# VEOS by Data Driven Planning

## 测试条件

-固定测试场景 -固定工况 -不开空调（减少空调能耗干扰）-往返路线（减少地形差异干扰）-观测噪声：地形，压缩机，电池 SOC,(大灯,tbox,...) -测量驾驶风格：纵向控制问题中，特定工况下油门踏板（和刹车踏板）的使用情况 -通过独立的 UDP 数据记录交叉验证测量和性能 -总共实验 1400 次

### 驾驶风格

- 无 AI 和带 AI 的基准驾驶风格比较

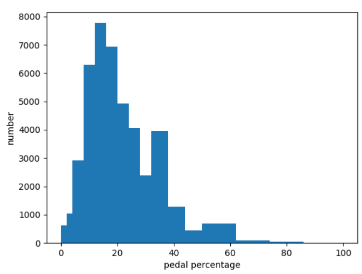


图 1.1 无 AI 的基准风格分布

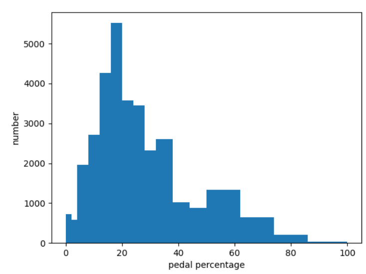


图 1.2 带 AI 的基准风格总平均分布

- 驾驶风格按周期变化：驾驶风格相对同一个司机是固定的

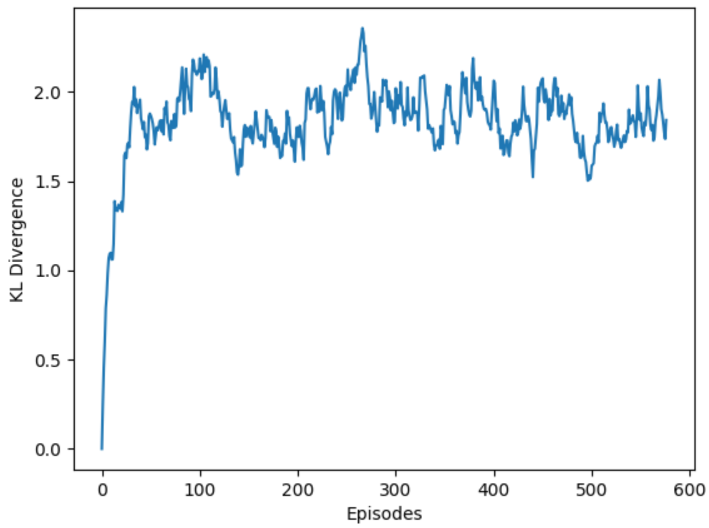


图 2 驾驶风格变化按 KL 散度评估，风格相对固定

- 驾驶风格有 AI 和无 AI 比较

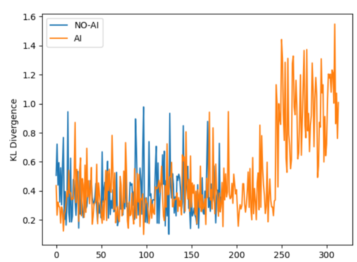


图 3.1 驾驶风格有 AI 和无 AI 比较，后面打开 coastdown

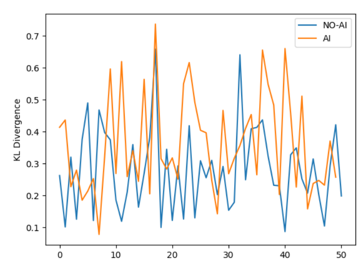


图 3.2 另一位驾驶员有 AI 与无 AI 比较

- 不同驾驶风格与 SAC 下驾驶风格总体比较:

| | SAC | DDPG-CD | SAC-CD | Gonghao-no CD |
|---|---|---|---|---|
| KL-D | 0 | 0.234 | 0.311 | 0.334 |

**能耗**

- 电动力默认 Pedal Map (PM) vs 自建 Pedal Map
    - 默认 PM: 高速时请求力矩会降低
    - 自建 PM: 分段线性, 请求力矩分段线性单调

