

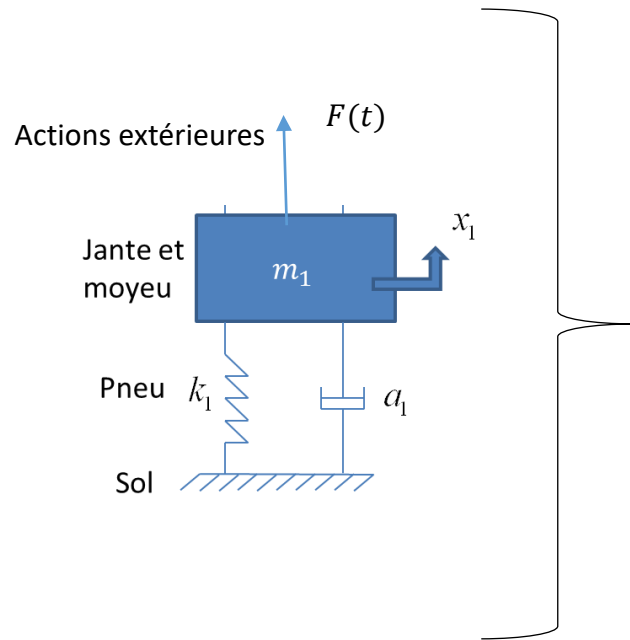
Simulation des systèmes de solides rigides polyarticulés

Simulation causale et acausale

Charles Pontonnier
ENS RENNES



Activité 1

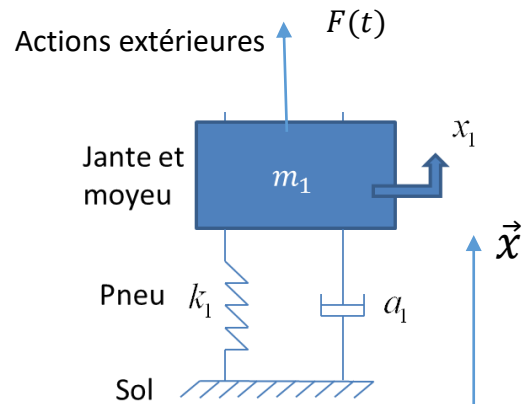


Simuler et récupérer le mouvement

Equation différentielle

L'évolution de ce système est régie par l'équation différentielle suivante

$$m_1 \ddot{x}_1 = F - k_1 x_1 - a_1 \dot{x}_1$$

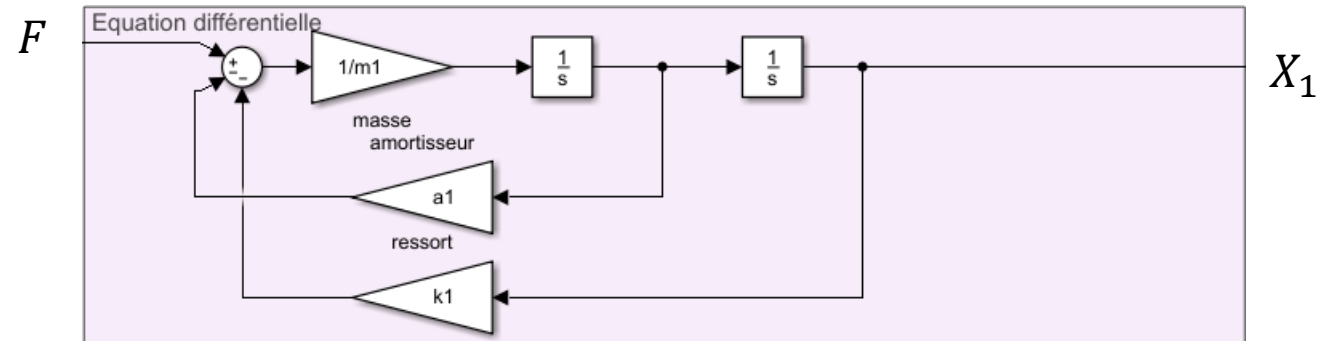
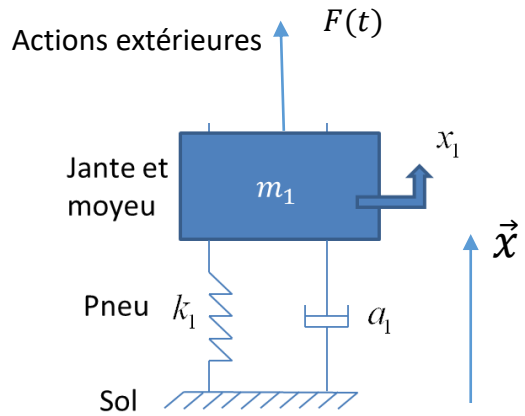


Transformée de Laplace

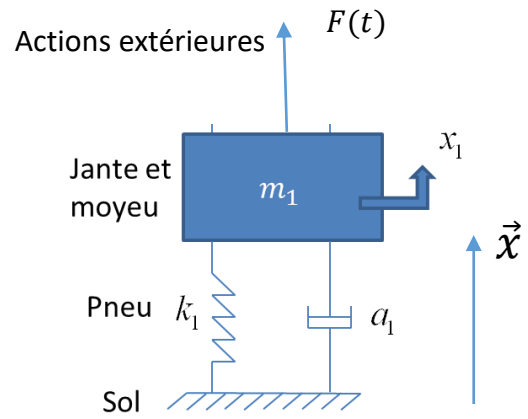
L'évolution de ce système est régie par l'équation différentielle suivante

$$m_1 \ddot{x}_1 = F - k_1 x_1 - a_1 \dot{x}_1$$

$$m_1 p^2 X_1 = F - k_1 X_1 - a_1 p X_1$$



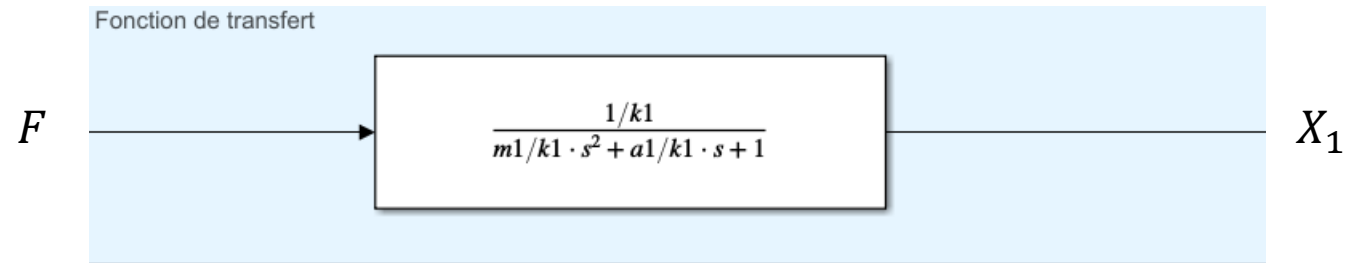
Fonction de transfert



$$m_1 \ddot{x}_1 = F - k_1 x_1 - a_1 \dot{x}_1$$

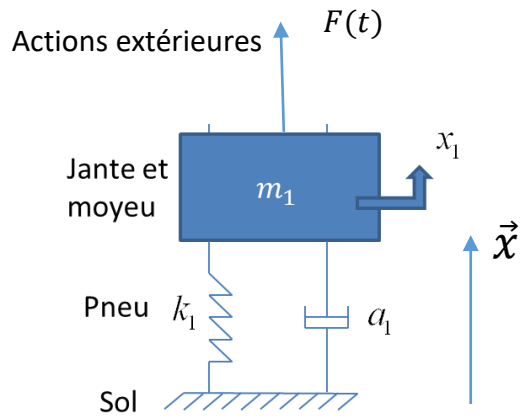
$$m_1 p^2 X_1 = F - k_1 X_1 - a_1 p X_1$$

$$\frac{X_1}{F} = \frac{\frac{1}{k_1}}{\frac{m_1}{k_1} p^2 + \frac{a_1}{k_1} p + 1}$$



