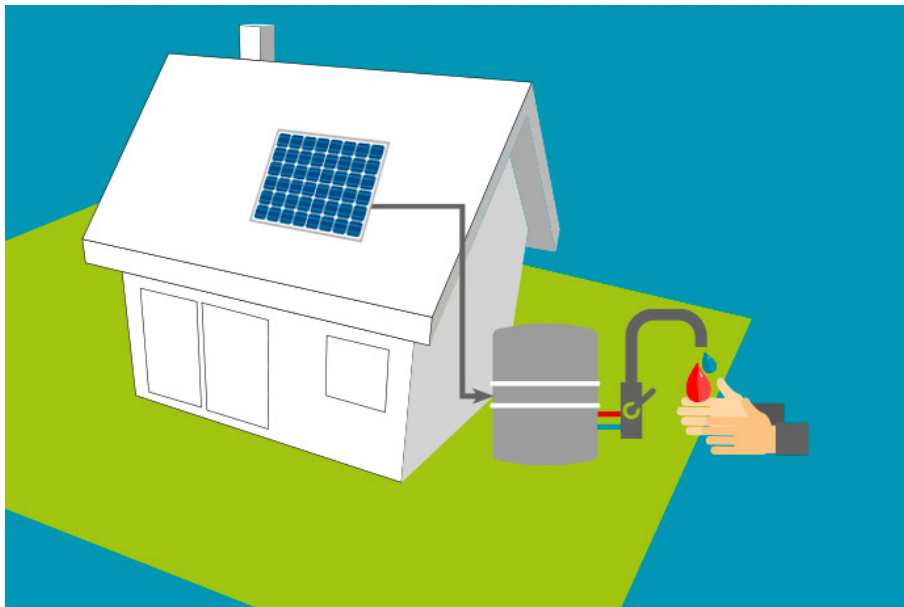

Les économies potentielles en utilisant le surplus de la production d'électricité par les panneaux photovoltaïques pour chauffer le ballon d'eau chaude à l'aide d'une résistance



Émile HEYBERGER Firas LASSOUED

Année 2022/2023

Introduction :

Pendant la journée, la production d'électricité à partir des panneaux photovoltaïques est plus grande que la consommation. L'idée est de stocker ce surplus d'énergie produite par les photovoltaïques dans l'eau chaude, c'est-à-dire d'envoyer le surplus d'électricité vers un boiler électrique qui va se charger de chauffer l'eau sanitaire pour l'utiliser plus tard dans la journée pour gaspiller moins et coûter moins cher. Donc avec l'électricité stockée dans l'eau chaude, vous ne reverrez plus votre énergie sur le réseau et vous éviterez donc de faire gonfler votre tarif.

Il existe plusieurs types de ballons d'eau chaude sanitaire :

- ballon d'eau chaude électrique à résistance électrique.
- ballon d'eau chaude électrique à résistance stéatite.
- ballon d'eau chaude électrique thermodynamique.
- ballon d'eau chaude thermique.

Les ballons d'eau chaude à résistance sont les équipements les mieux adaptés à l'autoconsommation.[1].

On a besoin d'environ 4500 kWh pour chauffer un ballon d'eau chaude sanitaire de 200 litres sur une année.[2].

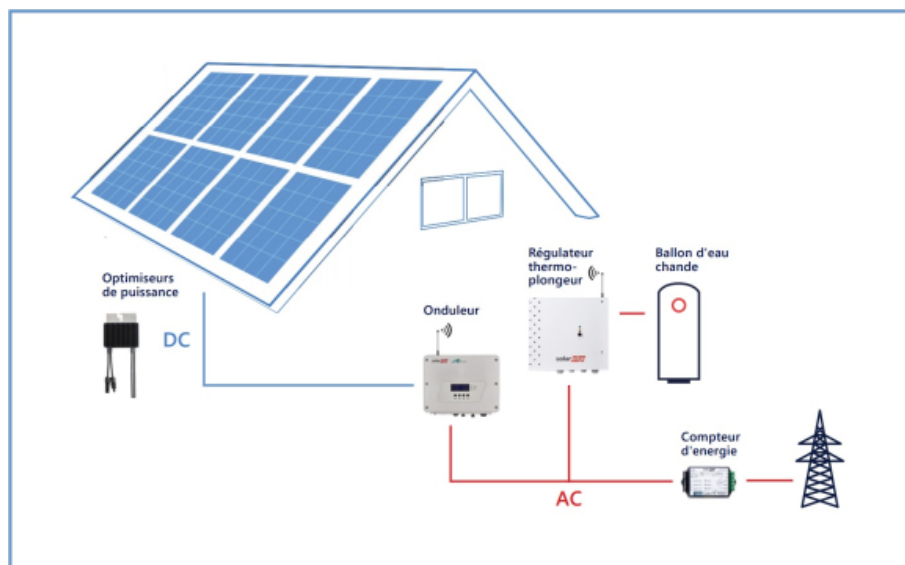


figure 1 : Chauffer l'eau par les panneaux solaires

Objectif :

