

# 智能搜救机器人在障碍地形的自主构型规划\*

陈柏良, 黄开宏, 潘海南, 肖军浩, 吴文启, 卢惠民  
(国防科技大学 智能科学学院, 湖南 长沙 410073)

**摘要:**为了解决带有辅助摆臂的智能搜救机器人自动规划构型以实现自主越障的难题,提出一种能够适应复杂地面形状的搜救机器人越障构型规划新方法,其核心是一种高适应性、高效率的机器人姿态预测算法。通过将地形表示为离散的点集,建立了搜救机器人的单侧姿态预测数学模型;进一步提出了快速求解该问题的算法,每秒可预测1 000~1 500个姿态。基于此,设计了机器人越障过程中状态、动作的评价指标,运用动态规划算法与滚动优化思想构建了具有优化能力的、能够实时运行的构型规划器。仿真与实物实验的结果表明,该方法能够使机器人自主调整构型穿越复杂地形,且相较强化学习算法和人工操作具有更平稳的越障效果。

**关键词:** 搜救机器人;障碍地形;姿态预测;动作规划

**中图分类号:** TP242.6 **文献标志码:** A **开放科学(资源服务)标识码(OSID):**

**文章编号:** 1001-2486(2023)06-132-11



听语音  
与作者互动  
聊科研

## Autonomous configuration planning for intelligent search and rescue robots in rough terrains

CHEN Bailiang, HUANG Kaihong, PAN Hainan, XIAO Junhao, WU Wenqi, LU Huimin

(College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China)

**Abstract:** In order to solve the configuration planning problem for intelligent search and rescue robots with assisted flippers to achieve autonomous obstacle crossing, a novel method for planning robot configuration during obstacle crossing was proposed that can be applied to complex terrains. The core of the proposed method is an adaptable and efficient robot pose prediction algorithm. By representing the terrain as a series of discrete point sets, a mathematical model for predicting the one-sided pose of the tracked robot was established; further, a fast solver for this model was proposed, which can predict 1 000~1 500 poses per second. Based on this, the evaluation metrics of the robot's state and action in the obstacle-crossing process were established, and an optimization-based real-time flippers action planner was realized by using the dynamic programming algorithm and rolling optimization. The simulation and real-robot experiments show that the proposed approach enables the robot to control the flippers to cross rough terrains autonomously. It performs more smoothly than the reinforcement-learning method and manual operation when crossing obstacles.

**Keywords:** search and rescue robots; rough terrains; pose prediction; motion planning

智能搜救机器人是一种可变构型机器人,其利用摆臂结构或子履带结构调整自身构型并提高越障能力,更好地适应城市废墟、户外林地等障碍地形,如图1所示。其代价是操作自由度较多(6个),如果完全依赖远程人工遥控,会给操作人员带来很大的认知负担,容易导致控制失误<sup>[1]</sup>。因此,越障过程中机器人构型的自主规划是当前一个急需解决的问题。

根据实现方法,现有的构型规划算法主要分为三类:底盘姿态控制法、简化模型分析法和机器

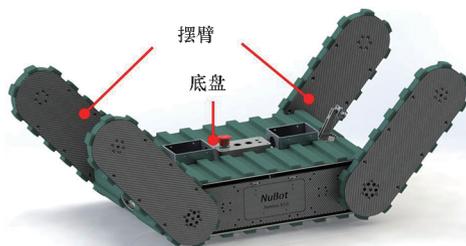


图1 NuBot-Rescue 四摆臂履带机器人

Fig. 1 The NuBot-Rescue tracked robot with four flippers  
学习法。

\* 收稿日期:2023-02-28

基金项目:国家自然科学基金资助项目(U1913202,U22A2059,62203460);湖南省自然科学基金资助项目(2021JC0004)

作者简介:陈柏良(1994—),男,湖南长沙人,博士研究生,E-mail:chenbailiang12@nudt.edu.cn;

卢惠民(通信作者),男,福建南平人,教授,博士,博士生导师,E-mail:lhmnew@nudt.edu.cn

底盘姿态控制法是根据地形起伏给出期望的底盘姿态(包括底盘俯仰角、摆臂角度等)并以此为摆臂控制目标。文献[2]根据机器人三维全局路径的起伏曲率来调整摆臂角度,使底盘与摆臂的整体构型接近路径的包络线。底盘姿态控制法可以实现基本的越障功能,但仍需要专门的越障构型来应对较为复杂的障碍,而且这些方法较少考虑越障动作的优化问题,存在局限性。

简化模型分析法是针对摆臂履带机器人与特定地形的接触过程进行运动学与动力学分析<sup>[3]</sup>,得到最优的越障过程。特定的地形通常指单个台阶<sup>[4-5]</sup>、楼梯<sup>[6]</sup>等结构化场景。文献[4]中建立了机器人翻越单个台阶过程的运动学和动力学模型,并总结出使摆臂力矩最小的控制规律。基于简化模型的方法分析详尽,但不能应对复杂、不规则、非结构化的地形等情况。

机器学习法既考虑了越障动作优化问题,也对不规则的地形具有一定适应性。其主要采用深度学习或强化学习方法获得机器人在当前状态下越障效果最好的机器人期望姿态或摆臂动作<sup>[7-10]</sup>。文献[9-10]采用强化学习方法,以地形起伏信息和机器人姿态等为状态量,以危险姿态、颠簸程度等指标为惩罚项,训练出能够自主上下楼梯的摆臂控制策略。文献[8]将模仿学习与强化学习相结合,使机器人面对多种不同障碍时能够快速训练出适应性较好的越障策略。然而,此类方法在未训练过的地形以及极易倾覆的高难度地形中的适应性尚未得到充分验证。

上述三种方法的共同特点都是针对越障过程中的机器人构型和姿态进行分析处理。机器人通过调整摆臂角度改变自身构型,从而控制自身姿态以翻越高难度的障碍,并减轻越障过程的颠簸。因此,机器人在越障时的构型变化取决于机器人当前的姿态以及在即将到达的地形上所期望的姿态。如果能够提前预知机器人在即将到达的地形上的状态,机器人就可以有效地规划越障构型。目前主要有仿真分析法和迭代几何法实现了履带机器人的姿态预测。前者主要利用物理仿真引擎,得到机器人在稳定状态下的仿真预测值<sup>[11]</sup>。其适用于任意地形和机器人外形,但计算量大、实时性差。迭代几何法<sup>[12]</sup>则是根据机器人以及地形的起伏形状,迭代寻找接触点以确定姿态。这种方法能够在复杂地形中预测姿态,但是现有研究并未考虑摆臂采用多种构型时的姿态预测问题。

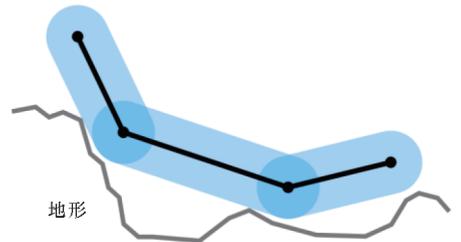
因此,本文针对传统方法无法兼顾优化越障

过程和适应复杂障碍的问题,以及强化学习方法训练过程不确定性强、高难度地形适应性存疑的不足,结合姿态预测对复杂地形的高适应性和动态规划算法寻找最优解的特点,提出一种既能应对复杂地形又具有优化能力的搜救机器人自主构型规划方法,该方法在仿真与实物实验中均展现出了优异的自动控制效果。

