

**Record Number:** 16730  
**Author, Monographic:** Heniche, M.//Secretan, Y.//Leclerc, M.  
**Author Role:**  
**Title, Monographic:** DISPERSIM 1.0a01: guide d'utilisation  
**Translated Title:**  
**Reprint Status:**  
**Edition:**  
**Author, Subsidiary:**  
**Author Role:**  
**Place of Publication:** Québec  
**Publisher Name:** INRS-Eau  
**Date of Publication:** 2000  
**Original Publication Date:** Juin 2000  
**Volume Identification:**  
**Extent of Work:** i, 112  
**Packaging Method:** pages incluant un annexe  
**Series Editor:**  
**Series Editor Role:**  
**Series Title:** INRS-Eau, rapport de recherche  
**Series Volume ID:** 558 g1  
**Location/URL:**  
**ISBN:** 2-89146-334-X  
**Notes:** Rapport annuel 2000-2001  
**Abstract:** Réalisé pour Pêches et Océans canada, Garde côtière  
**Call Number:** R000558 g1  
**Keywords:** rapport/ ok / guide/ dl

***DISPERSIM Guide d'utilisation***

***DISPERSIM version 1.0a01***

***Juin 2000***

# **DISPERSIM**

## **Guide d'utilisation**

Dispersion version 1.0a01  
Juin 2000

## ÉQUIPE DE RÉALISATION

### Institut National de la Recherche Scientifique - Eau

Mourad Heniche	Associé de Recherche, PhD
Michel Leclerc	Professeur, PhD
Yves Secretan	Professeur, PhD

©INRS-EAU, 2000  
ISBN : 2-89146-334-X

Pour les fins de citation :  
**Heniche M., Secretan Y., Leclerc M. (2000).**  
DISPERSIM 1.0a01, Guide d'utilisation. Rapport INRS-Eau R558-G1.

# Table des matières

<b>Chapitre 1 Aperçu général du logiciel</b>	<b>1-1</b>
Utilité du logiciel	1-1
Matériel requis	1-2
Licence d'utilisation	1-3
Démarrage du logiciel	1-3
Langue d'usage	1-4
Unités des grandeurs physiques	1-4
Arrêt du logiciel	1-4
<b>Chapitre 2 Processus de travail</b>	<b>2-1</b>
Description schématique	2-1
Structure du fichier de commandes	2-1
Structure du nom des fichiers	2-16
Progression de la simulation	2-16
<b>Chapitre 3 Comment gérer une simulation?</b>	<b>3-1</b>
Créer une nouvelle simulation	3-1
Définir la discrétisation du problème	3-1
Lire les données	3-2
Résoudre le problème	3-8
Imprimer les résultats	3-9
<b>Chapitre 4 Comment obtenir une solution de transport-diffusion?</b>	<b>4-1</b>
Comment valider les données d'entrée (phase préliminaire)?	4-1
Comment initialiser le modèle?	4-2
Comment converger la solution?	4-6
Comment calibrer le modèle?	4-14
Comment valider le modèle ?	4-15
Questions fréquentes	4-15
<b>Chapitre 5 Annexe</b>	<b>5-1</b>
Dictionnaire de langue	5-1
Librairie d'éléments finis	5-2
Élément fini CD2DNC_CCN	5-2
Élément fini CD2DNC_CLF	5-6
Élément fini CD2DNC_DBO	5-10
Élément fini CD2DNC_MES	5-14
Élément fini CD2DNC_MET	5-18
Élément fini CD2DNC_TOX	5-22
Librairie de schémas temporels	5-26
Traitement des données transitoires	5-26
Dépendances des blocs	5-27
Exemple de fichier de commandes	5-30
Formats des fichiers d'entrée	5-35
Formats des fichiers de résultats	5-44
<b>Chapitre 6 Glossaire</b>	<b>6-1</b>

# Dispersim - Guide d'utilisation

Le guide d'utilisation de DISPERSIM vous présente toute l'information relative à son utilisation. Il est important de noter que DISPERSIM est un moteur numérique qui ne dispose pas d'interface graphique. En outre, il fonctionne uniquement en mode texte. Les principales sections à partir desquelles vous pourrez obtenir de l'information sont:

- **Aperçu général du logiciel**
- **Processus de travail**
- **Comment gérer une simulation?**
- **Comment obtenir une solution de transport-diffusion?**
- **Annexe**

# Chapitre 1 Aperçu général du logiciel

- Utilité du logiciel ;
- Matériel requis ;
- Licence d'utilisation ;
- Démarrage du logiciel ;
- Langue d'usage ;
- Unités des grandeurs physiques;
- Arrêt du logiciel.

---

## Utilité du logiciel

Le logiciel DISPERSIM est un code éléments finis conçu et développé par l'INRS-Eau, un centre de recherche de l'Université du Québec, dans le but de servir d'outil destiné à la simulation bidimensionnelle horizontale de la dispersion de substances passives ou réactives dans les estuaires, les fleuves et les rivières. DISPERSIM est adressé à des usagers d'horizons diverses disposant toutefois de notions d'hydraulique fluviale et de qualité de l'eau.

La base du programme repose sur la résolution par éléments finis des équations eulériennes de transport-diffusion non conservatives, en régime permanent ou transitoire, gouvernant le processus de dispersion en milieu aqueux. DISPERSIM est un modèle bi-dimensionnel intégré sur la verticale, les variables calculées sont donc des valeurs moyennées sur la colonne d'eau. DISPERSIM peut être exploité tant dans un cadre académique pour la compréhension des mécanismes physico-chimique que dans un contexte de réalisation de projet sur des cas réels. En entrée, DISPERSIM demande comme données de base pour le processus de transport-diffusion les données hydrodynamiques à savoir :

- 1) composante de vitesse en x ;
- 2) composante de vitesse en y ;
- 3) profondeur de l'écoulement ;
- 4) diffusivité verticale ;
- 5) diffusivité horizontale.

De plus, DISPERSIM autorise la prise en compte de cinétiques de transformation spécifiques pour simuler les sources et les puits. Les cinétiques prises en considération sont les suivantes :

- 1) les matières en suspension (MES) ;
- 2) l'oxygène dissous ;
- 3) la demande biochimique en oxygène ;
- 4) les coliformes fécaux ;
- 5) les métaux lourds ;
- 6) les toxiques.

Chaque cinétique est confiée à un élément spécialisé, au nombre de six (voir **Librairie d'éléments finis**), au profit d'une plus grande simplicité d'utilisation, des fichiers de données de taille réduite et des délais d'exécution plus rapides.

En outre, DISPERSIM est un outil de modélisation mathématique par éléments finis des rejets urbains par temps de pluie lorsque la surcharge d'un réseau d'égout est rejetée dans un cours d'eau. Il peut être employé à des fins telles que l'établissement de règles de contrôle des rejets de temps de pluie, le dimensionnement des ouvrages d'assainissement. Étant sa capacité à prédire le transport de matière en suspension, on peut penser à des applications en sédimentologie telle que la détermination des zones de déposition pour optimiser les interventions de dragage dans le cadre de l'entretien des voies de navigation maritime.

## Matériel requis

DISPERSIM tourne sur ordinateur personnel (P.C.) sur les plateformes Win32 (Windows 95/98, Windows NT3.51/4.0 etc...) et gère dynamiquement la mémoire dont il a besoin. Pour de bonnes performances de rapidité d'exécution il est conseillé de le faire tourner en mémoire vive (RAM) car l'accès à la mémoire virtuelle sur disque détériore dramatiquement la vitesse de l'exécution.

La mémoire requise dépend directement de la taille de l'application à simuler. L'indice permettant de quantifier la taille en mégabytes d'une simulation est le nombre de degrés de liberté total (NDLT). Pour les paramètres de résolution préconisés, la mémoire allouée par DISPERSIM est estimée à environ  $6.5 \times 10^{-4}$  fois NDLT en mégabytes. À titre indicatif, au Tableau 1 nous avons l'espace en mémoire requis pour des valeurs de NDLT variant entre 10 000 et 100 000.

NDLT (x1000)	10	20	30	40	50	60	70	80	90	100
Espace(Meg)	6,5	13	19,5	26	32,5	39	45,5	52	58,5	65



## Licence d'utilisation

DISPERSIM est un logiciel protégé. Pour en faire usage il est nécessaire après l'avoir installé d'effectuer l'enregistrement dont la procédure à suivre est décrite dans le fichier `dispersim_enregistrement.txt` disponible dans le même répertoire que `dispersim.exe`.

Après enregistrement, un type de licence d'utilisation parmi deux (2) est attribué :

- 1 - DEMONSTRATION : attribuée pour des exercices de démonstration. L'accès à certaines fonctionnalités du logiciel n'est pas autorisé. En outre, aucun calcul ni impression de résultats n'est possible.
- 2 - COMPLÈTE : permet d'avoir accès à toutes les fonctionnalités du logiciel. La durée dans le temps peut être limitée ou illimitée.

---

## Démarrage du logiciel

Sur les plateformes Win32 (Windows 95/98, Windows NT3.51/4.0 etc...) le démarrage du logiciel est activée, dans une fenêtre en mode texte MS-DOS(command prompt), en tapant :

```
dispersim
```

Juste après le lancement, DISPERSIM affiche systématiquement au démarrage le nom du logiciel. Ensuite, il affiche la date, l'heure et l'espace mémoire occupé. Puis, il affiche les informations relatives à la licence du logiciel : nom du logiciel, code d'accès et type de licence.

Ces informations peuvent être affichées soit au prompt soit dans un fichier de sortie. En effet, le logiciel lit les instructions de l'unité FORTRAN numéro 5 et imprime les informations relatives au déroulement de la simulation dans l'unité FORTRAN numéro 6. Les unités fortran d'entrée/sortie 5 et 6 peuvent être redirigées sur les fichiers `fichier.inp` de commandes et `fichier.out` de sortie respectivement comme suit :

```
dispersim < fichier.inp > fichier.out
```

Si aucun fichier d'entrée/sortie n'est spécifié on parlera d'exécution au prompt. Dans ce cas, tous les messages de sortie de DISPERSIM sont alors affichés directement à l'écran. De plus, immédiatement après le démarrage, le logiciel est en mode balayage qui est désactivé en tapant la commande STOP.

Dans le cas où l'exécution est gérée par un fichier de commandes le mode balayage est désactivé automatiquement.

Les messages de sortie de DISPERSIM sont affichés sur fichier de sortie s'il a été défini au préalable.

---

## Langue d'usage

DISPERSIM dispose d'un module de traduction de langue qui permet de traduire les messages internes au logiciel. La langue d'usage est définie dans le fichier de configuration `dispersim.ini` au moyen de la variable `LANGUE` la syntaxe se présente comme suit :

```
_LANGUE=' .xxx '
```

Les caractères `xxx` définissent la langue d'usage (voir **Dictionnaire de langue**). Par défaut le français a préséance.

---

## Unités des grandeurs physiques

Dans DISPERSIM, les unités des grandeurs physiques sont exprimées dans le Système International (SI).

---

## Arrêt du logiciel

Pour une exécution sans fichier d'entrée, on procède en deux temps pour arrêter le logiciel. Dans un premier temps il faut sortir du mode balayage en tapant au clavier la commande `STOP` suivi d'un *Entrée*. Ensuite on reprend la même opération une seconde fois pour arrêter le logiciel.

# Chapitre 2 Processus de travail

DISPERSIM est un logiciel modulaire, où chaque module est dédié à une tâche spécifique. L'objectif du processus de travail est de présenter tous les modules existant dans DISPERSIM, leur fonction ainsi que les prérequis et dépendances pour les exécuter. Les développements de ce chapitre sont relatifs à :

- Description schématique
- Structure du fichier de commandes
- Structure du nom des fichiers
- Progression de la simulation

---

## Description schématique

Après le démarrage du logiciel DISPERSIM, la simulation consiste à faire travailler les différents modules pour atteindre l'objectif ciblé. C'est au moyen de commandes qu'il est possible d'accomplir les différentes tâches.

Schématiquement, on peut classer les commandes en trois groupes :

- le groupe de lecture des données d'entrée
- le groupe de résolution
- le groupe d'impression des résultats en sortie

---

## Structure du fichier de commandes

- Définition
- Blocs
- Variables

---

## Définition

Le fichier de commandes est composé d'instructions et de commentaires optionnels qui permettent de personnaliser

chaque simulation. Les commentaires peuvent être précédés par l'un ou l'autre des deux symboles # ou !.

Tous les commentaires précédés par un # sont automatiquement imprimés en sortie. Ce qui n'est pas le cas pour ceux précédés par un !.

Les instructions sont constituées d'un jeu de commandes formé de Blocs et de Variables. Les instructions que nous retrouvons systématiquement dans le fichier de commandes sont les choix du type d'élément fini et du schéma temporel, les définitions de la formulation, des fichiers de données et de résultats et l'appel à des blocs pour exécuter des tâches spécifiques. Toute simulation doit être ponctuée par l'appel au bloc STOP qui met un terme à l'exécution de DISPERSIM. Le fichier de commandes se présente donc comme suit (voir **Exemple de fichier de commandes**) :

```
#Spécimen de fichier de commandes pour DISPERSIM
# Informations ou identifications relatives à la
# simulation
```

```
! Choisir le type d'élément fini (voir ELTYP).
```

```
Instruction
```

```
! Choisir le type de schéma temporel (voir
! STEMP).
```

```
Instruction
```

```
! Définir la formulation (voir FORM)
```

```
Instruction
```

```
! Définition des fichiers de données et de
! résultats.
```

```
Instruction
```

```
! Liste des instructions pour l'acquisition des
! données, la simulation et le post-traitement.
```

```
Instruction
```

```
.
.
.
```

```
! Fin de la simulation.
```

```
STOP
```

Remarque : Chaque instruction doit être écrite sur une seule ligne de 1024 caractères de long au maximum.

## Blocs

Chaque commande est identifiée par un bloc d'exécution. Chaque bloc est muni d'une unité d'impression optionnelle qui retourne en sortie, en partie ou en intégralité, les données d'entrée et/ou les informations spécifiques à chaque bloc. Le niveau d'impression des informations est piloté par l'entier positif M, qui par défaut est égal à 0. La syntaxe des blocs est :

BLOC

ou bien

BLOC[M]

Certains blocs sont accompagnés d'une table de valeurs réelles :

BLOC(réel, réel, ..., réel)

ou bien

BLOC[M](réel, réel, ...réel)

Le nom de chaque bloc est un mot réservé qui ne doit pas excéder 4 caractères au maximum. Dans DISPERSIM on dénombre 17 blocs :

- COND
- COOR
- ELEM
- ERR
- FIN
- FORM
- INIT(réel, réel, ..., réel)
- PRCO
- PREL
- PRGL(réel, réel, ..., réel)

- PRNO
  - POST
  - RESI
  - SOLC
  - SOLR
  - SOLV(réel,réel,réel)
  - STOP
- 

## COND

- Fonction

Bloc de lecture des CONDitions aux limites.

- Variable associée

MCND et TASCND

- Prérequis

Définir obligatoirement la variable MCND l'appel à COND.

Définir la variable TASCND (=0 par défaut) si les conditions aux limites évoluent dans le temps avant l'appel à COND.

Appeler les blocs COOR, ELEM, PRGL, PRNO, PREL et INIT(réel,réel,...,réel) avant l'appel à COND.

---

## COOR

- Fonction

Bloc de lecture des COORdonnées du maillage.

- Variable associée

MCOR

- Prérequis

Définir obligatoirement le variable MCOR avant l'appel à COOR.

Appeler le bloc FORM avant l'appel à COOR.

---

## ELEM

- Fonction

Bloc de lecture des ÉLÉMENTS du maillage.

- Variable associée

## MELE

- Prérequis

Définir obligatoirement la variable MELE avant l'appel à ELEM.  
Appeler les blocs FORM et COOR avant l'appel à ELEM.

---

## ERR

- Fonction

Bloc de calcul des ERReurs numériques.

- Variable associée

MERR

- Prérequis

Définir la variable MERR si l'impression des erreurs numériques dans un fichier est souhaitée avant l'appel à ERR.

Appeler les blocs FORM, COOR, ELEM, PRGL, PRNO et PREL et INIT(réel,réel,...,réel) avant l'appel à ERR.

---

## FIN

- Fonction

Bloc d'impression FINale de la solution en format binaire.

- Variable associée

MFIN et FFFIN.

- Prérequis

Définir les variables MFIN, si l'impression de la solution dans un fichier est souhaitée (voir fort recommandée), et FFFIN avant l'appel à FIN.

Appeler les blocs FORM, COOR, ELEM, PRGL, PRNO et PREL et INIT(réel,réel,...,réel) avant l'appel à FIN.

---

## FORM

- Fonction

Bloc de définition de la formulation du problème. Doit être impérativement appelé au début de chaque simulation.

- Variable associée

ELTYP et STEMP.

- Prérequis

Définir absolument les variables ELTYP et STEMP avant l'appel à FORM.

Mise en garde : Après l'appel à FORM, toutes les données en mémoire virtuelle sont initialisées.

---

## INIT(réel,réel,....,réel)

- Fonction

Bloc de mise à jour de la solution INITiale du problème. La syntaxe du bloc est accompagnée d'une table de réels optionnelle contenant les paramètres d'initialisation associés au type d'élément fini (voir **Librairie d'éléments finis**).

- Variable associée

MINI, FFINI et TASINI

- Prérequis

Définir la variable MINI si la solution initiale est stockée sur fichier avant l'appel à INIT.

Dans le cas où la solution initiale doit être lue sur fichier : définir les variables FFINI et TASINI, si la solution initiale évolue dans le temps, avant l'appel à INIT.

Appeler les blocs FORM, COOR, ELEM, PRGL, PRNO et PREL avant l'appel à INIT.

---

## PRCO

- Fonction

Bloc de calcul des pointeurs pour un PRéCOnditionnement de type matrice ILU.

- Variable associée

ILU et DELPRT.

- Prérequis

Appeler les blocs FORM, COOR et ELEM avant l'appel à PRCO.

---

## PREL

- Fonction

Bloc de lecture des PRopriétés ÉLémentaires.

- Variable associée

MPRE et TASPRES

- Prérequis



Définir, dépendamment du type d'élément fini (voir **Librairie d'éléments finis**), la variable MPRE avant l'appel de PREL.

Définir la variable TASPRES avant l'appel de PREL si les propriétés élémentaires évoluent dans le temps.

Appeler les blocs FORM, COOR, ELEM, PRGL et PRNO avant l'appel à PREL.

---

## PRGL(réel,réel,....,réel)

- Fonction

Bloc de lecture des PRopriétés GLobales.

- Variable associée

Aucune

- Prérequis

Appeler le bloc FORM avant l'appel à PRGL.

---

## PRNO

- Fonction

Bloc de lecture des PRopriétés NOdales.

- Variable associée

MPRN et TASP RN

- Prérequis

Définir, dépendamment du type d'élément fini (voir **Librairie d'éléments finis**), la variable MPRN avant l'appel à PRNO.

Définir la variable TASP RN si les propriétés nodales évoluent dans le temps avant l'appel à PRNO.

Appeler les blocs FORM, COOR, ELEM et PRGL avant l'appel à PRNO.

---

## POST

- Fonction

Bloc de POST-traitement des résultats.

- Variable associée

MPST

- Prérequis

Définir la variable MPST avant l'appel de POST si l'impression du post-traitement dans un fichier est souhaitée.

---

Appeler les blocs FORM, COOR, ELEM, PRGL, PRNO, PREL et INIT(réel,réel,...,réel) avant l'appel à POST.

---

## RESI

- Fonction

Bloc de calcul des RÉSIDus.

- Variable associée

MRES

- Prérequis

La définition de la variable MRES avant l'appel de RESI est obligatoire si l'impression des résidus dans un fichier est souhaitée.

Appeler les blocs FORM, COOR, ELEM, PRGL, PRNO, PREL, INIT(réel,réel,...,réel) et COND avant l'appel à RESI.

---

## SOLC

- Fonction

Bloc de lecture des SOLlicitations Concentrées.

- Variable associée

MSLC et TASSLC

- Prérequis

Définir obligatoirement la variable MSLC si les sollicitations concentrées sont stockées sur fichier avant l'appel à SOLC.

