

---

# Algorithme de la Boule Pesante

pour l'Optimisation des Paramètres de Tir  
d'un Projectile à Masse Variable

---



samedi 20 décembre 2025

NKAMWA WAKIEU Randy  
[school.algorush.fr](https://school.algorush.fr)

# Plan de la Présentation

---



## Problématique et Contexte

Comprendre le défi physique du projectile à masse variable



## Méthodologie : La Boule Pesante

Découvrir l'algorithme et son intuition mathématique



## Application et Implémentation

Comment nous avons mis l'algorithme en œuvre



## Résultats et Visualisations

Les performances et les résultats obtenus



## Conclusion

Synthèse, avantages, limites et perspectives

# Le Problème Physique

## Projectile à Masse Variable

Le projectile perd de sa masse de manière exponentielle due à la consommation de carburant.

Masse totale:

$$M(t) = M_0 + m_{\text{restante}}(t)$$

Masse restante:

$$m_{\text{restante}}(t) = m_0 \cdot \exp(-\lambda t)$$

### Objectif

Déterminer les paramètres de tir initiaux optimaux ( $V_0$ ,  $\alpha$ ,  $V_e$ ) — vitesse initiale, angle d'élévation, et vitesse d'éjection — afin de minimiser l'erreur de repérage en abscisse.

## Forces en Jeu



### Poussée

Proportionnelle au débit massique et à la vitesse d'éjection relative  $V_e$ .

$$F_{\text{poussée}}(t) = V_e \cdot \lambda \cdot m_0 \cdot \exp(-\lambda t)$$

*Décroît exponentiellement avec le temps*



### Gravité

Force constante agissant verticalement.



### Traînée Aérodynamique

Modélisée par des forces dépendant de l'intensité des perturbations atmosphériques  $V_r$ .

$$F_{\text{traînée},i} = -\frac{1}{2} \cdot \rho \cdot C_i \cdot A_i \cdot V_r^2$$

# Principe de la Boule Pesante

L'optimisation est souvent comparée à la recherche du point le plus bas dans un paysage vallonné:

## 💡 Intuition Physique

- L'algorithme imite le mouvement d'une **boule pesante** dans un paysage d'optimisation
- Le **momentum** (inertie) permet de franchir les petits obstacles
- La boule accumule de la vitesse dans les descentes, ce qui aide à sortir des minima locaux

## 🔑 Concepts Clés

- **Gradient:** Direction de la pente la plus raide
- **Momentum:** Capacité à conserver la vitesse et direction passées
- **Inertie:** Résistance au changement de direction



### Gradient Classique



S'arrête au premier creux



Ralentit dans les zones plates



Pas de mémoire des directions précédentes



### Boule Pesante



Traverse les petites bosses



Accélère dans les longues descentes



Conserve l'élan des étapes précédentes

# Formulation Mathématique

## Relation de Vitesse

$$v_{k+1} = \gamma v_k - \alpha \nabla f(\theta_k)$$

## Relation de Mise à Jour des Paramètres

$$\theta_{k+1} = \theta_k + v_{k+1}$$

où  $\gamma \in [0, 1)$  est le coefficient d'inertie et  $\alpha$  le pas d'apprentissage

## Concepts Clés



$\theta$  : Vecteur des paramètres à optimiser (angle, vitesse initiale, vitesse d'éjection)

$v$  : Vecteur de "vitesse" ou de "momentum", représentant la direction et l'amplitude du déplacement



$\gamma$  : Coefficient d'inertie ou de momentum, détermine la part de la vitesse précédente conservée



$\alpha$  : Pas d'apprentissage, contrôlant l'influence du gradient actuel sur la mise à jour

$\nabla f(\theta_k)$  : Gradient de la fonction coût  $f$  au point  $\theta_k$ , indiquant la direction de la plus forte augmentation

