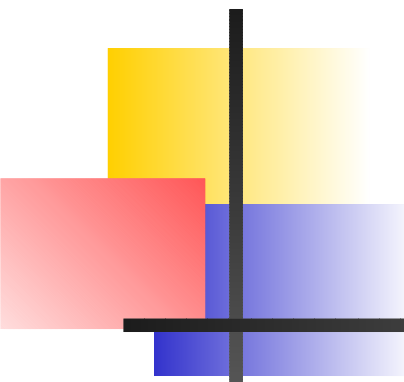
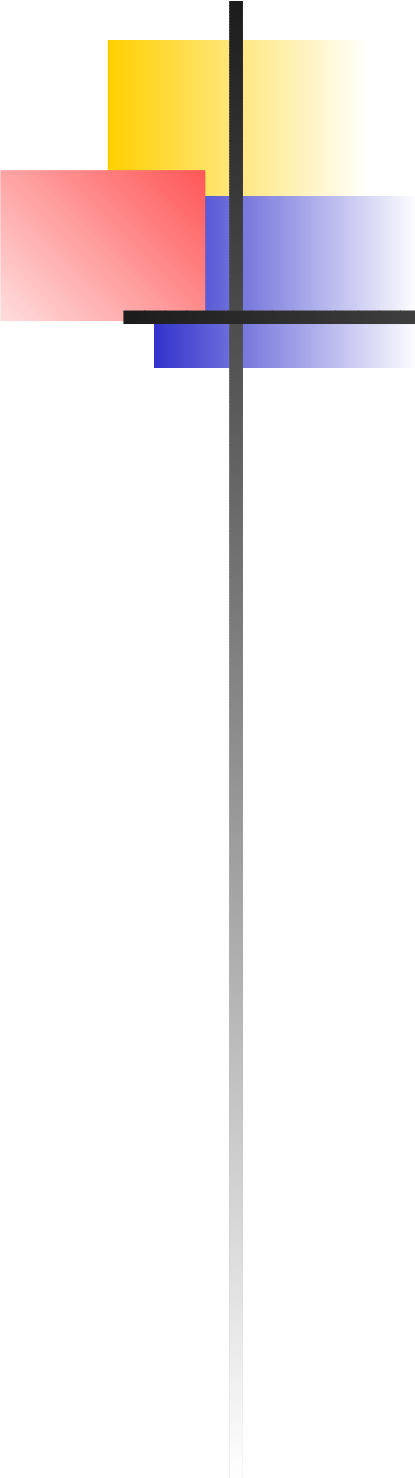




Algorithmes polynomiaux pour le calcul numérique prouvé de constantes

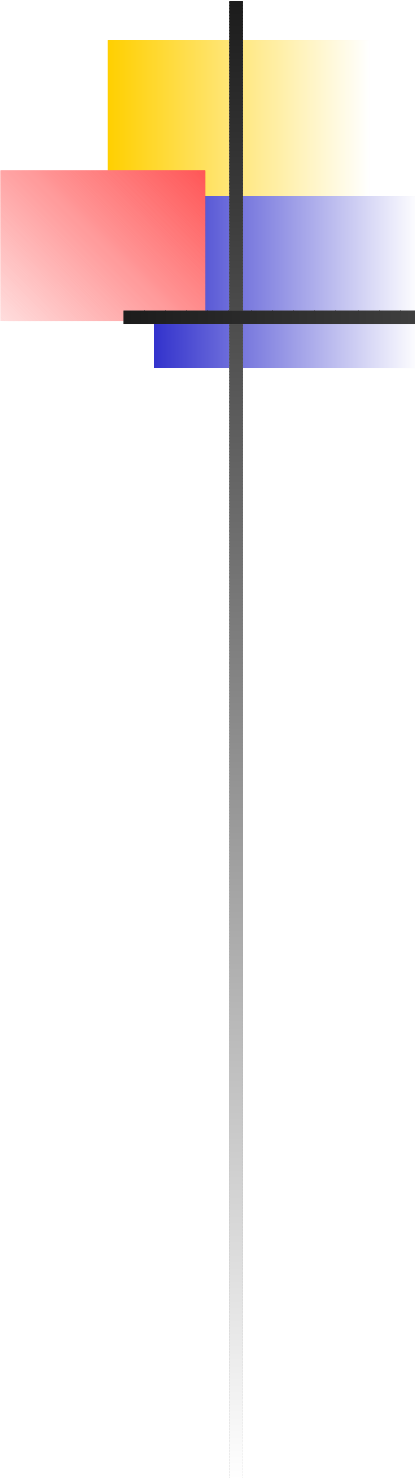
Loïck LHOTE et Brigitte VALLÉE

GREYC, université de Caen



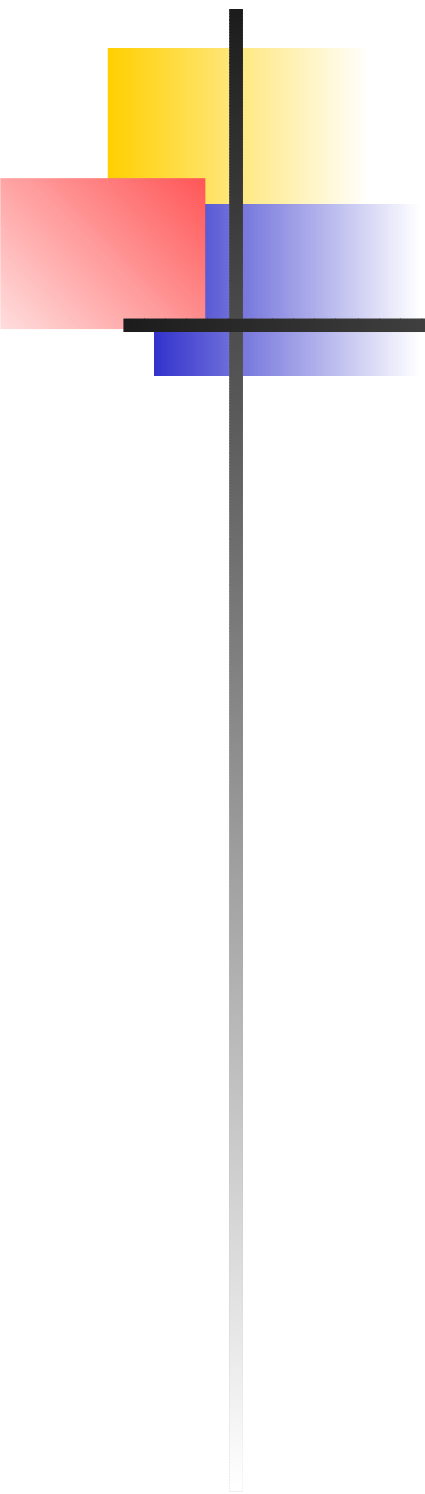
Analyse Dynamique

analyse classique



Analyse Dynamique

analyse Dynamique



Analyse Dynamique

analyse Dynamique+constantes

Systemes Dynamiques (SD)

- un intervalle I



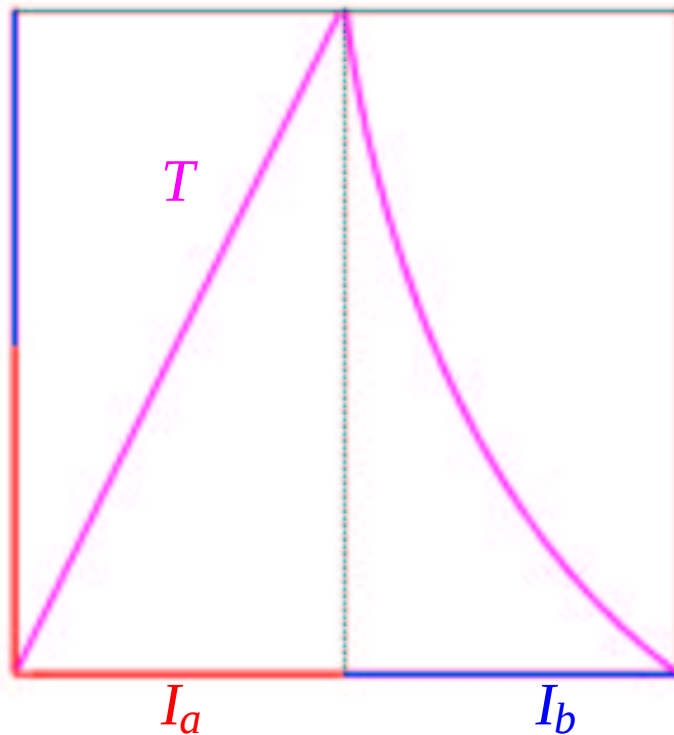
Systemes Dynamiques (SD)

- un intervalle I
- une partition $\mathcal{P} = (I_a)_{a \in \mathcal{A}}$ de I



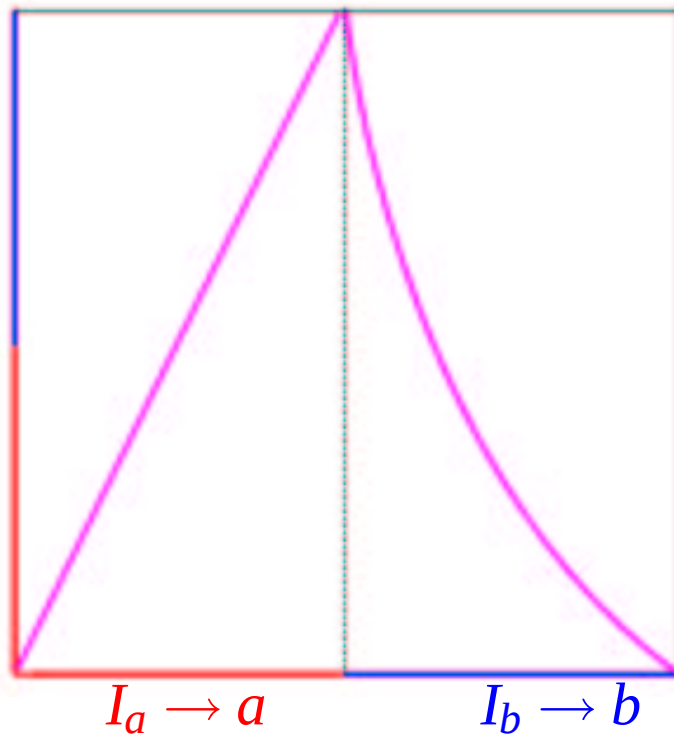
ex: $\mathcal{P} = (I_a, I_b)$

Systemes Dynamiques (SD)



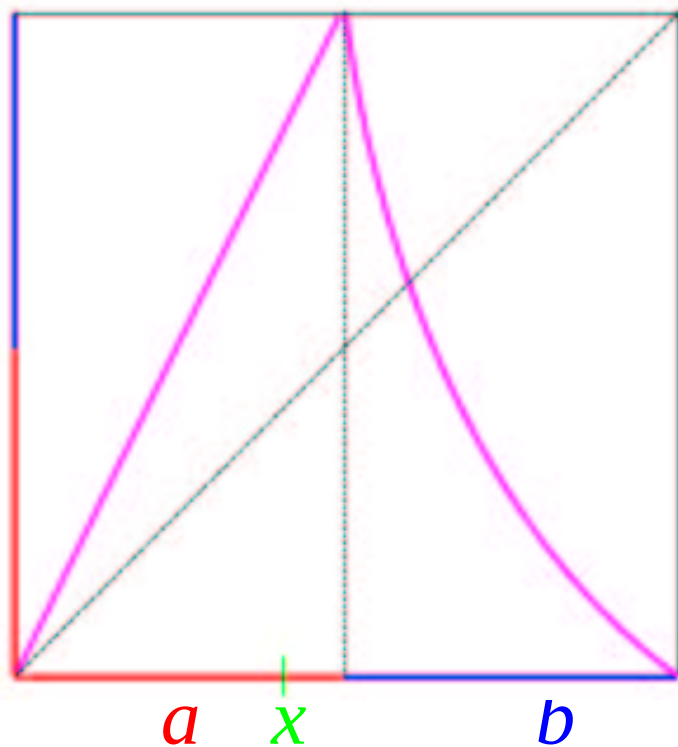
- un intervalle I
- une partition $\mathcal{P} = (I_a)_{a \in \mathcal{A}}$ de I
ex: $\mathcal{P} = (I_a, I_b)$
- une fonction T (ou shift)
surjective par morceaux
st. monotone par morceaux

Systemes Dynamiques (SD)



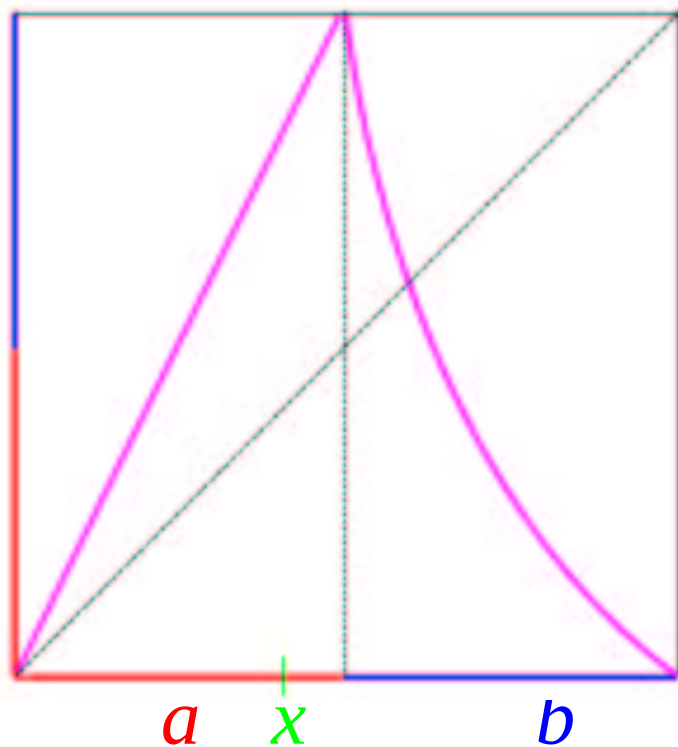
- un intervalle I
- une partition $\mathcal{P} = (I_a)_{a \in \mathcal{A}}$ de I
ex: $\mathcal{P} = (I_a, I_b)$
- une fonction T (ou shift)
surjective par morceaux
st. monotone par morceaux
- un codage de la partition
ex: $I_a \rightarrow a, \quad I_b \rightarrow b$

Mots



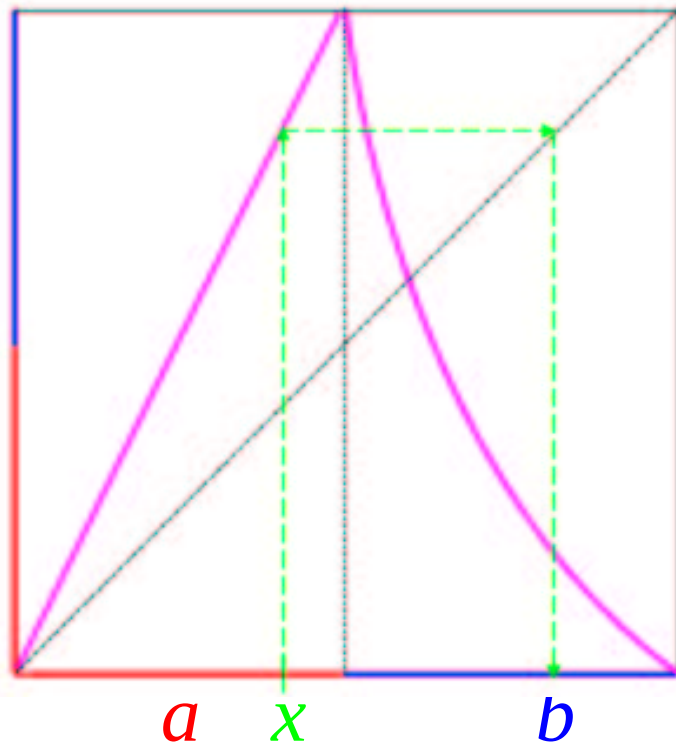
$$M(x) =$$

Mots



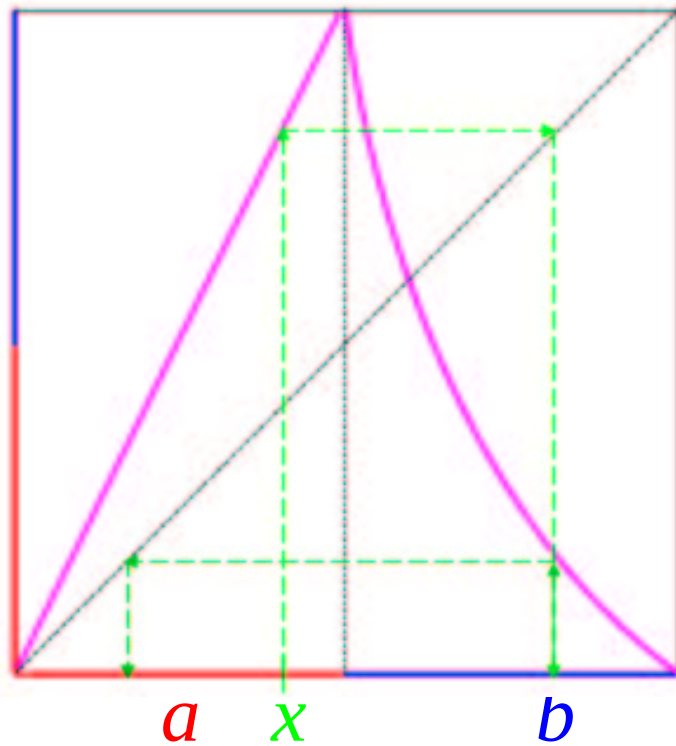
$$M(x) = a$$

Mots



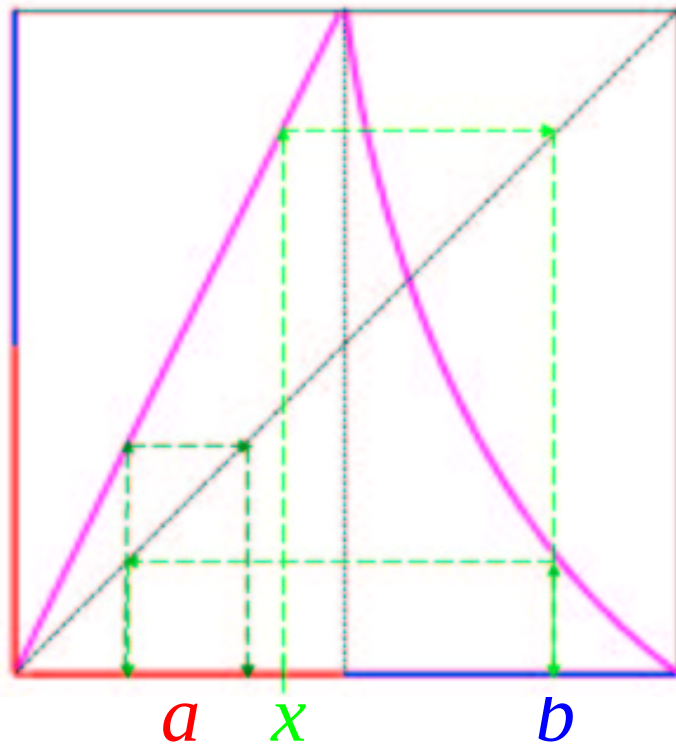
$$M(x) = a b$$

Mots



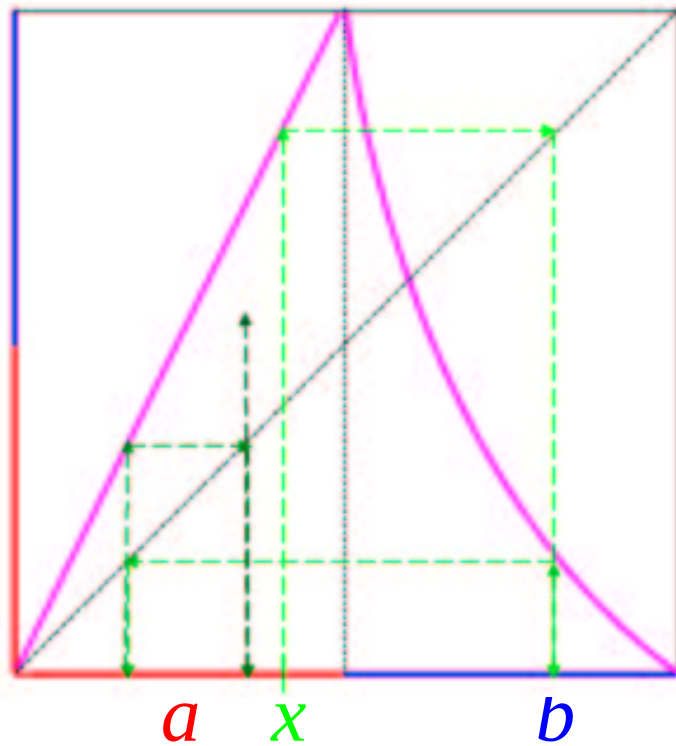
$$M(x) = a b a$$

Mots



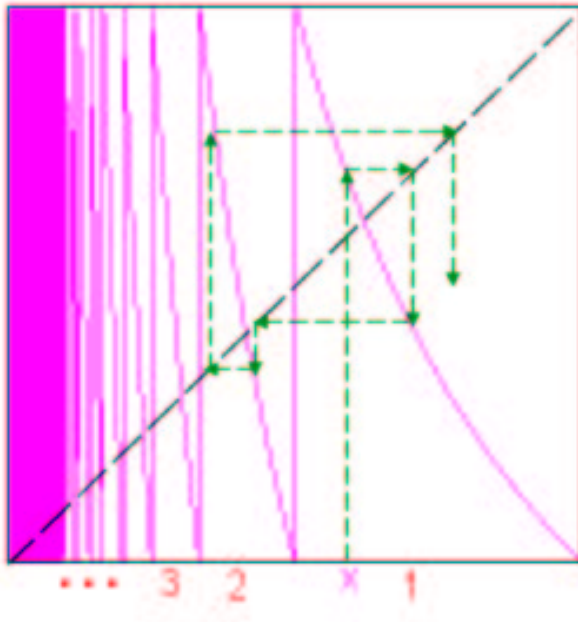
$$M(x) = a b a a$$

Mots



$$M(x) = a b a a \dots$$

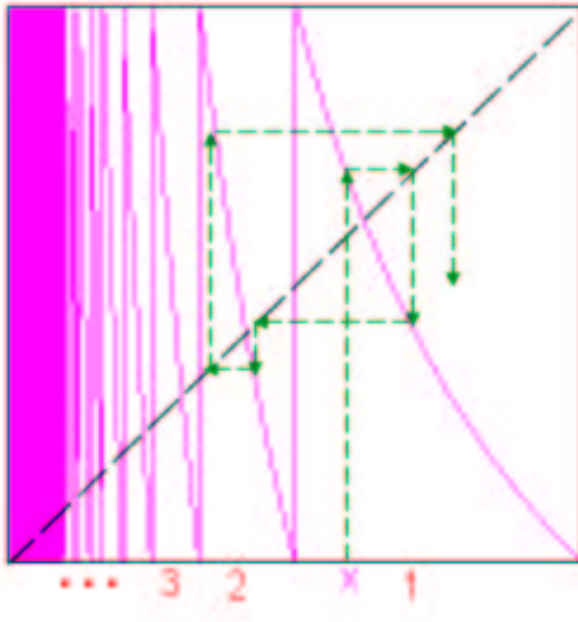
Ex: SD des Fractions Continues



$$M(\textcolor{blue}{x}) = 1\ 1\ 2\ 2\ 1\ \dots$$

- intervalle: $I = [0, 1]$
- partition: $I_j = \left[\frac{1}{j+1}, \frac{1}{j} \right]$
- shift: $T_{FC}(x) = \left\{ \frac{1}{x} \right\}$
- codage: $I_j \rightarrow j$

