

CHAPTER 04

机器的"四肢"控制与行动

从空调温控到自动驾驶

反馈控制如何驱动智能体

通识选修课《具身智能：从看见到行动》

如果机器人没有"手"和"脚"

假设大脑能精准识别行人并计算出"应该刹车"，但执行机构反应迟钝——会发生什么？

再聪明的大脑也没有用。**行动，是智能的"出口"。**

人类的反射弧

感受器 → 感受到刺激

传入神经 → 传到中枢

中枢 → 做出决策

传出神经 → 指挥效应器

肌肉 → 收缩产生动作

生物体的行动链条

机器人的"反射弧"

传感器 → 采集环境数据

控制器 → 运行算法决策

执行器 → 电机/液压驱动

机械结构 → 产生物理动作

具身智能的行动链条

本章核心命题：行动不只是智能的附属品，而是智能的完成态。

夏天的空调——最熟悉的自动控制系统

反馈控制的四个环节

设定值 Setpoint
目标温度 25° C

→

传感器 Sensor
实时测量房间温度

→

控制器 Controller
计算误差，决定动作

→

执行器 Actuator
压缩机+风扇

闭环
Loop

闭环工作原理

1. **设定目标**：用户设定 25° C → 系统的"目标值"
2. **测量实际**：传感器检测当前温度（如 28° C）
3. **计算误差**：Error = 设定值 − 实际值 = 25 − 28 = −3° C
4. **决策动作**：误差为负 → 加大制冷功率
5. **执行动作**：压缩机加速 → 温度下降
6. **再次测量**：温度降到 25.5° C → 误差 = −0.5° C → 减小制冷
7. **循环往复**：最终稳定在 25° C ± 0.5° C

PID控制——工业文明的"隐藏基石"

1922年发明 · 100+年

比例-积分-微分控制（Proportional-Integral-Derivative Control）

$$输出 = Kp \times 误差 + Ki \times \int 误差 dt + Kd \times d(误差)/dt$$

无处不在的应用

空调冰箱	汽车巡航	飞机飞行	SpaceX火箭	机器人
温度控制	速度控制	姿态控制	着陆控制	关节控制

核心思想

根据"误差"（设定值与实际值之差）的**三个维度**来计算控制输出：

P（比例）看当前误差 → I（积分）看历史累积 → D（微分）看变化趋势

PID三维度详解——P、I、D各司其职

P 比例 Proportional

"差多少，调多少"

- 误差越大，调节力度越大
- 公式：输出 = Kp × 误差
- 问题：无法消除稳态误差

I 积分 Integral

"历史欠账，慢慢补上"

- 累积历史误差消除静差
- 公式：输出 = Ki × ∫误差
- 问题：可能带来超调

D 微分 Derivative

"提前刹车，防止冲过头"

- 观察误差变化趋势
- 公式：输出 = Kd × d(误差)/dt
- 作用：抑制超调

生活类比：开车保持与前车50米距离

P = 追上去

当前距离80米 → 踩油门追

D = 别追过头

距离快速缩小 → 提前松油门

I = 精确对齐

稳定在52米 → 微调消除残余误差

P负责大方向，D负责防超调，I负责消除残余误差 —— 三者缺一不可

PID参数整定方法论

调参口诀："先调P让系统动起来，再加I消除静差，最后D抑制振荡"

