

Dimensionner une installation photovoltaïque autonome/en

 Bababa



Dernière modification le 28/11/2025

 Difficulté Moyen

 Durée 1 heure(s)

 Coût 0 EUR (€)

Description

Outils pour dimensionner une installation photovoltaïque hors réseau

Sommaire

Description
Sommaire
Introduction
Étape 1 - évaluer les
besoins
Étape 2 - dimensionner le
stockage
Étape 3 - dimensionner la
production
Étape 4 - dimensionner le régulateur de
charge
Étape 5 - Optimiser le
dimensionnement
Étape 6 - Résultats
optimisés
Commentaires

Introduction

La plupart des installations photovoltaïques sont aujourd'hui branchées sur le réseau électrique erdf/enedis, que ce soit sur des formules commerciales dites en "injection du surplus", ou en "injection totale".

Avec les vieux compteurs, il est encore possible de "faire tourner à l'envers" le compteur électrique lorsque les panneaux produisent (un peu comme si linky soustrayait de votre consommation ce que vos panneaux produisent quelle que soit l'heure à laquelle la production a lieu)

Le recours au réseau électrique est très pratique car cela évite le besoin de stocker l'électricité produite.

Cependant, que ce soit pour des raisons d'évolution sociétale (laissons nous le droit de rêver à autre modèle urbanistique où les sites autonomes écologiques sans besoins d'infrastructures sont encouragés), pour des raisons de contraintes naturelles, ou par choix, on peut souhaiter être indépendant des réseaux, 100% autonome en énergie électrique

J'ai initialement fait un petit bout de code pour dimensionner correctement en réseau autonome un mobilhome que je souhaitais rendre autonome, puis pour brancher un frigo et un congélateur (qui nécessitent un approvisionnement relativement constant, c'est à dire sans coupure)

L'ines propose déjà des outils ici:

http://ines.solaire.free.fr/pvisole_1.php

et ici:

<https://autocalcul.ines-solaire.org/etude/localisation/>

Cependant, les hypothèses sont à 1kWh produit par kWc en hiver pour le premier lien et on n'a que des moyennes pour le second lien.

En outre, on n'a pas de possibilité de "data-tester" le nombre de jours de blackout selon le dimensionnement.

Ce tuto permet donc de dimensionner en "data-testant", c'est à dire avec des hypothèses de production en hiver au jour le jour, moins "moyennée", et permet aussi de tester des dimensionnement plus économes en capacité de batterie (qui coutent encore cher en 2024).

Compte tenu du nombre de brevets déposés dans le domaine du stockage électrique des 15 dernières années, et compte tenu des évolutions géopolitiques au sein des brics, il est probable qu'on aboutisse à des coûts de stockage nettement diminués dans les prochaines années et l'algorithme pour optimiser le dimensionnement au stockage au plus juste sera peut être moins pertinent économiquement parlant dans les années à venir.

Mais un stockage à très bas coût remettrait très fortement en question le système pétro dollar, donc on a sans doute le temps de voir venir ;)

Dans tous les cas, les batteries usagées produisant pas mal de

déchet, cela reste intéressant d'avoir ce bout de logique pour dimensionner au plus juste et être dans une optique low-tech en évitant le surdimensionnement lorsqu'on peut se permettre de rares épisodes sans électricité.

Démo web interactive ici:

<https://vpn.matangi.dev/sun>

Matériaux

Outils

 Dimensionner_une_installation_photovoltaique_autonome_jrc_a_u_t_o_n_o_m_i_e.ods

Étape 1 - évaluer les besoins

Il s'agit dans un premier temps d'estimer votre consommation électrique.

Pour cela, il convient de prendre en compte les bonnes échelles temporelles:

consommation moyenne d'une part, mais aussi les variations journalières (la consommation selon l'heure de la journée) et saisonnières (la consommation en hiver et en été).

Cet exercice est intéressant à la fois pour dimensionner adéquatement la production, mais aussi pour penser comment réduire sa consommation. Par exemple, il est aberrant de chauffer avec des radiateurs électriques si votre source d'énergie est uniquement photovoltaïque.

Un autre exemple est le chauffe-eau électrique qui consomme des quantités conséquentes d'énergie qui sont relativement linéaires du nombre de douches chaudes que vous prenez et qui peuvent interroger la source d'énergie pertinente pour chauffer votre eau ou le nombre de douches pertinents pour une production non fossile (on pourra utilement s'interroger sur l'opportunité de basculer sur une source d'énergie solaire thermique et/ou chauffer au bois ou au biogaz)

NB: pour une douche de 10min à 40°C, il faut environ 3kWh, soit environ le biométhane générable par les fientes journalières d'une centaine de poules (en ordre de grandeur)

Pour mesurer votre consommation, vous pouvez utiliser votre compteur électrique.

Vous pouvez ainsi mesurer votre consommation journalière mais aussi la consommation horaire de chaque appareil électrique (le plus simple étant d'enlever le fusible au compteur de tous les appareils électriques qu'on veut sortir de la mesure).

Étape 2 - dimensionner le stockage

En site autonome, le chemin des électrons produits par les panneaux photovoltaïque est celui qui va à la moindre résistance. Ainsi, si les électrons doivent "choisir" entre un appareil électrique en fonctionnement et les batteries, ils iront alimenter l'appareil branché.

Cette considération peut être importante si vous avez une consommation électrique importante en journée (lorsque les panneaux produisent), mais dans la plupart des cas, les variations de production journalière et la différence horaire entre la production et la consommation font qu'on dimensionne les batteries sur le besoin journalier maximum.

Exemple: besoin journalier de 4kWh

On prend le pourcentage maximum de décharge accepté par les batteries selon la technologie utilisée. Par exemple 80% pour des batteries lithium.

Exemple: $(\text{besoin journalier}) / (\text{pourcentage decharge max}) = (4) / (0,8) = 5\text{kWh}$

NB: les capacités des batteries sont souvent exprimées en Ah. Pour obtenir les Ah à partir de la capacité en kWh, simplement diviser les Wh par la tension aux bornes de vos batteries. Par exemple en 24V:

$500\text{Wh} / 24\text{V} = 208\text{Ah}$

Pour dimensionner le stockage correctement en fonction du besoin journalier maximum, on va ensuite chercher à savoir combien de jours sans soleil le stockage doit pouvoir tenir vos besoins journaliers maximum

Pour trouver le nombre de jours maximum sans soleil, vous pouvez soit mesurer vous même l'ensoleillement (se reporter à mon autre tuto ici : Mesure de l'ensoleillement-luminosité avec un ordinateur monocarte (raspberry-orangepi),

soit utiliser les données téléchargeable de la modélisation jrc européenne pvgis qui font référence (attention, il s'agit d'une modélisation,

Modif 27/02: après vérification, il s'agit de données satellites horaires mondiales donc des données empiriques en entrée notamment pour les nuages mais les hypothèses du modèle sur la quantité d'irradiation qui passe à travers les nuages sont peu explicitées, voir https://joint-research-centre.ec.europa.eu/photovoltaic-geographical-information-system-pvgis/getting-started-pvgis/pvgis-data-sources-calculation-methods_en?prefLang=fr)

Pour télécharger les données jrc, taper cette commande dans votre terminal linux:

```
wget --max-redirect=10 -O output.csv "https://re.jrc.ec.europa.eu/api/v5_2/seriescalc?lat=44.203142&lon=0.616363&loss=14&angle=45&aspect=0&startyear=2005&endyear=2020&pvcaculation=1&peakpower=1&pvttechchoice=crystSi&browser=0&outputformat=csv"
```

Les parametres lat= et lon= vous permettent d'ajuster la latitude et la longitude à votre lieu

startyear= et endyear= sont les bornes de début et de fin de la periode considéré (ici 15 ans entre 2005 et 2020)

peakpower= correspond à la puissance photovoltaïque en kWc (ici 1kWc)

On utilise ensuite un bout de code python/pandas pour obtenir le nombre de jours consécutifs maximum pour qu'1 kWc produise 1000Wh, 2000Wh et 3000Wh

```
import pandas as pd
df = pd.read_csv('output.csv', skiprows=10, skipfooter=11, sep=',', engine='python')
df['time']=df['time'].astype(str)
df['time']=pd.to_datetime(df['time'],format="%Y%m%d:%H%M")
df=df.set_index('time')
daily_data = df.resample('D').sum()

max_streak = 0
current_streak = 0
current_sum=0
target=1000
streaks=[]
for value in daily_data['P']:
    if current_sum <= target:
        current_sum+=value
        current_streak += 1
        max_streak = max(max_streak, current_streak)
    else:
        streaks.append(current_streak)
        current_sum=0
        current_streak=0
print("nb de jours consécutifs maximum pour qu'1kWc produise 1kWh: "+str(max_streak)+" j")
print("nombre d'occurences: "+ str(streaks.count(max_streak)))
print("nb de jours consécutifs moyen pour qu'1kWc produise 1kWh: "+str(sum(streaks)/len(streaks))+ " j")

max_streak = 0
current_streak = 0
current_sum=0
target=2000
streaks=[]
for value in daily_data['P']:
    if current_sum <= target:
        current_sum+=value
        current_streak += 1
        max_streak = max(max_streak, current_streak)
    else:
        streaks.append(current_streak)
        current_sum=0
        current_streak=0
print("nb de jours consécutifs maximum pour qu'1kWc produise 2kWh: "+str(max_streak)+" j")
print("nombre d'occurences: "+ str(streaks.count(max_streak)))
print("nb de jours consécutifs moyen pour qu'1kWc produise 2kWh: "+str(sum(streaks)/len(streaks))+ " j")

max_streak = 0
current_streak = 0
current_sum=0
target=3000
streaks=[]
for value in daily_data['P']:
    if current_sum <= target:
        current_sum+=value
        current_streak += 1
        max_streak = max(max_streak, current_streak)
    else:
        streaks.append(current_streak)
        current_sum=0
        current_streak=0
print("nb de jours consécutifs maximum pour qu'1kWc produise 3kWh: "+str(max_streak)+" j")
print("nombre d'occurences: "+ str(streaks.count(max_streak)))
print("nb de jours consécutifs moyen pour qu'1kWc produise 3kWh: "+str(sum(streaks)/len(streaks))+ " j")
```