Modèle causal

Modèle d'état



$$m_1 \ddot{x}_1 = F - k_1 x_1 - a_1 \dot{x}_1$$

→ Variables d'état (rappel : autant de variables que l'ordre du système)

$$x = [x_1 \ \dot{x}_1]^t$$

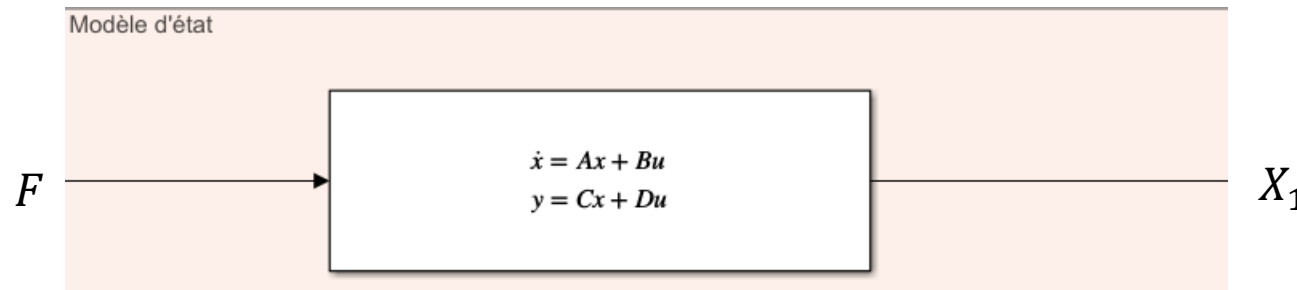
→ Commande

$$u = F(t)$$

→ Modèle d'état → linéaire, chouette !

Sortie : x_1

$$\begin{cases} \begin{bmatrix} \dot{x}_1 \\ \ddot{x}_1 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k_1}{m} & -\frac{a_1}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ \dot{x}_1 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} u \\ y = [1 \ 0] \begin{bmatrix} x_1 \\ \dot{x}_1 \end{bmatrix} \end{cases}$$



Résolution simulation causale



$$\begin{cases} \dot{x} = f(x) + g(x)u \\ y = h(x) \end{cases}$$

x état du système à un instant t ,

u commande de ce système

y sortie observée du système (les variables que l'on veut contrôler en d'autres termes).

Le solveur met en forme le système d'équations en un modèle d'état intégrable

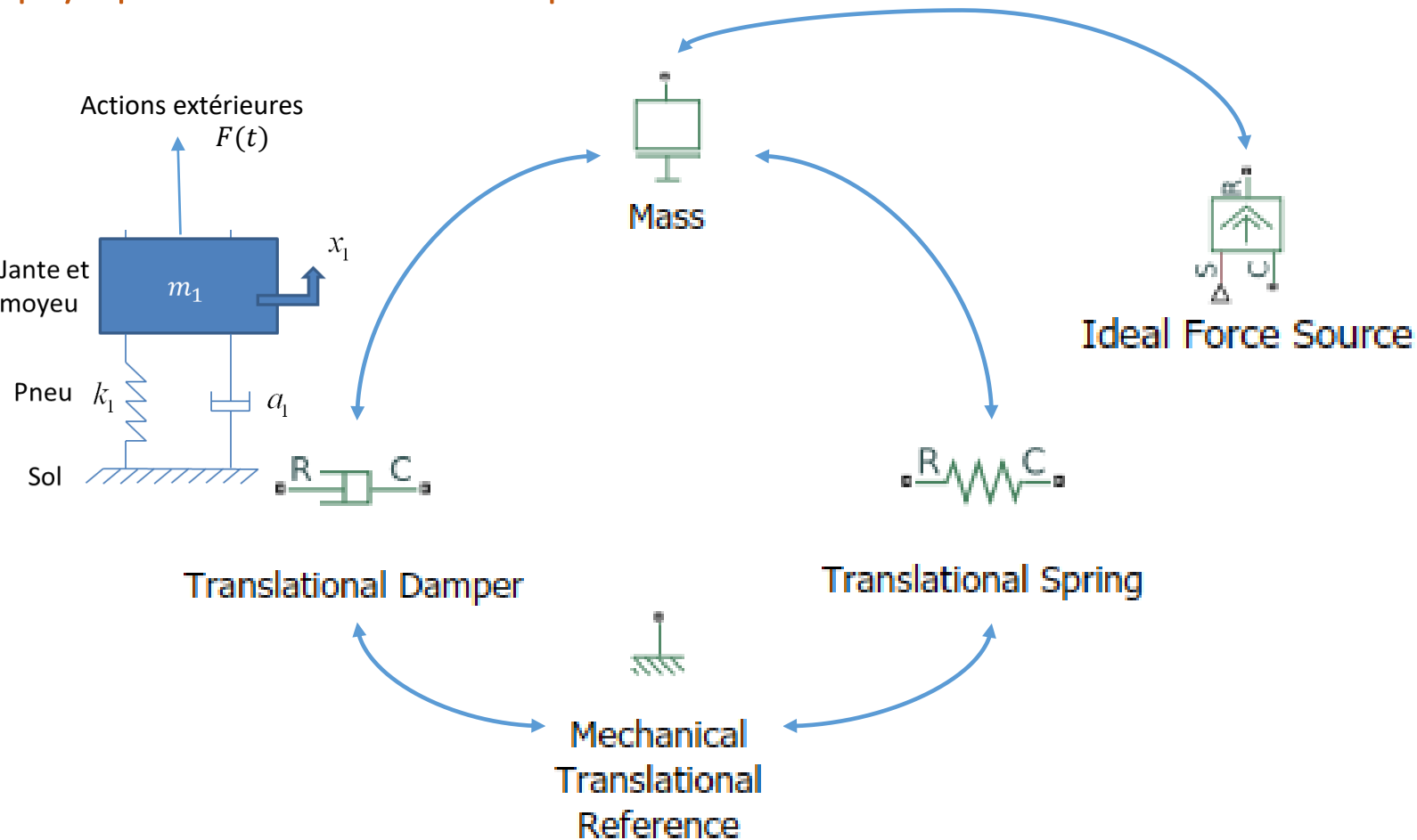


Application d'un schéma temporel (Runge Kutta, Dormand Prince, ...)



