

IMN430

Chapitre 2

Techniques de base en visualisation

Partie 3: Données vectorielles

Olivier Godin & Michaël Bernier

Université de Sherbrooke

23 janvier 2017

Plan de la présentation

- 1 Introduction
- 2 Visualisation de données vectorielles
- 3 Références

Introduction

1 Introduction

2 Visualisation de données vectorielles

3 Références

Introduction

Dans ce chapitre-ci, on s'attarde aux méthodes de base permettant **d'exprimer les données sous forme visuelle.**

Tel que mentionné dans le chapitre d'introduction, les **algorithmes de mappage** qui mettent en correspondance des données avec des primitives graphiques sont au cœur de la visualisation.

Ces algorithmes seront catégorisés **selon le type de donnée qu'ils prennent en entrée.**

Introduction

Ici, une attention toute particulière sera portée aux algorithmes de mappage de **données scalaires**.

Par la suite, une introduction aux représentations de **données vectorielles** sera aussi présentée, mais ce sujet sera approfondi au chapitre 5.

Visualisation de données vectorielles

1 Introduction

2 Visualisation de données vectorielles

- Divergence et vorticité
- Flèches et glyphs
- Code de couleurs vectoriel

3 Références

Divergence et vorticité

1 Introduction

2 Visualisation de données vectorielles

- Divergence et vorticité
- Flèches et glyphes
- Code de couleurs vectoriel

3 Références

Généralités

En visualisation, on rencontre souvent des données vectorielles. Un vecteur \mathbf{v} est simplement un **tuple composé de n scalaires** :

$$\mathbf{v} = (v_1, \dots, v_n), \quad v_i \in \mathbb{R}.$$

Un vecteur peut représenter, par exemple, une **position**, une **direction** ou une **force** dans \mathbb{R}^n .

Généralités

Dans les algorithmes de visualisation, on considère souvent que **les vecteurs possèdent trois composantes**.

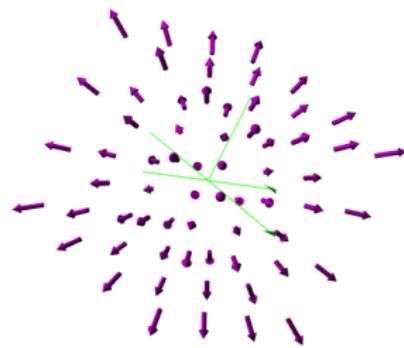
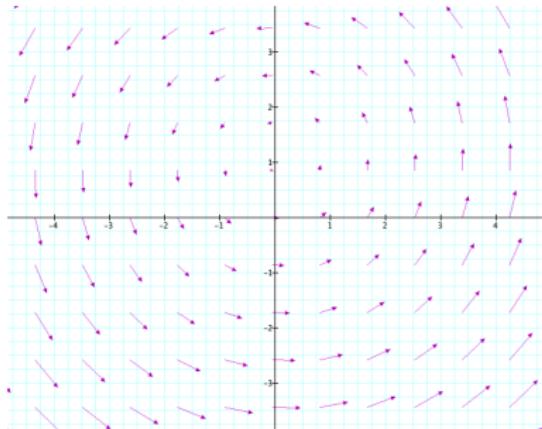
- Si les données comportent **plus de trois dimensions**, on applique une technique de réduction de la dimension pour les ramener à trois.
- Si les données comportent **moins de trois dimensions**, il suffit de mettre les composantes superflues égales à zéro.

Généralités

Un champ vectoriel est une fonction $f : \mathcal{D} \rightarrow \mathbb{R}^3$ où \mathcal{D} est un sous-ensemble de \mathbb{R}^2 ou \mathbb{R}^3 . Un ensemble de données vectorielles est obtenu par **échantillonnage d'un champ vectoriel** sur un domaine discret.

Dans cette section, on propose une introduction à la visualisation de données vectorielles. Des techniques plus avancées seront abordées au chapitre 5.

