Calculer avec des équations différentielles. Calculabilité, Complexité.

Olivier Bournez

Luminy, Mai 2016

Calculer avec des équations différentielles. Calculabilité, Complexité.

Olivier Bournez [1]

Luminy, Mai 2016

[1] Basé sur des travaux communs avec différentes personnes dont Manuel Campagnolo, Daniel Graça, Emmanuel Hainry, Amaury Pouly.
De nombreux transparents repris de l'exposé de (l'excellente) thèse de Amaury Pouly.

Le sujet de cet exposé

$$\begin{cases} y' & = & p(y(t)) \\ y(0) & = & y_0 \end{cases} \tag{1}$$

où p est (un vecteur) de polynômes.

Le sujet de cet exposé

$$\left\{ \begin{array}{lll} y' & = & p(y(t)) \\ y(0) & = & y_0 \end{array} \right. \qquad (1)$$

où p est (un vecteur) de polynômes.

- Question: Quel est le temps nécessaire pour résoudre une telle équation différentielle?

Le sujet de cet exposé

$$\begin{cases} y' & = & p(y(t)) \\ y(0) & = & y_0 \end{cases} \tag{1}$$

où p est (un vecteur) de polynômes.

- Question: Quel est le temps nécessaire pour résoudre une telle équation différentielle?
  - Etant donnés p, y_0, t and ϵ, calculer $y(t)$ à ϵ près.

Le sujet de cet exposé

$$\begin{cases} y' & = & p(y(t)) \\ y(0) & = & y_0 \end{cases} \quad (1)$$

où p est (un vecteur) de polynômes.

- Question: Quel est le temps nécessaire pour résoudre une telle équation différentielle?
  - Etant donnés p, y_0, t and ϵ, calculer $y(t)$ à ϵ près.
- Question proche: Quel peut-on calculer avec des équations différentielles?

Le sujet de cet exposé

NACA Lewis Flight Propulsion Laboratory's Differential Analyser

- Question: quelle est la puissance de cette machine?

Menu

Qu'est ce qu'un ordinateur ?

Ordinateur portable

Qu'est ce qu'un ordinateur ?

Ordinateur portable

Super-calculateur

Qu'est ce qu'un ordinateur ?

Ordinateur portable

Super-calculateur

Serveurs

Qu'est ce qu'un ordinateur ?

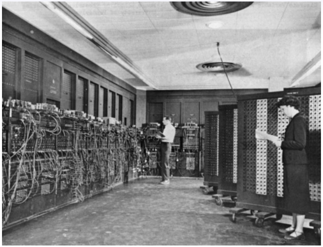
Ordinateur portable

Super-calculateur

Serveurs

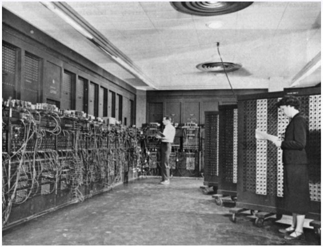
Commodore 64

Qu'est-ce qu'un ordinateur?



ENIAC

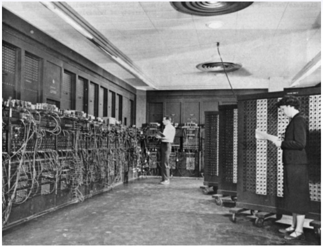
Qu'est-ce qu'un ordinateur?



ENIAC

Prédicteur de Marée de Kelvin

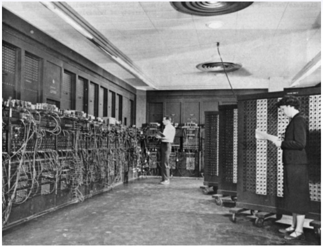
What is a computer ?



ENIAC

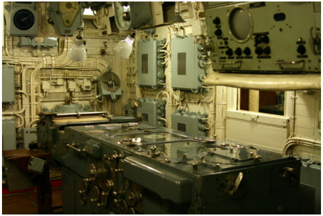
Prédicteur de Marée de Kelvin

What is a computer ?



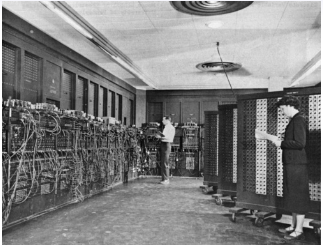
ENIAC

Prédicteur de Marée de Kelvin



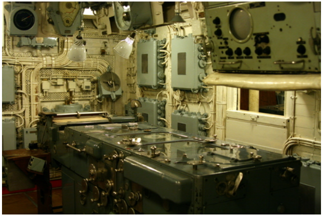
Admiralty Fire Control Table

What is a computer ?



ENIAC

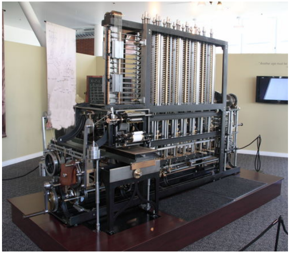
Prédicteur de Marée de Kelvin



Admiralty Fire Control Table

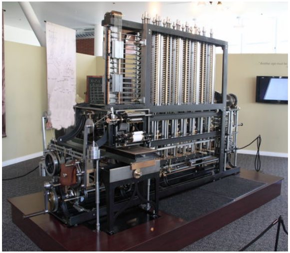
Analyseur Différentiel

What is a computer ?

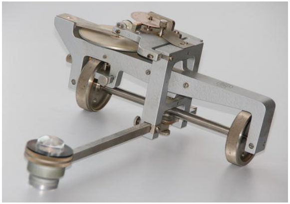


Machine à différences

What is a computer ?

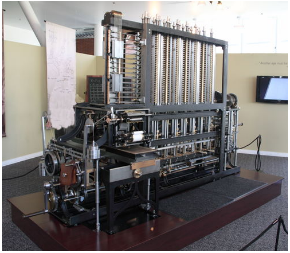


Machine à différences

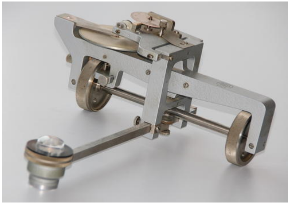


Planimètre Linéaire

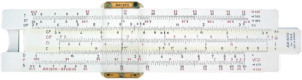
What is a computer ?



Machine à différences

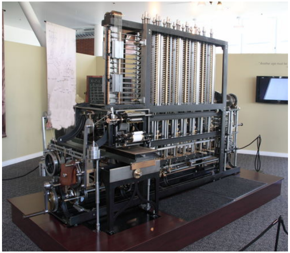


Planimètre Linéaire

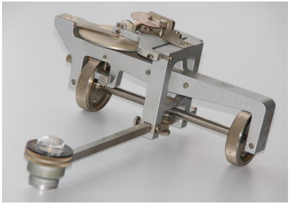


Règle à calcul

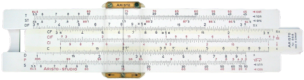
What is a computer ?



