

基于积分模型预测控制的全向移动机器人轨迹跟踪研究^①颜佳晴^{②*} 鲁聪达^{③*} 蔡颖杰^{**} 彭翔^{*} 潘婷^{*} 周圣云^{**}

(* 浙江工业大学机械工程学院 杭州 310023)

(** 深圳怡丰机器人科技有限公司 深圳 518116)

摘 要 为了减小全向移动机器人在运行中所产生的稳态误差,提出了一种具有积分作用的模型预测控制方法,即积分模型预测控制(IMPC)。积分模型预测控制不仅能够保留模型预测控制中预测和优化的功能,并且可以抑制系统稳态误差,从而使控制器具有更好的鲁棒性和跟踪能力。为了验证积分模型预测控制的先进性,将积分模型预测控制器与传统模型预测控制器进行性能对比,采用四差速全向移动机器人作为模拟实验模型,测试了2种控制器在3种不同路径下的轨迹跟踪性能。测试结果证明,积分模型预测控制器比传统模型预测控制器拥有更好的跟踪性能,验证了所提出方法的有效性和先进性。

关键词 轨迹跟踪;全向移动机器人;积分模型预测控制(IMPC);稳态误差

0 引言

近年来移动机器人的应用越来越广泛。从港口的集装箱装卸、电子商务包裹分拣、智能化仓储到智能停车,移动机器人开始进入人们的生活并改变了生活方式。因此,移动机器人越来越受到研究人员的关注。目前,移动机器人的研究内容主要包括模型、控制、导航定位、避障、路径规划、点稳定等。^[1-5]

移动机器人的控制是指通过输出控制率来驱动移动机器人按照预设的路径行驶^[6]。模型预测控制(model predictive control, MPC)作为一种现代智能控制,是移动机器人控制算法的研究热点。基于运动学模型的预测控制策略已经应用于移动机器人和自动驾驶汽车的路径跟踪^[6-8]。广义预测控制被用来解决路径跟随问题^[9]。在这些研究中,预测控制仅输出角速度,而线速度则是通过特殊的速度发生器产生。随着研究的深入,当今的模型预测控制不仅是简单的控制速度和角速度,越来越多的研究集中在如何优化 MPC 控制器上。例如,文献[10]

通过将协调器添加到 MPC 控制器,可以通过线性返回移动机器人中的跟踪误差来选择合适的权重参数。文献[11]考虑到摩擦力对 MPC 控制器的影响,设计了一种模型预测控制器进行摩擦补偿,以达到较好的控制效果。

在移动机器人领域,解决摩擦补偿是一个重要的问题。在摩擦补偿的研究中,提出了许多控制方法,提出了模型技术控制中的各种摩擦模型。这些方法可以分为基于模型的技术^[12-14]和基于非模型的技术。在基于模型的补偿方案中,使用静态或动态摩擦模型来缓解由于摩擦效应而导致的性能下降。在基于非模型的补偿方案中,在控制设计中不使用摩擦模型。摩擦补偿是通过适当选择控制增益参数或使用非基于模型的观测器来实现的^[15]。但是,基于摩擦模型的模型预测控制的计算量十分巨大,不能满足实际应用中可靠、稳定、快速响应的要求。非模型的补偿方案中解决摩擦的非线性影响,是基于改进 PID 控制器。除了需要精确的模型外,这些技术还具有补偿摩擦以外干扰的优势^[11]。但

① 国家自然科学基金(51775510,51875525)资助项目。

② 男,1994年生,硕士生;研究方向:移动机器人运动控制;E-mail: 875055094@qq.com

③ 通信作者,E-mail: lcd@zjut.edu.cn
(收稿日期:2020-07-31)

是移动机器人在移动中除了摩擦以外,还有其他外部干扰所形成的稳态误差影响控制器的性能。为了消除模型预测控制中的稳态误差并减少处理器的计算量,并在移动机器人的跟踪控制中获得令人满意的性能,本文设计了一种具有积分作用的模型预测控制器,称为积分模型预测控制(integral model predictive control, IMPC)。本文的主要工作概括如下。

(1) 设计具有积分作用的模型预测控制器。通过在传统模型预测控制器中添加积分作用,消除 MPC 中的稳态误差,并获得更好的轨迹跟踪效果。

(2) 在分析四差速全向移动机器人运动特性基础上,对差速驱动单元进行运动学分析,建立了完整的四差速全向移动机器人的运动学模型。

(3) 将设计的 IMPC 控制器应用于四差速全向移动机器人,与传统 MPC 控制器进行对比测试。

1 全向移动机器人运动学分析

1.1 差动驱动模块的运动学分析

全向移动机器人的测试平台如图 1 所示。与传统全向移动机器人相比,该测试平台使用差速驱动作为驱动形式,通过两个轮子的速度差实现驱动模块的直行或转弯。全向移动机器人通过 4 对差速驱动单元的共同协作实现移动机器人的全向运行,因此称为四差速全向移动机器人。



图 1 全向移动机器人的测试平台

差速驱动模块在 t_i 和 t_{i+1} 两个时刻的位姿如图 2 所示。差速驱动单元左右两个驱动轮之间的距离为 l , 左轮速度为 v_l , 右轮速度为 v_r , 两个相邻时刻所转过的角度为 θ , d 为右轮比左轮多行驶的距离。

