

Journées APMEP — Toulon, 18-21/10/2025

Courbe du chien et problème des quatre scarabées

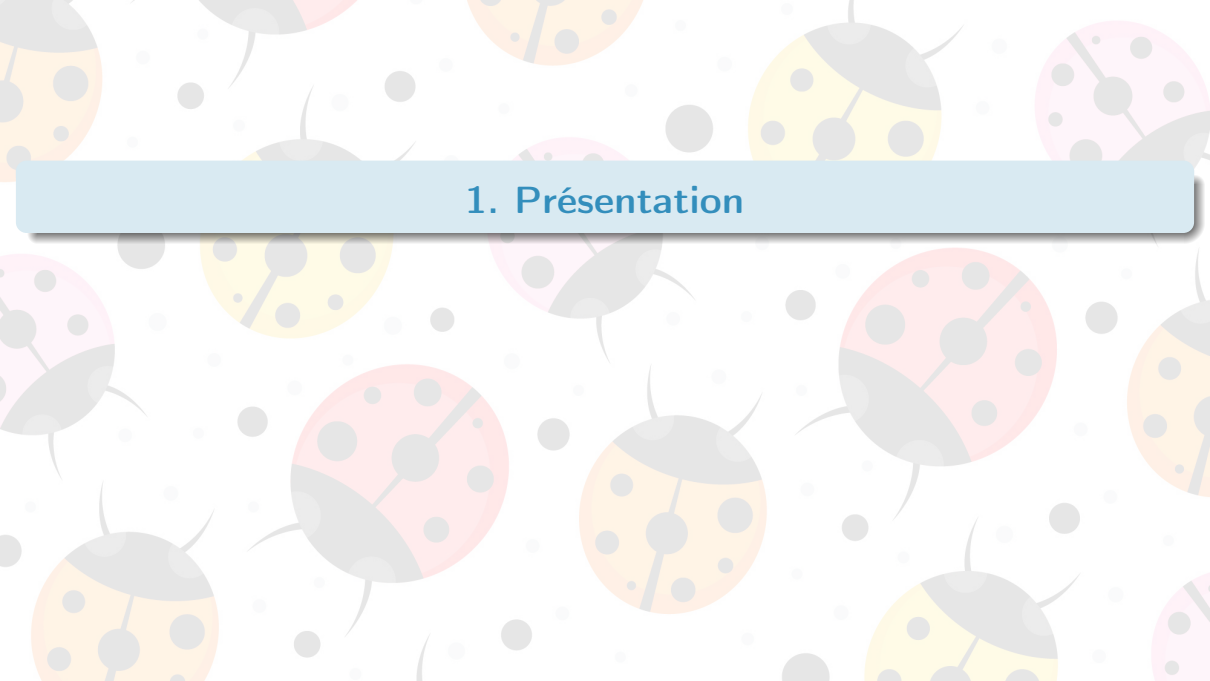
Exploration de trajectoires mêlant modélisations, calculs analytiques et simulations

Gloria Faccanoni

<https://faccanoni.univ-tln.fr/>
gloria.faccanoni@univ-tln.fr

Outline

1. Présentation
2. La courbe du chien
3. Le problème des 4 scarabées
4. Conclusion

The background of the slide is a repeating pattern of stylized ladybugs in orange, pink, and yellow, each with grey spots and antennae. Interspersed among the ladybugs are small grey dots of varying sizes. A light blue horizontal bar with a slight drop shadow is positioned across the middle of the slide.

1. Présentation

Qui suis-je ?

**Enseignante-Chercheuse
en Mathématiques**

Enseignement



Calcul scientifique

Licence, Master,
Écoles d'ingénieurs

Recherche



Modélisation,
étude mathématique et numérique,
simulations

d'écoulements dans des
réacteurs nucléaires

Démarche en calcul scientifique

1 Comprendre le problème physique

- Identifier les grandeurs clés (ex : vitesse, pression...)
- Recueillir les données (géométrie, conditions initiales...)

2 Construire et analyser le modèle mathématique

- Traduire les lois physiques en équations (conservation...)
- Identifier des propriétés particulières sans connaître la solution (invariants, énergie, monotonie...)

3 Approximations : modèle mathématique $\rightarrow$ modèle numérique

- Discretisation (espace/temps) $\rightarrow$ nombre fini d'inconnues $\rightarrow$ schéma numérique (relations discrètes)
- Analyse théorique des propriétés du schéma
- Implémentation $\rightarrow$ calcul sur des tests bien choisis

4 Évaluation de la qualité des approximations

- Vérifier la cohérence avec les propriétés de la solution exacte (cf. le point 2)
- Comparer à des solutions exactes dans des cas simplifiés
- Comparer aux observations réelles

Démarche en calcul scientifique

1 Comprendre le problème physique

- Identifier les grandeurs clés (ex : vitesse, pression...)
- Recueillir les données (géométrie, conditions initiales...)

2 Construire et analyser le modèle mathématique

- Traduire les lois physiques en équations (conservation...)
- Identifier des propriétés calculables sans connaître la solution (invariants, énergie, monotonie,...)

3 Approximation du modèle mathématique par un modèle numérique

- Discretisation (espace/temps) → nombre fini d'inconnues → schéma numérique (relations discrètes)
- Analyse théorique des propriétés du schéma
- Implémentation → calcul sur des tests bien choisis

4 Évaluation de la qualité des approximations

- Vérifier la cohérence avec les propriétés de la solution exacte (cf. le point 2)
- Comparer à des solutions exactes dans des cas simplifiés
- Comparer aux observations réelles

Démarche en calcul scientifique

1 Comprendre le problème physique

- Identifier les grandeurs clés (ex : vitesse, pression...)
- Recueillir les données (géométrie, conditions initiales...)

2 Construire et analyser le modèle mathématique

- Traduire les lois physiques en équations (conservation...)
- Identifier des propriétés calculables sans connaître la solution (invariants, énergie, monotonie,...)

3 Approximations : Modèle mathématique → Modèle numérique

- Discrétisation (espace/temps) $\rightsquigarrow$ nombre fini d'inconnues $\rightsquigarrow$ schéma numérique (relations discrètes)
- Analyse théorique des propriétés du schéma
- Implémentation → calcul sur des tests bien choisis

4 Évaluation de la qualité des approximations

- Vérifier la cohérence avec les propriétés de la solution exacte (cf. le point 2)
- Comparer à des solutions exactes dans des cas simplifiés
- Comparer aux observations réelles

Démarche en calcul scientifique

❶ Comprendre le problème physique

- Identifier les grandeurs clés (ex : vitesse, pression...)
- Recueillir les données (géométrie, conditions initiales...)

❷ Construire et analyser le modèle mathématique

- Traduire les lois physiques en équations (conservation...)
- Identifier des propriétés calculables sans connaître la solution (invariants, énergie, monotonie,...)

❸ Approximations : Modèle mathématique → Modèle numérique

- Discrétisation (espace/temps) $\rightsquigarrow$ nombre fini d'inconnues $\rightsquigarrow$ schéma numérique (relations discrètes)
- Analyse théorique des propriétés du schéma
- Implémentation → calcul sur des tests bien choisis

❹ Évaluation de la qualité des approximations

- Vérifier la cohérence avec les propriétés de la solution exacte (cf. le point 2)
- Comparer à des solutions exactes dans des **cas simplifiés**
- Comparer aux observations réelles

Réacteurs $\rightsquigarrow$ animaux...

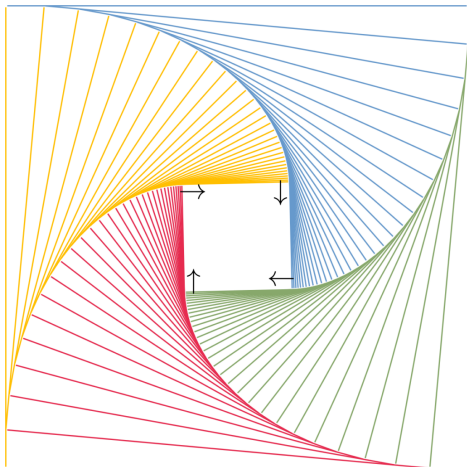


Image de Peter Rowlett réalisée en TikZ/LaTeX dans
<https://aperiodical.com/2023/10/2-four-bugs/>

Couverture de *Scientific American* (juillet 1965)
illustrant un problème de
poursuite
lié aux énigmes de
Gardner dans
*Mathematical Puzzles
and Diversions*.

Réacteurs $\rightsquigarrow$ animaux...

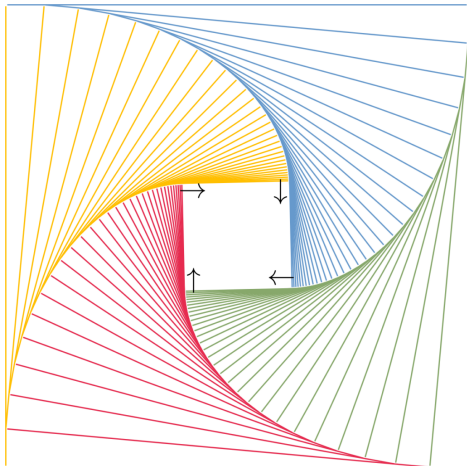
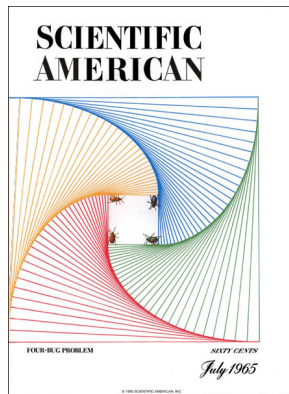


Image de Peter Rowlett réalisée en TikZ/LaTeX dans
<https://aperiodical.com/2023/10/2-four-bugs/>

Couverture de *Scientific American* (juillet 1965)
 illustrant un problème de
 poursuite
 lié aux énigmes de
 Gardner dans
*Mathematical Puzzles
 and Diversions*.



The Amorous Bugs

16. The Amorous Bugs

FOUR BUGS—A, B, C and D—occupy the corners of a square 10 inches on a side. A and C are male, B and D are female. Simultaneously A crawls directly toward B, B toward C, C toward D and D toward A. If all four bugs crawl at the same constant rate, they will describe four congruent logarithmic spirals which meet at the center of the square.

How far does each bug travel before they meet? The problem can be solved without calculus.



Four bugs — A, B, C, and D — occupy the corners of a square 10 inches along a side. A and C are male, B and C are female.

Simultaneously A crawls directly towards B, B towards C, C towards D and D towards A. If all four bugs crawl at the same constant rate, they will describe four congruent logarithmic spirals which meet at the centre of the square.

How far does each beetle travel before they meet? The problem can be solved without calculus.

The Amorous Bugs

16. The Amorous Bugs

FOUR BUGS—A, B, C and D—occupy the corners of a square 10 inches on a side. A and C are male, B and D are female. Simultaneously A crawls directly toward B, B toward C, C toward D and D toward A. If all four bugs crawl at the same constant rate, they will describe four congruent logarithmic spirals which meet at the center of the square.

How far does each bug travel before they meet? The problem can be solved without calculus.



Four bugs — A, B, C, and D — occupy the corners of a square 10 inches along a side. A and C are male, B and C are female.

Simultaneously A crawls directly towards B, B towards C, C towards D and D towards A. If all four bugs crawl at the same constant rate, they will describe four congruent **logarithmic spirals** which meet at the centre of the square.

How far does each beetle travel before they meet? The problem can be solved without calculus.

Illustration de la démarche : Courbes de poursuite

① Poursuite classique



Chien → Chat / Maître
Prédateur → Proie

② Poursuite mutuelle

4 insectes aux sommets d'un carré se
poursuivent mutuellement

Comment **modéliser**, **approcher** et/ou **calculer analytiquement** ces trajectoires ?

Pour faciliter l'exposé, modélisation en deux dimensions, mais généralisation facile à des dimensions supérieures.

Illustration de la démarche : Courbes de poursuite

① Poursuite classique



Chien → Chat / Maître
Prédateur → Proie

② Poursuite mutuelle



4 insectes aux sommets d'un carré se
poursuivent mutuellement

Comment **modéliser**, **approcher** et/ou **calculer analytiquement** ces trajectoires ?

Pour faciliter l'exposé, modélisation en deux dimensions, mais généralisation facile à des dimensions supérieures.

Illustration de la démarche : Courbes de poursuite

① Poursuite classique



Chien → Chat / Maître
Prédateur → Proie

② Poursuite mutuelle



4 insectes aux sommets d'un carré se
poursuivent mutuellement

Comment **modéliser**, **approcher** et/ou **calculer analytiquement** ces trajectoires ?

Pour faciliter l'exposé, modélisation en deux dimensions, mais généralisation facile à des dimensions supérieures.

Illustration de la démarche : Courbes de poursuite

① Poursuite classique



Chien → Chat / Maître
Prédateur → Proie

② Poursuite mutuelle



4 insectes aux sommets d'un carré se
poursuivent mutuellement

Comment **modéliser**, **approcher** et/ou **calculer analytiquement** ces trajectoires ?

Pour faciliter l'exposé, modélisation en deux dimensions, mais généralisation facile à des dimensions supérieures.

2. La courbe du chien

- 2.1 Position du problème
- 2.2 Mise en équation et approximations
- 2.3 Invariant
- 2.4 Solution exacte

Introduction

- **Courbe du chien** : trajectoire d'un poursuivant qui se dirige en permanence vers une cible en mouvement (chat $\rightarrow$ souris, chien $\rightarrow$ lièvre/maître, missile $\rightarrow$ avion...).
- Le poursuivant n'a accès **qu'à la position actuelle** de sa cible : il ne prédit ni sa trajectoire ni sa vitesse.

Problème

- Quelle est la trajectoire du poursuivant ?
- Si les deux trajectoires se croisent : **où** et **quand** ?



Position du problème

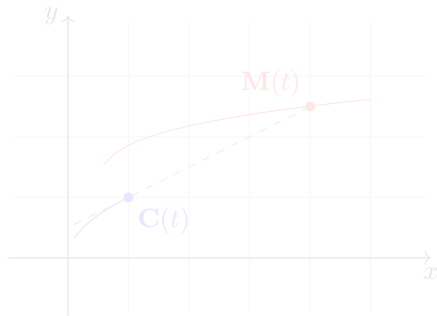
À l'instant t :

- position du maître (connue) :

$$\mathbf{M}(t) = (x_m(t), y_m(t))$$

- position du chien (inconnue) :

$$\mathbf{C}(t) = (x(t), y(t))$$



2 hypothèses

- Maître : **vitesse constante** v , trajectoire donnée.
- Chien : **vitesse constante** kv (avec $k > 0$), en **direction** de la position instantanée du maître.

