Energy Informatics Mandatory Assignment 1

Lucas Georges Gabriel Charpentier, Sondre Wold

October 2023

Introduction

In this assignment, we aim to develop efficient control policies for microgrids of different configurations and sizes. These policies are developed using reinforcement learning, where we optimize a policy based on the monetary cost of operating the microgrid given an energy load from groups of residential homes.

Data

The input to our microgrid and RL algorithms, depending on the specific configuration, includes data points per hour for a whole year from the following:

- Solar irradiance
- Wind speed
- Energy consumption from residential houses.
- Grid price

In Tables 1 to 4 we include statistics describing the price of importing from the grid (Table 1), the energy consumption of different house cluster sizes (Table 2), the wind speed at the turbine location (Table 4) and the solar irradiance at the solar panel location (Table 3). As for the grid export price, it is set at a fixed 0.2$/kWh.

Statistic	Value	Statistic	Value
μ	0.102	25%	0.06
σ	0.086	50%	0.06
min	0.06	75%	0.09
max	0.35	-	-

Table 1: Statistics on the grid import prices over a year. All values are in $/kWh

Statistic	Value (25 houses)	Value (100 houses)	Value (272 houses)
μ	87	247	787
σ	37	90	302
min	27	92	334
25%	61	186	575
50%	81	227	708
75%	111	298	997
max	213	506	1686

Table 2: Statistics on electricity consumption over a year for various numbers of houses. Values are in kW

Statistic	Value	Statistic	Value
μ	225	25%	0.05
σ	307	50%	12
min	0	75%	439
max	1030	-	-

Table 3: Statistics on Solar Irradiance over a year. Values are in W/m^2

Statistic	Value	Statistic	Value
μ	3.94	25%	1.84
σ	2.76	50%	3.45
min	0	75%	5.46
max	21.39	-	-

Table 4: Statistics on Wind Speed over a year. Values are in m/s

Parameters	Value
$a(m^2)$-Panel area	300
δ-Efficiency	95
r^s_{omc}(\$/kWh)	0.95

Table 5: Solar panel parameters

Parameters	Value	Parameters	Value
v^{ci} (m/s)-Cut in speed	3	η_t-Gearbox efficiency	0.95
v^{co} (m/s)-Cut out speed	11	η_g-Generator efficiency	0.95
v^r (m/s)-Rate speed	7	θ-Power coefficient	0.593
$\rho(kg/m^3)$-Air density	1.225	r^w_{omc}(\$/kWh)	0.085
$r(m)$-Blade radius	25	N_w(unit)	1

Table 6: Wind turbine parameters

Software

We use the following task-specific packages in order to solve the assignment:

- PYTHON-MICROGRID (Gonzague Henri and Cordier, 2020): for defining a microgrid object that includes our energy modules and the interaction with a power grid.
- RLLIB (Moritz et al., 2018): for implementing and training reinforcement learning algorithms on the microgrid.

General Setup

The general information (cost, size, parameters, etc.) of the elements of the microgrid can be found in Tables 5 to 8.

Based on the above value, we decided to drop the battery from the microgrid model. Given that from Table 1, the maximum import price from the grid is 0.35\$/kWh but that the battery costs 0.95\$/kWh to either charge or discharge, there is no good argument in using the battery over importing from the grid (we assume the grid to be an infinite source of energy). The same can be said

Parameters	Value	Parameters	Value
e(kWh)-Capacity	300	b-Charging rate (kW)	2.5
SOC_{max}(%)	95	SOC_{min}(%)	5
r^b_{omc}(\$/kWh)	0.95	η-Efficiency	0.99

Table 7: Battery storage parameters

Parameters	Value
n_g(unit)-Number of generators	1
G_p(kW)-Generator capacity	600
r^g_{omc}(\$/kWh)	0.55

Table 8: Gas generator parameters

about the generator, however, in this case, the values are closer (0.35 and 0.55) therefore we keep it in the microgrid.