1

调 P（比例增益）

先让 Ki=0，Kd=0，只保留比例项。逐渐增大 Kp，找到快速响应但不震荡的值。

Kp太小 → 反应迟钝 | Kp太大 → 剧烈震荡

2

加 I（积分增益）

在调好 Kp 的基础上，逐渐增加 Ki，找到能消除稳态误差但又不引起明显震荡的值。

Ki太小 → 静差消除慢 | Ki太大 → 系统震荡加剧

3

加 D（微分增益）

在调好 Kp 和 Ki 的基础上，逐渐增加 Kd，找到能抑制超调但对噪声不过于敏感的值。

Kd太小 → 超调抑制不明显 | Kd太大 → 对噪声敏感

调参本质：在**"响应速度"**和**"稳定性"**之间找平衡 —— 好的PID调参就是在这对矛盾中找到最优解

从空调到机器人——反馈控制的通用性

机械臂定位

- 传感器（编码器）测量当前位置
- 控制器计算位置误差
- 执行器（电机）调整关节角度
- 每个关节独立PID，多关节协同

无人机悬停

- 传感器（气压计、IMU）测高度和姿态
- 控制器计算高度/姿态误差
- 执行器调整四个螺旋桨转速
- 四电机PID精确协调

自动驾驶车道保持

- 传感器（摄像头）检测车道线
- 控制器计算横向误差+航向误差
- 执行器（转向电机）调方向盘
- 响应速度至关重要

波士顿动力 Atlas

- **28个主动自由度**
- 数十个PID控制器协同网络
- 高度控制 + 姿态控制 + 位置控制
- 站立、行走、跳跃、后空翻

课堂互动①——PID调参体验

小组讨论 3分钟

1 为什么空调设定25°C后，温度不会"精准停"在25.000°C？

提示：PID控制器在持续调节，总是在"过一点"和"欠一点"之间寻找平衡，系统的惯性让完美的"零误差"很难达到。

2 纯P控制做"车道保持"会出现什么现象？为什么需要D和I？

提示：P控制看到偏离就大幅修正，修正过头又向另一边偏离 → 来回摆动。D"提前刹车"，I消除长期残余误差。

3 骑自行车保持直立——你的大脑在运行PID控制吗？P、I、D分别对应什么？

提示：P ≈ 身体倾斜时本能向反方向施力；D ≈ 倾斜速度加快时提前调整；I ≈ 长期微调找到最适合自己的平衡点。

路径规划——从A点到B点，走哪条路？

给定**起点、终点、障碍物** → 找可行/最优路径

A*算法

$f(n) = g(n) + h(n)$

g = 已走路径代价

h = 到终点的估计代价

优势：保证最优路径

局限：需预先构建地图

RRT算法

随机采样 + 快速探索

在环境中随机撒点

逐步扩展覆盖整棵树

优势：适合高维复杂环境

局限：路径非最优

人工势场法

目标吸引 + 障碍排斥

像磁铁一样被目标吸引

像同极磁铁被障碍排斥

优势：计算简单、实时性好

局限：易陷局部最优

三种算法对比

维度	A*算法	RRT算法	人工势场法
核心思想	实际代价+估计代价	随机采样扩展	引力+斥力
适用场景	地图已知、求最优	高维复杂环境	计算资源有限

轨迹规划——不仅"去哪里"，还有"怎么去"

路径 = 几何曲线（在哪里转弯）　　轨迹 = 路径 + 时间（什么时候转、转多快）

生活类比

宿舍到教室："慢悠悠散步" vs "狂奔" —— 路径相同，轨迹不同

轨迹规划的约束条件

速度限制	加速度限制	曲率限制	Jerk限制	能量消耗
不能超限速	不能急加急刹	转弯不能太急	加速度变化率	尽量节省能源

目标

在所有约束条件的"交集"中，找到一条最优轨迹

不是"最快"的（可能不安全），也不是"最慢"的（可能效率低）

而是**"在安全、舒适、效率之间找到最佳平衡"**的

自动驾驶运动规划的三层架构

高层 战略性

行为规划 Behavioral Planning

决定"做什么"：变道？超车？停车让行？

基于对整体交通形势的理解

中层 战术性

路径规划 Path Planning

决定"走哪里"：具体选择哪条路线完成变道？

沿车道中心线还是偏一侧避开卡车？

底层执行

速度规划 Speed Planning

决定"走多快"：什么时候加速？减速速率多少？

每个时间点的具体速度控制

行为决定"做什么" → 路径决定"走哪里" → 速度决定"走多快"

