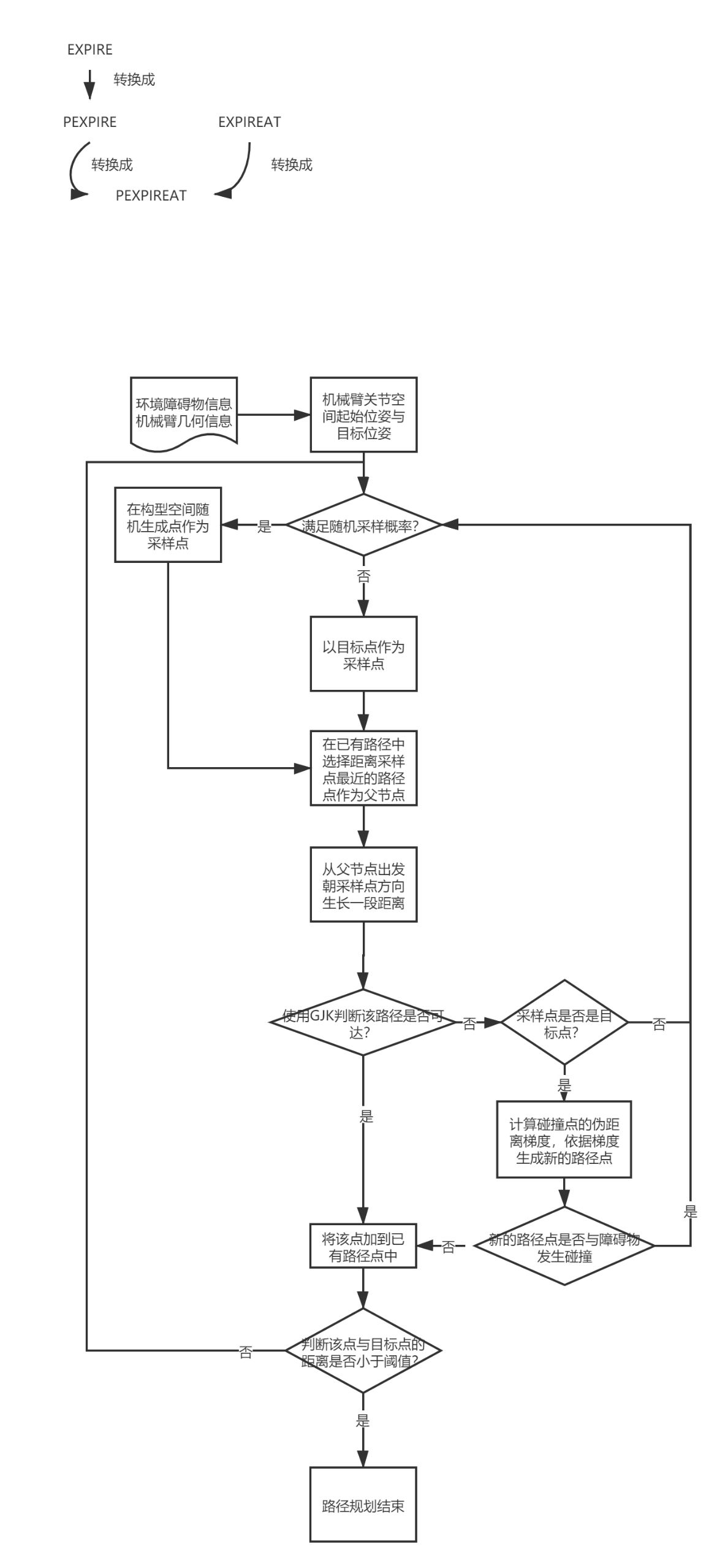
**说 明 书 摘 要**

本发明提供了一种基于伪距离函数的工业机器人路径规划方法。使用凸多面体对机械臂连杆以及工作环境障碍物进行包络，将机械臂的几何形状以及环境障碍物以凸多面体顶点的形式记录下来。以传统RRT（Rapidly-exploring random trees）算法为基础在机器人的关节空间进行路径规划。首先确定路径的起点与终点，然后以一定的概率在关节空间随机取点作为采样点或者将终点作为采样点，在原有的路径点中选择与采样点最近的一点为父节点，并朝采样点方向生长一段距离生成新的路径点；将父节点与新路径点之间的路径离散化，利用机械臂的运动学方程将关节空间的关节参数映射成工作空间几何外形，使用GJK算法[1]判断机械臂是否与环境障碍物发生碰撞，如果发生碰撞则计算碰撞点的伪距离[2]对各关节参数的梯度，将碰撞点沿梯度方向做出修正作为新的路径点。该算法与传统RRT相比生成的路径更优，鲁棒性更好，在障碍物较多或存在狭窄通道的环境下优势更加明显。

**摘 要 附 图**



**权 利 要 求 书**

1. 一种基于伪距离的RRT路径规划方法，包括如下步骤：

步骤1：用凸多面体对障碍物以及机械臂进行包络；

步骤2：输入起始点关节位形以及目标点关节位形；

步骤3：以一定概率生成随机点或者以目标点作为采样点，选择已有路径点中与采样点最近的点作为父节点沿采样点方向生长一段距离产生新的路径点；

步骤4：把父节点与新路径点之间的路径离散化，通过机械臂的运动学方程把机械臂关节空间的关节参数映射成机械臂工作空间的几何外形；使用GJK算法判断机械臂是否与环境障碍物发成了碰撞；如果没有发生碰撞则进入步骤6；否则进入步骤5。

