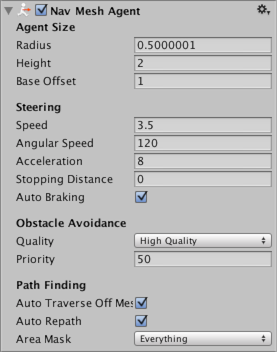
# 导航网格代理 (NavMesh Agent)

NavMeshAgent 组件可帮助您创建在朝目标移动时能够彼此避开的角色。代理 (Agent) 使用导航网格来推断游戏世界，并知道如何避开彼此以及其他移动障碍物。寻路和空间推断是使用导航网格代理的脚本 API 进行处理的。

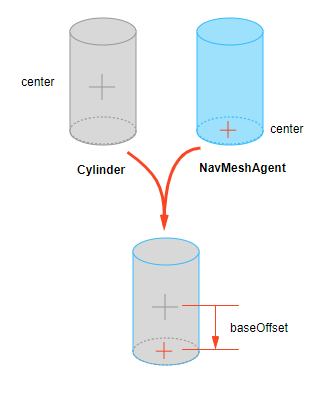


## 属性

| 属性 | 功能 |
| --- | --- |
| Agent Size | |
| **Radius** | 代理的半径，用于计算障碍物与其他代理之间的碰撞。 |
| **Height** | 代理通过头顶障碍物时所需的高度间隙。 |
| **Base offset** | 碰撞圆柱体相对于变换轴心点的偏移。 |
| Steering | |
| **Speed** | 最大移动速度（以世界单位/秒表示）。 |
| **Angular Speed** | 最大旋转速度（度/秒）。 |
| **Acceleration** | 最大加速度（以世界单位/平方秒表示）。 |
| **Stopping distance** | 当靠近目标位置的距离达到此值时，代理将停止。 |
| **Auto Braking** | 启用此属性后，代理在到达目标时将减速。对于巡逻等行为（这种情况下，代理应在多个点之间平滑移动）应禁用此属性 |
| Obstacle Avoidance | |
| **Quality** | 障碍躲避质量。如果拥有大量代理，则可以通过降低障碍躲避质量来节省 CPU 时间。如果将躲避设置为无，则只会解析碰撞，而不会尝试主动躲避其他代理和障碍物。 |
| **Priority** | 执行避障时，此代理将忽略优先级较低的代理。该值应在 0–99 范围内，其中较低的数字表示较高的优先级。 |
| Path Finding | |
| **Auto Traverse OffMesh Link** | 设置为 true 可自动遍历网格外链接 (Off-Mesh Link)。如果要使用动画或某种特定方式遍历网格外链接，则应关闭此功能。 |
| **Auto Repath** | 启用此属性后，代理将在到达部分路径末尾时尝试再次寻路。当没有到达目标的路径时，将生成一条部分路径通向与目标最近的可达位置。 |
| **Area Mask** | Area Mask 描述了代理在寻路时将考虑的[区域类型](file:///E:\UnityDocumentation\Manual\nav-AreasAndCosts.html)。在准备网格进行导航网格烘焙时，可设置每个网格区域类型。例如，可将楼梯标记为特殊区域类型，并禁止某些角色类型使用楼梯。 |

## 详细信息

代理由直立圆柱体定义，而该圆柱体的大小由 Radius 和 Height 属性指定。圆柱体随对象移动，但即使对象本身旋转也始终保持直立。圆柱体的形状用于检测并响应其他代理和障碍物之间的碰撞。当游戏对象的锚点不在圆柱体的底部时，可使用 Base Offset 属性来获取高度差。



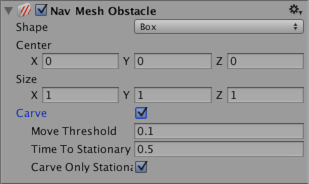
圆柱体的高度和半径实际上在\_两个\_不同的位置指定：[导航网格烘焙设置](file:///E:\\UnityDocumentation\\Manual\\nav-BuildingNavMesh.html)和各代理的属性。

* \_导航网格烘焙设置\_描述了所有导航网格代理如何碰撞或避开静态世界几何体。为了保持内存在预算范围内和 CPU 负载在控制范围内，只能在烘焙设置中指定一种大小。
* \_导航网格代理属性\_值描述代理如何与移动障碍物及其他代理碰撞。