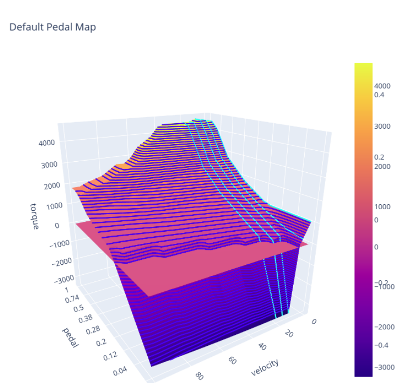


图 4.1 EP 默认 PM

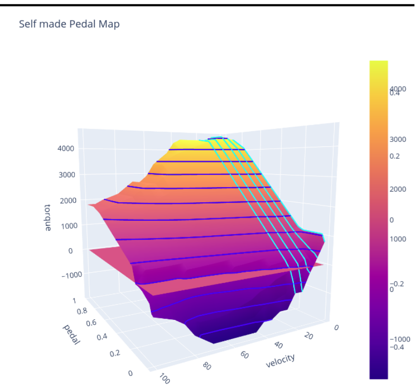


图 4.2 自建 PM

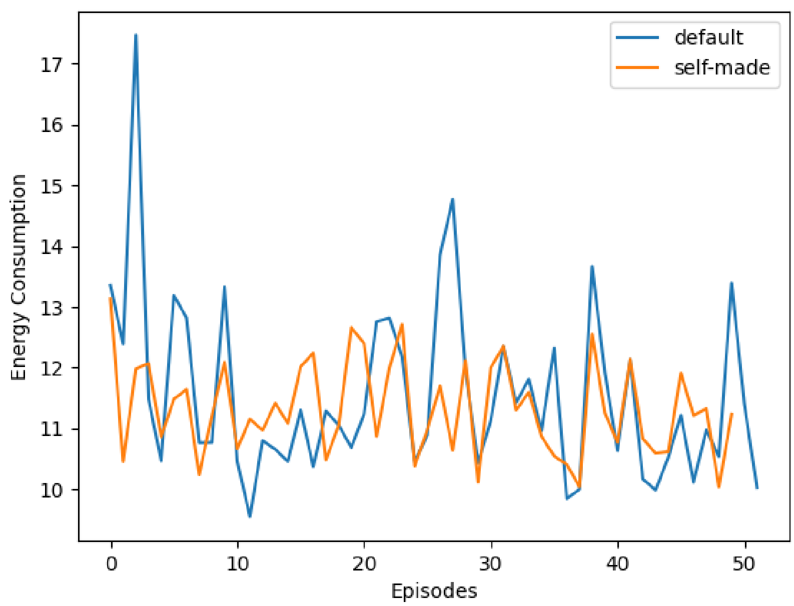


图 5 EP 默认 PM 与自建 PM 能耗比较,

- 具备较强能量回收的 pedal map

**能耗结果**

历次带 AI tensorboard - 襄阳 vs. 上海 (解决时间同步问题和漏帧问题) - 确认收敛过程 - 能耗持续降低过程

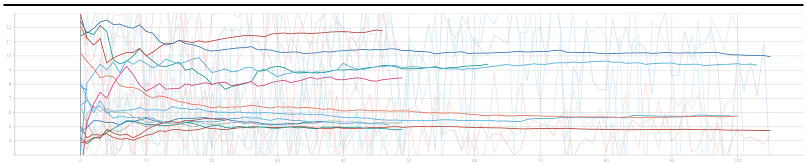

图 6 SAC 算法襄阳和上海对比

- 上海优化改进过程
  - 能耗持续降低

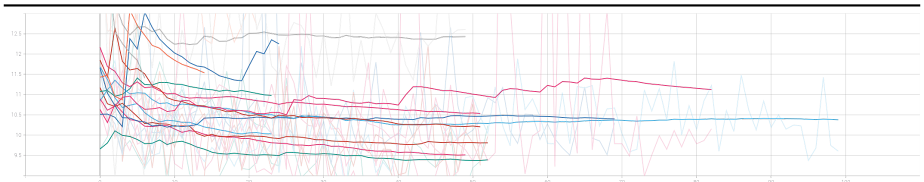

图 6 上海算法改进过程

- SAC 持续模式

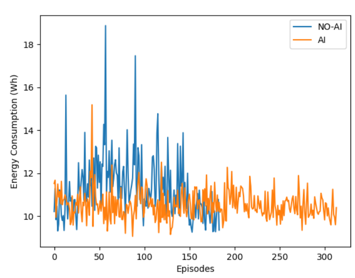


图 7.1 驾驶风格有 AI 和无 AI 比较，后面打开 coastdown

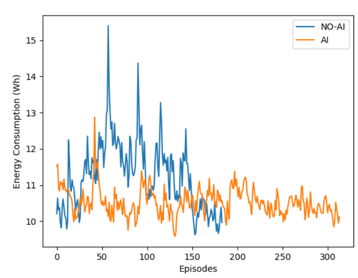


图 7.2 另一位驾驶员有 AI 与无 AI 比较

5.(无 AI vs. 有 AI) 各 50 次 6. 有 AI 持续模式 7. 不同驾驶员带 AI 优化过程 8. 怠速表关闭/打开 9. 熵变小/策略趋向确定性 10.DDPG vs. SAC

- baselines

- driver styles analysis (analysis)
- pedal map comparison
- A2C
- different initial map

- Declining in each epoch
  - RL agent cooperative / Human driver adapting?
  - Epoch tendcy
  - declining in total loss / inclining in total reward
  - entropy declining / becoming more and more deterministic
  - expected wh declining
  - seems cooperative, at least no conflict
  - baseline: strong regen -> higher efficiency
  - methods
  - achievements
  - status
- long-term in resume mode (model and table resumed)
- weak regen (fix coastdown / constrained action space)
- strong regen (exploit coastdown / relax action space)

## Analysis

weaker regen: not stronger regen, but better motion control for the test case

- possible models

debug

- tools
  - **driving style analysis (quantitative)**
    * vehicle interfaces and systems (stable and reliable)
    * synchronization
  - data logging (for analysis and offline algo)
  - udp episodic analysis (cross check)
  - energy consumpt cross-check by UDP messages
  - model resume tool
  - different driver storage with resume and from scratch
  - debug (latency analysis)
  - verifying DL algo with cpu only resources
