Fire Monkeys
Modélisation de feu en temps réel

Benjamin Aupetit - Julien Champeau - Arnaud Emilien

Jun 14 2010

Objectif

Réaliser un modéle de combustion d'objets en 3D temps réel.
image

- Le feu
- La fumée
- Iteraction avec des objets
- Propagation sur l'objet
- Combustion d'objet

Les étapes de notre démarche

- découvrir le milieu scientifique
- étudier différents articles
- Concevoir notre propre modèle
- Implémenter le modèle
- Améliorer le modèle GPU (objectif initial)

Modèle d'intéraction

Interractions entre le modèle de feu et le modèle d'objet.

- Présence des objets (i.e. : pas de flamme a l'interieur des objets)
- Transimission des informations de chaleur d'un modèle à l'autre.
- Gestion de la "pyrolise" des objets.

Principe

- Modèle basé sur le travail de **Jos Stam**, notemment **Stable Fluids**(SIGGRAPH 99 Conference Proceedings).
- Résolution de manière approchée des équation de Navier-Stokes pour la dynamique des fluides icompréssibles.
- Un rendu utilisant le principe de “BillBoard”.

La diffusion

La diffusion représente la capacité du fluide à se déplacer dans le milieu ambiant.

$$\frac{\partial \vec{u}}{\partial t} = \nu \nabla^2 \vec{u} \tag{1}$$

Figure: Equation de diffusion

Diffusion

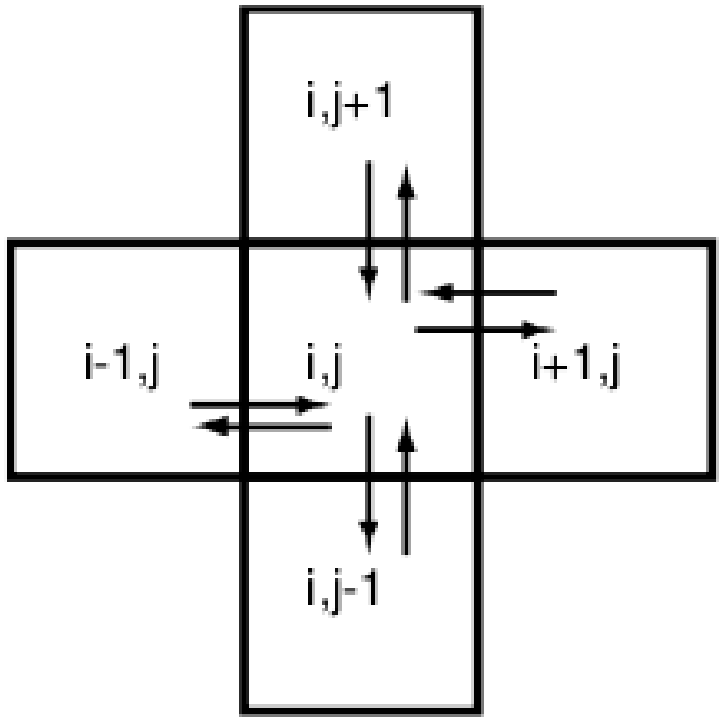

Figure: Diffusion sur une grille 2D

L'advection

Le champ de vitesse va servir à transporter la quantité de fluide dans l'espace. L'auto-advection du champ de vitesse est ce qui va permettre le mouvement des particules de gaz dans le milieu.