Dans le cas où les sollicitations concentrées sont lues sur fichier : définir la variable TASSLC si les sollicitations concentrées évoluent dans le temps avant l'appel à SOLC.

Appeler les blocs FORM, COOR, ELEM, PRGL, PRNO, PREL et INIT(réel,réel,...,réel) avant l'appel à SOLC.

---

## SOLR

- Fonction

Bloc de lecture des SOLlicitations Réparties.

- Variable associée

MSLR et TASSLR

- Prérequis

Définir obligatoirement la variable MSLR si les sollicitations réparties sont stockées sur fichier avant l'appel à SOLR.

Dans le cas où les sollicitations réparties sont lues sur fichier : définir la variable TASSLR si les sollicitations concentrées évoluent dans le temps avant l'appel à SOLR.

Appeler les blocs FORM, COOR, ELEM, PRGL, PRNO, PREL et INIT(réel,réel,...,réel) avant l'appel à SOLR.

---

## SOLV(réel,réel,réel)

- Fonction

Bloc faisant appel au SOLVeur. La syntaxe du bloc est accompagnée d'une table de réels optionnelle contenant les limiteurs de solution associés à chaque type de degré de liberté.

- Variable associée

MEXE pour l'état d'avancement de la simulation. Pour les autres, elles dépendent du schéma de résolution défini par la variable STEMP.

- Prérequis

Appeler les blocs FORM, COOR, ELEM, PRGL, PRNO, PREL, INIT(réel,réel,...,réel), COND, SOLC (optionnel), SOLR (optionnel) et PRCO (dépendamment de IMPR) avant l'appel à SOLV.

---

## STOP

- Fonction

Bloc d'arrêt du logiciel.

- Variable associée

Aucune

- Prérequis

Aucun.

---

## Variables

Les variables sont toujours suivies d'un champ dynamique qui les définit. La syntaxe des variables est :

`_VARIABLE=valeur`

Le champ dynamique `valeur` se présente sous trois formes :

- Type chaîne de caractères entre guillemets simples ( ' ').
  - Type entier.
  - Type réel.
- 

## Type chaîne de caractères

Les variables du type chaîne de caractères servent en général à définir les noms des fichiers de données et de résultats. Leur utilisation peut être optionnelle ou obligatoire. Dans DISPERSIM on dénombre 18 variables du type chaîne de caractères :

- ELTYP
- FFFIN
- FFINI
- MCND
- MCOR
- MELE
- MERR
- MEXE
- MFIL
- MFIN
- MINI
- MPRE
- MPRN
- MPST
- MRES
- MSLC
- MSLR
- STEMP

---

### ELTYP

Variable fondamentale pour toute simulation. Elle définit le TYPE d'Élément fini devant être utilisé (voir **Librairie d'éléments finis**).

---

## **FFFIN**

Variable définissant le format (ASCII ou binaire) du fichier des degrés de liberté. Elle peut prendre pour valeur ASCII (par défaut) ou BIN.

---

## **FFINI**

Variable définissant le format (ASCII ou binaire) du fichier de solution initiale. Elle peut prendre pour valeur ASCII (par défaut) ou BIN.

---

## **MCND**

Variable définissant le nom ou l'extension du nom du fichier des CoNDitions aux limites.

---

## **MCOR**

Variable définissant le nom ou l'extension du nom du fichier des COoRdonnées des noeuds du maillage.

---

## **MELE**

Variable définissant le nom ou l'extension du nom du fichier des connectivités des ÉLÉments du maillage.

---

## **MERR**

Variable définissant le nom ou l'extension du nom du fichier des résultats des ERReurs numériques.

---

## **MEXE**

Variable définissant le nom ou l'extension du nom du fichier de suivi de la progression de l'ÉXEcution de la simulation.

---

## **MFIL**

Variable définissant le nom générique ou le répertoire de tous les Fichiers d'entrée et de sortie.

---

## **MFIN**

Variable définissant le nom ou l'extension du nom du fichier d'impression de la solution FINale.

---

---

## MINI

Variable définissant le nom ou l'extension du nom du fichier de solution INITiale.

---

## MPRE

Variable définissant le nom ou l'extension du nom du fichier des PPropriétés Élémentaires.

---

## MPRN

Variable définissant le nom ou l'extension du nom du fichier des PPropriétés Nodales.

---

## MPST

Variable définissant le nom ou l'extension du nom du fichier des résultats de PoSt-Traitement.

---

## MRES

Variable définissant le nom ou l'extension du nom du fichier des résultats de RÉSidus.

---

## MSLC

Variable définissant le nom ou l'extension du nom du fichier des SoLlicitations Concentrées.

---

## MSLR

Variable définissant le nom ou l'extension du nom du fichier des SoLlicitations Réparties.

---

## STEMP

Variable fondamentale pour toute simulation. Elle définit le Schéma TEMPorel devant être utilisé (voir **Librairie de schémas temporels**).

---

## Type entier

Les variables du type entier servent en règle général à définir les paramètres de la résolution. Leur utilisation peut être optionnelle

ou obligatoire. Dans DISPERSIM on dénombre 6 variables du type entier :

- ILU
- IMPR
- NITER
- NPAS
- NPREC
- NRDEM

---

## ILU

Variable définissant le niveau de remplissage pour un préconditionnement de type matrice ILU. Par défaut, elle est égale à 0.

---

## IMPR

Variable définissant le type de Matrice de PRéconditionnement :

\_IMPR=0 pour matrice identité

\_IMPR=1 pour matrice masse diagonale

\_IMPR=2 pour matrice tangente diagonale

\_IMPR=3 pour matrice ILU

Par défaut elle est égale à 1.

---

## NITER

Variable définissant le Nombre d'ITÉrations. Par défaut, elle est égale à 25.

---

## NPAS

Variable définissant le Nombre de PAS pour la division du temps  $t$  en incrément  $\Delta t$  dans le cas d'une simulation transitoire ou non stationnaire. Par défaut, elle est égale à 1.

---

## NPREC

Variable définissant le Nombre de PREConditionnements. Par défaut, elle est égale à 1.

---

## NRDEM

Variable définissant le Nombre de ReDÉMarrages. Par défaut, elle est égale à 25.

---

## Type réel

Les variables du type réel servent à définir les paramètres de la méthode de résolution ainsi que les données dans un contexte non stationnaire. Leur utilisation peut être optionnelle ou obligatoire. Dans DISPERSIM on dénombre 12 variables du type réel :

- ALFA
- DELPRT
- DPAS
- EPSDL
- OMEGA
- TASCND
- TASINI
- TASPRES
- TASPRTN
- TASSLC
- TASSLR
- TINI

---

### ALFA

Variable définissant le coefficient  $\alpha$  (ALFA) du schéma temporel d'Le schéma temporel EULER. Elle peut prendre des valeurs comprises entre 0 et 1 inclus. Par défaut, elle est égale à 1.

---

### DELPRT

Variable définissant le coefficient  $\Delta$  (DELta) de PeRTurbation de la matrice de préconditionnement du type ILU. Par défaut, elle est égale à  $10^{-8}$ .

---

### DPAS

Variable définissant le PAS de temps  $\Delta t$  (Delta t) pour la discrétisation du temps. Par défaut, elle est égale à  $10^{+12}$ .

---



## EPSDL

Variable définissant la précision  $\varepsilon$  (EPSilon) des Degrés de Liberté. Par défaut, elle est égale à  $10^{-06}$ .

---

## OMEGA

Variable définissant le facteur  $\omega$  (OMÉGA) de relaxation des degrés de liberté. Par défaut, elle est égale à 1.

---

## TASCND

Variable définissant le Temps ASSocié à la lecture initiale des CoNDitions aux limites non stationnaires (voir **Traitement des données transitoires**).

---

## TASINI

Variable définissant le Temps ASSocié à la lecture initiale de la solution INItiale non stationnaire (voir **Traitement des données transitoires**).

---

## TASPRE

Variable définissant le Temps ASSocié à la lecture initiale des PRopriétés Élémentaires non stationnaires (voir **Traitement des données transitoires**).

---

## TASPRN

Variable définissant le Temps ASSocié à la lecture initiale des PRopriétés Nodales non stationnaires (voir **Traitement des données transitoires**).

---

## TASSLC

Variable définissant le Temps ASSocié à la lecture initiale des SoLlicitations Concentrées non stationnaires (voir **Traitement des données transitoires**).

---

## TASSLR

Variable définissant le Temps ASSocié à la lecture initiale des SoLlicitations Réparties non stationnaires (voir **Traitement des données transitoires**).

---

## TINI

Variable définissant le Temps INItial de la simulation.

---

## Structure du nom des fichiers

Les noms des fichiers de données et de résultats de DISPERSIM sont systématiquement définis au moyen de la concaténation des valeurs de deux Variables du Type chaîne de caractères. Dans l'ordre, la première commune à tous est MFIL et la seconde est associée au bloc pertinent (voir Blocs).

### Exemple :

Le nom du fichier associé au bloc XXX admettant MXXX pour variable du type chaîne de caractères est déterminé par l'une ou l'autre des deux procédures ci-dessous :

```
_MFIL=' nom1 '
```

```
_MXXX=' nom2 '
```

ou bien

```
_MFIL=' '
```

```
_MXXX=' nom1nom2 '
```

Dans les deux cas de figure, le nom du fichier résultant de l'association, dans l'ordre, de MFIL et MXXX est `nom1nom2`.

Remarque : La longueur max des noms des fichiers ne peut dépasser 217 caractères.

---

## Progression de la simulation

- Définition
  - Volume de l'exécution
  - Message à l'arrêt
- 

## Définition

La progression de la simulation est accessible sur le fichier de suivi de simulation défini au moyen de la variable MEXE (voir Structure du nom des fichiers).

Typiquement durant une exécution nous avons dans le fichier de suivi de la simulation une seule ligne de message contenant un entier compris entre 1 et 100. Elle exprime, en pourcentage (%)

du Volume de l'exécution, l'état d'avancement de la procédure de résolution du problème.

---

## Volume de l'exécution

Le volume de l'exécution est un entier égal au nombre d'itérations maximum possible. Il est calculé automatiquement pendant le mode balayage. Il est affiché ensuite au prompt ou dans le fichier de sortie (voir **Démarrage du logiciel**) à la sortie du mode balayage.

---

## Message à l'arrêt

Après arrêt de l'exécution de DISPERSIM nous avons alors systématiquement le message suivant dans le fichier de suivi de la simulation :

```
100  
END
```

Dans le fichier de sortie ou au prompt, la date et l'heure de fin de la simulation sont affichées ainsi que sa durée. Enfin, l'espace mémoire est également indiqué.

Remarque : La durée d'une même simulation peut varier si les conditions de l'exécution changent.

## Chapitre 3 Comment gérer une simulation?

L'objectif de ce chapitre est de présenter la procédure à suivre pour construire un fichier de commandes (voir **Structure du fichier de commandes**), à partir des **Variables** et des **Blocs**, pour gérer une simulation. À chaque étape sont indiqués la variable à définir et le bloc à appeler. Pour s'assurer de respecter les règles de dépendances entre les blocs et les variables (voir **Dépendances des blocs**), il est conseillé de suivre la démarche ci-dessous :

- Créer une nouvelle simulation
- Définir la discrétisation du problème
- Lire les données
- Résoudre le problème
- Imprimer les résultats

---

### Créer une nouvelle simulation

Lorsqu'on crée une nouvelle simulation, bien que facultatif, il est toutefois conseillé d'insérer dans le fichier de commandes du texte, à raison de 1024 caractères par ligne au maximum, qui puisse identifier le processus physique à simuler. Cela consiste en particulier à indiquer la provenance des données de terrain, le choix des conditions aux limites, des sollicitations et des conditions initiales. De plus, on peut également indiquer à quel stade correspond la simulation : initialisation, calibration, prédiction...

En règle général, plus il y a d'information mieux c'est. Cela peut être particulièrement utile en phase d'analyse. Pour que l'information apparaisse dans le fichier de sortie il faut s'assurer que chaque ligne de message soit précédée par un #.

---

### Définir la discrétisation du problème

Définir les variables **ELTYP** (voir **Librairie d'éléments finis**) et **STEMP** (voir **Librairie de schémas temporels**). Appeler le bloc **FORM**.

**Exemple :**

Pour discrétiser un problème de transport-diffusion d'un contaminant conservatif par exemple, dans un contexte stationnaire la commande associée est :

```
_ELTYP= 'CD2DNC_CCN'  
_STEMP= 'STATIQ'  
FORM[0]
```

ou bien dans un contexte transitoire :

```
_ELTYP= 'CD2DNC_CCN'  
_STEMP= 'EULER'  
FORM[0]
```

---

## Lire les données

En règle générale, les données sont stockées dans des fichiers ASCII. L'opération de lecture de chaque type de donnée consiste à définir le nom du fichier de données au moyen des variables appropriées (voir **Structure du nom des fichiers**), de définir le temps associé et le pas de temps s'il y a lieu dans le cas d'une simulation dans le temps ; d'activer ensuite le bloc pertinent pour procéder à la lecture. Pour intégrer les données dans DISPERSIM voici dans l'ordre la procédure à suivre :

- Lire les coordonnées
- Lire les connectivités
- Lire les propriétés globales
- Lire les propriétés nodales
- Lire les propriétés élémentaires
- Lire la solution initiale
- Lire les conditions aux limites
- Lire les sollicitations concentrées
- Lire les sollicitations réparties

## Lire les coordonnées

Les coordonnées sont stockées dans un fichier ASCII (voir **Fichier des coordonnées**). Définir les variables **MFIL** et **MCOR**. Appeler le bloc **COOR**.

**Exemple :**

Le fichier des coordonnées des éléments est test.cor , la commande associée est :

```
_MFIL='test'  
_MCOR='.cor'  
COOR[0]
```

---

## Lire les connectivités

Les connectivités sont stockées dans un fichier ASCII (voir **Fichier des connectivités**). Définir les variables **MFIL** et **MELE**. Appeler le bloc **ELEM**.

**Exemple :**

Le fichier des connectivités des éléments est test.ele, la commande associée est :

```
_MFIL='test'  
_MELE='.ele'  
ELEM[0]
```

---

## Lire les propriétés globales

Les propriétés globales sont intégrées à partir du fichier de commandes. Appeler le bloc **PRGL** et remplir la table des propriétés globales qui suit.

**Exemple :**

Pour intégrer les propriétés globales, au nombre de 4 pour fixer les idées, la commande associée est :

```
PRGL[0] (1.e-06, 1., 1., 2.33)
```

## Lire les propriétés nodales

Les propriétés nodales sont stockées dans un fichier ASCII (voir **Fichier des propriétés nodales**). Définir seulement les variables **MFIL** et **MPRN** pour assigner le nom du fichier dans le cas stationnaire. Ajouter au besoin **TASPRN** dans le cas transitoire. Appeler le bloc **PRNO**.

### Exemple :

Le fichier des propriétés nodales stationnaires est test.prn, la commande associée est :

```
_MFIL='test'  
_MPRN='.prn'  
PRNO[0]
```

Dans le cas transitoire nous avons à spécifier le temps si les propriétés nodales varient dans le temps, par exemple charger celles associées à t=24h00 :

```
_MFIL='test'  
_MPRN='.prn'  
_TASPRN=86400.  
PRNO[0]
```

---

## Lire les propriétés élémentaires

Les propriétés élémentaires sont stockées dans un fichier ASCII (voir **Fichier des propriétés élémentaires**). Définir seulement les variables **MFIL** et **MPRE** pour assigner le nom du fichier dans le cas stationnaire. Ajouter au besoin **TASPRE** dans le cas transitoire. Appeler le bloc **PREL**.

### Exemple :

Le fichier des propriétés élémentaires stationnaires est test.pre, la commande associée est :

```
_MFIL='test'  
_MPRE='.pre'  
PREL[0]
```

Dans le cas transitoire nous avons à spécifier le temps si les propriétés élémentaires varient dans le temps, par exemple charger celles associées à t=12h00:

```
_MFIL='test'  
_MPRE=' .pre'  
_TASPRE=43200.  
PREL[0]
```

---

## Lire la solution initiale

Deux cas peuvent se présenter :

- 1 - Les paramètres de la solution initiale sont introduits au prompt ou par le fichier de commandes. Appeler le bloc **INIT(réel,réel,...,réel)** et remplir s'il y a lieu la table des paramètres (nombres réels) d'initialisation qui suit.
- 2 - La solution initiale est stockée dans un fichier (voir **Fichier des degrés de liberté**). Définir les variables **MFIL**, **MINI**, **FFINI** et **TASINI**. Appeler le bloc **INIT(réel,réel,...,réel)**.

### Exemple 1:

L'initialisation est effectuée sans fichier de données et tous les degrés de libertés sont mis à zéro, la commande associée est :

```
INIT[0]
```

Si par contre des paramètres, au nombre de 3 pour fixer les idées, doivent être spécifiés nous avons :

```
INIT[0] (1.0, -2, 100)
```

### Exemple 2:

La solution initiale stationnaire est stockée dans le fichier ASCII test.deb, la commande associée est :

```
_MFIL='test'  
_MINI=' .deb'  
INIT[0]
```

Si par contre le fichier est en format binaire nous avons :



```
_MFIL='test'  
_MINI='.deb'  
_FFINI='BIN'  
INIT[0]
```

Dans le cas transitoire nous avons à spécifier le temps si la solution initiale stockée au format binaire varie dans le temps, par exemple charger celle associée à t=3h00 :

```
_MFIL='test'  
_MINI='.deb'  
_FFINI='BIN'  
_TASINI=10800.  
INIT[0]
```

Remarque : Le format du fichier lu par le bloc INIT est identique à celui généré par le bloc FIN.

---

## Lire les conditions aux limites

Les conditions aux limites sont stockées dans un fichier ASCII (voir **Fichier des conditions aux limites**). Définir seulement les variables **MFIL** et **MCND** pour assigner le nom du fichier dans le cas stationnaire. Ajouter au besoin **TASCND** pour le cas transitoire. Appeler le bloc **COND**.

### Exemple :

Le fichier des conditions aux limites stationnaires est test.prn, la commande associée est :

```
_MFIL='test'  
_MCND='.cnd'  
COND[0]
```

Dans le cas transitoire nous avons à spécifier le temps si les conditions aux limites varient dans le temps, par exemple charger celles associées à t=1h00 :

```
_MFIL='test'
```

```
_MCND=' .cnd'  
_TASCND=3600.  
COND[0]
```

---

## Lire les sollicitations concentrées

Les sollicitations concentrées sont stockées dans un fichier ASCII (voir **Fichier des sollicitations**). Définir seulement les variables **MFIL** et **MSLC** pour assigner le nom du fichier dans le cas stationnaire. Ajouter au besoin **TASSLC** dans le cas transitoire. Appeler le bloc **SOLC**.

**Exemple :**

Le fichier des sollicitations concentrées stationnaires est `test.slc`, la commande associée est :

```
_MFIL='test'  
_MSLC=' .slc'  
SOLC[0]
```

Dans le cas transitoire nous avons à spécifier le temps si les sollicitations concentrées varient dans le temps, par exemple charger celles associées à `t=-0h30` :

```
_MFIL='test'  
_MSLC=' .slc'  
_TASSLC=-1800.  
SOLC[0]
```

---

## Lire les sollicitations réparties

Les sollicitations réparties sont stockées sur fichier ASCII (voir **Fichier des sollicitations**). Définir seulement les variables **MFIL** et **MSLR** pour assigner le nom du fichier dans le cas stationnaire. Ajouter au besoin **TASSLR** dans le cas transitoire. Appeler le bloc **SOLR**.

**Exemple :**

Le fichier des sollicitations réparties stationnaires est `test.slr`, la commande associée est :

```
_MFIL='test'  
_MSLR='.slr'  
SOLR[0]
```

Dans le cas transitoire nous avons à spécifier le temps si les sollicitations réparties varient dans le temps, par exemple charger celles associées à `t=0h00` :

```
_MFIL='test'  
_MSLR='.slr'  
_TASSLR=0.  
SOLC[0]
```

---

## Résoudre le problème

La procédure d'activation de la résolution se fait en deux étapes :

- 1 - Étape obligatoire si **IMPR=3**. Assigner une valeur à **ILU** et à **DELPRT**. Activer le bloc **PRCO**.
- 2 - Assigner une valeur à **TINI**, **NPAS**, **IMPR**, **NPREC**, **NRDEM**, **NITER**, **OMEGA** et **EPSDL**. Appeler le bloc **SOLV(réel,réel,réel)** et remplir si souhaité la table des limiteurs qui suit.

