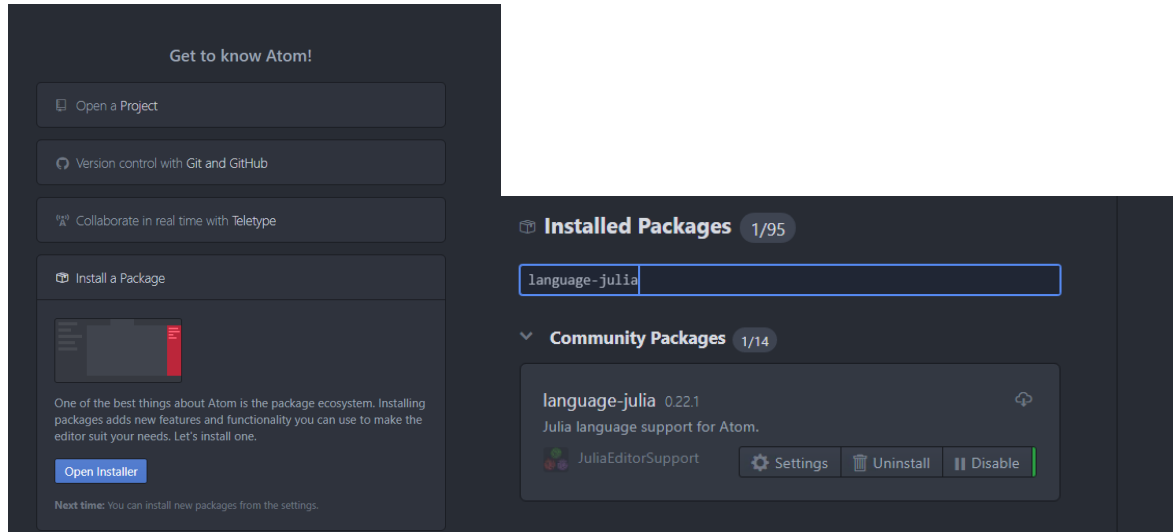


# Instructions pour l'installation et l'utilisation de Julia

Installer Julia : [Download Julia \(julialang.org\)](https://julialang.org)

Installer Atom : <https://atom.en.softonic.com/> et installer le package « language-julia »



## Utilisation de Julia :

- Ouvrir l'invite de commandes (vous pouvez ouvrir File Explorer à l'endroit souhaité et taper « cmd » à la place du chemin du dossier et appuyer sur Entrée)
- Taper « julia » dans l'invite de commandes
- Pour lancer un script, taper `include("nom_du_script.jl")` dans l'invite de commandes Julia

```
C:\Windows\system32>julia
Documentation: https://docs.julialang.org
Type "?" for help, "]?" for Pkg help.
Version 1.7.3 (2022-05-06)
Official https://julialang.org/ release

julia> include("eod.jl")
```

Installer des packages ([Pkg · The Julia Language](https://docs.julialang.org/en/v1/stdlib/Pkg/)) :

- Appuyer sur la touche « ] » (Alt Gr + touche °) dans l'invite de commandes Julia
- Taper « add XXXX.jl » avec XXXX le nom du package à installer

```
D:\Users\4106GX\OneDrive - GRTgaz\Documents>julia
Documentation: https://docs.julialang.org
Type "?" for help, "]?" for Pkg help.
Version 1.7.3 (2022-05-06)
Official https://julialang.org/ release

(@v1.7) pkg> add JuMP
```

- Appuyer sur la touche Retour en arrière pour revenir à l'invite de commandes Julia

A faire : installer les packages JuMP, HiGHS et XLSX
--

JuMP est le package Julia pour l'optimisation ([Introduction · JuMP](https://jump.dev/JuMP.jl/stable/tutorials/getting_started/getting_started_with_JuMP/) [https://jump.dev/JuMP.jl/stable/tutorials/getting\\_started/getting\\_started\\_with\\_JuMP/](https://jump.dev/JuMP.jl/stable/tutorials/getting_started/getting_started_with_JuMP/)). Il est facile à prendre en main et permet de résoudre un grand nombre de problèmes (exemples : Economic Dispatch & Unit Commitment : [Power Systems · JuMP](#)).

HiGHS est l'un des solveurs open-source compatibles avec JUMP pour résoudre des problèmes d'optimisation linéaires continus ou avec des variables entières. Il existe cependant beaucoup de solveurs compatibles avec JuMP ([Installation Guide · JuMP](#)). Il faudra donc dans ce projet réfléchir au choix du solveur. Evidemment il faudra choisir un solveur compatible avec le type de problème d'optimisation à résoudre. Ensuite, il faudra choisir un solveur avec un algorithme adapté : a-t-on besoin d'obtenir l'optimum global ou local ou bien une solution approchée obtenue par une heuristique est-elle acceptable ? A-t-on un problème difficile à résoudre (par exemple problème de grande taille) ? Il faudra se documenter pour savoir quels solveurs sont considérés comme l'état de l'art pour telle ou telle classe de problème. Il ne faudra pas hésiter à comparer des solveurs entre eux si on n'est pas satisfait du solveur choisi (sur des petits problèmes il n'y aura peut-être pas de différence significative mais sur des problèmes réels il y aura sûrement des différences importantes).

