`IETS_RT_Dispatch`-论文调研

创新点

- ADP
    - 近似动态规划到底是什么？
    - 从动态规划到近似动态规划：BERTSEKAS 2014 在THU的暑期课程笔记
    - 近似动态规划和强化学习 - ADP&RL
- IL(Imitation Learning)
    - 模仿学习简洁教程

目标

`IETS`实时调度问题的目标是设计一个调度策略以最小化操作成本：

$$F = \min_{\pi \in \Pi} \mathbf{E}\left(\sum_{t \in \mathscr{T}} C(t)\right), \tag{9}$$

解决方案

> *ADP+IL*

3.1 DP

1 外部信息

`IETS`的外部信息包括：`RES`的生成的随机过程、环境温度、电热负载、实时价格。

$$W_t = \{\varepsilon(t), P_i^{\mathrm{EL}}(t), Q_i^{\mathrm{EL}}(t), H_n^{\mathrm{TL}}(t), \tau^{\mathrm{AM}}(t), \overline{P}_n^{\mathrm{RES}}(t), \overline{Q}_n^{\mathrm{RES}}(t)\}. \tag{11}$$

备注：

Active/reactive electrical load

Ambient temperature

$$W_t = \{\varepsilon(t), P_i^{\mathrm{EL}}(t), Q_i^{\mathrm{EL}}(t), H_n^{\mathrm{TL}}(t), \tau^{\mathrm{AM}}(t), \overline{P}_n^{\mathrm{RES}}(t), \overline{Q}_n^{\mathrm{RES}}(t)\}.$$

Electricity purchase price

Thermal load in node

Upper active/reactive power output limit of RES

2 系统状态

$$S_t=\{W_t, R_t\}$$

- 外部信息
- 资源状态

$$R_t = \{E_n^{\mathrm{TS}}(t), E_n^{\mathrm{BS}}(t)\}. \tag{12}$$

备注：

$$R_t = \{E_n^{\mathrm{TS}}(t), E_n^{\mathrm{BS}}(t)\}.$$

Energy stored in TS/BS at time-slot t

3 决策变量

$$X_t = \{H_n^{\mathrm{CHP}}(t), P_n^{\mathrm{CHP}}(t), a_{\mathrm{d},n}^{\mathrm{BS}}(t), a_{\mathrm{c},n}^{\mathrm{BS}}(t), a_{\mathrm{d},n}^{\mathrm{TS}}(t), a_{\mathrm{c},n}^{\mathrm{TS}}(t), H_{\mathrm{c},n}^{\mathrm{TS}}(t), H_{\mathrm{d},n}^{\mathrm{TS}}(t), P_{\mathrm{c},n}^{\mathrm{BS}}(t), P_{\mathrm{d},n}^{\mathrm{BS}}(t), P_n^{\mathrm{RES}}(t), Q_n^{\mathrm{RES}}(t), X_t^{\mathrm{A}}\}.$$

备注：

$a_{\mathrm{c},n}^{\mathrm{TS}}(t)/a_{\mathrm{c},n}^{\mathrm{BS}}(t)$ Binary variables indicating if TS/BS is charged

$a_{\mathrm{d},n}^{\mathrm{TS}}(t)/a_{\mathrm{d},n}^{\mathrm{BS}}(t)$ Binary variables indicating if TS/BS is discharged

Heat power output of CHP

$P_{\mathrm{c},n}^{\mathrm{BS}}(t)/P_{\mathrm{d},n}^{\mathrm{BS}}(t)$ Charging/discharging power of BS

$$X_t = \{H_n^{\mathrm{CHP}}(t), P_n^{\mathrm{CHP}}(t), \underline{a_{\mathrm{d},n}^{\mathrm{BS}}(t), a_{\mathrm{c},n}^{\mathrm{BS}}(t), a_{\mathrm{d},n}^{\mathrm{TS}}(t), a_{\mathrm{c},n}^{\mathrm{TS}}(t)}, \underline{H_{\mathrm{c},n}^{\mathrm{TS}}(t), H_{\mathrm{d},n}^{\mathrm{TS}}(t)}, \underline{P_{\mathrm{c},n}^{\mathrm{BS}}(t), P_{\mathrm{d},n}^{\mathrm{BS}}(t)}, \underline{P_n^{\mathrm{RES}}(t), Q_n^{\mathrm{RES}}(t)}, X_t^{\mathrm{A}}\}.$$