In terms of how we use each part of the system, for the grid, we use it as an energy balancer, when there is too much generation, we sell the excess, while if there is a lack of electricity, the grid fills in the gaps. While the other parts of the microgrid (solar PV, wind turbine, and gas generator) are left to the RL agent to decide whether to turn it on or off, and whether to sell or use it for the energy load. Finally, since we discretize the actions, we must split the gas generator production discretely rather than continuously. We decide to split it into 11 actions; action 0 means it is off, while actions 1-10 increases production by 10% of the maximum production (i.e. increments of 60kW). The energy production is denoted E_s, E_w, and E_g for solar, wind, and gas generator respectively. The total energy cost for production is denoted O_m. The energy sold back to the grid is denoted E^u_s, E^u_w, and E^u_g for solar, wind, and gas generator respectively. The energy used for the energy load is denoted E^i_s, E^i_w, and E^i_g for solar, wind, and gas generator respectively. The energy imported from the grid is denoted E_u, and the energy sold back is S_u. The energy produced and used to support the microgrid is denoted with S_i. The price to sell to the grid is P^u_g, and the price to buy from the grid is P^i_g. The total cost of importing from the grid is O_u, and the profit from exporting to the grid is O_s. The load needed to supply the households is denoted by L.

To calculate energy production for our renewable source we use Equation (1) for solar energy, and Equations (2) and (3) for wind energy.

$$E_s = \frac{S_r \cdot a \cdot \delta}{1000}, \text{ where } S_r \text{ is solar irradiance in W}/m^2 \tag{1}$$

$$R_w = 0.5 \cdot \rho \cdot \pi \cdot r^2 \cdot \bar{v_w}^3 \cdot \theta \cdot \eta_t \cdot \eta_g, \text{ where } v_w \text{ is the wind speed in } m/s \tag{2}$$

$$E_w = \begin{cases} 0 & \text{if } v^{ci} \leq v_w \leq v^{co} \\ N_w \cdot R_w \cdot \min(1, \frac{v_w - v^{ci}}{v^r - v^{ci}}) & \text{else} \end{cases} \tag{3}$$

To calculate the total used to support the energy load we use Equation (4).

$$S_i(t) = E^i_s(t) + E^i_w(t) + E^i_g(t) \tag{4}$$

To calculate the total energy exported to the grid we use Equation (5). These values can be the whole production or the extra generation that is over the load L.

$$S_u(t) = E_s^u(t) + E_w^u(t) + E_g^u(t) \tag{5}$$

The energy imported from the grid is given by Equation (6).

$$E_u(t) = L(t) + S_u(t) - S_i(t) \tag{6}$$

Then for the total cost of energy production, we use Equation (7).

$$O_m(t) = E_s(t) \cdot r_{omc}^s + E_w(t) \cdot r_{omc}^w + E_g(t) \cdot r_{omc}^g \tag{7}$$

The total cost of importing from the grid is defined by Equation (8).

$$O_u(t) = E_u(t) \cdot P_g^i(t) \tag{8}$$

The profit made by exporting to the grid is calculated with Equation (9).

$$O_s(t) = S_u(t) \cdot P_g^u(t) \tag{9}$$

For the first 3 questions, we define our cost function $C(t)$ by Equation (10).

$$C(t) = O_u(t) + O_m(t) - O_s(t) \tag{10}$$

Algorithms

For all our experiments, we compare two algorithms. These are outlined here.

Baseline As a baseline policy we randomly sample an action at a time step and execute it on the grid. The number of available actions depends on the specific configuration required by the experiments that follow.

DQN As an RL-based policy, we train a Deep Q-Network (Mnih et al., 2013), an off-policy approach. Similar to Q learning, this model tries to learn a value function that can guide the selection of actions leading to the greatest cumulative reward in interaction with an environment. Instead of estimating these values in a table using the Bellman equation, this function can be approximated using a deep neural network. This is much more flexible. DQNs use an epsilon-greedy strategy for exploration, where the model either explores the action space at random or selects the currently computed best action, given a probability that is often decayed over time. We initialise our DQN model using the following hyperparameters:

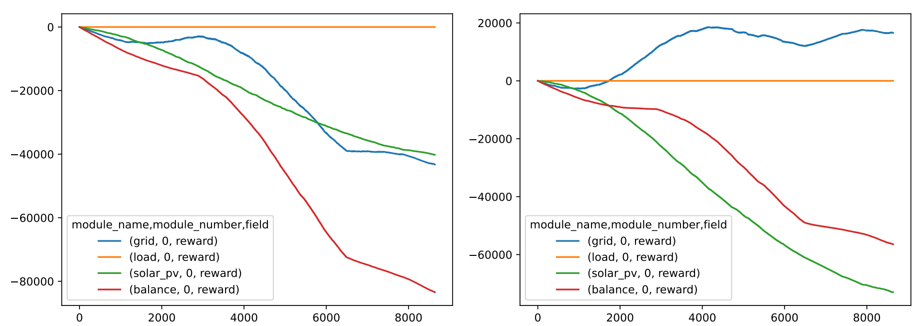
- LEARNING RATE: $1e-4$
- FUTURE REWARD GAMMA: 0.9