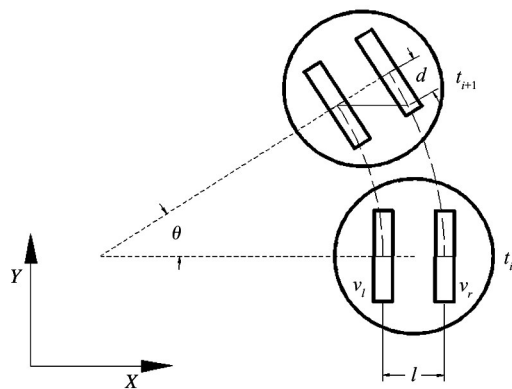


图 2 差速模块相邻两个时刻的位姿

通过几何分析得到差速驱动模块左右轮的速度与其中心点线速度 v 和角速度 ω 之间的关系:

$$\begin{cases} v = \frac{v_r + v_l}{2} \\ \omega = \frac{v_r - v_l}{l} \end{cases} \quad (1)$$

也可以写为

$$\begin{cases} v_r = v + \frac{\omega l}{2} \\ v_l = v - \frac{\omega l}{2} \end{cases} \quad (2)$$

式(1)表示通过差速驱动单元左右驱动轮的速度,可以计算得到差速驱动单元参考点的线速度和角速度。式(2)表示通过差速驱动单元参考点的线速度和角速度,可以计算得出驱动单元的左右轮速。

1.2 全向移动机器人整体分析

图 3 为四差速全向移动机器人的整体结构和坐标系。移动机器人 4 组差速驱动单元均匀分布在车体四周,每组驱动单元有 2 个驱动轮。 $X-Y$ 为全局坐标系, X_R-Y_R 为局部坐标系。设全局坐标系与局部坐标系之间的夹角为 φ 。对差速驱动进行运动学分析,得出单舵轮驱动与差速驱动可以通过式(1)和式(2)实现 v_l, v_r 与 v, ω 之间的相互转换,因此差速驱动等效为舵轮驱动。

假设四差速全向移动机器人 2 个前部驱动模块的速度与转角相同, 2 个后部驱动模块的速度与转角也相同。根据假设,移动机器人前部 2 个差速驱动单元用一个单舵轮替换,用 A 表示;同理,移动机器人后部 2 个差速驱动单元用一个单舵轮替换,用 B 表示; C 表示移动机器人的运动质心(参考点),

A 、 B 两点到 C 点的距离分别由 l_f 和 l_r 表示。 φ 表示移动机器人车头方向, V 表示质心的速度, 速度 V 与车辆轴线所形成的角度用 β 表示。简化后的四差速全向移动机器人模型如图 4 所示, 此模型称为单车模型。

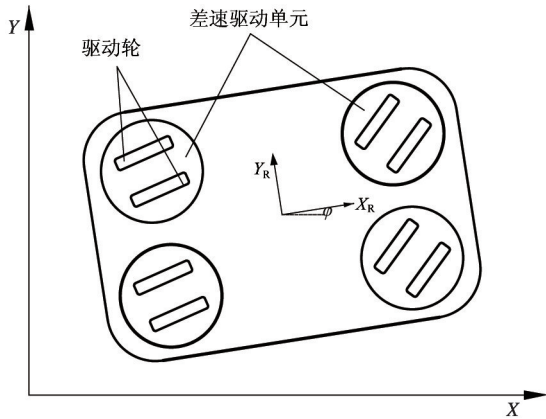


图3 四差速全向移动机器人的整体结构和坐标系

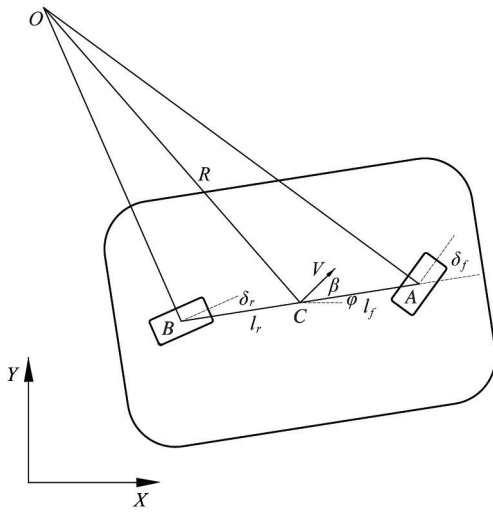


图4 单车模型

通过对单车模型的几何关系分析, 得到:

$$(\tan(\delta_f) - \tan(\delta_r)) \cos(\beta) = \frac{l_f + l_r}{R} \quad (3)$$

因为移动的机器人的角速度表示为 θ , 由此可得出移动机器人角速度为

$$\dot{\varphi} = \frac{V}{R} \quad (4)$$

联立式(3)和式(4), 可以得出:

$$\dot{\varphi} = \frac{V \cos(\beta)}{l_f + l_r} (\tan(\delta_f) + \tan(\delta_r)) \quad (5)$$

因此, 单车模型的运动学方程表示为

$$\begin{cases} \dot{X} = V \cos(\varphi + \beta) \\ \dot{Y} = V \sin(\varphi + \beta) \\ \dot{\varphi} = \frac{V \cos(\beta)}{l_f + l_r} (\tan(\delta_f) + \tan(\delta_r)) \end{cases} \quad (6)$$

式(6)为四差速全向移动机器人的运动学模型。3 个输入量分别是 δ_f 、 δ_r 和 V 。式(7)为 β 角的计算公式, 移动机器人的航向角 φ 可由定位模块提供。

$$\beta = \tan^{-1} \left(\frac{l_f \tan \delta_r + l_r \tan \delta_f}{l_f + l_r} \right) \quad (7)$$