Machine à différences



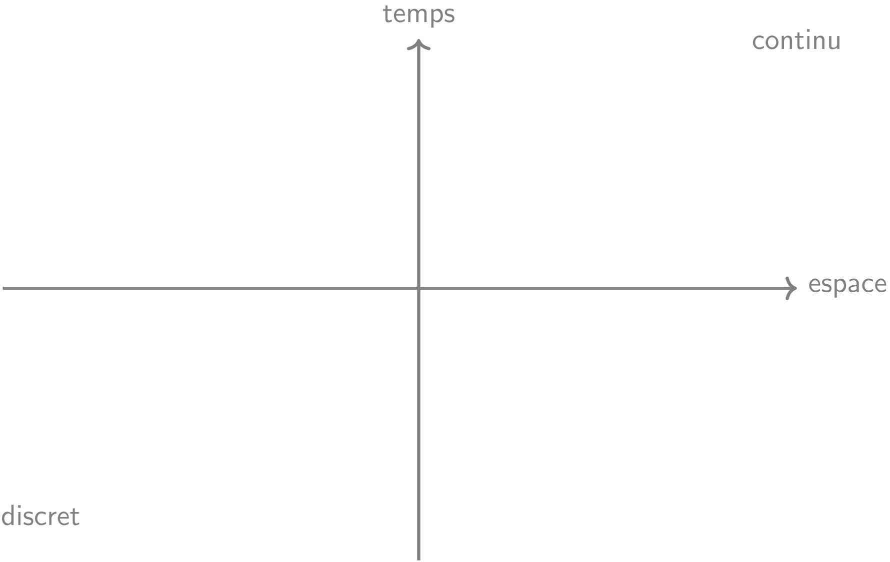
Planimètre Linéaire



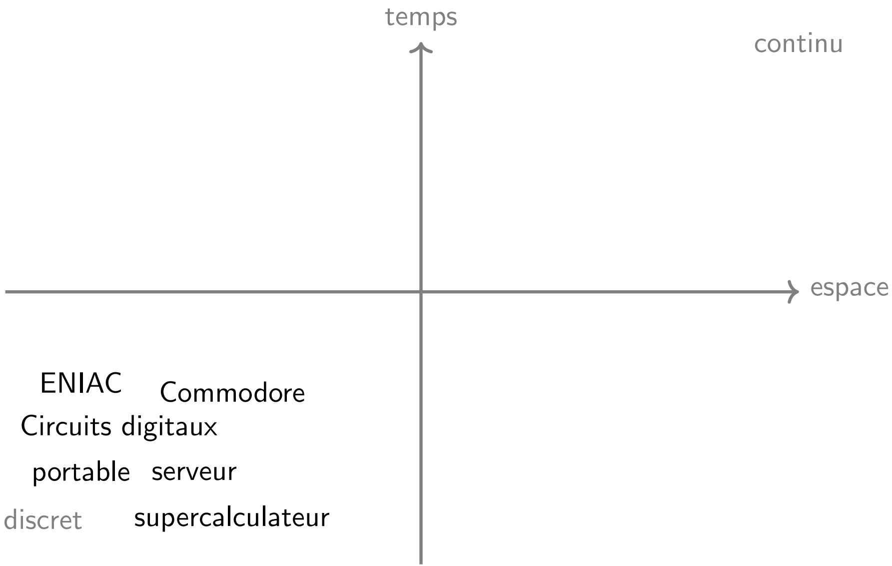
Règle à calcul

Mécanisme d'Anticythère

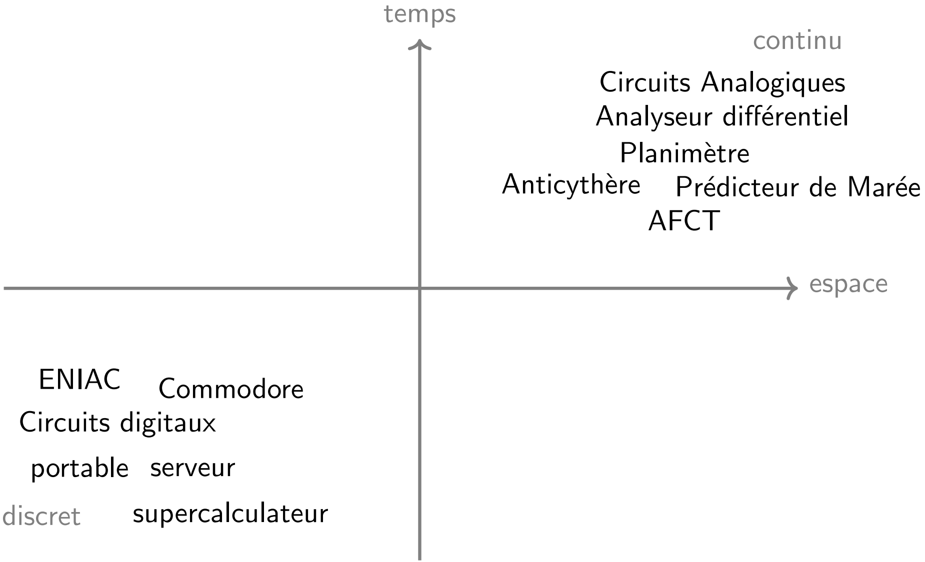
Une première classification



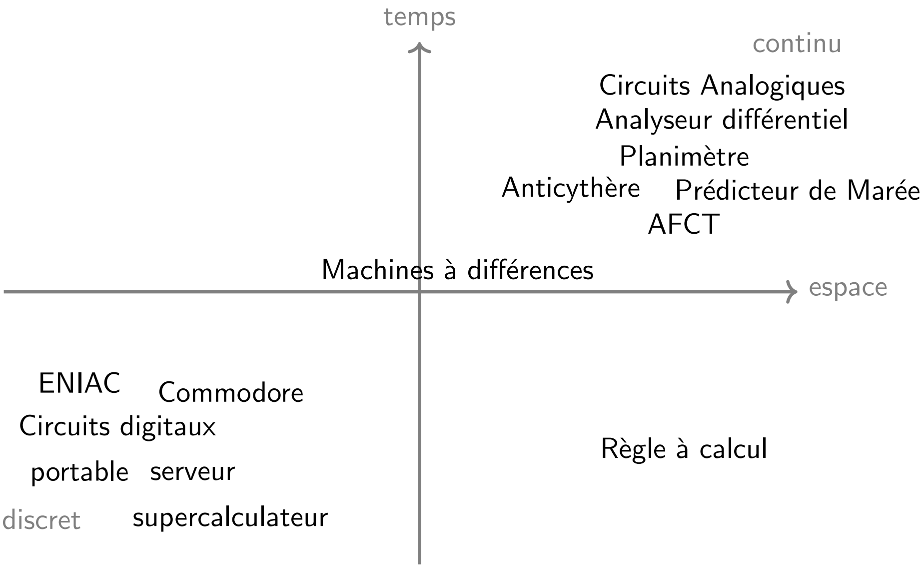
Une première classification



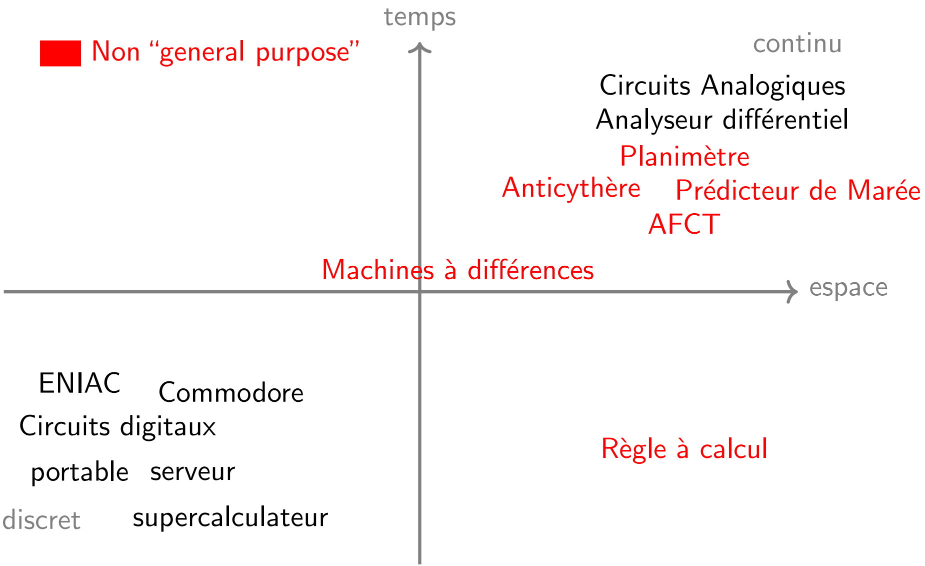
Une première classification



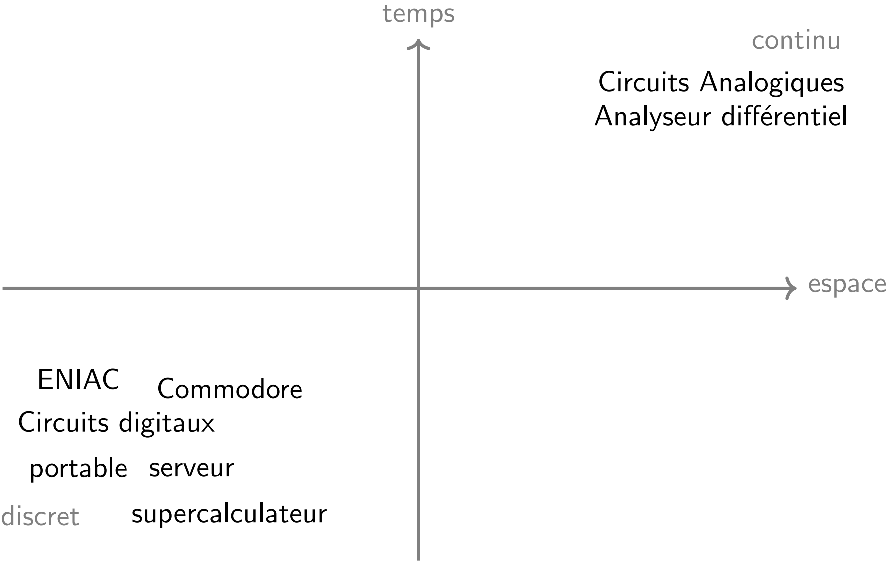
Une première classification



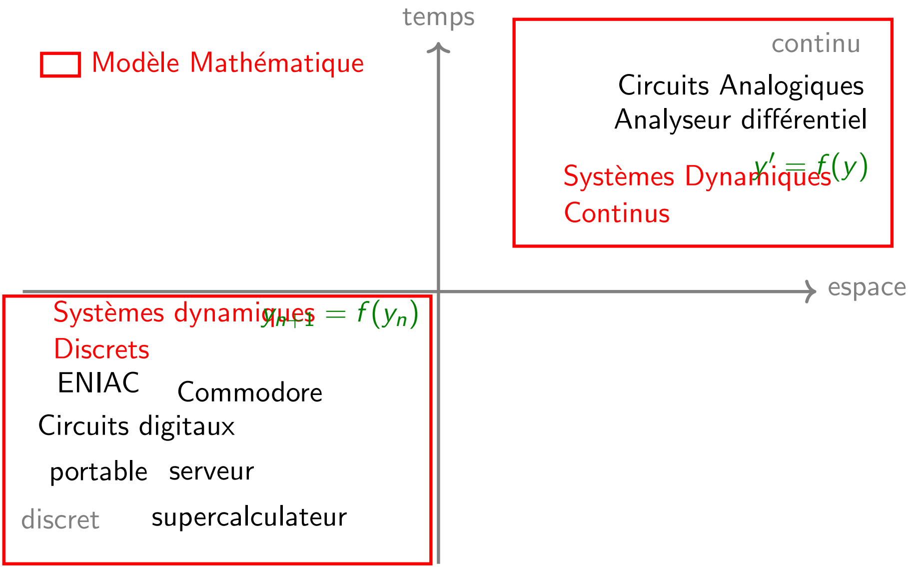
Une première classification



Une première classification



Une première classification



Systèmes dynamiques

Systèmes dynamiques
sur un espace continu

$\mathcal{H} = (X, f)$

Temps discret

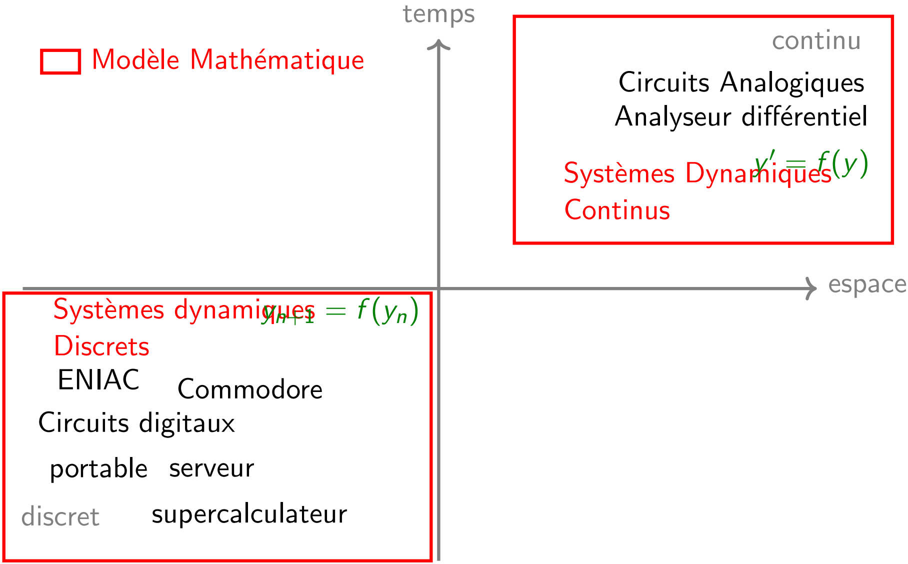
$$\left\{ \begin{array}{lll} y(t+1) & = & f(y(t)) \\ y(0) & = & x \end{array} \right.$$

Temps continu

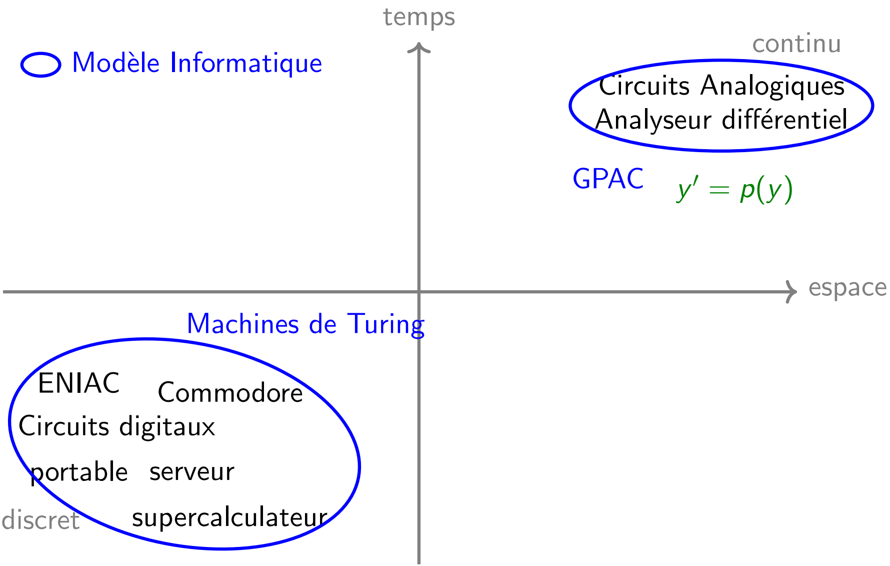
$$\left\{ \begin{array}{lll} y' & = & f(y(t)) \\ y(0) & = & x \end{array} \right.$$

Ici: $X = \mathbb{R}$ or $X = \mathbb{R}^d$.

Une première classification



Une première classification



Plus de modèles!

Machines physiques	**Modèle**
Portable, ...	Machines de Turing λ-calcul Fonctions récursives Circuits Systèmes dynamiques discrets
Analyseur différentiel, ...	GPAC Systèmes dynamiques continus

Plus de modèles!

Machines physiques	**Modèle**
Portable, ...	Machines de Turing λ-calcul Fonctions récursives Circuits Systèmes dynamiques discrets
Analyseur différentiel, ...	GPAC Systèmes dynamiques continus

Thèse de Church-Turing

Tous les modèles raisonnables de calcul sont équivalents.

Plus de modèles!

Machines physiques	**Modèle**
Portable, ...	Machines de Turing λ-calcul Fonctions récursives Circuits Systèmes dynamiques discrets
Analyseur différentiel, ...	GPAC Systèmes dynamiques continus

Thèse de Church-Turing

Tous les modèles raisonnables de calcul sont équivalents.

Corollaire implicite

Il y a des modèles qui sont trop généraux/non-raisonnables.

Plus de modèles!

Machines physiques	**Modèle**
Portable, ...	Machines de Turing λ-calcul Fonctions récursives Circuits Systèmes dynamiques discrets
Analyseur différentiel, ...	**GPAC$\rightarrow$raisonable ?** Systèmes dynamiques continus

Thèse de Church-Turing

Tous les modèles raisonnables de calcul sont équivalents.

Corollaire implicite

Il y a des modèles qui sont trop généraux/non-raisonnables.

Menu

Le General Purpose Analog Computer de Shannon

- Le GPAC est une abstraction mathématique dûe à Claude Shannon (1941) des Analyseurs Différentiels.

- [Graça Costa 03]: Cela correspond aux systèmes dynamiques continus de la forme

$$\begin{cases} y(0) = y_0 \\ y'(t) = p(y(t)) \end{cases}$$

  où

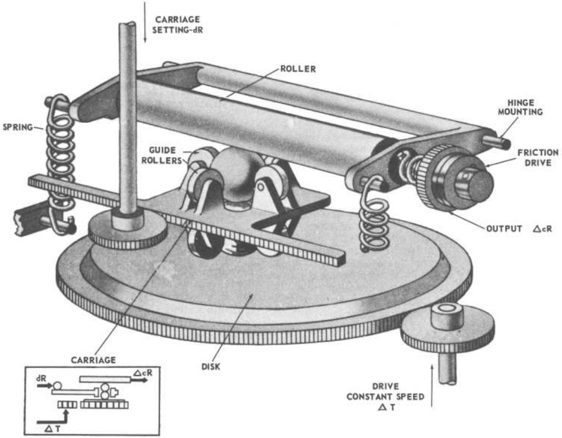
  - $y : I \to \mathbb{R}^n, t \in I$
  - et p est un (vecteur de) polynômes.

Les Analyseurs Différentiels Mécaniques

L'analyseur différentiel mécanique de Vannevar Bush's (1938)

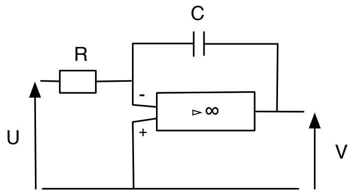
- Principes sous-jacents: Lord Kelvin 1876.
- Premier construit: Vannevar Bush 1931 au MIT.
- Applications: de la ballistique à la conception d'avions
- Intensivement utilisés lors de l'effort de guerre.
- Versions électroniques de la fin des années 40, jusqu'aux années 70

Composant (mécanique) clé



Bureau of Naval Personnel, *Basic Machines and How They Work*, 1964

Composant plus moderne

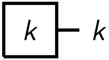


$$V(t) = -1/RC \int_0^t U(t)dt$$

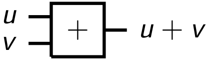
Le GPAC

Une abstraction mathématique de Claude Shannon (1941) des analyseurs différentiels.