物理世界的"铁律"——机器不能违背物理

惯性定律 F = ma

满载货车40吨 vs 小轿车1.5吨

100km/h刹停：150米 vs 40米

摩擦定律

干燥路面 μ≈0.7-0.8 | 结冰路面 μ≈0.1-0.2

冰面刹车距离是干燥路面的 7-8倍

电机极限

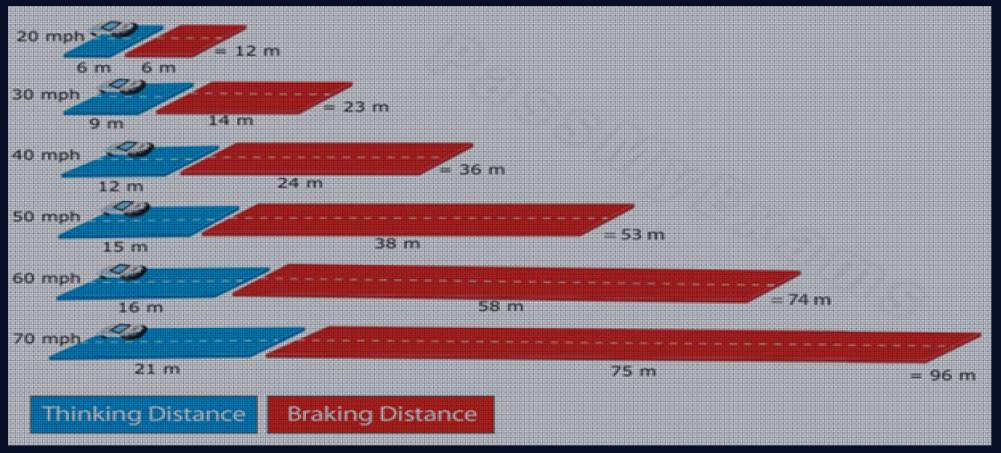
最大扭矩和最大转速是"硬天花板"

无论多急迫，电机只能输出设计范围内的力

能量限制

Atlas后空翻不能无限做——电池有限

激进驾驶风格显著增加能耗



核心思想

不是打破镣铐

而是**"戴着镣铐跳舞"**

在约束中找到最优解

"鬼探头"场景中的物理约束

场景：YOLO5检测到行人 | 车速60km/h | 行人距车头20米 | 干燥路面

物理计算

刹车距离 = $v^2/(2\mu g) = 16.7^2/(2\times0.7\times9.8) \approx$ **20.3米**

反应距离 = 100-300ms延迟 × 速度 ≈ **1.7-5米**

总停车距离 = 22-25米 > 20米

结论：无法在距离内完全刹停

决策树

① 能否在距离内刹停？ → 否

② 能否变道避让？ → 检查相邻车道

③ 最小伤害策略 → 全力刹车+调整方向

"带着镣铐跳舞"

不是"最优解"，而是"约束下的最好解"

Sim-to-Real中的物理鸿沟

仿真→真实：性能骤降（Performance Degradation）

四大简化误差源

摩擦模型近似

仿真用简单库仑摩擦模型，真实世界受温度、湿度、磨损影响

碰撞离散化

仿真是刚体瞬间反弹，真实世界是连续变形、吸收能量

电机延迟忽略

仿真执行器"瞬时响应"，真实电机有10-100ms延迟

柔性形变未建模

轮胎、悬挂等受力会形变，仿真简化为刚体

解决思路：追求对扰动鲁棒的策略

域随机化

训练时故意随机扰动物理参数，让策略"见多识广"

域适应

学习"仿真到真实"的映射函数，用少量真实数据微调

残差学习

训练残差模型预测仿真与真实的差异，实时补偿

渐进式迁移

仿真→半仿真（硬件在环）→真实世界，逐步过渡

"不要追求'完美仿真'，而应追求'对扰动鲁棒的策略'"

力控与柔顺——从"硬碰硬"到"软接触"



位置控制的问题

不考虑接触力 → 捏碎鸡蛋

力控 Force Control

目标不是到达位置，而是"施加某个力"

