

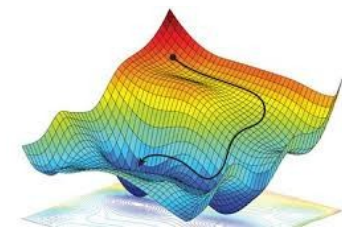
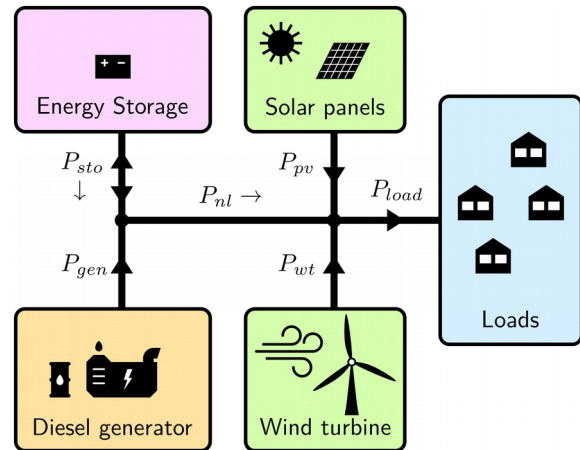
# Optimisation des systèmes énergétiques par différentiation automatique de code Julia

Sujet de stage de Master 2 pour le printemps 2020

Disciplines : génie électrique, méthodes numériques (optimisation)

## Objectifs et enjeux

La conception des systèmes énergétiques (ex. : microréseau) se fait généralement à l'aide d'outils logiciels qui optimisent le dimensionnement (taille des moyens de production, de stockage...) pour minimiser le coût (économique ou écologiques) du système sur son cycle de vie. Cette optimisation peut être chronophage, car l'évaluation de la fonction coût se fait par un calcul « boîte noire » (p. ex. simulation temporelle du fonctionnement du système sur de longues périodes). La durée de calcul est d'autant plus longue que le nombre de paramètres à optimiser est grand. Par conséquent, pour explorer des configurations de systèmes énergétiques *plus originales et plus performantes*, il faut disposer d'outils permettant de simuler et d'optimiser ces systèmes de façon *nettement plus rapide*.



Dans ce projet, nous proposons d'explorer et d'analyser les performances des solutions logicielles open source actuelles permettant la différentiation automatique (AD), c'est-à-dire le calcul numérique exact du gradient d'une fonction (au sens informatique), sans passer par les différences finies (coûteux en calcul et imprécis). Appliqué au dimensionnement de systèmes énergétiques, cela doit permettre d'utiliser des méthodes d'optimisation utilisant le gradient (ex. : BFGS, SQP...) dont il est connu qu'elles convergent nettement plus vite que les méthodes sans gradient.

Julia ([julialang.org](http://julialang.org)) est un langage de calcul scientifique récent dont la popularité va grandissante, en particulier dans la communauté de l'optimisation. Sa syntaxe proche de Matlab et Python facilite son apprentissage. Deux caractéristiques de Julia en font un outil très prometteur: la compilation Just-in-Time (JIT) qui accélère l'exécution et sa forte capacité d'introspection qui facilite des implémentations performantes de l'AD.

La différentiation automatique n'est pas une technique nouvelle, mais elle était jusqu'à présent réservée aux spécialistes, ou bien nécessitait de décrire le problème d'optimisation dans un langage spécialisé souvent propriétaire. Elle connaît un regain d'intérêt avec le Deep Learning, car la procédure d'apprentissage des réseaux de neurones repose fondamentalement sur l'AD. Ainsi, nous voulons profiter de l'effervescence actuelle autour de l'AD pour l'appliquer au problème du dimensionnement des systèmes énergétiques.

## Sujet du stage

Le cas test de système énergétique à dimensionner sera un microréseau sur un site isolé (données réelles de l'île d'Ouessant). Le ou la stagiaire s'appuiera sur des résultats préliminaires très prometteurs pour le dimensionnement de microréseau avec Julia, sur des cas cependant simplistes. Beaucoup de questions restent en suspens quant à la capacité à traiter des problèmes plus réalistes et donc plus complexes.

Il s'agira de mettre au point un simulateur de microréseau Julia plus complexe et de vérifier qu'il se prête bien à la différentiation automatique. Par ailleurs, l'effervescence sur l'AD fait qu'il y a plusieurs packages disponibles actuellement, mais un seul (ForwardDiff.jl) a été testé pour l'instant.

Ce stage comportera des interactions avec un autre stage de Master 2 au laboratoire G2Elab (à Grenoble) et dont le sujet est d'explorer les solutions d'AD, mais avec le langage Python. Le but est de comparer les avantages respectifs de ces deux langages open source.

## Livrables attendus

- Réalisation d'un simulateur de microréseau en Julia (sur la base des expériences précédentes du laboratoire)
- Comparaison des performances de différents packages d'AD en Julia
- Comparaison avec les cas tests étudiés au G2Elab avec le langage Python

## Compétences abordées

- Modélisation des systèmes énergétiques (en particulier : production photovoltaïque & éolienne, batteries, gestion d'énergie...)
- Programmation en Julia

NB : la connaissance préalable de Julia n'est *pas un prérequis*. Par contre, il est souhaitable de maîtriser un autre langage de calcul scientifique (Matlab, Python...).

## Contact

**Pierre Haessig**, Enseignant à CentraleSupélec, campus de Rennes,  
Chercheur dans l'équipe d'Automatique (AUT) de l'IETR.

- Email : [pierre.haessig@centralesupelec.fr](mailto:pierre.haessig@centralesupelec.fr)
- Page personnelle : <https://pierreh.eu/>