



**Mass
hopper**
**参数化
设计** 教程

中国建筑工业出版社

祁鹏远 (Skywoolf) 编著

Grasshopper参数化设计教程

祁鹏远 (Skywoolf) 编著

中国建筑工业出版社

图书在版编目 (CIP) 数据

Grasshopper 参数化设计教程 / 祁鹏远编著. -- 北京:
中国建筑工业出版社, 2017.7
ISBN 978-7-112-20728-2

I. ①G… II. ①祁… III. ①建筑设计-计算机辅助设计-三维动画软件-教材 IV. ①TU201.4

中国版本图书馆CIP数据核字 (2017) 第096989号

本书是由 NCF 参数化建筑联盟官方出品、站长祁鹏远 (Skywoolf) 亲自编著的一部以参数化设计思维培养和 Grasshopper 软件教学为核心内容的二合一教程。书中提出的“参数化六艺”、“阶梯式训练”、“实践化教学”等独具特色的设计教学模式, 能够让读者在演练一系列教学案例的同时, 由浅入深地掌握 Grasshopper 参数化设计思维和方法。

书中涉及 30 余个案例, 讲解生动细致; 内容层层递进, 可用于建筑设计师、建筑学相关专业学生及参数化建筑设计爱好者在各个阶段的学习和提高。

责任编辑: 尤凯曦 李阳 李明

责任校对: 李欣慰 焦乐

Grasshopper 参数化设计教程

祁鹏远 (Skywoolf) 编著

*

中国建筑工业出版社出版、发行 (北京海淀三里河路9号)

各地新华书店、建筑书店经销

北京中科印刷有限公司印刷

*

开本: 787×1092毫米 1/16 印张: 17 $\frac{1}{2}$ 字数: 427千字

2017年7月第一版 2017年9月第三次印刷

定价: 49.00元 (含增值服务)

ISBN 978-7-112-20728-2

(30391)

版权所有 翻印必究

如有印装质量问题, 可寄本社退换

(邮政编码 100037)



序言

喜闻鹏远出版 Grasshopper 参数化方面的书籍，这对于设计师来说真是一件幸事！多年来设计师的模型修改都是一大难题，无论是建筑设计还是工业产品设计，在 Grasshopper 问世之前 Rhino 用户都面临着这样尴尬的问题。设计并不是一个简单的过程，N 多个设计信息的输入都需要设计师来同时协调处理，但由于传统的设计思路、手段与技术工具的局限，使得我们往往只能凭借经验主观性地输出为数不多的设计成果。尽管我们无比努力，但也可能会遗失一些未曾发现的优质设计输出。现在这些问题都会迎刃而解。Grasshopper 参数化设计工具可以帮我们无损地将所有的设计输入转换为设计输出，这样既可以基于程序算法和设计条件筛选出最具功能性与美观性的设计输出，也可以基于甲方的特殊要求得到个性十足的设计成果，充分展现了基于程序的参数化设计魅力。Grasshopper 这一革命性的辅助设计工具，必将会带来一场革命性的设计创新。

与鹏远初识于 2009 年的 Shaper3d 论坛和 NCF 参数化建筑联盟。我记得他在大学时代就非常热衷于 Grasshopper 参数化设计的学习与研究，与国内众多的 Grasshopper 参数化设计师一同创立了 NCF 参数化建筑论坛，共同学习、交流，共同进步。难能可贵的是鹏远与众多的研究性学者不一样，不限于纸上谈兵的参数化设计理论研究，而是将所有的研究成果付诸实践。他在从业期间，坚持将 Grasshopper 参数化设计工具应用于实际的建筑设计案例中，帮助设计团队提高创作效率，为甲方节省时间与成本。更多项目实践证明，其研究的成果是能经受得起检验的。百闻不如一见，我相信鹏远的这本书，能够为初涉参数化设计漫长之路的你提供一盏启明灯，为已经在这条路上的你提供更多的宝贵参考与意见。

Jessesn.Chen

McNeel Asia 技术支持

Shaper3d (西普设计咨询) 创始人

Rhino 中国技术支持与推广中心负责人

记得那是 2009 年的夏天，

我打开了这个工具的界面，

脑海里浮现出了崭新的世界……

前言

给参数一个定位

一直以来，我身边的很多人总会问我这样一个问题：究竟什么是参数化？每次我都会重新思考这个问题的答案，然后尝试用自己匮乏的语言去建立一个提问者和解惑者之间的信息桥梁，然后把自己的理解努力地描述给对方。久而久之我也发现，每次随着提问者的特征不同，我的解释并不一样：回答因人而异，这是我所选择的交流方式，因为我并不认为这个问题的答案对于每个人而言是统一的，它对于目的不同、创作思维迥异的设计师们而言，其意义也应该是不同的。时间长了，自己也会习惯性地去想：究竟什么是参数化？