2 积分模型预测控制器设计

积分模型预测控制器的设计主要分为 2 部分。第 1 部分是 MPC 控制器根据参考位姿 $\mathbf{p}_r = [x_r, y_r, \theta_r]^T$ 与 $\mathbf{p} = [x, y, \theta]^T$ 移动机器人当前位姿之间的误差来输出的线速度和角速度。第 2 部分是积分作用设计, 用于补偿移动机器人运行中摩擦等稳态误差。IMPC 的结构如图 5 所示, 外部虚线框为 IMPC, 内部虚线框为传统 MPC。

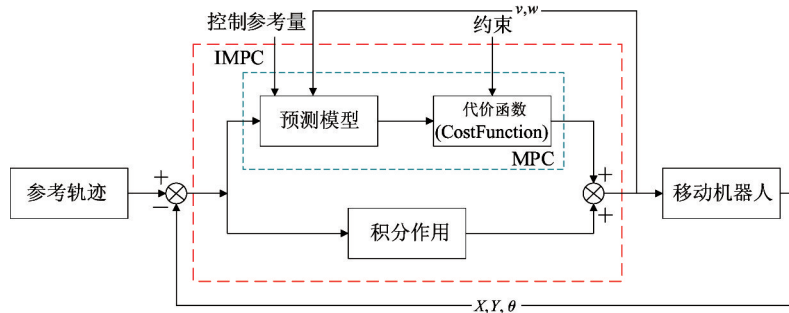


图5 积分模型预测控制器结构图

2.1 模型预测控制器设计

移动机器人运动学方程的一般式可表示为

$$\dot{\mathbf{X}} = f(\mathbf{X}, \mathbf{u}) \quad (8)$$

其上面的每一点都满足上述的运动学方程,用 r 表示参考量,可表示为

$$\dot{\mathbf{X}}_r = f(\mathbf{X}_r, \mathbf{u}_r) \quad (9)$$

其中, $\mathbf{X}_r = [x_r, y_r, \varphi_r]^T, \mathbf{u}_r = [v_r, \omega_r, \beta_r]^T$ 。

在任意点 $(\mathbf{X}_r, \mathbf{u}_r)$ 处泰勒展开,只保留一阶项,忽略高阶项,得到:

$$\begin{aligned} \dot{\mathbf{X}} = f(\mathbf{X}_r, \mathbf{u}_r) + \frac{\partial f(\mathbf{X}, \mathbf{u})}{\partial \mathbf{X}} \bigg|_{\substack{\mathbf{X}=\mathbf{X}_r \\ \mathbf{u}=\mathbf{u}_r}} (\mathbf{X} - \mathbf{X}_r) \\ + \frac{\partial f(\mathbf{X}, \mathbf{u})}{\partial \mathbf{u}} \bigg|_{\substack{\mathbf{X}=\mathbf{X}_r \\ \mathbf{u}=\mathbf{u}_r}} (\mathbf{u} - \mathbf{u}_r) \end{aligned} \quad (10)$$

将式(10)减去式(9),得到:

$$\begin{aligned} \dot{\tilde{\mathbf{X}}} = \begin{bmatrix} \dot{x} - \dot{x}_r \\ \dot{y} - \dot{y}_r \\ \dot{\varphi} - \dot{\varphi}_r \end{bmatrix} = \begin{bmatrix} 0 & 0 & -v_r \sin(\varphi_r + \beta_r) \\ 0 & 0 & v_r \cos(\varphi_r + \beta_r) \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x - x_r \\ y - y_r \\ \varphi - \varphi_r \end{bmatrix} \\ + \begin{bmatrix} \cos(\varphi_r + \beta_r) & 0 & -v_r \sin(\varphi_r + \beta_r) \\ \sin(\varphi_r + \beta_r) & 0 & v_r \cos(\varphi_r + \beta_r) \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} v - v_r \\ \omega - \omega_r \\ \beta - \beta_r \end{bmatrix} \end{aligned} \quad (11)$$

式(11)称为移动机器人的线性误差模型,对式(11)进行离散化处理,得到:

$$\tilde{\mathbf{X}}(k+1) = \mathbf{A}_{k,t} \tilde{\mathbf{X}}(k) + \mathbf{B}_{k,t} \tilde{\mathbf{u}}(k) \quad (12)$$

式中,

$$\begin{aligned} \mathbf{A}_{k,t} = \begin{bmatrix} 1 & 0 & -v_r \sin(\varphi_r + \beta_r) T \\ 0 & 1 & v_r \cos(\varphi_r + \beta_r) T \\ 0 & 0 & 1 \end{bmatrix}, \\ \mathbf{B}_{k,t} = \begin{bmatrix} \cos(\varphi_r + \beta_r) T & 0 & -v_r \sin(\varphi_r + \beta_r) T \\ \sin(\varphi_r + \beta_r) T & 0 & v_r \cos(\varphi_r + \beta_r) T \\ 0 & T & 0 \end{bmatrix}, \end{aligned} \quad \text{为}$$

采样时间。

采用式(13)作为目标函数。

$$\begin{aligned} \phi(k) = \sum_{i=1}^{N_p} \|\boldsymbol{\eta}(k+i|t) - \boldsymbol{\eta}_{ref}(k+i|t)\|_Q^2 \\ + \sum_{i=1}^{N_c-1} \|\Delta \mathbf{U}(k+i|t)\|_R^2 \end{aligned} \quad (13)$$

式中, N_p 为预测时域, N_c 为控制时域。

为了满足式(13)目标函数的形式,需将式(12)

