

# Modélisation sous Matlab/Simulink ou Scilab/Xcos

[thierry.mimar@acssysteme.com](mailto:thierry.mimar@acssysteme.com)



# Création d'un modèle

## ■ La démarche de modélisation

Conception

Développement

Calage

Validation



```
Fig1:  
CreateRectPgn(80,160,80+  
gn.FtInRegion(point))  
  
double x0c,y0c;  
for (int i=0;i<NUNCITY;i++)  
x0c=80+RectSizeX*(xEnd  
y0c=160+RectSizeX*(yEnd  
-gn.FtInRegion(point))  
- CreateEllip
```



# 1. Conception du modèle

## ■ Définition et objectif

### Qu'est-ce qu'un modèle

Représentation mathématique d'un processus permettant de simuler son comportement dans le temps

### Pourquoi un modèle

- comprendre le fonctionnement d'un système
- évaluer les performances d'un système
- dimensionner un système
- synthétiser une loi de commande
- valider les performances et la robustesse d'une loi de commande

### Approche « Model-Based Design »



# 1. Conception du modèle

## Finalité du modèle

### Le modèle de conception

- destiné à la synthèse d'une loi de commande
- contraintes de modélisation liées à la méthode de commande choisie
- généralement linéaire

### Le modèle de validation

- destiné à la simulation pour valider la loi de commande
- généralement non linéaire





# 1. Conception du modèle

## ■ Phénomènes à simuler

### En fonction des objectifs

La précision du modèle et des phénomènes représentés dépend avant tout de ce que l'on veut faire avec le modèle !

### Exemple

La modélisation d'une voiture pourra être très différente suivant que l'on veut :

- mettre au point un régulateur de vitesse,
- étudier l'usure des pneus,
- optimiser la consommation...



```
Fig1:  
CreateRectPgn(80,160,80+  
gn.PtInRegion(point))  
  
double x0c,y0c;  
for (int i=0;i<NUNCITY;i++)  
{  
x0c=80+RectSizeX*(xEnd  
y0c=160+RectSizeY*(yEnd  
-gn.RgnEllipse;  
- CreateEllipse
```



# 1. Conception du modèle

## Type de modèle

### Modèles « boîte blanche »

- modèles basés sur les équations physiques du système → modèle de connaissance
- souvent définis avec l'aide d'un ingénieur métier

### Modèles « boîte noire »

- modèles identifiés à partir de données d'essais (acquisitions sur le système réel ou simulation d'un modèle complexe) → modèle de comportement

### Modèles « boîte grise »

- modèles mixtes

# 1. Conception du modèle

## Type de modélisation

### Causale

- modèle constitué de composants « orientés ». Chaque élément (typiquement un fonction de transfert) possède une entrée qui impose une sortie (et non l'inverse)

### Acausale

- modèle composé d'objets qui définissent des relations entre les variables sans notion de cause à effet



```
gfn:  
CreateRectPgn(80,160,80+  
gn.PtInRegion(point))  
  
double x0c,y0c;  
for (int i=0;i<NUNCITY;i++)  
x0c=80+RectSizeX*(x2c  
y0c=160+RectSizeY*(y2c  
gn.RgnEllipse:  
- CreateEllipse
```



# 1. Conception du modèle

## Exemple : retenue d'eau d'un barrage

### Objectif du modèle

- Concevoir un correcteur de niveau qui sera implémenter dans le système de contrôle/commande de l'aménagement

### Phénomènes à simuler

- L'évolution dans le temps du niveau d'eau dans la retenue en fonction des débits entrant et sortant
- Les limites physiques du débit sortant et du niveau d'eau

### Type de modèle

- Boîte blanche, car on dispose des équations





```

fig:
CreateRectPgn(80,160,80+
jn.PtInRegion(point))
double x0c,y0c;
for (int i=0;i<NUNCITY;i++)
x0c=80+RectSizeX*(x2c
y0c=160+RectSizeY*(y2c
cban_rgnEllipse:
CreateEllipse