创作，是从自我内心之中挖掘冥想的过程，每个人追求的答案不同，只能自己去寻找。别人的路可以作借鉴，但如果不坚定地走向自己内心的深处，是永远不可能到达彼岸的。

参数化是什么？我给予自己这样一个答案：它是一种借助计算机运算能力来解决客观设计问题的工作方法，我运用它解决自己在设计创作及实践过程中所遇到的各种问题；这些问题有的很简单，比如调整一条曲线的形态——我总习惯在 Grasshopper 里把这条曲线关联的各项关系空间使用性能的参数（包括空间面积、空间直径等）都用参数模型显示出来，然后一边调

整形态，一边观察各项指标的变化，以此来寻找一个美观和实用的平衡点；复杂的问题如在一片丘陵地形之中，用算法生成合理的行车路网，或是在一片自由形态的坡折屋面之中，通过视线和环境的因素来考虑和调控整个屋顶形态等。这些过程都很有意思，道理也易于理解，但是这种工作方法让设计变得与众不同。由设计方法的不同导致问题关注深度和角度的不同，也就有了不一样的设计结果。所以参数化也是通往创新型设计思维的一条通道，我一直是这么想的。当然除此之外，我也用它做过一些复杂构造节点的深化设计和施工定位，模型总是越精细就越容易控制施工，我们一直在很努力地促使自己的作品尽量保持原样地落地，所以高效率的建模工具是必不可少的。这些就是我理解的参数化，它是我的工作方法，是辅助我思维的一种工具，也是协助我解决设计难题的利器。

参数化设计当然不是玩造型，不是标新立异的英雄主义。因为选择如何去做的是设计师，是个人的心理需求，是社会的舆论关注。而参数化本身只是一种设计的思维和方法，它能提高我们的工作效率，拓宽我们的构思领域，它也是在当下每个还在思考创新的建筑师都应该去了解 and 尝试的一种设计模式。如果硬要拿那些“不可思议”的项目来描述“参数化”的话，可以这么讲：那些奇怪的形体，是设计师的创意，是个人情怀，和参数化思维关系不大，但是使这些复杂的形态能够进行合理地优化调整并能用科学的依据将其实现的设计方法和技术手段，是用参数化设计的方法做到的。我经常跟学生们讲：“这个造型、这个设计结果不叫参数化设计。用来推敲、分析、控制这个形体，让这个空间好用、合理的设计过程才算是参数化设计工作中的一部分。但是参数化设计本身可不是给那些玩形状的人收拾残局的，它能启发设计师去重新思考建筑的生成过程，对整个行业设计创作过程的启迪和改变才是它最大的核心价值。”借用一句马克思主义哲学里的话，因为它提高了生产力（设计的生产力可以被理解成创造性思维能力），所以它的发展和成熟是具有必然性的。

这里，给予参数化设计的新人几点提示：

1. 我们鼓励平时练习用算法控制复杂模型的体态，体味这个设计过程是一件很能激发设计师创作灵感的事情。但一定要清楚这仅是造型，而不是建筑。建筑设计还要综合考虑功能空间和视觉艺术表达等多方面的因素，到建造阶段更是错综复杂的社会科学。所以切记不要轻易地把自己的异形体量当作建筑设计的课程作业或者盲目地应用于项目实践。

2. 如果我们已经开始用参数化设计的手法来做方案或是课题了，那这个方案的形体就算再复杂也应该是被一个或多个设计要素（如功能需求）制约着的，形体仅是空间的外在表达，只有满足了空间的塑造逻辑和某些价值或意义需求的作品才有灵魂。尽量多体会推敲和解决问题的过程，因为这是对设计基本能力的一种训练和培养；尽量多地讲述和分享每一个设计过程，因为这样可以强化自我对设计方法的理解。