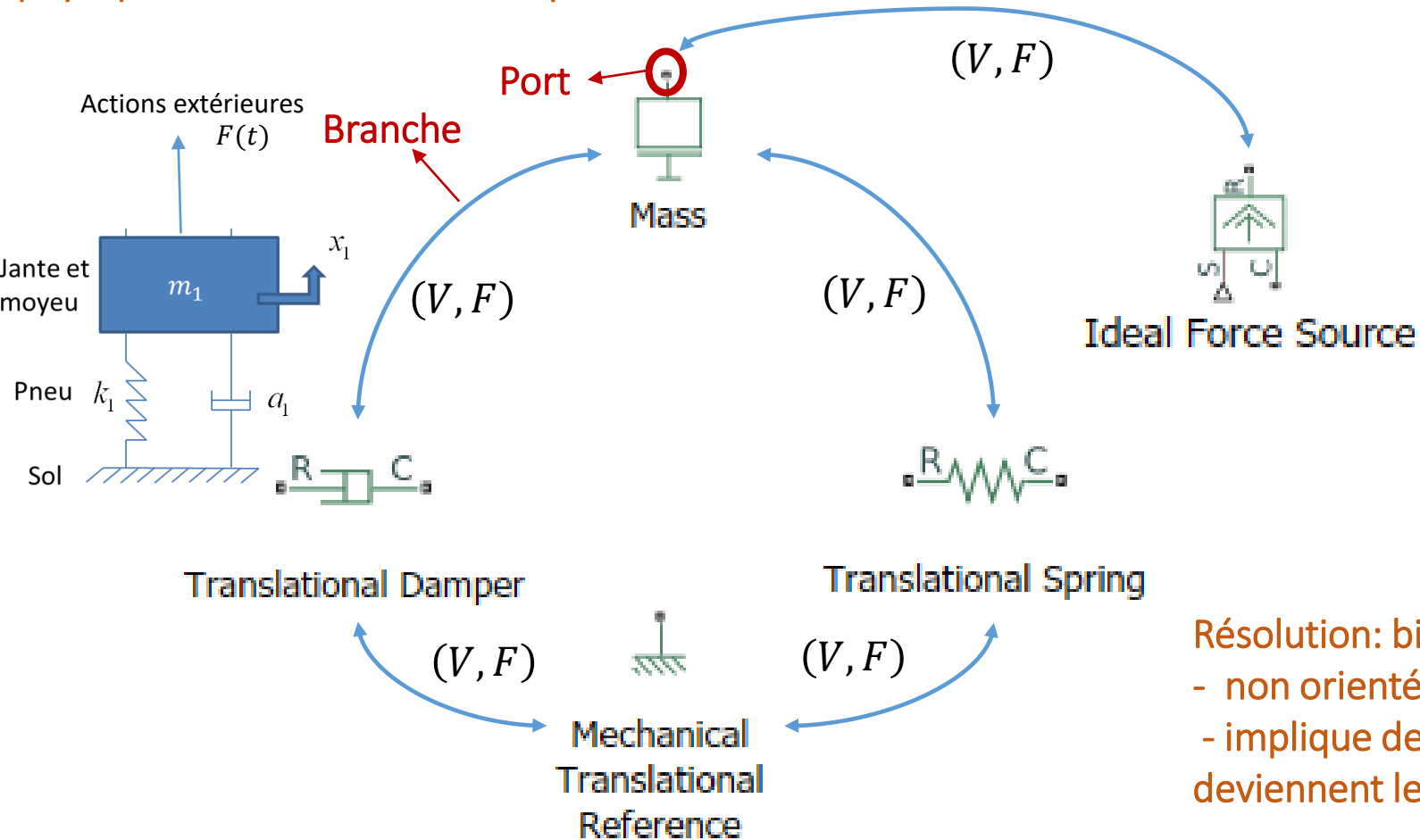
Modèle acausal, simulation acausale

Simuler le comportement de ces systèmes sur la base du **modèle physique de chacun de ses composants**



Modèle acausal, simulation acausale

Simuler le comportement de ces systèmes sur la base du **modèle physique de chacun de ses composants**



Les flux ne sont pas orientés (les relations entrée/sortie sont évaluées à partir des données connues du problème lors de la compilation)

Flux riches « niveau d'abstraction supérieur »

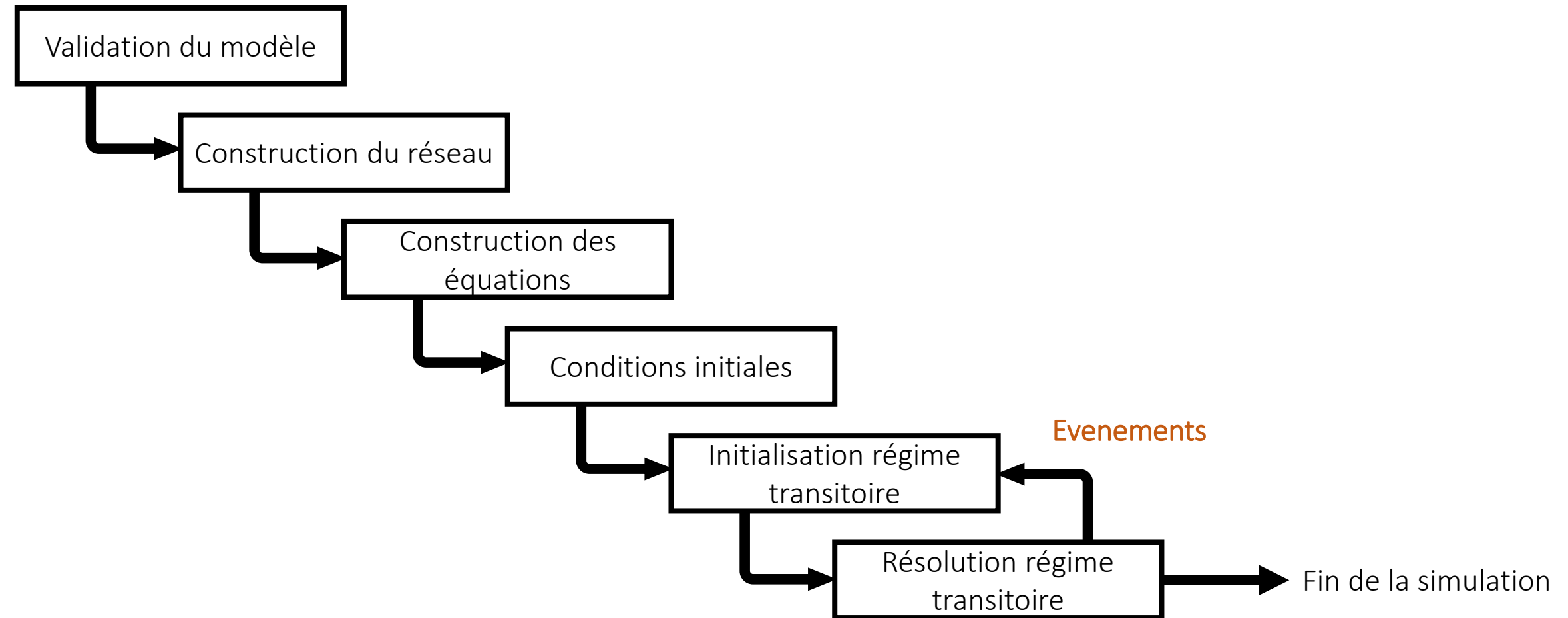
→ Flux (through) et potentiel (across)
 (I, U) (τ, θ) (V, F) (Q, p)

