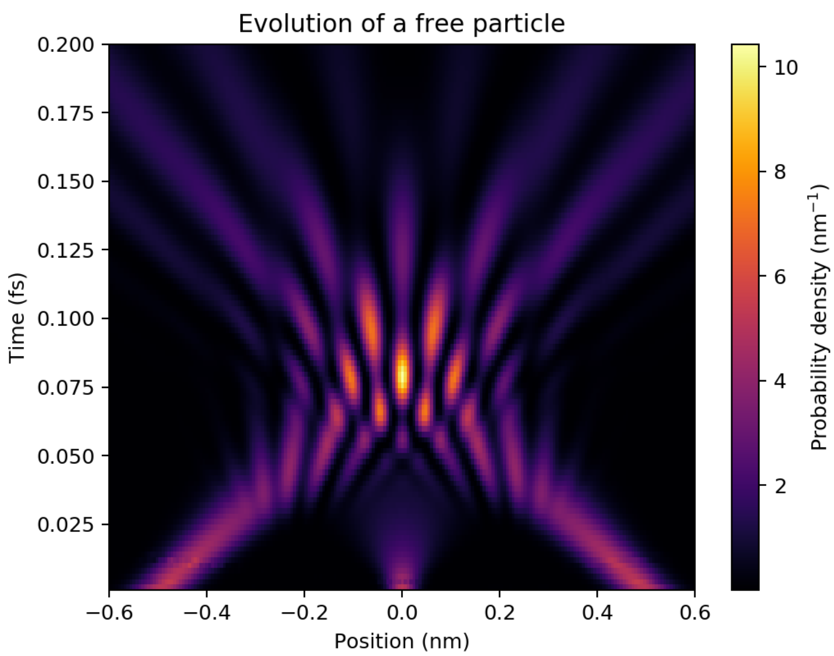
Projet Informatique
Équation de Schrödinger dépendant du temps à une dimension

Résolution grâce aux propagateurs de Schrödinger

RENAUDINEAU Adrien

23 janvier 2022

Table des matières



Projet réalisé dans le cadre du cours d'informatique 2021-2022 du Magistère de Physique Fondamentale d'Orsay[1].

1 Définition et objectif du projet

1.1 Introduction

Le but initial du projet est d'effectuer le même travail que le sujet proposé "Résolution numérique de l'équation de Schrödinger dépendant du temps à une dimension"[2] – avec un potentiel quelconque et à 1 dimension – mais en utilisant les propagateurs de l'équation de Schrödinger au lieu de la résoudre directement.

Dans ce contexte, cela ne simplifie pas les calculs, au contraire, mais cela permet notamment une plus grande liberté du choix de la fenêtre espace-temps de simulation.[3]

Tout le code source (en C pour les calculs et en Python pour la visualisation) est disponible ici sous licence MIT : https://github.com/Ardakaniz/pathintegral.

1.2 Définition

On définit[4] en premier lieu l'état formel $|xt\rangle$ qui est l'extension des états propres de l'observable de position à des temps quelconques (notamment négatifs), sans nécessairement de réalité physique :

$$|xt\rangle \mathrel{\hat{=}} e^{+\frac{i}{\hbar}\hat{H}(t-t_0)}\,|x\rangle$$

On définit alors le propagateur de Feynman, propagateur de l'équation de Schrödinger, ou simplement propagateur :

$$\langle x_f t_f | x_i t_i\rangle \mathrel{\hat{=}} \langle x_f|\, e^{-i\hat{H}(t_f-t_i)}\,|x_i\rangle \tag{1}$$

Feynman a démontré qu'il était possible d'exprimer (1) grâce à l'action $\mathcal{S}$, après discrètisation du temps :

$$\langle x_f t_f | x_i t_i\rangle = \int \mathcal{D}x(t) e^{\frac{i}{\hbar}\mathcal{S}[x(t)]} \tag{2}$$

avec :

$$\mathcal{D}x(t) = \lim_{N\to\infty} \left(\frac{m}{2\pi i\hbar\varepsilon}\right)^{N/2} \left[\prod_{k=1}^{N-1} \mathrm{d}x_k\right]$$

Cela permet notamment d'exprimer n'importe quelle fonction d'onde :

$$\Psi(x_1,t_1) = \int_{\mathbb{R}} \mathrm{d}x_0\, \langle x_1 t_1 | x_0 t_0\rangle\, \Psi(x_0,t_0)$$

En particulier, ici nous allons vouloir exprimer $\Psi(x,t)$ à partir de la fonction d'onde initiale $\Psi(x,t=0)$ d'où :

$$\Psi(x,t) = \int_{\mathbb{R}} \mathrm{d}x_0\, \langle xt | x_0 0\rangle\, \Psi(x_0,0) \tag{3}$$

Pour plus de clarté, on désignera $\langle xt | x_0 0\rangle$ par $K(x_0,x,t)$ par la suite.

1.3 Objectif

L'objectif technique du projet a consisté à trouver un moyen de calculer numériquement le propagateur (2), i.e. une intégrale fonctionnelle.

Il a réalité été utilisé une approximation semi-classique de ce propagateur, valable pour $\mathcal{S} \gg \hbar$, que nous expliciterons en partie 2 et discuterons en partie 4.