Active power output of CHP

Active/reactive power output of RES

$H_{\mathrm{c},n}^{\mathrm{TS}}(t)/H_{\mathrm{d},n}^{\mathrm{TS}}(t)$ Charging/discharging power of TS

X_t^A表示：`optimal energy flows in IETS, e.g., mass flow rates and flow temperature of TDN, voltages and branch flows of EDN`

4 状态转移

$$E_n^{\mathrm{BS}}(t) = E_n^{\mathrm{BS}}(t-1) + (\eta_{\mathrm{BS,c}} P_{\mathrm{c},n}^{\mathrm{BS}}(t) - P_{\mathrm{d},n}^{\mathrm{BS}}(t)/\eta_{\mathrm{BS,d}})\Delta t, \tag{14a}$$

$$E_n^{\mathrm{TS}}(t) = (1-\eta_{\mathrm{TS,d}})E_n^{\mathrm{TS}}(t-1) - H_{\mathrm{d},n}^{\mathrm{TS}}(t)\Delta t + \eta_{\mathrm{TS,c}} H_{\mathrm{c},n}^{\mathrm{TS}}(t)\Delta t. \tag{14b}$$

备注：

$\eta_{\mathrm{BS,c}}/\eta_{\mathrm{BS,d}}$ Charging/discharging efficiency of BS

$\eta_{\mathrm{TS,c}}/\eta_{\mathrm{TS,d}}$ Charging/decay efficiency of TS

$$E_n^{\mathrm{BS}}(t) = E_n^{\mathrm{BS}}(t-1) + (\underline{\eta_{\mathrm{BS,c}}} P_{\mathrm{c},n}^{\mathrm{BS}}(t) - P_{\mathrm{d},n}^{\mathrm{BS}}(t)/\underline{\eta_{\mathrm{BS,d}}})\Delta t,$$

$$E_n^{\mathrm{TS}}(t) = (1-\underline{\eta_{\mathrm{TS,d}}})E_n^{\mathrm{TS}}(t-1) - H_{\mathrm{d},n}^{\mathrm{TS}}(t)\Delta t + \underline{\eta_{\mathrm{TS,c}}} H_{\mathrm{c},n}^{\mathrm{TS}}(t)\Delta t.$$

寻找最优策略：

$$V_t(S_t) = \min_{X_t \in \Pi_t} (C_t(S_t, X_t) + \mathbf{E}(V_{t+1}(S_{t+1}))). \tag{15}$$

$$X_t = \operatorname*{argmin}_{X_t \in \Pi_t} (C_t(S_t, X_t) + \mathbf{E}(V_{t+1}(S_{t+1}))). \tag{16}$$

备注：

- 贝尔曼方程：又叫动态规划方程，是以`Richard Bellman`命名的，表示动态规划问题中相邻状态关系的方程。某些决策问题可以按照时间或空间分成多个阶段，每个阶段做出决策从而使整个过程取得效果最优的多阶段决策问题，可以用动态规划方法求解。某一阶段最优决策的问题，通过贝尔曼方程转化为下一阶段最优决策的子问题，从而初始状态的最优决策可以由终状态的最优决策(一般易解)问题逐步迭代求解。存在某种形式的贝尔曼方程，是动态规划方法能得到最优解的必要条件。绝大多数可以用最优控制理论解决的问题，都可以通过构造合适的贝尔曼方程来求解。

3.2 monotone-ADP

加速计算：

$$V_t^x(S_t^x) = \mathbf{E}[\min_{X_{t+1} \in \Pi_{t+1}} (C_{t+1}(S_{t+1}, X_{t+1}) + V_{t+1}^x(S_{t+1}^x))], \tag{17}$$

$$X_t = \operatorname*{argmin}_{X_t \in \Pi_t} (C_t(S_t, X_t) + V_t^x(S_t^x)). \tag{18}$$