"用5牛顿的力抓住鸡蛋" → 不会捏碎

柔顺控制 Compliant Control

控制力与位移的关系，像弹簧一样有弹性

生活类比

推门 → 位置控制

要让门到达"打开"的位置，所以用力推

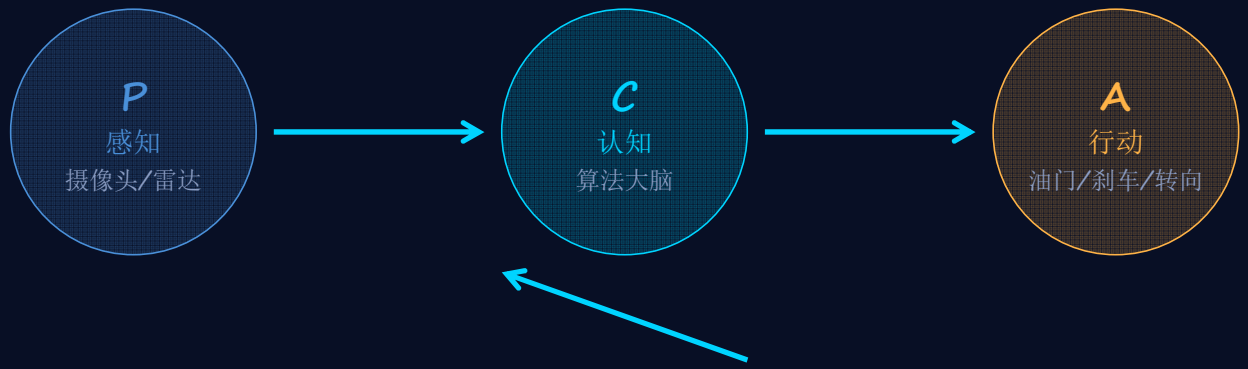
推婴儿车 → 力控

要让车以合适速度移动，所以施加合适的力

协作机器人(Cobot)安全性：碰人时"顺从退让" → 柔顺控制保障人机共存安全

PCA闭环的完整拼图——感知→认知→行动的"共舞"

不是"流水线"，而是循环往复、相互塑造的动态系统



学骑自行车的PCA闭环

车身向左倾斜 → 感知（身体感觉失衡） → 大脑意识到要摔倒 → 认知
→ 下意识将车把向右转 → 行动 → 车身回正 → 新的感知 → ... 螺旋上升

核心： 行动改变环境 → 环境变化产生新感知 → 新感知引发新认知 → 新认知指导新行动

延迟——闭环的"隐形杀手"

总延迟 = 感知延迟 + 计算延迟 + 执行延迟 = 100-500毫秒

感知延迟

摄像头 30fps ≈ 33ms/帧

激光雷达转一圈 ≈ 100ms

毫米波雷达 ≈ 50ms

计算延迟

目标检测 + 轨迹预测 + 路径规划 + 控制指令

端到端 ≈ 50-300ms

执行延迟

电机电磁响应 ≈ 10ms

液压系统 ≈ 50-100ms

机械传动 ≈ 20-50ms

120km/h 下的延迟代价

每 **100ms** 延迟 = **3.3米** 盲开

300ms 延迟 = **10米** —— 生死之差

降延迟措施

- 边缘计算：车载平台，5-15ms
- 任务优先级调度：感知控制优先
- 模型轻量化：蒸馏/剪枝/量化
- 硬件加速：专用AI芯片

从PCA到PCAP + MPC——"预见未来"的控制

PCAP闭环 = 加入"预测"

Prediction（预测）+ Perception（感知）
+ Cognition（认知）+ Action（行动）

从"反应式"到"前瞻式"

MPC模型预测控制

① 预测未来3秒行为
② 优化控制序列
③ 只执行第一步 → 重复

分层控制架构：MPC规划 + PID跟踪

上层 MPC：轨迹规划 —— 决定走哪条路径、以什么速度行驶（"预见未来"）

↓

下层 PID：轨迹跟踪 —— 控制方向盘、油门、刹车，精确沿规划轨迹行驶

端到端控制（FSD V12）

传感器直接映射控制输出 → 信息不损失，但存在"黑箱风险"（难以解释决策原因）

课堂互动②——综合讨论

分组辩论

1 设计"机器人夹取生鸡蛋"系统：位置控制、力控还是柔顺控制？

提示：纯位置控制会把鸡蛋捏碎（只关心到达目标位置，不关心用了多大力）。力控或柔顺控制可以精确控制力的大小。

2 感知延迟从50ms增加到200ms，哪些环节需要重新设计？

提示：需要更强的预测能力弥补延迟；可能需要更快的边缘计算硬件；也可能需要更保守的安全策略。

3 仓库搬运机器人：A*、RRT、人工势场法，你选哪个？为什么？

提示：地图已知且静态→A*最优；环境复杂多变→RRT灵活；计算资源有限→势场法简单但要小心局部最优陷阱。

本章小结——行动是智能的"出口"

PID控制

P看当前、I看历史、D看趋势

工业文明的隐藏基石

运动规划

A*寻最优、RRT探复杂

势场法模拟引斥力

物理极限

惯性/摩擦/电机/能量

戴着镣铐跳舞

PCA闭环

感知→认知→行动，循环往复、相互塑造

不是流水线，而是共舞

PCAP进化

从反应式到前瞻式

MPC规划 + PID跟踪

NEXT CHAPTER

平行宇宙：虚拟世界中的"试错"百万次

如果机器人可以在虚拟世界里试错，再进入现实世界？

"想"和"做"之间，隔着整个物理世界。