



POLYTECHNIQUE
MONTRÉAL

GLOW, un générateur de géométries surfaciques non-natives

A. Hébert

2024/07/07

Table des matières

Introduction
R&D
Le prototype SALOMON
Le générateur GLOW
Ressources

Introduction
R&D
Le prototype SALOMON
Le générateur GLOW
Ressources

Introduction

Introduction

R&D

Le prototype
SALOMON

Le générateur GLOW

Ressources

- Le traitement des géométries est un enjeu majeur en physique des réacteurs nucléaires.
- Nous allons présenter le traitement et l'utilisation des géométries présentes lors du **calcul de réseau**.
- Trois types de représentation géométriques sont couramment utilisées en France et au Canada
 1. Géométries gigognes (Apollo1/2/3, Dragon1/2/3/4/5, solveurs “coeur”)
 2. Géométries surfaciques (Apollo2/3, Dragon5)
 3. Géométries combinatoires (MCNP, Serpent, Tripoli, Moret)

Définitions:

1. Une géométrie **native** est une géométrie définie avec les mot-clés-valeurs du code de réseau ou du code de coeur.
2. Une géométrie **non-native** est une géométrie définie avec un outil externe (Silène, Salomon, Charm, Alamos ou **GLOW**).

Développement du générateur GLOW:

L'outil GLOW (*Geometry Layout for OpenCascade Workflows*) permettra la génération de géométries **surfaciques non-natives** en utilisant l'*Application Programming Interface* (API) de OpenCascade. Cet outil sera disponible sous license **Open Source**.

