智能工程

目 录

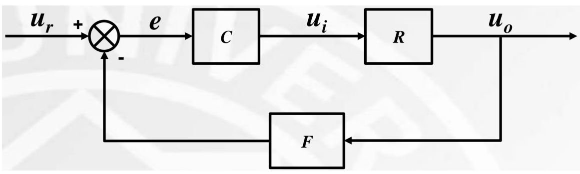
返回目录

图 片

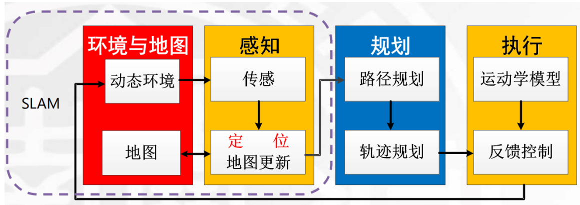
表 格

要 点

1 基础知识



(a) 课程内容



(b) 工作流程

图 1: 课程内容

表 1: 课程内容

	u_i	u_o	R	F	u_r	e	C
概念	系统输入	系统输出	系统模型	反馈单元	系统给定	系统误差	控制器
含义	能对被控对象施加作用的手段	作业目标相应的可测系统状态	系统输入输出映射	系统输出映射变换	系统作业目标	作业目标与系统当前测量状态差值	系统误差与输入映射
内容	机器人运动学			机器人控制		机器人感知	机器人轨迹规划

课程内容

课程案例 移动机器人->轮式机器人->两轮差速机器人。

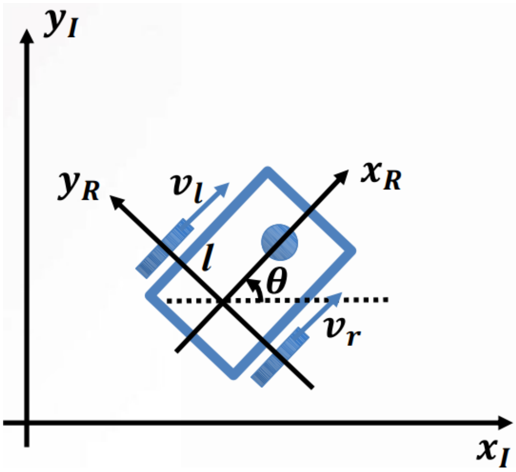


图 2: 两轮差速机器人模型

- 车轮半径r。
- 两轮转速φ_l, φ_r：$v_i = \varphi_i r$。
- 车轮到两轮中间点距离l。

1. 求正运动学模型3.3.4。
2. 设计运动控制器4.1。

2 机器人运动形态

2.1 移动机器人

自然界运动形态特点

- 能量利用率高。
- 适应野外复杂环境。
- 与身体尺寸、结构相适应。
- 运行速度高。

机器人实现自然界运动形态问题

- 机械结构、能量密度、感知与控制决策能力困难。
- 安全性、可靠性差。
- 成本高。
- 于人造环境低效。

运动（LOCOMOION） 机器人与环境的物理交互方式。

- 稳定性。
- 接触特性。
- 环境特性。

2.2 腿式机器人

2.2.1 腿式机器人

研究意义

- 复杂恶劣环境的高适应性。
- 点接触的高通过能力。
- 高实现难度：系统控制多自由度，实时感知环境。

腿数影响

- 机构复杂度。
- 控制复杂度。
- 环境适应性：腿越多，通过性越好，环境适应性越强。
- 系统稳定性 [1]：腿数增加，由动态稳定向静态稳定过度。
  - 动态稳定：执行器停止工作摔倒。运动过程中通常半数腿离地。
  - 静态稳定：执行器停止工作不摔倒。点接触需保证三腿同时着地，面接触需保证一条腿着地。

返回目录

运动规划

- 运动学分析。
- 动力学分析。

步态 [2] 一个行进周期内各腿抬落组合，k腿机器人的步态模式数量为$N = (2^k - 1)!$。

单位距离能耗（COT） $\text{cot} = \frac{\text{消耗能量}}{\text{重量}\times\text{运行距离}}$。

2.2.2 四足机器人

- 点接触：每条腿至少需要两个自由度，执行器较少，没有冗余。
- 行走（静态平衡）：一次移动一条腿，剩下腿支持身体，重心落在支持多边形内。适合攀爬，速度低，能效低。
- 奔跑（动态平衡）：一次移动多条腿，平衡建立在周期运动上。速度高，能效高，需要实时控制与执行。