3. 如果我们对参数化还不是很了解，却正面对一些要用到这门技术的课题或项目，无论时间多紧，也请不要直接用前人的参数模型来满足自己的方案需求。因为每段生成逻辑都具有独特性，模仿只能做些表面工作，对课题来说很难搞清楚实质的东西，对项目来说更是会遗留下很多施工的疑难问题。参数化并不复杂，我们可以找个集中的时间静下心来从头开始梳理，也只有这样才能最快地实现我们理想的设计追求和目的。

以上是 2013 年底的笔记，一字不差地诠释着我今天的理解。如今的参数化设计已经被大家所熟知，越来越多的应用和作品已如雨后春笋般呈现，年轻且富有创意的思维在国内外竞标中大放异彩。然而我们并不会停止前进的脚步，还有很多门没有打开，还有很多陋习没有摒弃。为了更理想的空间，为了潮流褪色之后化作经典，我们不忘初心、继续前行，期待有您相伴……

目录

Contents

v	前言	给参数一个定位		
1	参数化“六艺”			
6	工具概述	Tool Overview		
15	准备开始 ...	Let's go!		
16	Part A 生成思维	Generative Thinking		
18	Level 1		46	Level 4
28	Level 2		54	Level 5
36	Level 3		64	Level 6
66	Part B 数据结构	Data Structure		
68	Level 1		90	Level 4
74	Level 2		98	Level 5
82	Level 3		106	Level 6
108	Part C 曲面设计	Surface Design		
110	Level 1		134	Level 4
118	Level 2		142	Level 5
126	Level 3		150	Level 6
152	Part D 逻辑优化	Logic Optimization		
154	Level 1		174	Level 4
162	Level 2		182	Level 5
168	Level 3		190	Level 6
192	Part E 工程接轨	Project Integration		
194	Level 1		220	Level 4
204	Level 2		228	Level 5
214	Level 3		236	Level 6
238	Part F 算法研发	Algorithm Development		
240	Level 1		258	Level 4
246	Level 2		264	Level 5
252	Level 3		270	Level 6
272	后记			

参数化

“六艺”

“养国子以道，乃教之六艺：一曰五礼，二曰六乐，三曰五射，四曰五御，五曰六书，六曰九数。”

——《周礼·保氏》

决定创作结果成败的，往往不是设计者的某种技能是否出色，而是他是否拥有一套完善且闭合的技能体系。

参数化“六艺”从构思到营造，伴你一起走过回归设计的旅程……

浅谈参数设计的技能体系

多年以来, Grasshopper 的学习总会给我们带来一些困扰。初学者在认识这个工具后往往不知道该如何着手学习, 也很难再继续提升自己, 以至于普遍地产生了一段长时间的迷失。虽然在这个时期大家还会陆续地认识新的运算器, 接触新的算法案例, 但不同的新老知识和技能之间总是让人感觉很混乱, 难以结合成体系, 也就更难将其应用于实处。漫漫“编程”路, 踉跄地走过了五十, 又何时能到达一百……这一路的艰辛, 不是因为我们笨, 也不是因为这些技能有多难, 其实只是因为我们缺少了一套系统的知识体系来扶持我们的思考, 才导致了不必要的窘境。

