

Réduction de modèle

Vincent Vadez

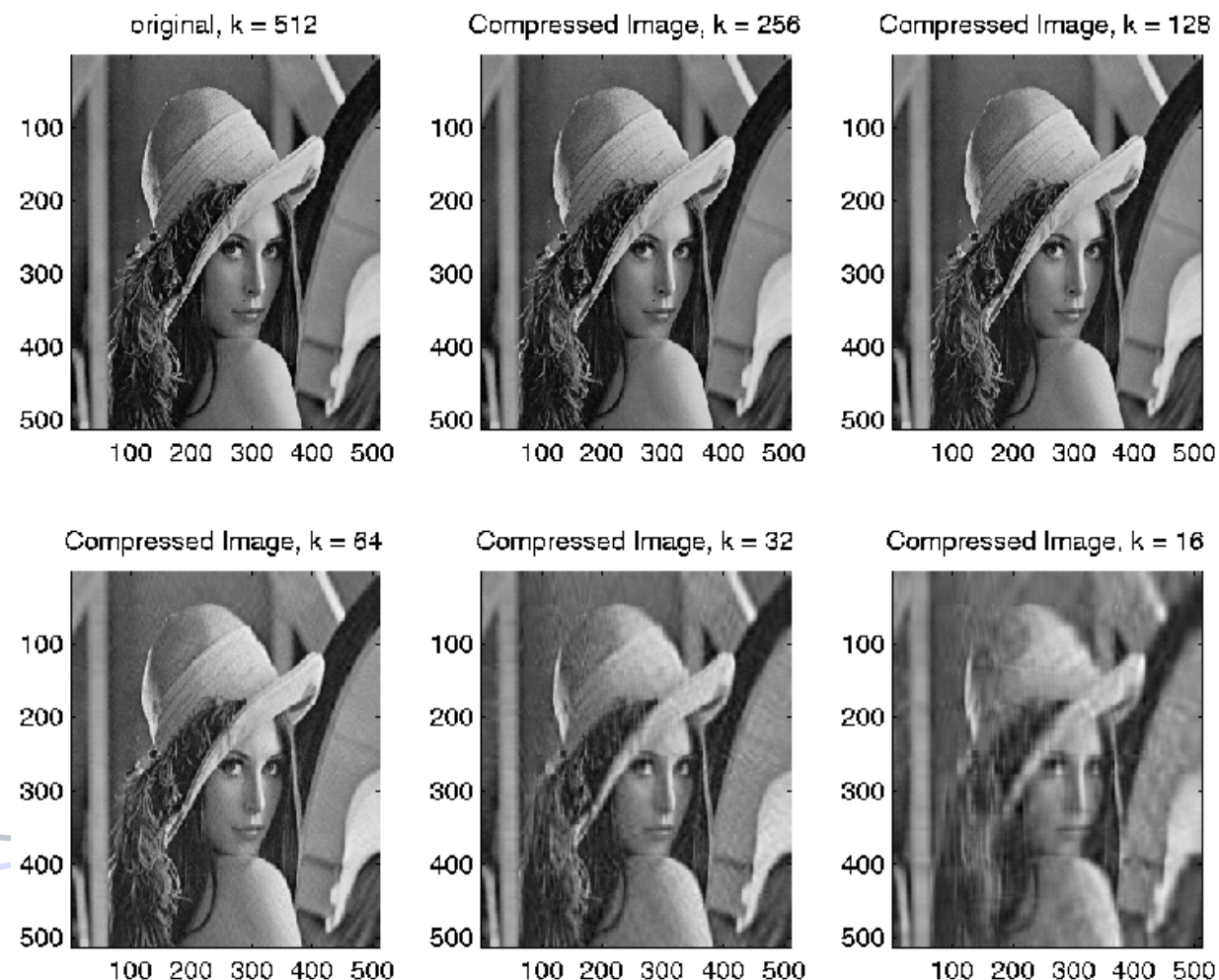
Dorea Technology

vincent.vadez@dorea.fr

La réduction de modèle, de quoi s'agit-il?

Qu'évoque la réduction de modèle pour vous?

- Vous en avez déjà tous fait, peut-être sans le savoir! (SVD projet MAM3, interpolation, changement de base réduite d'un système linéaire, développement de Taylor...)



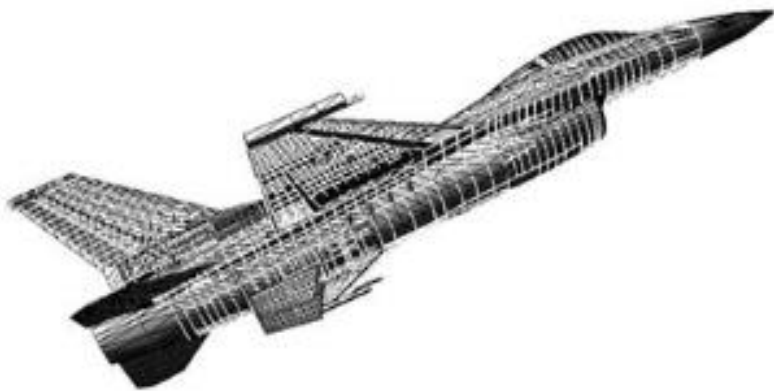
- Réduire la complexité de calcul des modèles mathématiques dans les simulations numériques
- **Réduction de la dimension d'espace d'état ou des degrés de liberté** associés au modèle, une approximation du modèle d'origine est calculée, communément appelée modèle d'ordre réduit

Contexte

- Modèles réduits: utiles quand impossible de mener des simulations numériques avec modèle complet
- Causes: limitations des ressources de calcul ou aux exigences du paramètre de simulation (temps réel, requêtes multiples)

De multiples applications

- Premiers travaux : Lumley, J.L. (1967). *The Structure of Inhomogeneous Turbulence*



(a)

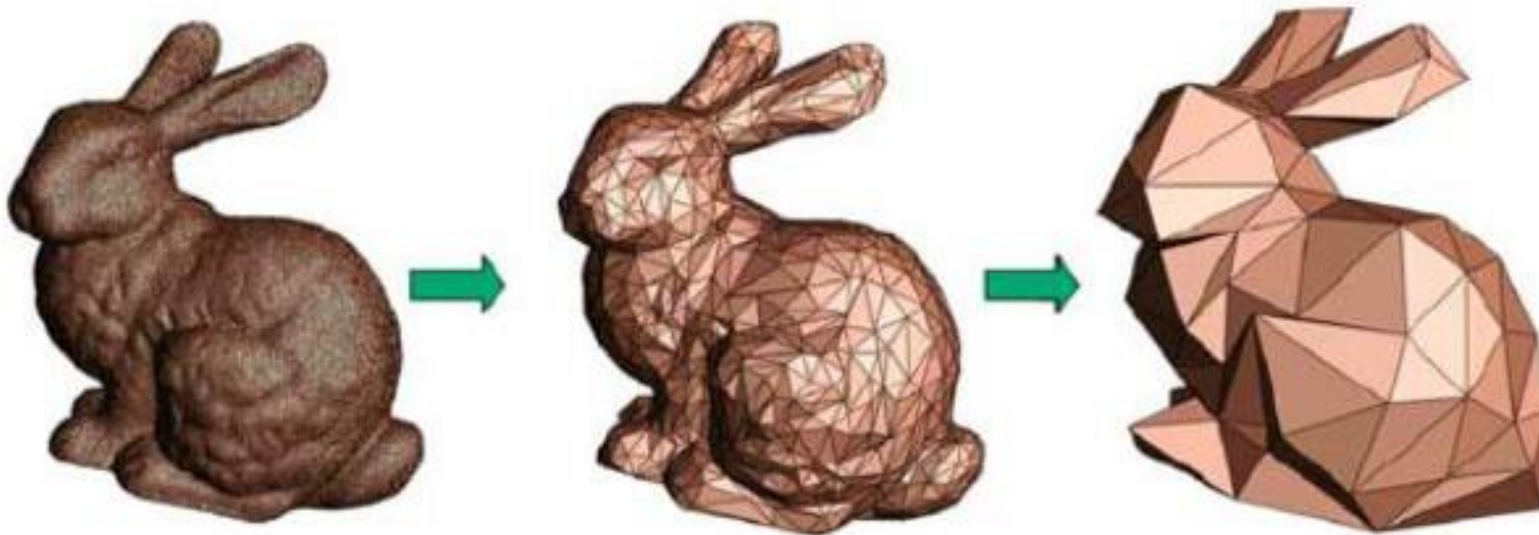


(b)

Fig. 2. F-16 aeroelastic model. (a) Detailed FE structural model and (b) fluid surface grid.

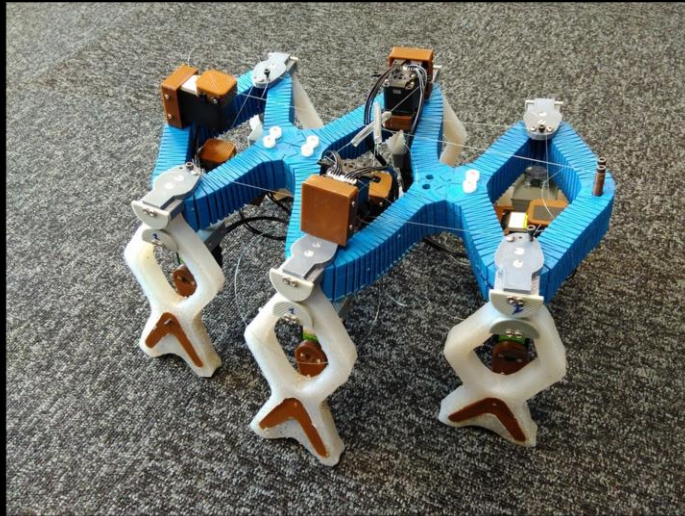
Full order model of an F16 fighter-aircraft with over 2.1 million degrees of freedom, reduced to a model of just 90 degrees of freedom (Lieu, T.; Farhat, C.; Lesoinne, M. (2006). "Reduced-order fluid/structure modeling of a complete aircraft configuration". *Computer Methods in Applied Mechanics and Engineering*. **195** (41–43): 5730–5742.)

De multiples applications



Réduction de modèle géométrique d'un maillage surfacique

De multiples applications



Real Robot



Reduction with coarse mesh



Reduction with fine mesh



SOFA Inria Lille

Contraintes

- Modèle réduit \rightarrow erreur d'approximation
- Besoin de conserver propriétés et caractéristiques du modèle complet
- Modélisation sur ordre réduit se doit d'être efficace et robuste au niveau informatique

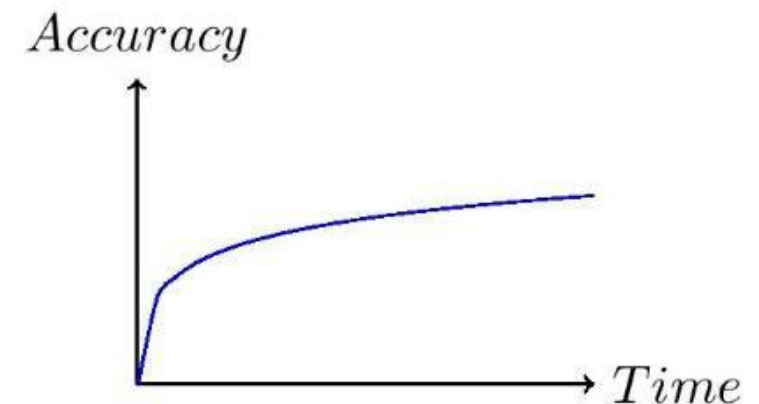
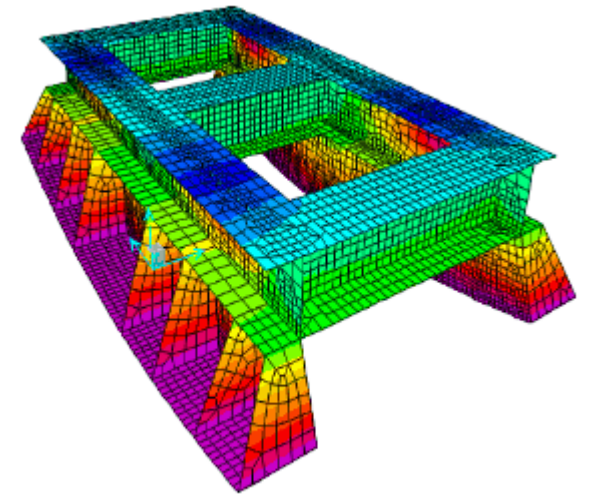
A retenir

Modèle initial trop volumineux de par le nombre de pas de temps, le nombre de degrés de liberté et/ou le nombre de paramètres. Ceci entraîne des limitations en termes de moyens matériels (CPU, mémoire), humains (préparation et analyse des résultats) et financiers.

