。

在视线整合度的概念的基础上，我们可以设定一个新的假想空间，深入研究一下。如下图所示：

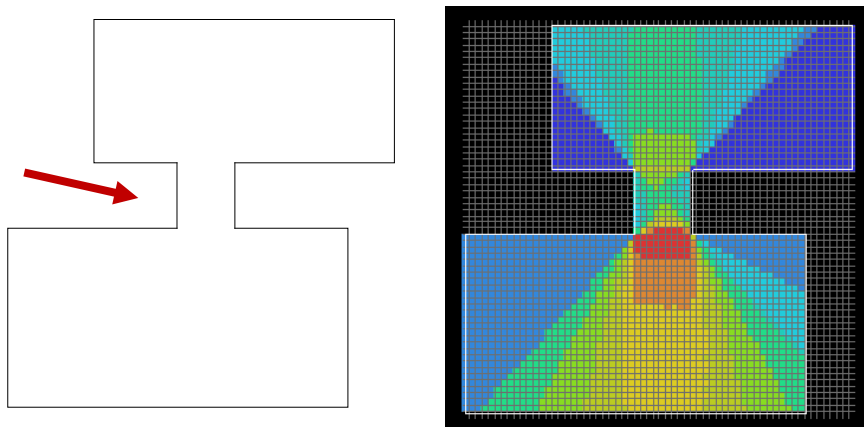


图 2.3-11 假想空间的 Visual Integration

这个假想空间由两个大的体块和一个连接通道组成。红色箭头所指的这个连接通道，在右侧的视线整合度的分析结果中，数值明显地下降了。出了这个通道，不论是从哪边出去，视线整合度的值立即上升到全系统最高的程度。我们可以将这个通道的长短变化一下，比较一下其视线整合度的变化趋势。

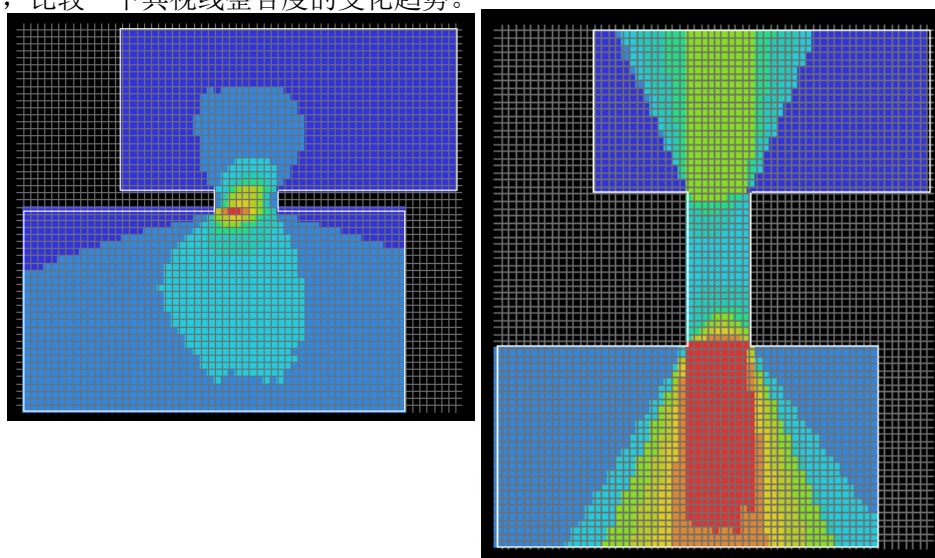


图 2.3-12 假想空间的 Visual Integration

对比一下三种情况，我们会发现，通道的长短，对于全系统的视线整合度的特征，影响很大。当通道很短的时候，通道这个地方，就是全系统当中视线整合度最高的地方，在通道这里，想要看到上下两个体块的情况，都很方便。

当通道拉伸到很长的情况时，通道的视线整合度就明显下降，而只要出了通道，视线整合度马上升高。通道本身就变成了一个较为独立的空间。在此基础上，可以跟实际的建筑作品进行比较。

下面我们来看一下伦佐·皮亚诺设计的金贝尔美术馆。这个美术馆与路易·康设计的老馆的关系如下图所示，平面图也摆在下面。

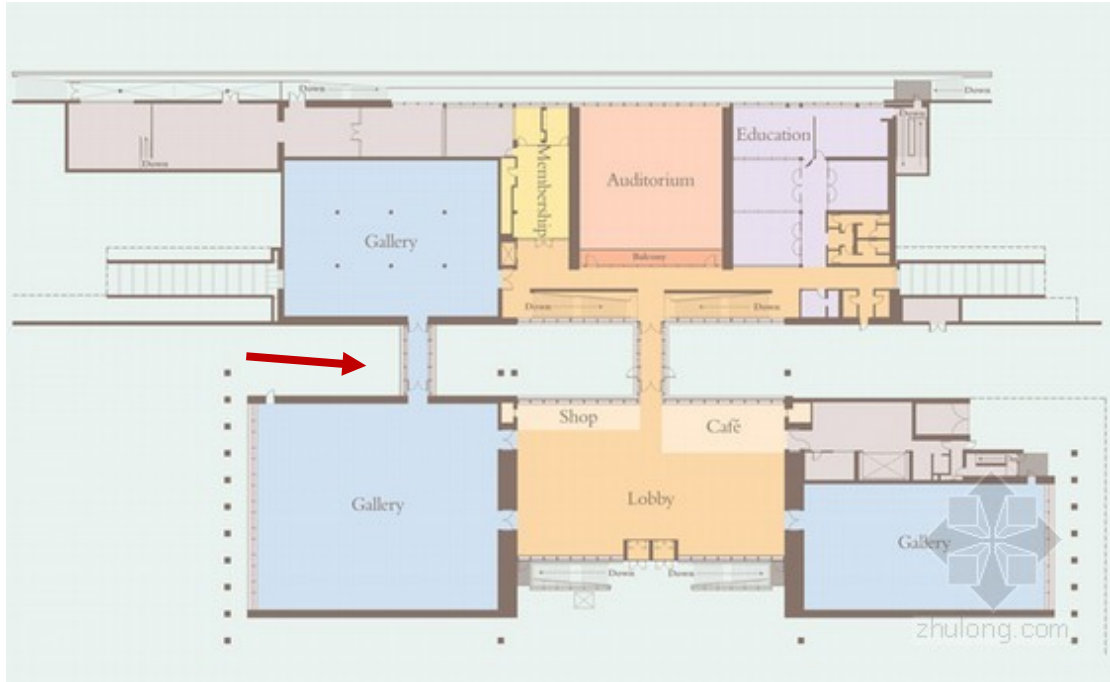


图 2.3-13 皮亚诺设计的新金贝尔美术馆

皮亚诺设计的金贝尔美术馆的两个展厅之间，这个连廊的处理，值得我们注意。就是上面平面图中红色箭头所指的这个地方。

我们先把展厅的照片摆上来看一下。如下图所示。



图 2.3-14 展厅内景

皮亚诺设计的两个展厅以及中间的连廊，是我们关注的重点。展厅都是混凝土墙面，连廊连接到混凝土墙面上，而连廊本身又达到了一定的长度，这种安排，直接弱化了连廊这个空间的视线整合度。混凝土墙不开窗，强调展厅空间的向内的那种凝聚和挤压。这是一种压缩的手段，是收。

连廊本身的处理上，又使用了落地窗，是一种完整地向自然景观开敞的状态。是放。

这一收一放，使得观众从一个展厅出来，经过连廊，到了另一个展厅，这个过程，有了一定的节奏感。

我们可以假设一下，空间边界的安排，变化一下。如果展厅的混凝土墙面都取消掉，都变成彻底透明的玻璃幕墙，而连廊变成混凝土隧道了，有什么样的效果。如下图所示。

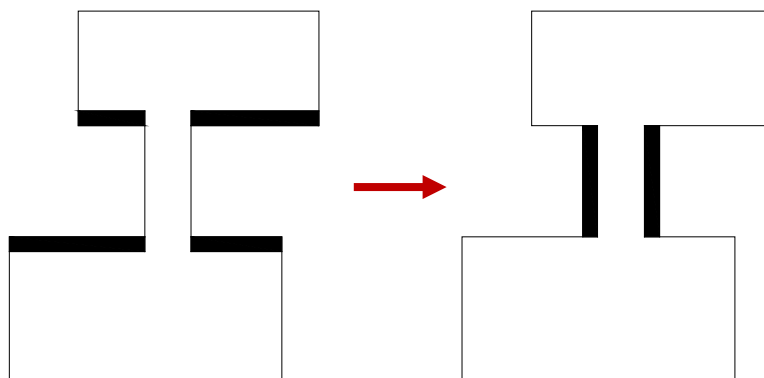


图 2.3-15 空间边界的安排

上面这个问题，留给读者思考吧。

在建筑设计的基础上，到了真正的使用的时候，还要更复杂一些。到了布展的时候，在空间的安排上，还可以继续深化。参见下图：



图 2.3-16 实际使用的状态

如上图所示，室内空间的这种分隔手段，是用不透明的硬的体块，去切割空间。在我们的假想空间中，可以放几个方块，再测试一下。视线整合度的计算结果如下图所示：

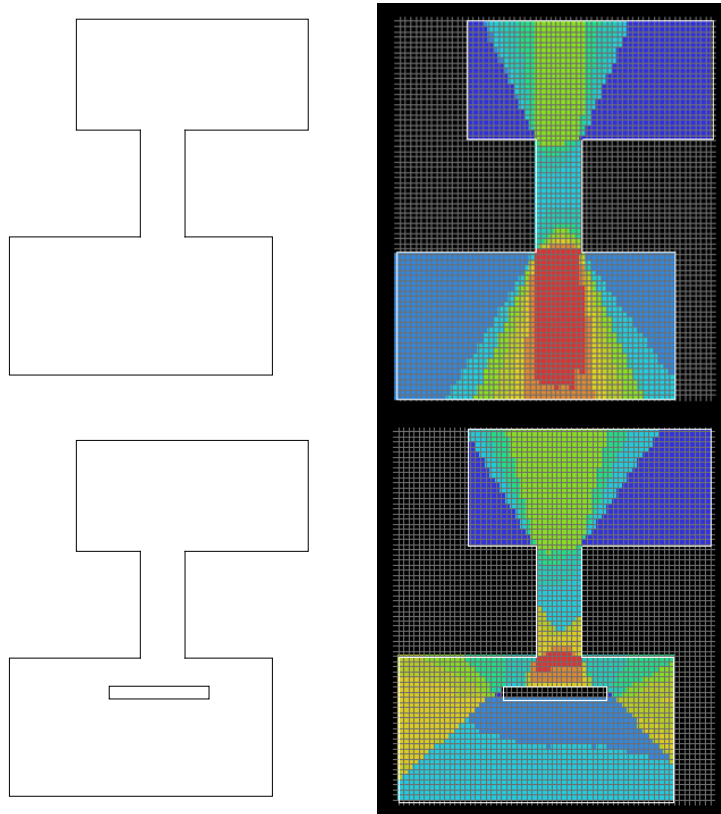


图 2.3-17 室内空间的使用

上面这张图的意思是，在已有的空间结构中，用硬的体块再去切割的话，并没有改变通道本身视线整合度低而两侧出口附近的整合度很高的状态，造成的改变都是局限在上下两个大的体块的内部。在展厅内部的空间分割，不会影响建筑设计所营造的几个体块之间过渡的那种节奏感。

到了这里，可以总结一下。上文提到的这个连接通道的长短的问题，其实就是古典建筑学当中，墙厚的问题。

西方人的砌体建筑的这种工程传统，就造成，他们的墙体厚度非常可观，进一步造成其门窗洞口的厚度也很大。门窗洞口本身，可以被视为一种独特的空间，这类空间具有其独特的艺术品质。这类空间可以独立地发挥某种艺术上的、审美上的功能。

甚至古典建筑的柱廊，也是这么个有厚度的空间。厚度不但为空间和材料的艺术表现提供了更多的可能性，更进一步地，也为建筑有关的时间层面上的问题，提供了一种解决思路。

西方的建筑大师，到了现在，对于墙厚问题也不是回避的态度，而更多的是积极探索、积极利用的那种态度。请注意下图中红色箭头所指的洞口：

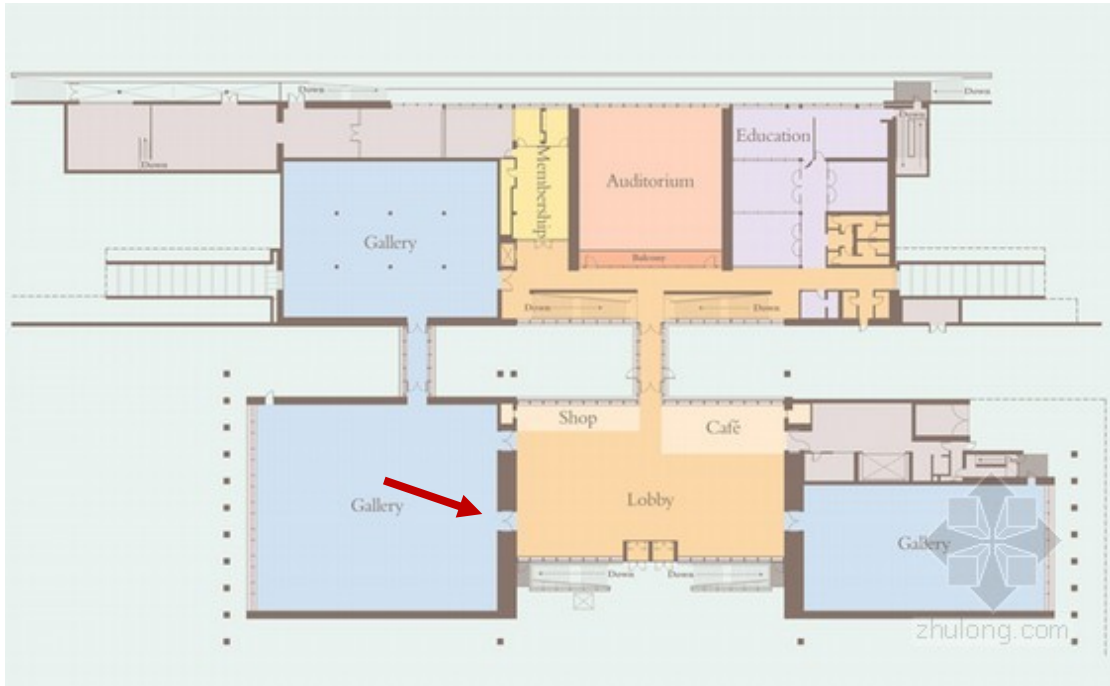


图 2.3-18 洞口厚度的问题

混凝土结构工程发展起来以后，现代建筑的墙体，失去了结构上的必然性，失去了结构上的重要性，可以达到没有重量、无限轻薄、无可把握的状态。墙体的消隐，好处自不必说，坏处是损失了砌体建筑的时代，空间转折的一些自然而然的可能性。

柯布西耶提出了现代建筑的五项原则，包括：底层架空，屋顶花园，自由平面，自由立面，水平长窗。这都是在墙体的消隐的基础上，才提的。

回头去看柯布西耶本人的作品，他并未放弃在墙厚问题上的探讨。

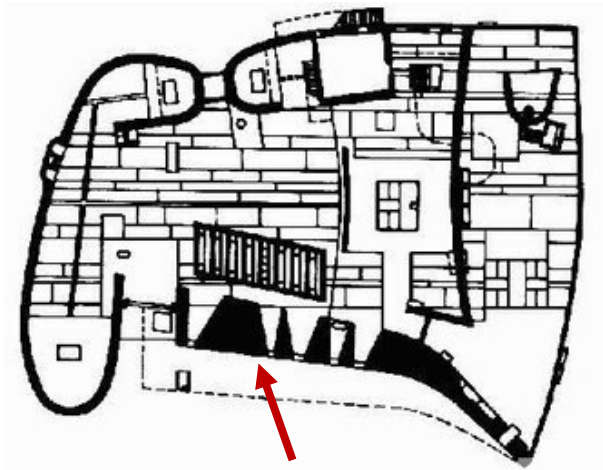


图 2.3-19 朗香教堂的情况

也许，这个墙体厚度的问题，门窗洞口的厚度的问题，墙体的艺术地位的问题，空间之间衔接过渡有何种可能性的问题，在有了空间句法的分析工具以后，更加值得我们深入地反思一下吧。

3) Visual Clustering Coefficient

这个 Visual Clustering Coefficient，直译比较困难，大概意思是说，空间边界在视觉方面的限定效果的强弱。先把计算结果摆上来看。

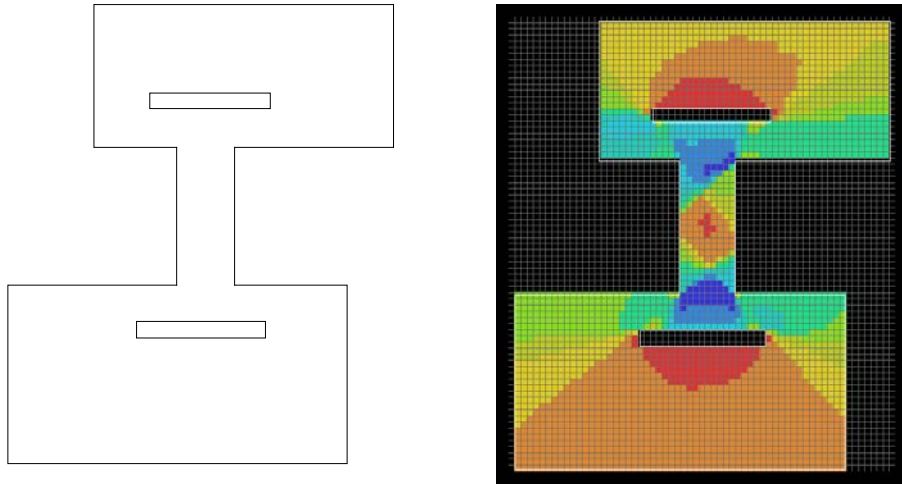
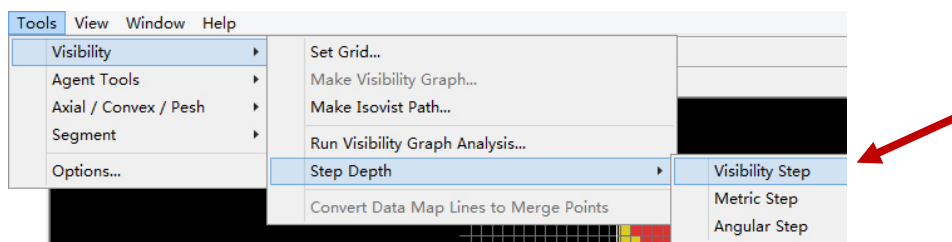


图 2.3-20 计算结果

DepthMapX 软件的能力有限，无法分辨空间边界的透明状态，只能假定空间边界都是不透明的实体界面。有经验的建筑师，可以在软件给出的结果的基础上，估计一下。

Visual Clustering Coefficient 的算法，最早是两位数学家，Duncan J. Watt 和 Steven H. Strogatz 提出的。他们研究的对象是人际关系，他们对人际关系进行了拓扑学分析，总结了计算公式。有了算法以后，DepthMap 很快就开发了相应的功能，获得了不少启发性的结果。

这个算法针对的是某个单独的元素。假设我们选定了一个单独的元素，称作 g_1 ，选中的这个元素处于黄色高亮的显示状态。通过 Visibility Step 的命令，可以找到所有与 g_1 相距一个视线深度的元素。如下图所示：



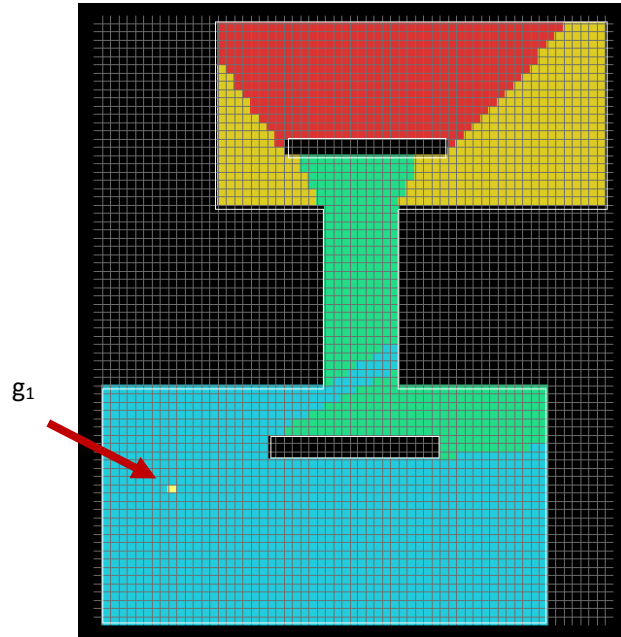


图 2.3-21 g_1 元素的 Visibility Step Depth 的情况

上图中，蓝色的区域，是距离 g_1 一个视线深度的区域，绿色的区域是距离 g_1 两个视线深度的区域。我们可以定义，蓝色区域当中包含 k 个元素，绿色区域当中包含 K 个元素，有了这两个值以后，就可以计算 g_1 对应的 Visual Clustering Coefficient 的值了。这个值将来反馈回来，记在 g_1 的头上，当系统中每个元素都照此做了一遍以后，就可以生成彩色的特征图了。

针对 g_1 元素，它的 Visual Clustering Coefficient 的计算公式是：

$$\text{Visual Clustering Coefficient } (g_1) = \frac{K}{k(k-1)}$$

这个公式的意思是，首先找到某个单独的元素，要研究这个元素的性质。从这个单独的元素眼中，向外看，一共看到多少个其他的元素，这个数值就是分母。就是上图中蓝色的那一块地方的元素数量。蓝色的区域也被称为 g_1 的视域。

接下来，在这个元素的视域范围内活动活动，再向外看，又看到了一些新的元素，这些新的元素的数目是多少，反馈回来，就是上面公式中的分子。

也就是说，用相距两个视线深度的元素的数量，和相距一个视线深度的元素的数量，做一下比较。

如果一个元素，它的视域范围内的东西很少，但是在这个视域范围内活动活动，再向外看，却可以发现很多东西，那么，这个元素被遮蔽得比较厉害。如果一个元素，其视域范围内东西就很多，在视域范围内活动活动，向外看，也没啥新东西了，那么这个元素就没有受到什么遮蔽。

这个话说得通么？

对应到上面的公式中，如果这个 Visual Clustering Coefficient 的值越高的，表示距离选定的这个元素，两个视线深度的东西更多，而距离它一个视线深度的东西却较少，这表示该元素受到较强烈的遮蔽，它附近的界面，在限制和约束视线方面，作用比较强。

如果这个 Visual Clustering Coefficient 的值越低的，表示它受到的遮蔽越弱，它附近的

空间界面在视觉上并无太大限制作用。

针对平面中的每个元素，都进行了 Visual Clustering Coefficient 的计算以后，每个元素头上都有数值了，就可以按照 DepthMapX 当中赋予颜色方法，制作一个数值特征图。

针对前面提到的假想空间，Visibility Clustering Coefficient 的计算结果中，红色的区域，是全系统当中，视线遮蔽得最厉害的地方，对么？这个计算结果也可以解释成视线受到压迫的强烈程度。下面弄个大师作品上来分析一下。