## 1 姿态预测

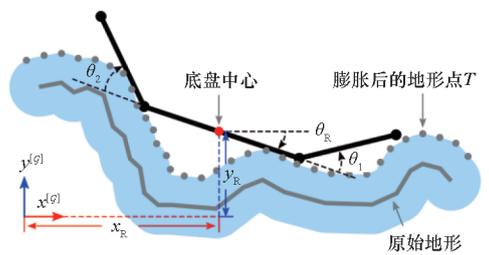
### 1.1 机器人与地形建模

研究对象为如图1所示的智能搜救机器人,其在一般履带底盘的基础上增加了四个可以独立转动的子履带(称为摆臂)。这种机器人的左侧与右侧结构是相同的,在建模时可以将机器人左侧与右侧分别使用同样的方法分析。当滚动角较小时,机器人模型可以简化为图2(a)所示的单侧履带模型:侧向投影机器人的左侧或右侧得到三段式模型,中间段为底盘,左右两段分别为后摆臂与前摆臂。在侧向投影中,当摆臂具有与主履带大致相同的厚度时,可以将地面向外膨胀履带的弯曲半径<sup>[5]</sup>,得到进一步简化的机器人无厚度线段模型,如图2(b)所示。



(a) 单侧履带简化模型

(a) The single-side pose modelling



(b) 膨胀地形简化模型及符号说明

(b) The terrain-expanded model and the notation definition

图2 单侧履带机器人模型简化过程与符号定义

Fig.2 The tracked robot's single-side pose modelling and notation definition

机器人下方的地形采用离散的点集表示,如图2(b)中的地形点。通过机载激光雷达或深度相机可以感知局部地形并结合建图算法<sup>[13-14]</sup>获

得原始点云;再将点云膨胀并重新采样得到膨胀后的地形点集,记为集合  $T$ ,其中的点记为  $p \in T$ 。

### 1.2 姿态预测问题定义

机器人在地形中的姿态,是由机器人自身构型以及与地形的接触点决定的。设地面参考系为  $[G]$ ,姿态预测问题可以描述为下列形式。

给定:

- 1)  $T$ :地形的点集。
- 2)  $x_R$ :底盘中心在  $[G]$  中的  $x$  轴坐标。
- 3)  $\theta_1$ :机器人前摆臂角度,逆时针为正。
- 4)  $\theta_2$ :机器人后摆臂角度,顺时针为正。

求机器人在静态稳定状态下的:

- 1)  $y_R$ :底盘中心在  $[G]$  中的  $y$  轴坐标。
- 2)  $\theta_R$ :机器人在  $[G]$  中的俯仰角,逆时针为正。

约束条件:

- 1) 机器人质心的前后两侧各存在一个以上与地形的接触点;
- 2) 由接触点确定的机器人姿态应当使机器人位于其范围内所有地形点的上方。

图 2(b)展示了上述参数的具体意义。因此,求解姿态就是从点集  $T$  中找出满足约束条件的地形点即接触点,进而确定姿态参数。这里预测的是机器人在“稳定状态”下的姿态,基于此可以提出两个约束条件。条件 1 是机器人在地面上保持稳定的基本条件;条件 2 代表机器人的任一部位都不能与地面出现碰撞干涉的基本物理约束。

### 1.3 姿态预测问题的数学描述

#### 1.3.1 机器人模型的数学描述

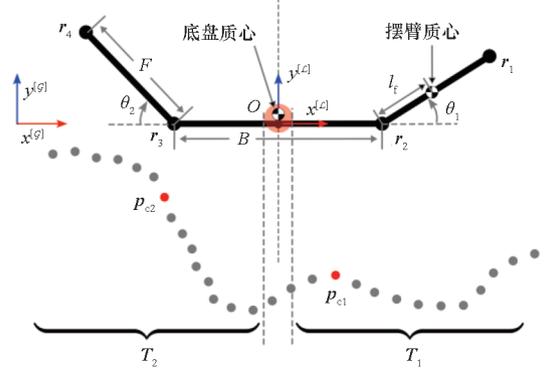
以  $[G]$  中的待预测位置  $(x_R, 0)$  为原点建立一个  $y$  轴与重力方向平行、 $x$  轴方向为机器人朝向的局部坐标系  $[L]$ 。令机器人的底盘中心位置与原点重合,底盘与  $x$  轴平行,可以得到如图 3(a) 所示的模型。该模型可以用分段直线描述各部件:

$$\mathcal{R} = \begin{cases} I_{r_2 r_1} & \text{前摆臂 } r_2 r_1 \text{ 段} \\ I_{r_3 r_2} & \text{底盘 } r_3 r_2 \text{ 段} \\ I_{r_4 r_3} & \text{后摆臂 } r_4 r_3 \text{ 段} \end{cases} \quad (1)$$

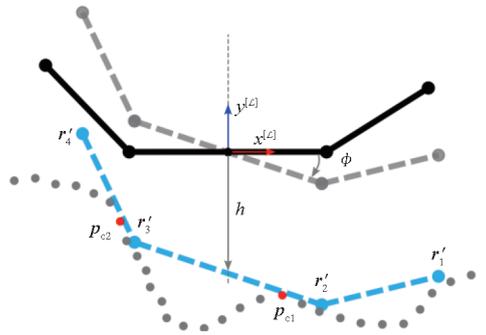
其中:前摆臂末端点  $r_1$ ,坐标  $(\frac{1}{2}B + F\cos\theta_1, F\sin\theta_1)$ ;前摆臂转轴点  $r_2$ ,坐标  $(\frac{1}{2}B, 0)$ ;后摆臂转轴点  $r_3$ ,坐标  $(-\frac{1}{2}B, 0)$ ;后摆臂末端点  $r_4$ ,坐标  $(-\frac{1}{2}B - F\cos\theta_2, F\sin\theta_2)$ ;  $B$  为底盘的长度,  $F$  为摆臂的长度。

#### 1.3.2 接触关系的数学描述

地形点与机器人接触可以表示为点在分段直线模型上,即机器人  $\mathcal{R}$  在  $[L]$  中首先绕原点旋转俯仰角  $\phi$ ,再沿  $y$  轴平移  $h$  变换为  $\mathcal{R}'$ ,并与地形点  $p_{c1}(x_{c1}, y_{c1}) \in T, p_{c2}(x_{c2}, y_{c2}) \in T$  接触,如图 3(b) 所示。



(a) 在局部坐标系中水平放置的机器人的分段直线模型  
(a) A segmented linear model of the robot placed horizontally in the local coordinate system



(b) 将机器人模型先旋转  $\phi$ ,再平移  $h$ ,与接触点重合  
(b) The robot model is first rotated  $\phi$ , then translated  $h$  to coincide with the contact points

图 3 机器人模型以及机器人姿态与接触点的关系

Fig. 3 Robot model and the relationship between robot pose and contact points

为满足约束条件 1,地形点需分别位于机器人质心前、后侧。点在直线上的几何关系可以用直线的齐次坐标向量与点的齐次坐标向量的点积为零表示。而  $p_{c1}, p_{c2}$  既可能位于摆臂上,也可能位于底盘上,因此对于  $p_{c1}$ ,有式(2)成立:

$$\begin{cases} I_{r_2 r_1}^T \cdot p_{c1} = 0 & p_{c1} \text{ 位于前摆臂上, } x_{c1} \in (x_{r_2}, x_{r_1}) \\ I_{r_3 r_2}^T \cdot p_{c1} = 0 & p_{c1} \text{ 位于底盘上, } x_{c1} \in [x_{com}, x_{r_2}] \end{cases} \quad (2)$$