Motivations

- Maillage d'objets (triangles ou tétraèdres) indispensable pour calculs physiques via éléments finis
- Inconvénients: très chronophage (plus le nombre d'éléments du maillage augmente, plus le calcul physique est fiable, mais plus le temps de calcul augmente).
- En laboratoire: précision au détriment du temps de calcul

En entreprise: compromis entre temps de calcul et précision souhaitée (modèle d'origine rarement acceptable niveau temps de calculs...)



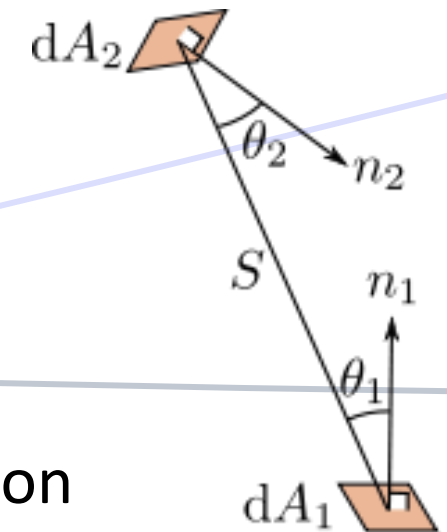
Exemple: calcul radiatif d'un satellite

- Problème quadratique en fonction du nombre d'éléments de maillage surface n ($O(n^2)$). Une pièce ($\sim 1\%$ du sat) peut avoir plusieurs millions de faces. (considérer que chaque face voit potentiellement les $n-1$ autres).



$$\dot{Q}_{1 \rightarrow 2} = \sigma A_1 F_{1 \rightarrow 2} (T_1^4 - T_2^4)$$

$$F_{1 \rightarrow 2} = \frac{1}{A_1} \int_{A_1} \int_{A_2} \frac{\cos \theta_1 \cos \theta_2}{\pi s^2} dA_2 dA_1$$

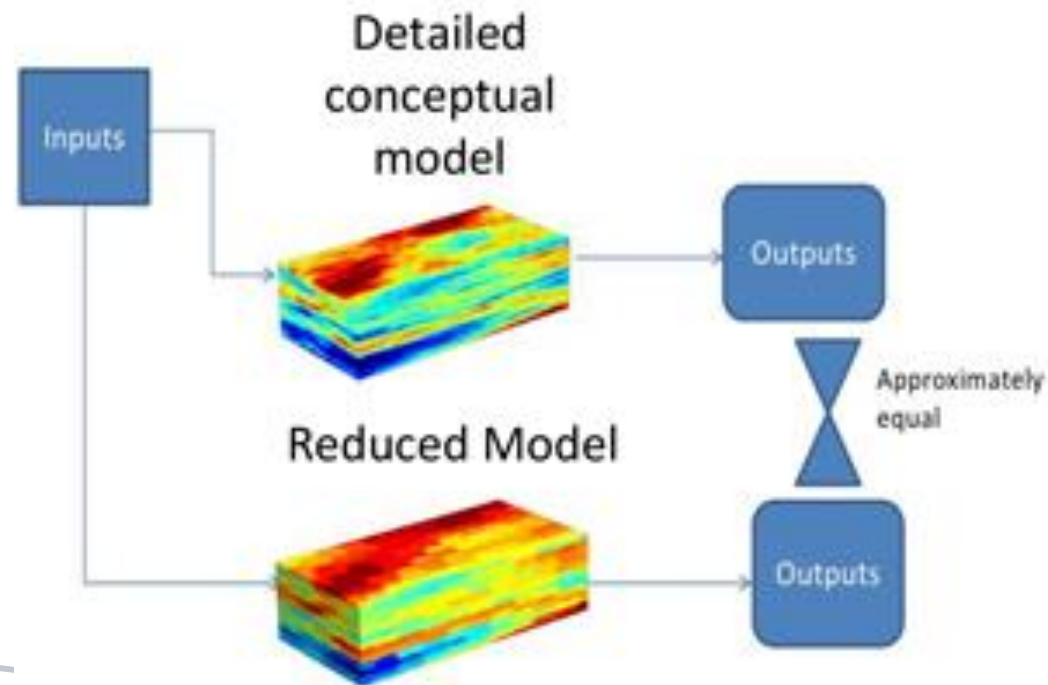


- Calculs infaisables pour du temps réel \rightarrow nécessité de réduction

Possibilités d'ajustement

- Simplifier/approximer le modèle sans trop le dégrader (pour ne pas avoir des modèles totalement différents)
- Autre solution: découper le modèle en sous-systèmes

Résumé du principe

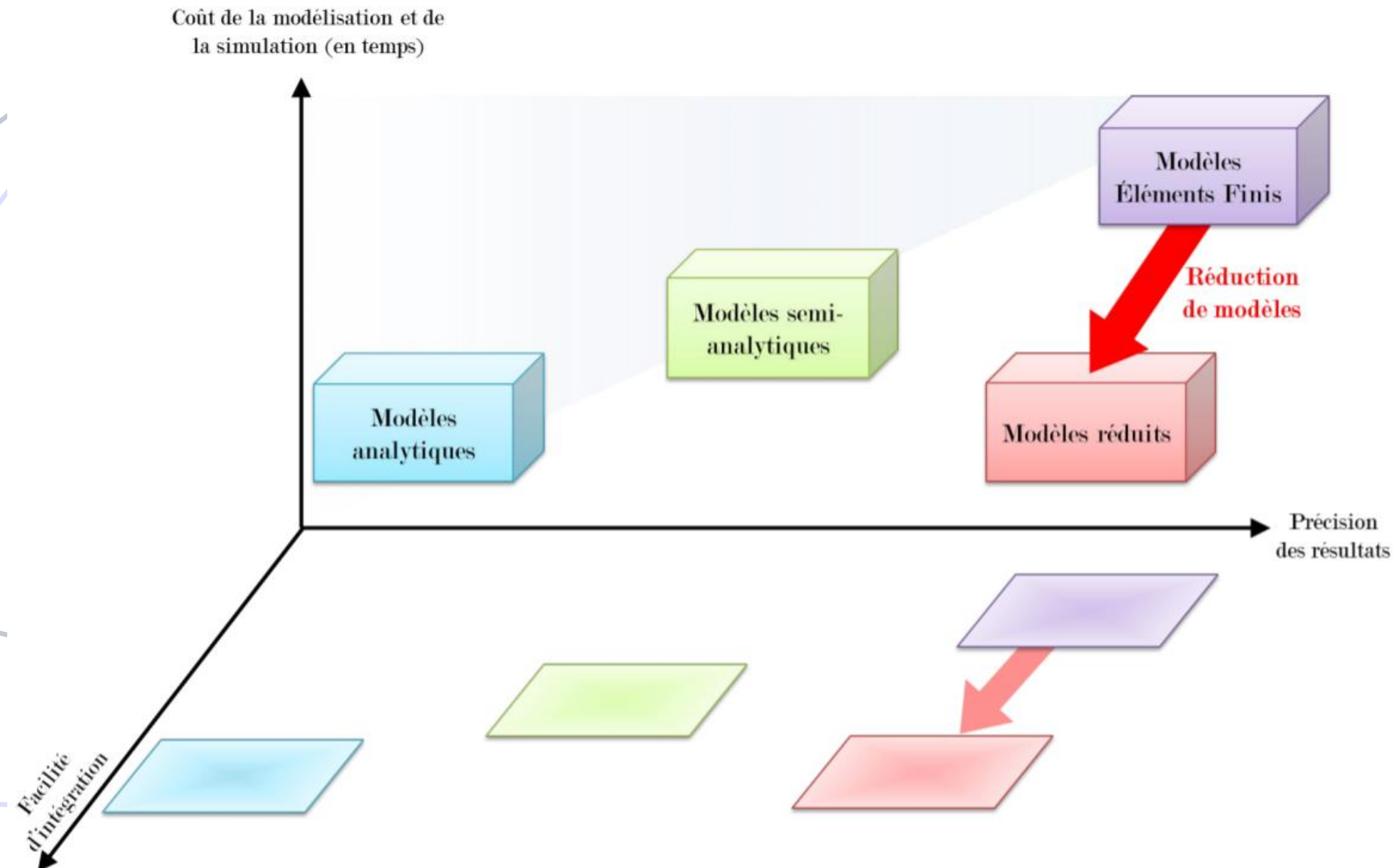


Décomposition en sous-systèmes

- Découpage du système en noeuds (isovaleurs ou isoquantités) en fonction de la géométrie du modèle et de l'environnement (mais requiert l'expertise métier, jusqu'à l'apprentissage via machine learning?)

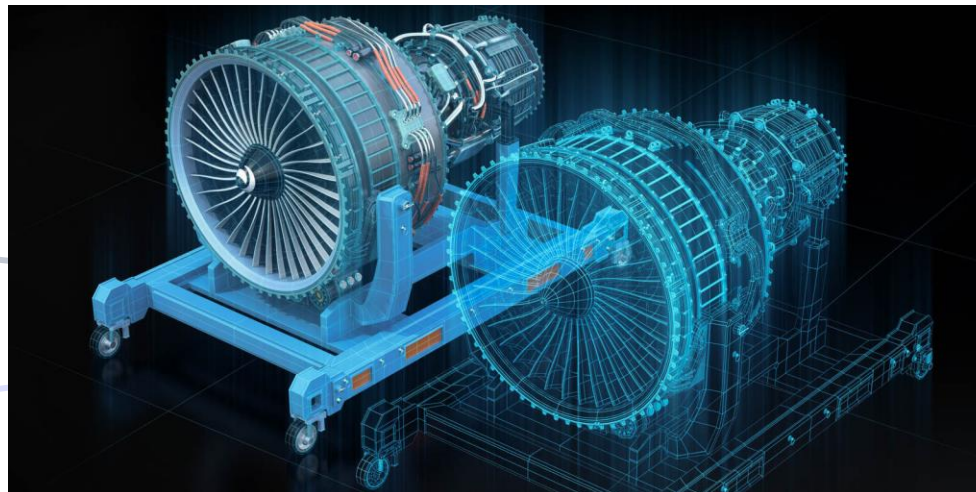
Décomposition en sous-systèmes

- Intégration des sous-modèles avec plusieurs modes dans un modèle complexe (très utilisé en aéronautique, aérospatial, aérodynamique, électronique intégrée, biologie, automobile...)
- Remarque: facilité d'intégration du modèle réduit dans des plateformes de simulation de systèmes (Simulink, Abaqus, Modelica, AMEsim etc) -> très utilisées pour la conception de systèmes complexes



Jumeaux numériques (digital twins)

- Représentation numérique de l'objet permettant d'effectuer des simulations afin d'anticiper le comportement de l'objet **en temps réel**.
- Exemples d'application: satellite, moteurs d'avion, effet d'un médicament sur un patient, éoliennes, navires, chauffage/ventilation, locomotives...



État de l'art de la réduction du modèle mathématique (non exhaustif)

- Méthode de la synthèse modale
- Méthodes de sous-structuration
- Méthodes optimales
- Réduction pour des problèmes non-linéaires

Méthode de la synthèse modale

- Extraction préliminaire des valeurs propres et des vecteurs propres de l'équation différentielle sans second membre : $x' = Ax$
- Valeurs propres et vecteurs propres : pulsations propres et modes propres en dynamique des structures et comme étant les constantes de temps et les modes propres de diffusion en thermique
- Différentes méthodes: Jacobi, puissance itérée, itération de LANCZOS, la méthode d'itération sur les sous-espaces

Exemple: puissance itérée

Soit la matrice $A \in M_{n,n}(\mathbb{R})$, avec A diagonalisable.