- TRAINING STEPS: 5 runs over all episodes.
- INITIAL EPSILON: 1
- MINIMUM EPSILON: 0

Question 1.1: RL solution for the microgrid with only solar power, and the microgrid supplies energy to the energy load

We define a Microgrid with a solar panel that has an operating cost of 0.15$ per kWh produced. The microgrid is also connected to the grid, so it can buy energy when needed and sell surplus production from the panel. The load profile comes from the energy consumption of 25 residential homes.

At a given time step, we can either: 1) turn on the solar power generation, 2) turn off the solar generation, 3) buy from the grid and 4) sell to the grid. We define our solar panel generation so that its production is always the maximum possible generation given the solar irradiance at a time step, that is: we discretize the action space for the sake of simplicity. However, we conjecture that this would also be the case in production, as there is no argument for operating a single solar panel in a fine-grained manner under the provided configurations. We would in general prefer producing as much green energy as possible, compared to buying from the grid where the source is unknown.



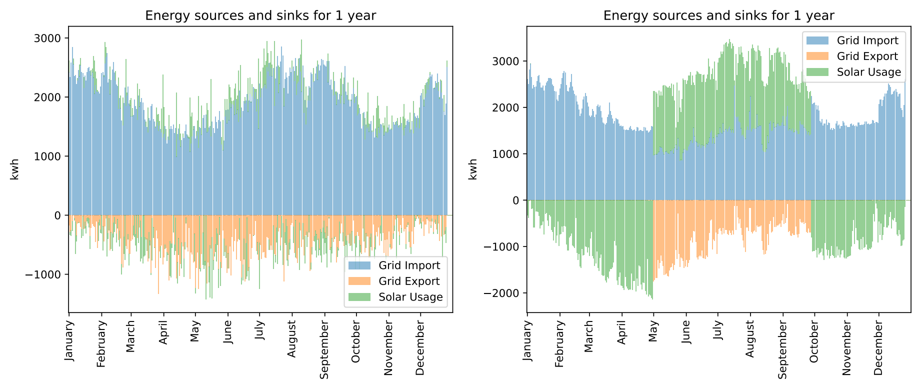
(a) Random policy

(b) DQN policy

Figure 1: The contribution of the different modalities on the reward function over one year for 25 households.

Results

The cost of our DQN-based policy as compared to a random policy can be seen in Figure 1. The random policy achieves a cumulative cost of around $-80k$\$, while the DQN reduces the cost to around $-50k$\$. The actions of our policy over a year of operation can be seen in Figure 2. We see that during the winter months, our grid is dependent on buying electricity from the grid in order to meet the load demand. From May to October, the price of electricity from the grid is higher than the production cost using the solar panel. Consequently, as can be seen from the right side of the figure, our DQN policy learns to use solar energy during this period in order to meet the load demand.



(a) Random policy usage statistics

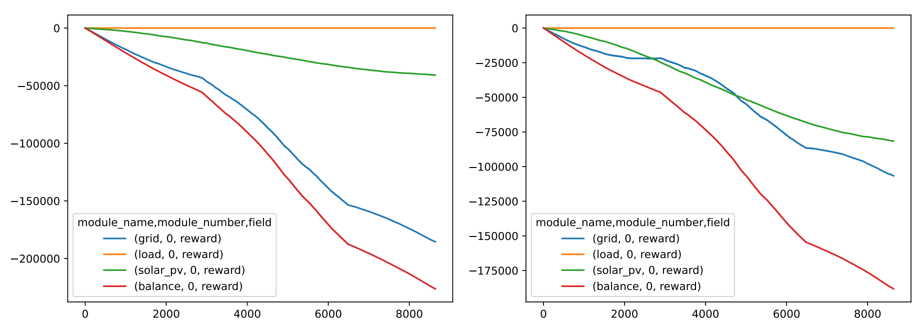
(b) DQN policy usage statistics

Figure 2: The usage of solar power and the grid for 25 households over a year of operation.