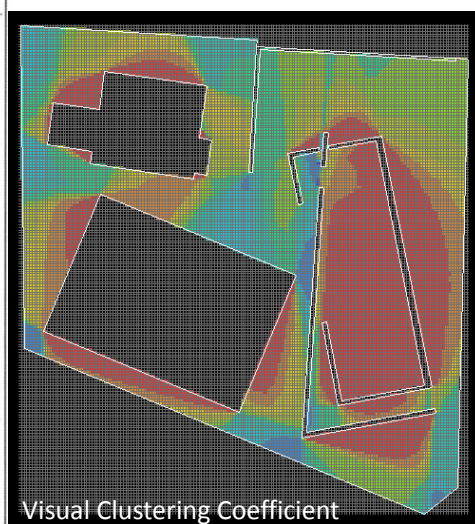
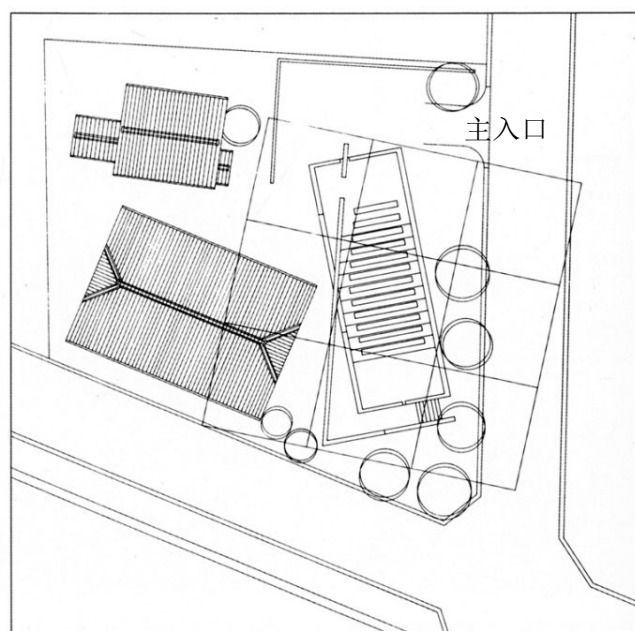


图 2.3-22 光的教堂

根据上面的分析可以看出，光的教堂，其室内空间，即使是在这么个小院子围合的空间系统当中，都算视线压迫的程度比较高的。在这样的视线压迫的基础上，再开个十字形的窗，空间的宗教主题的表现，就更加有力了。

安藤忠雄设计的这个光的教堂，不用空间句法分析，也能得到相同的结论。当初设计的意图就是这样的。软件并不能告诉我们应该做什么，不过，我们做了一个东西以后，验证一下想法实现的程度，还是可以的。比如，在室内设计的时候，某种半围合的手段，创造的这个局部空间，到底围合效果如何，可以用这个 **Visual Clustering Coefficient** 来衡量一下。在软件给出的计算结果的基础上，根据设计师自己的经验，在材料、灯光的配合下，可以修正视线遮蔽的实际效果，实现特别细腻的那种视线控制。

关于视线分析的其他公式，放在附录中了。

本章小结

第三章 轴线建模

DepthMap 的这种轴线建模方式，已经彻底淘汰。本教程这部分及以后的内容，不再更新了。

分析平台采用 ArcGIS，用 Cadiff 大学出品的那个 sDNA 插件进行分析就行了。在 CAD 中，轴线建模的时候，直接采用道路中心线的方式建模即可。

本教程前面的章节中，了解了用无限细分的方格网去代替原来的空间这种思路。这是数学上的微分的思路。也可以采用概括的方式，用直线段去概括空间，用直线之间的连接关系，去代替原来的空间结构当中的连接关系。根据 Bill Hillier 的说法，数学上已经证明，用直线来代替的思路，在数学上具有唯一性，建模思路是成立的，方法论上是靠得住的。我并未看到给出证明的那个文献，数学上的证明，与图论有关。而图论的一整套知识，也超出了我的范围。本教程中，就默认，轴线图的方法是可靠的，希望读者不要去纠缠方法论的问题。

用直线来概括空间，可以称作“将空间结构转译为轴线图”，大概的感觉，参见下图。

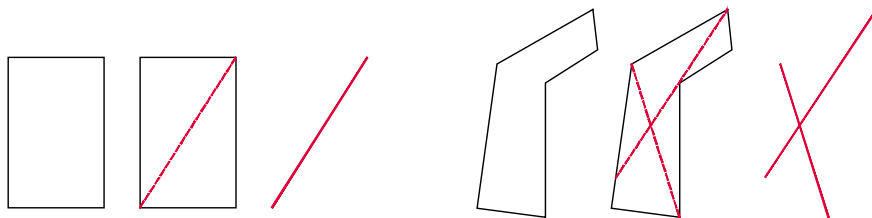


图 1.3-1 将空间结构转译为轴线图

在空间句法中，建立 Axial Map 的规则，是这样的：

保持凸空间的连接关系不变，用最长且最少的轴线穿过所有的凸空间。

这个所谓“最长且最少”，在文献中，是“longest and fewest”，两个条件必须同时满足。DepthMapX 当中，也提供了这个功能，对于不太复杂的空间系统，空间边界在 CAD 中完全用多段线闭合的情况下，软件可以自动生成轴线图。

但是这个 DepthMapX 的软件，体积一共才 2M 不到，这么小的软件，稳定性各方面是存在问题的。不能一味靠软件去生成轴线图。我做过西安的研究，城市底图拿过来，发现 CAD 的制图规范都没有，线条很乱，而且最后手工勾出来的轴线一共九千多条，这么大的量，靠着软件去生成，根本没戏。

更加合理的工作方法，是学会手工绘制轴线图。绘制轴线图，就是个手艺，要通过一定量的练习，通过跟老师交流、纠错，逐渐掌握绘制轴线图的方法。

有了经验以后，眼睛会变得比较贼，哪里容易出问题，瞄一眼就知道，抓住轴线图的一些细节，检查一下，如果细节说得过去，这个轴线图就是靠得住的。

绘制轴线图的技术，我是跟在王浩峰老师后面，跟了一年有余，不断地纠错，不断地打磨，才入了门。在写这个教程的时候，仍然有些细节的处理上，不是很牢靠。读者如果是初学，首先是找到好的老师，没有合适的老师，自己折腾，很难靠谱。在有了好的老师的基础上，再慢慢练习，慢慢熟练，慢慢掌握。

轴线图的这种概括方式，去研究大尺度的，比如城市空间，是有特色的。特色在于，特

别复杂的空间结构，过去是很琐碎的，而在轴线上，杂七杂八的东西就没有，整个系统的状态很简单，因果关系很简单，全系统的状态可以说一目了然。敝人本科是学建筑学的，弄这个轴线图一段时间以后，我个人的关注的视角，也逐渐从小尺度的建筑空间，逐渐转移到较为宏观的城市的角度。还是大的好玩儿。

第一节 用 DepthMapX 自动生成轴线图

随便假设一个空间结构，在 CAD 当中，空间的边界用多段线闭合好，通过 dxf 格式的文件，导入到 DepthMapX 中来。

在 DepthMapX 当中，通过 Ctrl + N 新建一个工作空间，然后通过 Ctrl + I 导入已经准备好的 dxf 文件。导入进来后，底图是居中最大化显示的，不需要做任何事，直接去窗口顶部，命令行那里，有个 Axial Map 的命令，找到这个图标，左键点击一下，鼠标指针本身，也变成了铅笔的形状。到我们要研究的空间的内部，随便找一个地方，左键点击一下，软件就会自动生成轴线图。如下图所示。

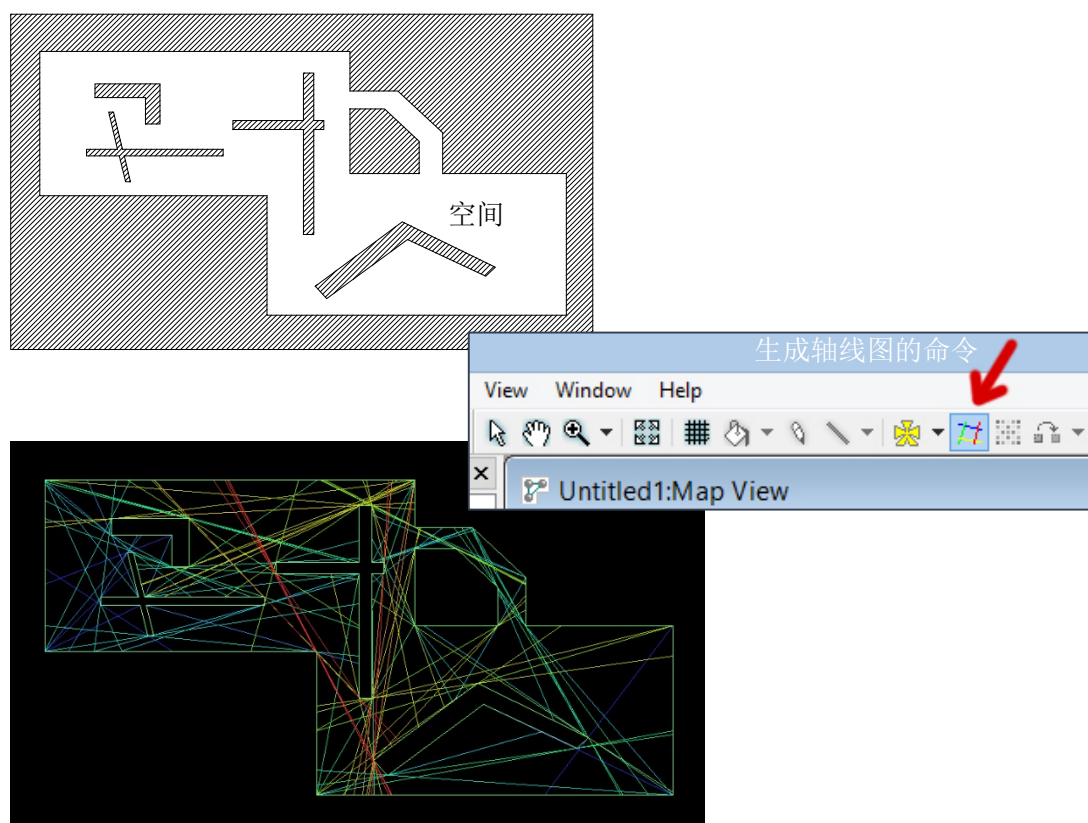


图 1.3.1-1 将空间结构转译为轴线图

上面这个还不是“最长且最少”，还要再加工一下。从 DepthMapX 顶部，下拉菜单，选择 Tools → Axial/Convex/Pesh → Reduce to Fewest Line Map，就可以得到数学上严格的最长且最少的轴线图了。操作步骤及结果参见下图。

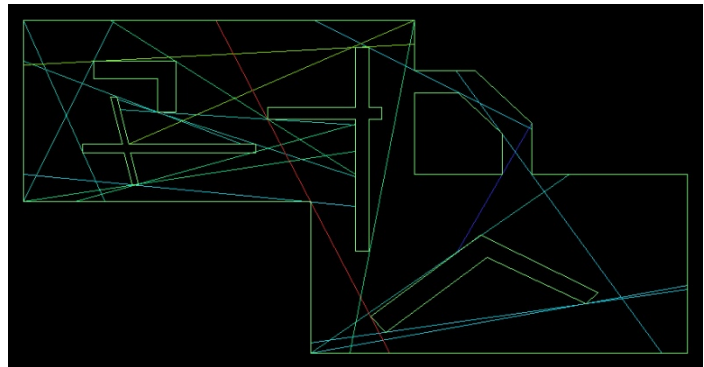
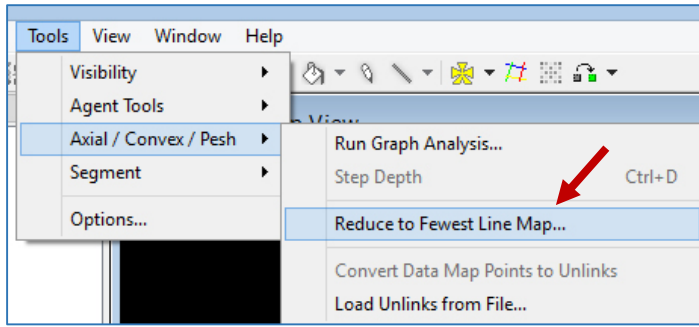


图 1.3.1-2 Reduce to Fewest

用最长且最少的轴线概括了空间结构以后，就可以把轴线图抽象成一个拓扑结构了。每个轴线算一个元素，线与线有交叉的，算成两者直接相连。

假设有三条轴线，可以将他们的关系简化成这样：

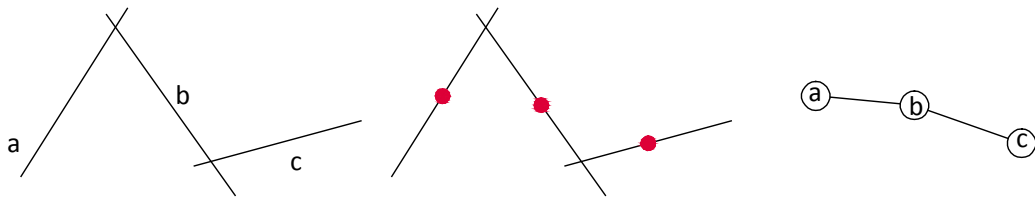


图 1.3.1-3 将轴线图变成拓扑结构

将轴线图变成拓扑结构图以后，就可以按照前面介绍过的套路，进行空间重映射，计算整合度什么的，完了回来给每个元素赋予颜色了。

这一节所介绍的，就是个基本道理，如果算法不过关，不可能产生唯一的“longest and fewest”的轴线图。

尽管算法过关了，可是软件的稳定性的问题，包括底图整理的难度，都造成，研究城市空间等大系统的时候，我们根本不会去用这个功能。轴线图都是在 CAD 当中手绘出来的。

绘制轴线图，就是个手艺。几万条线的大系统，虽然在每个局部都是有明确的绘制规则的，但是总体上的把握，就是凭手艺，凭感觉来的。

绘制轴线图的方法，放在下节介绍。

第二节 用轴线图研究城市

用轴线图去表达城市的空间结构，大概是个什么感觉，可以先摆两张图看看。如下图所示。



图 1.3.2-1 将城市空间结构转译为轴线图

上图中，黑白色块的这个图，是个 jpg 格式的图片，从某个出版物上面扫描下来的。为了写论文，收集资料的时候，要把这个图准备好。然后，在 CAD 中导入这个 jpg 图，衬在下面，把尺度弄对，按照我们的规范，手工描出街网的轴线图。有了轴线图后，就可以导入到 DepthMapX 当中，用软件去进行了。

在做城市研究的时候，轴线图是最基本的。在此基础上，还有线段模型等建模思路。

没有轴线图的时候，有些城市功能的分析，也是可以分析的。例如，将城市当中的某一类功能，做个调研，将其空间分布的状态，进行总结，也是可以的。如下图所示。



图 1.3.2-2 城市功能的分布，调研资料

上图是 1940 年前后，西安的具有一定规模的工厂企业的分布情况。这些企业包括面粉厂，化工厂等。不用空间句法的分析，仅仅是通过直观的感受，我们也可以说事儿。比如，我们可以说，哪里哪里的工厂分布比较多。有了空间句法以后，我们可以说的话就多了，分析的思路也有变化。

这是敝人的硕士论文当中的内容。到了答辩的时候，王鲁民教授问了两个问题，切中间句法这种研究方法的要害，不妨在本文中先扯一下这事儿，再展开介绍后面的内容。

这两个问题是：

1.城市空间，本身是个复杂的系统，有街道的空间，有建筑物内部的空间，为什么在空间句法当中，仅仅建立街网的模型？

2.城市空间中，靠近中心附近，建筑物的容积率可能更高，能够容纳更多的城市活动。仅仅用街道的情况，去描述城市当中的各种活动，是将三维空间中的状态，变成了二维的，合理性在哪？

回答第一个问题，轴线模型建立的问题。

轴线模型，最基础的合理性在于，它是用来概括凸空间的。无论是建筑物内部的空间，还是街网空间，都可以用轴线图的方式来概括。

但做研究的时候，有个层级的划分。比如有人研究城市当中的步行系统，那么他建模的时候，轴线图上的每一根轴线，都是在概括那些人可以走到的空间的。有的人研究机动车的通行系统，那么每一条轴线，概括的都是机动车可以开得到的空间。要有个研究范围的划定，要有个取舍。

做西安的研究，那个 jpg 底图的信息很有限，只能分辨到色块和当中的道路，所以建模的时候，只能深入到街网这个层级。

基础资料总是制约着我们。对于读者来说，针对你的研究对象，到底能够收集到什么样深度的素材，也是很要紧的。有一条素材，我们能够说一句话。没有素材，很多东西通过想象力，通过感觉，通过感悟，也能知道，但是在学术上就成了没有支撑的东西，很难获得承认。

回答第二个问题，容积率的事儿¹。容积率的事儿，直接回答，逻辑过程太复杂，拆成三个小问题，可能读者理解起来，会方便一些。

- ① 建筑物与街道是什么关系？
- ② 轴线概括了什么？
- ③ 功能主义建筑学的根本要求是什么？

街道与建筑物是较为分割的两个系统。街道往往是公共空间，对于街道的使用，在现代城市中，是没有身份限制的。街道作为公共空间，平等地向所有人开放。你虽然都是平等地开放的，但是人家愿意不愿意从你这里走，却是很大的问题。

建筑空间与街道空间不同，建筑物一定会有各种隔离措施，将内外空间区分开来。

有的时候，并不需要真正的物理上的隔离，仅仅需要社会上的人的观念，人的共识，就可以形成隔离。英国不是有个流传很广的说法么，“风能进，雨能进，就是国王不能进”，这是法治社会才有的情况。

建筑物与街道的关系，是开放空间与隔离起来的的空间的关系。由于隔离起来的的空间，在使用上具有排他性，就造成，在不同的人眼中，空间的状态是不同的。而公共空间，在不同的人眼中，都是一样的。

建筑物与街网的对立，私有的、隔离的空间与公有的、开放的的空间的对立，就造成，轴线图这种方式，概括的，其实是一种界面。也就是说，轴线图概括的是城市当中，公共空间与私有空间的互相转换的这个界面。

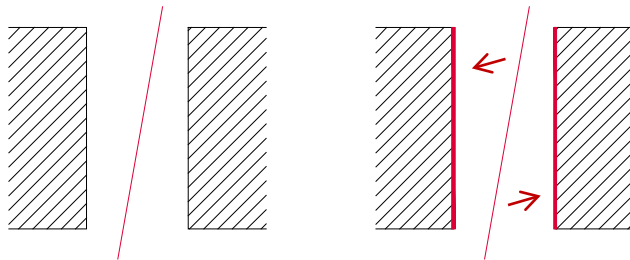


图 1.3.2-3 将轴线的值，push 到空间界面

假设我们在轴线图的基础上，计算得到了某个单独的轴线的整合度的数值，我们可以把这个数值，推送给它两侧的空间边界。这个功能，在 ArcGIS 中，早就有了。

在推送数值的这个思路的基础上，如果有一个十字路口，有多条轴线，我们就会推送不一样的值到不同的空间界面，这样，会给我们很新鲜的一种观察空间的方法。参见下图。

¹ Bill Hillier 写的那个《空间是机器：建筑组构理论》当中，对这个问题有深入的阐释。除此之外还有一些文献。问题是他说的东西太多了，相关的文献太多了，翻译成汉语，存在一定的扭曲，语言跟思想变成了两张皮，让人眼花缭乱，很难从总体上加以把握。本文中说的东西，依据是有的，但是从经典文献当中，逻辑上一步一步推到这里，逻辑过程有点儿复杂。

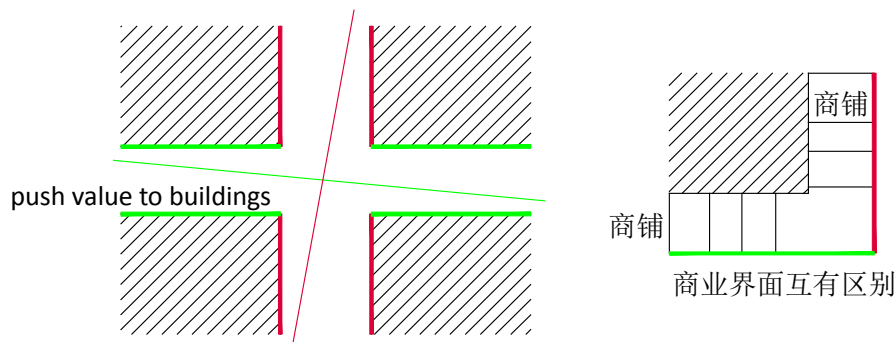


图 1.3.2-4 将轴线图的值，push 到空间界面

上面这个图，意思是，将轴线图的数值，比如整合度的数值，推送给其相邻的建筑物界面，这样，我们就知道建筑物的界面，有什么特征了。在城里边儿转悠转悠，经常有体会，一条街上小商店，类型都是差不多的，但是一转弯，就变成另一种类型的商业集中在一起了。从空间句法这里，可以给个解释，就是空间界面本身，是有区别的²。

没有空间句法之前，空间界面的区别，没法谈。

在此基础上，继续推理。

因为功能主义的现代建筑，和功能主义的现代城市规划方法，强调的就是建筑物在功能上的专门化，强调的就是土地利用在功能上的专门化，就造成，出行成了大问题。

我们中国的传统，不是这样的。我们是弄个院子，把自己围起来，吃喝拉撒都内向地解决。在西方人的生活中，大门不出二门不入的情况，是难以想象的。西方人是吃在一处，住在另一处，工作又跑到另外一处，在不同的地方，干不同的事儿，人的活动在城市当中广泛地分布，是外向地解决的。

轴线图所概括的这个公共空间，其实就是一个出行的空间。甚至可以说，轴线图表达的是出行这个功能，在空间上的状态。

那么好了，针对某个具体的空间，它发挥什么功能，其实是由人的出行决定的。在什么地方，方便干什么事儿，人们是有判断的。在什么地方，方便干什么事儿，人们就会提出要求。

某个地方，承载什么功能，要看人怎么要求。随着时间，人的要求是在变的。随着时间，建筑空间的状态也是在变化的。这就形成了一种“功能调试”的状态。

一个具体的地块，上面承载了什么样的功能，上面承载了多少的容积率，是随着历史进程，随着城市发展的进程，逐步在功能上调试，在空间上增殖，慢慢发展起来的。

城市空间的状态，也不是一成不变的。在人对于功能的要求，在社会对于功能的要求，在这些要求的基础上，空间的状态会做出相应的改变。空间的合理利用，全社会都盯着呢。如果空间怎么改造改造，就更有利了，人们力所能及的情况下，马上就改造了。城市是个复杂的巨系统，城市一直处于变化当中，城市的空间结构，一直处于变化当中。所有的针对城市的研究，其实都是截取出来一个时间的切片，研究的是某个时间段内的情况。