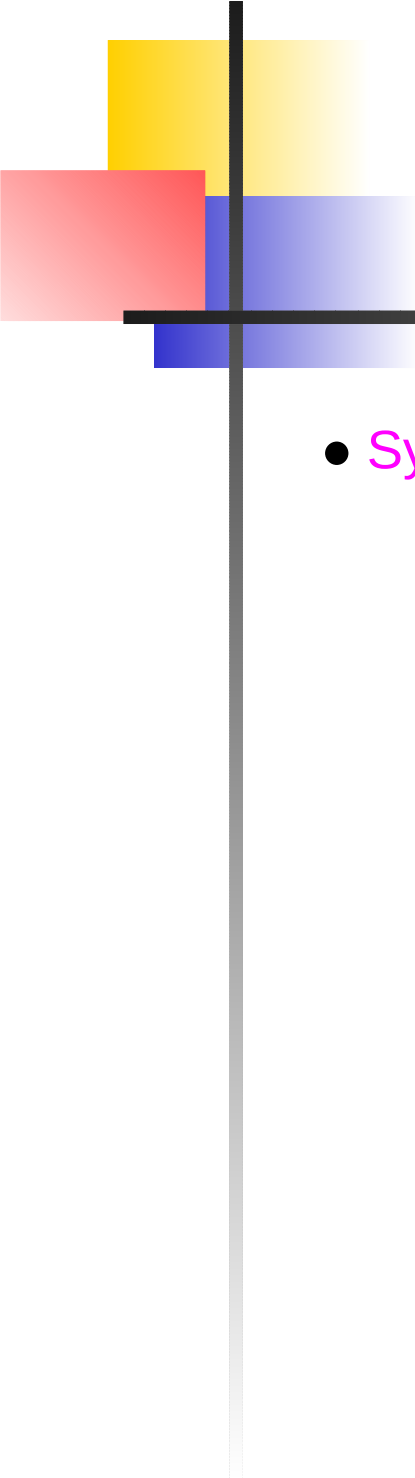
Ex: SD des Fractions Continues



$$M(x) = 1 \ 1 \ 2 \ 2 \ 1 \ \dots$$

- intervalle: $I = [0, 1]$
- partition: $I_j = \left[\frac{1}{j+1}, \frac{1}{j} \right]$
- shift: $T_{FC}(x) = \left\{ \frac{1}{x} \right\}$
- codage: $I_j \rightarrow j$

$$x = \frac{1}{1 + \frac{1}{1 + \frac{1}{2 + \frac{1}{2 + \frac{1}{1 + \dots}}}}}$$



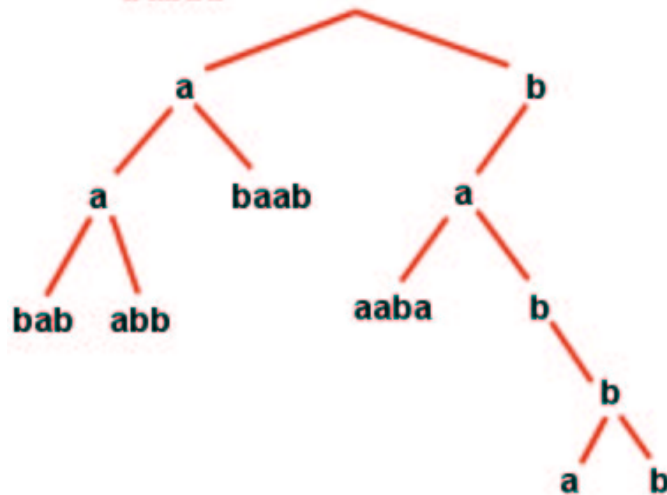
Exemple 1 d'utilisation

- **Système Dynamique**= source

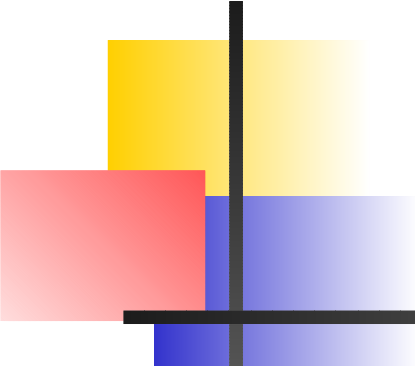
Exemple 1 d'utilisation

- **Système Dynamique**= source

mots: aabab, abaab, aaabb, baaba, babba, babbb



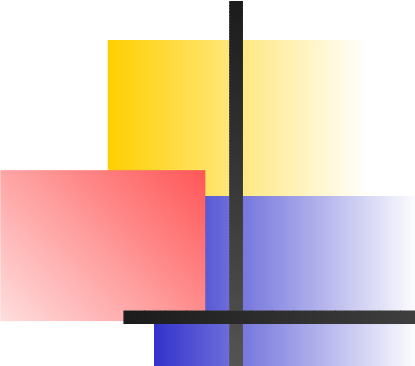
- paramètres de la source?
entropie, proba. de 2-coïncidence,...
- action sur les structures?
hauteur moyenne des tries,
longueur moyenne de cheminement,



Exemple 2 d'utilisation: algorithmes euclidiens

- Algorithme d'Euclide classique

Départ: (a_0, a_1) avec $a_0 \geq a_1 > 0$

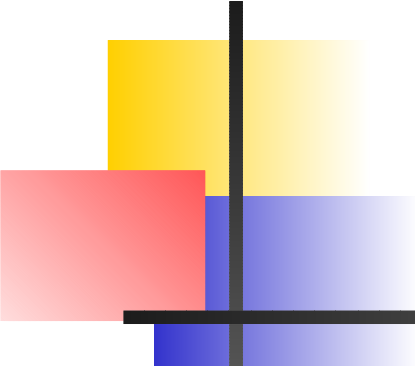


Exemple 2 d'utilisation: algorithmes euclidiens

- Algorithme d'Euclide classique

Départ: (a_0, a_1) avec $a_0 \geq a_1 > 0$

$$a_0 = a_1 q_1 + a_2$$



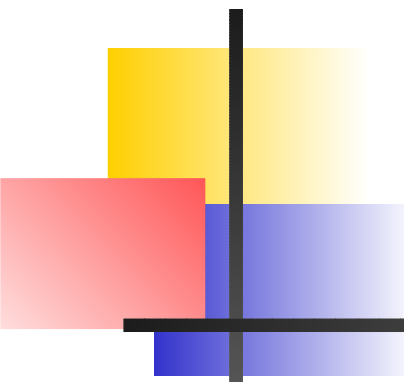
Exemple 2 d'utilisation: algorithmes euclidiens

- Algorithme d'Euclide classique

Départ: (a_0, a_1) avec $a_0 \geq a_1 > 0$

$$a_0 = a_1 q_1 + a_2$$

$$a_1 = a_2 q_2 + a_3$$



Exemple 2 d'utilisation: algorithmes euclidiens

- Algorithme d'Euclide classique

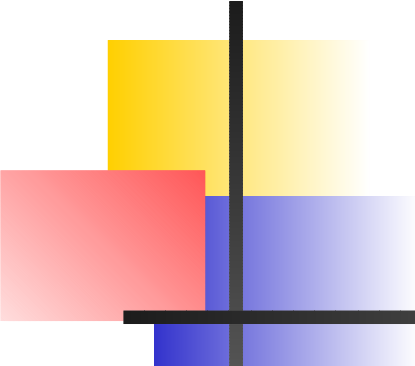
Départ: (a_0, a_1) avec $a_0 \geq a_1 > 0$

$$a_0 = a_1 q_1 + a_2$$

$$a_1 = a_2 q_2 + a_3$$

$$\vdots$$

$$a_{p-1} = a_p q_p + 0$$



Exemple 2 d'utilisation: algorithmes euclidiens

- Algorithme d'Euclide classique

Départ: (a_0, a_1) avec $a_0 \geq a_1 > 0$

$$a_0 = a_1 q_1 + a_2$$

$$a_1 = a_2 q_2 + a_3$$

$$\vdots$$

$$a_{p-1} = a_p q_p + 0$$

Retourner a_p

Exemple 2 d'utilisation: algorithmes euclidiens

- Algorithme d'Euclide classique

Départ: (a_0, a_1) avec $a_0 \geq a_1 > 0$

$$a_0 = a_1 q_1 + a_2$$

$$\rightarrow T_{FC}\left(\frac{a_1}{a_0}\right) = \frac{a_2}{a_1}$$

$$a_1 = a_2 q_2 + a_3$$

$$\vdots$$

$$a_{p-1} = a_p q_p + 0$$

Retourner a_p

Exemple 2 d'utilisation: algorithmes euclidiens

- Algorithme d'Euclide classique

Départ: (a_0, a_1) avec $a_0 \geq a_1 > 0$

$$a_0 = a_1 q_1 + a_2 \quad \rightarrow \quad T_{FC}\left(\frac{a_1}{a_0}\right) = \frac{a_2}{a_1}$$

$$a_1 = a_2 q_2 + a_3 \quad \rightarrow \quad T_{FC}\left(\frac{a_2}{a_1}\right) = \frac{a_3}{a_2}$$

$$\vdots$$

$$a_{p-1} = a_p q_p + 0$$

Retourner a_p

Exemple 2 d'utilisation: algorithmes euclidiens

- Algorithme d'Euclide classique

Départ: (a_0, a_1) avec $a_0 \geq a_1 > 0$

$$a_0 = a_1 q_1 + a_2$$

$$\rightarrow T_{FC}\left(\frac{a_1}{a_0}\right) = \frac{a_2}{a_1}$$

$$a_1 = a_2 q_2 + a_3$$

$$\rightarrow T_{FC}\left(\frac{a_2}{a_1}\right) = \frac{a_3}{a_2}$$

$$\vdots$$
$$\vdots$$

$$a_{p-1} = a_p q_p + 0$$

$$\rightarrow T_{FC}\left(\frac{a_p}{a_{p-1}}\right) = 0$$

Retourner a_p

Exemple 2 d'utilisation: algorithmes euclidiens

- Algorithme d'Euclide classique

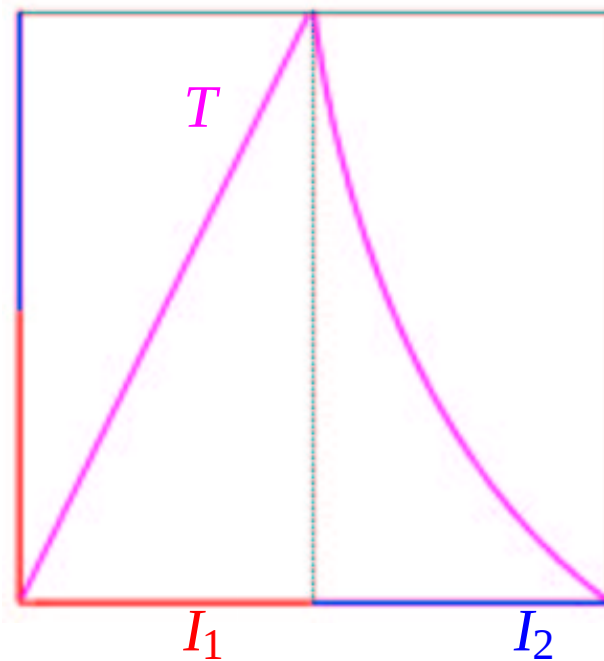
Départ: (a_0, a_1) avec $a_0 \geq a_1 > 0$

$$\begin{array}{ll} a_0 = a_1 q_1 + a_2 & \rightarrow T_{FC}\left(\frac{a_1}{a_0}\right) = \frac{a_2}{a_1} \\ a_1 = a_2 q_2 + a_3 & \rightarrow T_{FC}\left(\frac{a_2}{a_1}\right) = \frac{a_3}{a_2} \\ \vdots & \vdots \\ a_{p-1} = a_p q_p + 0 & \rightarrow T_{FC}\left(\frac{a_p}{a_{p-1}}\right) = 0 \end{array}$$

Retourner a_p

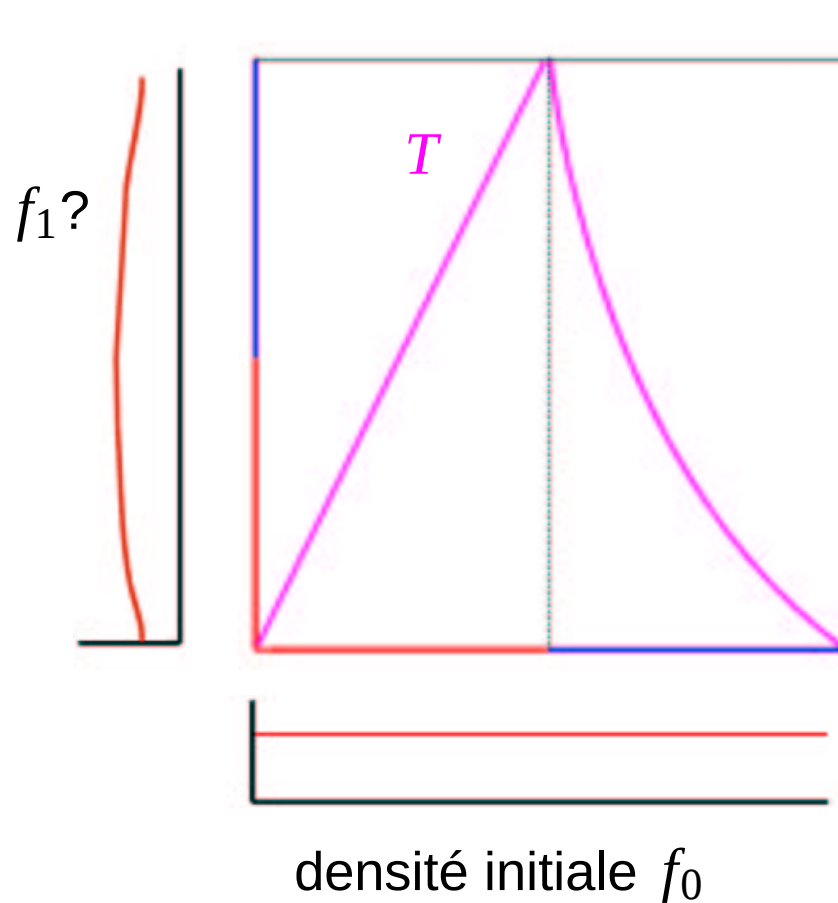
→ se généralise à tous les algorithmes euclidiens

opérateurs de transfert



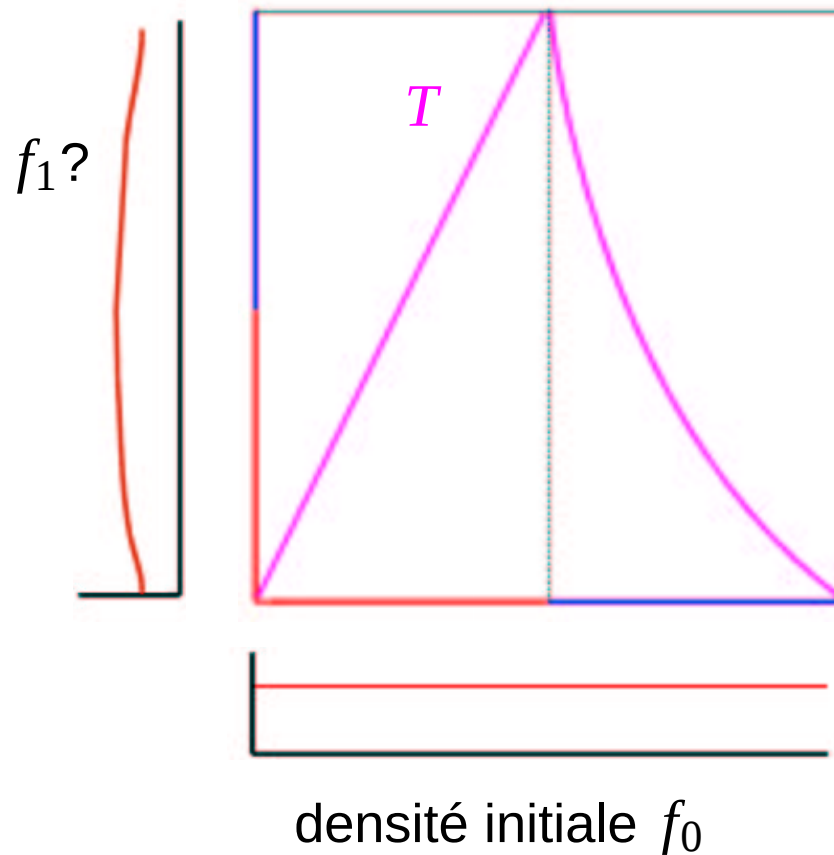
- notation: $h_m = T|_{I_m}^{-1}$

opérateurs de transfert



- notation: $h_m = T_{|I_m}^{-1}$

opérateurs de transfert

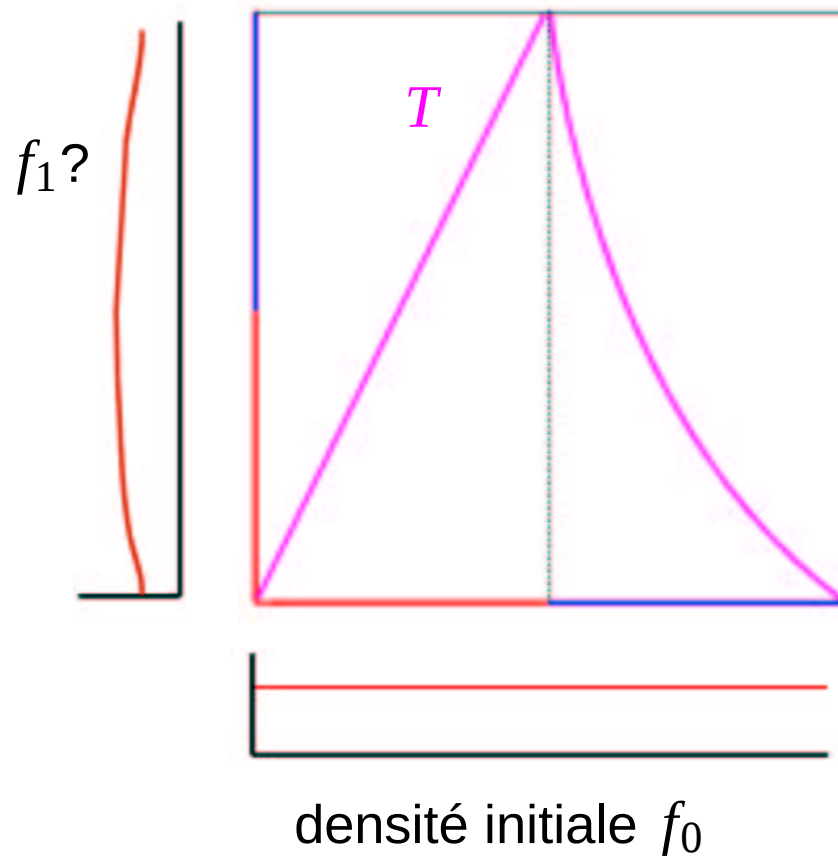


- notation: $h_m = T_{|I_m}^{-1}$

transformateur de densité

$$f_1 = \mathbf{G}[f_0] = \sum_{m \in \mathcal{A}} |h'_m| f_0 \circ h_m$$

opérateurs de transfert



- notation: $h_m = T_{|I_m}^{-1}$

transformateur de densité

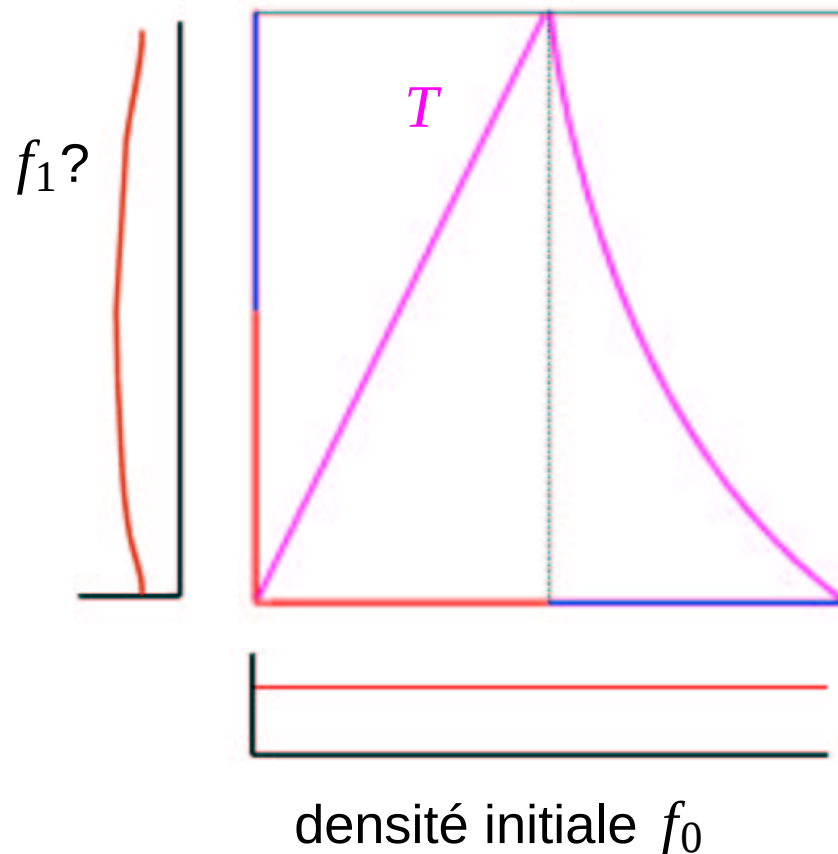
$$f_1 = \mathbf{G}[f_0] = \sum_{m \in \mathcal{A}} |h'_m| f_0 \circ h_m$$