Dans notre cas, on obtient:

nb de jours consécutifs maximum pour qu'1kWc produise 1kWh: 4 j

nombre d'occurences: 1

nb de jours consécutifs moyen pour qu'1kWc produise 1kWh: 1.1124367317425885 j

nb de jours consécutifs maximum pour qu'1kWc produise 2kWh: 5 j

nombre d'occurences: 3

nb de jours consécutifs moyen pour qu'1kWc produise 2kWh: 1.2801404603979711 j

nb de jours consécutifs maximum pour qu'1kWc produise 3kWh: 7 j

nombre d'occurences: 1

nb de jours consécutifs moyen pour qu'1kWc produise 3kWh: 1.4647827920708563 j

Que nous disent ces chiffres?

Que le nombre maximal de jours avec un faible ensoleillement varie de 4 à 7j selon le niveau d'ensoleillement espéré.

Mais aussi que le nombre d'occurrences est relativement faible: 1 à 3 occurrences seulement de ces épisodes avec très très peu de luminosité et de production sur 15 ans.

(si vous n'êtes pas entièrement satisfait des données modélisées au regard de votre expérience de la production photovoltaïque en fonction de la météo, et mon hypothèse est

que les panneaux produisent plus que ce que les données de la modélisation jrc nous donne, je vous encourage à

utiliser des données empiriques en utilisant mon autre tuto ici: Mesure de l'ensoleillement-luminosité avec un ordinateur monocarte (raspberry-orangepi))

Et enfin que le nombre de jours moyens pour qu'un kWc de module produise 1kWh à 3kWh est compris entre 1 et 1,5 j.

On a alors le choix:

-soit surdimensionner le stockage pour tenir 4j de besoins pour faire face à ces épisodes rares sans soleil : 20kWh de batteries pour 4kWh de besoins

-soit dimensionner le stockage pour tenir 1 à 2j et avoir une source électrique d'appoint à base de groupe électrogène fossile : 5 à 10kWh de batteries pour 4kWh de besoins

pour éviter un surdimensionnement coûteux pour de très rares occasions

-soit, si la criticité des besoins n'est pas trop importante, dimensionner le stockage pour tenir 1 à 2j et

se débrouiller sans électricité lorsqu'il y a des épisodes longs sans soleil : 5 à 10kWh de batteries pour 4kWh de besoins

-(ou jouer sur l'augmentation du dimensionnement de la production -le kWh produit étant très nettement moins cher que le kWh stocké en 2024 - dans l'étape suivante pour tout recharger dès le premier jour de soleil suivant les épisodes peu ensoleillés)

Nous sommes en mai 2024, et on sort juste du consensus informel industriel à 1€/Wh pour les batteries "performantes".

Si vous voulez dimensionner votre stockage "à l'échelle" après avoir réduit vos besoins électriques, le mieux

est encore de faire de la récupération de batterie de voitures.

Voir photos issues de l'atelier lowtech de l'astrolab pour de la récupération de batterie de voiture lithium ion.

En mai 2024, on s'en tire à 3k€ pour 14kWh.

Il est encore difficile de se passer de BMS à cause des normes sécuritaires,

mais on peut en trouver sur aliexpress (voir photo pour les références).

Draft

Récupérée de "https://lowtechlab.dokit.app/index.php?title=Dimensionner_une_installation_photovoltaique_autonome/en&oldid=152032"









Étape 3 - dimensionner la production

On suppose qu'on dimensionne les batteries à 10kWh pour 4kWh de besoin -soit 2j d'autonomie-

On peut dimensionner la production de plusieurs façon: 1. De façon à ce que les batteries soient entièrement rechargées sur un intervalle moyen en décembre/janvier (les mois les moins ensoleillés de l'année) 2. De façon à ce que les batteries soient entièrement rechargées en un jour d'ensoleillement "moyéné" en décembre/janvier

On notera que le dimensionnement calé sur les mois d'hiver permet de produire suffisamment pour subvenir aux besoin toute l'année, mais entraine une surproduction d'énergie en été qu'il serait utile de récupérer (par exemple en hydrolisant de l'eau pour produire de l'hydrogene puis en transformant l'hydrogene en méthane, cf scenario negawatt 2011) notamment si ce type d'installation venait à se généraliser.

On va réutiliser le bout de code précédent et l'adapter pour trouver les memes statistiques que précédemment mais uniquement en comptant les mois de décembre et janvier.

```

import pandas as pd
df = pd.read_csv('output.csv', skiprows=10, skipfooter=11, sep=',', engine='python')
df['time']=df['time'].astype(str)
df['time']=pd.to_datetime(df['time'],format="%Y%m%d:%H%M")
df=df.set_index('time')
daily_data = df.resample('D').sum()
dec_jan_data = daily_data[(daily_data.index.month == 12) <span>|</span> (daily_data.index.month == 1)]
print("Production moyenne journalière en janvier et décembre")
print(str(dec_jan_data['P'].mean()/1000)+" kWh")
max_streak = 0
current_streak = 0
current_sum=0
target=1000
streaks=[]
for value in dec_jan_data['P']:
    if current_sum <= target:
        current_sum+=value
        current_streak += 1
        max_streak = max(max_streak, current_streak)
    else:
        streaks.append(current_streak)
        current_sum=0
        current_streak=0
print("nb de jours consécutifs maximum pour qu'1kWc produise 1kWh: "+str(max_streak)+" j")
print("nombre d'occurrences: "+ str(streaks.count(max_streak)))
print("nb de jours consécutifs moyen pour qu'1kWc produise 1kWh: "+str(sum(streaks)/len(streaks))+ " j")

max_streak = 0
current_streak = 0
current_sum=0
target=2000
streaks=[]
for value in dec_jan_data['P']:
    if current_sum <= target:
        current_sum+=value
        current_streak += 1
        max_streak = max(max_streak, current_streak)
    else:
        streaks.append(current_streak)
        current_sum=0
        current_streak=0
print("nb de jours consécutifs maximum pour qu'1kWc produise 2kWh: "+str(max_streak)+" j")
print("nombre d'occurrences: "+ str(streaks.count(max_streak)))
print("nb de jours consécutifs moyen pour qu'1kWc produise 2kWh: "+str(sum(streaks)/len(streaks))+ " j")

max_streak = 0
current_streak = 0
current_sum=0
target=3000
streaks=[]
for value in dec_jan_data['P']:
    if current_sum <= target:
        current_sum+=value
        current_streak += 1
        max_streak = max(max_streak, current_streak)
    else:
        streaks.append(current_streak)
        current_sum=0
        current_streak=0
print("nb de jours consécutifs maximum pour qu'1kWc produise 3kWh: "+str(max_streak)+" j")
print("nombre d'occurrences: "+ str(streaks.count(max_streak)))
print("nb de jours consécutifs moyen pour qu'1kWc produise 3kWh: "+str(sum(streaks)/len(streaks))+ " j")

```

Dans notre cas, on obtient:

Production moyenne journalière en janvier et décembre 1.9924362701612903 kWh

nb de jours consécutifs maximum pour qu'1kWc produise 1kWh: 4 j

nombre d'occurrences: 1

nb de jours consécutifs moyen pour qu'1kWc produise 1kWh: 1.3539192399049882 j

nb de jours consécutifs maximum pour qu'1kWc produise 2kWh: 5 j

nombre d'occurrences: 3

nb de jours consécutifs moyen pour qu'1kWc produise 2kWh: 1.8022598870056497 j

nb de jours consécutifs maximum pour qu'1kWc produise 3kWh: 7 j

nombre d'occurrences: 1

nb de jours consécutifs moyen pour qu'1kWc produise 3kWh: 2.263157894736842 j

1. En dimensionnant sur l'intervalle moyen d'1kWh: 1kWc produit en moyenne 1kWh en 1.35j, soit 0,74kWh ramené sur un intervalle d'un jour, soit une production de 10kWh/0,74=13,5kWc En dimensionnant sur l'intervalle moyen de 2kWh: 1kWc produit en moyenne 2kWh en 1,8j, soit 1,11kWh ramené sur un intervalle d'un jour, soit une production de 10kWh/1,11=9kWc En dimensionnant sur l'intervalle moyen de 3kWh: 1kWc produit en moyenne 3kWh en 2,26j, soit 1,33kWh ramené sur un intervalle d'un jour, soit une production de 10kWh/1,33=7,5kWc

2. En dimensionnant sur un jour moyené, 1kWc produit en moyenne 1,99kWh par jour, soit un besoin de 10kWh/1,99=5kWc

On remarque que les données de références de l'ines pour la production d'1kWc en décembre correspondent à 1kWc==1kWh environ.