2.2.3 双足机器人

两种方案

- 静态稳定（日本）：面接触，重心左右变换，速度低，环境适应性差，能效低。
- 动态稳定（美国）：点接触，重心适时调整，速度高，环境适应性强，能效高。

动态稳定运动机理 [3]

- 倒立摆模型：类似纯滚动，步距越小越趋于圆。步态不自然，重心变化（需做功），落地冲击大。
- 无源动态行走：摆动与向前摔落结合，势能转化为动能。
- 弹簧负载倒立摆（SLIP）：仿照动物腿肌肉，增加弹簧来缓冲并储存能量，运动具有对称性。其周期往复运动的动态稳定性可由庞加莱变换线性化后验证，条件为$\lambda < 1$（见PPT.2.34-43）。
- 串联弹性驱动（SEA）：更为高效，更符合生物的自然属性，基于运动学的位置控制，基于动力学的力矩控制。可由其获得稳定平台（见PPT.2.48-50）。

2.3 轮式机器人

研究意义

- 人造环境下的高效性：滚动摩擦，无重心起伏。
- 结构简单，可靠性高，成本低。
- 控制简单，系统复杂度低。

轮数对稳定性的影响 轮数增加，由动态稳定向静态稳定过度。

- 动态稳定：执行器停止工作摔倒。倒立摆模型。
- 静态稳定：执行器停止工作不摔倒。陀螺效应，随动轮效应。

3 机器人运动学

3.1 运动学模型

表征机器人驱动（输入）和机器人空间位姿（输出）关系。
机械臂运动学模型与移动机器人运动学模型的区别：

- 机械臂本体坐标系固定，精度高；移动机器人本体坐标系随动，精度低。
- 非完整约束 [4] ：移动机器人只知道码盘变化量无法获取位姿，其状态取决于路径。这来源于不可积的微分约束（比如车轮的侧向滑动约束）。
- 微分运动学（Differential Kinematics）：速度空间替代位置空间。

3.2 车轮

3.2.1 类型

[5]

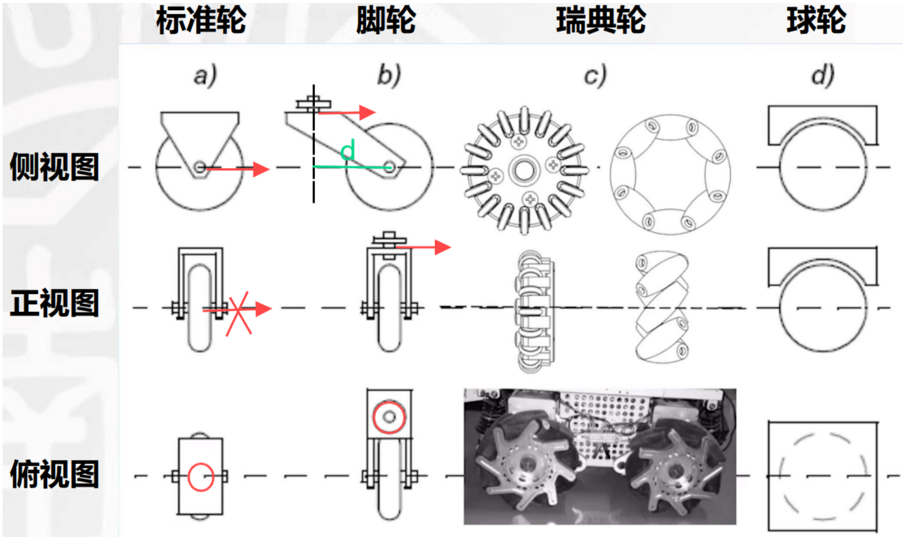


图 3: 车轮类型

标准轮（STANDARD WHEEL）

- 两个自由度：沿轮平面滚动+沿垂直轴转动。
  一个约束：沿轮轴滑动。
- 分类
  - 标准固定轮：无法旋转，只有一个自由度。
  - 标准转向轮（舵轮）。

脚轮（CASTOR WHEEL）

- 三个自由度：沿轮平面滚动+沿垂直轴转动+沿路轴运动。
  无约束。
- 偏心距d：触地点到垂直旋转轴距离。
- 扭矩压力，易损坏。

瑞典轮（SWEDISH WHEEL）

- 三个自由度：沿轮平面滚动（被动）+沿轮轴转动（主动）+沿垂直轴转动（被动）。
  无约束。

返回目录

- 麦克纳姆轮（Macanum wheel）：45°，至少需要4个共同使用。
  连续切换轮：90°，至少需要3个共同使用。
- 对地面冲击大，噪音大，易损坏，成本高。

