

---

# Présentation du solveur SPARK (qui sert de base à la plateforme de simulation SimSpark)



---

Journée INES-LAMA  
[pierre.tittlein@univ-savoie.fr](mailto:pierre.tittlein@univ-savoie.fr)

---

# Sommaire

Je vais vous présenter:

- ❑ A quoi il sert
- ❑ Comment il fonctionne
- ❑ Comment on l'utilise

---

# SPARK: à quoi ça sert

- SPARK sert à résoudre des équations algébro-différentielles et discrètes.
- Exemple d'applications:
  - Modèle zonal (thermo-aérodynamique + humidité)
  - Modèles de systèmes (clim solaire...)
  - Possibilité de programmer avec SPARK dans Energy +

A quoi il sert?

---

# SPARK

- ❑ A quoi il sert
- ❑ Comment il fonctionne
- ❑ Comment on l'utilise

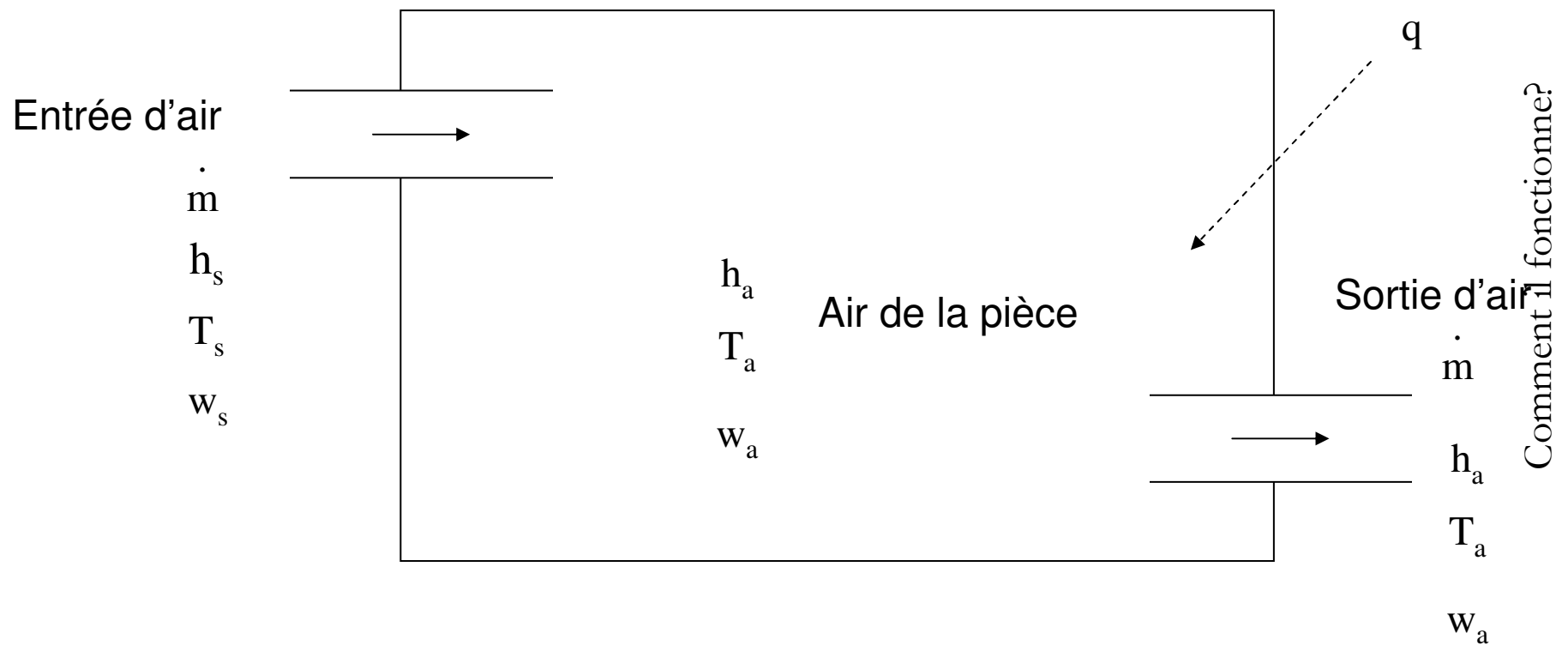
# Principales caractéristiques

- Description du problème orienté objet. L'objet de base est l'équation.
- Il se sert de la méthode des graphes pour réduire le système d'équation à résoudre en composants forts.
- Il possède une dizaine de méthode d'inversion testées en cascade.
- Le pas de temps de calcul est variable.

Comment il fonctionne?

# Comment il fonctionne???

Exemple simple (ex1)



# Équations disponibles:

## ❖ Equation 1: bilan d'énergie dans la pièce

$$\dot{m} \cdot h_s + q - \dot{m} \cdot h_a = 0$$

## ❖ Equation 2: enthalpie dans la pièce

$$h_a = a \cdot T_a + w_a \cdot (b + c \cdot T_a)$$

## ❖ Equation 3: enthalpie de l'air qui entre

$$h_s = a \cdot T_s + w_s \cdot (b + c \cdot T_s)$$

8 variables connues

$$\dot{m}, q, a, b, c, w_a, w_s, T_s$$

3 variables inconnues

$$h_s, h_a, T_a$$

→ Le problème est bien posé: 3 équations-3 inconnues

Comment il fonctionne?

# Comment on ferait à la main?

**Equation 3**

$$h_s = a \cdot T_s + w_s \cdot (b + c \cdot T_s)$$

**Equation 1**

$$h_a = h_s + \frac{q}{\dot{m}}$$

**Equation 2**

$$T_a = \frac{h_a - w_a \cdot b}{a + w_a \cdot c}$$

Comment il fonctionne?



# Comment fait SPARK? (ex1)

*On lui donne:*

**Equation 1**

$$\begin{cases} h_a = h_s + \frac{q}{\dot{m}} \\ h_s = h_a - \frac{q}{\dot{m}} \end{cases}$$

**Equation 2**

$$\begin{cases} T_a = \frac{h_a - w_a \cdot b}{a + w_a \cdot c} \\ h_a = a \cdot T_a + w_a \cdot (b + c \cdot T_a) \end{cases}$$

**Equation 3**

$$h_s = a \cdot T_s + w_s \cdot (b + c \cdot T_s)$$

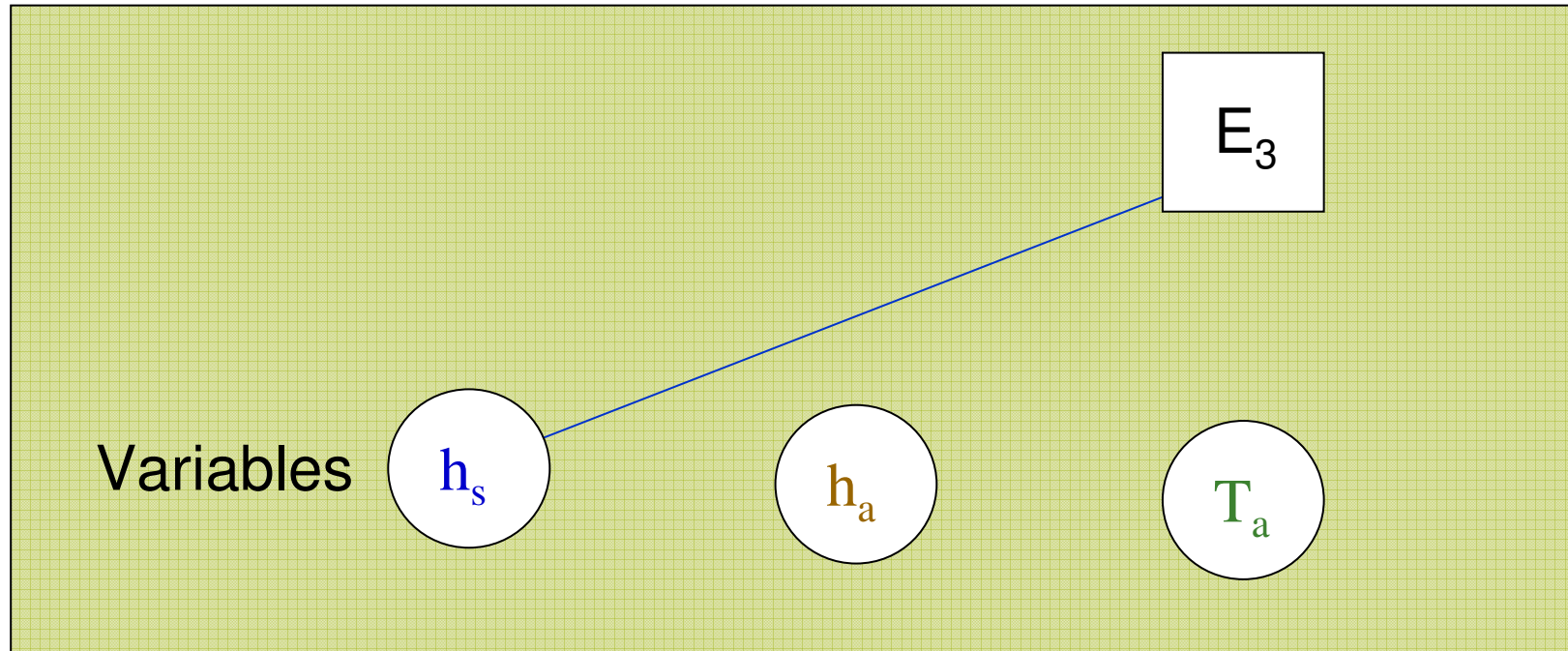
Comment il fonctionne?