步骤5：如果采样点是随机点，则放弃该点重新返回到步骤3；如果发生了碰撞且采样点是目标点，则在碰撞点计算伪距离对机械臂各个关节的梯度，将碰撞点沿着梯度方向进行修正；如果没有发生碰撞，将该点加入到已有路径中；

步骤6：判断新路径点与目标点的距离是否小于设定阈值，如果小于则路径生成完毕，否则返回步骤3；

2、根据权利要求1所述的基于伪距离函数的工业机器人路径规划方法，其特征在于使用凸多面体对障碍物以及机械臂进行包络，将机械臂的几何外形以及环境中障碍物以凸多面体顶点的形式记录下来。

3、根据权利要求1所述的基于伪距离函数的工业机器人路径规划方法，其特征在于步骤5中如果新路径点与父节点之间的路径发生碰撞，则通过伪距离计算发生碰撞的机械臂连杆与障碍物之间的浸入距离，并计算碰撞点伪距离关于机械臂各个关节参数的梯度，将碰撞点沿梯度方向做出调整生成新的路径点。

**说 明 书**

**一种基于伪距离函数的工业机器人路径规划方法**

**技术领域**

本发明涉及一种基于伪距离的工业机器人路径规划方法，用于在离线编程环境，给定机械臂起始、终止位姿以及环境障碍物下，在关节空间快速生成一系列位姿点，继而生成一条无碰撞路径。

**背景技术**

机器人离线编程是机器人在实际生产加工前的预演规划过程，与传统的手动示教相比，离线编程具有规划精度高、操作环境安全、不占用机器人等优点；规划一条使机械臂从位姿A移动到位姿B的无碰撞的路径是离线编程中最常见的功能之一。该应用场景下最常见的算法是RRT（Rapidly-exploring random trees）算法。RRT是一种通用的方法，不管什么机器人类型、不管自由度是多少、不管约束有多复杂都能使用，且实现原理简单。但其缺点也是非常明显的，首先RRT找到的路径只能满足可达性，并不能对路径质量做出保证，相同条件下两次RRT生成的路径差别可能很大。其次，对于复杂的障碍物环境（例如路径需要穿过狭长通道），在给定尝试次数下，RRT能找到可达路径的概率较低。