球轮（SPHERICAL WHEEL）

- 三个自由度（全主动）：沿两个正交轮轴转动+沿垂直轴转动。
  无约束。
- 成本高，可靠性差。

3.2.2 选取

- 数量：至少三轮同时着地，才能保证静态稳定性。四轮可以提升稳定性，但需要适当的悬架系统。
- 大小：越大的轮子通过性越好，但需要更大的扭矩。
- 多数形态都有非完整约束。

3.3 运动学建模

3.3.1 空间描述与状态表达

坐标系

- 惯性参考坐标系I：作业目标、控制指令、传感器感知测量信息。
- 机器人参考坐标系R：控制器误差输入、控制器控制指令。
- 笛卡尔坐标系：右手法则。

位姿（POSE）

位置空间求导得到速度空间：

$$\xi_I = \begin{bmatrix} x_I \\ y_I \\ \theta_I \end{bmatrix}, \xi_R = \begin{bmatrix} x_R \\ y_R \\ \theta_R \end{bmatrix} \xRightarrow{\text{求导}} \dot{\xi}_I = \begin{bmatrix} \dot{x}_I \\ \dot{y}_I \\ \dot{\theta}_I \end{bmatrix}, \dot{\xi}_R = \begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta}_R \end{bmatrix}$$

返回目录

惯性参考坐标系旋转得到机器人参考坐标系：

$$\dot{\xi}_R = R\theta\dot{\xi}_I$$

旋转阵 $R(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$ 为单位正交阵，$R^T = R^{-1}$。

3.3.2 *ICR*法

6

瞬时旋转/曲率中心（ICR） 刚体上各点角速度相同。

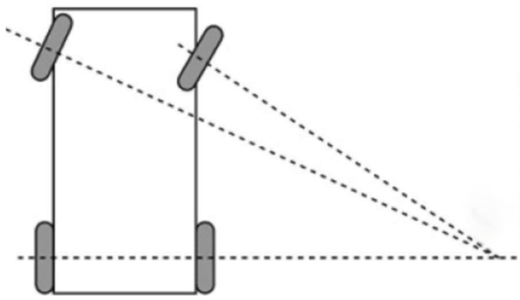


图 4: 瞬心

步骤

- 坐标系变换。
- 确定约束。
- 计算瞬心：各轮轮轴到该点距离与速度成正比。
- 求解$\xi_R = \begin{bmatrix} \dot{x}_R & \dot{y}_R & \dot{\theta}_R \end{bmatrix}^T$。