做一定的转换:

$$\boldsymbol{\xi}(k|t) = \begin{bmatrix} \tilde{\mathbf{X}}(k|t) \\ \tilde{\mathbf{u}}(k-1|t) \end{bmatrix} \quad (14)$$

得到一个新的状态空间表达式:

$$\begin{aligned} \boldsymbol{\xi}(k+1|t) &= \tilde{\mathbf{A}}_{k,t} \boldsymbol{\xi}(k|t) + \tilde{\mathbf{B}}_{k,t} \Delta \mathbf{U}(k|t) \\ \boldsymbol{\eta}(k|t) &= \tilde{\mathbf{C}}_{k,t} \boldsymbol{\xi}(k|t) \end{aligned} \quad (15)$$

式中, $\tilde{\mathbf{A}}_{k,t} = \begin{bmatrix} \mathbf{A}_{k,t} & \mathbf{B}_{k,t} \\ \mathbf{0}_{m \times n} & \mathbf{I}_m \end{bmatrix}, \tilde{\mathbf{B}}_{k,t} = \begin{bmatrix} \mathbf{B}_{k,t} \\ \mathbf{I}_m \end{bmatrix}, n$ 为状态向量维度, m 为控制量维度。

经过推导,可以得到系统的预测输出表达式:

$$\mathbf{Y}(t) = \boldsymbol{\psi}_t \boldsymbol{\xi}(t|t) + \boldsymbol{\Theta}_t \Delta \mathbf{U}(t) \quad (16)$$

将式(16)代入目标函数式(13),可得到完整形式的目标函数表达式。

控制量的表达式表示为

$$\begin{aligned} \mathbf{u}_{\min}(t+k) \leq \mathbf{u}(t+k) \leq \mathbf{u}_{\max}(t+k) \\ k = 0, 1, \dots, N_c - 1 \end{aligned} \quad (17)$$

控制增量的表达式表示为

$$\begin{aligned} \Delta \mathbf{u}_{\min}(t+k) \leq \Delta \mathbf{u}(t+k) \leq \Delta \mathbf{u}_{\max}(t+k) \\ k = 0, 1, \dots, N_c - 1 \end{aligned} \quad (18)$$

通过对控制量增量的约束代替对控制量的约束,因此在设计约束条件时也应针对控制增量进行设计,以满足目标函数的要求。因此,需要对式(18)进行转换,求得相应的转换矩阵。

控制量与控制增量之间关系为

$$\mathbf{u}(t+k) = \mathbf{u}(t+k-1) + \Delta \mathbf{u}(t+k) \quad (19)$$

假设:

$$\mathbf{U}_t = \mathbf{1}_{N_c} \otimes \mathbf{u}(k-1) \quad (20)$$

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 1 & 1 & 0 & \cdots & 0 \\ 1 & 1 & 1 & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 1 & 1 & \cdots & 1 & 1 \end{bmatrix} \otimes \mathbf{I}_m \quad (21)$$

其中, $\mathbf{1}_{N_c}$ 为行数为 N_c 的列向量, $\mathbf{I}_m$ 为维度为 m 的单位矩阵, $\otimes$ 为克罗内克积, $\mathbf{u}(k-1)$ 为上一时刻的控制量。

结合式(19)、式(20)和式(21),式(17)可改写为

$$\mathbf{U}_{\min} \leq \mathbf{A} \Delta \mathbf{U}_t + \mathbf{U}_t \leq \mathbf{U}_{\max} \quad (22)$$

其中, $U_{\max}$ 和 $U_{\min}$ 分别为控制时域内的控制量最大值与最小值。

将控制量增量式目标函数转化为二次型形式, 结合约束条件, 转化为式(22)的优化问题。

$$\begin{aligned} \phi(\xi(t), u(t-1), \Delta U(t)) = & [\Delta U(t)^T, \varepsilon]^T H_t [\Delta U(t)^T, \varepsilon] + G_t [\Delta U(t)^T, \varepsilon] \\ \text{s. t. } \Delta U_{\min} \leq \Delta U_t \leq \Delta U_{\max} \\ U_{\min} \leq A \Delta U_t + U_t \leq U_{\max} \end{aligned} \quad (23)$$

$$\text{式中, } H_t = \begin{bmatrix} \Theta_t^T Q \Theta_t + R & 0 \\ 0 & \rho \end{bmatrix}, G_t = [2e_t^T Q \Theta_t \quad 0], e_t$$

为跟踪误差。

对式(23)进行求解, 可得到一系列控制量增量, 如式(24)所示。

$$\Delta U_t^* = [\Delta u_t^*, \Delta u_{t+1}^*, \dots, \Delta u_{t+N_c-1}^*]^T \quad (24)$$

将式(24)中的第1个元素作为控制增量输入系统, 可得到:

$$u(t) = u(t-1) + \Delta u_t^* \quad (25)$$

进入下一个控制周期后, 不断重复以上过程, 实现四差速全向移动机器人的轨迹跟踪控制。

2.2 积分作用设计

实际应用中, 由于移动机器人存在运行中的摩擦等外部因素影响, 整个系统存在稳态误差。因此, 通过对累积误差增加积分作用, 即补偿移动机器人的摩擦和其他外部影响, 使移动机器人获得更好的响应性和鲁棒性。

积分作用可累积历史误差, 但传统的积分控制在启动时会在很短的时间内对系统输出造成较大的偏差, 很容易引起积分累积, 导致控制量超过临界值。该值对应于执行器允许的最大动作范围, 这会导致较大的过冲甚至震荡。使用改进的积分器校正移动机器人的航向角误差和位置误差, 最后将角速度输入到移动机器人。系统启动时, 将使用模型预测控制来使系统稳定。当控制量接近给定值时, 即当移动机器人的当前点接近目标点时, 加入积分作用消除稳态误差, 提高控制系统精度和鲁棒性。