对于  $p_{c2}$ ,有式(3)成立:

$$\begin{cases} I_{r_3 r_2}^T \cdot p_{c2} = 0 & p_{c2} \text{ 位于底盘上, } x_{c2} \in [x_{r_3}, x_{com}) \\ I_{r_4 r_3}^T \cdot p_{c2} = 0 & p_{c2} \text{ 位于后摆臂上, } x_{c2} \in [x_{r_4}, x_{r_3}) \end{cases} \quad (3)$$

其中:  $\mathbf{l}_{r'_i+r'_i}$  ( $i=1,2,3$ ) 为以齐次坐标向量形式表示的变换后的机器人模型  $\mathcal{R}'$ , 其含义为经过  $\mathbf{r}'_{i+1}$ 、 $\mathbf{r}'_i$  两点的直线;  $\mathbf{p}_{c_j}$  ( $j=1,2$ ) 为以齐次坐标向量形式表示的接触点, 如式(4)所示。

$$\begin{cases} \mathbf{l}_{r'_i+r'_i} = \begin{bmatrix} y_{r'_{i+1}} - y_{r'_i} \\ -(x_{r'_{i+1}} - x_{r'_i}) \\ x_{r'_{i+1}}y_{r'_i} - x_{r'_i}y_{r'_{i+1}} \end{bmatrix} \\ \mathbf{p}_{c_j} = \begin{bmatrix} x_{c_j} \\ y_{c_j} \\ 1 \end{bmatrix} \end{cases} \quad (4)$$

式中,  $(x_{r'_i}, y_{r'_i})$  为  $\mathbf{r}'_i$  的坐标, 是图 3(a) 中初始状态下的  $\mathbf{r}_i$  经过旋转和平移变换后得到的, 如式(5):

$$\begin{bmatrix} x_{r'_i}(h, \phi) \\ y_{r'_i}(h, \phi) \end{bmatrix} = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} x_{r_i} \\ y_{r_i} \end{bmatrix} + \begin{bmatrix} 0 \\ h \end{bmatrix} \quad (5)$$

在约束条件 1 中, 接触点还需要满足分布在质心前后的要求, 即式(2)、式(3)中接触点的  $x$  轴坐标取值范围要远离质心。机器人模型质心在  $[\mathcal{L}]$  中的坐标  $(x_{\text{com}}, y_{\text{com}})$  可以表示为式(6):

$$\begin{cases} x_{\text{com}} = \frac{1}{m_B + 2m_F} [m_F l_f (\cos\theta_1 - \cos\theta_2) + m_B x_b] \\ y_{\text{com}} = \frac{1}{m_B + 2m_F} [m_F l_f (\sin\theta_1 + \sin\theta_2) + m_B y_b] \end{cases}$$

$$\Rightarrow \left( x_{\text{com}} - \frac{m_B x_b}{m_B + 2m_F} \right)^2 + \left( y_{\text{com}} - \frac{m_B y_b}{m_B + 2m_F} \right)^2$$

$$= 2[1 - \cos(\theta_1 + \theta_2)] \left( \frac{m_F l_f}{m_B + 2m_F} \right)^2$$

$$\leq \left( \frac{2m_F l_f}{m_B + 2m_F} \right)^2 \quad (6)$$

其中,  $(x_b, y_b)$  是底盘质心的坐标,  $l_f$  是摆臂质心到摆臂转轴的距离,  $m_B$  和  $m_F$  是机器人单侧底盘以及单个摆臂的质量。假设底盘质心与底盘形状中心重合即  $(x_b, y_b) = (0, 0)$ , 则有式(7):

$$x_{\text{com}} \in \left[ -\frac{2m_F l_f}{m_B + 2m_F}, \frac{2m_F l_f}{m_B + 2m_F} \right] \quad (7)$$

在实际中,  $m_B$  通常比  $m_F$  大 8~10 倍, 因此  $(x_{\text{com}}, y_{\text{com}})$  只会在半径约  $\frac{1}{5}l_f$  的圆内变化, 只需使接触点不在该范围内即可满足稳定性要求。因此, 可以初步将所有地形点在  $[\mathcal{L}]$  下以  $x_{\text{com}}$  的上下边界为界分成  $\mathbf{T}_1$ 、 $\mathbf{T}_2$  前后两个子集, 并设定候选接触点分别位于其中, 即:  $\mathbf{p}_{c1} \in \mathbf{T}_1$ ,  $\mathbf{p}_{c2} \in \mathbf{T}_2$ 。

与此同时, 所有位于变换之后机器人范围内的地形点  $\mathbf{p}$  还需要满足约束条件 2 的要求。可以用  $\mathbf{l}_{r'_i+r'_i}$  ( $i=1,2,3$ ) 与  $\mathbf{p}$  的点积正负表示点相对直

线的位置, 因此该条件可以转化为式(8):

$$\begin{cases} \mathbf{l}_{r'_2r'_1}^T \cdot \mathbf{p} \leq 0 & x_p \in (x_{r'_2}, x_{r'_1}) \\ \mathbf{l}_{r'_3r'_2}^T \cdot \mathbf{p} \leq 0 & x_p \in [x_{r'_3}, x_{r'_2}], \mathbf{p}(x_p, y_p) \in \mathbf{T} \\ \mathbf{l}_{r'_4r'_3}^T \cdot \mathbf{p} \leq 0 & x_p \in [x_{r'_4}, x_{r'_3}] \end{cases} \quad (8)$$

需要注意的是, 变换后的机器人模型摆臂末端点  $\mathbf{r}'_1$ 、 $\mathbf{r}'_4$  有可能位于底盘下方即  $x_{r'_2} > x_{r'_1}$  或  $x_{r'_4} > x_{r'_3}$ , 此时式(8)中的符号取反。

最终满足所有条件的候选接触点  $\mathbf{p}_{c1}$ 、 $\mathbf{p}_{c2}$  所对应的  $(h, \phi)$  即为机器人在当前位置的预测姿态参数:  $(y_R, \theta_R) = (h, \phi)$ 。

联立式(2)~(5)、式(8)可以得到如下形式的姿态预测问题的数学描述:

求解:  $h, \phi, \mathbf{p}_{c1}(x_{c1}, y_{c1}) \in \mathbf{T}_1, \mathbf{p}_{c2}(x_{c2}, y_{c2}) \in \mathbf{T}_2$

条件:

$$\begin{cases} \mathbf{l}_{r'_2r'_1}^T \cdot \mathbf{p}_{c1} = 0 & \mathbf{p}_{c1} \text{ 位于前摆臂上, } x_{c1} \in (x_{r'_2}, x_{r'_1}) \\ \mathbf{l}_{r'_3r'_2}^T \cdot \mathbf{p}_{c1} = 0 & \mathbf{p}_{c1} \text{ 位于底盘上, } x_{c1} \in \left( \frac{2m_F l_f}{m_B + 2m_F}, x_{r'_2} \right) \end{cases} \quad (9)$$

$$\begin{cases} \mathbf{l}_{r'_3r'_2}^T \cdot \mathbf{p}_{c2} = 0 & \mathbf{p}_{c2} \text{ 位于底盘上, } x_{c2} \in \left( x_{r'_3}, -\frac{2m_F l_f}{m_B + 2m_F} \right) \\ \mathbf{l}_{r'_4r'_3}^T \cdot \mathbf{p}_{c2} = 0 & \mathbf{p}_{c2} \text{ 位于后摆臂上, } x_{c2} \in [x_{r'_4}, x_{r'_3}] \end{cases} \quad (10)$$

$$\begin{cases} \mathbf{l}_{r'_2r'_1}^T \cdot \mathbf{p} \leq 0 & x_p \in (x_{r'_2}, x_{r'_1}) \\ \mathbf{l}_{r'_3r'_2}^T \cdot \mathbf{p} \leq 0 & x_p \in [x_{r'_3}, x_{r'_2}], \mathbf{p}(x_p, y_p) \in \mathbf{T} \\ \mathbf{l}_{r'_4r'_3}^T \cdot \mathbf{p} \leq 0 & x_p \in [x_{r'_4}, x_{r'_3}] \end{cases} \quad (11)$$

$$\begin{bmatrix} x_{r'_i} \\ y_{r'_i} \end{bmatrix} = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} x_{r_i} \\ y_{r_i} \end{bmatrix} + \begin{bmatrix} 0 \\ h \end{bmatrix} \quad (12)$$