```



# 1. Conception du modèle

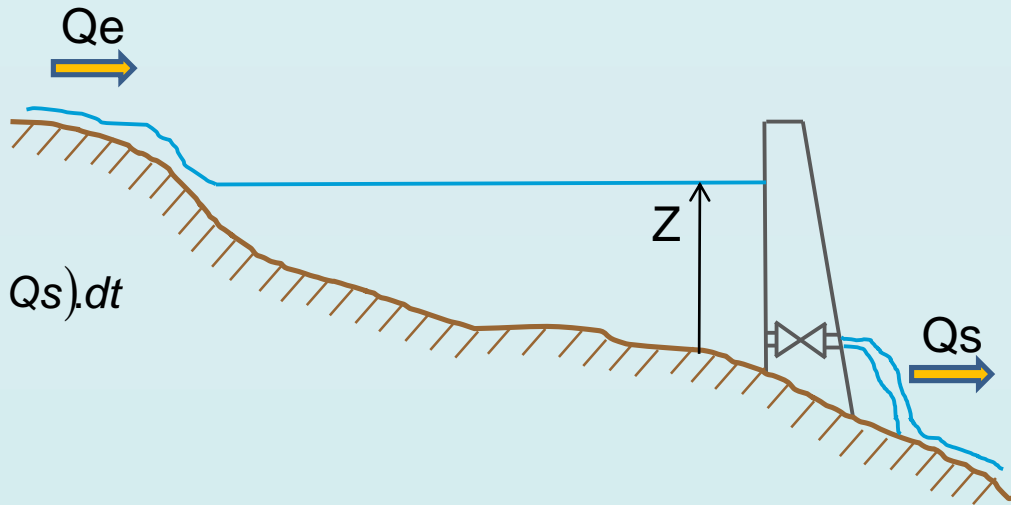
Exemple : retenue d'eau d'un barrage

Équations dynamiques de la retenue

$Z$  : niveau d'eau, m  
 $S$  : surface du plan d'eau,  $m^2$   
 $Q_e$  : débit entrant ( $m^3/s$ )  
 $Q_s$  : débit sortant ( $m^3/s$ )

$$Z(t) = Z_0 + \frac{1}{S} \int_0^t (Q_e - Q_s) . dt$$

$$\frac{dZ}{dt} = \frac{1}{S} (Q_e - Q_s)$$





## 2. Développement du modèle

### Environnement de simulation

#### Modélisation causale

- Matlab/Simulink (société Mathworks)
- Scilab/Xcos (logiciel libre et gratuit)

#### Modélisation physique (acausale)

- Simscape dans Matlab/Simulink (société Mathworks)
- Modelicac dans Scilab/Xcos (logiciel libre et gratuit)
- AMESim (société LMS International)
- Dymola (société Dassault Systèmes)
- MATRIXx/SystemBuild (société National Instruments)



```
function CreateRectPgn(80,160,80+  
pgn.PtInRegion(point))  
  
double x0c,y0c;  
for (int i=0;i<NUNCITY;i++)  
x0c=80+RectSizeX*(xEnd  
y0c=160+RectSizeX*(yEnd  
pgn.PgnEllipse;  
pgn.PgnEllipse;
```



## 2. Développement du modèle

### ■ Environnement Matlab/Simulink

#### Matlab

- langage de programmation de haut niveau
- environnement interactif pour le calcul numérique, la visualisation et la programmation

#### Simulink

- plate-forme de simulation et de modélisation de systèmes dynamiques
- environnement graphique permettant de créer des modèles à partir d'un ensemble de bibliothèques, puis de les simuler
- Intégré à Matlab et compatible avec ses nombreuses boites à outils

→ *Démonstration*



## 2. Développement du modèle

### ■ Environnement Scilab/Xcos

#### Scilab

- ✔ outil de programmation scientifique similaire à Matlab
- ✔ libre et gratuit
- ✔ néanmoins plus limité

#### Xcos

- ✔ outil de modélisation et de simulation similaire à Simulink
- ✔ libre et gratuit
- ✔ néanmoins plus limité

→ *Démonstration*





```
function  
CreateRectPgn(80,160,80+  
gn.PtInRegion(point))  
  
double x0c,y0c;  
for (int i=0;i<NUNCITY;i++)  
x0c=80+RectSizeX*(x2c  
y0c=160+RectSizeY*(y2c  
-gn.PgnEllipse;  
- CreateEllipticalPgn
```



## 2. Développement du modèle

### ■ Modélisation sous forme de schéma bloc Concepts communs à Simulink et Xcos :

- Un **éditeur de schémas** permettant de coder des équations sous forme graphique
- Des **bibliothèques de blocs** thématiques (sources, visualisation, enregistrement, opérations mathématique, discontinuités...)
- Un moteur de simulation numérique qui résout le diagramme puis appelle un **solveur** ad hoc pour simuler un **scénario** défini par l'utilisateur
- La possibilité de créer des « sous-systèmes » pour **encapsuler** des parties
  - amélioration de la lecture des diagrammes
  - restitution de l'architecture du système physique (différents organes ou sous-ensembles)
  - capitalisation des développements (création de bibliothèques de blocs métier)