3.3.3 约束方程法

7

要求 在水平面上运动，车轮与地面点接触，不变形，安装在钢体表面，舵机转轴与地面垂直。

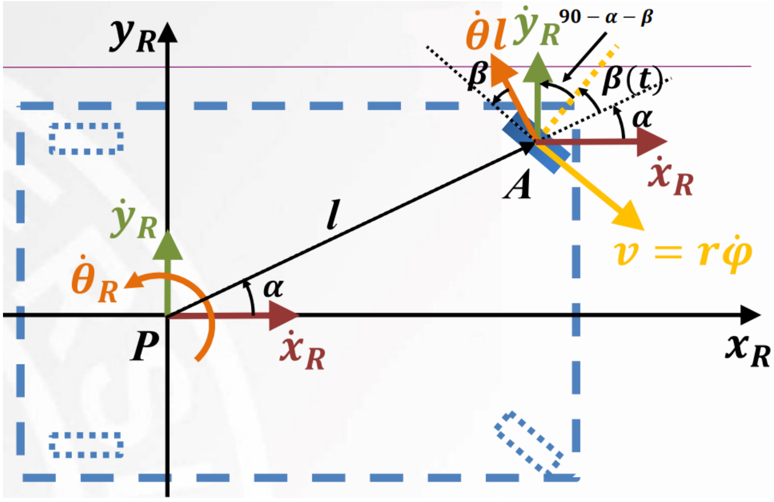


图 5: 标准轮约束方程

标准轮

- 纯滚动：$\begin{bmatrix}\sin(\alpha+\beta(t)) & -\cos(\alpha+\beta(t)) & -l\cos\beta(t)\end{bmatrix} R\theta\dot{\xi}_I = r\dot{\varphi}$。
- 无滑动：$\begin{bmatrix}\cos(\alpha+\beta(t)) & \sin(\alpha+\beta(t)) & l\sin\beta(t)\end{bmatrix} R\theta\dot{\xi}_I = 0$。
- 主动轮两个约束，随动轮只有滑动约束。

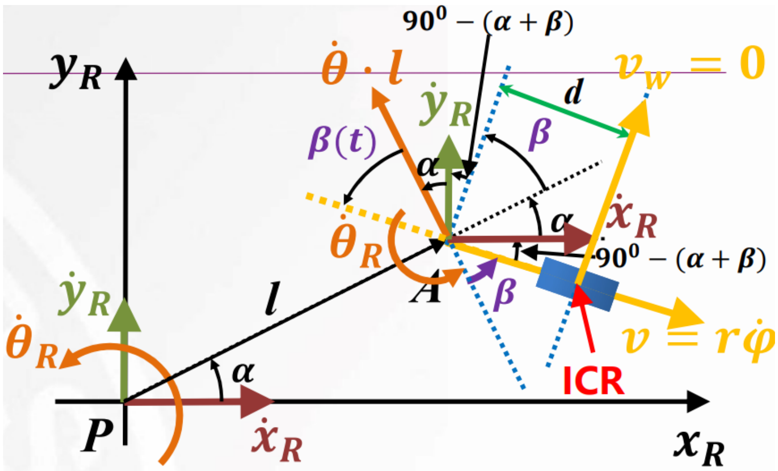


图 6: 脚轮约束方程

脚轮

- 纯滚动：$\begin{bmatrix}\sin(\alpha+\beta) & -\cos(\alpha+\beta) & -l\cos\beta\end{bmatrix} R\theta\dot{\xi}_I = r\dot{\varphi}$。

返回目录

- 无滑动：$\begin{bmatrix}\cos(\alpha+\beta) & \sin(\alpha+\beta) & d+l\sin\beta\end{bmatrix} R\theta\dot{\xi}_I = -d\dot{\beta}$。
- 主动轮两个约束，随动轮无约束。

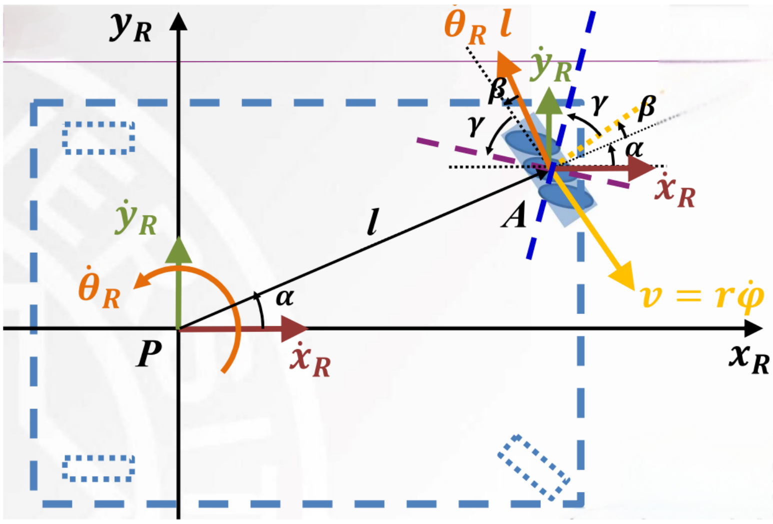


图 7: 瑞典约束方程

瑞典轮

- 纯滚动：$\begin{bmatrix}\cos(\alpha+\beta+\gamma) & \sin(\alpha+\beta+\gamma) & l\sin(\beta+\gamma)\end{bmatrix} R\theta\dot{\xi}_I = r\dot{\varphi}\sin\gamma + r_{sw}\dot{\varphi}_{sw}$。
- 无滑动：$\begin{bmatrix}\sin(\alpha+\beta+\gamma) & -\cos(\alpha+\beta+\gamma) & -l\cos(\beta+\gamma)\end{bmatrix} R\theta\dot{\xi}_I = r\dot{\varphi}\cos\gamma$。
- 需满足主动轮小轮两个约束，随动轮无约束。