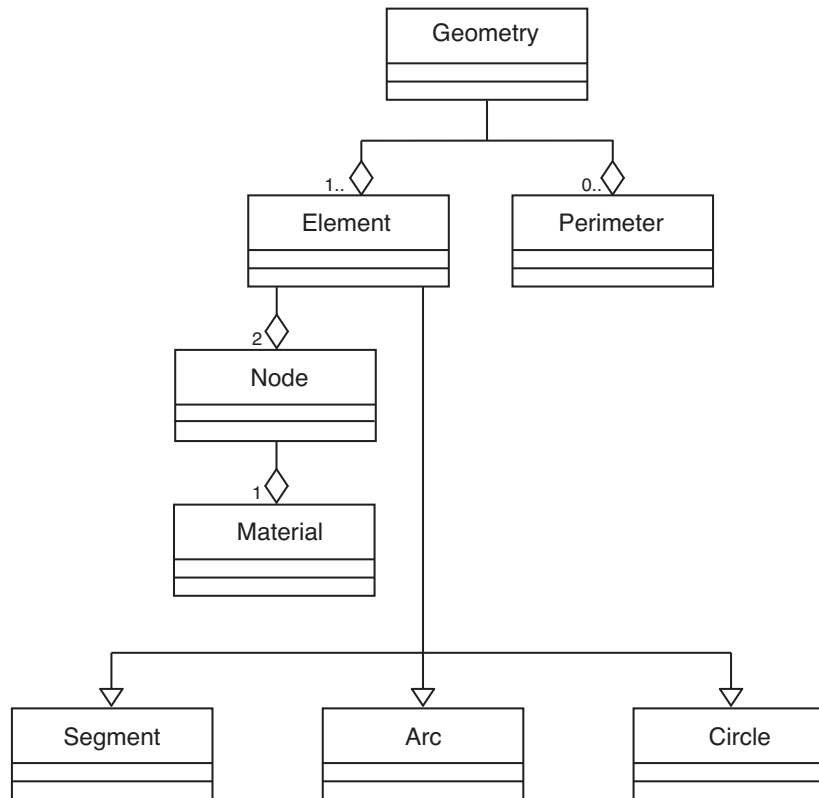
Open CASCADE Technology

Géométries surfaciques

Introduction

R&D
Le prototype
SALOMON
Le générateur GLOW
Ressources

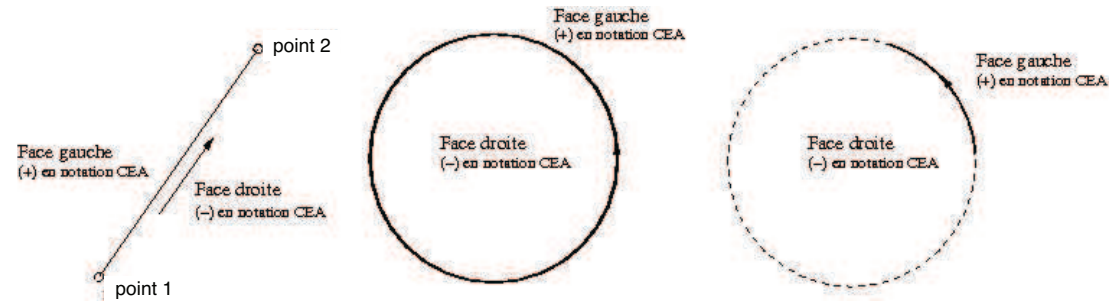
- Introduites par Richard Sanchez dans Apollo2 en 1993. Utilisées dans Dragon5 (voir diagramme UML).
- Limitées aux géométries 2D.
- Ne présentent aucune limitation pour aucun design 2D.
- Géométries natives (pour les plus simples) ou non-natives.
- Peuvent être décrites en langage PostScript directement.



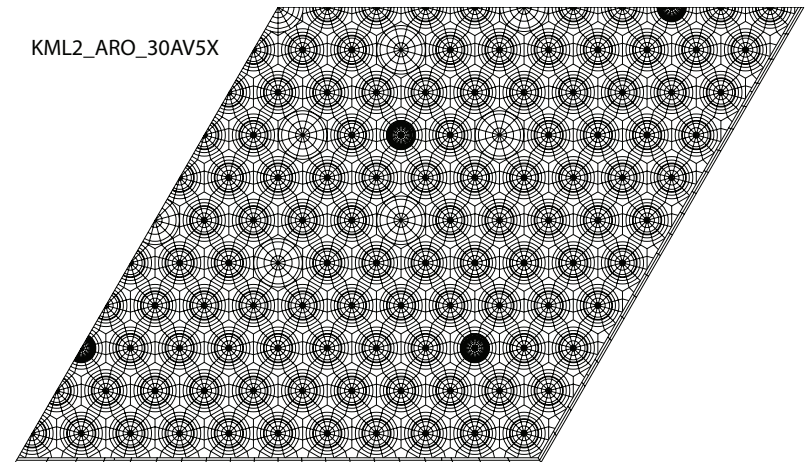
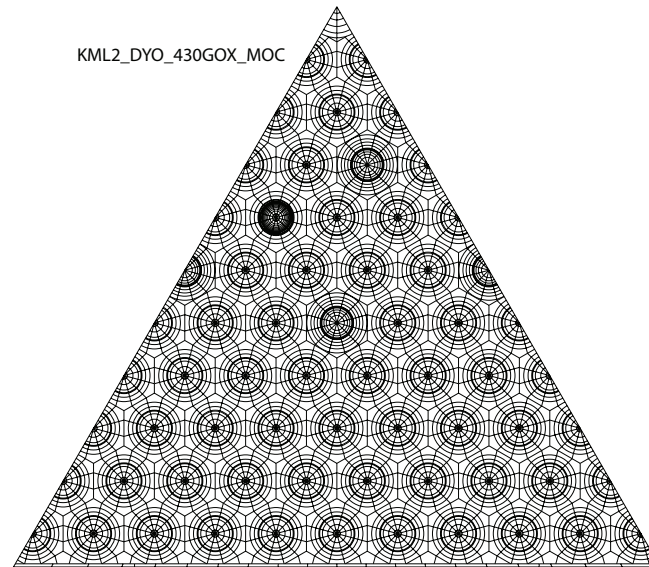
Introduction