Généralités



Divergence

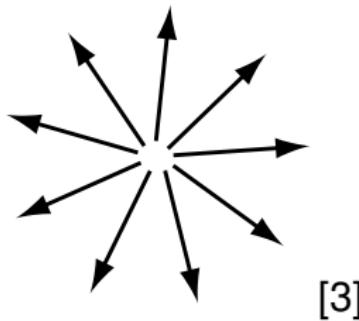
Soit un champ vectoriel $\mathbf{v} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$. La **divergence** associée à $\mathbf{v} = (v_x, v_y, v_z)$ correspond à la donnée scalaire

$$\operatorname{div} \mathbf{v} = \frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z}.$$

La divergence représente **la densité du flux sortant d'un point** dans un champ vectoriel.

Divergence

Une **divergence positive** en un point p indique que le flux quitte le point p .

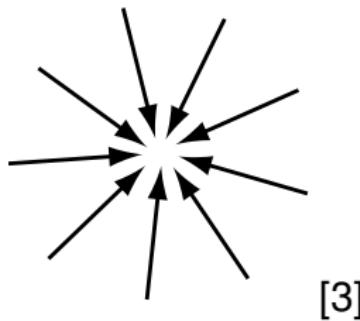


[3]

Les points présentant une divergence positive sont appelés **sources**.

Divergence

Une **divergence négative** en un point p indique que le flux est absorbé par le point p .

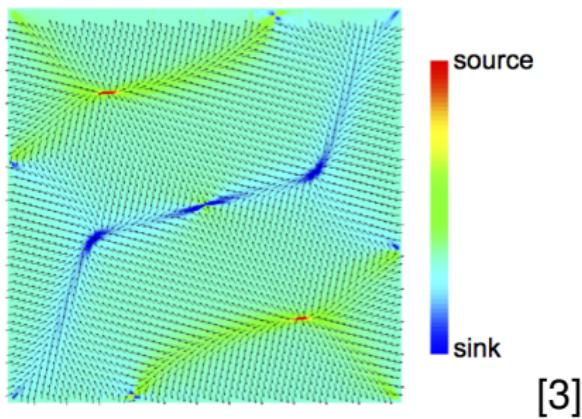


[3]

Les points présentant une divergence négative sont appelés **puits**.

Divergence

Dans la figure suivante, on superpose un **champ vectoriel** (flèches) et sa **divergence** (code de couleurs).

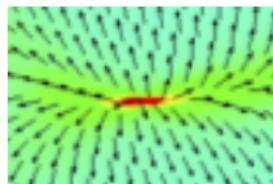


[3]

Que remarque-t-on ?

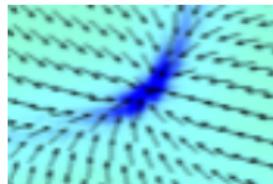
Divergence

Les **sources** (divergence positive) sont identifiées par la couleur rouge. On en discerne deux et on remarque que **le flux émerge de ces régions**.



[3]

Les **puits** (divergence négative) sont associés à la couleur bleue. On remarque facilement que **ces secteurs absorbent le flux**.



[3]

Vorticité

Soit un champ vectoriel $\mathbf{v} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$. La **vorticité** associée à $\mathbf{v} = (v_x, v_y, v_z)$ correspond à la donnée vectorielle

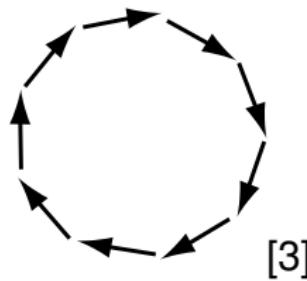
$$\text{rot } \mathbf{v} = \left(\frac{\partial v_z}{\partial y} - \frac{\partial v_y}{\partial z}, \frac{\partial v_x}{\partial z} - \frac{\partial v_z}{\partial x}, \frac{\partial v_y}{\partial x} - \frac{\partial v_x}{\partial y} \right).$$

La vorticité $\text{rot } \mathbf{v}$ est un champ vectoriel qui est localement perpendiculaire au plan de rotation de \mathbf{v} et dont l'amplitude représente la **vitesse angulaire de rotation** de \mathbf{v} autour de $\text{rot } \mathbf{v}$.