# Comment fait SPARK? (ex1)

*Il utilise alors la méthode des graphes pour trouver la meilleure séquence de résolution :*

**1 inverse**

$$h_s = a \cdot T_s + w_s \cdot (b + c \cdot T_s)$$

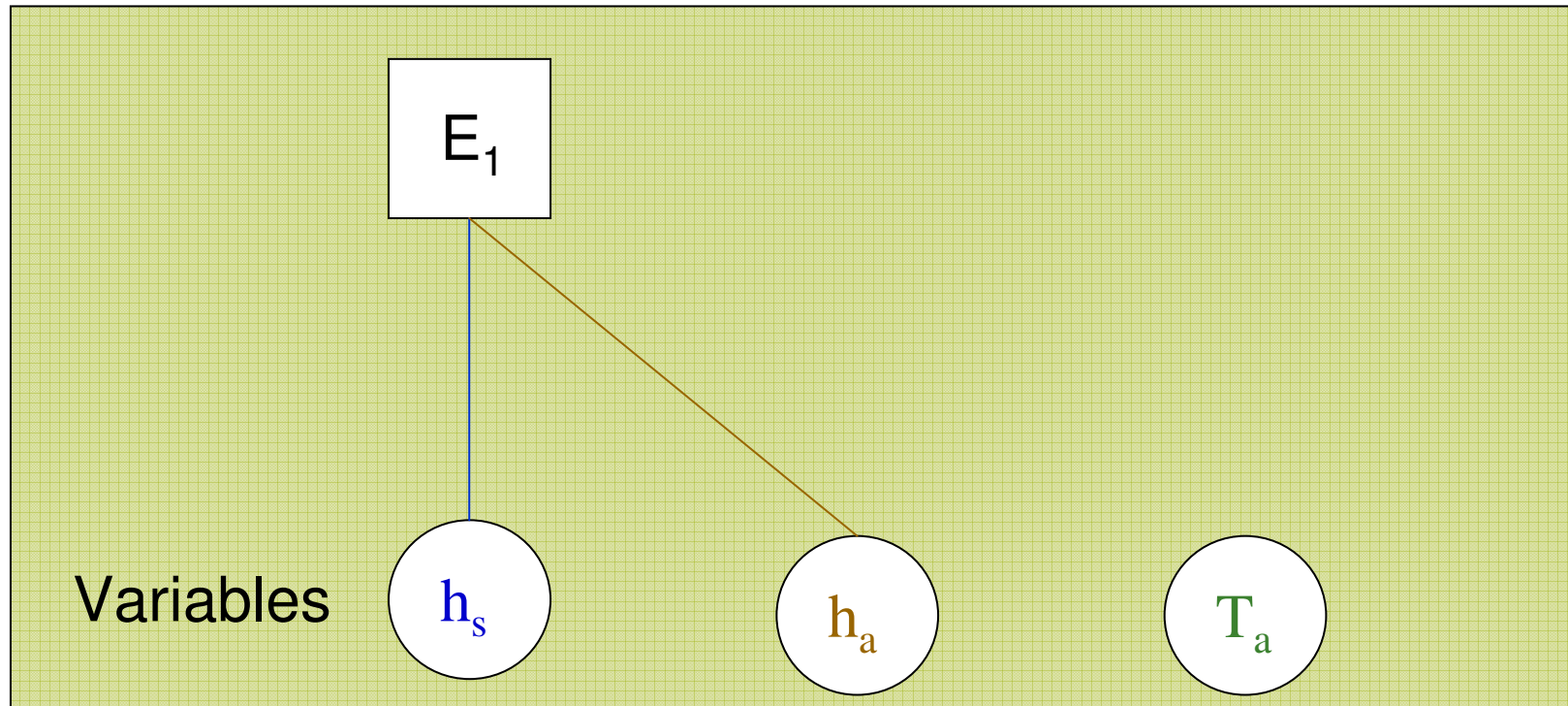


Comment il fonctionne?

# Comment fait SPARK? (ex1)

2 inverses

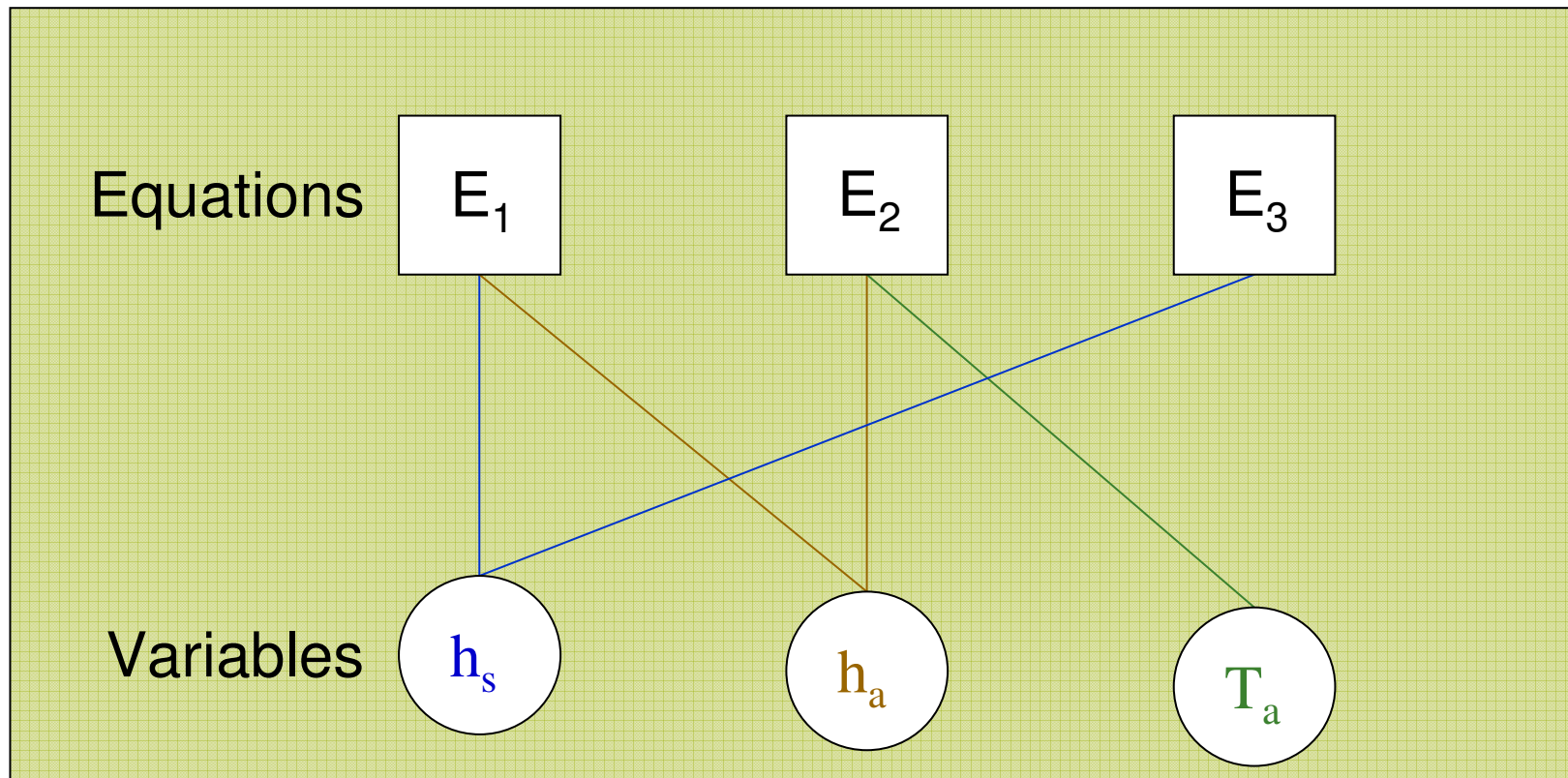
$$\begin{cases} h_a = h_s + \frac{q}{\dot{m}} \\ h_s = h_a - \frac{q}{\dot{m}} \end{cases}$$



Comment il fonctionne?