Position du problème

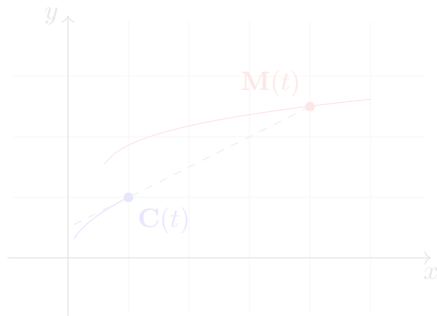
À l'instant t :

- position du maître (connue) :

$$\mathbf{M}(t) = (x_m(t), y_m(t))$$

- position du chien (inconnue) :

$$\mathbf{C}(t) = (x(t), y(t))$$



2 hypothèses

- Maître : **vitesse constante** v , trajectoire donnée.
- Chien : **vitesse constante** kv (avec $k > 0$), en **direction** de la position instantanée du maître.

Position du problème

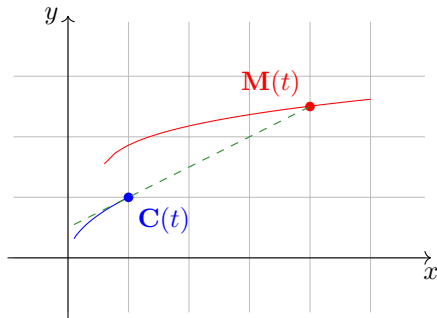
À l'instant t :

- position du maître (connue) :

$$\mathbf{M}(t) = (x_m(t), y_m(t))$$

- position du chien (inconnue) :

$$\mathbf{C}(t) = (x(t), y(t))$$



2 hypothèses

- Maître : **vitesse constante** v , trajectoire donnée.
- Chien : **vitesse constante** kv (avec $k > 0$), en **direction** de la position instantanée du maître.

Programmer avant de mathématiser

Soit Δt un intervalle de temps (fixé arbitrairement) et notons

- $\mathbf{M}_n = \mathbf{M}(t_n)$ la position (exacte) du maître à l'instant $t_n = n\Delta t$,
- $\mathbf{C}_n \approx \mathbf{C}(t_n)$ la position (approchée) du chien au même instant.

Principe d'un programme naïf

À l'instant t_n , le chien avance en ligne **droite** vers $\mathbf{M}_n$ à vitesse kv pendant Δt
 $\rightsquigarrow$ la trajectoire du chien est approchée par une courbe affine par morceaux (spline linéaire)

Récurrance :

$$\begin{cases} \mathbf{C}_0 \text{ position initiale connue} \\ \mathbf{C}_{n+1} = \mathbf{C}_n + \Delta t \cdot kv \cdot \frac{\mathbf{M}_n - \mathbf{C}_n}{\|\mathbf{M}_n - \mathbf{C}_n\|} \end{cases}$$

Arrêt si $\|\mathbf{M}_n - \mathbf{C}_n\| < \varepsilon$ (tolérance).

[Voir la simulation dans l'annexe](#)

Programmer avant de mathématiser

Soit Δt un intervalle de temps (fixé arbitrairement) et notons

- $\mathbf{M}_n = \mathbf{M}(t_n)$ la position (exacte) du maître à l'instant $t_n = n\Delta t$,
- $\mathbf{C}_n \approx \mathbf{C}(t_n)$ la position (approchée) du chien au même instant.

Principe d'un programme naïf

À l'instant t_n , le chien avance en ligne **droite** vers $\mathbf{M}_n$ à vitesse kv pendant Δt
 $\rightsquigarrow$ la trajectoire du chien est approchée par une courbe affine par morceaux (spline linéaire)

Récurrance :

$$\begin{cases} \mathbf{C}_0 \text{ position initiale connue} \\ \mathbf{C}_{n+1} = \mathbf{C}_n + \Delta t \cdot kv \cdot \frac{\mathbf{M}_n - \mathbf{C}_n}{\|\mathbf{M}_n - \mathbf{C}_n\|} \end{cases}$$

Arrêt si $\|\mathbf{M}_n - \mathbf{C}_n\| < \varepsilon$ (tolérance).

[Voir la simulation dans l'annexe](#)

Programmer avant de mathématiser

Soit Δt un intervalle de temps (fixé arbitrairement) et notons

- $\mathbf{M}_n = \mathbf{M}(t_n)$ la position (exacte) du maître à l'instant $t_n = n\Delta t$,
- $\mathbf{C}_n \approx \mathbf{C}(t_n)$ la position (approchée) du chien au même instant.

Principe d'un programme naïf

À l'instant t_n , le chien avance en ligne **droite** vers $\mathbf{M}_n$ à vitesse kv pendant Δt
 $\rightsquigarrow$ la trajectoire du chien est approchée par une courbe affine par morceaux (spline linéaire)

Récurrence :

$$\begin{cases} \mathbf{C}_0 \text{ position initiale connue} \\ \mathbf{C}_{n+1} = \mathbf{C}_n + \Delta t \cdot kv \cdot \frac{\mathbf{M}_n - \mathbf{C}_n}{\|\mathbf{M}_n - \mathbf{C}_n\|} \end{cases}$$

Arrêt si $\|\mathbf{M}_n - \mathbf{C}_n\| < \varepsilon$ (tolérance).

[Voir la simulation dans l'annexe](#)

Programmer avant de mathématiser

Soit Δt un intervalle de temps (fixé arbitrairement) et notons

- $\mathbf{M}_n = \mathbf{M}(t_n)$ la position (exacte) du maître à l'instant $t_n = n\Delta t$,
- $\mathbf{C}_n \approx \mathbf{C}(t_n)$ la position (approchée) du chien au même instant.

Principe d'un programme naïf

À l'instant t_n , le chien avance en ligne **droite** vers $\mathbf{M}_n$ à vitesse kv pendant Δt
 $\rightsquigarrow$ la trajectoire du chien est approchée par une courbe affine par morceaux (spline linéaire)

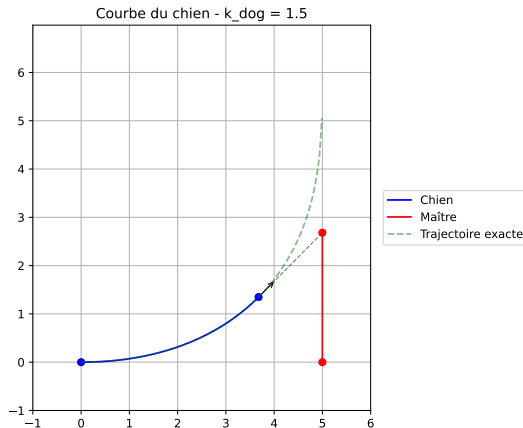
Récurrance :

$$\begin{cases} \mathbf{C}_0 \text{ position initiale connue} \\ \mathbf{C}_{n+1} = \mathbf{C}_n + \Delta t \cdot kv \cdot \frac{\mathbf{M}_n - \mathbf{C}_n}{\|\mathbf{M}_n - \mathbf{C}_n\|} \end{cases}$$

Arrêt si $\|\mathbf{M}_n - \mathbf{C}_n\| < \varepsilon$ (tolérance).

[Voir la simulation dans l'annexe](#)

Exemples de trajectoires



Constat

- La trajectoire du chien dépend de :
 - k (la vitesse relative),
 - la trajectoire du maître.
- Le chien peut ne jamais rejoindre le maître.

Mise en équation

La trajectoire du chien satisfait

un système de 2 EDO d'ordre 1 :

$$\begin{cases} \mathbf{C}(0) \text{ connue} \\ \mathbf{C}'(t) = kv \frac{\mathbf{M}(t) - \mathbf{C}(t)}{\|\mathbf{M}(t) - \mathbf{C}(t)\|} \end{cases}$$

$$\rightsquigarrow \begin{cases} x(0), y(0) \text{ connues} \\ x'(t) = \varphi_1(t, x(t), y(t)) \stackrel{\text{def}}{=} kv \frac{x_m(t) - x(t)}{\sqrt{(x_m(t) - x(t))^2 + (y_m(t) - y(t))^2}} \\ y'(t) = \varphi_2(t, x(t), y(t)) \stackrel{\text{def}}{=} kv \frac{y_m(t) - y(t)}{\sqrt{(x_m(t) - x(t))^2 + (y_m(t) - y(t))^2}} \end{cases}$$

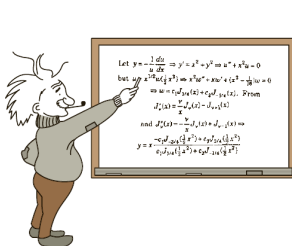
Mise en équation

La trajectoire du chien satisfait

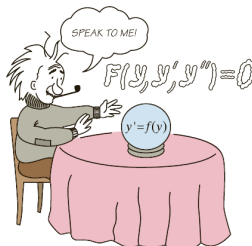
un système de 2 EDO d'ordre 1 :

$$\begin{cases} \mathbf{C}(0) \text{ connue} \\ \mathbf{C}'(t) = kv \frac{\mathbf{M}(t) - \mathbf{C}(t)}{\|\mathbf{M}(t) - \mathbf{C}(t)\|} \end{cases} \rightsquigarrow \begin{cases} x(0), y(0) \text{ connues} \\ x'(t) = \varphi_1(t, x(t), y(t)) \stackrel{\text{def}}{=} kv \frac{x_m(t) - x(t)}{\sqrt{(x_m(t) - x(t))^2 + (y_m(t) - y(t))^2}} \\ y'(t) = \varphi_2(t, x(t), y(t)) \stackrel{\text{def}}{=} kv \frac{y_m(t) - y(t)}{\sqrt{(x_m(t) - x(t))^2 + (y_m(t) - y(t))^2}} \end{cases}$$

Solution ?



(a) analytical



(b) qualitative



(c) numerical

Pour $M(t)$ quelconque :

- Solution analytique ? **Non** (sauf cas particulier)
- Propriétés de la solution sans trouver sa forme exacte ? **Oui**
- Solution approchée ? **Oui** (schémas numériques)

C'est quoi un schéma numérique pour des EDO ?

On cherche $\mathbf{y} : [t_0, T] \rightarrow \mathbb{R}^2$ tel que :

$$\begin{cases} \mathbf{y}'(t) = \varphi(t, \mathbf{y}(t)), & t \in [t_0, T] \\ \mathbf{y}(t_0) = \mathbf{y}_0, & \text{(valeur initiale)} \end{cases}$$

Hypothèse : $\exists!$ solution

Dans notre exemple :

$$\mathbf{y}(t) = \mathbf{C}(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}$$

$$\varphi(t, \mathbf{C}) = \begin{pmatrix} \varphi_1(t, x(t), y(t)) \\ \varphi_2(t, x(t), y(t)) \end{pmatrix}$$

Principe :

- discrétiser le temps : $t_n = t_0 + n\Delta t$
- pour chaque nœud t_n , calculer la valeur $\mathbf{u}_n \approx \mathbf{y}_n = \mathbf{y}(t_n)$

C'est quoi un schéma numérique pour des EDO ?

On cherche $\mathbf{y}: [t_0, T] \rightarrow \mathbb{R}^2$ tel que :

$$\begin{cases} \mathbf{y}'(t) = \varphi(t, \mathbf{y}(t)), & t \in [t_0, T] \\ \mathbf{y}(t_0) = \mathbf{y}_0, & \text{(valeur initiale)} \end{cases}$$

Hypothèse : $\exists!$ solution

Dans notre exemple :

$$\mathbf{y}(t) = \mathbf{C}(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}$$

$$\varphi(t, \mathbf{C}) = \begin{pmatrix} \varphi_1(t, x(t), y(t)) \\ \varphi_2(t, x(t), y(t)) \end{pmatrix}$$

Principe :

- discrétiser le temps : $t_n = t_0 + n\Delta t$
- pour chaque nœud t_n , calculer la valeur $\mathbf{u}_n \approx \mathbf{y}_n = \mathbf{y}(t_n)$

C'est quoi un schéma numérique pour des EDO ?

On cherche $\mathbf{y}: [t_0, T] \rightarrow \mathbb{R}^2$ tel que :

$$\begin{cases} \mathbf{y}'(t) = \varphi(t, \mathbf{y}(t)), & t \in [t_0, T] \\ \mathbf{y}(t_0) = \mathbf{y}_0, & \text{(valeur initiale)} \end{cases}$$

Hypothèse : $\exists!$ solution

Dans notre exemple :

$$\mathbf{y}(t) = \mathbf{C}(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}$$

$$\varphi(t, \mathbf{C}) = \begin{pmatrix} \varphi_1(t, x(t), y(t)) \\ \varphi_2(t, x(t), y(t)) \end{pmatrix}$$

Principe :

- discrétiser le temps : $t_n = t_0 + n\Delta t$
- pour chaque nœud t_n , calculer la valeur $\mathbf{u}_n \approx \mathbf{y}_n = \mathbf{y}(t_n)$

C'est quoi un schéma numérique pour des EDO ?

On cherche $\mathbf{y}: [t_0, T] \rightarrow \mathbb{R}^2$ tel que :

$$\begin{cases} \mathbf{y}'(t) = \varphi(t, \mathbf{y}(t)), & t \in [t_0, T] \\ \mathbf{y}(t_0) = \mathbf{y}_0, & \text{(valeur initiale)} \end{cases}$$

Hypothèse : $\exists!$ solution

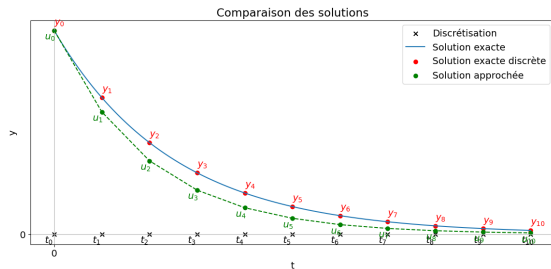
Dans notre exemple :

$$\mathbf{y}(t) = \mathbf{C}(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}$$

$$\varphi(t, \mathbf{C}) = \begin{pmatrix} \varphi_1(t, x(t), y(t)) \\ \varphi_2(t, x(t), y(t)) \end{pmatrix}$$

Principe :

- discrétiser le temps : $t_n = t_0 + n\Delta t$
- pour chaque nœud t_n , calculer la valeur $\mathbf{u}_n \approx \mathbf{y}_n = \mathbf{y}(t_n)$



Construction d'un schéma

Souvent $\{u_n\}$ est obtenue par une récurrence par intégration de l'EDO :

$$y(t_{n+1}) - y(t_n) = \int_{t_n}^{t_n+\Delta t} y'(t) dt = \int_{t_n}^{t_n+\Delta t} \varphi(t, y(t)) dt \approx \sum_{j \in J} w_j \varphi(t_j, y(t_j))$$

$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \sum_{j \in J} w_j \varphi(t_j, \mathbf{u}_j) \end{cases}$$

Exemple : schéma d'**Euler Explicite**



$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t \varphi(t_n, \mathbf{u}_n) \end{cases}$$

C'est le "Programme naïf" !