Supposons que A possède n valeurs propres telles que $|\lambda_1| < |\lambda_2| < \dots < |\lambda_n|$ et soient $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n$ les vecteurs propres associés.

Alors, si \mathbf{x}_0 est un vecteur quelconque:

$$\mathbf{x}_0 = a_1 \mathbf{u}_1 + a_2 \mathbf{u}_2 + \dots + a_n \mathbf{u}_n$$

Supposons que $a_n \neq 0$, on calcule la suite $\mathbf{x}_{k+1} = A\mathbf{x}_k$

$$\begin{aligned} \text{ie: } \mathbf{x}_k = A^k \mathbf{x}_0 &= \lambda_1^k a_1 \mathbf{u}_1 + \lambda_2^k a_2 \mathbf{u}_2 + \dots + \lambda_n^k a_n \mathbf{u}_n \\ &= \lambda_n^k \left(\left(\lambda_1 / \lambda_n \right)^k a_1 \mathbf{u}_1 + \dots + \left(\lambda_{n-1} / \lambda_n \right)^k a_{n-1} \mathbf{u}_{n-1} + a_n \mathbf{u}_n \right) \end{aligned}$$

Si $i \neq n$, $(\lambda_i / \lambda_n)^k \rightarrow 0$ donc le terme dominant devient $\lambda_n^k a_n \mathbf{u}_n$

Exemple: puissance itérée

Plus grande valeur propre: $|\lambda_n| \approx (\mathbf{x}_{k+1} / ||\mathbf{x}_k||)$

Vecteur propre associé: $\mathbf{u}_n \approx \mathbf{x}_k$

On normalise la suite:

$$\mathbf{b}_k = (\mathbf{x}_k / ||\mathbf{x}_k||) \text{ et } \mathbf{x}_{k+1} = A\mathbf{b}_k$$

On obtient: ${}^t\mathbf{b}_k \mathbf{x}_{k+1} = {}^t\mathbf{b}_k A\mathbf{b}_k \rightarrow \lambda_n {}^t\mathbf{b}_k \mathbf{b}_k = \lambda_n$

Exemple: puissance itérée

Algorithme:

Données : A, \vec{x}_0 et ε
début
 $\vec{x} \leftarrow \vec{x}_0$
 $\lambda_{anc} \leftarrow 1; \lambda \leftarrow 0$
 tant que $|\lambda - \lambda_{anc}| > \varepsilon$ **faire**
 $\lambda_{anc} \leftarrow \lambda$
 $\vec{b} \leftarrow \frac{\vec{x}}{\|\vec{x}\|}$
 $\vec{x} \leftarrow A \vec{b}$
 $\lambda \leftarrow {}^t \vec{b} \vec{x}$
 retourner λ, b
fin

Exemple: puissance inverse

Problème: recherche de la valeur propre de plus petit module
-> Revient à appliquer puissance itérée à A^{-1}

- 1) $PA = LU$ (décomposition LU de A)
- 2) On construit les suites \mathbf{b}_k et λ_k pour :
 - 1) Résoudre $LU\mathbf{x}_{k+1} = P\mathbf{b}_k$
 - 2) On pose $\mathbf{b}_k = \mathbf{x}_k / \|\mathbf{x}_k\|$ et $\lambda_k = 1/\mathbf{x}_{k+1}^t \mathbf{b}_k$
- 3) Jusqu'à convergence, ie: $|\lambda_{k+1} - \lambda_k| < \text{epsilon}$

Exemple: puissance inverse

Données : A, P, L, U tels que $PA = LU$, \vec{x}_0 et ε

Algorithme:

```

début
   $\vec{x} \leftarrow \vec{x}_0$ 
   $\lambda_{anc} \leftarrow 1$ 
   $\lambda \leftarrow 0$ 
  tant que  $|\lambda - \lambda_{anc}| > \varepsilon$  faire
     $\lambda_{anc} \leftarrow \lambda$ 
     $\vec{b} \leftarrow \frac{\vec{x}}{\|\vec{x}\|}$ 
    résoudre  $LU\vec{x} = P\vec{b}$ 
     $\lambda \leftarrow \frac{1}{\vec{x}^T \vec{b}}$ 
  retourner  $\lambda$  et  $\vec{b}$ 
fin
  
```

Méthodes de sous-structuration

- Diviser la structure principale en sous-structures puis approcher champ de solution de chaque sous-structure sur la base de ses modes propres
- Exemples: méthodes de Guyan, Craig Bampton, modes de composantes etc.

Exemple: méthode de Guyan

$$\mathbf{K} \cdot \mathbf{d} = \mathbf{f}$$

avec \mathbf{K} matrice de raideur, \mathbf{f} le vecteur force et \mathbf{d} le vecteur déplacement

En séparant les degrés de liberté master/slave, il vient:

$$\begin{bmatrix} \mathbf{K}_{mm} & \mathbf{K}_{ms} \\ \mathbf{K}_{sm} & \mathbf{K}_{ss} \end{bmatrix} \begin{Bmatrix} \mathbf{d}_m \\ \mathbf{d}_s \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_m \\ \mathbf{0} \end{Bmatrix}$$

Puis:
$$\begin{Bmatrix} \mathbf{d}_m \\ \mathbf{d}_s \end{Bmatrix} = \begin{bmatrix} \mathbf{I} \\ -\mathbf{K}_{ss}^{-1} \mathbf{K}_{sm} \end{bmatrix} \{\mathbf{d}_m\} = \{\mathbf{T}_G\} \{\mathbf{d}_m\}$$

Le système devient donc: $\mathbf{K}_G \mathbf{d}_m = \mathbf{f}_m$ avec $\mathbf{K}_G = \mathbf{T}_G^T \mathbf{K} \mathbf{T}_G$ et \mathbf{T}_G la matrice de transformation de la réduction de Guyan

Méthodes optimales

- Méthodes dites “a posteriori”
- Exemples: POD, SVD, POD snapshots, PGD

Exemple: Proper Generalized Decomposition

- 1) Formulation variationnelle du problème
- 2) Discrétisation du domaine (éléments finis)
- 3) Solution \mathbf{u} approximée par une représentation séparée

$$\mathbf{u} \approx \mathbf{u}^N(x_1, x_2, \dots, x_d) = \sum_{i=1}^N \mathbf{X}_{1i}(x_1) \cdot \mathbf{X}_{2i}(x_2) \cdots \mathbf{X}_{di}(x_d)$$

Avec les produits de fonctions $\mathbf{X}_1(x_1), \mathbf{X}_2(x_2), \dots, \mathbf{X}_d(x_d)$, chacun dépendant d'une ou plusieurs variables

- 4) Algorithme de résolution (point fixe)

Exemple: Proper Generalized Decomposition

La solution est recherchée en appliquant un algorithme glouton (en général point fixe) à la formulation faible du problème.

Pour chaque itération i de l'algorithme, un mode de la solution est calculé. Chaque mode est constitué d'un ensemble de valeurs numériques des produits de fonctions $\mathbf{X}_1(x_1), \dots, \mathbf{X}_d(x_d)$, qui enrichissent l'approximation de la solution.

Réduction pour des problèmes non linéaires

- La méthode de réduction a priori (APR) est une méthode incrémentale basée sur un enrichissement de la base de projection à l'aide des sous-espaces de Krylov. Le principe de base de cette méthode peut se résumer en quatre étapes :
 - 1. Initialisation de la base
 - 2. Résolution du système réduit
 - 3. Utilisation de la solution du système réduit et du modèle pour déterminer un résidu
 - 4. Enrichissement de la base par le résidu
- Les étapes 2, 3 et 4 sont répétées jusqu'à convergence du modèle réduit vers une précision fixée par l'utilisateur.

Exemple d'application: thermique spatiale

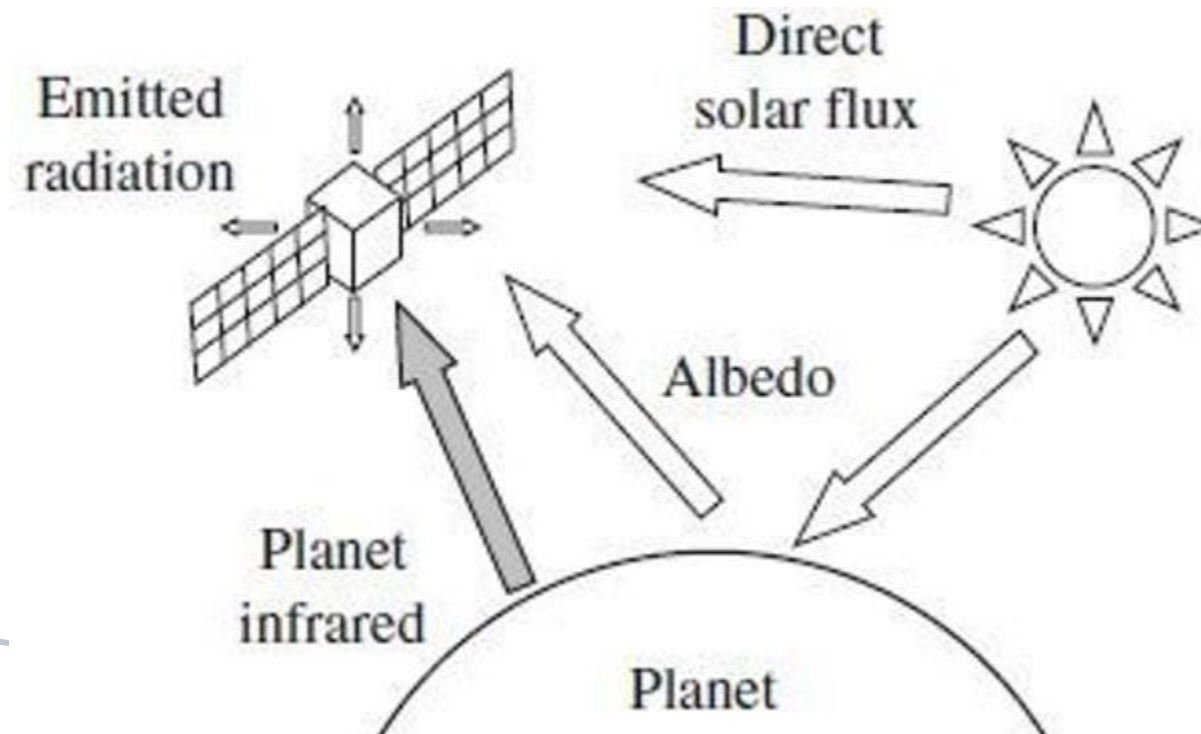
- Rappels de thermique
- Exemple de réduction du modèle mathématique
- Exemple de réduction du modèle géométrique

Différents transferts de chaleur

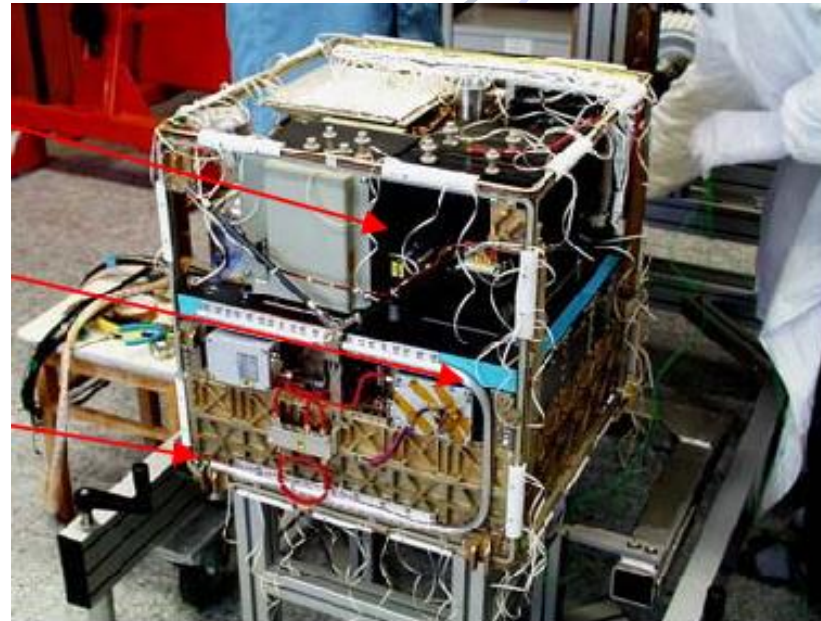
- Q : transfert de chaleur(W)
- A : aire de la surface(m^2)
- L : longueur considérée (m)
- h : coefficient de transfert de chaleur conductif ($W.K^{-1}$)
- ϵ : émissivité de la surface (pas d'unité)
- σ : constante de Stefan-Boltzmann ($W.m^{-2}.K^{-4}$)
- m : masse du composant(kg)
- C_p : capacité thermique massique ($J.kg^{-1}.K^{-1}$)
- T_1 & T_2 : température des corps (K)
- Remarque: pas de convection dans l'espace