# Comment fait SPARK? (ex1)

Il “dessine” tous les inverses disponibles dans un graphe

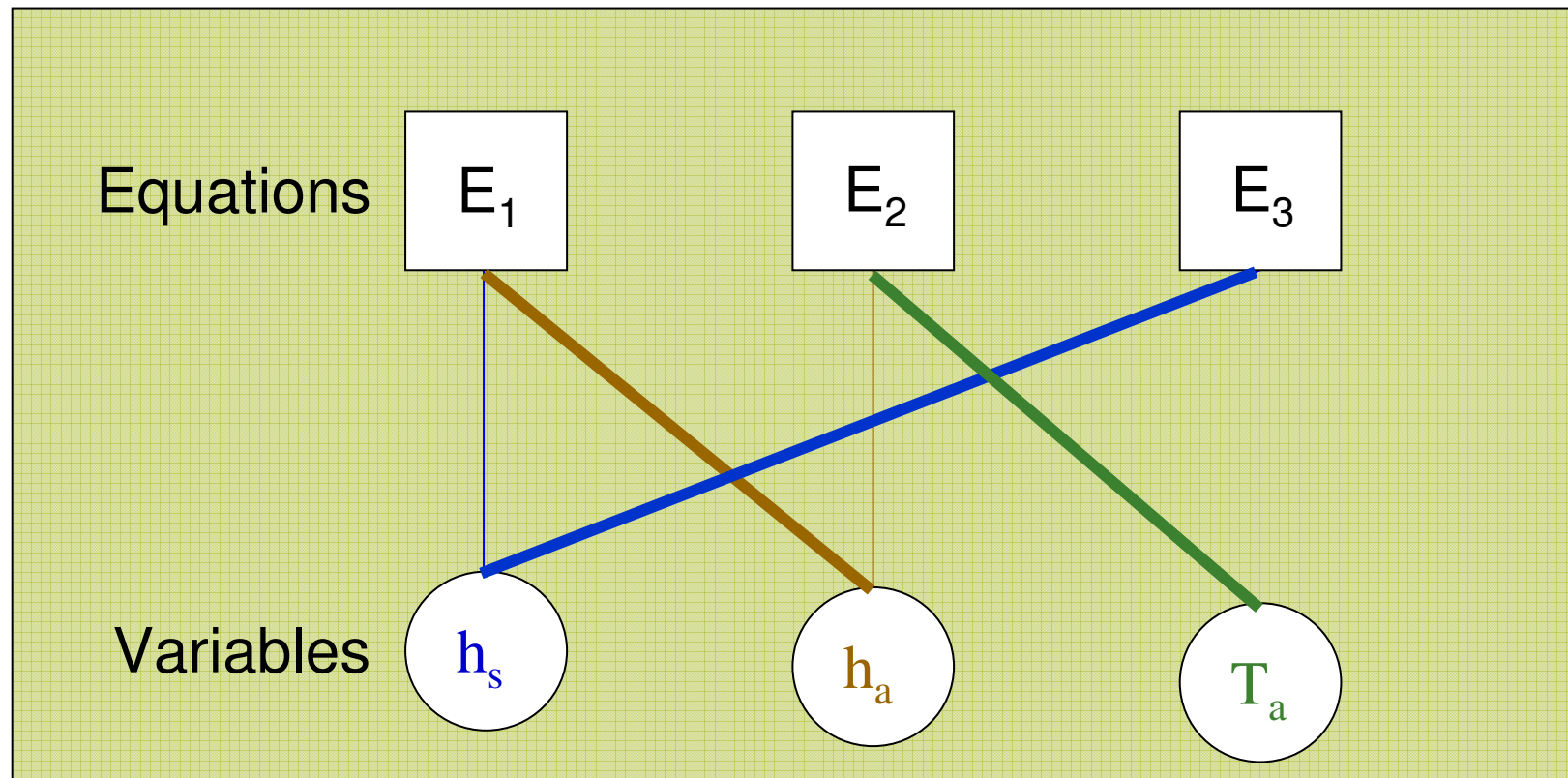


Comment il fonctionne?

# Comment fait SPARK? (ex1)

BUT

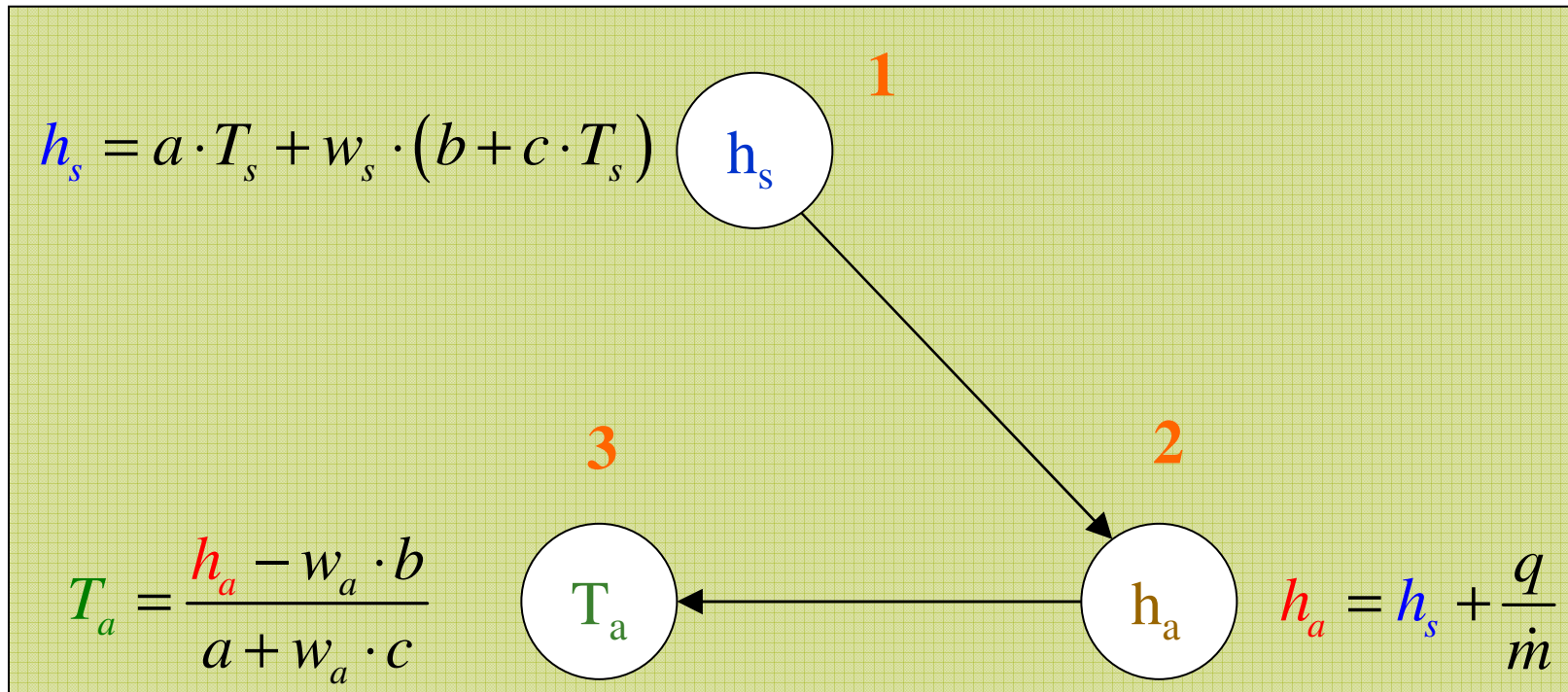
Sélectionner les bons inverses pour calculer les inconnues



Comment il fonctionne?