Construction d'un schéma

Souvent $\{u_n\}$ est obtenue par une récurrence par intégration de l'EDO :

$$y(t_{n+1}) - y(t_n) = \int_{t_n}^{t_n+\Delta t} y'(t) dt = \int_{t_n}^{t_n+\Delta t} \varphi(t, y(t)) dt \approx \sum_{j \in J} w_j \varphi(t_j, y(t_j))$$

$$\rightsquigarrow \begin{cases} u_0 = y_0 \\ u_{n+1} = u_n + \sum_{j \in J} w_j \varphi(t_j, u_j) \end{cases}$$

Exemple : schéma d'Euler Explicite



$$\rightsquigarrow \begin{cases} u_0 = y_0 \\ u_{n+1} = u_n + \Delta t \varphi(t_n, u_n) \end{cases}$$

C'est le "Programme naïf" !

Construction d'un schéma

Souvent $\{u_n\}$ est obtenue par une récurrence par intégration de l'EDO :

$$\mathbf{y}(t_{n+1}) - \mathbf{y}(t_n) = \int_{t_n}^{t_n+\Delta t} \mathbf{y}'(t) dt = \int_{t_n}^{t_n+\Delta t} \varphi(t, \mathbf{y}(t)) dt \approx \sum_{j \in J} w_j \varphi(t_j, \mathbf{y}(t_j))$$

$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \sum_{j \in J} w_j \varphi(t_j, \mathbf{u}_j) \end{cases}$$

Exemple : schéma d'Euler Explicite



$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t \varphi(t_n, \mathbf{u}_n) \end{cases}$$

C'est le "Programme naïf" !

Construction d'un schéma

Souvent $\{u_n\}$ est obtenue par une récurrence par intégration de l'EDO :

$$\mathbf{y}(t_{n+1}) - \mathbf{y}(t_n) = \int_{t_n}^{t_n+\Delta t} \mathbf{y}'(t) dt = \int_{t_n}^{t_n+\Delta t} \varphi(t, \mathbf{y}(t)) dt \approx \sum_{j \in J} w_j \varphi(t_j, \mathbf{y}(t_j))$$

$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \sum_{j \in J} w_j \varphi(t_j, \mathbf{u}_j) \end{cases}$$

Exemple : schéma d'Euler Explicite



$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t \varphi(t_n, \mathbf{u}_n) \end{cases}$$

C'est le "Programme naïf" !

Construction d'un schéma

Souvent $\{u_n\}$ est obtenue par une récurrence par intégration de l'EDO :

$$\mathbf{y}(t_{n+1}) - \mathbf{y}(t_n) = \int_{t_n}^{t_n+\Delta t} \mathbf{y}'(t) dt = \int_{t_n}^{t_n+\Delta t} \boldsymbol{\varphi}(t, \mathbf{y}(t)) dt \approx \sum_{j \in J} w_j \boldsymbol{\varphi}(t_j, \mathbf{y}(t_j))$$

$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \sum_{j \in J} w_j \boldsymbol{\varphi}(t_j, \mathbf{u}_j) \end{cases}$$

Exemple : schéma d'Euler Explicite



$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t \boldsymbol{\varphi}(t_n, \mathbf{u}_n) \end{cases}$$

C'est le "Programme naïf" !

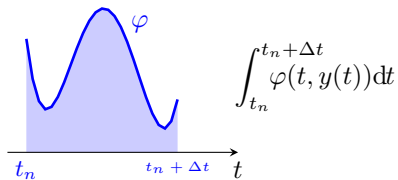
Construction d'un schéma

Souvent $\{u_n\}$ est obtenue par une récurrence par intégration de l'EDO :

$$\mathbf{y}(t_{n+1}) - \mathbf{y}(t_n) = \int_{t_n}^{t_n+\Delta t} \mathbf{y}'(t) dt = \int_{t_n}^{t_n+\Delta t} \boldsymbol{\varphi}(t, \mathbf{y}(t)) dt \approx \sum_{j \in J} w_j \boldsymbol{\varphi}(t_j, \mathbf{y}(t_j))$$

$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \sum_{j \in J} w_j \boldsymbol{\varphi}(t_j, \mathbf{u}_j) \end{cases}$$

Exemple : schéma d'**Euler Explicite**



$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t \boldsymbol{\varphi}(t_n, \mathbf{u}_n) \end{cases}$$

C'est le "Programme naïf" !

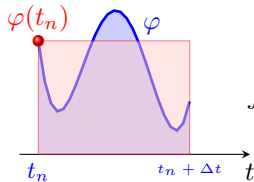
Construction d'un schéma

Souvent $\{u_n\}$ est obtenue par une récurrence par intégration de l'EDO :

$$\mathbf{y}(t_{n+1}) - \mathbf{y}(t_n) = \int_{t_n}^{t_n+\Delta t} \mathbf{y}'(t) dt = \int_{t_n}^{t_n+\Delta t} \boldsymbol{\varphi}(t, \mathbf{y}(t)) dt \approx \sum_{j \in J} w_j \boldsymbol{\varphi}(t_j, \mathbf{y}(t_j))$$

$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \sum_{j \in J} w_j \boldsymbol{\varphi}(t_j, \mathbf{u}_j) \end{cases}$$

Exemple : schéma d'**Euler Explicite**



$$\int_{t_n}^{t_n+\Delta t} \boldsymbol{\varphi}(t, \mathbf{y}(t)) dt \approx \Delta t \boldsymbol{\varphi}(t, \mathbf{y}(t_n))$$

$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t \boldsymbol{\varphi}(t_n, \mathbf{u}_n) \end{cases}$$

C'est le "Programme naïf" !

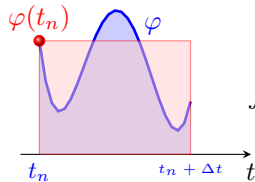
Construction d'un schéma

Souvent $\{u_n\}$ est obtenue par une récurrence par intégration de l'EDO :

$$\mathbf{y}(t_{n+1}) - \mathbf{y}(t_n) = \int_{t_n}^{t_n+\Delta t} \mathbf{y}'(t) dt = \int_{t_n}^{t_n+\Delta t} \boldsymbol{\varphi}(t, \mathbf{y}(t)) dt \approx \sum_{j \in J} w_j \boldsymbol{\varphi}(t_j, \mathbf{y}(t_j))$$

$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \sum_{j \in J} w_j \boldsymbol{\varphi}(t_j, \mathbf{u}_j) \end{cases}$$

Exemple : schéma d'**Euler Explicite**



$$\int_{t_n}^{t_n+\Delta t} \boldsymbol{\varphi}(t, \mathbf{y}(t)) dt \approx \Delta t \boldsymbol{\varphi}(t, \mathbf{y}(t_n))$$

$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t \boldsymbol{\varphi}(t_n, \mathbf{u}_n) \end{cases}$$

C'est le "Programme naïf" !

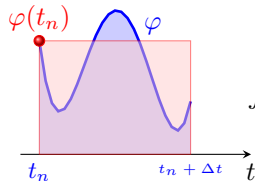
Construction d'un schéma

Souvent $\{u_n\}$ est obtenue par une récurrence par intégration de l'EDO :

$$\mathbf{y}(t_{n+1}) - \mathbf{y}(t_n) = \int_{t_n}^{t_n+\Delta t} \mathbf{y}'(t) dt = \int_{t_n}^{t_n+\Delta t} \boldsymbol{\varphi}(t, \mathbf{y}(t)) dt \approx \sum_{j \in J} w_j \boldsymbol{\varphi}(t_j, \mathbf{y}(t_j))$$

$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \sum_{j \in J} w_j \boldsymbol{\varphi}(t_j, \mathbf{u}_j) \end{cases}$$

Exemple : schéma d'**Euler Explicite**



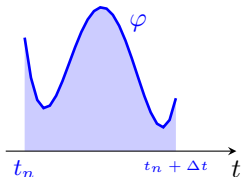
$$\int_{t_n}^{t_n+\Delta t} \boldsymbol{\varphi}(t, \mathbf{y}(t)) dt \approx \Delta t \boldsymbol{\varphi}(t, \mathbf{y}(t_n))$$

$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t \boldsymbol{\varphi}(t_n, \mathbf{u}_n) \end{cases}$$

C'est le "Programme naïf" !

D'autres schémas populaires

- Schéma de **Crank-Nicolson** (ou du trapèze)



- Schéma de **Heun**

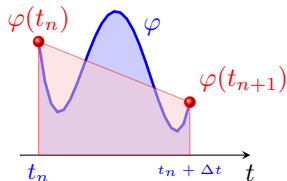
$$\begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \tilde{\mathbf{u}}_{n+1} = \mathbf{u}_n + \Delta t \varphi(t_n, \mathbf{u}_n) & [\text{Predictor}] \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \frac{\Delta t}{2} (\varphi(t_n, \mathbf{u}_n) + \varphi(t_{n+1}, \tilde{\mathbf{u}}_{n+1})) & [\text{Corrector}] \end{cases}$$

Schéma **explicite**

Cf. <https://moodle.univ-tln.fr/course/view.php?id=4840>

D'autres schémas populaires

- Schéma de **Crank-Nicolson** (ou du trapèze)



$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \frac{\Delta t}{2} (\varphi(t_n, \mathbf{u}_n) + \varphi(t_{n+1}, \mathbf{u}_{n+1})) \end{cases}$$

- Schéma de **Heun**

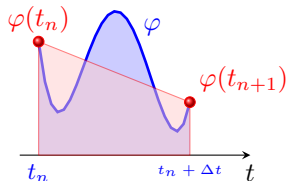
$$\begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \tilde{\mathbf{u}}_{n+1} = \mathbf{u}_n + \Delta t \varphi(t_n, \mathbf{u}_n) & \text{[Predictor]} \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \frac{\Delta t}{2} (\varphi(t_n, \mathbf{u}_n) + \varphi(t_{n+1}, \tilde{\mathbf{u}}_{n+1})) & \text{[Corrector]} \end{cases}$$

Schéma **explicite**

Cf. <https://moodle.univ-tln.fr/course/view.php?id=4840>

D'autres schémas populaires

- Schéma de **Crank-Nicolson** (ou du trapèze)



$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \frac{\Delta t}{2} (\varphi(t_n, \mathbf{u}_n) + \varphi(t_{n+1}, \mathbf{u}_{n+1})) \end{cases}$$

Schéma implicite

- Schéma de **Heun**

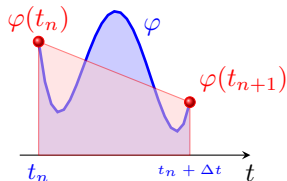
$$\begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \tilde{\mathbf{u}}_{n+1} = \mathbf{u}_n + \Delta t \varphi(t_n, \mathbf{u}_n) & [\text{Predictor}] \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \frac{\Delta t}{2} (\varphi(t_n, \mathbf{u}_n) + \varphi(t_{n+1}, \tilde{\mathbf{u}}_{n+1})) & [\text{Corrector}] \end{cases}$$

Schéma explicite

Cf. <https://moodle.univ-tln.fr/course/view.php?id=4840>

D'autres schémas populaires

- Schéma de **Crank-Nicolson** (ou du trapèze)



$$\rightsquigarrow \begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \frac{\Delta t}{2} (\varphi(t_n, \mathbf{u}_n) + \varphi(t_{n+1}, \mathbf{u}_{n+1})) \end{cases}$$

Schéma implicite

- Schéma de **Heun**

$$\begin{cases} \mathbf{u}_0 = \mathbf{y}_0 \\ \tilde{\mathbf{u}}_{n+1} = \mathbf{u}_n + \Delta t \varphi(t_n, \mathbf{u}_n) & [\text{Predictor}] \\ \mathbf{u}_{n+1} = \mathbf{u}_n + \frac{\Delta t}{2} (\varphi(t_n, \mathbf{u}_n) + \varphi(t_{n+1}, \tilde{\mathbf{u}}_{n+1})) & [\text{Corrector}] \end{cases}$$

Schéma explicite

Cf. <https://moodle.univ-tln.fr/course/view.php?id=4840>

Analyse des schémas

Calcul numérique

- Permet d'étudier des solutions approchées lorsque les solutions analytiques sont inaccessibles
- Les résultats doivent être interprétés avec prudence

Questions clés (sans connaître la solution exacte) :

- le problème est-il (mathématiquement et) numériquement bien posé ?
- quel schéma choisir ?
- peut-on estimer/majorer l'erreur en fonction de Δt ? (ordre de convergence)
- peut-on choisir Δt quelconque ? (stabilité)
- peut-on déterminer certaines propriétés de la solution ?
- peut-on comparer avec la solution exacte dans un cas simple ?

Analyse des schémas

Calcul numérique

- Permet d'étudier des solutions approchées lorsque les solutions analytiques sont inaccessibles
- Les résultats doivent être interprétés avec prudence

Questions clés (**sans connaître la solution exacte**) :

- le problème est-il (mathématiquement et) numériquement bien posé ?
- quel schéma choisir ?
- peut-on estimer/majorer l'erreur en fonction de Δt ? (ordre de convergence)
- peut-on choisir Δt quelconque ? (stabilité)
- peut-on déterminer certaines propriétés de la solution ?
- peut-on comparer avec la solution exacte dans un cas simple ?

Analyse des schémas

Calcul numérique

- Permet d'étudier des solutions approchées lorsque les solutions analytiques sont inaccessibles
- Les résultats doivent être interprétés avec prudence

Questions clés (**sans connaître la solution exacte**) :

- le problème est-il (mathématiquement et) numériquement bien posé ?
- quel schéma choisir ?
- peut-on estimer/majorer l'erreur en fonction de Δt ? (ordre de convergence)
- peut-on choisir Δt quelconque ? (stabilité)
- peut-on déterminer certaines propriétés de la solution ?
- peut-on comparer avec la solution exacte dans un cas simple ?

Analyse des schémas

Calcul numérique

- Permet d'étudier des solutions approchées lorsque les solutions analytiques sont inaccessibles
- Les résultats doivent être interprétés avec prudence

Questions clés (**sans connaître la solution exacte**) :

- le problème est-il (mathématiquement et) numériquement bien posé ?
- quel schéma choisir ?
- peut-on estimer/majorer l'erreur en fonction de Δt ? (ordre de convergence)
- peut-on choisir Δt quelconque ? (stabilité)
- peut-on déterminer certaines propriétés de la solution ?
- peut-on comparer avec la solution exacte dans un cas simple ?

Analyse des schémas

Calcul numérique

- Permet d'étudier des solutions approchées lorsque les solutions analytiques sont inaccessibles
- Les résultats doivent être interprétés avec prudence

Questions clés (**sans connaître la solution exacte**) :

- le problème est-il (mathématiquement et) numériquement bien posé ?
- quel schéma choisir ?
- peut-on estimer/majorer l'erreur en fonction de Δt ? (ordre de convergence)
- peut-on choisir Δt quelconque ? (stabilité)
- peut-on déterminer certaines propriétés de la solution ?
- peut-on comparer avec la solution exacte dans un cas simple ?