R&D
Le prototype
SALOMON
Le générateur GLOW
Ressources

Éléments surfaciques



Exemples de géométries surfaciques



Modèles géométriques

Exemples de géométries surfaciques

Introduction

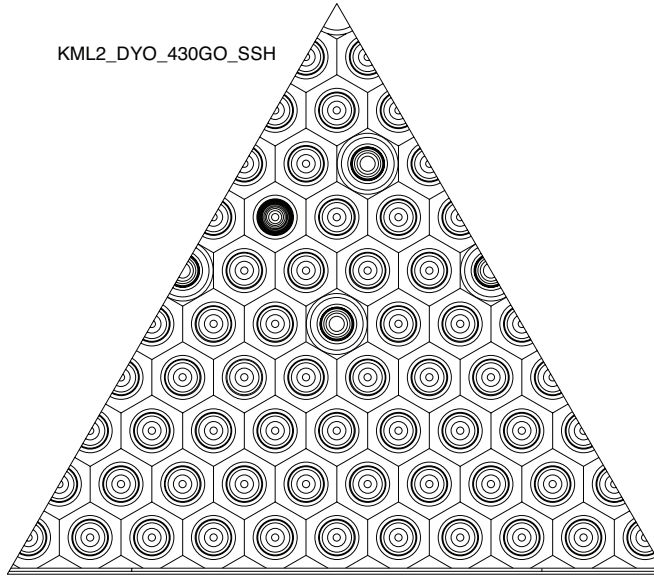
R&D

Le prototype
SALOMON

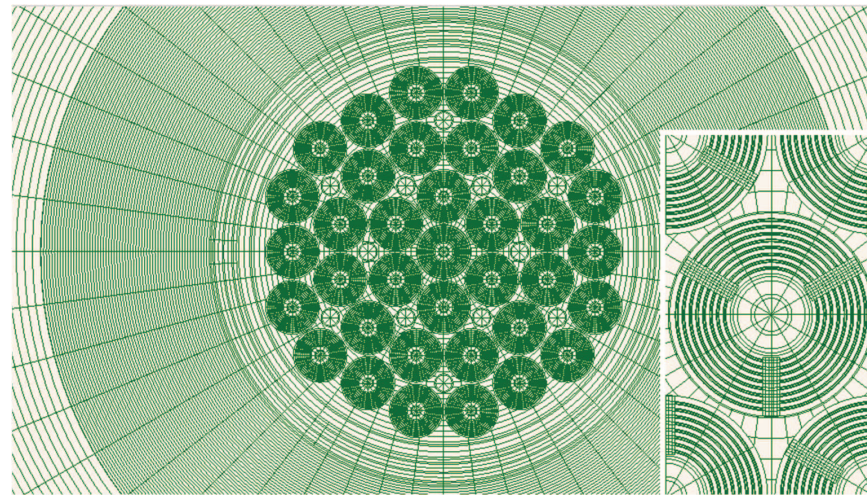
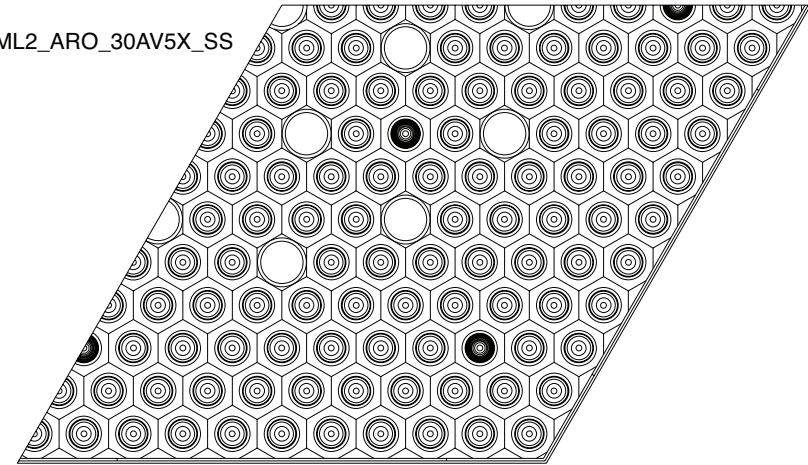
Le générateur GLOW

Ressources

KML2_DYO_430GO_SSH



KML2_ARO_30AV5X_SS



Le développement des géométries surfaciques pour le calcul de réseau est le résultat de décennies de R&D en France.