# Comment fait SPARK? (ex1)

Il construit alors le graphe de calcul



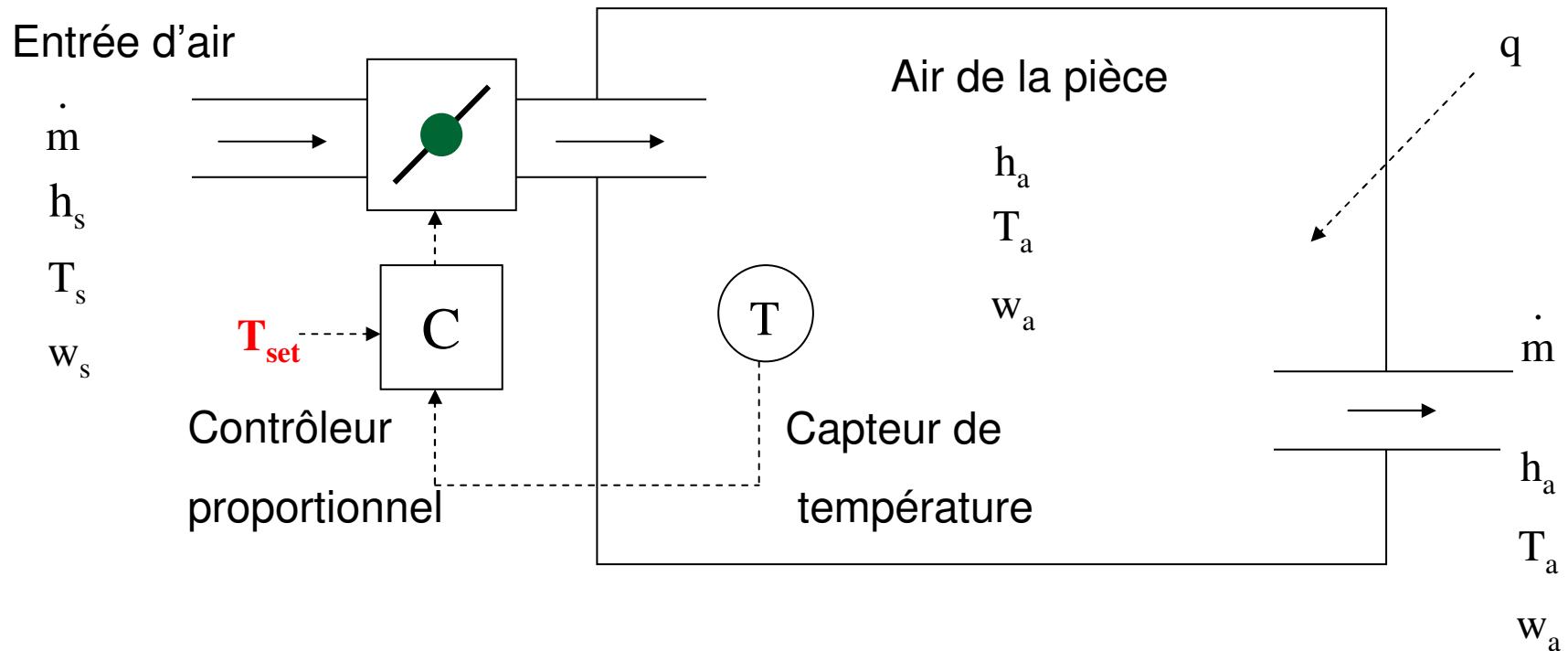
Comment il fonctionne?

Aucune itération, aucune matrice à inverser...

Il fait comme on ferait à la main

# Autre exemple (ex2)

## Schéma du modèle



Comment il fonctionne?

# Équations disponibles

- ❖ **Équation 1: bilan d'énergie dans la pièce**

$$\dot{m} \cdot h_s + q - \dot{m} \cdot h_a = 0$$

- ❖ **Équation 2: enthalpie de la pièce**

$$h_a = a \cdot T_a + w_a \cdot (b + c \cdot T_a)$$

- ❖ **Équation 3: enthalpie de l'air qui entre**

$$h_s = a \cdot T_s + w_s \cdot (b + c \cdot T_s)$$

- ❖ **Équation 4: contrôleur proportionnel**

$$\dot{m} = \max(0, k_p (T_a - T_{set}))$$

9 variables connues

$$q, a, b, c, w_a, w_s, T_s, T_{set}, k_p$$

4 variables inconnues

$$\dot{m}, h_s, h_a, T_a$$

→ Le problème est bien posé: 4 équations for 4 inconnues

Comment il fonctionne?



# La méthode brutale...

## Méthode de Newton

$$x = (h_s, h_a, T_a, \dot{m})^T$$

$$F(x) = \begin{bmatrix} F_1(x) \\ F_2(x) \\ F_3(x) \\ F_4(x) \end{bmatrix} \quad \leftarrow \text{Fonctions résiduelles}$$

$$x^{k+1} = x^k - J^{-1}(x^k) \cdot F(x^k)$$

Condition initiale:  $x_0$

$F_4(x)$   
Ne dépend pas de  $h_s$

## Jacobian matrix

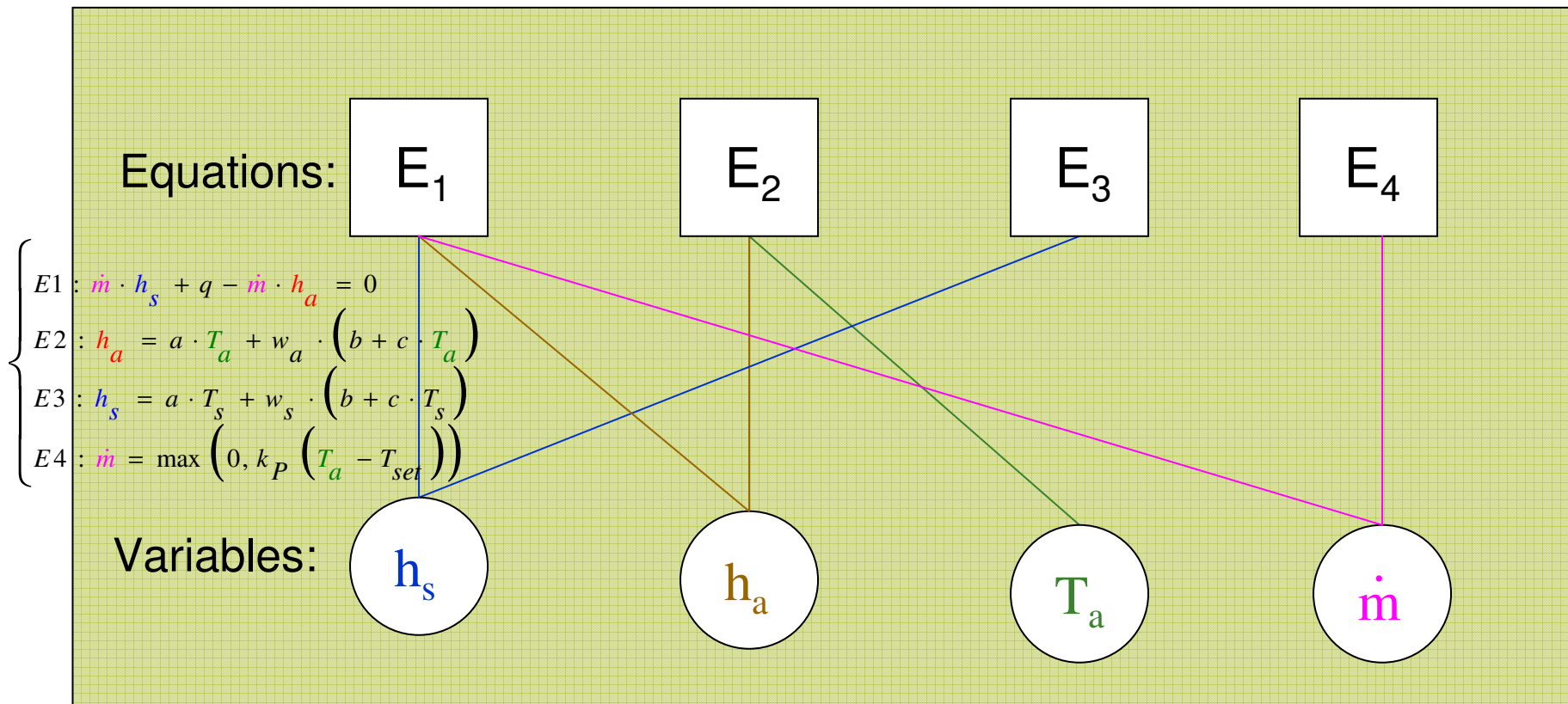
$$J = \begin{bmatrix} \frac{\partial F_1}{\partial x_1} & \frac{\partial F_1}{\partial x_2} & 0 & \frac{\partial F_1}{\partial x_4} \\ 0 & \frac{\partial F_2}{\partial x_2} & \frac{\partial F_2}{\partial x_3} & 0 \\ \frac{\partial F_3}{\partial x_1} & 0 & 0 & 0 \\ 0 & 0 & \frac{\partial F_4}{\partial x_3} & \frac{\partial F_4}{\partial x_4} \end{bmatrix} \quad \text{Comment il fonctionne?}$$