Analyse des schémas

Calcul numérique

- Permet d'étudier des solutions approchées lorsque les solutions analytiques sont inaccessibles
- Les résultats doivent être interprétés avec prudence

Questions clés (**sans connaître la solution exacte**) :

- le problème est-il (mathématiquement et) numériquement bien posé ?
- quel schéma choisir ?
- peut-on estimer/majorer l'erreur en fonction de Δt ? (ordre de convergence)
- peut-on choisir Δt quelconque ? (stabilité)
- peut-on déterminer certaines propriétés de la solution ?
- peut-on comparer avec la solution exacte dans un cas simple ?

Analyse des schémas

Calcul numérique

- Permet d'étudier des solutions approchées lorsque les solutions analytiques sont inaccessibles
- Les résultats doivent être interprétés avec prudence

Questions clés (**sans connaître la solution exacte**) :

- le problème est-il (mathématiquement et) numériquement bien posé ?
- quel schéma choisir ?
- peut-on estimer/majorer l'erreur en fonction de Δt ? (ordre de convergence)
- peut-on choisir Δt quelconque ? (stabilité)
- peut-on **déterminer certaines propriétés de la solution** ?
- peut-on **comparer avec la solution exacte dans un cas simple** ?

Détermination d'un invariant

- On cherche un **invariant**, i.e. $\mathcal{E}(t) = E(t, x(t), y(t))$ tel que $\mathcal{E}'(t) = 0$ pour tout t :

$$\mathcal{E}'(t) = \frac{\partial E}{\partial t} + \frac{\partial E}{\partial x} x'(t) + \frac{\partial E}{\partial y} y'(t) = \frac{\partial E}{\partial t} + \frac{\partial E}{\partial x} \varphi_1(t, x(t), y(t)) + \frac{\partial E}{\partial y} \varphi_2(t, x(t), y(t))$$

Contrainte : montrer que $\mathcal{E}'(t) \equiv 0$ sans utiliser l'expression explicite de $x(t)$ et $y(t)$.

- L'invariant permet de "mesurer" la qualité d'une approximation $(\tilde{x}_n, \tilde{y}_n)$ en calculant

$$\|E(t_n, \tilde{x}_n, \tilde{y}_n) - E(0, x_0, y_0)\|$$

- Cas particulier : si $(x_0, y_0) = (0, 0)$ et $M(t) = (a, vt)$, $a > 0$ constant, alors on peut choisir

$$E(t, x, y) = k \ln \left(\sqrt{(a-x)^2 + (vt-y)^2} + (vt-y) \right) - (k-1) \ln(a-x)$$

Remarque : Euler explicite ne conserve pas bien cet invariant

Détermination d'un invariant

- On cherche un **invariant**, i.e. $\mathcal{E}(t) = E(t, x(t), y(t))$ tel que $\mathcal{E}'(t) = 0$ pour tout t :

$$\mathcal{E}'(t) = \frac{\partial E}{\partial t} + \frac{\partial E}{\partial x} x'(t) + \frac{\partial E}{\partial y} y'(t) = \frac{\partial E}{\partial t} + \frac{\partial E}{\partial x} \varphi_1(t, x(t), y(t)) + \frac{\partial E}{\partial y} \varphi_2(t, x(t), y(t))$$

Contrainte : montrer que $\mathcal{E}'(t) \equiv 0$ **sans utiliser l'expression explicite de $x(t)$ et $y(t)$** .

- L'invariant permet de "mesurer" la qualité d'une approximation $(\tilde{x}_n, \tilde{y}_n)$ en calculant

$$\|E(t_n, \tilde{x}_n, \tilde{y}_n) - E(0, x_0, y_0)\|$$

- Cas particulier : si $(x_0, y_0) = (0, 0)$ et $M(t) = (a, vt)$, $a > 0$ constant, alors on peut choisir

$$E(t, x, y) = k \ln \left(\sqrt{(a-x)^2 + (vt-y)^2} + (vt-y) \right) - (k-1) \ln(a-x)$$

Remarque : Euler explicite ne conserve pas bien cet invariant

Détermination d'un invariant

- On cherche un **invariant**, i.e. $\mathcal{E}(t) = E(t, x(t), y(t))$ tel que $\mathcal{E}'(t) = 0$ pour tout t :

$$\mathcal{E}'(t) = \frac{\partial E}{\partial t} + \frac{\partial E}{\partial x} x'(t) + \frac{\partial E}{\partial y} y'(t) = \frac{\partial E}{\partial t} + \frac{\partial E}{\partial x} \varphi_1(t, x(t), y(t)) + \frac{\partial E}{\partial y} \varphi_2(t, x(t), y(t))$$

Contrainte : montrer que $\mathcal{E}'(t) \equiv 0$ **sans utiliser l'expression explicite de $x(t)$ et $y(t)$** .

- L'invariant permet de "mesurer" la qualité d'une approximation $(\tilde{x}_n, \tilde{y}_n)$ en calculant

$$\|E(t_n, \tilde{x}_n, \tilde{y}_n) - E(0, x_0, y_0)\|$$

- Cas particulier : si $(x_0, y_0) = (0, 0)$ et $M(t) = (a, vt)$, $a > 0$ constant, alors on peut choisir

$$E(t, x, y) = k \ln \left(\sqrt{(a-x)^2 + (vt-y)^2} + (vt-y) \right) - (k-1) \ln(a-x)$$

Remarque : Euler explicite ne conserve pas bien cet invariant

Détermination d'un invariant

- On cherche un **invariant**, i.e. $\mathcal{E}(t) = E(t, x(t), y(t))$ tel que $\mathcal{E}'(t) = 0$ pour tout t :

$$\mathcal{E}'(t) = \frac{\partial E}{\partial t} + \frac{\partial E}{\partial x} x'(t) + \frac{\partial E}{\partial y} y'(t) = \frac{\partial E}{\partial t} + \frac{\partial E}{\partial x} \varphi_1(t, x(t), y(t)) + \frac{\partial E}{\partial y} \varphi_2(t, x(t), y(t))$$

Contrainte : montrer que $\mathcal{E}'(t) \equiv 0$ **sans utiliser l'expression explicite de $x(t)$ et $y(t)$** .

- L'invariant permet de "mesurer" la qualité d'une approximation $(\tilde{x}_n, \tilde{y}_n)$ en calculant

$$\|E(t_n, \tilde{x}_n, \tilde{y}_n) - E(0, x_0, y_0)\|$$

- Cas particulier : si $(x_0, y_0) = (0, 0)$ et $M(t) = (a, vt)$, $a > 0$ constant, alors on peut choisir

$$E(t, x, y) = k \ln \left(\sqrt{(a-x)^2 + (vt-y)^2} + (vt-y) \right) - (k-1) \ln(a-x)$$

Remarque : Euler explicite ne conserve pas bien cet invariant

Détermination d'un invariant

- On cherche un **invariant**, i.e. $\mathcal{E}(t) = E(t, x(t), y(t))$ tel que $\mathcal{E}'(t) = 0$ pour tout t :

$$\mathcal{E}'(t) = \frac{\partial E}{\partial t} + \frac{\partial E}{\partial x} x'(t) + \frac{\partial E}{\partial y} y'(t) = \frac{\partial E}{\partial t} + \frac{\partial E}{\partial x} \varphi_1(t, x(t), y(t)) + \frac{\partial E}{\partial y} \varphi_2(t, x(t), y(t))$$

Contrainte : montrer que $\mathcal{E}'(t) \equiv 0$ **sans utiliser l'expression explicite de $x(t)$ et $y(t)$** .

- L'invariant permet de "mesurer" la qualité d'une approximation $(\tilde{x}_n, \tilde{y}_n)$ en calculant

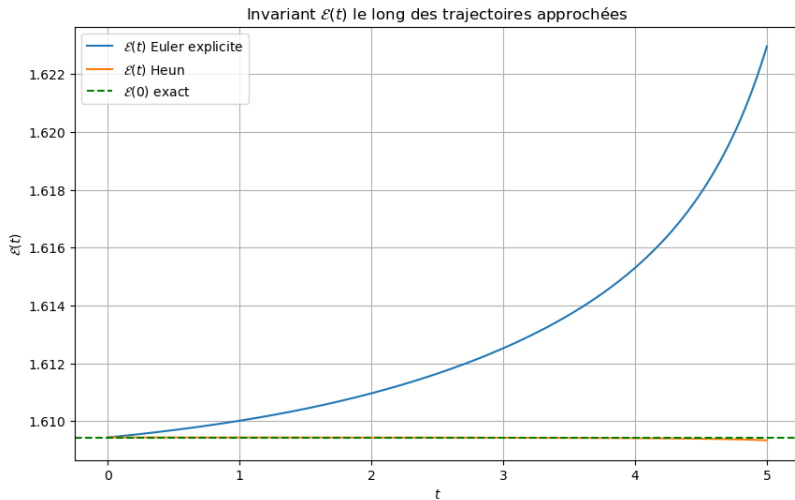
$$\|E(t_n, \tilde{x}_n, \tilde{y}_n) - E(0, x_0, y_0)\|$$

- Cas particulier : si $(x_0, y_0) = (0, 0)$ et $M(t) = (a, vt)$, $a > 0$ constant, alors on peut choisir

$$E(t, x, y) = k \ln \left(\sqrt{(a-x)^2 + (vt-y)^2} + (vt-y) \right) - (k-1) \ln(a-x)$$

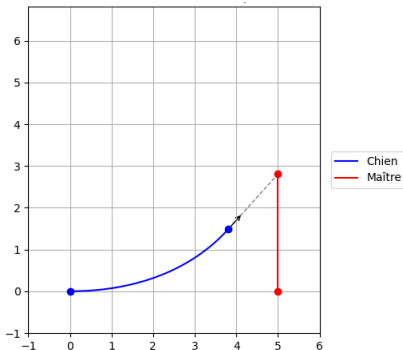
Remarque : Euler explicite ne conserve pas bien cet invariant

Cas $M(t) = (a, vt)$ – invariant : Euler vs Heun



Simulations avec $x_0 = y_0 = 0$, $v = 1$, $a = 5$, $k = 1.5$, $\Delta t = 0.01$

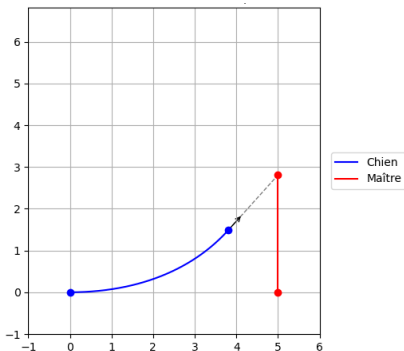
Cas particulier : le maître se déplace verticalement



Si $M(t) = (a, vt)$ on peut

- donner un invariant sans calculer la solution exacte
- calculer une expression explicite de $x \mapsto y = \eta(x)$
- calculer une expression explicite de $x \mapsto t = \tau(x)$

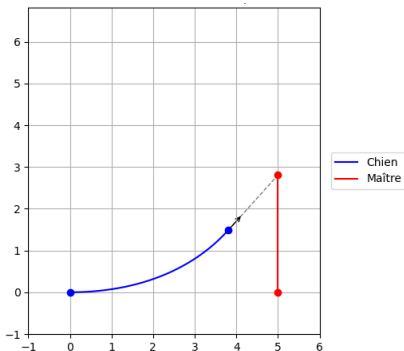
Cas particulier : le maître se déplace verticalement



Si $M(t) = (a, vt)$ on peut

- donner un invariant sans calculer la solution exacte
- calculer une expression explicite de $x \mapsto y = \eta(x)$
- calculer une expression explicite de $x \mapsto t = \tau(x)$

Cas particulier : le maître se déplace verticalement



Si $M(t) = (a, vt)$ on peut

- donner un invariant sans calculer la solution exacte
- calculer une expression explicite de $x \mapsto y = \eta(x)$
- calculer une expression explicite de $x \mapsto t = \tau(x)$

Cas $M(t) = (a, vt)$ – Changement de variables

Dans cette configuration, on peut paramétrer la trajectoire par x au lieu de t :

$$\begin{cases} t \mapsto x(t) \\ t \mapsto y(t) \end{cases} \iff \begin{cases} x \mapsto y = \eta(x) \\ x \mapsto t = \tau(x) \end{cases}$$

- Rapport des deux EDO :

$$\boxed{\eta'(x)} = \frac{y'(t)}{x'(t)} = \frac{y_m(t) - y(t)}{x_m(t) - x(t)} = \frac{vt - y(t)}{a - x(t)} = \frac{v\tau(x) - \eta(x)}{a - x}$$

- Norme du vecteur vitesse constante :

$$\sqrt{(x'(t))^2 + (y'(t))^2} = kv \rightsquigarrow (x'(t))^2(1 + (\eta'(x))^2) = (kv)^2 \rightsquigarrow \boxed{\tau'(x)} = \frac{dt}{dx} = \frac{\sqrt{1 + (\eta'(x))^2}}{kv}$$

Cas $M(t) = (a, vt)$ – Changement de variables

Dans cette configuration, on peut paramétrer la trajectoire par x au lieu de t :

$$\begin{cases} t \mapsto x(t) \\ t \mapsto y(t) \end{cases} \iff \begin{cases} x \mapsto y = \eta(x) \\ x \mapsto t = \tau(x) \end{cases}$$

- Rapport des deux EDO :

$$\boxed{\eta'(x)} = \frac{y'(t)}{x'(t)} = \frac{y_m(t) - y(t)}{x_m(t) - x(t)} = \frac{vt - y(t)}{a - x(t)} = \boxed{\frac{v\tau(x) - \eta(x)}{a - x}}$$

- Norme du vecteur vitesse constante :

$$\sqrt{(x'(t))^2 + (y'(t))^2} = kv \rightsquigarrow (x'(t))^2(1 + (\eta'(x))^2) = (kv)^2 \rightsquigarrow \boxed{\tau'(x)} = \frac{dt}{dx} = \boxed{\frac{\sqrt{1 + (\eta'(x))^2}}{kv}}$$

Cas $M(t) = (a, vt)$ – Changement de variables

Dans cette configuration, on peut paramétrer la trajectoire par x au lieu de t :

$$\begin{cases} t \mapsto x(t) \\ t \mapsto y(t) \end{cases} \iff \begin{cases} x \mapsto y = \eta(x) \\ x \mapsto t = \tau(x) \end{cases}$$

- Rapport des deux EDO :

$$\boxed{\eta'(x)} = \frac{y'(t)}{x'(t)} = \frac{y_m(t) - y(t)}{x_m(t) - x(t)} = \frac{vt - y(t)}{a - x(t)} = \boxed{\frac{v\tau(x) - \eta(x)}{a - x}}$$

- Norme du vecteur vitesse constante :

$$\sqrt{(x'(t))^2 + (y'(t))^2} = kv \rightsquigarrow (x'(t))^2(1 + (\eta'(x))^2) = (kv)^2 \rightsquigarrow \boxed{\tau'(x)} = \frac{dt}{dx} = \boxed{\frac{\sqrt{1 + (\eta'(x))^2}}{kv}}$$

Cas $M(t) = (a, vt)$ – Nouveau système d'EDO

Nouveau système de deux EDO d'ordre 1 :

$$\begin{cases} (a-x)\eta'(x) = v\tau(x) - \eta(x) \\ (kv)\tau'(x) = \sqrt{1 + (\eta'(x))^2} \end{cases}$$

avec $\eta(0) = 0$ et $\tau(0) = 0$.

