

Expériences numériques sur des modèles de système 0D/1D avec Modelica et Python

Modelica est un langage libre pour programmer des modèles physiques sous forme de systèmes algébriques d'**équations différentielles** (<http://book.xogeny.com/front/intro/>). Les équations s'écrivent pour ainsi dire en langage (mathématique) naturel. Grâce à l'approche acausale, la mise en équations et leur résolution numérique sont deux problèmes complètement distincts, le second étant pour l'essentiel pris en charge par des outils, dont le logiciel libre OpenModelica (<https://www.openmodelica.org/>). Le paradigme d'objets de Modelica est mis à profit pour construire des bibliothèques de modules réutilisables et combinables, permettant même la construction de modèles exclusivement depuis une interface graphique !

Des développements récents, en particulier la **norme FMI** (*functional mock-up interface*, <http://fmi-standard.org/>) et les outils associés, permettent d'appliquer à ces modèles numériques toutes la panoplie d'algorithmes mathématiques disponibles notamment dans l'univers **Python**.

Cette formation offre un panorama des outils disponibles pour réaliser des plans d'expérience de simulations avec des modèles écrits en Modelica, ainsi que pour utiliser des programmes Python à l'intérieur de modèles Modelica. Une connaissance préalable de l'un ou l'autre des langages Modelica et Python est préférable, mais pas indispensable.


Informations pratiques

- Accueil : 9h30
- Fin : 17h30
- Lieu : Phimeca Engineering, 18 boulevard de Reuilly, 75012, Paris
- Tarif : 850 € (400 € pour les étudiants).

Les inscriptions se font par courriel à girard@phimeca.com. Ne tardez pas : le nombre de places disponibles est restreint pour garantir l'interactivité de la séance.

Programme

Modélisation des systèmes avec Modelica


- Qu'est-ce que la « modélisation système » ?
- Pourquoi choisir Modelica?
- Les outils pour Modelica
 - Dymola
 - OpenModelica
 - Outils additionnels
-  Installation d'OpenModelica

Pourquoi compléter Modelica avec un langage procédural ?


- Pilotage

- Réaliser un plan d'expérience pour
 - l'optimisation,
 - l'estimation de paramètres et le calage,
 - l'analyse de sensibilité,
 - la propagation d'incertitudes...
- La norme FMI (*functional mockup-up interface*) : des boîtes noires standardisées pour les modèles numériques
- Inclusion
 - Utiliser des bibliothèques d'algorithmes mathématiques dans des modèles Modelica
 - Efficacité calculatoire
 - Gestion commode des données externes
 - *software in the loop*, émulateur de modèles numériques coûteux
- Pourquoi choisir Python?
 - Polyvalence, facilité d'usage, communauté
 - L'API C/Python, et autres ponts
 - Python et la norme FMI: PyFMI

Piloter des modèles Modelica avec Python

- Interface de données d'entrée-sortie
- Piloter des FMU avec le module Python PyFMI
-  Estimation de paramètre et propagation d'incertitude avec un modèle d'épidémie

Inclure du code Python dans des modèles Modelica

- Le mot-clef Modelica `external` : inclusion de code C ou Fortran
- Brancher Python sur le connecteur C
-  Inclusion d'un modèle réduit en Python d'un simulateur coûteux dans un modèle Modelica