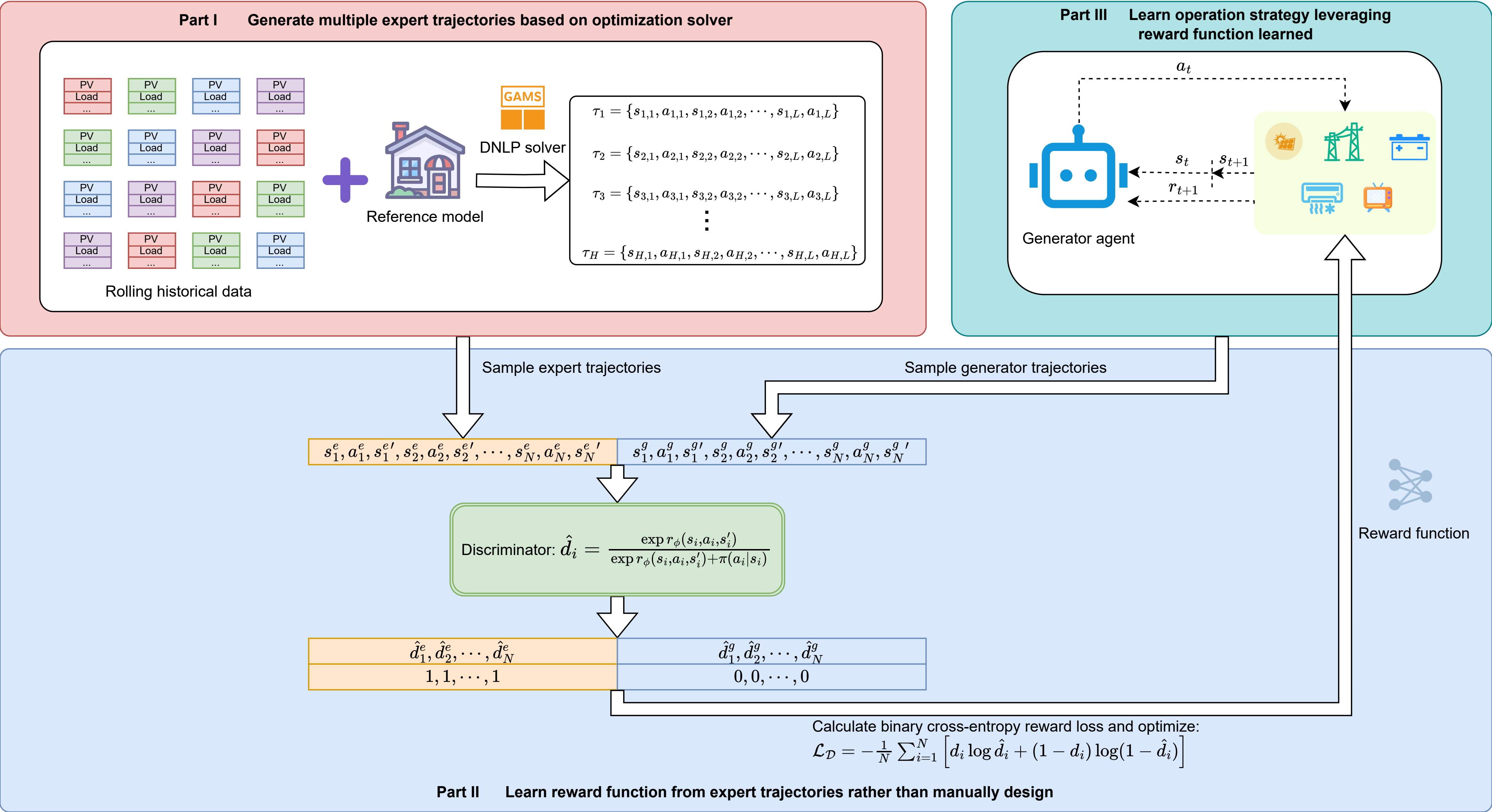
Part I Generate multiple expert trajectories based on optimization solver
PV
Load
...
Rolling historical data
Reference model
GAMS
DNLP solver
$\tau_1 = \{s_{1,1}, a_{1,1}, s_{1,2}, a_{1,2}, \cdots, s_{1,L}, a_{1,L}\}$
$\tau_2 = \{s_{2,1}, a_{2,1}, s_{2,2}, a_{2,2}, \cdots, s_{2,L}, a_{2,L}\}$
$\tau_3 = \{s_{3,1}, a_{3,1}, s_{3,2}, a_{3,2}, \cdots, s_{3,L}, a_{3,L}\}$
$\vdots$
$\tau_H = \{s_{H,1}, a_{H,1}, s_{H,2}, a_{H,2}, \cdots, s_{H,L}, a_{H,L}\}$
Part III Learn operation strategy leveraging reward function learned
a_t
s_t
s_{t+1}
r_{t+1}
Generator agent
Sample expert trajectories
Sample generator trajectories
$s_1^e, a_1^e, s_1^{e\prime}, s_2^e, a_2^e, s_2^{e\prime}, \cdots, s_N^e, a_N^e, s_N^{e\prime}$
$s_1^g, a_1^g, s_1^{g\prime}, s_2^g, a_2^g, s_2^{g\prime}, \cdots, s_N^g, a_N^g, s_N^{g\prime}$
Discriminator: $\hat{d}_i = \frac{\exp r_\phi(s_i,a_i,s_i')}{\exp r_\phi(s_i,a_i,s_i')+\pi(a_i|s_i)}$
$\hat{d}_1^e, \hat{d}_2^e, \cdots, \hat{d}_N^e$
$1, 1, \cdots, 1$
$\hat{d}_1^g, \hat{d}_2^g, \cdots, \hat{d}_N^g$
$0, 0, \cdots, 0$
Reward function
Calculate binary cross-entropy reward loss and optimize:
$\mathcal{L}_\mathcal{D} = -\frac{1}{N}\sum_{i=1}^{N}\left[d_i \log \hat{d}_i + (1-d_i)\log(1-\hat{d}_i)\right]$
Part II Learn reward function from expert trajectories rather than manually design