XLSX est un package qui permet de lire des fichiers Excel, ce qui peut être très pratique pour lire des données d'entrée et ainsi éviter de les écrire à la main. Attention : pour lire un fichier Excel, celui-ci doit être fermé sinon cela entraîne des problèmes d'accès.

## TP phase n°1 : EOD pas à pas

Afin de prendre en main Julia et son module optimisation JuMP, nous allons construire un problème d'Equilibre Offre-Demande (EOD) simple portant uniquement sur le vecteur électricité pour une seule et unique zone. Nous allons construire ce problème pas à pas.

### 1. EOD simple

Nous commençons par une approche simple : nous avons une consommation d'électricité à satisfaire (2200 MWh) et cinq moyens de production d'électricité pour satisfaire cette demande :

- Une centrale nucléaire avec une capacité de 900 MW et un prix de marché égal à 14€/MWh,
- Une deuxième centrale nucléaire avec une capacité de 900 MW et un prix de marché égal à 16€/MWh,
- Une centrale CCG (Cycle Combiné Gaz) avec une capacité de 300 MW et un prix de marché égal à 45€/MWh
- Une centrale hydraulique avec une capacité de 300 MW et un prix de marché égal à 48€/MWh,
- Un parc éolien avec une capacité de 300 MW et un prix de marché égal à 0€/MWh.

L'objectif consiste à déterminer le plan de production qui satisfait cette demande au moindre coût.

Un cas aussi simple peut évidemment se résoudre à la main. Cependant, il peut aussi être formulé comme un problème d'optimisation qui est implémenté en Julia dans le fichier « eod\_step\_by\_step\_to\_complete.jl » entre les lignes 10 et 52.

### 2. EOD avec contrainte de puissance minimum

Le premier problème d'EOD que nous avons vu a le mérite d'être simple mais il n'est pas réaliste. En effet, pour chaque moyen de production, il existe des contraintes techniques de fonctionnement qu'il

faut prendre en compte. La première contrainte de fonctionnement que nous étudions est la contrainte de puissance minimale de fonctionnement : certains moyens de production, s'ils sont démarrés, doivent fonctionner à une puissance minimale. Dans notre exemple, les deux centrales nucléaires doivent fonctionner à 300 MW minimum si elles sont démarrées et la centrale CCG doit fonctionner à 150 MW minimum si elle est démarrée.

Exercice : modéliser cette contrainte de puissance minimum (indice : introduire des variables binaires qui indiquent si chaque groupe de production est en fonctionnement ou à l'arrêt) et l'ajouter dans le problème d'optimisation dans le fichier « eod\_step\_by\_step\_to\_complete.jl » entre les lignes 56 et 101.

### 3. EOD avec contrainte de rampe

Les deux problèmes d'EOD que nous venons de voir portaient uniquement sur une heure mais en pratique, il faut prévoir un plan de production pour plusieurs heures consécutives. Dans cette troisième version du problème d'EOD, nous cherchons donc un plan de production pour trois heures consécutives (consommations à satisfaire : 2200 MWh, 2450 MWh et 1900 MWh). De plus, nous introduisons une nouvelle contrainte technique de fonctionnement : la contrainte de rampe. Cette contrainte limite la variation de puissance d'un moyen de production entre deux heures consécutives. Dans notre exemple, nous imposons une variation maximale de 350 MW/h pour les centrales nucléaires et une variation maximale de 200 MW/h pour la centrale CCG.

Remarque : ces données sont fictives et ont pour seul objectif de complexifier progressivement le problème d'optimisation, en réalité la contrainte de rampe sur ces moyens de production n'est pas limitante au pas de temps horaire.

Exercice : modéliser cette contrainte de rampe et l'ajouter dans le problème d'optimisation dans le fichier « eod\_step\_by\_step\_to\_complete.jl » entre les lignes 105 et 157.

### 4. EOD avec contrainte de durée de fonctionnement

Une autre contrainte de fonctionnement importante concerne la durée de fonctionnement des moyens de production. En effet, certains moyens de production ont une contrainte de durée minimum de fonctionnement et/ou d'arrêt. Dans notre exemple, les centrales nucléaires ont une durée minimum de fonctionnement de vingt-quatre heures et la centrale CCG a une durée minimum de fonctionnement de trois heures.