perturbation

opérateurs de transfert

$$\mathbf{G}_s[f] = \sum_{m \in \mathcal{A}} |h'_m|^s f \circ h_m$$

opérateurs de transfert



• notation: $h_m = T_{|I_m}^{-1}$

transformateur de densité

$$f_1 = \mathbf{G}[f_0] = \sum_{m \in \mathcal{A}} |h'_m| f_0 \circ h_m$$

perturbation

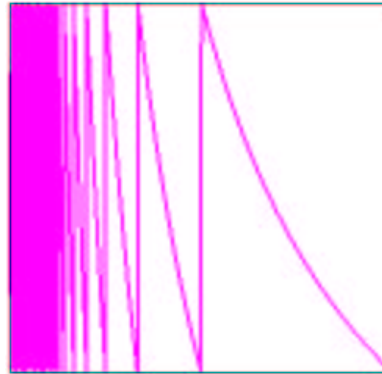
opérateurs de transfert

$$\mathbf{G}_s[f] = \sum_{m \in \mathcal{A}} |h'_m|^s f \circ h_m$$

généralisation

opérateurs contraints

exemples operateurs de transfert



- Branches inverses:

$$h_m(x) = \frac{1}{m+x}, \quad m > 0, x \in [0, 1].$$

- Opérateur de transfert

$$\mathbf{G}_s[f](x) = \sum_{m>0} \frac{1}{(m+x)^{2s}} f\left(\frac{1}{m+x}\right)$$

- Opérateur de transfert contraint

$$\mathbf{G}_s[f](x) = \sum_{m \text{ pair}} \frac{1}{(m+x)^{2s}} f\left(\frac{1}{m+x}\right)$$

Utilisation des opérateurs de transfert

Exemple pour les algorithmes euclidiens:

- Série de Dirichlet de coût:

$$F_C(s) = \sum_{(u,v) \in \Omega} \frac{C(u,v)}{v^s} = \sum_{n \geq 1} \frac{x_n}{n^s}$$

- Pour $C(u,v) = 1$

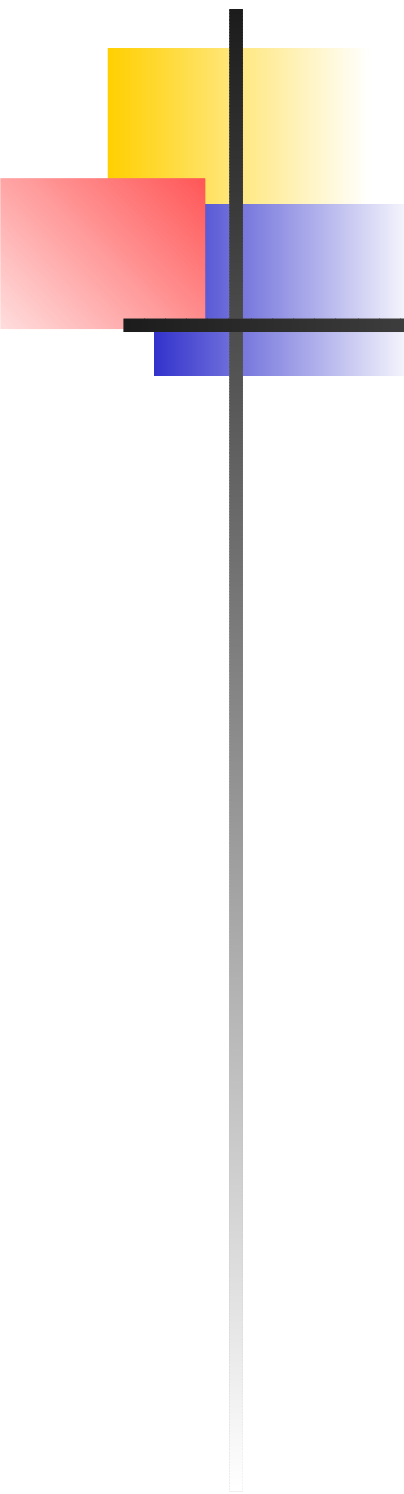
$$F_C(s) = \sum_{(u,v) \in \Omega} \frac{1}{v^s} = (\mathbf{I} - \mathbf{G}_{s/2})^{-1}[1](0)$$

singularité simple en $s = 2 \Rightarrow x_n = O(n(\log n)^0)$ (en fait $= n - 1$)

- Pour $C(u,v) = \text{coût en bits sur l'entrée } (u,v)$

$$F_C(s) = (\mathbf{I} - \mathbf{G}_{s/2})^{-1} \circ \dots \circ (\mathbf{I} - \mathbf{G}_{s/2})^{-1} \circ \dots \circ (\mathbf{I} - \mathbf{G}_{s/2})^{-1}[1](0)$$

singularité triple en $s = 2 \Rightarrow x_n = O(n(\log n)^2)$

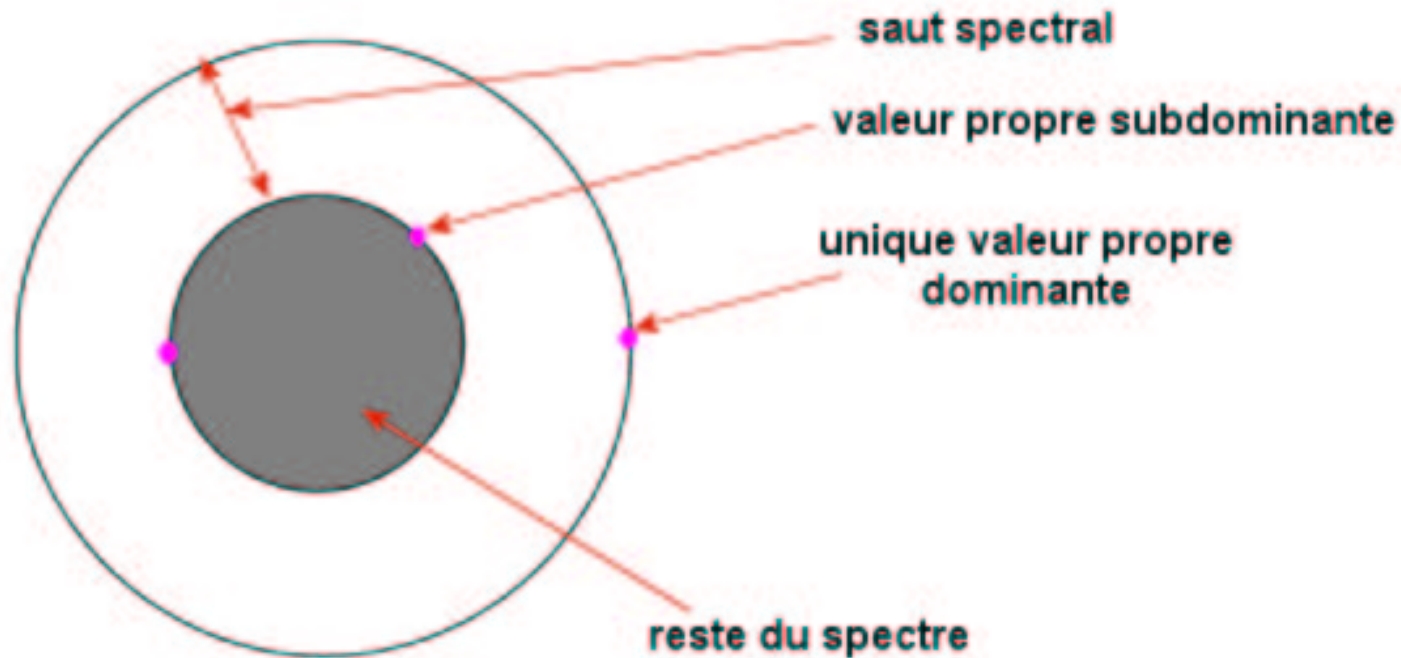


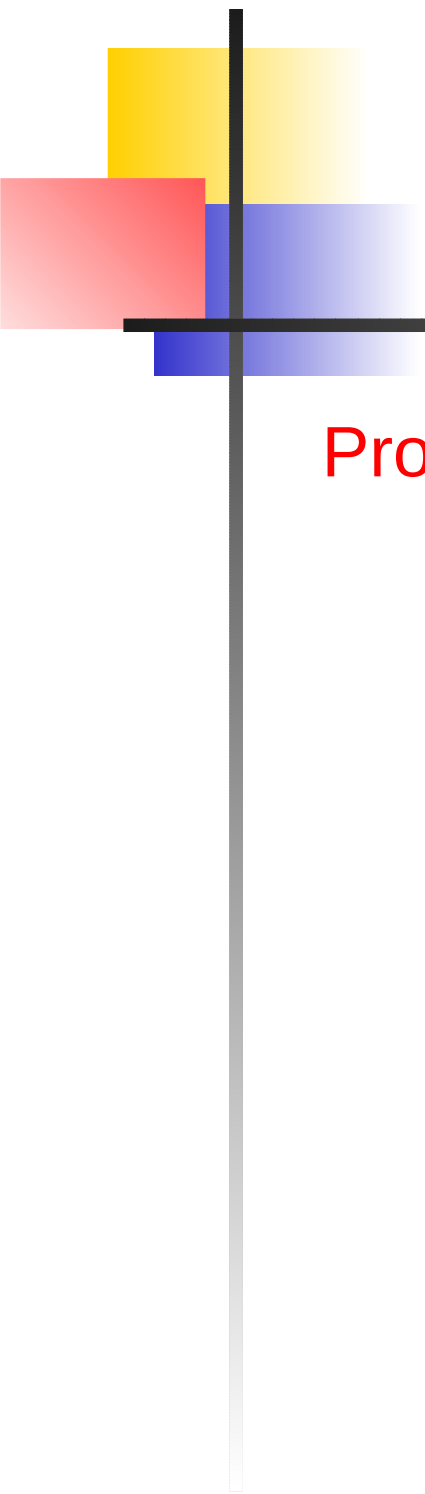
Analyse Dynamique

analyse Dynamique+constantes

Spectre des opérateurs de transfert

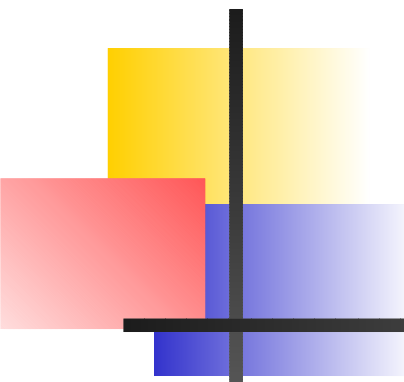
- spectre des opérateurs de transfert (sur un espace fonctionnel convenable):





Objectif

Proposer une méthode générale pour calculer $\lambda(s)$.

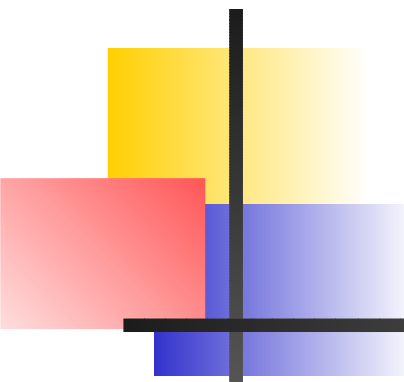


Objectif

Proposer une méthode générale pour calculer $\lambda(s)$.

Application au calcul de trois types de constantes:

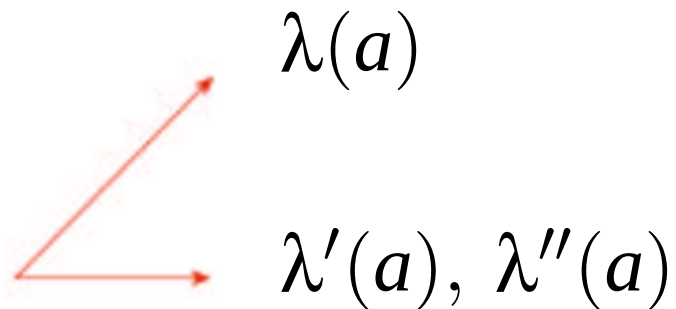

$$\lambda(a)$$

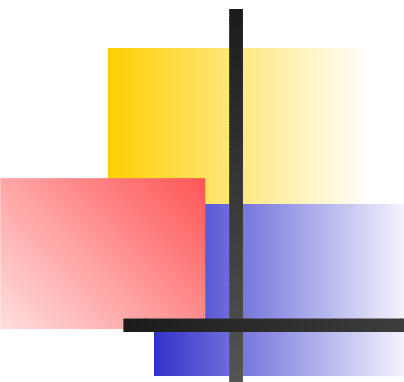


Objectif

Proposer une méthode générale pour calculer $\lambda(s)$.

Application au calcul de trois types de constantes:

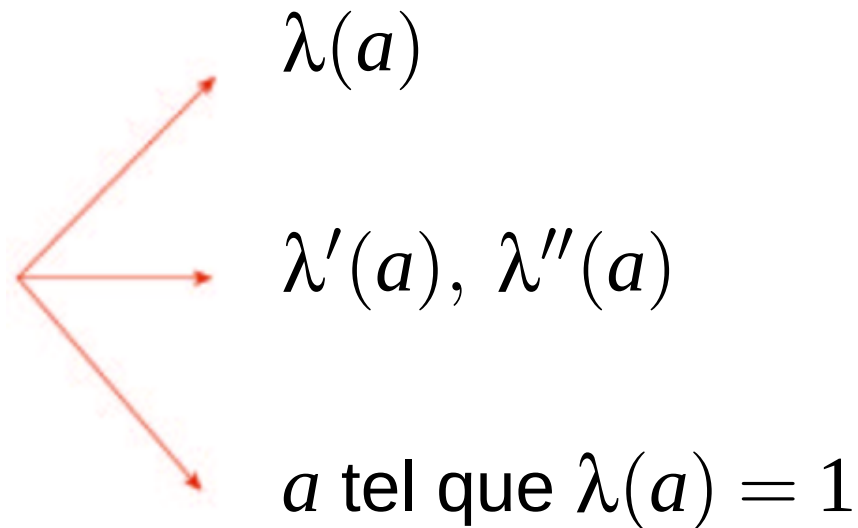


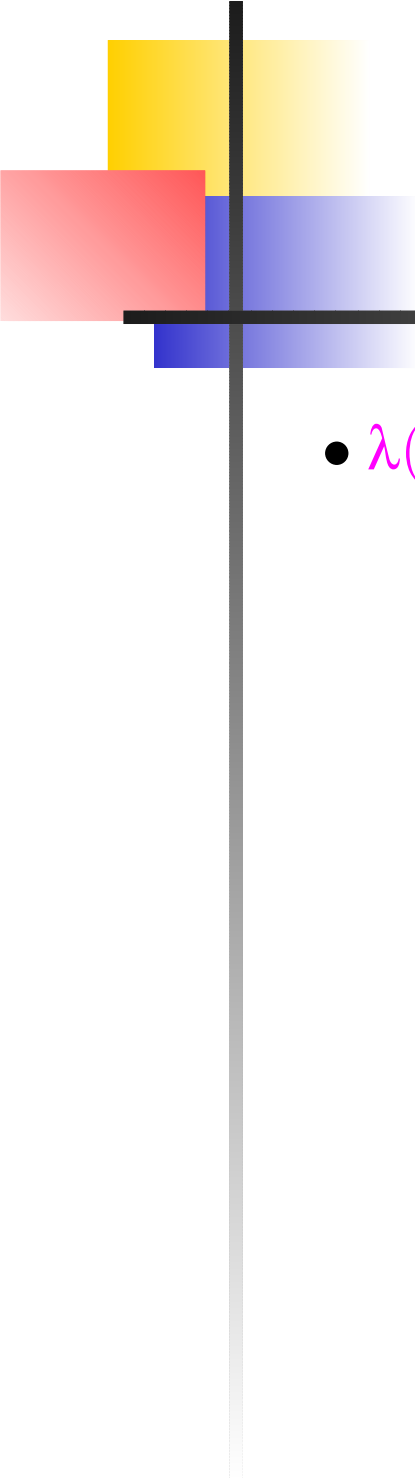


Objectif

Proposer une méthode générale pour calculer $\lambda(s)$.