积分作用表示为

$$u_l(t) = K \sum_{n=1}^T e(t) \quad (26)$$

移动机器人的误差由航向误差和位置误差组

成, 得到:

$$u_l(t) = K \sum_{n=1}^T (e_l(t) + e_h(t)) \quad (27)$$

本文通过预测模型、滚动优化和反馈矫正3个步骤计算, 得出了传统模型预测控制器的控制量 $u_M(t)$ 。

$$u_M(t) = u_M(t-1) + \Delta u_t^* \quad (28)$$

式(28)表示当前控制量等于上一时刻的控制量加上控制量的增量。在加入积分作用后, 得到:

$$\begin{aligned} u(t) = u_M(t) + u_l(t) = u_M(t-1) \\ + \Delta u_t^* + K \sum_{n=1}^T (e_l(t) + e_h(t)) \end{aligned} \quad (29)$$

本文通过积分作用校正移动机器人的航向角误差和位置误差, 系统启动时, 将使用模型预测控制来使系统稳定, 当控制量接近给定值时, 即当移动机器人的当前点接近目标点时, 采用积分控制消除稳态误差, 根据所设定的积分器的作用时间, 将式(29)进行进一步优化。

$$u(t) = \begin{cases} u_M(t-1) + \Delta u_t^* & e > e_{ref} \\ u_M(t-1) + \Delta u_t^* + K \sum_{n=1}^T (e_l(t) + e_h(t)) & e < e_{ref} \end{cases} \quad (30)$$

式中, e_{ref} 表示误差临界值。至此, 式(30)为本文所设计的积分模型预测控制器。

3 实验

3.1 积分模型预测控制器实验设计和过程

图6是进行轨迹跟踪实验测试的场地。测试场地长约35.5 m, 宽约16 m, 地面平整, 无较大的起伏和缝隙, 摩擦系数较高。在场地柱子周边和沿墙壁一侧都安装有激光反射柱, 为激光导航提供定位信息。

图7是通过激光定位模块所绘制出的测试场地激光特征图, 定位模块可以将环境中的绝大部分信息扫描进地图, 例如反光板、墙面、柱子等。在移动机器人进行定位时, 定位模块可以根据当前位置信息与所扫描的特征地图进行匹配, 进而确定移动机器人的当前位置, 提供给移动机器人准备的位置信息。



图6 测试场地

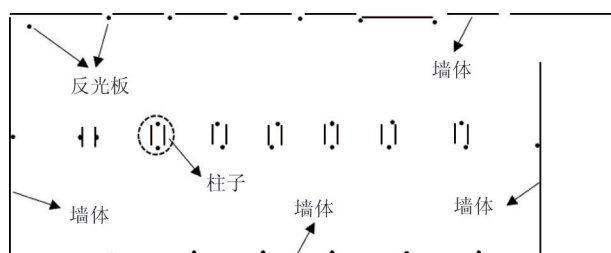


图7 测试场地特征图

实验前确定全向移动机器人在环境中的当前位置,找到与当前位置最近的参考点并初始化。通过调试个人电脑可以给移动机器人下发本地任务,移动机器人根据任务信息搜索最近路径并开始移动,在移动过程中会不断记录当前位置与理论位置之间的位置误差和航向角误差,输出外部文本文件。积分模型预测控制器则会不断优化输出,不断减小机器人行进中的误差。

将设计的积分模型控制器应用于四差速全向移动机器人平台进行轨迹跟踪测试,验证四差速移动机器人平台的可行性,也验证积分模型预测控制在实际移动机器人运动轨迹跟踪控制中的有效性。在本文测试实验中设计3种不同的轨迹进行测试,分别是直线、圆角矩形和“8”字形,综合分析移动机器人模型和模型预测控制算法的性能,图8所示为移动机器人正进行圆角矩形轨迹跟踪测试。

3.2 积分模型预测控制实验数据分析

3.2.1 直线轨迹

直线轨迹跟踪是最基本的轨迹跟踪问题之一。在移动机器人的移动过程中,能够以最快的速度运行,而直线运行时发生的抖动也最容易观察到。因

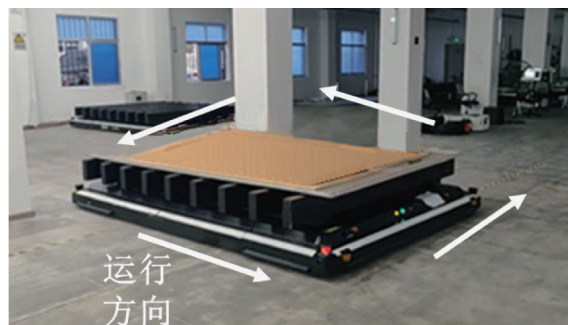


图8 移动机器人圆角矩形测试

此,IMPC 控制的稳定性和鲁棒性通过直线路径验证。

在直线轨迹测试中,移动机器人以 1.2 m/s 的固定速度自主行进了 14.5 m 长的直线路径。图9为两种算法在直线测试中的跟踪结果,实线为参考轨迹,虚线为 IMPC 跟踪轨迹,点线为 MPC 跟踪轨迹,从图9中观察到两种算法都能顺利按照预设轨迹行进。通过对直线跟踪结果进行局部放大,观察到两种算法的跟踪效果,IMPC 跟踪轨迹与 MPC 跟踪轨迹相比,更加靠近参考轨迹,即误差更小。图10和图11显示了 IMPC 和 MPC 在直线路径上的横向误差和航向误差的比较,其中点划线为 MPC 误差,实线为 IMPC 误差。从图10和图11中明显观察到,IMPC 算法能够大大减小移动机器人行驶中的横向误差和航向误差。在直线运行时,传统的 MPC 控制器的横向误差稳定在 -10 mm,航向误差稳定在 0.7° 。在加入积分作用后,IMPC 可以大幅减小

