

# Physique appliquée

Lignes de champ

2021 – 2022

AHMADI Zabiullah, KURTESHI Agnon

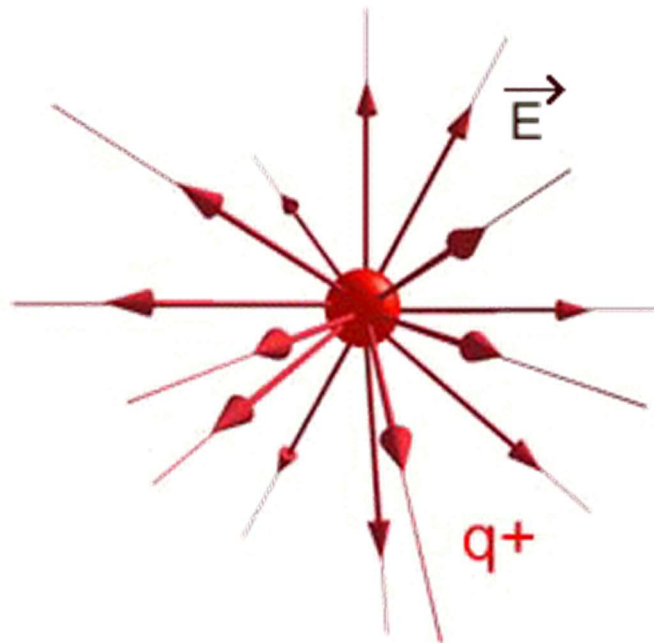


Figure 1 : Ligne de champ

# Table des matières

I.	Introduction :	4
	Mise en contexte .....	4
	Annonce du plan .....	4
II.	Rappels théoriques sur les champs électriques.....	5
	Quelques formules .....	5
	Ligne de champ .....	5
III.	Théorie en pratique (programmation).....	6
	Structure de données .....	6
IV.	Algorithme.....	7
	Génération des points aléatoires dans l'univers.....	7
	Génération des charges aléatoires.....	7
	Calcul de la charge et de la norme .....	7
	Dessiner les lignes de champ électriques .....	7
V.	Problèmes rencontrés :	8
VI.	Résultat et validation :	8
	Une charge .....	8
	Deux charges de même polarité .....	9
	Deux charges de polarité différente .....	9
VII.	Conclusion :	10
	Rappel de la réalisation .....	10
	Résultats importants .....	10
	Amélioration possible.....	10

# Table des illustrations

Figure 1 : Ligne de champ .....	1
Figure 2: Simulation avec une charge .....	8
Figure 3: Simulation avec deux charges négatives.....	9
Figure 4: Simulation avec deux charges positives.....	9
Figure 5: Simulation avec deux charges de polarité différente .....	9

# I. Introduction :

## Mise en contexte

Les lignes de champ ne sont pas faciles à se représenter. Introduites par Michael Faraday, un des plus grands découvreurs dans les domaines de l'électricité et du magnétisme, elles permettent de représenter la direction, la grandeur et le sens des champs électriques dans l'espace.

Afin de pouvoir expérimenter l'interaction entre les charges dans l'espace, nous avons développé un programme qui permet de visualiser les lignes de champ dans l'espace.

Grâce à ce dernier, nous pouvons ainsi avoir un support visuel simulant la direction des lignes de champ en fonction de la polarité des charges.

Pour implémenter cette simulation, nous avons utilisé le langage C. Par ailleurs, nous avons eu un fichier source contenant le squelette de base du programme. Ensuite, nous avons créé les fonctions qui permettent de mener à bien notre projet.

## Annonce du plan

Dans un premier temps, nous allons faire un rappel théorique sur ce qui compose les lignes de champ, et nous verrons ensuite l'algorithme qui nous a permis de réaliser le programme. Et pour finir, nous verrons les résultats du programme.