² 每个人都发财，是不可能的。每个商铺都成功，也是不可能的。商铺的成功与否，并不需要真刀真枪地去经营一下，来验证。通过空间句法这种研究方法，商铺的商业价值，其实是可以得到衡量的。在本文写作的2013年，新建的楼盘，下面几乎都是一圈商铺围着。这种设计思路，将会造成什么麻烦，在空间句法这里边儿，其实是可以预测的。

针对某个具体的地块，人对它提什么功能上的要求，与这个地块所处的出行条件，是有紧密的关系的。甚至，话可以倒过来说，当空间的状态改变了以后，也会影响到出行的状态。空间的状态，与出行的功能上的要求，是个互相驱动的关系。

出行的事儿，与空间结构的事儿，互相具有规定性，是双向的约束。容积率是空间结构在局部的增殖程度的反映，容积率是空间结构繁密程度的一个表征，容积率的事儿，其实就是空间结构的事儿。

容积率其实还是个较为简陋的指标，还可以从投资密度（每平方千米多少亿元）、能源消耗的功率密度（每平方千米多少瓦特）等指标，去衡量空间结构。根据新闻报道，深圳有的超高层办公楼，这个楼内一年就能产生几十亿的税收。如果将税收的情况，与土地利用做一个比较，可以自己定义一个税收密度（每平方千米多少亿元），也能给我们提供一些城市空间结构的信息。可以将这些东西的值，推送给轴线，将来再把轴线图单独拎出来，进行考查，等等吧，方法很多。只要是能想得到的信息，只要是这条信息带有地理位置的，就可以将其与轴线图关联起来，这种思路是很自然的，不论是在空间句法当中，还是在 ArcGIS 当中，都有类似的功能。

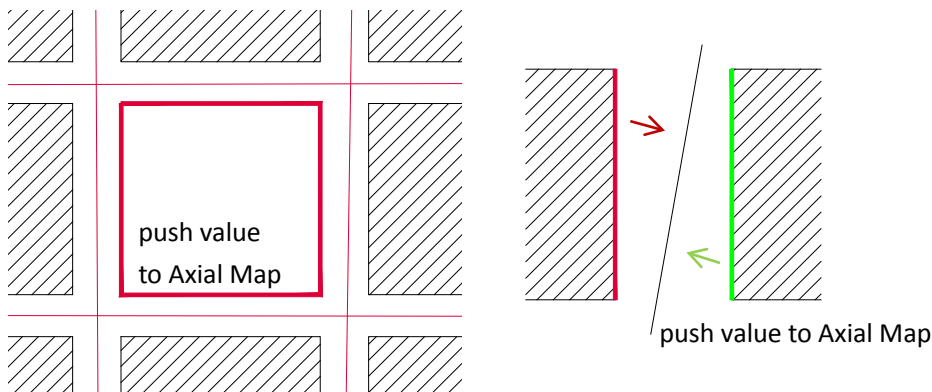


图 1.3.2-5 将地块的信息，推送给轴线

因为轴线图概括的，是出行的这个功能，而出行的事儿与空间结构的事儿是直接相关的，所以，用轴线图可以从一个侧面，去研究容积率的事儿，去研究地块功能的事儿。

上面这段话比较绕，读者如果想要深刻理解这里边儿的思想，可以详细研究一下《空间是机器》这本书的英文版。

第三节 手绘轴线图的方法

在 CAD 当中手工绘制这个轴线图，有两个要求：

1. 交接处要稍微出头
2. 对于空间的概括要准确

在手绘的时候，一定要遵循“出头”的原则。因为，轴线图经常会发生微调，在 CAD 当中，一个线段，端点稍微移动一下位置，与这条线相交的其他的线，可能就脱开了。不注意的话，会造成连接关系的丢失。参见下图。

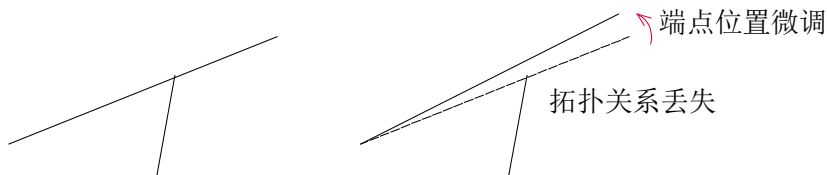


图 1.3.3-1 轴线交接的可靠性

将城市空间结构转译为轴线图的过程中，一边画一边调整。在不断的调整过程中，去寻找那些更加简化，更加合理的概括方式。

面对一条 S 形的街道，如何用轴线图来概括它呢？参见下图。

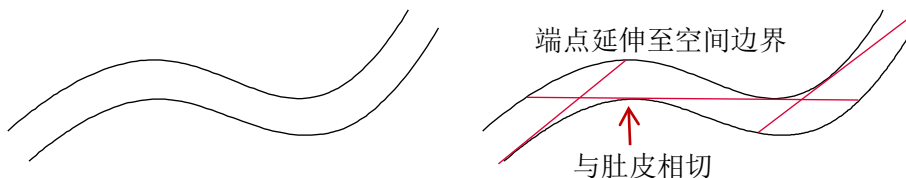


图 1.3.3-2 S 形街道的处理办法

上图中，应用的就是“最长且最少”的轴线图的思想。

当然，有的时候，空间界面的边界，是个圆肚皮，从哪个位置相切，可以变的。轴线的位置，可以根据与它相连的空间的状态，进行微调。参见下图。

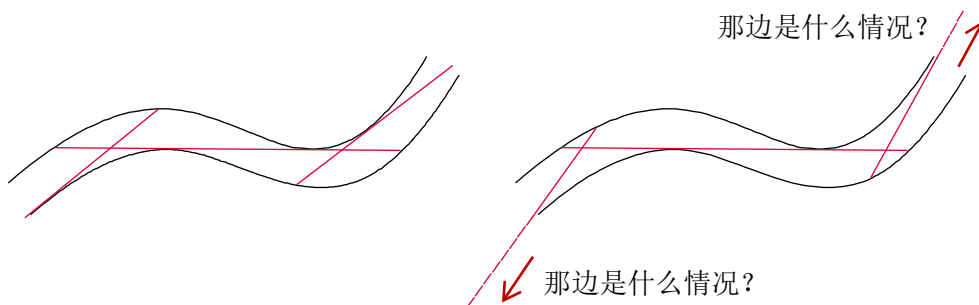


图 1.3.3-3 对轴线图进行微调

上图的意思是，先有了一个轴线图的状态，然后可以进行微调，让轴线尽量延伸得远，同时，与空间边界的相切关系要正确，有的时候，稍微偏离空间边界，割到地块里面去一些，

这不要紧。只是割进去的程度不要太离谱就行了。轴线割进去太多，可能就需要将其切分成几段了。

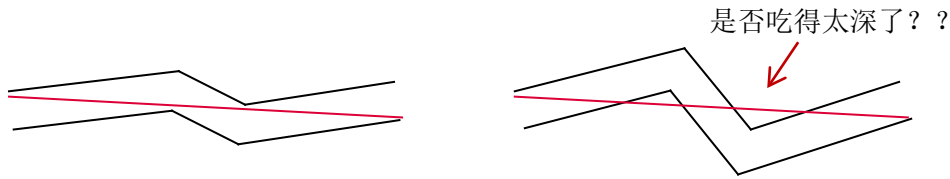


图 1.3.3-4 对轴线图进行微调

除了曲线的街道的麻烦，还有道路的交叉口、立交桥一类的问题，比较麻烦。

现代的城市中，交通系统的复杂程度越来越高，交通系统本身出现了专业化的趋势。交通系统越来越分化，不同的子系统之间，转换关系越来越复杂，交通节点的单体建筑的复杂程度甚至超出了大系统的复杂程度。因为复杂，所以引人注目。

本教程中，将典型的情况先罗列一下，希望读者借此反思空间系统的合理概括的问题。将来练手艺的时候，有个判断标准。

情况 1：平交的交通转盘

交通转盘的情况，是所有道路都连接在转盘上，所有流量都在里面绕一下，绕到自己要的那个出口，再绕出去。按照绘制轴线图的简单规则，大概可以按如下方式来处理：

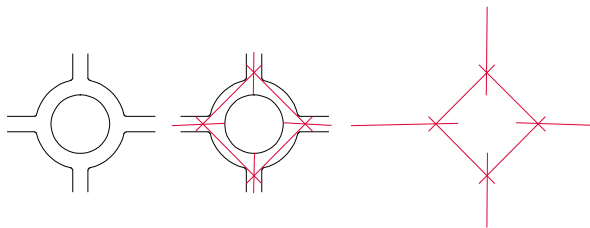


图 1.3.3-5 交通转盘的处理

为了符合轴线“最长”的要求，线段的端点往往都在角落里面，与道路中心线呈很大的夹角。轴线图的走向，往往不是那么横平竖直的，看起来歪三扭四的。

上图中的这种处理方式，也就是认为，两条街在拓扑关系上不是直接相连，而是要通过交通转盘，转折一下，拓扑关系上要增加一步，至多两步。

情况 2：不规则的交通转盘

在西安的古城，一个城门处，出现了长方形的转盘，如下图所示，怎么处理呢？

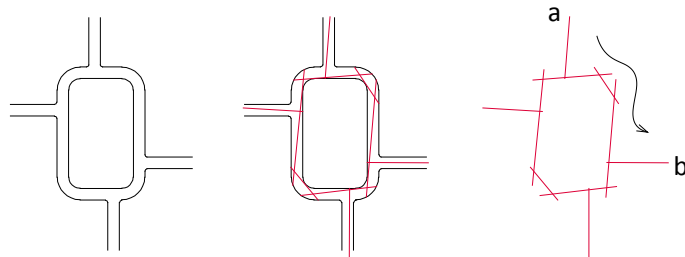


图 1.3.3-6 交通转盘的处理

上图中的这个交通转盘，按照上图给出的轴线图这样处理，合适吗？

我们认为是不合适的。考查 a 与 b 的拓扑距离，软件当中是按最短拓扑步数来计算的，按照箭头的走向，软件认为 a 与 b 相距 4 步。实际的交通状况，是按照逆时针的方向转的，那样 a 与 b 的距离就更多了。

而这个交通转盘，除了解决交通流量的问题，并无其他城市功能活动发生。在转弯处的许多轴线，能代表什么意义呢？再者，一个司机开着车，从 a 去 b，从空间认知的角度来说，是认为从 a 出发，一步到了转盘，再一步到了 b，中间怎么转的，不会注意。

所以，我们可以将轴线图调整成下图这样：

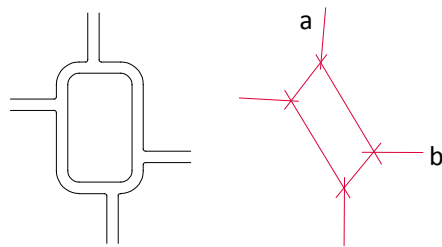


图 1.3.3-7 简化一下

像上图这样，软件计算的时候，a 与 b 距离仍然是两步。从轴线模型的意义来说，从 a 出发，一步到了转盘，再一步到了 b。是比较合理的。

对于交通转盘，如果不这样简化，而坚持要用众多的断轴线去拼圆弧，是什么后果呢？我们假设一个更加夸张一点儿的情况。如下图所示。

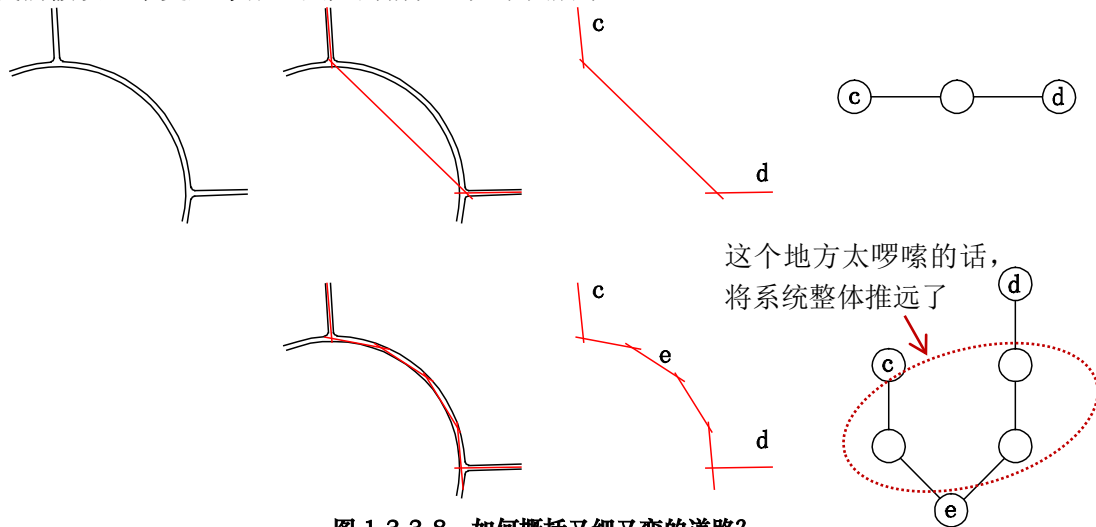


图 1.3.3-8 如何概括又细又弯的道路？

注意，我们现在的大前提，是 e 位于纯交通性的道路上。

如果按照上图第一种情况，用较为直接的方式去概括，那么 c 与 d 的关系是拓扑步数较少的情况。而按照上图中的下面这种方式，非要用许多短的轴线去拼这个圆弧，那么，你看 c 和 d 的距离就更远了，而这些步数，在城市功能方面，能有什么价值呢？

以 e 为中心，进行一下空间重映射，看一下拓扑结构图，就更加明白了。如果道路的幅面再窄一些，就需要更多的步数。转折的步数越多，就相当于以 e 为中心，把整个系统都推得更远了。

越是用细密的轴线去拼这个圆弧，就越使得类似于 e 这样的居中的轴线，在全局深度上吃亏，进而在整合度的数值上吃亏。所以，用最简单的一步连接的方式，去处理交通转盘，是合理的。

情况 3：带立交的转盘

在西安的朱雀路环岛，出现了如下图所示的情况。如何用轴线图去概括呢？

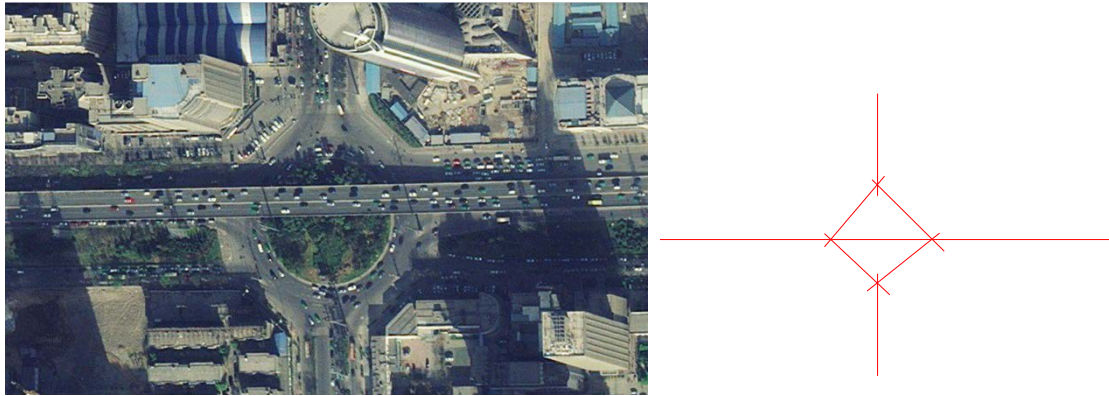


图 1.3.3-8 西安的朱雀路环岛，如何用轴线去概括？

王浩峰老师给出了如右侧所示的这种轴线模型。水平方向是一根轴线直接拉通的，而垂直方向是需要通过交通环岛转折的。高架系统并不是封闭的，沿着高架走一段，就有出口，所以在轴线模型的处理上，就用单线就行。

情况 4：隔离的高架系统

有的大城市中，高架系统在城市当中隔离起来，走了很远都没有出入口。而在高架系统的下面，普通的城市道路互相联系，密度也很高。在平面图上，两者又重叠在一起，该如何处理呢？参见下图。

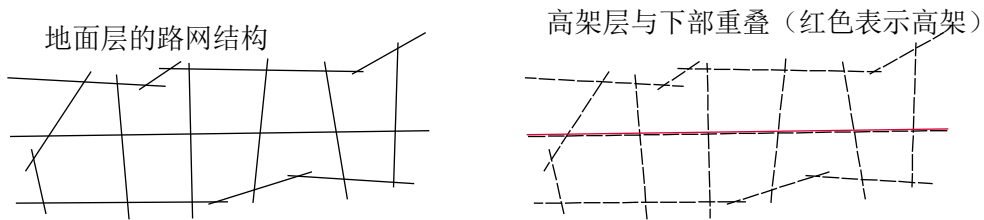


图 1.3.3-9 隔离的高架系统如何处理？

在建模的时候，为了将来操作的方便，可以分别建模。地面层的轴线，就按照一般的建模规则，画出来，而上部的高架系统，轴线图就整体偏移一个距离。保持上下两层各自的拓扑关系，在各自的子系统内部是正确的，就行了。将来，到了 DepthMapX 当中，再用 unlink 命令，将两者分开就行了。参见下图。

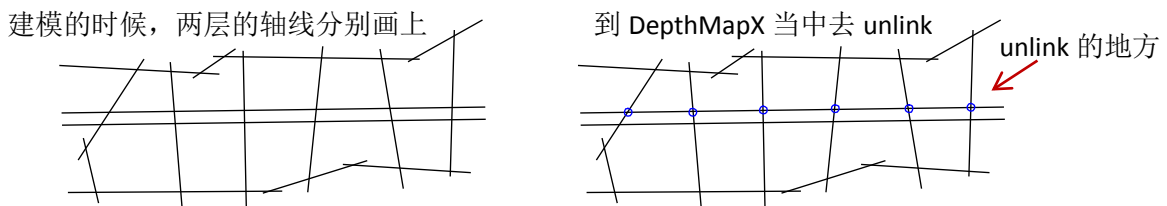


图 1.3.3-10 在 CAD 中各自建模，在 DepthMapX 中设置拓扑关系

情况 5： 复杂的立交系统

在资料比较详尽的情况下，立交系统的建模方式可以有多种。在城市的尺度上，多费个三五个拓扑步数，去描述立交系统，也是可以的。或者仅仅考虑道路的联通关系的正确，转折的细节都甩掉，也是可以的。

在上一版的教程中，以郑州的紫荆山立交桥为例，进行了分析。本文中继续引用这个例子。立交系统的状态，参见下图。



图 1.3.3-11 郑州的紫荆山立交桥

比较抠细节的建模方法，就是忠实地将空间的转折关系都建出来，主路是主路的轴线，辅路是辅路的轴线，该 link 的就 link，该 unlink 的就 unlink，这也是可以的。

或者，干脆从交通关系考虑，这不是个五岔路口么，龙去脉能走得通就行了，细节都抛掉，也是可以的。如下图所示。

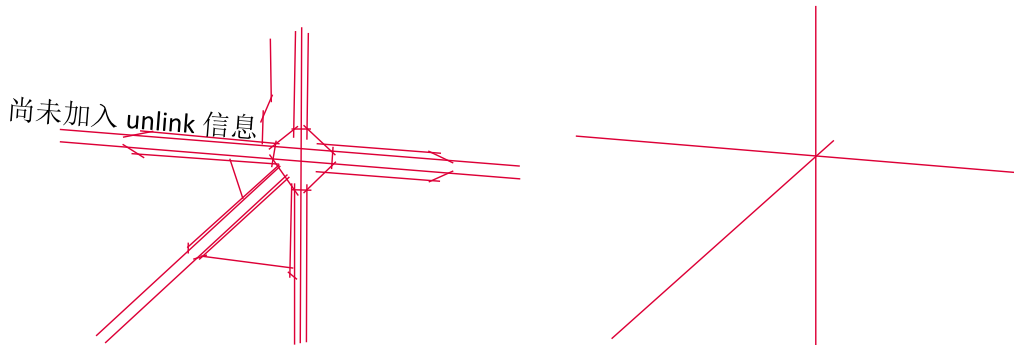


图 1.3.3-12 建模的思路

在面对城市尺度，面对几万条轴线的情况下，不要贸然决定采用哪种建模方式。先把城市当中的立交系统，都观察一下。交通系统的不同子系统之间，找那种最复杂的转换节点，比如深圳的高铁站，上海的虹桥枢纽，找这种最复杂的节点，先研究一下，然后再决定建模的方法。

有的时候，交通系统本身就是研究的目标，越详尽越好，那么建模的时候要照顾到。还有些情况下，是投资决策的时候要用，只关心有限的几个地段的情况，那么就可以做相应的简化。

只要轴线模型的逻辑上，是一致的，就行。要简约的都简约，要抠细节的都抠细节，全系统的轴线模型，其建立规则，在逻辑上是内在一致的，就行了。

这个绘制轴线模型的事儿，也是个手艺，多找些典型的案例，练练手，慢慢地就找到感觉了。

情况 6： 地铁等交通系统的处理

在考察地面交通的时候，只关心车行系统的时候，按照道路的情况建模就行，地铁的事儿可以甩开不要。但是要研究人的出行的时候，研究人的消费行为的时候，研究城市当中的功能分布的时候，就要考虑了。

越是大容量的、快速的交通系统，在宏观上，就越能够对人流物流产生拉伸作用。地铁这样的系统，因为是高度专用的，出入口都是专用的，所以在建模的时候，甚至可以把轴线模型都省了，直接用 Link 命令，将地铁口附近的路面连起来就行了。参见下图。

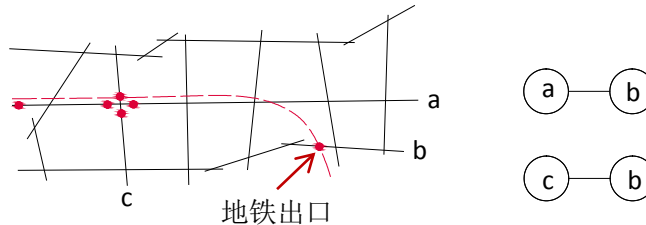


图 1.3.3-13 地铁的处理

上图的意思是，先在 CAD 当中勾出轴线模型，同时，用手工的方式，记下 a 与 b 是 link 上的，将来导入到 DepthMapX 当中以后，再用 link 命令给他们连上³。

研究人流的情况的时候，只要有地铁口的，就可以认为两个街道是拓扑上直接相连的，可以用 Link 命令，将其手工连接到一起。如上图中，手工将 b 和 c，link 起来，两者之间的关系就成了直接相连了。

根据敝人的粗浅研究，同样一个轴线模型，有地铁和没地铁，将来轴线图的计算结果，差别不是一般的大。

第四节 局部模型的建立

在基础性的研究项目中，一定要竭尽人力物力财力，去进行最大范围的资料收集与整理。追求的是研究成果的重大意义。

在一般性的商业项目中，有总的成本控制，有利润的保留与分配的问题，不可能将研究范围扩展到难以收拾的程度。必须在周密分析的基础上，将研究范围划定在一个尽可能小的区域内。