Heat Transfer Mechanism	Governing Equation
Conduction	$Q = kA \frac{T_2 - T_1}{L}$
Convection	$Q = hA(T_2 - T_1)$
Radiation	$Q = \epsilon \sigma A(T_2^4 - T_1^4)$
Heat Absorbed	$Q = mc_p(T_2 - T_1)$

Différentes sources de chaleur

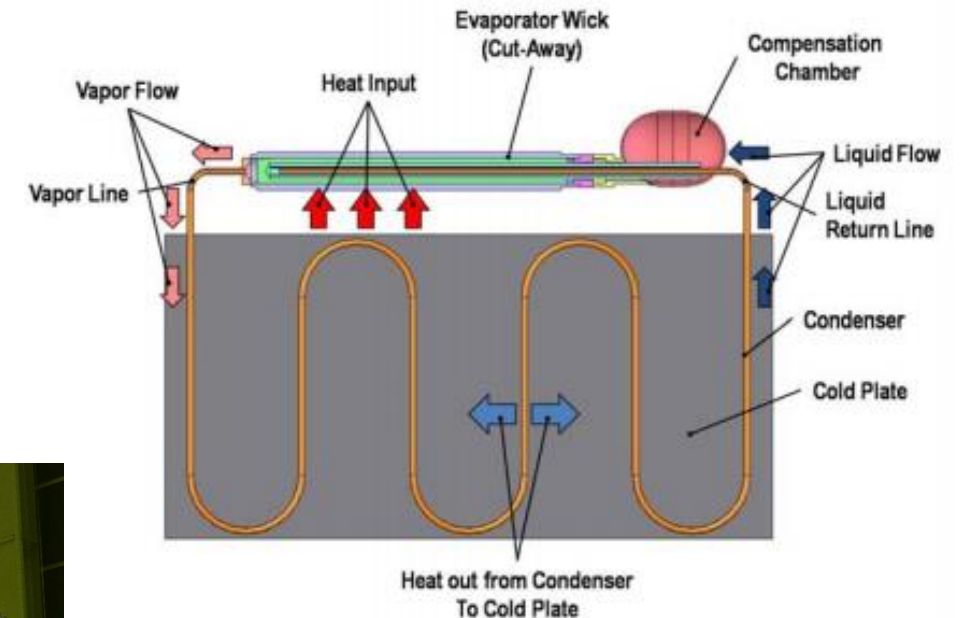
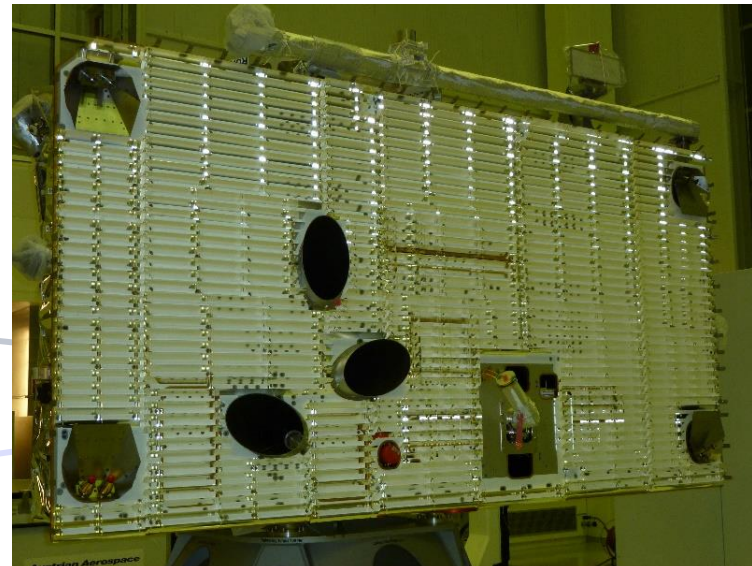


Différents outils de régulation de la température



Caloducs

Radiateurs



Boucles fluides

Exemple d'application: méthode des conductances équivalentes

- Méthode de réduction de matrice conductrice similaire utilisée à Thales Alenia Space & Airbus
 - Utilisée pour la réduction de modèles mathématiques des satellites
 - Efficacité des outils prouvée (en usage depuis de nombreuses années)
- La méthodologie peut être utilisée par le fournisseur du sous-système pour fournir un modèle réduit plus précis

Reference: TMRT A thermal model reduction tool, 23rd European Workshop on Thermal and ECLS Software

Théorie de la réduction

- Équation conductive thermique du système:

- 3 types de noeuds:

- K = gardés
- S = supprimés
- G = groupés (définissant des noeuds moyens A)

$$[C_{DD}]\{T_D\} + \{Q_D\} + \{P_D\} - [M_{DD}]\left\{\frac{\partial T_D}{\partial t}\right\} = \{0\}$$

$$\begin{bmatrix} C_{KK} & C_{KS} & C_{KG} \\ C_{SK} & C_{SS} & C_{SG} \\ C_{GK} & C_{GS} & C_{GG} \end{bmatrix} \begin{Bmatrix} T_K \\ T_S \\ T_G \end{Bmatrix} + \begin{Bmatrix} Q_K \\ Q_S \\ Q_G \end{Bmatrix} + \begin{Bmatrix} P_K \\ 0 \\ P_G \end{Bmatrix} - \begin{bmatrix} M_{KK} & 0 & 0 \\ 0 & M_{SS} & 0 \\ 0 & 0 & M_{GG} \end{bmatrix} \begin{Bmatrix} \frac{\partial T_K}{\partial t} \\ \frac{\partial T_S}{\partial t} \\ \frac{\partial T_G}{\partial t} \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}$$

Reference: TMRT A thermal model reduction tool, 23rd European Workshop on Thermal and ECLS Software

Théorie de la réduction

- Définition d'un noeud moyen: $T_A = \sum_G a_{AG} T_G$
- a_{AG} = ratio aire/capacitance du noeud G par rapport au noeud A
- Hypothèse physique: flux, échanges radiatifs ou convectifs proportionnels à l'aire du noeud

$$P_G = a_{GA} P_A$$

Théorie de la réduction

- Approximation:
$$\sum_j \left(M_{i,j} \frac{\partial T_j}{\partial t} \right) \approx \sum_j (M_{i,j}) \frac{\partial T_i}{\partial t}$$

Remarque: les facteurs de la matrice M sont plus importants pour les noeuds j qui sont conductivement proches du noeud i

Théorie de la réduction

- Équations équivalentes:
 - Équation de la chaleur du système réduit:

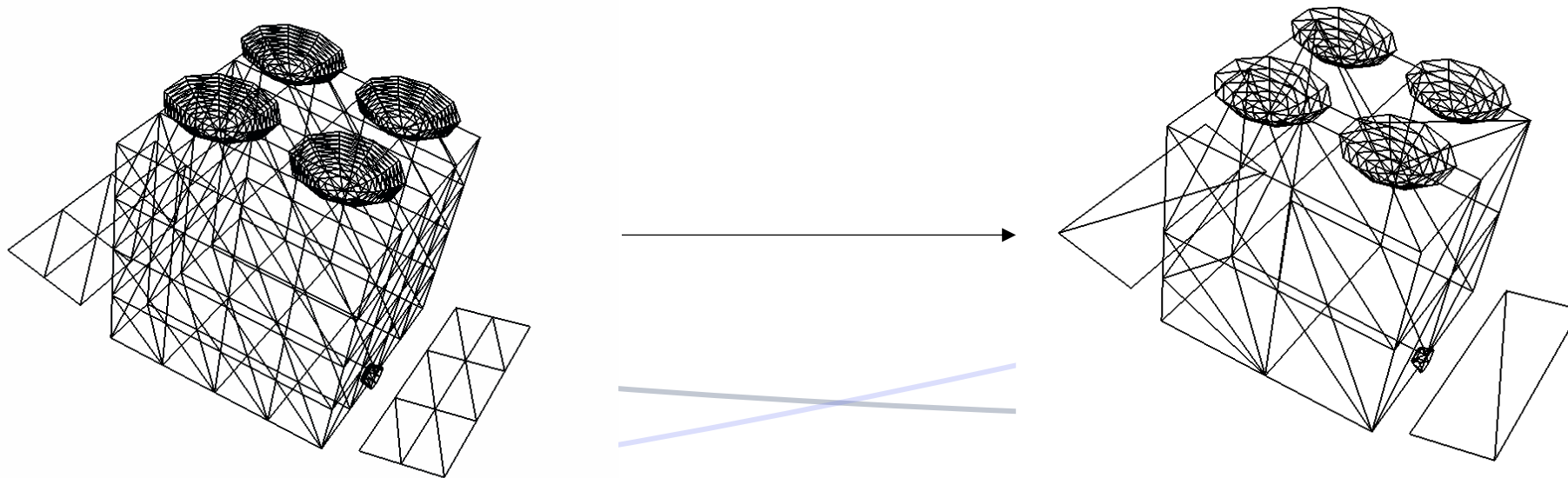
$$[C'_{RR}] \begin{Bmatrix} T_K \\ T_A \end{Bmatrix} + [P_W D_{RD}] \begin{Bmatrix} Q_K \\ Q_S \\ Q_R \end{Bmatrix} + \begin{Bmatrix} P_K \\ P_A \end{Bmatrix} - [M'_{RR}] \begin{Bmatrix} \frac{\partial T_K}{\partial t} \\ \frac{\partial T_A}{\partial t} \end{Bmatrix} = \{0\}$$

Rappel, système d'origine:

$$\begin{bmatrix} C_{KK} & C_{KS} & C_{KG} \\ C_{SK} & C_{SS} & C_{SG} \\ C_{GK} & C_{GS} & C_{GG} \end{bmatrix} \begin{Bmatrix} T_K \\ T_S \\ T_G \end{Bmatrix} + \begin{Bmatrix} Q_K \\ Q_S \\ Q_G \end{Bmatrix} + \begin{Bmatrix} P_K \\ 0 \\ P_G \end{Bmatrix} - \begin{bmatrix} M_{KK} & 0 & 0 \\ 0 & M_{SS} & 0 \\ 0 & 0 & M_{GG} \end{bmatrix} \begin{Bmatrix} \frac{\partial T_K}{\partial t} \\ \frac{\partial T_S}{\partial t} \\ \frac{\partial T_G}{\partial t} \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}$$

Réduction du modèle géométrique

- Problème: aucune connexion avec la physique (aucune garantie sur le respect des propriétés)