- En dérivant la première équation par rapport à x , on obtient une EDO d'ordre 2 :

$$-\eta'(x) + (a-x)\eta''(x) = v\tau'(x) - \eta'(x) \quad \rightsquigarrow \quad (a-x)\eta''(x) = \frac{\sqrt{1 + (\eta'(x))^2}}{k}$$

- Posons $p = \eta'$. On a $p(0) = 0$ et

$$(a-x)p'(x) = \frac{1}{k}\sqrt{1 + p^2(x)}$$

Cas $M(t) = (a, vt)$ – Nouveau système d'EDO

Nouveau système de deux EDO d'ordre 1 :

$$\begin{cases} (a-x)\eta'(x) = v\tau(x) - \eta(x) \\ (kv)\tau'(x) = \sqrt{1 + (\eta'(x))^2} \end{cases}$$

avec $\eta(0) = 0$ et $\tau(0) = 0$.

- En dérivant la première équation par rapport à x , on obtient une EDO d'ordre 2 :

$$-\eta'(x) + (a-x)\eta''(x) = v\tau'(x) - \eta'(x) \quad \rightsquigarrow \quad (a-x)\eta''(x) = \frac{\sqrt{1 + (\eta'(x))^2}}{k}$$

- Posons $p = \eta'$. On a $p(0) = 0$ et

$$(a-x)p'(x) = \frac{1}{k}\sqrt{1 + p^2(x)}$$

Cas $M(t) = (a, vt)$ – Nouveau système d'EDO

Nouveau système de deux EDO d'ordre 1 :

$$\begin{cases} (a-x)\eta'(x) = v\tau(x) - \eta(x) \\ (kv)\tau'(x) = \sqrt{1 + (\eta'(x))^2} \end{cases}$$

avec $\eta(0) = 0$ et $\tau(0) = 0$.

- En dérivant la première équation par rapport à x , on obtient une EDO d'ordre 2 :

$$-\eta'(x) + (a-x)\eta''(x) = v\tau'(x) - \eta'(x) \quad \rightsquigarrow \quad (a-x)\eta''(x) = \frac{\sqrt{1 + (\eta'(x))^2}}{k}$$

- Posons $p = \eta'$. On a $p(0) = 0$ et

$$(a-x)p'(x) = \frac{1}{k}\sqrt{1 + p^2(x)}$$

Cas $M(t) = (a, vt)$ – Calcul de $x \mapsto y = \eta(x)$

- Étape 1 : calcul de p avec $p(0) = 0$ (EDO à variables séparables)

$$(a-x)p'(x) = \frac{1}{k} \sqrt{1+p^2(x)} \rightsquigarrow p(x) = \sinh \left(-\frac{1}{k} \ln \left(1 - \frac{x}{a} \right) \right) = \frac{1}{2} \left(\left(1 - \frac{x}{a} \right)^{-\frac{1}{k}} - \left(1 - \frac{x}{a} \right)^{\frac{1}{k}} \right)$$

- Étape 2 : $p = \eta' \rightsquigarrow$ calcul de η avec $\eta(0) = 0$

- $k = 1$

$$\eta(x) = \frac{a}{4} \left(\left(1 - \frac{x}{a} \right)^2 - \ln \left(1 - \frac{x}{a} \right)^2 - 1 \right) \rightsquigarrow \eta(x) \xrightarrow{x \rightarrow a} +\infty$$

$\rightsquigarrow$ le chien s'approche asymptotiquement du maître

- $k \neq 1$

$$\eta(x) = \frac{ak}{k^2 - 1} + \frac{k}{2} \left(\frac{a^{-1/k}}{k+1} (a-x)^{\frac{k+1}{k}} - \frac{a^{1/k}}{k-1} (a-x)^{\frac{k-1}{k}} \right)$$

$\rightsquigarrow$ si $k > 1$, le chien attrapera le maître en

$$\eta(a) = \frac{a}{k^2 - 1}$$

Cas $M(t) = (a, vt)$ – Calcul de $x \mapsto y = \eta(x)$

- Étape 1 : calcul de p avec $p(0) = 0$ (EDO à variables séparables)

$$(a-x)p'(x) = \frac{1}{k} \sqrt{1+p^2(x)} \rightsquigarrow p(x) = \sinh \left(-\frac{1}{k} \ln \left(1 - \frac{x}{a} \right) \right) = \frac{1}{2} \left(\left(1 - \frac{x}{a} \right)^{-\frac{1}{k}} - \left(1 - \frac{x}{a} \right)^{\frac{1}{k}} \right)$$

- Étape 2 : $p = \eta' \rightsquigarrow$ calcul de η avec $\eta(0) = 0$

- $k = 1$

$$\eta(x) = \frac{a}{4} \left(\left(1 - \frac{x}{a} \right)^2 - \ln \left(1 - \frac{x}{a} \right)^2 - 1 \right) \rightsquigarrow \eta(x) \xrightarrow{x \rightarrow a} +\infty$$

$\rightsquigarrow$ le chien s'approche asymptotiquement du maître

- $k \neq 1$

$$\eta(x) = \frac{ak}{k^2 - 1} + \frac{k}{2} \left(\frac{a^{-1/k}}{k+1} (a-x)^{\frac{k+1}{k}} - \frac{a^{1/k}}{k-1} (a-x)^{\frac{k-1}{k}} \right)$$

$\rightsquigarrow$ si $k > 1$, le chien attrapera le maître en

$$\eta(a) = \frac{a}{k^2 - 1}$$

Cas $M(t) = (a, vt)$ – Calcul de $x \mapsto y = \eta(x)$

- Étape 1 : calcul de p avec $p(0) = 0$ (EDO à variables séparables)

$$(a-x)p'(x) = \frac{1}{k} \sqrt{1+p^2(x)} \rightsquigarrow p(x) = \sinh \left(-\frac{1}{k} \ln \left(1 - \frac{x}{a} \right) \right) = \frac{1}{2} \left(\left(1 - \frac{x}{a} \right)^{-\frac{1}{k}} - \left(1 - \frac{x}{a} \right)^{\frac{1}{k}} \right)$$

- Étape 2 : $p = \eta' \rightsquigarrow$ calcul de η avec $\eta(0) = 0$

- $k = 1$

$$\eta(x) = \frac{a}{4} \left(\left(1 - \frac{x}{a} \right)^2 - \ln \left(1 - \frac{x}{a} \right)^2 - 1 \right) \rightsquigarrow \eta(x) \xrightarrow{x \rightarrow a} +\infty$$

$\rightsquigarrow$ le chien s'approche asymptotiquement du maître

- $k \neq 1$

$$\eta(x) = \frac{ak}{k^2 - 1} + \frac{k}{2} \left(\frac{a^{-1/k}}{k+1} (a-x)^{\frac{k+1}{k}} - \frac{a^{1/k}}{k-1} (a-x)^{\frac{k-1}{k}} \right)$$

$\rightsquigarrow$ si $k > 1$, le chien attrapera le maître en

$$\eta(a) = \frac{a}{k^2 - 1}$$

Cas $M(t) = (a, vt)$ – Calcul de $x \mapsto t = \tau(x)$

Calcul de $t = \tau(x)$ avec $\tau(0) = 0$

$$(kv)\tau'(x) = \sqrt{1 + (\eta'(x))^2} = \sqrt{1 + \sinh^2\left(-\frac{1}{k} \ln\left(1 - \frac{x}{a}\right)\right)} = \frac{1}{2} \left(\left(1 - \frac{x}{a}\right)^{-\frac{1}{k}} + \left(1 - \frac{x}{a}\right)^{\frac{1}{k}} \right)$$

• $k = 1$

$$\tau(x) = -\frac{a}{4v} \left(\left(1 - \frac{x}{a}\right)^2 + \ln\left(1 - \frac{x}{a}\right)^2 - 1 \right) \rightsquigarrow \tau(x) \xrightarrow{x \rightarrow a} +\infty$$

• $k \neq 1$

$$\tau(x) = \frac{ak}{v(k^2 - 1)} - \frac{1}{2v} \left(\frac{a^{-1/k}}{k+1} (a-x)^{\frac{k+1}{k}} + \frac{a^{1/k}}{k-1} (a-x)^{\frac{k-1}{k}} \right)$$

$\rightsquigarrow$ si $k > 1$, le chien attrapera le maître à l'instant

$$\tau(a) = \frac{ak}{v(k^2 - 1)}$$

Cas $M(t) = (a, vt)$ – Calcul de $x \mapsto t = \tau(x)$

Calcul de $t = \tau(x)$ avec $\tau(0) = 0$

$$(kv)\tau'(x) = \sqrt{1 + (\eta'(x))^2} = \sqrt{1 + \sinh^2\left(-\frac{1}{k} \ln\left(1 - \frac{x}{a}\right)\right)} = \frac{1}{2} \left(\left(1 - \frac{x}{a}\right)^{-\frac{1}{k}} + \left(1 - \frac{x}{a}\right)^{\frac{1}{k}} \right)$$

• $k = 1$

$$\tau(x) = -\frac{a}{4v} \left(\left(1 - \frac{x}{a}\right)^2 + \ln\left(1 - \frac{x}{a}\right)^2 - 1 \right) \rightsquigarrow \tau(x) \xrightarrow{x \rightarrow a} +\infty$$

• $k \neq 1$

$$\tau(x) = \frac{ak}{v(k^2 - 1)} - \frac{1}{2v} \left(\frac{a^{-1/k}}{k+1} (a-x)^{\frac{k+1}{k}} + \frac{a^{1/k}}{k-1} (a-x)^{\frac{k-1}{k}} \right)$$

$\rightsquigarrow$ si $k > 1$, le chien attrapera le maître à l'instant

$$\tau(a) = \frac{ak}{v(k^2 - 1)}$$

Cas $M(t) = (a, vt)$ – Calcul de $x \mapsto t = \tau(x)$

Calcul de $t = \tau(x)$ avec $\tau(0) = 0$

$$(kv)\tau'(x) = \sqrt{1 + (\eta'(x))^2} = \sqrt{1 + \sinh^2\left(-\frac{1}{k} \ln\left(1 - \frac{x}{a}\right)\right)} = \frac{1}{2} \left(\left(1 - \frac{x}{a}\right)^{-\frac{1}{k}} + \left(1 - \frac{x}{a}\right)^{\frac{1}{k}} \right)$$

• $k = 1$

$$\tau(x) = -\frac{a}{4v} \left(\left(1 - \frac{x}{a}\right)^2 + \ln\left(1 - \frac{x}{a}\right)^2 - 1 \right) \rightsquigarrow \tau(x) \xrightarrow{x \rightarrow a} +\infty$$

• $k \neq 1$

$$\tau(x) = \frac{ak}{v(k^2 - 1)} - \frac{1}{2v} \left(\frac{a^{-1/k}}{k+1} (a-x)^{\frac{k+1}{k}} + \frac{a^{1/k}}{k-1} (a-x)^{\frac{k-1}{k}} \right)$$

$\rightsquigarrow$ si $k > 1$, le chien attrapera le maître à l'instant

$$\tau(a) = \frac{ak}{v(k^2 - 1)}$$

Cas $M(t) = (a, vt)$ – Calcul de $x \mapsto t = \tau(x)$

Calcul de $t = \tau(x)$ avec $\tau(0) = 0$

$$(kv)\tau'(x) = \sqrt{1 + (\eta'(x))^2} = \sqrt{1 + \sinh^2\left(-\frac{1}{k} \ln\left(1 - \frac{x}{a}\right)\right)} = \frac{1}{2} \left(\left(1 - \frac{x}{a}\right)^{-\frac{1}{k}} + \left(1 - \frac{x}{a}\right)^{\frac{1}{k}} \right)$$

• $k = 1$

$$\tau(x) = -\frac{a}{4v} \left(\left(1 - \frac{x}{a}\right)^2 + \ln\left(1 - \frac{x}{a}\right)^2 - 1 \right) \rightsquigarrow \tau(x) \xrightarrow{x \rightarrow a} +\infty$$

• $k \neq 1$

$$\tau(x) = \frac{ak}{v(k^2 - 1)} - \frac{1}{2v} \left(\frac{a^{-1/k}}{k+1} (a-x)^{\frac{k+1}{k}} + \frac{a^{1/k}}{k-1} (a-x)^{\frac{k-1}{k}} \right)$$

$\rightsquigarrow$ si $k > 1$, le chien attrapera le maître à l'instant

$$\tau(a) = \frac{ak}{v(k^2 - 1)}$$

Cas $M(t) = (a, vt)$ – Calcul de $x \mapsto t = \tau(x)$

Calcul de $t = \tau(x)$ avec $\tau(0) = 0$

$$(kv)\tau'(x) = \sqrt{1 + (\eta'(x))^2} = \sqrt{1 + \sinh^2\left(-\frac{1}{k} \ln\left(1 - \frac{x}{a}\right)\right)} = \frac{1}{2} \left(\left(1 - \frac{x}{a}\right)^{-\frac{1}{k}} + \left(1 - \frac{x}{a}\right)^{\frac{1}{k}} \right)$$

- $k = 1$

$$\tau(x) = -\frac{a}{4v} \left(\left(1 - \frac{x}{a}\right)^2 + \ln\left(1 - \frac{x}{a}\right)^2 - 1 \right) \rightsquigarrow \tau(x) \xrightarrow{x \rightarrow a} +\infty$$

- $k \neq 1$

$$\tau(x) = \frac{ak}{v(k^2 - 1)} - \frac{1}{2v} \left(\frac{a^{-1/k}}{k+1} (a-x)^{\frac{k+1}{k}} + \frac{a^{1/k}}{k-1} (a-x)^{\frac{k-1}{k}} \right)$$

$\rightsquigarrow$ si $k > 1$, le chien attrapera le maître à l'instant

$$\tau(a) = \frac{ak}{v(k^2 - 1)}$$

Synthèse

Si $k > 1$, le chien rejoint le maître.

- **Point de rencontre :**

$$(x_F, y_F) = \left(a, \frac{a}{k^2 - 1} \right)$$

- Instant de capture :

$$t_F = \frac{a k}{v(k^2 - 1)}$$

- Distance parcourue par le chien :

$$d_F = v_{\text{chien}} t_F = k v t_F = \frac{a k^2}{k^2 - 1}$$

Synthèse

Si $k > 1$, le chien rejoint le maître.

- **Point de rencontre :**

$$(x_F, y_F) = \left(a, \frac{a}{k^2 - 1} \right)$$

- **Instant de capture :**

$$t_F = \frac{a k}{v(k^2 - 1)}$$

- Distance parcourue par le chien :

$$d_F = v_{\text{chien}} t_F = k v t_F = \frac{a k^2}{k^2 - 1}$$

Synthèse

Si $k > 1$, le chien rejoint le maître.

