Distributed Deep Reinforcement Learning for Intelligent Load Scheduling in Residential Smart Grids

CS885 Paper Presentation

Alexander James

November 22, 2021

University of Waterloo

Problem Overview

Context

- Energy consumption in residential areas is increasing and improvements in energy efficiency of the power grid and appliances is not keeping pace with the increase in demand.

The Problem

- **scheduling appliance power consumption** to minimize the load on the residential power grid.

Appliance Types

Shiftable Appliances		Non-shiftable Appliances
Interruptible	Non-interruptible	
EV	Dishwasher	Water Heater
Air Conditioner	Washing Machine	Lighting
Furnace		Refrigerator
		Kitchen Appliances
		Other Appliances

Problem Formulation

Entities Involved

There are several entities involved in the problem formulation

1. *Energy Consumption Controllers (ECC)*: Each household in the residential area has an ECC that schedules the power consumption for each of the appliances in the households.
2. *Aggregator*: A component managed by a trusted third-party entity that collects the power consumption information for each household from the ECC.

Notation

- Denote the set of households as $\mathcal{N}$.
- Time is discretized and divided into equal length timeslots denoted as $\mathcal{T} = \{1, 2, \ldots, T\}$, where T denotes the fixed scheduling horizon.
- The set of appliances in a household $i \in \mathcal{N}$ is denoted as $\mathcal{A}_i$.
- $s_{i,j,t}$ denotes the state of appliance j in household i at time t. The state is the value $E_{i,j,t}$ which is the energy required by appliance j in household i at time t.

The High-Level Objective

- Let NSA_i, NIA_i and IA_i denote the sets of *non-shiftable* appliances, non-interruptible appliances and interruptible appliances respectively for each household $i \in \mathcal{N}$.
- There is no scheduling decision to be made for all appliances $j \in NSA_i$ because these appliances must be given power immediately. Thus in each timeslot, ECC_i needs to compute $P_{i,t}$ which is the amount of power consumed by all appliances $j \in NIA_i \cup IA_i$ for household i.

Pricing Model

The equation:

$$\lambda_t = \alpha_1 (L_t)^2 + \alpha_2 L_t + \alpha_3$$

Describes the cost of energy at timeslot t (λ_t) as a function of the load on the power grid (L_t). α is a vector in $\mathbb{R}^3$ that allows energy providers to alter the price of energy depending on their infrastructure and how efficient the production of energy scales.

Pricing Model

The load (L_t) on the grid at a given timeslot t is computed as

$$L_t = \sum_{i \in \mathcal{N}} P_{i,t} + \sum_{i \in \mathcal{N}} \sum_{j \in NSA_i} E_{i,j,t}$$

The first term in the sum is the power consumed by the non-interruptible and interruptible appliances and the second term is the power consumed by the non-shiftable assignments.

Pricing Model

Use a *Real-time pricing model* (RTP) which encourages ECCs to minimize the peak-to-average ratio of the load on the power grid. The real-time price of energy in a given timeslot t is computed as

$$RTP_t = \begin{cases} \lambda_t, & 0 \leq L_t \leq \delta_1 \\ \sigma_1 \lambda_t, & \delta_1 < L_t \leq \delta_2 \\ \sigma_2 \lambda_t, & L_t > \delta_2 \end{cases}$$

Solution

Consumption Scheduling Game

- In the consumption scheduling game, each *ECC* must schedule the energy consumption of the NIAs and the IAs in its corresponding household.
- Since the price of energy (RTP_t) is computed in real time, each *ECC* must perform scheduling *without knowing the price of energy in the current timeslot*.

Consumption Scheduling Game

Then the consumption scheduling game can be formally described as follows:

- *States*: $s_{i,j,t}\,\forall j \in \mathcal{A}_i, i \in \mathcal{N}$
- *Observations*: States at time t and $t-1$ as well as RTP_t and RTP_{t-1}.
- *Actions:* $P_{i,t}\,\forall i \in \mathcal{N}$
- *Reward Function*: Described in subsequent slides.

Reward Function

The cost of electricity at time t for the ith household:

$$r^1_{i,t} = P_{i,t} \times RTP_t$$

Penalize agents for not satisfying energy requests within the time horizon:

$$r^2_{i,t} = \begin{cases} 0, & t \neq T \\ \epsilon_1, & t = T, E_{i,j,t} > 0 \\ \epsilon_2, & t = T, E_{i,j,t} = 0 \end{cases}$$

Where $\epsilon_1 < 0$ and $\epsilon_2 > 0$.