Coût:  $O(n^3) = O(4^3)$

- approche conventionnelle par inversion de matrice
- on résout un système 4x4 d'équations non linéaires

# Comment fait SPARK? (ex2)

Il “dessine” tous les inverses disponibles dans un graphe

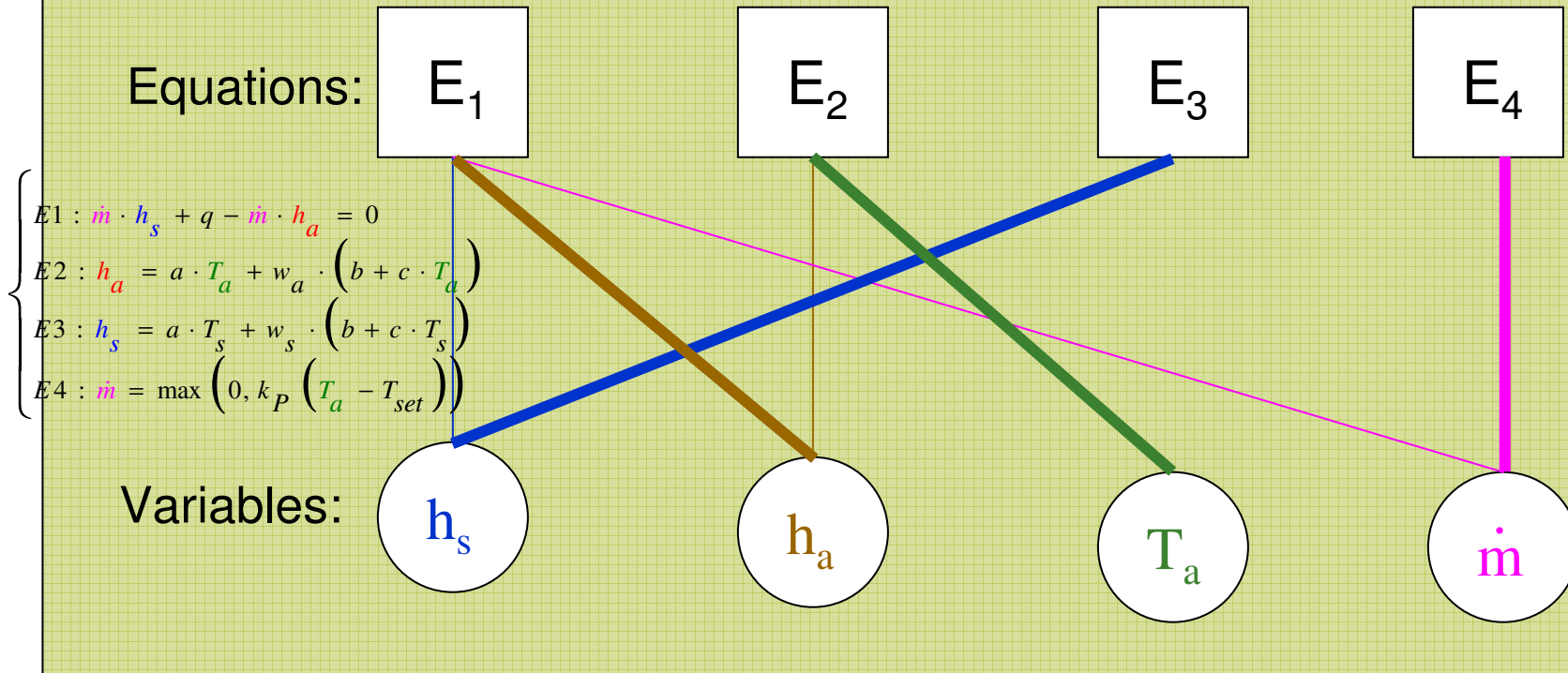


# Comment fait SPARK? (ex2)

BUT

Sélectionner les bons inverses  
pour calculer les

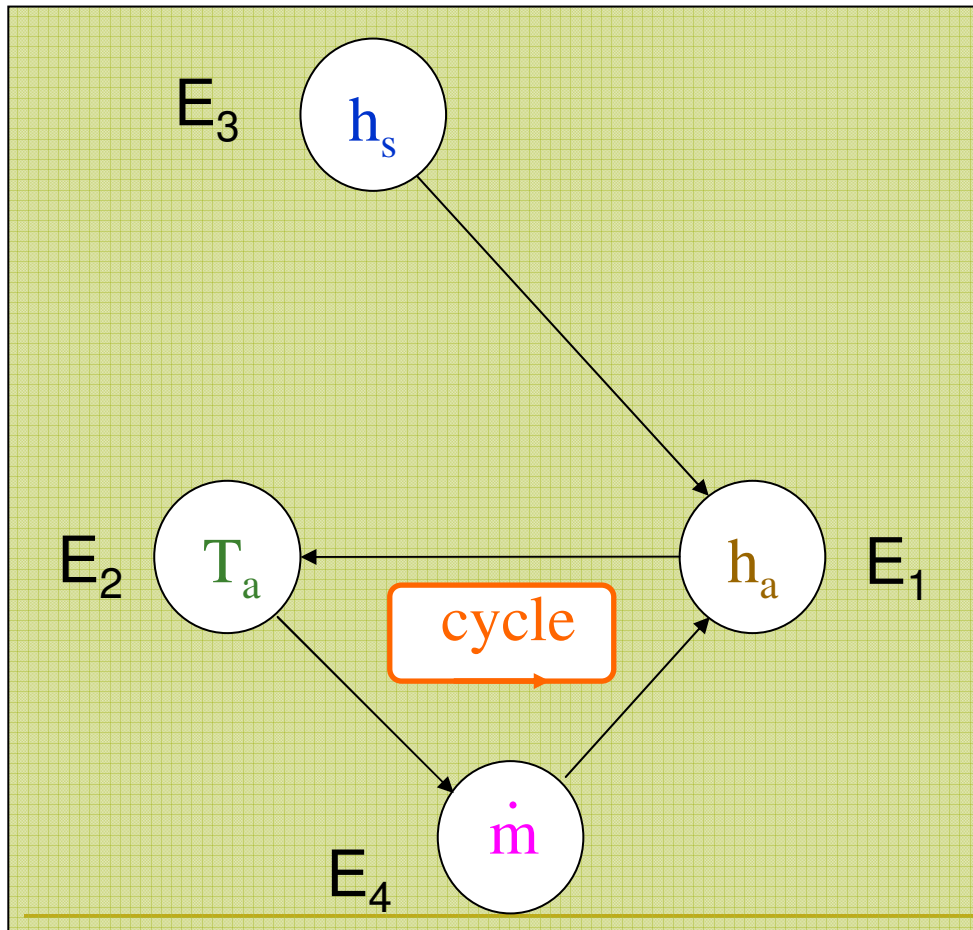
inconnues



Comment il fonctionne?

# Comment fait SPARK? (ex2)

Represente le flux de données dans un graphe orienté



$$E_1 \quad h_a = \frac{\dot{m} \cdot h_s + q_i}{\dot{m}}$$

$$E_2 \quad T_a = \frac{h_a - w_a \cdot b}{a + w_a \cdot c}$$

$$E_3 \quad h_s = a \cdot T_s + w_s \cdot (b + c \cdot T_s)$$

$$E_4 \quad \dot{m} = \max(0, k_P (T_a - T_{set}))$$

Comment il fonctionne?