Résolution: bilan de puissance à chaque nœud

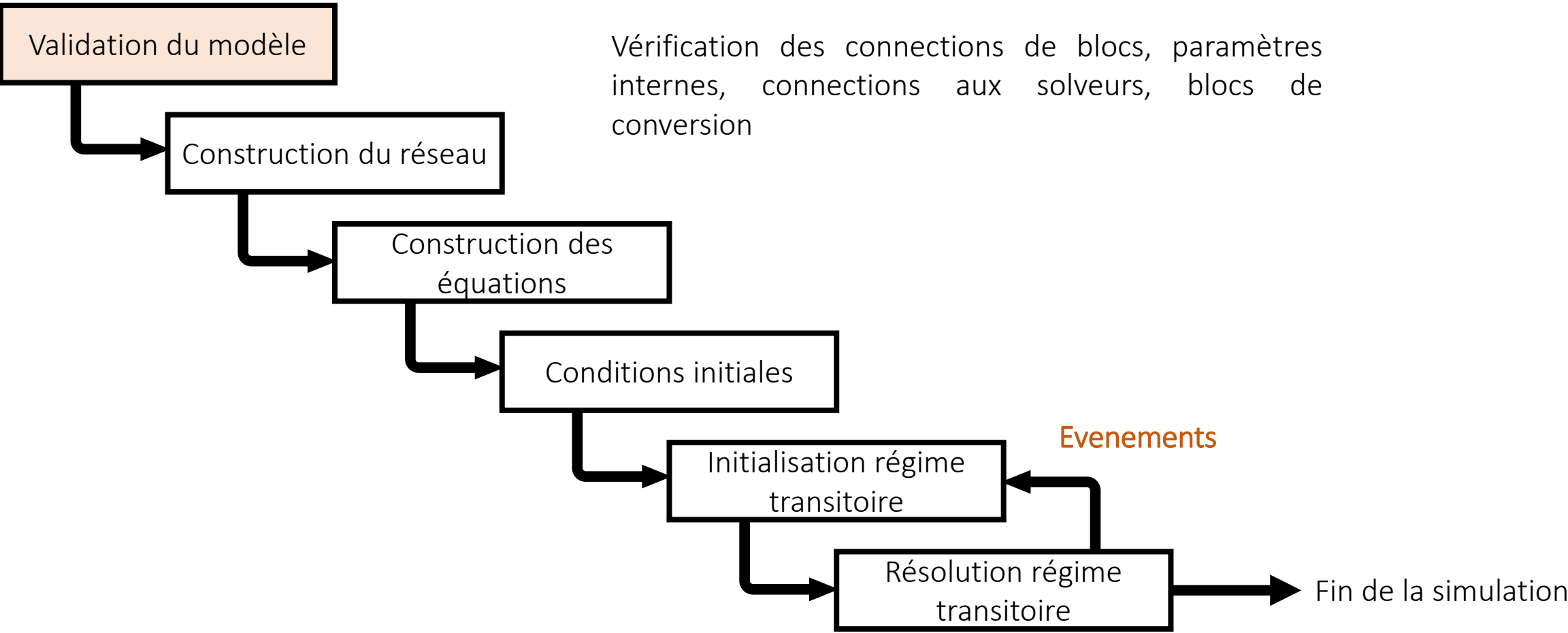
- non orienté

- implique de fixer a priori un certain nombre de variables qui deviennent les entrées du problème

