

移动

功能

本节教大家能够让 Pepper 通过代码实现来移动到某一个位置。经常用于让 Pepper 试图找到一个路径，并达到目标点的位置。

请记住：Pepper 机器人可以躲避障碍物，并安全的到达目标点。

基本的 QiSDK API 调用接口如下：

```
// 获取目标点的针率
Frame targetFrame = ...;

// 创建 GoTo 行为
GoTo goTo = GoToBuilder.with(qiContext)
                        .withFrame(targetFrame)
                        .build();

// 异步执行 GoTo 行为
goTo.run();
```

预备知识

什么是 Frame?

Frame 是 QiSDK API 中的概念，一般会在 Pepper 机器人室内导航中使用。Frame 是用于记录导航中目标点的位置信息，此信息一般包括机器人所在位置的 x, y, z 的三维空间值。

什么是 Transform?

Transform 也是 QiSDK API 中的概念，它代表中一个几何变化，它包括一个 Vector3 和一个 Quaternion。初看上去，大家可能不太理解，这里面设计到一些高阶数学上的知识。

为了更好理解它在应用的使用，这里给大家一个简单的介绍：在机器人的几何空间中，它包含了 x, y, z 三维空间信息，同时还有一个旋转的信息，那么这个信息是由一个四元组（Quaternion）来表示，至于为什么使用四元数来表示旋转，[请参考这里](#)。我们完全可以类比 Excel 中的 Offset 功能，它的作用是定义了一个偏移量，不相对于任何参照物。

什么是 Mapping?

Mapping 是 Pepper 提供的本地地图服务。

案例

场景：让 Pepper 移动到一个指定的目标点

让 Pepper 移动到你前方的一米位置处。实例如下：

```
Actuation actuation = qiContext.getActuation();

Frame robotFrame = actuation.robotFrame();

// 定义一个沿 X 轴方向的 1 米的偏移量。
Transform transform = TransformBuilder.create().fromXTranslation(1);

Mapping mapping = qiContext.getMapping();

FreeFrame targetFrame = mapping.makeFreeFrame();

// 0L 是指时间戳，可以指定时间来更新 FreeFrame。
targetFrame.update(robotFrame, transform, 0L);

GoTo goTo = GoToBuilder.with(qiContext)
                        .withFrame(targetFrame.frame())
                        .build();

goTo.run();
```

性能与局限性

避障

Pepper 机器人能够在移动过程中躲避障碍物。意味着它为了达到目标点的路线中，遇到障碍物，它会选择合适的路径绕开障碍物，并达到目标点。

异常处理

特定的环境下可能干扰机器人达到目标的准确性，并引起 GoTo 执行错误，比如：

- * 灯光会影响到传感器。
- * 障碍物不能被识别到。
- * 粗糙的地面。
- * 临时障碍物的干预，比如人或者动物。

选择 GoTo 还是 Animate?

GoTo 和 Animate 都可以让 Pepper 机器人移动，但如何选择使用那种方式：

- * 抛除避障功能外，Animate 适合于不变的轨道场景。
- * GoTo 适用于在到达目标点的路径是随机变动的。

参考

如果要学习更多有关 GoTo 类的使用，请参考：

- 如何使用并理解 [GoTo 的使用](#)
- GoTo 的 [javadoc](#)