Vorticité

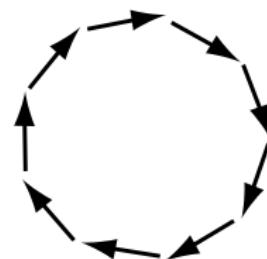
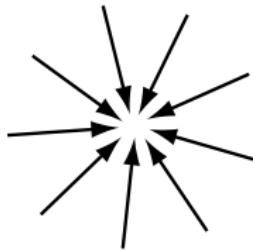
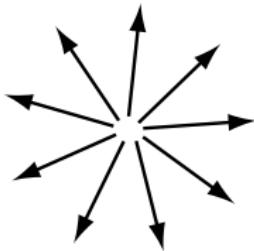
Ainsi, le vecteur de vorticité donne l'information sur **la vitesse et la direction de rotation** d'un champ vectoriel, à chaque point de celui-ci.

Les secteurs présentant une forte vorticité sont appelés **vortex**. Un vortex est une région où le champ vectoriel tourne localement autour d'un point.



Vorticité

Les notions de divergence et de vorticité sont **complémentaires**.

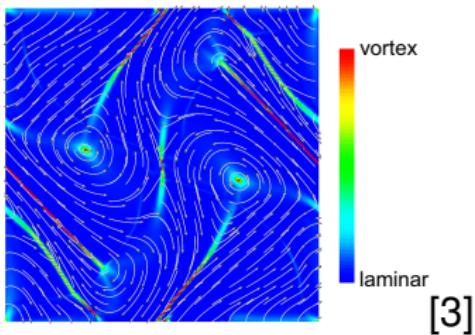


**Divergence élevée
Vorticité nulle**

**Divergence
nulle
Vorticité élevée**

Vorticité

La figure suivante superpose un **champ vectoriel** (flèches) et l'amplitude de la **vorticité** (code de couleurs).



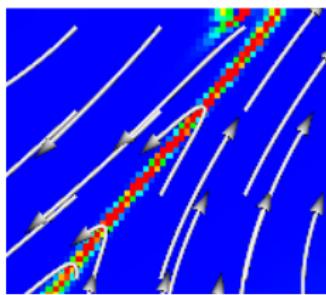
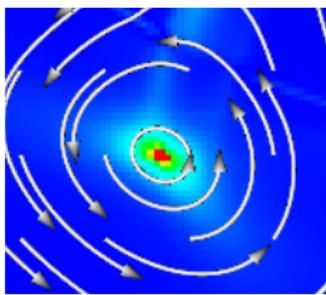
Les régions en **rouge** sont associées à **de fortes valeurs de vorticité**, tandis que celles en **bleu** présentent **peu ou pas de rotation**.

Vorticité

On aperçoit deux types de régions ayant une forte vorticité :

- les **petits points rouges**, qui représentent **les vortex** (on le constate en voyant le comportement du champ vectoriel).
- les **longues bandes rouges**, qui indiquent **les endroits où le champ vectoriel change brusquement de direction**.

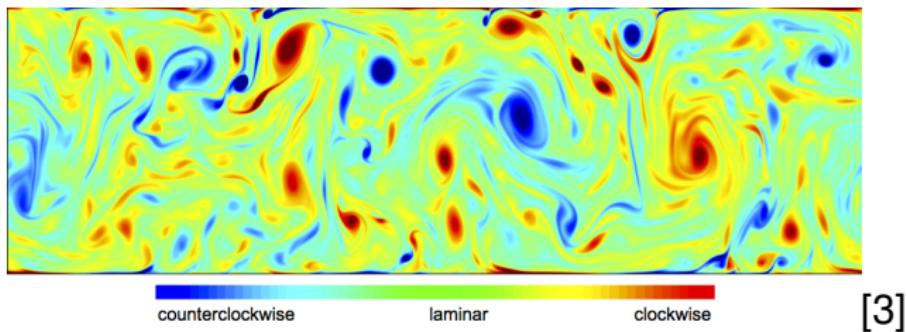
Vorticité



Les bandes rouges ne sont pas des vortex, mais plutôt des **lignes de séparation entre des régions** où les vecteurs vont dans des directions opposées.