(<https://autocalsol.ines-solaire.org/etude/localisation/> et http://ines.solaire.free.fr/pvisole_1.php pour site isolés)

On peut ici parler d'hypothèses plus ou moins "conservatrices", c'est à dire d'hypothèses plus ou moins "prudentes", afin d'avoir des batteries toujours rechargées en hiver. On remarquera que quelles que soit l'hypothèse retenue pour satisfaire les besoins de façon continue, y compris en hiver, on se retrouvera avec des surproductions importantes en été. Il est alors judicieux d'utiliser des systèmes de récupération de la surproduction en l'absence de réseau dans lequel on peut injecter le surplus (par exemple en hydrolisant de l'eau pour produire de l'hydrogene puis en transformant l'hydrogene en méthane, cf scénario negawatt 2011).

Étape 4 - dimensionner le régulateur de charge

rappel de base: branchement en série (+ sur - et + sur -) on ajoute le voltage et on garde le meme amperage, branchement en parallele (+ sur +, - sur -) on ajoute l'amperage et on garde le meme voltage

idem pour les batteries: à mettre en parallele pour garder la meme tension (voltage)

La première problématique du photovoltaïque lowtech autonome est le dimensionnement de l'installation.

Vous pouvez utiliser la feuille libreoffice en piece jointe en haut de ce tutoriel pour du dimensionnement "bricolé".

Les étapes précédentes ont permis de calculer à partir du modèle jrc la taille de la batterie et la production moyenne en hiver d'un kWc.

Vous pouvez lire les caractéristiques des panneaux qu'on vous a donné ou trouvés sur leboncoin à pas cher:

-puissance crete: elles s'additionnent pour obtenir la puissance nécessaire trouvée lors de la phase de dimensionnement.

-tension : 12V,24V ou 48V, voir regles série/parallele pour leur additions

-intensité: variable selon les modèles mais souvent inferieure à 10A, voir regles série/parallele pour leur additions

Pour recharger des batteries, en principe, si vous connectez votre panneaux en direct sur une batterie, il suffit que la tension à la sortie de vos panneaux soit la même que celles de vos batter ie, et ca devrait charger.

Il y a un composant important à retenir pour charger correctement vos batteries:

le regulateur ou controleur de charge (nb: generalement, les onduleurs dits "hybrides", cad avec une sortie batterie et une sortie reseau, integrent un regulateur de charge, le plus souvent mppt, qui conditionne les specifications des batteries qu'on peut lui brancher)

il en existe de trois sortes: les tor (tout ou rien) les mppt (Maximum power point tracking) et les pwm (Pulse Width Modulation)

Ils sont composés d'un adaptateur DC/DC (courant continu vers courant continu) et d'un coupe circuit. Le mppt comprend également un adaptateur d'impédance (il a une résistance pour adapter l'amperage injecté dans la batterie). Les mppt accepte des puissances nominales plus élevées, cad des tensions et intensité plus élevées.

Le régulateur ou controleur de charge permet principalement de couper le circuit quand la batterie est rechargée en surveillant la tension et l'intensité de charge. Il coupe le circuit si leurs valeurs dépassent les intervalles de référence (pour cela le regulateur arrete la charge temporairement et mesure la tension aux bornes des batteries).

Le mppt a un "algorithme" électronique intégré qui va chercher le point de puissance optimal grace a son adaptateur de resistance.

Si vous connectez plusieurs panneaux et plusieurs batteries, il est recommandé d'avoir un regulateur pour couper la charge correctement lorsque la batterie est chargée.

Les tensions de charge de référence sont 12V,24V et 48V.

Cependant, les prix des modèles augmentent avec la puissance nominales (qui va dépendre de l'amperage) qu'ils acceptent.

Pour limiter l'intensité du courant de la production photovoltaïque, il est plus judicieux d'utiliser des panneaux de plus forte puissance qui sont généralement à des tensions plus élevées

(rappel $P=U \cdot I$,

rappel $E=P \cdot t$ se conserve dans un systeme fermé).

note: si le systeme de stockage par batterie ou l'appareil connecté à vos panneaux n'absorbe pas toute la puissance produite, et si le régulateur de charge ne coupe pas le circuit, le reste sera dégagé en chaleur.

L'amperage va aussi dependre de la capacité de stockage de vos batteries, dimensionnées pour couvrir vos besoins pendant une periode définie au dimensionnement.

Le courant de charge est calculé en divisant par 4 ou 5 la capacité nominale de la batterie exprimée en Ah qui devrait alors se recharger en 4 ou 5h. Cependant une batterie se rechargera aussi avec un courant de charge de la capacité nominale de la batterie divisée par 20 mais plus lentement (en 20h).

Pour dimensionner correctement le regulateur de charge on peut :

-soit monter tous les panneaux en série (ce qui fait fortement augmenter la tension et il faudra peut etre un transfo ensuite pour trouver un regulateur qui accepte cette tension)

-soit monter tous les panneaux en parallele (ce qui fait fortement augmenter l'intensité)

-soit monter les panneaux en serie/parelle pour ajuster tension et intensité aux caractéristiques de votre régulateur.

Il y a enfin un dernier point sur lequel être attentif: le déclenchement de la recharge de la batterie par le regulateur/controleur de charge (qui déclenche quand la tension de la batterie diminue en dessous d'un certain seuil).

En effet, si la puissance soustrée à la batterie est trop faible, il est possible que le temps nécessaire à la décharger avec votre consommation journalière pour déclencher la recharge dans le régulateur dépasse le temps d'ensoleillement journalier. La batterie ne se recharge alors pas du tout pendant la journée.

Dans ce cas, votre batterie ne se rechargera qu'un jour sur deux ou sur trois (selon le seuil de déclenchement de la recharge du regulateur).

C'est un paramètre à prendre en compte dans le dimensionnement (non inclus dans la feuille de calcul).

Le régulateur a 3 phases:

bulk: le regulateur laisse passer le courant

floating: le regulateur alterne interrupteur fermée et ouvert à une fréquence donnée pour maintenir la batterie chargée

En outre il faut prendre des précautions car la charge des batteries présente certains risques.

absorption (pour les mppt): la tension de charge augmente un peu pour créer une difference de potentiel suffisante pour continuer à charger la batterie presque pleine.

En théorie le courant de charge diminue lorsque la batterie est presque rechargée (courant de queue etc.)

La charge de batteries en parallele ou en série sur des batteries usagées qui n'ont pas les mêmes tensions ou intensité présente en théorie des risques. En effet vous lirez un peu partout que la résistance des fils pour relier ces batteries

crée des différences de potentiels entre les batteries qui produisent des décharges d'une batterie envers une autre etc.

créant des risques d'explosion, de degazage pour les batteries plomb etc.

Il faut bien se rappeler que les batteries sont des assemblages de composants unitaires de faible tension mis en séries et en parallele pour obtenir un générateur de l'intensité et la tension voulue et qu'à priori faire de même avec des batterie entiere ne présente pas vraiment de risques..

On parle souvent de "battery management system" (bms) "intégré" pour les batteries lithium ion.

En réalité le régulateur de charge est déjà un "bms". En théorie, le bms intégré s'assure que les tensions et les intensités de chaque unité composant la batterie est la même et la rééquilibre au besoin.

On peut bien sûr s'interroger si tout ceci n'est pas une façon de rendre le stockage de l'énergie plus cher avec des composants BMS artificiellement chers et si ce n'est pas une façon d'éviter de réutiliser des unités de batteries usagées.

Il est par exemple étonnant qu'il n'existe pas de BMS pour rééquilibrer automatiquement des batteries plomb acide, ce qui rendrait utilisable toutes les batteries mises au rebut de l'industrie automobile pour stocker l'énergie photovoltaïque sans risque!

Dans tous les cas, si vous réutilisez des batteries au plomb, utilisez un régulateur pour éviter de continuer à charger vos batteries rechargées (risques de production d'hydrogene) -ou si vous n'en utilisez pas dimensionnez avec beaucoup de soin-, évitez les décharges profondes, et maintenez les batteries à une temperature constante autant que possible.

Étape 5 - Optimiser le dimensionnement

On va maintenant utiliser un bout de code python pour itérer (un peu sur le principe d'essais et erreurs) en augmentant progressivement la puissance des modules, la capacité de la batterie, et le nombre de jours de blackout admissibles et en testant au jour le jour le résultat stockage - consommation électrique + production sur les données jrc précédemment téléchargées.

Explications : Le code est commenté pour expliquer ce qui est fait et les noms des variables sont relativement explicites.

Pour la boucle d'itération: On démarre à puissance0=(consommation journaliere)/2 (hypothese hiver 1 kWc produit 2kWh en moyenne) On démarre à batterie0=consommation journaliere/0,8 au lithium et consommation journaliere/0,5 au plomb

On itère 14 fois les puissances par pas de 1/3*puissance0 kWc On itère 40 fois les capacités de batterie par pas de 1/2*batterie0 kWh On itère 20 fois les jours admissibles de blackout par pas de 1j

On démarre à batterie pleine et on teste chaque jour si le stock d'électricité dans la batterie moins la consommation plus la production photovoltaïque aboutit à un blackout ou permet de subvenir à la consommation journaliere. S'il y a black out, on arrete la boucle au bout du nombre de jours consécutifs admissibles de black out.

Pré-requis logiciels:

Pour utiliser python sous un autre os que linux, débrouillez vous avec vos daubes propriétaires intrusives.