Question 1.2: the same settings as Question 1.1, except that you need to use 100 household profiles

We reproduce the setup defined above but execute it on 100 household profiles.

Results

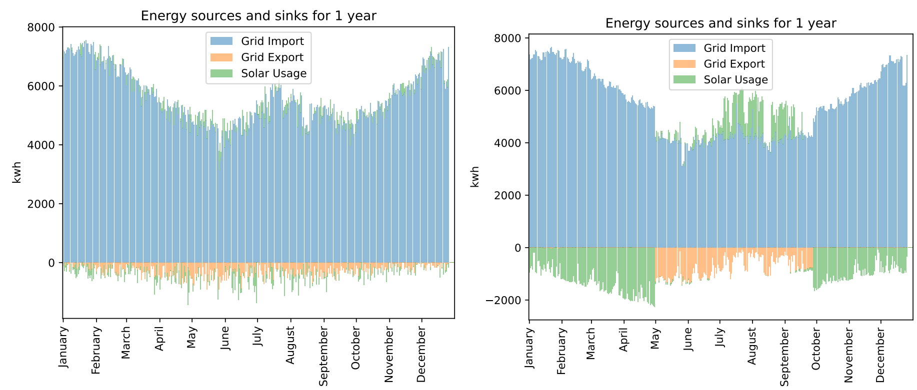


(a) Random policy

(b) DQN policy

Figure 3: The contribution of the different modalities on the reward function over one year for 100 households.

The results of our DQN-based policy can be seen in Figure 3. We see the same picture here as was the case for 25 houses. Our policy is better managing the load, and consequently need to purchase less energy from the grid in order

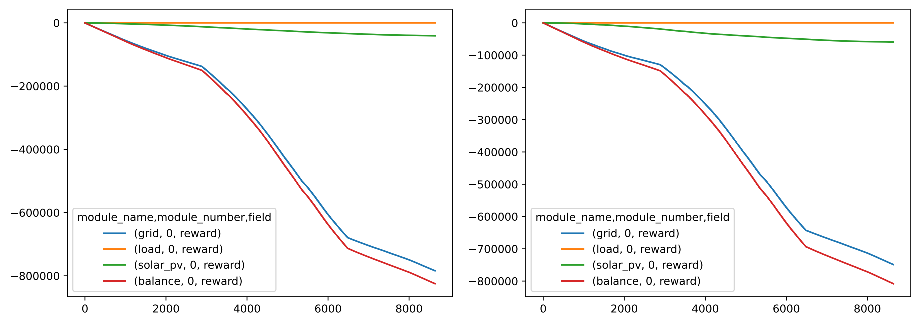


(a) Random policy usage statistics

(b) DQN policy usage statistics

Figure 4: The usage of solar power and the grid for 100 households over a year of operation.

to meet demand. Usage statistics of the actions taken can be seen in Figure 4. However, now that the load is coming from 100 houses, the solar panel is not sufficiently efficient given the solar irradiance values, and the effect of our policy weakens as we become more dependent on the grid to meet demand.



(a) Random policy

(b) DQN policy

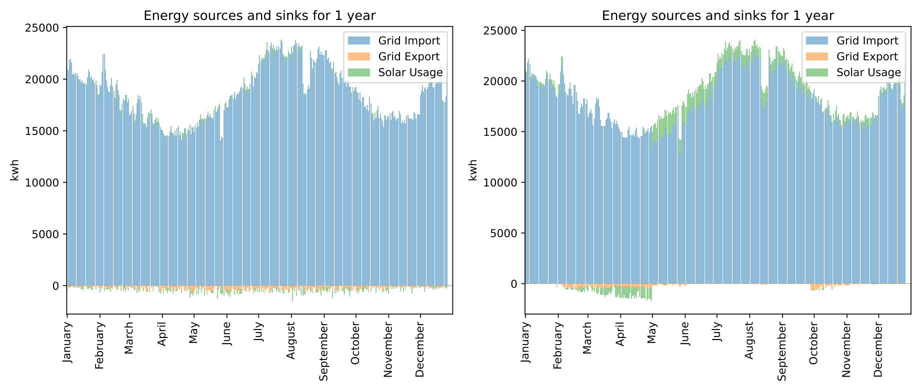
Figure 5: The contribution of the different modalities on the reward function over one year for 272 households.