Vorticité

La figure suivante propose un autre code de couleurs pour visualiser la vorticité. L'information véhiculée concerne cette fois-ci **la direction de rotation**.



C'est donc tout le contraire du code de couleurs précédent qui ne s'intéressait qu'à la vitesse et non la direction de rotation.

Flèches et glyphs

1 Introduction

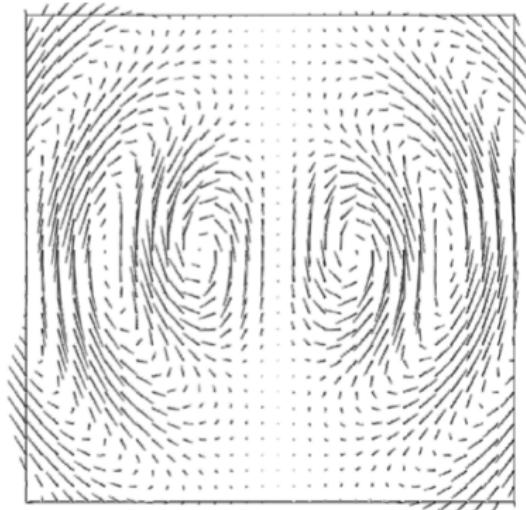
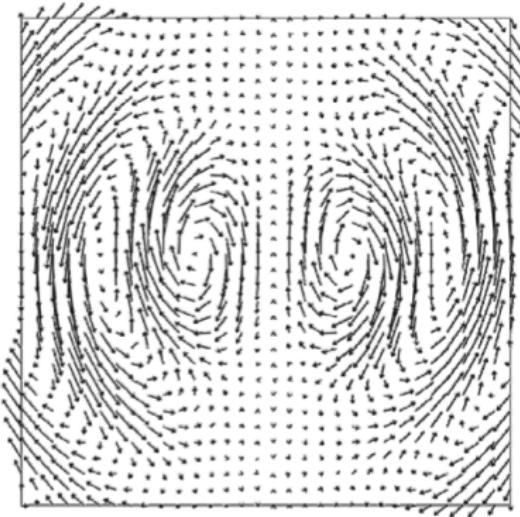
2 Visualisation de données vectorielles

- Divergence et vorticité
- Flèches et glyphs
- Code de couleurs vectoriel

3 Références

Glyphes et vecteurs

La représentation d'un champ vectoriel avec des glyphs est **la plus simple et la plus populaire** des techniques de visualisation pour ce type de données.



[5]

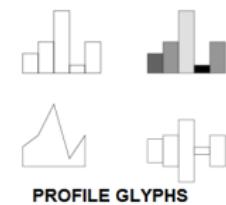
Glyphes et vecteurs

Cette technique associe un glyphe (une représentation visuelle) **avec chaque élément de l'ensemble de données vectorielles.**

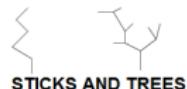
Les propriétés visuelles du glyphe seront définies par la valeur du vecteur à représenter. Celui-ci pourra influencer la **position**, la **taille**, la **couleur** et l'**orientation** du glyphe.

Glyphes et vecteurs

Un compromis devra souvent être fait entre **la densité des glyphes** et **le nombre d'attributs** à lui faire représenter : on doit résERVER plus d'espace pour des glyphes plus complexes.



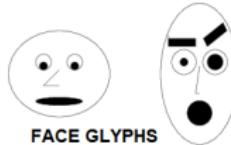
STARS AND METROGLYPHS



STICKS AND TREES



AUTOGLYPH/BOX GLYPH



FACE GLYPHS



ARROWS/WEATHERVANES

[4]

Vecteurs et hérissons

Le glyphe le plus simple est **la ligne**, qui permet d'illustrer la **position**, **l'orientation** et **l'amplitude** d'un ensemble de vecteurs.