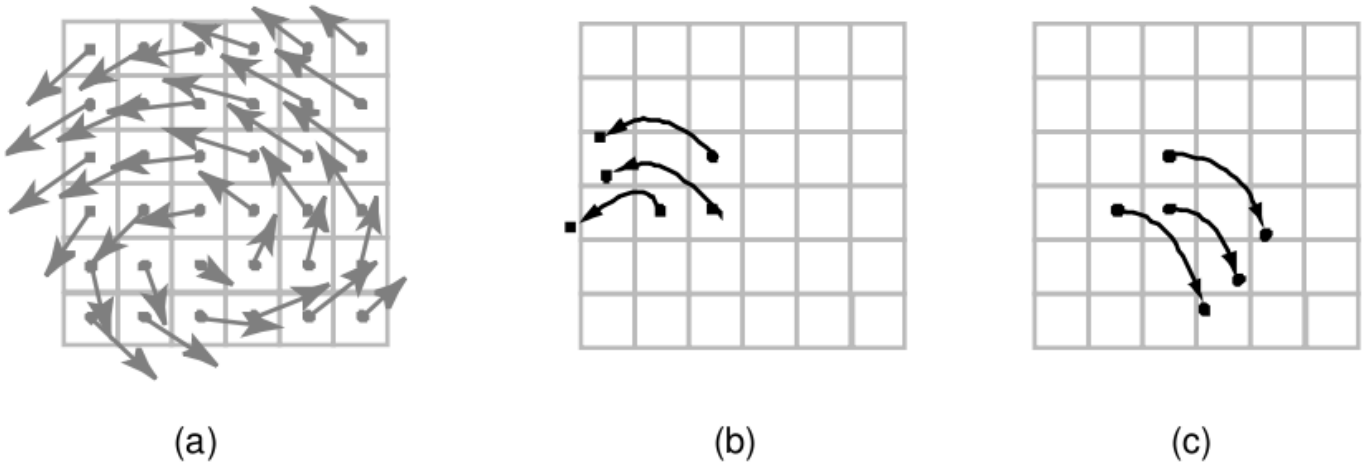

Figure: Principe de l'advection évoquée par **Jos Stam**

La projection

La projection permet de forcer le champ de vitesse à conserver la masse. Pour faire cela il suffit de soustraire le gradient au champ de vitesse après la diffusion et après l'advection de celui-ci.

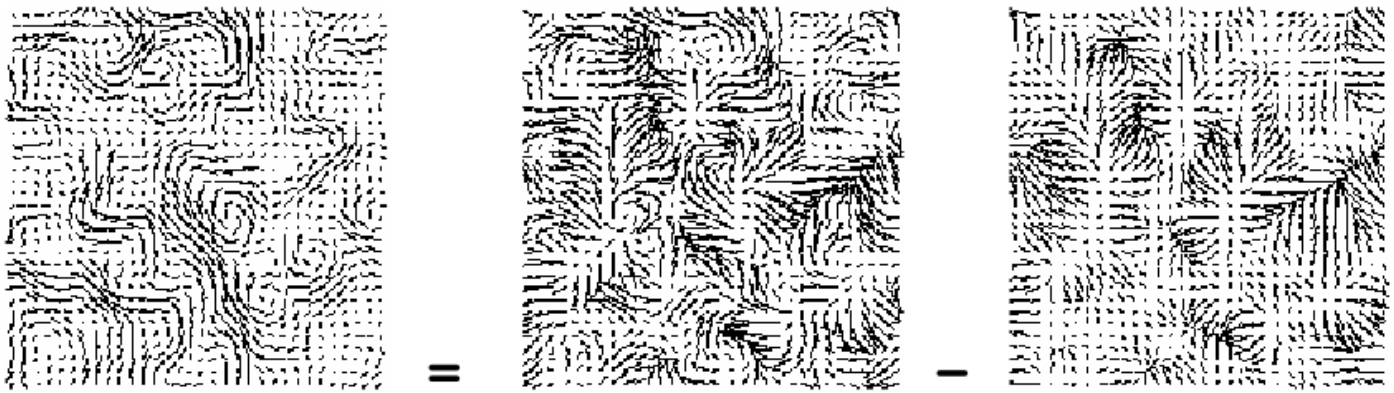

Figure: Correction du champ de vitesse par la soustraction du gradient

Le rendu

Pour le rendu, nous utilisons une technique de “billboard”, ce qui permet d’avoir un cout d’affichage assez faible.