Introduction

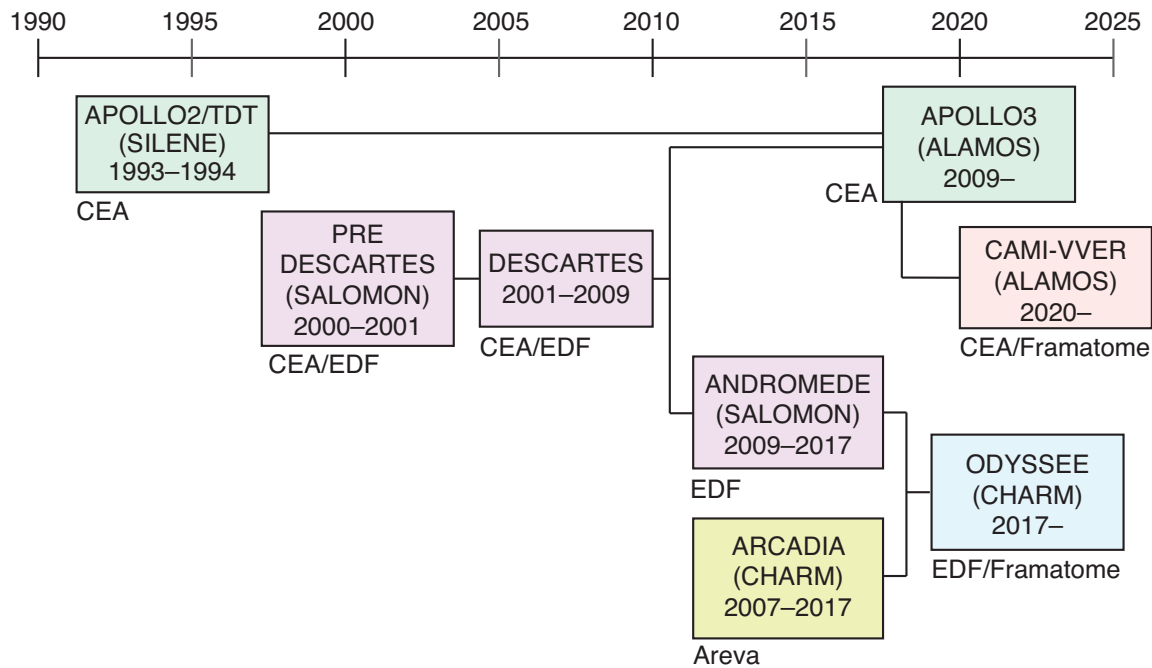
R&D

Le prototype

SALOMON

Le générateur GLOW

Ressources



Introduction

R&D

Le prototype
SALOMON

Le générateur GLOW

Ressources

Développement initial 1993–1994

Le développement initial des géométries surfaciques comporte deux composantes:

SILENE: Outil graphique de saisie d'une géométrie surfacique non-native. Silène a été programmé par Zarko Stankovsky en langage Java.

R. Sanchez and Z. Stankovsky, SILENE & TDT: A code for CP calculations in XY geometry, CEA-CONF-11953 (1993).

TDT: Outil de génération de trajectoires (tracking) d'une géométrie surfacique dans l'environnement APOLLO2. TDT a été programmé par Richard Sanchez, assisté d'une thésarde au Ph. D. (Naïma Lyoussi-Charrat).

N. Lyoussi-Charrat, Calcul de transport neutronique dans le code APOLLO2 par la méthode ..., thèse Ph. D. (1994).

- Cette contribution SILENE/TDT est limitée aux géométries non-natives.
- La contribution de Naïma Lyoussi-Charrat sur les probabilités de collision n'a pas été intégrée à APOLLO2. Par contre, la méthode des caractéristiques basée sur TDT a été implémentée.

Prototypage Pré-Descartes 2000–2001

- Le projet Pré-Descartes avait pour but de réaliser des prototypes rapides de solutions innovantes et audacieuses en amont du projet Descartes dont le début était fixé à Septembre 2021.
 - ◆ Un prototype rapide est une expérimentation informatique menée sur une courte période de temps et basée sur des composants Open-Source (Salome, Python2, Dragon2).

Les analyses suivantes ont été menées:

1. Conception du prototype G2S : pour transformer une géométrie gigogne 2D en géométrie surfacique ou une géométrie surfacique en géométrie combinatoire.
 - \approx 13,000 lignes de Fortran-90
2. Conception du prototype SALT : pour effectuer un tracking 2D sur une géométrie surfacique. Ce tracking est compatible avec la méthode des probabilités de collision (PIJ) et avec la méthode des caractéristiques (MOC).

X. Warin, Notice théorique de la méthode des caractéristiques 2D et du générateur de trajectoires SALT (2002).

 - \approx 7,000 lignes de Fortran-90
3. Conception du prototype SALOMON pour générer une géométrie surfacique non-native en utilisant le module GEOM de SALOME.
 - \approx 1,900 lignes de Python2

Situation actuelle 2017–

1. La nouvelle interface graphique, ALAMOS, spécifiquement développée pour APOLLO3 avec une interface de programmation d'application (API) Python, est le successeur de SILÈNE.