Question 1.3: the same settings as Question 1.1, except that you need to use 200-500 household profiles

We reproduce the setup defined above but execute it on 272 household profiles.

Results

The results of our DQN-based policy can be seen in Figure 5. Here, the load of the group of houses is so large that our network is unable to learn an efficient policy, thereby closing in on the performance of the random model. This is confirmed in the usage plot, which can be seen in Figure 6. The interpretation of this is that the solar panel is unable to meet the load demand from 272 houses, so the microgrid has to purchase the overall majority of its power from the grid, The effect of the panel is consequently negligible, which is why the random policy is not that much worse with respect to cost.



(a) Random policy usage statistics

(b) DQN policy usage statistics

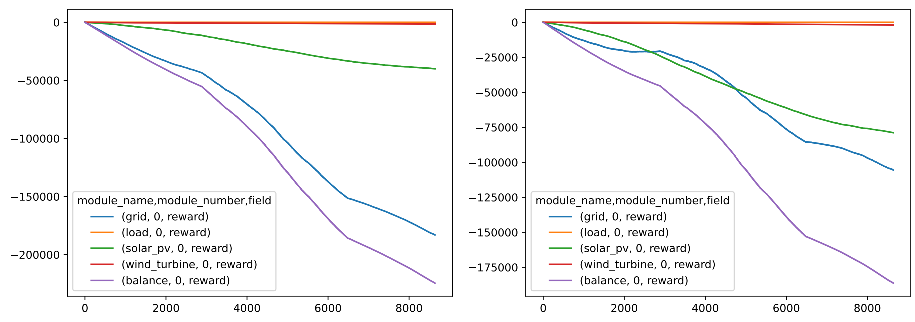
Figure 6: The usage of solar power and the grid for 272 households over a year of operation.

1 Question 2: RL solution for the microgrid with solar and wind power

For this question, we compare the effect of a learned policy on a microgrid that has both solar and wind power generation available. As with the solar panels, we discretize the actions space for the wind turbine so that it is either turned off or producing at max capacity given the wind speed, with the same rationale as for the solar panels. We compare the same algorithms as in the previous questions.

Results

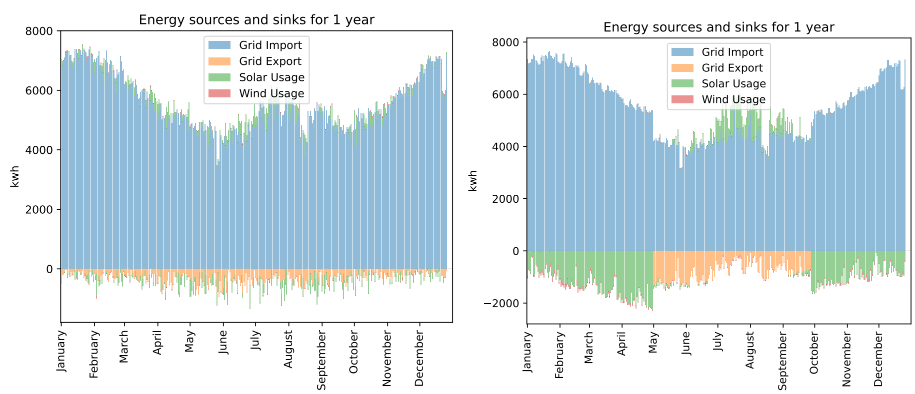
The results of our DQN-based policy compared to a random one can be seen in Figure 7. The usage statistics can be seen in Figure 8. For the cost at the end of the year, we save about $40k$\$ with our DQN-based policy network compared to sampling random actions. The effect of the wind turbine, marked with red in both of these figures, is negligible throughout the year. The capacity of the turbine as defined by the specifications in the assignment, together with the wind speed, produces too little in order to make a mark on the overall balance of the microgrid. The theoretical maximum possible generation in a day of the wind turbine is $300kWh$, and that is given an average daily wind speed of between $7-11$ meters per second. On average, we **actually** generate 70.5 kWh per day given the data. Our average load is $5928kWh$ per day, so a theoretical maximum wind production would only provide about 5% of the average daily load. It is no surprise then, that the wind production is always activated, as can be seen from the right side of Figure 8, that is the small red ticks on all the bars, as it is very cheap.