Pour un ensemble de données vectorielles **définies sur un domaine discret** \mathcal{D} , on associe la ligne

$$\ell = x \rightarrow x + k\mathbf{v}(x)$$

pour chaque point $x \in \mathcal{D}$ possédant un vecteur $\mathbf{v}(x)$. Le paramètre k sert de facteur d'échelle.

Vecteurs et hérissons

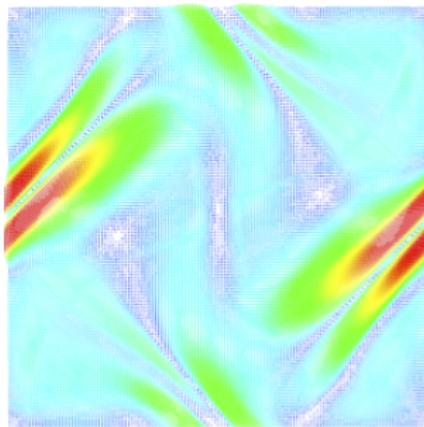
Les représentations de champ vectoriel à l'aide de lignes sont souvent appelés **diagrammes en hérisson**, ou encore *hedgehogs*.



Vecteurs et hérissons

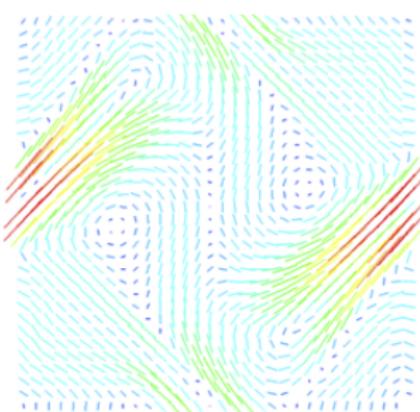
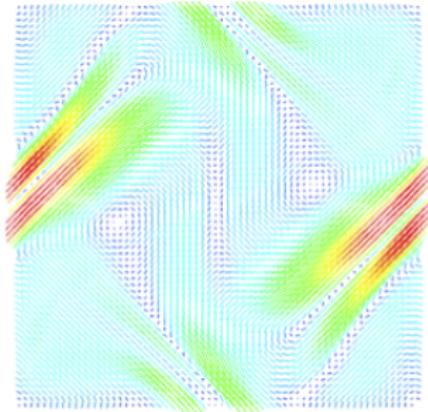
Soit un champ vectoriel $\mathbf{v} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ discrétisé sur un ensemble \mathcal{D} de 256×256 points.

La figure suivante montre la représentation en hérisson de \mathbf{v} avec un **sous-échantillonnage de facteur 2**.



Vecteurs et hérissons

Les deux images suivantes montrent le même champ vectoriel, mais **avec un sous-échantillonnage de facteurs 4 et 8**, respectivement.



[3]

Quels sont les **avantages** et les **inconvénients** associés au sous-échantillonnage ?

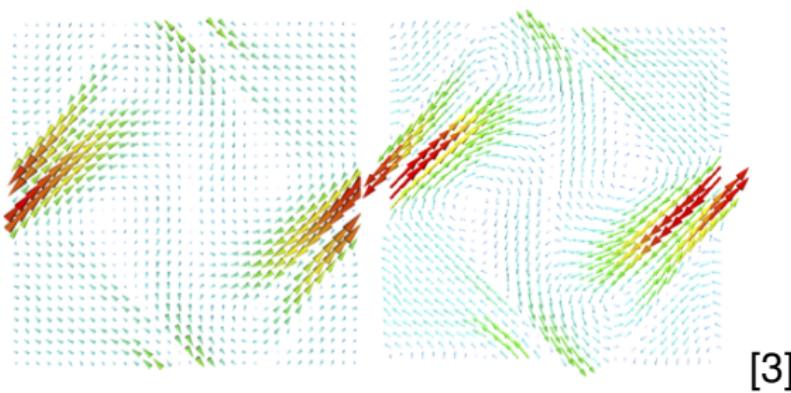
Vecteurs et hérissons

Dans ces graphiques, la longueur des lignes est **proportionnelle à l'amplitude du champ vectoriel**. La constante k est déterminée en fonction du facteur de sous-échantillonnage.