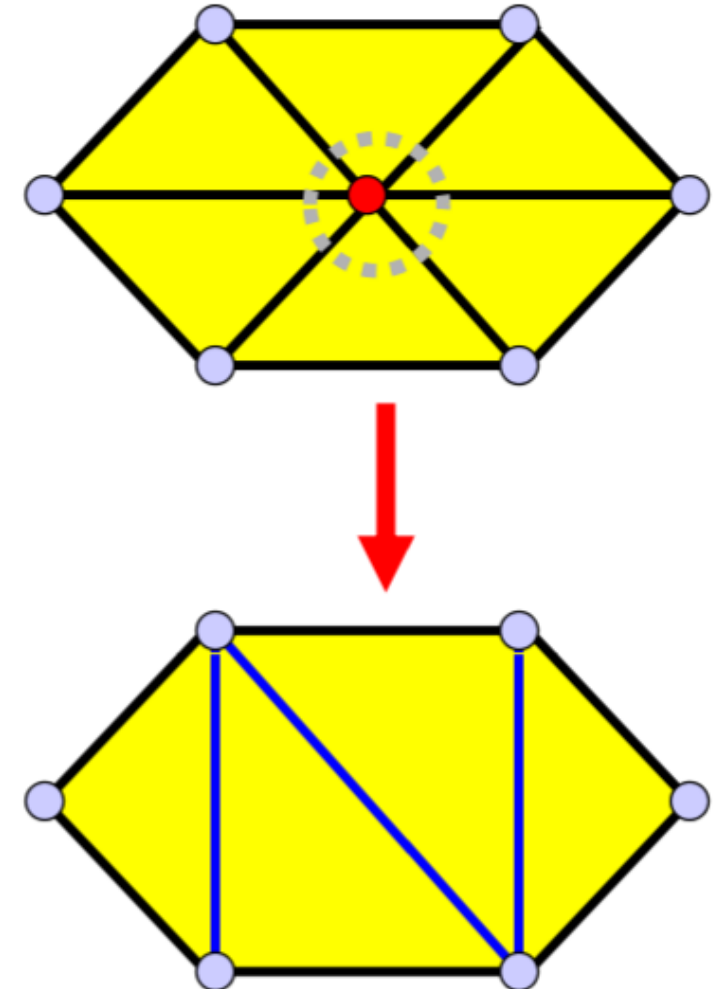
État de l'art (non exhaustif)

- Suppression des sommets, edge collapse / halfedge collapse
- Fonctions de coût et de placement
 - Lindstrom-Turk: preserve la forme, le volume et les frontières
 - Garland-Heckbert: encode la distance approximée au maillage d'origine en utilisant des matrices quadriques attribuées à chacun des sommets
 - Préservation du volume "memoryaware"
 - Préservation des normales



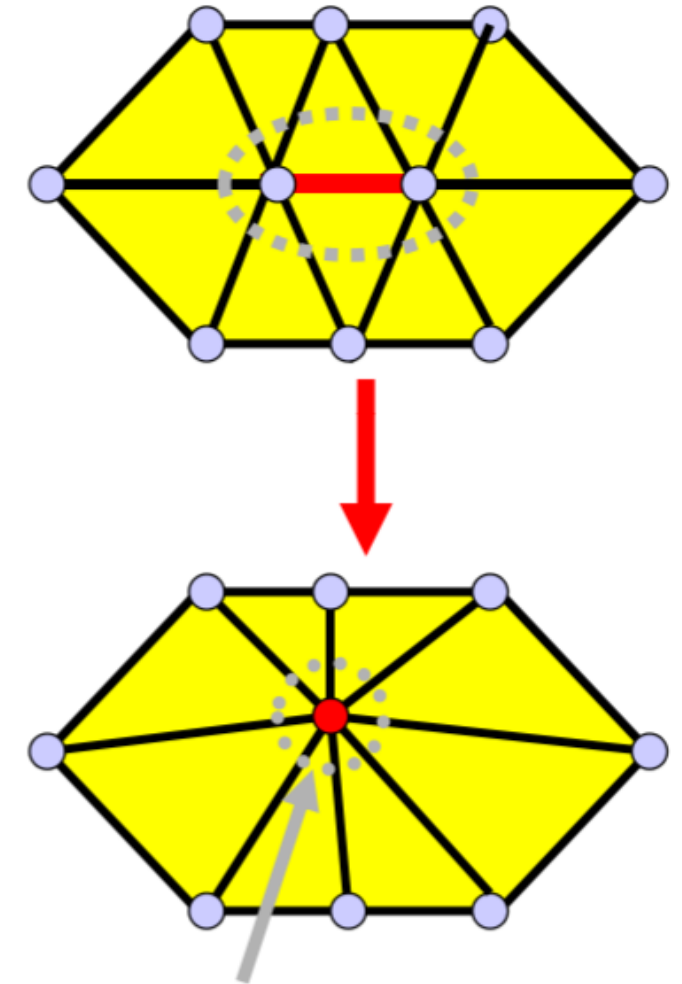
Suppression de sommets

- Supprime un sommet et ses faces et côtés adjacents, créant un trou de valence k .
- Le trou est triangulé en ajoutant $k-2$ triangles.
- Le nombre de sommets et de triangles est réduit de 1 et 2, respectivement.



Edge collapse

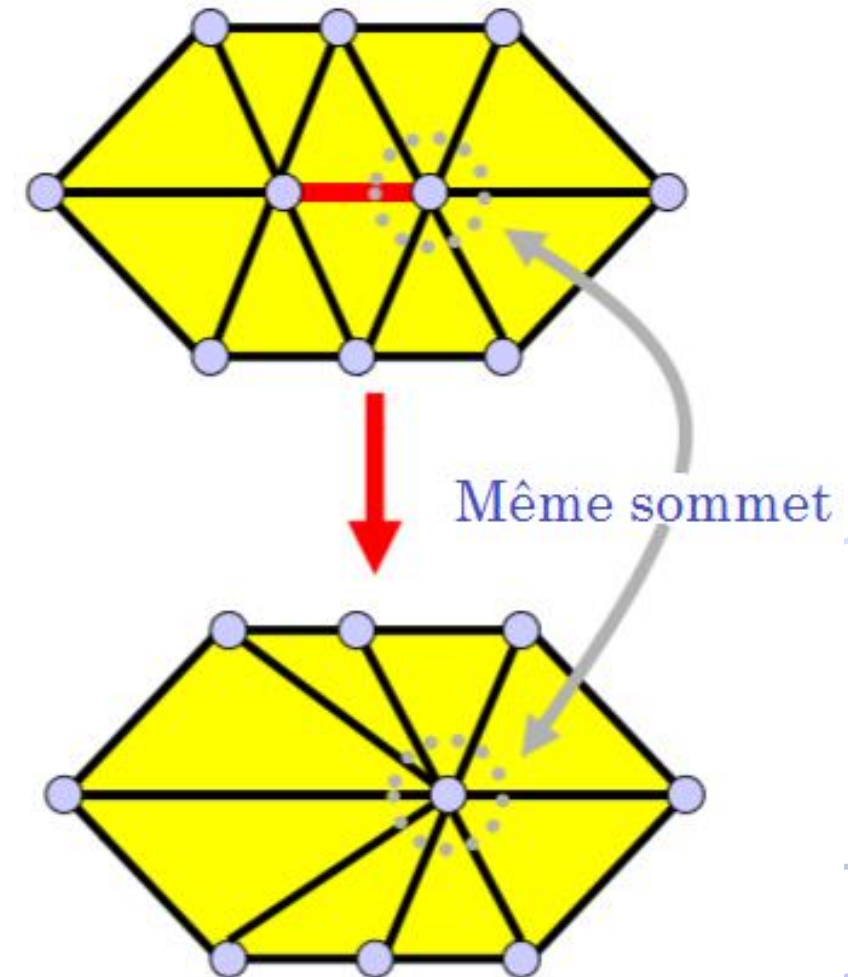
- Sélectionne un côté et le “collapse” en un nouveau point. Les deux triangles adjacents sont transformés en deux côtés.
- Le nombre de sommets et de triangles est réduit de 1 et de 2, respectivement.
- Remarque: liberté de placement du nouveau sommet!



Nouveau point

Halfedge collapse

- Soit un côté sélectionné avec pour sommets p et q , l'opérateur du halfedge collapse déplace p sur q ou q sur p (cas particulier du edge collapse)
- Remarques: - déplacer q sur p ou p sur q constituent deux opérations différentes
 - ici aucun degré de liberté dans le placement du nouveau point



Minimisation de la fonction de coût

- Respect du volume de la géométrie
- Respect de l'orientation des faces
- Respect de la forme des frontières
- Etc
- => via descente de gradient appliquée à la fonction de coût considérée

ongler dans
ccès rapide

Copier
Coller

Couper
Copier le chemin d'accès
Coller le raccourci

Presse-papiers

Déplacer vers
Copier vers

Organiser

Supprimer
Renommer

Nouvel élément
Accès rapide

Nouveau dossier

Propriétés
Ouvrir

Ouvrir
Modifier
Historique

Sélectionner tout
Aucun
Inverser la sélection

Sélectionner

← → ↕ ↑ > Ce PC > Bureau > DORGeomProcessor

Rechercher dans : DORGeomP...

★ Accès rapide

Desktop

Documents

Téléchargements

Images

Vincent

data

dialogs

OFF

Tests Vincent

Ce PC

Bureau

Documents

Images

Musique

Téléchargements

Vidéos

Windows (C:)

DATA (D:)

Réseau

Nom	Modifié le	Type	Taille
.qglviewer.xml	25/03/2021 17:09	Document XML	2 Ko
error.log	25/03/2021 17:08	Document texte	0 Ko
orig_offfile.off	25/03/2021 17:08	Fichier OFF	49 Ko
proto.exe	25/03/2021 17:08	Application	3 009 Ko
proto.pdb	25/03/2021 17:08	Base de données ...	33 612 Ko
Reduced_model_output.obj	25/03/2021 17:09	Fichier objet	2 Ko
temp_before_reduction.obj	25/03/2021 17:09	Fichier objet	5 Ko
Decimation_aif_mesh.obj	25/03/2021 13:34	Fichier objet	1 Ko
reader.py	18/03/2021 15:27	Python File	5 Ko
convert.py	18/03/2021 15:26	Python File	23 Ko
geom_converter.py	18/03/2021 15:23	Python File	3 Ko
env.bat	18/03/2021 10:05	Fichier de comma...	1 Ko
DORGeomProcessor.bat	08/03/2021 10:12	Fichier de comma...	1 Ko
convert.pyc	08/03/2021 09:59	Compiled Python ...	23 Ko
reader.pyc	06/03/2021 16:37	Compiled Python ...	6 Ko
proto_shell.py	21/01/2021 19:17	Python File	6 Ko
geom3D.py	21/01/2021 19:13	Python File	20 Ko
thermal_computer.py	08/01/2021 16:29	Python File	4 Ko
Sources	25/03/2021 17:07	Dossier de fichiers	
dialogs	25/03/2021 16:36	Dossier de fichiers	
tmp	25/03/2021 13:51	Dossier de fichiers	
Tests Vincent	25/03/2021 13:32	Dossier de fichiers	
__pycache__	18/03/2021 15:27	Dossier de fichiers	
Debug	18/03/2021 15:20	Dossier de fichiers	
images	08/03/2021 09:58	Dossier de fichiers	
models	08/03/2021 09:58	Dossier de fichiers	
DLL	26/02/2021 14:58	Dossier de fichiers	

Progress: 0%

Estimated remaining time: --:--:--

3D View Skin Reduction Simulation

Geometrical Model

Name: C:/Projets/geomprocessor/models/OFF files/sculpt

Triangles: 7342 Quadrangles: 0

Points: 3669 Edges: 11013

Clusters: 1 Length: 0.999992

Width: 1 Height: 0.999888

Center: (0, 0, 0)

Rebuild

Tolerance: 0,00014 Planarity: 0,00100

Rectangularity(%): 10,00 Rebuild

☐ Convert to quad. ☐ All triangles

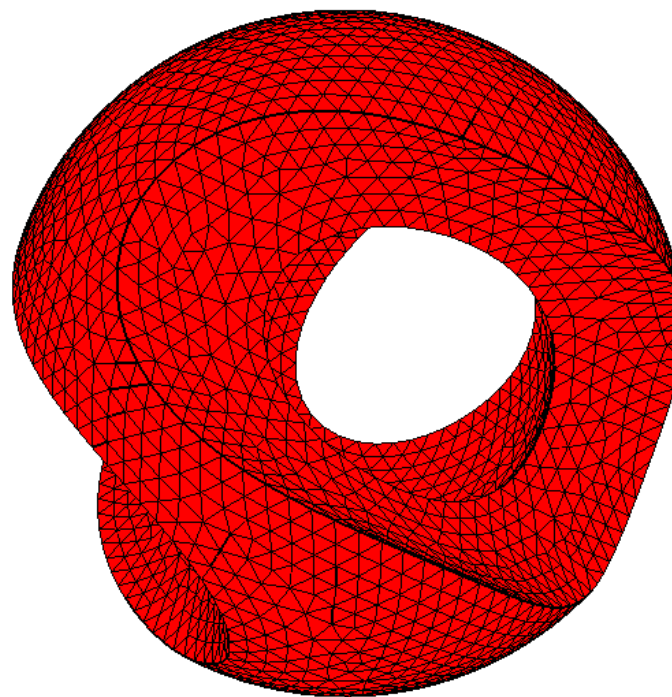
3D View

View mode:    