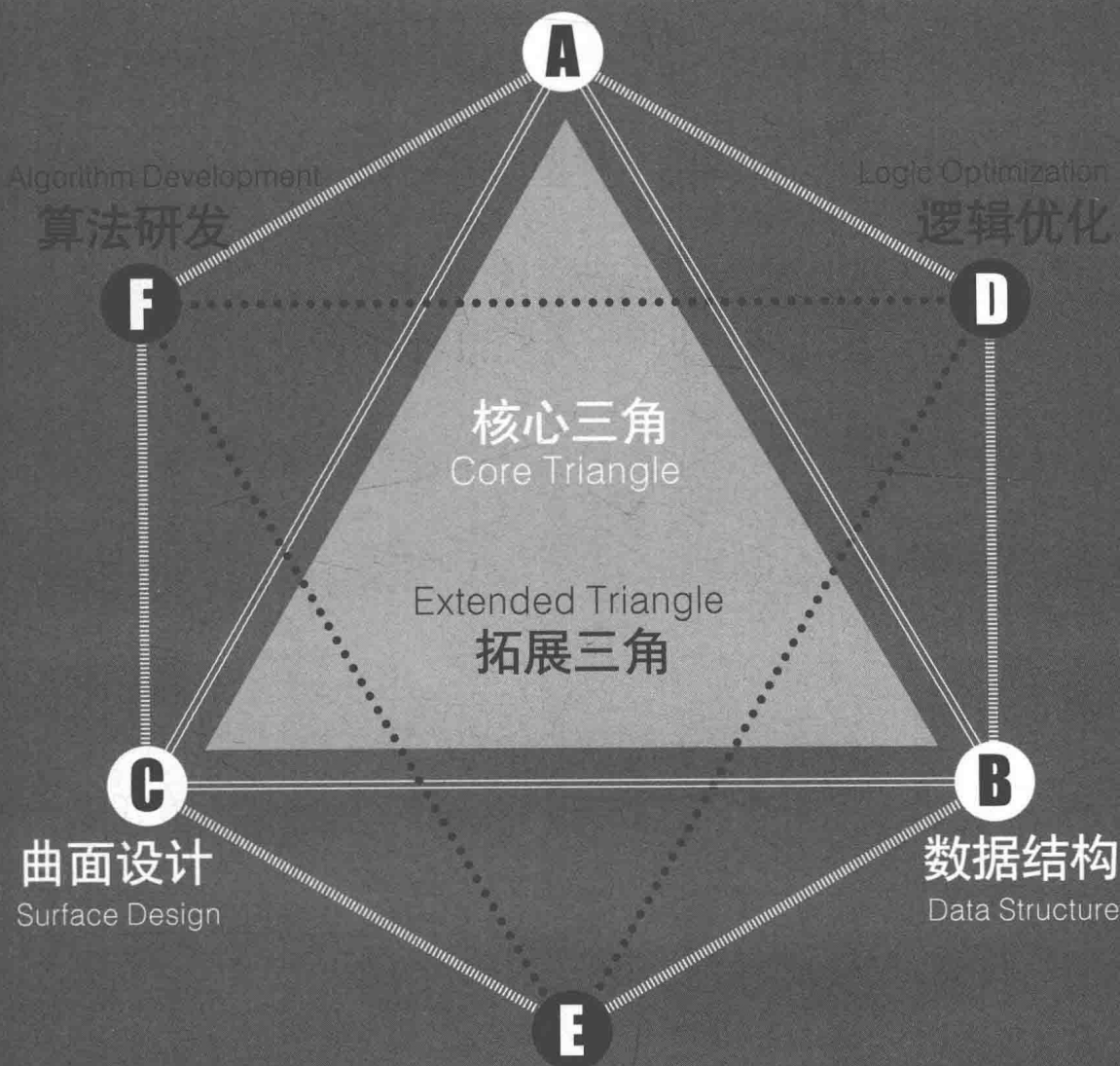
“六艺”的提出, 正是为了引导我们获取一套完整的技能体系, 它既是我们构成参数化逻辑思维的六种基本技能, 同时也是挖掘我们自身思维潜能的一套知识结构图谱。有了它, 我们就可以在此基础上逐渐地建立起一套属于自己的核心逻辑体系。每当我们再获取其他新的技能算法, 我们便可以将其归纳到这个逻辑体系当中, 并通过体系中的知识关系将其与已有的信息形成比对、反思, 从更多的维度对这些新的技能进行消化、分解、融合。只有这样, 我们学到的东西才能更有机地我们的技能储备结合在一起, 也只有这样才能真正地将我们所学之技应用于实践。

本书以“六艺”为核心框架, 每个章节独立为一类技能的分支, 每章案例练习由浅入深分成多个等级, 其中 Level1~Level3 为基础章节, 一般零基础的新人均可以独立完成并有所收获, Level4~Level5 往往是为了帮大家更深入地形成思维体系而进行的多分支知识点结合性训练。所以对于初学者而言, 建议最初先将白色核心三角的技能完成到 Level3, 然后再平行推进 6 个体系。对于已经熟悉了本软件的朋友, 可以自行在任意分支上通过案例默写来给自己定级。然后有针对性地训练自己的短板, 加深各分支之间的技能结合和知识理解。最后的 Level6 是技能回归创作的一种境界, 需要大家认真体会。无论我们的专业技能有多强, 其初衷都只有一个, 就是拿出优秀的作品, 所以回归设计就是技能最好的闭合。