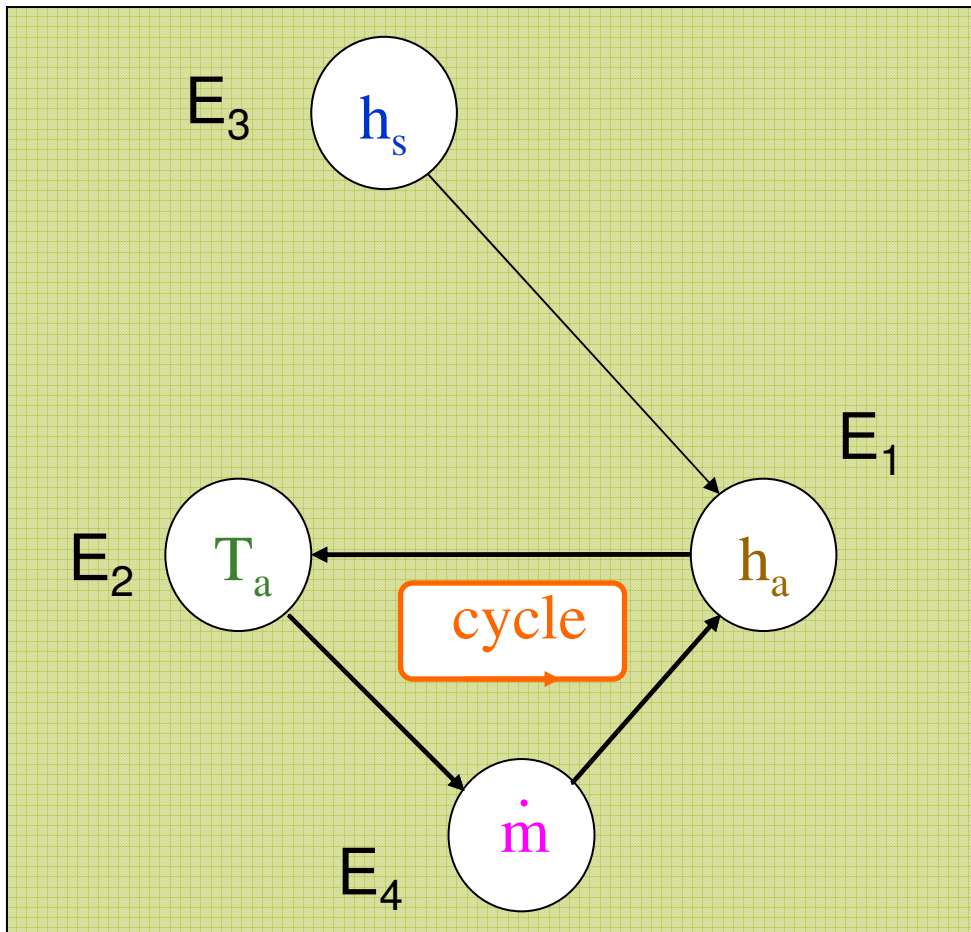
→ Il faut briser le cycle !

# Comment fait SPARK? (ex2)

Où briser le cycle



Les variables dont on se sert pour briser le cycle sont des variables d'itération



Coupes possibles:

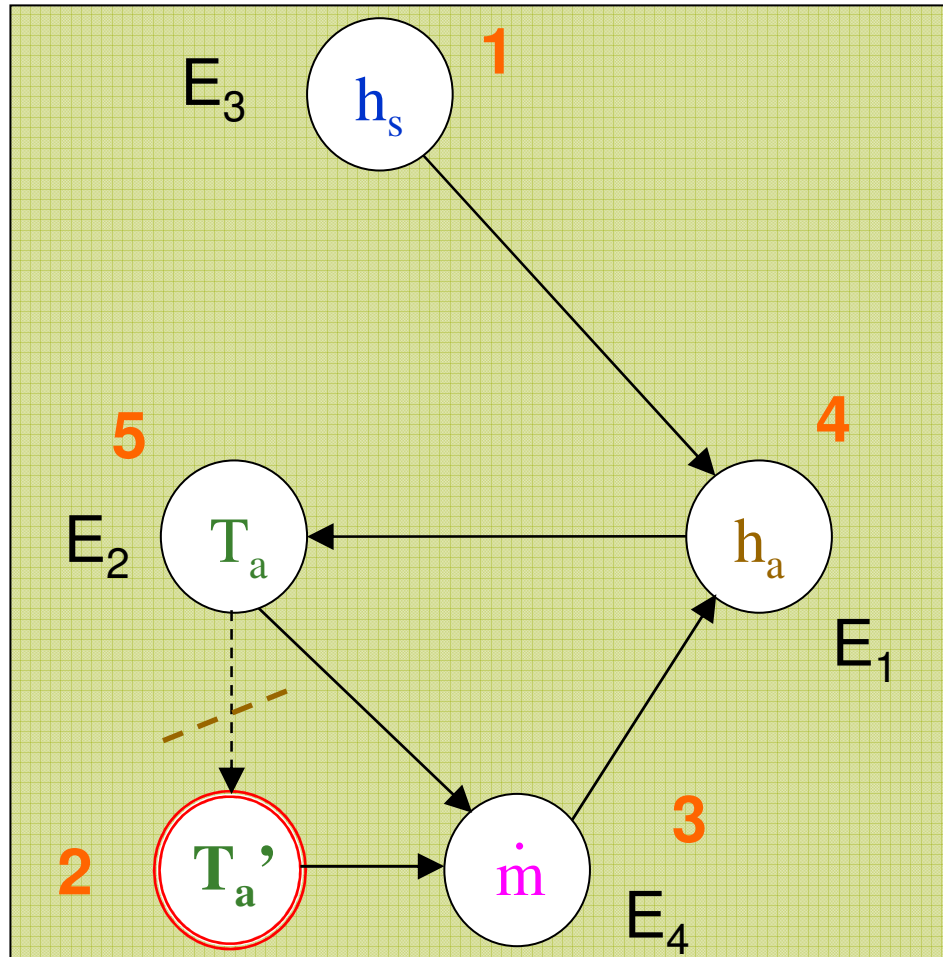
1.  $\{\dot{m}\}$
2.  $\{T_a\}$  ←
3.  $\{h_a\}$
4.  $\{T_a, \dot{m}, h_a, h_s\}$

Comment il fonctionne?

- la coupe 4 n'est pas optimale
- on choisi la coupe la plus petite possible

# Comment fait SPARK? (ex2)

## Coupe du graphe



➤ La coupe introduit un nouveau point dans le graphe:  $T_a'$

➤  $T_a$  est la variable coupée ou « **Break variable** »

➔ Méthode de résolution:

$$T_a' = H(T_a)$$

Ex: Newton-Raphson

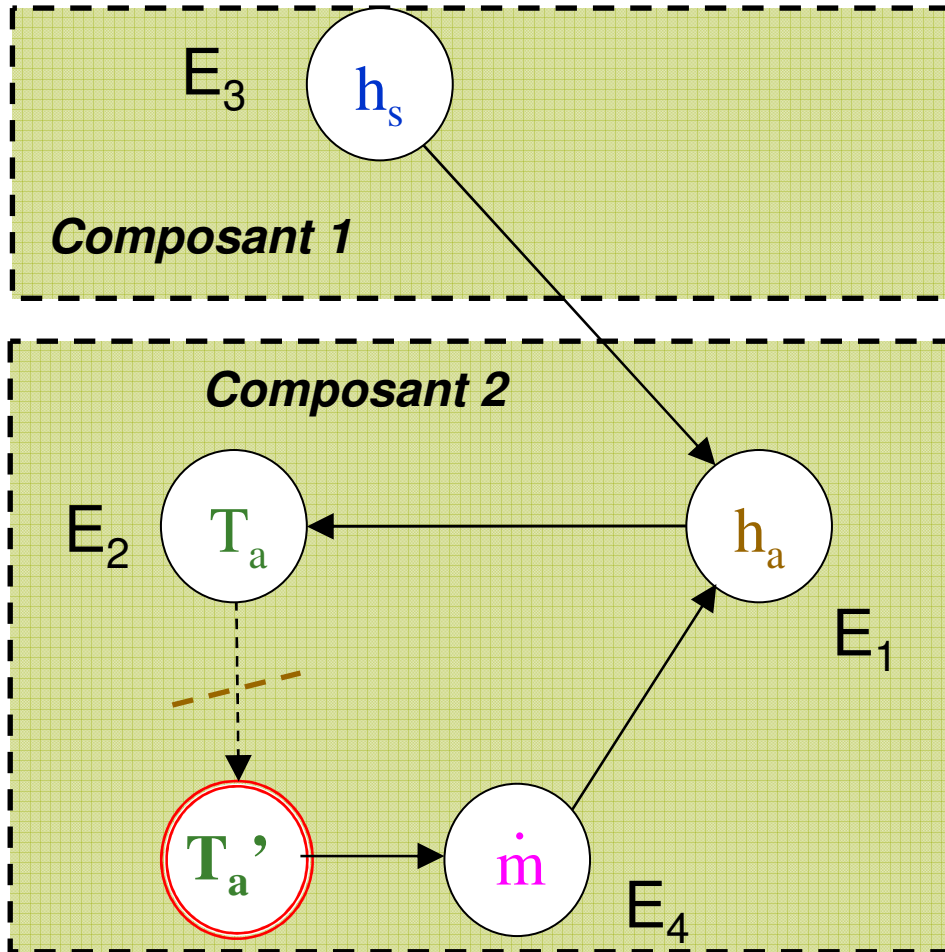
$$T_a^{k+1} = T_a^k - \frac{F(T_a^k)}{\frac{\partial F}{\partial T_a}(T_a^k)}$$