**发明内容**

针对现有技术的不足，本发明在传统RRT算法的基础上做出了改进。本发明的目的是提供一种适用于工业机械臂的快速、稳定的无碰撞路径生成算法。其特征在于，包括如下步骤：

步骤1：用凸多面体对障碍物以及机械臂进行包络；

步骤2：输入起始点关节位形以及目标点关节位形；

步骤3：以一定概率生成随机点或者以目标点作为采样点，选择已有路径点中与采样点最近的点作为父节点沿采样点方向生长一段距离产生新的路径点；

步骤4：把父节点与新路径点之间的路径离散化，通过机械臂的运动学方程把机械臂关节空间的关节参数映射成机械臂工作空间的几何外形；使用GJK算法判断机械臂是否与环境障碍物发成了碰撞；如果没有发生碰撞则进入步骤6；否则进入步骤5。

步骤5：如果采样点是随机点，则放弃该点重新返回到步骤3；如果发生了碰撞且采样点是目标点，则在碰撞点计算伪距离对机械臂各个关节的梯度，将碰撞点沿着梯度方向进行修正；如果没有发生碰撞，将该点加入到已有路径中；

步骤6：判断新路径点与目标点的距离是否小于设定阈值，如果小于则路径生成完毕，否则返回步骤3；

对于步骤1中所述的环境障碍物以及机械臂我们用如下记号进行表示：



其中，。

对于步骤4，我们定义父节点为，新节点为，在父节点与新节点之间生成一段路径：



将该路径离散化，得到一系列位姿点，其中，对这些位姿点使用机器人的运动学方程求得每个连杆的位置，继而确定机械臂每一个连杆的顶点集合，使用GJK算法判断对发生碰撞，则将该位形记录为，进入步骤5。

对于步骤5中所述伪距离是描述两凸集之间在一度规函数下的距离。我们定义是一个紧凸集，且，定义Q的度规函数为，可以把理解为一种伪范数，我们称之为Q范数，记为。很自然的，我们称为以原点为中心的球，很容易得证对于任意的，。假设A和B是上的两个紧凸集，那从A到B在Q范数下的伪距离定义为：



我们可以使用GJK算法判断凸集A与凸集B是否相交。

我们将A,B和Q分别由各自的顶点集的凸包表示，即，，其中分别是A和B凸包的顶点。的求解过程可以转化为求解线性规划的问题。如果



如果



其中由伪距离函数的Lipschitz连续性可知关于几乎处处可微。当时，设最优解为，其中设表征最优解中的正元素。定义。则的导数为：



其中；，其中除第k列为1外，其余列都为0；，其中



。

类似的，当时，的导数为：



针对步骤4中发生碰撞的姿态，我们求算，并确定：



。

我们按下式计算的导数：



易知上式可以理解为关于的梯度。通过的梯度，我们可以构造出关节空间的调整增量：



其中代表步长，这里我们取值为，对做出修正，然后再次用GJK算法判断是否存在碰撞，如果仍然存在碰撞，则放弃掉这个点；如果没有碰撞，则将该点作为新的路径点加入到已有路径点中。

**附图说明**

通过阅读参照以下附图对非限制性实施例所作的详细描述，本发明的其它特征、目的和优点将会变得更明显：

图1为本发明流程。

图2为在复杂障碍物环境下平面三连杆机械臂的路径规划效果。

图3为在复杂障碍物环境下工业机械臂UR10的路径规划效果。

**具体实施方式**

下面结合具体实施例对本发明进行详细说明。以下实施例将有助于本领域的技术人员进一步理解本发明，但不以任何形式限制本发明。应当指出的是，对本领域的普通技术人员来说，在不脱离本发明构思的前提下，还可以做出若干变形和改进。这些都属于本发明的保护范围。

如图1是本实施例的具体流程，结合图2所示实施例做出阐述。

步骤1：首先，用多边形对机械臂以及环境障碍物进行包络

障碍物：



机械臂连杆：



其中，，。

步骤2：设置机械臂关节位形的起始点和终点：



步骤3：初始化一个数组结构，在该结构中每一个元素由一个路径点,以及该路径点的父节点编号组成。设置随机采样概率0.3，然后随机在之间取一值，如果则在位形空间中随机取一点；否则。