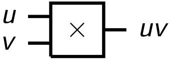
- Blocs de base:



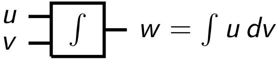
Bloc constante



Bloc addition

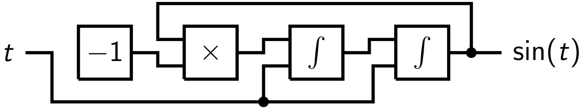


Bloc multiplication



Bloc intégration

Exemple: Générer cos et sin avec un GPAC



$$\begin{cases} y'(t) = z(t) \\ z'(t) = -y(t) \\ y(0) = 0 \\ z(0) = 1 \end{cases} \Rightarrow \begin{cases} y(t) = \sin(t) \\ z(t) = \cos(t) \end{cases}$$

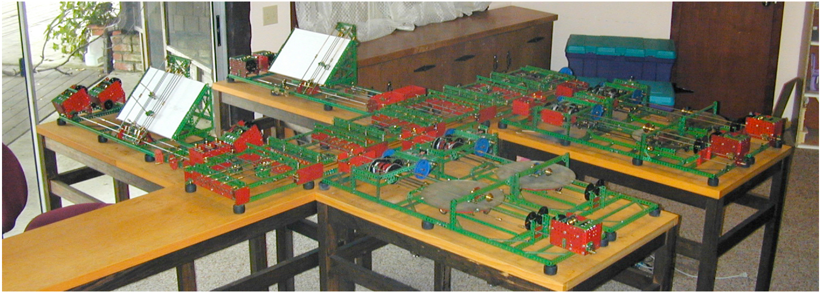
Publicité de Scientific American, Mars 1953.

Voir aussi:

- **Bernd Ulmann. Analog computing.** Walter de Gruyter, 2013.
  or
- **Doug Coward's Analog Computer Museum**

http://dcoward.best.vwh.net/analog/

Analyseurs Différentiels en Mécano



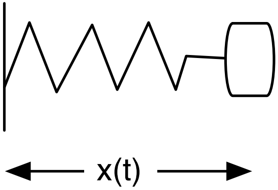
Douglas Rayner Hartree, 1933

Tim Robinson, 2004

Some videos

- https://www.youtube.com/watch?v=36LHGAonlDA
- https://www.youtube.com/watch?v=FctKVbAwHg8
- https://www.youtube.com/watch?v=hIinz4fKGpo

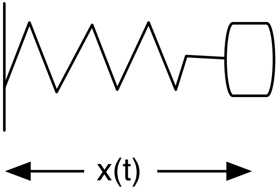
Générer des fonctions avec des GPACs: Exemple. Oscillateur Harmonique



EDO 1:

$$x'' + p^2 x = 0$$

Générer des fonctions avec des GPACs: Exemple. Oscillateur Harmonique

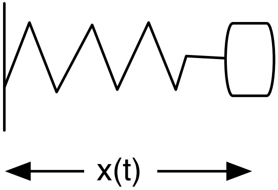


EDO 1:

$$x'' + p^2 x = 0$$

Solution $x(t) = sin(At + B)$

Générer des fonctions avec des GPACs: Exemple. Oscillateur Harmonique



EDO 1:

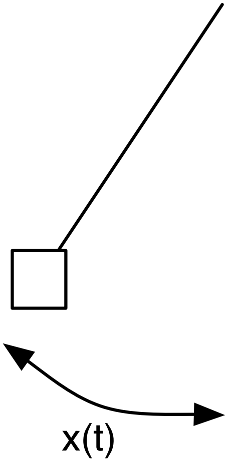
$$x'' + p^2 x = 0$$

Solution $x(t) = sin(At + B)$

EDO 2:

$$\left\{ \begin{array}{lll} x' & = & y \\ y' & = & -p^2 x. \end{array} \right.$$

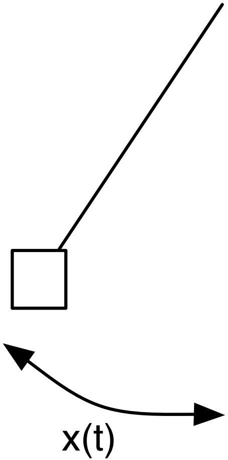
Générer des fonctions avec des GPACs: Exemple. Pendule



Supposons que l'on veut résoudre

$$x'' + p^2 \sin(x) = 0.$$

Générer des fonctions avec des GPACs: Exemple. Pendule



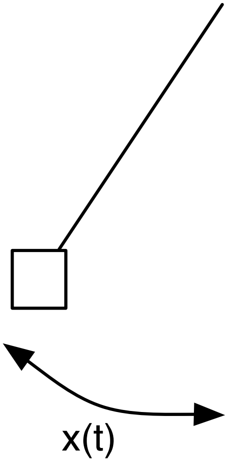
Supposons que l'on veut résoudre

$$x'' + p^2 \sin(x) = 0.$$

Programme:
On définit

$$\begin{cases} y &= x' \\ z &= \sin(x) \\ u &= \cos(x) \end{cases}$$

Générer des fonctions avec des GPACs: Exemple. Pendule



Supposons que l'on veut résoudre

$$x'' + p^2 \sin(x) = 0.$$

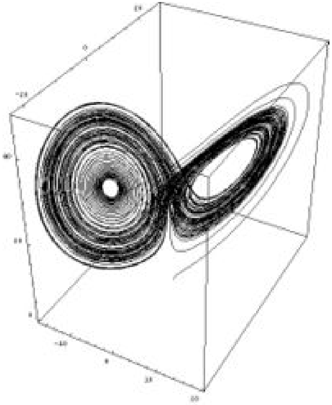
Programme:
On définit

$$\left\{ \begin{array}{lcl} y & = & x' \\ z & = & \sin(x) \\ u & = & \cos(x) \end{array} \right.$$

pour obtenir

$$\left\{ \begin{array}{lcl} x' & = & y \\ y' & = & -p^2 z \\ z' & = & yu \\ u' & = & -yz \end{array} \right.$$

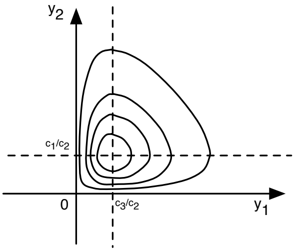
Système de Lorenz (1963)



$$\begin{cases} x' &= \sigma(y-x) \\ y' &= \rho x - y - xz \\ z' &= xz - bz \end{cases}$$

Image pour: $\sigma = 10$, $\rho = 28$, $b = 8/3$

Equations de Lotka/Volterra



$$\begin{cases} y_1' &= c_1 y_1 - c_2 y_1 y_2 \\ y_2' &= c_2 y_1 y_2 - c_3 y_2 \end{cases}$$

Un autre exemple

Le problème de Cauchy:

$$\begin{cases} x_1' &= \sin^2 x_2 \\ x_2' &= x_1 \cos x_2 - e^{x_1+t} \end{cases} \qquad \begin{cases} x_1(0) &= 0 \\ x_2(0) &= 0 \end{cases}$$

Un autre exemple

Le problème de Cauchy:

$$\left\{ \begin{array}{rcl} x_1' & = & \sin^2 x_2 \\ x_2' & = & x_1 \cos x_2 - e^{x_1+t} \end{array} \right. \qquad \left\{ \begin{array}{rcl} x_1(0) & = & 0 \\ x_2(0) & = & 0 \end{array} \right.$$

peut se ramener au problème de Cauchy

$$\left\{ \begin{array}{l} \\ \\ \\ \\ \end{array} \right. \qquad \left\{ \begin{array}{l} \\ \\ \\ \\ \end{array} \right.$$

où $y_1(t) = x_1(t)$ et $y_2(t) = x_2(t)$.

Un autre exemple

Le problème de Cauchy:

$$\begin{cases} x_1' &= \sin^2 x_2 \\ x_2' &= x_1 \cos x_2 - e^{x_1+t} \end{cases} \qquad \begin{cases} x_1(0) &= 0 \\ x_2(0) &= 0 \end{cases}$$

peut se ramener au problème de Cauchy

$$\begin{cases} y_1' &= y_3^2 \\ \\ \\ \\ \end{cases} \qquad \begin{cases} y_1(0) &= 0 \\ \\ \\ \\ \end{cases}$$

où $y_1(t) = x_1(t)$ et $y_2(t) = x_2(t)$.
en considérant $y_3 = \sin x_2$,

Un autre exemple

Le problème de Cauchy:

$$\left\{ \begin{array}{lcl} x_1' & = & \sin^2 x_2 \\ x_2' & = & x_1 \cos x_2 - e^{x_1+t} \end{array} \right. \qquad \left\{ \begin{array}{lcl} x_1(0) & = & 0 \\ x_2(0) & = & 0 \end{array} \right.$$

peut se ramener au problème de Cauchy

$$\left\{ \begin{array}{lcl} y_1' & = & y_3^2 \\ y_2' & = & y_1 y_4 - y_5 \\ & & \\ & & \\ & & \end{array} \right. \qquad \left\{ \begin{array}{lcl} y_1(0) & = & 0 \\ y_2(0) & = & 0 \\ & & \\ & & \\ & & \end{array} \right.$$

où $y_1(t) = x_1(t)$ et $y_2(t) = x_2(t)$.
en considérant $y_3 = \sin x_2$, $y_4 = \cos x_2$,

Un autre exemple

Le problème de Cauchy:

$$\begin{cases} x_1' &=& \sin^2 x_2 \\ x_2' &=& x_1 \cos x_2 - e^{x_1+t} \end{cases} \qquad \begin{cases} x_1(0) &=& 0 \\ x_2(0) &=& 0 \end{cases}$$

peut se ramener au problème de Cauchy

$$\begin{cases} y_1' &=& y_3^2 \\ y_2' &=& y_1 y_4 - y_5 \\ y_3' &=& y_4(y_1 y_4 - y_5) \\ & & \\ & & \end{cases} \qquad \begin{cases} y_1(0) &=& 0 \\ y_2(0) &=& 0 \\ y_3(0) &=& 0 \\ & & \\ & & \end{cases}$$

où $y_1(t) = x_1(t)$ et $y_2(t) = x_2(t)$.
en considérant $y_3 = \sin x_2,\ y_4 = \cos x_2,\ y_5 = e^{x_1+t}$

Un autre exemple

Le problème de Cauchy:

$$\begin{cases} x_1' &= \sin^2 x_2 \\ x_2' &= x_1 \cos x_2 - e^{x_1+t} \end{cases} \qquad \begin{cases} x_1(0) &= 0 \\ x_2(0) &= 0 \end{cases}$$

peut se ramener au problème de Cauchy

$$\begin{cases} y_1' &= y_3^2 \\ y_2' &= y_1 y_4 - y_5 \\ y_3' &= y_4(y_1 y_4 - y_5) \\ y_4' &= -y_3(y_1 y_4 - y_5) \end{cases} \qquad \begin{cases} y_1(0) &= 0 \\ y_2(0) &= 0 \\ y_3(0) &= 0 \\ y_4(0) &= 1 \end{cases}$$

où $y_1(t) = x_1(t)$ et $y_2(t) = x_2(t)$.
en considérant $y_3 = \sin x_2,\ y_4 = \cos x_2,\ y_5 = e^{x_1+t}$

Un autre exemple

Le problème de Cauchy:

$$\begin{cases} x_1' &= \sin^2 x_2 \\ x_2' &= x_1 \cos x_2 - e^{x_1+t} \end{cases} \qquad \begin{cases} x_1(0) &= 0 \\ x_2(0) &= 0 \end{cases}$$

peut se ramener au problème de Cauchy

$$\begin{cases} y_1' &= y_3^2 \\ y_2' &= y_1 y_4 - y_5 \\ y_3' &= y_4(y_1 y_4 - y_5) \\ y_4' &= -y_3(y_1 y_4 - y_5) \\ y_5' &= y_5(y_3 + 1) \end{cases} \qquad \begin{cases} y_1(0) &= 0 \\ y_2(0) &= 0 \\ y_3(0) &= 0 \\ y_4(0) &= 1 \\ y_5(0) &= 1 \end{cases}$$

où $y_1(t) = x_1(t)$ et $y_2(t) = x_2(t)$.
en considérant $y_3 = \sin x_2,\ y_4 = \cos x_2,\ y_5 = e^{x_1+t}$

Propriétés formelles: Propriétés de clôture

- Si f et g sont deux fonctions pIVP, alors
  - $f+g$ est une fonction pIVP.
  - $f-g$ est une fonction pIVP.
  - $f \times g$ est une fonction pIVP.
  - f/g est une fonction pIVP.
  - $f \circ g$ est une fonction pIVP.
  - f' est une fonction pIVP.
  - Si f est une fonction pIVP injective, alors f^{-1} est une fonction pIVP.

- Tout problème de Cauchy

$$\begin{cases} x' & = & f(t,x) \\ x(0) & = & x_0 \end{cases}$$

sur $\mathbb{R}^n$ où chaque composante de f est une fonction pIVP peut être mise sous forme d'une EDO polynomiale.