(a) Random policy

(b) DQN policy

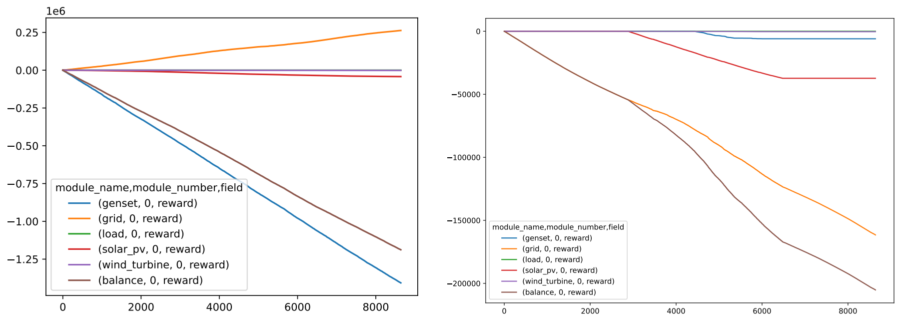
Figure 7: The contribution of the different modalities on the reward function over one year for 100 households for a microgrid with both wind and solar power.



(a) Random policy usage statistics

(b) DQN policy usage statistics

Figure 8: The usage of solar and wind power and supply to the grid for 100 households over a year of operation.



(a) Random policy

(b) DQN policy

Figure 9: The contribution of the different modalities on the reward function over one year for 100 households for a microgrid with a wind turbine, solar panel and gas turbine

2 Question 3: RL solution for the microgrid with solar & wind power and gas turbine generator considered

For this question we compare the effect of a learned policy on a microgrid that has two renewable sources, wind power and solar power, but also a gas turbine. We keep the renewables binary (either turned off or on at max capacity given the weather), but the gas turbine we split into a more fine-grained action space. We divide its generation capacity of $600kwh$ into 10 bins. At any episode then, our microgrid can either produce different levels of energy using the generator, turn the solar and wind on or off, and sell or buy from the grid.

Results

The results of our DQN-based policy compared to a random one can be seen in Figure 9. The usage statistics can be seen in Figure 10 for the random policy and Figure 11 for our trained DQN-based policy. The biggest difference between the two is that our DQN-based policy learns to ignore the generator. As stated in the introduction, the generator is very expensive to operate, so it is almost always better to buy directly from the grid. The random policy does not cater to this, naturally, which leads to a significantly higher cost at the end of the year. Our DQN-based control saves over 1 million dollars for 100 houses compared to the random policy, due to this fact. InFigure 11, we see that our policy only uses the gas generator during a brief period in summer. We conjecture that this is still sub-optimal and that this shows that our DQN is undertrained. Due to computational resource limitations, we were unable to train this network for a sufficiently long enough time.

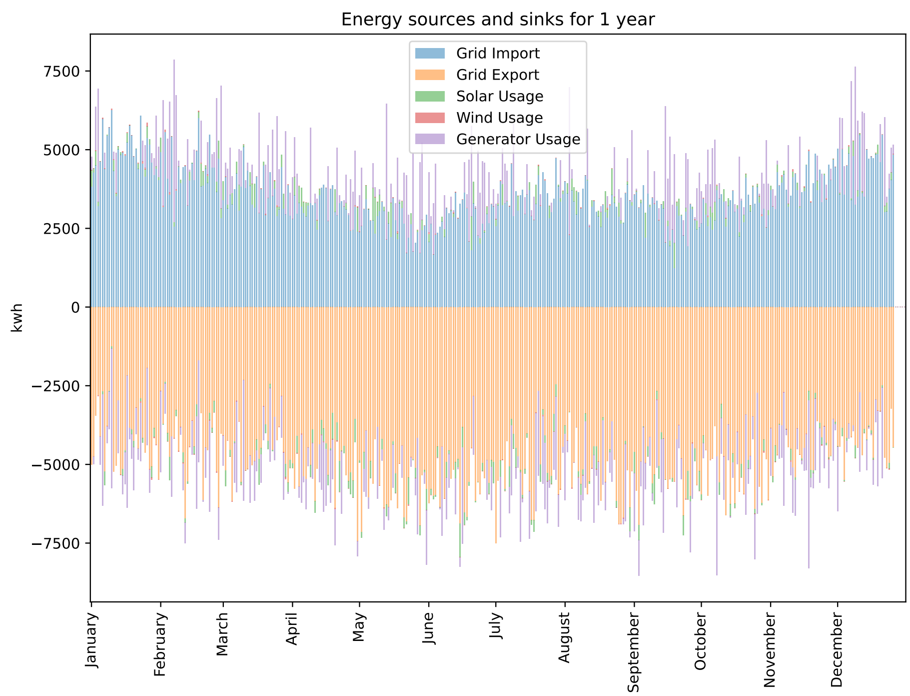


Figure 10: The energy sources and sinks for our microgrid with wind, solar and generation capacity over 1 year using a random policy.