如何划分研究范围，王浩峰老师的教案当中提过，我这里就直接引用过来了：

1. 以环城高速作为边界；
2. 在目标区域外围，设置一个足够大的缓冲区。

环城高速、护城河、铁路线等，对于城市往往产生生硬的割裂作用，从卫星图上看，建成区在这附近，往往有生硬的边界。

³ 现在软件功能还不完善。软件无法自动处理，读者要手工操作，一定要有严谨的态度，要有耐心。谁和谁是直接 link 的，可以弄个 Excel 表格，先记好。将来导入到 DepthMapX 当中以后，还要花时间，去一个一个核对。轴线模型的正确性，是说话的基础。必须多次校对。

古代的城市，其城墙本身，就是个硬的边界。现代的城市，我们就从卫星图上去直接观察，都能够找到一些硬的边界。在一些硬的边界范围内，城市空间容易坍缩成一些区域性的次中心。这些事儿，科斯托夫不是写了两个书么，一个是《城市的形成》，一个是《城市的组合》，读者可以看下这两本书，对于城市的边界的情况，从历史的角度，进行一定的思考。

通过直观的感受，去找那些硬的边界，从而确定研究范围，进行轴线建模和分析，是有合理性的。

上述第二条，所谓缓冲区，就是考虑，在一个拓扑图中，带 GPS 坐标的，居于中心附近的那些元素，在拓扑关系上是占优势的。靠近地理上的边缘位置的，在拓扑关系上会吃亏。

如果设立了一定的缓冲区，使得我们盯着的这几条街道，都处于很靠近中心的位置，那么就能抵消掉 GPS 坐标的那种影响，将来以目标区域这些元素分别作空间重映射，会有数学上的合理性。否则，一条街比另一条街的整合度高，你还没开始解释呢，人家就提醒你，不用算都知道靠边儿的吃亏。设置缓冲区以后，两条街道都占有拓扑结构中心附近的位置，一条街比另一条街整合度高，你就可以说，这种情况，是如何如何地有学术上的意义和价值。

至于多大的缓冲区就算足够大了，还是要专门考虑一下的。

这里扯一下“等速系统”和“差速系统”的事儿。

在古代的城市中，所有的交通流量，都是非常低速的，可以称之为等速系统。

而现代城市中，交通空间本身，是出现了分化的。城市的交通系统，往往是按照速度等级，进行分化发展的。我称之为差速系统。

不同的速度等级的子系统，互相之间的衔接转换，是个大问题。因为不同速度等级的子系统，要衔接的时候，往往涉及到交通流量的重新组织、涉及到人流和物流的中转、等待的问题，是否能够流畅，关系就太大了。例如，深圳的南海大道，包括南山大道，在南山中心区附近，堵车的情况比较严重。从我们的研究来看，这个地方当初设计的时候，是按照等速系统的思路，进行的设计，路网间隔啥的，都比较均匀。但是这个南海大道与深南大道的交界处，同时又很明确，是深圳整体的交通系统当中，高速系统和低速系统转换的一个地方，这个地方有差速系统的配置问题。不研究还真不知道是这样。

差速系统的转换节点，是个大问题，在学术上具有突出的重要性。

差速系统的转换节点，有自己的内在的逻辑，有的时候通过直观的观察和感受，也能知道，但是更可靠的办法，是通过空间句法的一套计算与分析，去发现转换节点。分析过后，我们也能知道，将来的工程当中，可以有什么样的解决问题的办法。

除了节点的事儿，还可以扯点儿别的。根据敝人所研究过的几个古城，可以发现，等速系统的话，城市的高整合度的街道（生活性的街道）与高选择度的街道（通过性的街道），往往是重合的。只有在现代的城市中，在差速系统高度发达的情况下，才出现高整合度的街道与高选择度的街道相分离的现象。这也是预料之中的事儿，因为自从汽车的时代以来，“town without traffic”和“traffic without town”，就一直是城市规划当中的重大话题。

在二十世纪早期，汽车刚出来的时候，现代城市规划的思想刚出现的时候，人们的重点，在于通过手术刀一般的精准切割，将古城劈开，生生地将一些高选择度的街道拓展成宽阔的大道，从而极大地提升城市交通的容量。

现代城市规划刚出来的时候，重点在于开辟高速系统的大网格。

后来，高速的车行系统大行其道，甚至所有的街道都必须按照车行系统的要求进行设置，这样，手术刀横竖撇捺到处乱割，将城市生活割得支离破碎，本来城里边儿呆着，什么东西都挤在一起，人们干个什么事儿，走个两条街就到了，方便得很。现在好了，城市劈开八瓣

儿，功能都分散了，互相之间离着大老远，人们不得不抓紧时间，从一个地方赶去另一个地方。城市越大，人们就越疲于奔命。西方社会在 1970 年代，有了石油危机以后，就觉得，太过僵硬的城市发展方式，不合适。差速系统有高速的子系统，有低速的子系统，一味地按照高速的子系统的要求，去进行城市建设，不合适。为了人的舒服，为了城市功能分布的合理与方便，要尊重城市当中的低速系统的要求，要尊重人的步行尺度上，这个最低速的交通系统的要求。被差速系统所割裂的城市的生活空间，被割裂的步行尺度的空间，需要进行缝合。

一个研究者，如果不了解一个城市的差速系统的发展的历程，就不可能真正理解这座城市的性格与气质。

回到本节的主题，回到局部模型的问题。在设立缓冲区的时候，最为稳妥的，是先去实地调研一下，看看在研究的这个目标范围内，城市当中的差速系统是如何配置的？罗马不是一天建成的，差速系统也不是一天就建成的。近代以来，城市当中差速系统的发展，基本上是每一版总体规划的核心内容，可以将历史上的总体规划都拿来，先梳理一下，这个差速系统的来龙去脉是怎样的，有用的信息可能也不少。在此基础上，再根据现在的主观判断，去划定合适的缓冲区的范围。

最后，再罗嗦一下，如果研究目标的范围内，是有地铁的，存在类似这样的大容量的交通系统，那么缓冲区就要弄得大一些。地铁这样的交通系统，对于人流的拉伸作用，是巨大的。如果城市当中仅有一两条地铁线，那就扩张一下尺度就算了。如果城市当中的地铁系统已经比较完善了，地铁系统的网格已经形成了，那么干脆不设置缓冲区了，直接将研究范围放大，直接将城市整体的模型建出来，是合适的。甚至，在现代交通技术的基础上，城市的那种行政区划的范围，都显得太狭隘了，人的活动都是跨行政区的，要从区域的角度，从城市群的高度，去进行建模。

将来做城市设计，或者城市规划的时候，如果将差速系统的配置改了，那么我们也可以合理估计一下，这将会带来什么样的影响，波及的范围有多大。

有无地铁，在研究方法上，甚至发生范式转换了都。

本章小结

第四章 轴线模型的计算与分析

轴线模型建好以后，可以导入到 DepthMapX 当中来，先验证一下正确性。大部分情况下，轴线模型都会存在瑕疵，比如一两条轴线与其他所有的轴线都不相交，或者 unlink 的地方忘了 unlink，等等，总会有这些麻烦。

导入 DepthMapX 当中，利用其中的几个计算结果，可以先验证一下轴线的正确性。经过几轮的调整，轴线模型满意了，再进行后面的步骤。

第一节 验证轴线模型的正确性

在 CAD 当中，建立轴线模型的时候，可以专门设置一个图层，方便将来梳理 Unlink 的地方。

所有 Unlink 的地方，都手工加上一个圆圈，而所有圆圈都位于专门设置的这个图层上。将来轴线图绘制完了以后，将其他图层关闭，用鼠标在这个专门的图层上框选，就能知道一共几个物体，就能知道总共有几个 unlink 的地方了。这样在 CAD 当中和在 DepthMapX 当中，两边儿来回核对，方便一些。

Unlink 的点，到了 DepthMapX 当中，还是要手工重新设置一遍的。

如果要做 link 的，也是同样的思路。

下面弄个最简单的轴线模型，操作一下看看。下图是个假想的方格网城市，当中有街道画错了，红色的 a 轴线与所有其他的轴线都不相交，是个独立的轴线。而 b 轴线附近，有那么一坨，自己内部是有联系的，但是与其他地方断开的。这里是为了表示的方便，在 CAD 中弄成了红色，导入 DepthMapX 的时候，是不认颜色的。

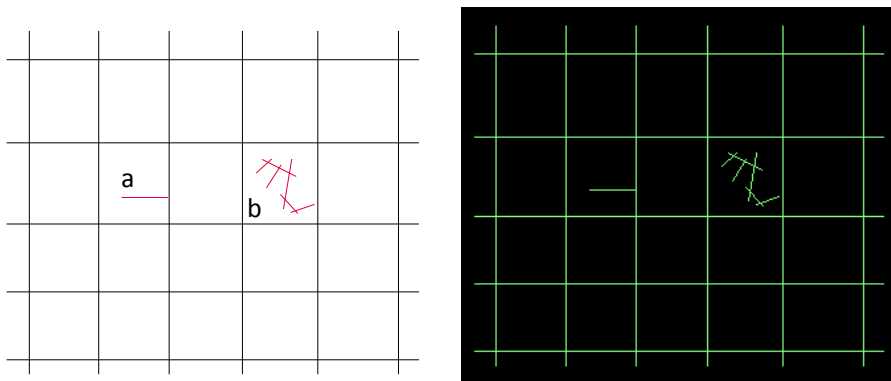


图 1.4.1-1 将轴线图导入 DepthMapX 中

城市当中的道路，都是能走得进去的。自我封闭起来的那种空间，对于我们的研究来说，是没有意义的。在轴线模型中，从任意一个轴线出发，到达任意一个轴线，都应该是走得通

的，不可能出现互相分割成不相连的子系统的情况。上图这样的情况，就是一个图中，有三个子系统。在我们的学术研究中，这种情况是不可能出现的。

所谓的验证轴线模型的正确性，也就是，要确保所有的轴线，宏观上是处于同一个系统当中。

轴线图的错误很常见。有的时候，较长的轴线稍微把端点挪挪位置，肚子那里的连接关系就丢失了。出错的地方可能会很多，靠着眼睛一点儿一点儿去寻找，也不现实。必须要利用软件来排除错误。

这里再啰嗦一下软件的操作步骤。从 CAD 当中，将轴线图勾出来，导出为 dxf 格式的文件。到了 DepthMapX 当中以后，Ctrl+N 创立一个新的工作空间，Ctrl+I 导入准备好的文件。

接下去，通过 DepthMapX 当中的下拉菜单，Map → Convert Drawing Map，弹出对话框。在对话框中选择 Axial Map，生成一个新的轴线图的图层。如下图所示。

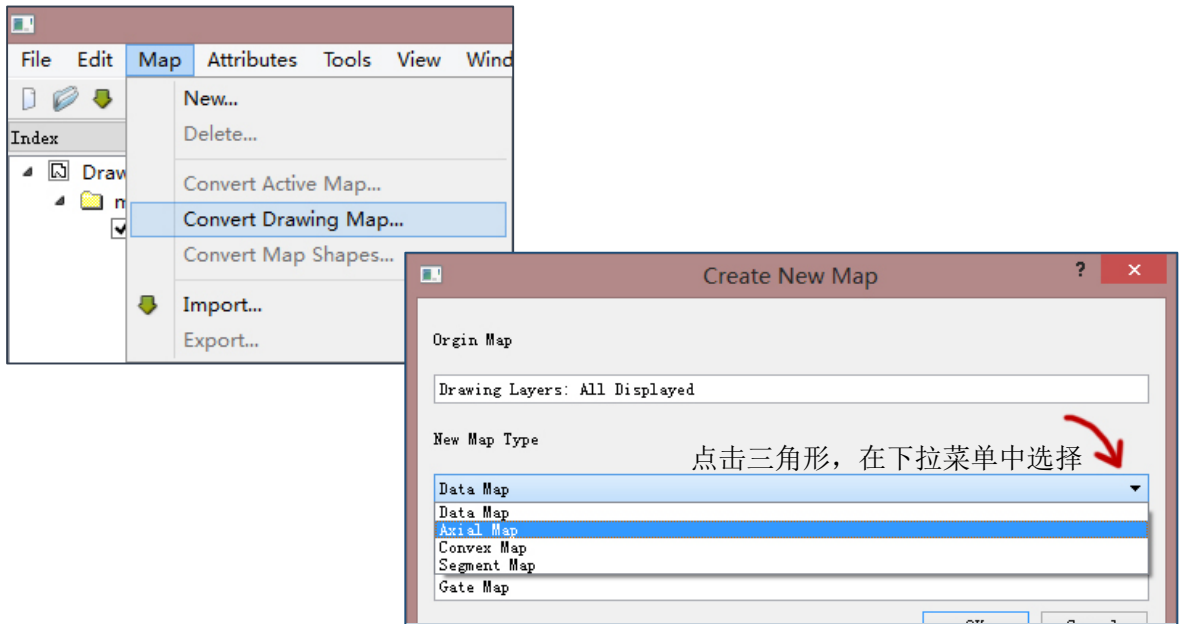
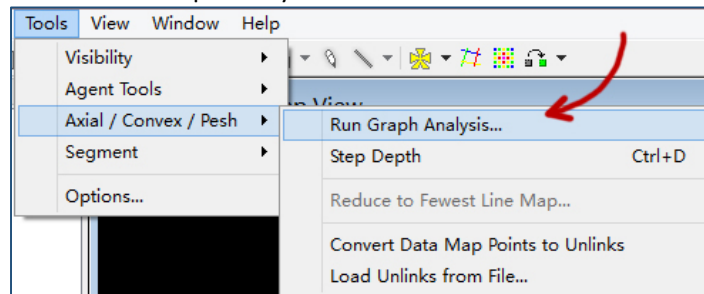


图 1.4.1-2 生成 Axial Map

这就将导入进来的东西认成轴线了。DepthMapX 窗口左下角的参数选择栏那里，出现了一些新的参数。暂时没什么用，先不管。我们接下去操作，从下拉菜单当中选择，Tools → Axial / Convex / Pesh → Run Graph Analysis，弹出对话框。如下图所示。



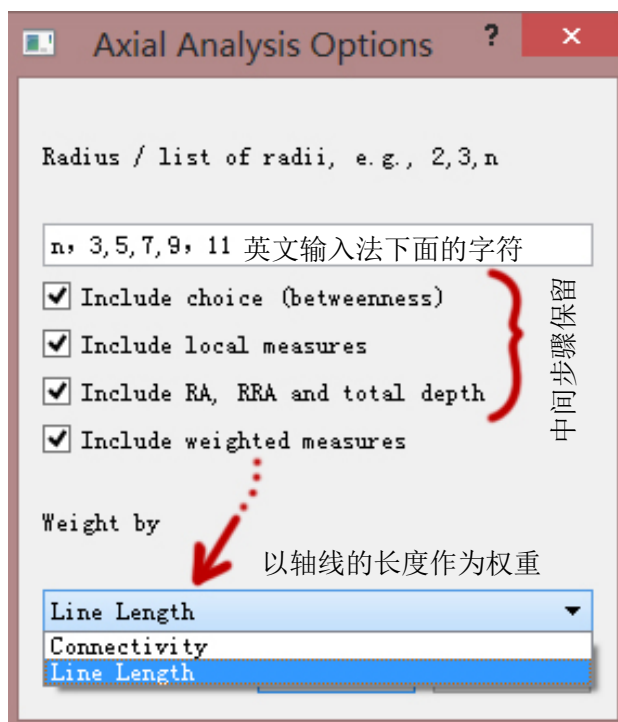


图 1.4.1-3 轴线图的计算

在所有情况下，都是这么设置。读者直接将这个对话框的情况记住就行了。最上面，输入半径，输入 n，然后三五七九，中间用逗号隔开，表示空间重映射以后，以这些拓扑步数为半径，截取局部的拓扑结构出来，进行计算。一共四个方框可以打勾，前三个打上勾，表示计算的中间过程保留，会在参数选择栏那里，出现一些东西，将来可以查看。最下面这个方框，打上勾，表示已什么什么为权重，进行计算结果的修正。所有的情况下，我们都选择用轴线长度进行权重。

对话框设置好以后，OK 一下。DepthMapX 左下角的参数选择栏，出现了一大堆新东西。这些就是轴线分析的结果。

因为轴线图本身可能存在这样那样的问题，请读者切记，不要直接用那些计算结果，先要通过检查 Node Count 这个参数，验证一下轴线模型的正确性。如下图所示。

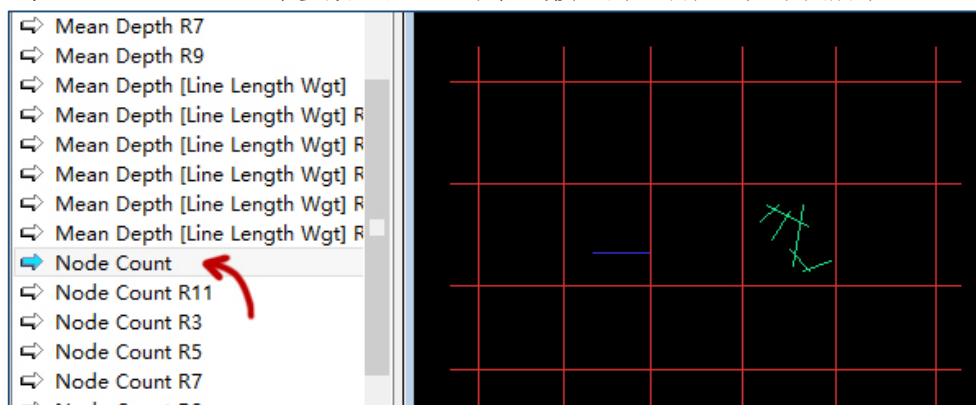


图 1.4.1-2 根据 Node Count 的计算结果，找到有问题的轴线

这个 Node Count 的意思，就是从某个元素走出去，一共能够走到几个其他的元素？因为现在一张图里面有三个子系统，所以各个子系统的头上，记着不同的数字。

我们一定要确保，所有的轴线都连在相同的一个空间系统上，所有的轴线都是走得通的。一旦出现了一片红色，中间夹杂着几个蓝色的线条，我们就知道模型有断开的地方。把这些细小的蓝色的线头找到，在 CAD 中，将轴线图修正好，将来再重新通过 dxf 格式的文件重新导入一次。

例如图中的这种情况，我就回到 CAD 当中，将断开的地方，与旁边的大路接上，再导入一次。有的读者可能会抠细节，路网到底是怎么个衔接，不先搞清楚，随便那么一延伸，怎么能行呢？确实常会遇到这样的问题。有的时候从卫星地图上看不清楚，有的时候官方提供的 CAD 的底图也很成问题，丢三落四，图层混乱。实在搞不清楚的情况下，有些尽头的小路，干脆删了得了。因为城市的尺度上，轴线可能达到几万条，一些小巷子，就是把轴线丢了，对于全局的影响也是微乎其微。

下图这样，断开的地方，把端点都扯过来，再导入到 DepthMapX 当中去，按照前面讲的步骤，先 convert 成 Axial Map，然后，Run Graph Analysis，从得到的一堆结果当中，选中 Node Count，看到一片绿色了，这就放心了。

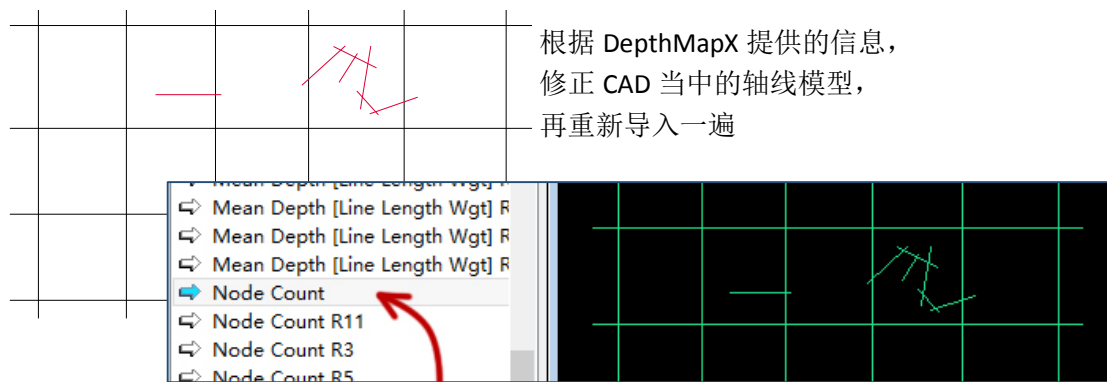


图 1.4.1-2 轴线图的计算

重说一遍，一定要确保 Node Count 是一片绿色的情况下，才能放心没有单独存在的那些轴线。

一定要确保 Node Count 是一片绿色的情况下，才能进行下面的工作。

一定要事先检查 Node Count !