De [Graça 2007]

Digression: A propos du papier de Shannon en 1941

- La caractérisation de Shannon en 1941 n'est pas complète. Plusieurs problèmes à propos des définitions, corrigées par [PourEl-Richards74], [Lipshitz-Rubel87], [Graça-Costa03].

- En très (trop) bref:
  - Shannon suppose que les circuits considérés ont une sortie, et une sortie unique.
  - Le résultat qui relie les GPAC avec les fonctions d.a. possède un bug.

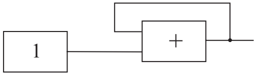


Figure 1: A circuit that admits no solutions as outputs.

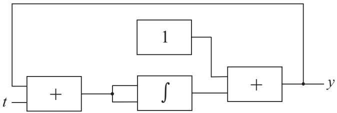


Figure 2: A circuit that admits two distinct solutions as outputs.

Résultats de non-calculabilité (non-générabilité)

Consequence: Une fonction pIVP (fonction unaire $f : I \subset \mathbb{R} \to \mathbb{R}$ générée par un GPAC) doit être différentiellement algébrique (d.a.): [Shannon'41]

i.e. elle satisfait une équation différentielle algébrique de la forme $p\left(t, y, y', \ldots, y^{(n)}\right) = 0$, où p est un polynôme non-nul de toutes ses variables.

Fonctions non-d.a.:

- Fonction Gamma $\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt$ [Hölder 1887].
- Fonction Zeta de Riemann $\zeta(x) = \sum_{k=0}^\infty \frac{1}{k^x}$ [Hilbert].

Conséquences:

GPAC générable $\subsetneq$ calculable (par machine de Turing).

Menu

Conséquence

Affirmation?

GPAC générable $\subseteq$ Differentiellement Algébrique $\subsetneq$ Calculable.

- Cependant, la notion de fonction générée par GPAC suppose des calculs en "temps réel" - une notion très restreinte de calcul.

A propos des fonctions calculées

Shannon considère une notion de calcul très restreinte.

Générable	Calculable

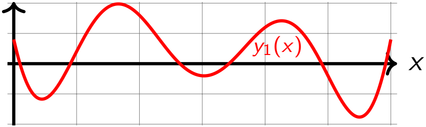
A propos des fonctions calculées

Shannon considère une notion de calcul très restreinte.

Générable

$$\begin{cases} y(0) = y_0 \\ y'(x) = p(y(x)) \end{cases} \qquad x \in \mathbb{R}$$

$$f(x) = y_1(x)$$



Calculable

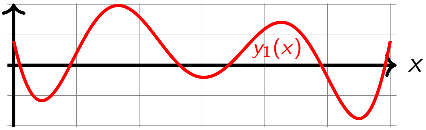
A propos des fonctions calculées

Shannon considère une notion de calcul très restreinte.

Générable

$$\begin{cases} y(0) = y_0 \\ y'(x) = p(y(x)) \end{cases} \qquad x \in \mathbb{R}$$

$$f(x) = y_1(x)$$

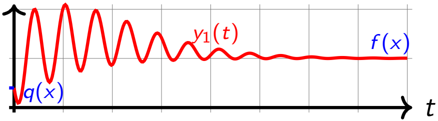


Notion de Shannon

Calculable

$$\begin{cases} y(0) = q(x) & x \in \mathbb{R} \\ y'(t) = p(y(t)) & t \in \mathbb{R}_+ \end{cases}$$

$$f(x) = \lim_{t \to \infty} y_1(t)$$

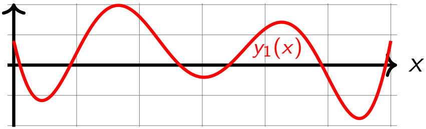


Notion moderne

A propos des fonctions calculées

Shannon considère une notion de calcul très restreinte.

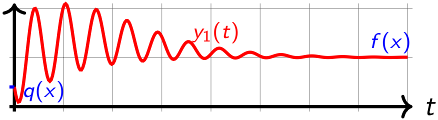
Générable



$\sin, \cos, \exp, \log, \ldots$

Plus faible que les fonctions calculables
[Shannon 41]

Calculable



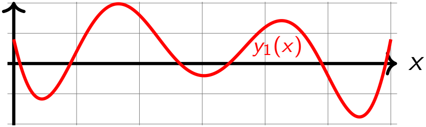
$\sin, \cos, \exp, \log, \Gamma, \zeta, \ldots$

Puissance de Turing
[B. Campagnolo Graça Hainry 2006]

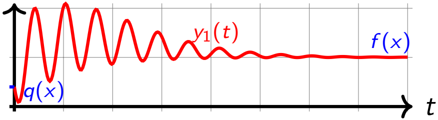
A propos des fonctions calculées

Shannon considère une notion de calcul très restreinte.

Générable

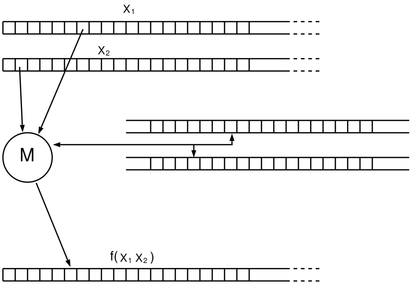


Calculable



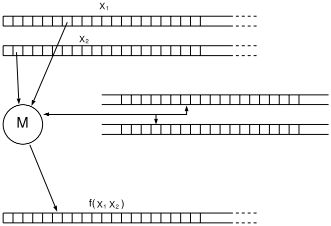
- **Théorème** Le GPAC est équivalent aux machines de Turing

Calculabilité classique: machines de Turing



Thèse de Church	"What is effectively calculable is computable"
Thèse M	"What can be calculated by a machine is computable

Modèle fondamental de la calculabilité: machines de Turing



- **Observation fondamentale:**
  - Une machine de Turing est un système dynamique à temps discret particulier.

Idée 1 : Une vue alternative des machines de Turing

- Soit M une machine de Turing avec un ruban, et m états, et 10 symboles.
- Si

$$...B\ B\ B\ a_{-k}\ a_{-k+1} ...\ a_{-1}\ a_0\ a_1 ...\ a_n\ B\ B\ B ...$$

est le contenu du ruban de M, il peut être vu comme

$$\begin{aligned} y_1 &= a_0 a_1 ... a_n \\ y_2 &= a_{-1} a_{-2} ... a_{-k} \end{aligned} \tag{2}$$

- La configuration de M est alors donnée par 3 valeurs: son état s, y_1 et y_2.
- La fonction de transition de M correspond alors à une fonction $\omega : Q \times \Sigma^* \times \Sigma^* \to Q \times \Sigma^* \times \Sigma^*$

Idée 1 : Une vue alternative des machines de Turing

- Soit M une machine de Turing avec un ruban, et m états, et 10 symboles.
- Si

$$...B\ B\ B\ a_{-k}\ a_{-k+1} ...\ a_{-1}\ a_0\ a_1 ...\ a_n\ B\ B\ B ...$$

est le contenu du ruban de M, il peut être codé comme

$$\begin{aligned} y_1 &= a_0 10^{-1} + a_1 10^{-2} + ... + a_n 10^{-n-1} \\ y_2 &= a_{-1} 10^{-1} + a_{-2} 10^{-2} + ... + a_{-k} 10^{-k}. \end{aligned} \qquad (2)$$

- La configuration de M est alors donnée par 3 valeurs: son état s, y_1 et y_2.
- La fonction de transition de M correspond alors à une fonction $\omega : \mathbb{Q}^3 \to \mathbb{Q}^3$

ω?

$$
\begin{aligned}
y_1 &= a_0 10^{-1} + a_1 10^{-2} + \ldots + a_n 10^{-n-1} \\
y_2 &= a_{-1} 10^{-1} + a_{-2} 10^{-2} + \ldots + a_{-k} 10^{-k}.
\end{aligned}
\tag{3}
$$

Machine de Turing	
Espace d'états $\{q_1, q_2, \cdots, q_m\} \times \Sigma^*$	
Etat $(q_i, a_{-m} \ldots a_{-1}, a_0 \ldots a_n)$	

ω?

$$
\begin{aligned}
y_1 &= a_0 10^{-1} + a_1 10^{-2} + \ldots + a_n 10^{-n-1} \\
y_2 &= a_{-1} 10^{-1} + a_{-2} 10^{-2} + \ldots + a_{-k} 10^{-k}.
\end{aligned}
\qquad (3)
$$

Machine de Turing	
Espace d'états $\{q_1, q_2, \cdots, q_m\} \times \Sigma^*$	Espace d'états $[1, m+1] \times [0, 1]$
Etat $(q_i, a_{-m}...a_{-1}, a_0...a_n)$	Etat $x = s + y_2,\ y = y_1$

ω?

$$\begin{aligned} y_1 &= a_0 10^{-1} + a_1 10^{-2} + \ldots + a_n 10^{-n-1} \\ y_2 &= a_{-1} 10^{-1} + a_{-2} 10^{-2} + \ldots + a_{-k} 10^{-k}. \end{aligned} \quad (3)$$

Machine de Turing	
Espace d'états $\{q_1, q_2, \cdots, q_m\} \times \Sigma^*$	Espace d'états $[1, m+1] \times [0, 1]$
Etat $(q_i, a_{-m} \ldots a_{-1}, a_0 \ldots a_n)$	Etat $x = s + y_2,\ y = y_1$
q_1: if 2 is read, then write 4; goto q_2	

ω?

$$
\begin{aligned}
y_1 &= a_0 10^{-1} + a_1 10^{-2} + \ldots + a_n 10^{-n-1} \\
y_2 &= a_{-1} 10^{-1} + a_{-2} 10^{-2} + \ldots + a_{-k} 10^{-k}.
\end{aligned} \tag{3}
$$

Machine de Turing	
Espace d'états $\{q_1, q_2, \cdots, q_m\} \times \Sigma^*$	Espace d'états $[1, m+1] \times [0, 1]$
Etat $(q_i, a_{-m}\ldots a_{-1}, a_0\ldots a_n)$	Etat $x = s + y_2,\ y = y_1$
q_1: if 2 is read, then write 4; goto q_2	$\begin{cases} x := x+1 \\ y := y + \frac{2}{10} \end{cases}$ if $\begin{cases} 1 \leq x < 2 \\ \frac{2}{10} \leq y < \frac{3}{10} \end{cases}$

ω?

$$\begin{aligned} y_1 &= a_0 10^{-1} + a_1 10^{-2} + ... + a_n 10^{-n-1} \\ y_2 &= a_{-1} 10^{-1} + a_{-2} 10^{-2} + ... + a_{-k} 10^{-k}. \end{aligned} \quad (3)$$

Machine de Turing	
Espace d'états $\{q_1, q_2, \cdots, q_m\} \times \Sigma^*$	Espace d'états $[1, m+1] \times [0, 1]$
Etat $(q_i, a_{-m}...a_{-1}, a_0...a_n)$	Etat $x = s + y_2,\ y = y_1$
q_1: if 2 is read, then write 4; goto q_2	$\begin{cases} x := x+1 \\ y := y + \frac{2}{10} \end{cases}$ if $\begin{cases} 1 \leq x < 2 \\ \frac{2}{10} \leq y < \frac{3}{10} \end{cases}$
q_5: if 3 is read, then move right; goto q_1	

ω?

$$\begin{aligned} y_1 &= a_0 10^{-1} + a_1 10^{-2} + \ldots + a_n 10^{-n-1} \\ y_2 &= a_{-1} 10^{-1} + a_{-2} 10^{-2} + \ldots + a_{-k} 10^{-k}. \end{aligned} \quad (3)$$

Machine de Turing	
Espace d'états $\{q_1, q_2, \cdots, q_m\} \times \Sigma^*$	Espace d'états $[1, m+1] \times [0,1]$
Etat $(q_i, a_{-m} \ldots a_{-1}, a_0 \ldots a_n)$	Etat $x = s + y_2$, $y = y_1$
q_1: if 2 is read, then write 4; goto q_2	$\begin{cases} x := x+1 \\ y := y + \frac{2}{10} \end{cases}$ if $\begin{cases} 1 \leq x < 2 \\ \frac{2}{10} \leq y < \frac{3}{10} \end{cases}$
q_5: if 3 is read, then move right; goto q_1	$\begin{cases} x := \frac{x-5}{10} + \frac{3}{10} + 1 \\ y := 10 * y - 3 \end{cases}$ if $\begin{cases} 5 \leq x < 6 \\ \frac{3}{10} \leq y < \frac{4}{10} \end{cases}$

ω?

$$
\begin{aligned}
y_1 &= a_0 10^{-1} + a_1 10^{-2} + \dots + a_n 10^{-n-1} \\
y_2 &= a_{-1} 10^{-1} + a_{-2} 10^{-2} + \dots + a_{-k} 10^{-k}.
\end{aligned} \tag{3}
$$

Machine de Turing	
Espace d'états $\{q_1, q_2, \cdots, q_m\} \times \Sigma^*$	Espace d'états $[1, m+1] \times [0, 1]$
Etat $(q_i, a_{-m} \dots a_{-1}, a_0 \dots a_n)$	Etat $x = s + y_2,\ y = y_1$
q_1: if 2 is read, then write 4; goto q_2	$\begin{cases} x := x + 1 \\ y := y + \frac{2}{10} \end{cases}$ if $\begin{cases} 1 \leq x < 2 \\ \frac{2}{10} \leq y < \frac{3}{10} \end{cases}$
q_5: if 3 is read, then move right; goto q_1	$\begin{cases} x := \frac{x-5}{10} + \frac{3}{10} + 1 \\ y := 10 * y - 3 \end{cases}$ if $\begin{cases} 5 \leq x < 6 \\ \frac{3}{10} \leq y < \frac{4}{10} \end{cases}$
q_3: if 5 is read, then move left; goto q_7	