使用 根据各轮主/随动状态列运动约束方程，得到最多三个独立约束方程（对应平面的三维位姿）。

以下以N个标准轮（N_f个固定，N_s个转向）的机器人为例：

- 滚动约束

$$J_1(\beta_s)R(\theta)\dot{\xi}_I - J_2\dot{\varphi} = 0$$

其中 $J_1(\beta_s) = \begin{bmatrix} J_{1f(N_f\times 3)} \\ J_{1s}(\beta_s)_{(N_s\times 3)} \end{bmatrix}, \varphi(t) = \begin{bmatrix} \varphi_f(t) \\ \varphi_s(t) \end{bmatrix}, J_2 = \mathrm{diag}(r_1, \cdots, r_N)$为轮径对角阵。

- 滑动约束

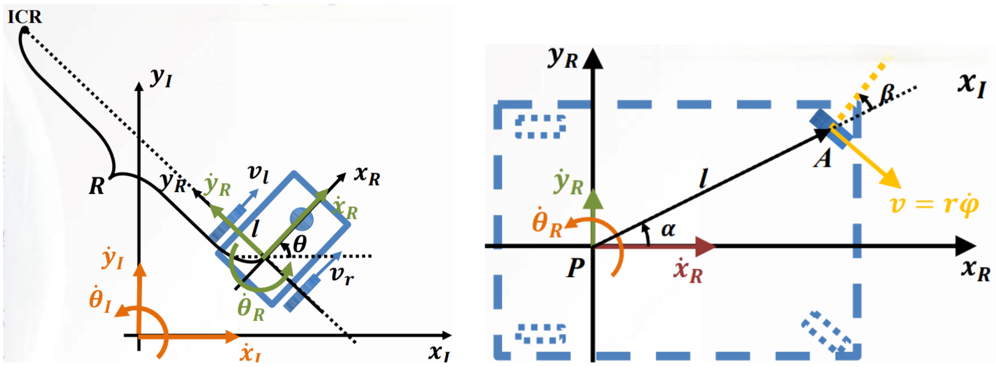
$$C_1(\beta_s)R(\theta)\dot{\xi}_I = 0$$

返回目录

其中$C_1(\beta_s)=\begin{bmatrix} C_{1f(N_f\times 3)} \\ C_{1s}(\beta_s)_{(N_s\times 3)} \end{bmatrix}$。

3.3.4 例子

以下以两轮差速机器人（见1）为例，$\alpha=\frac{\pi}{2},\beta=0$：



(a) ICR法

(b) 约束方程法

图 8: 两轮差速机器人正运动学建模

ICR法

两轮差速机器人的瞬心在两轮轮轴上，设其到机器人两轮中间的距离为R，有：

$$\dot{\theta}=\frac{\dot{x}_R}{R}=\frac{\dot{\varphi}_l r}{R-l}=\frac{\dot{\varphi}_r r}{R+l}$$

解得$R=\frac{\dot{\varphi}_r+\dot{\varphi}_l}{\dot{\varphi}_r-\dot{\varphi}_l}$，代回即可。

约束方程法

- 纯滚动：$\begin{bmatrix}\sin(\alpha+\beta) & -\cos(\alpha+\beta) & -l\cos\beta\end{bmatrix}\dot{\xi}_R=r\dot{\varphi}$。
- 无滑动：$\begin{bmatrix}\cos(\alpha+\beta) & \sin(\alpha+\beta) & l\sin\beta\end{bmatrix}\dot{\xi}_R=0$。

正运动学模型

$$\dot{\xi}_R=\begin{bmatrix}\dot{x}_R \\ \dot{y}_R \\ \dot{\theta}_R\end{bmatrix}=\frac{r}{2}\begin{bmatrix}\dot{\varphi}_l+\dot{\varphi}_r \\ 0 \\ \frac{\dot{\varphi}_r-\dot{\varphi}_l}{l}\end{bmatrix}$$