  - analysis
  - optimal motion planning
  - limiting action space,
  - exploit regen, activate coastdown part
  - better assistance for manual motion control for eco
  - reward shaping (penalize braking could be cooperative)

- need recurrency to encode system dynamics
- observing, acting rate, BP rate

#### 方法

强化学习方法，以大数据为基础的奖励驱动优化方法 - **没有模型** - 车辆动力学的模型和知识 - 电机模型 - 电源管理系统模型 - 符合学习直觉： - 利用大数据建立内部模型 - 自适应动态过程 -

#### 理论分析

- not like this: big data -> NN -> label ==> good result
- learn from data (distribution) not label (label is supervision)
  - distribution, law of large number n>30, (multiplicity with samples)
  - dynamic environment -> drifting distribution
  - **advantages**: previously impossible cases can be solved elegantly by big data.
- basic observability/controllability
  - observation enough? fully observable -> which should I observe?
  - control signal sufficient/efficient?
  - long-term dependency
- Model
  - Complete observable model (MDP)
  - human driver model $Th = Th(\mathbf{O_h})$
  - pedal map $\tilde{PM}(Th) = Trq$
  - $Trq = \tilde{PM} \circ Th = \tilde{PM}(Th(O_h))$
  - $\mathbf{O_h} = (vel, road, objects)$
  - Objective: Optimal Motion Planning
    * $\min_{Trq}(\Sigma_i(u \cdot i) \cdot dt)$
      - follow the optimal motion planning (follow the optimal speed curve)
      - reduce unnecessary large torque
      - maintain a speed when regenerative brake occurs (exploit regenerative brake)
  - Implementation, POMDP
    * $\mathbf{O_{rl}} = (vel, Th)$
    * $\mathbf{O_{rlx}} = (vel, Th, MotionPlan)$

## Outlook

### fully autonomous

1. optimal motion control/prediction

2. exploit regen

### assistance system

1. fix driving style and analysis
2. different reward

### Challenge

动态过程未知，奖励不完全知道，并非简单将数据灌入神经网络, 需要考虑几个因素。

- sample efficiency
- offline data utilization
- reward shaping
- memory

### Counter measures