avec  $T_a' = T_a^{k+1}; T_a^k = T_a$

Comment il fonctionne?

# Comment fait SPARK? (ex2)

## Décomposition en composants forts



- trouver les sommets dépendants
- on trouve 2 sous problèmes indépendants

Comment il fonctionne?

➔ les sous problèmes sont appelés composants et sont résolus séparément

# Comment fait SPARK? (ex2)

## ■ Résolution du composant 1

- Pas d'itération

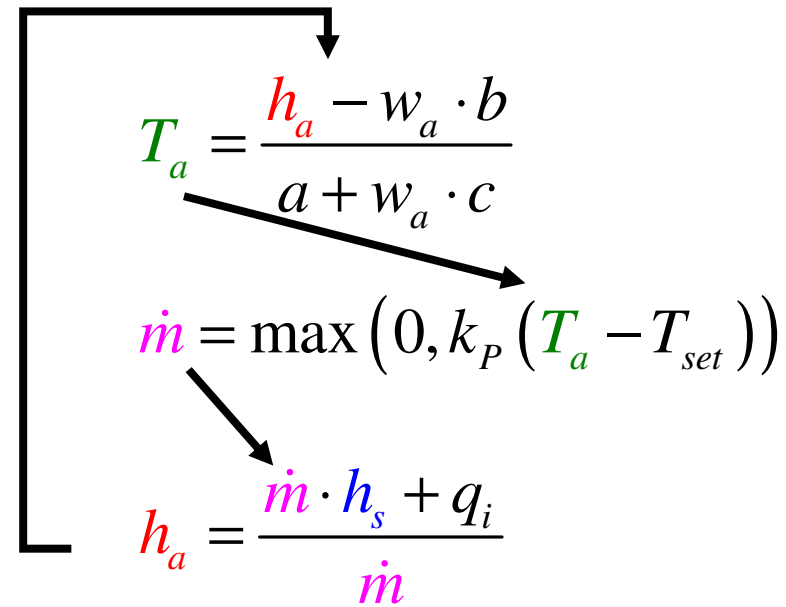
## ■ Résolution du composant 2

- Solution itérative requise
- Valeur initiale pour  $T_a$
- Fonction résiduelle:

$$F(T_a) = T_a - \frac{h_a - w_a \cdot b}{a + w_a \cdot c} \equiv 0$$

Coût:  $O(n^3) = O(1^3)$

$$h_s = a \cdot T_s + w_s \cdot (b + c \cdot T_s)$$



Comment il fonctionne?

→ la méthode de SPARK est 64 fois plus rapide que la méthode brutale



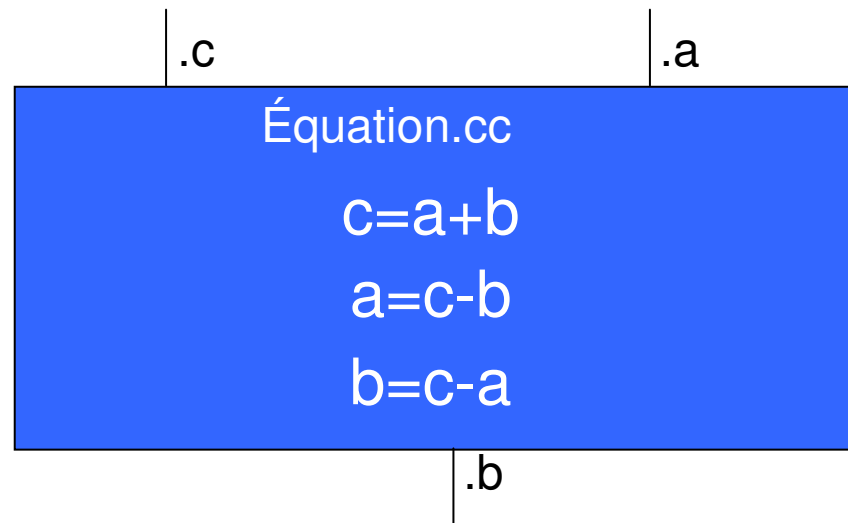
---

# SPARK

- ❑ A quoi il sert
- ❑ Comment il fonctionne
- ❑ Comment on l'utilise

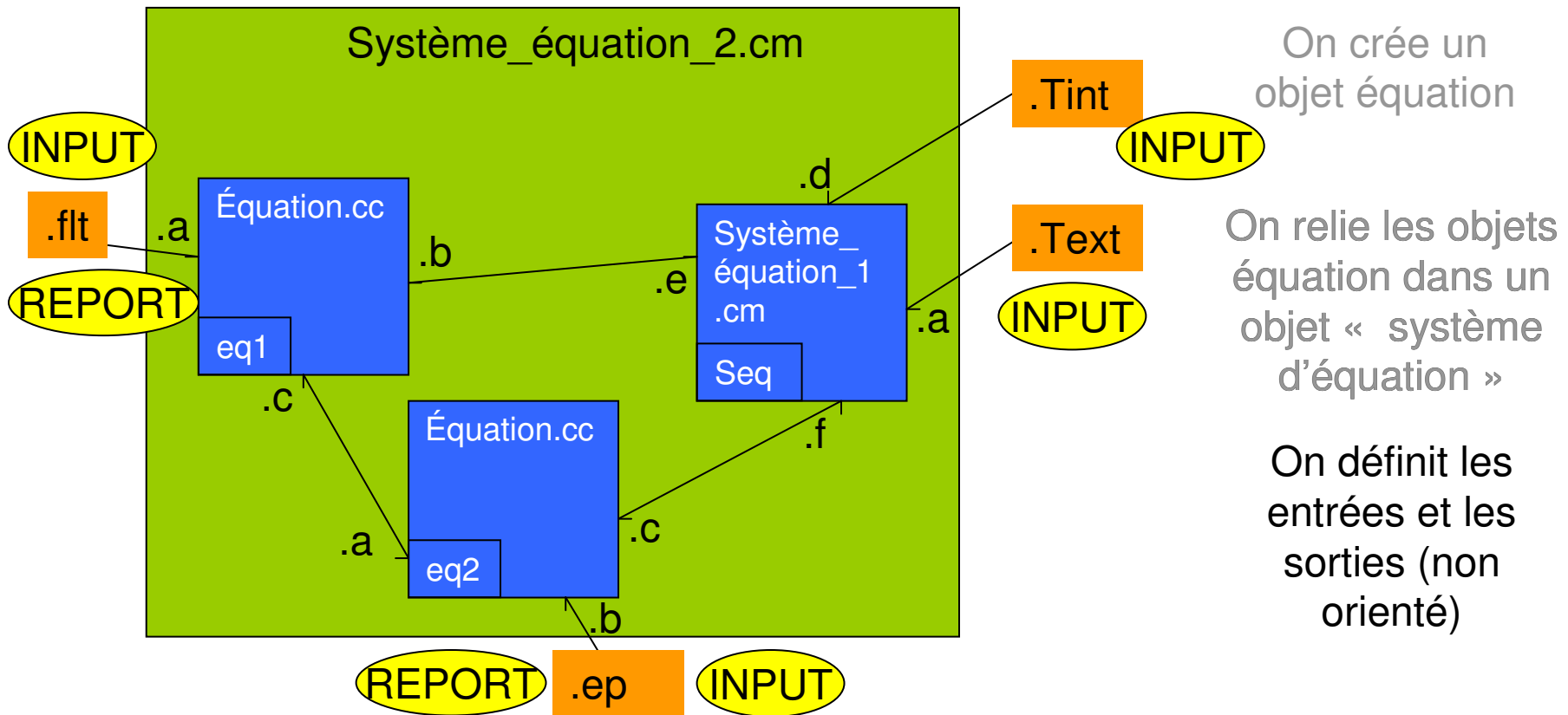
# Comment construire un problème

On crée un objet équation



Comment on l'utilise?