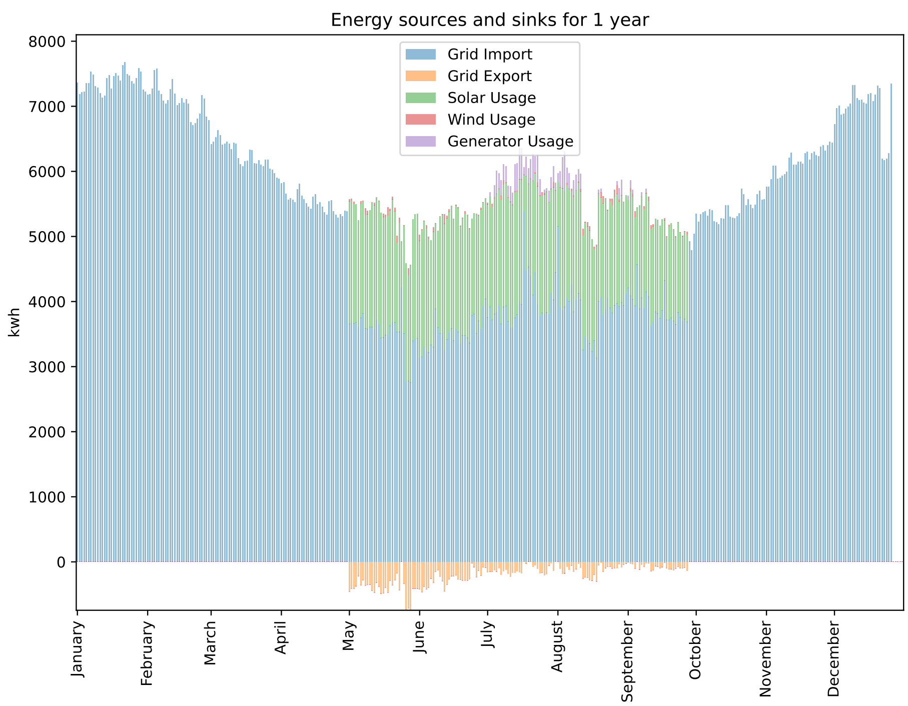


Figure 11: The energy sources and sinks for our microgrid with wind, solar and generation capacity over 1 year using a DQN policy.

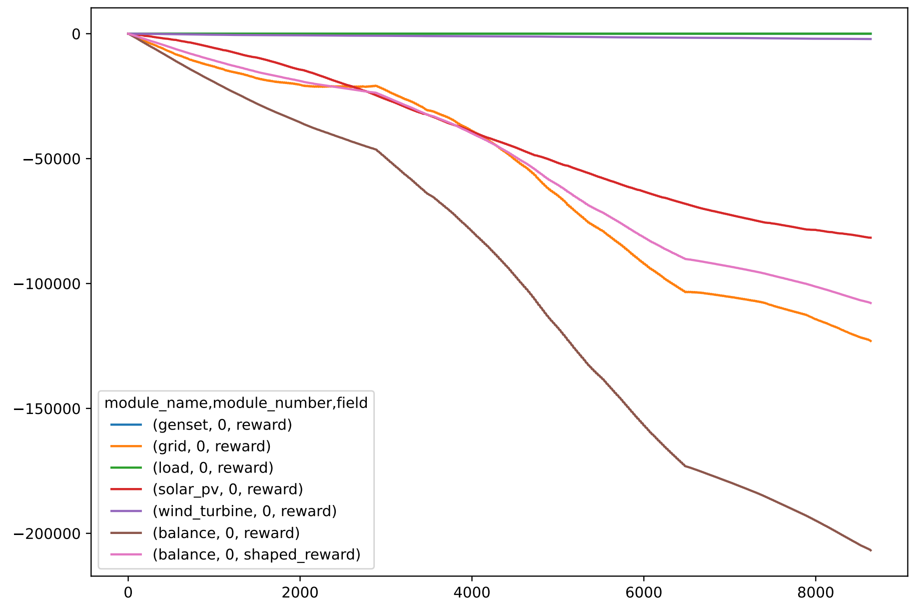


Figure 12: The contribution of the different modalities on both the normal cost function and the newly defined cost function.