返回目录

3.4 自由度

概念

- 衡量机器人改变运行状态的能力。
- 需满足实际作业需求，考虑实现成本。
- 机器人设计基础、算法依据（一般自由度相同的机器人可采用相同的控制和规划算法）。

- 平面运动机器人自由度最大不能超过3。

分类 [8]

- 移动度（Degree of Mobility）δ_m：瞬时改变机器人运动状态的能力。

$$\delta_m = \dim[C_1(\beta_s)] = 3 - \mathrm{rank}[C_1(\beta_s)] \in [0, 3]$$

- 转向度（Degree of Steerability）δ_s：间接改变机器人运动状态的能力。

$$\delta_s = \mathrm{rank}[C_{1s}(\beta_s)] \in [0, 2]$$

- 机动度（Degree of Maneuverability）δ_M：改变机器人运动状态的能力。

$$\delta_M = \delta_m + \delta_s$$

  - 机动度相同，结构不一定相同。
  - $\delta_M = 2$，瞬心位于一条直线上；$\delta_M = 3$，瞬心可分布于空间任何一点。

实例

- 全向机器人：
  - Type(3,0)：完整约束全方位移动机器人。
  - Type(2,1)：一个同心轮+两个瑞典轮。
  - Type(1,2)：多舵机全方位移动机器人。
- 非全向机器人：
  - Type(2,0)：差分移动机器人。
  - Type(1,1)：自动驾驶汽车（阿克曼转向）、自行车、叉车。

返回目录

4 机器人运动控制

4.1 运动控制

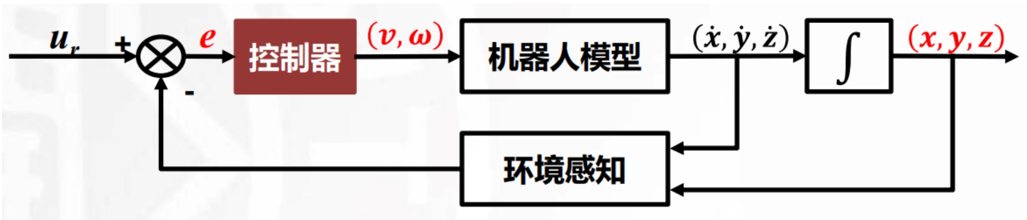


图 9: 运动控制器

误差（惯性系下给定与反馈）$\xRightarrow{\text{变换}}$输入（机器人系下控制输入）。

特点

- 大多存在滑动约束，是非完整系统，有侧向偏差和姿态偏差。
- 非线性，控制器设计复杂，还需要根据可获得的反馈信号选取，按顺序调节控制参数，并且不能同时实现定点控制和跟踪控制。
- 不存在能完成控制目标的连续时不变（静态）反馈控制率。
- 受标定精度影响大，且由于执行单元性能约束，控制输入要合理限幅。

分类

- 定点（镇定）控制（Regulation Control）：以指定姿态到达指定位置。
- 跟踪控制：
  - 轨迹跟踪控制（Trajectory Tracking Control）：跟随给定轨迹（包含速度、姿态信息）。
  - 路径跟踪控制（Path Tracking Control）：跟随给定路线。

开环控制 将运动轨迹分割成直线和圆弧，存在以下问题：

- 直线和圆弧的曲率不一致，不连续。

返回目录

- 难以实现定义若干合适轨迹。
- 速度加速度约束。
- 无法自适应调整轨迹来面对环境变化。
- 所得轨迹不光滑。

控制器性能评价 取正定李雅普诺夫函数，其导数负定则系统渐进稳定。

两轮差速机器人运动控制 以下以两轮差速机器人（见1）为例。

$$\dot{\xi}_R = \begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta}_R \end{bmatrix} = \frac{r}{2}\begin{bmatrix} \dot{\varphi}_l + \dot{\varphi}_r \\ 0 \\ \frac{\dot{\varphi}_r - \dot{\varphi}_l}{l} \end{bmatrix} = \begin{bmatrix} v_1 \\ 0 \\ v_2 \end{bmatrix} \xRightarrow{\text{变换}} \dot{\xi}_I = \begin{bmatrix} \dot{x}_I \\ \dot{y}_I \\ \dot{\theta}_I \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