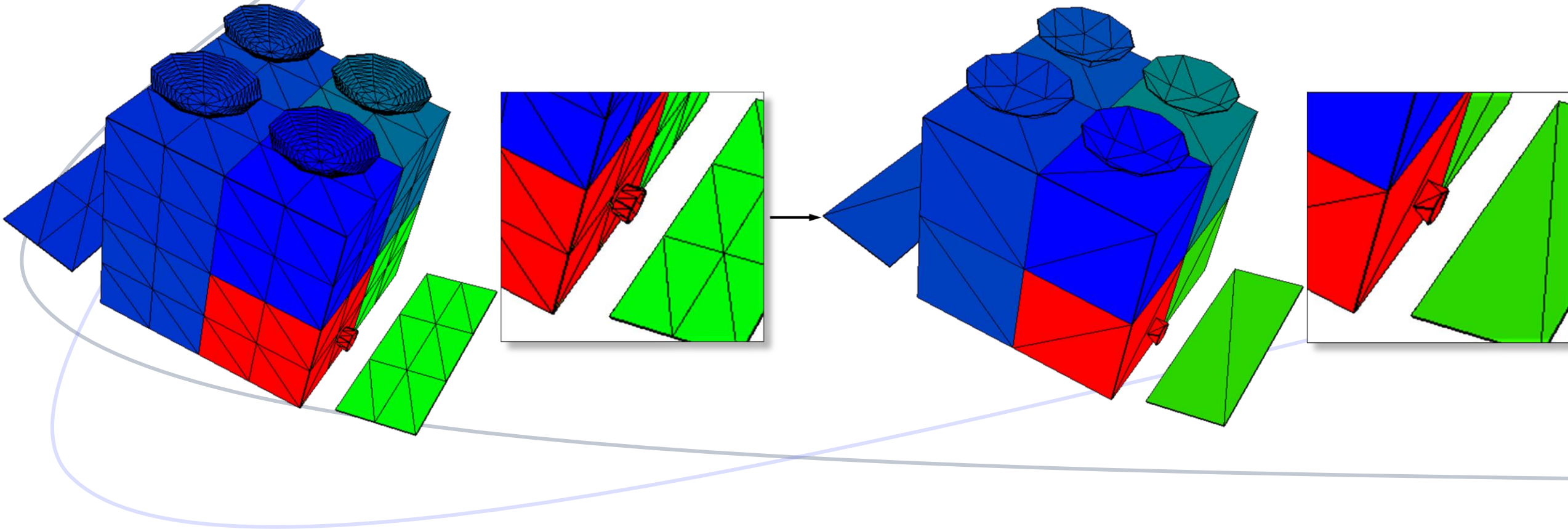
Transparency: 

OpenGL

Vendor: Intel
 Renderer: Intel(R) UHD Graphics 750
 Version: 4.6.0 - Build 30.0.101.1340
 Lang. Version: 4.60 - Build 30.0.101.1340



Réduction géométrique guidée par la physique

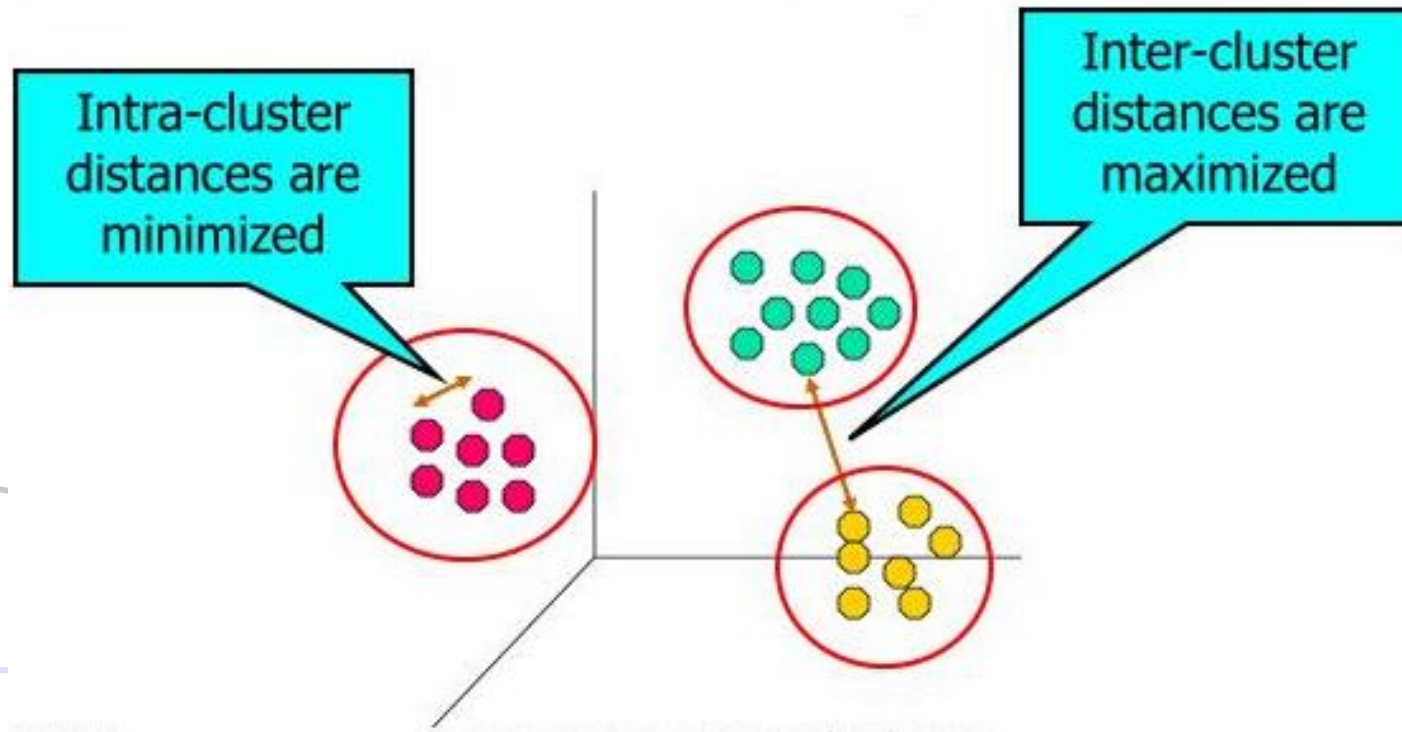


Références

- [1] Frédéric da Silva. Méthodologies de réduction de modèles multiphysiques pour la conception et la commande d'une chaîne de traction électrique. Autre. Université Paris-Saclay, 2015. Français. ffNNT : 2015SACLC022ff. fftel-01275878
- [2] Peter Lindstrom and Greg Turk. Fast and memory efficient polygonalsimplification. In IEEE Visualization, pages 279–286, 1998.
- [3] M. Garland and P. S. Heckbert. Surface simplification using quadric errormetrics. In Proc. SIGGRAPH '97, pages 209–216, 1997.
- [4] David Cohen-Steiner, Pierre Alliez, Mathieu Desbrun. Variational Shape Approximation. [Research Report] RR-5371, INRIA. 2004, pp.29. inria-0007063
- [5] TMRT A thermal model reduction tool, 23rd European Workshop on Thermal and ECLS Software
- [6] <https://pages.mtu.edu/~shene/COURSES/cs3621/SLIDES/Simplification.pdf>

Clustering

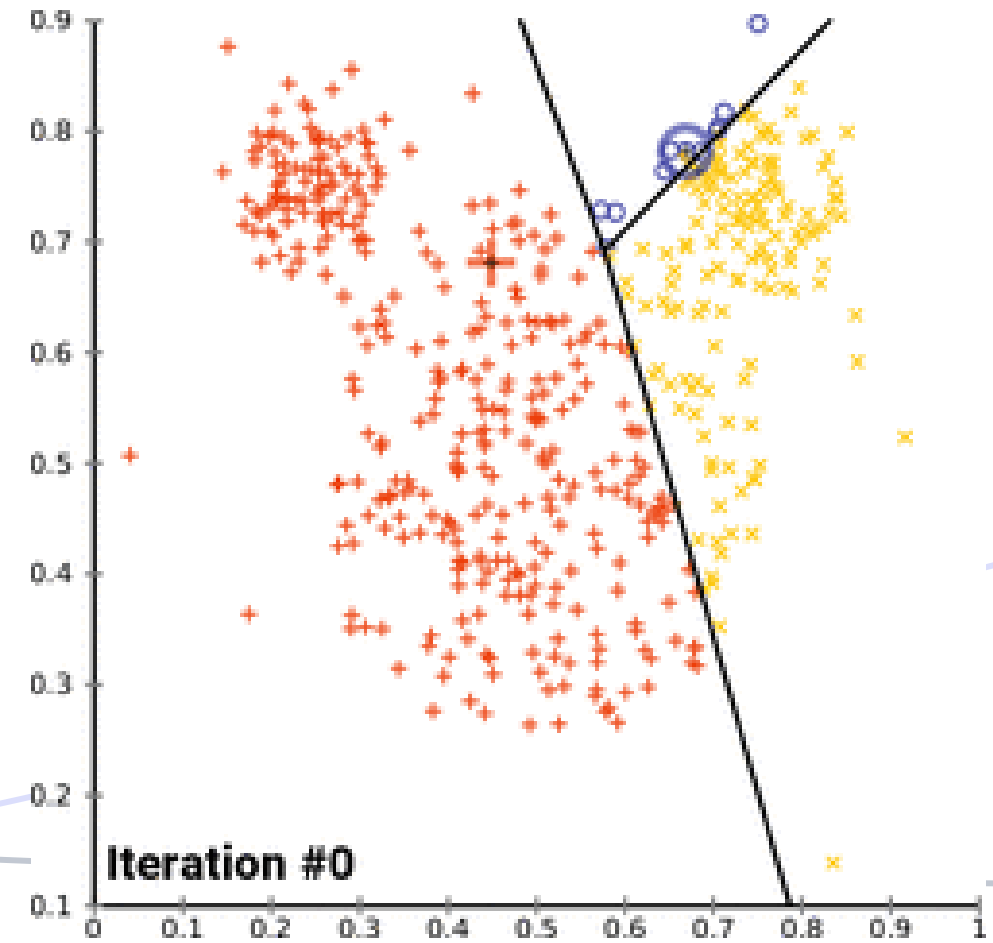
- Partitionnement de données en sous-ensembles partageant des caractéristiques communes
- Bon clustering si : clusters homogènes et bien différenciés