$i=1,2,3,4$

该问题并不总是有可行解, 而无效解对应的机器人构型与地形位置, 将在后续的构型规划中被视为危险状态进行规避。

#### 1.4 姿态预测问题的解析求解方法

至此, 姿态预测问题的数学模型已经建立, 实际运用时需要高效的求解算法。当地形点总数量有限时, 可以利用式(2)、式(3)组成的等式方程组快速地遍历求解局部地形点集中所有候选接触点对  $(\mathbf{p}_{c1}, \mathbf{p}_{c2})$  对应的机器人姿态解析解, 并根据区间范围条件、不等式条件筛选得到满足所有约束条件的姿态。基于这种思路, 设计了一种可以适用于复杂地形的履带机器人姿态预测方法。

首先从地形点集  $\mathbf{T}_1$ 、 $\mathbf{T}_2$  中选取一对候选接触

点( $p_{c1}, p_{c2}$ ),由于接触点既可能位于摆臂上也可能位于底盘上,因此式(2)、式(3)可以根据下列情况各自提出一个等式组成待求解的等式方程组:

- 1)  $p_{c1}$ 在前摆臂上,  $p_{c2}$ 在后摆臂上;
- 2)  $p_{c1}$ 在前摆臂上,  $p_{c2}$ 在底盘上;
- 3)  $p_{c1}$ 在底盘上,  $p_{c2}$ 在后摆臂上;
- 4)  $p_{c1}$ 在底盘上,  $p_{c2}$ 在底盘上。

这些方程组的基本形式是一致的,接下来以第二种情况为例,其他三种情况将对应参数替换即可。将式(4)、式(5)代入由式(2)第一行等式、式(3)第一行等式组成的方程组,令  $z = \tan(\phi/2)$ , 则  $\sin\phi = 2z/(1+z^2)$ ,  $\cos\phi = (1-z^2)/(1+z^2)$ 。整理得到关于  $h$  和  $z$  的方程组如式(13):

$$\begin{cases} (A_1 - X_1h)z^2 + (B_1 - 2Y_1h)z + (C_1 + X_1h) = 0 \\ (A_2 - X_2h)z^2 + (B_2 - 2Y_2h)z + (C_2 + X_2h) = 0 \end{cases} \quad (13)$$

其中的系数  $A_i, B_i, C_i, X_i, Y_i (i=1,2)$  为只与机器人模型的原坐标  $(x_{r_i}, y_{r_i})$ 、候选接触点坐标  $(x_{c_i}, y_{c_i})$  有关的常数。将该方程组视为关于  $z$  的多项式,如果  $z$  的解存在,则这两个多项式之间必然存

在公因式,即两者的结式为零。假设式(13)的最高项系数  $(A_i - X_ih)$  均不为零,展开结式可以得到关于  $h$  的四次方程,利用多项式求根算法可以求出  $h$  的解;进一步利用二次方程求根公式与反正切函数得到  $h$  对应的  $\phi$ 。采用相同的方法,可以求出其他三种情况下的解。根据接触点  $x$  轴坐标的范围条件以及等式约束条件,可从这四种情况中找到一组满足接触点条件的姿态参数  $(h, \phi)$ 。

基于上述解析求解方法,在给定一对候选接触点坐标后,可以计算出满足约束条件 1 的姿态参数。在此之后,根据式(8)判断是否满足约束条件 2。如果不满足,则重新从  $T_1, T_2$  中选取一对候选接触点,最终找到机器人在给定地形上采取指定前后摆臂角度(构型)时的单侧机器人姿态。

姿态预测算法的整体流程如图 4 所示。当  $T_1, T_2$  中地形点总数为 30 ~ 50 时,指定  $x_R$  以及  $\theta_1, \theta_2$  时的单个姿态的平均求解时间约为 0.75 ms,即每秒预测 1 000 ~ 1 500 个姿态,能够满足实时进行构型规划的需求;采用点集的形式描述地面,使算法能够应对各种上下起伏不规则的地形。

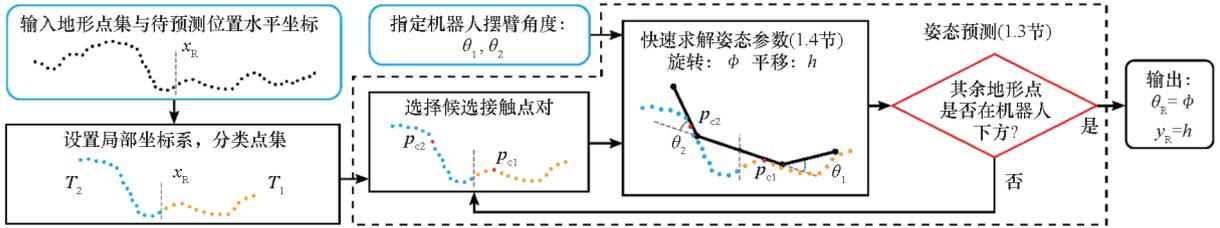


图 4 姿态预测算法框图

Fig. 4 Framework of pose prediction algorithm

## 2 构型规划

基于快速的姿态预测算法,进一步设计越障构型规划算法。算法以传统的动态规划方法为基础,寻找局部感知范围内的最优越障构型序列;结合滚动优化的思想设计不断更新构型序列的规划器,实现机器人越障整体过程的优化。

### 2.1 机器人状态与动作评价

动态规划算法需要首先对越障过程中的机器人状态以及状态之间的转移过程进行合理的评价。

如图 5 所示,左侧为地形与机器人在当前位置  $P_1^i$  的状态,地形中红色的点表示当前时刻用于姿态预测的局部地形点。姿态预测算法以  $P_1^i$  为起点沿前进方向设定路径点  $P_2^i, P_3^i, P_4^i$ ;在每个路径点处按照预设的机器人构型  $(\theta_1, \theta_2)$  进行姿态

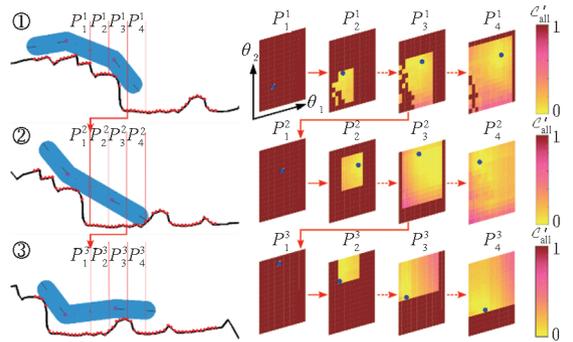


图 5 机器人下壕沟过程的构型规划示意图

Fig. 5 Diagram of robot configuration planning when going down the trench

预测,即可得到由路径点位置、前后摆臂角度、机器人俯仰角、离地高度等元素构成的离散状态空间。机器人在指定位置与摆臂角度时的姿态定义为离散的状态  $S_{P, \theta_1, \theta_2} = (y_R(P, \theta_1, \theta_2), \theta_R(P, \theta_1, \theta_2))$ ,