## II. Rappels théoriques sur les champs électriques

### Quelques formules

La formule du champ électrique en un point généré par une charge électrique  $q_1$  :

$$\vec{E} = K \frac{q_1}{r^2}$$

- $E$  est l'intensité du champ électrique par rapport à notre charge  $q_1$ .
- $K$  est la constante électrostatique étant d'environ  **$8.98 \cdot 10^9 \text{ Nm}^2 / \text{C}$** .
- $r$  est la distance entre la charge et le point où le champ électrique est mesuré.

Les lignes de champ en un point sont dues au vecteur résultant des lignes de champ des charges.

$$\vec{E} = \sum_i^N E_i \cdot \frac{q_i \vec{P}}{||q_i \vec{P}||}$$

Pour dessiner les lignes de champ, on met plusieurs points aléatoires dans notre écran qui a une taille fixe.

Ensuite, on calcule l'intensité des lignes de champ pour chacun des points à l'aide de la formule suivante :

$$P_{suivant} = P + \delta x \cdot \frac{\vec{E}}{||\vec{E}||}$$

Cela va calculer la ligne de champ à chaque point et avancer de  $\delta x$  dans la direction du champ électrique.

De plus, nous avons également besoin de limiter notre champ électrique à notre taille d'écran.

$$\delta x = \frac{1}{\sqrt{\text{largeur}^2 + \text{hauteur}^2}}$$

### Ligne de champ

Le champ électrique est représenté par un vecteur, et pour le représenter, on parle de champ vectoriel. Pour chaque point de l'espace, le champ électrique va associer un vecteur qui aura une amplitude et une direction en ce point.

Pour visualiser un champ électrique, il faut utiliser plusieurs lignes de champ afin de pouvoir visualiser la direction du champ électrique.

Les lignes de champs sont dessinées afin d'indiquer la direction de la force exercée par des charges électriques sur une charge test positive. La convention veut que les lignes de champ sortent de la charge positive pour entrer dans la charge négative.

### III. Théorie en pratique (programmation)

#### Structure de données

##### 1. Charge :

```
typedef struct
{
    double q;
    vec2 pos;
} charge_t;
```

La structure « charge\_t » va permettre de représenter une charge.

##### 2. Vecteur :

```
typedef struct _vec2
{
    double x, y;
} vec2;
```

La structure « vec2 » va permettre de pouvoir faire les calculs en relation avec les vecteurs.

##### 3. Coordonnées :

```
typedef struct _coordinates
{
    uint32_t row, column;
} coordinates;
```

La structure « coordinates » va permettre de représenter un vecteur en coordonnées de l'écran.

## IV. Algorithme

### Génération des points aléatoires dans l'univers

Nous avons une fonction « generate\_random\_points » qui génère des nombres entre [0 : 1, 0 : 1] (taille de l'univers, axe x et y) qui nous permettrons ensuite de voir si la ligne de champ traverse ses points, et de pouvoir dessiner les lignes de champ.

### Génération des charges aléatoires

Afin de pouvoir placer une charge, nous commençons par déterminer sa charge électrique afin de voir si c'est une charge négative ou positive. Ensuite, on génère la position de la charge. Ces deux valeurs sont choisies aléatoirement.

### Calcul de la charge et de la norme

Une particule chargée, soit positivement, ou négativement, génère un champ électrique autour d'elle. Maintenant, il faut pouvoir calculer le champ électrique qui est en unité de [N]/[C].

Pour ce faire, nous avons utilisé une fonction « compute\_champs » qui permet de nous renvoyer l'intensité de champ électrique. Elle reprend exactement la formule montrée plus haut.

### Dessiner les lignes de champ électriques

Pour chaque point aléatoire présent dans l'univers, nous allons calculer le champ résultant en ce point :

$$P_{suivant} = P + \delta x \cdot \frac{\vec{E}}{||\vec{E}||}$$

$$P_{suivant} = P - \delta x \cdot \frac{\vec{E}}{||\vec{E}||}$$

## V. Problèmes rencontrés :

Lors de notre réalisation, nous avons eu du mal à bien afficher le segment entre le point actuel et celui de suivant. Cela est dû au fait que dans notre simulation, les points sont très proches entre eux. Donc, ils ressemblent à un point, mais l'affichage des lignes correcte.