# Application au Tir Balistique

## 🎯 Paramètres d'Optimisation



$V_0$

Vitesse initiale



$\alpha$

Angle d'élévation



$V_e$

Vitesse d'éjection

## 📈 Fonction Coût

$$f(\theta) = |x_{chute}(\theta) - x_{cible}|$$

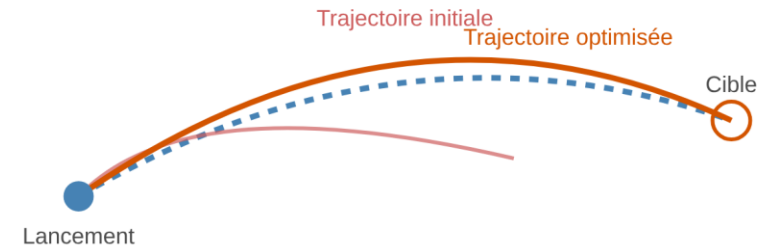
où  $\theta = (V_0, \alpha, V_e)$  représente le vecteur des paramètres de lancement.

## 📊 Calcul du Gradient

### Avantages du gradient analytique :

- Élimination des simulations numériques coûteuses
- Pas besoin de calculs par différences finies
- Précision accrue sans erreurs de discrétisation

## Optimisation Balistique



### Optimisation par la Boule Pesante

L'algorithme optimise les paramètres de tir pour minimiser l'erreur en abscisse.

→ Mise à jour de la vitesse

$$v_{k+1} = \gamma v_k - \alpha \nabla f(\theta_k)$$

→ Mise à jour des paramètres

$$\theta_{k+1} = \theta_k + v_{k+1}$$

# Implémentation Pratique

## 1 Initialisation

Définir les paramètres initiaux:

- Vecteur initial de paramètres:  $\theta_0 = (V_0, \alpha, V_e)$
- Vitesse (momentum) initiale:  $v_0 = 0$

## 2 Calcul du Gradient

Évaluer le gradient analytiquement:

$$\nabla f(\theta_k) = \text{dérivée de } |x_{\text{chute}}(\theta_k) - x_{\text{cible}}|$$

## 3 Mise à Jour de la Vitesse

Intégrer le momentum et la direction du gradient:

$$v_{k+1} = \gamma v_k - \alpha \nabla f(\theta_k)$$

## 4 Mise à Jour des Paramètres

Ajuster les paramètres de tir:

$$\theta_{k+1} = \theta_k + v_{k+1}$$

## 5 Test d'Arrêt

Vérifier les critères de convergence:

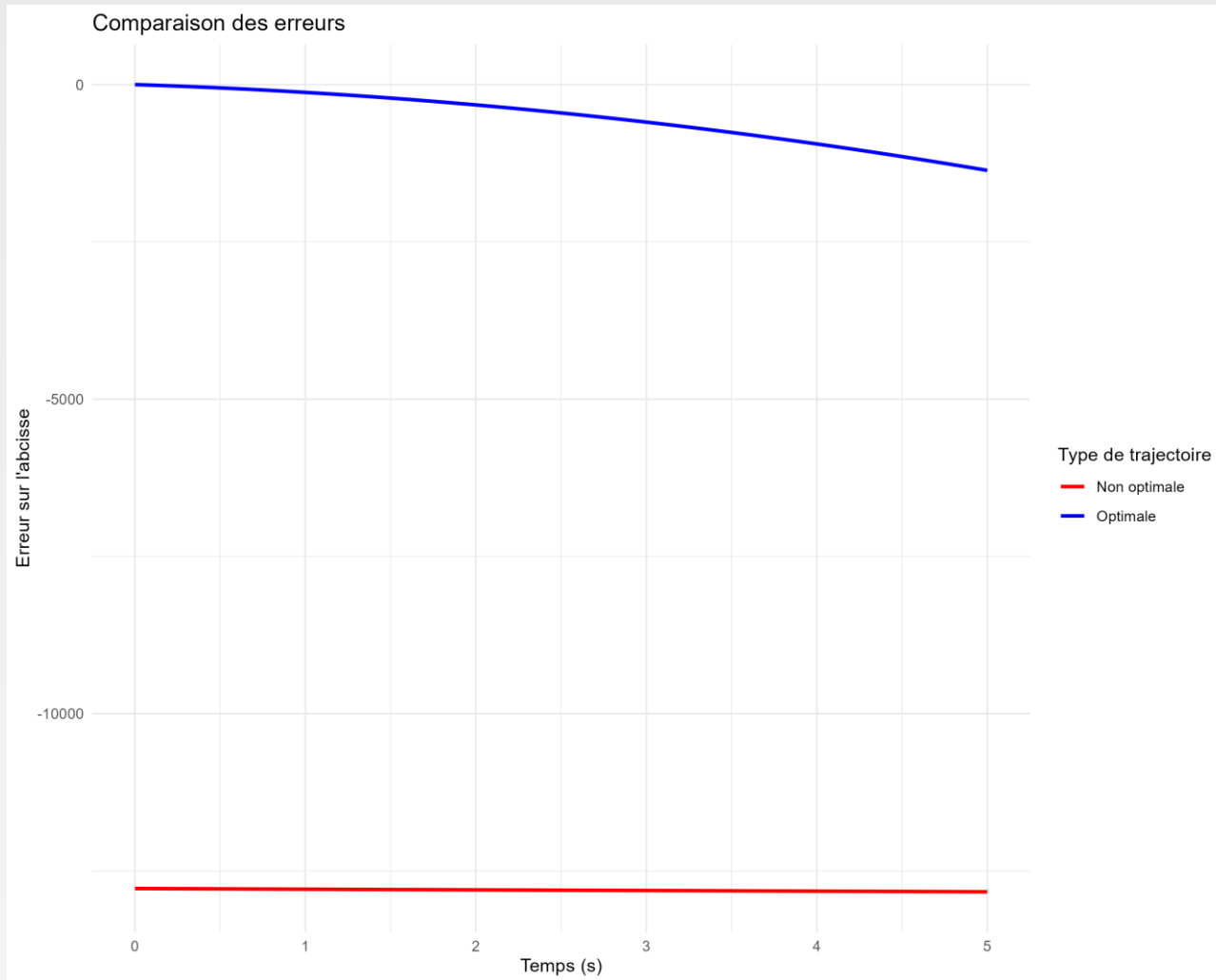
- Variation de la fonction coût < seuil
- Norme du gradient < seuil
- Nombre d'itérations maximal atteint

## 6 Solution

Les paramètres optimaux:

$$\theta_{\text{optimal}} = \theta_k \text{ à l'itération } k \text{ où le critère d'arrêt est vérifié}$$

# Trajectoires Optimisées



## ⇔ Comparaison

- ✗ Trajectoire initiale : déviation significative par rapport à la cible
- ✓ Trajectoire optimisée : atteint la cible avec précision