其中  $P$  代表路径点。前、后摆臂在相邻路径点之间变化的角度分别记为  $a_1, a_2$ , 范围为  $\{-bk, \dots, 0, \dots, bk\}$ ,  $b \in \mathbb{Z}$ , 又称摆臂动作; 其中  $k$  为摆臂角度离散化分辨率,  $b$  为每当机器人前进一个路径点时摆臂角度最多可以变化的单位分辨率个数。

根据人类操作员的经验, 设计了代价函数  $\mathcal{C}$  来评价在状态  $S_{P, \theta_1, \theta_2}$  时采取动作  $(a_1, a_2)$  的代价:

$$\mathcal{C}(P, \theta_1, \theta_2, a_1, a_2) = \begin{cases} 10, & S_{P+1, \theta_1+a_1, \theta_2+a_2} \text{ 不稳定或超出空间范围} \\ \mathcal{C}_\theta(S_{P, \theta_1, \theta_2}, S_{P+1, \theta_1+a_1, \theta_2+a_2}) + \mathcal{C}_y(S_{P, \theta_1, \theta_2}) + \\ \mathcal{C}_s(S_{P, \theta_1, \theta_2}) + \mathcal{C}_l(S_{P, \theta_1, \theta_2}) + \mathcal{C}_r(a_1, a_2), & \text{其他} \end{cases} \quad (14)$$

其中,  $\mathcal{C}_\theta(S_{P, \theta_1, \theta_2}, S_{P+1, \theta_1+a_1, \theta_2+a_2})$  为前后两个状态间俯仰角变化的代价,  $\mathcal{C}_y(S_{P, \theta_1, \theta_2})$  为当前状态下质心高度与底盘范围内地面平均高度的偏差的代价,  $\mathcal{C}_s(S_{P, \theta_1, \theta_2})$  为当前状态下俯仰方向稳定性的代价,  $\mathcal{C}_l(S_{P, \theta_1, \theta_2})$  为当前状态下摆臂力矩的代价,  $\mathcal{C}_r(a_1, a_2)$  为摆臂调整幅度的代价。

利用式(14)对离散状态空间中所有的状态与动作进行评价, 作为构型规划的依据。

### 2.2 动态规划

在获得每个路径点处机器人各种构型时的状态以及采取各种动作时的代价值后, 运用动态规划算法寻找机器人状态空间中连接起始状态和目标状态的最优状态转移路径, 使得式(15)的累计代价值最小。

$$\mathcal{C}_{\text{all}}(S_{P, \theta_1, \theta_2}) = \min \{ \mathcal{C}_{\text{all}}(S_{P-1, \theta_1-a_1, \theta_2-a_2}) + \mathcal{C}(P-1, \theta_1-a_1, \theta_2-a_2) \} \quad (15)$$

首先指定机器人的起始状态(如图5中位置  $P_1^i$  和此时机器人的构型), 以及期望的目标点位置  $P_4^i$ ; 随后计算从起始状态开始每一次状态转移的累计最小代价  $\mathcal{C}_{\text{all}}$ , 依次迭代直到  $P_4^i$ , 得到如图5右侧所示的各路径点处的相对累计代价图  $\mathcal{C}'_{\text{all}}$ (减去了当前路径点中最小的  $\mathcal{C}_{\text{all}}$ , 对应图中蓝点位置的状态)。选取  $P_4^i$  位置  $\mathcal{C}'_{\text{all}}$  最小的状态作为目标状态, 并回溯累计代价最低的状态转移过程, 得到最优的越障构型变换序列, 即图5中蓝点的连线。

### 2.3 基于滚动优化的构型序列更新

机器人的感知范围有限导致构型规划的距离有限, 因此运用滚动优化的方式在前进的同时不断更新构型序列, 如图5中的3个阶段所示。

机器人在当前位置  $P_1^i$  采集相应信息, 预测前方  $P_2^i, P_3^i, P_4^i$  路径点处的姿态并进行构型规划。随着机器人前进, 摆臂执行规划的动作并到达  $P_3^i$  路径点。此时机器人被视为到达了新的位置  $P_1^{i+1}$ , 根据当前的状态对路径点  $P_2^{i+1}, P_3^{i+1}, P_4^{i+1}$  进行新一轮姿态预测与规划, 得到更新后的构型序列并执行, 如此循环。基于此, 算法持续输出在机器人前方局部范围内的构型序列, 通过不断滚动向前实现整体越障效果的优化。

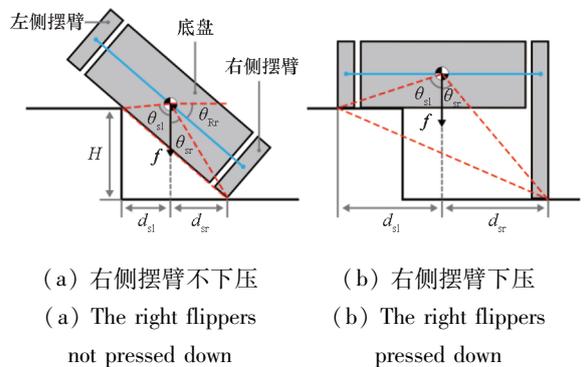
### 2.4 非对称地形构型规划策略

左右非对称的地形会使机器人倾斜, 导致姿态预测算法的基本条件“滚动角度较小”无法满足, 同时也将影响滚转方向的稳定性, 使机器人更容易侧翻。图6中机器人滚转方向的稳定性可以使用力角稳定性指标<sup>[15]</sup>分析, 如式(16):

$$\beta = \min \{ \theta_{sl} \cdot \|d_{sl}\| \cdot \|f\|, \theta_{sr} \cdot \|d_{sr}\| \cdot \|f\| \} \quad (16)$$

其中:  $\theta_{sl}, \theta_{sr}$  分别指机器人左侧与右侧的接触点形成的翻转轴法向与合力  $f$  (不包括支持力, 静态稳定时主要是重力) 的夹角;  $d_{sl}, d_{sr}$  指左侧与右侧的翻转轴到合力方向的距离。

当所处地形相对机器人左右高度差  $H$  较大时, 如图6(a)所示, 机器人滚转角  $\theta_{Rr}$  很大, 极易侧翻。此时将地形较低一侧的摆臂下压可以提高机器人稳定性, 如图6(b)所示; 且机器人被抬起使滚转角接近  $0^\circ$  时, 能满足姿态预测算法的假设。



(a) 右侧摆臂不下压 (b) 右侧摆臂下压  
(a) The right flippers not pressed down (b) The right flippers pressed down

图6 机器人在单侧台阶上的滚转稳定性分析示意图  
Fig.6 Diagram of the rolling stability analysis of the robot on single-side steps

因此, 设计了一种辅助侧摆臂角度计算方法, 以减少机器人在面对不对称障碍时的滚转角变化, 在保持稳定的同时让姿态预测算法能够有效运行。首先分析局部地形中不同方位的起伏程度, 选择地形较高的方位中的点云数据作为越障侧姿态预测与构型规划的输入, 地形