第二节 在 DepthMapX 中处理连接关系

本节所述内容，前边儿已经涉及到了。本节主要是弄个实际的例子，一步一步操作一下，帮助读者熟悉这个工具的用法。

大连这个城市挺有意思的，有很多故事，更有利的是，现存的城市测绘资料也很多。日本鬼子早年的城市测绘图，水平一流。我们拿明治三十九年，公元 1906 年的大连城市地图来，转译成轴线模型分析一下。参见下图。

老地图一般是以 jpg 的格式保存的，在导入到 CAD 的时候，要把尺寸弄准确。尽管拓扑学上，是不考虑实际坐标的，轴线模型中，不考虑坐标的问题，不考虑尺度的问题，但是后面线段模型那里，要用到。制图的时候，要把尺度弄对。

城市地图，出版的时候，都带比例尺的，根据底图可以有个参考的尺度，这是一个办法。另外，因为城市当中，很多街道，很多街块，到了现在还在，可以从 Google Earth 上面，找那种特征明确的点，利用 Google Earth 的测距的功能，测出尺寸。这样也有个参考的尺度。

确定基本尺度的时候，要多种方式都试一试，因为底图的误差一般都比较大。底图的绘制方式，采用的不同的坐标系，就可能造成误差，包括后来扫描的时候，镜头上带个偏差，啥的，也可能造成扭曲变形。根据敝人的经验，各方汇集的资料，无法完全对得上。尺度的事儿，最牢靠的是在 ArcGIS 当中进行配准。

在 CAD 当中，通过缩放命令，尽量能够将尺度对上，就很好了。



图 1.4.2-1 1906 年的大连

上图中，红色箭头所指的部分，是规划路网。这样，我们建模的时候，可以先建一个纯粹现状的，不带规划路网，计算一遍。然后将规划路网加上，再计算一遍。在此基础上，两个轴线图再做比较研究，就方便了。CAD 中，两种状态的轴线图，如下图所示。

这种比较研究，最大的用处，就是评价城市设计。不同的方案，各有什么特点，可以通过轴线图来比较一下。

图片导入 CAD 当中以后， 图片有个边框。左键点击图片边框，图片就处于选中的状态了，而 CAD 窗口的顶部，会蹦出来名为 Image 的一个工具条，通过调整 Fade 的值，默认为零，调到五六十，图片整体就变暗了，将来在上面描轴线，视觉效果上更加舒服一些。参见下图。

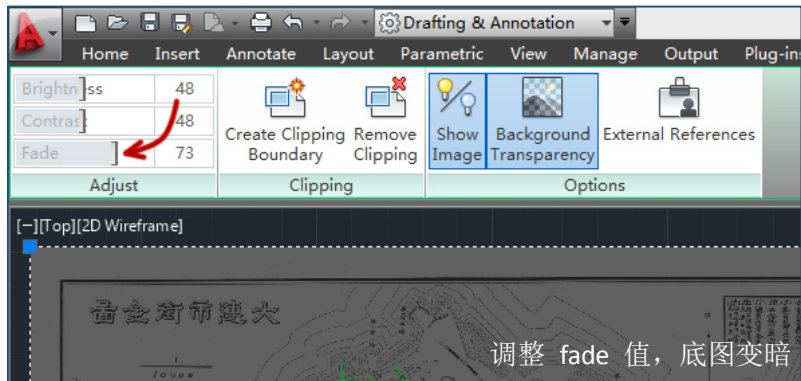


图 1.4.2-2 调整 CAD 中的 Fade 的值

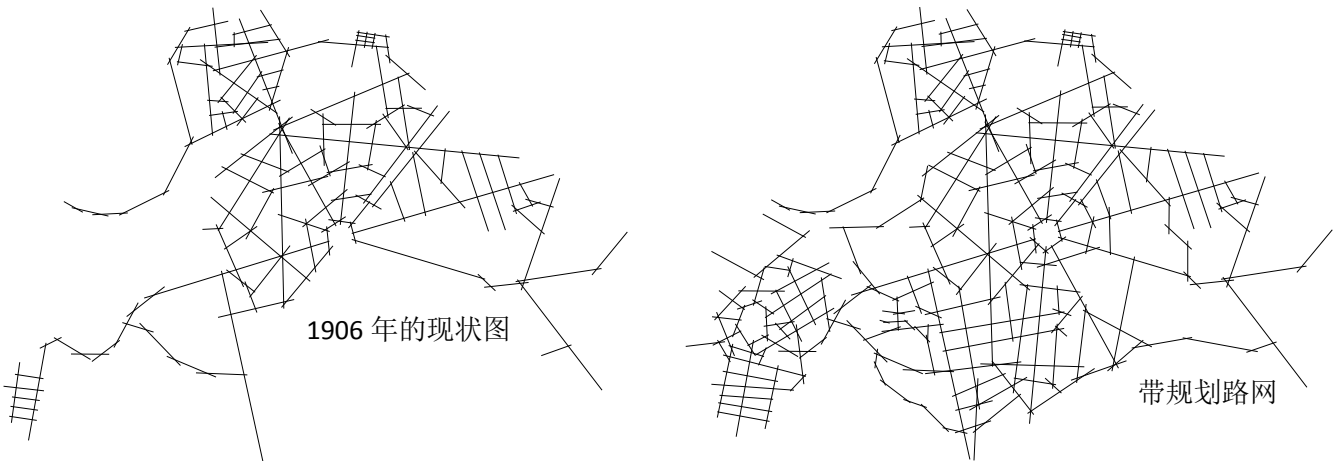


图 1.4.2-3 按不同的规则，分别建立 Axial Map

在 CAD 当中，分别建好模型以后，分别导入到 DepthMapX 中，进行轴线分析。

软件操作的步骤是：在 DepthMapX 中，Ctrl + N 创立一个新的工作空间， Ctrl + I 导入 dxf 格式的文件。导入进来后，首先要 convert 成 Axial Map ，然后 Run Graph Analysis，得到一堆计算结果。

在计算结果中，首先查看 Node Count，如果是一片绿色，就可以继续进行下面的分析工作。如果是一片红色，说明有断开的地方，要把隔离开的那些小系统找到，回到 CAD 中进行修改，将丢失的连接关系补上。检查完了以后，再重新导入一次。

在严格地进行了检查步骤以后，轴线的计算结果就可以拿来用了。

在 DepthMapX 中，两个 graph 文件的窗口都打开，将窗口弄到一样大小，并列着。再用 Recenter View 命令，将轴线图居中显示，这样，再从程序窗口左下角的参数选择栏中，分别点击两者的 Integration [HH]，两个轴线图的整合度情况，比较起来就方便了。



通过 Recentre View 命令，
轴线图最大化居中显示

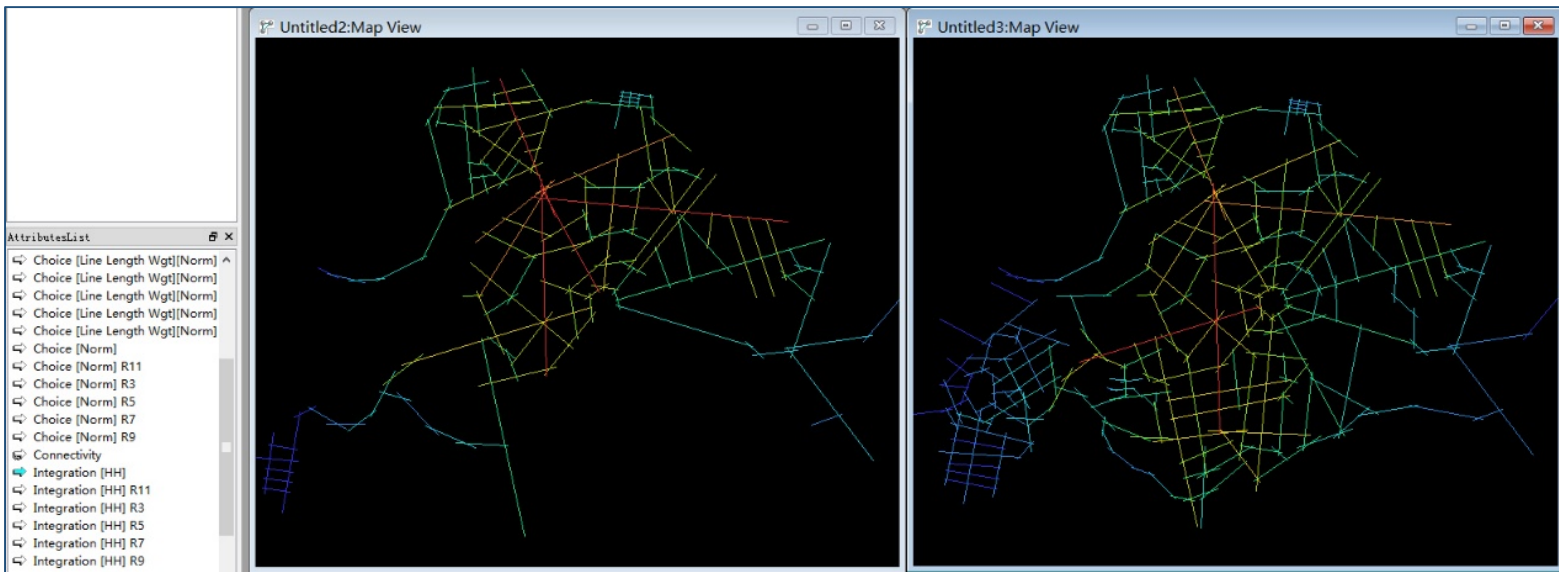


图 1.4.2-4 在 DepthMapX 当中，窗口并列，方便比较

在这样的工作方法的基础上，进行比较研究，比较直观、方便。上图中，显示的都是基于 Axial Map 的 Integration [HH] 的情况。

第三节 Integration Core

轴线模型计算完成以后，会自动跳到 Integration [HH] 上面。直接在软件当中，进行观察比较，将来还面临输出的问题。直接通过截屏输出，也可以，或者利用 DepthMapX 的输出 eps 的功能，输出计算结果，也可以。

具体功能在 Edit 菜单下面，如下图所示。

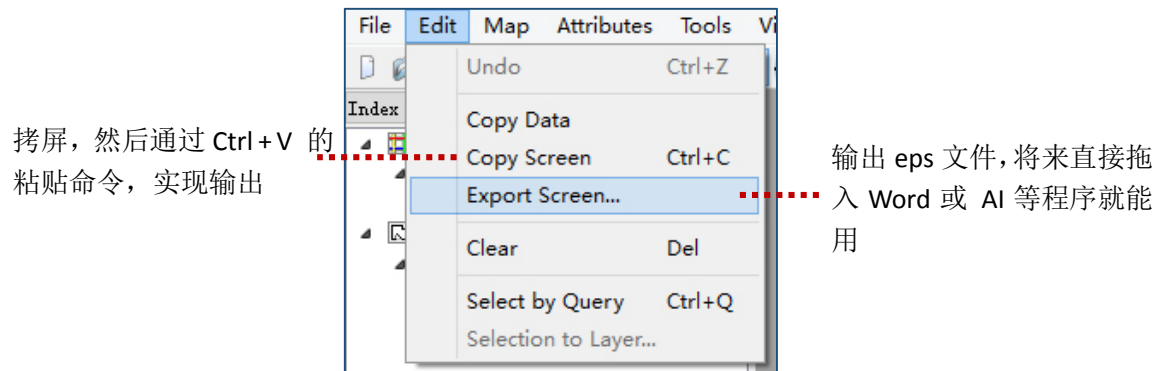


图 1.4.3-1 输出计算结果

观察了大连的 1906 年的两个轴线图的情况以后，本节重点是“整合度核心”的事儿。这是 Bill Hillier 的那个黑皮子书中，也提过的概念。

描述整合度核心的概念的时候，用的英文是“the 10% most integrated lines”。意思是说，在一个空间结构中，总有少数的那么一些互相勾搭在一起的轴线，整合度很高。不论是在古城，还是在规模巨大的现代城市，都有这个现象。可以用数学再深入定义一下。

整合度的数学定义是这样的：把所有轴线的整合度的数值，都加在一起，得到一个加总的值， I ，然后将所有元素按照整合度数值的高低，进行降序排列，从最高的元素开始，依次向下选择元素，将其整合度的数值相加，直到加起来的这个值达到了 I 的十分之一，这些参与运算的最高整合度值的元素，这个集合，就被称为“整合度核心”。

注意，如果是 200 条轴线以下的小系统，就是 the 10% most integrated lines，如果是 200 条轴线以上的大系统，就是 the 5% most integrated lines.

一个系统中到底有多少元素，可以从 DepthMapX 窗口的右下角查看。参见下图。

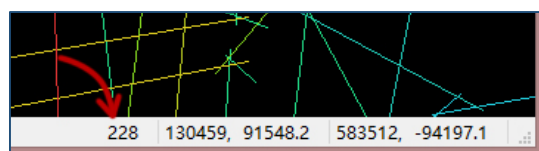


图 1.4.3-2 查看元素数目

整合度的建筑学意义，在前面介绍凸空间的时候，就已经涉及到了。整合度高的，就是可达性高。

那么，所谓的整合度核心，就是可达性最高的那些街道的集合。因为可达性高，这些街道空间往往也是面子工程的集中地，在经济、文化和政治上具有重要性，在这些街道上聚集着不少重要的城市功能。

有了整合度核心的概念以后，我们来研究一下孟母三迁的故事。他们搬家，是从远离城市的墓地，搬到了城市边缘的杀猪宰羊的功能性服务区，最后搬到了市中心，整合度核心附近的文庙。这是城市化过程中的典型情况。

我没明白，从郊区往城里搬，别人的妈也是这么干的，凭啥孟子他妈就占理了呢？

想想也是，孟子是当老师的，一般当老师大的人，都是嗓门大，音调高。孟子同学恐怕从小嗓门就大，住在坟地边上，没事儿帮着嚎两句，感情比人家真的死了亲爹的还要真挚。孟子他妈一听，那心情，简直了。

不搬家真的受不了了。

所以，我们中国人讲的故事，尤其是圣人的故事，听听就算了，好事儿都是安上去的，实际情况如何，恐怕远没有那么浪漫。

言归正传，城市中，整合度核心附近，往往经济繁荣，人气很旺。城市功能的发展，会要求整合度核心附近的城市空间结构，进行最经常、最大量、最细致的优化。随着时间的推移，城市空间扩张，整合度核心有可能从最初的有限的几条街道，慢慢扩张开来，也有可能，整合度核心慢慢挪到别的地方去了，敝人称之为整合度核心的漂移。

把握住了整合度核心的扩张和漂移的历史过程，往往也就抓住了城市发展的主线。



老的大连，是俄国人做的规划。后来日本人占据的时候，在原有规划的基础上，又一边做实施，一边提出新版的规划。从城市地图上看，一直是将现状和规划路网都详尽地表现出来的。

年代迟一些的地图，我们拿过来看，直观地可以感受到，日本鬼子还是忠实于城市规划的要求，慢慢地将当初图上的东西实施了。如上图所示。

这些图在百度上都搜得到，感兴趣的读者可以拿来练练手。

本节的最后，再扯一个 Bill Hillier 在那个黑皮子书中提到的概念，叫做“空间结构的惯性” (Hillier, 2008, p. 35)。意思是说，长期以来，财富与建筑物是不能分离的，建筑物是物质财富的唯一载体，人们一定会寻求任何可能的手段，去保护自己的财富。这就造成，城市空间结构，一旦生成，就具有了强大的维持自身稳定的能力。这是一个方面。

另一个方面，建筑物上的投资，虽然可能带来巨大的财富，但是同时，也可能蕴含着重大的风险。对于普通的老百姓来说，房子再烂，只要能糊弄过去，就不愿意动。在传统的中国，什么样的人家才大兴土木呢？都是要娶媳妇的家庭，才投资于建筑。平时，几乎就不会去弄这个事儿。

所以，城市空间结构的改变，城市当中建筑物这类的物质实体的更迭，强烈地受到历史因素的影响。我们把城市的街网，抽象成拓扑学的结构，不同的历史时期，分别建模，然后沿着时间轴，将这些空间结构排列出来，做一对比，往往会发现，许多东西具有顽强的生命力，几百年过去，根本就没有发生本质性的变化。

人们常提到什么不朽的丰碑，其实丰碑很容易朽，反倒是丰碑所处的那个地方，与城市其他空间的拓扑关系，这个关系，很难朽。物质实体与抽象出来的这个拓扑学关系相比，还是拓扑关系更为本质。

读者中如果有人，了解犹太人那些事儿的话，我就提醒你，耶路撒冷不是有一个犹太人的教堂被毁，后来在原址上盖了个清真寺么，这个事儿就是最为根本的例证，证明了空间结构的惯性，是多么的根深蒂固。

如果 Bill Hillier 说的这个惯性的事儿，能够获得认同的话（如果他说话的逻辑您能够认可），那么，站在现在这个时间点上，我们就可以想想将来的事儿了。

正因为空间结构的惯性，涉及到最为本质的问题，所以，牛逼的城市规划师，和牛逼的建筑师，最根本的工作，不是瞄准实的，而是瞄准虚的。真正牛逼的人，都是在法律的框架内，在尊重人民群众的财产权的情况下，还能有办法，去改善城市空间结构的人。

如果通过暴力手段，将城市空间结构铲平，那么一切与城市本体有关的历史记忆、文化积淀，就都会消失。

城市空间结构都颠覆了，妄谈历史传承啊。就像一个广东人抱怨什么事儿，用的语言竟然是“这是咋整的？！”，就全完了。城市空间结构的那种独特性，和语言文化的那种独特性，是一样的，都是无价的。

没有历史传承的人，是孤独的；没有历史传承的城市，是苍白的。保护城市的空间结构，是重要的研究视角。

第四节 Synergy

还是 1906 年的大连的例子。现在我们已经做了轴线模型的计算了，除了直观地去看轴线图，还可以看散点图，所谓的 Scatter Plot。

散点图一般是带颜色的。散点图这个窗口打开之前，在参数选择栏当中，当前的选中状态是哪个，哪个就被默认为 Scatter Plot 的横轴和纵轴的显示对象。将来通过更改横轴和纵轴的定义，来查看不同的参数之间的关系。

打开散点图，是从 DepthMapX 窗口顶部，下拉菜单那里，点击 Window → Scatter Plot。



图 1.4.4-1 查看 Scatter Plot

不论最初生成散点图的时候，X 轴和 Y 轴是什么，都可以再变，如上图箭头所指，通过下拉菜单选取就是了。我们把 X 轴定义为 Integration[HH]，Y 轴定义为 Integration R3。这样，下面显示的散点图，当中的点，就有意义了。每个点，代表的都是一个元素。针对某个特定的点，这个点的横坐标的来源，就是 Integration[HH]，纵坐标的来源，就是 Integration R3。点在这个坐标系的位置，就综合地反映出了该元素头上记着的两个参数的值。

横轴和纵轴代表什么意思，可以手工选择。而现在散点图上的某个具体的点，显示什么颜色，还是由参数选择栏的当前状态决定的。也就是一张散点图，可以表达三个方面的信息。如下页图所示。

如果觉得散点图表达的信息太多了，可以把颜色关掉。正式发表文章的时候，面临费用问题，彩色印刷可能无法保证。弄成黑白的散点图，可能会更加符合规范。

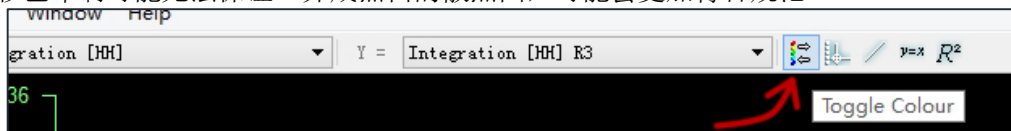


图 1.4.4-2 彩色开关

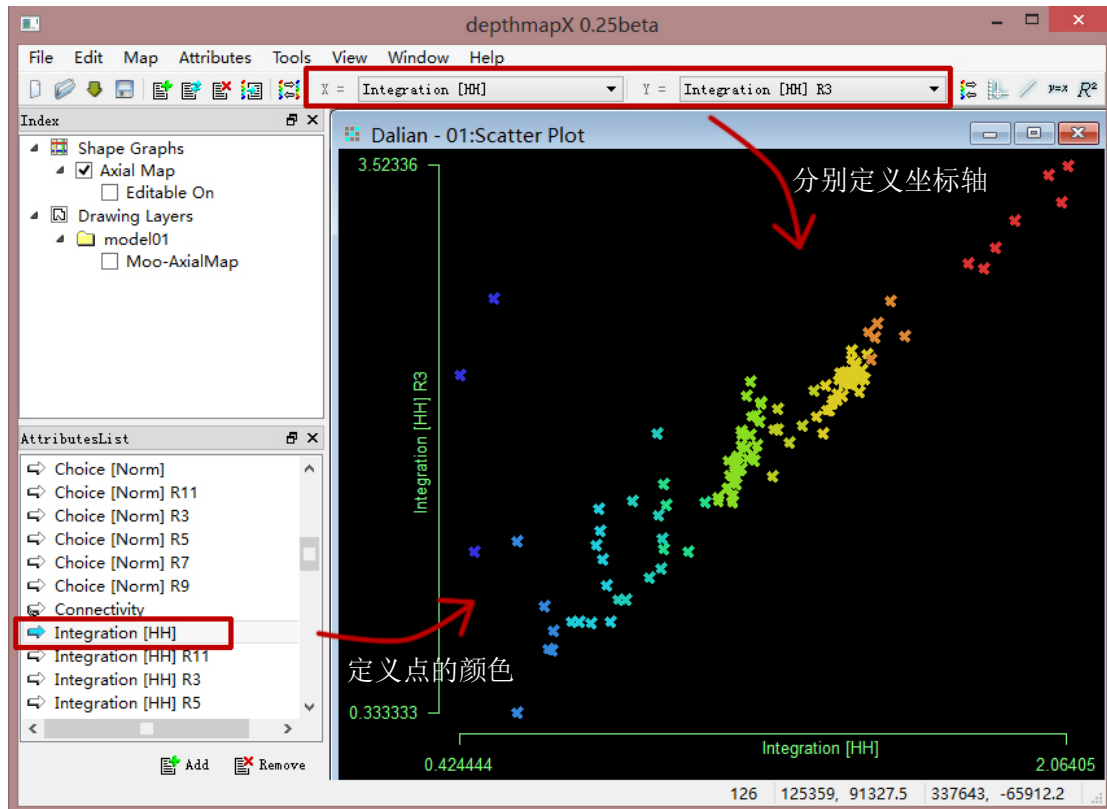


图 1.4.4-3 散点图的意义

彩色开关后面，还有几个开关，读者可以手贱一点儿，分别点击点击，尝试尝试。这些开关的意思，如下图所示。

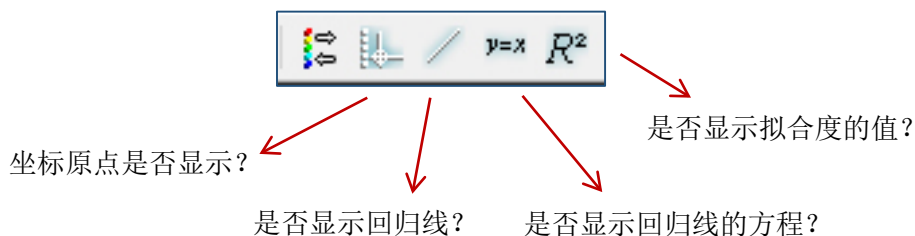


图 1.4.4-4 散点图的开关

所谓回归线，英文是 Trend Line，就是求一条直线，所有的点到该直线的距离，加总起来，是个最小值。

$y=x$ 那个按钮，意思是，用 $y = ax + b$ 的形式，来表示这条 trend line。

R 平方这个按钮，是数学上的拟合度，这个值越高的，表示用 trend line 去预测散点图的实际情况，就越准确。拟合度的值越高的，表示 trend line 这种简化的走势表示方法，就越靠谱。

R 平方的值，在 0.5 以下的，认为横轴与纵轴不相关。R 平方的值，在 0.5 以上的，认为横轴与纵轴是相关的。当 R 平方的值上升到 0.7 以上的时候，认为横轴与纵轴之间显著相关。

这些开关，如果打开，相关的信息就能够随着散点图一起输出出去了。哪些信息有用，就输出哪些信息。没用的就关掉好了。

本节的主题是协同度，就是看全局整合度和局部整合度之间的相关关系。

请读者先回忆一下，Integration[HH]（下文中，为了说话方便，简称 Rn）是什么意思，Integration R3（简称 R3）是什么意思。

前者，就是在空间重映射以后，将全系统都考虑进来，算一下整合度的值。后者，就是空间重映射以后，以 3 个拓扑步数为限，截取中心空间附近的这么个小范围出来，计算这个新的拓扑结构的整合度。也就是说，前者是全局整合度，后者是局部整合度。现在将全局整合度和局部整合度进行相关性分析，就有了建筑学上的一些意义了。

其实，这个协同度，既可以是 Rn 与 R3 的协同，也可以是 Rn 与 R5 的协同，又可以是别的组合方式。只是 Rn 与 R3 的这个情况，最典型，一说协同度，就是默认是这两个了。

关于这个协同度，在中文版的《空间是机器》的 103 页，有所涉及。不过翻译很成问题，图片的说明也没有，看了跟没看似的。

对这个协同度，敝人的理解也不深，下面的解释，都是我自己的想法，供读者参考吧。

首先，算法是明确的，不论是 Rn 还是 R3，从拓扑学上，算法是很明确的。从较为功利的角度来说，整合度不是衡量可达性么，那么不论是全局的还是局部的，可达性高的，还是好啊。

理想情况是，针对某个元素，它的 Rn 和 R3 的值都较高。可是实际上，全系统中，Rn 和 R3 的值同时高的街道，就没几条。绝大部分街道，是 Rn 也不怎么高，R3 也不怎么高。除此之外，往往有少数的那么些街道，其 Rn 和 R3 的值，有一个畸高，另一个畸低。这都是城市当中有特色的区域，或者有问题的区域。