Application au calcul de trois types de constantes:





constantes de type $\lambda(a)$

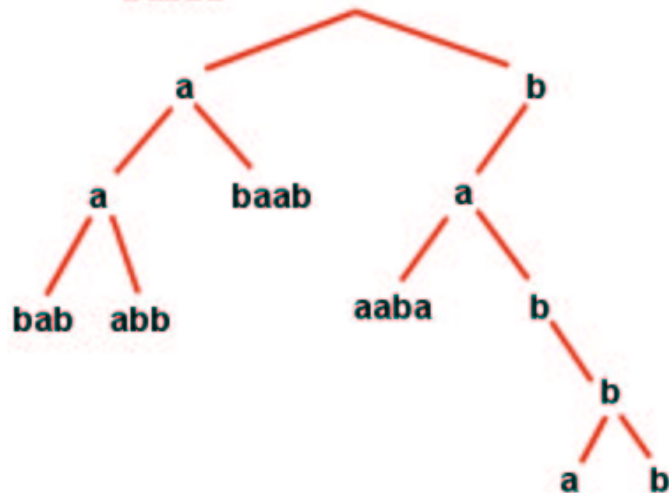
- $\lambda(2)$ = probabilité de 2-coïncidence
→ liée à la probabilité que deux mots commencent par le même préfixe

constantes de type $\lambda(a)$

- $\lambda(2)$ = probabilité de 2-coïncidence
→ liée à la probabilité que deux mots commencent par le même préfixe

exemple: structure de trie → code les dictionnaires

mots: aabab, abaab, aaabb, baaba, babba, babbb

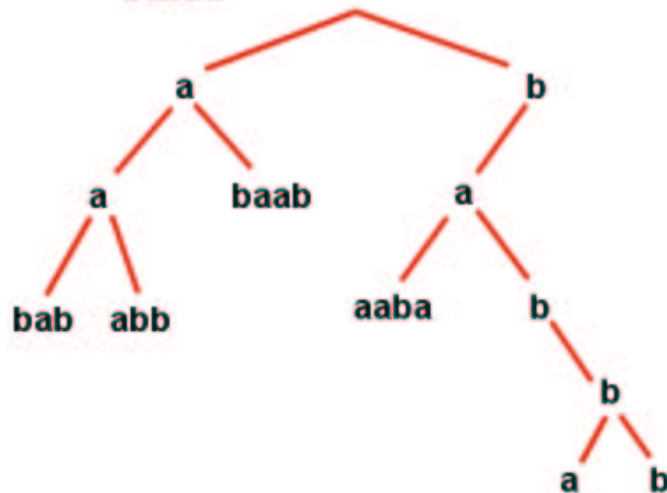


constantes de type $\lambda(a)$

- $\lambda(2)$ = probabilité de 2-coïncidence
→ liée à la probabilité que deux mots commencent par le même préfixe

structure de trie: code les dictionnaires

mots: aabab, abaab, aaabb, baaba, babba, babbb



- hauteur moyenne $\approx$

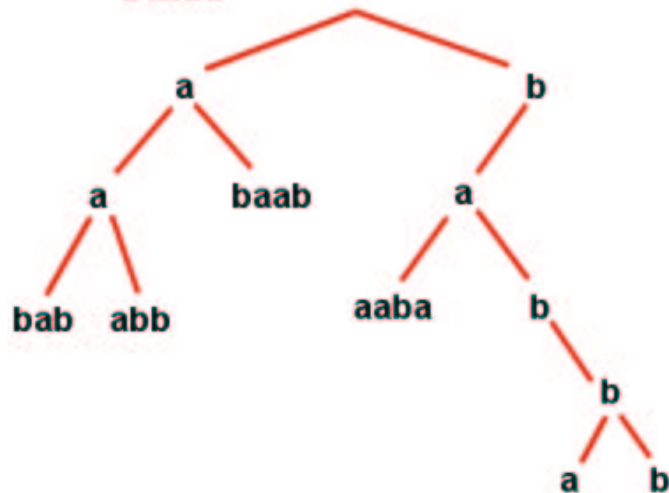
$$2\log n / |\log \lambda(2)|$$

constantes de type $\lambda(a)$

- $\lambda(2)$ = probabilité de 2-coïncidence
→ liée à la probabilité que deux mots commencent par le même préfixe

structure de trie: code les dictionnaires

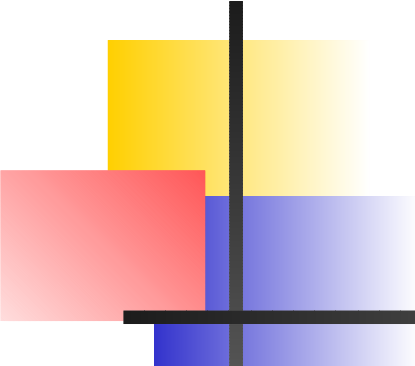
mots: aabab, abaab, aaabb, baaba, babba, babbb



- hauteur moyenne $\approx$

$$2 \log n / |\log \lambda(2)|$$

- Algorithme de Gauss (LLL)
- $\lambda(r)$ → comparaison de deux nombres



constantes de type $\lambda'(a), \lambda''(a)$

- $-\lambda'(1)$ = entropie de la source (du SD)
→ mesure l'incertitude de la source

“la source qui produit a ou b uniformement est plus incertaine que la source qui produit a avec proba 1/4 et b avec proba 3/4”

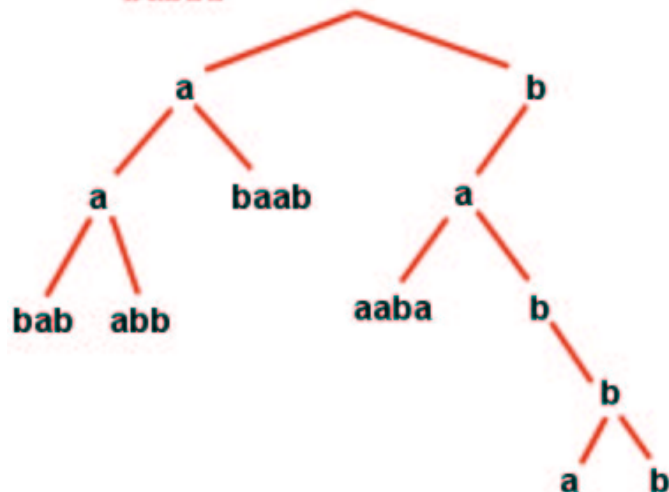
constantes de type

$\lambda'(a), \lambda''(a)$

- $-\lambda'(1)$ = entropie de la source (du SD)
→ mesure l'incertitude de la source

“la source qui produit a ou b uniformement est plus incertaine que la source qui produit a avec proba 1/4 et b avec proba 3/4”

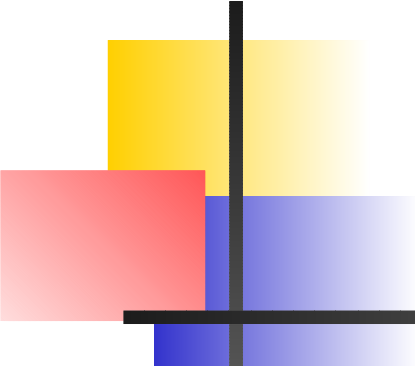
mots: aabab, abaab, aaabb, baaba, babba, babbb



influence sur les tries:

- longueur moyenne de cheminement $\approx$

$$-(n \log n) / \lambda'(1)$$



constantes de type $\lambda'(a), \lambda''(a)$

Algorithme d'Euclide classique:

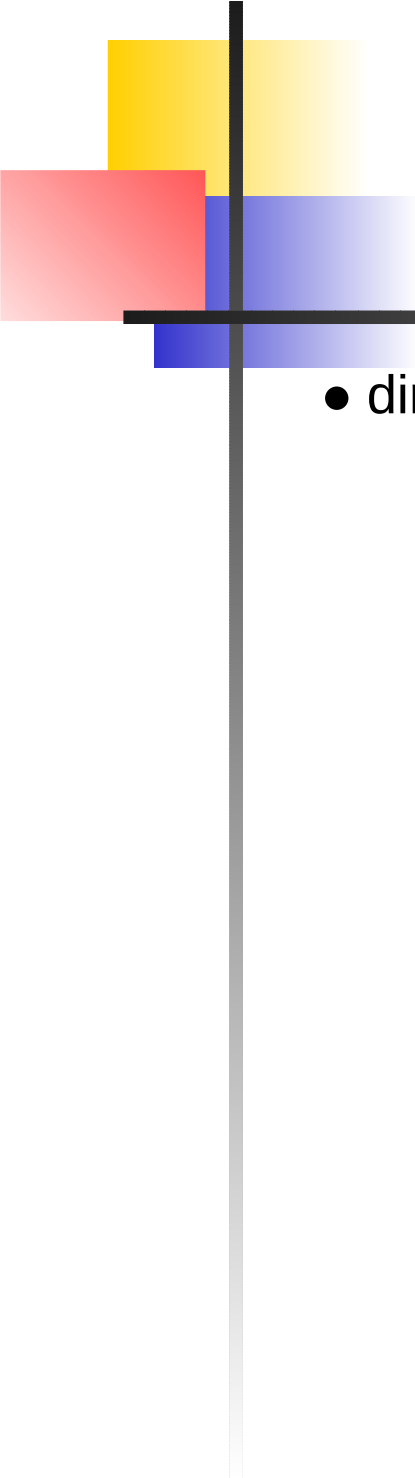
- Le nombre d'étapes est asymptotiquement gaussien
- Sur des entrées (u, v) telles que $0 < u < v \leq N$, on a
 - Espérance:

$$E[X_N] \approx -\frac{1}{\lambda'(1)} \log(N)$$

- Variance:

$$\text{Var}[X_N] \approx c_H \log(N) = -2 \frac{\lambda''(1) - \lambda'(1)^2}{\lambda'(1)^3}$$

avec c_H la constante de Hensley.



constantes de type a tq

$$\lambda(a) = 1$$

- dimension de Hausdorff d'ensemble de Cantor

constantes de type a tq

$$\lambda(a) = 1$$

- dimension de Hausdorff d'ensemble de Cantor
ex: ensemble triadique de Cantor



constantes de type a tq

$$\lambda(a) = 1$$

- dimension de Hausdorff d'ensemble de Cantor

ex: ensemble triadique de Cantor



1 intervalle de longueur 1

2 intervalles de longueur $1/3$

4 intervalles de longueur $1/9$

$\vdots$

2^n intervalles de longueur $(1/3)^n$

constantes de type a tq

$$\lambda(a) = 1$$

- dimension de Hausdorff d'ensemble de Cantor

ex: ensemble triadique de Cantor



1 intervalle de longueur 1

2 intervalles de longueur $1/3$

4 intervalles de longueur $1/9$

$\vdots$

2^n intervalles de longueur $(1/3)^n$

- compact, non dénombrable mais

$$\text{mesure} = \lim 2^n \times \left(\frac{1}{3}\right)^n = 0$$

constantes de type a tq

$$\lambda(a) = 1$$

- dimension de Hausdorff d'ensemble de Cantor

ex: ensemble triadique de Cantor



1 intervalle de longueur 1

2 intervalles de longueur $1/3$

4 intervalles de longueur $1/9$

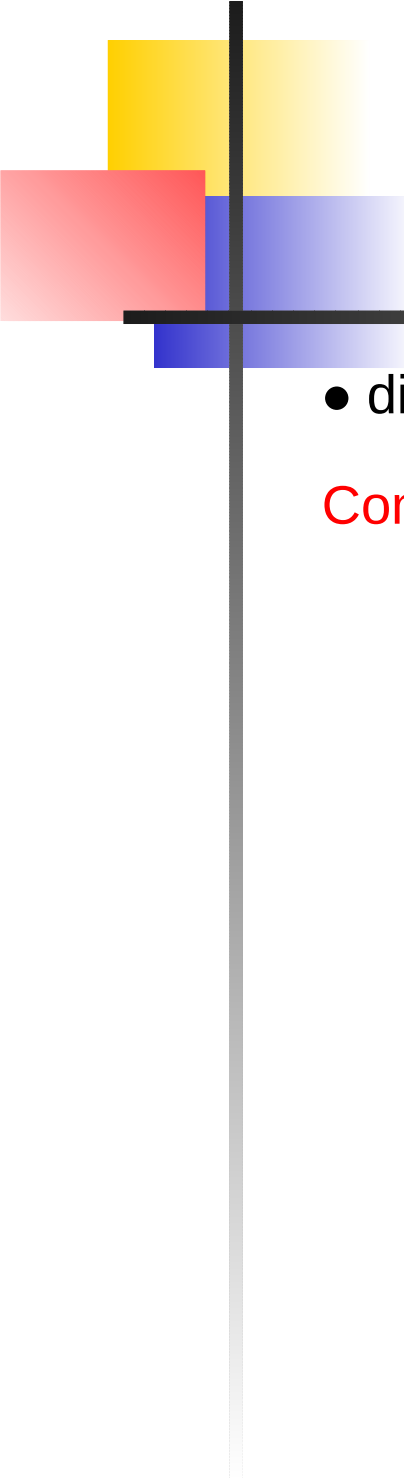
$\vdots$

2^n intervalles de longueur $(1/3)^n$

- compact, non dénombrable mais

$$\text{mesure} = \lim 2^n \times \left(\frac{1}{3}\right)^n = 0$$

Comment mesurer cet ensemble?



constantes de type a tq

$$\lambda(a) = 1$$

- dimension de Hausdorff d'ensemble de Cantor

Comment mesurer cet ensemble?

constantes de type *a tq*

$$\lambda(a) = 1$$

- dimension de Hausdorff d'ensemble de Cantor

Comment mesurer cet ensemble?

$$\text{mesure} = \lim 2^n \times \left(\frac{1}{3}\right)^n$$

constantes de type *a tq*

$$\lambda(a) = 1$$

- dimension de Hausdorff d'ensemble de Cantor

Comment mesurer cet ensemble?

$$\text{mesure} = \lim 2^n \times \left(\frac{1}{3}\right)^{n^s}$$

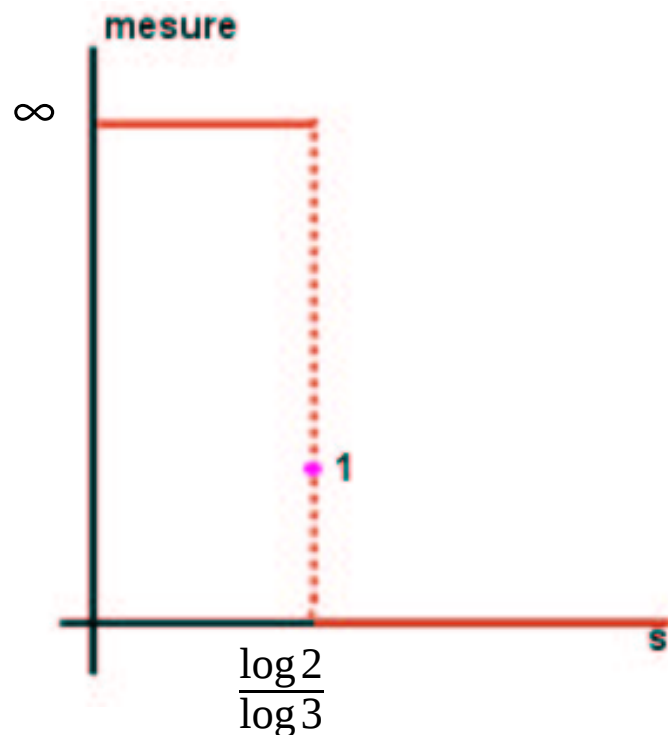
constantes de type a tq

$$\lambda(a) = 1$$

- dimension de Hausdorff d'ensemble de Cantor

Comment mesurer cet ensemble?

$$\text{mesure} = \lim 2^n \times \left(\frac{1}{3}\right)^{ns}$$



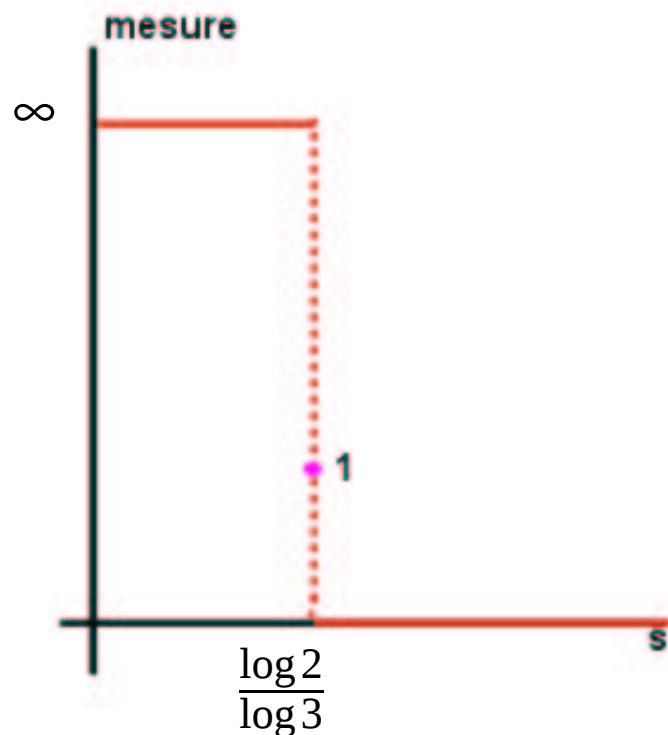
constantes de type a tq

$$\lambda(a) = 1$$

- dimension de Hausdorff d'ensemble de Cantor

Comment mesurer cet ensemble?

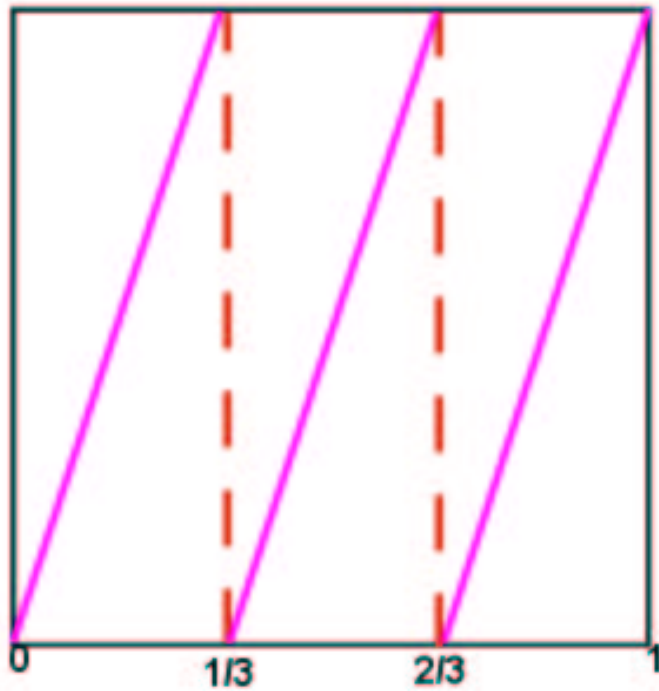
$$\text{mesure} = \lim 2^n \times \left(\frac{1}{3}\right)^{ns}$$



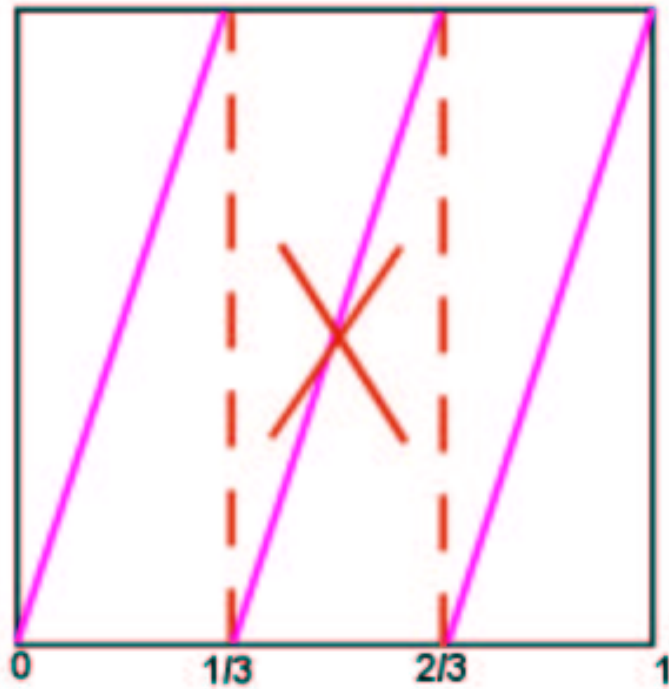
$$\text{mesure} = 1$$

$$\text{dimension de Hausdorff} = \frac{\log 2}{\log 3}$$

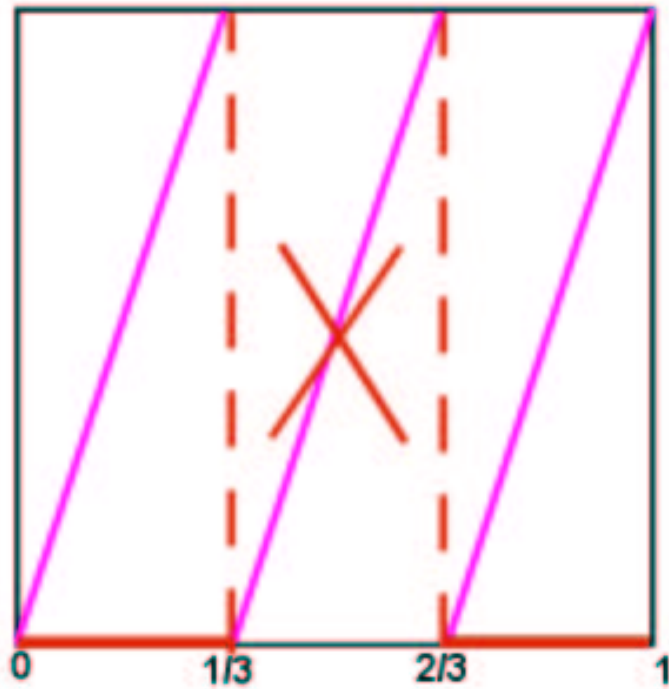
lien avec les SD



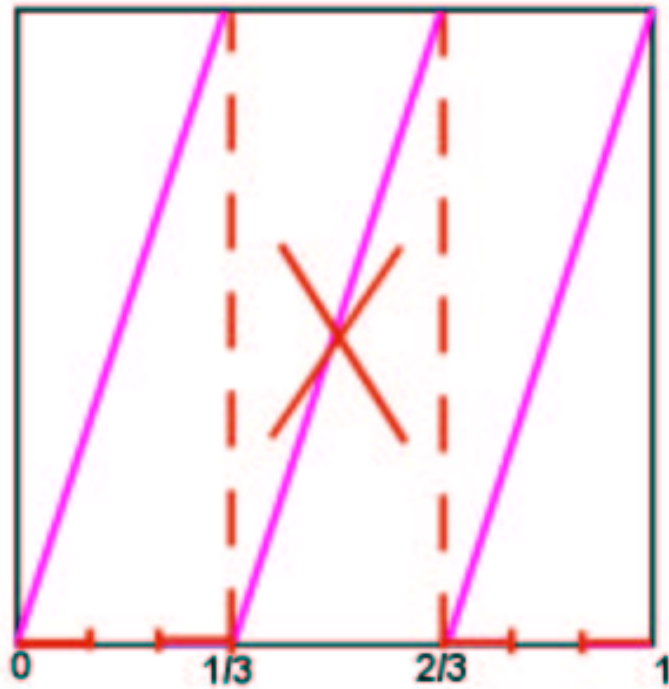
lien avec les SD



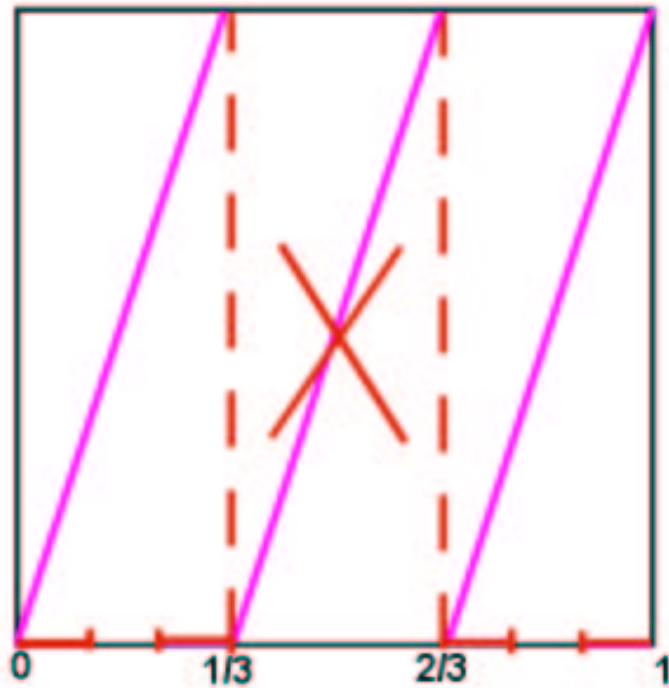
lien avec les SD



lien avec les SD



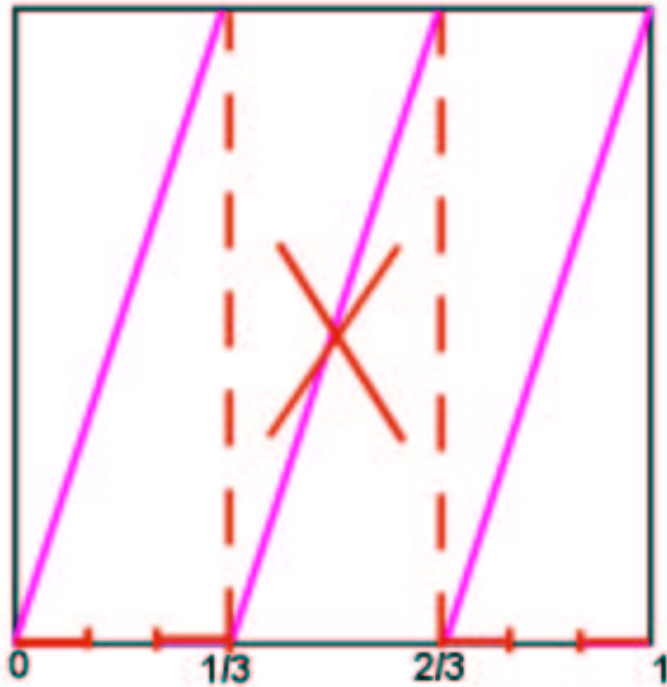
lien avec les SD



- opérateur de transfert

$$\mathbf{G}_s[f] = \frac{1}{3^s} f\left(\frac{x}{3}\right) + \frac{1}{3^s} f\left(\frac{x-1}{3}\right) + \frac{1}{3^s} f\left(\frac{x-2}{3}\right)$$