- **Point de rencontre :**

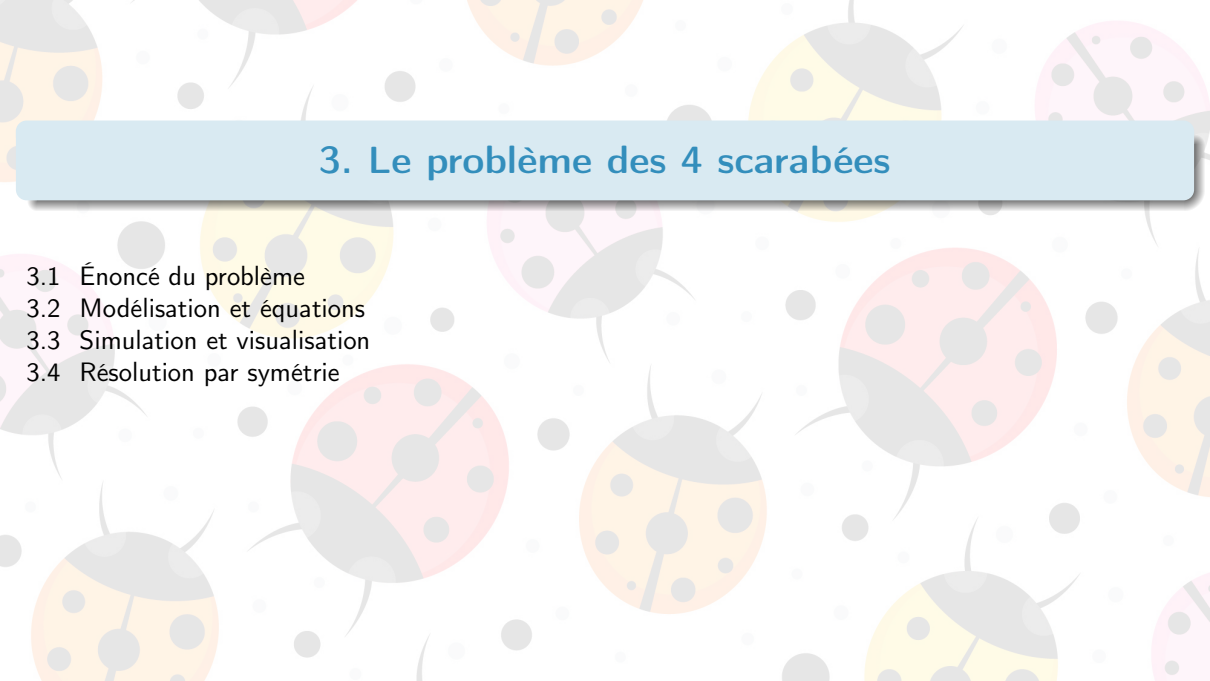
$$(x_F, y_F) = \left(a, \frac{a}{k^2 - 1} \right)$$

- **Instant de capture :**

$$t_F = \frac{a k}{v(k^2 - 1)}$$

- **Distance parcourue par le chien :**

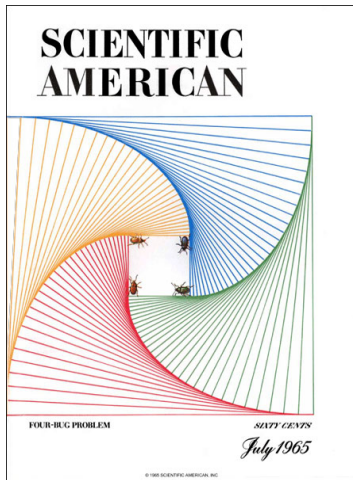
$$d_F = v_{\text{chien}} t_F = k v t_F = \frac{a k^2}{k^2 - 1}$$

The background of the slide features a repeating pattern of stylized ladybugs in orange, pink, and yellow, each with grey spots and antennae. Interspersed among the ladybugs are small, light grey dots of varying sizes.

3. Le problème des 4 scarabées

- 3.1 Énoncé du problème
- 3.2 Modélisation et équations
- 3.3 Simulation et visualisation
- 3.4 Résolution par symétrie

Les 4 scarabées de Gardner – énoncé

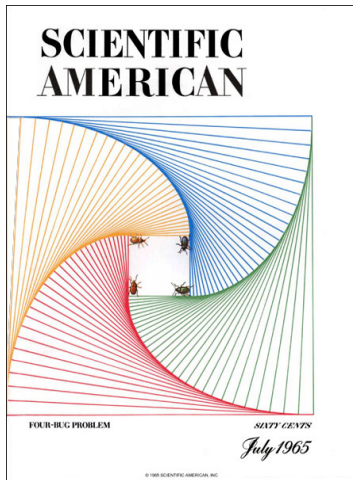


Quatre scarabées – A , B , C et D – occupent les sommets d'un carré de 10 pouces de côté. A et C sont mâles, B et D sont femelles. Simultanément, A se dirige vers B , B vers C , C vers D , et D vers A . Si tous les scarabées se déplacent à vitesse constante, **ils décrivent quatre spirales logarithmiques congruentes** qui se rejoignent au centre du carré. Quelle distance parcourt chaque scarabée avant de se rencontrer ? Le problème peut se résoudre sans calcul différentiel.

Nota Bene

Les mouvements sont **interdépendants** ($\neq$ courbe du chien)

Les 4 scarabées de Gardner – énoncé



Quatre scarabées – A , B , C et D – occupent les sommets d'un carré de 10 pouces de côté. A et C sont mâles, B et D sont femelles. Simultanément, A se dirige vers B , B vers C , C vers D , et D vers A . Si tous les scarabées se déplacent à vitesse constante, **ils décrivent quatre spirales logarithmiques congruentes** qui se rejoignent au centre du carré. Quelle distance parcourt chaque scarabée avant de se rencontrer ? Le problème peut se résoudre sans calcul différentiel.

Nota Bene

Les mouvements sont **interdépendants** ($\neq$ courbe du chien)

Position du problème

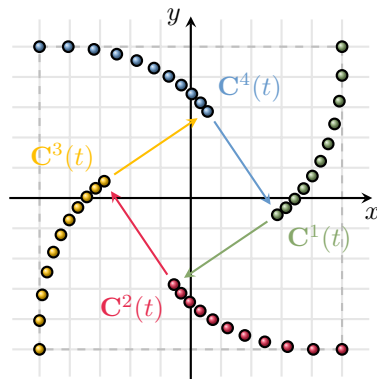
Hypothèses

4 scarabées ($k \in \{1, 2, 3, 4\}$)

- à $t = 0$ aux sommets d'un carré de côté ℓ ,
- se déplacent à la même vitesse constante v ,
- k poursuit $k + 1 \pmod{4}$.

Position du k -ème scarabée à l'instant t :

$$\mathbf{C}^k(t) = (x^k(t), y^k(t)), \quad k \in \{1, 2, 3, 4\}$$



Programmer avant de mathématiser

Δt intervalle de temps (fixé arbitrairement)

$\mathbf{C}_n^k \approx \mathbf{C}^k(t_n)$ position approchée du scarabée k à l'instant $t_n = n\Delta t$.

Principe d'une approximation

À l'instant t_n , chaque scarabée k avance en ligne droite à vitesse v pendant Δt vers la position du scarabée $k+1 \pmod{4}$.

Récurrance :

$$\begin{cases} \mathbf{C}_0^k \text{ position initiale connue} \\ \mathbf{C}_{n+1}^k = \mathbf{C}_n^k + \Delta t v \frac{\mathbf{C}_n^{k+1} - \mathbf{C}_n^k}{\|\mathbf{C}_n^{k+1} - \mathbf{C}_n^k\|} \end{cases}$$

Critère d'arrêt : $\|\mathbf{C}_n^{k+1} - \mathbf{C}_n^k\| < \varepsilon$

Simulations

Résultat observé

Trajectoires : spirale (logarithmique), symétriques, convergeant vers le centre.

Programmer avant de mathématiser

Δt intervalle de temps (fixé arbitrairement)

$\mathbf{C}_n^k \approx \mathbf{C}^k(t_n)$ position approchée du scarabée k à l'instant $t_n = n\Delta t$.

Principe d'une approximation

À l'instant t_n , chaque scarabée k avance en ligne droite à vitesse v pendant Δt vers la position du scarabée $k + 1 \pmod{4}$.

Récurrance :

$$\begin{cases} \mathbf{C}_0^k \text{ position initiale connue} \\ \mathbf{C}_{n+1}^k = \mathbf{C}_n^k + \Delta t v \frac{\mathbf{C}_n^{k+1} - \mathbf{C}_n^k}{\|\mathbf{C}_n^{k+1} - \mathbf{C}_n^k\|} \end{cases}$$

Critère d'arrêt : $\|\mathbf{C}_n^{k+1} - \mathbf{C}_n^k\| < \varepsilon$

Simulations

Résultat observé

Trajectoires : spirale (logarithmique), symétriques, convergeant vers le centre.

Programmer avant de mathématiser

Δt intervalle de temps (fixé arbitrairement)

$\mathbf{C}_n^k \approx \mathbf{C}^k(t_n)$ position approchée du scarabée k à l'instant $t_n = n\Delta t$.

Principe d'une approximation

À l'instant t_n , chaque scarabée k avance en ligne droite à vitesse v pendant Δt vers la position du scarabée $k + 1 \pmod{4}$.

Récurrance :

$$\begin{cases} \mathbf{C}_0^k \text{ position initiale connue} \\ \mathbf{C}_{n+1}^k = \mathbf{C}_n^k + \Delta t v \frac{\mathbf{C}_n^{k+1} - \mathbf{C}_n^k}{\|\mathbf{C}_n^{k+1} - \mathbf{C}_n^k\|} \end{cases}$$

Critère d'arrêt : $\|\mathbf{C}_n^{k+1} - \mathbf{C}_n^k\| < \varepsilon$

Simulations

Résultat observé

Trajectoires : spirale (logarithmique), symétriques, convergeant vers le centre.

Programmer avant de mathématiser

Δt intervalle de temps (fixé arbitrairement)

$\mathbf{C}_n^k \approx \mathbf{C}^k(t_n)$ position approchée du scarabée k à l'instant $t_n = n\Delta t$.

Principe d'une approximation

À l'instant t_n , chaque scarabée k avance en ligne droite à vitesse v pendant Δt vers la position du scarabée $k + 1 \pmod{4}$.

Récurrance :

$$\begin{cases} \mathbf{C}_0^k \text{ position initiale connue} \\ \mathbf{C}_{n+1}^k = \mathbf{C}_n^k + \Delta t v \frac{\mathbf{C}_n^{k+1} - \mathbf{C}_n^k}{\|\mathbf{C}_n^{k+1} - \mathbf{C}_n^k\|} \end{cases}$$

Critère d'arrêt : $\|\mathbf{C}_n^{k+1} - \mathbf{C}_n^k\| < \varepsilon$

Simulations

Résultat observé

Trajectoires : spirale (logarithmique), symétriques, convergeant vers le centre.

Programmer avant de mathématiser

Δt intervalle de temps (fixé arbitrairement)

$\mathbf{C}_n^k \approx \mathbf{C}^k(t_n)$ position approchée du scarabée k à l'instant $t_n = n\Delta t$.

Principe d'une approximation

À l'instant t_n , chaque scarabée k avance en ligne droite à vitesse v pendant Δt vers la position du scarabée $k + 1 \pmod{4}$.

Récurrance :

$$\begin{cases} \mathbf{C}_0^k \text{ position initiale connue} \\ \mathbf{C}_{n+1}^k = \mathbf{C}_n^k + \Delta t v \frac{\mathbf{C}_n^{k+1} - \mathbf{C}_n^k}{\|\mathbf{C}_n^{k+1} - \mathbf{C}_n^k\|} \end{cases}$$

Critère d'arrêt : $\|\mathbf{C}_n^{k+1} - \mathbf{C}_n^k\| < \varepsilon$

Simulations

Résultat observé

Trajectoires : spirale (logarithmique), symétriques, convergeant vers le centre.

Mise en équation

Trajectoire du scarabée k :

$$(\mathbf{C}^k)'(t) = v \frac{\mathbf{C}^{k+1}(t) - \mathbf{C}^k(t)}{\|\mathbf{C}^{k+1}(t) - \mathbf{C}^k(t)\|}.$$

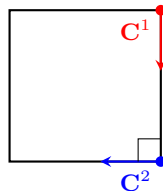
Il s'agit de **8 EDOs couplées** (pour $t \mapsto x^k(t)$ et $t \mapsto y^k(t)$, $k = 1, \dots, 4$) :

$$\begin{cases} (x^k)'(t) = v \frac{x^{k+1}(t) - x^k(t)}{\sqrt{(x^{k+1}(t) - x^k(t))^2 + (y^{k+1}(t) - y^k(t))^2}}, \\ (y^k)'(t) = v \frac{y^{k+1}(t) - y^k(t)}{\sqrt{(x^{k+1}(t) - x^k(t))^2 + (y^{k+1}(t) - y^k(t))^2}}. \end{cases}$$

Raisonnement intuitif — Gardner

Le mouvement d'un scarabée est **perpendiculaire** à celui du scarabée qu'il poursuit, donc

- à chaque instant, les 4 scarabées restent aux quatre sommets d'un carré de centre $(0,0)$ qui rétrécit et tourne
- temps de capture = temps si le poursuivi est immobile
- la vitesse relative dans la direction radiale est constante : le scarabée se rapproche du centre de manière uniforme en "angle constant".



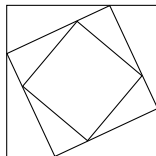
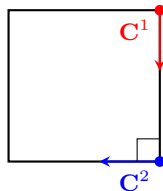
Conclusion

- Distance parcourue par chaque scarabée : ℓ (côté du carré initial)
- Durée de la poursuite : ℓ/v
- Trajectoire : spirale logarithmique <https://mathworld.wolfram.com/LogarithmicSpiral.html>

Raisonnement intuitif — Gardner

Le mouvement d'un scarabée est **perpendiculaire** à celui du scarabée qu'il poursuit, donc

- à chaque instant, les 4 scarabées restent aux quatre sommets d'un carré de centre $(0,0)$ qui rétrécit et tourne
- temps de capture = temps si le poursuivi est immobile
- la vitesse relative dans la direction radiale est constante : le scarabée se rapproche du centre de manière uniforme en "angle constant".



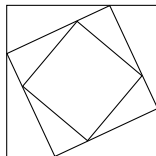
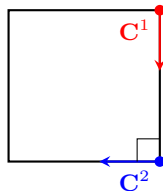
Conclusion

- Distance parcourue par chaque scarabée : ℓ (côté du carré initial)
- Durée de la poursuite : ℓ/v
- Trajectoire : spirale logarithmique <https://mathworld.wolfram.com/LogarithmicSpiral.html>

Raisonnement intuitif — Gardner

Le mouvement d'un scarabée est **perpendiculaire** à celui du scarabée qu'il poursuit, donc

- à chaque instant, les 4 scarabées restent aux quatre sommets d'un carré de centre $(0,0)$ qui rétrécit et tourne
- temps de capture = temps si le poursuivi est immobile
- la vitesse relative dans la direction radiale est constante : le scarabée se rapproche du centre de manière uniforme en "angle constant".