4.2 定点控制器

9

控制目标

机器人参考坐标系下误差$e = \begin{bmatrix} x & y & \theta \end{bmatrix}^T$，

设计控制阵$K = \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \end{bmatrix}$，其中$k_{ij} = k(t, e)$，

得到控制输入$\begin{bmatrix} v(t) \\ \omega(t) \end{bmatrix} = Ke$，使$\lim_{t\to\infty} e(t) = 0$。

误差信号转换

惯性系下，实际状态$q = \begin{bmatrix} x & y & \theta \end{bmatrix}^T$与参考状态$q_r \begin{bmatrix} x_r & y_r & \theta_r \end{bmatrix}^T$之差为开环误差：

$$\tilde{q} = \begin{bmatrix} \tilde{x} & \tilde{y} & \tilde{\theta} \end{bmatrix}^T = \begin{bmatrix} x - x_r & y - y_r & \theta - \theta_r \end{bmatrix}^T$$

返回目录

1. 转换到机器人系

$$e = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = R(\theta)^T \tilde{q} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{\theta} \end{bmatrix} \xRightarrow{\text{闭环}} \dot{e} = \begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \\ \dot{e}_3 \end{bmatrix} = R(\theta)\dot{\tilde{q}} + R\dot{(\theta)}\tilde{q} = \begin{bmatrix} v_1 + v_2 e_2 \\ -v_2 e_1 \\ v_2 \end{bmatrix}$$

2. 转换到极坐标系

$$\begin{cases} \rho & = \sqrt{\tilde{x}^2 + \tilde{y}^2} \\ \beta & = -\arctan 2(-\tilde{y}, -\tilde{x}) \\ \alpha & = -\beta - \tilde{\theta} \end{cases} \xRightarrow{\text{闭环}} \begin{bmatrix} \dot{\rho} \\ \dot{\beta} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} -\cos\alpha & 0 \\ \frac{\sin\alpha}{\rho} & 0 \\ \frac{\sin\alpha}{\rho} & -1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

机器人系非线性控制器

设计控制器$\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} -k_1 e_1 \\ -k_2 e_2 + e_2^2 \sin(t) \end{bmatrix}$，代入得$\begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \\ \dot{e}_3 \end{bmatrix} = \begin{bmatrix} -k_1 e_1 + v_2 e_2 \\ -v_2 e_1 \\ -k_2 e_2 + e_2^2 \sin(t) \end{bmatrix}$。

其有误差时扰动，效果不佳。

极坐标系线性控制器

设计控制器$\begin{cases} v_1 & = k_\rho \rho \\ v_2 & = k_\alpha \alpha + k_\beta \beta \end{cases}$，代入得

$$\begin{bmatrix} \dot{\rho} \\ \dot{\beta} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} -k_\rho \rho \cos\alpha \\ -k_\rho \sin\alpha \\ k_\rho \sin\alpha - k_\alpha \alpha - k_\beta \beta \end{bmatrix} \xRightarrow{\alpha \to 0} \begin{bmatrix} -k_\rho \rho \\ -k_\rho \alpha \\ k_\rho \alpha - k_\alpha \alpha - k_\beta \beta \end{bmatrix}$$

其中前两行非线性耦合，在$\alpha \to 0$时指数性稳定，非全局稳定。

极坐标系线性控制器

设计控制器$\begin{cases} v_1 &= k_\rho \rho \cos\alpha \\ v_2 &= k_\alpha \alpha + \frac{k_\rho \sin\alpha\cos\alpha}{\alpha}(\alpha - k_\beta \beta) \end{cases}$，代入得

$$\begin{bmatrix} \dot\rho \\ \dot\beta \\ \dot\alpha \end{bmatrix} = \begin{bmatrix} -k_\rho \rho \cos\alpha\cos\alpha \\ -k_\rho \cos\alpha\sin\alpha \\ k_\rho\cos\alpha\sin\alpha - k_\alpha\alpha - \underbrace{\frac{k_\rho\sin\alpha\cos\alpha}{\alpha}}_{\alpha\to 0, \frac{\sin\alpha}{\alpha}=1}(\alpha - k_\beta\beta) \end{bmatrix}$$