### Exemple 1:

On souhaite effectuer une résolution stationnaire avec une matrice ILU au niveau de remplissage 0, 1 préconditionnement, 10 redémarrages, 25 itérations et une précision de  $10^{-6}$ , la commande associée est :

```
_ILU=0  
PRCO  
_IMPR=3  
_NPREC=1  
_NRDEM=10  
_NITER=25  
_OMEGA=1.0  
_EPSDL=1.e-06
```

```
SOLV[0]
```

**Exemple 2:**

On souhaite effectuer une résolution transitoire par un schéma d'Euler implicite (ALFA=1) pour simuler un processus d'une heure avec un pas de temps d'une minute et en fixant à zéro le temps de départ de la simulation. On emploie une matrice ILU au niveau de remplissage 0, 1 préconditionnement, 10 redémarrages, 25 itérations et une précision de  $10^{-6}$ , la commande associée est :

```
_ILU=0  
PRCO  
_ALFA=1  
_TINI=0  
_DPAS=60  
_NPAS=60  
_IMPR=3  
_NPREC=1  
_NRDEM=10  
_NITER=25  
_OMEGA=1.0  
_EPSDL=1.e-06  
SOLV[0]
```

Remarque : Les solutions intermédiaires ne sont pas accessibles en post-résolution, seule la solution finale peut être exploitée ultérieurement.

---

## Imprimer les résultats

L'opération d'impression de chaque type de résultat consiste à définir le nom du fichier de résultats (voir **Structure du nom des fichiers**) au moyen des variables appropriées et d'activer le bloc pertinent pour procéder à l'impression. Voici les différents types de résultats pouvant être imprimés par DISPERSIM :

- Imprimer les degrés de liberté
- Imprimer l'estimation des erreurs numériques
- Imprimer le post-traitement
- Imprimer les résidus

## Imprimer les degrés de liberté

Les degrés de libertés sont stockés dans un fichier binaire (voir **Fichier des degrés de liberté**). Définir les variables **MFIL** et **MFIL** pour assigner le nom du fichier. Appeler le bloc **FIN**.

### Exemple :

Les degrés de liberté doivent être stockés dans le fichier ASCII `test.fin`, la commande associée est :

```
_MFIL='test'  
_MFIL='.fin'  
FIN[0]
```

ou bien dans un fichier binaire

```
_MFIL='test'  
_MFIL='.fin'  
_FFFIL='BIN'  
FIN[0]
```

Remarque : Le format du fichier généré par le bloc **FIN** est identique à celui lu par le bloc **INIT(réel,réel,...,réel)**.

---

## Imprimer l'estimation des erreurs numériques

Les erreurs numériques sont stockées dans fichier ASCII (voir **Fichier des erreurs numériques**). Définir les variables **MFIL** et **MERR** pour assigner le nom du fichier. Appeler le bloc **ERR**.

### Exemple :

Les erreurs numériques doivent être stockés dans le fichier `test.err`, la commande associée est :

```
_MFIL='test'  
_MERR='.err'  
ERR[0]
```

## Imprimer le post-traitement

Le post-traitement est stocké dans fichier ASCII (voir **Fichier de post-traitement**). Définir les variables **MFIL** et **MPST** pour assigner le nom du fichier. Appeler le bloc **POST**.

### Exemple :

Le post-traitement doit être stocké dans le fichier `test.pst`, la commande associée est :

```
_MFIL='test'  
_MPST='.pst'  
POST[0]
```

---

## Imprimer les résidus

Les résidus sont stockés dans un fichier ASCII (voir **Fichier des résidus**). Définir seulement les variables **MFIL** et **MRES** pour assigner le nom du fichier dans le cas stationnaire. Ajouter **DPAS** et **ALFA** dans le cas transitoire. Appeler le bloc **RESI**.

### Exemple 1:

Les résidus associés à une solution stationnaire doivent être stockés dans le fichier `test.res`, la commande associée est :

```
_MFIL='test'  
_MRES='.res'  
RESI[0]
```

### Exemple 2:

Les résidus associés à une solution transitoire calculée avec un schéma d'Le schéma temporel EULER implicite (**ALFA=1**) et un pas de temps d'une minute, doivent être stockés dans le fichier `test.res`, la commande associée est :

```
_MFIL='test'  
_MRES='.res  
_ALFA=1.0  
_DPAS=60.'
```

RESI[0]

# Chapitre 4 Comment obtenir une solution de transport-diffusion?

Ce chapitre vise à prodiguer quelques précautions à observer pour obtenir une solution de transport-diffusion dans les meilleurs délais. L'expérience nous enseigne qu'un projet de modélisation par éléments finis n'est pas une procédure linéaire mais plutôt itérative qu'on souhaite convergente. Cela à raison, car un exercice de simulation d'un cas réel est une activité complexe qui implique la préparation d'un modèle numérique de terrain, la génération des données hydrodynamiques pour le processus de transport, la détermination des paramètres physiques, tels que les diffusivités et les coefficients des cinétiques de puits et sources, et des conditions aux limites. Ensuite, il faut initialiser le modèle, post-traiter et analyser la solution et calibrer le modèle. L'étape suivante et consiste à valider le modèle. Lorsque ces étapes sont franchies, on peut penser à des exercices de prédiction.

Ici, nous nous concentrons sur certaines dispositions qui méritent d'être consultées pour réaliser efficacement une simulation de transport-diffusion proprement dite:

- Comment valider les données d'entrée (phase préliminaire)?
- Comment initialiser le modèle?
- Comment converger la solution?
- Comment calibrer le modèle?
- Comment valider le modèle ?
- Questions fréquentes

---

## Comment valider les données d'entrée (phase préliminaire)?

Nous entendons par "données d'entrée" celles transmises directement au simulateur dans le fichier de commande. Il ne s'agit pas des données de base de terrain ayant servi à construire le modèle de terrain. Ces données doivent être contrôlées directement sur le ou les logiciels utilisés pour la préparation des données d'entrée de DISPERSIM. En pratique il s'agit de s'assurer que/de:

- 1 - l'élément fini utilisé pour mailler est bien disponible dans la librairie d'éléments de DISPERSIM ;



- 2 - la limite (peau) du maillage éléments fini représente exactement le contour du domaine de simulation ;
- 3 - le maillage recouvre bien l'ensemble du lit de l'écoulement ;
- 4 - la projection des données hydrodynamiques sur le maillage éléments fini n'engendre pas de valeurs corrompues ;
- 5 - lancer une simulation avec DISPERSIM avec pour instructions de lire les données et d'imprimer le post-traitement ;
- 6 - analyser la solution et vérifier si elle respecte bien les conditions aux limites et initiales.

---

## Comment initialiser le modèle?

L'initialisation du modèle vise à calculer la toute première solution sur le domaine de simulation. C'est une étape importante car dans certaines situations cette première solution peut servir de solution initiale pour chercher d'autres scénarios. Cela est particulièrement le cas lorsqu'on cherche à modéliser sur de longs tronçons de rivières le transport d'un contaminant qui obéit à une cinétique complexe. En pratique, c'est par un choix judicieux des conditions aux limites et initiales qu'on procède à l'initialisation du modèle. Voici comment les choisir :

- Scénarios de conditions aux limites
- Scénarios de conditions initiales

---

## Scénarios de conditions aux limites

Les conditions aux limites sont introduites sur le contour du domaine de simulation, en d'autres termes sur la peau du maillage. Physiquement, le contour du domaine est formé d'un ensemble de frontières ouvertes et fermées. Chaque type de frontière nécessite un traitement particulier :

- Frontière fermée
- Frontière ouverte

---

### Frontière fermée

On parle de frontière fermée pour définir une frontière imperméable, l'écoulement normal étant nul. Sur une frontière fermée, on impose généralement une condition de Neuman. Une condition de Neuman stipule que le gradient de concentration est nul dans la direction normale à la frontière. La condition de Neuman sur une frontière fermée est implicite dans DISPERSIM

et n'a donc pas besoin d'être spécifiée explicitement, cette condition est appliquée par défaut.

---

## Frontière ouverte

On désigne par frontière ouverte une frontière par laquelle l'écoulement normal est non nul. De plus, on distingue une frontière ouverte d'entrée c'est à dire une frontière par laquelle un débit non nul transit pour entrer dans le domaine de simulation. À l'opposé, une frontière de sortie est une frontière par laquelle transit un débit non nul pour sortir du domaine de simulation. Pour chacune des deux frontières, il faut imposer une condition spécifique.

- Frontière ouverte d'entrée
  - Frontière ouverte de sortie
- 

### Frontière ouverte d'entrée

Sur la frontière d'entrée, on impose habituellement soit une condition de Dirichlet soit une condition de Cauchy. Une condition de Cauchy a l'avantage d'être moins contraignante pour le processus de résolution.

Une condition de Dirichlet impose de spécifier directement la valeur de la concentration. Une condition de Cauchy permet d'introduire la valeur de la concentration à l'extérieur du domaine vis à vis de la frontière ouverte, valeur que devrait atteindre le degré de liberté impliqué.

#### Exemple :

Le noeud numéro 69 se trouve sur une frontière ouverte d'entrée. La valeur de la concentration est égale à  $0.75 \text{ kg/m}^3$ . Dans le fichier des conditions aux limites on traduit cette information comme condition de Dirichlet de la façon suivante :

```
1000000000  0.75  
69
```

ou bien de Cauchy par :

```
2000000000  0.75  
69
```

Pour plus d'information, consultez les formats **Fichier des conditions aux limites**.

Remarque : Dans le cas où l'écoulement est nul, une condition de Cauchy est identique à une condition de Neuman autrement

dit la dérivée de la concentration le long de la direction normale à la frontière ouverte est nulle.

---

### Frontière ouverte de sortie

Sur la frontière de sortie, on impose habituellement une condition de sortie libre. Autrement dit, on ne suppose rien sur la solution puisqu'aucune valeur n'est spécifiée, le modèle s'ajuste automatiquement pour déterminer la solution à la sortie.

#### Exemple :

Le noeud numéro 69 se trouve sur une frontière ouverte de sortie. Dans le fichier des conditions aux limites on traduit cette information comme condition de sortie libre :

```
3000000000 0.00
```

```
69
```

Pour plus d'information, consultez les formats du **Fichier des conditions aux limites**.

---

## Scénarios de conditions initiales

Un choix judicieux des conditions initiales est très important pour assurer la délicate convergence du processus de résolution. Pour amorcer la résolution il est bien important de démarrer la résolution à partir d'un état d'équilibre. Dans DISPERSIM, deux stratégies d'initialisation sont proposées :

- Solution initiale constante
  - Solution initiale de référence
- 

### Solution initiale constante

En règle générale, on exploite cette procédure d'initialisation lors de la toute première exécution et elle consiste à fixer tous les degrés de liberté du même type à une valeur constante. Les degrés de liberté d'un même noeud pourront avoir des valeurs initiales distinctes. Les valeurs d'initialisation de concentration à choisir sont dépendantes non seulement des conditions aux limites mais également du mode de dispersion.

Pour transporter un panache par exemple, une valeur initiale nulle est appropriée. Si par contre il s'agit de résoudre un problème de diffusion pure, fixer la valeur initiale à une valeur proche de celle appliquée à la limite ouverte d'entrée est un choix opportun. Dans le cas d'une cinétique de dégradation, il

faut tenir compte de la décroissance potentielle de la concentration dans l'adoption de la valeur initiale.

**Exemple :**

Dans le fichier de commande de DISPERSIM, on initialise avec les valeurs C1, C2,..., Cn les degrés de liberté nodaux 1, 2,..., n respectivement d'une solution comme suit :

```
! pas de fichier d'initialisation à lire
!_MINI=''
INIT(C1,C2,...,Cn)
```

si toutes les valeurs C1, C2,..., Cn sont toutes nulles alors il suffit d'écrire :

```
! pas de fichier d'initialisation à lire
!_MINI=''
INIT
```

Remarque : La solution initiale n'a aucune influence sur la convergence si et seulement si on simule la dispersion d'un contaminant conservatif ou qui réagit à une cinétique de dégradation du premier ordre en régime permanent avec une méthode de résolution directe (voir Méthode de résolution).

---

## Solution initiale de référence

La solution initiale provient d'une simulation précédente qu'on réinjecte dans un fichier d'initialisation. Cette forme d'initialisation est pratique lorsqu'on cherche à faire converger la solution en plusieurs étapes. On fait également usage de cette stratégie lorsqu'on poursuit une étude d'impact aux conditions de transport-diffusion, hydrodynamiques, d'aménagements, etc...

**Exemple :**

Dans le fichier de commande cela se traduit comme suit :

```
! solution initiale stockée dans le
!fichier sol-reference.deb en format ASCII
_FFINI='ASCII'
_MINI='sol-reference.deb'
INIT
```

---

## Comment converger la solution?

- Méthode de résolution
- Mise à jour de la solution

- Comportement du solveur
- La convergence est atteinte?
- Conseil pratique

---

## Méthode de résolution

Dans DISPERSIM, la méthode de résolution du système d'équations algébrique est faite par la méthode itérative GMRES non-linéaire, selon un schéma de "Newton-Inexact", avec préconditionnement. Les différents aspects de la méthode sont relatives à :

- Algorithme de résolution par GMRES
- Matrices de préconditionnement
- Espace mémoire
- Précision

---

## Algorithme de résolution par GMRES

Le fonctionnement de l'algorithme de résolution par GMRES est présenté sur la Figure 1. On constate qu'il y a trois boucles. La première est pilotée par la variable **NPREC**, la seconde par la variable **NRDEM** et la troisième par **NITER** laquelle ne peut excéder la valeur de **NDLT**, le nombre de degrés de libertés (variables) total. La variable **NITER** joue un double rôle puisqu'elle permet non seulement de fixer le nombre d'itérations mais également de gouverner la dimension du sous-espace solution égale au produit de **NITER** par **NDLT**.

Formellement, la première boucle est destinée au calcul de la Matrices de préconditionnement, la seconde à la Mise à jour de la solution et la troisième au calcul du sous-espace solution de type Krylov par la méthode GMRES proprement dite.

Les paramètres de fonctionnement sont :

- 1 - le nombre de préconditionnements **NPREC**
- 2 - le nombre de redémarrages **NRDEM**
- 3 - le nombre d'itérations **NITER**

La théorie stipule que pour un problème linéaire, l'algorithme GMRES converge au plus en **NDLT** itérations, ce qui serait tout à fait irréalisable en pratique étant donné l'Espace mémoire exorbitant que cela pourrait supposer pour un problème normal. Dans le cas nonlinéaire par contre, il n'existe pas de méthode pour déterminer les valeurs optimales des paramètres de

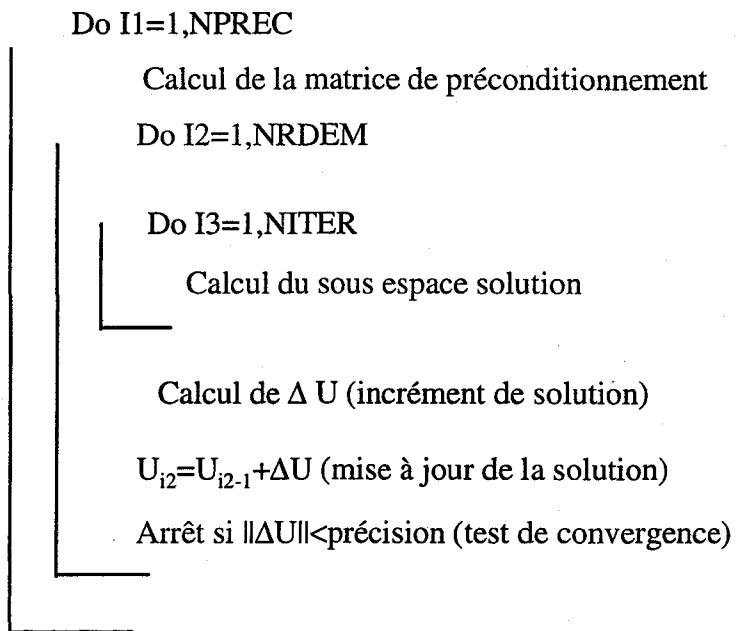
fonctionnement. Toutefois, l'expérience sur une large gamme de problèmes suggère les valeurs par défaut suivantes :

```
_NPREC=1  
_NRDEM=25  
_NITER=25
```

On pourra accroître le nombre de préconditionnements (NPREC) pour des simulations importantes.

Le critère d'arrêt de l'algorithme de résolution est basé sur la norme de l'incrément, c'est-à-dire, le progrès de la solution, qui doit être inférieure à la Précision fixée via la variable **EPSDL**.

**Figure 1 : Algorithme de résolution par GMRES nonlinéaire.**



## Matrices de préconditionnement

La matrice de préconditionnement joue un rôle très important dans l'algorithme de résolution par GMRES. Un bon préconditionneur améliore grandement les performances du solveur. Pour fixer les idées, deux matrices de préconditionnement données peuvent faire converger ou diverger le problème. On parle de divergence de l'algorithme de résolution si la norme de l'incrément tend à augmenter (explose).

Dans le cas des équations de transport-diffusion, l'expérience montre que le préconditionnement matrice ILU, à niveau de remplissage minimum (ILU=0), est efficace. Pour en faire usage, les instructions dans le fichier de commande sont :

```
:  
_ILU=0
```

PRCO

\_IMPR=3

:

Si l'Espace mémoire l'autorise, de meilleurs résultats sont obtenus en choisissant le remplissage maximum (ILU=-1). Dans ce cas, les instructions sont :

:

\_ILU=-1

PRCO

\_IMPR=3

:

Dans le cas contraire, si même une matrice matrice ILU à niveau de remplissage minimum (ILU=0) ne peut être exploitée pour des raisons de capacité mémoire, le compromis à faire est d'avoir recours à un préconditionnement dit diagonal (voir **IMPR**).

Remarque : La performance du préconditionnement ILU est fortement influencée par la numérotation des noeuds du maillage. Pour obtenir les meilleurs résultats, il est conseillé d'utiliser un algorithme de numérotation qui minimise la largeur de bande.

---

## Espace mémoire

Dans l'algorithme de résolution, ce sont la matrice de préconditionnement **ILU** et la dimension du sous-espace solution via **NITER** qui conditionnent l'espace mémoire requis pour la procédure de résolution.

L'espace mémoire à prévoir pour stocker la matrice ILU avec un niveau de remplissage n (ILU=n) est d'environ :

$$\sigma \cdot ((3/2) \cdot NKG + (3 \cdot n + 1) \cdot NDLT + 1/2) \text{ en mégabytes ;}$$

tandis que celui associé au sous espace solution est donné par

$$\sigma \cdot NITER \cdot NDLT$$

où  $\sigma = 8/2^{20}$  est le coefficient de conversion de mots réels à mégabytes et NKG le nombre de termes non nuls dans la matrice ILU. On peut obtenir cette information dans le fichier de sortie associé à la commande :

\_ILU=0

PRCO[0]

---

## Précision

La précision applicable sur le processus itératif de Mise à jour de la solution est définie au moyen de la variable **EPSDL**. Plus la précision est grande, plus le volume de calcul augmente.

Pour la simulation de problèmes académiques, on fixe EPSDL à une valeur de l'ordre de :

`_EPSDL=1.E-10`

Pour la simulation efficace de cas réels, on pose temporairement une valeur de l'ordre de :

`_EPSDL=1.E-03`

et en approchant de la solution, on réduit progressivement la valeur de EPSDL jusqu'à la précision souhaitée qui peut typiquement se situer au voisinage de :

`_EPSDL=1.E-06`

---

## Mise à jour de la solution

Dans l'algorithme de résolution (voir Algorithme de résolution par GMRES), la mise à jour de la solution se fait après chaque redémarrage comme suit :

$$U_{i2}=U_{i2-1}+\Delta U$$

Où  $U_{i2}$  désigne le vecteur solution au (i2) ième redémarrage et  $\Delta U$  l'incrément de solution

Cependant, DISPERSIM propose de moduler la mise à jour de la solution ce qui peut s'avérer fort utile dans le cas où la solution varie beaucoup et/ou les non-linéarités sont importantes. Le calcul de l'incrément  $\Delta U$  de la solution est alors fait comme suit :

$$\Delta U = \text{signe}(\Delta U) \max(\Delta U_{\max}, |\omega \Delta U|)$$

où  $\Delta U_{\max}$  est le limiteur. Un limiteur est associé à chaque type de variable (degré de liberté).