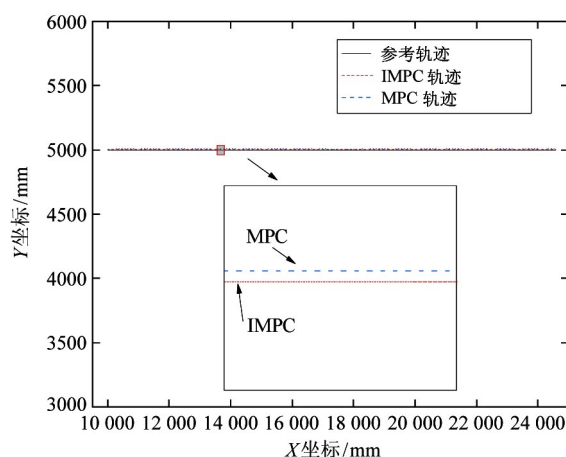


图9 直线测试跟踪结果

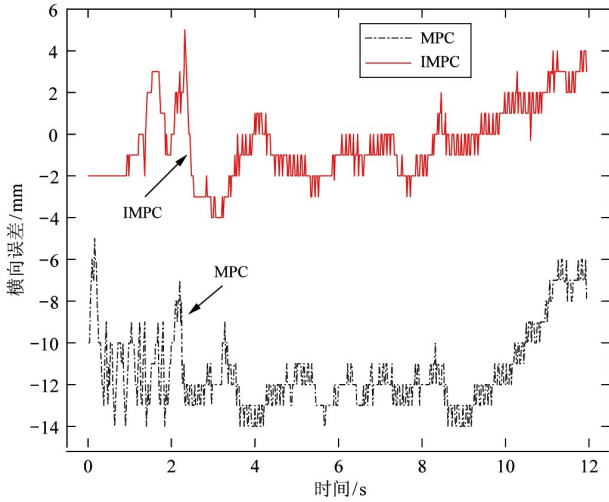


图 10 直线测试横向误差

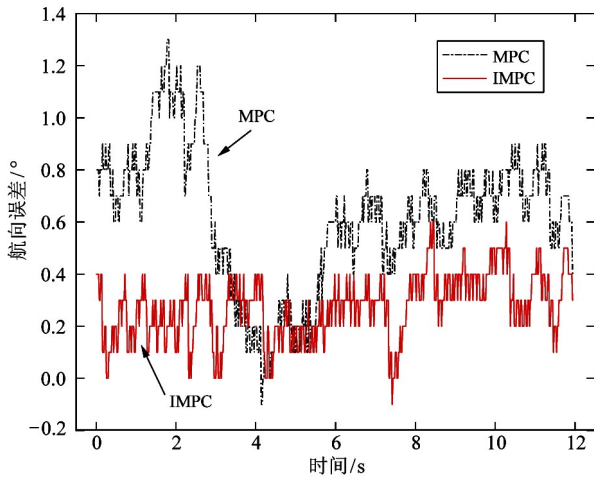


图 11 直线测试航向误差

系统的稳态误差,横向误差在 0 mm 附近波动,航向误差在 0°附近波动。

3.2.2 圆角矩形

在圆角矩形测试中,移动机器人以 1 m/s 的平均速度自动行进 36 m 长的圆角矩形路径。图 12 为两种算法在相同路径下的跟踪结果,从图 12 中观察到两种算法都能顺利按照预设轨迹行进。通过对圆角矩形左上部分进行局部放大,观察到两种算法的跟踪效果,其中实线为参考轨迹,虚线为 IMPC 跟踪轨迹,点线为 MPC 跟踪轨迹,IMPC 跟踪轨迹与 MPC 跟踪轨迹相比,更加靠近参考轨迹。图 13 和图 14 显示了 IMPC 和 MPC 在圆角矩形路径上的横向误差和航向误差的比较,其中点划线为 MPC 误差,实线为 IMPC 误差。很明显,当运行一个圆角矩形时,