Exercice : modéliser cette contrainte de durée de fonctionnement (indice : introduire de nouvelles variables binaires UP égales à 1 si le groupe démarre à l'instant t et 0 sinon) et l'ajouter dans le problème d'optimisation dans le fichier « eod\_step\_by\_step\_to\_complete.jl » entre les lignes 161 et 226.

## TP phase n°2 : EOD hebdomadaire

Nous allons maintenant étudier un problème d'EOD un peu plus réaliste mais portant toujours uniquement sur le vecteur électricité pour une seule zone : un problème d'EOD sur une semaine entière (soit 168 heures) avec

- quatre moyens de production renouvelable :
  - solaire,
  - éolien,
  - hydraulique fil de l'eau,
  - une dernière catégorie fatale regroupant la biomasse, les déchets et la cogénération) ;
- cinq moyens de production thermique :
  - nucléaire,
  - charbon,
  - gaz semi-base,
  - gaz pointe,
  - fioul ;
- un moyen de production hydraulique de type lac ;
- une STEP (Station de Transfert d'Energie par Pompage) ;
- une batterie.

Dans toute la suite, nous utilisons les variables suivantes décrites dans le tableau suivant.

Indices	Variables
$t$ : pas de temps	$P_{g,t}^{th}$ : production des centrales thermiques
$g$ : centrale thermique	$P_t^{hy}$ : production de la centrale hydraulique
$h$ : centrale hydraulique	$P_t^{res}$ : production de la centrale éolienne
	$UC_{g,t}^{th}$ : binaire indiquant si la centrale thermique $g$ est en fonctionnement ou à l'arrêt à l'instant $t$
<b>Exposants</b>	$UP_{g,t}^{th}$ : binaire indiquant si la centrale thermique $g$ est démarrée à l'instant $t$
$th$ : moyen de production thermique	$DO_{g,t}^{th}$ : binaire indiquant si la centrale thermique $g$ est arrêtée à l'instant $t$
$hy$ : moyen de production hydraulique	
$res$ : moyen de production renouvelable	

### 1. Introduction des variables d'énergie non fournie et d'énergie en excès

Dans ce problème d'EOD, nous introduisons deux nouvelles variables :  $P_t^{uns}$  et  $P_t^{exc}$  qui représentent respectivement l'énergie non fournie (unsupplied energy) et l'énergie en excès (spilled energy) à chaque pas de temps  $t$ . En effet, dans certaines situations, il est possible de ne pas réussir à satisfaire la demande (par exemple en cas de panne) ou bien d'avoir trop d'énergie (par exemple trop d'énergies renouvelables ou de centrales ne pouvant s'arrêter). Ces situations ont un coût car ce sont des situations que l'on souhaite éviter : en particulier, une situation d'énergie non fournie peut coûter très cher. Ainsi, ces deux nouvelles variables apparaissent dans la fonction objectif et dans la contrainte d'équilibre offre-demande.

$$\min \sum_{t=1}^T \left( \sum_{h=1}^{N_{hy}} c_{h,t}^{hy} P_{h,t}^{hy} + \sum_{g=1}^{N_{th}} c_{g,t} P_{g,t}^{th} + c^{uns} P_t^{uns} + c^{exc} P_t^{exc} \right)$$

## SOUS CONTRAINTES

Equilibre offre-demande :

$$\sum_{g=1}^{N_{th}} P_{g,t}^{th} + \sum_{h=1}^{N_{hy}} P_{h,t}^{hy} + P_t^{res} + P_t^{uns} = load_t + P_t^{exc} \quad \forall t = 1..T$$

Ce problème est modélisé dans le fichier « eod\_1\_week\_to\_complete.jl ».

### 2. Modélisation de l'hydraulique lac

Jusqu'à présent nous avons modélisé la production hydraulique comme une production pilotable mais sans stock. Or, certains ouvrages hydrauliques comme les centrales « lac » permettent de stocker une certaine quantité d'eau. De manière simplifiée, cela signifie que la quantité d'énergie disponible pendant la semaine est limitée. Ainsi, il faut utiliser l'énergie disponible aux moments les plus judicieux. Dans notre exemple, le réservoir de l'hydraulique lac est équivalent à 3 TWh électrique stocké.

Exercice : modéliser cette contrainte de stock et l'ajouter dans le problème d'optimisation dans le fichier « eod\_1\_week\_to\_complete.jl » ligne 120.

### 3. Modélisation d'une STEP hebdomadaire

Un autre type d'ouvrage hydraulique, les STEP (stations de pompage turbinage) permettent de stocker de l'énergie. Plus précisément, les STEP sont composées de deux bassins aux altitudes différentes et l'idée est de pomper l'eau du bassin inférieur vers le bassin supérieur lorsque la demande en électricité est faible. A l'inverse, lorsque la demande est élevée, l'eau du bassin supérieur est transférée dans le bassin inférieur pour activer la turbine et générer de l'électricité (comme une centrale hydraulique classique). Dans notre exemple, une STEP hebdomadaire est disponible avec une capacité de pompage/turbinage de 1200 MW et un rendement de 75% pour le pompage.