更加极端的一些街道，其 Rn 的值很低，同时 R3 的值也很低，这样的区域，从经济地理学的角度来看，肯定是非常不利的区域。

根据敝人自己研究过的几个案例，包括我们深圳大学建筑研究所做过的一些研究，深圳有几个城中村，城中村范围内的街道，可能全局整合度不高，但局部整合度非常高，这表示，这个地方是封闭起来，自己发展自己的，外界对它的影响不大。

在阿拉伯国家的案例中，城市中心地带，有可能存在一些街道，其全局整合度和局部整合度都很低。按理说，越是靠近地理上的中心位置的，那些地方越有繁荣的条件，经过历史的积淀，Rn 和 R3 都应该较高才是。如果在城市靠近地理上的中心的地方，出现了 Rn 和 R3 都很低的街道，这就说明这座城市不一般，其社会发展和经济发展的逻辑，与我们所认为的常识，根本不同。

敝人认为，不一定非要衡量整合度这个算法下面的 Rn 与 R3 的协同度，我们可以用 Rn 和容积率，进行相关性分析，也可以用轴线模型的任意参数，推送到 ArcGIS 当中，去和任意的社会调查的结果进行相关性分析。

只要一项技术经济指标，或者社会调查，它的数据是带有地址的，它的调查是与地址相关的，就可以与空间句法的数学模型进行相关性分析。数值可以在不同的模型之间来回推送。每个城市都是有特色的，总会有一些具体的空间，在相关性分析中，显示出自己的个性。我们首先要发现那些有特色的空间，进而去探究那些空间之所以存在的原因。

在功能主义的大背景下，协同度的分析，就是要找到那些协同得不好的地方，将其空间结构进行一番大刀阔斧的改造，使其数学上的指标往我们认为的好的方向发展。这就是我们的赤果果的目的。

第五节 选择度

2012 年以来, 这个 Choice 的事儿, 有了新的进展, Bill Hillier 等人发表了一篇新的论文, 对选择度的相关算法, 进行了改进。该文章的题目, 叫做“Normalizing least angle choice in DepthMap” (Hillier, Yang, & Turner, 2012)。这个文章的附录 2 当中, 有关于 Choice 的算法的数学上的讨论, 包括经验系数是怎么给的, 都谈到了。有兴趣的读者可以深入看下。

在最早的时候, 选择度的问题, 是个纯粹的拓扑学的问题。

假设有两个空间结构, 如下图所示。因为拓扑连接方式的不同, 造成中心空间的选择度不同。

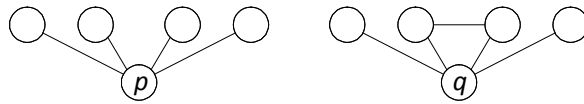


图 1.4.5-1 选择度的拓扑学定义

空间句法的概念, 很多都是“全系统加总”的。所谓选择度, 就是考察一个空间“出现在最短拓扑路径上的次数”。对于凸空间 p 来说, 把它自己排除在外, 以全系统中任意一个元素为起点, 任意一个元素为终点, 最短的拓扑路径是怎样的? p 是否出现在这个最短路径上? 如果是, p 的选择度的值, $\text{Choice}(p)$ 就加 1。穷尽所有可能的情况以后, 就得到了元素 p 的选择度的值。全系统当中任意一个元素, 都计算了选择度以后, 每个元素头上都有 Choice 的值了, 就可以按照空间句法的赋予颜色的规则, 去形成一个彩色的选择度的彩色特征图了。

上面的情况, 可以手算一下。然后用软件再验算一下, 看看两者是否一致。

在 CAD 中, 用方框表示元素, 导入 DepthMapX 以后, 手工设置元素之间的 link 关系, 本教程前面的章节中有过操作方法的介绍, 读者还熟悉么? 如下图所示。

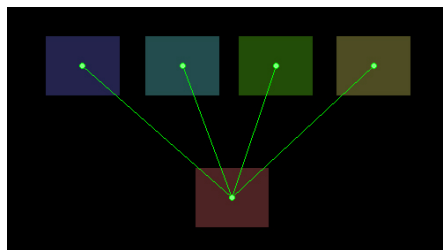


图 1.4.5-2 选择度的拓扑学定义

通过算法的设计, 我们就知道, 选择度可以衡量, 一个元素吸引到的穿越交通的潜力。

尽管, 实际发生的城市活动, 其路径选择可能不会是最短的拓扑路径, 但至少, 选择最短拓扑路径的可能性是很大的。从任意一个元素出发, 到到达任意一个元素, 我们是穷尽了所有的可能性的, 这就意味着, 城市当中的活动, 其出发点和终点, 都是被这个数学模型所覆盖了的。选择度的算法, 并不是说一定会如何如何, 而是说, 潜力比较大, 可能性比较高。

既然是考虑可能性, 那么, 就不只是拓扑关系, 对这个“可能性”发生影响, 其他的因素也有可能的。在简单的 Choice 算法的基础上, 可以进行修正。

根据前述文章的附录 2(Hillier et al., 2012, p. 191), 最简单的修正办法, 是除以另一个我们已经提到过的拓扑学的变量, Total Depth。这个 Total Depth, 也是纯粹的拓扑学的算法得来的, 本教程在第一章第四节中, “空间深度的考察”那一节, 有针对 Total Depth 的详细的说明。修正过的选择度, 英文是 “Normalized Choice”, 公式如下:

$$\text{NormalizedChoice}(p) = \frac{\text{Choice}(p)}{\text{Total Depth}(p)} \quad (\text{公式 1})$$

上述公式, 意思是说, Choice 值, 与系统当中的元素的数目, 有很大的关系。而 Total Depth 也是这样。元素数目越多的系统, 其中的某个元素, 其 Choice 的值就可能越高, 同时 Total Depth 的值就可能越高。这样两者相除, 可以将元素数目的变动情况, 给剔除掉。参见下图。

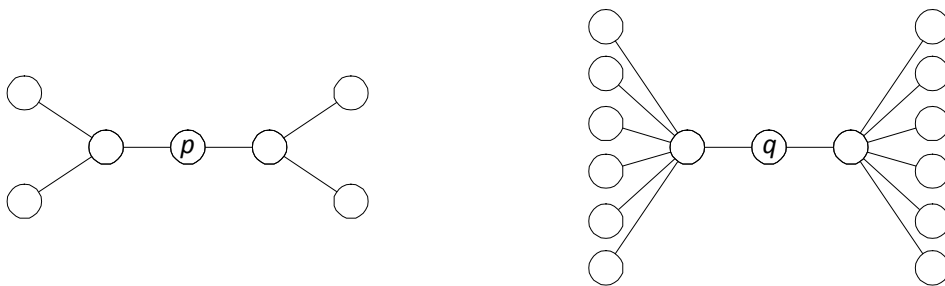


图 1.4.5-2 选择度的值被元素数目所影响

选择度是衡量一个元素吸引穿越交通的潜力的, 而全局深度是衡量一个元素吸引到达交通的潜力的。两者都考虑进来, 就综合性地衡量了一个元素吸引 “交通” 的潜力。文献中的原话是, “this can be interpreted as the spatial benefit-cost ratio” (Hillier et al., 2012, p. 191 公式 1 的解释部分)。读者可能会有疑问, 为啥上式中的两个东西, 不用乘法, 而用除法。从我们的分析中, choice 的值, 越高越有利, 而 Total Depth 的值, 越低越有利。同样是变得 “有利”, 两个东西的变化趋势却不一致。取个 Total Depth 的倒数, 变化趋向就一致了, NormalizedChoice 的值越高的, 综合起来越有利。

在公式 1 的基础上, 还可以加入限制条件, 以多少步为半径, 将分子分母分别限制一下。公式摆在下面:

$$\text{NormalizedChoice}(p) \text{ within step}(r) = \frac{\text{Choice}(p) \text{ within}(r)}{\text{Total Depth}(p) \text{ within}(r)} \quad (\text{公式 2})$$

在 DepthMapX 当中, 在窗口左下角的参数选择栏, 点击 Choice[Norm] 这个参数, 就是修正过的 Choice 的情况了。

本节涉及到的那篇论文, 还提到其他的公式, 那都是线段模型里边儿的东西。下一章的内容才能涉及到。

大连那个例子, 选择度的计算结果, 如下图所示。

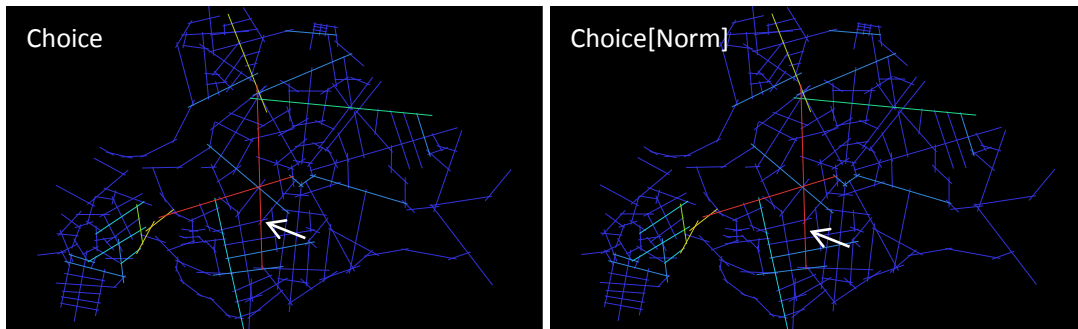


图 1.4.5-3 选择度

上图中，白色箭头所指的那条轴线，对应于现在大连市中山区友好路。左边的这种情况，其 Choice 的值为 20954，右边儿这种情况，Choice[Norm]的值为 0.81688888。分别都是最高值。两个图像显示出来的色彩关系，也是极其相似。彩色的特征图，在加了拓扑半径的限制以后，就不完全一致。面对实际的研究项目的时候，到底用哪个，读者可以多斟酌一下。

第六节 Ctrl+D

在 Axial Map 的状态下，针对某个特定的元素，可以分析其“Step Depth”，快捷键是 Ctrl+ D。在前面的章节中，视线分析的那个地方，针对方格网，也有这个 Step Depth 的命令。在线段模型中，也有。

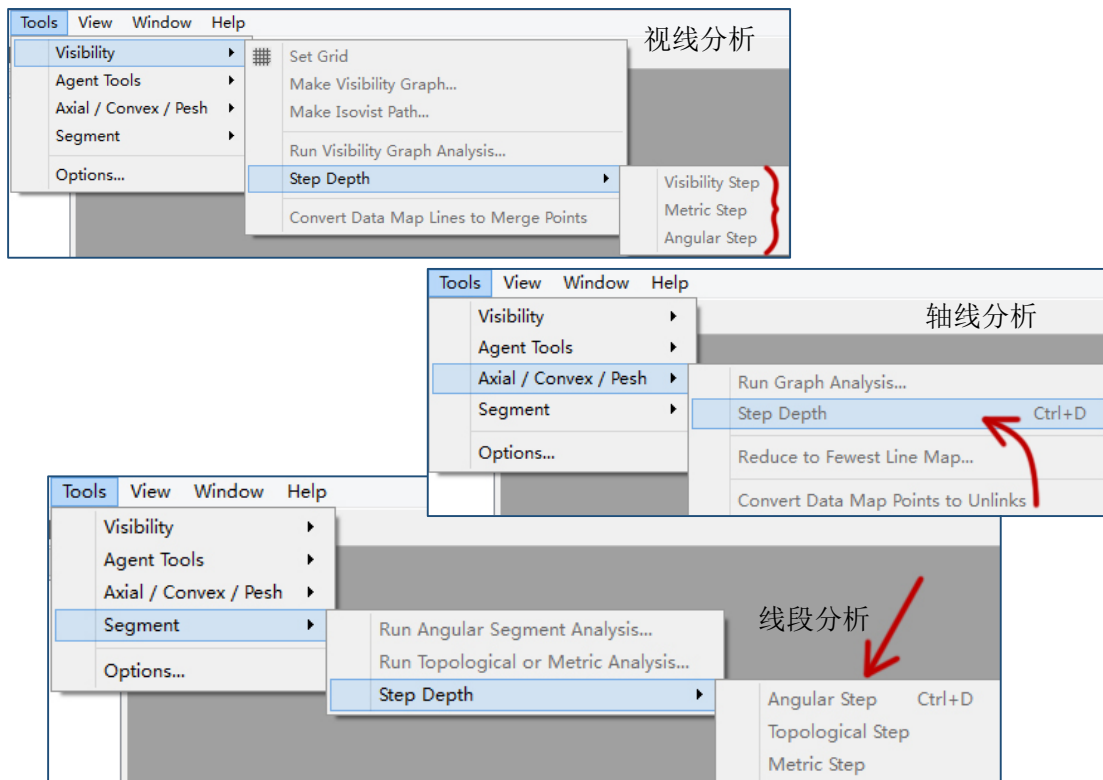


图 1.4.6-1 Step Depth

这个 Step Depth 的事儿，开始计算之前，一定要有选择集。

通过左键单击,或者拖动,可以弄个选择集出来。被选中的元素,都是黄色高亮显示的。在轴线图中,选择集非空的情况下,直接 **Ctrl+D**,就有了针对这个选择集的计算结果了。计算结果会保存在一个新的参数中,自动蹦出来的,即参数选择栏中的 **Step Depth**。

下图中,白色箭头所指的轴线,就是选择集。该轴线代表的是现在的大连的中山路。从选择集出发,到达任意一个其他的轴线,最短的拓扑路径是怎样的?有多少个拓扑步数?这个值就记在终点元素的头上。距离选择集拓扑步数越少的轴线, **Step Depth** 的值就越低,在图上就越偏冷色。

这个 **Step Depth** 也是衡量拓扑深度的一个方法。

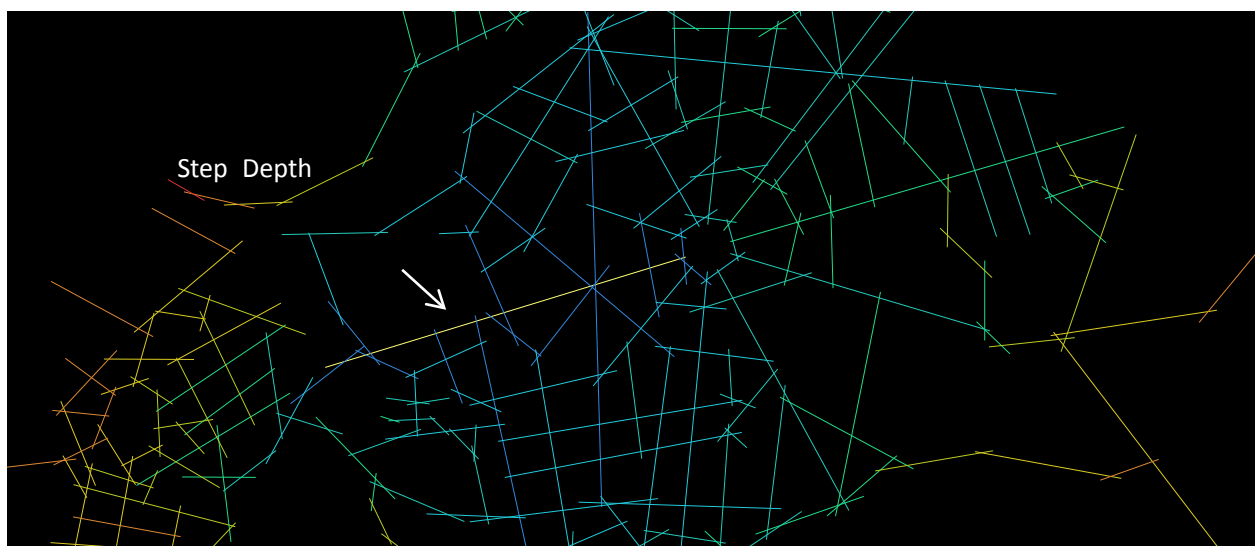


图 1.4.6-1 以白色箭头所指元素为中心,考察 Step Depth

从视线深度,到拓扑深度,到角度的深度,连空间句法的软件都叫 **DepthMap**,读者应该能体会到一些东西了吧?有了一种数学思路,顺杆爬,可以构造出许多种概念。

在空间句法中,理解了深度问题,理解了各种“深度”是依靠什么数学思想进行定义的,就理解了它的精髓。

本章小结

第五章 线段模型的计算与分析

仅仅有了轴线模型，还是有局限性。

如果你跟一个建筑师，或者城市规划师谈，竟然告诉人家，你的理论是不考虑空间的 GPS 坐标的，人家一定会皱起眉头。

首先，因为 GPS 坐标没有考虑在内，两条街之间的米制距离无法考虑进来。这是把空间的尺度丢了啊，这是大问题；

其次，一条街的不同段，其实不是一码事儿。延伸很远的一条街，与几条街相交，被打断成几段，靠近市中心的一段，与靠近郊区的一段，不论是从城市功能的分布来看，还是从人的空间认知来看，都存在着巨大的差异。简单地用一条轴线去概括，失之于笼统了。

如果把不被打断的一段街道，看成是一个元素，参与拓扑计算的话，就会有优势了。如下图所示。

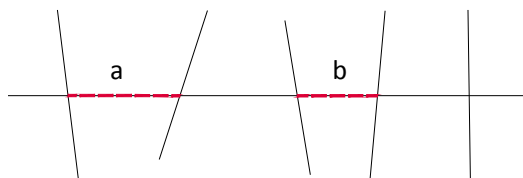


图 1.5-1 以线段作为参与计算的元素

这个线段模型的思路，很早就想到了，但是真正做成软件，变成研究人员的实用工具，出现得就比较晚了。参与计算的元素，不是轴线，而是不被打断的一个线段。线段模型是在轴线模型的基础上，进一步加工而来的，不但参与计算的元素的数目会增加，而且，元素之间的连接关系也会变得比较复杂。听说空间句法的早期，开发软件的时候，用的都是小型机，那种机器的价码，高得特别离谱。哪怕是用小型机，早期的文献当中，还是经常说，“硬件不允许”啥的。在本文写作的 2013 年，个人电脑的运算能力已经相当高了，王浩峰老师用 i7 处理器，去处理元素总数 70 万的超大系统，跑个四五天，也能完成任务。

从我们研究所的一些测试项目来看，线段模型的优势确实是比较大。一方面，米制距离可以考虑进来了，可以在线段模型的基础上，用米制半径，去限制参与计算的元素，等等吧；另一方面，任意两个元素之间的角度关系，这个角度模型，更加细致了。

空间句法这种研究方法，有一个好，只要你能提出新的数学模型，拿这个模型跟城市当中的人流物流的情况对一对，跟城市当中的功能分布的情况对一对，对得上的是学术成果，对不上的也是学术成果，懂么？其实老毛当初就是这么玩儿的。自己先讲许多话，然后找个熟悉马裂的人来，把自己的文章跟马裂的著作对一对，对得上的是继承和发扬马裂煮义，对不上的是结合中国实际的毛泽东思想。就是这么玩儿的。

在线段模型的基础上，再进行修正啊，权重啊什么的，数学模型本身就很丰富了。将来再通过 ArcGIS 里边儿的功能，将任意的带有地址的数据库弄进来，与数学模型的自身的情况进行一下相关性分析，那火力不是一般的猛。

第一节 线段模型的生成

导入 dxf 格式的文件，到 DepthMapX 中，可以直接将导入进来的东西 convert 成为 Segment Map，但是经过测试，软件有瑕疵，这个功能不稳定。我们还是啰嗦一点儿，先转换成 Axial Map，然后，再转换成 Segment Map。如下图所示。注意第二次 convert 的时候，用的是 convert active map，与第一次不同。

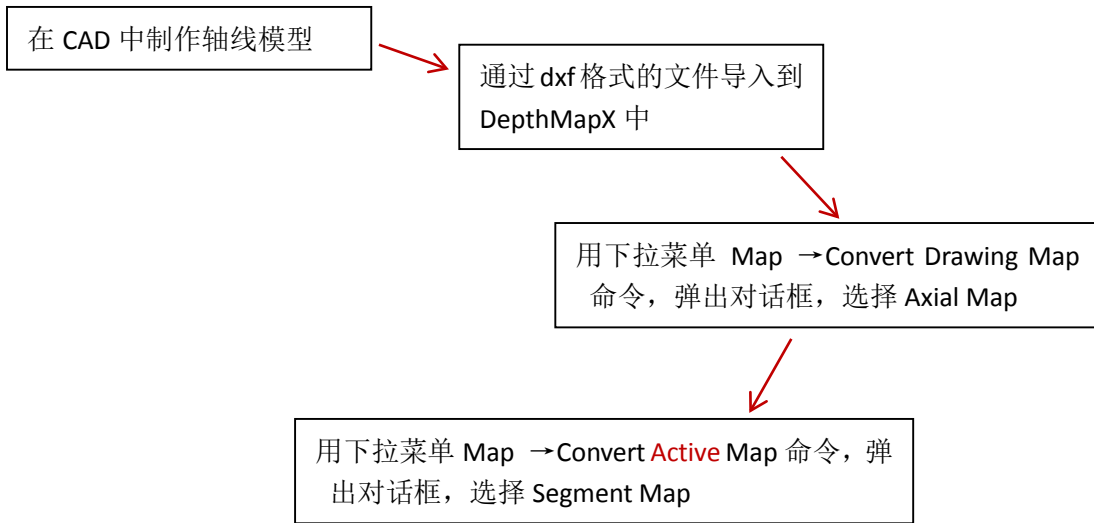


图 1.5.1-1 生成线段模型

最后一步，变成 Segment Map 的时候，那个对话框，还是要拎出来看下。如下图所示。不论是什么情况，对话框都如下这样设置。

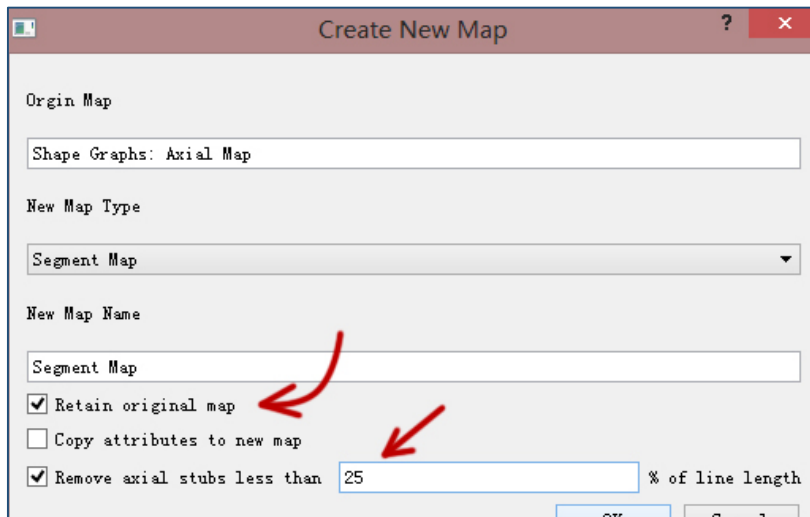


图 1.5.1-2 生成线段模型的对话框

红色箭头所指的地方，所谓 Retain original map，意思是，前一步的 Axial Map 的图层，不要删掉。而下面那个 Remove axial stubs less than 那个对话框，意思是，如果一条轴线与其