L'objectif physique du projet... reste encore à définir.

[...]

---

1. http://www.magistere-physique.universite-paris-saclay.fr/
2. http://hebergement.u-psud.fr/mpo-informatique/sujets_projets_21_22.pdf p.13
3. En plus d'avoir un intérêt pédagogique car mettant en pratique mes cours sur les intégrales de chemin et car bien moins guidé dans la résolution.
4. http://pperso.th.u-psud.fr/page_perso/Moreau/Integrale_chemin-Cours.2020.pdf p.2

2 Résolution analytique

2.1 Calcul du propagateur

Rappelons l'expression du propagateur (2) :

$$\langle x_f t_f | x_i t_i \rangle = \int \mathcal{D}x(t) e^{\frac{i}{\hbar}\mathcal{S}[x(t)]}$$

En annexe A, on démontre que lorsque $\mathcal{S} \gg \hbar$, on peut l'approximer par :

$$K(x_i, x_f, t_f) \approx \sqrt{-\frac{1}{2\pi i\hbar}\frac{\partial^2 \mathcal{S}_{cl}}{\partial x_f \partial x_i}} e^{\frac{i}{\hbar}S_{cl}(x_i,x_f,t_f)}$$

Cela permet d'obtenir quand même un résultat exact avec un potentiel constant ou harmonique.

2.1.1 Calcul de l'action classique

Il suffit de résoudre les équations du mouvement, puis d'intégrer le lagrangien sur la trajectoire...

Exemple de l'oscillateur harmonique On considère une particule de masse $m = 1$ dans le potentiel 1D $V(x) = \frac{1}{2}x^2$.

On exprime alors le lagrangien :

$$\mathcal{L}(x, \dot{x}) = \frac{1}{2}\dot{x}^2 - \frac{1}{2}x^2$$

Après dérivation, on obtient l'équation du mouvement :

$$\frac{\mathrm{d}^2 x}{\mathrm{d}t^2} + x = 0$$

Dont la solution générale s'exprime :

$$x(t) = A\cos t + B\sin t$$

Les conditions aux limites $x(0) = x_i$ et $x(t_f) = x_f$ nous permettent de fixer A et B et alors on a :

$$\begin{cases} x(t) = x_i \cos t + (x_f - x_i \cos t_f)\,\frac{\sin t}{\sin t_f} \\ \dot{x}(t) = -x_i \sin t + (x_f - x_i \cos t_f)\,\frac{\cos t}{\sin t_f} \end{cases}$$

Ainsi :

$$\mathcal{L}(x, \dot{x}) = \frac{1}{2}\left(x_i^2 + x_f^2 - 2x_i x_f \cos t_f + x_i^2 \cos^2 t_f\right)\left(\sin^2 t - \cos^2 t\right) +$$

Enfin,

$$\mathcal{S}_{cl}(x_i, x_f, t_f) = \frac{1}{2\sin t_f}\left[\left(x_f^2 + x_i^2\right)\cos t_f\right]$$

2.1.2 Calcul de la dérivée de l'action

$\mathcal{S}_{cl}$ représente la fonction principale de Hamilton, on a donc notamment la relation :

$$\frac{\partial \mathcal{S}_{cl}}{\partial x_f} = p_f$$

D'où :

$$\frac{\partial^2 \mathcal{S}_{cl}}{\partial x_f \partial x_i} = \frac{\partial p_f}{\partial x_i} = m\frac{\partial \dot{x}_f}{\partial x_i} = m\frac{\partial \dot{x}(t_f)}{\partial x_i}$$

Finalement,

$$K(x_i, x_f, t_f) \approx \sqrt{-\frac{m}{2\pi i\hbar}\frac{\partial \dot{x}(t_f)}{\partial x_i}} e^{\frac{i}{\hbar}S_{cl}(x_i,x_f,t_f)}$$

2.2 Calcul de la fonction d'onde

bla[5]

3 Résolution numérique

ble

4 Limites, problèmes, et possibles améliorations

blu

5 Conclusion

bli (Je veux faire de la physique numérique. http://www.ens-lyon.fr/MasterSDM/fr/master-2/m2-modelisation-numerique/presentation-du-m2-chimie-et-physique-computationnelle)

A Démonstration de l'approximation semi-classique

blo

$$\langle x_f t_f | x_i t_i \rangle \approx \sqrt{\frac{m}{2\pi i\hbar\phi(t_f)}} e^{\frac{i}{\hbar} S_{cl}(x_f,t_f,x_i,t_i)} \tag{4}$$

$$S_{cl}(x_f,t_f,x_i,t_i) = S[x_{cl}] = \int_{t_i}^{t_f} \mathcal{L}(x,\dot{x})\mathrm{d}t \tag{5}$$

$$\delta S[x_{cl}] = 0$$

Tous les éléments pour démontrer et comprendre au lien http://www.blau.itp.unibe.ch/lecturesPI.pdf

B Algorithmes

Liste des algorithmes

qsd

```c
#include <stdlib.h>
#include <stdio.h>

int main(void) {
  printf("Hello, World!\n");

  return EXIT_SUCCESS;
}
```

Algorithme 1 – Hello World en C

5. https://www.physics.mcgill.ca/~hilke/719.pdf p. 30