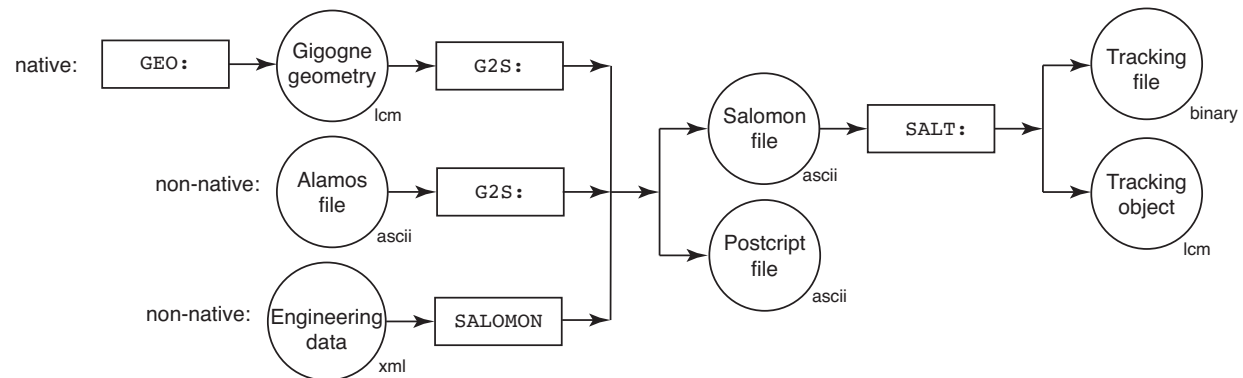
D. Tomatis *et al.*, Overview of SERMA's Graphical User Interfaces for Lattice Transport Calculations (2022).

2. Le code APOLLO3 permet de traiter les géométries surfaciques natives et non-natives (générées par SILÈNE ou par ALAMOS) avec la méthode des probabilités de collision (PIJ), avec la méthode des caractéristiques (MOC) ou avec la méthode aux ordonnées discrètes (solveur MINARET) si les éléments surfaciques sont des segments de droite.

J.-F. Vidal *et al.*, New reference Apollo3 calculation scheme for light water reactors, (2020).

3. Le code DRAGON5 permet de traiter les géométries surfaciques natives et non-natives (générées par SALOMON ou par ALAMOS) avec la méthode des probabilités de collision (PIJ) ou avec la méthode des caractéristiques (MOC). De nouvelles symétries ont été ajoutées au module SALT : pour pouvoir traiter les assemblages VVER.

A. Hébert, DRAGON5 and DONJON5, the contribution of ÉPM to the SALOME platform, (2020).



Le prototype SALOMON

Introduction

R&D

Le prototype
SALOMON

Le générateur GLOW

Ressources

Le prototype SALOMON, développé par Yann Pora de EDF R&D, est une application Python2 qui procède en deux étapes:

1. Lecture de la description technologique XML de la géométrie et génération d'une représentation surfacique BRep en utilisant les API du module GEOM de SALOME. Seules les géométries des réacteurs PWR peuvent être traitées pour l'instant.
2. Transcription de la représentation BRep de GEOM au format TDT utilisé par les codes APOLLO2 (MOC) ou DRAGON5 (module SALT:).

La description technologique XML du cas UOX_TBH_1cell.xml suit:

```
<Header ID="START">
<HeaderComponentID> MyLattice_HTP900_assemblage_mox__CellConfig__STD_ </HeaderComponentID>
<BoundCond> mirror </BoundCond>
<CalcSym> 4 </CalcSym>
<Compound ID="MyLattice_HTP900_assemblage_mox__CellConfig__STD_">
<MainComponentID> MyLattice_HTP900_assemblage_mox__CellConfig__STD__PLT </MainComponentID>
<XCoordinateList> 0.0 </XCoordinateList>
<YCoordinateList> 0.0 </YCoordinateList>
</Compound>
<Lattice ID="MyLattice_HTP900_assemblage_mox__CellConfig__STD__PLT">
<XNbComponents> 3 </XNbComponents><Y NbComponents> 3 </Y NbComponents>
<ISym> 1 </ISym>
<ComponentIDList>
  Lame_C LameH Lame_C
  Lame_V C0101 Lame_V
  Lame_C LameH Lame_C
</ComponentIDList>
</Lattice>
<Cell ID="Lame_C">
<MaterialList> TSTR.'TMil_MOC'.'MODE' </MaterialList>
<GeomCellID> GLame_Coin </GeomCellID>
</Cell>
<Cell ID="Lame_V">
<MaterialList> TSTR.'TMil_MOC'.'MODE' </MaterialList>
<GeomCellID> GLame_Verticale </GeomCellID>
</Cell>
<Cell ID="LameH">
<MaterialList> TSTR.'TMil_MOC'.'MODE' </MaterialList>
<GeomCellID> GLame_Horizontale </GeomCellID>
</Cell>
```