通常在这两个位置设置相同的代理大小。但在某些情况下，例如，一个沉重的士兵可能有更大的半径，所以其他代理会在他周围保持更大间隔空间，但在其他情况下他避开环境的方式是相同的。

# 导航网格障碍物 (Nav Mesh Obstacle)

**导航网格障碍物 (Nav Mesh Obstacle)** 组件允许您描述[导航网格代理](file:///E:\UnityDocumentation\Manual\class-NavMeshAgent.html)在世界中导航时应避开的移动障碍物（例如，由物理系统控制的木桶或板条箱）。当障碍物正在移动时，导航网格代理会尽力避开它。当障碍物静止时，它会在导航网格中雕刻一个孔。导航网格代理随后将改变它们的路径以绕过障碍物，或者如果障碍物导致路径被完全阻挡，则寻找其他不同路线。



| **属性** | **功能** |
| --- | --- |
| **Shape** | 障碍物几何体的形状。选择最适合对象形状的选项。 |
| **Box** |  |
| Center | 盒体的中心（相对于变换位置）。 |
| Size | 盒体的大小。 |
| **Capsule** |  |
| Center | 胶囊体的中心（相对于变换位置）。 |
| Radius | 胶囊体的半径。 |
| Height | 胶囊体的高度。 |
| **Carve** | 勾选 Carve 复选框后，导航网格障碍物会在导航网格中创建一个孔。 |
| Move Threshold | 当导航网格障碍物的移动距离超过 Move Threshold 设置的值时，Unity 会将其视为移动状态。使用此属性可设置该阈值距离来更新移动的雕孔。 |
| Time To Stationary | 将障碍物视为静止状态所需等候的时间（以秒为单位）。 |
| Carve Only Stationary | 启用此属性后，只有在静止状态时才会雕刻障碍物。请参阅下面的[移动的导航网格障碍物的逻辑](file:///E:\UnityDocumentation\Manual\class-NavMeshObstacle.html#LogicMovingObstacles)以了解更多信息。 |

## 详细信息

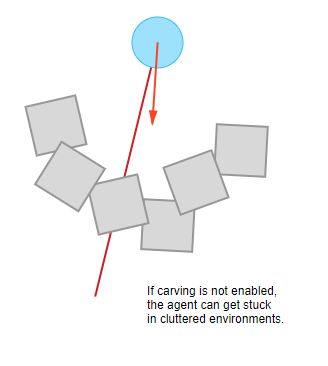
导航网格障碍物可通过两种方式影响导航网格代理在游戏中的导航：

### 障碍

未启用 **Carve** 时，导航网格障碍物的默认行为类似于碰撞体的行为。导航网格代理会尝试避免与导航网格障碍物的碰撞，当靠近时，它们会与导航网格障碍物碰撞。障碍躲避行为是非常基本的，具有一条短半径。因此，导航网格代理可能无法在导航网格障碍物很混乱的环境中找到方向。此模式最适合用于障碍物不断移动的情况（例如，车辆或玩家角色）。

### 雕刻

启用 **Carve** 时，障碍物处于静止状态时将在导航网格中雕刻一个孔。移动时，障碍物即为障碍物。在导航网格中雕刻一个孔后，寻路器 (pathfinder) 能够让导航网格代理绕过雕有障碍物的位置周围，或者如果当前路径被障碍物阻挡，则寻找另一条路线。对于通常会阻碍导航但可被玩家或其他游戏事件（如爆炸）移动的导航网格障碍物（例如板条箱或木桶），最好为其开启雕刻功能。



## 移动的导航网格障碍物的逻辑

当导航网格障碍物的移动距离超过 **Carve** > **Move Threshold** 设置的值时，Unity 会将其视为移动状态。当导航网格障碍物移动时，雕刻的孔也会移动。但是，为了减少 CPU 开销，只在必要时才重新计算该孔。此计算的结果将在下一帧更新中可用。重新计算逻辑有两个选项：