Dans notre projet on va faire une étude sur Les économies potentielles en utilisant le surplus de la production d'électricité par les panneaux photovoltaïque pour chauffer le ballon d'eau chaude à l'aide d'une résistance.

Extraction des données :

On commence tout d'abord par l'extraction des données afin de les interpréter après, on télécharge les données à partir de Grafana sur un fichier .csv.

On utilise un code python pour extraire les données en utilisant un ID.

Les données choisies pour ce projet sont :

- La production totale PV.
- La consommation totale du foyer.
- Les températures de la chaudière.
- Les consommations de gaz.

Nous avons reçu une liste des différents données avec la liste suivante :

```
'id': '13272'  'name': 'Production Photovoltaïque"Global"Production instantanée'.  
'id': '5224'   'name': 'Full_Home_PZEM004/Puissance'.  
'id': '1512'  'name': 'Compteur électrique/ConsoAujourd'hui'.  
'id': '1514'  'name': 'Compteur électrique/ConsoMoyMinute'.  
'id': '1516'  'name': 'Compteur électrique/ConsoMoyHeure'.  
'id': '2004'  'name': 'Température Eau Chaude/temperature'.  
'id': '2010'  'name': 'Température de débit/temperature'.  
'id': '132879' 'name': 'Compteur Gaz""ConsoGaz3clos""ConsoJour3clos'.
```

Par exemple pour extraire les données de la production photovoltaïque totale, on utilise l'ID : 13272 et pour la consommation 5224,

```
datasheet_prod = client.query('SELECT "value" FROM "jeedom"."autogen"."132725" WHERE time > now() - 36d')  
datasheet_conso = client.query('SELECT "value" FROM "jeedom"."autogen"."5224" WHERE time > now() - 36d')
```

figure 2 : ID des données

On obtient donc un fichier excel où il existe les données pour les utiliser pour le code de Machine Learning.

Le code suivant nous permet d'afficher ces données et de les transformer en un fichier csv :

```

with open("/Users/emileheyberger/Downloads/smartsystem/dataraw_prod.csv", "w", newline='') as fp:
    writer = csv.writer(fp, dialect='excel')
    # print(header_list[1:])
    value_header = header_list_prod[1]
    offset = sum(c.isalpha() for c in value_header)
    print(offset)
    #header_list[1:] = sorted(header_list[1:], key=lambda x: int(x[offset:]))
    header_list_prod[1:] = ['value']
    print(header_list_prod)
    writer.writerow(header_list_prod)
    for line in exported_data_prod:
        # print(line)
        writer.writerow([line[kn] for kn in header_list_prod])

```

figure 3 : Extraction des données en fichier csv

Les figures ci-dessous présentent les données extraites pour la production et la consommation :

time	value
2023-01-09 00:00:07+00:00	191.0
2023-01-09 00:00:27+00:00	186.0
2023-01-09 00:00:37+00:00	185.0
2023-01-09 00:01:07+00:00	189.0
2023-01-09 00:01:16+00:00	187.0
...	...
2023-02-01 01:22:28+00:00	2724.0
2023-02-01 01:27:58+00:00	2635.0
2023-02-01 13:35:38.631741+00:00	3494.0
2023-02-01 13:35:48.711827+00:00	3490.0
2023-02-01 13:35:59.407794+00:00	3478.0

figure 4 : les données de Consommation

time	value
2023-01-09 00:00:00+00:00	-4.388200
2023-01-09 00:30:00+00:00	-4.443455
2023-01-09 01:00:00+00:00	-4.352053
2023-01-09 01:30:00+00:00	-4.393909
2023-01-09 02:00:00+00:00	-4.267100
...	...
2023-02-01 11:30:00+00:00	1144.966120
2023-02-01 12:00:00+00:00	1124.824815
2023-02-01 12:30:00+00:00	1014.370739
2023-02-01 13:00:00+00:00	938.570143
2023-02-01 13:30:00+00:00	887.468222