Sous linux, python est généralement installé et pour utiliser le code partagé dans ce tuto, il suffira de copier coller le code dans un fichier processing.py puis entré

```
python processing.py
```

Cependant, il faut installer la librairie pandas qui est très largement utilisé dans le monde de la finance et dans le monde scientifique, notamment pour sa gestion efficace des séries temporelles et sa capacité de vectorisation des données.

Pour cela voici les commandes à entrer dans linux debian avant de lancer le script processing.py pour être tranquille:

```

sudo apt install python3 python3-venv python3-pip python-is-python3

cd ~ && python -m venv venv

source venv/bin/activate

pip install pandas

```

Penser ensuite à activer l'environnement virtuel dans lequel est installé pandas chaque fois que vous utilisez le script (apres un reboot ou si vous fermez et reouvrez votre terminal) en lançant la commande:

```
cd ~ && source venv/bin/activate
```

Nous sommes en 2024 et si vous êtes cibles d'entraves anti éco-terroristes de psychopathes comme moi, et en bon scientifique qui se respecte vous inspectez vos instruments de mesures avant de les utiliser, vous pouvez inspectez le code source de pandas qui est évidemment logiciel libre ici : <https://github.com/pandas-dev/pandas>, ou vous pouvez faire l'hypothèse discutable qu'on peut avoir confiance dans un logiciel aussi massivement utilisé dans le monde de la finance et de la science.

Python reposant sur des librairies C pour un certains nombres d'opérations de base, le hack, y compris le hack scientifique, n'est jamais impossible mais on laissera de coté ces considérations pro-lowtech qui n'entrent pas dans le champ de réflexion de ce tutoriel.

bout de code: (update 8.4.24: ajout commentaires, input angle&orientation&% decharge batterie)

```

import pandas as pd
import sys
import os
import math
import time

# Demander à l'utilisateur d'entrer la latitude et la longitude
x_input = input("Entrez la latitude de votre lieux: ")
y_input = input("Entrez la longitude de votre lieux: ")

# Remplacer les virgules par des points
x_input = float(x_input.replace(',', '.'))
y_input = float(y_input.replace(',', '.'))

#Entrer l'orientation de votre lieu
aspect_input=input("Entrez l'orientation de votre lieux (est,ouest,nord,sud)")
dictaspect={'est':90,'ouest':90,'nord':180,'sud':0}
if aspect_input in dictaspect:
    aspect=dictaspect[aspect_input]
else:
    aspect=0

#Entrer l'angle des modules
angle_input=input("Entrez l'angle des modules en °")
try:
    angle=float(angle_input)
except Exception as err:
    print(f"error, angle utilisé: 45°")
    angle=45

# Telecharger les données
try:
    os.system(f'wget --max-redirect=10 -O output.csv "https://re.jrc.ec.europa.eu/api/v5_2/seriescalc?lat={x_input}&lon={y_input}&loss=14&angle={angle}&aspect={aspect}&startyear=2005&endyear=2020&pvcaculation=1&peakpower=1&pvtchchoice=crystSi&browser=0&outputformat=csv"')
except:
    print("les données n'ont pas pu être téléchargées, exiting")
    sys.exit()

df = pd.read_csv('output.csv', skiprows=10, skipfooter=11, sep=',', engine='python')
df['time']=df['time'].astype(str)
df['time']=pd.to_datetime(df['time'],format="%Y%m%d-%H%M")
df=df.set_index('time')
daily_data = df.resample('D').sum()

max_streak = 0
current_streak = 0
current_sum=0
target=1000
streaks=[]
for value in daily_data['P']:
    if current_sum <= target:
        current_sum+=value
        current_streak += 1
        max_streak = max(max_streak, current_streak)
    else:
        streaks.append(current_streak)
        current_sum=0
        current_streak=0
print("\n")
print("nb de jours consécutifs maximum pour qu'1kWc produise 1kWh: "+str(max_streak)+" j")
print("nombre d'occurrences: "+ str(streaks.count(max_streak)))
print("nb de jours consécutifs moyen pour qu'1kWc produise 1kWh: "+str(sum(streaks)/len(streaks))+ " j")

max_streak = 0
current_streak = 0
current_sum=0
target=2000
streaks=[]
for value in daily_data['P']:
    if current_sum <= target:
        current_sum+=value
        current_streak += 1
        max_streak = max(max_streak, current_streak)
    else:
        streaks.append(current_streak)
        current_sum=0
        current_streak=0

```

```

        current_sum=0
        current_streak=0
    print("\n")
    print("nb de jours consécutifs maximum pour qu'1kWc produise 2kWh: "+str(max_streak)+" ")
    print("nombre d'occurrences: " + str(streaks.count(max_streak)))
    print("nb de jours consécutifs moyen pour qu'1kWc produise 2kWh: "+str(sum(streaks)/len(streaks))+ " ")

    max_streak = 0
    current_streak = 0
    current_sum=0
    target=3000
    streaks=[]
    for value in daily_data['P']:
        if current_sum <= target:
            current_sum+=value
            current_streak += 1
            max_streak = max(max_streak, current_streak)
        else:
            streaks.append(current_streak)
            current_sum=0
            current_streak=0
    print("\n")
    print("nb de jours consécutifs maximum pour qu'1kWc produise 3kWh: "+str(max_streak)+" ")
    print("nombre d'occurrences: " + str(streaks.count(max_streak)))
    print("nb de jours consécutifs moyen pour qu'1kWc produise 3kWh: "+str(sum(streaks)/len(streaks))+ " ")

    daily_data['P']=daily_data['P']/1000
    #Production d'un kWh à partir des data jrc:
    resultday=daily_data['P'].resample('D').sum()
    resultday.index=pd.to_datetime(resultday.index,format="%Y%m%d")
    print("\nProduction moyenne par jour (kWh):\n", resultday.mean())
    print("Production minimum par jour (kWh):\n", resultday.min())
    print("Production maximum par jour (kWh):\n", resultday.max())

    # Calculer les sommes de production par semaine
    resultweek=daily_data['P'].resample('W').sum()

    # Calculer les sommes de production par mois
    resultmonth=daily_data['P'].resample('M').sum()

    # Calculer les sommes de production par trimestre
    resulttrim=resultday.resample('Q').sum()
    resulttrim=resulttrim.rename_axis('trimestre')
    print("\nProduction par trimestre (kWh):\n",resulttrim.to_string())

    # Calculer les sommes de production par an
    resultyear=daily_data['P'].resample('Y').sum()
    print("\nProduction annuelle moyennes d'1kWc (kWh):\n",resultyear.mean())

    # Calculer le nombre de jours consécutifs maximum sans production
    max_streak = 0
    current_streak = 0
    for value in resultday:
        if value == 0:
            current_streak += 1
            max_streak = max(max_streak, current_streak)
        else:
            current_streak = 0 # Reset the streak if the value is not zero
    print(f"\nNombre de jours consecutifs maximum à production 0: {max_streak}")

    # Moyenne par trimestre pour chaque trimestre
    moyenne_trimestrielle_par_trimestre = resulttrim.groupby(resulttrim.index.quarter).mean()

    # Minimum par trimestre pour chaque trimestre
    min_trimestrielle_par_trimestre = resulttrim.groupby(resulttrim.index.quarter).min()

    # Maximum par trimestre pour chaque trimestre
    max_trimestrielle_par_trimestre = resulttrim.groupby(resulttrim.index.quarter).max()

    # Imprimer les résultats
    print("\nMoyenne par trimestre pour chaque trimestre (kWh):\n", moyenne_trimestrielle_par_trimestre)
    print("\nMinimum par trimestre pour chaque trimestre (kWh):\n", min_trimestrielle_par_trimestre)
    print("\nMaximum par trimestre pour chaque trimestre (kWh):\n", max_trimestrielle_par_trimestre)

    # Demander à l'utilisateur d'entrer sa consommation d'électricité journalière
    inputelecconsoday = input("Entrez la consommation d'électricité journalière (kWh): ")

    # Remplacer les virgules par des points
    inputelecconsoday = float(inputelecconsoday.replace(',','.'))

    # Demander à l'utilisateur batterie plomb ou lithium
    typebatterie = input("Voulez vous utiliser des batteries au plomb? Taper oui ou o pour répondre oui (Lithium par défaut si rien n'est indiqué)")

    # Dimensionnement batterie initiale pour avoir 24h d'autonomie:
    if typebatterie=="oui" or typebatterie=="o" or typebatterie=="y" or typebatterie=="yes":
        batterie0=int(math.ceil(inputelecconsoday/0.5))
        typebatterie="plomb"
    else:
        batterie0=int(math.ceil(inputelecconsoday/0.8))
        typebatterie="lithium"
    print(f"Seuil batterie initiale avec hypothèse entrée et données fournies par l'utilisateur (batterie {typebatterie})")
    hypothese:(conso elec journalière / 0.8 au lithium et conso elec journalière / 0.5 au plomb)
    {batterie0} kWh")

    # Pourcentage de décharge maximum
    d_input=input("Entrer le pourcentage de décharge maximum autorisé (20% si vous voulez que votre batterie ne se décharge pas à moins de 20%. Appuyer sur entrée si vous voulez utiliser les valeurs par défaut de 50% pour le plomb et de 0% sinon (hypothese solid state battery)")
    try:
        d=float(d_input)/100
    except Exception as err:

```

```

if typebatterie=='plomb':
    d=0.5
else:
    d=0

# Demander à l'utilisateur combien de jours sans elec il ou elle peut supporter
inputjnoelec = input("Combien de jours maximum consécutifs sans électricité (ou avec un groupe electrogene) pouvez vous supporter (0 par défaut si reponse vide)")

try:
    jnoelec=int(inputjnoelec)
except Exception as err:
    jnoelec=0
print(f"Nombre de jours maximum consécutifs sans électricité (ou avec un groupe electrogene) pris en hypothèse {jnoelec} j")

# Dimensionnement puissance crete des modules (kWc) initiale pour produire suffisamment en hiver (hypothese 2kWh par kWc par jour):
puissance0=int(math.ceil(inputelecconsoday/2))

print(f"Seuil puissance initiale avec hypothèse entrée et données fournies par l'utilisateur
hypothèse: conso elec journaliere/2 (2kWh produit par kWc en hiver)
{puissance0} kWc")

print(f"Seuil volume de batterie initiale avec hypothèse entrée et données fournies par l'utilisateur
hypothèse: conso elec journaliere/0.8 au lithium et 0.5 au plomb
{batterie0} kWh")

batterie0_input = input("\nSi vous souhaitez corriger la valeur initiale de batterie (kWh) pour les itérations, entrer votre valeur, sinon appuyer sur entree")
try:
    _=float(batterie0_input)
    batterie0=_
except Exception as err:
    print(f"Erreur de type ou valeur utilisateur vide, poursuite avec utilisation de batterie0={batterie0}kWh")

puissance0_input = input("\nSi vous souhaitez corriger la valeur initiale de puissance (kWc) pour les itérations, entrer votre valeur, sinon appuyer sur entree")
try:
    _=float(puissance0_input)
    puissance0=_
except Exception as err:
    print(f"Erreur de type ou valeur utilisateur vide, poursuite avec utilisation de puissance0={puissance0}kWc")

# Itérations algorithmiques stockage&consommation

#fonction check surface volume
def iter(data,consoelecday,v0,p0,joursnoelec):
    #elec est l'énergie dans la batterie : electricite initiale=v0 (volume de batterie à 10)
    elec=v0
    #current_streak est le nombre de jours de blackout consécutif
    current_streak = 0
    #listjnoelec est la liste dans laquelle on enregistre les jours de blackout
    listjnoelec=[]
    #boucle sur la date entrée en argument de la fonction
    for i in range(len(data)):
        #recupday est l'électricité récupérée (produite) au jour i (la puissance d'1kWc*p0)
        #p0 est la puissance crete entree en argument
        recupday=data.iloc[i]*p0
        #consoday est la consommation électrique du jour i entree en argument de la fonction
        consoday=consoelecday
        #energie mise à jour avec consoday et recupday
        elec=elec+recupday-consoday
        #si l'énergie mise à jour est supérieure au volume de batterie
        if elec>v0:
            #print("batterie pleine")
            #le nombre de jours de balckout consécutif est remis à zero
            current_streak=0
            #l'énergie est égale au volume de la batterie
            elec=v0 #hypothese gestion du trop plein ok
            continue
        #si l'énergie mise à jour est inférieure au pourcentage max de decharge et le nombre de jours consécutifs de blackout est inférieure au nombre de jours toléré
        elif elec<d*v0 and current_streak<joursnoelec:
            #si le nombre de jours consécutifs de blackout est égal à zero
            if current_streak==0:
                #ajouter la date du jour et un compteur à la liste listjnoelec
                listjnoelec.append([data.index[i].strftime("%d-%m-%Y"),1])
            #sinon
            else:
                #incrémenter le compteur de la dernier entree de listjnoelec
                listjnoelec[-1][-1]+=1
            #remise à zero de la valeur négative de elec
            elec=0
            #incrémentation du nombre de jours consécutifs de blackout
            current_streak+=1
            continue
        #si l'énergie est inférieure au pourcentage max de decharge et le nombre de jours consécutifs de blackout supérieur au nombre de jours tolérés
        elif elec<d*v0 and current_streak>=joursnoelec:
            #la fonction renvoie une valeur nulle
            return (0,0)
        #sinon
        else:
            #remise à zero du nombre de jours consécutifs de blackout
            current_streak=0
    #variable nombre d'épisode de blackout = longueur de la la liste listjnoelec
    nb_episode_no_elec=len(listjnoelec)
    #si listjnoelec n'est pas vide
    if len(listjnoelec)!=0:
        #liste des dures des épisodes sans elec
        list_duree_episode_no_elec=[length[1] for length in listjnoelec]
        #moyenne de cette liste
        duree_moy_episode_no_elec=sum(list_duree_episode_no_elec)/len(list_duree_episode_no_elec)
    #sinon
    else:

```

```

#duree moyenne egale à zero
duree_moy_episode_no_elec=0
#print("les puissances et batteries permettent de subvenir à la consommation d'électricité sur le dataset")
#renvoyer les variables necessaires au traitement des résultats
return (v0,p0,journoelec,nb_episode_no_elec,duree_moy_episode_no_elec,listjnoelec)
#hypothèse récupérateur plein à t0
elec=batterie0
resultpuissancevolume=(batterie0,puissance0,jnoelec,0)

#boucle iteration
listpuissance0=[puissance0*(1+i*0.33) for i in range(0,999)]
listbatterie0=[batterie0*(1+i*0.5) for i in range(0,999)]
listresult_blackout=[]
listresult_blackout=[]

# increment avec hypothèse jnoelec=0 (k in range 0,1)
for i in range(0,14): #boucle iteration surface modules
    for j in range (0,i+40):#boucle iteration volume batterie
        for k in range(0,1):
            resultpuissancevolume=iter(resultday,inputelecconsoday,listbatterie0[j],listpuissance0[i],k)
            #si les resultats ne sont pas nuls
            if resultpuissancevolume!=(0,0):
                #ajouter les resultats à une liste de resultats
                listresult_blackout.append(resultpuissancevolume)
                #arreter la boucle k
                break
            else:#sinon continuer
                continue
        #arreter la boucle j si les resultats ne sont pas nul
        if resultpuissancevolume!=(0,0):
            break
        else:
            continue

#idem avec increment hypothèse jnoelec entre 0 et 13 (i in range 0,14)
for i in range(0,14):
    for j in range (0,i+40):
        for k in range(0,21):
            resultpuissancevolume=iter(resultday,inputelecconsoday,listbatterie0[j],listpuissance0[i],k)
            if resultpuissancevolume!=(0,0):
                listresult_blackout.append(resultpuissancevolume)
                break
            else:
                continue
        if resultpuissancevolume!=(0,0):
            break
        else:
            continue

# Affichage des résultats des boucles d'itérations
for k in listresult_blackout:
    print(f"avec les données fournies par l'utilisateur, et
    une batterie de {k[0]}kWh
    et une puissance crete de {k[1]}kWc,
    et une hypothèse de {k[2]}j sans elec possible
    et {k[3]} épisodes sans elec (ou avec un groupe electrogene)
    entre 2005 et 2020:
    Liste des épisodes sans elec (nb de jours):
    {k[5]}
    pour une duree moyenne d'épisode sans elec de {k[4]}j
    on satisfait aux besoins utilisateurs ({inputelecconsoday}kWh/jour)entrées en hypothèse")

for k in listresult_blackout:
    print(f"avec les données fournies par l'utilisateur, et
    une batterie de {k[0]}kWh
    et une puissance crete de {k[1]}kWc,
    et une hypothèse de {k[2]}j sans elec possible
    et {k[3]} épisodes sans elec (ou avec un groupe electrogene):
    entre 2005 et 2020:
    Liste des épisodes sans elec (nb de jours):
    {k[5]}
    pour une duree moyenne d'épisode sans elec de {k[4]}j
    on satisfait aux besoins utilisateurs ({inputelecconsoday}kWh/jour)entrées en hypothèse")

print("\n\n\n")
print("affichage des résultats optimisés")
# Optimisation résultats:

# premier résultat en incrémentant la puissance crete dont la capacité de batterie est minimale
batterie_blackout=[k[0] for k in listresult_blackout]
minbatterie_blackout=min(batterie_blackout)
for k in listresult_blackout:
    if k[0]==minbatterie_blackout:
        print(f"avec les données fournies par l'utilisateur, et
        une batterie de {k[0]}kWh
        et une puissance crete de {k[1]}kWc,
        et une hypothèse de {k[2]}j sans elec possible
        et {k[3]} épisodes sans elec (ou avec un groupe electrogene):
        entre 2005 et 2020:
        Liste des épisodes sans elec (nb de jours):
        {k[5]}
        pour une duree moyenne d'épisode sans elec de {k[4]}j
        on satisfait aux besoins utilisateurs ({inputelecconsoday}kWh/jour)entrées en hypothèse")
        break

# premier résultat en incrémentant la puissance crete fournissant un nb de jours de blackout inferieur ou égal
# à celui entrée par l'utilisateur
for k in listresult_blackout:
    if k[2]<=jnoelec:
        print(f"avec les données fournies par l'utilisateur, et

```

une batterie de {k[0]}kWh
et une puissance crete de {k[1]}kWc,
et une hypothèse de {k[2]} sans elec possible
et {k[3]} episodes sans elec (ou avec un groupe electrogene):
entre 2005 et 2020:
Liste des episodes sans elec (nb de jours):
{k[5]}
pour une duree moyenne d'épisode sans elec de {k[4]}
on satisfait aux besoins utilisateurs ((inputelecconsoday)kWh/jour)entrées en hypothèse""
break

Étape 6 - Resultats optimisés

Pour notre exemple, on aboutit aux résultats suivant (avec 1j de blackout autorisé): Les "résultats optimisés sont tout en bas":