On crée un fichier système d'équation (extension « .cm »)



On crée un objet équation

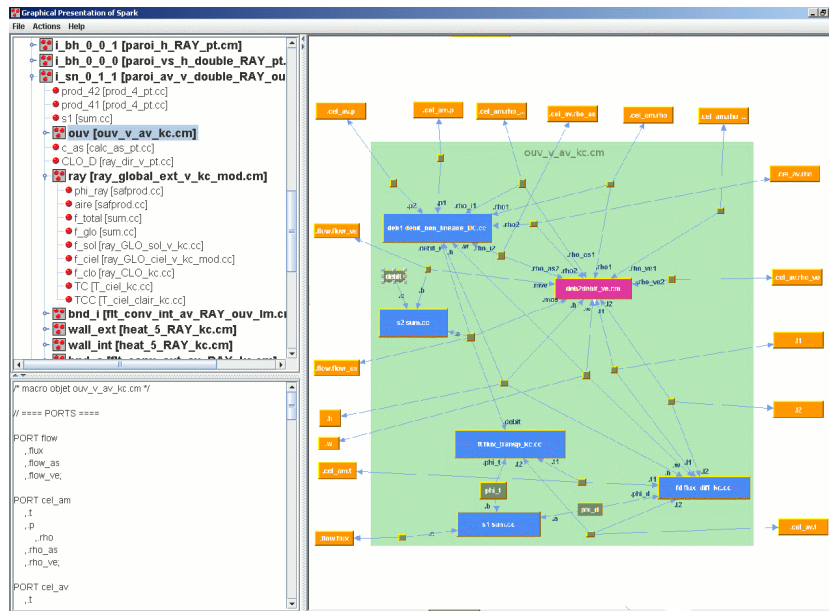
On relie les objets équation dans un objet « système d'équation »

On définit les entrées et les sorties (non orienté)

Comment on l'utilise?

# Outil Graphique

Le logiciel « ViewSpark » en cours de développement pour générer une représentation graphique automatiquement.



Comment on l'utilise?

- i\_bh\_0\_0\_1 [paroi\_h\_RAY\_pt.cm]
- i\_bh\_0\_0\_0 [paroi\_vs\_h\_double\_RAY\_pt.cm]
- i\_sn\_0\_1\_1 [paroi\_av\_v\_double\_RAY\_out.cm]
  - prod\_42 [prod\_4\_pt.cc]
  - prod\_41 [prod\_4\_pt.cc]
  - s1 [sum.cc]
  - **ouv [ouv\_v\_av\_kc.cm]**
    - s2 [sum.cc]
    - s1 [sum.cc]
    - ft [flux\_transp\_kc.cc]
    - fd [flux\_diff\_kc.cc]
    - **deb2 [debit\_ve.cm]**
      - deb1 [debit\_non\_lineaire\_DC.cc]
    - c\_as [calc\_as\_pt.cc]
    - CLO\_D [ray\_dir\_v\_pt.cc]
  - ray [ray\_global\_ext\_v\_kc\_mod.cm]
    - phi\_ray [safprod.cc]
    - aire [safprod.cc]
    - f\_total [sum.cc]
    - f\_glo [sum.cc]
    - f\_sol [ray\_GLO\_sol\_v\_kc.cc]
    - f\_ciel [ray\_GLO\_ciel\_v\_kc\_mod.cc]
    - f\_clo [ray\_CLO\_kc.cc]

```

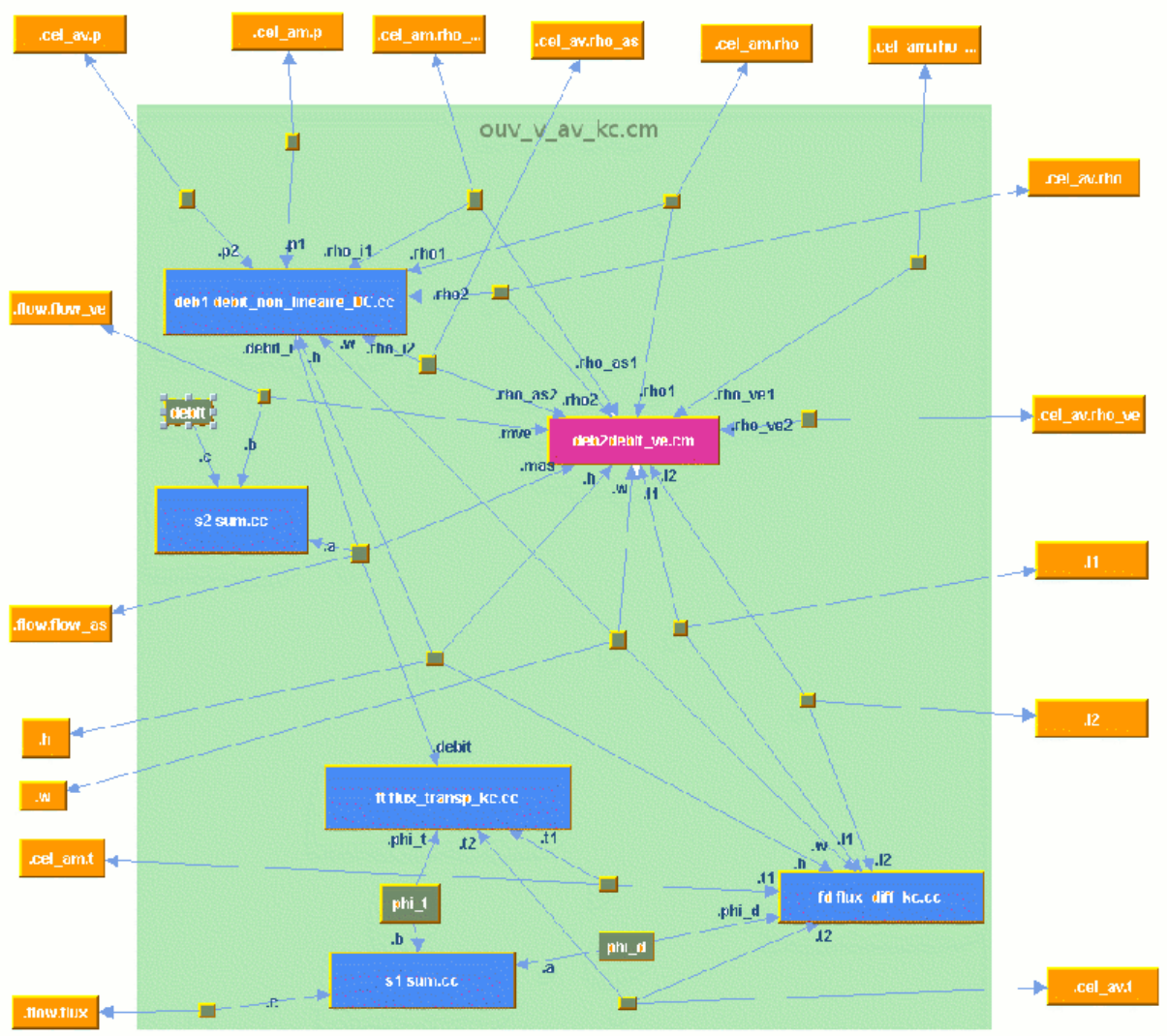
/* macro objet ouv_v_av_kc.cm */

// ===== PORTS =====

PORT flow
,flux
,flow_as
,flow_ve;

PORT cel_am
,t
,p
,rho
,rho_as
,rho_ve;

PORT cel_av
,t
    
```



---

# Conclusion

- Très facile d'interchanger des modèles
- Lisibilité des modèles (objets basés sur les équations)
- Non orientation du problème
- Langage simple (pas besoin de connaître fortran, C++...)
- Robustesse du solveur
- Lien avec Energy +
- Gratuit: <http://gundog.lbl.gov/VS/spark.html>

Questions???