Ainsi deux possibilités sont offertes pour intervenir lors de la mise à jour de la solution :

- Relaxer la mise à jour de la solution
- Limiter la mise à jour de la solution

---

## Relaxer la mise à jour de la solution

Relaxer la solution signifie "atténuer" l'amplitude de l'incrément  $\Delta U$ . La procédure de mise à jour de la solution est faite comme suit :

$$U_{i2}=U_{i2-1}+\omega \Delta U$$

où  $\omega$  est le facteur de relaxation.

**Exemple :**

On veut sous-relaxer la solution avec facteur de 0.8 :



`_OMEGA=0.8`

En pratique, la sous-relaxation ( $\omega < 1$ ) permet au solveur de mieux converger dans le cas où le problème est difficile. En contrepartie, elle ralentit le processus de résolution même au voisinage de la solution.

---

## Limiter la mise à jour de la solution

La technique consiste à imposer une limite absolue sur la valeur de l'incrément  $\Delta U$ .

### Exemple :

On veut limiter la mise à jour  $\Delta U$  de la solution avec une valeur  $\Delta U_{\max}$  de 0.1 sur la concentration ce qui se traduit dans le fichier de commande comme suit en supposant un (01) degré de liberté par noeud :

```
SOLV[0] (0.1)
```

En pratique, le limiteur n'intervient dans le processus de mise à jour de la solution que lorsque l'incrément  $\Delta U$  devient supérieur au limiteur  $\Delta U_{\max}$ . En outre, au voisinage de la solution le processus de résolution n'est pas ralenti.

La limitation de la mise à jour de la solution peut se faire de concert avec la relaxation.

---

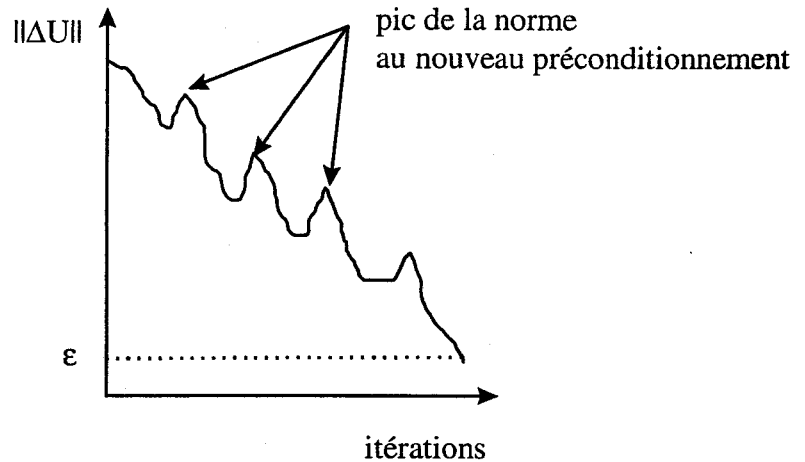
## Comportement du solveur

Sur la Figure 2 est représentée la courbe de convergence typique de GMRES associé à un préconditionnement de type ILU.

En règle générale, on observe un pic de la norme de  $\Delta U$  à chaque mise à jour de la matrice ILU, autrement dit à chaque nouveau préconditionnement.

La convergence du problème est atteinte selon la précision  $\varepsilon$  retenue si au préconditionnement suivant le pic de la norme reste inférieur à  $\varepsilon$  (voir La convergence est atteinte?).

**Figure 2 : Comportement caractéristique de la norme de l'incrément de solution  $\Delta U$  en fonction des itérations dans le cas de GMRES avec la matrice ILU comme préconditionneur.**

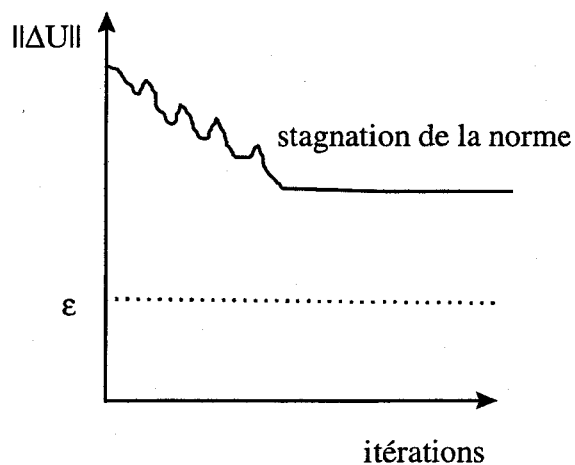


Dans certaines situations, il peut arriver que GMRES stagne ou oscille (Figure 3). Le premier remède possible est d'augmenter le nombre d'itérations **NITER**. En effet, ce paramètre n'a pas seulement pour effet de prolonger la boucle de calcul, il permet aussi d'enrichir l'espace de recherche (la dimension du sous-espace solution).

Si le résultat reste inchangé faire une tentative en modifiant la diffusivité vers le haut d'abord puis vers le bas si nécessaire.

Si le problème persiste, il est probable que la solution ne peut être trouvée à cause d'une carence de discrétisation quelque part dans le domaine de simulation. Il faut alors identifier visuellement la zone qui pose problème (zone à résidu élevé). Ensuite, il faut raffiner localement le maillage à cet endroit et reprendre l'exercice de simulation.

**Figure 3 : Stagnation du processus de convergence de la norme.**



## La convergence est atteinte?

Pour déclarer que la convergence de la solution calculée est atteinte, au sens numérique, il faut s'assurer que :

- 1 - GMRES a convergé ;
- 2 - La norme du résidu associé à chaque degré de liberté est au plus du même ordre que  $\epsilon$ .

Si convergence il y a, la simulation est terminée.

Dans le cas d'une convergence locale seulement, il faut reprendre la simulation en partant de la dernière solution et en augmentant le nombre d'itérations **NITER**. Répéter cette procédure au besoin.

Si le problème persiste analyser visuellement les résultats et deux scénarios sont possibles :

- 1 - Les résultats sont valides dans l'ensemble. Dans ce cas isoler les zones à résidus élevés. Densifier le maillage dans ces zones. Reprendre les simulations sur le nouveau maillage. Répéter cette procédure au besoin.
- 2 - Localement, les résultats sont mauvais (pollution de la solution par des oscillations parasites). Dans ce cas, il faut reprendre la simulation avec une initialisation à froid:
  - a) en augmentant soit la diffusivité moléculaire soit le coefficient de pondération de la diffusivité longitudinale si la zone problématique est localisée dans le lit de l'écoulement.
  - b) en diminuant soit la diffusivité moléculaire soit le coefficient de pondération de la diffusivité longitudinale si la zone problématique est localisée près d'une rive.

---

## Conseil pratique

Ce conseil vise les simulations de longue durée, pendant toute une nuit, par exemple. Pour se mettre à l'abri des imprévus et pour ne pas perdre le bénéfice de plusieurs heures de calculs il est fortement conseillé d'imprimer périodiquement les degrés de liberté et mieux encore, dans des fichiers différents si l'espace disque est disponible. Ainsi, après une panne de courant, par exemple, il sera possible de repartir la simulation à partir de la plus récente solution obtenue ou acceptable.

### Exemple :

Pour un même volume de calcul, au lieu de la commande 1 :

```
!---- commande 1 : impression unique des ddl
```

```
!      dans test.fin
_NPREC=3
SOLV[0] (0.25,0.25,0.1)
_MFIL='test'
_MFIN='.fin'
FIN[0]
```

utilisez plutôt :

```
!---- commande 2 : impression périodique des ddl
!      dans test.fin#i.
_MFIL='test'
_NPREC=1
SOLV[0] (0.25,0.25,0.1)
_MFIN='.fin#1'
FIN[0]
SOLV[0] (0.25,0.25,0.1)
_MFIN='.fin#2'
FIN[0]
SOLV[0] (0.25,0.25,0.1)
_MFIN='.fin#3'
FIN[0]
```

---

## Comment calibrer le modèle?

Une solution convergée n'est pas nécessairement acceptable d'un point de vue physique; autrement dit, la solution calculée peut manquer d'objectivité. À partir d'un scénario réel, l'exercice de calibration vise à vérifier le degré de similitude entre les observations sur le terrain et les résultats de simulation dans les mêmes conditions de transport-diffusion. Si la solution d'initialisation ainsi calculée est jugée acceptable en première approximation, on peut passer à l'étape de validation (voir Comment valider le modèle ?)

Dans le cas où la solution d'initialisation ne semble pas acceptable pour diverses raisons, panache irréaliste, niveau de concentration exagérément haut ou bas, oscillations parasites ou valeurs extrêmes des concentrations par exemple, il faut poursuivre l'exercice d'initialisation jusqu'à satisfaction par divers ajustements à apporter soit au maillage soit aux paramètres du modèle. Le raffinement de maillage devra être effectué dans les zones du domaine qui se distinguent par des erreurs d'approximation relativement importantes (voir Contrôle des

erreurs d'approximation) et également au voisinage des rives (Influence des profondeurs négatives ou nulles).

Si après avoir apporté les corrections nécessaire, le biais systématique et inacceptable persiste il faut à ce moment questionner les données de terrain d'entrée.

---

## Contrôle des erreurs d'approximation

Le bloc d'estimation des erreurs d'approximation quant à lui indique si potentiellement en raffinant le maillage dans les zones à niveau d'erreur élevé la solution est susceptible de changer. Les unités des erreurs sont identiques à celles des variables auxquelles elles sont associées. En règle générale, les erreurs sont les plus importantes dans les zones de fortes variations hydrodynamiques, autour des points d'injection de concentration, au voisinage d'un front ou un panache de concentration etc...

---

## Influence des profondeurs négatives ou nulles

Dans les zones sèches, c'est dire les zones à profondeurs négatives ou nulles la concentration est imposée automatiquement à zéro comme condition de Dirichlet.

Les zones situées le long des rives, qu'on appelle également zones de transition, sont traitées différemment. La concentration n'est pas imposée à zéro comme dans les zones sèches. C'est une condition de Neuman qui est imposée automatiquement de façon à éviter les gradients fictifs de concentration d'origines numérique. Il peut alors arriver que sur un noeud sec on observe une concentration non nulle. Pour améliorer les choses si nécessaire, modifier le maillage en ne conservant que les zones mouillées.

---

## Comment valider le modèle ?

Cette étape vient en principe après l'étape de calibration. Elle consiste à reprendre les paramètres qui ont servi à calibrer le modèle pour reproduire, avec une hydrodynamique pratiquement inchangée, un scénario distinct de transport-diffusion et de s'assurer que les résultats de simulation sont conformes avec ceux observés sur le terrain.

Remarque : Si aucun scénario réel, hormis celui exploité pour la calibration, n'est disponible on ne peut, à ce moment là, dire que le modèle a été validé.

## Questions fréquentes

- Comment évaluer les diffusivités?
- Quels sont les effets d'une diffusivité excessive?
- La résolution a-t-elle divergé?
- Comment atténuer les oscillations parasites?
- Comment minimiser le niveau d'erreur?

---

### Comment évaluer les diffusivités?

L'évaluation des diffusivités est toujours un exercice délicat puisque la théorie sur la turbulence est encore un domaine de recherche ouvert et très actif. Il reste que le logiciel DISPERSIM évalue le tenseur symétrique des diffusivités à partir d'une combinaison algébrique mettant en relation les diffusivités verticale, horizontale (toutes deux dépendantes de l'hydrodynamique) et moléculaire ainsi que la direction de l'écoulement.

La diffusivité verticale ou de fond est évaluée à l'aide de la formule de Taylor ; elle est proportionnelle à la vitesse de cisaillement au fond. La diffusivité horizontale est quant à elle basée sur le concept de longueur de mélange qui tient compte de la variabilité locale de l'écoulement.

En outre, la difficulté dans la détermination du tenseur de diffusivité est donc transférée vers l'évaluation de la diffusivité moléculaire et des différents coefficients de pondération :

- 1 - coefficient de pondération de la diffusivité verticale ;
- 2 - coefficient de pondération de la diffusivité horizontale ;
- 3 - coefficient de pondération de la diffusivité longitudinale.

En pratique, l'ajustement des coefficients de pondération s'inscrit dans le processus de calibration du modèle.

---

### Quels sont les effets d'une diffusivité excessive?

Le phénomène de dissipation excessive est occasionné par une surévaluation des diffusivités. Une dissipation excessive masque le processus de transport au profit de celui de la diffusion. Dans ce cas on change la physique du problème. En pratique, une

dissipation excessive a tendance à étaler le panache de concentration et à diluer les concentrations.

---

## La résolution a-t-elle divergé?

Le processus de résolution a divergé si on observe tous les phénomènes suivants :

- 1 - la norme de l'incrément de solution tend à augmenter et même, explose ;
- 2 - la norme du résidu explose ;
- 3 - dans tous le domaine ou localement, on note de fortes oscillations parasites de la solution.

---

## Comment atténuer les oscillations parasites?

En règle générale, les oscillations parasites non physiques sont d'origine purement numériques. Elle sont causées par un maillage inadéquat pour les conditions aux limites exploitées. C'est précisément le cas lorsqu'on exploite une condition de Dirichlet au lieu d'une condition de Cauchy sur un maillage grossier. Dans ces conditions le remède consiste à raffiner le maillage près des frontières ainsi qu'autour des points d'injection ponctuels.

Une autre source d'apparition d'oscillations est la modélisation du transport à haut nombre de Peclet. Dans ce cas les oscillations, même sur un maillage fin, sont dues au fait qu'une solution lisse n'existe pas. Le seul moyen de lisser la solution est de rajouter de la dissipation ou de raffiner le maillage.

---

## Comment minimiser le niveau d'erreur?

En supposant le modèle bien paramétrisé, les erreurs entachant la solution sont directement liées à la variabilité des données hydrodynamiques et de la géométrie du domaine modélisé. La procédure de raffinement du maillage vise donc à minimiser le niveau d'erreur à une juste proportion dans la mesure où des données de terrain existent en quantité suffisante pour le supporter.

Pour éviter de raffiner le maillage inutilement, ce qui occasionnerait des temps de calculs prohibitifs, il est

recommandé d'intervenir localement là où le niveau d'erreur est jugé trop important. Les zones problématiques peuvent être identifiées à l'analyse des cartes des résidus et des erreurs d'approximations numériques.



## Chapitre 5 Annexe

- Dictionnaire de langue
- Librairie d'éléments finis
- Librairie de schémas temporels
- Traitement des données transitoires
- Dépendances
- Exemple de fichier de commandes
- Formats des fichiers d'entrée
- Formats des fichiers de résultats

---

### Dictionnaire de langue

Le dictionnaire de langue de DISPERSIM propose trois (3) langues de travail :

- DISPERSIM en version anglaise
- DISPERSIM en version espagnole
- DISPERSIM en version française

---

### DISPERSIM en version anglaise

Pour l'usage de l'anglais (ENGLISH) définir la variable LANGUE à :

```
_LANGUE=' .eng'
```

---

### DISPERSIM en version espagnole

Pour l'usage de l'ESPagnol définir la variable LANGUE à :

```
_LANGUE=' .esp'
```

## DISPERSIM en version française

Pour l'usage du Français définir la variable LANGUE à :

`_LANGUE=' .frc'`

---

### Librairie d'éléments finis

La librairie d'éléments finis de DISPERSIM propose six (6) types d'éléments finis :

- Élément fini CD2DNC\_CCN
  - Élément fini CD2DNC\_CLF
  - Élément fini CD2DNC\_DBO
  - Élément fini CD2DNC\_MES
  - Élément fini CD2DNC\_MET
  - Élément fini CD2DNC\_TOX
- 

### Élément fini CD2DNC\_CCN

Baptisé CD2DNC\_CCN en référence aux équations de Convection-Diffusion à 2 dimensions sous forme Non Conservative de Contaminants CoNservatifs.

- Comment atteindre l'élément CD2DNC\_CCN?
  - La fonction de l'élément CD2DNC\_CCN
  - Les propriétés de l'élément CD2DNC\_CCN
  - Les conditions aux limites de l'élément CD2DNC\_CCN
  - Les sollicitations de l'élément CD2DNC\_CCN
  - Solution initiale de l'élément CD2DNC\_CCN
- 

#### Comment atteindre l'élément CD2DNC\_CCN?

Pour atteindre l'élément fini CD2DNC\_CCN, dans le fichier de commandes la variable ELTYP doit être définie comme suit :

`_ELTYP='CD2DNC_CCN'`

---

#### La fonction de l'élément CD2DNC\_CCN

CD2DNC\_CCN est un élément fini triangulaire linéaire à trois (3) noeuds, un sur chaque sommet. Il est utilisé pour la discrétisation des équations bidimensionnelles de convection-diffusion sous forme non conservative sans puits ni source. Il est dédié à l'étude de la dispersion de contaminants conservatifs en milieu fluvial et estuarien. Il admet un (1) degré de liberté par noeud, la concentration  $C(\text{kg/m}^3)$ .

Remarque : L'élément CD2DNC\_CCN peut être exploité également pour la simulation d'un champ de température en milieu fluvial. Dans ce cas, la variable d'état n'est plus la concentration mais la température  $T(^{\circ}\text{C})$ .

---

## Les propriétés de l'élément CD2DNC\_CCN

- Propriétés géométriques de l'élément CD2DNC\_CCN
  - Propriétés globales de l'élément CD2DNC\_CCN
  - Propriétés élémentaires de l'élément CD2DNC\_CCN
  - Propriétés nodales de l'élément CD2DNC\_CCN
- 

### Propriétés géométriques de l'élément CD2DNC\_CCN

Nombre de dimensions NDIM=2 ;

Nombre de noeuds par élément NNEL=3.

---

### Propriétés globales de l'élément CD2DNC\_CCN

Nombre de propriétés globales à lire NPRGL=4 :

- 1 - diffusivité moléculaire ( $\text{m}^2/\text{s}$ ) ;
  - 2 - coefficient de pondération de la diffusivité verticale ;
  - 3 - coefficient de pondération de la diffusivité horizontale ;
  - 4 - coefficient de pondération de la diffusivité longitudinale.
- 

### Propriétés élémentaires de l'élément CD2DNC\_CCN

Aucune propriété élémentaire à lire NPREL=0.

---

### Propriétés nodales de l'élément CD2DNC\_CCN

Nombre de propriétés par noeud à lire NPRNL=5 :

- 1 - composante de vitesse suivant x ( $\text{m/s}$ ) ;
  - 2 - composante de vitesse suivant y ( $\text{m/s}$ ) ;
  - 3 - profondeur de l'écoulement (m) ;
  - 4 - diffusivité verticale ( $\text{m}^2/\text{s}$ ) ;
-

5 - diffusivité horizontale ( $m^2/s$ ).

---

## Les conditions aux limites de l'élément CD2DNC\_CCN

Les conditions aux limites (C.L.) sont appliquées sur les degrés de liberté au contour du domaine. Le nombre de degré de liberté par noeud  $NDLN=1$  ; la concentration  $C(kg/m^3)$ . L'application des conditions aux limites dépend du type de frontière. On en dénombre trois (3) :

- Condition de frontière fermée de l'élément CD2DNC\_CCN
  - Condition de frontière ouverte d'entrée de l'élément CD2DNC\_CCN
  - Condition de frontière ouverte de sortie de l'élément CD2DNC\_CCN
  - Codes des conditions aux limites de l'élément CD2DNC\_CCN
- 

### Condition de frontière fermée de l'élément CD2DNC\_CCN

On parle de frontière fermée pour définir une frontière imperméable, l'écoulement normal étant nul. Sur une frontière fermée, on impose généralement une condition de Neuman. Une condition de Neuman stipule que le gradient de concentration est nul dans la direction normale à la frontière. Cette condition est appliquée par défaut.