lien avec les SD



- opérateur de transfert

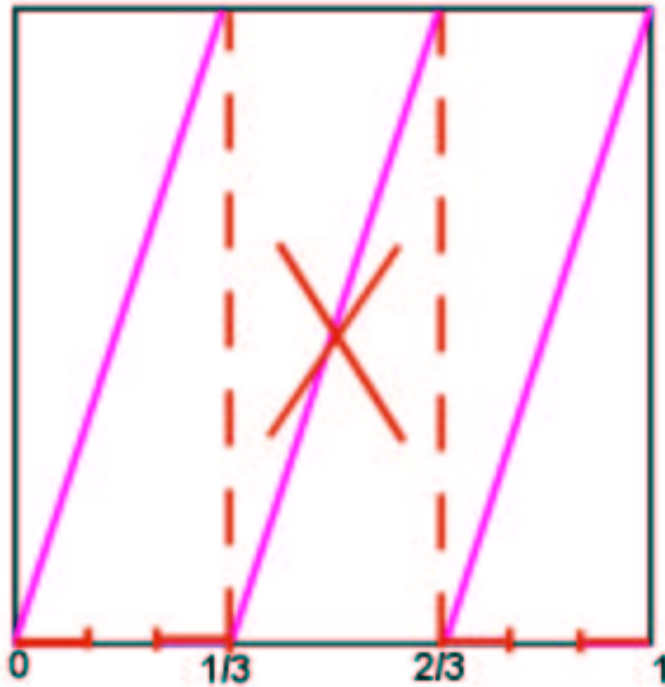
$$\mathbf{G}_s[f] = \frac{1}{3^s} f\left(\frac{x}{3}\right) + \frac{1}{3^s} f\left(\frac{x-1}{3}\right) + \frac{1}{3^s} f\left(\frac{x-2}{3}\right)$$



- opérateur de transfert contraint

$$\mathbf{G}_s[f] = \frac{1}{3^s} f\left(\frac{x}{3}\right) + \quad \quad \quad + \frac{1}{3^s} f\left(\frac{x-2}{3}\right)$$

lien avec les SD



- opérateur de transfert

$$\mathbf{G}_s[f] = \frac{1}{3^s} f\left(\frac{x}{3}\right) + \frac{1}{3^s} f\left(\frac{x-1}{3}\right) + \frac{1}{3^s} f\left(\frac{x-2}{3}\right)$$



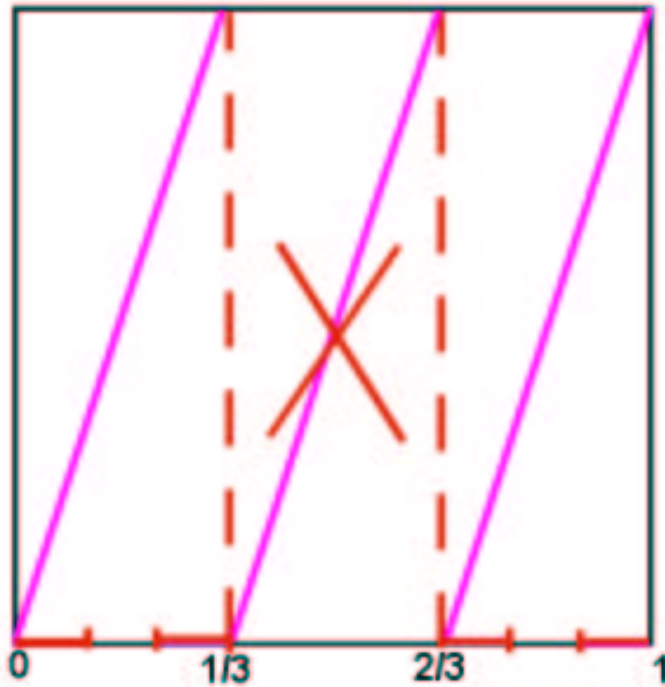
- opérateur de transfert contraint

$$\mathbf{G}_s[f] = \frac{1}{3^s} f\left(\frac{x}{3}\right) + \frac{1}{3^s} f\left(\frac{x-2}{3}\right)$$



unique valeur propre dominante $\lambda(s)$

lien avec les SD



- opérateur de transfert

$$\mathbf{G}_s[f] = \frac{1}{3^s} f\left(\frac{x}{3}\right) + \frac{1}{3^s} f\left(\frac{x-1}{3}\right) + \frac{1}{3^s} f\left(\frac{x-2}{3}\right)$$



- opérateur de transfert contraint

$$\mathbf{G}_s[f] = \frac{1}{3^s} f\left(\frac{x}{3}\right) + \frac{1}{3^s} f\left(\frac{x-2}{3}\right)$$

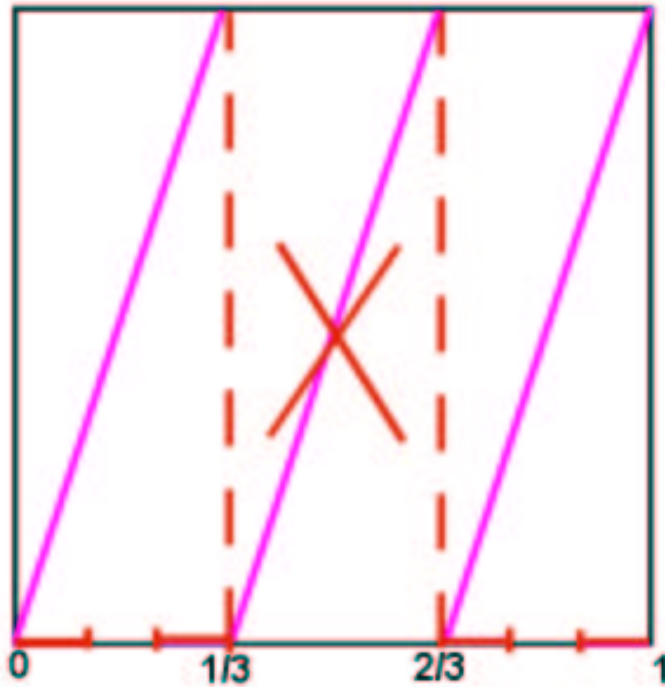


unique valeur propre dominante $\lambda(s)$



dim. de Hausdorff = s tel que $\lambda(s) = 1$

lien avec les SD



- opérateur de transfert

$$\mathbf{G}_s[f] = \frac{1}{3^s} f\left(\frac{x}{3}\right) + \frac{1}{3^s} f\left(\frac{x-1}{3}\right) + \frac{1}{3^s} f\left(\frac{x-2}{3}\right)$$



- opérateur de transfert contraint

$$\mathbf{G}_s[f] = \frac{1}{3^s} f\left(\frac{x}{3}\right) + \frac{1}{3^s} f\left(\frac{x-2}{3}\right)$$

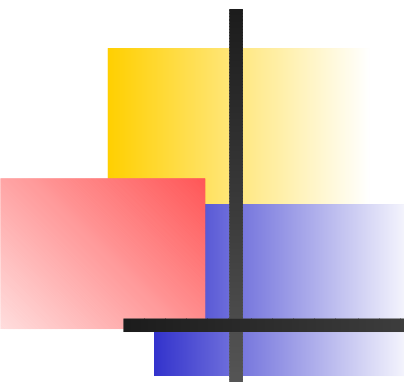


unique valeur propre dominante $\lambda(s)$



dim. de Hausdorff = s tel que $\lambda(s) = 1$

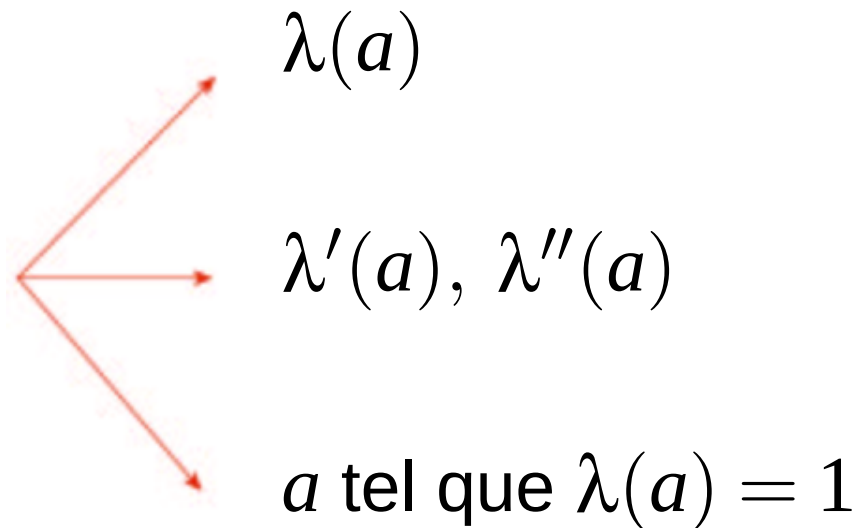
- Généralisable à tout SD ayant au moins 3 branches

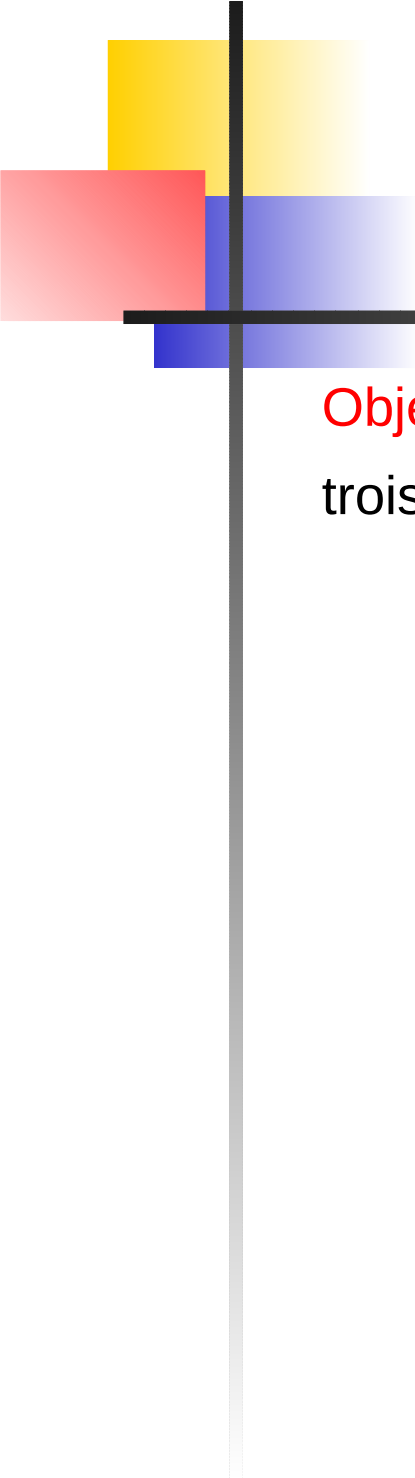


Objectif

Proposer une méthode générale pour calculer $\lambda(s)$.

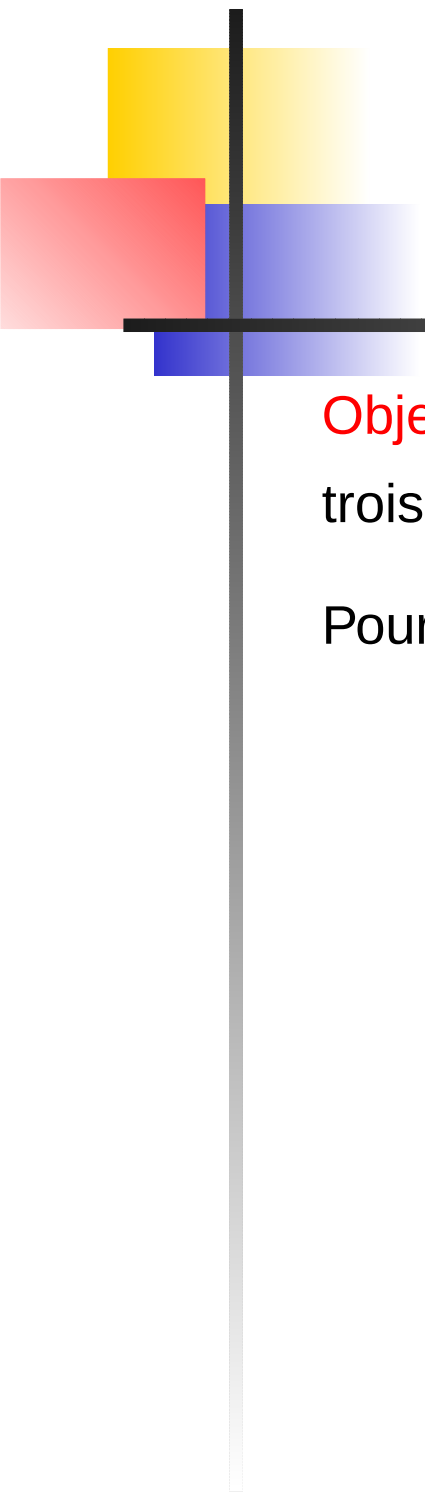
Application au calcul de trois types de constantes:





Objectifs

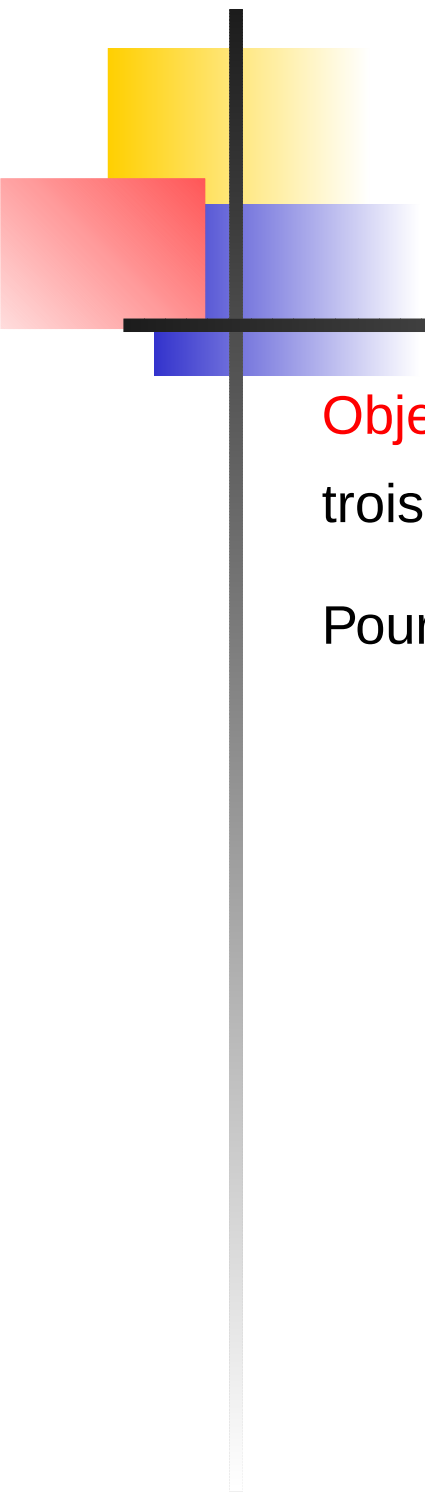
Objectif: proposer une methode pour le calcul (numérique) prouvé des trois types de constantes.



Objectifs

Objectif: proposer une methode pour le calcul (numérique) prouvé des trois types de constantes.

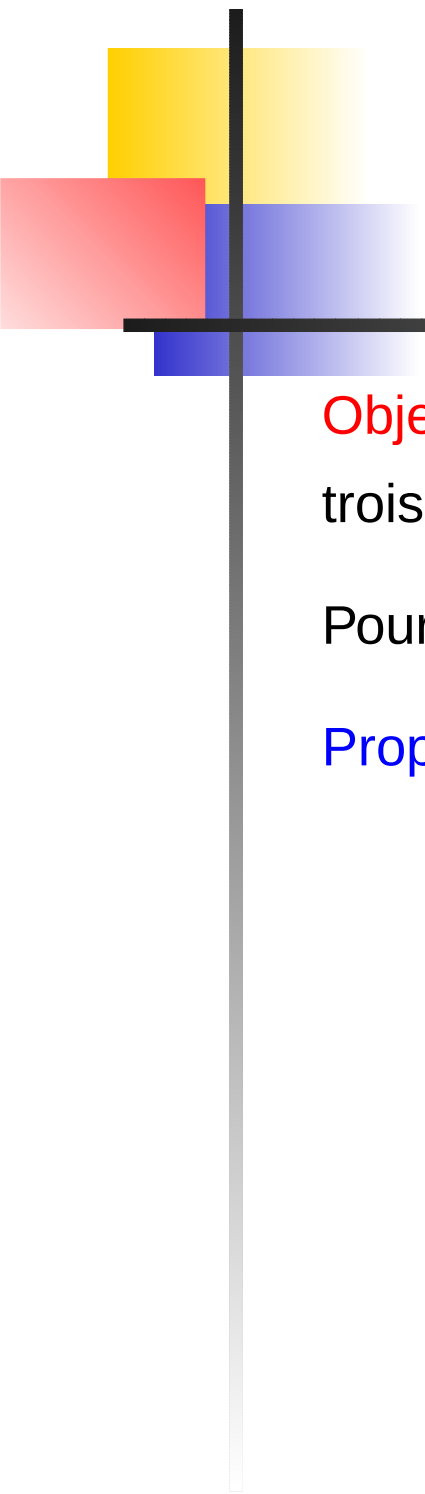
Pour tous les SD?



Objectifs

Objectif: proposer une methode pour le calcul (numérique) prouvé des trois types de constantes.

Pour tous les SD? NON



Objectifs

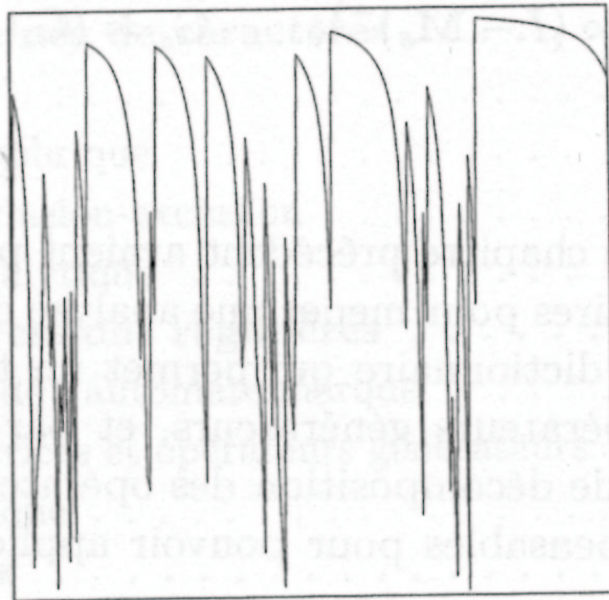
Objectif: proposer une methode pour le calcul (numérique) prouvé des trois types de constantes.