IMPC 的横向误差稳定在 5 mm 左右,航向误差稳定在 0.2°左右,因此 IMPC 具有更好的跟踪性能。

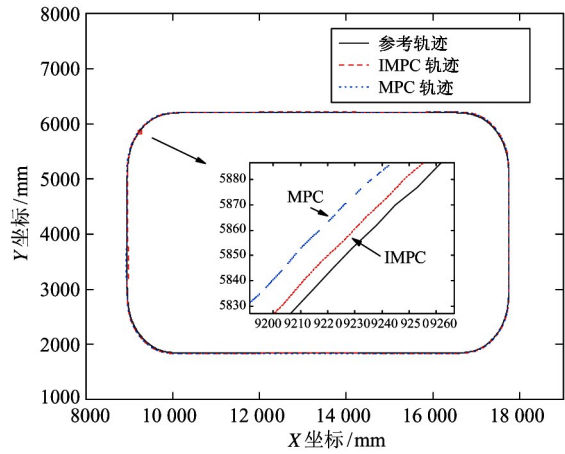


图 12 圆角矩形测试跟踪结果

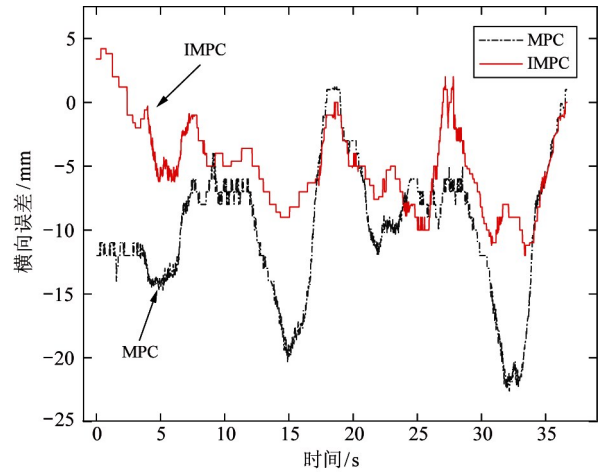


图 13 圆角矩形测试横向误差

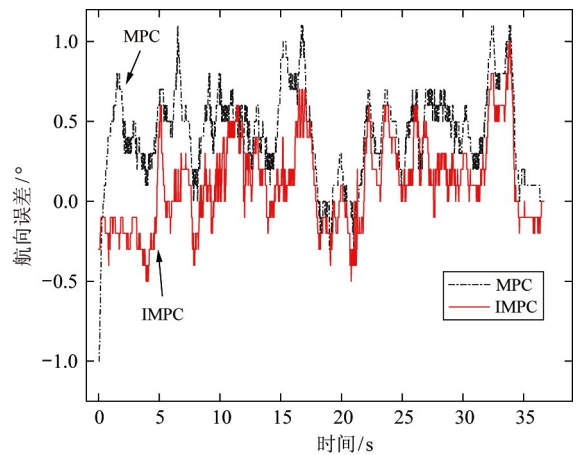


图 14 圆角矩形测试航向误差

3.2.3 8 字曲线

在 8 字形曲线测试中,移动机器人以 0.8 m/s 的速度自动行驶 28 m 长的 8 字形路径。图 15 显示了 8 型路径实验的 IMPC 和 MPC 控制器的路径跟踪结果,其中图中实线为参考轨迹,虚线为 IMPC 跟踪轨迹,点线为 MPC 跟踪轨迹。通过对跟踪结果的局部放大,观察到 IMPC 算法的实际路径更接近参考路径。图 16 和图 17 显示了 IMPC 和 MPC 在圆角矩形路径上的横向误差和航向误差的比较,其中点划线为 MPC 误差,实线为 IMPC 误差。通过图 16 和图 17 观察到,IMPC 控制器可以显著减少运动过程中移动机器人的航向误差和横向误差,并且误差波动较小、驾驶更加稳定。加上积分作用后,航向误差的最大值为 0.5° ,最小误差为 0° 。横向误差的最大值为 -4.7 mm ,最小误差为 -1.1 mm 。平均误差为 0.2° 和 -3.2 mm 。