## VI. Résultat et validation :

### Une charge

Maintenant que nous avons vu ce qui compose le programme, nous pouvons l'exécuter et ainsi faire apparaître les lignes de champ.

Afin de pouvoir tester si notre programme fonctionne, nous avons testé d'afficher une seule charge et de voir si les lignes de champ sont bien droites. En effet, comme il n'y a qu'une charge, les lignes de champ ne sont pas soumises à une attraction ou une répulsion. C'est un détail qui permet de valider le fonctionnement du programme.

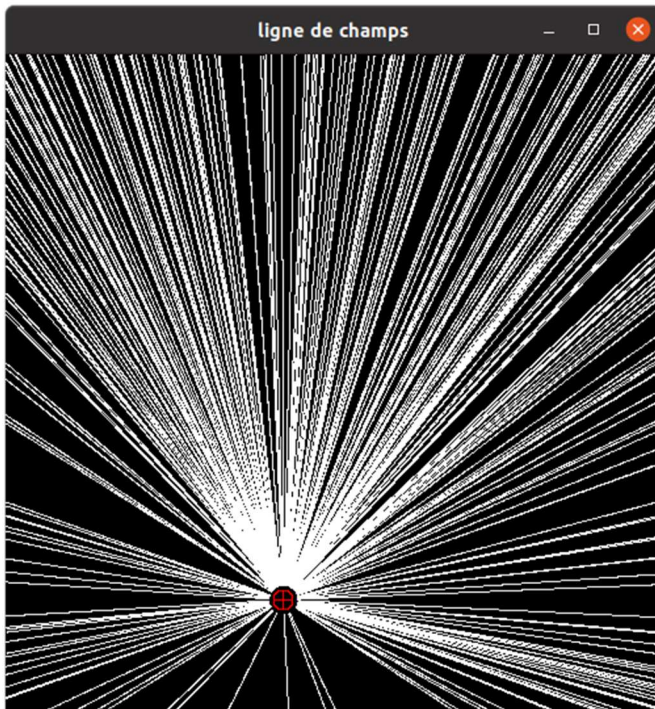


Figure 2: Simulation avec une charge



## Deux charges de même polarité

Maintenant, on peut le tester avec deux charges pour voir si les lignes de champ s'attirent ou se repoussent entre les deux charges.