Les couleurs associées aux lignes sont aussi **définies en fonction de l'amplitude des vecteurs**. Il y a donc une redondance dans l'information.

Vecteurs et hérissons

On peut utiliser des formes plus complexes à la place des lignes. Deux choix classiques sont **les cônes** et **les flèches**.



Ceux-ci ont l'avantage de fournir l'information sur **la direction du champ vectoriel**, en plus de l'orientation de celui-ci.

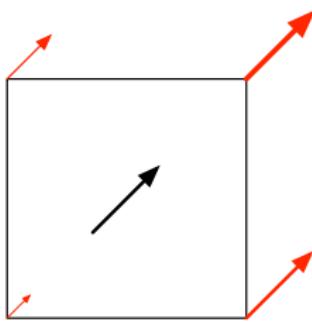
Vecteurs et hérissons

Le compromis à faire entre **le nombre d'attributs illustrés par un glyphe** et **l'espace qu'il occupe à l'écran** est une préoccupation centrale de ce mode de visualisation. Plus l'espace occupé est grand, plus le facteur de sous-échantillonnage doit être élevé.

En comparaison, l'utilisation d'un code de couleurs en visualisation de données scalaires permettait d'**associer une représentation visuelle à chaque donnée**. On dira qu'il s'agit d'une représentation **dense**.

Vecteurs et hérissons

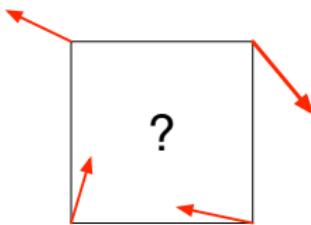
Attardons-nous aux **conséquences du sous-échantillonnage**. Pour ce faire, considérons la portion de champ vectoriel suivante.



Comme les vecteurs associés aux quatre coins de la cellule sont **très semblables**, un sous-échantillonnage n'affectera pas notre interprétation de la véritable nature du champ vectoriel.

Vecteurs et hérissons

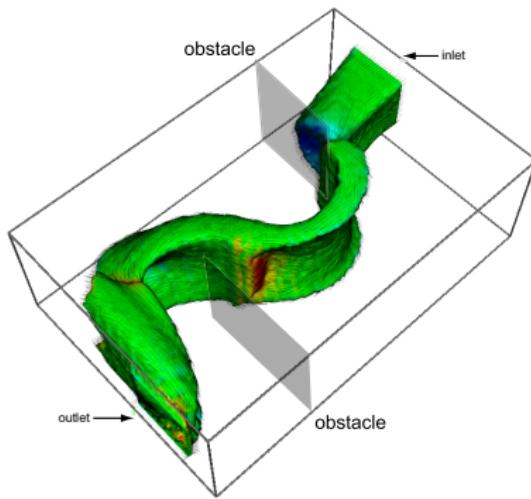
À l'opposé, si les vecteurs originaux ont **des directions et des amplitudes différentes**, le sous-échantillonnage causera des problèmes.



Dans une telle situation, comment calcule-t-on le vecteur à afficher ?
Quelle sera sa signification ?

Glyphes en 3D

Les glyphes permettent aussi la visualisation de champs vectoriels 3D. Considérons l'exemple d'un **écoulement d'eau dans un bassin** où sont situés deux obstacles qui dévient le parcours du liquide.