在已有路径点中寻找，其中。从出发沿着方向移动一段距离生成一个新的路径点，其中,其中表示步长，在本例中我们设为0.2。

在父节点与新节点之间生成一段路径：



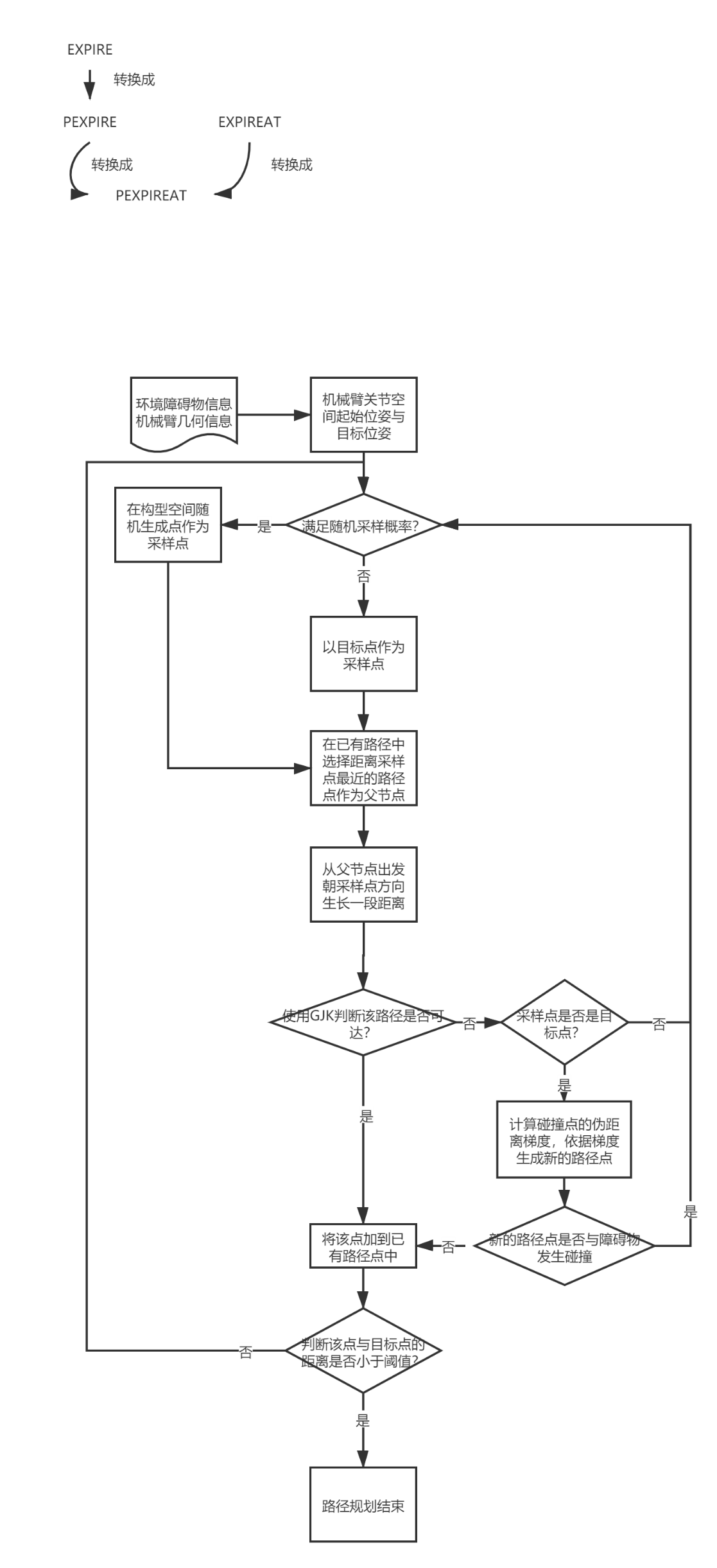
将该路径离散化，得到一系列位姿点，其中，对这些位姿点使用机器人的运动学方程求得每个连杆的位置，继而确定机械臂每一个连杆的顶点集合，使用GJK算法判断对是否发生碰撞，其中。如果没有发生碰撞，则将以及在中的编号作为一个新元素加入到中，进入步骤6；否则进入步骤5。