最终的`MILP`：

$$X_t = \underset{X_t \in \Pi_t}{\operatorname{argmin}}(C_t(S_t, X_t) + \sum_{g \in G} \gamma_g \overline{V}_t^x(\ominus R_g)), \tag{20a}$$

subject to (1a)-(3b), (4a)-(4l), (4n), (5a)-(7h), and (8).

$$E_n^{\mathrm{BS},x}(t) = E_n^{\mathrm{BS},x}(t-1) + (\eta_{\mathrm{BS,c}} P_{\mathrm{c},n}^{\mathrm{BS}}(t) - P_{\mathrm{d},n}^{\mathrm{BS}}(t)/\eta_{\mathrm{BS,d}})\Delta t, \tag{20b}$$

$$E_n^{\mathrm{TS},x}(t) = (1 - \eta_{\mathrm{TS,d}}) E_n^{\mathrm{TS},x}(t-1) - H_{\mathrm{d},n}^{\mathrm{TS}}(t)\Delta t + \eta_{\mathrm{TS,c}} H_{\mathrm{c},n}^{\mathrm{TS}}(t)\Delta t, \tag{20c}$$

$$R_t^x = \sum_{g \in G} \gamma_g R_g, \tag{20d}$$

$$\sum_{g \in G} \gamma_g = 1. \tag{20e}$$

Algorithm 1. ***off-line pre-learning process of monotone-ADP*.**

Step 1) generate a set of training samples Ω, set the initial $\overline{V}_t^{x,0}$, and set maximum iteration N.
Step 2) set $n = 1$.
Step 3) choose a sample $\omega \in \Omega$ and set $t = 1$.
Step 4) select $R_t^{x,n} \in \mathbb{R}$, observe exogenous information, obtain real-time dispatch at time-slot $t+1$ by solving (20a), and observe the specific value $\overline{v}_t^{x,n}$ at $R_{t+1}^{x,n}$ by solving.
$\overline{v}_t^{x,n}(R_t^{x,n}) = \min(C_{t+1}(S_{t+1}, X_{t+1}) + \overline{V}_{t+1}^{x,n-1}(R_{t+1}^{x,n})).(21)$
Step 5) update value function at $R_t^{x,n}$ using step size α_n.
$\overline{z}_t^{x,n}(R_t^{x,n}) = \alpha_n \overline{v}_t^{x,n}(R_t^{x,n}) + (1-\alpha_n)\overline{V}_t^{x,n-1}(R_t^{x,n}).(22)$
Step 6) perform monotonicity preservation projection Π_M to maintain the value function monotonicity and update new value function approximation [27].
$\overline{V}_t^{x,n} = \Pi_M(\overline{V}_t^{x,n-1}, \overline{z}_t^{x,n}(R_t^{x,n})).(23)$
Step 7) set $t = t+1$, return to Step 4 if $t < T$.
Step 8) set $n = n+1$, return to Step 3 if $n < N$.

3.3. ADP-IL

Algorithm 2. ***off-line pre-learning process of ADP-IL.***

Step 1) generate a set of expert demonstrations Ω_E and set the initial $\overline{V}_t^{x,0} = \overline{V}_{\max,t}^{x}$.
Step 2) set $n = 1$.
Step 3) choose a sample $e \in \Omega_E$ and set $t = 1$.
Step 4) select $R_t^{x,n} = R_t^{x,e}$ and update $\overline{v}_t^{x,n}(R_t^{x,e})$ where.
$\overline{v}_t^{x,n}(R_t^{x,e}) = \sum_{i=t}^{T} C_i^e(S_i^e, X_i^e)$.(25)
Step 5) update the value function using step size α_n and then perform monotonicity preservation projection Π_M by solving (24).
Step 6) set $t = t + 1$, return to Step 4 if $t < T$.
Step 7) set $n = n + 1$, return to Step 3 until all demonstrations are used.
Step 8) run Algorithm 1 with initial $\overline{V}_t^{x,0}$ obtained by imitation learning.