较低的方位中的点云数据作为辅助侧摆臂角度计算的输入。如图 7 所示,当灰色圆点  $T_{li}$  表示的左侧地形比红色方形  $T_{ri}$  表示的右侧地形平均值高时,黑色线段表示的机器人左侧为越障侧,淡蓝色表示的机器人右侧为辅助侧。利用  $T_{li}$  进行姿态预测与构型规划,可以获得在当前位置处期望的机器人左侧姿态  $(y_R, \theta_R)$ , 接触点  $p_{cl1}$ 、 $p_{cl2}$ , 以及左侧摆臂角度  $\theta_{l1}$ 、 $\theta_{l2}$ 。此时若要使滚转角接近  $0^\circ$ , 则右侧的摆臂需要下压将车体撑起, 使右侧摆臂转轴  $r_{r1}$ 、 $r_{r2}$  与左侧摆臂转轴的侧向投影位置相近。因此右侧(辅助侧)摆臂角度  $\theta_{r1}$ 、 $\theta_{r2}$  的计算只需在给定摆臂转轴位置的条件下, 求得满足摆臂与地形接触、摆臂不与地形碰撞干涉、摆臂角度在允许范围内三个条件的摆臂角度即可, 此时辅助侧摆臂与地形的接触点为  $p_{cr1}$ 、 $p_{cr2}$ 。

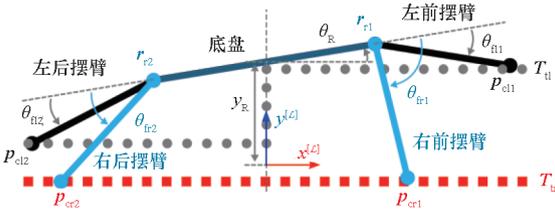


图 7 机器人攀爬左侧台阶时的辅助侧摆臂动作示意图  
Fig. 7 Diagram of the assistant side flipper action when the robot is climbing the left side step

综上所述,以姿态预测为基础,设计了基于动态规划并能滚动更新的机器人构型规划器,并提出了一种能够适应非对称地形的规划策略。

### 3 实验与讨论

为了测试所提出的机器人自主构型规划算法(以下简称构型规划算法)的效果,在 2 个 Gazebo 仿真场景与 3 个真实场景中进行了实验,并与基于深度强化学习的方法<sup>[10]</sup>进行了对比。

#### 3.1 实验设置与对比指标

##### 3.1.1 参数设置

如图 1 所示,实验使用的 NuBot-Rescue 机器人的基本物理参数为:底盘轴距  $B = 0.552$  m,摆臂轴距  $F = 0.336$  m。通过机载激光雷达结合建图算法<sup>[16]</sup>输出局部地形点云;通过惯性测量单元(inertial measurement unit, IMU)感知机器人姿态。Gazebo 仿真中的机器人与实物机器人具有相同的物理参数和传感器配置。仿真履带运动控制方法采用文献[17]提出的“不可变形履带的快速仿真方法”。

系统软硬件架构如图 8 所示。采用机器人操作系统(ROS)将所有程序运行在机载电脑(Intel i9-9900T, 16 GB)中,主要包括:同步定位与建图节点(5 Hz)、姿态预测与构型规划节点(1 ~ 2 Hz)、电机控制节点(100 Hz)、人工控制节点(100 Hz)、交互界面。机载电脑从激光雷达、IMU、相机、电机驱动器等机载设备获得环境感知信息与机器人状态感知信息;通过无线通信将感知信息传递到遥控端,并接收前进方向、速度等人工控制指令信息。在电机控制节点中将人工控制指令与规划的构型序列融合,输出摆臂、履带的控制指令。

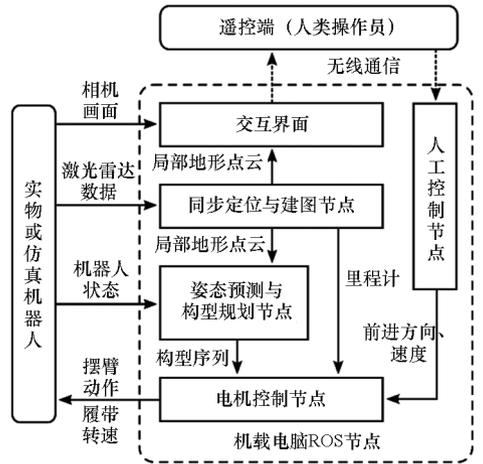


图 8 机器人软硬件架构

Fig. 8 Framework of the robot's software and hardware

在算法的参数方面,局部地形感知域为以机器人底盘中心向前 1.5 m、向后 0.5 m,左右各 0.5 m 的长方形范围;预测路径点间距 0.2 m,路径点数量至多 4 个;摆臂角度分辨率  $k = 10^\circ$ ,摆臂可调整单位数  $b = 3$ ,摆臂活动范围  $-70^\circ \sim 70^\circ$ 。

##### 3.1.2 对比指标

用于量化对比的指标主要有:单次任务机器人姿态角变化率绝对值的总和  $\hat{\theta}$ (俯仰  $\hat{\theta}_R$ 、滚转  $\hat{\theta}_{roll}$ )<sup>[10]</sup>、单次任务过程中的姿态角速度最大值  $|\dot{\theta}|_{max}$ (俯仰  $|\dot{\theta}_R|_{max}$ 、滚转  $|\dot{\theta}_{roll}|_{max}$ )、单次任务耗时  $t_{cost}$  等三种。其中  $\hat{\theta}$  由式(17)计算:

$$\hat{\theta} = \int_0^{t_{cost}} |\dot{\theta}| dt = \sum_{t=1}^{T-1} |\theta(t+1) - \theta(t)| \quad (17)$$

式中,  $T$  表示单次任务的时间总步数。 $\hat{\theta}$  能够衡量机器人运行时的整体摇晃程度,  $|\dot{\theta}|_{max}$  能够衡量瞬时摇晃程度。除废墟场景外,每个场

景各实验 10 次,取上述指标的均值进行分析。实验的起点设置为机器人中心距离障碍物边缘 1 m 处,终点设置为机器人中心离开障碍物后 1 m 处。

### 3.2 仿真实验

#### 3.2.1 台阶场景

0.4 m 高的单台阶是机器人能够通过的极限高度,用于测试算法对高难度对称地形的适应性。主要包括上、下台阶两个过程,如图 9 所示。

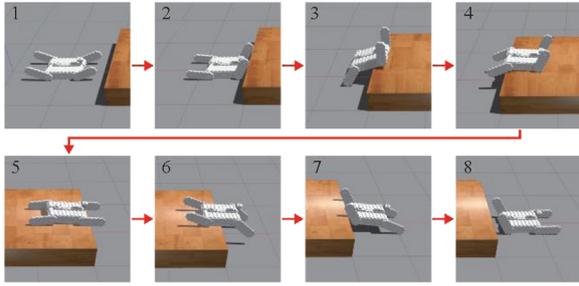


图 9 0.4 m 单台阶场景越障过程

Fig. 9 Processes of the robot moving through the 0.4 m single-step scenarios in simulation

量化指标如表 1 所示,对比强化学习方法和人工操作<sup>[10]</sup>,构型规划算法在单台阶越障场景中具有比强化学习方法更好的通行平稳程度和相近的通行时间,且都明显优于人工操作。