Exercice : modéliser cette STEP (indice : utiliser trois variables Pcharge\_STEP, Pdecharge\_STEP et stock\_STEP qui indiquent respectivement le pompage de la STEP à un instant t, le turbinage de la STEP à un instant t et le stock disponible à un instant t) et l'ajouter dans le problème d'optimisation dans le fichier « eod\_1\_week\_to\_complete.jl » ligne 122. Analyser l'impact de la STEP sur les résultats et sur le coût total

### 4. Modélisation d'une batterie de quelques heures

Le dernier élément que nous ajoutons est une batterie de quelques heures. En termes de modélisation, une batterie fonctionne comme une STEP, seule la durée de stockage est différente.

Dans notre exemple, nous avons une batterie de 280 MW qui a un rendement de 85% (pour la charge et pour la décharge) et une durée d'utilisation de deux heures.

Exercice : modéliser cette batterie et l'ajouter dans le problème d'optimisation dans le fichier « eod\_1\_week\_to\_complete.jl » ligne 125. Analyser l'impact de la batterie sur les résultats et sur le coût total

## Le projet

Le problème d'EOD sur lequel nous avons travaillé comporte encore des limites : l'hydraulique lac est modélisé grossièrement, les interdépendances avec les voisins ou avec les autres vecteurs énergétiques ne sont pas prises en compte, les effacements de la consommation ne sont pas modélisés et les investissements ne sont pas du tout représentés dans ce problème. L'objectif du projet consiste à améliorer le problème d'EOD vu en cours en choisissant un sujet d'approfondissement parmi ceux proposés (vous pouvez aussi faire une autre proposition). Plusieurs modélisations pourront être testées et comparées afin de déterminer les avantages et les inconvénients de chacune. Chaque choix de modélisation devra être clairement expliqué, justifié, testé et analysé.

NB : le problème d'EOD du projet sera légèrement différent de celui vu en cours (moyens de production différents, contraintes en plus ou en moins, etc). Il faudra donc adapter le problème aux données fournies.

## Les consignes

- Le projet sera fait en groupe de trois personnes.
- Chaque groupe devra :
  - Présenter son choix de sujet, les pistes qui seront explorées et éventuellement de premiers résultats pendant la séance du vendredi 03/02/2023 (5 minutes de présentation, 10 minutes de discussion) et fournir en amont de la séance un ou deux transparents présentant le projet ;
  - Fournir un rapport détaillant le problème traité, les solutions envisagées et testées. Chaque étape doit être décrite précisément en justifiant chaque choix :
    - Quel est le problème traité ?
    - Quelle approche a été choisie ?
    - Quelle modélisation du problème ?
    - Quelles données ont été choisies ?
    - Quel type de problème d'optimisation obtenez-vous ?
    - Quel solveur d'optimisation a été choisi pour la résolution ? Pourquoi ? Quel algorithme utilise le solveur choisi ?
    - Les résultats doivent être analysés et leur pertinence doit être illustrée par des graphes et des tableaux,
    - Les avantages et inconvénients de chaque méthode doivent être précisés

Ainsi, le rapport devra contenir une analyse critique de tout ce qui a été fait. En particulier, toutes les limites devront être explicitées et des perspectives pourront être proposées ;

- Fournir le code utilisé (code en Julia utilisant le module JuMP) ;
- Le rapport et le code sont à rendre le vendredi 03/03/2023 à 12h au plus tard sur LMS ;
- Présenter son projet à l'aide de slides lors d'une soutenance finale le vendredi 10/03/2023 (10 minutes de présentation, 10 minutes de question)

**ATTENTION** : le respect du temps de présentation compte dans l'évaluation (la présentation doit durer entre 9 et 10 minutes, elle ne doit surtout pas dépasser les 10 minutes)

L'évaluation porte d'une part sur des critères relatifs à la qualité de la production tels que

- la pertinence des modèles ;
- la qualité du code développé ;
- les performances obtenues (la résolution de votre problème d'optimisation doit être rapide) ;
- la critique des modèles / algorithmes développés et des résultats obtenus ;
- la qualité des rendus écrits ;
- la qualité de la soutenance orale (respect du temps de présentation, présentation claire et précise, réponses pertinentes aux questions, prise de recul sur le travail effectué).

Elle porte d'autre part sur des critères relatifs au processus de travail du groupe

- la prise d'initiatives ;
- le travail d'équipe.

Pour toute question, nous contacter via les adresses mail suivantes :

- [julie.sliwak@grtgaz.com](mailto:julie.sliwak@grtgaz.com)
- [marion.li@rte-france.com](mailto:marion.li@rte-france.com)