



Fiche d'accompagnement de la démonstration sur le développement de simulateurs OpenFLUID

Objectifs: Découvrir en pratique les principes de développement de simulateurs pour OpenFLUID, via l'environnement de développement dédié

Pré-requis: Cours sur le développement de simulateurs pour OpenFLUID

1 Le modèle couplé

Le modèle développé est un modèle couplé de transfert du ruissellement de surface sur des unités spatiales de classe SU. A partir d'un signal de pluie, le **partage entre ruissellement et infiltration** est assuré par une modélisation de type **SCS Curve Number**. Le transfert du ruissellement sous la forme de **débit** à l'exutoire de chaque unité spatiale est assuré par un modèle simplifié de transfert par **décalage dans le temps de l'onde** (lag). Ce modèle de transfert utilise la **longueur d'écoulement** et la **pente** des unités spatiales.

1.1 Modèle de partage ruissellement-infiltration

Le modèle de partage ruissellement-infiltration à partir du signal de pluie est basé sur la méthode SCS-CN (USDA, TR-55¹).

Soit R le ruissellement, P la pluie, S le coefficient de rétention

$$0.0001 \leq S \leq 0.008$$

$$\text{Si } P \leq (0.2 \cdot S) \text{ alors } R = 0.0$$

$$\text{Si } P > (0.2 \cdot S) \text{ alors } R = \frac{(P - 0.2 \cdot S)^2}{(P + 0.8 \cdot S)}$$

1. Technical Release 55 : Urban Hydrology for Small Watersheds. USDA (U.S. Department of Agriculture). 1986. http://www.nrcs.usda.gov/Internet/FSE_DOCUMENTS/stelprdb1044171.pdf

1.2 Modèle de transfert

Le modèle de transfert utilise le ruissellement, la longueur d'écoulement et la pente de chaque unité spatiale de classe SU.

Soit d la distance de transfert, t le temps de transfert, v la vitesse de transfert

$$v = \frac{d}{t}$$

$$t = \frac{d}{v}$$

Soit Δt la durée d'un pas de temps de simulation, n le nombre de pas de temps pour le transfert sur une unité donnée

$$n \cdot \Delta t = \frac{d}{v}$$

$$n = \frac{d}{v \cdot \Delta t}$$

Avec v équivalent à la racine carrée de la pente p de chaque unité, multipliée par un coefficient de célérité moyenne c , on obtient

$$n = \frac{d}{c \cdot \sqrt{p} \cdot \Delta t}$$

Le calcul du débit en sortie Q d'une unité spatiale sera égale à la somme des apports des unités amont, auquel on ajoute le débit local.

Soit Q le débit local en sortie d'une unité donnée de type SU, A l'aire de cette unité, H la hauteur de ruissellement, Δt la durée d'un pas de temps

$$Q = \frac{H \cdot A}{\Delta t}$$

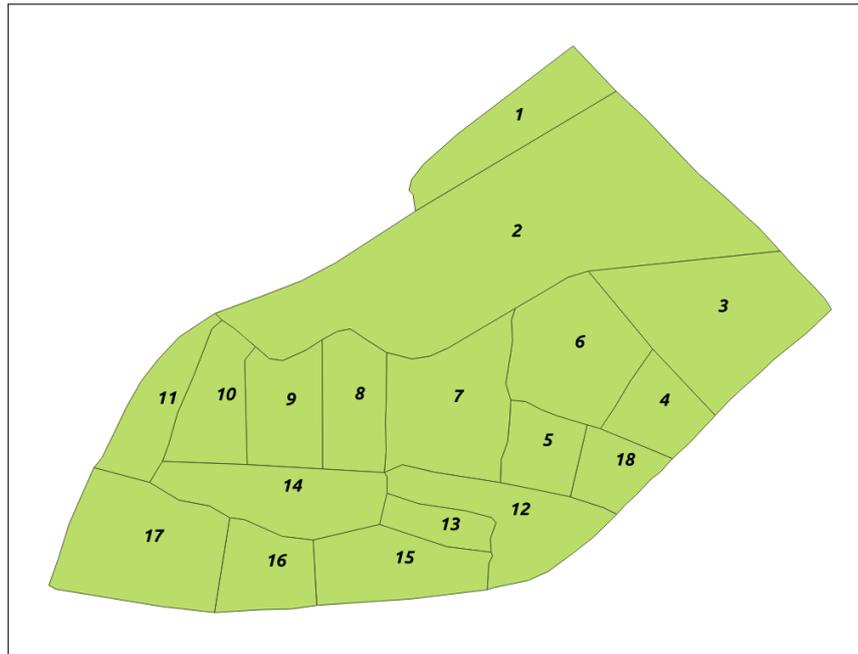
Note: Les hypothèses simplificatrices suivantes sont utilisées au cours de cette démonstration :

- Le mécanisme de prise en compte des apports amont est simplifié
- Les unités spatiales ne débordent pas