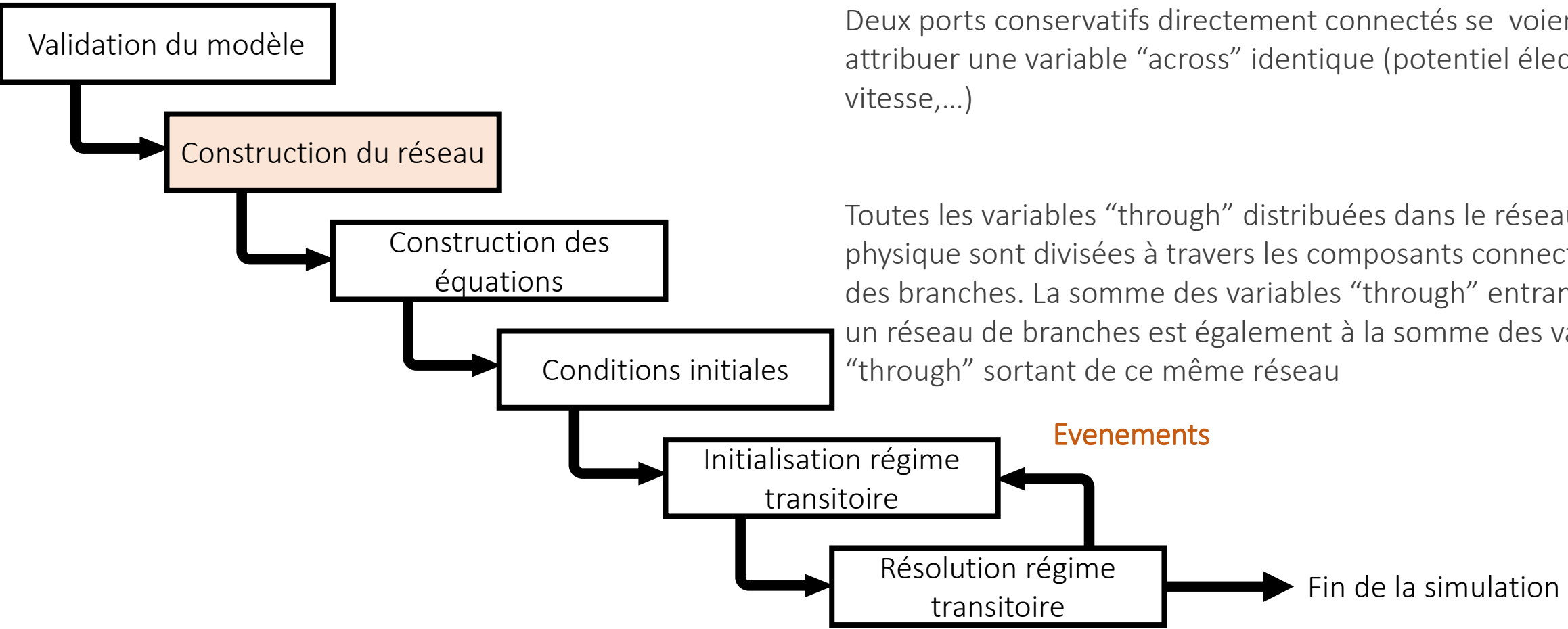
Résolution simulation acausale



Résolution simulation acausale



Résolution simulation acausale

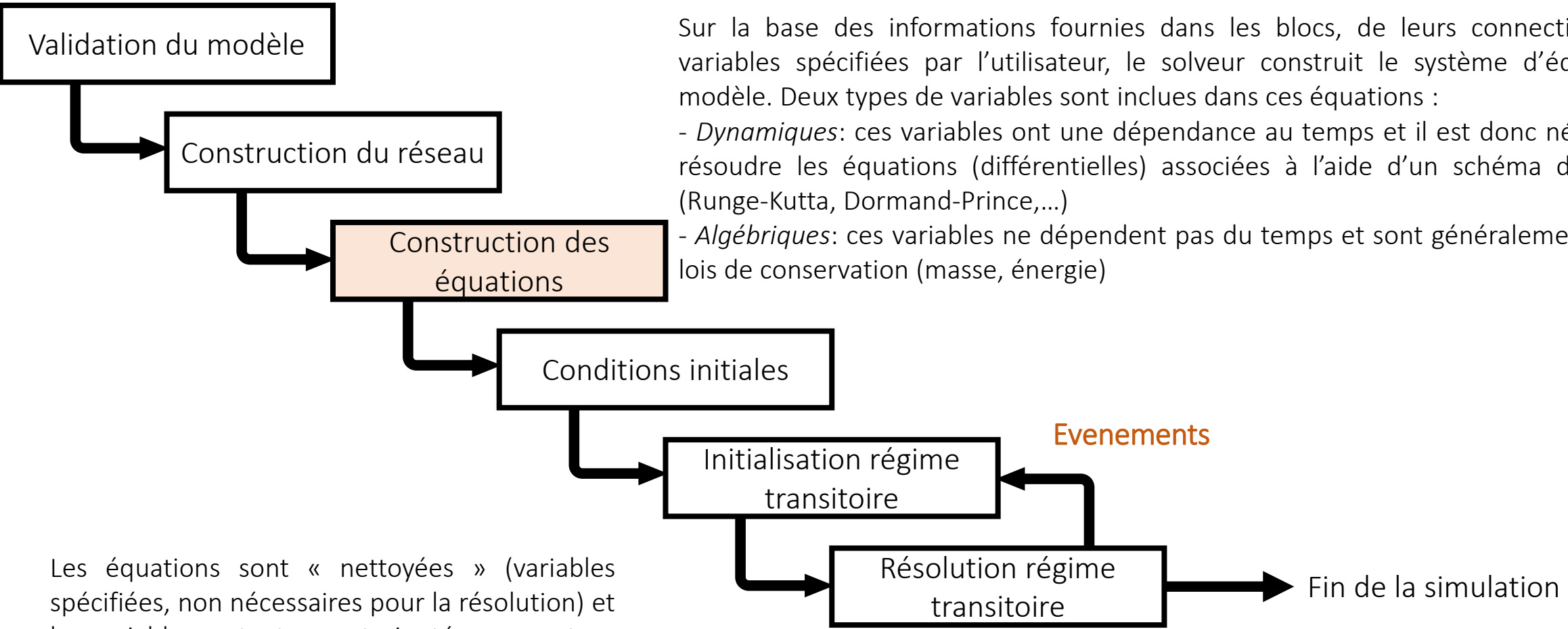


Deux ports conservatifs directement connectés se voient attribuer une variable "across" identique (potentiel électrique, vitesse,...)

Toutes les variables "through" distribuées dans le réseau physique sont divisées à travers les composants connectés par des branches. La somme des variables "through" entrant dans un réseau de branches est également à la somme des variables "through" sortant de ce même réseau



Résolution simulation acausale



Sur la base des informations fournies dans les blocs, de leurs connections et des variables spécifiées par l'utilisateur, le solveur construit le système d'équations du modèle. Deux types de variables sont incluses dans ces équations :

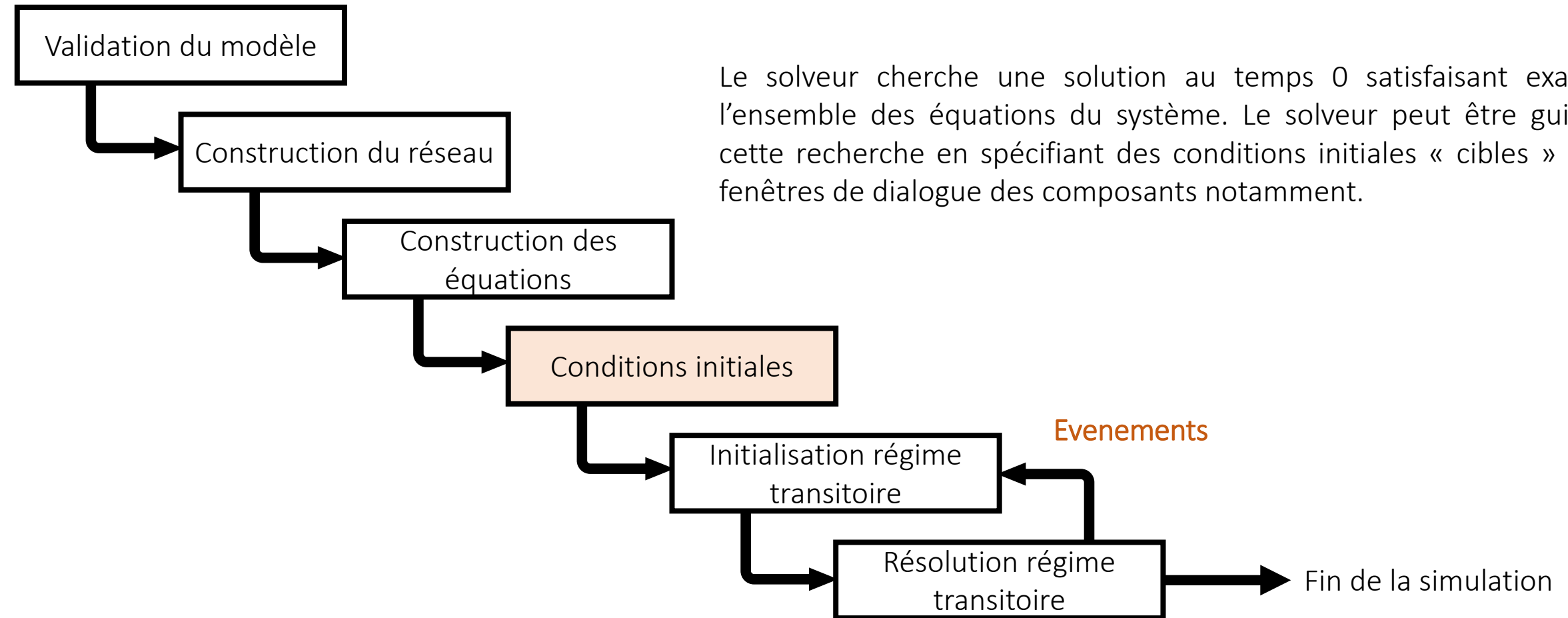
- *Dynamiques*: ces variables ont une dépendance au temps et il est donc nécessaire de résoudre les équations (différentielles) associées à l'aide d'un schéma d'intégration (Runge-Kutta, Dormand-Prince,...)
- *Algébriques*: ces variables ne dépendent pas du temps et sont généralement dues aux lois de conservation (masse, énergie)

Les équations sont « nettoyées » (variables spécifiées, non nécessaires pour la résolution) et les variables restantes sont ajoutées au vecteur d'état global du modèle