---

### Condition de frontière ouverte d'entrée de l'élément CD2DNC\_CCN

On parle de frontière ouverte d'entrée lorsque l'écoulement passe au travers pour entrer dans le domaine. Sur une frontière ouverte d'entrée on impose soit une condition de Dirichlet (la valeur de la concentration est imposée directement) soit une condition de Cauchy (la valeur de la concentration va tendre vers la valeur imposée). D'un point de vue numérique, il est parfois plus souhaitable d'exploiter une condition de Cauchy de façon à atténuer l'amplitude des oscillations parasites de concentration qui polluent la solution (voir **Comment atténuer les oscillations parasites?**).

---

### Condition de frontière ouverte de sortie de l'élément CD2DNC\_CCN

On parle de frontière ouverte de sortie lorsque l'écoulement passe au travers pour sortir du domaine. Sur une frontière ouverte de sortie on impose soit une condition de sortie libre soit une condition de Neuman, prise en compte par défaut, qui stipule que le gradient de concentration est nul dans la direction normale à la frontière. D'un point de vue physique, il est parfois plus souhaitable d'exploiter une condition de sortie libre plus adéquate qu'une condition de Neuman.

## Codes des conditions aux limites de l'élément CD2DNC\_CCN

La convention de code permettant d'introduire les C.L. est (voir Fichier des conditions aux limites) :

- 1 - 1000000000 pour une condition de Dirichlet ;
- 2 - 2000000000 pour une condition de Cauchy ;
- 3 - 3000000000 pour une condition de sortie libre.

Remarques :

Si aucune condition n'est posée sur une frontière une condition de type Neuman sera imposée par défaut par le simulateur.

Dans les zones sèches à profondeur nulles ou négatives, une condition de Dirichlet est appliquée automatiquement avec une valeur de concentration imposée à zéro.

Dans les zones de transition, une condition de Neuman est appliquée automatiquement ce qui permet d'éviter les effets indésirables de couche limite de concentration au voisinage des rives.

---

## Les sollicitations de l'élément CD2DNC\_CCN

- Sollicitations concentrées de l'élément CD2DNC\_CCN
- Sollicitations réparties de l'élément CD2DNC\_CCN

---

### Sollicitations concentrées de l'élément CD2DNC\_CCN

Les sollicitations concentrées servent à introduire les charges réparties équivalentes résultant de la discrétisation par éléments finis du produit  $QHC^0$ , où  $Q(m^3/s)$  est le débit injecté dans le domaine,  $H(m)$  la profondeur et  $C^0(kg/m^3)$  la concentration injectée dans le domaine modélisé. La convention de signe suppose positif (+) un débit entrant et négatif (-) un débit sortant du domaine. Les sollicitations concentrées peuvent être appliquées en n'importe quel point du domaine. En pratique, elles servent à introduire les sources d'injection ponctuelles dans le modèle. Dans le fichier des sollicitations concentrées, le produit  $QHC^0$  est associé à chaque degré de liberté nodal (voir Fichier des sollicitations).

---

### Sollicitations réparties de l'élément CD2DNC\_CCN

Les sollicitations réparties servent à introduire le débit  $Q(m^3/s)$  à la concentration du milieu injecté dans le domaine modélisé. La convention de signe suppose positif (+) un débit entrant et négatif (-) un débit sortant du domaine. Les sollicitations réparties peuvent être appliquées en n'importe quel point du domaine. En pratique, elles servent à introduire le débit des sources et puits d'eau ponctuels dans le modèle. Dans le fichier

des sollicitations réparties le débit Q est associé à chaque degré de liberté (voir Fichier des sollicitations).

---

## Solution initiale de l'élément CD2DNC\_CCN

Nombre de termes d'initialisation NTINI=1 :

1 - valeur de la concentration constante dans tous le domaine.

---

## Élément fini CD2DNC\_CLF

Baptisé CD2DNC\_CLF en référence aux équations de Convection-Diffusion à 2 Dimensions sous forme Non Conservative des Coliformes Fécaux.

- Comment atteindre l'élément CD2DNC\_CLF?
  - La fonction de l'élément CD2DNC\_CLF
  - Les propriétés de l'élément CD2DNC\_CLF
  - Les conditions aux limites de l'élément CD2DNC\_CLF
  - Les sollicitations de l'élément CD2DNC\_CLF
  - Solution initiale de l'élément CD2DNC\_CLF
- 

## Comment atteindre l'élément CD2DNC\_CLF?

Pour atteindre l'élément fini CD2DNC\_CLF, dans le fichier de commandes la variable ELTYP doit être définie comme suit :

```
_ELTYP='CD2DNC_CLF'
```

---

## La fonction de l'élément CD2DNC\_CLF

CD2DNC\_CLF est un élément fini triangulaire linéaire à trois (3) noeuds, un sur chaque sommet. Il est utilisé pour la discrétisation des équations bidimensionnelles de convection-diffusion sous forme non conservative avec une cinétique de dégradation du premier ordre. Il est dédié à l'étude de la contamination bactérienne du milieu fluvial et estuarien. Il admet un (1) degré de liberté par noeud, la concentration C(kg/m<sup>3</sup>) en coliformes fécaux variable représentative de la contamination bactérienne.

---

## Les propriétés de l'élément CD2DNC\_CLF

- Propriétés géométriques de l'élément CD2DNC\_CLF
  - Propriétés globales de l'élément CD2DNC\_CLF
-

- propriétés élémentaires de l'élément CD2DNC\_CLF
  - Propriétés nodales de l'élément CD2DNC\_CLF
- 

### Propriétés géométriques de l'élément CD2DNC\_CLF

Nombre de dimensions NDIM=2 ;

Nombre de noeuds par élément NNEL=3.

---

### Propriétés globales de l'élément CD2DNC\_CLF

Nombre de propriétés globales à lire NPRGL=5 :

- 1 - diffusivité moléculaire ( $m^2/s$ ) ;
  - 2 - coefficient de pondération de la diffusivité verticale ;
  - 3 - coefficient de pondération de la diffusivité horizontale ;
  - 4 - coefficient de pondération de la diffusivité longitudinale ;
  - 5 - coefficient de dégradation des coliformes fécaux ( $s^{-1}$ ).
- 

### propriétés élémentaires de l'élément CD2DNC\_CLF

Aucune propriété élémentaire à lire NPREL=0.

---

### Propriétés nodales de l'élément CD2DNC\_CLF

Nombre de propriétés par noeud à lire NPRNL=5 :

- 1 - composante de vitesse suivant x ( $m/s$ ) ;
  - 2 - composante de vitesse suivant y ( $m/s$ ) ;
  - 3 - profondeur de l'écoulement (m) ;
  - 4 - diffusivité verticale ( $m^2/s$ ) ;
  - 5 - diffusivité horizontale ( $m^2/s$ ).
- 

### Les conditions aux limites de l'élément CD2DNC\_CLF

Les conditions aux limites (C.L.) sont appliquées sur les degrés de liberté au contour du domaine. Le nombre de degré de liberté par noeud NDLN=1 ; la concentration C( $kg/m^3$ ). L'application des conditions aux limites dépend du type de frontière. On en dénombre trois (3) :

- Condition de frontière fermée de l'élément CD2DNC\_CLF
- Condition de frontière ouverte d'entrée de l'élément CD2DNC\_CLF
- Condition de frontière ouverte de sortie de l'élément CD2DNC\_CLF

- Codes des conditions aux limites de l'élément CD2DNC\_CLF

---

### Condition de frontière fermée de l'élément CD2DNC\_CLF

On parle de frontière fermée pour définir une frontière imperméable, l'écoulement normal étant nul. Sur une frontière fermée, on impose généralement une condition de Neuman. Une condition de Neuman stipule que le gradient de concentration est nul dans la direction normale à la frontière. Cette condition est appliquée par défaut.

---

### Condition de frontière ouverte d'entrée de l'élément CD2DNC\_CLF

On parle de frontière ouverte d'entrée lorsque l'écoulement passe au travers pour entrer dans le domaine. Sur une frontière ouverte d'entrée, on impose soit une condition de Dirichlet (la valeur de la concentration est imposée directement) soit une condition de Cauchy (la valeur de la concentration va tendre vers la valeur imposée). D'un point de vue numérique, il est parfois plus souhaitable d'exploiter une condition de Cauchy de façon à atténuer l'amplitude des oscillations parasites de concentration qui polluent la solution (voir **Comment atténuer les oscillations parasites?**).

---

### Condition de frontière ouverte de sortie de l'élément CD2DNC\_CLF

On parle de frontière ouverte de sortie lorsque l'écoulement passe au travers pour sortir du domaine. Sur une frontière ouverte de sortie, on impose soit une condition de sortie libre soit une condition de Neuman, prise en compte par défaut, qui stipule que le gradient de concentration est nul dans la direction normale à la frontière. D'un point de vue physique, il est parfois plus souhaitable d'exploiter une condition de sortie libre plus adéquate qu'une condition de Neuman.

---

### Codes des conditions aux limites de l'élément CD2DNC\_CLF

La convention de code permettant d'introduire les C.L. est (voir Fichier des conditions aux limites) :

- 1 - 1000000000 pour une condition de Dirichlet ;
- 2 - 2000000000 pour une condition de Cauchy ;
- 3 - 3000000000 pour une condition de sortie libre.

Remarques :

Si aucune condition n'est posée sur une frontière, une condition de type Neuman sera imposée par défaut par le simulateur.

Dans les zones sèches à profondeur nulles ou négatives, une condition de Dirichlet est appliquée automatiquement avec une valeur de concentration imposée à zéro.



Dans les zones de transition, une condition de Neuman est appliquée automatiquement ce qui permet d'éviter les effets indésirables de couche limite de concentration au voisinage des rives.

---

## Les sollicitations de l'élément CD2DNC\_CLF

- Sollicitations concentrées de l'élément CD2DNC\_CLF
- Sollicitations réparties de l'élément CD2DNC\_CLF

---

### Sollicitations concentrées de l'élément CD2DNC\_CLF

Les sollicitations concentrées servent à introduire les charges réparties équivalentes résultant de la discrétisation par éléments finis du produit  $QHC^0$ , où  $Q(m^3/s)$  est le débit injecté dans le domaine,  $H(m)$  la profondeur et  $C^0(kg/m^3)$  la concentration injectée dans le domaine modélisé. La convention de signe suppose positif (+) un débit entrant et négatif (-) un débit sortant du domaine. Les sollicitations concentrées peuvent être appliquées en n'importe quel point du domaine. En pratique, elles servent à introduire les sources d'injection ponctuelles dans le modèle. Dans le fichier des sollicitations concentrées, le produit  $QHC^0$  est associé à chaque degré de liberté nodal (voir Fichier des sollicitations).

---

### Sollicitations réparties de l'élément CD2DNC\_CLF

Les sollicitations réparties servent à introduire le débit  $Q(m^3/s)$  à la concentration du milieu injecté dans le domaine modélisé. La convention de signe suppose positif (+) un débit entrant et négatif (-) un débit sortant du domaine. Les sollicitations réparties peuvent être appliquées en n'importe quel point du domaine. En pratique, elles servent à introduire le débit des sources et puits d'eau ponctuels dans le modèle. Dans le fichier des sollicitations réparties, le débit  $Q$  est associé à chaque degré de liberté (voir Fichier des sollicitations).

---

### Solution initiale de l'élément CD2DNC\_CLF

Nombre de termes d'initialisation NTINI=1 :

1 - valeur de la concentration constante dans tous le domaine.

---

## Élément fini CD2DNC\_DBO

Baptisé CD2DNC\_DBO en référence aux équations de Convection-Diffusion à 2 Dimensions sous forme Non Conservative de la Demande Biochimique en oxygène et de l'Oxygène dissous.

- Comment atteindre l'élément CD2DNC\_DBO?
- La fonction de l'élément CD2DNC\_DBO
- Les propriétés de l'élément CD2DNC\_DBO
- Les conditions aux limites de l'élément CD2DNC\_DBO
- Les sollicitations de l'élément CD2DNC\_DBO
- Solution initiale de l'élément CD2DNC\_DBO

---

## Comment atteindre l'élément CD2DNC\_DBO?

Pour atteindre l'élément fini CD2DNC\_DBO, dans le fichier de commandes la variable ELTYP doit être définie comme suit :

```
_ELTYP='CD2DNC_DBO'
```

---

## La fonction de l'élément CD2DNC\_DBO

CD2DNC\_DBO est un élément fini triangulaire linéaire à trois (3) noeuds, un sur chaque sommet. Il est utilisé pour la discrétisation des équations bidimensionnelles de convection-diffusion sous forme non conservative avec une cinétique de dégradation du premier ordre. Il est dédié à l'étude de la consommation d'oxygène dissous dans le milieu fluvial et estuarien. Il admet deux (2) degrés de liberté par noeud, dans l'ordre la concentration d'oxygène dissous et la Demande Biochimique en Oxygène (DBO).

---

## Les propriétés de l'élément CD2DNC\_DBO

- Propriétés géométriques de l'élément CD2DNC\_DBO
- Propriétés globales de l'élément CD2DNC\_DBO
- Propriétés élémentaires de l'élément CD2DNC\_DBO
- Propriétés nodales de l'élément CD2DNC\_DBO

---

### Propriétés géométriques de l'élément CD2DNC\_DBO

Nombre de dimensions NDIM=2 ;

Nombre de noeuds par élément NNEL=3.

---

### Propriétés globales de l'élément CD2DNC\_DBO

Nombre de propriétés globales à lire NPRGL=9 :

1 - diffusivité moléculaire ( $m^2/s$ ) ;

- 2 - coefficient de pondération de la diffusivité verticale ;
- 3 - coefficient de pondération de la diffusivité horizontale ;
- 4 - coefficient de pondération de la diffusivité longitudinale ;
- 5 - température (°C) ;
- 6 - coefficient de réaération atmosphérique (s<sup>-1</sup>) ;
- 7 - coefficient de dégradation de la DBO soluble (s<sup>-1</sup>) ;
- 8 - fraction particulaire de la DBO ;
- 9 - concentration d'oxygène à saturation calculée à partir de la température si égale zéro (kg/m<sup>3</sup>).

---

### Propriétés élémentaires de l'élément CD2DNC\_DBO

Aucune propriété élémentaire à lire NPREL=0.

---

### Propriétés nodales de l'élément CD2DNC\_DBO

Nombre de propriétés par noeud à lire NPRNL=7 :

- 1 - composante de vitesse suivant x (m/s) ;
- 2 - composante de vitesse suivant y (m/s) ;
- 3 - profondeur de l'écoulement (m) ;
- 4 - diffusivité verticale (m<sup>2</sup>/s) ;
- 5 - diffusivité horizontale (m<sup>2</sup>/s) ;
- 6 - vitesses de sédimentation des matières en suspension (m/s) ;
- 7 - demande benthique (kg/m<sup>2</sup>/s).

---

### Les conditions aux limites de l'élément CD2DNC\_DBO

Les conditions aux limites (C.L.) sont appliquées sur les degrés de liberté au contour du domaine. Le nombre de degré de liberté par noeud NDLN=1 ; la concentration C(kg/m<sup>3</sup>). L'application des conditions aux limites dépend du type de frontière. On en dénombre trois (3) :

- Condition de frontière fermée de l'élément CD2DNC\_DBO
- Condition de frontière ouverte d'entrée de l'élément CD2DNC\_DBO
- Condition de frontière ouverte de sortie de l'élément CD2DNC\_DBO
- Codes de conditions aux limites de l'élément CD2DNC\_DBO

---

### Condition de frontière fermée de l'élément CD2DNC\_DBO

On parle de frontière fermée pour définir une frontière imperméable, l'écoulement normal étant nul. Sur une frontière fermée, on impose généralement une condition de Neuman. Une condition de Neuman stipule que le gradient de concentration est nul dans la direction normale à la frontière. Cette condition est appliquée par défaut.

---

#### Condition de frontière ouverte d'entrée de l'élément CD2DNC\_DBO

On parle de frontière ouverte d'entrée lorsque l'écoulement passe au travers pour entrer dans le domaine. Sur une frontière ouverte d'entrée, on impose soit une condition de Dirichlet (la valeur de la concentration est imposée directement) soit une condition de Cauchy (la valeur de la concentration va tendre vers la valeur imposée). D'un point de vue numérique, il est parfois plus souhaitable d'exploiter une condition de Cauchy de façon à atténuer l'amplitude des oscillations parasites de concentration qui polluent la solution (voir **Comment atténuer les oscillations parasites?**).

---

#### Condition de frontière ouverte de sortie de l'élément CD2DNC\_DBO

On parle de frontière ouverte de sortie lorsque l'écoulement passe au travers pour sortir du domaine. Sur une frontière ouverte de sortie, on impose soit une condition de sortie libre soit une condition de Neuman, prise en compte par défaut, qui stipule que le gradient de concentration est nul dans la direction normale à la frontière. D'un point de vue physique, il est parfois plus souhaitable d'exploiter une condition de sortie libre plus adéquate qu'une condition de Neuman.

---

#### Codes de conditions aux limites de l'élément CD2DNC\_DBO

La convention de code permettant d'introduire les C.L. est (voir Fichier des conditions aux limites) :

- 1 - 1100000000 pour une condition de Dirichlet ;
- 2 - 2200000000 pour une condition de Cauchy ;
- 3 - 3300000000 pour une condition de sortie libre.

Remarques :

Les degrés de liberté sont ordonnés comme suit :

- 1- oxygène dissous ;
- 2- DBO.

Le même type de conditions aux limites doit être appliqué sur un même noeud à l'oxygène dissous et à la DBO.

Si aucune condition n'est posée sur une frontière, une condition de type Neuman sera imposée par défaut par le simulateur.

Dans les zones sèches à profondeur nulles ou négatives, une condition de Dirichlet est appliquée automatiquement avec une valeur de concentration imposée à zéro.

Dans les zones de transition, une condition de Neuman est appliquée automatiquement ce qui permet d'éviter les effets indésirables de couche limite de concentration au voisinage des rives.

---

## Les sollicitations de l'élément CD2DNC\_DBO

- Sollicitations concentrées de l'élément CD2DNC\_DBO
- Sollicitations réparties de l'élément CD2DNC\_DBO

---

### Sollicitations concentrées de l'élément CD2DNC\_DBO

Les sollicitations concentrées servent à introduire les charges réparties équivalentes résultant de la discrétisation par éléments finis du produit  $QHC^0$ , où  $Q(m^3/s)$  est le débit injecté dans le domaine,  $H(m)$  la profondeur et  $C^0(kg/m^3)$  la concentration injectée dans le domaine modélisé. La convention de signe suppose positif (+) un débit entrant et négatif (-) un débit sortant du domaine. Les sollicitations concentrées peuvent être appliquées en n'importe quel point du domaine. En pratique, elles servent à introduire les sources d'injection ponctuelles dans le modèle. Dans le fichier des sollicitations concentrées, le produit  $QHC^0$  est associé à chaque degré de liberté nodal (voir Fichier des sollicitations).

---

### Sollicitations réparties de l'élément CD2DNC\_DBO

Les sollicitations réparties servent à introduire le débit  $Q(m^3/s)$  à la concentration du milieu injecté dans le domaine modélisé. La convention de signe suppose positif (+) un débit entrant et négatif (-) un débit sortant du domaine. Les sollicitations réparties peuvent être appliquées en n'importe quel point du domaine. En pratique, elles servent à introduire le débit des sources et puits d'eau ponctuels dans le modèle. Dans le fichier des sollicitations réparties, le débit  $Q$  est associé à chaque degré de liberté (voir Fichier des sollicitations).

---

## Solution initiale de l'élément CD2DNC\_DBO

Nombre de termes d'initialisation NTINI=2 :

- 1 - valeur initiale de la concentration en oxygène dissous ;
- 2 - valeur initiale de concentration en DBO.

Remarque : La valeur initiale est appliquée à tous les degrés de liberté du domaine.

## Élément fini CD2DNC\_MES

Baptisé CD2DNC\_MES en référence aux équations de Convection-Diffusion à 2 Dimensions sous forme Non Conservative des Matières En Suspension.