Pour tous les SD? NON

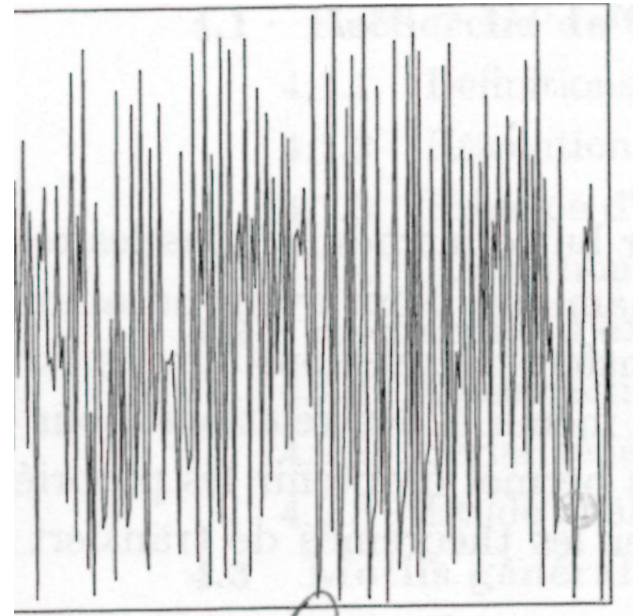
Propriété de contraction

Propriétés de contraction

- pas de point fixe indifférent



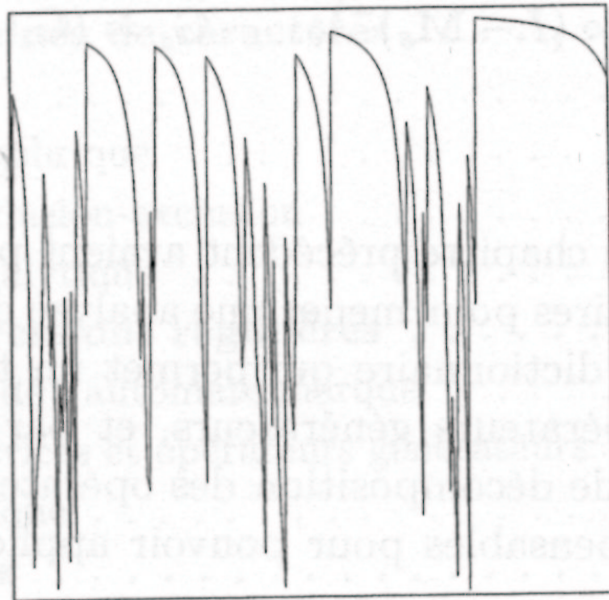
avec



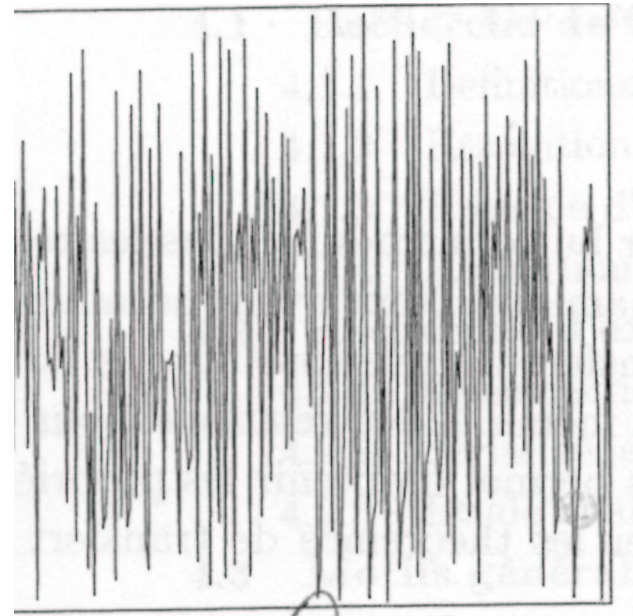
sans

Propriétés de contraction

- pas de point fixe indifférent



avec

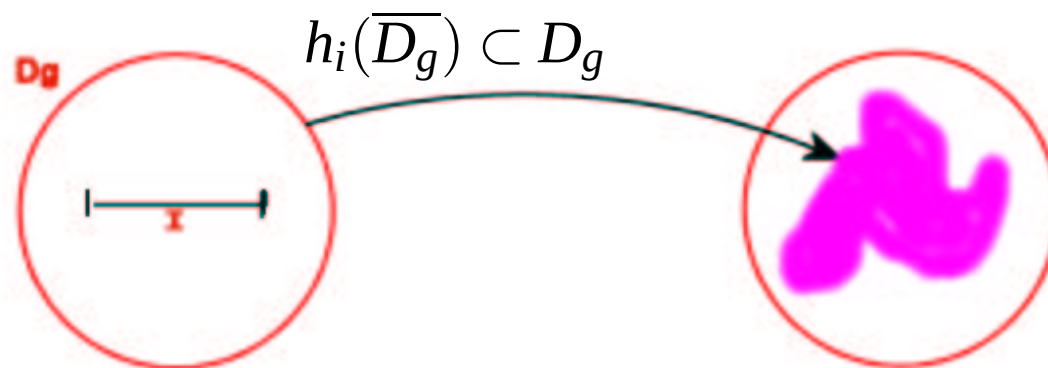


sans

conséquence: ralentissement des algorithmes

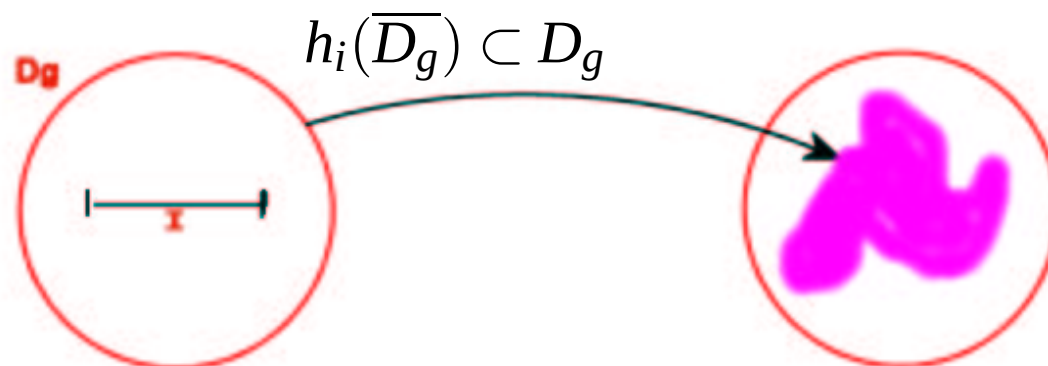
Propriétés de contraction

- Condition suffisante: contraction

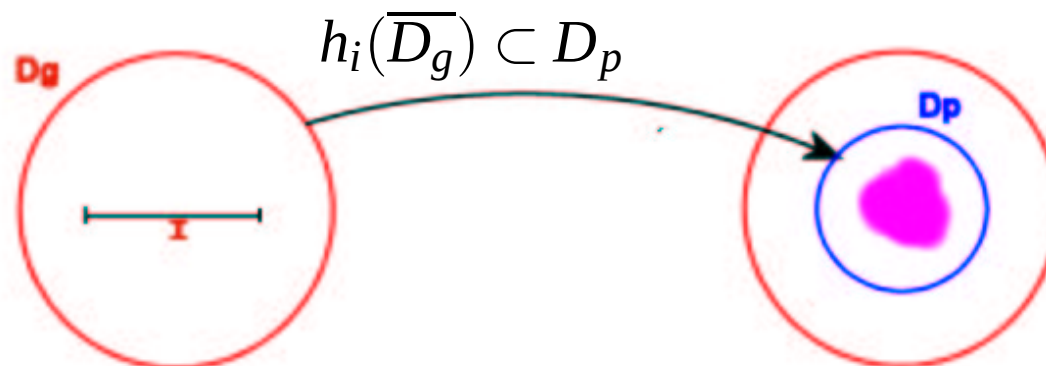


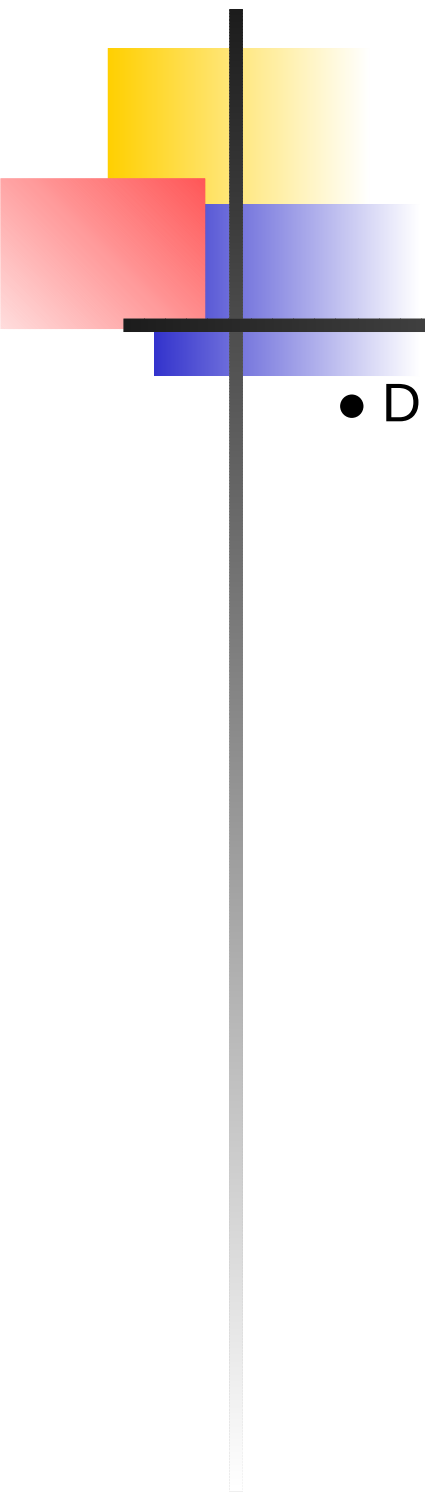
Propriétés de contraction

- Condition suffisante: contraction



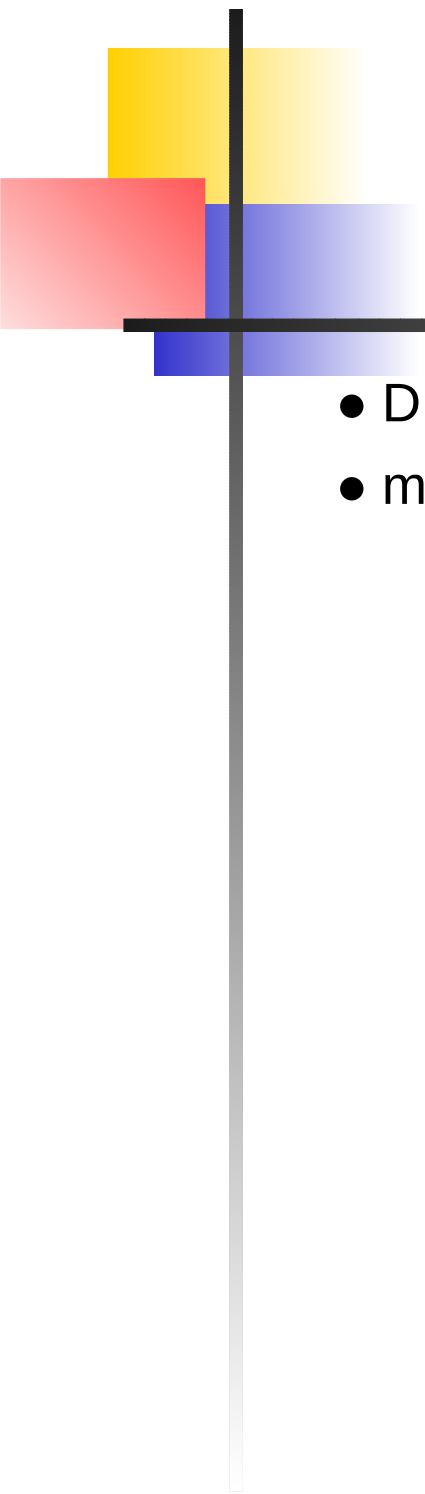
- condition de contraction forte





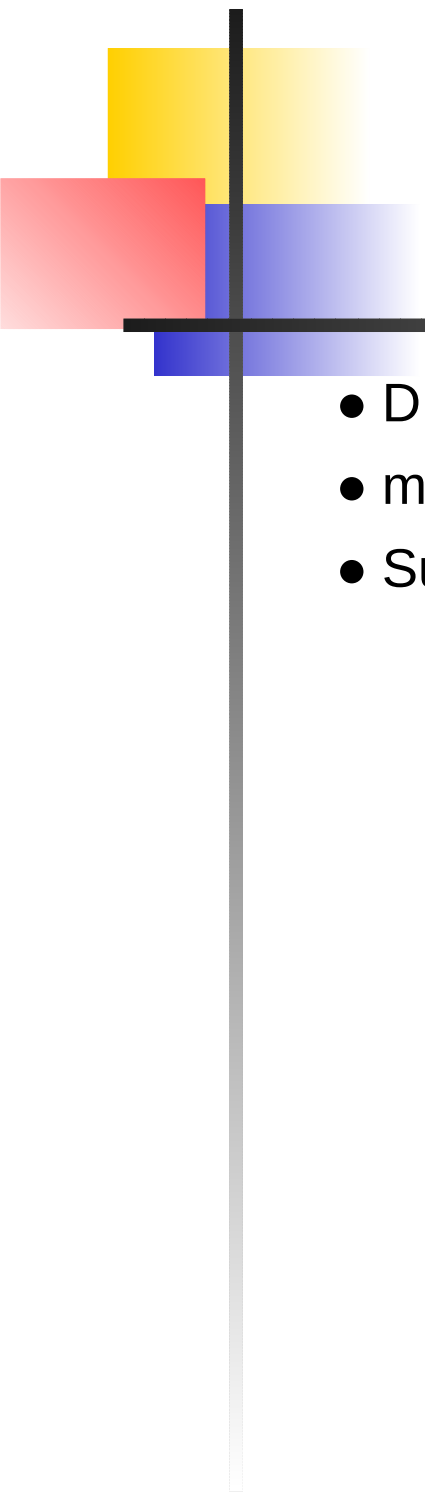
DFV méthode

- DFV= Daudet, Flajolet et Vallée



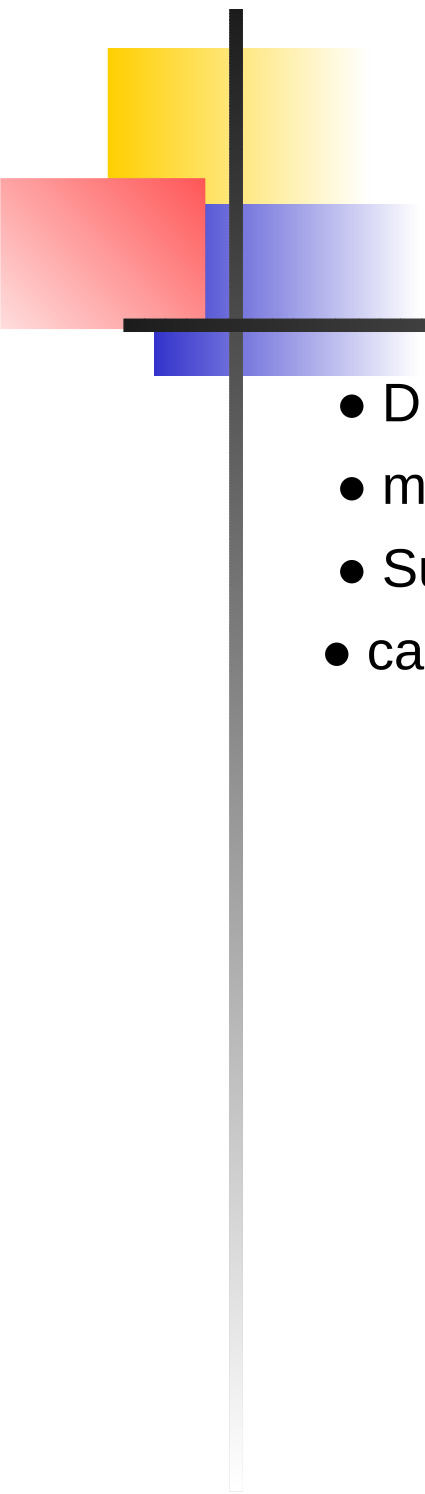
DFV méthode

- DFV= Daudet, Flajolet et Vallée
- méthode utilisée mais non prouvée



DFV méthode

- DFV= Daudet, Flajolet et Vallée
- méthode utilisée mais non prouvée
- Sur un espace convenable, $\mathbf{G}_s$ est une matrice infinie

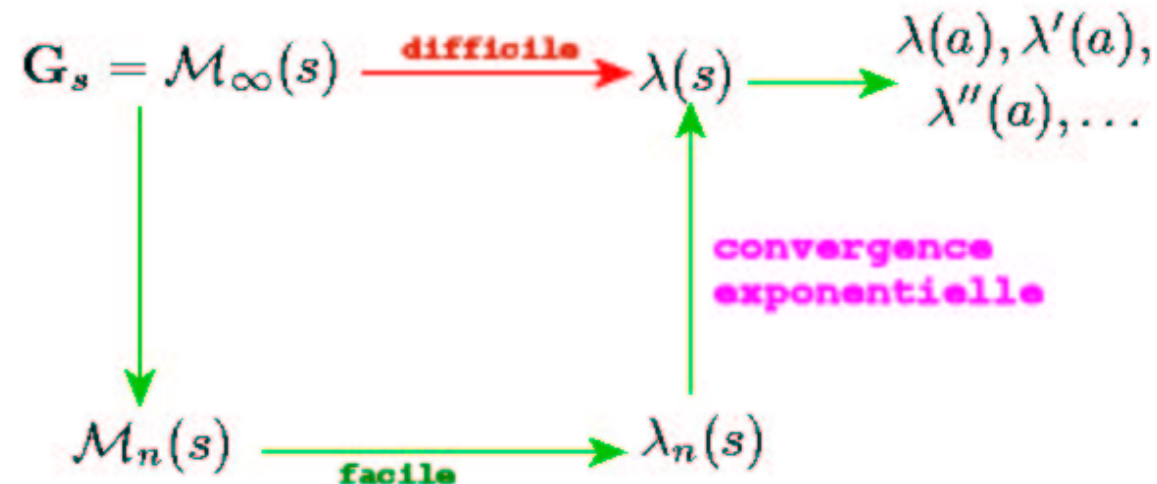


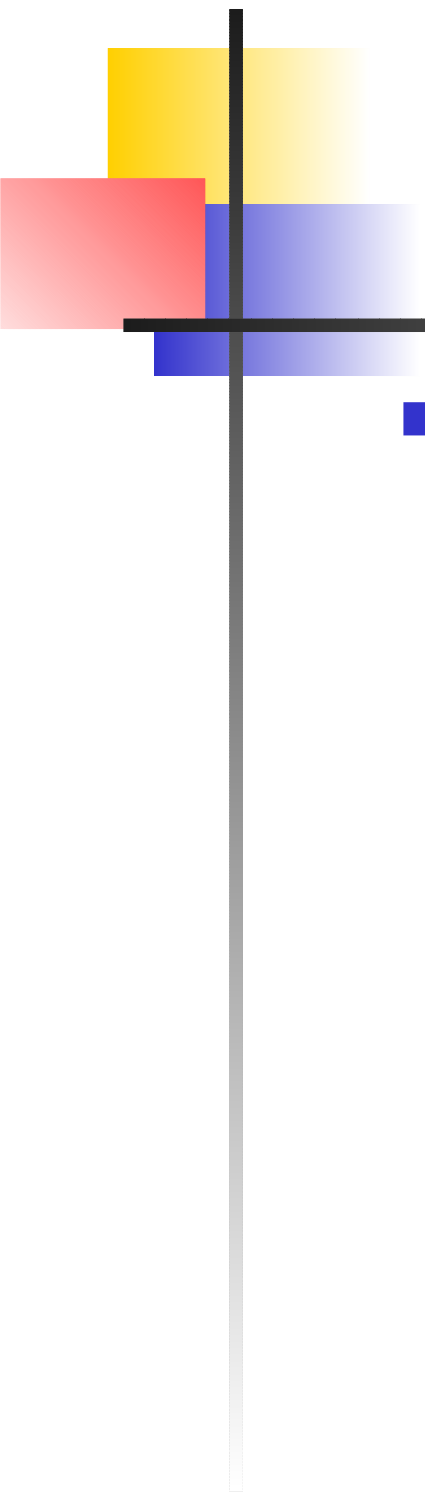
DFV méthode

- DFV= Daudet, Flajolet et Vallée
- méthode utilisée mais non prouvée
- Sur un espace convenable, $\mathbf{G}_s$ est une matrice infinie
- calcul de valeurs propres facile ssi on a des matrices finies

DFV méthode

- DFV= Daudet, Flajolet et Vallée
- méthode utilisée mais non prouvée
- Sur un espace convenable, $\mathbf{G}_s$ est une matrice infinie
- calcul de valeurs propres facile ssi on a des matrices finies