* 仅当导航网格障碍物静止时才进行雕刻
* 导航网格障碍物移动后进行雕刻

### 仅当导航网格障碍物静止时才进行雕刻

这是默认行为。要启用此选项，请勾选导航网格障碍物组件的 **Carve Only Stationary** 复选框。在此模式下，当导航网格障碍物移动时，雕刻的孔将被移除。当导航网格障碍物已停止移动并且静止时间超过 **Carving Time To Stationary** 设置的值时，障碍物将被视为静止状态，并再次更新雕刻的孔。当导航网格障碍物为移动状态时，导航网格代理会使用碰撞躲避功能避开它，但不会在它周围规划路径。

**Carve Only Stationary** 通常是性能方面的最佳选择，当与导航网格障碍物相关联的游戏对象由物理系统控制时，是很适合的选项。

### 导航网格障碍物移动后进行雕刻

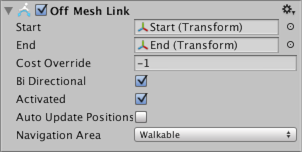
要启用此模式，请取消勾选导航网格障碍物组件的 **Carve Only Stationary** 复选框。取消勾选此复选框的情况下，当障碍物移动的距离超过 **Carving Move Threshold** 设置的值时，雕刻的孔会更新。此模式适用于大型缓慢移动的障碍物（例如，步兵避开的坦克）。

**注意**：使用导航网格查询方法时，应考虑到更改导航网格障碍物与该更改对导航网格生效之间存在一帧延迟。

# 网格外链接 (Off-Mesh Link)

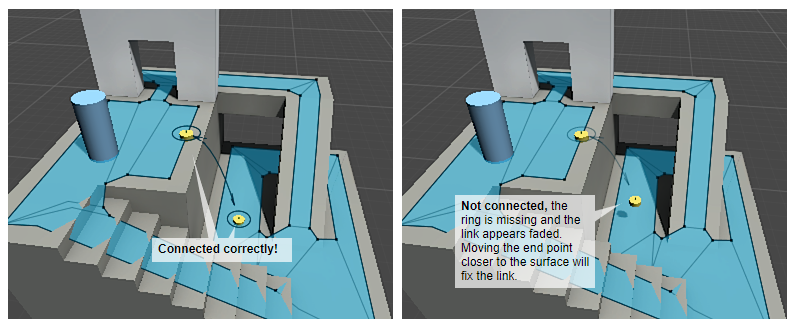
OffMeshLink 组件允许您合并无法使用可行走表面来表示的导航捷径。例如，跳过沟渠或围栏，或在通过门之前打开门，全都可以描述为网格外链接。

## 属性



| 属性 | 功能 |
| --- | --- |
| **Start** | 描述网格外链接起始位置的对象。 |
| **End** | 描述网格外链接起始位置的对象。 |
| **Cost Override** | 如果值为正，则在计算处理路径请求的路径成本时使用该值。否则，使用默认成本（此游戏对象所属区域的成本）。如果 Cost Override 设置为值 3.0，则在网格外链接上移动的成本将是在默认导航网格区域上移动相同距离的成本的三倍。如果希望让代理通常优先选择步行，但当步行距离明显更长时使用网格外链接，则 Cost Override 设置将变得有用。 |
| **Bi-Directional** | 如果启用此属性，则可以在任一方向上遍历链接。否则，只能按照从 Start 到 End 的方向遍历链接。 |
| **Activated** | 指定寻路器 (pathfinder) 是否将使用此链接（如果将此属性设置为 false，则将忽略它）。 |
| **Auto Update Positions** | 如果启用此属性，当端点移动时，网格外链接将重新连接到导航网格。如果禁用，即使移动了端点，链接也将保持在其起始位置。 |
| **Navigation Area** | 描述链接的[导航区域类型](file:///E:\UnityDocumentation\Manual\nav-AreasAndCosts.html)。该区域类型允许您对相似区域类型应用常见的遍历成本，并防止某些角色根据代理的区域遮罩 (Area Mask) 访问网格外链接。 |

## 详细信息



如果代理未遍历网格外链接，请确保两个端点都已正确连接。正确连接的端点应在接入点周围显示一个圆圈。

另一个常见原因是导航网格代理 (Navmesh Agent) 的 Area Mask 没有包含网格外链接的区域。