表 1 仿真台阶与楼梯场景中的不同算法指标对比

Tab. 1 Quantitative evaluation results in the single-step and steep staircase scenarios in simulation

场景	$\hat{\theta}_R/\text{rad}$			$(\hat{\theta}_{R\max} - \hat{\theta}_{R\min})/\text{rad}$			$t_{\text{cost}}/\text{s}$		
	本文	文献[10]	人工	本文	文献[10]	人工	本文	文献[10]	人工
0.4 m 单台阶	<b>2.98</b>	3.43	4.02	<b>0.94</b>	1.11	2.92	<b>27.81</b>	28.07	46.17
陡峭楼梯	5.15	<b>4.53</b>	6.30	<b>1.26</b>	2.60	3.88	<b>36.76</b>	41.21	76.53

### 3.3 实物实验

#### 3.3.1 单侧台阶场景

在单侧台阶场景实物实验中,机器人需要攀爬位于左侧的高 0.3 m 台阶,如图 11(a) 所示。运行中的量化指标如表 2 所示。结果表明构型规划算法能够使机器人稳定地攀爬具有较高难度的单侧台阶,且滚转方向的摇晃程度较小。

#### 3.3.2 斜向楼梯场景

在斜向楼梯场景中,机器人需要斜向通过楼梯地形,每级楼梯长 0.38 m、高 0.12 m,共 8 级。具体效果如图 11(b) ~ (c) 所示。

量化指标如表 2 所示,相比上楼梯的过程,下楼梯的过程中  $\hat{\theta}_R$ 、 $\hat{\theta}_{\text{roll}}$  更接近,但  $|\dot{\theta}_R|_{\max}$ 、 $|\dot{\theta}_{\text{roll}}|_{\max}$

#### 3.2.2 陡峭楼梯场景

陡峭楼梯场景由上 33.7° 楼梯和下 45° 楼梯两部分组成,其中上楼梯时台阶高度 0.2 m,台阶宽度 0.3 m,共 6 级;下楼梯时台阶高度 0.2 m,台阶宽度 0.2 m,共 6 级。整体过程如图 10 所示。

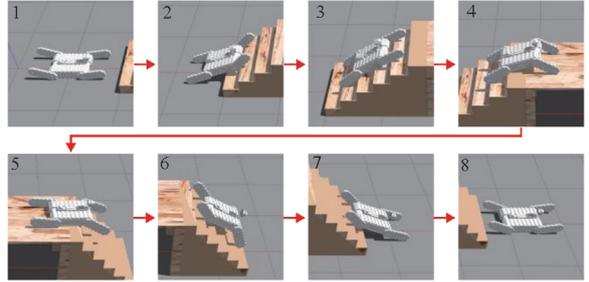


图 10 陡峭楼梯场景越障过程

Fig. 10 Processes of the robot moving through the steep staircase scenarios in simulation

运行中的量化指标如表 1 所示。其中  $\hat{\theta}_R$  较深度强化学习方法稍大,因为下楼梯的过程中算法并未将摆臂完全展平,机器人随着楼梯起伏而轻微摇摆。但是,构型规划算法的第二项指标 ( $\hat{\theta}_{R\max} - \hat{\theta}_{R\min}$ ) 更小,表明  $\hat{\theta}_R$  的上限与深度强化学习方法的上限几乎相同。并且,构型规划算法的任务完成时间更短,间接表明越障过程更流畅。

更大,说明瞬时晃动更大。这是由于机器人质心从平台下到楼梯上时会向左前方倾转;而上台阶时右后摆臂下压使底盘较平,在上到台阶顶部时倾转较小,从而晃动较小。

#### 3.3.3 废墟场景

废墟场景由水泥砖块、家具、树枝、麻袋等物体堆砌而成。机器人需要正向、反向通过废墟,整体水平移动距离约 8 m。具体效果如图 12 所示。

整体而言,机器人流畅而平稳地通过了废墟。在上坡时前后摆臂放平以提高稳定性;在下坡时提前将前摆臂下压以降低俯仰角晃动。在局部地形的左右高差较大时,左右摆臂采取不同的动作,虽然幅度较小但有效地防止了机器人侧翻。每次

实验时的量化指标单独列出,如表 3 所示。废墟场景整体起伏大、距离长,使得  $\hat{\theta}_R$ 、 $\hat{\theta}_{roll}$  较高。但借助构型规划算法,瞬时晃动  $|\dot{\theta}_R|_{max}$ 、 $|\dot{\theta}_{roll}|_{max}$  相对较小,越障过程较平稳。

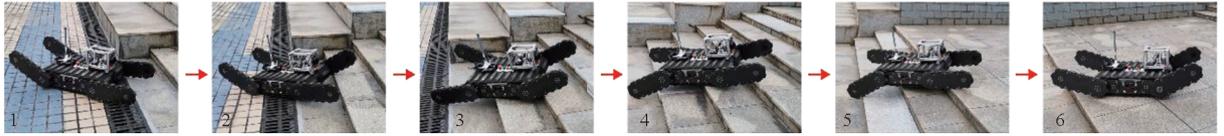
综上所述,在仿真实验中,构型规划算法能够

使机器人顺利通过高难度的障碍地形,且相比深度强化学习算法具有更好的效果;在实物实验中,进一步测试了算法的实际运行性能,结果表明算法能够有效适应难度较高的单侧台阶,轻松地斜向上下楼梯,平稳地通过复杂的废墟地形。



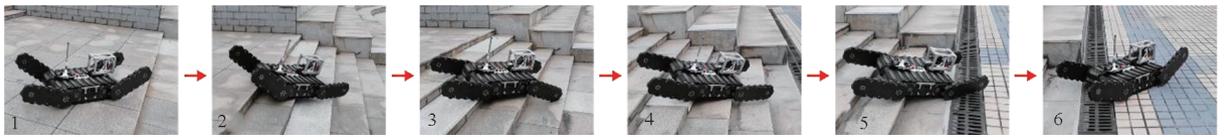
(a) 0.3 m 单侧台阶场景越障过程

(a) Obstacle crossing process in the 0.3 m single-side-step scenario



(b) 斜向上楼梯场景越障过程

(b) Obstacle crossing process in the up-side oblique stairs scenario



(c) 斜向下楼梯场景越障过程

(c) Obstacle crossing process in the down-side oblique stairs scenario

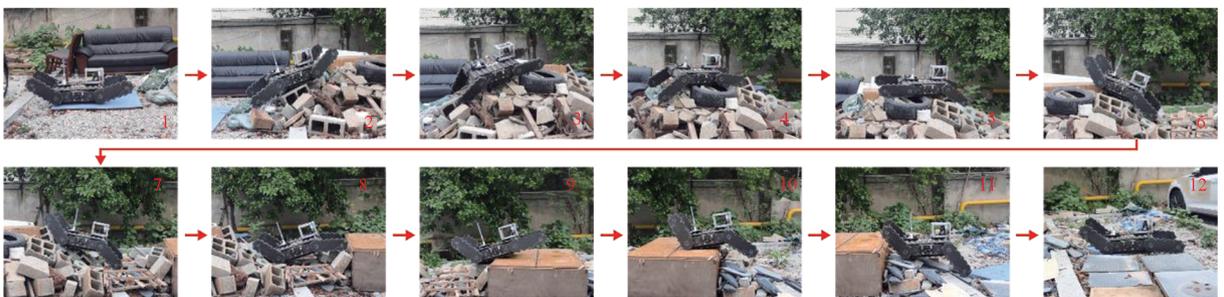
图 11 机器人在真实单侧台阶与斜向楼梯场景的越障过程

Fig. 11 Processes of the robot moving through the single-side-step and oblique stairs scenarios in real-world