Question 4: the same settings as Question 3, except that we change how the energy cost is defined

For this question, we redefine our price from importing from the grid as:

$$O_u(t) = \frac{0.25 \cdot E_u^2(t) \cdot P_g^u(t)}{1000} + 0.5 \cdot E_u(t) \cdot P_g^u(t) \tag{11}$$

We keep the same setup as above, namely a microgrid that has a solar and wind module, together with a gas turbine. The load is still from 100 houses over one year of operation.

Results

Figure 12 shows how redefining the cost function influences the balance of the microgrid with respect to each of the producing capacities. In order to compare the effect of this new cost function, compare the pink line in Figure 12 with the brown line on the right side of Figure 9. The cost is reduced from about $-200k$ to $-100k$. The interesting part, however, is the different strategies taken with the new cost function. Comparing Figure 13 and Figure 11, we now effectively sell all our produced solar energy. This is probably due to the new way to

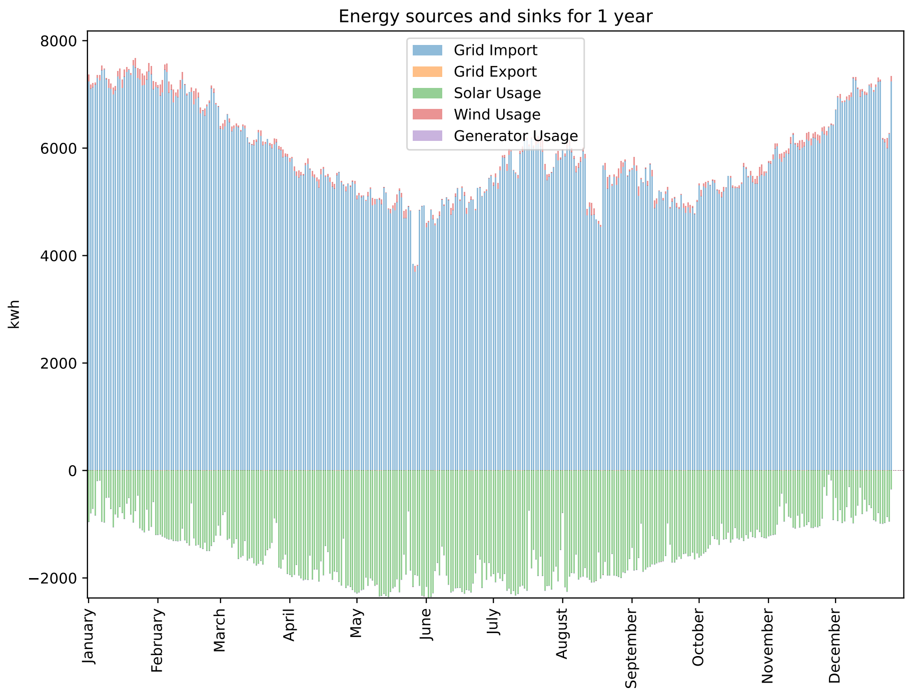


Figure 13: The action usage statistics given the redefined cost function.

calculate the price of importing from the grid. Even using the maximum values from our data (and assuming they happen at the same time), the new max price per kWh is 0.22\$. But, on average it is only (based on average values) 0.056 \$/kWh and for at least half the time it is lower than 0.05\$/kWh, in this case selling all the energy produced by solar PV is better than using it to supply the energy load.

References

Gonzague Henri, Tanguy Levent, A. H. R. A. and Cordier, P. (2020). pymgrid: An open-source python microgrid simulator for applied artificial intelligence research.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. (2013). Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.

Moritz, P., Nishihara, R., Wang, S., Tumanov, A., Liaw, R., Liang, E., Elibol, M., Yang, Z., Paul, W., Jordan, M. I., and Stoica, I. (2018). Ray: A distributed framework for emerging ai applications.