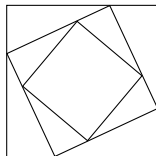
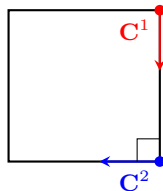
Conclusion

- Distance parcourue par chaque scarabée : ℓ (côté du carré initial)
- Durée de la poursuite : ℓ/v
- Trajectoire : spirale logarithmique <https://mathworld.wolfram.com/LogarithmicSpiral.html>

Raisonnement intuitif — Gardner

Le mouvement d'un scarabée est **perpendiculaire** à celui du scarabée qu'il poursuit, donc

- à chaque instant, les 4 scarabées restent aux quatre sommets d'un carré de centre $(0,0)$ qui rétrécit et tourne
- temps de capture = temps si le poursuivi est immobile
- la vitesse relative dans la direction radiale est constante : le scarabée se rapproche du centre de manière uniforme en “angle constant”.



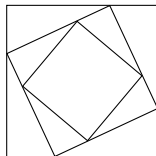
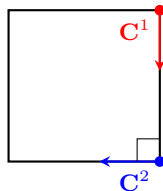
Conclusion

- Distance parcourue par chaque scarabée : ℓ (côté du carré initial)
- Durée de la poursuite : ℓ/v
- Trajectoire : spirale logarithmique <https://mathworld.wolfram.com/LogarithmicSpiral.html>

Raisonnement intuitif — Gardner

Le mouvement d'un scarabée est **perpendiculaire** à celui du scarabée qu'il poursuit, donc

- à chaque instant, les 4 scarabées restent aux quatre sommets d'un carré de centre $(0,0)$ qui rétrécit et tourne
- temps de capture = temps si le poursuivi est immobile
- la vitesse relative dans la direction radiale est constante : le scarabée se rapproche du centre de manière uniforme en “angle constant”.



Conclusion

- Distance parcourue par chaque scarabée : ℓ (côté du carré initial)
- Durée de la poursuite : ℓ/v
- Trajectoire : spirale logarithmique <https://mathworld.wolfram.com/LogarithmicSpiral.html>

Symétrie $\rightsquigarrow$ Passage aux coordonnées polaires

Paramétrisation :

$$\begin{cases} x^k(t) = r^k(t) \cos(\theta^k(t)), \\ y^k(t) = r^k(t) \sin(\theta^k(t)). \end{cases}$$

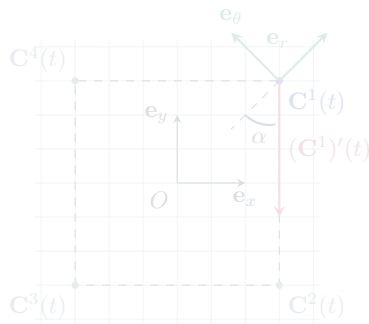
Notation

La position $\mathbf{C}^k(t)$ s'écrit :

- Cartésiennes : $(x^k(t), y^k(t))$.
- Polaires : $(r^k(t) : \theta^k(t))$.

Le vecteur vitesse $(\mathbf{C}^k)'(t)$ s'écrit :

- Cartésiennes : $((x^k)'(t), (y^k)'(t))$.
- Polaires : $((r^k)'(t) : r^k(t)(\theta^k)'(t))$.



<https://mathworld.wolfram.com/PolarCoordinates.html>

Symétrie $\rightsquigarrow$ Passage aux coordonnées polaires

Paramétrisation :

$$\begin{cases} x^k(t) = r^k(t) \cos(\theta^k(t)), \\ y^k(t) = r^k(t) \sin(\theta^k(t)). \end{cases}$$

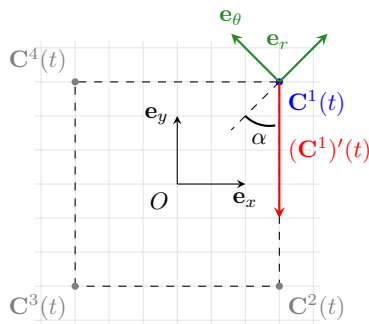
Notation

La position $\mathbf{C}^k(t)$ s'écrit :

- Cartésiennes : $(x^k(t), y^k(t))$.
- Polaires : $(r^k(t) : \theta^k(t))$.

Le vecteur vitesse $(\mathbf{C}^k)'(t)$ s'écrit :

- Cartésiennes : $((x^k)'(t), (y^k)'(t))$.
- Polaires : $((r^k)'(t) : r^k(t)(\theta^k)'(t))$.



<https://mathworld.wolfram.com/PolarCoordinates.html>

Équations différentielles en coordonnées polaires

Par symétrie, il suffit d'étudier le scarabée $k = 1$ (on omettra le k), de position initiale

$$\mathbf{C}(0) = \left(\frac{\ell}{2}, \frac{\ell}{2} \right) = \left(\frac{\ell}{\sqrt{2}} : \frac{\pi}{4} \right).$$

En projetant la vitesse $\mathbf{C}'(t) = (r'(t) : r(t)\theta'(t))$ sur $(\mathbf{e}_r, \mathbf{e}_\theta)$ on obtient

$$\mathbf{C}'(t) = \left(-v \cos\left(\frac{\pi}{4}\right) : -v \sin\left(\frac{\pi}{4}\right) \right) = \left(-\frac{v}{\sqrt{2}} : -\frac{v}{\sqrt{2}} \right)$$

ce qui donne

$$\begin{cases} r'(t) = -\frac{v}{\sqrt{2}}, \\ r(t)\theta'(t) = -\frac{v}{\sqrt{2}}, \\ r(0) = \frac{\ell}{\sqrt{2}}, \\ \theta(0) = \frac{\pi}{4}. \end{cases} \rightsquigarrow \begin{cases} r(t) = r(0) - \frac{v}{\sqrt{2}}t = \frac{\ell - vt}{\sqrt{2}} \\ \theta(t) = \theta(0) + \ln\left(\frac{r(t)}{r(0)}\right) = \frac{\pi}{4} + \ln\left(1 - \frac{vt}{\ell}\right) \end{cases}$$

Équations différentielles en coordonnées polaires

Par symétrie, il suffit d'étudier le scarabée $k = 1$ (on omettra le k), de position initiale

$$\mathbf{C}(0) = \left(\frac{\ell}{2}, \frac{\ell}{2} \right) = \left(\frac{\ell}{\sqrt{2}} : \frac{\pi}{4} \right).$$

En projetant la vitesse $\mathbf{C}'(t) = (r'(t) : r(t)\theta'(t))$ sur $(\mathbf{e}_r, \mathbf{e}_\theta)$ on obtient

$$\mathbf{C}'(t) = \left(-v \cos\left(\frac{\pi}{4}\right) : -v \sin\left(\frac{\pi}{4}\right) \right) = \left(-\frac{v}{\sqrt{2}} : -\frac{v}{\sqrt{2}} \right)$$

ce qui donne

$$\begin{cases} r'(t) = -\frac{v}{\sqrt{2}}, \\ r(t)\theta'(t) = -\frac{v}{\sqrt{2}}, \\ r(0) = \frac{\ell}{\sqrt{2}}, \\ \theta(0) = \frac{\pi}{4}. \end{cases} \rightsquigarrow \begin{cases} r(t) = r(0) - \frac{v}{\sqrt{2}}t = \frac{\ell - vt}{\sqrt{2}} \\ \theta(t) = \theta(0) + \ln\left(\frac{r(t)}{r(0)}\right) = \frac{\pi}{4} + \ln\left(1 - \frac{vt}{\ell}\right) \end{cases}$$

Équations différentielles en coordonnées polaires

Par symétrie, il suffit d'étudier le scarabée $k = 1$ (on omettra le k), de position initiale

$$\mathbf{C}(0) = \left(\frac{\ell}{2}, \frac{\ell}{2} \right) = \left(\frac{\ell}{\sqrt{2}} : \frac{\pi}{4} \right).$$

En projetant la vitesse $\mathbf{C}'(t) = (r'(t) : r(t)\theta'(t))$ sur $(\mathbf{e}_r, \mathbf{e}_\theta)$ on obtient

$$\mathbf{C}'(t) = \left(-v \cos\left(\frac{\pi}{4}\right) : -v \sin\left(\frac{\pi}{4}\right) \right) = \left(-\frac{v}{\sqrt{2}} : -\frac{v}{\sqrt{2}} \right)$$

ce qui donne

$$\begin{cases} r'(t) = -\frac{v}{\sqrt{2}}, \\ r(t)\theta'(t) = -\frac{v}{\sqrt{2}}, \\ r(0) = \frac{\ell}{\sqrt{2}}, \\ \theta(0) = \frac{\pi}{4}. \end{cases} \rightsquigarrow \begin{cases} r(t) = r(0) - \frac{v}{\sqrt{2}}t = \frac{\ell - vt}{\sqrt{2}} \\ \theta(t) = \theta(0) + \ln\left(\frac{r(t)}{r(0)}\right) = \frac{\pi}{4} + \ln\left(1 - \frac{vt}{\ell}\right) \end{cases}$$

Synthèse

- **Trajectoire du scarabée k :**

$$\mathbf{C}^k(t) = \left(r(t) : \theta(t) - \frac{\pi}{4} \right)$$

- Trajectoire logarithmique

$$\theta(t) = \theta(0) + \ln \left(\frac{r(t)}{r(0)} \right) \rightsquigarrow r(t) = r(0)e^{\theta(t)-\theta(0)} \rightsquigarrow r(\theta) = r(0)e^{(\theta-\theta(0))} = \frac{\ell}{\sqrt{2}}e^{(\theta-\frac{\pi}{4})}$$

- **Durée de la poursuite :** la capture a lieu quand $r(t) = 0 \rightsquigarrow t_F = \frac{\ell}{v}$
- **Distance parcourue par chaque scarabée** $= vt_F = \ell$

Remarque : une approche plus élégante consiste à utiliser les nombres complexes

Synthèse

- Trajectoire du scarabée k :

$$\mathbf{C}^k(t) = \left(r(t) : \theta(t) - \frac{\pi}{4} \right)$$

- Trajectoire logarithmique

$$\theta(t) = \theta(0) + \ln \left(\frac{r(t)}{r(0)} \right) \rightsquigarrow r(t) = r(0)e^{\theta(t)-\theta(0)} \rightsquigarrow r(\theta) = r(0)e^{(\theta-\theta(0))} = \frac{\ell}{\sqrt{2}}e^{(\theta-\frac{\pi}{4})}$$

- Durée de la poursuite : la capture a lieu quand $r(t) = 0 \rightsquigarrow t_F = \frac{\ell}{v}$
- Distance parcourue par chaque scarabée $= vt_F = \ell$

Remarque : une approche plus élégante consiste à utiliser les nombres complexes

Synthèse

- **Trajectoire du scarabée k :**

$$\mathbf{C}^k(t) = \left(r(t) : \theta(t) - \frac{\pi}{4} \right)$$

- **Trajectoire logarithmique**

$$\theta(t) = \theta(0) + \ln \left(\frac{r(t)}{r(0)} \right) \rightsquigarrow r(t) = r(0)e^{\theta(t)-\theta(0)} \rightsquigarrow r(\theta) = r(0)e^{(\theta-\theta(0))} = \frac{\ell}{\sqrt{2}}e^{(\theta-\frac{\pi}{4})}$$

- **Durée de la poursuite :** la capture a lieu quand $r(t) = 0 \rightsquigarrow t_F = \frac{\ell}{v}$

- Distance parcourue par chaque scarabée $= vt_F = \ell$

Remarque : une approche plus élégante consiste à utiliser les nombres complexes

Synthèse

- **Trajectoire du scarabée k :**

$$\mathbf{C}^k(t) = \left(r(t) : \theta(t) - \frac{\pi}{4} \right)$$

- **Trajectoire logarithmique**

$$\theta(t) = \theta(0) + \ln \left(\frac{r(t)}{r(0)} \right) \rightsquigarrow r(t) = r(0)e^{\theta(t)-\theta(0)} \rightsquigarrow r(\theta) = r(0)e^{(\theta-\theta(0))} = \frac{\ell}{\sqrt{2}}e^{(\theta-\frac{\pi}{4})}$$

- **Durée de la poursuite :** la capture a lieu quand $r(t) = 0 \rightsquigarrow t_F = \frac{\ell}{v}$
- **Distance parcourue par chaque scarabée** $= vt_F = \ell$

Remarque : une approche plus élégante consiste à utiliser les nombres complexes

Synthèse

- **Trajectoire du scarabée k :**

$$\mathbf{C}^k(t) = \left(r(t) : \theta(t) - \frac{\pi}{4} \right)$$

- **Trajectoire logarithmique**

$$\theta(t) = \theta(0) + \ln \left(\frac{r(t)}{r(0)} \right) \rightsquigarrow r(t) = r(0)e^{\theta(t)-\theta(0)} \rightsquigarrow r(\theta) = r(0)e^{(\theta-\theta(0))} = \frac{\ell}{\sqrt{2}}e^{(\theta-\frac{\pi}{4})}$$

- **Durée de la poursuite :** la capture a lieu quand $r(t) = 0 \rightsquigarrow t_F = \frac{\ell}{v}$
- **Distance parcourue par chaque scarabée** $= vt_F = \ell$

Remarque : une approche plus élégante consiste à utiliser les nombres complexes

Généralisations : polygones réguliers à n côtés

- n scarabées initialement aux sommets d'un polygone régulier, se poursuivant à vitesse v constante. Par symétrie, on étudie un seul scarabée.

- On notera

- $s = -1$ si la poursuite est dans le sens horaire, $s = 1$ sinon.
- $\alpha = \frac{\pi}{2} - \beta$ l'angle entre le vecteur vitesse et le rayon, avec $\beta = \frac{\pi}{n}$.

- La projection de la vitesse v sur le repère polaire donne

$$(r'(t) : r(t)\theta'(t)) = \mathbf{C}'(t) = (-v \cos(\alpha) : sv \sin(\alpha)) = (-v \sin(\beta) : sv \cos(\beta)).$$

- Solution

$$\begin{cases} r'(t) = -v \sin(\beta), \\ r(t)\theta'(t) = sv \cos(\beta), \end{cases} \rightsquigarrow \begin{cases} r(t) = r(0) - v \sin(\beta)t, \\ \theta(t) = \theta(0) + \frac{s}{\tan(\beta)} \ln\left(\frac{r(0)}{r(t)}\right), \end{cases}$$

- Conclusion : les trajectoires sont des spirales logarithmiques convergeant vers le centre, le temps pour y parvenir est $t_F = \frac{r(0)}{v \sin(\beta)}$ et la distance parcourue est $\frac{r(0)}{\sin(\beta)}$.