- Comment atteindre l'élément CD2DNC\_MES?
- La fonction de l'élément CD2DNC\_MES
- Les propriétés de l'élément CD2DNC\_MES
- Les conditions aux limites de l'élément CD2DNC\_MES
- Les sollicitations de l'élément CD2DNC\_MES
- Solution initiale de l'élément CD2DNC\_MES

---

### Comment atteindre l'élément CD2DNC\_MES?

Pour atteindre l'élément fini CD2DNC\_MES, dans le fichier de commandes la variable ELTYP doit être définie comme suit :

```
_ELTYP= 'CD2DNC_MES'
```

---

### La fonction de l'élément CD2DNC\_MES

CD2DNC\_MES est un élément fini triangulaire linéaire à trois (3) noeuds, un sur chaque sommet. Il est utilisé pour la discrétisation des équations bidimensionnelles de convection-diffusion sous forme non conservative avec une cinétique de dégradation du premier ordre. Il est dédié à l'étude du transport de matières en suspension en milieu fluvial et estuarien. Il admet un (1) degré de liberté par noeud la concentration.

---

### Les propriétés de l'élément CD2DNC\_MES

- Propriétés géométriques de l'élément CD2DNC\_MES
- Propriétés globales de l'élément CD2DNC\_MES
- Propriétés élémentaires de l'élément CD2DNC\_MES
- Propriétés nodales de l'élément CD2DNC\_MES

---

#### Propriétés géométriques de l'élément CD2DNC\_MES

Nombre de dimensions NDIM=2 ;

Nombre de noeuds par élément NNEL=3.

### Propriétés globales de l'élément CD2DNC\_MES

Nombre de propriétés globales à lire NPRGL=15 :

- 1 - diffusivité moléculaire ( $m^2/s$ ) ;
- 2 - coefficient de pondération de la diffusivité verticale ;
- 3 - coefficient de pondération de la diffusivité horizontale ;
- 4 - coefficient de pondération de la diffusivité longitudinale ;
- 5 - accélération gravitationnelle ( $m/s^2$ ) ;
- 6 - masse volumique de l'eau ( $kg/m^3$ ) ;
- 7 - viscosité cinématique de l'eau ( $m^2/s$ ) ;
- 8 - code pour sédiments cohésifs (1) ou non cohésifs (2) ;
- 9 - vitesse de sédimentation constante si supérieure à zéro ( $m/s$ ) ;
- 10 - contrainte critique de déposition ( $kg/m/s^2$ ) ;
- 11 - contrainte critique d'érosion calculée par le simulateur si égale zéro ( $kg/m/s^2$ ) ;
- 12 - diamètre moyen des sédiments (m) ;
- 13 - masse volumique moyenne des sédiments ( $kg/m^3$ ) ;
- 14 - constante de sédimentation ( ) ;
- 15 - coefficient d'érosion ( $kg/m^2/s$ ).

Remarques : La vitesse de sédimentation est constante si elle est spécifiée supérieure à zéro. Sinon, elle est calculée par le simulateur.

Si spécifiée égale à zéro, la contrainte critique d'érosion est calculée par le simulateur selon l'approximation de van Rijn du diagramme de Shields.

---

### Propriétés élémentaires de l'élément CD2DNC\_MES

Aucune propriété élémentaire à lire NPREL=0.

---

### Propriétés nodales de l'élément CD2DNC\_MES

Nombre de propriétés par noeud à lire NPRNL=6 :

- 1 - composante de vitesse suivant x ( $m/s$ ) ;
- 2 - composante de vitesse suivant y ( $m/s$ ) ;
- 3 - profondeur de l'écoulement (m) ;
- 4 - diffusivité verticale ( $m^2/s$ ) ;
- 5 - diffusivité horizontale ( $m^2/s$ ) ;
- 6 - vitesse de cisaillement ( $m/s$ ).

## Les conditions aux limites de l'élément CD2DNC\_MES

Les conditions aux limites (C.L.) sont appliquées sur les degrés de liberté au contour du domaine. Le nombre de degré de liberté par noeud  $NDLN=1$  ; la concentration  $C(kg/m^3)$ . L'application des conditions aux limites dépend du type de frontière. On en dénombre trois (3) :

- Condition de frontière fermée de l'élément CD2DNC\_MES
- Condition de frontière ouverte d'entrée de l'élément CD2DNC\_MES
- Condition de frontière ouverte de sortie de l'élément CD2DNC\_MES
- Codes de conditions aux limites de l'élément CD2DNC\_MES

---

### Condition de frontière fermée de l'élément CD2DNC\_MES

On parle de frontière fermée pour définir une frontière imperméable, l'écoulement normal étant nul. Sur une frontière fermée, on impose généralement une condition de Neuman. Une condition de Neuman stipule que le gradient de concentration est nul dans la direction normale à la frontière. Cette condition est appliquée par défaut.

---

### Condition de frontière ouverte d'entrée de l'élément CD2DNC\_MES

On parle de frontière ouverte d'entrée lorsque l'écoulement passe au travers pour entrer dans le domaine. Sur une frontière ouverte d'entrée, on impose soit une condition de Dirichlet (la valeur de la concentration est imposée directement) soit une condition de Cauchy (la valeur de la concentration va tendre vers la valeur imposée). D'un point de vue numérique, il est parfois plus souhaitable d'exploiter une condition de Cauchy de façon à atténuer l'amplitude des oscillations parasites de concentration qui polluent la solution (voir **Comment atténuer les oscillations parasites?**).

---

### Condition de frontière ouverte de sortie de l'élément CD2DNC\_MES

On parle de frontière ouverte de sortie lorsque l'écoulement passe au travers pour sortir du domaine. Sur une frontière ouverte de sortie, on impose soit une condition de sortie libre soit une condition de Neuman, prise en compte par défaut, qui stipule que le gradient de concentration est nul dans la direction normale à la frontière. D'un point de vue physique, il est parfois plus souhaitable d'exploiter une condition de sortie libre plus adéquate qu'une condition de Neuman.

---

### Codes de conditions aux limites de l'élément CD2DNC\_MES



La convention de code permettant d'introduire les C.L. est (voir Fichier des conditions aux limites) :

- 1 - 1000000000 pour une condition de Dirichlet ;
- 2 - 2000000000 pour une condition de Cauchy ;
- 3 - 3000000000 pour une condition de sortie libre.

Remarques :

Si aucune condition n'est posée sur une frontière, une condition de type Neuman sera imposée par défaut par le simulateur.

Dans les zones sèches à profondeur nulles ou négatives, une condition de Dirichlet est appliquée automatiquement avec une valeur de concentration imposée à zéro.

Dans les zones de transition, une condition de Neuman est appliquée automatiquement ce qui permet d'éviter les effets indésirables de couche limite de concentration au voisinage des rives.

---

## Les sollicitations de l'élément CD2DNC\_MES

- Sollicitations concentrées de l'élément CD2DNC\_MES
- Sollicitations réparties de l'élément CD2DNC\_MES

---

### Sollicitations concentrées de l'élément CD2DNC\_MES

Les sollicitations concentrées servent à introduire les charges réparties équivalentes résultant de la discrétisation par éléments finis du produit  $QHC^0$ , où  $Q(m^3/s)$  est le débit injecté dans le domaine,  $H(m)$  la profondeur et  $C^0(kg/m^3)$  la concentration injectée dans le domaine modélisé. La convention de signe suppose positif (+) un débit entrant et négatif (-) un débit sortant du domaine. Les sollicitations concentrées peuvent être appliquées en n'importe quel point du domaine. En pratique, elles servent à introduire les sources d'injection ponctuelles dans le modèle. Dans le fichier des sollicitations concentrées, le produit  $QHC^0$  est associé à chaque degré de liberté nodal (voir Fichier des sollicitations).

---

### Sollicitations réparties de l'élément CD2DNC\_MES

Les sollicitations réparties servent à introduire le débit  $Q(m^3/s)$  à la concentration du milieu injecté dans le domaine modélisé. La convention de signe suppose positif (+) un débit entrant et négatif (-) un débit sortant du domaine. Les sollicitations réparties peuvent être appliquées en n'importe quel point du domaine. En pratique, elles servent à introduire le débit des sources et puits d'eau ponctuels dans le modèle. Dans le fichier des sollicitations réparties le débit  $Q$  est associé à chaque degré de liberté (voir Fichier des sollicitations).

## Solution initiale de l'élément CD2DNC\_MES

Nombre de termes d'initialisation NTINI=1 :

1 - valeur de la concentration constante dans tous le domaine.

---

## Élément fini CD2DNC\_MET

Baptisé CD2DNC\_MET en référence aux équations de Convection-Diffusion à 2 Dimensions sous forme Non Conservative des Métaux.

- Comment atteindre l'élément CD2DNC\_MET?
  - La fonction de l'élément CD2DNC\_MET
  - Les propriétés de l'élément CD2DNC\_MET
  - Les conditions aux limites de l'élément CD2DNC\_MET
  - Les sollicitations de l'élément CD2DNC\_MET
  - Solution initiale de l'élément CD2DNC\_MET
- 

## Comment atteindre l'élément CD2DNC\_MET?

Pour atteindre l'élément fini CD2DNC\_MET, dans le fichier de commandes la variable ELTYP doit être définie comme suit :

```
_ELTYP=' CD2DNC_MET'
```

---

## La fonction de l'élément CD2DNC\_MET

CD2DNC\_MET est un élément fini triangulaire linéaire à trois (3) noeuds, un sur chaque sommet. Il est utilisé pour la discrétisation des équations bidimensionnelles de convection-diffusion sous forme non conservative avec une cinétique de dégradation du premier ordre. Il est dédié à l'étude de la contamination en métaux du milieu fluvial et estuarien. Il admet un (1) degré de liberté par noeud la concentration.

---

## Les propriétés de l'élément CD2DNC\_MET

- Propriétés géométriques de l'élément CD2DNC\_MET
  - Propriétés globales de l'élément CD2DNC\_MET
  - Propriétés élémentaires de l'élément CD2DNC\_MET
  - Propriétés nodales de l'élément CD2DNC\_MET
-

### Propriétés géométriques de l'élément CD2DNC\_MET

Nombre de dimensions NDIM=2 ;

Nombre de noeuds par élément NNEL=3.

---

### Propriétés globales de l'élément CD2DNC\_MET

Nombre de propriétés globales à lire NPRGL=6 :

- 1 - diffusivité moléculaire ( $m^2/s$ ) ;
  - 2 - coefficient de pondération de la diffusivité verticale ;
  - 3 - coefficient de pondération de la diffusivité horizontale ;
  - 4 - coefficient de pondération de la diffusivité longitudinale ;
  - 5 - coefficient de dégradation global ( $s^{-1}$ ) ;
  - 6 - coefficient de partage.
- 

### Propriétés élémentaires de l'élément CD2DNC\_MET

Aucune propriété élémentaire à lire NPREL=0.

---

### Propriétés nodales de l'élément CD2DNC\_MET

Nombre de propriétés par noeud à lire NPRNL=8 :

- 1 - composante de vitesse suivant x (m/s) ;
  - 2 - composante de vitesse suivant y (m/s) ;
  - 3 - profondeur de l'écoulement (m) ;
  - 4 - diffusivité verticale ( $m^2/s$ ) ;
  - 5 - diffusivité horizontale ( $m^2/s$ ) ;
  - 6 - concentration de matières en suspension ( $kg/m^3$ ) ;
  - 7 - vitesse de sédimentation (m/s) ;
  - 8 - probabilité de déposition.
- 

### Les conditions aux limites de l'élément CD2DNC\_MET

Les conditions aux limites (C.L.) sont appliquées sur les degrés de liberté au contour du domaine. Le nombre de degré de liberté par noeud NDLN=1 ; la concentration C( $kg/m^3$ ). L'application des conditions aux limites dépend du type de frontière. On en dénombre trois :

- Condition de frontière fermée de l'élément CD2DNC\_MET
- Condition de frontière ouverte d'entrée de l'élément CD2DNC\_MET

- Condition de frontière ouverte de sortie de l'élément CD2DNC\_MET
  - Codes de conditions aux limites de l'élément CD2DNC\_MET
- 

### Condition de frontière fermée de l'élément CD2DNC\_MET

On parle de frontière fermée pour définir une frontière imperméable, l'écoulement normal étant nul. Sur une frontière fermée, on impose généralement une condition de Neuman. Une condition de Neuman stipule que le gradient de concentration est nul dans la direction normale à la frontière. Cette condition est appliquée par défaut.

---

### Condition de frontière ouverte d'entrée de l'élément CD2DNC\_MET

On parle de frontière ouverte d'entrée lorsque l'écoulement passe au travers pour entrer dans le domaine. Sur une frontière ouverte d'entrée, on impose soit une condition de Dirichlet (la valeur de la concentration est imposée directement) soit une condition de Cauchy (la valeur de la concentration va tendre vers la valeur imposée). D'un point de vue numérique, il est parfois plus souhaitable d'exploiter une condition de Cauchy de façon à atténuer l'amplitude des oscillations parasites de concentration qui polluent la solution.

---

### Condition de frontière ouverte de sortie de l'élément CD2DNC\_MET

On parle de frontière ouverte de sortie lorsque l'écoulement passe au travers pour sortir du domaine. Sur une frontière ouverte de sortie, on impose soit une condition de sortie libre soit une condition de Neuman, prise en compte par défaut, qui stipule que le gradient de concentration est nul dans la direction normale à la frontière. D'un point de vue physique, il est parfois plus souhaitable d'exploiter une condition de sortie libre plus adéquate qu'une condition de Neuman.

---

### Codes de conditions aux limites de l'élément CD2DNC\_MET

La convention de code permettant d'introduire les C.L. est (voir Fichier des conditions aux limites) :

- 1 - 1000000000 pour une condition de Dirichlet ;
- 2 - 2000000000 pour une condition de Cauchy ;
- 3 - 3000000000 pour une condition de sortie libre.

Remarques :

Si aucune condition n'est posée sur une frontière, une condition de type Neuman sera imposée par défaut par le simulateur.

Dans les zones sèches à profondeur nulles ou négatives, une condition de Dirichlet est appliquée automatiquement avec une valeur de concentration imposée à zéro.

Dans les zones de transition, une condition de Neuman est appliquée automatiquement ce qui permet d'éviter les effets indésirables de couche limite de concentration au voisinage des rives.

---

## Les sollicitations de l'élément CD2DNC\_MET

- Sollicitations concentrées de l'élément CD2DNC\_MET
- Sollicitations réparties de l'élément CD2DNC\_MET

---

### Sollicitations concentrées de l'élément CD2DNC\_MET

Les sollicitations concentrées servent à introduire les charges réparties équivalentes résultant de la discrétisation par éléments finis du produit  $QHC^0$ , où  $Q(m^3/s)$  est le débit injecté dans le domaine,  $H(m)$  la profondeur et  $C^0(kg/m^3)$  la concentration injectée dans le domaine modélisé. La convention de signe suppose positif (+) un débit entrant et négatif (-) un débit sortant du domaine. Les sollicitations concentrées peuvent être appliquées en n'importe quel point du domaine. En pratique, elles servent à introduire les sources d'injection ponctuelles dans le modèle. Dans le fichier des sollicitations concentrées, le produit  $QHC^0$  est associé à chaque degré de liberté nodal (voir Fichier des sollicitations).

---

### Sollicitations réparties de l'élément CD2DNC\_MET

Les sollicitations réparties servent à introduire le débit  $Q(m^3/s)$  à la concentration du milieu injecté dans le domaine modélisé. La convention de signe suppose positif (+) un débit entrant et négatif (-) un débit sortant du domaine. Les sollicitations réparties peuvent être appliquées en n'importe quel point du domaine. En pratique, elles servent à introduire le débit des sources et puits d'eau ponctuels dans le modèle. Dans le fichier des sollicitations réparties, le débit  $Q$  est associé à chaque degré de liberté (voir Fichier des sollicitations).

---

## Solution initiale de l'élément CD2DNC\_MET

Nombre de termes d'initialisation  $NTINI=1$  :

1 - valeur de la concentration constante dans tous le domaine.

## Élément fini CD2DNC\_TOX

Baptisé CD2DNC\_TOX en référence aux équations de Convection-Diffusion à 2 Dimensions sous forme Non Conservative des substances TOXiques.

- Comment atteindre l'élément CD2DNC\_TOX?
- La fonction de l'élément CD2DNC\_TOX
- Les propriétés de l'élément CD2DNC\_TOX
- Les conditions aux limites de l'élément CD2DNC\_TOX
- Les sollicitations de l'élément CD2DNC\_TOX
- Solution initiale de l'élément CD2DNC\_TOX

---

### Comment atteindre l'élément CD2DNC\_TOX?

Pour atteindre l'élément fini CD2DNC\_TOX, dans le fichier de commandes la variable ELTYP doit être définie comme suit :

```
_ELTYP='CD2DNC_TOX'
```

---

### La fonction de l'élément CD2DNC\_TOX

CD2DNC\_TOX est un élément fini triangulaire linéaire à trois (3) noeuds, un sur chaque sommet. Il est utilisé pour la discrétisation des équations bidimensionnelles de convection-diffusion sous forme non conservative avec une cinétique de dégradation du premier ordre. Il est dédié à l'étude de la contamination en substances toxiques du milieu fluvial et estuarien. Il admet un (1) degré de liberté par noeud, la concentration.

---

### Les propriétés de l'élément CD2DNC\_TOX

- Propriétés géométriques de l'élément CD2DNC\_TOX
- Propriétés globales de l'élément CD2DNC\_TOX
- Propriétés élémentaires de l'élément CD2DNC\_TOX
- Propriétés nodales de l'élément CD2DNC\_TOX

---

#### Propriétés géométriques de l'élément CD2DNC\_TOX

Nombre de dimensions NDIM=2 ;

Nombre de noeuds par élément NNEL=3.

### Propriétés globales de l'élément CD2DNC\_TOX

Nombre de propriétés globales à lire NPRGL=8 :

- 1 - diffusivité moléculaire ( $m^2/s$ ) ;
- 2 - coefficient de pondération de la diffusivité verticale ;
- 3 - coefficient de pondération de la diffusivité horizontale ;
- 4 - coefficient de pondération de la diffusivité longitudinale ;
- 5 - contenu en carbone des matières en suspension ;
- 6 - coefficient de dégradation global ( $s^{-1}$ ) ;
- 7 - coefficient de volatilisation ;
- 8 - solubilité.

---

### Propriétés élémentaires de l'élément CD2DNC\_TOX

Aucune propriété élémentaire à lire NPREL=0.

---

### Propriétés nodales de l'élément CD2DNC\_TOX

Nombre de propriétés par noeud à lire NPRNL=8 :

- 1 - composante de vitesse suivant x (m/s) ;
- 2 - composante de vitesse suivant y (m/s) ;
- 3 - profondeur de l'écoulement (m) ;
- 4 - diffusivité verticale ( $m^2/s$ ) ;
- 5 - diffusivité horizontale ( $m^2/s$ ).
- 6 - concentration de matières en suspension ( $kg/m^3$ ) ;
- 7 - vitesse de sédimentation (m/s) ;
- 8 - probabilité de déposition.

---

### Les conditions aux limites de l'élément CD2DNC\_TOX

Les conditions aux limites (C.L.) sont appliquées sur les degrés de liberté au contour du domaine. Le nombre de degré de liberté par noeud NDLN=1 ; la concentration  $C(kg/m^3)$ . L'application des conditions aux limites dépend du type de frontière. On en dénombre trois :

- Condition de frontière fermée de l'élément CD2DNC\_TOX
- Condition de frontière ouverte d'entrée de l'élément CD2DNC\_TOX
- Condition de frontière ouverte de sortie de l'élément CD2DNC\_TOX

- Codes de conditions aux limites de l'élément CD2DNC\_TOX
- 

### Condition de frontière fermée de l'élément CD2DNC\_TOX

On parle de frontière fermée pour définir une frontière imperméable, l'écoulement normal étant nul. Sur une frontière fermée, on impose généralement une condition de Neuman. Une condition de Neuman stipule que le gradient de concentration est nul dans la direction normale à la frontière. Cette condition est appliquée par défaut.

---

### Condition de frontière ouverte d'entrée de l'élément CD2DNC\_TOX

On parle de frontière ouverte d'entrée lorsque l'écoulement passe au travers pour entrer dans le domaine. Sur une frontière ouverte d'entrée, on impose soit une condition de Dirichlet (la valeur de la concentration est imposée directement) soit une condition de Cauchy (la valeur de la concentration va tendre vers la valeur imposée). D'un point de vue numérique, il est parfois plus souhaitable d'exploiter une condition de Cauchy de façon à atténuer l'amplitude des oscillations parasites de concentration qui polluent la solution (voir **Comment atténuer les oscillations parasites?**).