ω?

$$
\begin{aligned}
y_1 &= a_0 10^{-1} + a_1 10^{-2} + \ldots + a_n 10^{-n-1} \\
y_2 &= a_{-1} 10^{-1} + a_{-2} 10^{-2} + \ldots + a_{-k} 10^{-k}.
\end{aligned} \tag{3}
$$

Machine de Turing	
Espace d'états $\{q_1, q_2, \cdots, q_m\} \times \Sigma^*$	Espace d'états $[1, m+1] \times [0, 1]$
Etat $(q_i, a_{-m}\ldots a_{-1}, a_0\ldots a_n)$	Etat $x = s + y_2$, $y = y_1$
q_1: if 2 is read, then write 4; goto q_2	$\begin{cases} x := x+1 \\ y := y + \frac{2}{10} \end{cases}$ if $\begin{cases} 1 \leq x < 2 \\ \frac{2}{10} \leq y < \frac{3}{10} \end{cases}$
q_5: if 3 is read, then move right; goto q_1	$\begin{cases} x := \frac{x-5}{10} + \frac{3}{10} + 1 \\ y := 10 * y - 3 \end{cases}$ if $\begin{cases} 5 \leq x < 6 \\ \frac{3}{10} \leq y < \frac{4}{10} \end{cases}$
q_3: if 5 is read, then move left; goto q_7	$\begin{cases} x := 10(x-3) - j + 7 \\ y := \frac{y}{10} + \frac{j}{10} \end{cases}$ if $\begin{cases} 3 + \frac{j}{10} \leq x < 3 + \frac{j+1}{10} \\ \frac{5}{10} \leq y < \frac{6}{10} \end{cases}$

ω?

$$\begin{aligned} y_1 &= a_0 10^{-1} + a_1 10^{-2} + \ldots + a_n 10^{-n-1} \\ y_2 &= a_{-1} 10^{-1} + a_{-2} 10^{-2} + \ldots + a_{-k} 10^{-k}. \end{aligned} \tag{3}$$

Machine de Turing	
Espace d'états $\{q_1, q_2, \cdots, q_m\} \times \Sigma^*$	Espace d'états $[1, m+1] \times [0,1]$
Etat $(q_i, a_{-m} \ldots a_{-1}, a_0 \ldots a_n)$	Etat $x = s + y_2,\ y = y_1$
q_1: if 2 is read, then write 4; goto q_2	$\begin{cases} x := x+1 \\ y := y + \frac{2}{10} \end{cases}$ if $\begin{cases} 1 \leq x < 2 \\ \frac{2}{10} \leq y < \frac{3}{10} \end{cases}$
q_5: if 3 is read, then move right; goto q_1	$\begin{cases} x := \frac{x-5}{10} + \frac{3}{10} + 1 \\ y := 10 * y - 3 \end{cases}$ if $\begin{cases} 5 \leq x < 6 \\ \frac{3}{10} \leq y < \frac{4}{10} \end{cases}$
q_3: if 5 is read, then move left; goto q_7	$\begin{cases} x := 10(x-3) - j + 7 \\ y := \frac{y}{10} + \frac{j}{10} \end{cases}$ if $\begin{cases} 3 + \frac{j}{10} \leq x < 3 + \frac{j+1}{10} \\ \frac{5}{10} \leq y < \frac{6}{10} \end{cases}$ for $j \in \{0, 1, \ldots, 9\}$.

ω?

$$\begin{aligned} y_1 &= a_0 10^{-1} + a_1 10^{-2} + \ldots + a_n 10^{-n-1} \\ y_2 &= a_{-1} 10^{-1} + a_{-2} 10^{-2} + \ldots + a_{-k} 10^{-k}. \end{aligned} \quad (3)$$

Machine de Turing	PAM
Espace d'états $\{q_1, q_2, \cdots, q_m\} \times \Sigma^*$	Espace d'états $[1, m+1] \times [0,1]$
Etat $(q_i, a_{-m} \ldots a_{-1}, a_0 \ldots a_n)$	Etat $x = s + y_2$, $y = y_1$
q_1: if 2 is read, then write 4; goto q_2	$\begin{cases} x := x+1 \\ y := y + \frac{2}{10} \end{cases}$ if $\begin{cases} 1 \leq x < 2 \\ \frac{2}{10} \leq y < \frac{3}{10} \end{cases}$
q_5: if 3 is read, then move right; goto q_1	$\begin{cases} x := \frac{x-5}{10} + \frac{3}{10} + 1 \\ y := 10 * y - 3 \end{cases}$ if $\begin{cases} 5 \leq x < 6 \\ \frac{3}{10} \leq y < \frac{4}{10} \end{cases}$
q_3: if 5 is read, then move left; goto q_7	$\begin{cases} x := 10(x-3) - j + 7 \\ y := \frac{y}{10} + \frac{j}{10} \end{cases}$ if $\begin{cases} 3 + \frac{j}{10} \leq x < 3 + \frac{j+1}{10} \\ \frac{5}{10} \leq y < \frac{6}{10} \end{cases}$ for $j \in \{0, 1, \ldots, 9\}$.

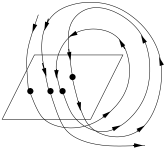
La fonction de transition à une étape d'une machine de Turing est une fonction ω affine par morceaux

Simuler une machine de Turing avec des EDOs

- Idée 1:
  - Le contenu du ruban d'une machine de Turing M avec m états et 10 symboles $\ldots B\,B\,B\,a_{-k}\,a_{-k+1}\ldots a_{-1}\,a_0\,a_1\ldots a_n\,B\,B\,B\ldots$ peut s'encoder par $s = m$

$$x_1 = a_0 + a_1 10 + \ldots + a_n 10^n \qquad x_2 = a_{-1} + a_{-2} 10 + \ldots + a_{-k} 10^{k-1}. \tag{4}$$

  - La configuration de M est alors donnée par trois entiers: son état s, x_1 et x_2.
  - La fonction de transition de M correspond alors à une fonction $\omega : \mathbb{N}^3 \to \mathbb{N}^3$
- Idée 2: plonger cela en $\omega : \mathbb{R}^3 \to \mathbb{R}^3$ de fonction analytique.
- Idée 3: plonger cela en $y' = p(y)$ d'une façon analytique.



Idée 3: Les horloges de Branicky (1995): avec des fonctions non-analytiques

- On alterne des affectations du type $z_2 := \omega(z_1)$, $z_1 := z_2$.

Idée 3: Les horloges de Branicky (1995): avec des fonctions non-analytiques

- On alterne des affectations du type $z_2 := \omega(z_1)$, $z_1 := z_2$.
- Observation clé: la solution de

$$y' = c(g - y)^3 \phi(t)$$

approche à $t = 1/2$ le "but" g avec une précision arbitraire, indépendemment de la condition initiale à $t = 0$

pour toute fonction ϕ d'intégrale poositive si c est suffisement grand.

Idée 3: Les horloges de Branicky (1995): avec des fonctions non-analytiques

- On alterne des affectations du type $z_2 := \omega(z_1)$, $z_1 := z_2$.
- Observation clé: la solution de

$$y' = c(g - y)^3 \phi(t)$$

approche à $t = 1/2$ le "but" g avec une précision arbitraire, indépendemment de la condition initiale à $t = 0$

pour toute fonction ϕ d'intégrale poositive si c est suffisement grand.

  - Si on prefère, cela fait grossièrement $y(1/2) := g$.

Idée 3: Les horloges de Branicky (1995): avec des fonctions non-analytiques

- On alterne des affectations du type $z_2 := \omega(z_1)$, $z_1 := z_2$.
- Observation clé: la solution de

  $$y' = c(g-y)^3\phi(t)$$

  approche à $t = 1/2$ le "but" g avec une précision arbitraire, indépendemment de la condition initiale à $t = 0$

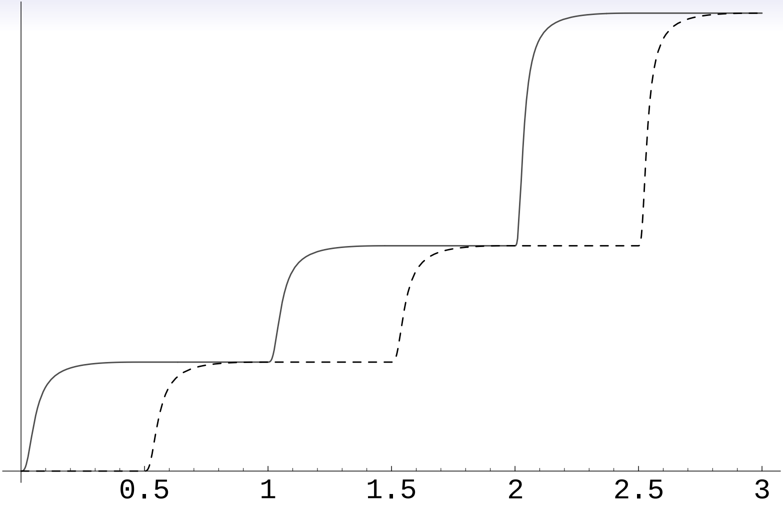
  pour toute fonction ϕ d'intégrale poositive si c est suffisemment grand.

  - Si on prefère, cela fait grossièrement $y(1/2) := g$.
- Le système suivant est une solution:

  $$\left\{\begin{array}{rcl} z_1' & = & c_1(z_2 - z_1)^3\theta(-\sin(2\pi t)) \\ z_2' & = & c_2(\omega(z_1) - z_2)^3\theta(\sin(2\pi t)) \end{array}\right. \quad \left\{\begin{array}{rcl} z_1(0) & = & x_0 \\ z_2(0) & = & x_0 \end{array}\right.$$

  en prenant des fonctions:

  - θ telle que $\theta(x) = 0$ si $x \leq 0$, $\theta(x) = x^2$ si $x \geq 0$.



Simulation des itérations de $h(n) = 2^n$ avec des EDOs.

A part que la vie n'est pas si simple...

- On veut des équations différentielles polynomiales
  - les fonctions non-analytiques (par exemple θ, Δ) sont interdites.

A part que la vie n'est pas si simple...

- On veut des équations différentielles polynomiales
  - les fonctions non-analytiques (par exemple θ, Δ) sont interdites.
- Necessite de "programmer" avec des EDOs
  - et gérer des erreurs.

Résumé de ce que l'on obtient

$$TIME_{TM}(t) \subseteq TIME_{GPAC}(t)$$

Résumé de ce que l'on obtient

$$TIME_{TM}(t) \subseteq TIME_{GPAC}(t)$$

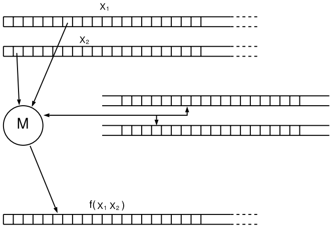
Dit d'une autre façon:

- Au temps $t \in \mathbb{N}$, $y(t) = (s(t), x_1(t), x_2(t))$ encode l'état d'une machine de Turing au temps t.

$$x_1 = a_0 + a_1 10 + \ldots + a_n 10^n \qquad x_2 = a_{-1} + a_{-2} 10 + \ldots + a_{-k} 10^{k-1}.$$

Analyse récursive

Dûe à Turing, Grzegorczyk, Lacombe. Ici présentation de Weihrauch.



Un ruban représente un nombre réel

Chaque nombre réel x est representé par une suite infinite $(x_n)_n \in \mathbb{Q}$,

$$||x_n - x|| \leq 2^{-n}.$$

M se comporte comme une machine de Turing

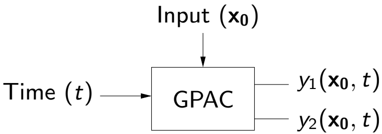
Rubans d'entrée en lecture seule.
Rubans de sortie en écriture seule.
M produit une représentation de $f(x_1, x_2)$ à partir de représentations de x_1, x_2.

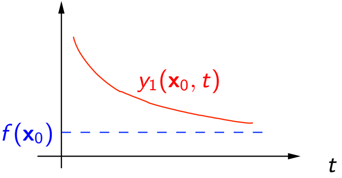
GPAC calculable vs GPAC générable

Definition

Une fonction $f : [a, b] \to \mathbb{R}$ est GPAC-calculable ss'il existe des polynomes calculables $p : \mathbb{R}^{n+1} \to \mathbb{R}^n$, $p_0 : \mathbb{R} \to \mathbb{R}$, et $n - 1$ valeures réelles calculables $\alpha_1, \ldots, \alpha_{n-1}$ telles que:

1. $(y_1, \ldots, y_n)$ est la solution du problème de Cauchy $y' = p(y, t)$ avec la condition initiale $(\alpha_1, \ldots, \alpha_{n-1}, p_0(x))$ au temps $t_0 = 0$
2. $\lim_{t \to \infty} y_2(t) = 0$
3. $|f(x) - y_1(t)| \leq y_2(t)$ pour tout $x \in [a, b]$ et tout $t \in [0, +\infty)$.