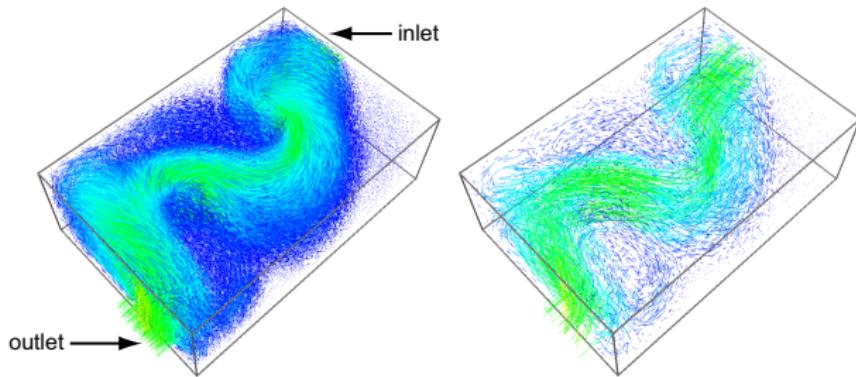
Glyphes en 3D

Le mouvement de l'eau est défini par un ensemble de $128 \times 85 \times 42$ vecteurs. **Si on illustre la totalité de ceux-ci** avec un diagramme en hérisson 3D, on s'expose, comme dans le cas 2D, à un problème de **recouvrement des glyphs**.

La solution est alors la même : **procéder à un sous-échantillonnage**.

Glyphes en 3D

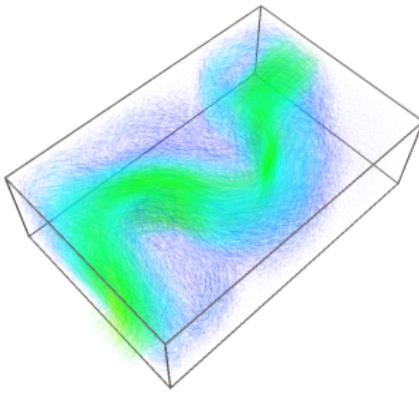
Les figures suivantes ont été obtenues en **choisissant aléatoirement 100 000 et 10 000 vecteurs** dans l'ensemble original.



Si la première présente encore de l'occlusion à cause de la trop grande densité des glyphs, la deuxième permet sans problème de **visualiser le comportement complet du liquide.**

Glyphes en 3D

On peut aussi **contourner le problème du recouvrement tout en conservant un nombre élevé de glyphs**. La solution est alors d'opter pour des glyphs ayant un certain niveau de **transparence**.



On compte ici 100 000 glyphs et **les effets de l'occultation sont grandement diminués**.

Code de couleurs vectoriel

1 Introduction

2 Visualisation de données vectorielles

- Divergence et vorticité
- Flèches et glyphs
- Code de couleurs vectoriel

3 Références

Champ vectoriel 2D

Il est maintenant clair que **l'utilisation de glyphes** pour illustrer un champ vectoriel **nécessite un sous-échantillonnage** (parfois important) des données originales afin d'avoir l'espace nécessaire pour afficher les glyphes.

Serait-il envisageable d'avoir une **représentation dense** pour un champ vectoriel ?

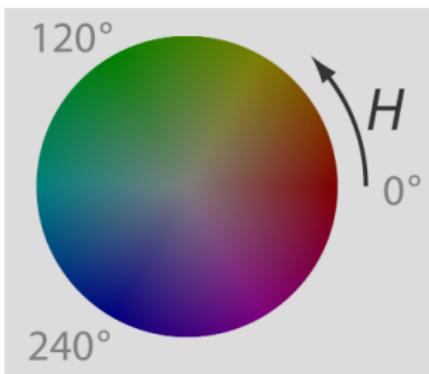
Champ vectoriel 2D

C'est ce que propose les **codes de couleurs vectoriels**. À l'image du placage de couleurs pour les données scalaires, un code de couleur vectoriel **associe une couleur à chaque point d'une surface** où est défini un champ vectoriel discrétilisé.

La couleur permet d'illustrer l'**orientation**, la **direction** et l'**amplitude** des vecteurs.

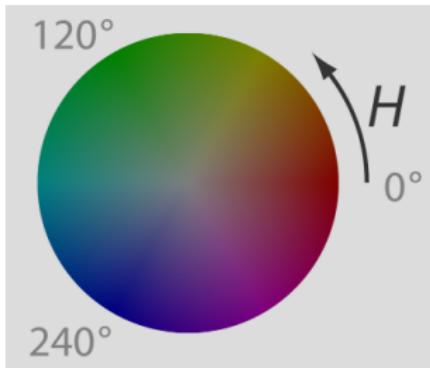
Champ vectoriel 2D

Un code de couleur vectoriel repose sur l'utilisation d'un **cercle chromatique**.