他轴线相交，那边儿留了个尾巴，如果尾巴的长度小于这条轴线的总长度的 25%，那么就可以将其删掉。在制作轴线模型的时候，不是特别强调，交接的地方要出头么，变成线段模型的时候，那些出头的小毛刺，可以通过这个对话框中的百分比的设置，给它自动删掉。熟悉 ArcGIS 的读者马上明白，这个不是 ArcGIS 里边儿的悬垂度么？

通过上面的步骤，将 CAD 输入进来的轴线图变成 DepthMapX 当中的线段模型以后，就可以进行下一步的计算了。

第二节 角度分析的软件操作

所有的线段分析的工具，都在下拉菜单的 Tools → Segment 下面。

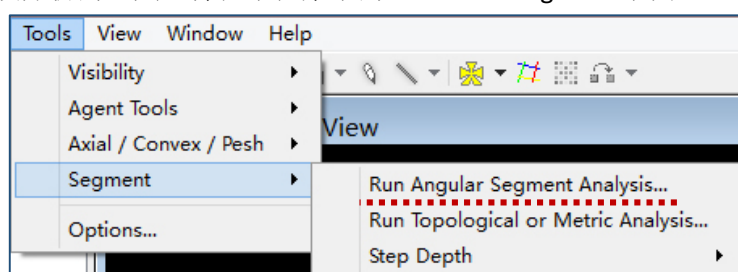
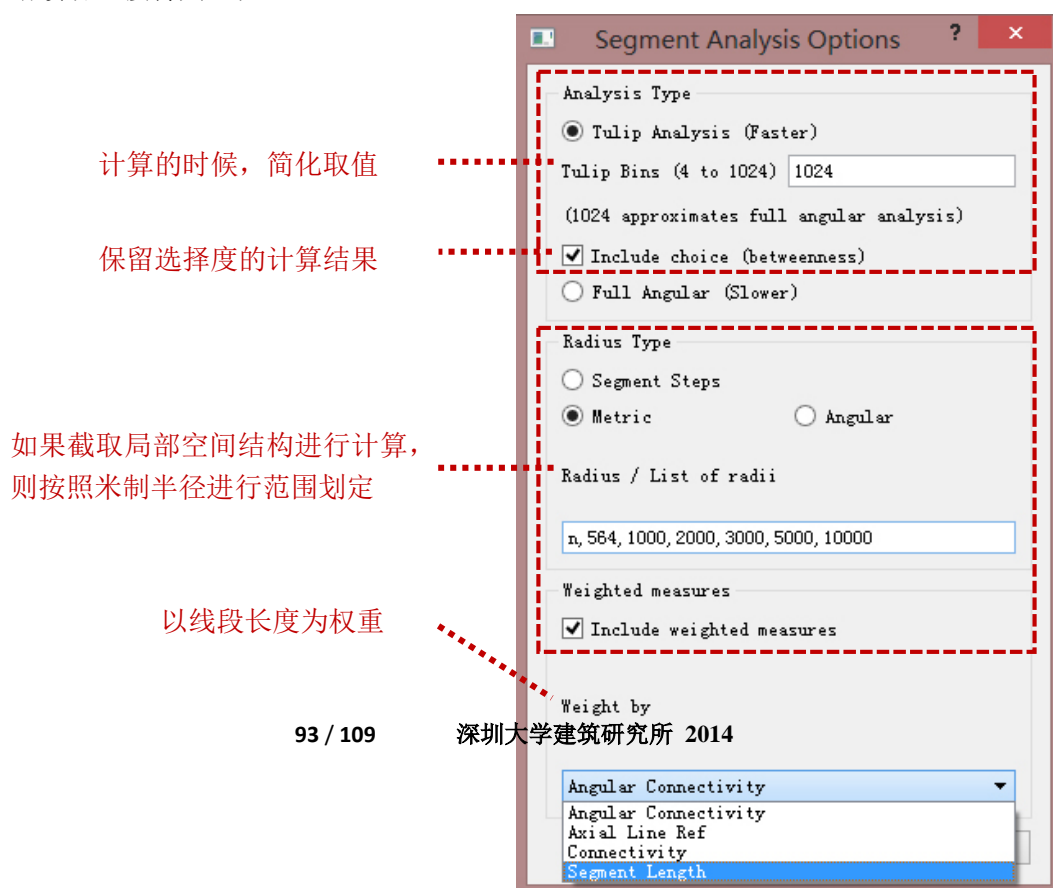


图 1.5.2-1 线段模型的计算

本节的主题是角度分析，就是点击下拉菜单中的 Run Angular Segment Analysis 这个选项，弹出对话框。其他的两个选项，操作也很简单，数学上的意思也很简单，本文中就不涉及了。

针对线段模型的角度分析，弹出来的对话框，如下图所示。在所有情况下，都是如此设置。请读者直接将其记住。



在所有的情况下，这个对话框都是这么设置的。请读者将其直接记住。

详细解释一下这个对话框的内容。最上面所谓的 **Analysis Type** 的事儿，意思是，两个线段直接相连，他们夹角是多少度？不管是用角度表示，还是用弧度表示，都有个精确到小数点后面多少位的问题。软件当中，为了节省计算资源，将 π 等分成 1024 个区间，两个线段的夹角，落在哪个区间了，就用该区间的近似值表示。这是一种简化的算法。

下面有个方框，就是 **choice(betweenness)** 的那个勾，可以打上。将来计算的结果当中，这个 **choice** 的计算结果会给出来。

对话框中间的部分，是设置限定条件的。我们以米制半径为限制条件。**n** 表示不限制条件，直到穷尽了所有的可能性以后，计算结束。**n** 屁股后面，用英文输入法下面的逗号，隔开，然后输入各种米制距离。假设我们输入 1000，意思是，针对一个线段，以其线段中点为原点，以 1000 米为半径，画一个圆，在这个圆以内的，包括与这个圆相交的元素，就选进来了。将入选的元素，作为一个局部的空间结构提取出来，又可以进行整合度之类的计算了。

半径设定好以后，还可以选一下权重。**include weighted measures** 那个方框，勾选上，下面的下拉菜单，就可以选择以什么为权重了。在所有的情况下，都选择以 **Segment Length** 为权重。因为线段的长度，可能隐含着一些信息。一个街道越长的话，就暗示，这条街两侧的内容比较多，这条街上会有比较多的城市活动。也就是说，我们从数学上简单地假设，一条街越长的话，这条街越重要。通过这个 **Segment Length** 进行修正以后，有些参数的算法会更加靠谱一点儿。

设置完成以后，OK 一下，软件就开始算。计算完成后，自动跳到 **T1024 Choice** 上了。旧版当中，是自动跳到 **T1024 Integration** 上了。不要紧，手工从参数选择栏当中，左键单击，可以查看所有的计算结果。

敝人通过 **eps** 格式的文件，将 **T1024 Integration** 的结果，输出到 **WORD** 中。为了避免歧义，这个图的说明中，要说，这是基于 **Segment Map** 的 **Integration** 的情况，半径是多少多少。读者如果是一步一步从头读的本教程，那么就会知道，我们有凸空间的模型，有无限细分的方格网的模型，有轴线模型，有线段模型，都能计算整合度。为了避免混淆，应该将基于什么模型，用的什么算法，说清楚。

轴线模型这里边儿东西最多，算法最为复杂，但同时，轴线模型也最有学术价值，在各种场合使用得最多。读者可以多体会一下。

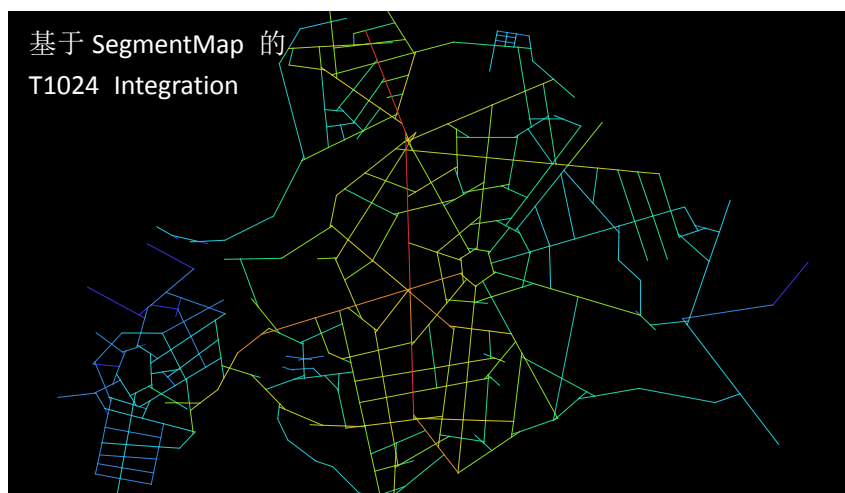


图 1.5.2-3 基于 Segment Map 的 Global Integration

第三节 角度基础上的整合度和选择度

1) 线段模型的整合度

整合度的事儿，在前面的章节中，占用的篇幅就已经不少了。

线段模型当中，从任意一个元素出发，到达任意一个其他的元素，可以有多种“最短路径”的选择方式，本节重点是以角度为衡量标准，而测算出来的最短路径。

线段模型当中，某条路径上，可以将元素都看成向量，向量之间的夹角，进行加总，就是这条路径的 Angular Total Depth。

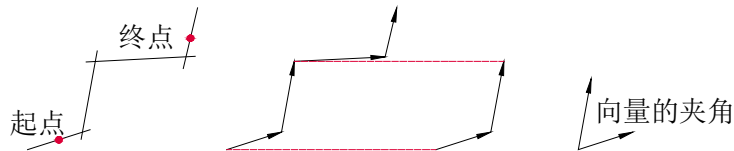


图 1.5.3-1 路径的转角加总

这条路径上，从起点开始，依次进行向量夹角的计算，直到终点为止，得到的值加总，就是这条路径的 Angular Total Depth。起点终点相同，可以找到一条路径，其 Angular Total Depth 是最小值。

以某一个元素为中心，进行空间重映射，到达系统中任意一个其他的元素，都进行角度意义上的最短路径的选择以后，将这些路径的 Angular Total Depth 再来一个全系统加总，数值反馈回来，记在中心空间的头上，就是它的 Angular Total Depth。

有了这个 Angular Total Depth 以后，再按照老的套路，可以计算 Integration 的值。在软件当中，面对几万个轴线，没有必要精确地进行角度计算，将一个 π 等分成 1024 个区间，两个向量之间的夹角，落在哪个区间，就取这个区间的角度值为近似值。在近似值的基础上，可以计算近似的 Angular Total Depth 的值，接着计算近似的 Integration 的值。这就是软件给出的 T1024 Integration 的含义。

2) 期望选择度

前面轴线模型那里，从拓扑学的道理上，给出了选择度的算法。选择度就是出现在最短路径上的次数。

现在还可以将此算法加以改进。

过去的简单加总的算法，存在的主要的问题是，最短拓扑路径可能不是唯一的。参见下图。好几条路径都是最短路径的情况下，要为不同的路径，分配权重的。注意，我们这里说的是拓扑意义上的最短路径。

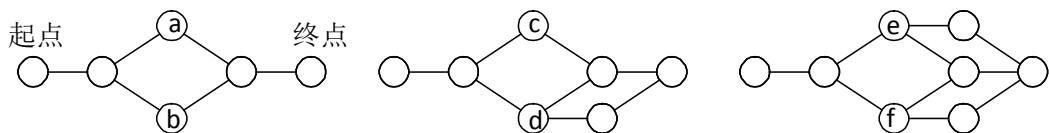


图 1.5.3-2 路径的权重问题

上图的意思是，三个拓扑结构，我们都是从左边开始走，到右边结束。第一个情况，a 和 b 都是处于最短拓扑路径上。现在观察中间这个情况，我们对比一下 c 和 d 的情况，因为从 d 往终点走，可选的路径比 c 多，我们可以视为两者的重要性不同。最后发展到 e 和 f 的情况，两者在路径选择当中的重要性就又相等了。

上图中，从起点往终点走，a、b、c、d、e、f 都处于最短拓扑路径上，当人们面对岔路口的时候，以丢硬币的方式，来决定走哪条路的时候，最终实际走出来的路线，就成了一个概率问题。都是最短拓扑路径，可是不同的路径之间，被选择的概率并不相同。按照 Bill Hillier 的说法，我们可以将这种选择路径的问题，理解为沿着拓扑路径进行选择度的分配(Hillier et al., 2012, pp. 157-160)的问题。

下面详细思考一下路径选择的概率问题。

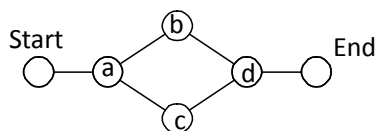


图 1.5.3-3 可能性的分配

上图中，从起点往终点走，从起点开始，100%的人要走到 a 点，那么 a 点被选择的概率是 100%。从 a 点出来，可以走 b 也可以走 c，那么如果以抛硬币的方式来决定，b 被选择的概率是 50%，而 c 被选择的概率是 50%。

从 b 出来，继续往终点走，100%的人都要经过 d，那么 d 元素被选择的概率是 (b 被选择的概率) * (从 b 出来时，d 被选择的概率) = 50% * 100% = 50%。对于 d 来说，这还没完。从 c 出来的那条路线，也要经过 d，还可以计算一下。最终，所有可能性加在一起，d 元素被选择的概率是 50% + 50% = 100%。从 d 出来，往终点走，也是 100%的人要到达终点，终点被选择的概率是 (d 被选择的概率) * (从 d 出来时，终点被选择的概率) = 100% * 100% = 100%。

这个百分数可以换成小数，50%的概率就是 0.5，这样就清爽了。选择度的分配，以小数表示，可以再弄个例子，帮助读者理解一下。

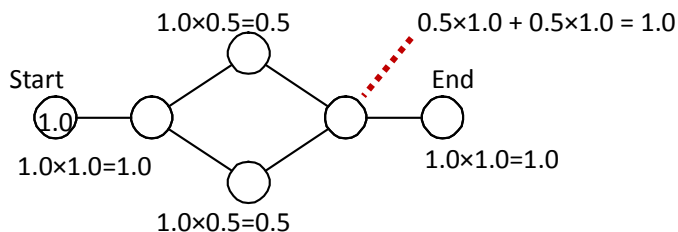


图 1.5.3-4 被选择的概率

上图的意思是，每次路径选择的时候，都是按照概率来的，这样，不同的元素被选择的概率是不同的。

重复一下前面的章节中提到的概念，Choice 的值，就是针对某个具体的元素，出现在最

短路径上面一次，choice 的值就加一。现在有了概率的分析以后，我们可以用(概率)•(1 次)得到的这个值，再进行加总。可以将其称为“期望选择度”。

数学上，一个东西出现的概率是 0.5，就说，期望值是 0.5。选择度加总的时候，如果是包含了概率的分析的，就是“被选中的期望值”的加总，简称为“期望选择度”。英文是 Expected Choice。

有了这个期望选择度以后，参照 Normalized Choice 的公式，可以拿 Total Depth 修正一下期望选择度。公式如下。

$$\text{NormalizedExpectedChoice (a)} = \frac{\text{Expected Choice (a)}}{\text{Total Depth (a)}} \quad (\text{公式 1})$$

这个公式，分子分母都是针对 a 元素的。也就是说，这个修正过后的期望选择度，还是针对具体的元素说的。

上面的公式当中，并未限定半径。我们可以按照某种条件，限制一下参与计算的元素的范围。就是说，按照某种限制条件，截取一个局部的空间系统出来，再进行上述的 NormalizedExpectedChoice 的计算，也是可以的。

因为线段模型是可以将 GPS 坐标考虑进来的，那么，就以线段的中点为圆心，以一定的米制距离为半径，进行限制，就可以了。公式如下。

$$\text{NormalizedExpectedChoice (a) within(r)} = \frac{\text{Expected Choice (a) within (r)}}{\text{Total Depth (a) within(r)}} \quad (\text{公式 2})$$

上面这个公式中，r 的数值，是一个米制的距离。

到了这里，不妨对本小节做一个总结。本小节中用到的新概念和提到的新公式，都会看着比较眼熟。以前都出现过类似的东西。本小节就是在选择度的基础上，进一步将概率问题考虑进来，而弄的一些事儿。最短的拓扑路径可能并非唯一。如果最短拓扑路径并不唯一，我们可以按照概率的思想，对其期望选择度进行考察。

3) Angular Choice

前面的小节中，这个 Choice 的定义，还是按照拓扑学的最短路径来的。当然还有别的最短路径的定义方式。可以按照最小转角的法则，来找最短路径。一个元素，如果出现在转角最小的路径上一次，它的 Angular Choice 的值就加一。针对具体的元素 b，以系统中任意一个元素为起点，任意的另一个元素为终点，找到其最小转角的路径，如果 b 出现在该路径中，就加一，穷尽了所有的可能性之后，就可以知道 b 元素的 Angular Choice 的值了。

按照老套路，我们可以对一个元素的 Angular Choice 的值，进行修正。公式如下。

$$\text{NormalizedAngularChoice (b)} = \frac{\text{Angular Choice (b)}}{\text{Total Angular Depth (b)}} \quad (\text{公式 3})$$

不论是 Angular Choice 还是 Total Angular Depth，都可以加入新的限制条件。因为线段模型是可以将 GPS 坐标考虑进来的，那么，就以线段的中点为圆心，以一定的米制距离为半径，进行限制，就可以了。公式如下。

$$\text{NormalizedAngularChoice (b) within (r)} = \frac{\text{Angular Choice (b) within (r)}}{\text{Total Angular Depth (b) within (r)}} \quad (\text{公式 4})$$

公式 4 中，r 是米制距离。

到了这里，还没完，主要是软件编程的时候，会遇到麻烦。主要是线段模型的复杂度比较高，Choice 的值，Total Angular Depth 的值，可能会变得非常高，最后连科学计数法都用上了。可能系统中有一百个元素，九十九个元素的 Total Angular Depth 的值都在一个较小的范围，只有一个元素，Total Angular Depth 的值一下子冲到十的几十次方，空间句法的赋予颜色的方法，是将最大值和最小值之间的这个数值段，进行十等分，每段儿赋予不同的颜色。如果少数几个元素的值，突然冲得很高，那么将来图上反映出来的就是一片蓝色，加上几个有限的红色的元素。这就失去了彩色特征图的意义了。

可以用取对数的方法，加工一下。

自然对数曲线，如下图所示，我们把 x 轴看作输入，y 轴看作输出。去了自然对数以后，x 轴的输入非常大的情况下，y 轴输出的结果，这个增量其实非常小。而且这是一个单调函数，不会改变由大到小的排列顺序。

取自然对数，或者取以十为底的对数，函数的图像都是差不多的，取对数的方式，就经常用来把波动很大的数列，在元素的排序不变的情况下，压缩波动范围。

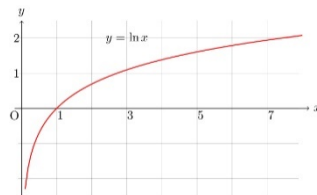


图 1.5.3-5 自然对数曲线

写软件的时候，公式 4 写成如下的形式：

$$\text{NormalizedAngularChoice (b) within (r)} = \frac{\log\{\text{Angular Choice (b) within (r)}+1\}}{\log\{\text{Total Angular Depth (b) within (r)}+3\}} \quad (\text{公式 5})$$

上面公式中，分子分母的大括号中，后面都加了个常数，主要原因是，大括号里面的值，要大于 1，将来取了对数，得到的东西才是正值。对于我们的研究来说，负值无意义。所以要确保大括号里面的东西大于 1。分子上的，Angular Choice 的最坏的情况，就是等于零，从未出现在最短路径上，那么后边儿+1，就符合要求了。分母上的，有些元素 Total Angular Depth 的值，是空的，软件自动为其赋值为-1，按说是+2 就满足要求了。但问题是，分母不能为零，log1=0，又是错误。所以+2 也可能会造成麻烦。干脆写软件的时候，就后边儿+3，这样分母取完对数，肯定是大于零的正数。

其实后边儿加个常数，再大一些也可以的。因为 Angular Choice 的值可能高达几万，取对数之前，后边儿加个五加个十，影响都是微乎其微。分母的 Total Angular Depth 也是这样的。扯这些是希望读者知道，写软件的时候，公式可能还要稍微修改修改才能用。如果读者自己有编程的能力的话，这些处理办法都是老生常谈的事儿了，广为人知。

第四节 再谈整合度

线段模型当中的整合度，是在角度的基础上，在 **Total Angular Depth** 的基础上，发展起来的。这么干的主要原因，在于，经过研究，人们对于路径的选择，更倾向于最小转角的路径。

本节的内容，来自于 **Bill Hillier** 本人的讲座内容。原始文献是段进的那个《空间句法与城市规划》(段进, 2006)，原书 36 页左右。

有人假设，人在城市中活动的时候，会选择米制距离的最短路径。有些做交通规划的人就是这么想的。这种思路可以被称为“**Shortest Path**”；

有人假设，人在城市中活动的时候，会选择拓扑学意义上的最短路径。这种思路可以被称为“**Fewest Turns**”。我们当初在 CAD 当中，制作轴线图的时候，用的原则是最常且最少，意味着，抽象出来的这个拓扑路径，已经尽量把不必要的转折都省略了，剩下的都是必要的转折，每一个拓扑步数，都代表着一个必要的转折，每个转折，都是有意义的。

还有人假设，人在城市中活动的时候，会选择转角总和最小的那个路径。这种思路可以被称为“**Least Angle**”。

我们可以先进行一下城市活动的调查。样本数弄得高一些，将一些具体的步行的人流活动，起点在哪，终点在哪，路径是怎么选的，记录好。

然后，针对一个实际发生的人流，我们知道它的起点和终点，我们回到数学模型上，找到这两个点，分别找到三种路径，也就是起点和终点确定的情况下，可以手工计算，找到 **Shortest Path**、**Fewest Turns** 和 **Least Angle** 的三种算法的情况下，分别是什么路径。

最后一步，我们将实际发生的人流，和三种路径对一下，看看哪个选择路径的方式出现的概率更高。

书中给出的结论是，人流的实际情况，与米制距离的 **Shortest Path** 的相关性其实很低，相关系数的峰值才 0.4 不到，可以认为不相关。

人流的实际情况，与最短拓扑路径---**Fewest Turns** 和最小转角---**Least Angle** 的相关性会高一些，最着人流的起点和终点的米制距离逐步扩大，相关系数的峰值能达到 0.8 以上。

最短拓扑路径和最小转角，这两个在建模的时候，有隐含的相关性。因为每一个拓扑步数，就是一个转折。

在线段模型当中，把中心元素到其他任意一个元素的最小转角的路径找出来，将该路径的角度值进行加总，再进行全系统加总，得到这个元素的 **Total Angular Depth** 后，就可以进一步加工得到 **Integration**，算法的这种设计，其根本原因就在于，他们事先研究了实际发生的人流，证明实际发生的人流与某个算法相符，于是回到数学模型中，继续对该算法进行深入加工，逻辑上还是说得通的。