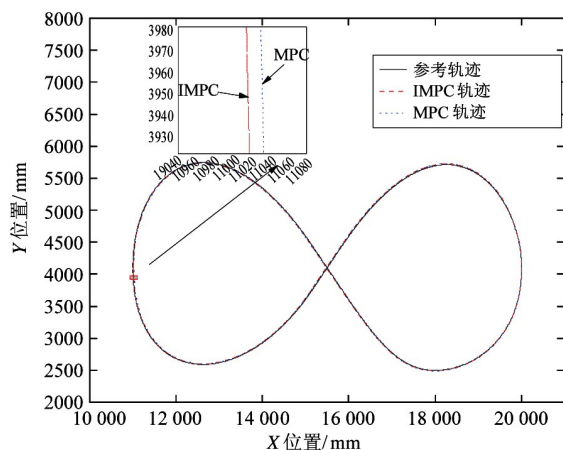


图 15 8 字形测试跟踪结果

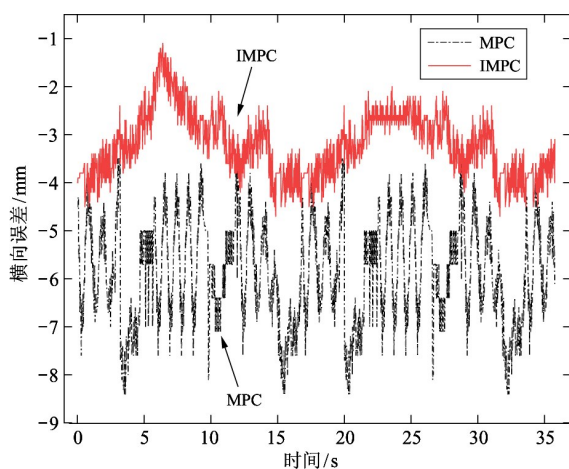


图 16 8 字形测试横向误差

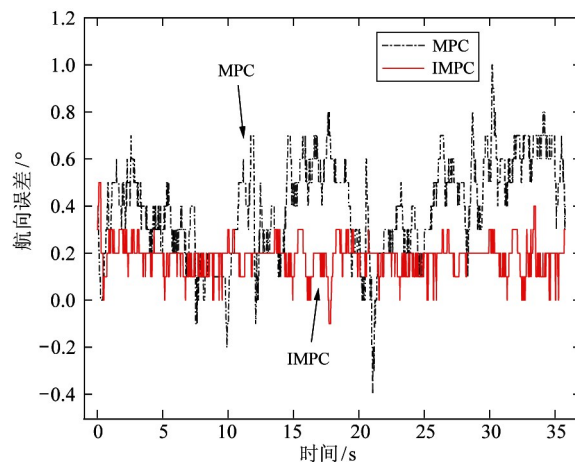


图 17 8 字形测试航向误差

4 结论

本文提出了一种积分模型预测控制方法,并以四差速全向移动机器人作为研究对象,对移动机器人的轨迹跟踪控制进行了研究分析,利用机器人学、运动学、控制理论和计算机学等理论知识,结合大实车实验等综合性研究,设计出了跟踪性能强、鲁棒性好的轨迹跟踪系统。首先,建立运动学模型并线性化非线性移动机器人模型;其次,设计积分模型预测控制,包括 MPC 设计和积分作用设计;最后,测试了直线、圆角矩形和 8 字形 3 个不同轨迹的实验。实验结果表明,与传统 MPC 相比,IMPC 的横向误差和航向误差较小,实际路径更接近参考路径,具有更好的跟踪性和稳定性。

参考文献

- [1] Liu K, Gao H, Ji H, et al. Adaptive sliding mode based disturbance attenuation tracking control for wheeled mobile robots[J]. *International Journal of Control Automation and Systems*, 2019, 18(5): 1288-1298
- [2] Sun Q, Tian Y, Diao M. Cooperative localization algorithm based on hybrid topology architecture for multiple mobile robot system[J]. *IEEE Internet of Things Journal*, 2018, 5(6): 4753-4763
- [3] Yang H, Fan X, Shi P, et al. Nonlinear control for tracking and obstacle avoidance of a wheeled mobile robot with nonholonomic constraint[J]. *IEEE Transactions on Control Systems Technology*, 2015, 24(2): 103-125
- [4] He X, Geng Z. Arbitrary point-to-point stabilization control in specified finite time for wheeled mobile robots

- based on dynamic model [J]. *Nonlinear Dynamics*, 2019, 97(2): 937-954
- [5] 占伟, 屈军锁, 芦鑫, 等. 基于改进蚁群算法的移动机器人全局路径规划[J]. *现代电子技术*, 2018, 41(24):180-183
- [6] Raffo G V, Gomes G K, Normey-Rico J E, et al. A predictive controller for autonomous vehicle path tracking [J]. *IEEE Transactions on Intelligent Transportation Systems*, 2009, 10(1): 92-102
- [7] Klančar G, Škrjanc I. Tracking-error model-based predictive control for mobile robots in real time[J]. *Robotics and Autonomous Systems*, 2007, 55(6):460-469
- [8] 黎伟豪. 全向移动机器人的控制算法及其运动规划[D]. 广州:华南理工大学自动化科学与工程学院, 2019
- [9] 曹阳, 贺登博, 喻凡, 等. 基于主动转向的车辆路径跟随广义预测控制[J]. *上海交通大学学报*, 2016, 50(3):401-406
- [10] Jardine P T, Kogan M, Givigi S N, et al. Adaptive predictive control of a differential drive robot tuned with reinforcement learning[J]. *International Journal of Adaptive Control and Signal Processing*, 2019, 33(2):410-423
- [11] Barreto S J C L, Conceicao A G S, Dorea C E T, et al. Design and implementation of model-predictive control with friction compensation on an omnidirectional mobile robot [J]. *IEEE/ASME Transactions on Mechatronics*, 2014, 19(2):467-476
- [12] 王圣翔, 任超, 马书根. 含有摩擦补偿的全方位移动机器人自抗扰控制[J]. *控制理论与应用*, 2017, 34(10):1285-1292
- [13] Ren C, Li X, Yang X, et al. Extended state observer based sliding mode control of an omnidirectional mobile robot with friction compensation[J]. *IEEE Transactions on Industrial Electronics*, 2019, 66(12):9480-9489
- [14] Wang S X, Ren C, Ma S G. Active disturbance rejection control with friction compensation for an omnidirectional mobile robot[J]. *Control Theory and Applications*, 2017, 34(10):1285-1292
- [15] Ren C, Li X, Yang X, et al. Extended state observer based sliding mode control of an omnidirectional mobile robot with friction compensation[J]. *IEEE Transactions on Industrial Electronics*, 2019, 24(4): 245-256

Research on trajectory tracking of omnidirectional mobile robot based on integral model predictive control

Yan Jiaqing*, Lu Congda*, Cai Yingjie**, Peng Xiang*, Pan Ting*, Zhou Shenyun**

(* College of Mechanical Engineering, Zhejiang University of Technology, Hangzhou 310023)

(** Shenzhen Yeeffing Robotics Co. Ltd, Shenzhen 518116)

Abstract

In order to reduce the steady-state errors caused by an omnidirectional mobile robot during movement, a model predictive control method with integral function is proposed, which is called integral model predictive control (IMPC). Integral model predictive control not only preserves the predictive and optimization features of model predictive control, but also suppresses the steady-state error of the system, so that the controller has better robustness and tracking ability. In order to verify the advanced nature of the integral model predictive control, the performance of the integral model predictive controller is compared with the traditional model predictive controller. The trajectory tracking performance of the two controllers under three different paths is tested using a four-differential omnidirectional moving robot as a simulated experimental model. The test results prove that the integral model prediction controller has better tracking performance than the traditional model prediction controller, which validates the effectiveness and sophistication of the proposed method.

Key words: trajectory tracking, omnidirectional mobile robot, integral model predictive control (IMPC), steady-state error