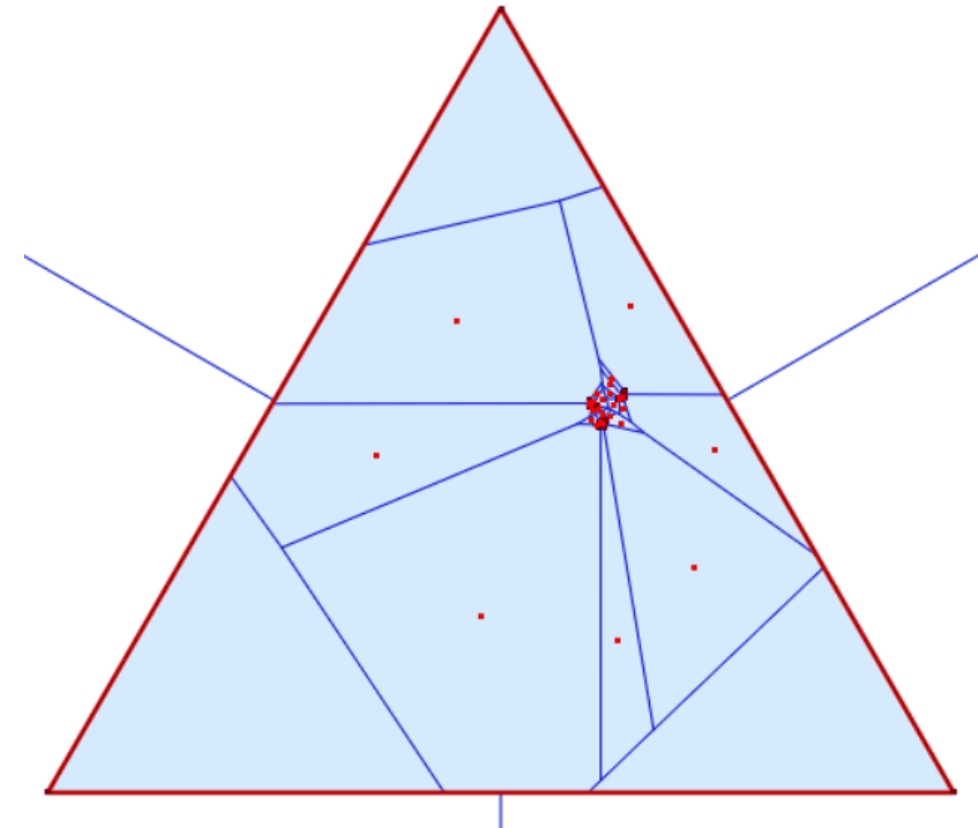
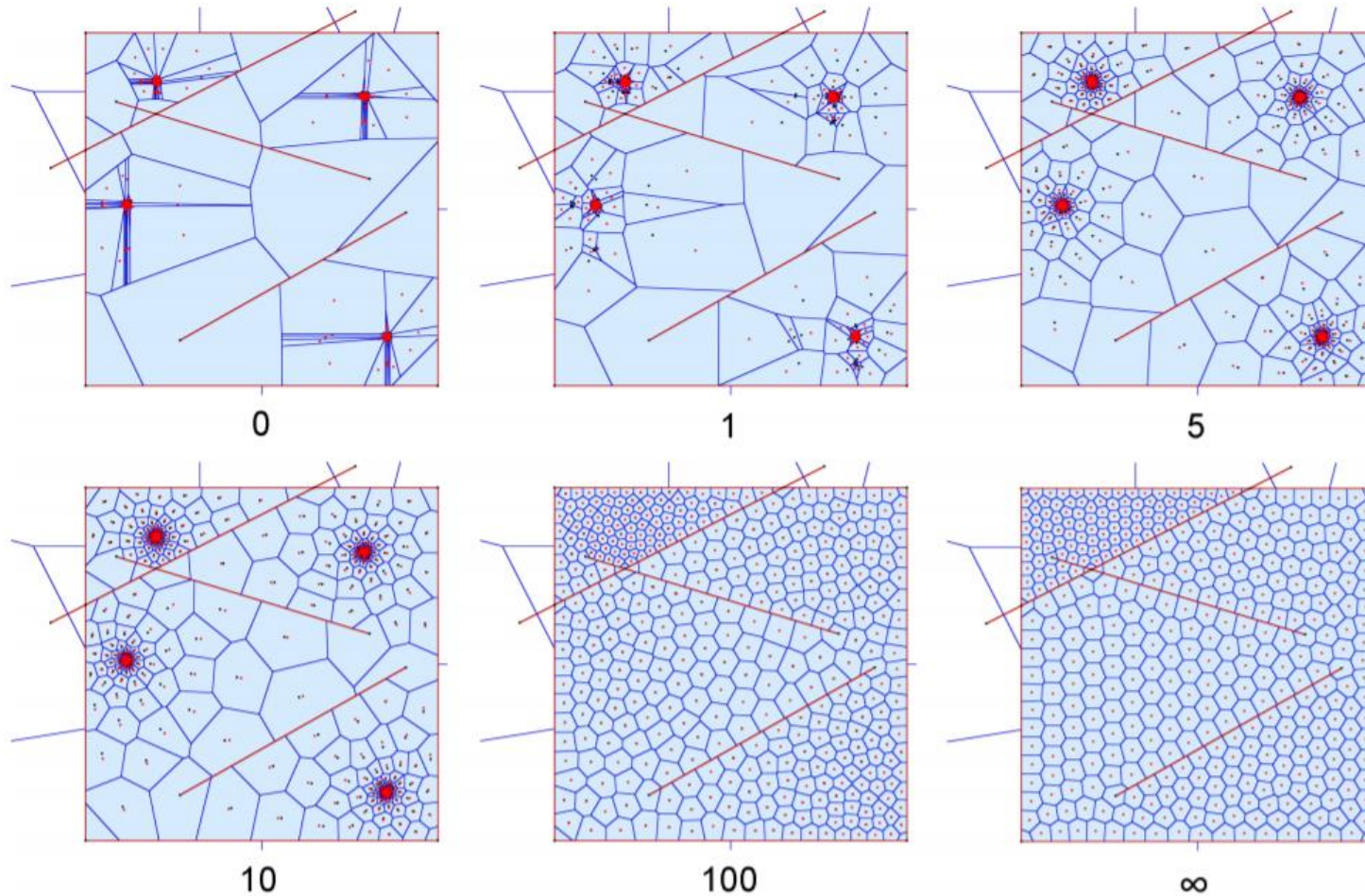
- Par des méthodes de partitionnement:
 - Différentes partitions évaluées via différents critères (k-means, CLARANS etc)
- Par des méthodes hiérarchiques
 - Décomposition hiérarchique des données via différents critères (BIRCH, Diana etc)
- Par des méthodes basées sur une densité
 - Via connectivité et fonctions de densité (OPTICS, DBSCAN etc)

K-means

- Etape 1: k (entrée) centroides initiaux auxquels sont assignés les points les plus proches
- Etape 2: calcul des nouveaux centroides via moyenne des points du cluster
- Etape 3: répéter étape 2 jusqu'à convergence



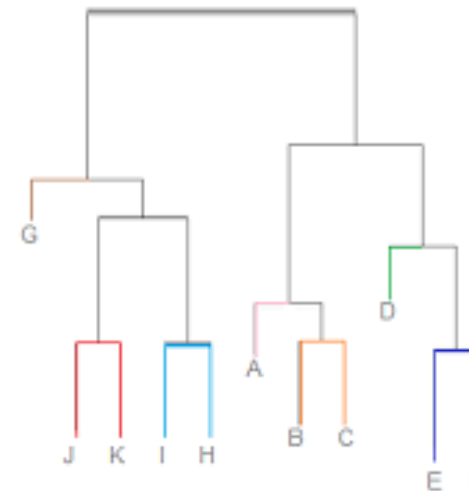
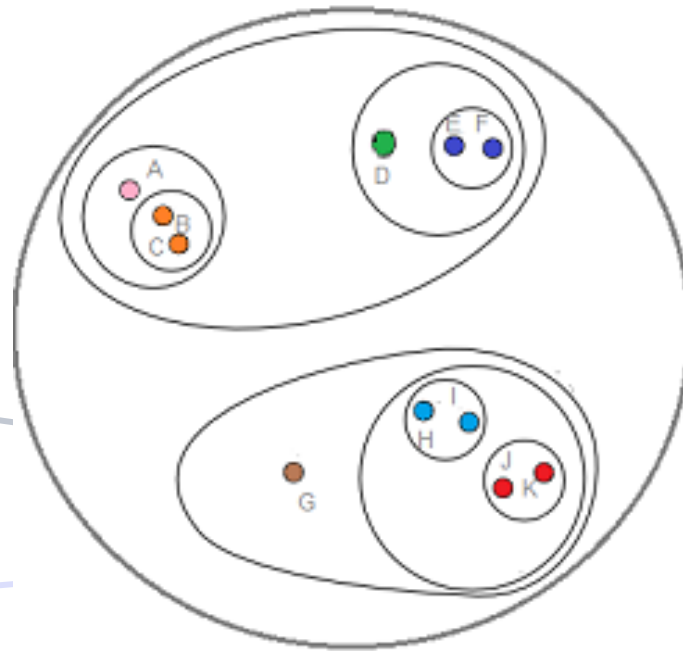
Algorithme de Lloyd pour les diagrammes de Voronoï



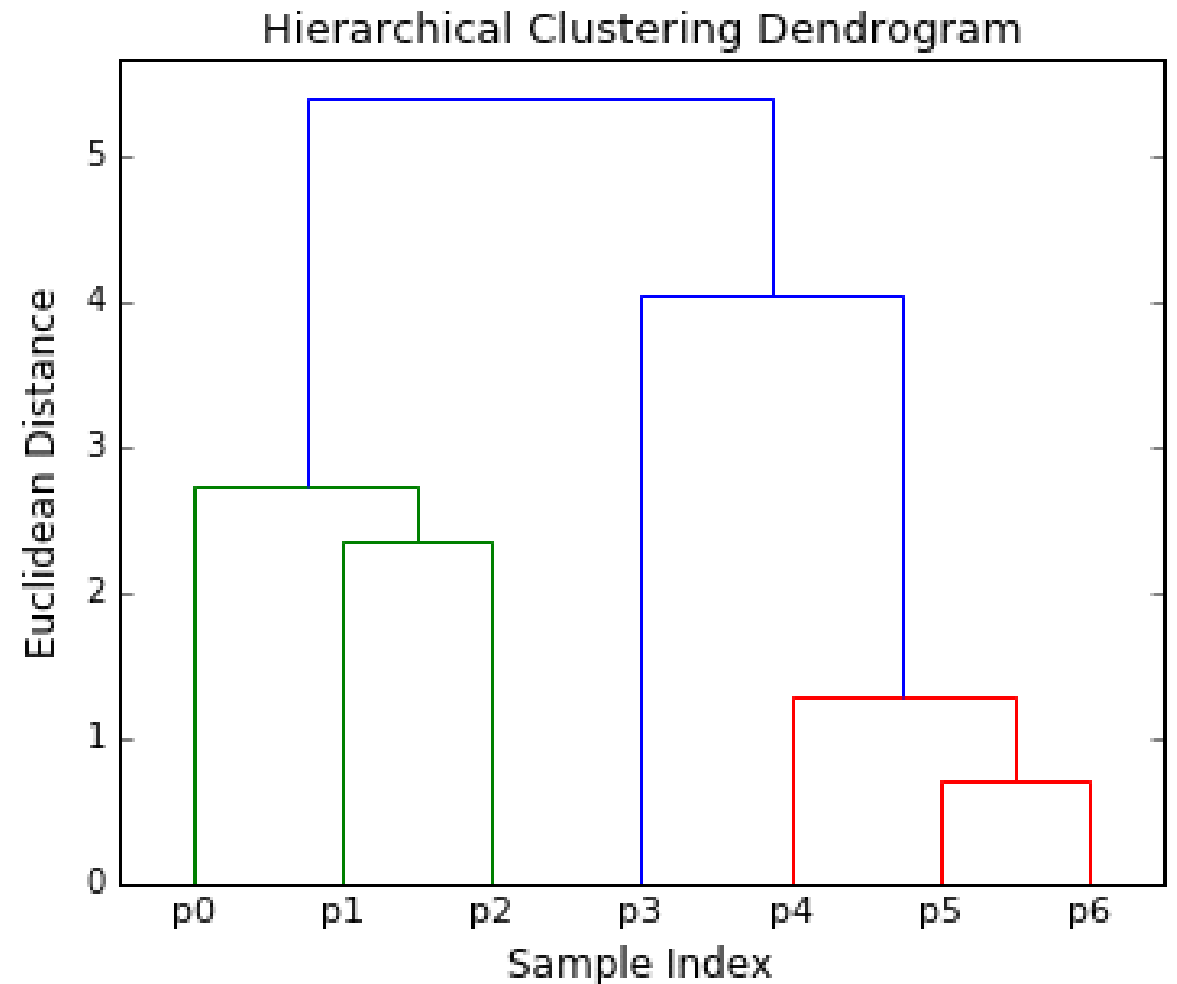
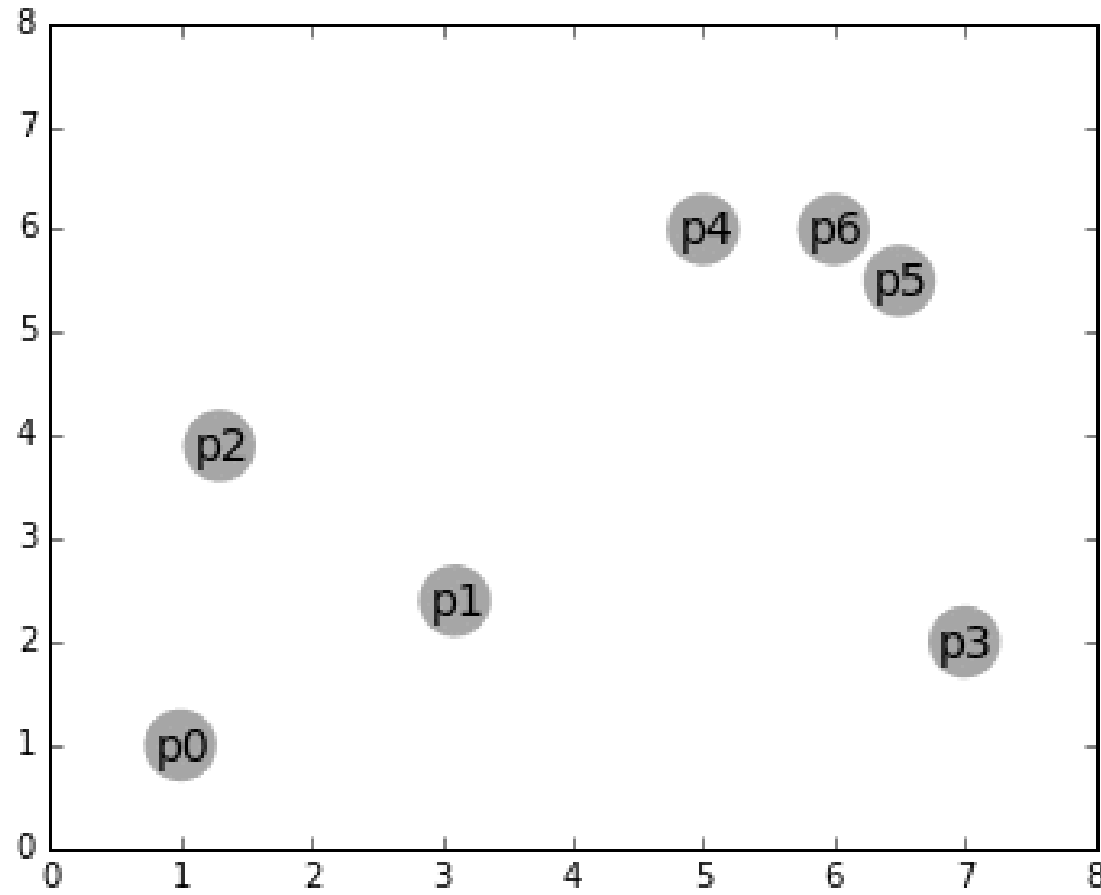
Tournois, Alliez, Devillers: 2D Centroidal Voronoi Tessellations with Constraints.