→ *Démonstration*



```

function CreateRectPgn(80,160,80+
jg.PtInRegion(point))
double x0c,y0c;
for (int i=0;i<NUMCITY;i++)
x0c=80+RectSizeX*(x2c
y0c=160+RectSizeY*(y2c
pgn.PgnEllipsoe:
pgn.PgnEllipsoe:

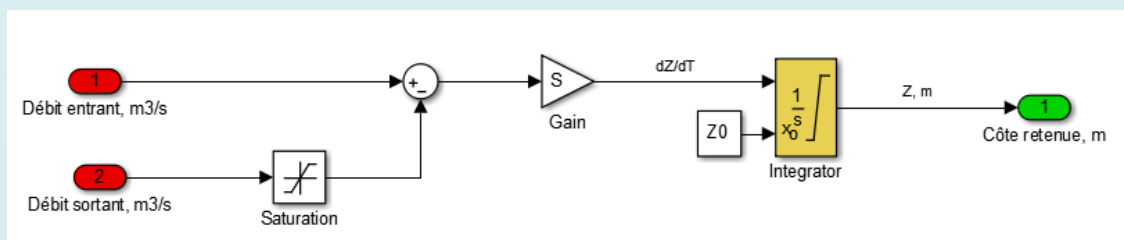
```



## 2. Développement du modèle

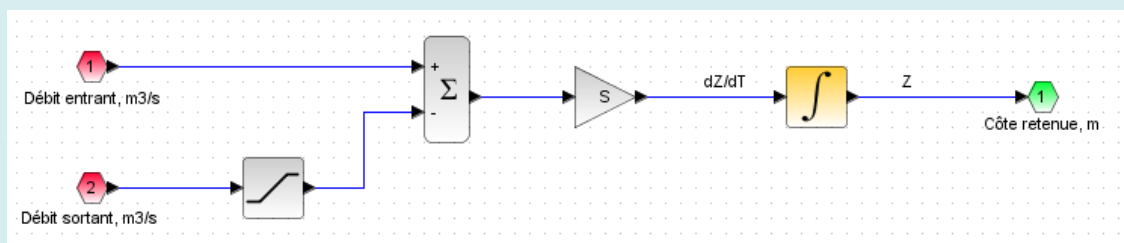
Exemple : retenue d'eau d'un barrage

Matlab/Simulink



Scilab/Xcos

$$Z(t) = Z_0 + \frac{1}{S} \int (Q_e - Q_s) dt$$



MathWorks®

scilab  
enterprises

id4CAR  
Moteur d'idées pour véhicules spécifiques et mobilité durable

NATIONAL  
INSTRUMENTS®

SHERPA  
ENGINEERING

FAAR®  
2004 - 2014 industry

acsystème®  
analyse & contrôle des systèmes



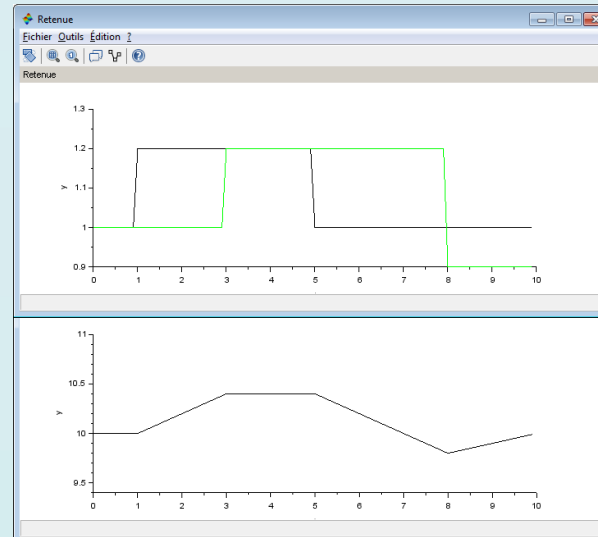
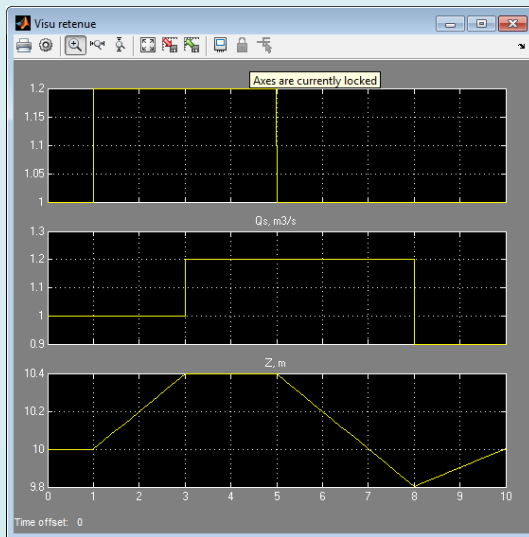
CAP'TRONIC