Généralisations : polygones réguliers à n côtés

- n scarabées initialement aux sommets d'un polygone régulier, se poursuivant à vitesse v constante. Par symétrie, on étudie un seul scarabée.
- On notera
 - $s = -1$ si la poursuite est dans le sens horaire, $s = 1$ sinon.
 - $\alpha = \frac{\pi}{2} - \beta$ l'angle entre le vecteur vitesse et le rayon, avec $\beta = \frac{\pi}{n}$.
- La projection de la vitesse v sur le repère polaire donne

$$(r'(t) : r(t)\theta'(t)) = \mathbf{C}'(t) = (-v \cos(\alpha) : sv \sin(\alpha)) = (-v \sin(\beta) : sv \cos(\beta)).$$

- Solution

$$\begin{cases} r'(t) = -v \sin(\beta), \\ r(t)\theta'(t) = sv \cos(\beta), \end{cases} \rightsquigarrow \begin{cases} r(t) = r(0) - v \sin(\beta)t, \\ \theta(t) = \theta(0) + \frac{s}{\tan(\beta)} \ln\left(\frac{r(0)}{r(t)}\right), \end{cases}$$

- Conclusion : les trajectoires sont des spirales logarithmiques convergeant vers le centre, le temps pour y parvenir est $t_F = \frac{r(0)}{v \sin(\beta)}$ et la distance parcourue est $\frac{r(0)}{\sin(\beta)}$.

Généralisations : polygones réguliers à n côtés

- n scarabées initialement aux sommets d'un polygone régulier, se poursuivant à vitesse v constante. Par symétrie, on étudie un seul scarabée.
- On notera
 - $s = -1$ si la poursuite est dans le sens horaire, $s = 1$ sinon.
 - $\alpha = \frac{\pi}{2} - \beta$ l'angle entre le vecteur vitesse et le rayon, avec $\beta = \frac{\pi}{n}$.
- La projection de la vitesse v sur le repère polaire donne

$$(r'(t) : r(t)\theta'(t)) = \mathbf{C}'(t) = (-v \cos(\alpha) : sv \sin(\alpha)) = (-v \sin(\beta) : sv \cos(\beta)).$$

- Solution

$$\begin{cases} r'(t) = -v \sin(\beta), \\ r(t)\theta'(t) = sv \cos(\beta), \end{cases} \rightsquigarrow \begin{cases} r(t) = r(0) - v \sin(\beta)t, \\ \theta(t) = \theta(0) + \frac{s}{\tan(\beta)} \ln\left(\frac{r(0)}{r(t)}\right), \end{cases}$$

- Conclusion : les trajectoires sont des spirales logarithmiques convergeant vers le centre, le temps pour y parvenir est $t_F = \frac{r(0)}{v \sin(\beta)}$ et la distance parcourue est $\frac{r(0)}{\sin(\beta)}$.

Généralisations : polygones réguliers à n côtés

- n scarabées initialement aux sommets d'un polygone régulier, se poursuivant à vitesse v constante. Par symétrie, on étudie un seul scarabée.
- On notera
 - $s = -1$ si la poursuite est dans le sens horaire, $s = 1$ sinon.
 - $\alpha = \frac{\pi}{2} - \beta$ l'angle entre le vecteur vitesse et le rayon, avec $\beta = \frac{\pi}{n}$.
- La projection de la vitesse v sur le repère polaire donne

$$(r'(t) : r(t)\theta'(t)) = \mathbf{C}'(t) = (-v \cos(\alpha) : sv \sin(\alpha)) = (-v \sin(\beta) : sv \cos(\beta)).$$

• Solution

$$\begin{cases} r'(t) = -v \sin(\beta), \\ r(t)\theta'(t) = sv \cos(\beta), \end{cases} \rightsquigarrow \begin{cases} r(t) = r(0) - v \sin(\beta)t, \\ \theta(t) = \theta(0) + \frac{s}{\tan(\beta)} \ln\left(\frac{r(0)}{r(t)}\right), \end{cases}$$

- Conclusion : les trajectoires sont des spirales logarithmiques convergeant vers le centre, le temps pour y parvenir est $t_F = \frac{r(0)}{v \sin(\beta)}$ et la distance parcourue est $\frac{r(0)}{\sin(\beta)}$.

Généralisations : polygones réguliers à n côtés

- n scarabées initialement aux sommets d'un polygone régulier, se poursuivant à vitesse v constante. Par symétrie, on étudie un seul scarabée.
- On notera
 - $s = -1$ si la poursuite est dans le sens horaire, $s = 1$ sinon.
 - $\alpha = \frac{\pi}{2} - \beta$ l'angle entre le vecteur vitesse et le rayon, avec $\beta = \frac{\pi}{n}$.
- La projection de la vitesse v sur le repère polaire donne

$$(r'(t) : r(t)\theta'(t)) = \mathbf{C}'(t) = (-v \cos(\alpha) : sv \sin(\alpha)) = (-v \sin(\beta) : sv \cos(\beta)).$$

- Solution

$$\begin{cases} r'(t) = -v \sin(\beta), \\ r(t)\theta'(t) = sv \cos(\beta), \end{cases} \rightsquigarrow \begin{cases} r(t) = r(0) - v \sin(\beta)t, \\ \theta(t) = \theta(0) + \frac{s}{\tan(\beta)} \ln\left(\frac{r(0)}{r(t)}\right), \end{cases}$$

- Conclusion : les trajectoires sont des spirales logarithmiques convergeant vers le centre, le temps pour y parvenir est $t_F = \frac{r(0)}{v \sin(\beta)}$ et la distance parcourue est $\frac{r(0)}{\sin(\beta)}$.

The background of the slide is a repeating pattern of stylized ladybugs in orange, pink, and yellow, each with grey spots and antennae. Interspersed among the ladybugs are small grey dots of varying sizes. A light blue horizontal bar with a slight drop shadow is positioned across the middle of the slide.

4. Conclusion

Conclusion

Extensions des modèles et situations complexes

- Angle constant entre direction du chien et droite chien-maître
- Chien prédit la position du maître en $t + \Delta t$
- Vitesse variable du maître et/ou du chien (fonction du temps et/ou de la distance)
- Plusieurs proies
- Géométrie des poursuivants (polygones, figures quelconques, 3D) ... Simulations
- Problèmes de poursuite-évasion (stratégie du poursuivant vs de l'évadé), approche déterministe ou stochastique
- Problèmes de contrôle optimal (atteindre la cible le plus rapidement possible et/ou éviter le poursuivant le plus longtemps possible)

Calcul numérique appliqué aux courbes de poursuite

- Choix parmi plusieurs schémas numériques
- Simulations dans les cas complexes
- Solutions exactes : souvent partielles et limitées à des cas "simples" mais essentielles pour validation

Conclusion

Extensions des modèles et situations complexes

- Angle constant entre direction du chien et droite chien-maître
- Chien prédit la position du maître en $t + \Delta t$
- Vitesse variable du maître et/ou du chien (fonction du temps et/ou de la distance)
- Plusieurs proies
- Géométrie des poursuivants (polygones, figures quelconques, 3D) ... Simulations
- Problèmes de poursuite-évasion (stratégie du poursuivant vs de l'évadé), approche déterministe ou stochastique
- Problèmes de contrôle optimal (atteindre la cible le plus rapidement possible et/ou éviter le poursuivant le plus longtemps possible)

Calcul numérique appliqué aux courbes de poursuite

- Choix parmi plusieurs schémas numériques
- Simulations dans les cas complexes
- Solutions exactes : souvent partielles et limitées à des cas "simples" mais essentielles pour validation

Conclusion

Extensions des modèles et situations complexes

- Angle constant entre direction du chien et droite chien-maître
- Chien prédit la position du maître en $t + \Delta t$
- **Vitesse variable du maître et/ou du chien (fonction du temps et/ou de la distance)**
- Plusieurs proies
- Géométrie des poursuivants (polygones, figures quelconques, 3D) ... Simulations
- Problèmes de poursuite-évasion (stratégie du poursuivant vs de l'évadé), approche déterministe ou stochastique
- Problèmes de contrôle optimal (atteindre la cible le plus rapidement possible et/ou éviter le poursuivant le plus longtemps possible)

Calcul numérique appliqué aux courbes de poursuite

- Choix parmi plusieurs schémas numériques
- Simulations dans les cas complexes
- Solutions exactes : souvent partielles et limitées à des cas "simples" mais essentielles pour validation

Conclusion

Extensions des modèles et situations complexes

- Angle constant entre direction du chien et droite chien-maître
- Chien prédit la position du maître en $t + \Delta t$
- Vitesse variable du maître et/ou du chien (fonction du temps et/ou de la distance)
- **Plusieurs proies**
- Géométrie des poursuivants (polygones, figures quelconques, 3D) ... Simulations
- Problèmes de poursuite-évasion (stratégie du poursuivant vs de l'évadé), approche déterministe ou stochastique
- Problèmes de contrôle optimal (atteindre la cible le plus rapidement possible et/ou éviter le poursuivant le plus longtemps possible)

Calcul numérique appliqué aux courbes de poursuite

- Choix parmi plusieurs schémas numériques
- Simulations dans les cas complexes
- Solutions exactes : souvent partielles et limitées à des cas "simples" mais essentielles pour validation

Conclusion

Extensions des modèles et situations complexes

- Angle constant entre direction du chien et droite chien-maître
- Chien prédit la position du maître en $t + \Delta t$
- Vitesse variable du maître et/ou du chien (fonction du temps et/ou de la distance)
- Plusieurs proies
- **Géométrie des poursuivants (polygones, figures quelconques, 3D) ...** Simulations
- Problèmes de poursuite-évasion (stratégie du poursuivant vs de l'évadé), approche déterministe ou stochastique
- Problèmes de contrôle optimal (atteindre la cible le plus rapidement possible et/ou éviter le poursuivant le plus longtemps possible)

Calcul numérique appliqué aux courbes de poursuite

- Choix parmi plusieurs schémas numériques
- Simulations dans les cas complexes
- Solutions exactes : souvent partielles et limitées à des cas "simples" mais essentielles pour validation

Conclusion

Extensions des modèles et situations complexes

- Angle constant entre direction du chien et droite chien-maître
- Chien prédit la position du maître en $t + \Delta t$
- Vitesse variable du maître et/ou du chien (fonction du temps et/ou de la distance)
- Plusieurs proies
- Géométrie des poursuivants (polygones, figures quelconques, 3D) ... Simulations
- Problèmes de poursuite-évasion (stratégie du poursuivant vs de l'évadé), approche déterministe ou stochastique
- Problèmes de contrôle optimal (atteindre la cible le plus rapidement possible et/ou éviter le poursuivant le plus longtemps possible)

Calcul numérique appliqué aux courbes de poursuite

- Choix parmi plusieurs schémas numériques
- Simulations dans les cas complexes
- Solutions exactes : souvent partielles et limitées à des cas "simples" mais essentielles pour validation

Conclusion

Extensions des modèles et situations complexes

- Angle constant entre direction du chien et droite chien-maître
- Chien prédit la position du maître en $t + \Delta t$
- Vitesse variable du maître et/ou du chien (fonction du temps et/ou de la distance)
- Plusieurs proies
- Géométrie des poursuivants (polygones, figures quelconques, 3D) ... Simulations
- Problèmes de poursuite-évasion (stratégie du poursuivant vs de l'évadé), approche déterministe ou stochastique
- Problèmes de contrôle optimal (atteindre la cible le plus rapidement possible et/ou éviter le poursuivant le plus longtemps possible)

Calcul numérique appliqué aux courbes de poursuite

- Choix parmi plusieurs schémas numériques
- Simulations dans les cas complexes
- Solutions exactes : souvent partielles et limitées à des cas "simples" mais essentielles pour validation

Conclusion

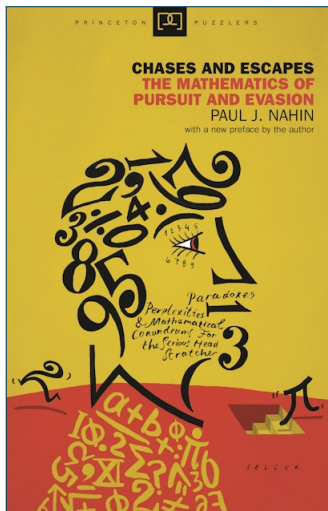
Extensions des modèles et situations complexes

- Angle constant entre direction du chien et droite chien-maître
- Chien prédit la position du maître en $t + \Delta t$
- Vitesse variable du maître et/ou du chien (fonction du temps et/ou de la distance)
- Plusieurs proies
- Géométrie des poursuivants (polygones, figures quelconques, 3D) ... Simulations
- Problèmes de poursuite-évasion (stratégie du poursuivant vs de l'évadé), approche déterministe ou stochastique
- Problèmes de contrôle optimal (atteindre la cible le plus rapidement possible et/ou éviter le poursuivant le plus longtemps possible)

Calcul numérique appliqué aux courbes de poursuite

- Choix parmi plusieurs schémas numériques
- Simulations dans les cas complexes
- Solutions exactes : souvent partielles et limitées à des cas "simples" mais essentielles pour validation

À lire



- Paul J. Nahin, *Chases and Escapes : The Mathematics of Pursuit and Evasion*, Princeton University Press, 2007.
- Antoine Hollard, *Courbes de poursuites mutuelles*, *Bulletin de l'Union des Professeurs des Classes Préparatoires Scientifiques*, 2021, pp. 53–68.
<https://mathcurve.com/poursuites%20hollard.pdf>
- Svetlana Moiseeva, *Optimal Control Problems Curves of Pursuit*, 2011. https://digitalrepository.unm.edu/math_etds/31

Webographie



Simulations :

<https://ressources.univ-lemans.fr/AccesLibre/UM/Pedago/physique/02/meca/poursuite.html>

<https://phyanim.sciences.univ-nantes.fr//Meca/Cinematique/4mouches.php>



MathCurve :

<https://www.mathcurve.com/courbes2d/poursuite/poursuite.shtml>

<https://www.mathcurve.com/courbes2d/poursuite/poursuitemutuelle.shtml>

<https://mathcurve.com/courbes2d/poursuite/poursuite.shtml>

<https://www.mathcurve.com/courbes3d/poursuite/poursuite.shtml>



MathWorld : <https://mathworld.wolfram.com/PursuitCurve.html> and

<https://mathworld.wolfram.com/MiceProblem.html>



MathPages : <https://www.mathpages.com/home/kmath492/kmath492.htm>



The Mathematical Tourist :

<https://mathtourist.blogspot.com/2020/12/pursuing-pursuit-curves.html>



StackExchange :

<https://math.stackexchange.com/questions/77652/4-bugs-chasing-each-other-differential-equation>

<https://math.stackexchange.com/questions/1492187/differential-equation-math-puzzle>

Simulations de la courbe du chien – cas 1

Simulations de la courbe du chien – cas 2

Simulations de la courbe du chien – cas 3

[Retour aux résultats](#)

Simulations poursuite mutuelle - polygone régulier

[Retour aux résultats](#)

Simulations poursuite mutuelle - quelconque

[Retour](#)