基础

Foundation

生成思维 Generative Thinking



曲面设计
Surface Design

数据结构
Data Structure

Project Integration 工程接轨

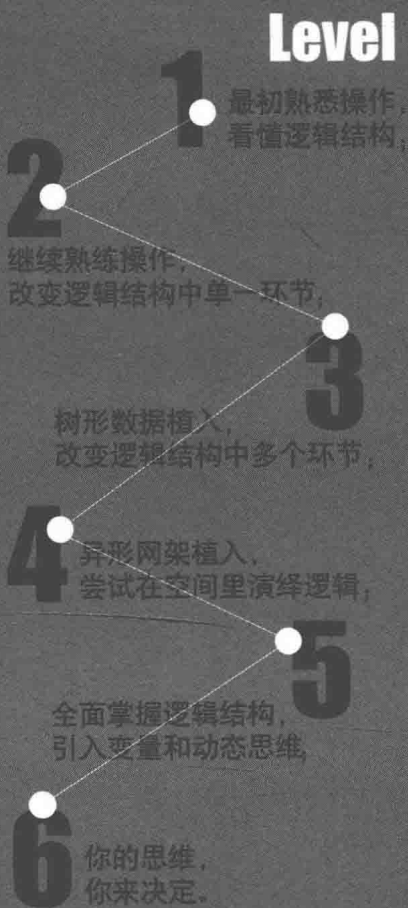
Practice

实践

Part A

生成思维

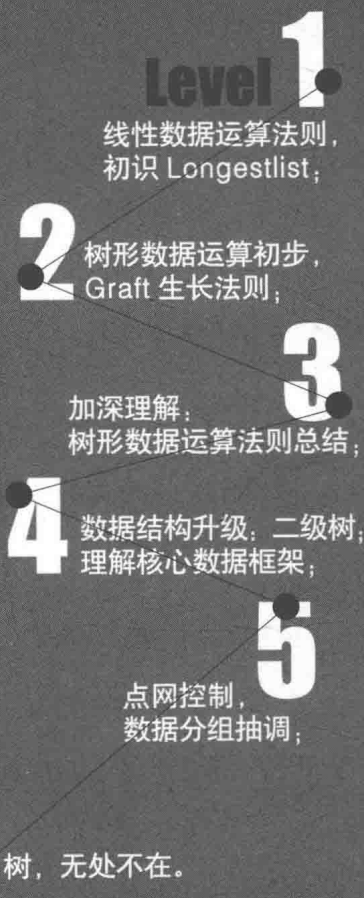
Generative Thinking



Part B

数据结构

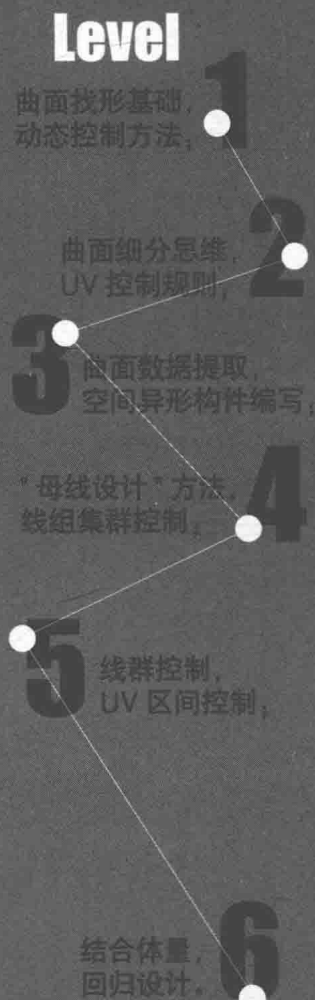
Data Structure



Part C

曲面设计

Surface Design



Part E

工程接轨

Project Integration

Level

实践体系认知，
结构一级框架；

1

2

结构二级框架，
构造级节点建模；

3

算法细节控制，
Voronoi2D 构件定位；

4

曲面空间网格优化，
定位点坐标输出；

5

幕墙细分深化，
工程实例模拟；

6

项目接轨，
立足实践。

Part F

算法研发

Algorithm Development