Reward Function

Now the reward function can be defined as follows:

$$r_{i,t} = r_{i,t}^2 - r_{i,t}^1$$

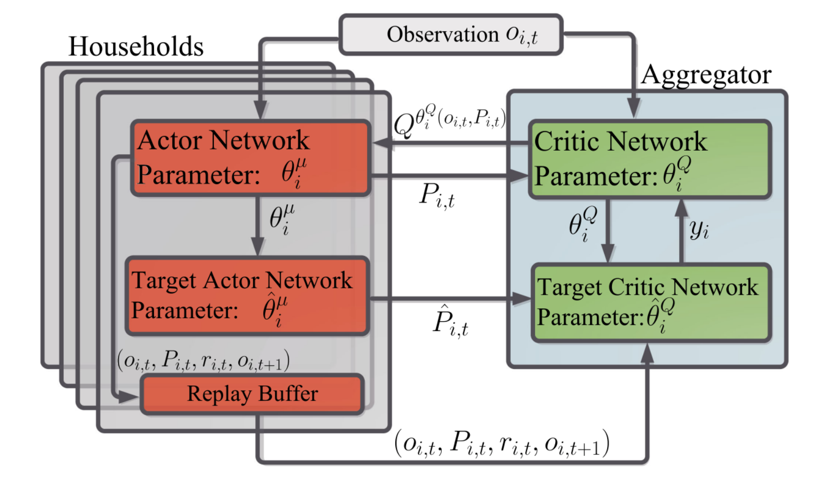
The actual goal of the RL problem is to maximize the cumulative discounted reward:

$$R(o_{i,k}, P_{i,k}) = \sum_{t=k}^{T} (\gamma_i)^{t-k} r_{i,t}$$

where γ_i is the discount factor for household i.

Distributed Deep-RL Solution

Distributed RL architecture



Critic update step

$$\min_{\theta_i^Q} \mathbb{E}\left[Q^{\theta_i^Q}(o_{i,t}, P_{i,t}) - y_i\right]^2$$

where

$$y_i = r_{i,t} + \gamma \hat{Q}^{\theta_i^Q}(o_{i,t+1}, \hat{P}_{i,t+1})$$

Actor update step

$$\nabla_{P_{i,t}} \pi_{\theta_i^Q}(o_{i,t}, P_{i,t})$$

A note on exploration

- Policy network encodes a deterministic actor, which means that exploration must be implemented explicitly.
- Implement exploration by adding a random noise term to the output of each of the actor networks.
- Details of the noise term are not specified in the paper.

Distributed RL algorithm

Algorithm 1: Distributed Power Consumption Scheduling (DPCS).

Initialize actor and critic networks with random weights θ_i^μ and θ_i^Q

The weight of target actor and target critic networks are assigned as $\hat{\theta}_i^\mu \leftarrow \theta_i^\mu$ and $\hat{\theta}_i^Q \leftarrow \theta_i^Q$

Initialize the replay buffer $\mathcal{D}_i$ for each ECC and set $l = 0$

for *episode* = 1 **to** M_{ep} **do**

    Obtain the observation $o_{i,t}$ for all households

    for t = *1* **to** T **do**

        Obtain $P_{i,t}$ for all i

        Execute $P_{i,t}$ and obtain $o_{i,t+1}$ for all i

        Store $(\mathbf{s}_{i,t}, P_{i,t}, r_{i,t}, \mathbf{s}_{i,t+1})$ to $\mathcal{D}_i$

        Allocate the power to the appliances

        $l = l + 1$

    if $l == \beta$ **then**

        Every ECC samples M data from $\mathcal{D}_i$ to aggregator

        for *ECC* = *i* **to** N **do**

            Calculate the y_i based on (14) and (15)

            Update the critic by minimizing 13

            Update the actor network by (11);

        Update the parameters of target networks (16)

        $l = 0$

Use $P_{i,t}{=}\pi_{\theta_i^\mu}(o_{i,t})$ for real-time consumption scheduling

Evaluation

Dataset

- Use a dataset * from Peach Street incorporated which includes the energy consumption profiles for more than 1000 households.

*Available at `https://dataport.cloud`

Simulation Parameters

Parameter Name	Parameter Value
α_1	0.02
α_2	0.02
α_3	0.50
σ_1	1.1
σ_2	1.3
δ_1	50 kW
δ_2	100 kW
ϵ_1	-60 $\times$ unfulfilled demand
ϵ_2	50

Baseline Algorithms

C-DDPG Centralized Deep Deterministic Policy Gradient. In this algorithm the DDPG agent has all the observations from all the ECCs.