Entrez la latitude de votre lieux: 44.2
Entrez la longitude de votre lieux: 0.6
Entrer l'orientation de votre lieux (est,ouest,nord,sud)sud
Entrez l'angle des modules en °45
--2024-04-08 20:04:29-- https://re.jrc.ec.europa.eu/api/v5_2/seriescalc?
lat=44.2&lon=0.6&loss=14&angle=45.0&aspect=0&startyear=2005&endyear=2020&pvcaculation=1&peakpower=1&pvttechchoice=crystSi&browser=0&outputformat=csv
Résolution de re.jrc.ec.europa.eu (re.jrc.ec.europa.eu)... 64:ff9b::8bbf:dd12, 139.191.221.18
Connexion à re.jrc.ec.europa.eu (re.jrc.ec.europa.eu)[64:ff9b::8bbf:dd12]:443... connecté.
requête HTTP transmise, en attente de la réponse... 200 OK
Taille : 6291134 (6,0M) [text/html]
Enregistre : «output.csv»

output.csv 100%[=====>] 6,00M 1,03MB/s ds 7,2s

2024-04-08 20:04:43 (852 KB/s) - «output.csv» enregistré [6291134/6291134]

nb de jours consécutifs maximum pour qu'1kWc produise 1kWh: 4 j
nombre d'occurences: 1
nb de jours consécutifs moyen pour qu'1kWc produise 1kWh: 1.1124367317425885 j

nb de jours consécutifs maximum pour qu'1kWc produise 2kWh: 5 j
nombre d'occurences: 3
nb de jours consécutifs moyen pour qu'1kWc produise 2kWh: 1.2801404603979711 j

nb de jours consécutifs maximum pour qu'1kWc produise 3kWh: 7 j
nombre d'occurences: 1
nb de jours consécutifs moyen pour qu'1kWc produise 3kWh: 1.4647827920708563 j

Production moyenne par jour (kWh):
3.4641937012320336
Production minimum par jour (kWh):
0.0
Production maximum par jour (kWh):
6.65871

Production par trimestre (kWh):
trimestre
2005-03-31 270.56038
2005-06-30 375.15833
2005-09-30 399.84068
2005-12-31 227.76775
2006-03-31 247.03158
2006-06-30 406.70165
2006-09-30 389.10805
2006-12-31 233.93622
2007-03-31 229.46528
2007-06-30 348.05120
2007-09-30 390.13982
2007-12-31 250.21269
2008-03-31 250.87365
2008-06-30 347.15819
2008-09-30 385.60970
2008-12-31 202.44585
2009-03-31 286.11329
2009-06-30 350.51328
2009-09-30 397.86835
2009-12-31 237.76417
2010-03-31 270.32507
2010-06-30 366.21454
2010-09-30 394.93503
2010-12-31 236.08537
2011-03-31 271.73968
2011-06-30 400.68298
2011-09-30 374.99079
2011-12-31 255.64396
2012-03-31 294.48135
2012-06-30 344.82346
2012-09-30 394.74104
2012-12-31 217.83773
2013-03-31 259.24581
2013-06-30 323.20427
2013-09-30 402.14528
2013-12-31 228.40704
2014-03-31 268.96648
2014-06-30 385.06742
2014-09-30 388.64375
2014-12-31 252.08195
2015-03-31 242.45034
2015-06-30 402.47347
2015-09-30 400.23336
2015-12-31 273.02084
2016-03-31 248.06779
2016-06-30 354.35530
2016-09-30 344.38888

2016-09-30 411.73628
2016-12-31 224.28598
2017-03-31 267.90392
2017-06-30 410.47864
2017-09-30 363.44330
2017-12-31 251.00240
2018-03-31 231.96132
2018-06-30 362.16785
2018-09-30 426.21631
2018-12-31 219.18595
2019-03-31 313.23495
2019-06-30 368.76834
2019-09-30 407.61175
2019-12-31 201.18704
2020-03-31 283.16855
2020-06-30 389.45566
2020-09-30 395.65480
2020-12-31 240.10074
Freq: Q-DEC

Production annuelle moyennes d'1kWc (kWh):
1265.2967493749998

Nombre de jours consecutifs maximum à production 0: 2

Moyenne par trimestre pour chaque trimestre (kWh):

trimestre

1 264.724340
2 370.954661
3 395.182393
4 234.435355

Name: P, dtype: float64

Minimum par trimestre pour chaque trimestre (kWh):

trimestre

1 229.46528
2 323.20427
3 363.44330
4 201.18704

Name: P, dtype: float64

Maximum par trimestre pour chaque trimestre (kWh):

trimestre

1 313.23495
2 410.47864
3 426.21631
4 273.02084

Name: P, dtype: float64

Entrez la consommation d'électricité journaliere (kWh): 4

Voulez vous utiliser des batteries au plomb? Taper oui ou o pour repondre oui (Lithium par défaut si rien n'est indiqué)

Seuil batterie initiale avec hypothèse entrée et données fournies par l'utilisateur (batterie lithium))

hypothese:(conso elec journaliere / 0.8 au lithium et conso elec journaliere / 0.5 au plomb)

5 kWh

Entrer le pourcentage de décharge maximum autorisé (20% si vous voulez que votre batterie ne se décharge pas à moins de 20%. Appuyer sur entree si vous voulez

utiliser les valeurs par défaut de 50% pour le plomb et de 0% sinon (hypothese solid state battery)

Combien de jours maximum consécutifs sans electricité (ou avec un groupe electrogene) pouvez vous supporter (0 par défaut si reponse vide)1

Nombre de jours maximum consécutifs sans electricité (ou avec un groupe electrogene) pris en hypothèse

1 j

Seuil puissance initiale avec hypothèse entrée et données fournies par l'utilisateur

hypothèse: conso elec journaliere/2 (2kWh produit par kWc en hiver)

2 kWc

Seuil volume de batterie initiale avec hypothèse entrée et données fournies par l'utilisateur

hypothèse: conso elec journaliere/0.8 au lithium et 0.5 au plomb

5 kWh

Si vous souhaitez corriger la valeur initiale de batterie (kWh) pour les itérations, entrer votre valeur, sinon appuyer sur entree

erreur de type ou valeur utilisateur vide, poursuite avec utilisation de batterie0=5kWh

Si vous souhaitez corriger la valeur initiale de puissance (kWc) pour les itérations, entrer votre valeur, sinon appuyer sur entree

erreur de type ou valeur utilisateur vide, poursuite avec utilisation de puissance0=2kWc

avec les données fournies par l'utilisateur, et

une batterie de 75.0kWh

et une puissance crete de 2.0kWc,

et une hypothèse de 0j sans elec possible

et 0 episodes sans elec (ou avec un groupe electrogene)

entre 2005 et 2020:

Liste des episodes sans elec (nb de jours):

[]

pour une duree moyenne d'épisode sans elec de 0j

on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse

avec les données fournies par l'utilisateur, et

une batterie de 22.5kWh

et une puissance crete de 2.66kWc,

et une hypothèse de 0j sans elec possible

et 0 episodes sans elec (ou avec un groupe electrogene)

entre 2005 et 2020:

Liste des episodes sans elec (nb de jours):

[]

pour une duree moyenne d'épisode sans elec de 0j

on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse

avec les données fournies par l'utilisateur, et

une batterie de 17.5kWh

et une puissance crete de 3.320000000000003kWc,

et une hypothèse de 0j sans elec possible

et 0 episodes sans elec (ou avec un groupe electrogene)

entre 2005 et 2020:

[illegible]