Level 1

三角函数曲面生成，
函数模块运算演示；

2

遗传算法应用，
代数模块运算演示；

3

迭代算法演示，
L-系统生成思维；

4

迭代算法应用，
几何模块运算演示；

5

人工智能初步，
计算机初级逻辑基础；

6

人机合一，
天下无敌！

Part D

逻辑优化

Logic Optimization

Level 1

理解数据联动的意义，
让变量更加清晰；

2

了解逻辑优化原则，
减轻计算机运算负荷；

3

合理利用树形数据，
避免重复性操作；

4

“最小单元”思维，
挖掘更简洁的生成逻辑；

5

清理重复数据，
获得最干净的结论；

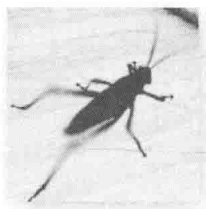
6

逻辑打包，
封存你的能力！

Tool Overview

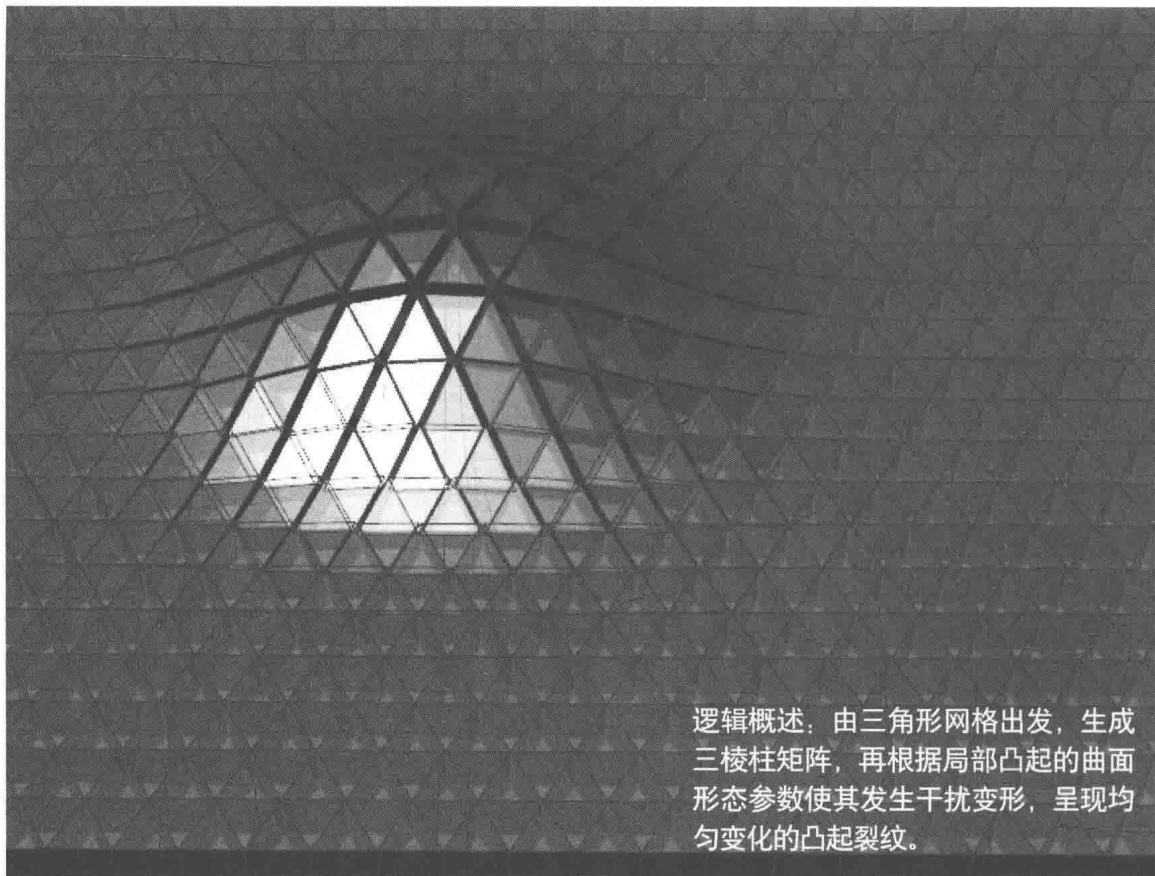
工具概述

Grasshopper (简称 GH) 是 Rhino 平台上一款很特别的建模插件。其建模方式与传统建模软件相比有很多明显的不同。最主要的特点是 GH 可通过一系列模块化的建模指令(运算器)来搭建起一个模型完整的生成逻辑,并通过计算机运算执行这些指令来生成最终的模型。



