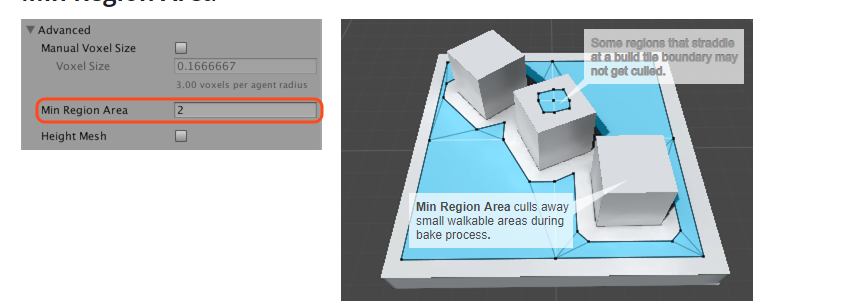
**高级导航网格烘焙设置**

**Min Region Area**



借助 *Min Region Area* 高级构建设置，可剔除未连接的小型导航网格区域。表面积小于指定值的导航网格区域将被移除。

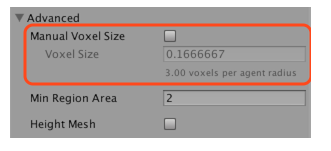
请注意，尽管有 *Min Region Area* 设置，可能也无法删除某些区域。导航网格是以区块网格形式并行构建而成的。如果某个区域跨越区块边界，则不会移除该区域。此情况的原因是区域修剪发生在构建过程中无法访问周围区块的阶段。

**体素大小 (Voxel Size)**

通过 Manual Voxel Size 设置可更改烘焙过程的准确度。

导航网格烘焙过程使用体素化从任意级别几何体构建导航网格。在算法的第一轮中，场景被光栅化为体素，然后提取可行走表面，最后将可行走表面变成导航网格。体素大小描述了生成的导航网格表示场景几何体的准确程度。

默认的精度设置为每个代理半径为 3 个体素，即整个代理宽度为 6 个体素。此设置可在精确度和烘焙速度之间实现良好折衷。体素大小减半将使内存使用量增加 4 倍，构建场景需要的时间也为 4 倍。



通常情况下不应调整体素大小，但有两种情况可能需要这样做：构建\_更小的代理半径\_或\_更准确的\_导航网格。

**更小的代理半径**

针对人为减小的代理半径进行烘焙时，导航网格烘焙系统也会减小体素大小。如果其他代理尺寸保持不变，可能没有必要增加导航网格构建分辨率。

最简单的方法如下：

1.将 *Agent Radius* 设置为实际代理半径。 2.启用 *Manual Voxel Size\_，这样将采用当前的体素大小并“将其冻结”。 3.设置人为减小的*Agent Radius\_，因为您已经选中了 \_Manual Voxel Size\_，体素大小不会改变。

**更准确的导航网格**

如果您的关卡有很多密集点，您可能希望通过减小体素来提高准确度。Voxel Size 下的标签显示了体素大小和代理半径之间的关系。较好的范围介于 2–8 之间，好于通常会导致构建时间大幅延长的设置。

当您有意在游戏中构建紧凑的走廊时，请注意，除了代理半径之外还应留出至少 *4 \* voxelSize* 的空隙，特别是走廊成角度的情况下。

如果需要更小的走廊且超出了导航网格烘焙能够支持的范围，请考虑使用网格外链接。这些链接具有额外的好处，您可以检测它们何时被使用，并可以播放特定动画等。