如果，则放弃掉该路径点，重新进入步骤3；如果，记录下路径上与环境障碍物发生碰撞的第一个路径点，并记录。计算在处的梯度然后得到关节位姿的修正值，其中。令，再次用GJK判断是否发生碰撞，如果存在碰撞进入步骤3；否则将该路径点加入到中，进入步骤6。

在步骤6中判断是否成立，如果成立则表示路径已生成，否则进入步骤3。

图3为本算法在六轴机械臂UR10上的应用，设置了一个起始点，一个终点以及一个中间点，规划出的无碰路径如图所示。

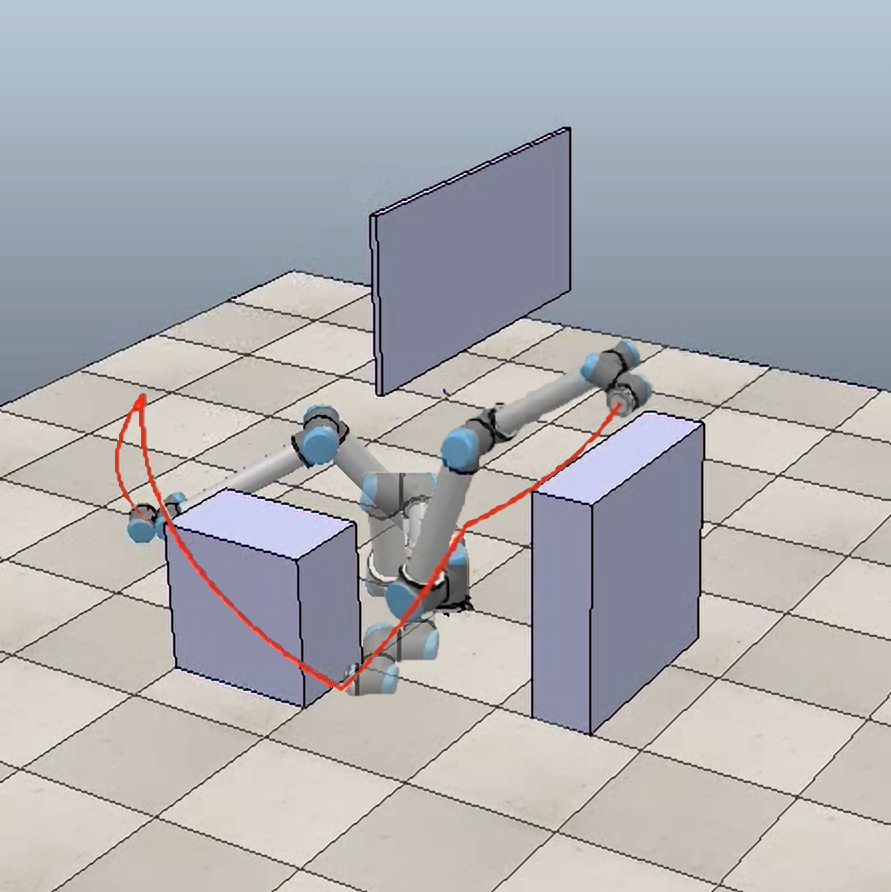
**说 明 书 附 图**



**图 1**



**图 2**



起始点

中间点

目标点

图 3

**参 考 文 献**

[1]. E. G. Gilbert, D. W. Johnson and S. S. Keerthi, "A fast procedure for computing the distance between complex objects in three-dimensional space," in IEEE Journal on Robotics and Automation, vol. 4, no. 2, pp. 193-203, April 1988, doi: 10.1109/56.2083.

[2]. Xiangyang Zhu, Han Ding and S. K. Tso, "A pseudodistance function and its applications," in IEEE Transactions on Robotics and Automation, vol. 20, no. 2, pp. 344-352, April 2004, doi: 10.1109/TRA.2004.824682.