其全局渐近稳定。

4.3 轨迹跟踪控制器

10

控制目标与误差变换

惯性系下，实际轨迹$q(t) = \begin{bmatrix} x(t) & y(t) & \theta(t) \end{bmatrix}^T$与参考轨迹$q_r \begin{bmatrix} x_r(t) & y_r(t) & \theta_r(t) \end{bmatrix}^T$之差为开环误差：

$$\tilde q(t) = \begin{bmatrix} \tilde x(t) & \tilde y(t) & \tilde\theta(t) \end{bmatrix}^T = \begin{bmatrix} x(t) - x_r(t) & y(t) - y_r(t) & \theta(t) - \theta_r(t) \end{bmatrix}^T$$

控制目标为$\lim_{n\to\infty} \tilde q(t) = 0$。

开环误差转换坐标系后求闭环误差，进而得到辅助误差信号：

$$e = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde x \\ \tilde y \\ \tilde\theta \end{bmatrix} \xRightarrow{\text{闭环}} \dot e = \begin{bmatrix} v_1 + v_2 e_2 - v_{1r}\cos e_3 \\ -v_2 e_1 + v_{1r}\sin e_3 \\ v_2 - v_{2r} \end{bmatrix}$$

控制器

设计控制器$\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} -k_1 e_1 + v_{1r}\cos e_3 \\ -v_{1r}\frac{\sin e_3}{e_3} e_2 - k_2 e_3 + v_{r2} \end{bmatrix}$，代入得

$$\begin{bmatrix} \dot e_1 \\ \dot e_2 \\ \dot e_3 \end{bmatrix} = \begin{bmatrix} -k_1 e_1 + v_2 e_2 \\ -v_2 e_1 + v_{1r}\sin e_3 \\ -k_2 e_3 - v_{1r}\frac{\sin e_3}{e_3} e_2 \end{bmatrix}$$

4.4 路径跟踪控制器

11

控制目标与误差变换

惯性系下，实际路径$q(s) = \begin{bmatrix} x(s) & y(s) & \theta(s) \end{bmatrix}^T$与参考路径$q_r \begin{bmatrix} x_r(s) & y_r(s) & \theta_r(s) \end{bmatrix}^T$之差为开环误差：

$$\tilde{q}(s) = \begin{bmatrix} \tilde{x}(s) & \tilde{y}(s) & \tilde{\theta}(s) \end{bmatrix}^T = \begin{bmatrix} x(s)-x_r(s) & y(s)-y_r(s) & \theta(s)-\theta_r(s) \end{bmatrix}^T$$

其中$s \in [0,1]$为路径参考变量，控制目标为$\lim_{n\to\infty} \tilde{q}(s) = 0$。

作变换$\begin{cases} y_1 & = x + b\cos\theta \\ y_2 & = y + b\sin\theta \end{cases}$，

进而得到闭环误差$\begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \end{bmatrix} = \begin{bmatrix} \cos\theta & -b\sin\theta \\ \sin\theta & b\cos\theta \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} = T(\theta) \begin{bmatrix} v \\ \omega \end{bmatrix}$。

逆运算得到$\begin{bmatrix} v \\ \omega \end{bmatrix} = T^{-1}(\theta) \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\frac{\sin\theta}{b} & \frac{\cos\theta}{b} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$，

故$\begin{cases} \dot{y}_1 & = u_1 \\ \dot{y}_2 & = u_2 \\ \dot{\theta} & = \frac{u_2\cos\theta - u_1\sin\theta}{b} \end{cases}$。

控制器

设计控制器$\begin{cases} u_1 = \dot{y}_{1d} + k_1(y_{1d} - y_1) \\ u_2 = \dot{y}_{2d} + k_2(y_{2d} - y_2) \end{cases}$，有$\begin{cases} \dot{\tilde{y}}_1 = -k_1\tilde{y}_1 \\ \dot{\tilde{y}}_2 = -k_2\tilde{y}_2 \end{cases}$，系统指数性收敛。

返回目录

5 机器人感知

5.1 传感器

5.2 激光定位

5.3 SVD与ICP

5.4 卡尔曼滤波

5.5 蒙特卡洛定位

5.6 SLAM

6 机器人轨迹规划

返回目录