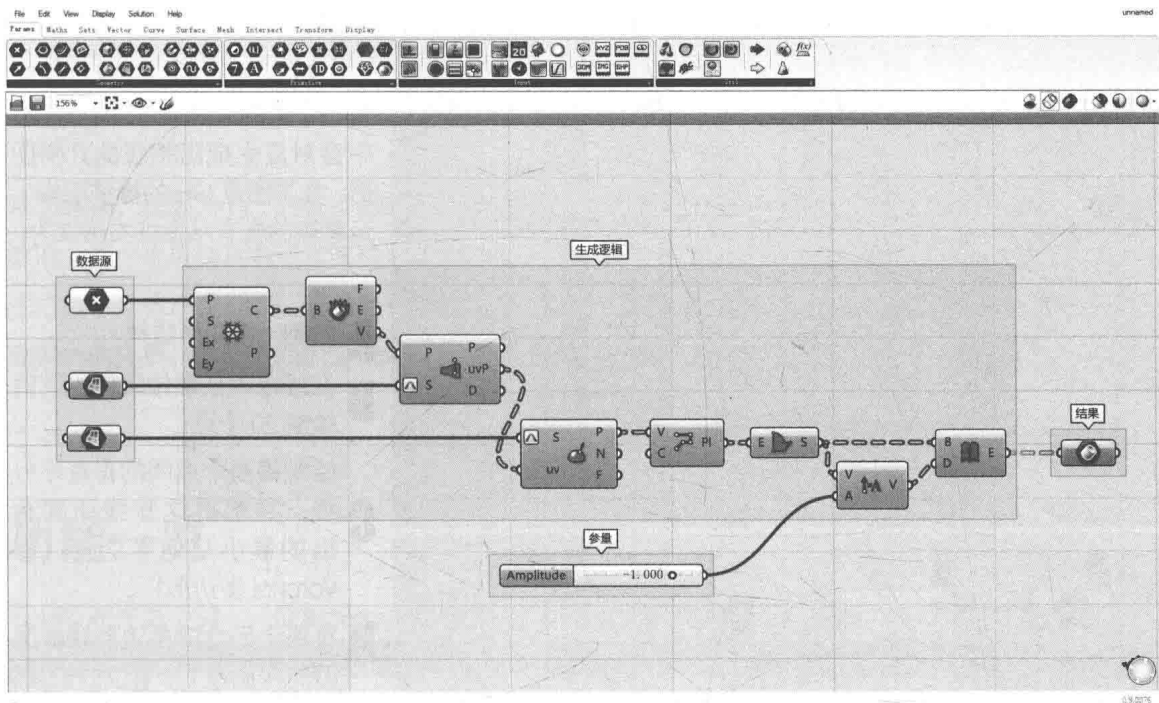
0.9.0076

在整个建模过程中, GH 扮演着编写生成逻辑的角色, Rhino 作为一个运算平台将生成的模型直观地、即时地展示给我们。那下面我们就来了解一下 GH 建模的案例。



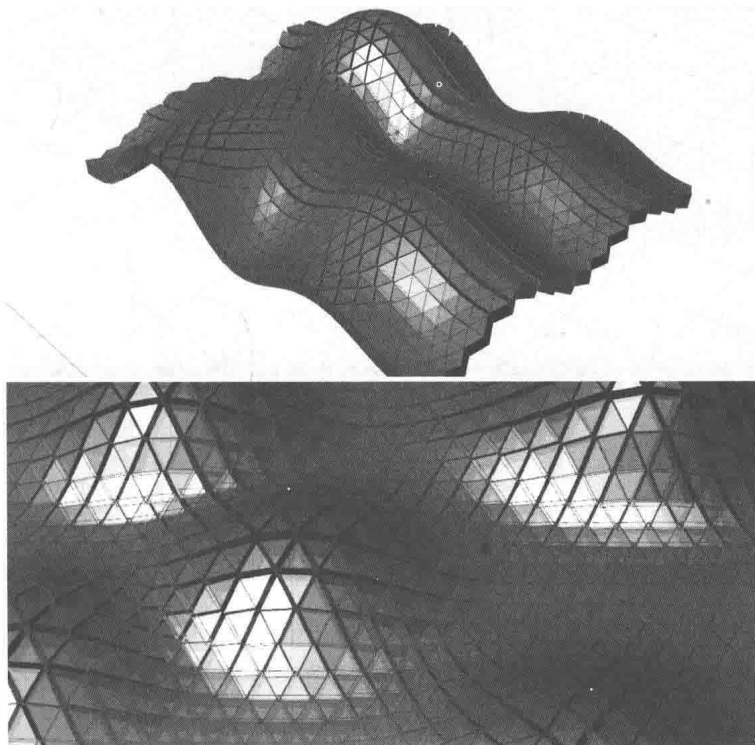
逻辑概述: 由三角形网格出发, 生成三棱柱矩阵, 再根据局部凸起的曲面形态参数使其发生干扰变形, 呈现均匀变化的凸起裂纹。