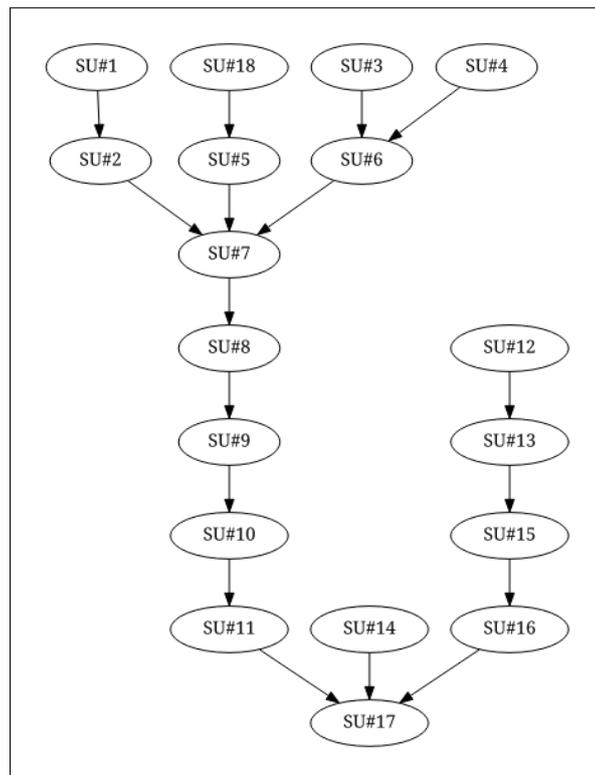
2 La zone étudiée

La zone étudiée qui servira de support pour cette démonstration est un sous-bassin-versant du bassin-versant du Bourdic, situé à 25km au nord-ouest de Montpellier, près de la commune de Saint-Bauzille-de-la-Sylve.

Pour cette démonstration, cette zone étudiée a été découpée en 18 unités spatiales de classe SU. Les figures ci-après donnent un aperçu de la géométrie et de la connectivité de ces unités spatiales.



Géométrie du domaine spatial



Connectivité du domaine spatial

3 Le jeu de données OpenFLUID

Le jeu de données OpenFLUID utilisé pour cette démonstration a été préparé afin d'alléger la démonstration :

- Les observateurs sont pré-configurés pour les sorties du modèle couplé
- Le fichier de pluie et de distribution de la pluie dans l'espace sont déjà présents
- La période de simulation est pré-positionnée

4 Développement des simulateurs

4.1 Production ruissellement-infiltration (modèle SCS-CN)

Ce modèle sera implanté dans un simulateur OpenFLUID ayant pour identifiant `water.surf.runoff-infilt.scs-cn`.

Ce simulateur :

- requiert l'attribut spatial `S` pour l'équation du modèle SCS-CN,
- requiert la variable `water.atm-surf.H.rain` contenant la hauteur de pluie,
- produit la variable `water.surf.H.runoff` contenant le ruissellement calculé,
- produit la variable `water.surf.H.infiltration` contenant l'infiltration calculée.

Lors du développement de ce simulateur, nous aborderons :

- la déclaration dans la signature d'attributs requis, la déclaration de variables requises et produites,
- l'utilisation de boucles spatiales pour parcourir les unités spatiales d'une classe,
- l'initialisation de variables de simulation,
- l'accès aux attributs spatiaux,
- l'accès aux variables de simulation,
- la production de variables de simulation.

Nous aborderons également la mise en place d'un générateur de variables de simulation dans le modèle couplé.

L'attribut `S` nécessaire pour le fonctionnement de ce simulateur pourra être positionné à la valeur 0.0005 pour toutes les unités spatiales dans un premier temps, puis ajusté globalement ou indépendamment sur chaque unité spatiale par la suite.

4.2 Transfert de surface (Modèle lag)

Ce modèle sera implanté dans un simulateur OpenFLUID ayant pour identifiant `water.surf.transfer.lag`.

Ce simulateur :

- requiert le paramètre de simulateur `meance1` contenant le coefficient de célérité moyenne,
- requiert l'attribut spatial `area` contenant la surface de chaque unité spatiale,
- requiert l'attribut spatial `slope` contenant la pente de chaque unité spatiale,
- requiert l'attribut spatial `flowdist` contenant la distance d'écoulement de chaque unité spatiale,
- requiert la variable `water.surf.H.runoff` contenant le ruissellement,
- produit la variable `water.surf.Q.downstream` contenant le débit calculé.

Lors du développement de ce simulateur, en complément des notions abordées lors du développement du précédent simulateur, nous aborderons :

- la déclaration dans la signature de paramètres de simulateur,
- l'utilisation de paramètres de simulateur,
- l'utilisation de la connectivité entre unités spatiales,
- l'utilisation de données internes pour conserver l'état du modèle entre deux pas de temps.

Le paramètre `meance1` de ce simulateur pourra être positionné à la valeur 0.5 dans un premier temps, puis ajusté par la suite.