figure 5 : les données de production

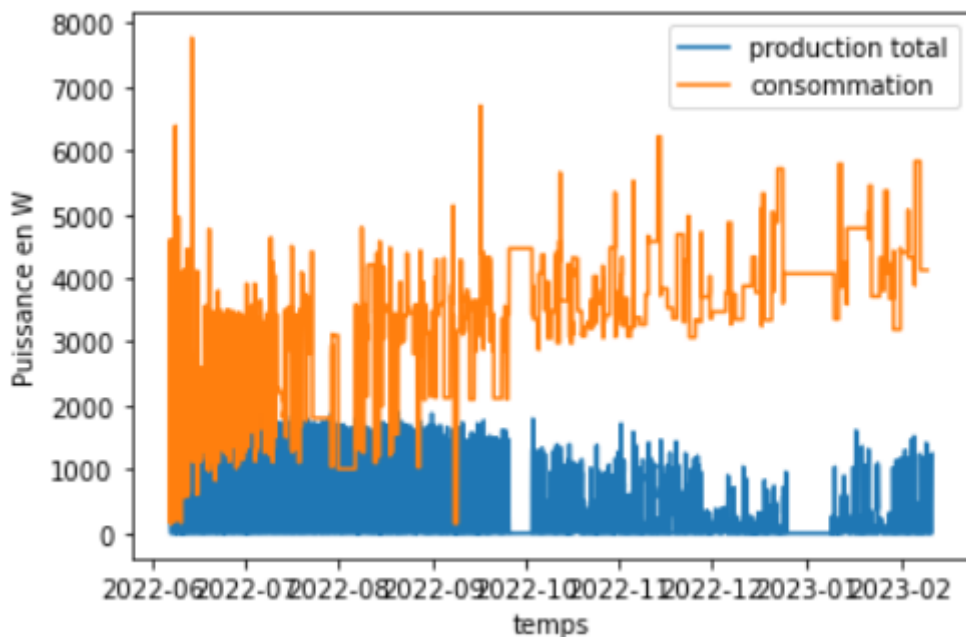


figure 6 : Evolution de la consommation et la production

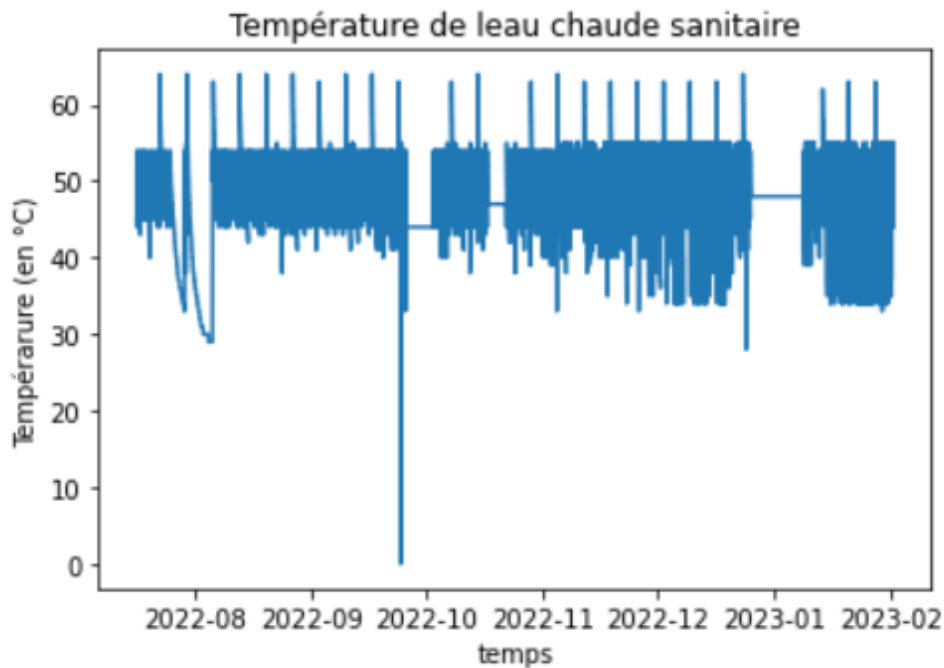


figure 7 : Température de l'eau chaude

Résultat :

Au cours des 250 derniers jours, 8.9 kWh de surplus ont été injectés sur le réseau ainsi 0.9 € ont pu être gagnés en vendant le surplus à 0,10€/kWh.

Alors que si cette énergie avait été utilisée pour chauffer l'eau chaude sanitaire, en considérant que l'énergie électrique nécessaire est équivalente à l'énergie en gaz, ceci aurait permis d'économiser 1.1 € de gaz.