操作界面



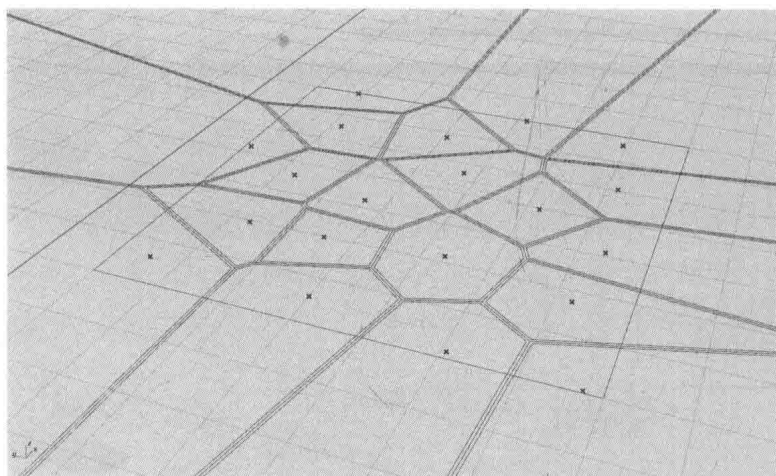
GH 界面用来编辑每段生成算法（以运算器形式显示）和它们之间的数据关系（以运算器之间的连线模式显示），最终构架出一套完整的模型生成逻辑算法。

Rhino 界面作为模型的观察窗口，可即时地显示 GH 编写的生成逻辑所运算出的结果。



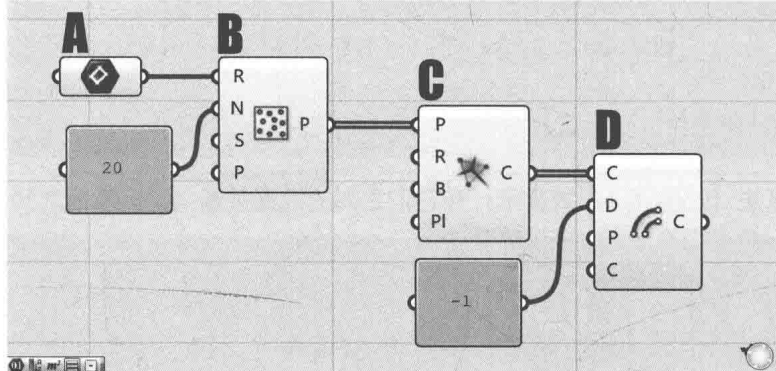
由于 GH 编写的成果不是模型本身，而是整个模型的生成过程，所以稍微改变生成逻辑的一些参数，即可改变模型最终的运算结果。图示范例改变了算法最初的基准面形态，相应的深化建模不再需要人工修改，由 GH 逻辑直接生成。

建模流程



每当我们在 GH 窗口中编辑一个新的运算器，Rhino 平台中会对应生成出相应的几何图形。左下图是 GH 的操作面板，运算器的含义从左往右依次为：

- A** 绘制一个矩形线框；
- B** 在线框内任意位置随机同时绘制 20 个点；
- C** 绘制每两个点间的垂直平分线，并求得这些线所围合出的最小几何多边形（即 Voronoi 多边形）；
- D** 将每个闭合的多边形线框同时向内偏移 1 个单位的长度。



以上我们可以了解 GH 的建模操作过程，不难看出，像 C 步骤这样一个复杂的运算指令，在 GH 中一步即可实现。

建模特性

即时可见：这种功能与传统建模并无区别，但在编程软件领域里却是一种革新。普遍的编程建模需要设计师自行编写代码，在代码编写完整前，这些指令并没有建模能力。所以每次需要编写、校对，然后试运行。这种类似程序开发的高技术的工作一直以来成为编程语言建模无法靠近大众设计师创意工作的一层壁垒。但 GH 的操作解决了这个难题，每个运算器都是一组完善的打包代码，它们既具有灵活组合的能力，同时又拥有独立的可视化功能。这使得设计师可以在轻松编辑它们的同时看到运算的结果，适用于方案的推导和辅助。

逻辑建模：GH 编写的不仅是一个模型的结果，而是模型如何生成的一个过程。在整个设计过程中，设计师可以通过一段逻辑生成很多个符合逻辑条件的结果，比如上图的随机点，其实可以有无数种