另外，我觉得还要强调一个与伦敦这座城市相关的问题。空间句法这种思路，这种做事的感觉，老早之前就有了。

我们的鸦片战争那会儿，一八五几年，有个叫 **John Snow** 的人，来到了伦敦。这个人来自约克郡的煤炭工人的家庭，早年在家乡作为学徒，入了医学的门，来到伦敦后，经过一段时间的努力，拿到了资格证书，成了一个发展得不错的医生。1854 年，伦敦爆发了流行病，霍乱，很多人病死。这个传染病引起了斯诺的兴趣。

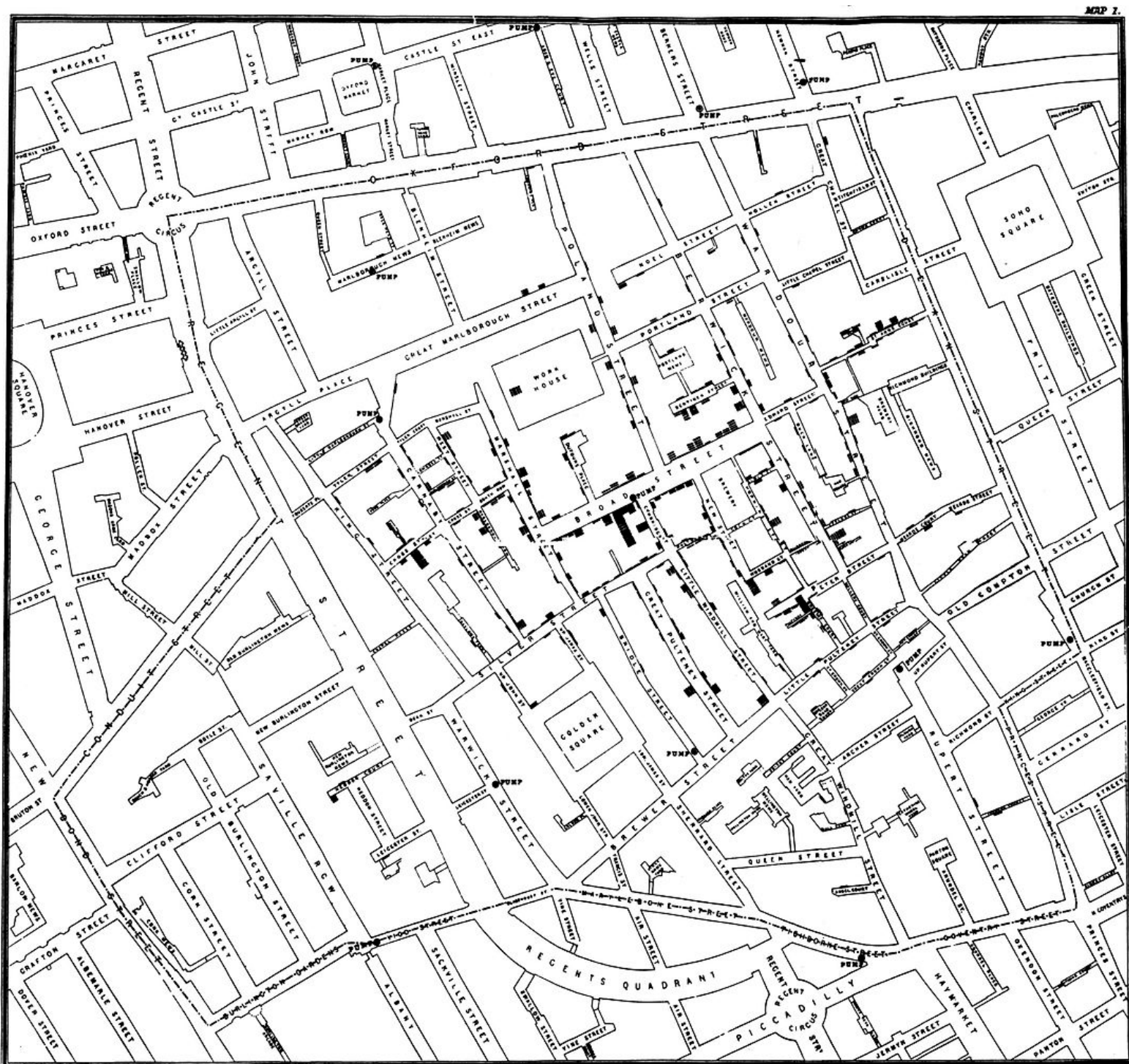
斯诺找来一些帮手，开始了医学调查。他开始对居民进行走访，并在地图上将病例所处的位置标好。按现在话说，就是开发了一个带有 **GPS** 信息的医学数据库。通过研究霍乱病例在空间上的分布状态，斯诺发现，病例的空间分布的状态，是在某个公用的水泵的管辖范围

内。离开了这个水泵的供水范围，没有出现霍乱的爆发或流行。斯诺同时发现，喜欢喝酒的人受到感染的机会少，因为酒精有一定的杀毒灭菌的作用。

斯诺后来针对水泵进行调查，发现水井附近，有个粪坑，患过病的人的粪便，会通过这个粪坑，将饮用水源给污染了。

在这些发现的基础上，斯诺采取了措施，如严格控制可疑水源的使用，针对城市当中的其他水源地，要采取妥善的保护措施，等等。这样，斯诺就成了现代公共卫生的这个领域的开创者。

把斯诺的图摆上来看看，对于做空间句法研究的人来说，这真有种似曾相识的感觉。



C. F. Coe's Map, Southampton 1844 London

SCALE 80 INCHES TO A MILE.

以前干成过事儿，对于后人，总是会产生持久的启发。流行疾病在城市空间当中的分布，可能会揭示一些事儿，那么，扩展到任何一类其他的城市活动，比如某个类型的商业，会不会在城市空间当中的分布，也是有规律可循的？这背后会不会隐藏着什么更加深刻的原因？

空间句法的这一套，并未满足于制作城市地图。当然，制作城市地图本身，这也是个数学建模的事儿。空间句法的创新的地方，在于依托于拓扑学等数学方法，可以在城市地图的基础上，将空间抽象出来，将空间变成数学上的概念，将空间变成可以参与运算的元素。这样，设计出来一大堆算法，对于空间本身获得了更多的数学上的理解。这样，在数学的基础上，人们思考问题的深度，思考问题的角度，就非常多了，将来再跟城市活动的实际情况一对，互相之间有个参照，这火力就更猛烈了。没有空间句法的时候，凭着感觉，人家也能进行归纳总结。不用空间句法，有些研究也能做，但是描述和分析问题的时候，精确性就很差了。如果说要精确地分析与“功能”沾边的那些问题，空间句法是目前为止最好的工具了。

其实什么事儿，发展到了实用的技术的阶段，已经是很没有挑战性的了。真正的困难在于，如何去思考？怎么去想这个事儿，思考的方法是什么？空间句法对于我们的价值也就在于，原来还可以这么看问题？！

软件的技术，操作具体项目的技术，这些不是最重要的。最重要的是思考问题的方法。

第五节 街网密度的问题

街网密度的事儿，在空间句法当中有两种衡量的办法，一个是看单位面积上，街道的长度加总起来，是什么情况；另一个是看单位面积上，有多少个不受打断的街道段。

本科阶段的教材中，不是有街网的线密度、面密度么，线密度空间句法当中也有，面密度没有。空间抽象成轴线了，道路面积的信息，是没法直接弄了。不过，放在 ArcGIS 当中，面密度的事儿，还是可以研究的。DepthMapX 这个软件，体积很小，算法比较简陋，就满足于线密度和元素密度的水平了。

线段分析的步骤，按照本教程前面所提的，一步一步操作下来，T1024 Total Segment Length 和 Node Count 已经有了，这两个参数，后面还可以加米制半径的限制条件，生成新的计算结果。

还是大连那个例子。在 CAD 当中，找那些有代表性的街道，看看它在百度地图上是什么长度，把尺寸对好，然后将轴线图导入到 DepthMapX 当中，再变成线段模型，进行分析。

还是线段分析的那个对话框，在对话框中，我是这么输入的：

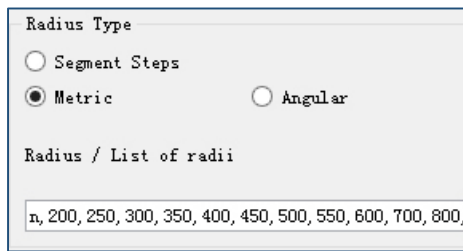


图 1.5.5-1 在线段模型的角度分析中，设定米制半径

计算完成后，直接到窗口左下角的参数选择栏当中，去点击 T1024 Node Count R300 metric 这个参数，得到的结果如下图所示。



图 1.5.5-2 基于 Segment Map 的 Node Count R300 metric 的情况

这个 Node Count R300 metric 的算法，是说，针对某个元素 E_i ，现在参与运算的都是不被打断的一个直线段，以这个 E_i 的中点为圆心，以米制距离的 300 米为半径，画一个圆，只要有元素出现在这个圆内，或者与这个圆相交，都算进来，数一数一共有多少元素，这个数值反馈回来，就记在 E_i 头上，就是它的 Node Count R300 metric 的值。

上图中，红色的区域表示数值最高，我们可以发现，红色区域大概分成三块儿，这就是城市路网密度最高的几个局部区域了。

路网密度高，代表什么意思，读者可以先反思一下。这只是以 300 米为半径，进行的考察，半径可以逐步放大的，比如将半径放到 564 米，情况就变成下图这样了。这样的变化，又能告诉我们什么呢？

564 米的意思是，以 564 米为半径，画一个圆，这个圆的面积正好是 1 平方千米，方便用密度单位来说话。别的半径，比如 3000，你要想说“密度”的事儿，还要手工去除一下面积，稍微费点事。



图 1.5.5-3 基于 Segment Map 的 Node Count R564 metric 的情况

上面这是按照街道段数来的，Total Segment Length 是按照长度来的。仍然是画个圆，圆内部或者与圆相交的元素都算进来，将其长度做一个加总，数值反馈回来，记在中心空间的头上，就是线密度的情况了。

我的导师曾经提到“核心街区”的概念，在师兄的论文里面出现过，不妨在这里提一下，供读者参考。

核心街区的意思是说，将考察街网密度的时候，往往是小半径的条件下，红色区域与老城区相重合，逐渐放大半径，红色区域往往会漂移到现代城市的中心。造成此种情况的原因，是前现代化时期，城市街块往往较为细碎，路网密度较高，后来，在这种城市结构的基础上发展现代城市，新出来的这一块，并不能摆脱依附关系，依赖于老城。直到某个确定的时间节点，城市会突然地猛烈地扩张，老城会被撇在一边，新城与老城会发生某种分裂。重要的是要考察这个时间节点，在这个时间节点之前，街网密度较高的那个区域，发挥重要的城市功能，在各种城市活动中所占的比重都很大，所以就被称为“核心街区”。

中国不允许贫民窟的存在，所以提出这个核心街区的概念，感觉上还是比较偏“普世”的。西方人不是这样的，他们老说要普世普世，意思就是现实当中很不普世呗。早期，犹太人在欧洲的时候，总是被限制在 Ghetto 里面，这就是犹太人的核心街区了。核心街区是要按人种，按身份，分别考察的。到了现代社会，在拉美国家，印度，等等，大城市当中的贫民窟，也存在这种按照身份进行隔离的情况。城市扩张的时候，贫民窟按照贫民窟的方式进行扩张，按城市规划进行扩张的就是另外一种方式。城市的功能上，往往从一开始就是割裂的，各是各的。

按照普世的理想，进行城市建设的，也可能引发意想不到的麻烦。底特律就普世了一把，结果什么人都混在一起，1967 年发生了骚乱，是黑人发动的骚乱。1950 年代，底特律白人占 55%，骚乱以后，白人开始离开这座城市，在本文写作的 2013 年，白人占比 11% 左右。底特律的人口从上世纪六十年代的 170 万人，下降到现在的七十多万人。要知道，全美国的

人口总量，一直是在增长的。底特律衰落了现在，都成了探险圣地了。许多人都在谈论底特律起死回生的事儿，又是城市规划又是产业投资的，其实问题的根子在哪啊？

回到本节的主题，这个街网密度的问题。这不是一个简单的技术问题，可以将街网密度的事儿，沿着历史的时间轴展开，来考察一下街网密度上升与下降的情况，里面蕴含的意义，应该是相当大了吧。。。。。

第六节 通过 Properties 对话框查看计算结果

从参数选择栏那里，任意一个参数，可以右键单击，在右键菜单中，选择 Column Properties 这个选项，弹出对话框。从 DepthMapX 窗口顶部的下拉菜单，也有这个命令。参见下图。

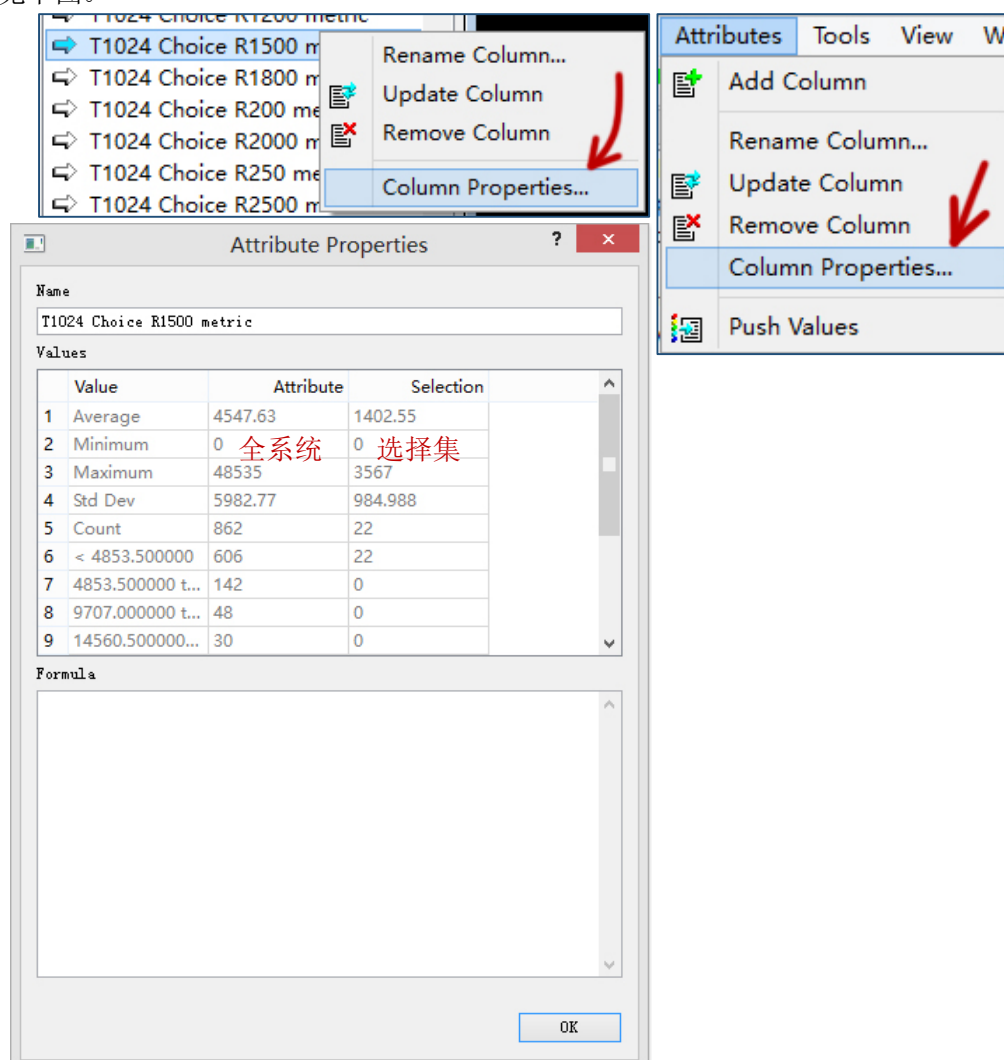


图 1.5.6-1 查看 Attribute Properties 对话框

弹出来的对话框，最左边 Value 那一列，表示这一行的数值是什么意思。Attribute 那一列表示全系统的状态，Selection 那一列表示选择集的状态。

比如上图中，看行号，第 3 行，Maximum，这一行都是最大值，全系统的最大值是多少多少，选择集的最大值是多少多少，都罗列出来了。

看 Value 那一列出现的单词，Average 是平均值，Minimum 是最小值，Maximum 是最大值，Std Dev 是标准方差，Count 是元素总数，这一列下面有些数值段。

对话框中，看行号，第 6 行，写着小于多少多少，意思是，在小于某个数值的这个区间内，全系统有 606 个元素的数值，处于这个数值段，而我手工框选了一个选择集，选择集中有 22 个元素，处于这个数值段内。

这些数值是从哪取来的，要看你在参数选择栏当中，选中了哪个。对话框最上面那个 Name 那里，也显示了数值的来源。

这个对话框的方便之处就在于，全系统的情况，和选择集的情况，对比起来比较方便。比如现在对比 1000 米半径的整合度，我们对四个不同的地点感兴趣，那就分别框选四个地点，从 Properties 对话框，读取选择集的数值，特别方便。

本章小结

结语

本教程的重点是 DepthMapX 这个软件。按照由简单到复杂的原则，依次安排了凸空间、无限细分的方格网、轴线模型、线段模型的内容，将软件操作捋了一遍。

写这个东西的主要原因是，学习空间句法存在难度。西方人写的关于空间句法的东西，汗牛充栋了已经，直接去面对西方人的东西，会陷进去。人的精力是有限的，敝人付出大量的精力，弄这么个教程，如果能够帮助读者节省时间和精力，更快地进入到空间句法这个门里面，敝人就值了。

西方人很厉害，但是西方人也有自己的麻烦。西方人的厉害在于，学术有关的、工程有关的标准，这个评价标准，都是他们定的。西方人的麻烦在于，现实世界的复杂性，并不能按照人的意愿，一厢情愿地加以抽象化、概括化，学术上确立的标准，总是存在漏洞，从而将标准本身给否了。

如果读者通过学习空间句法，没有学到思考问题的方法，没有看到空间句法本身能够成立的逻辑边界，而是仅仅学到了技术，仅仅是通过技术而确定了标准，以空间句法确立的标准为标准，而增强了否定别人，否定现状，甚至否定一切的能力，将空间句法弄成放之四海而皆准，那么鄙人的工作就彻底失败了。

知识越多，并未使得我们远离混乱的充满偶然性的状态，反而是知识和技术越发展，我们认知和掌握的事物越多，我们就越发现未知的事物越多。建筑学的迷人之处就在于，技术、历史、经济、政治、法律、宗教、家庭等等社会系统，都对我们的学科有重大的影响，任何将建筑学简单化、抽象化的企图，都会面临重大的问题。建筑学本来并不神秘，可是西方人做学问的方法，也就是“科学的方法”，把问题弄复杂了。

科学这个词，是从日本传来的。所谓的科学，就是分科的学问，就是分门别类的学问。当代的建筑学不同于以往的地方，就在于其内部的高度分化。如果说中国传统的建筑学是按照身份等级进行社会阶层意义上的分化的话，那么，在当代的建筑学就主要表现在学术本身在功能方面的分化，由此带来了建筑物的功能方面的分化。是学术的高度分化，学术的高度专业化和自我规范化，这些东西在先，而建筑物的功能分化在后。

现在的建筑学，其本身就是一个在功能上不断分化的系统。因此，建筑学的研究方法，也要求每一次针对一个单一的功能，进行详尽的研究。这造成很大的麻烦。

鄙人扯的这些东西，德国哲学家鲁曼都说过。鄙人的意思是，空间句法本身，就是一个封闭起来的独立技术体系，您要是学建筑入了门，有了自己的鉴别能力，有了自己的明辨是非的能力，那么您把空间句法的这个技术系统拿过来，从侧面帮助一下，很受用。如果您对于建筑学还没入门，没什么鉴别是非的能力，那么空间句法就是个漩涡，您进去就不一定出得来了。这玩意像肾上腺素一样，稍微来一点儿，人就快炸锅了，火力不是一般的猛。

再重说一遍，建筑学本来并不神秘，也不是深不可测，但是西方人做学问的这个方法，就造成了建筑学内部不同的学术门类，和不同的技术体系的高度分化，造成是事实上的隔离。俗话说隔行如隔山，说的正是各个学术门类之间、各个技术体系之间的不可化约性和各系统意义的不可翻译性。

从空间句法的这些东西来看，技术系统越发展到高级的阶段，就越呈现出游戏的特征，越伴随着理论本身的不确定性，同时，将理论运用到实际中的时候，风险也是越来越大。认为理论就是福音，只会顺风顺水地推动学术和工程应用的发展，只会使事情变得顺利，这种

看法是错的。

本节的最后，鄙人觉得，空间句法的研究方法，可以被总结为，是数学与人类学的综合。空间句法调查的什么功能啊，行为活动啊，这都可以被理解为是人类学的调查。而空间本身，又被概括成了数学模型。所以鄙人斗胆总结一句，空间句法就是数学与人类学的综合。

其实这个数学与人类学综合的事儿，我们的老祖宗早就干了。易经就是这样的啊。易经的基本思路是说，针对同一个事物，可以从象、数、理三个层面上去分析探讨。易经的图式化的表达，就是世界上最早的二进制。

DepthMapX 这个软件，其实也是“象数理”，一样的。

理，既是物理，又是事理，又包含“人理”，各种理，只要是你有个逻辑上说得通的关联，就可以将其定义为数学问题；数，是将参与计算的元素，按照某种理，按照某种规则，进行全系统的加总，得到的数值，记在中心元素的头上；而象，就是软件提供的彩色的数值特征图，全系统的状态可以说是一眼望到底。

不管研究对象是新的，还是理论猜想是新的，放到空间句法中来，象数理，对得上的，是新发现；对不上的，也是新发现。

有两句著名的诗句，“闲坐小窗读周易，不觉春去已多时”。空间句法的这个体系，带给我们的，应该也是那种思维的乐趣吧。把纷繁复杂、千头万绪的现象，变成思维上可把握的十分简洁明了的东西，会非常的过瘾。

欢迎读者猛烈批评，Email: aquitaine@qq.com。

[the End]

Date of Finish: September 3, 2014

a Moo.Yee. project

引用必须按照学术规范注明出处。

抄袭将遭严厉追究。

参考文献

- Hillier, B. (2008). *Space is the Machine: a Configurational Theory of Architecture* (杨滔, 张信 & 王晓京, Trans. 3 ed.): 中国建筑工业出版社.
- Hillier, B., Yang, T., & Turner, A. (2012). Normalising least angle choice in Depthmap, and how it opens up new perspectives on the global and local analysis of city space. *The Journal of Space Syntax*, 3(2).
- Thaler, U. (2005). *Narrative and Syntax: new perspectives on the Late Bronze Age palace of Pylos, Greece*. Paper presented at the Space Syntax: 5th International Symposium, Amsterdam.
- Turner, A. (2001). *DepthMap: A Program to Perform Visibility Graph Analysis*. Paper presented at the 3rd International Symposium on Space Syntax, Georgia Institute of Technology: GA.
- Turner, A. (2004). Depthmap4: a researcher's hand book. <http://eprints.ucl.ac.uk/2651/1/2651.pdf>
- 段进. (2006). *空间句法与城市规划*: 东南大学出版社.
- 赵红军. (2005). *交易效率、城市化与经济发展*: 上海人民出版社.