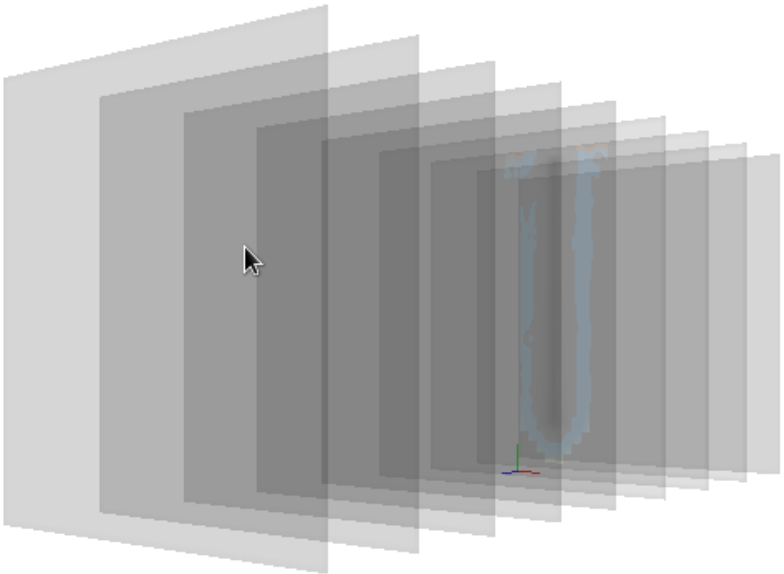

Figure: Vue des plans affichés

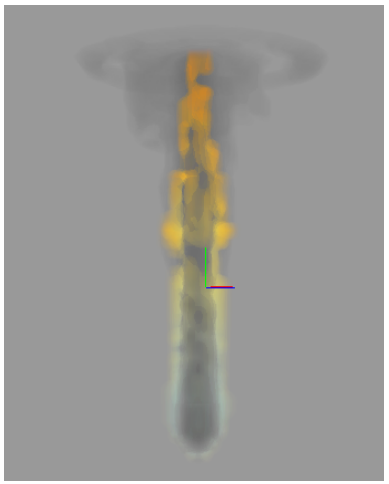

Figure: Affichage normal face caméra

Pour amélioré le rendu et le rendre plus réaliste nous avons aussi implémenter un bruit de Perlin qui agit sur l'image.