---

### Condition de frontière ouverte de sortie de l'élément CD2DNC\_TOX

On parle de frontière ouverte de sortie lorsque l'écoulement passe au travers pour sortir du domaine. Sur une frontière ouverte de sortie, on impose soit une condition de sortie libre soit une condition de Neuman, prise en compte par défaut, qui stipule que le gradient de concentration est nul dans la direction normale à la frontière. D'un point de vue physique, il est parfois plus souhaitable d'exploiter une condition de sortie libre plus adéquate qu'une condition de Neuman.

---

### Codes de conditions aux limites de l'élément CD2DNC\_TOX

La convention de code permettant d'introduire les C.L. est (voir Fichier des conditions aux limites) :

- 1 - 1000000000 pour une condition de Dirichlet ;
- 2 - 2000000000 pour une condition de Cauchy ;
- 3 - 3000000000 pour une condition de sortie libre.

Remarques :

Si aucune condition n'est posée sur une frontière, une condition de type Neuman sera imposée par défaut par le simulateur.

Dans les zones sèches à profondeur nulles ou négatives, une condition de Dirichlet est appliquée automatiquement avec une valeur de concentration imposée à zéro.



Dans les zones de transition, une condition de Neuman est appliquée automatiquement ce qui permet d'éviter les effets indésirables de couche limite de concentration au voisinage des rives.

---

## Les sollicitations de l'élément CD2DNC\_TOX

- Sollicitations concentrées de l'élément CD2DNC\_TOX
- Sollicitations réparties de l'élément CD2DNC\_TOX

---

### Sollicitations concentrées de l'élément CD2DNC\_TOX

Les sollicitations concentrées servent à introduire les charges réparties équivalentes résultant de la discrétisation par éléments finis du produit  $QHC^0$ , où  $Q(m^3/s)$  est le débit injecté dans le domaine,  $H(m)$  la profondeur et  $C^0(kg/m^3)$  la concentration injectée dans le domaine modélisé. La convention de signe suppose positif (+) un débit entrant et négatif (-) un débit sortant du domaine. Les sollicitations concentrées peuvent être appliquées en n'importe quel point du domaine. En pratique, elles servent à introduire les sources d'injection ponctuelles dans le modèle. Dans le fichier des sollicitations concentrées, le produit  $QHC^0$  est associé à chaque degré de liberté nodal (voir Fichier des sollicitations).

---

### Sollicitations réparties de l'élément CD2DNC\_TOX

Les sollicitations réparties servent à introduire le débit  $Q(m^3/s)$  à la concentration du milieu injecté dans le domaine modélisé. La convention de signe suppose positif (+) un débit entrant et négatif (-) un débit sortant du domaine. Les sollicitations réparties peuvent être appliquées en n'importe quel point du domaine. En pratique, elles servent à introduire le débit des sources et puits d'eau ponctuels dans le modèle. Dans le fichier des sollicitations réparties, le débit  $Q$  est associé à chaque degré de liberté (voir Fichier des sollicitations).

---

## Solution initiale de l'élément CD2DNC\_TOX

Nombre de termes d'initialisation NTINI=1 :

1 - valeur de la concentration constante dans tous le domaine.

---

## Librairie de schémas temporels

La librairie de schémas temporels de DISPERSIM propose deux méthodes pour la discrétisation du temps :

- Le schéma temporel EULER

- Le schéma temporel STATIQ

---

## Le schéma temporel EULER

Il s'agit d'un schéma de résolution dépendant du temps dont l'approximation est du type EULER (approximation du premier ordre). Son utilisation est requise pour simuler des processus physiques non permanents ou transitoires. Pour atteindre le schéma d'EULER la syntaxe est :

```
_STEMP= ' EULER '
```

---

## Le schéma temporel STATIQ

Il s'agit d'un schéma de résolution indépendant du temps ou STATIQ. Son utilisation est requise pour simuler des processus physiques permanents dans le temps. Pour atteindre le schéma STATIQ la syntaxe est :

```
_STEMP= ' STATIQ '
```

---

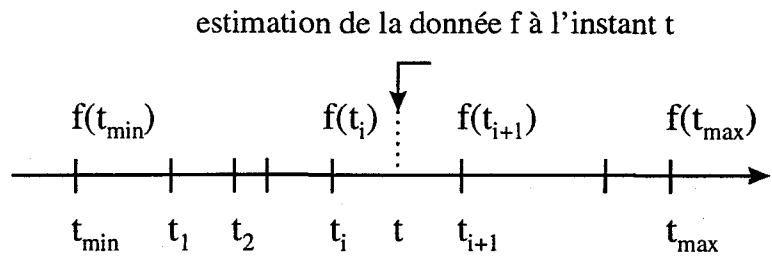
## Traitement des données transitoires

Les données d'un fichier d'entrée sont dites transitoires dès lors que nous avons deux ou plusieurs séquences de valeurs auxquelles est systématiquement associée une valeur du temps (voir Formats des fichiers d'entrée). Le logiciel DISPERSIM a la capacité de traiter les données d'entrées transitoires dans le cadre d'une simulation évoluant dans le temps. Plus précisément, il est possible de prédire les données d'entrée à un temps donné au moyen d'une procédure d'interpolation. Le principe de détermination des valeurs des données transitoires, valable pour n'importe quelle donnée d'entrée, est le suivant :

sur la Figure 4 on représente l'intervalle de variation d'une fonction  $f$ . L'intervalle est compris entre les temps  $t_{\min}$  et  $t_{\max}$ . Aux différents temps  $t_1, t_2, t_i, \dots, t_{i+1}, \dots$  est associée une valeur de  $f$ . Ainsi, au temps  $t$ , la valeur de  $f$  est égale à :

- si  $t < t_{\min}$ ,  $f(t) = f(t_{\min})$
- si  $t_i < t < t_{i+1}$ ,  $f(t) = at + b$  avec  $a = [f(t_{i+1}) - f(t_i)] / (t_{i+1} - t_i)$  et  $b = 1/2[f(t_i) + f(t_{i+1}) - a(t_i + t_{i+1})]$
- si  $t > t_{\max}$ ,  $f(t) = f(t_{\max})$

Figure 4 : Intervalle temporel de variation d'une donnée f.



Remarque : Les séquences de données transitoires doivent être rangées séquentiellement de manière à avoir une croissance monotone du temps.

## Dépendances des blocs

Les blocs sont non seulement dépendant entre eux mais dépendent également des variables. Des tableaux sont employés pour identifier les dépendances au moyen de la convention de notation suivante :

- 1 - dépendance obligatoire : X ;
- 2 - dépendance optionnelle : O ;

Les différents types de dépendances recensés sont:

- Dépendances entre les blocs
- Dépendances blocs-variables

## Dépendances entre les blocs

Dans le Tableau 2 ci-dessous, les blocs appelés sont listés dans les colonnes 2 à 17 et les blocs de dépendance sont listés dans la première colonne. Seul le bloc STOP n'a pas été listé puisqu'il ne dépend d'aucun bloc et réciproquement.

Tableau 2 : Dépendances entre les blocs.																
	COND	COOR	ELEM	ERR	FIN	FORM	INIT	POST	PRCO	PREL	PRGL	PRNO	RESI	SOLC	SOLR	SOLV
COND													X	X	X	X
COOR	X		X	X			X	X	X	X		X	X	X	X	X
ELEM	X			X			X	X	X	X		X	X	X	X	X
FORM		X									X					
INIT	X			X	X			X					X	X	X	X
PRCO																O
PREL	X			X			X	X					X	X	X	X
PRGL	X			X			X	X		X		X	X	X	X	X
PRNO	X			X			X	X					X	X	X	X
SOLC													O			O
SOLR													O			O

## Dépendances blocs-variables

- Dépendances blocs-chaînes de caractères
- Dépendances blocs-variables entières
- Dépendances blocs-variables réelles

## Dépendances blocs-chaînes de caractères

Dans le Tableau 3 ci-dessous, les blocs appelés sont listés dans les colonnes 2 à 15. Les blocs PRCO, PRGL et STOP n'ont pas été listé puisqu'ils ne dépendent d'aucune variable. Les variables de dépendance du type chaîne de caractère sont listées dans la première colonne.

Tableau 3 : Dépendances des blocs par rapport aux chaînes de caractères.															
	COND	COOR	ELEM	ERR	FIN	FORM	INIT	POST	PREL	PRNO	RESI	SOLC	SOLR	SOLV	
ELTYP						X									
MCND	X														
MCOR		X													
MELE			X												

MERR				X										
MEXE														O
MFIL	O	O	O	O	O		O	O	O	O	O	O	O	O
MFIN					X									
MINI							O							
MPRE									O					
MPRN										O				
MPST							X							
MRES											X			
MSLC												O		
MSLR													O	
STEMP						X								

### Dépendances blocs-variables entières

Dans le Tableau 4 ci-dessous deux blocs seulement dépendent des variables entières.

Tableau 4 : Dépendances des blocs par rapport aux variables entières.		
	PRCO	SOLV
ILU	O	
IMPR		O
NITER		O
NPAS		O
NPREC		O
NRDEM		O

### Dépendances blocs-variables réelles

Dans le ci-dessous, un total de dix blocs, listés dans les colonnes 2 à 11, dépend de variables réelles listées dans la première colonne.

Tableau 5 : Dépendances des blocs par rapport aux variables réelles.											
	COND	INIT	POST	PRCO	PREL	PRNO	RESI	SOLC	SOLR	SOLV	

ALFA								○				○
DELPRT				○								
DPAS								○				○
EPSDL												○
OMEGA												○
TINI				○								○
TASCND	○											
TASINI			○									
TASPRE							○					
TASPRN								○				
TASSLC										○		
TASSLR											○	

## Exemple de fichier de commandes

- Cas stationnaire
- Cas non-stationnaire ou transitoire

### Cas stationnaire

```

#-----
#      EXEMPLE DE FICHIER DE COMMANDES POUR DISPERSIM
#      Répertoire de simulation :
# c :\dispersim\simul\
#      Nom générique des fichiers de données et de
# résultats : test
#      Simulation stationnaire
#      Niveau d'impression minimum :M=0
#-----
!--- DÉFINITION DE LA FORMULATION
!--- TYPE D'ÉLÉMENT
_ELTYP='CD2DNC_CCN'
!--- SCHÉMA TEMPOREL
_STEMP='STATIQ'

```

```

FORM[0]
!--- DÉFINITION DES FICHIERS
_MFIL=' c :\dispersim\simul\test'
!--- FICHER DE PROGRESSION DE LA SIMULATION
MEXE='.dat'
!--- LECTURE DES COORDONNÉES
_MCOR='.cor'
COOR[0]
!--- LECTURE DES CONNECTIVITÉS
_MELE='.ele'
ELEM[0]
!--- LECTURE DES PROPRIÉTÉS GLOBALES
PRGL[0](1.E-06, 1.0, 1.0, 2.33)
!--- LECTURE DES PROPRIÉTÉS NODALES
_MPRN='.prn'
PRNO[0]
!--- APPEL DU BLOC DES PROPRIÉTÉS ÉLÉMENTAIRES
PREL[0]
!--- INITIALISATION DE LA SOLUTION
INIT[0]
!--- LECTURE DES CONDITIONS AUX LIMITES
_MCND='.cnd'
COND[0]
!--- LECTURE DES SOLLICITATIONS CONCENTRÉES
_MSLC='.slc'
SOLC[0]
!--- LECTURE DES SOLLICITATIONS RÉPARTIES
_MSLR='.slr'
SOLR[0]
!--- PRÉCONDITIONNEMENT
_ILU=-1
PRCO[0]
!--- RÉOLUTION STATIONNAIRE PAR GMRES
! NONLINÉAIRE ET PRECONDITONNEMENT ILU AVEC
! 1 PRECONDITIONNEMENT, 1 REDÉMARRAGE,
! 1 ITÉRATION, PRÉCISION 10-6
_IMPR=3
_NPREC=1

```

```
_NRDEM=1
_NITER=1
_EPSDL=1.0E-06
_OMEGA=1
SOLV[0]
!--- RÉSULTATS : IMPRESSION DE LA SOLUTION
! FINALE
_MFIN='.fin'
FIN[0]
!--- RÉSULTATS : RÉSIDUS
_MRES='.res'
RESI[0]
!--- RÉSULTATS : POST-TRAITEMENT
_MPST='.pst'
POST[0]
!--- RÉSULTATS : ERREURS NUMÉRIQUES
_MERR='.err'
ERR[0]
!--- ARRÊT DE LA SIMULATION
STOP
```

---

## Cas non-stationnaire ou transitoire

```
#-----
#      EXEMPLE DE FICHIER DE COMMANDES POUR DISPERSIM
#      Répertoire de simulation :
# c :\dispersim\simul\
#      Nom générique des fichiers de données et de
# résultats : test
#      Simulation non-stationnaire (les conditions
# aux limites et la
#      solution initiale varient dans le temps)
#      Niveau d'impression minimum :M=0
#-----
!--- DÉFINITION DE LA FORMULATION
!--- TYPE D'ÉLÉMENT
_ELTP='CD2DNC_CCN'
```



```
!--- SCHÉMA TEMPOREL
_STEMP='EULER
FORM[0]'
!--- DÉFINITION DES FICHIERS
_MFIL=' c :\dispersim\simul\test'
!--- FICHER DE PROGRESSION DE LA SIMULATION
MEXE='.dat'
!--- LECTURE DES COORDONNÉES
_MCOR='.cor'
COOR[0]
!--- LECTURE DES CONNECTIVITÉS
_MELE='.ele'
ELEM[0]
!--- LECTURE DES PROPRIÉTÉS GLOBALES
PRGL[0](1.E-06, 1.0, 1.0, 2.33)
!--- LECTURE DES PROPRIÉTÉS NODALES
_MPRN='.prn'
PRNO[0]
!--- APPEL DU BLOC DES PROPRIÉTÉS ÉLÉMENTAIRES
PREL[0]
!--- INITIALISATION DE LA SOLUTION SUR FICHER
_MINI='.deb'
_TASINI=0.
INIT[0]
!--- LECTURE DES CONDITIONS AUX LIMITES
_MCND='.cnd'
_TASCND=0.'
COND[0]
!--- LECTURE DES SOLlicitATIONS CONCENTRÉES
_MSLC='.slc'
SOLC[0]
!--- LECTURE DES SOLlicitATIONS RÉPARTIES
_MSLR='.slr'
SOLR[0]
!--- PRÉCONDITIONNEMENT
_ILU=-1
PRCO[0]
!--- RÉOLUTION NON-STATIONNAIRE PAR EULER-
```

```
! IMPLICITE ET GMRES NONLINÉAIRE
! PRECONDITIONNÉ PAR ILU AVEC 10 PAS DE TEMPS
! DE 60 SECONDES
! 1 PRECONDITIONNEMENT, 1 REDÉMARRAGE,
! 1 ITÉRATION, PRÉCISION 10-6
_TINI=0.00
_DPAS=60.00
_ALFA=1.0
_IMPR=3
_NPAS=10
_NPREC=1
_NRDEM=1
_NITER=1
_EPSDL=1.0E-06
_OMEGA=1
SOLV[0]
!--- RÉSULTATS : IMPRESSION DE LA SOLUTION
FINALE
_MFIN='.fin'
FIN[0]
!--- RÉSULTATS : RÉSIDUS
_MRES='.res'
RESI[0]
!--- RÉSULTATS : POST-TRAITEMENT
_MPST='.pst'
POST[0]
!--- RÉSULTATS : ERREURS NUMÉRIQUES
_MERR='.err'
ERR[0]
!--- ARRÊT DE LA SIMULATION
STOP
```

---

## Formats des fichiers d'entrée

Cette section est consacrée à la présentation des formats des différents fichiers de données nécessaires à DISPERSIM pour effectuer une simulation.

On distingue deux catégories de données à savoir celles qui sont statiques et celles qui sont variables dans le temps. Les données dynamiques sont introduites par séquences, correspondant

chacune à un temps précis qui doit être systématiquement défini en premier (voir Traitement des données transitoires). En toute logique un fichier de données statiques se comporte qu'une seule séquence indépendante du temps. Pour certains fichiers d'entrée un exemple simple est proposé. Les fichiers d'entrées lus par DISPERSIM sont :

- Fichier des conditions aux limites
- Fichier des conditions aux limites
- Fichier des coordonnées
- Fichier des propriétés élémentaires
- Fichier des propriétés nodales
- Fichier des sollicitations
- Fichier de solution initiale

## Fichier des conditions aux limites

Les conditions aux limites constituent une information du type dynamique. Le format associé se présente comme suit :

Séquence	Lignes	Format	Variables	Type
1	1	libre	Tmp-cl	
	2	10I1,3F24.0	pour chaque groupe de noeuds icod,(vcl(i),i=1,NDLN)	
	3 à m	13I6	(kdimp(i),i=1,NBN)	

	m+1	l6	0 (fin de séquence 1)	ASCII
2	1	libre	Tmp-cl	
	2	10l1,3F24.0	pour chaque groupe de noeuds icod,(vcl(i),i=1,NDLN)	
	3 à m	13l6	(kdimp(i),i=1,NBN)	
	m+1	l6	0 (fin de séquence 2)	
.	.	.	.	
.	.	.	.	
.	.	.	.	
n	1	libre	Tmp-cl	
	2	10l1,3F24.0	pour chaque groupe de noeuds icod,(vcl(i),i=1,NDLN)	
	3 à m	13l6	(kdimp(i),i=1,NBN)	
	m+1	l6	0 (fin de séquence n)	

Tmp-cl : TeMPs associé à chaque séquence des Conditions aux Limites.

NBN : Nombre de Noeuds partageant les mêmes Conditions aux limites.

NDLN : Nombre de Degrés de Liberté par Noeud.

ICOD : entier indiquant le CODE affecté à chaque degré de liberté.

vcl : table des VaLeurs des Conditions aux Limites associées à chaque degré de liberté.

kdimp : table des numéros de noeuds à Degré de liberté IMPosé.

**Exemple :**

Aux noeuds 1, 92 et 3567 les valeurs des conditions aux limites sont -12.351 et 116 associés aux d.d.l. 1 et 3 respectivement. Aux noeuds 44 et 5225 les valeurs imposées sont 10.009 et 115 associés aux d.d.l. 2 et 3 respectivement. Ce qui donne

```

0.0
1010000000-12.351      0.0      116.
      1   92  3567
0110000000 0.0      10.009     115.
      44  5255
0
    
```

## Fichier des connectivités

Les connectivités des éléments du maillage constituent une information du type statique. Le format associé se présente comme suit :

Lignes	Format	Variables	Type
1	libre	NELT, NNEL	ASCII
2 à NELT+1	libre	Noeud1,Noeud2, ...,NoeudNNEL	

NELT : Nombre d'ÉLéments Total.

NNEL : Nombre de Noeuds par Élément.

### Exemple :

Considérons un maillage à 2 éléments et 6 noeuds par élément.  
Ce qui donne

```

2   6
1   5   9   6   3   2
1   4   7   8   9   5

```

## Fichier des coordonnées

Les coordonnées des noeuds du maillage constituent une information du type statique. Le format associé se présente comme suit :

Lignes	Format	Variables	Type
1	libre	NNT, NDIM	ASCII
2 à NNT+1	libre	coord. X,Y,Z	

NNT : Nombre de Noeuds Total.

NDIM : Nombre de DIMensions.

**Exemple :**

Considérons un maillage à 5 noeuds et 2 dimensions. Ce qui donne

```
5 2
0.000 0.000
0.000 1.000
0.000 2.000
1.000 0.000
1.0 1.000
```

---

## Fichier des propriétés élémentaires

Les propriétés élémentaires constituent une information du type dynamique. Le format associé se présente comme suit :

Séquence	Lignes	Format	Variables	Type
	1	libre	NELT,NPREL,Tmp-pre	

1	2 à (NNT+1)	libre	in,(vpre(i),i= 1,...,NPREL)	ASCII
2	1	libre	NELT,NPREL,Tmp-pre	
	2 à (NNT+1)	libre	in,(vpre(i),i= 1,...,NPREL)	
.	.	.	.	
.	.	.	.	
.	.	.	.	
n	1	libre	NELT,NPREL,Tmp-pre	
	2 à (NNT+1)	libre	in,(vpre(i),i= 1,...,NPREL)	

Tmp-pre : Temps associé à chaque séquence des Propriétés Élémentaires.