Le prototype SALOMON

Introduction

R&D

Le prototype
SALOMON

Le générateur GLOW

Ressources

```
<GeomCell ID="GLame_Coin">
<Type> uniform </Type>
<XSize> 0.042 </XSize>
<YSize> 0.042 </YSize>
<XNbMesh> 1 </XNbMesh>
<YNbMesh> 1 </YNbMesh>
</GeomCell>
<GeomCell ID="GLame_Verticale">
<Type> uniform </Type>
<XSize> 0.042 </XSize>
<YSize> 1.260 </YSize>
<XNbMesh> 1 </XNbMesh>
<YNbMesh> 1 </YNbMesh>
</GeomCell>
<GeomCell ID="GLame_Horizontale">
<Type> uniform </Type>
<XSize> 1.260 </XSize>
<YSize> 0.042 </YSize>
<XNbMesh> 1 </XNbMesh>
<YNbMesh> 1 </YNbMesh>
</GeomCell>
<Cell ID="C0101">
<GeomCellID> MAILCOMB </GeomCellID>
<MaterialList> TSTR.'TMil_MOC'. 'COMB39'.1
                TSTR.'TMil_MOC'. 'COMB39'.2
                TSTR.'TMil_MOC'. 'COMB39'.3
                TSTR.'TMil_MOC'. 'COMB39'.4
                TSTR.'TMil_MOC'. 'VIDE'
                TSTR.'TMil_MOC'. 'GAINCOMB'
                TSTR.'TMil_MOC'. 'MODE'
</MaterialList>
</Cell>
<GeomCell ID="MAILCOMB">
<Type> standard </Type>
<XSize> 1.26000000e+00 </XSize>
<YSize> 1.26000000e+00 </YSize>
<OuterRadiusList> 0.288712 0.365195 0.397962 0.4083 0.4165 0.4775 </OuterRadiusList>
<NbSectorList> 1 1 1 1 1 1 </NbSectorList>
<InitialAngleList> 0.0 0.0 0.0 0.0 0.0 0.0 </InitialAngleList>
</GeomCell>
</Header>
```

Le prototype SALOMON

Introduction

R&D

Le prototype
SALOMON

Le générateur GLOW

Ressources

L'exécution de SALOMON demande a définition d'une variable d'environnement dans le fichier `.profile`:

```
export SALOME_KERNEL=$HOME/salome_6.6.0/KERNEL_6.6.0 # Salome kernel
```

SALOMON est exécuté par la commande suivante:

```
cd SALOMON  
./SALOMON.sh -o MesSorties DonneesXML/UOX_TBH_1cell.xml
```

Cette exécution produira la création de deux fichiers:

MesSorties/UOX_TBH_1cell.dat

MesSorties/UOX_TBH_1cell.mat

Le fichier `UOX_TBH_1cell.dat` est un fichier surfacique au format TDT qui peut être directement utilisé par DRAGON5. Le fichier `UOX_TBH_1cell.mat` donne la correspondance entre les identificateurs de milieux alphanumériques (utilisés par XML et par APOLLO2) et entiers (utilisés par DRAGON5).

Le prototype SALOMON

Introduction

R&D

Le prototype
SALOMON

Le générateur GLOW

Ressources

L'exécution de DRAGON5 a besoin de trois fichiers et d'un directory nommés

UOX_TBH_1cell_g2s.access: script bash de récupération du fichier surfacique TDT

UOX_TBH_1cell_g2s.x2m: fichier de données DRAGON5

UOX_TBH_1cell_g2s.save: script bash de sauvegarde des résultats

UOX_TBH_1cell_g2s_proc: directory qui contiendra le fichier UOX_TBH_1cell.dat.