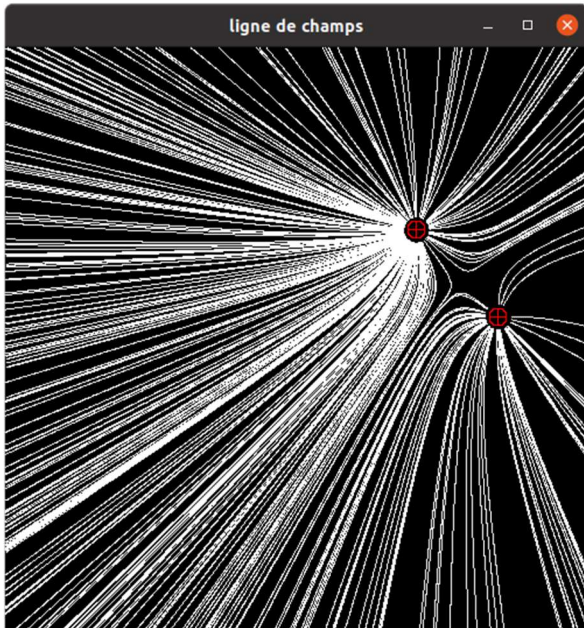


Figure 4: Simulation avec deux charges positives

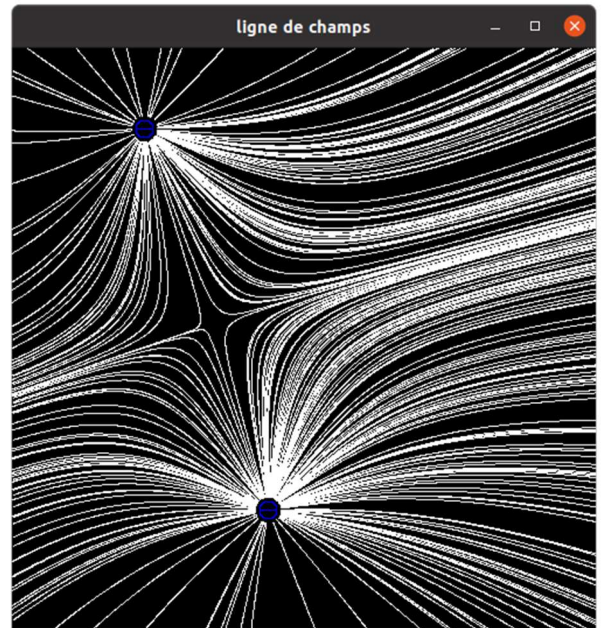


Figure 3: Simulation avec deux charges négatives

Avec deux charges positives ou deux charges négatives, on peut voir que les lignes de champ se repoussent. Ce phénomène permet de pouvoir valider le fonctionnement de notre programme.

## Deux charges de polarité différente

On teste maintenant avec deux charges de polarité différente. On peut voir que les lignes de champ s'attirent entre les deux charges, et donc, on peut logiquement dire que le programme est fonctionnel.

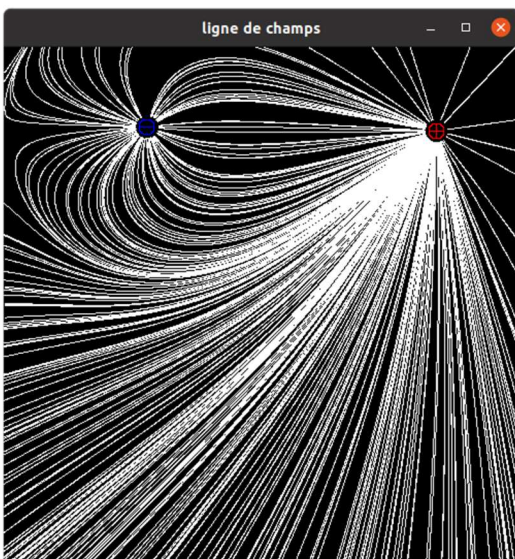


Figure 5: Simulation avec deux charges de polarité différente

## VII. Conclusion :

### Rappel de la réalisation

Comme vu précédemment, notre programme consiste à simuler un phénomène lié à la physique des charges électriques. Pour arriver à programmer ce dernier, nous avons utilisés plusieurs formules, comme celle du champ électrique. Ensuite, dans un univers de taille  $[0 : 1, 0 : 1]$ , nous avons dessiner pleins de points où, avec un algorithme, nous allons pouvoir calculer les lignes de champ passant sur ces points. C'est le cœur même du projet, car, c'est comme ça que nous allons dessiner les lignes.

### Résultats importants

Les choses les plus importantes à retenir, c'est de voir que le programme s'exécute sans erreur, et de plus qu'il simule exactement ce que nous voulons, et bien évidemment, simule le réel phénomène physique. Cela renvoie au fait que le programme est fonctionnel. Comme vu précédemment, avec une charge, les lignes ne sont soumises à aucune force, et donc, elles sont toutes droites, mais dès qu'on rajoute une charge en plus, des phénomènes sur les lignes de champ apparaissent.

### Amélioration possible

Une amélioration possible, c'est le mouvement des charges. En effet, avec la simulation actuelle, nous avons des charges statiques, et on ne peut que visualiser les lignes de champ. Mais, avec le mouvement des charges, on pourrait de plus observer l'interaction des charges entre elles et leur mouvement dans l'espace. Il faudrait pour cela une gestion du temps, afin que les charges ne bougent pas trop vite, et que l'on ne puisse pas observer leur mouvement.