DFV méthode: théorie

- $\mathbf{G}_s : \mathcal{A}_\infty(D_p) \rightarrow \mathcal{A}_\infty(D_g) \subset \mathcal{A}_\infty(D_g)$ avec
 $\mathcal{A}_\infty(D) = \{\text{fonctions holomorphes sur } D \text{ et continues sur le bord}\}$



DFV méthode: théorie

- $\mathbf{G}_s : \mathcal{A}_\infty(D_p) \rightarrow \mathcal{A}_\infty(D_g) \subset \mathcal{A}_\infty(D_g)$ avec
 $\mathcal{A}_\infty(D) = \{\text{fonctions holomorphes sur } D \text{ et continues sur le bord}\}$
- toute fonction de $\mathcal{A}_\infty(D)$ s'écrit

$$f(z) = \sum_{k \geq 0} a_k (z - x_0)^k$$



DFV méthode: théorie

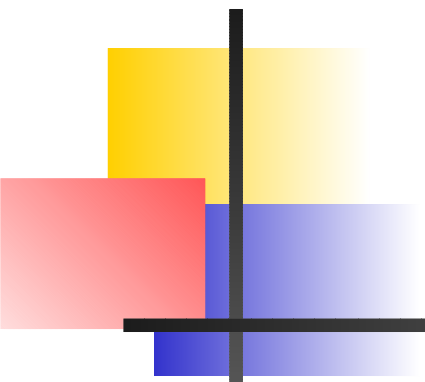
- $\mathbf{G}_s : \mathcal{A}_\infty(D_p) \rightarrow \mathcal{A}_\infty(D_g) \subset \mathcal{A}_\infty(D_g)$ avec
 $\mathcal{A}_\infty(D) = \{\text{fonctions holomorphes sur } D \text{ et continues sur le bord}\}$

- toute fonction de $\mathcal{A}_\infty(D)$ s'écrit

$$f(z) = \sum_{k \geq 0} a_k (z - x_0)^k$$

- opérateur de troncature:

$$\pi_n[f](s) = \sum_{k=0}^n a_k (z - x_0)^k$$



DFV méthode: théorie

- $\mathbf{G}_s : \mathcal{A}_\infty(D_p) \rightarrow \mathcal{A}_\infty(D_g) \subset \mathcal{A}_\infty(D_g)$ avec
 $\mathcal{A}_\infty(D) = \{\text{fonctions holomorphes sur } D \text{ et continues sur le bord}\}$

- toute fonction de $\mathcal{A}_\infty(D)$ s'écrit

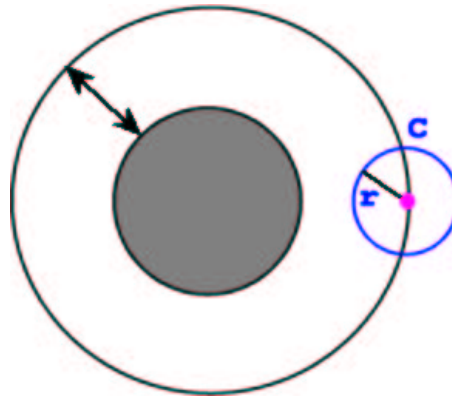
$$f(z) = \sum_{k \geq 0} a_k (z - x_0)^k$$

- opérateur de troncature:

$$\pi_n[f](s) = \sum_{k=0}^n a_k (z - x_0)^k$$

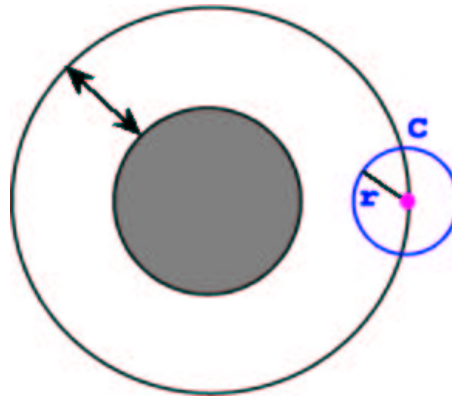
- $\mathcal{M}_n = \pi_n \circ \mathbf{G}_s \circ \pi_n$

Analyse fonctionnelle



■ cercle C comme dans le dessin

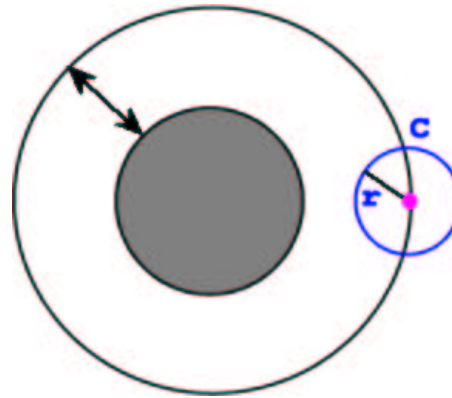
Analyse fonctionnelle



- cercle C comme dans le dessin
- on pose

$$\alpha_C = \sup_{z \in C} \|(\mathcal{M}_\infty - z Id)^{-1}\|$$

Analyse fonctionnelle



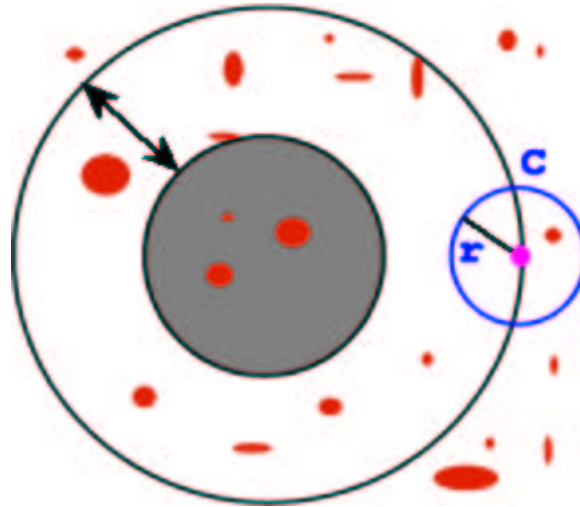
- cercle C comme dans le dessin
- on pose

$$\alpha_C = \sup_{z \in C} \|(\mathcal{M}_\infty - z \text{Id})^{-1}\|$$

- Si

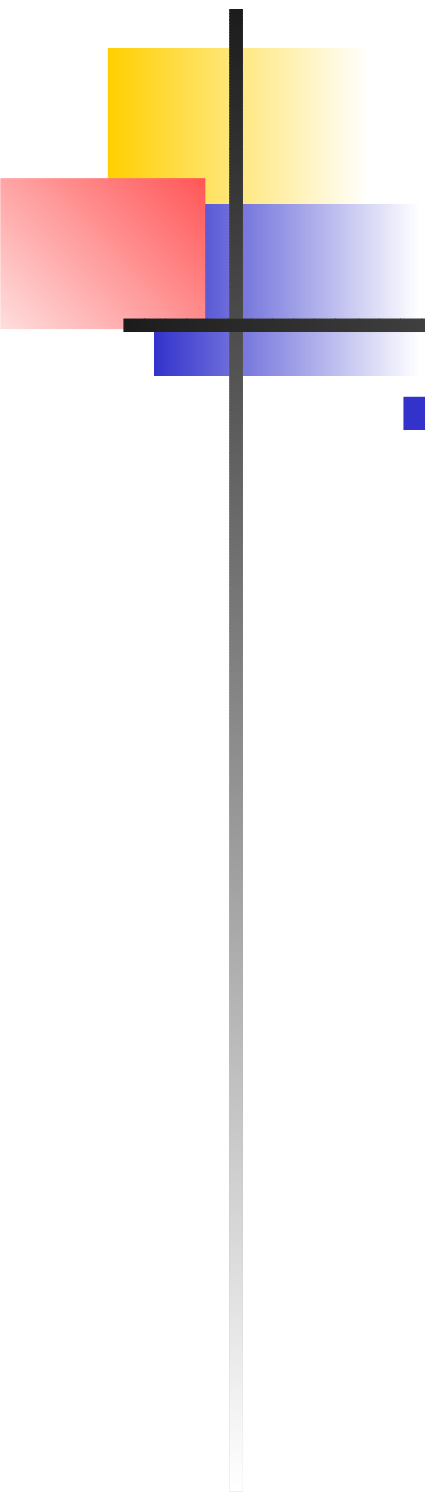
$$\|\mathcal{M}_\infty - \mathcal{M}_n\| \leq \min \left\{ \frac{1}{2\alpha_C}, \frac{1}{2r_C\alpha_C^2}, \frac{1}{8r^2\alpha_C^3} \right\}$$

Analyse fonctionnelle



Alors $\mathcal{M}_n$ admet une unique valeur propre $\lambda_n(s)$ dans C et

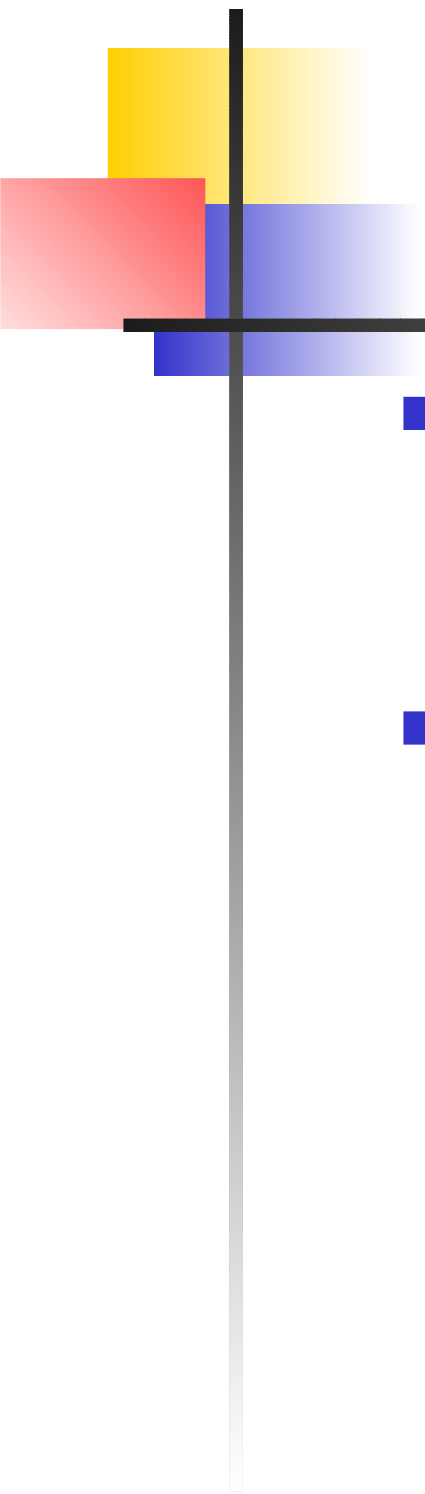
$$|\lambda_n(s) - \lambda(s)| \leq 2r\alpha_C \|\mathcal{M}_n - \mathcal{M}_\infty\|$$



Premier résultat

■ Premier résultat:

$$\|\mathcal{M}_n - \mathcal{M}_\infty\| \leq K \left(\frac{R_p}{R_g} \right)^n$$

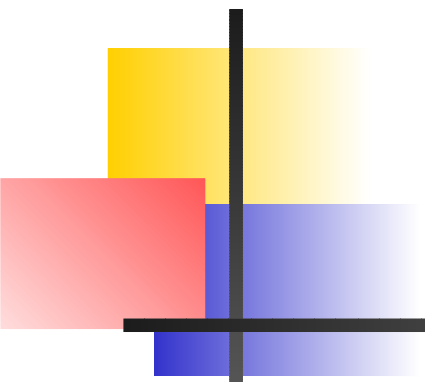


Premier résultat

- Premier résultat:

$$\|\mathcal{M}_n - \mathcal{M}_\infty\| \leq K \left(\frac{R_p}{R_g} \right)^n$$

- conséquence: la précision obtenue sur $\lambda(s)$ est linéaire en la taille de la matrice

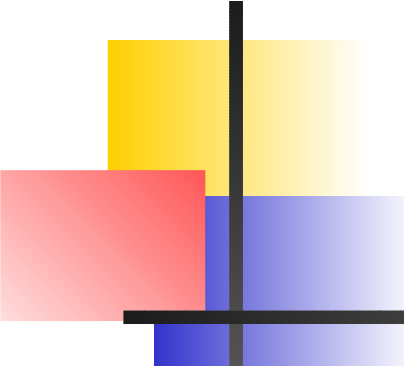


Premier résultat

- Premier résultat:

$$\|\mathcal{M}_n - \mathcal{M}_\infty\| \leq K \left(\frac{R_p}{R_g} \right)^n$$

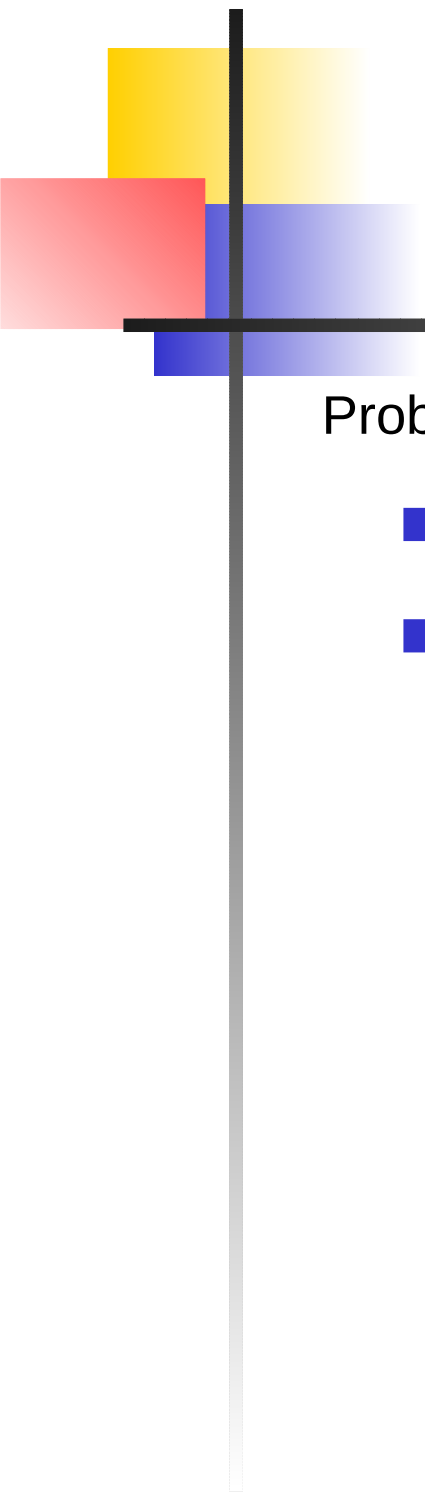
- conséquence: la précision obtenue sur $\lambda(s)$ est linéaire en la taille de la matrice
- corollaire: Si les $\mathcal{M}_n$ sont calculables en temps polynomial, $\lambda(s)$ est calculable en temps polynomial.



vers des résultats numériques

Problèmes:

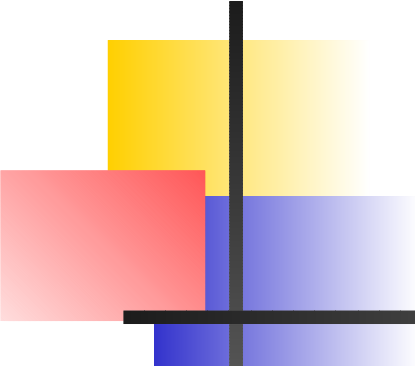
- déterminer un cercle C



vers des résultats numériques

Problèmes:

- déterminer un cercle C
- calculer α_C

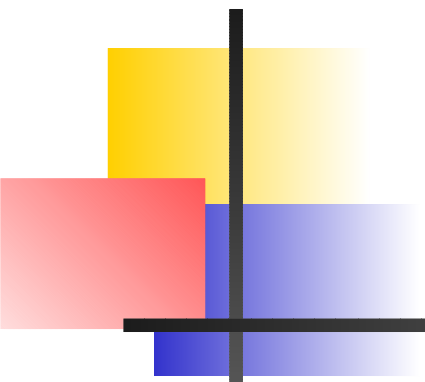


vers des résultats numériques

Problèmes:

- déterminer un cercle C
- calculer α_C

Le problème reste ouvert dans le cadre général des systèmes dynamiques à forte contraction



vers des résultats numériques

Problèmes:

- déterminer un cercle C
- calculer α_C

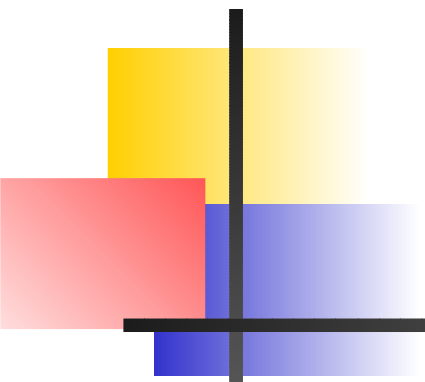
Le problème reste ouvert dans le cadre général des systèmes dynamiques à forte contraction → cas de la source des fractions continues



Cercle C

Cercle C :

- estimation simple de $\lambda(s)$
- minoration du saut spectral



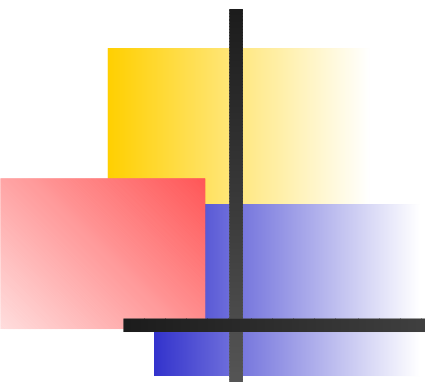
Cercle C

Cercle C:

- estimation simple de $\lambda(s)$:

$$\text{Si } \mu \leq \frac{\mathbf{G}_s[f]}{f} \leq \nu, \text{ alors } \mu \leq \lambda(s) \leq \nu$$

- minoration du saut spectral



Cercle C

Cercle C:

- estimation simple de $\lambda(s)$

Si $\mu \leq \frac{\mathbf{G}_s[f]}{f} \leq \nu$, alors $\mu \leq \lambda(s) \leq \nu$

- minoration du saut spectral

■ $\text{Tr} \mathbf{G}_s^2 = \sum_{\lambda \text{ vp de } \mathbf{G}_s} \lambda^2$



Cercle C

Cercle C:

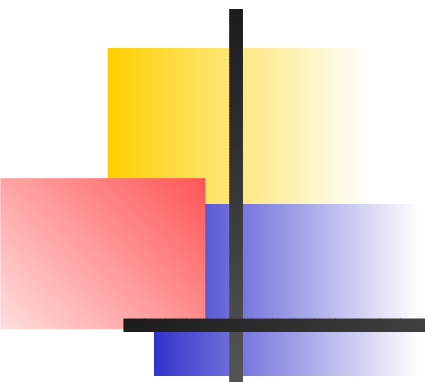
- estimation simple de $\lambda(s)$

Si $\mu \leq \frac{\mathbf{G}_s[f]}{f} \leq \nu$, alors $\mu \leq \lambda(s) \leq \nu$

- minoration du saut spectral

- $\text{Tr} \mathbf{G}_s^2 = \sum_{\lambda \text{ vp de } \mathbf{G}_s} \lambda^2$

- $\gamma^2 \leq \text{Tr} \mathbf{G}_s^2 - \lambda(s)^2$



Cercle C

Cercle C:

- estimation simple de $\lambda(s)$

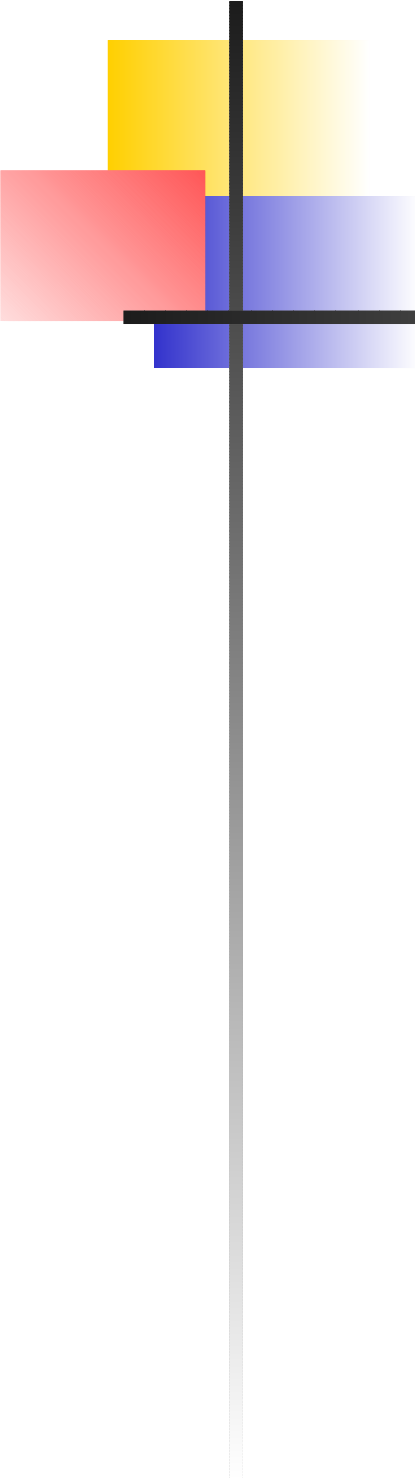
Si $\mu \leq \frac{\mathbf{G}_s[f]}{f} \leq \nu$, alors $\mu \leq \lambda(s) \leq \nu$