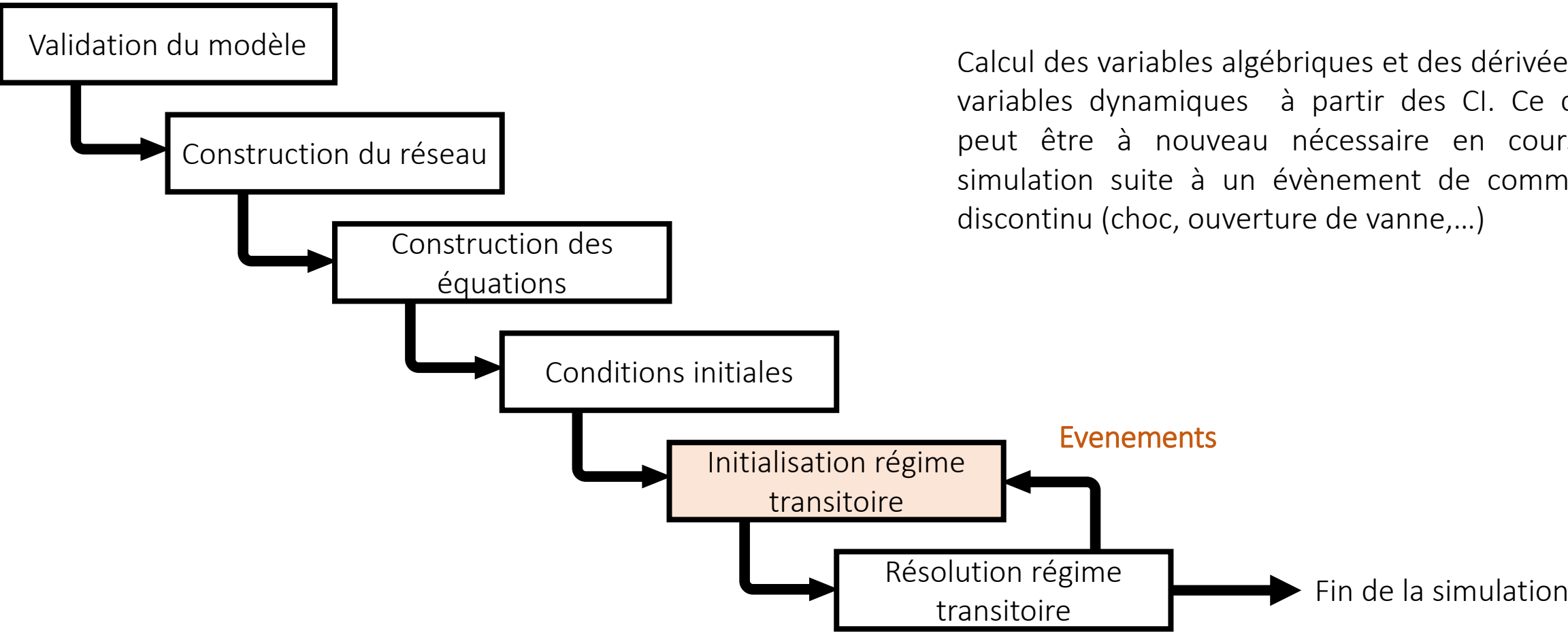


Résolution simulation acausale

Le solveur cherche une solution au temps 0 satisfaisant exactement l'ensemble des équations du système. Le solveur peut être guidé dans cette recherche en spécifiant des conditions initiales « cibles » dans les fenêtres de dialogue des composants notamment.



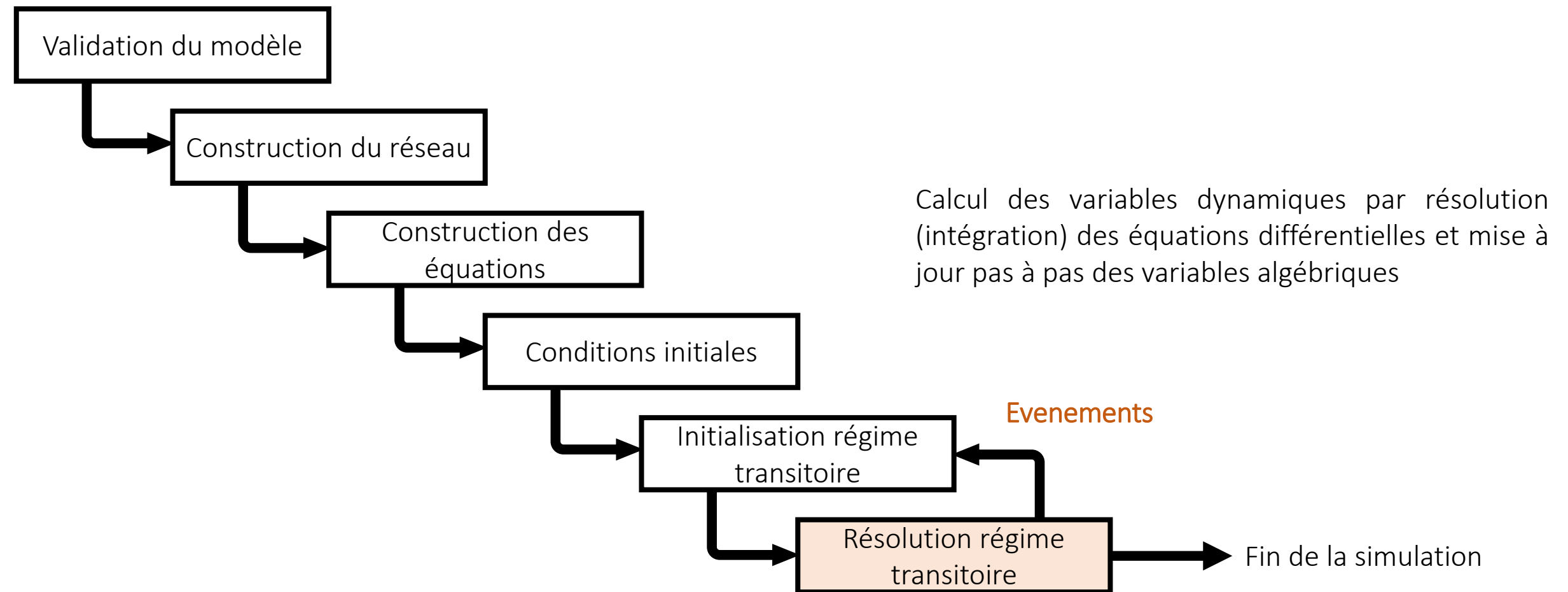
Résolution simulation acausale













Calcul des variables algébriques et des dérivées des variables dynamiques à partir des CI. Ce calcul peut être à nouveau nécessaire en cours de simulation suite à un évènement de commande discontinu (choc, ouverture de vanne,...)



Résolution simulation acausale



Variables « through », variables « across »

Type et couleur du connecteur acausal	Nom et unité du flux Variable(s) « Trough » Capteur série	Nom et unité de l'effort(s)/potentiels(s) Variable(s) « Across » Capteur parallèle
Mécanique (translation) 	Force (N)	Vitesse (m·s ⁻¹) Position (m)
Mécanique (rotation) 	Couple (N·m)	Vitesse (rad·s ⁻¹) Position (rad)
Mécanique (6D)  (v1)  (v2)	3 forces (N) 3 couples (N·m)	3 vitesses linéaires (m·s ⁻¹) 3 positions linéaires (m) 3 vitesses rotations (rad·s ⁻¹) 3 positions rotations (rad)
Electrique 	Courant (A)	Tension (V)
Hydraulique  Pneumatique 	Débit (m ³ ·s ⁻¹)	Pression (Pa)
Magnétique 	Induction (Wb)	Flux (A)
Thermique  Liquide 	Flux thermique (J/s)	Température (°K ou °C)



Causal vs acausal

Causal

- Parfaite maîtrise du niveau de modélisation
- Structure du modèle plus adapté à la mise en place d'une démarche de contrôle commande
- Maîtrise de l'introduction des non linéarités



Acausal



Causal vs acausal

Causal

- Parfaite maîtrise du niveau de modélisation
- Structure du modèle plus adapté à la mise en place d'une démarche de contrôle commande
- Maîtrise de l'introduction des non linéarités



Acausal

- Les grandeurs physiques sont considérées nulles à $t=0$ (conditions de Heaviside)
- Difficile à lire et à modifier
- Nécessite une parfaite connaissance théorique des phénomènes étudiés
- Impose l'utilisation d'outils mathématiques avancés (transformée de Laplace)
- La modélisation des systèmes complexes entraîne de nombreux bouclages dans le modèle ce qui pose des problèmes de résolution numérique pour les solveurs.