UOX_TBH_1cell_g2s.access

```
#!/bin/sh
if [ $# = 0 ]
then
echo "usage: UOX_TBH_1cell_g2s.access directory" 1>&2
exit 1
fi
MACH='uname -s'
Sysx="'echo $MACH | cut -b -6'"
if [ $Sysx = "CYGWIN" ]; then
MACH='uname -o'
elif [ $Sysx = "Darwin" ]; then
MACH='uname -sm'
elif [ $Sysx = "SunOS" ]; then
MACH='uname -sm'
fi
ln -s "$1"/data/UOX_TBH_1cell_g2s_proc/UOX_TBH_1cell.dat .
ls -l
echo "UOX_TBH_1cell_g2s access script terminated"
```

Le prototype SALOMON

Introduction

R&D

Le prototype
SALOMON

Le générateur GLOW

Ressources

UOX_TBH_1cell_g2s.x2m

```
SEQ_ASCII UOX_TBH :: FILE './UOX_TBH_1cell.dat' ;
SEQ_ASCII UOX_TBH.ps :: FILE './UOX_TBH_1cell_g2s.ps' ;
SEQ_ASCII Lines.m :: FILE './Lines.m' ;
MODULE G2S: SALT: TLM: END: ;
INTEGER an2d := 8 ;
REAL densur := 20.0 ;
LINKED_LIST TRACK ;
SEQ_BINARY TF_EXC ;
*
UOX_TBH.ps := G2S: UOX_TBH :: DRAWMIX ;
TRACK TF_EXC := SALT: UOX_TBH ::
  EDIT 3
  ALLG
  TSPC EQW2 <<an2d>> <<densur>> REND LONG
;
Lines.m := TLM: TRACK TF_EXC ::
  EDIT 99
  NTPO 1
  DIRECTIONS NoPause DIR 0 ;
ECHO "test UOX_TBH_1cell_g2s completed" ;
END: ;
```

UOX_TBH_1cell_g2s.save

```
#!/bin/sh
#
if [ $# = 0 ]
then
  echo "usage: UOX_TBH_1cell_g2s.save directory" 1>&2
  exit 1
fi
echo "access UOX_TBH_1cell_g2s.save"
MACH='uname -s'
Sysx="'echo $MACH | cut -b -6'"
if [ $Sysx = "CYGWIN" ]; then
  MACH='uname -o'
elif [ $Sysx = "AIX" ]; then
  MACH='uname -s'
else
  MACH='uname -sm | sed 's/[ ]/_/'
fi
ls -l
mv *.ps $1/"$MACH"
mv *.m $1/"$MACH"
echo "UOX_TBH_1cell_g2s.save completed"
```

Le prototype SALOMON

Introduction

R&D

Le prototype
SALOMON

Le générateur GLOW

Ressources

L'exécution de DRAGON5 est réalisée par la commande suivante:

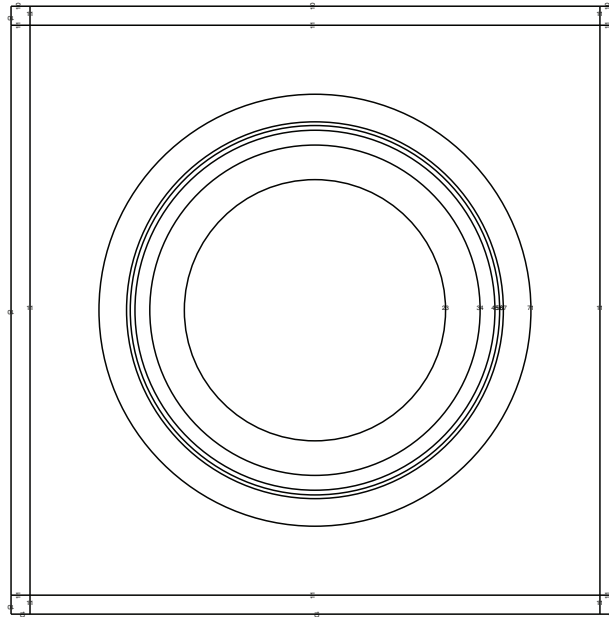
```
cd Version5_ev3318/Dragon  
./rdragon UOX_TBH_1cell_g2s.x2m
```

Cette exécution produira la création de deux fichiers:

UOX_TBH_1cell_g2s.ps : Fichier postscript produit par le module G2S : et décrivant la géométrie surfacique