Résultat de B., Campagnolo, Graça, Hainry

Theorem

Soient a et b deux réels calculables
Supposons que la fonction $f : [a, b] \to \mathbb{R}$ est $\mathcal{C}^2$.
La fonction $f : [a, b] \to \mathbb{R}$ est calculable si et seulement si elle est GPAC-calculable.

De façon provocative:

- Le GPAC n'est pas plus faible que les machines modernes, d'un point de vue de la calculabilité.

Menu

Moralité & Une question importante

- $TIME_{TM}(t) \subseteq TIME_{GPAC}(t)$.
- Question importante:
  - Formulation 1: Est-ce que le GPAC peut calculer plus vite que les machines de Turing (à un temps polynomial près) ?

Moralité & Une question importante

- $TIME_{TM}(t) \subseteq TIME_{GPAC}(t)$.
- Question importante:
  - Formulation 1: Est-ce que le GPAC peut calculer plus vite que les machines de Turing (à un temps polynomial près) ?
  - Formulation 2: $TIME_{GPAC}(t) \subseteq TIME_{TM}(poly(\lceil t \rceil))$?

Moralité & Une question importante

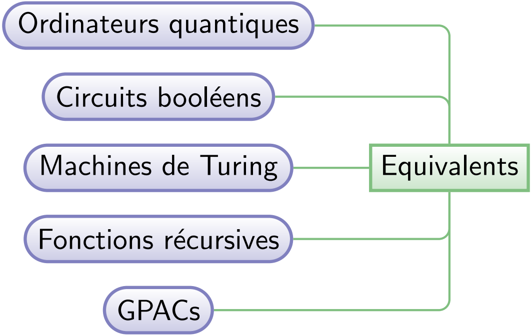
- $TIME_{TM}(t) \subseteq TIME_{GPAC}(t)$.
- Question importante:
  - Formulation 1: Est-ce que le GPAC peut calculer plus vite que les machines de Turing (à un temps polynomial près) ?
  - Formulation 2: $TIME_{GPAC}(t) \subseteq TIME_{TM}(poly(\lceil t \rceil))$?
  - Formulation 3: Est-ce que les équations différentielles polynomiales peuvent être résolues en temps polynomial?

Différents types d'équivalence

- Calculabilité: calculent les mêmes fonctions

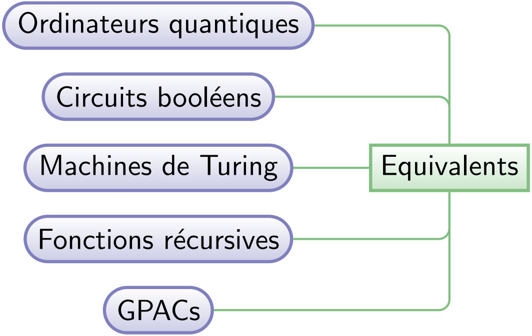
Différents types d'équivalence

- Calculabilité: calculent les mêmes fonctions



Différents types d'équivalence

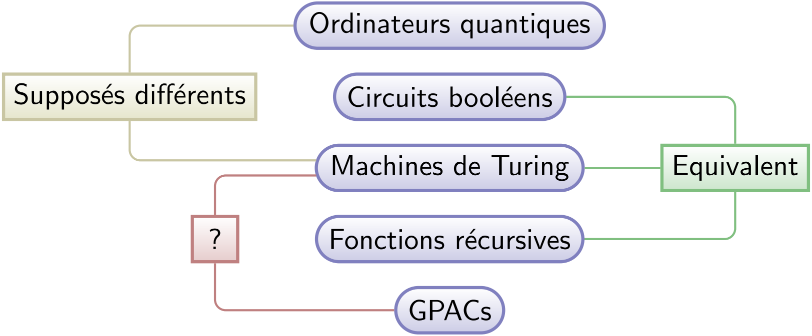
- Calculabilité: calculent les mêmes fonctions



- Complexité: mêmes fonctions avec la même complexité (temps,...)

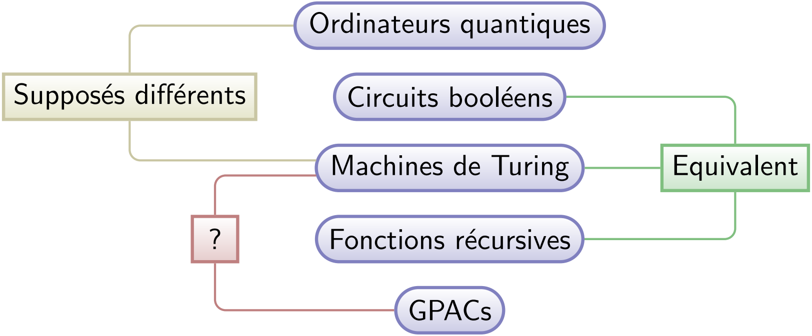
Différents types d'équivalence

- Calculabilité: calculent les mêmes fonctions
- Complexité: mêmes fonctions avec la même complexité (temps,...)



Différents types d'équivalence

- Calculabilité: calculent les mêmes fonctions
- Complexité: mêmes fonctions avec la même complexité (temps,...)



Résultat principal de la thèse d'Amaury Pouly

Les machines de Turing et les GPACs sont équivalents pour la complexité en temps.

Question fondamentale: Est-ce que ce problème peut se résoudre en temps polynomial?

- Peut-on résoudre une équation différentielle en temps polynomial?

Question fondamentale: Est-ce que ce problème peut se résoudre en temps polynomial?

- Peut-on résoudre une équation différentielle en temps polynomial?

- Autrement dit: est-ce que le problème suivant peut se résoudre en temps polynomial?

Problème `PIVP-SOLVE`
Entrée: $y_0 \in \mathbb{R}^d, p : \mathbb{R}^d \to \mathbb{R}^d$ polynôme, $t_0, t \in \mathbb{R}, \varepsilon \in \mathbb{R}_+$
Hypothèses:

$$y(t_0) = y_0 \qquad y'(u) = p(y(u)) \qquad u \in [t_0, t]$$

Sortie: x tel que $\|x - y(t)\|_\infty \leqslant \varepsilon$

Méthode d'Euler

- La méthode d'Euler est
  - une méthode sympathique sur les domaines compacts (bornés).

Méthode d'Euler

- La méthode d'Euler est
  - une méthode sympathique sur les domaines compacts (bornés).
  - mais non polynomiale ... sur des domaines non-bornés !!

Méthode d'Euler

- La méthode d'Euler est
  - une méthode sympathique sur les domaines compacts (bornés).
  - mais non polynomiale ... sur des domaines non-bornés !!
    - L'erreur en approchant la solution d'une EDO $y' = f(y)$, $y(0) = x$ par la méthode d'Euleur avec précision ε on $[0, T]$, en supposant les erreurs d'arrondi bornées par σ, avec N étapes, est donnée par

$$||y(T) - y_N^*|| \leq \frac{h}{\lambda}\left[\frac{R}{2} + \frac{\sigma}{h^2}\right](e^T\lambda - 1), \qquad (5)$$

où y_N^* est l'approximation après N étapes, h le pas de discrétisation, λ la constante de Lipschitz de f sur $[0, T]$, et $R = \max\{||y''(t)||, t \in [0, T]\}$.

Méthode d'Euler

- La méthode d'Euler est
  - une méthode sympathique sur les domaines compacts (bornés).
  - mais non polynomiale …sur des domaines non-bornés !!
    - L'erreur en approchant la solution d'une EDO $y' = f(y)$, $y(0) = x$ par la méthode d'Euleur avec précision ε on $[0, T]$, en supposant les erreurs d'arrondi bornées par σ, avec N étapes, est donnée par

$$||y(T) - y_N^*|| \leq \frac{h}{\lambda}\left[\frac{R}{2} + \frac{\sigma}{h^2}\right](e^T\lambda - 1), \tag{5}$$

      où y_N^* est l'approximation après N étapes, h le pas de discrétisation, λ la constante de Lipschitz de f sur $[0, T]$, et $R = \max\{||y''(t)||, t \in [0, T]\}$.

- Le nombre N d'étapes pour simuler le système est polynomiale en R et $\frac{1}{\varepsilon}$, mais exponentielle en T !!
    - On a $N = T/h = O(\frac{1}{\epsilon}T(e^T\lambda - 1)\frac{1}{\lambda}\left[\frac{R}{2} + \frac{\sigma}{h^2}\right])$.

Avec une méthode moins élémentaire?

- On considère sa méthode préférée.
- La méthode est d'ordre d pour un certain d,
  - h est (essentiellement) remplacé par h^d dans le transparent d'avant
  - toujours non polynomial en T !!

Aucun espoir d'y arriver !!

$$\begin{cases} y_1' &= y_1 \\ y_2' &= y_1 y_2 \\ y_3' &= y_2 y_3 \\ \vdots & \vdots \quad \vdots \\ y_n' &= y_{n-1} y_n \end{cases}$$

Aucun espoir d'y arriver !!

- $e^{e^{e^{\cdots^{e^t}}}}$ est solution de $\left\{\begin{array}{ccc} y_1' & = & y_1 \\ y_2' & = & y_1 y_2 \\ y_3' & = & y_2 y_3 \\ \vdots & \vdots & \vdots \\ y_n' & = & y_{n-1} y_n \end{array}\right.$

Aucun espoir d'y arriver !!

- $e^{e^{e^{\cdots^{e^t}}}}$ est solution de $\left\{\begin{array}{lll} y_1' & = & y_1 \\ y_2' & = & y_1 y_2 \\ y_3' & = & y_2 y_3 \\ \vdots & \vdots & \vdots \\ y_n' & = & y_{n-1} y_n \end{array}\right.$

- Cela ne peut pas se calculer en un temps polynomial sur $\mathbb{R}$,
  - puisque juste écrire cette valeur en binaire ne peut même pas se faire en temps polynomial.

Aucun espoir d'y arriver !!

- $e^{e^{e^{\cdots^{e^t}}}}$ est solution de $\left\{\begin{array}{lcl} y_1' & = & y_1 \\ y_2' & = & y_1 y_2 \\ y_3' & = & y_2 y_3 \\ \vdots & \vdots & \vdots \\ y_n' & = & y_{n-1} y_n \end{array}\right.$

- Cela ne peut pas se calculer en un temps polynomial sur $\mathbb{R}$,
  - puisque juste écrire cette valeur en binaire ne peut même pas se faire en temps polynomial.

- Corollaire: **Les EDO polynomiales ne peuvent pas être résolues en temps polynomial sur $\mathbb{R}$, dans le cas général.**

Aucun espoir d'y arriver !!

- $e^{e^{e^{\cdots^{e^t}}}}$ est solution de $\begin{cases} y_1' &= & y_1 \\ y_2' &= & y_1 y_2 \\ y_3' &= & y_2 y_3 \\ \vdots & \vdots & \vdots \\ y_n' &= & y_{n-1} y_n \end{cases}$

- Cela ne peut pas se calculer en un temps polynomial sur $\mathbb{R}$,
  - puisque juste écrire cette valeur en binaire ne peut même pas se faire en temps polynomial.

- Corollaire: **Les EDO polynomiales ne peuvent pas être résolues en temps polynomial sur $\mathbb{R}$, dans le cas général.**
  - Remarque: Elles peuvent se résoudre en temps polynomial sur $[0, T]$ pour un T fixé.