## 2. Développement du modèle

Exemple : retenue d'eau d'un barrage

Matlab/Simulink

Scilab/Xcos



MathWorks®

scilab  
enterprises

id4CAR  
Moteur d'idées pour véhicules spécifiques et mobilité durable

NATIONAL  
INSTRUMENTS®

SHERPA  
ENGINEERING

FAAR®  
2004 - 2014 industry

acsystème®  
analyse & contrôle des systèmes



CAP'TRONIC

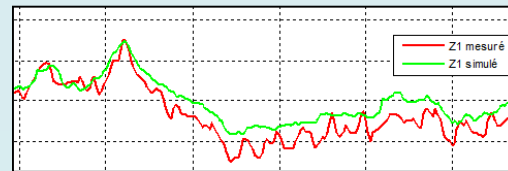
# 3. Calage du modèle

## Objectif

- on dispose d'un modèle capable de simuler les phénomènes physiques recherchés de manière cohérente (sens de variation des sorties en fonction des entrées, ordre de grandeurs des variables...)
- il s'agit d'ajuster les paramètres du modèle de manière à affiner sa représentativité (ses « prédictions ») vis-à-vis du système réel.

## Moyen

- réalisation d'essais sur le système réel
- confrontation des acquisitions et des résultats de simulation obtenus en injectant les mêmes excitations en entrée du modèle
- techniques d'identification et d'optimisation





## 4. Validation du modèle

### Objectif

- s'assurer de la capacité du modèle à représenter fidèlement le comportement du système avant de l'exploiter
- vérifier sa plage de validité, quantifier ses limites

### Moyen

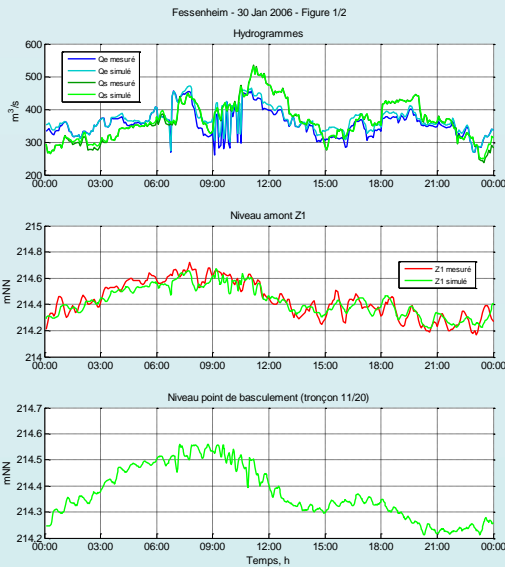
- confrontation d'acquisitions et de résultats de simulation sur des jeux de données différents de ceux ayant servis au calage
- analyse qualitative et quantitative