entre 2005 et 2020: Liste des épisodes sans elec (nb de jours): [] pour une duree moyenne d'épisode sans elec de 0j on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse avec les données fournies par l'utilisateur, et une batterie de 10.0kWh et une puissance crete de 10.58kWc, et une hypothèse de 0j sans elec possible et 0 épisodes sans elec (ou avec un groupe electrogene) entre 2005 et 2020: Liste des épisodes sans elec (nb de jours): [] pour une duree moyenne d'épisode sans elec de 0j on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse avec les données fournies par l'utilisateur, et une batterie de 5.0kWh et une puissance crete de 2.0kWc, et une hypothèse de 10j sans elec possible et 185 épisodes sans elec (ou avec un groupe electrogene): entre 2005 et 2020: Liste des épisodes sans elec (nb de jours): [['14-01-2005', 1], ['19-01-2005', 3], ['23-01-2005', 2], ['01-02-2005', 1], ['11-02-2005', 3], ['17-11-2005', 1], ['26-11-2005', 3], ['02-12-2005', 1], ['04-12-2005', 2], ['07-12-2005', 2], ['12-12-2005', 6], ['22-12-2005', 2], ['25-12-2005', 3], ['31-12-2005', 2], ['03-01-2006', 10], ['18-01-2006', 4], ['29-01-2006', 7], ['23-02-2006', 1], ['09-03-2006', 2], ['21-11-2006', 2], ['25-11-2006', 2], ['28-11-2006', 1], ['30-11-2006', 5], ['08-12-2006', 3], ['20-12-2006', 7], ['08-01-2007', 1], ['20-01-2007', 4], ['30-01-2007', 4], ['06-02-2007', 3], ['08-10-2007', 1], ['10-10-2007', 1], ['01-12-2007', 1], ['07-12-2007', 3], ['15-12-2007', 1], ['25-12-2007', 5], ['01-01-2008', 3], ['05-01-2008', 5], ['11-01-2008', 1], ['18-01-2008', 2], ['22-01-2008', 2], ['30-01-2008', 1], ['01-02-2008', 1], ['07-03-2008', 1], ['11-03-2008', 2], ['17-05-2008', 2], ['28-10-2008', 1], ['30-10-2008', 2], ['02-11-2008', 2], ['06-11-2008', 2], ['15-11-2008', 9], ['30-11-2008', 2], ['03-12-2008', 2], ['12-12-2008', 9], ['25-12-2008', 2], ['04-01-2009', 9], ['14-01-2009', 1], ['23-01-2009', 2], ['19-04-2009', 2], ['16-08-2009', 1], ['11-11-2009', 1], ['02-12-2009', 1], ['04-12-2009', 1], ['06-12-2009', 3], ['14-12-2009', 3], ['23-12-2009', 2], ['28-12-2009', 2], ['04-01-2010', 4], ['09-01-2010', 4], ['14-01-2010', 1], ['16-01-2010', 2], ['05-05-2010', 2], ['01-11-2010', 1], ['04-11-2010', 3], ['08-11-2010', 5], ['25-11-2010', 1], ['28-11-2010', 1], ['02-12-2010', 2], ['17-12-2010', 2], ['22-12-2010', 3], ['07-01-2011', 3], ['13-01-2011', 3], ['31-01-2011', 2], ['04-02-2011', 1], ['23-02-2011', 4], ['02-03-2011', 2], ['29-10-2011', 1], ['05-11-2011', 4], ['25-11-2011', 4], ['25-12-2011', 5], ['14-12-2011', 3], ['21-12-2011', 3], ['13-01-2012', 3], ['23-01-2012', 1], ['08-02-2012', 1], ['15-02-2012', 1], ['20-10-2012', 2], ['06-11-2012', 1], ['20-11-2012', 4], ['25-11-2012', 5], ['14-12-2012', 1], ['22-12-2012', 1], ['05-01-2013', 6], ['19-01-2013', 1], ['22-01-2013', 1], ['26-01-2013', 2], ['08-11-2013', 1], ['10-11-2013', 1], ['15-11-2013', 1], ['17-11-2013', 3], ['23-11-2013', 3], ['03-12-2013', 4], ['25-01-2014', 3], ['01-02-2014', 1], ['15-02-2014', 1], ['24-11-2014', 3], ['01-12-2014', 3], ['06-12-2014', 1], ['13-12-2014', 8], ['24-12-2014', 2], ['27-12-2014', 2], ['03-01-2015', 4], ['10-01-2015', 1], ['24-01-2015', 1], ['29-01-2015', 2], ['03-03-2015', 1], ['12-11-2015', 5], ['19-11-2015', 1], ['26-11-2015', 1], ['04-12-2015', 2], ['03-01-2016', 8], ['21-01-2016', 3], ['01-02-2016', 3], ['16-10-2016', 2], ['05-11-2016', 1], ['18-11-2016', 1], ['22-11-2016', 4], ['20-12-2016', 1], ['24-12-2016', 3], ['30-12-2016', 6], ['11-01-2017', 2], ['06-12-2017', 3], ['10-12-2017', 2], ['13-12-2017', 6], ['22-12-2017', 4], ['28-12-2017', 3], ['03-01-2018', 5], ['14-01-2018', 3], ['20-01-2018', 4], ['30-01-2018', 2], ['05-02-2018', 2], ['08-02-2018', 1], ['20-02-2018', 1], ['28-03-2018', 1], ['28-10-2018', 2], ['02-11-2018', 1], ['09-11-2018', 2], ['04-12-2018', 1], ['09-12-2018', 1], ['13-12-2018', 4], ['21-12-2018', 5], ['29-12-2018', 4], ['08-01-2019', 1], ['13-01-2019', 2], ['22-01-2019', 1], ['30-01-2019', 2], ['23-10-2019', 2], ['02-11-2019', 1], ['04-11-2019', 5], ['16-11-2019', 3], ['01-12-2019', 2], ['07-12-2019', 3], ['12-12-2019', 3], ['23-12-2019', 3], ['04-01-2020', 2], ['29-02-2020', 1], ['04-03-2020', 2], ['23-04-2020', 1], ['12-05-2020', 1], ['06-10-2020', 1], ['03-12-2020', 2], ['10-12-2020', 2], ['13-12-2020', 3], ['18-12-2020', 2], ['21-12-2020', 5], ['28-12-2020', 2]] pour une duree moyenne d'épisode sans elec de 2.5513513513513515j on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse avec les données fournies par l'utilisateur, et une batterie de 5.0kWh et une puissance crete de 2.66kWc, et une hypothèse de 8j sans elec possible et 103 épisodes sans elec (ou avec un groupe electrogene): entre 2005 et 2020: Liste des épisodes sans elec (nb de jours): [['19-01-2005', 3], ['23-01-2005', 1], ['01-02-2005', 1], ['12-02-2005', 2], ['26-11-2005', 3], ['05-12-2005', 1], ['08-12-2005', 1], ['12-12-2005', 6], ['23-12-2005', 1], ['26-12-2005', 2], ['31-12-2005', 1], ['04-01-2006', 4], ['09-01-2006', 4], ['20-01-2006', 2], ['29-01-2006', 4], ['03-02-2006', 2], ['22-11-2006', 1], ['04-12-2006', 1], ['10-12-2006', 1], ['22-12-2006', 5], ['22-01-2007', 2], ['30-01-2007', 4], ['08-02-2007', 1], ['01-12-2007', 1], ['08-12-2007', 2], ['26-12-2007', 4], ['03-01-2008', 1], ['06-01-2008', 4], ['11-01-2008', 1], ['23-01-2008', 1], ['07-03-2008', 1], ['17-05-2008', 2], ['07-11-2008', 1], ['16-11-2008', 8], ['01-12-2008', 1], ['04-12-2008', 1], ['15-12-2008', 6], ['26-12-2008', 1], ['05-01-2009', 8], ['24-01-2009', 1], ['19-04-2009', 2], ['16-08-2009', 1], ['11-11-2009', 1], ['08-12-2009', 1], ['14-12-2009', 3], ['23-12-2009', 2], ['05-01-2010', 3], ['09-01-2010', 2], ['12-01-2010', 1], ['06-05-2010', 1], ['05-11-2010', 2], ['12-11-2010', 1], ['22-12-2010', 3], ['08-01-2011', 1], ['31-01-2011', 2], ['06-11-2011', 3], ['25-11-2011', 4], ['07-12-2011', 2], ['21-12-2011', 3], ['14-01-2012', 2], ['20-10-2012', 2], ['21-11-2012', 2], ['25-11-2012', 2], ['28-11-2012', 1], ['14-12-2012', 1], ['22-12-2012', 1], ['05-01-2013', 6], ['19-01-2013', 1], ['18-11-2013', 2], ['24-11-2013', 2], ['04-12-2013', 2], ['25-01-2014', 3], ['25-11-2014', 2], ['02-12-2014', 2], ['14-12-2014', 7], ['05-01-2015', 2], ['13-11-2015', 2], ['04-01-2016', 5], ['10-01-2016', 1], ['23-01-2016', 1], ['17-10-2016', 1], ['23-11-2016', 1], ['25-11-2016', 1], ['20-12-2016', 1], ['25-12-2016', 2], ['31-12-2016', 5], ['12-01-2017', 1], ['22-12-2017', 4], ['29-12-2017', 1], ['04-01-2018', 3], ['16-01-2018', 1], ['21-01-2018', 3], ['31-01-2018', 1], ['04-12-2018', 1], ['15-12-2018', 2], ['22-12-2018', 4], ['30-12-2018', 3], ['31-01-2019', 1], ['17-11-2019', 2], ['02-12-2019', 1], ['08-12-2019', 1], ['13-12-2019', 2], ['29-12-2020', 1]] pour une duree moyenne d'épisode sans elec de 2.203883495145631j on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse avec les données fournies par l'utilisateur, et une batterie de 5.0kWh et une puissance crete de 3.3200000000000003kWc, et une hypothèse de 6j sans elec possible et 66 épisodes sans elec (ou avec un groupe electrogene): entre 2005 et 2020: Liste des épisodes sans elec (nb de jours): [['20-01-2005', 2], ['23-01-2005', 1], ['01-02-2005', 1], ['12-02-2005', 2], ['27-11-2005', 1], ['12-12-2005', 6], ['27-12-2005', 1], ['06-01-2006', 2], ['09-01-2006', 1], ['11-01-2006', 2], ['29-01-2006', 3], ['03-02-2006', 2], ['24-12-2006', 3], ['23-01-2007', 1], ['01-02-2007', 2], ['27-12-2007', 3], ['23-01-2008', 1], ['07-03-2008', 1], ['17-05-2008', 2], ['16-11-2008', 4], ['21-11-2008', 1], ['23-11-2008', 1], ['19-12-2008', 2], ['26-12-2008', 1], ['05-01-2009', 1], ['07-01-2009', 6], ['19-04-2009', 1], ['16-08-2009', 1], ['14-12-2009', 3], ['24-12-2009', 1], ['05-01-2010', 1], ['10-01-2010', 1], ['12-01-2010', 1], ['05-11-2010', 2], ['23-12-2010', 2], ['01-02-2011', 1], ['07-11-2011', 1], ['26-11-2011', 3], ['21-12-2011', 3], ['14-01-2012', 1], ['20-10-2012', 2], ['21-11-2012', 1], ['26-11-2012', 1], ['14-12-2012', 1], ['05-01-2013', 6], ['24-11-2013', 2], ['05-12-2013', 1], ['26-01-2014', 2], ['26-11-2014', 1], ['02-12-2014', 2], ['17-12-2014', 3], ['13-11-2015', 1], ['05-01-2016', 1], ['07-01-2016', 2], ['17-10-2016', 1], ['23-11-2016', 1], ['01-01-2017', 4], ['23-12-2017', 3], ['06-01-2018', 1], ['22-01-2018', 2], ['31-01-2018', 1], ['16-12-2018', 1], ['23-12-2018', 3], ['30-12-2018', 3], ['02-12-2019', 1], ['14-12-2019', 1]] pour une duree moyenne d'épisode sans elec de 1.8636363636363635j on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse avec les données fournies par l'utilisateur, et une batterie de 5.0kWh et une puissance crete de 3.98kWc, et une hypothèse de 6j sans elec possible et 43 épisodes sans elec (ou avec un groupe electrogene): entre 2005 et 2020: Liste des épisodes sans elec (nb de jours): [['20-01-2005', 2], ['01-02-2005', 1], ['13-02-2005', 1], ['12-12-2005', 5], ['27-12-2005', 1], ['07-01-2006', 1], ['12-01-2006', 1], ['29-01-2006', 1], ['31-01-2006', 1], ['04-02-2006', 1], ['25-12-2006', 2], ['02-02-2007', 1], ['28-12-2007', 2], ['23-01-2008', 1], ['07-03-2008', 1], ['17-05-2008', 2], ['16-11-2008', 4], ['21-11-2008', 1], ['08-01-2009', 2], ['11-01-2009', 2], ['19-04-2009', 1], ['16-08-2009', 1], ['15-12-2009', 2], ['24-12-2009', 1], ['05-01-2010', 1], ['10-01-2010', 1], ['06-11-2010', 1], ['23-12-2010', 2], ['01-02-2011', 1], ['07-11-2011', 1], ['26-11-2011', 3], ['22-12-2011', 2], ['21-10-2012', 1], ['05-01-2013', 6], ['25-11-2013', 1], ['26-01-2014', 1], ['03-12-2014', 1], ['17-10-2016', 1], ['01-01-2017', 4], ['23-12-2017', 3], ['23-01-2018', 1], ['23-12-2018', 1], ['25-12-2018', 1], ['30-12-2018', 3]] pour une duree moyenne d'épisode sans elec de 1.697674418604651j on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse avec les données fournies par l'utilisateur, et une batterie de 5.0kWh et une puissance crete de 4.640000000000001kWc, et une hypothèse de 4j sans elec possible et 22 épisodes sans elec (ou avec un groupe electrogene): entre 2005 et 2020:

Liste des episodes sans elec (nb de jours):
 [[12-12-2005', 1], [14-12-2005', 3], [27-12-2005', 1], [26-12-2006', 1], [07-03-2008', 1], [17-05-2008', 2], [17-11-2008', 3], [09-01-2009', 1], [11-01-2009', 2], [19-04-2009', 1], [16-08-2009', 1], [15-12-2009', 2], [23-12-2010', 2], [27-11-2011', 2], [23-12-2011', 1], [06-01-2013', 4], [25-11-2013', 1], [03-12-2014', 1], [02-01-2017', 3], [24-12-2017', 2], [23-12-2018', 1], [01-01-2019', 1]]
 pour une duree moyenne d'episode sans elec de 1.6818181818181819j
 on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse
 avec les données fournies par l'utilisateur, et
 une batterie de 5.0kWh
 et une puissance crete de 5.300000000000001kWc,
 et une hypothèse de 4j sans elec possible
 et 16 episodes sans elec (ou avec un groupe electrogene):
 entre 2005 et 2020:
 Liste des episodes sans elec (nb de jours):
 [[14-12-2005', 3], [27-12-2005', 1], [17-05-2008', 2], [17-11-2008', 3], [09-01-2009', 1], [12-01-2009', 1], [19-04-2009', 1], [16-08-2009', 1], [15-12-2009', 1], [24-12-2010', 1], [28-11-2011', 1], [06-01-2013', 4], [25-11-2013', 1], [03-12-2014', 1], [03-01-2017', 2], [01-01-2019', 1]]
 pour une duree moyenne d'episode sans elec de 1.5625j
 on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse
 avec les données fournies par l'utilisateur, et
 une batterie de 5.0kWh
 et une puissance crete de 5.96kWc,
 et une hypothèse de 2j sans elec possible
 et 10 episodes sans elec (ou avec un groupe electrogene):
 entre 2005 et 2020:
 Liste des episodes sans elec (nb de jours):
 [[15-12-2005', 2], [17-05-2008', 2], [18-11-2008', 2], [19-04-2009', 1], [16-08-2009', 1], [15-12-2009', 1], [24-12-2010', 1], [06-01-2013', 2], [09-01-2013', 1], [03-12-2014', 1]]
 pour une duree moyenne d'episode sans elec de 1.4j
 on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse
 avec les données fournies par l'utilisateur, et
 une batterie de 5.0kWh
 et une puissance crete de 6.62kWc,
 et une hypothèse de 2j sans elec possible
 et 5 episodes sans elec (ou avec un groupe electrogene):
 entre 2005 et 2020:
 Liste des episodes sans elec (nb de jours):
 [[16-12-2005', 1], [18-05-2008', 1], [19-04-2009', 1], [16-08-2009', 1], [06-01-2013', 2]]
 pour une duree moyenne d'episode sans elec de 1.2j
 on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse
 avec les données fournies par l'utilisateur, et
 une batterie de 5.0kWh
 et une puissance crete de 7.28kWc,
 et une hypothèse de 1j sans elec possible
 et 4 episodes sans elec (ou avec un groupe electrogene):
 entre 2005 et 2020:
 Liste des episodes sans elec (nb de jours):
 [[18-05-2008', 1], [19-04-2009', 1], [16-08-2009', 1], [07-01-2013', 1]]
 pour une duree moyenne d'episode sans elec de 1.0j
 on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse
 avec les données fournies par l'utilisateur, et
 une batterie de 5.0kWh
 et une puissance crete de 7.94kWc,
 et une hypothèse de 1j sans elec possible
 et 4 episodes sans elec (ou avec un groupe electrogene):
 entre 2005 et 2020:
 Liste des episodes sans elec (nb de jours):
 [[18-05-2008', 1], [19-04-2009', 1], [16-08-2009', 1], [07-01-2013', 1]]
 pour une duree moyenne d'episode sans elec de 1.0j
 on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse
 avec les données fournies par l'utilisateur, et
 une batterie de 5.0kWh
 et une puissance crete de 8.600000000000001kWc,
 et une hypothèse de 1j sans elec possible
 et 3 episodes sans elec (ou avec un groupe electrogene):
 entre 2005 et 2020:
 Liste des episodes sans elec (nb de jours):
 [[18-05-2008', 1], [19-04-2009', 1], [16-08-2009', 1]]
 pour une duree moyenne d'episode sans elec de 1.0j
 on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse
 avec les données fournies par l'utilisateur, et
 une batterie de 5.0kWh
 et une puissance crete de 9.260000000000002kWc,
 et une hypothèse de 1j sans elec possible
 et 3 episodes sans elec (ou avec un groupe electrogene):
 entre 2005 et 2020:
 Liste des episodes sans elec (nb de jours):
 [[18-05-2008', 1], [19-04-2009', 1], [16-08-2009', 1]]
 pour une duree moyenne d'episode sans elec de 1.0j
 on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse
 avec les données fournies par l'utilisateur, et
 une batterie de 5.0kWh
 et une puissance crete de 9.92kWc,
 et une hypothèse de 1j sans elec possible
 et 3 episodes sans elec (ou avec un groupe electrogene):
 entre 2005 et 2020:
 Liste des episodes sans elec (nb de jours):
 [[18-05-2008', 1], [19-04-2009', 1], [16-08-2009', 1]]
 pour une duree moyenne d'episode sans elec de 1.0j
 on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse
 avec les données fournies par l'utilisateur, et
 une batterie de 5.0kWh
 et une puissance crete de 10.58kWc,
 et une hypothèse de 1j sans elec possible
 et 3 episodes sans elec (ou avec un groupe electrogene):
 entre 2005 et 2020:
 Liste des episodes sans elec (nb de jours):
 [[18-05-2008', 1], [19-04-2009', 1], [16-08-2009', 1]]
 pour une duree moyenne d'episode sans elec de 1.0j
 on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse

amontage des réservoirs optimisés

avec les données fournies par l'utilisateur, et

une batterie de 10.0kWh

et une puissance crete de 5.96kWc,

et une hypothèse de 0j sans elec possible

et 0 episodes sans elec (ou avec un groupe electrogene):

entre 2005 et 2020:

Liste des episodes sans elec (nb de jours):

[]

pour une duree moyenne d'épisode sans elec de 0j

on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse

avec les données fournies par l'utilisateur, et

une batterie de 5.0kWh

et une puissance crete de 7.28kWc,

et une hypothèse de 1j sans elec possible

et 4 episodes sans elec (ou avec un groupe electrogene):

entre 2005 et 2020:

Liste des episodes sans elec (nb de jours):

[['18-05-2008', 1], ['19-04-2009', 1], ['16-08-2009', 1], ['07-01-2013', 1]]

pour une duree moyenne d'épisode sans elec de 1.0j

on satisfait aux besoins utilisateurs (4.0kWh/jour)entrées en hypothèse