Complexité en temps pour les systèmes continus

- Turing machines: $T(x) =$ nombre d'étapes sur l'entrée x

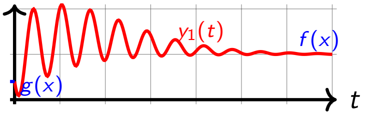
Complexité en temps pour les systèmes continus

- Turing machines: $T(x) =$ nombre d'étapes sur l'entrée x
- GPAC: problème de contraction du temps

Définition intuitive

$T(x, \mu) =$ temps t tel que $|y_1(t) - f(x)| \leqslant e^{-\mu}$

$y(0) = g(x) \qquad y' = h(y)$



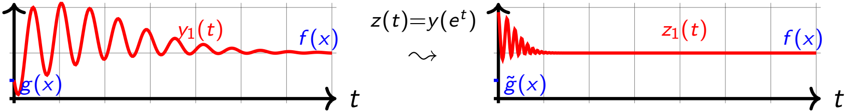
Complexité en temps pour les systèmes continus

- Turing machines: $T(x)$ = nombre d'étapes sur l'entrée x
- GPAC: problème de contraction du temps

Définition intuitive

$T(x,\mu)$ = temps t tel que $|y_1(t) - f(x)| \leqslant e^{-\mu}$

$y(0) = g(x) \qquad y' = h(y) \qquad\qquad z(0) = \tilde{g}(x) \qquad z' = \tilde{h}(z)$



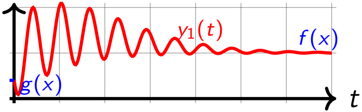
Complexité en temps pour les systèmes continus

- Turing machines: $T(x)$ = nombre d'étapes sur l'entrée x
- GPAC: problème de contraction du temps

Définition intuitive

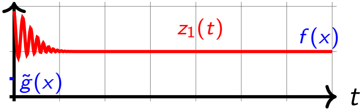
$T(x,\mu)$ = temps t tel que $|y_1(t) - f(x)| \leqslant e^{-\mu}$

$y(0) = g(x) \qquad y' = h(y)$



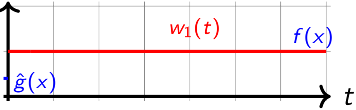
$z(t)=y(e^t)$ $\rightsquigarrow$

$z(0) = \tilde{g}(x) \qquad z' = \tilde{h}(z)$



$w(0) = \hat{g}(x) \qquad w' = \hat{h}(w)$

$w(t)=y\left(e^{e^t}\right)$ $\rightsquigarrow$



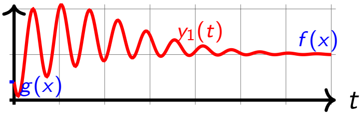
Complexité en temps pour les systèmes continus

- Turing machines: $T(x)$ = nombre d'étapes sur l'entrée x
- GPAC: problème de contraction du temps $\rightarrow$ **problème ouvert**

Définition intuitive

$T(x,\mu)$ = temps t tel que $|y_1(t) - f(x)| \leqslant e^{-\mu}$

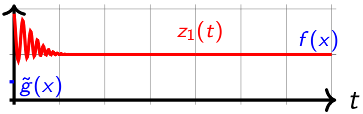
$y(0) = g(x) \qquad y' = h(y)$



$z(t)=y(e^t)$

$\rightsquigarrow$

$z(0) = \tilde{g}(x) \qquad z' = \tilde{h}(z)$



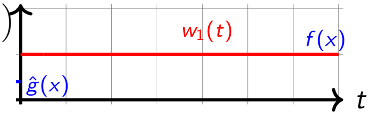
Observation

La définission est caduque: toutes les fonctions ont une complexité en temps arbitrairement petite.

$w(0) = \hat{g}(x) \qquad w' = \hat{h}(w)$

$w(t)=y\left(e^{e^t}\right)$

$\rightsquigarrow$



Contraction en temps en général

- Substitution $t = e^u - 1$ par exemple change la dynamique
  - $\mathcal{M} = (\mathbb{R}^m, f)$ avec ensemble d'acceptation F
  - en $(\mathbb{R}^{m+1}, (g, 1))$ avec ensemble d'acceptation $F \times \mathbb{R}$,

Considèrer

$$z(u) = y(e^u - 1),$$

donne une accélération en temps exponentielle, solution de

$$\frac{dz}{du} = f(z(u))e^u$$

donc correspondant à la dynamique

$$\frac{d(z,u)}{du} = (g(z,u), 1), \qquad \text{with } g(z,u) = f(z)e^u$$

Idée présente dans plusieurs articles (e.g. [Moore95]); présentation ici adaptée de [Ruohonen93]

Contraction en temps en général

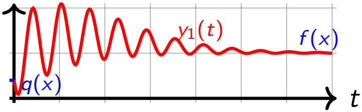
- Substitution $t = e^u - 1$ par exemple change la dynamique
  - $\mathcal{M} = (\mathbb{R}^m, f)$ avec ensemble d'acceptation F
  - en $(\mathbb{R}^{m+1}, (g, 1))$ avec ensemble d'acceptation $F \times \mathbb{R}$,

  Considèrer
  $$z(u) = y(e^u - 1),$$
  donne une accélération en temps exponentielle, solution de
  $$\frac{dz}{du} = f(z(u))e^u$$
  donc correspondant à la dynamique
  $$\frac{d(z, u)}{du} = (g(z, u), 1), \quad \text{with } g(z, u) = f(z)e^u$$
- La substitution $t = \tan(\pi u/2)$ donne une accélération infinie, en compressant tout calcul (même infini en temps) en un temps fini $0 \leq u < 1$.

Idée présente dans plusieurs articles (e.g. [Moore95]); présentation ici adaptée de [Ruohonen93]

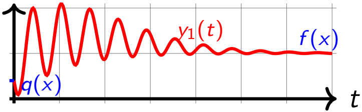
Le GPAC a quelque chose de spécial: La corrélation temps/espace du GPAC

$$y(0) = q(x) \qquad y' = p(y)$$



Le GPAC a quelque chose de spécial: La corrélation temps/espace du GPAC

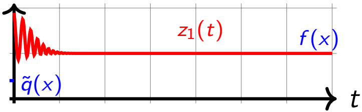
$y(0) = q(x) \qquad y' = p(y)$



$z(t) = y(e^t)$

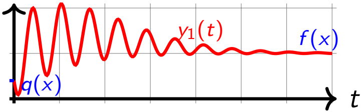
$\leadsto$

$z(0) = \tilde{q}(x) \qquad z' = \tilde{p}(z)$



Le GPAC a quelque chose de spécial: La corrélation temps/espace du GPAC

$y(0) = q(x) \qquad y' = p(y)$

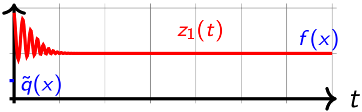


$w(t) = e^t$

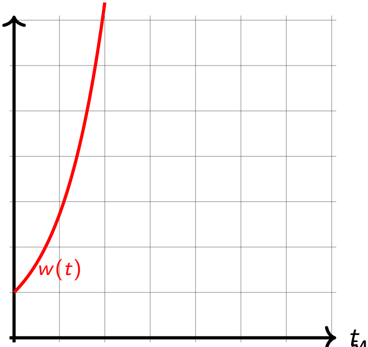
$z(t) = y(w(t))$

$\rightsquigarrow$

$z(0) = \tilde{q}(x) \quad z' = w\,p(z)$

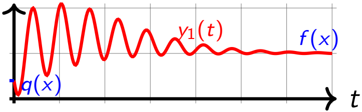


$w(0) = 1 \quad w' = w$



Le GPAC a quelque chose de spécial: La corrélation temps/espace du GPAC

$y(0) = q(x) \qquad y' = p(y)$

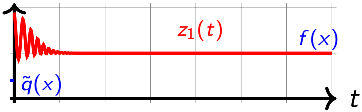


$w(t) = e^t$

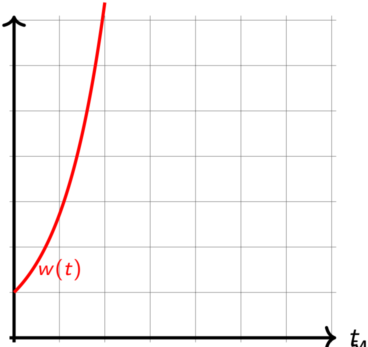
$z(t) = y(w(t))$

$\rightsquigarrow$

$z(0) = \tilde{q}(x) \qquad z' = w\,p(z)$



$w(0) = 1 \qquad w' = w$

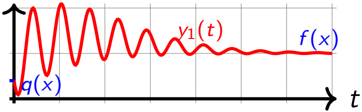


Observation #1

Le temps coûte de l'espace !

Le GPAC a quelque chose de spécial: La corrélation temps/espace du GPAC

$y(0) = q(x) \qquad y' = p(y)$

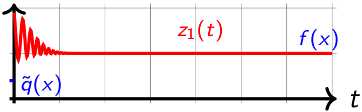


$w(t) = e^t$

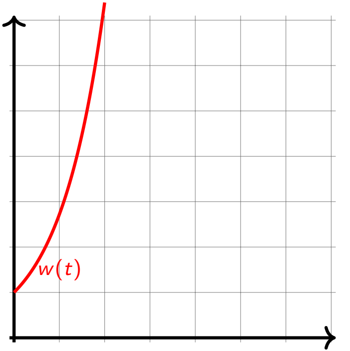
$z(t) = y(w(t))$

$\rightsquigarrow$

$z(0) = \tilde{q}(x) \quad z' = w\,p(z)$



$w(0) = 1 \quad w' = w$



Observation #1

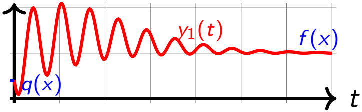
Le temps coûte de l'espace !

Observation #2

C'est une caractéristique du GPAC.

Le GPAC a quelque chose de spécial: La corrélation temps/espace du GPAC

$y(0) = q(x) \qquad y' = p(y)$

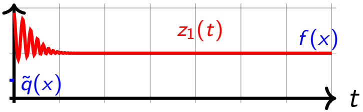


$w(t)=e^t$

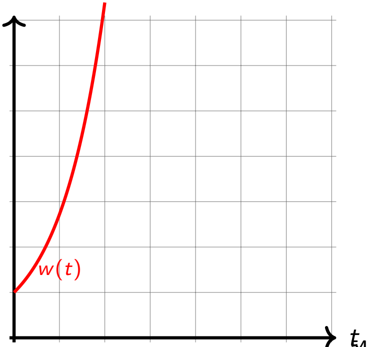
$z(t)=y(w(t))$

$\rightsquigarrow$

$z(0) = \tilde{q}(x) \qquad z' = w\,p(z)$



$w(0) = 1 \qquad w' = w$



Observation #1

Le temps coûte de l'espace !

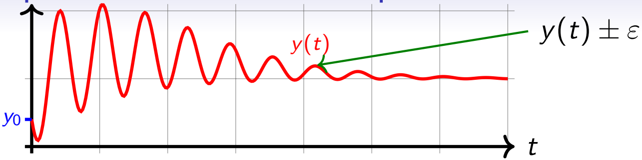
Observation #2

C'est une caractéristique du GPAC.

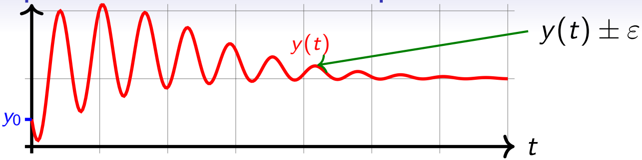
Conclusion

La complexité en temps du GPAC doit impliquer le temps et l'espace!

Complexité de résoudre un problème de Cauchy



Complexité de résoudre un problème de Cauchy



Problème `PIVP-SOLVE`

Entrée: $y_0 \in \mathbb{R}^d, p : \mathbb{R}^d \to \mathbb{R}^d$ polynôme , $t_0, t \in \mathbb{R}, \varepsilon \in \mathbb{R}_+$

Hypothèse:

$$y(t_0) = y_0 \qquad y'(u) = p(y(u)) \qquad u \in [t_0, t]$$

Sortie: x tel que $\|x - y(t)\|_\infty \leqslant \varepsilon$

Complexité de résoudre un problème de Cauchy

Problème `PIVP-SOLVE`

Entrée: $y_0 \in \mathbb{R}^d, p : \mathbb{R}^d \to \mathbb{R}^d$ polynôme , $t_0, t \in \mathbb{R}, \varepsilon \in \mathbb{R}_+$

Hypothèse:

$$y(t_0) = y_0 \qquad y'(u) = p(y(u)) \qquad u \in [t_0, t]$$

Sortie: x tel que $\|x - y(t)\|_\infty \leqslant \varepsilon$

Théorème

On peut calculer x en au plus

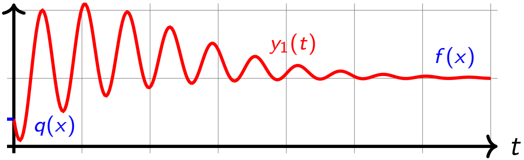
$$poly\left(\deg(p)^d, \ell, \log \|y_0\|_\infty, -\log \varepsilon\right)$$

opérations arithmétiques $(+, \times)$ sur $\mathbb{R}$, où:

$$\ell = \int_{t_0}^{t} \Sigma p \max(1, \|y(u)\|_\infty)^{\deg(p)} du \approx \text{longueur de } y \text{ over } [t_0, t]$$

où Σp est la somme des coefficients de p.