表 2 实物单侧台阶与斜向楼梯场景的量化指标

Tab. 2 Quantitative evaluation results in the single-side-step and oblique stairs scenarios in real-world

场景	$\hat{\theta}_R/\text{rad}$	$ \dot{\theta}_R _{max}/(\text{rad/s})$	$\hat{\theta}_{roll}/\text{rad}$	$ \dot{\theta}_{roll} _{max}/(\text{rad/s})$	$t_{cost}/\text{s}$
0.3 m 单侧台阶上	1.40	0.68	2.21	0.95	19.98
斜向楼梯上	4.40	0.93	5.24	1.15	40.48
斜向楼梯下	4.86	2.05	5.59	1.46	37.73



(a) 废墟场景正向越障过程

(a) Obstacle crossing process along forward direction in ruins scenario



(b) 废墟场景反向越障过程

(b) Obstacle crossing process along reverse direction in ruins scenario

图 12 机器人在废墟场景的越障过程

Fig. 12 Processes of the robot moving through the ruins scenario in real-world

表 3 实物废墟场景的量化指标

Tab. 3 Quantitative evaluation results in the ruins scenario in real-world

场景(方向)	$\hat{\theta}_R/\text{rad}$	$ \dot{\theta}_R _{\max}/(\text{rad/s})$	$\hat{\theta}_{\text{roll}}/\text{rad}$	$ \dot{\theta}_{\text{roll}} _{\max}/(\text{rad/s})$	$t_{\text{cost}}/\text{s}$
废墟场景(正向 1)	8.55	1.43	8.42	1.00	71.56
废墟场景(正向 2)	8.88	1.62	8.52	1.59	77.60
废墟场景(正向 3)	8.24	1.89	8.07	1.29	69.00
废墟场景(反向 1)	8.64	1.14	9.02	1.65	72.60
废墟场景(反向 2)	7.89	1.33	8.57	1.83	68.30
废墟场景(反向 3)	7.38	1.19	7.89	1.45	72.10

### 4 结论

本文提出了一种适用于障碍地形的搜救机器人越障构型规划方法。该方法以高效的姿态预测算法为核心,将不规则的地形剖面描述为离散的点云,并建立了履带机器人单侧姿态预测问题模型;进一步设计了该问题的解析求解算法,使得求解时间大为缩短;结合动态规划方法与滚动优化思想,构建了姿态预测—状态评价—构型规划三个步骤滚动进行的越障构型规划算法,并针对非对称地形设计了构型规划策略,获得了优化的机器人越障构型序列和对应的摆臂动作。最后在多个场景中进行了实验,并与深度强化学习方法进行了对比,仿真与实物实验结果表明该算法能够使机器人在复杂地形上实现优异的越障效果。在未来的工作中,将研究滚转角较大时机器人整体姿态预测算法,以更好地适应非对称地形;进一步提高姿态预测算法与构型规划算法的协同程度,探索效率更高、适用性更强的自主越障方法。

### 参考文献 (References)

[1] YANCO H A, NORTON A, OBER W, et al. Analysis of human-robot interaction at the DARPA robotics challenge

trials [ J ]. Journal of Field Robotics, 2015, 32 ( 3 ): 420 - 444.

[2] GIANNI M, FERRI F, MENNA M, et al. Adaptive robust three-dimensional trajectory tracking for actively articulated tracked vehicles [ J ]. Journal of Field Robotics, 2016, 33(7): 901 - 930.

[3] HOWARD T, PIVTORAIKO M, KNEPPER R A, et al. Model-predictive motion planning: several key developments for autonomous mobile robots [ J ]. IEEE Robotics & Automation Magazine, 2014, 21(1): 64 - 73.

[4] SINGH S, JADHAV B D, KRISHNA K M. Posture control of a three-segmented tracked robot with torque minimization during step climbing [ C ]//Proceedings of 2014 IEEE International Conference on Robotics and Automation (ICRA), 2014: 4200 - 4207.

[5] YUAN Y J, XU Q W, SCHWERTFEGER S. Configuration-space flipper planning on 3D terrain [ C ]//Proceedings of 2020 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), 2020: 318 - 325.

[6] ENDO D, WATANABE A, NAGATANI K. Stair climbing control of 4-degrees-of-freedom tracked vehicle based on internal sensors [ C ]//Proceedings of 2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), 2016: 112 - 117.

[7] PECKA M, ZIMMERMANN K, PETRLÍK M, et al. Data-driven policy transfer with imprecise perception simulation [ J ]. IEEE Robotics and Automation Letters, 2018, 3(4): 3916 - 3921.

[8] AZAYEV T, ZIMMERMANN K. Autonomous state-based

- flipper control for articulated tracked robots in urban environments [J]. *IEEE Robotics and Automation Letters*, 2022, 7(3): 7794 – 7801.
- [9] MITRIAKOV A, PAPADAKIS P, KERDREUX J, et al. Reinforcement learning based, staircase negotiation learning; simulation and transfer to reality for articulated tracked robots [J]. *IEEE Robotics & Automation Magazine*, 2021, 28(4): 10 – 20.
- [10] 潘海南, 陈柏良, 黄开宏, 等. 基于深度强化学习的履带机器人摆臂控制方法 [J/OL]. *系统仿真学报*, 2023: 1 – 11 (2023 – 02 – 16) [2023 – 02 – 25]. <https://doi-org-s.libyc.nudt.edu.cn/443/10.16182/j.issn100473x.joss.22-1105>.
- PAN H N, CHEN B L, HUANG K H, et al. Flipper control method based on deep reinforcement learning for tracked robot [J]. *Journal of System Simulation*, 2023: 1 – 11 (2023 – 02 – 16) [2023 – 02 – 25]. <https://doi-org-s.libyc.nudt.edu.cn/443/10.16182/j.issn100473x.joss.22-1105>. (in Chinese)
- [11] NOROUZI M, MIRO J V, DISSANAYAKE G. Planning stable and efficient paths for reconfigurable robots on uneven terrain [J]. *Journal of Intelligent & Robotic Systems*, 2017, 87(2): 291 – 312.
- [12] FABIAN S, KOHLBRECHER S, VON STRYK O. Pose prediction for mobile ground robots in uneven terrain based on difference of heightmaps [C]//*Proceedings of 2020 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2020: 49 – 56.
- [13] SHI C H, HUANG K H, YU Q H, et al. Extrinsic calibration and odometry for camera-LiDAR systems [J]. *IEEE Access*, 2019, 7: 120106 – 120116.
- [14] DENG W B, HUANG K H, CHEN X, et al. Semantic RGB-D SLAM for rescue robot navigation [J]. *IEEE Access*, 2020, 8: 221320 – 221329.
- [15] PAPAPOPOULOS E, REY D A. The force-angle measure of tipover stability margin for mobile manipulators [J]. *Vehicle System Dynamics*, 2000, 33(1): 29 – 48.
- [16] XU W, CAI Y X, HE D J, et al. FAST-LIO2: fast direct LiDAR-inertial odometry [J]. *IEEE Transactions on Robotics*, 2022, 38(4): 2053 – 2073.
- [17] PECKA M, ZIMMERMANN K, SVOBODA T. Fast simulation of vehicles with non-deformable tracks [C]//*Proceedings of 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017: 6414 – 6419.