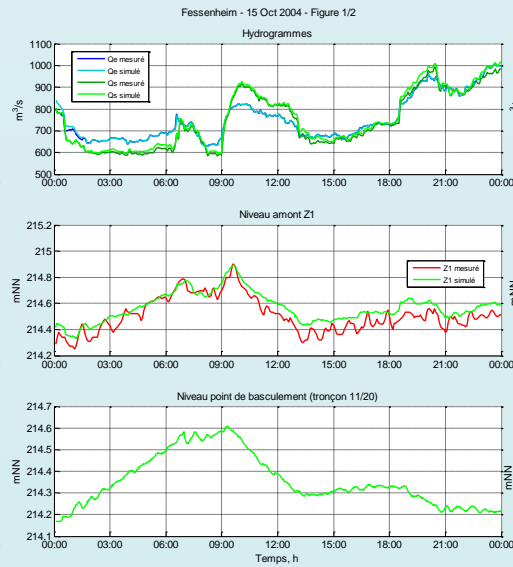
# 4. Validation du modèle

## Exemple : modèle de bief

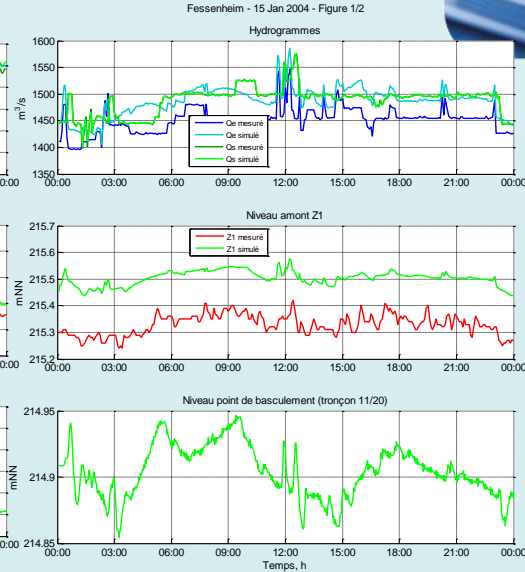
### Débit faible



### Débit moyen

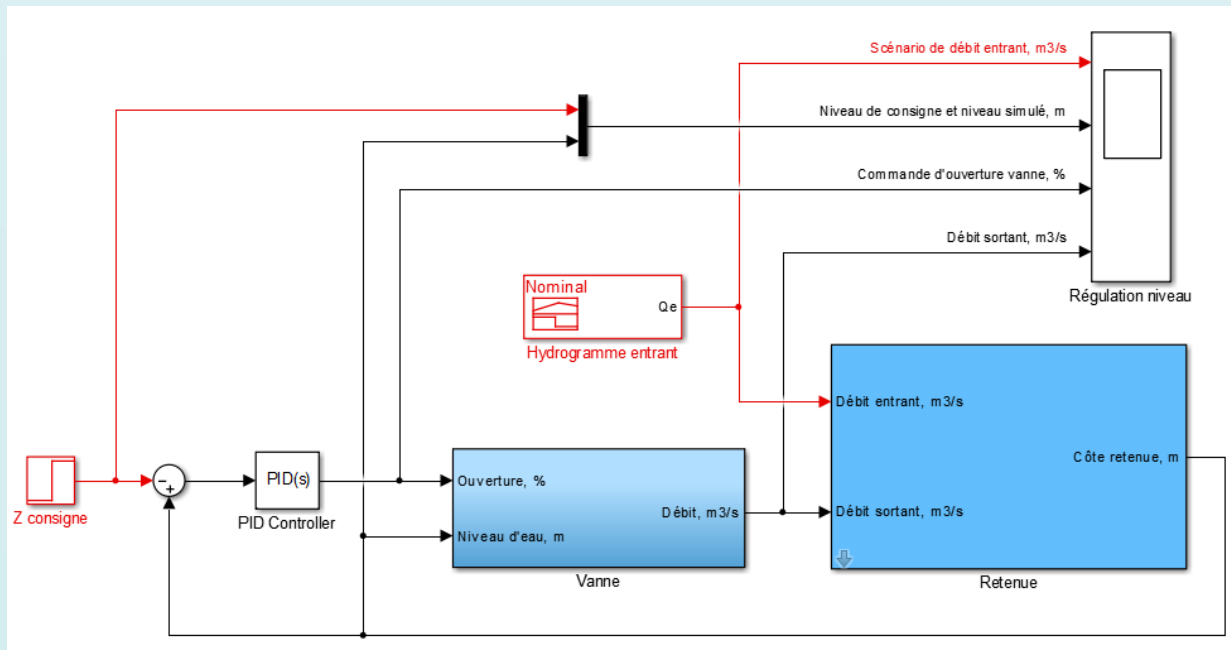


### Débits forts



## 5. Exploitation du modèle

Exemple : conception d'une régulation de niveau



MathWorks®

 [www.captronic.fr](http://www.captronic.fr)

iD<sup>4</sup>for CAR

Moteur d'idées pour véhicules spécifiques et mobilité durable

 **NATIONAL  
INSTRUMENTS™**

 **SHERPA  
ENGINEERING**

 **FAAR®**  
2004 - 2014 *industry*

 **acsystème®**  
analyse & contrôle des systèmes



JESSICA FRANCE - 17, rue des Martyrs - 38054 GRENOBLE

Cedex 09

CAP'TRONIC, un programme financé par :

JESSICA FRANCE fondée par :



**dgcis**

direction générale de la compétitivité  
de l'industrie et des services



**oseo**