Causal vs acausal

Causal

- Parfaite maîtrise du niveau de modélisation
- Structure du modèle plus adapté à la mise en place d'une démarche de contrôle commande
- Maîtrise de l'introduction des non linéarités



Acausal

- Très proche de la structure du système réel
- Facile à modifier, facile à lire
- Les conditions initiales peuvent être choisies quelconques
- Toutes les grandeurs physiques sont accessibles et mesurables au sein du modèle
- Pas d'équation à écrire
- Les connexions entre les composants transmettent un niveau d'information plus riche
- Aucun outil mathématique nécessaire
- Reconnaissance immédiate du domaine

- Les grandeurs physiques sont considérées nulles à $t=0$ (conditions de Heaviside)
- Difficile à lire et à modifier
- Nécessite une parfaite connaissance théorique des phénomènes étudiés
- Impose l'utilisation d'outils mathématiques avancés (transformée de Laplace)
- La modélisation des systèmes complexes entraîne de nombreux bouclages dans le modèle ce qui pose des problèmes de résolution numérique pour les solveurs.



Causal vs acausal

Causal

- Parfaite maîtrise du niveau de modélisation
- Structure du modèle plus adapté à la mise en place d'une démarche de contrôle commande
- Maîtrise de l'introduction des non linéarités



Acausal

- Très proche de la structure du système réel
- Facile à modifier, facile à lire
- Les conditions initiales peuvent être choisies quelconques
- Toutes les grandeurs physiques sont accessibles et mesurables au sein du modèle
- Pas d'équation à écrire
- Les connexions entre les composants transmettent un niveau d'information plus riche
- Aucun outil mathématique nécessaire
- Reconnaissance immédiate du domaine

- Les grandeurs physiques sont considérées nulles à $t=0$ (conditions de Heaviside)
- Difficile à lire et à modifier
- Nécessite une parfaite connaissance théorique des phénomènes étudiés
- Impose l'utilisation d'outils mathématiques avancés (transformée de Laplace)
- La modélisation des systèmes complexes entraîne de nombreux bouclages dans le modèle ce qui pose des problèmes de résolution numérique pour les solveurs.

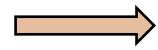


- Peu de maîtrise concernant le niveau de modélisation des composants utilisés
- Les modèles obtenus sont fortement non linéaires ce qui rend la démarche de contrôle commande délicate



Solutions nombreuses

- Matlab/Simscape
- Scilab/Xcos
- MapleSim
- Dymola
- ...



On va utiliser principalement Simscape/multibody



Ressource: voir ebook de Ivan Liebgott



Simscape Multibody



- Simulateur très répandu dans le domaine académique et industriel
- Bonne interface utilisateur
- Bien documenté
- Bien couplé avec Matlab
- **Simulation de systèmes discrets et continus**
- **Nombreux intégrateurs d'équations différentielles (notamment à pas variable)**
- Programmable
- Création de bibliothèques
- Hiérarchisation des modèles
- Décomposition des modèles



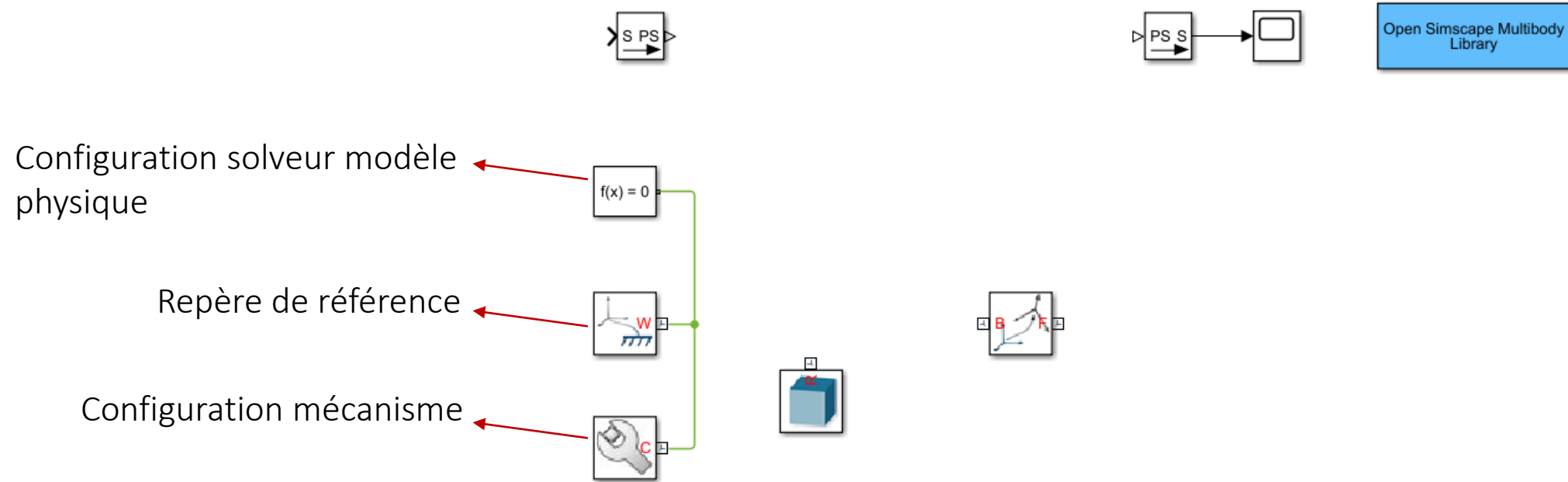
- Cher
- Lent
- Conçu pour des signaux unidirectionnels

Multibody

- Multibody est une toolbox Simulink faisant partie d'un package « Simscape »
- Cette toolbox permet de simuler des systèmes mécaniques actionnés ouverts ou fermés
- Elle permet également de récupérer des informations mécaniques (forces, vitesses, accélérations)

Un nouveau système

- Utiliser la commande « smnew »
- Mettre en liaison pivot un solide / à la référence



Configuration

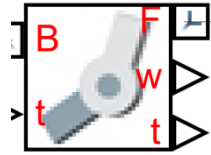
1. Configurer l'environnement (gravité pour le moment)
2. Configurer le solveur
3. Configurer les paramètres de masse et d'inertie ainsi que la géométrie du corps rigide
4. Eventuellement configurer les paramètres de simulation

Configuration

Je vous conseille vivement l'utilisation d'un « batch » pour paramétrer vos modèles

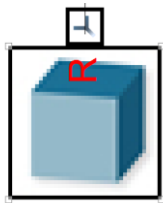
```
Documents and Settings\cpontonn\Bureau\Cours ENS\Approche système mécatro\IP\batch_IP1.m
1 %% Paramètres convoyeur palettes
2
3 n=21;
4 Jm=20.5*10^-4;
5 Jr=3*10^-4;
6 Jp=0.02674;
7 Mcp=350;
8 r=85*10^-3;
9 X=[0 0.068 0.068+0.663 0.8 0.8+6.5 7.3+0.022 7.3+0.022+2.256 7.3+0.022+2.256+0.022 7.3+0.
10 Y=[0 337.7 337.7 0 0 -108.5 -108.5 0 0];
11 Crete=53.1;
12 PID=[10 0 0];
13 PID2=[20 5 0];
14
```