On utilisera souvent l'**espace couleur HSV**, puisqu'il se prête bien à la visualisation sous la forme d'un cercle chromatique.

Champ vectoriel 2D



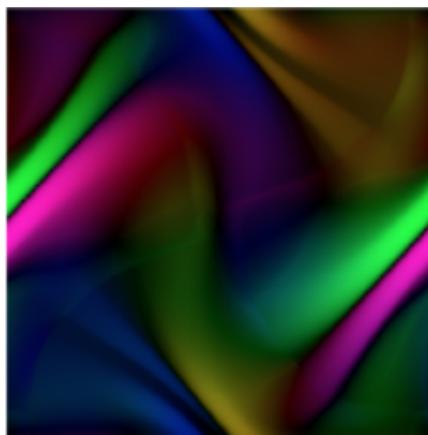
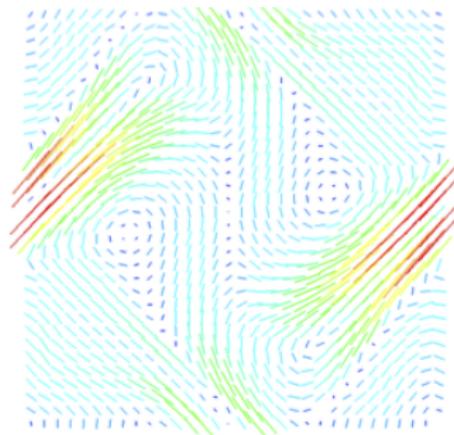
- Chaque **teinte** (*Hue*) correspond à un angle différent sur le cercle et donne ainsi l'information sur l'**orientation** et la **direction** du vecteur.
- La **saturation** représente quant à elle la distance entre le centre du cercle et une couleur, donnant ainsi l'information sur l'**amplitude** du vecteur.

Champ vectoriel 2D

Considérons que l'on utilise **un cercle chromatique de rayon un**. La première étape pour appliquer un code de couleurs consiste à **normaliser les vecteurs** du champ vectoriel de telle sorte que le plus long soit de longueur un.

De cette façon, chaque vecteur sera représenté par la couleur vers laquelle il pointe **si on le place au centre du cercle chromatique**.

Champ vectoriel 2D



Champ vectoriel 2D

D'entrée de jeu, on constate que le résultat visuel ne souffre d'**aucun problème de sous-échantillonnage**, comparativement à la représentation par glyphs.

En contrepartie, cette forme d'affichage est **beaucoup moins intuitive** que le diagramme en hérisson : l'opération d'**associer les couleurs à une direction et une amplitude** est, malgré la présence d'une légende, moins évidente.

Champ vectoriel 3D

Il est aussi possible d'établir **un code de couleurs pour les champs vectoriels 3D.**

La façon la plus simple d'y arriver est de directement **convertir les composantes v_x , v_y et v_z des vecteurs en couleur RGB.**

Champ vectoriel 3D

Le fait de **représenter chaque direction indépendamment**, comme des données scalaires non reliées, force l'utilisateur à lui-même faire la corrélation entre trois images à niveaux de gris.

Qui plus est, l'estimation de la direction d'un vecteur simplement à partir de trois composantes R, G et B n'est pas évidente.

Références

1 Introduction

2 Visualisation de données vectorielles

3 Références

Références I

-  C. D. Hansen and C. R. Johnson.
The visualization handbook, 2004.
-  W. Schroeder, K. Martin, and B. Lorensen.
The visualization toolkit : An object-oriented approach to 3d graphics, 2006.
-  A. C. Telea.
Data visualization : Principles and practice, 2008.
-  M. Ward, G. Grinstein, and D. Keim.
Interactive data visualization : Foundations, techniques and applications, 2010.
-  H. Wright.
Introduction to scientific visualization, 2007.