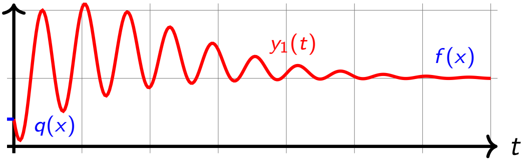
Deux notions équivalentes de complexité



$$\begin{cases} y(0) = q(x) \\ y'(t) = p(y(t)) \end{cases}$$

$$f(x) = \lim_{t \to \infty} y_1(t)$$

Deux notions équivalentes de complexité



$$\begin{cases} y(0) = q(x) \\ y'(t) = p(y(t)) \end{cases}$$

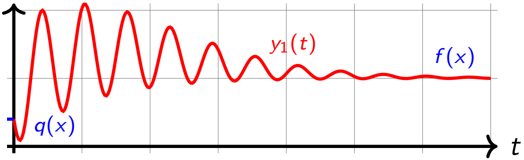
$$f(x) = \lim_{t \to \infty} y_1(t)$$

Basée sur la longueur: T

$$\ell(t) = \text{longueur de } y \text{ over } [0, t]$$
$$= \int_0^t \|p(y(u))\|_\infty \, du$$

$$T(x, \mu) = \text{longueur } \ell(t) \text{ telle que}$$
$$\|y_1(t) - f(x)\|_\infty \leqslant e^{-\mu}$$

Deux notions équivalentes de complexité



$$\begin{cases} y(0) = q(x) \\ y'(t) = p(y(t)) \end{cases}$$

$$f(x) = \lim_{t\to\infty} y_1(t)$$

Basée sur la longueur: T

$$\ell(t) = \text{longueur de } y \text{ over } [0,t] = \int_0^t \|p(y(u))\|_\infty \, du$$

$$T(x,\mu) = \text{longueur } \ell(t) \text{ telle que } \|y_1(t) - f(x)\|_\infty \leqslant e^{-\mu}$$

Basée sur temps/espace: (T,S)

$$T(x,\mu) = \text{temps } t \text{ tel que } \|y_1(t) - f(x)\|_\infty \leqslant e^{-\mu}$$

$$S(x,\mu) = \text{espace jusqu'à } T(x,\mu) = \sup_{[0,T(x,\mu)]} \|y\|_\infty$$

Caractérisation du temps polynomial

Définition: $\mathcal{L} \subseteq \{0,1\}^*$ est polytime-reconnaissable ssi pour tout w:

Caractérisation du temps polynomial

Définition: $\mathcal{L} \subseteq \{0,1\}^*$ est polytime-reconnaissable ssi pour tout w:

$$y(0) = q(\psi(w)) \qquad y' = p(y) \qquad \psi(w) = \sum_{i=1}^{|w|} w_i 2^{-i}$$

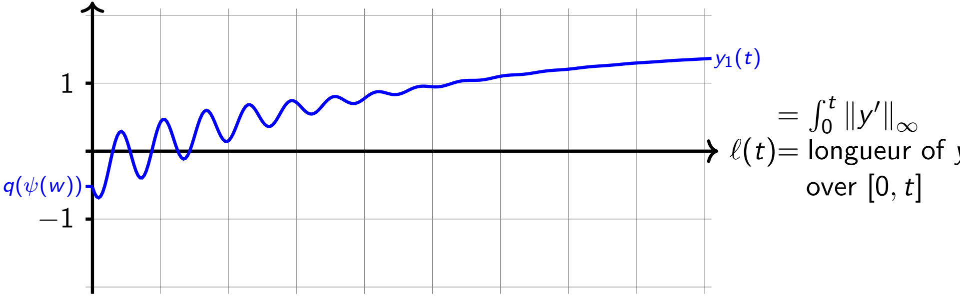
satisfait:

Caractérisation du temps polynomial

Définition: $\mathcal{L} \subseteq \{0,1\}^*$ est polytime-reconnaissable ssi pour tout w:

$$y(0) = q(\psi(w)) \qquad y' = p(y) \qquad \psi(w) = \sum_{i=1}^{|w|} w_i 2^{-i}$$

satisfait:

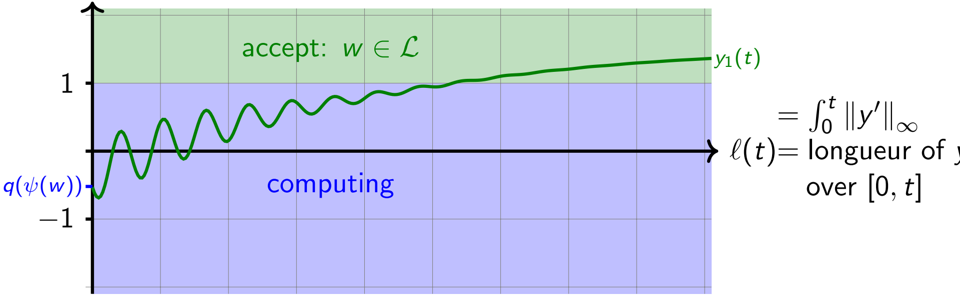


Caractérisation du temps polynomial

Définition: $\mathcal{L} \subseteq \{0,1\}^*$ est polytime-reconnaissable ssi pour tout w:

$$y(0) = q(\psi(w)) \qquad y' = p(y) \qquad \psi(w) = \sum_{i=1}^{|w|} w_i 2^{-i}$$

satisfait:

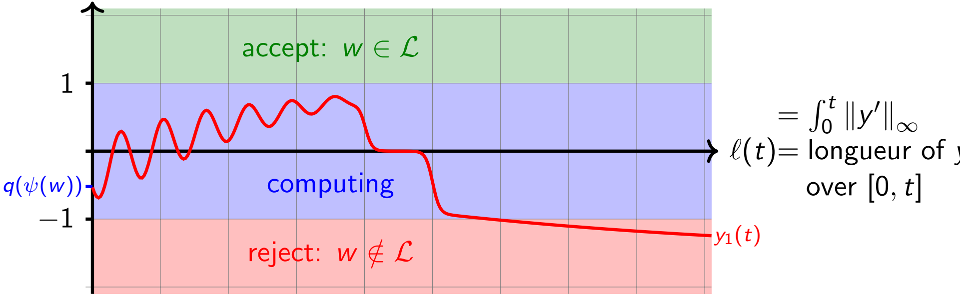


1. si $y_1(t) \geqslant 1$ alors $w \in \mathcal{L}$

Définition: $\mathcal{L} \subseteq \{0,1\}^*$ est polytime-reconnaissable ssi pour tout w:

$$y(0) = q(\psi(w)) \qquad y' = p(y) \qquad \psi(w) = \sum_{i=1}^{|w|} w_i 2^{-i}$$

satisfait:



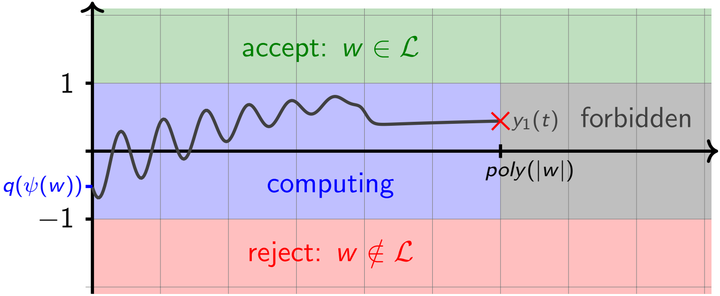
2. si $y_1(t) \leqslant -1$ alors $w \notin \mathcal{L}$

Caractérisation du temps polynomial

Définition: $\mathcal{L} \subseteq \{0,1\}^*$ est polytime-reconnaissable ssi pour tout w:

$$y(0) = q(\psi(w)) \qquad y' = p(y) \qquad \psi(w) = \sum_{i=1}^{|w|} w_i 2^{-i}$$

satisfait:



$= \int_0^t \|y'\|_\infty$

$\ell(t)$= longueur of y

over $[0, t]$

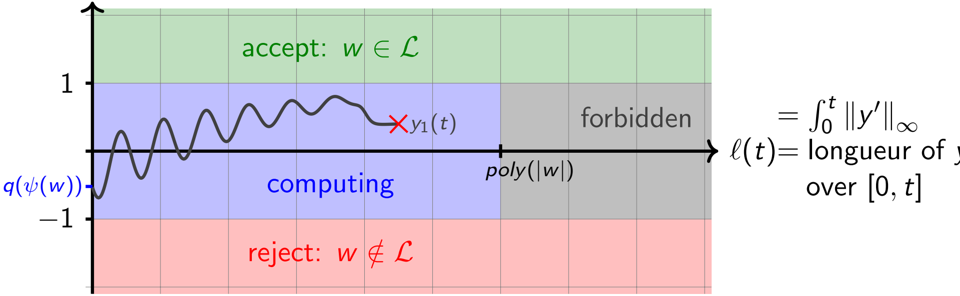
3. si $\ell(t) \geqslant poly(|w|)$ alors $|y_1(t)| \geqslant 1$

Caractérisation du temps polynomial

Définition: $\mathcal{L} \subseteq \{0,1\}^*$ est polytime-reconnaissable ssi pour tout w:

$$y(0) = q(\psi(w)) \qquad y' = p(y) \qquad \psi(w) = \sum_{i=1}^{|w|} w_i 2^{-i}$$

satisfait:



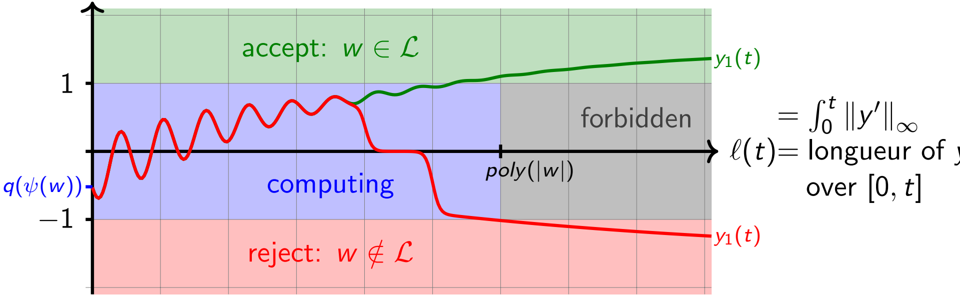
4. $\ell(t) \geqslant t$

Caractérisation du temps polynomial

Définition: $\mathcal{L} \subseteq \{0,1\}^*$ est polytime-reconnaissable ssi pour tout w:

$$y(0) = q(\psi(w)) \qquad y' = p(y) \qquad \psi(w) = \sum_{i=1}^{|w|} w_i 2^{-i}$$

satisfait:



Théorème

$\mathcal{L} \in \mathrm{P}$ si et seulement si $\mathcal{L}$ est polytime-reconnaissable.

Caractérisation du temps polynomial

Définition: $f : [a, b] \to \mathbb{R}$ est analog-polytime ssi pour tout x:

$$y(0) = q(x) \qquad y' = p(y)$$

satisfait:

Caractérisation du temps polynomial

Définition: $f : [a, b] \to \mathbb{R}$ est analog-polytime ssi pour tout x:

$$y(0) = q(x) \qquad y' = p(y)$$

satisfait:

1. $\forall n \in \mathbb{N}$, if $\ell(t) \geqslant poly(\|x\|_\infty, n)$ then $|y_1(t) - f(x)| \leqslant 10^{-n}$

$$\text{where } \ell(t) = \int_0^t \|y'(u)\|_\infty \, du$$

«si la courbe est suffisamment longue, alors la précision est suffisamment bonne »

Caractérisation du temps polynomial

Définition: $f : [a, b] \to \mathbb{R}$ est analog-polytime ssi pour tout x:

$$y(0) = q(x) \qquad y' = p(y)$$

satisfait:

1. $\forall n \in \mathbb{N}$, if $\ell(t) \geqslant poly(\|x\|_\infty, n)$ then $|y_1(t) - f(x)| \leqslant 10^{-n}$

   $$\text{where } \ell(t) = \int_0^t \|y'(u)\|_\infty \, du$$

   «si la courbe est suffisamment longue, alors la précision est suffisamment bonne »

2. $\forall t \in \mathbb{R}_+, \|y'(t)\|_\infty \geqslant 1$

«La longueur croît au moins linéairement avec le temps »

Caractérisation du temps polynomial

Définition: $f : [a, b] \to \mathbb{R}$ est analog-polytime ssi pour tout x:

$$y(0) = q(x) \qquad y' = p(y)$$

satisfait:

1. $\forall n \in \mathbb{N}$, if $\ell(t) \geqslant poly(\|x\|_\infty, n)$ then $|y_1(t) - f(x)| \leqslant 10^{-n}$

$$\text{where } \ell(t) = \int_0^t \|y'(u)\|_\infty \, du$$

   «si la courbe est suffisamment longue, alors la précision est suffisamment bonne »

2. $\forall t \in \mathbb{R}_+, \|y'(t)\|_\infty \geqslant 1$

«La longueur croît au moins linéairement avec le temps »

Résultat principal

$f : [a, b] \to \mathbb{R}$ est calculable en temps polynomial ssi f is analog-polytime.

Menu

Messages pour la maison

- Les ordinateurs ne sont pas nécessairement digitaux !

- Les EDOs de la forme

$$\begin{cases} y' & = & p(y(t)) \\ y(0) & = & y_0 \end{cases} \tag{6}$$

où p est (un vecteur) de polynômes,

ont leurs solutions qui correspondent à une classe de fonctions avec des propriétés très élégantes.

- Les EDOs de la forme

$$\begin{cases} y' & = & p(y(t)) \\ y(0) & = & y_0 \end{cases} \tag{6}$$

où p est (un vecteur) de polynômes,

ont leurs solutions qui correspondent à une classe de fonctions avec des propriétés très élégantes.

- **Une intuition:**
  - l' "analogue" de la notion de fonction calculable pour la calculabilité/complexité discrète.

Calculable vs Générable

- Modèles de calculs via des EDOs:
  - Fonctions GPAC générables doivent être différentiellement algébriques.
  - GPAC générable $\subsetneq$ Calculable.
  - Fonctions GPAC calculables correspondent aux problèmes de Cauchy polynomiaux.

- **Calculabilité GPAC = Calculabilité classique.**
- **Complexité GPAC = Complexité classique.**
  - Si le temps est mesuré par la longueur.
- **Faits inattendus / Complexité implicite:**
  - Les classes de complexité peuvent se définir par des EDOs !!

A (digital/discrete time) Picture

Church Thesis	“What is effectively calculable is computable”
Thesis M	“What can be calculated by a machine is computable”
Thesis?	“What can be calculated by a model is computable”

(following [Copeland2002])

Understanding computational power of models helps to understand

- limits of mechanical reasoning.
- limits of machines.
- limits of models.