NELT : Nombre d'Éléments Total.

NPREL : Nombre de Propriétés par Élément à Lire.

vpre : Table des Valeurs des Propriétés Élémentaires.

## Fichier des propriétés nodales

Les propriétés nodales constituent une information du type dynamique. Le format associé se présente comme suit :

Séquence	Lignes	Format	Variabiles	Type
----------	--------	--------	------------	------

1	1	libre	NNT,NPRNL,Tmp-prn	ASCII
	2 à (NNT+1)	libre	in,(vprn(i),i=1, ,NPRNL)	
2	1	libre	NNT, NPRNL,Tmp-prn	
	2 à (NNT+1)	libre	in,(vprn(i),i=1, ,NPRNL)	
.	.	.	.	
.	.	.	.	
.	.	.	.	
n	1	libre	NNT, NPRNL,Tmp-prn	
	2 à (NNT+1)	libre	in,(vprn(i),i=1, ,NPRNL)	

Tmp-prn : TeMPs associé à chaque séquence des Propriétés Nodales.

NNT : Nombre de Noeuds Total.

NPRNL : Nombre de Propriétés par Noeuds à Lire.

vprn : Table des Valeurs des Propriétés Nodales.

**Exemple :**

Considérons un maillage à 5 noeuds et 3 propriétés nodales par noeuds. Les trois valeurs nodales sont 115, 0.02 et 0 pour chaque noeud. Ce qui donne :

```

5   3   0.0
115.00  0.020  0.0
115.00  0.020  0.0
115.00  0.020  0.0
115.00  0.020  0.0
115.00  0.020  0.0
115.0  0.020  0.0
    
```

---

## Fichier des sollicitations

Les sollicitations concentrées et réparties constituent une information du type dynamique. Bien que les sollicitations concentrées et réparties sont stockées dans deux fichiers



différents, le format quant à lui est identique. Le format associé se présente comme suit :

Séquence	Lignes	Format	Variables	Type
1	1	libre	Tmp-sl	ASCII
	2	10I1,3F24.0	pour chaque groupe de noeuds icod,(vsldl(i),i=1,...,NDLN)	
	3 à m	13I6	(ksimp(i),i=1,...,NBN)	
	m+1	I6	0 (fin de séquence 1)	
2	1	libre	Tmp-sl	
	2	10I1,3F24.0	pour chaque groupe de noeuds icod,(vsldl(i),i=1,...,NDLN)	
	3 à m	13I6	(ksimp(i),i=1,...,NBN)	
	m+1	I6	0 (fin de séquence 2)	
.	.	.	.	
.	.	.	.	
.	.	.	.	
n	1	libre	Tmp-sl	
	2	10I1,3F24.0	pour chaque groupe de noeuds icod,(vsldl(i),i=1,...,NDLN)	
	3 à m	13I6	(ksimp(i),i=1,...,NBN)	
	m+1	I6	0 (fin de séquence n)	

Tmp-sl : TeMPs associé à chaque séquence des Sollicitations.

NBN : Nombre de Noeuds partageant les mêmes sollicitations.

NDLN : Nombre de Degrés de Liberté par Noeud.

icod : Entier indiquant le CODE affecté à chaque degré de liberté.

vsldl : Table des VaLeurs des SoLlicitations associées à chaque Degré de Liberté.

ksimp : Table des numéros de noeuds à Sollicitation IMPosé.

**Exemple :**

Aux noeuds 1, 92 et 3567 les valeurs des sollicitations imposées sont -12.351 et 116 associés aux d.d.l. 1 et 3 respectivement. Aux noeuds 44 et 5225 les valeurs imposées sont 10.009 et 115 associés aux d.d.l. 2 et 3 respectivement. Ce qui donne

0.0

```

1010000000-12.351          0.0          116.
      1    92  3567
0110000000 0.0            10.009        115.
      44  5255
0
    
```

## Fichier de solution initiale

La solution initiale constitue une information du type dynamique.  
Le format associé se présente comme suit :

Séquence	Lignes	Format	Variables	Type
1	1	libre ou non formaté	NNT,NDLN,Tmp-ini	ASCII (par défaut) ou binaire
	2 à (NNT+1)*NTTEMP	libre ou non formaté	(vddl(i),i=1,...,NDLN)	
2	1	libre ou non formaté	NNT,NDLN,Tmp-ini	
	2 à (NNT+1)*NTTEMP	libre ou non formaté	(vddl(i),i=1,...,NDLN)	
.	.	.	.	
.	.	.	.	
.	.	.	.	
n	1	libre ou non formaté	NNT,NDLN,Tmp-ini	
	2 à (NNT+1)*NTTEMP	libre ou	(vddl(i),i=1,...,NDLN)	

	non formaté	
--	-------------	--

Tmp-ini : TeMPs associé à chaque séquence de la solution INITiale.

NNT : Nombre de Noeuds Total.

NDLN : Nombre de Degrés de Liberté par Noeud.

NTTEMP : Nombre de Termes TEMPorels (=1 pour STEMP='STATIQ',=2 pour STEMP='EULER').

vddl : Table des Valeurs de chaque Degré De Liberté.

**Exemple :**

Considérons un maillage à 5 noeuds et 3 degrés de liberté par noeud. Les trois valeurs des degrés de liberté sont 0.0, 0.0 et 116.30 pour chaque noeud. Ce qui donne :

```

5   3   0.0
0.0  0.0  116.30
0.0  0.0  116.30
0.0  0.0  116.30
0.0  0.0  116.30
0.0  0.0  116.30
    
```

---

## Formats des fichiers de résultats

Cette section est consacrée à la présentation des formats des différents fichiers de résultats générés par DISPERSIM à la suite d'une simulation. On distingue deux catégories de résultats à savoir celles qui sont statiques et celles qui sont variables dans le temps. Les résultats dynamiques sont générés par séquences, correspondant chacune à un temps précis qui est systématiquement défini en premier. En toute logique un fichier de résultats statique ne comporte qu'une seule séquence indépendante du temps. Les fichiers de résultats générés par DISPERSIM sont :

- Fichier des degrés de liberté
- Fichier des erreurs numériques
- Fichier de post-traitement
- Fichier des résidus

## Fichier des degrés de liberté

Les degrés de liberté constituent une information du type dynamique. Le format associé se présente comme suit :

Séquence	Lignes	Format	Variables	Type
1	1	libre ou non formaté	NNT,NDLN,Tmp-fin	ASCII (par défaut) ou binaire
	2 à (NNT+1)*NTTEMP	13(1X,1PE24. 17) ou non formaté	(vddl(i),i=1,...,NDLN)	
2	1	libre ou non formaté	NNT,NDLN,Tmp-fin	
	2 à (NNT+1)*NTTEMP	13(1X,1PE24. 17) ou non formaté	(vddl(i),i=1,...,NDLN)	
.	.	..	..	
.	.	.	.	
n	1	non formaté	NNT,NDLN,Tmp-fin	
	2 à (NNT+1)*NTTEMP	13(1X,1PE24. 17) ou non formaté	(vddl(i),i=1,...,NDLN)	

Tmp-fin : TeMPs associé à chaque séquence de la solution INITiale.

NNT : Nombre de Noeuds Total.

NDLN : Nombre de Degrés de Liberté par Noeud.

NTTEMP : Nombre de Termes TEMPorels (=1 pour STEMP='STATIQ',=2 pour STEMP='EULER').

vddl : Table des Valeurs de chaque Degré De Liberté.

## Fichier des erreurs numériques

Les degrés de liberté constituent une information du type dynamique. Le format associé se présente comme suit :

Séquence	Lignes	Format	Variables	Type
1	1	libre	NNT,NDLN,Tmp-err	ASCII
	2 à (NNT+1)	13(1X,1PE24.17)	(verdl(i),i= 1,...,NDLN)	
2	1	libre	NNT,NDLN,Tmp-err	
	2 à (NNT+1)	13(1X,1PE24.17)	(verdl(i),i= 1,...,NDLN)	
.	.	.	.	
.	.	.	.	
.	.	.	.	
n	1	libre	NNT,NDLN,Tmp-err	
	2 à (NNT+1)	13(1X,1PE24.17)	(verdl(i),i= 1,...,NDLN)	

Tmp-err : Temps associé à chaque séquence des Erreurs numériques.

NNT : Nombre de Noeuds Total.

NDLN : Nombre de Degrés de Liberté par Noeud.

verdl : Table des Valeurs de l'Erreur sur les Degrés de Liberté.

## Fichier de post-traitement

Le post-traitement constitue une information du type dynamique.  
Le format associé se présente comme suit :

Séquence	Lignes	Format	Variables	Type
1	1	libre	NNT,NPOST,Tmp-pst	ASCII
	2 à (NNT+1)	13(1X,1PE24.17)	(vpst(i),i= 1,...,NPOST)	
2	1	libre	NNT,NPOST,Tmp-pst	
	2 à (NNT+1)	13(1X,1PE24.17)	(vpst(i),i= 1,...,NPOST)	
.	.	.	.	
.	.	.	.	
.	.	.	.	
n	1	libre	NNT,NPOST,Tmp-pst	
	2 à (NNT+1)	13(1X,1PE24.17)	(vpst(i),i= 1,...,NPOST)	

Tmp-pst : Temps associé à chaque séquence de PoST-traitement.

NNT : Nombre de Noeuds Total.

NPOST : Nombre de valeurs de POST-traitement.

vpst : Table des Valeurs du PoST-traitement.

## Fichier des résidus

Les résidus constituent une information du type dynamique. Le format associé se présente comme suit :

Séquence	Lignes	Format	Variables	Type
1	1	libre	NNT,NDLN,Tmp-res	ASCII
	2 à (NNT+1)	13(1X,1PE24.17)	(vredl(i),i= 1,...,NDLN)	
2	1	libre	NNT,NDLN,Tmp-res	
	2 à (NNT+1)	13(1X,1PE24.17)	(vredl(i),i= 1,...,NDLN)	
.	.	.	.	
.	.	.	.	
.	.	.	.	
n	1	libre	NNT,NDLN,Tmp-res	
	2 à (NNT+1)	13(1X,1PE24.17)	(vredl(i),i= 1,...,NDLN)	

Tmp-res Temps associé à chaque séquence des Résidus.

NNT : Nombre de Noeuds Total.

NDLN : Nombre de Degrés de Liberté par Noeud.

# Chapitre 6 Glossaire

## Approximation par éléments finis

Méthode de calcul par sous-domaines pour approcher les valeurs d'une fonction.

## Bilan d'oxygène dissous

Le bilan d'oxygène est établi en tenant compte de la dégradation de la Demande Biochimique en Oxygène (DBO) dissoute et déposée, de la demande benthique en oxygène et de la réaération atmosphérique. La DBO est choisie comme variable d'état représentative de la matière organique oxydable

## Cinétique de dégradation

Il s'agit d'un terme puits dans l'équation de transport-diffusion dont l'intensité est modulée par un coefficient de dégradation.

## Coliformes fécaux

En raison de la disponibilité des mesures, les coliformes fécaux sont les indicateurs les plus fréquemment utilisés pour analyser la contamination bactérienne. Le tau de décroissance des coliformes est gouverné par une cinétique de dégradation du premier ordre. Le coefficient de dégradation réunit les effets de sédimentation, de luminosité, de température, de prédation et de décroissance naturelle des populations. En pratique, des valeurs comprises entre 0 et 1 jour<sup>-1</sup> du coefficient de dégradation donnent de bons résultats de simulation par rapport aux observations.

## Conditions aux limites

Conditions devant être satisfaites par une fonction sur la frontière du domaine simulation.

## Condition de Cauchy

Condition qui stipule que la valeur d'une fonction à l'extérieur du domaine est connue.

## Condition de Dirichlet

Condition qui stipule que la valeur d'une fonction est connue.

## Condition de Neuman

Condition qui stipule que la variation d'une fonction suivant la direction normale à une frontière est nulle.

## Conditions initiales

Pour résoudre un problème non linéaire et/ou dépendant du temps il est impératif de préciser l'état initial du système. En pratique la question est transférée au choix d'une solution initiale qui satisfait en tout ou partie le système d'équations.

## Convergence d'un problème non linéaire

Un solveur non linéaire permet, à partir de conditions initiales données, de calculer par étapes successives la solution au problème posé. La solution est formée de proche en proche, à chaque étape un incrément ou une correction est déterminée pour mettre à jour la solution. Le processus de calcul atteint la convergence dès lors que la solution n'évolue plus, autrement dit l'incrément est tellement petit qu'il n'a plus aucune influence.



### **Convergence locale**

On parle de convergence locale ou de fausse convergence lorsque GMRES a convergé alors que la norme des résidus reste élevée.

### **Convection**

Voir accélération convective.

### **Connectivités**

Liste des numéros de noeuds géométrique associée à chaque élément fini du maillage.

### **Couvrant-découvrant**

Appellation pour exprimer la capacité du modèle hydrodynamique à prédire les zones couvertes et découvertes par les eaux. Les profondeurs peuvent ainsi être aussi bien positives que négatives. La convention de signe adoptée admet une région couverte à profondeur positive et une région découverte à profondeur négative. La ligne de démarcation entre les deux zones forme la ligne de rive.

### **Débit spécifique**

Le débit volumique par unité de largeur.

### **Degré de liberté**

Paramètre qui permet de définir l'état d'un système physique.

### **Discrétisation**

Opération qui assure la transformation d'un système temporel ou spatial continu en un système temporel ou spatial discret

### **Élément fini**

Entité géométrique pleine à une, deux ou trois dimensions constitué d'un nombre fini de noeuds disposés à l'intérieur ou sur le contour du domaine.

### **Équation de transport-diffusion**

On appelle équation de transport-diffusion une équation différentielle scalaire constituée de deux opérateurs :le transport et la diffusion.

### **Frontière fermée**

Frontière imperméable à écoulement normal nul. En l'absence d'adhérence, les vitesses sont tangentes à une frontière fermée.

### **Frontière ouverte**

Frontière perméable à écoulement normal non nul.

### **Formule de Taylor**

La formule de Taylor est destiné au calcul de la diffusivité de fond ou verticale égale au produit de la vitesse de cisaillement par la profondeur et par une constante variant entre 0.135 et 0.17.

### **Hydrodynamique**

Condition d'équilibre de l'eau en mouvement.

### **Hydrostatique**

Condition d'équilibre de l'eau au repos.

**Largeur de bande**

Entier positif obtenu par la différence entre les connectivités maximum et minimum.

**Ligne-de-ciel**

Méthode de stockage à largeur de bande variable des matrices éléments finis. Le coût mémoire du stockage en ligne-de-ciel est directement lié à la largeur de bande autrement dit à la numérotation du maillage. Il est minimum si la numérotation du maillage est optimisée.

**Ligne de rive**

Ligne qui observe la condition de profondeur nulle.

**Maillage**

Ensemble homogène d'éléments finis identiques ou complémentaires pour discrétiser le domaine de calcul.

**Matrice de préconditionnement**

Matrice particulière exploitée uniquement lorsqu'on fait appel à une méthode itérative pour effectuer la résolution.

**Matrice ILU**

Matrice de préconditionnement obtenue par la factorisation incomplète de la matrice éléments finis du système d'équations.

**Matrice de rigidité**

Matrice résultant de la discrétisation par éléments finis du modèle mathématique.

**Matrice tangente**

Résulte de la dérivation du vecteur résidu par rapport au vecteur solution.

**Méthode de résolution directe**

La procédure de résolution du système d'équations est effectuée en une seule étape.

**Méthode de résolution itérative**

La procédure de résolution du système d'équations est effectuée en plusieurs étapes successives appelées itérations.

**Modèle mathématique**

Ensemble d'équations, de lois constitutives et de conditions qui permet de reproduire le comportement d'un processus physique donné.

**Niveau d'eau**

Altitude  $h$  de la surface du plan d'eau par rapport à une référence.

**Niveau du fond**

Altitude  $z_f$  du fond du cours d'eau par rapport à une référence.

**NDIM**

Nombre de DIMensions.

**NDLN**

Nombre de Degrés de Liberté par Noeud.

**NDLT**

Nombre de Degrés de Liberté Total.

**NELT**

Nombre d'Éléments Total.

**NNEL**

Nombre de Noeuds par Élément.

**NNT**

Nombre de Noeuds Total.

**NPRNL**

Nombre de Propriétés Nodales à Lire.

**NTINI**

Nombre de Termes d'INItialisation.

**NTTEMP**

Nombre de Termes TEMPorels.

**Nombre de Peclet**

Nombre adimensionnel appelé également nombre de Reynolds local. Son expression est  $Pe=VD/\nu$  où  $V$  est la vitesse moyenne,  $D$  une dimension caractéristique de l'élément fini et  $\nu$  la viscosité cinématique de l'écoulement.

**Peau du maillage**

Maillage éléments finis formant la frontière du maillage du domaine de calcul.

**Pilotage de la résolution**

Procédure qui vise à assister le solveur en faisant varier progressivement un ou plusieurs paramètres.

**Préconditionnement**

Opération algébrique visant à transformer, au moyen d'une matrice de préconditionnement, le système d'équations de départ en un système d'équations équivalent mieux adapté pour la procédure de résolution par une méthode itérative.

**Profondeur**

Notée  $H$ , elle résulte de la différence entre le niveau de la surface libre  $h$  et celui du fond  $z_f$ .

**Propriétés élémentaires**

L'ensemble des paramètres propres à l'élément.

**Propriétés globales**

L'ensemble des paramètres propres au maillage.

**Propriétés nodales**

L'ensemble des paramètres propres aux noeuds.

**Puits**

On désigne par puits dans l'équation de transport-diffusion d'une fonction un terme dont le rôle consiste à consommer ladite fonction.

### **Relaxation**

Scalaire strictement positif qui pondère l'incrément de solution pour assurer la stabilité de la convergence d'un problème non linéaire. On parle de sur-relaxation s'il est supérieur à 1 et de sous-relaxation dans le cas contraire. Il est en règle général inférieur à 1 pour les situations de convergence difficile.

### **Résidu**

C'est une mesure qui permet de rendre compte à quel point une solution satisfait le modèle mathématique. Le résidu est nul si une solution satisfait exactement le modèle mathématique. Dans le contexte éléments finis, la notion de résidu est très importante. C'est un vecteur dont chaque composante est associé à chaque degré de liberté. L'opération de validation d'une solution consistera entre autre chose à s'assurer qu'elle minimise bien chaque composante du résidu. En pratique le contrôle de la norme L2 discrète du résidu devrait suffire à nous affranchir de vérifier ses composantes une à une.

### **Résolution**

Dans le contexte éléments finis, il s'agit d'une opération algébrique visant à déterminer la solution qui satisfait le système d'équations découlant de la discrétisation du modèle mathématique. Elle peut être effectuée de deux façons, soit par une méthode directe soit par une méthode itérative.

### **Schéma d'Euler**

Méthode de discrétisation du premier ordre de la dérivée d'une fonction  $f$  par rapport au temps  $t$ . Formellement le problème s'énonce comme suit :  $\partial f/\partial t \approx (f^{t+\Delta t} - f^t)/\Delta t$ , où  $\Delta t$  est l'incrément de temps.

### **Solveur**

Algorithme qui permet de calculer la solution du système d'équations linéaire ou non linéaire.

### **Source**

On désigne par source dans l'équation de transport-diffusion d'une fonction un terme dont le rôle consiste à alimenter ladite fonction.

### **Système continu**

Un système est continu s'il possède un nombre infini de degrés de liberté.

### **Système discret**

Un système est discret s'il possède un nombre fini de degrés de liberté. Sous forme matricielle on peut l'écrire comme suit :  $[K]\{U\} = \{F\}$  où  $[K]$  est la matrice de rigidité caractérisant le système,  $\{F\}$  est le vecteur des sollicitations connues et  $\{U\}$  est le vecteur solution inconnue.

### **Système linéaire**

La matrice du système d'équations est indépendante de la solution.

### **Système non linéaire**

Contrairement à un système linéaire, la matrice dépend de la solution.