Clustering hiérarchique

- Résulte en un ensemble de clusters imbriqués selon un arbre hiérarchique
- Peut être perçu comme un dendrogramme avec clustering obtenu en “coupant” selon le niveau désiré

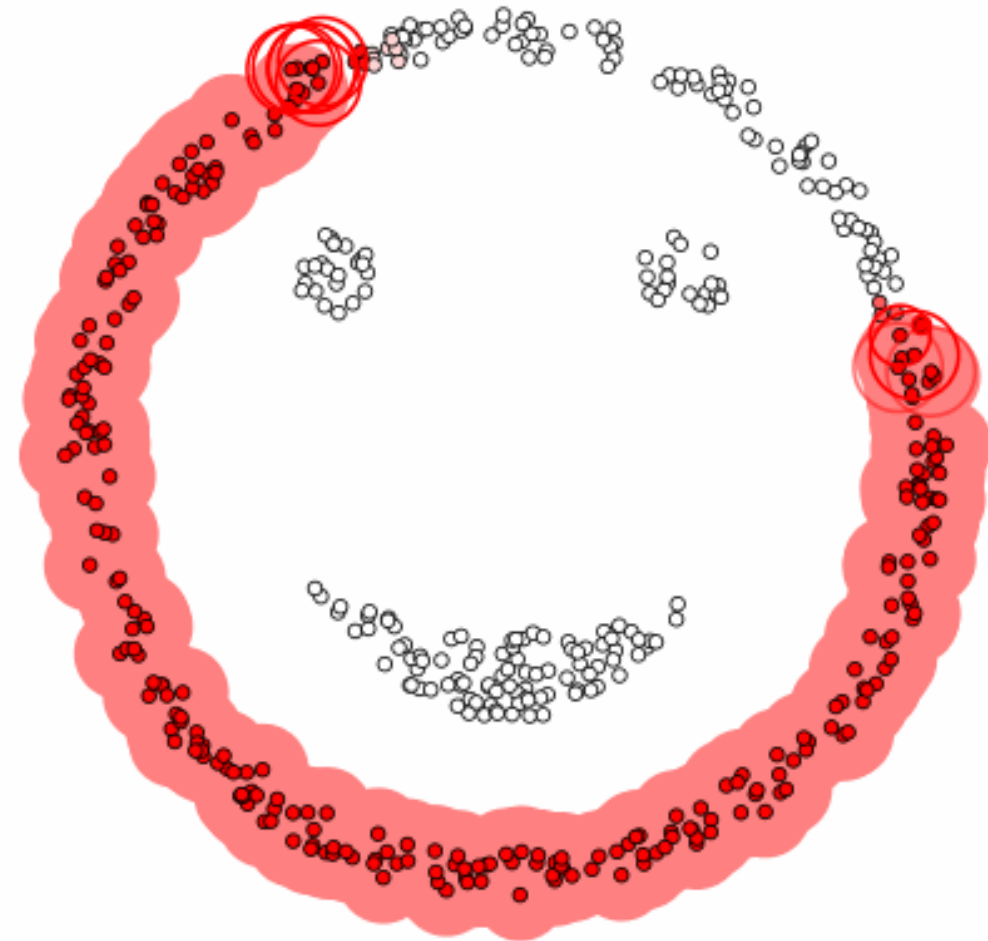


Clustering hiérarchique

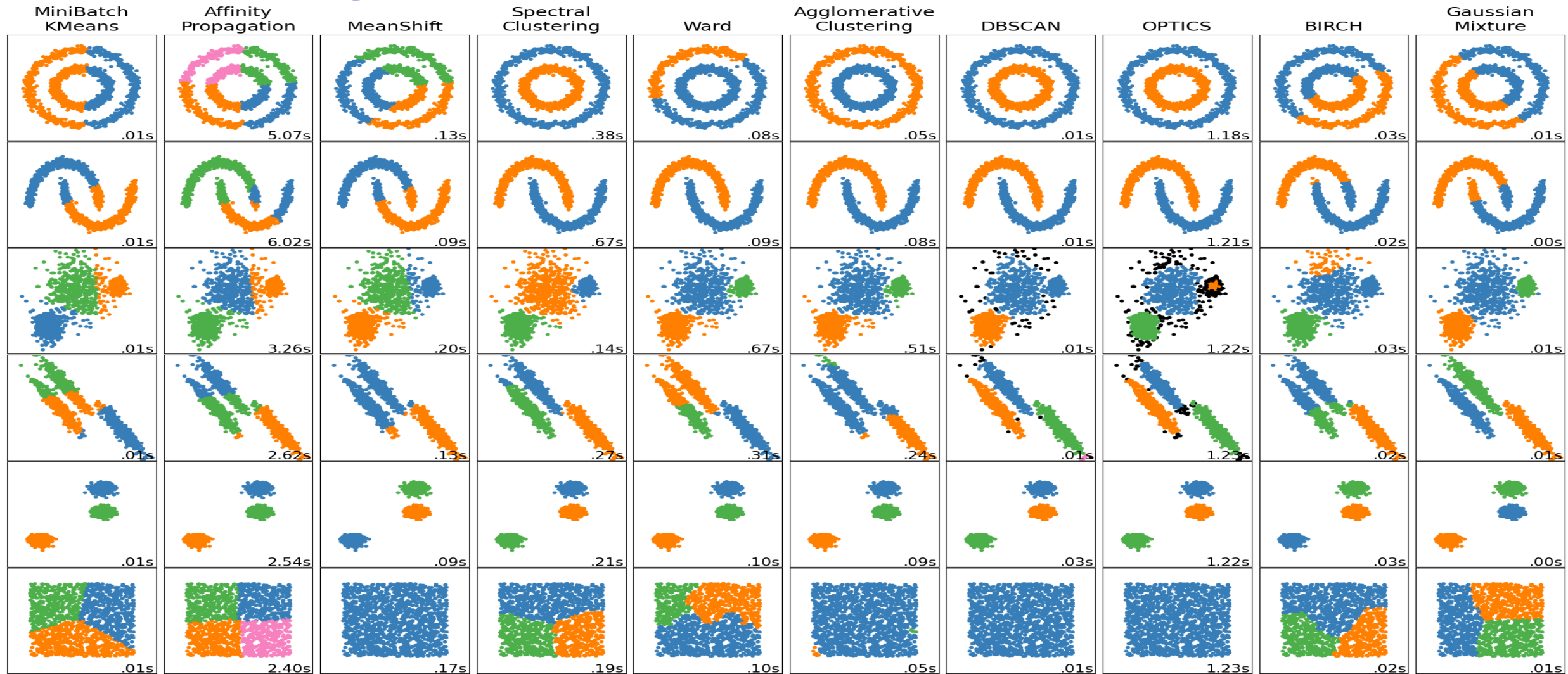


Clustering selon densité: DBSCAN

- Sélectionne arbitrairement un point p
- Collecte tous les points “densité-atteignable” depuis p selon un rayon Epsilon et un nombre minimum de points N_{min}
- Si p a une haute densité (bcp de points dans son rayon = **core point**): création d'un cluster
- Si p a une faible densité mais est dans le voisinage d'un core point: process réitéré pour un autre point p
- Sinon p point isolé
- Fin de l'algorithme quand tous les points ont été visité



Clustering via libraire Scikit-learn



A comparison of the clustering algorithms in scikit-learn

Method name	Parameters	Scalability	Usecase	Geometry (metric used)
K-Means	number of clusters	Very large <code>n_samples</code> , medium <code>n_clusters</code> with MiniBatch code	General-purpose, even cluster size, flat geometry, not too many clusters, inductive	Distances between points
Affinity propagation	damping, sample preference	Not scalable with <code>n_samples</code>	Many clusters, uneven cluster size, non-flat geometry, inductive	Graph distance (e.g. nearest-neighbor graph)
Mean-shift	bandwidth	Not scalable with <code>n_samples</code>	Many clusters, uneven cluster size, non-flat geometry, inductive	Distances between points
Spectral clustering	number of clusters	Medium <code>n_samples</code> , small <code>n_clusters</code>	Few clusters, even cluster size, non-flat geometry, transductive	Graph distance (e.g. nearest-neighbor graph)
Ward hierarchical clustering	number of clusters or distance threshold	Large <code>n_samples</code> and <code>n_clusters</code>	Many clusters, possibly connectivity constraints, transductive	Distances between points
Agglomerative clustering	number of clusters or distance threshold, linkage type, distance	Large <code>n_samples</code> and <code>n_clusters</code>	Many clusters, possibly connectivity constraints, non Euclidean distances, transductive	Any pairwise distance
DBSCAN	neighborhood size	Very large <code>n_samples</code> , medium <code>n_clusters</code>	Non-flat geometry, uneven cluster sizes, transductive	Distances between nearest points
OPTICS	minimum cluster membership	Very large <code>n_samples</code> , large <code>n_clusters</code>	Non-flat geometry, uneven cluster sizes, variable cluster density, transductive	Distances between points
Gaussian mixtures	many	Not scalable	Flat geometry, good for density estimation, inductive	Mahalanobis distances to centers
BIRCH	branching factor, threshold, optional global clusterer.	Large <code>n_clusters</code> and <code>n_samples</code>	Large dataset, outlier removal, data reduction, inductive	Euclidean distance between points

Non-flat geometry clustering is useful when the clusters have a specific shape, i.e. a non-flat manifold, and the standard euclidean distance is not the right metric. This case arises in the two top rows of the figure above.