Lines.m: Fichier Matlab permettant de tracer les lignes d'intégration générées par le module TLM:

Fichier UOX_TBH_1cell_g2s.ps



Le prototype SALOMON

Fichier généré par le script Matlab Lines.m

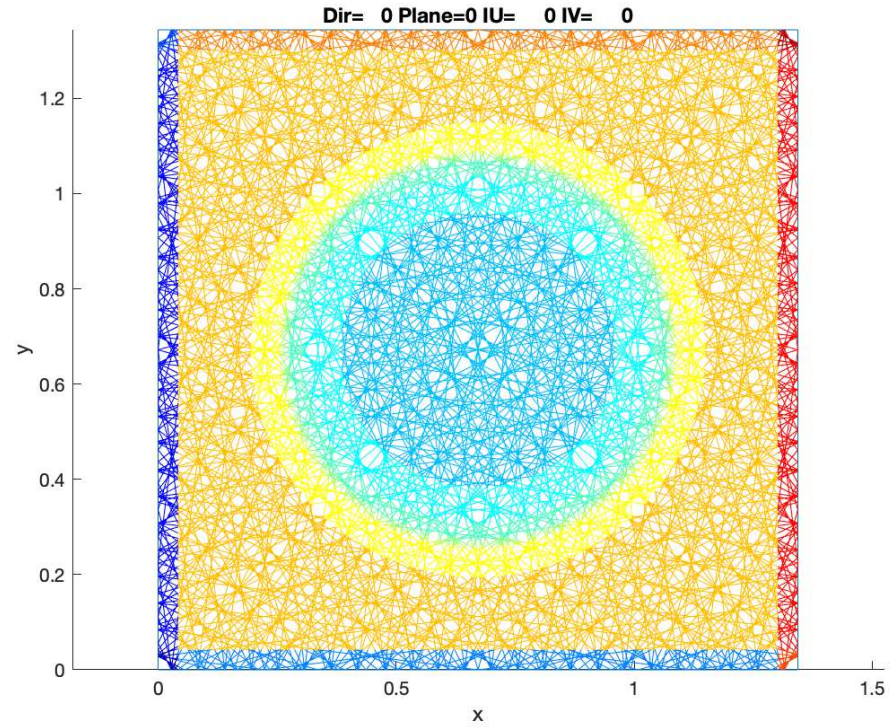
Introduction

R&D

Le prototype
SALOMON

Le générateur GLOW

Ressources



Le générateur GLOW

Introduction

R&D

Le prototype
SALOMON

Le générateur GLOW

Ressources

Le générateur GLOW est un outil actuellement développé à Polytechnique Montréal dans le but d'offrir une alternative Open Source à ALAMOS. Les principales différences entre ALAMOS et GLOW sont:

- Le fichier surfacique généré par GLOW est au format TDT APOLLO2. ALAMOS utilise un nouveau format pour APOLLO3.
- GLOW sera basé sur Open Cascade plutôt que sur SALOME afin de faciliter sa configuration.
- L'approche en deux étapes de SALOMON, tel que proposé par Yann Pora, sera utilisée.
- La première version de production va cibler les géométries hexagonales.
- La géométrie sera construite par le modèle *constructive solid geometry* (CSG) de Open Cascade, permettant des opérations booléennes sur les polygones 2D et 3D.

openCASCADE Engine

L'outil GLOW sera développé en quatre phases:

1. Familiarisation avec le prototype SALOMON de 2001 (voir document Descartes)
2. Adaptation du prototype SALOMON autour de OpenCascade (géométrie cartésienne). La version actuelle fonctionne autour de SALOME.
3. Création d'une syntaxe XML pour décrire les assemblages et colorsets en géométrie hexagonale.
4. Généralisation du prototype pour les géométries hexagonales (S30, RA60, R120, et COMPLETE) avec boîtier et/ou stiffeners en 2D. Généralisation ultérieure aux colorsets.

■ Academic:

Alain Hébert (alain.hebert@polymtl.ca)

■ Merlin website:

DRAGON5/DONJON5: <http://merlin.polymtl.ca>

■ Archives website:

- ◆ Accès à l'information sur les développements Dragon5
- ◆ Autres contributions académiques

<http://merlin.polymtl.ca/archives.htm>

■ Textbook:

A. Hébert, Applied Reactor Physics, Third Edition, PIP, 2020.