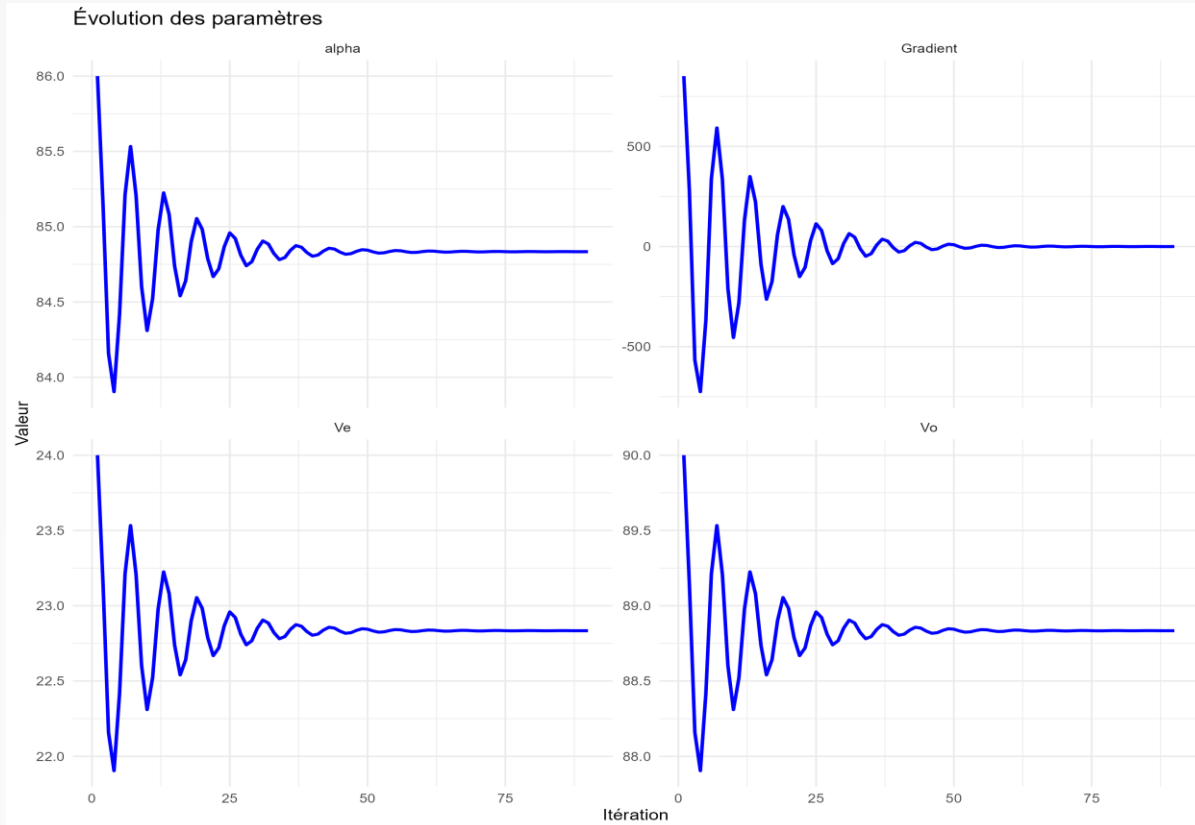
## 💡 Observations

Adjustement progressif des paramètres de tir

📈 Convergence vers la solution optimale



# Courbes de Convergence



## 🎯 Résultat de l'Optimisation

L'algorithme de la boule pesante atteint le minimum global avec une trajectoire plus directe, grâce à l'effet de momentum.

## 🏃 Trajectoires Comparées

La descente classique peut être piégée dans des creux locaux, tandis que la boule pesante franchit ces obstacles.

# Avantages et Limites



## Avantages

### Convergence Rapide

Accélère la recherche de l'optimum grâce à l'effet de momentum dans les régions plates.

### Évitement des Minima Locaux

Le momentum permet de franchir les "vallées" peu profondes et de progresser vers le minimum global.

### Robustesse

Efficace sur des surfaces d'optimisation non convexes et des fonctions coût complexes.



## Limites

### Sensibilité aux Hyperparamètres

Nécessite un réglage fin des paramètres  $\alpha$  (pas d'apprentissage) et  $\gamma$  (coefficient de momentum).

### Pas de Garantie de Convergence

Peut rester bloqué dans des minima locaux profonds, sans garantie de trouver le minimum global.

### Complexité Accrue




Implique la gestion d'une variable de vitesse supplémentaire par rapport aux méthodes de gradient simple.

# Conclusion et Perspectives

## ✓ Synthèse des Résultats

- Algorithme de la Boule Pesante : solution élégante et efficace pour des problèmes d'optimisation dynamique complexes
- Convergence rapide grâce à l'accumulation d'élan dans les régions plates
- Capacité avérée à éviter les minima locaux grâce au momentum
- Application pratique réussie à l'optimisation des paramètres de tir d'un projectile à masse variable

## 💡 Perspectives

-  Applications à d'autres systèmes  
Extension à la robotique, à l'aérospatiale ou à d'autres domaines d'ingénierie
-  Variantes stochastiques  
Exploration des algorithmes de Boule Pesante stochastiques pour des problèmes avec bruit ou de grandes bases de données
-  Optimisation des hyperparamètres  
Développement de méthodes adaptatives pour le réglage automatique de  $\alpha$  et  $\gamma$

# Questions et Discussion

👉 Merci de votre attention

Des questions ?



**Nom**

NKAMWA WAKIEU Randy



**Etablissement**

ENSPD



**Email**

viaatools@gmail.com



**Galerie**

[Cliquer pour voir la galerie](#)