- minoration du saut spectral

- $\text{Tr}\mathbf{G}_s^2 = \sum_{\lambda \text{ vp de } \mathbf{G}_s} \lambda^2$

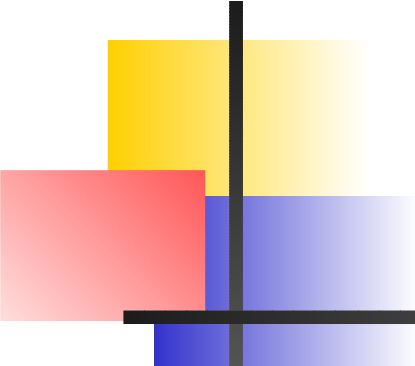
- $\gamma^2 \leq \text{Tr}\mathbf{G}_s^2 - \lambda(s)^2$

- $\text{Tr}\mathbf{G}_s^2$ est calculable en temps polynomial



Constante α_C

$$\alpha_C = \sup_{z \in C} \|(\mathbf{G}_s - z\mathbf{I})^{-1}\|$$

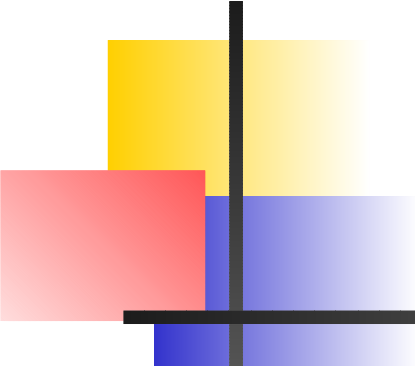


Constante α_C

$$\alpha_C = \sup_{z \in C} \|(\mathbf{G}_s - z\mathbf{I})^{-1}\|$$

Pb: en général difficile à calculer sauf si l'opérateur satisfait une propriété de **normalité**.

$$\|(\mathbf{G}_s - z\mathbf{I})^{-1}\| = \frac{1}{\text{dist}(C, \text{spectre})} = \frac{1}{r_C}$$



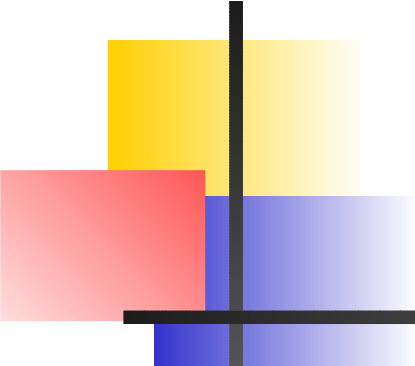
Constante α_C

$$\alpha_C = \sup_{z \in C} \|(\mathbf{G}_s - z\mathbf{I})^{-1}\|$$

Pb: en général difficile à calculer sauf si l'opérateur satisfait une propriété de **normalité**.

$$\|(\mathbf{G}_s - z\mathbf{I})^{-1}\| = \frac{1}{\text{dist}(C, \text{spectre})} = \frac{1}{r_C}$$

Pb: $\mathbf{G}$ n'est pas normal sur $\mathcal{A}_\infty(D_p)$



Constante α_C

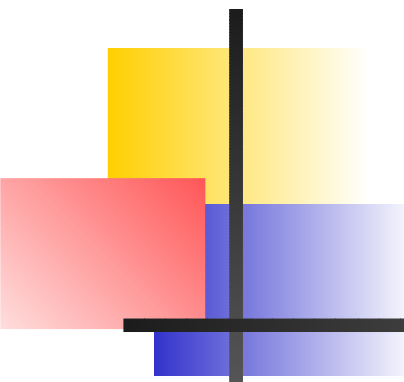
$$\alpha_C = \sup_{z \in C} \|(\mathbf{G}_s - z\mathbf{I})^{-1}\|$$

Pb: en général difficile à calculer sauf si l'opérateur satisfait une propriété de **normalité**.

$$\|(\mathbf{G}_s - z\mathbf{I})^{-1}\| = \frac{1}{\text{dist}(C, \text{spectre})} = \frac{1}{r_C}$$

Pb: $\mathbf{G}$ n'est pas normal sur $\mathcal{A}_\infty(D_p)$

$\mathbf{G}_s$ est normal sur un autre espace $\mathcal{H}$



Constante α_C

$$\alpha_C = \sup_{z \in C} \|(\mathbf{G}_s - z\mathbf{I})^{-1}\|$$

Pb: en général difficile à calculer sauf si l'opérateur satisfait une propriété de **normalité**.

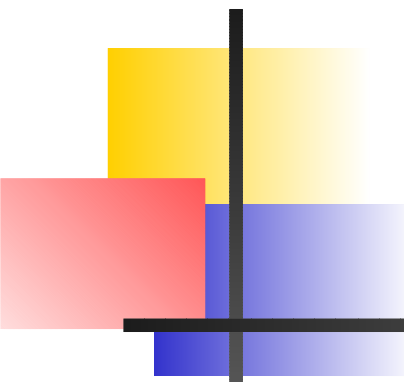
$$\|(\mathbf{G}_s - z\mathbf{I})^{-1}\| = \frac{1}{\text{dist}(C, \text{spectre})} = \frac{1}{r_C}$$

Pb: $\mathbf{G}$ n'est pas normal sur $\mathcal{A}_\infty(D_p)$

$\mathbf{G}_s$ est normal sur un autre espace $\mathcal{H}$

+

relations: $\mathcal{H} \subset \mathcal{A}_\infty(D_g)$ et $\mathbf{G}_s[\mathcal{A}_\infty(D_g)] \subset \mathcal{H}$



Constante α_C

$$\alpha_C = \sup_{z \in C} \|(\mathbf{G}_s - z\mathbf{I})^{-1}\|$$

Pb: en général difficile à calculer sauf si l'opérateur satisfait une propriété de **normalité**.

$$\|(\mathbf{G}_s - z\mathbf{I})^{-1}\| = \frac{1}{\text{dist}(C, \text{spectre})} = \frac{1}{r_C}$$

Pb: $\mathbf{G}$ n'est pas normal sur $\mathcal{A}_\infty(D_p)$

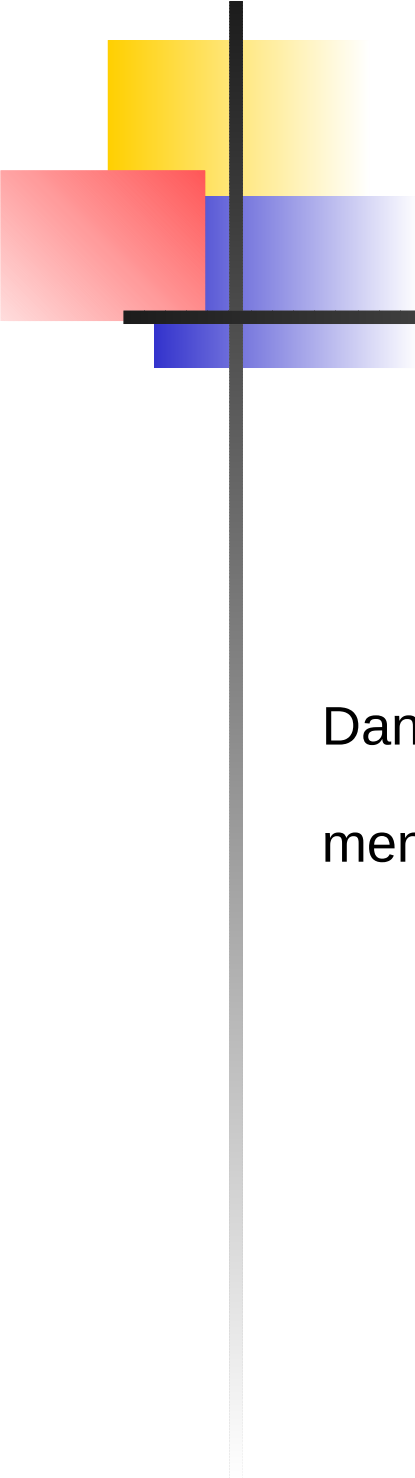
$\mathbf{G}_s$ est normal sur un autre espace $\mathcal{H}$

+

relations: $\mathcal{H} \subset \mathcal{A}_\infty(D_g)$ et $\mathbf{G}_s[\mathcal{A}_\infty(D_g)] \subset \mathcal{H}$

→

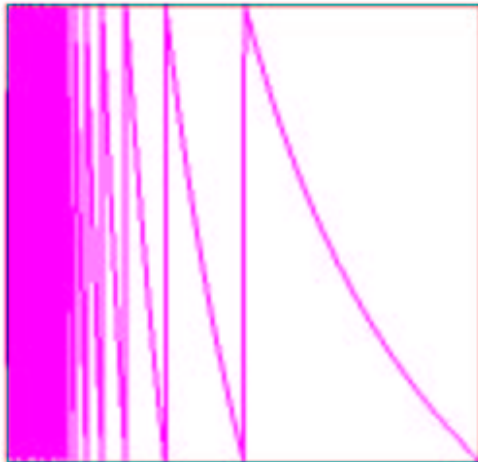
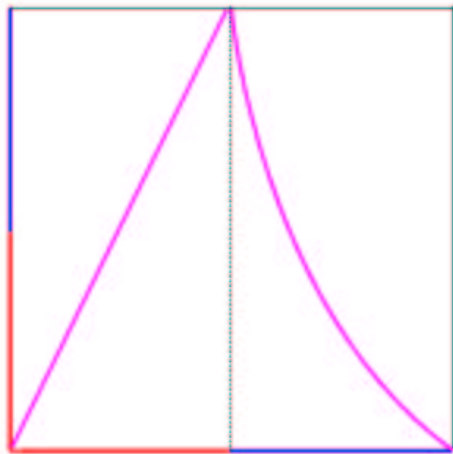
borne sur $\|(\mathbf{G}_s - z\mathbf{I})^{-1}\|$



Résultat 2

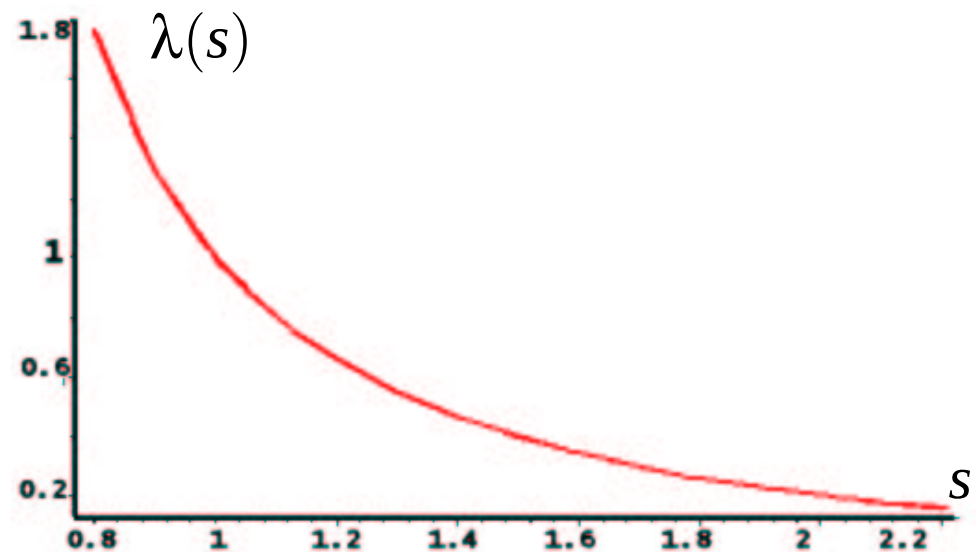
Dans le cadre des Fractions Continues, $\lambda(s)$ est calculable numériquement en temps polynomial

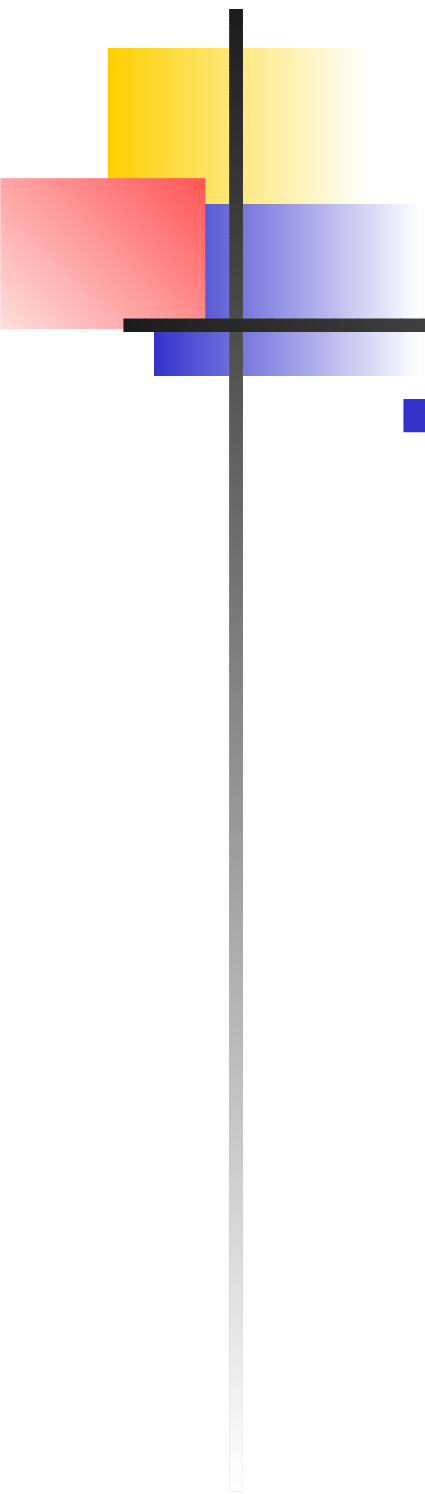
Constantes de type $\lambda(a)$



- probabilité de 2-coïncidence

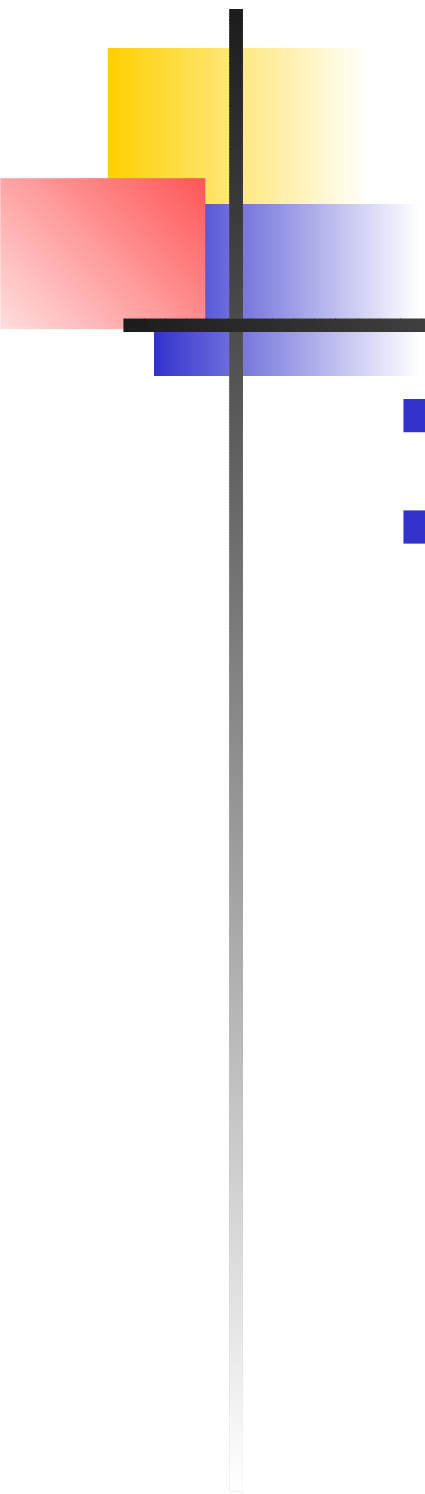
$$\lambda(2) = 0.51$$





Constantes de type $a \text{ tq } \lambda(a) = 1$

■ $A \subset N^*$



Constantes de type

$$a \text{ tq } \lambda(a) = 1$$

- $A \subset \mathbb{N}^*$
- $E_A = \{x \in [0, 1] \mid \text{développement en fraction continue contraint à } A\}$

Constantes de type $a \text{ tq } \lambda(a) = 1$

- $A \subset \mathbb{N}^*$
- $E_A = \{x \in [0, 1] \mid \text{développement en fraction continue contraint à } A\}$
- Opérateurs de transfert concernés:

$$\mathbf{G}_s[f](x) = \sum_{i \in A} \frac{1}{(i+x)^{2s}} f\left(\frac{1}{i+x}\right)$$



calcul numérique prouvé

- Valeurs pour $A = \{1, 2\}$

digits	temps	dimension de Hausdorff
5	2mn	0.53128
10	8mn	0.5312805062
15	25mn	0.531280506277205
20	1h	0.53128050627720514162
40	14h11	0.53128050627720514162...
45	23h10	0.53128050627720514162...

Calcul de $\lambda'(a)$, $\lambda''(a)$

- Formule de Taylor+inégalité triangulaire

$$\left| \lambda''(1) - \frac{\lambda_n(1+h) + \lambda_n(1-h) - 2\lambda_n(1)}{h^2} \right| \leq \frac{h^2}{12} \sup_{]1-h, 1+h[} |\lambda^{(4)}|$$
$$+ \frac{1}{h^2} [|\lambda_n(1+h) - \lambda(1+h)| + |\lambda_n(1-h) - \lambda(1-h)| + |\lambda_n(1) - \lambda(1)|]$$

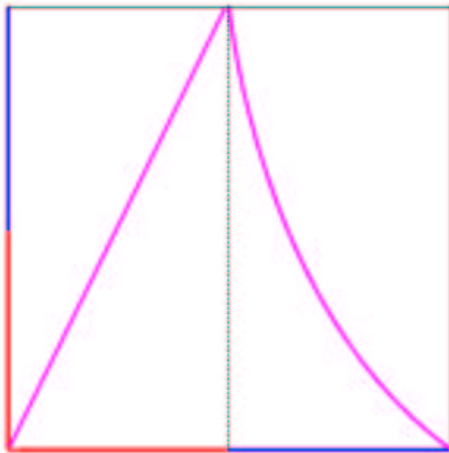
Calcul de $\lambda'(a)$, $\lambda''(a)$

- Formule de Taylor+inégalité triangulaire

$$\left| \lambda''(1) - \frac{\lambda_n(1+h) + \lambda_n(1-h) - 2\lambda_n(1)}{h^2} \right| \leq \frac{h^2}{12} \sup_{]1-h, 1+h[} |\lambda^{(4)}|$$
$$+ \frac{1}{h^2} [|\lambda_n(1+h) - \lambda(1+h)| + |\lambda_n(1-h) - \lambda(1-h)| + |\lambda_n(1) - \lambda(1)|]$$

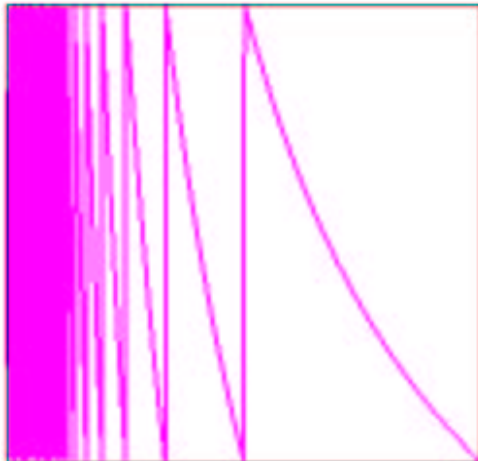
- Etape 1: majorer $\sup$ (possible dès que α_C est connu)
- Etape 2: déterminer h tel que *bleu* soit $< \frac{\text{précision}}{2}$
- Etape 3: déterminer n tel que *rouge* soit $< \frac{\text{précision}}{2}$

applications numériques



- entropie

$$-\lambda'(1) = 0.6766$$

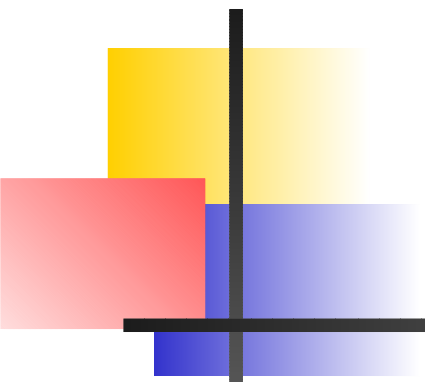


- entropie = $-\frac{\pi^2}{6 \log 2}$

- constante de Hensley = $-2 \frac{\lambda''(1) - \lambda'(1)^2}{\lambda'(1)^3}$

$$c_H = 0.516062408899991 \dots$$

Première valeur numérique PROUVÉE



Conclusion

- 2 types d'algorithmes basés sur la même méthode:
 - des algorithmes *théorique*: on ne connaît pas α_C
 - des algorithmes *numérique*: tout est connu (cas des Fractions Continues)
- première valeur numérique prouvée pour la constante de Hensley
- un nouvel algorithme polynomial pour les dimensions de Hausdorff
- généralisable à d'autres problèmes (FC à contraintes périodiques, analyse en distribution d'algorithmes Euclidiens, ...)



A faire

- Traitement des erreurs de calcul
 - pour le calcul de la matrice
 - pour le calcul de la valeur propre
- généraliser à d'autres SD
- Résoudre le problème des constantes α_C (difficile)