Quelques boites de base

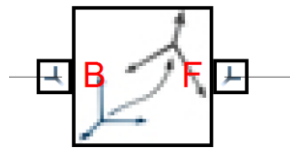


Moteur

Revolute joint → attention toutes les liaisons à 1ddl sont orientées selon z



Solid Solide → inertie, masse, géométrie et repère de référence

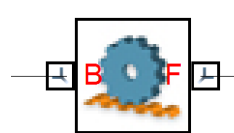


Rigid Transform

Transformation rigide → rotation et/ou translation des repères



Les transfos rigides appartiennent à une classe cinématique donnée



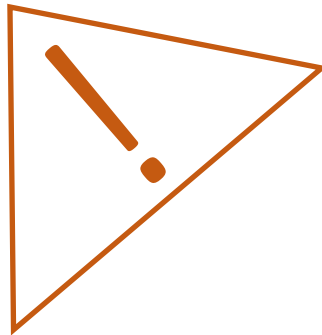
Rack and Pinion
Constraint

Contrainte cinématique → réduit les ddl...ATTENTION A LA CONFIGURATION DELICATE

Export d'un assemblage sous Solidworks

Add-on permettant d'exporter un assemblage solidworks avec ses propriétés vers MultiBody

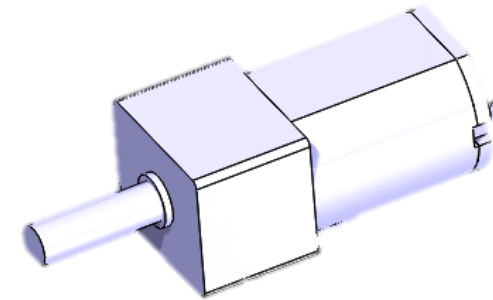
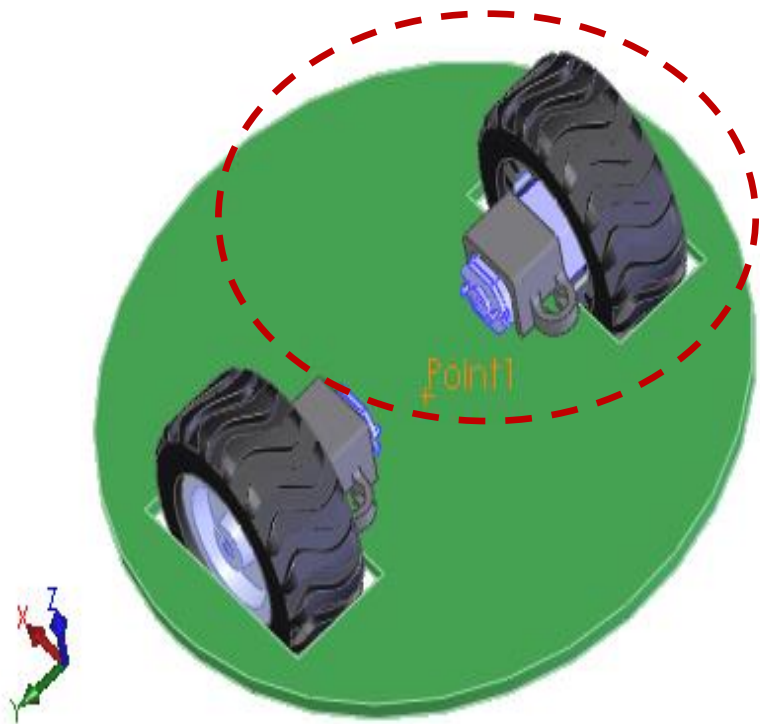
- Masses et inerties correctement exprimées
- Mobilités correctement définies
- Fermetures cinématiques respectées



Il y a des règles d'assemblage à respecter

Règle n°1: contraintes simples

- Privilégier les contraintes facilement traduisibles: coaxiale, coïncidence de plans, ...
- Eviter les contraintes complexes: distance, tangence, coïncidence de surfaces complexes,...



Liaison pivot: coaxiale + appui plan

Règle n°2: définir origines et orientations pour configuration initiale

- Libérer la base de la chaîne cinématique
- Contraindre la base de manière à confondre son origine et l'origine de l'assemblage
- L'orienter de manière à confondre ses axes principaux avec les axes d'assemblage
- Sur-contraindre l'assemblage afin de positionner en configuration initiale les différents solides du mécanisme (dépend de l'application visée)
- L'assemblage doit être totalement contraint à l'issue de cette manipulation

Sous Multibody

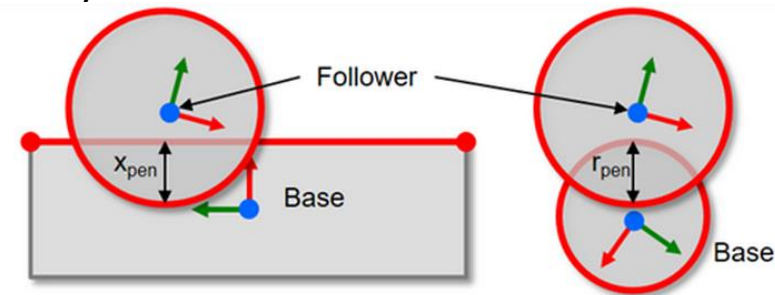
- Pas de gestion des contacts « par défaut »
- Librairie supplémentaire « Simscape Multibody Contact forces Library »
- « Trick » numérique: méthode de pénalité (équivalent à un « proportionnel dérivé »)
- Le contact est supposé toujours établi

Linéaire: réaction normale, **proportionnelle (pénétration) dérivée (vitesse de pénétration)**

$$f_c = kx_{pen} + b\dot{x}_{pen}$$

Non-linéaire: la raideur augmente **exponentiellement** avec la pénétration, l'amortissement augmente **linéairement avec la vitesse** de pénétration et **polynomialement avec la pénétration**

$$f_c = kx_{pen}^e + b\dot{x}_{pen}f(x_{pen}, x_{penmax})$$



Contact Stiffness (k): Spring stiffness for force law

Contact Damping (b): Damping constant for force law

Linear

A linear spring-damper resists penetration. Damping force is 0 as penetration decreases. Force is applied only along the direction of penetration.

Circle to Line		Circle to Circle, Ring	
$F_x = \begin{cases} k*x_{pen} + b*v_{pen} & x_{pen} > 0, v_{pen} > 0 \\ k*x_{pen} & x_{pen} > 0, v_{pen} < 0 \\ 0 & x_{pen} \leq 0 \end{cases}$	$F_y = 0$	$F_r = \begin{cases} k*x_{pen} + b*v_{pen} & r_{pen} > 0, v_{pen} > 0 \\ k*x_{pen} & r_{pen} > 0, v_{pen} < 0 \\ F_r = 0 & x_{pen} \leq 0 \end{cases}$	

Nonlinear

Same as Linear except:

1. Stiffness force increases exponentially with penetration
 $k*(x_{pen})^e$, where e is parameter **Penetration Exponent**
2. Damping force increases gradually during initial penetration
 $b*v_{pen}*(\text{Smooth Step})$, where Smooth Step is a polynomial whose value increases from 0 to 1 as the penetration increases from 0 to parameter **Penetration for Full Damping**