D-DDPG Distributed Deep Deterministic Policy Gradient. This variant uses a single DDPG agent for each ECC and trains them all separately (*i.e.*, without using a centralized critic)

Average Reward

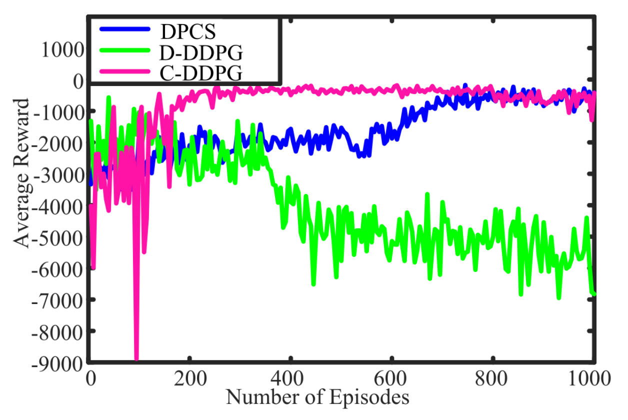


Figure 1: Average reward during training for DPCS, D-DDPG and C-DDPG

Reward Per-Household

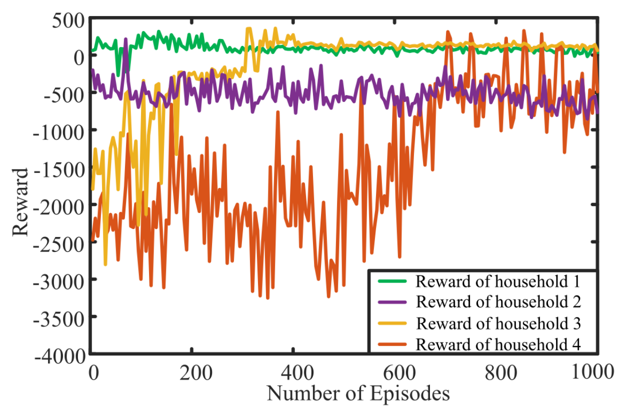


Figure 2: Reward per-household during training for DPCS

Load Profiles With and Without Scheduling

		Original	With scheduling
4 Households	peak (kW)	8.9874	8.3517
	mean (kW)	5.8644	5.7171
	Var	2.4648	2.1378
	PAR	0.4269	0.3790
	Cost (dime)	205.9080	193.3008
10 Households	peak (kW)	19.5222	18.4471
	mean (kW)	12.9803	12.9093
	Var	10.1750	9.3615
	PAR	0.4081	0.3570
	Cost (dime)	1483.0473	1369.6979
50 Households	peak (kW)	97.6112	92.4692
	mean (kW)	64.9013	64.6250
	Var	254.3740	226.7057
	PAR	0.4081	0.3597
	Cost (dime)	15813.3927	15053.4893

Load profiles for individual households

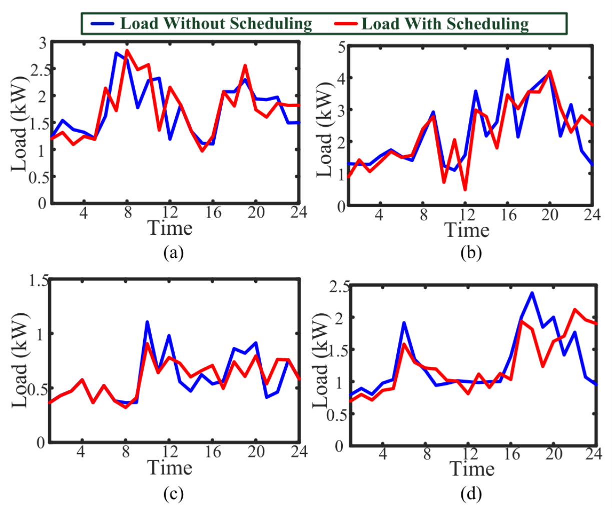


Figure 3: Load profiles per-household

Related Work

Related work using RL

Reference Number	Problem Description	Approach/Algorithm
[18]	Scheduling appliance power consumption	Q-learning
[19]	Scheduling appliance power consumption	Q-learning
[25]	Scheduling appliance power consumption	Deep Q-learning
[26]	Forecast electricity price & schedule EV charging	Deep Q-learning
[27]	Scheduling appliance power consumption	Policy gradient
[28]	Scheduling heating, ventilation and AC systems in households	Policy gradient

Related work not using RL

Reference Number	Problem Description	Approach/Algorithm
[4, 5]	Scheduling appliance power consumption	Game theoretic techniques
[6]	Scheduling appliance power consumption with real-time pricing	Genetic algorithm
[7]	Industrial power consumption scheduling	Mixed integer linear programming
[8]	Incentive scheme to encourage households to participate in load scheduling	Smoothing and regression
[9]	Scheduling appliance power consumption with privacy constraints	Mixed integer linear programming

Novelty of this work

- Deep RL formulation with continuous action space.
- Distributed RL solution based on deterministic policy gradient and actor critic networks.
- Distributed RL architecture ensures that households do not need to share power consumption profile with central authority.

Conclusion

Summary of Major Contributions

- Formulate scheduled energy consumption as a non-cooperative stochastic game.
- Use a distributed actor-critic method with deterministic policy gradient updates for the actors to produce energy consumption schedules for households.
- The schedules produced by DCPS are able to reduce the peak to average ratio of energy load by up to 12% in some cases.

Future Work

- Include household feedback as part of the input state for the actor network.

Questions?