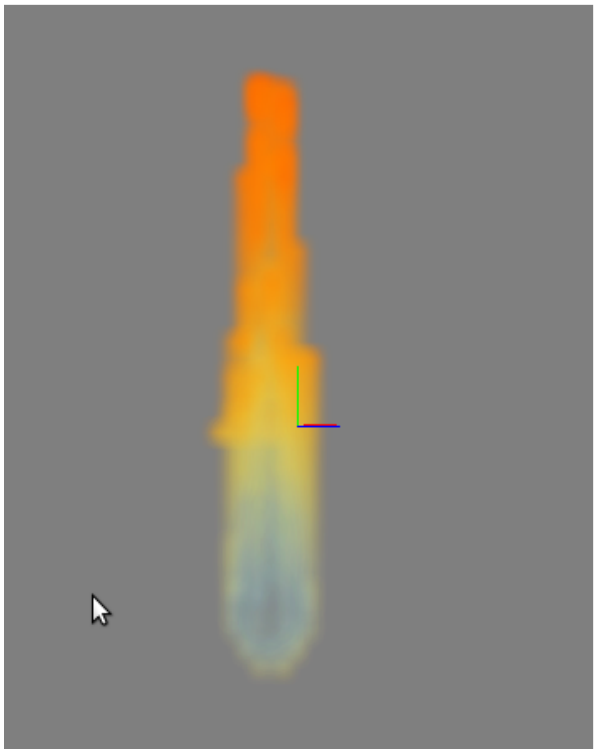

Figure: Sans le bruit de Perlin

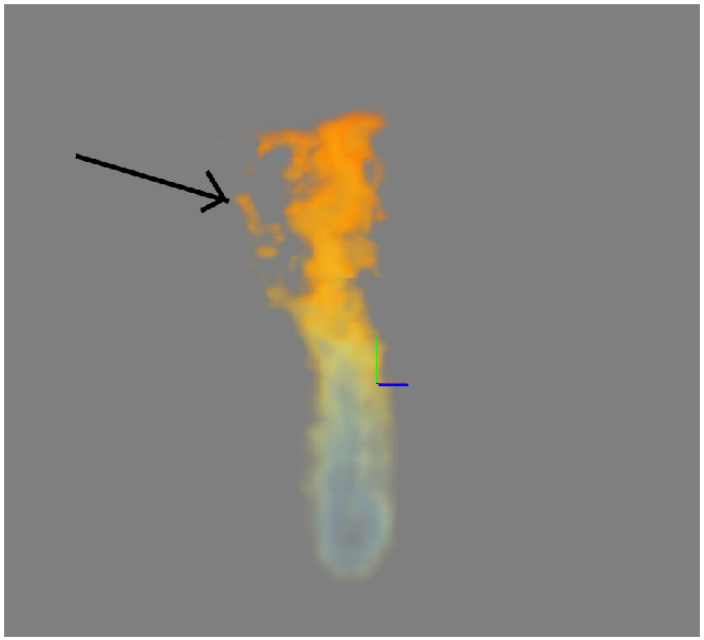

Figure: Avec le bruit de Perlin

Une représentation par voxel

Un objet = un champ de voxels
Interraction facilité avec le modèle d'objet
Permet de gerer des objets non uniforme

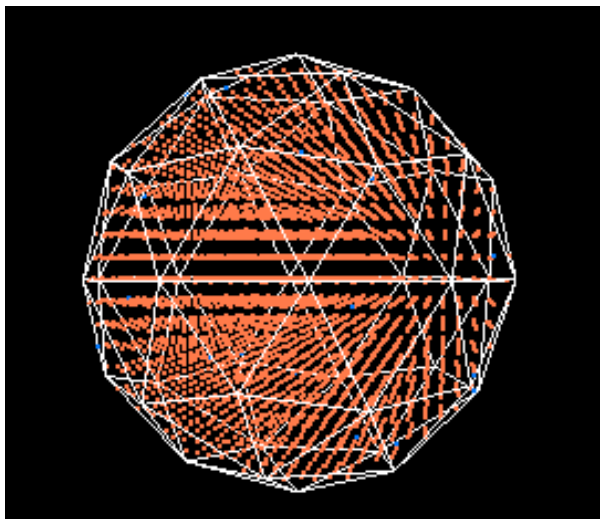

Figure: Exemple de champ de distance

Rendu

Besoin de recalculer la surface de l'objet localement.
$\Rightarrow$ utilisation d'un algorithme de marching cube adaptatif.

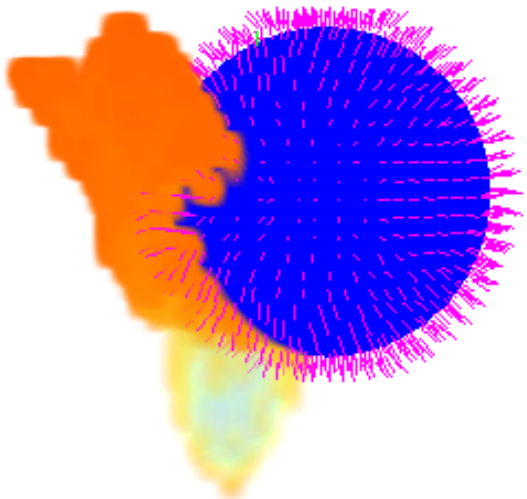

Figure: Sphere qui brûle, avec affichage du champ de répulsion

Le calcul sur GPU

Principe de fonctionnement

Les problèmes rencontrés

Réalisations

Démonstrations

Conclusion

Remerciements

- Marie-Paule Cani pour avoir accepté de nous encadrer, pour ses conseils et indices de recherche.
- Cyril Crassin pour nous avoir aidé à comprendre le fonctionnement de GLSL.
- Aurelie Catel pour le suivi de gestion de projet.
- Nintendo^{TM} pour Super Smash Bross Melee ©.