Analyse du Résultat :

Ainsi ce moyen aurait permis de faire gagner sur 250 jours, 0.2 € par rapport à la vente d'électricité.

Conclusion :

En sachant qu'une résistance coûte environ 50€, ce système est rentabilisé au bout de 170 années donc a priori ce système n'est pas rentable aujourd'hui au vu de l'autoproduction de l'installation, du coût de revient de l'énergie et du prix du gaz.

Références bibliographiques :

[1] Comment piloter son chauffe-eau.

https://www.oscaro-power.com/guide/fr_FR/j-adapte-et-je-pilote-ma-consommation/piloter-son-chauffe-eau

[2] Pourquoi faut-il coupler les panneaux photovoltaïques avec le ballon d'eau chaude ECS ?

<https://www.esole.eu/coupler-les-panneaux-photovoltaïques-ecs.html>

Annexes :

Code 1 [Extraction de données]:

```
from http.server import BaseHTTPRequestHandler, HTTPServer
import urllib
import time
from influxdb import InfluxDBClient
import sys
import csv
import pandas as pd
import matplotlib.pyplot as plt

#####
# SCRIPT SETTINGS
#####
# Set the port where you want the bridge service to run
PORT_NUMBER = 1234
# InfluxDB Server parameters
INFLUXDB_SERVER_IP = '82.65.155.71'
INFLUXDB_SERVER_PORT = 8086
INFLUXDB_USERNAME = 'eleves'
INFLUXDB_PASSWORD = 'SmarthouseG2Elab'
INFLUXDB_DB_NAME = 'jeedom'
#####

client = InfluxDBClient(INFLUXDB_SERVER_IP, INFLUXDB_SERVER_PORT,
INFLUXDB_USERNAME, INFLUXDB_PASSWORD,
INFLUXDB_DB_NAME,ssl=True, verify_ssl=False)

print(client.get_list_database())

client.switch_database('jeedom')

datasheet_prod1 = client.query('SELECT "value" FROM
"jeedom"."autogen"."132725" WHERE time >= now() -250d AND time < now()-150d')
datasheet_prod2 = client.query('SELECT "value" FROM
"jeedom"."autogen"."132725" WHERE time >= now() -150d')
datasheet_conso1 = client.query('SELECT "value" FROM "jeedom"."autogen"."5224"
WHERE time >= now() -250d AND time < now()-150d')
datasheet_conso2 = client.query('SELECT "value" FROM "jeedom"."autogen"."5224"
WHERE time >= now() -150d')
datasheet_temp = client.query('SELECT "value" FROM "jeedom"."autogen"."2004"
WHERE time > now() -100d')
```

```

exported_data_prod =
list(datasheet_prod1.get_points()+list(datasheet_prod2.get_points()))
exported_data_conso =
list(datasheet_conso1.get_points()+list(datasheet_conso2.get_points()))
exported_data_temp = list(datasheet_temp.get_points())
header_list_prod = list(exported_data_prod[0].keys())
header_list_conso = list(exported_data_conso[0].keys())
header_list_temp = list(exported_data_temp[0].keys())

```

```

with open("/Users/emileheyberger/Downloads/smartsystem/dataraw_prod.csv", "w",
newline="") as fp:

```

```

    writer = csv.writer(fp, dialect='excel')
#   print(header_list[1:])
    value_header = header_list_prod[1]
    offset = sum(c.isalpha() for c in value_header)
#   print(offset)
    #header_list[1:] = sorted(header_list[1:], key=lambda x: int(x[offset:]))
    header_list_prod[1:] = ['value']
#   print(header_list_prod)
    writer.writerow(header_list_prod)
    for line in exported_data_prod:
#       print(line)
        writer.writerow([line[kn] for kn in header_list_prod])

```

```

with open("/Users/emileheyberger/Downloads/smartsystem/dataraw_conso.csv",
"w", newline="") as fp:

```

```

    writer = csv.writer(fp, dialect='excel')
#   print(header_list[1:])
    value_header = header_list_conso[1]
    offset = sum(c.isalpha() for c in value_header)
#   print(offset)
    #header_list[1:] = sorted(header_list[1:], key=lambda x: int(x[offset:]))
    header_list_conso[1:] = ['value']
#   print(header_list_conso)
    writer.writerow(header_list_conso)
    for line in exported_data_conso:
#       print(line)
        writer.writerow([line[kn] for kn in header_list_conso])

```

```

with open("/Users/emileheyberger/Downloads/smartsystem/dataraw_temp.csv", "w",
newline="") as fp:

```



```

writer = csv.writer(fp, dialect='excel')
# print(header_list[1:])
value_header = header_list_temp[1]
offset = sum(c.isalpha() for c in value_header)
# print(offset)
#header_list[1:] = sorted(header_list[1:], key=lambda x: int(x[offset:]))
header_list_temp[1:] = ['value']
# print(header_list_temp)
writer.writerow(header_list_temp)
for line in exported_data_temp:
# print(line)
writer.writerow([line[kn] for kn in header_list_temp])

```

```

df_prod=pd.read_csv('/Users/emileheyberger/Downloads/smartsystem/dataraw_prod
.csv',parse_dates=["time"],index_col="time")
df_prod.drop_duplicates(inplace=True)
df_conso=pd.read_csv('/Users/emileheyberger/Downloads/smartsystem/dataraw_co
nso.csv',parse_dates=["time"],index_col="time")
df_conso.drop_duplicates(inplace=True)
df_temp=pd.read_csv('/Users/emileheyberger/Downloads/smartsystem/dataraw_tem
p.csv',parse_dates=["time"],index_col="time")
#df_temp.drop_duplicates(inplace=True)

```

#On met les différentes données sur la même échelle de temps

```

duree='5min'
df_prod=df_prod.resample(duree).ffill()
df_conso=df_conso.resample(duree).ffill()
df_temp=df_temp.resample(duree).ffill()

```

#On remet les fichiers dans le bon format

```

df_prod.to_csv('/Users/emileheyberger/Downloads/smartsystem/dataraw_newprod.c
sv')
df_prod=pd.read_csv('/Users/emileheyberger/Downloads/smartsystem/dataraw_new
prod.csv')
df_prod["time"] =pd.to_datetime(df_prod["time"])
df_prod["time"] = df_prod["time"].dt.floor('s')

df_conso.to_csv('/Users/emileheyberger/Downloads/smartsystem/dataraw_newcons
o.csv')

```

```
df_conso=pd.read_csv('/Users/emileheyberger/Downloads/smartsystem/dataraw_newconso.csv')
df_conso["time"] =pd.to_datetime(df_conso["time"])
df_conso["time"] = df_conso["time"].dt.floor('s')
```

```
df_temp.to_csv('/Users/emileheyberger/Downloads/smartsystem/dataraw_newtemp.csv')
df_temp=pd.read_csv('/Users/emileheyberger/Downloads/smartsystem/dataraw_newtemp.csv')
df_temp["time"] =pd.to_datetime(df_temp["time"])
df_temp["time"] = df_temp["time"].dt.floor('s')
```

#On trace les graphiques

```
x_prod = df_prod['time']
y_prod = df_prod['value']
```

```
x_conso = df_conso['time']
y_conso = df_conso['value']
```

```
surplus=[]
for k in range(600):
    if y_prod[k]>y_conso[k]:
        surplus.append((y_prod[k]-y_conso[k])*5/60)
    else:
        surplus.append(0)
```

```
x_temp = df_temp['time']
y_temp = df_temp['value']
```

```
plt.plot(x_prod,y_prod)
plt.plot(x_conso,y_conso)
plt.legend(['production total','consommation'])
plt.xlabel('temps')
plt.ylabel('Puissance en W')
plt.plot()
```

Code 2 :

```
somme_surplus=0
```

```
for k in surplus:
```

```
    somme_surplus+=k/1000
```

```
vente_surplus=0.10*somme_surplus
```

```
print("Au cours des 250 derniers jours,",round(somme_surplus,1),"kWh de surplus  
ont été injecté sur le réseau ainsi",round(vente_surplus,1),"€ ont pu être gagné en  
vendant le surplus à 0,10€/kWh.")
```

```
economie_gaz=0.128*somme_surplus
```

```
economie_moyen=economie_gaz-vente_surplus
```

```
print("Alors que si cette énergie aurait été utilisé pour chauffer l'eau chaude sanitaire,  
en considérant que l'énergie électrique nécessaire et équivalente à l'énergie en gaz,  
ceci aurait permis d'économiser",round(economie_gaz,1),"€ de gaz.")
```

```
print("Ainsi ce moyen aurait permis de faire gagner sur  
250j",round(economie_moyen,1),"€ par rapport à la vente d'électricité")
```

```
rentabilité=50/(economie_moyen/250)/365
```

```
print("En sachant qu'une résistance coûte environ 50€, ce système est rentabilisé au  
bout de", round(rentabilité,1),"années donc a priori ce système n'est pas rentable  
aujourd'hui au vu de l'autoproduction de l'installation, du coût de revente de l'énergie  
et du prix du gaz.")
```