3.4. ADP-IL based real-time IETS dispatch

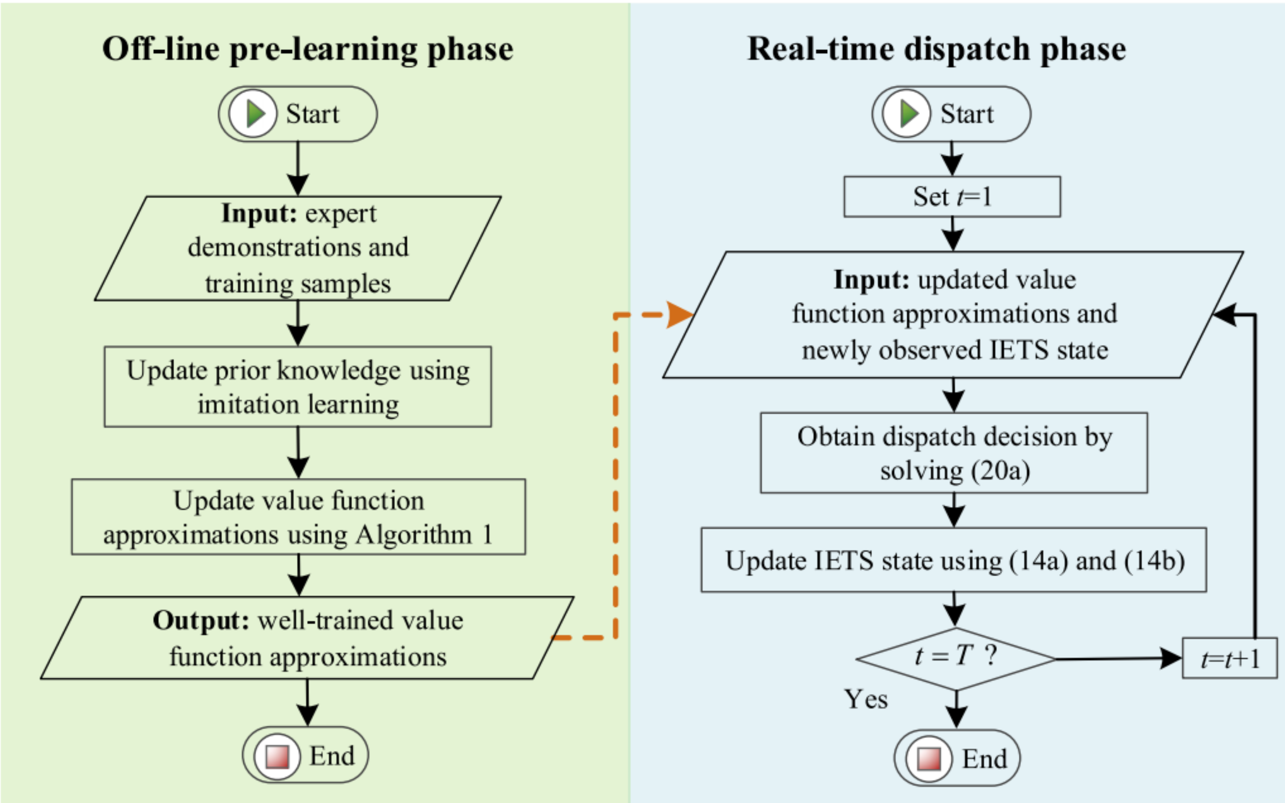

Fig. 1. Procedures of ADP-IL.

Nomenclature

Acronyms:

ADP	Approximate dynamic programming
BS/TS	Battery storage/thermal storage
CHP	Combined heat and power
IETS	Integrated electricity and thermal system
EDN/TDN	Electrical/thermal distribution networks
OOS	Optimal off-line solution
RES	Renewable energy source
MPC	Model predictive control