A universal differential equation

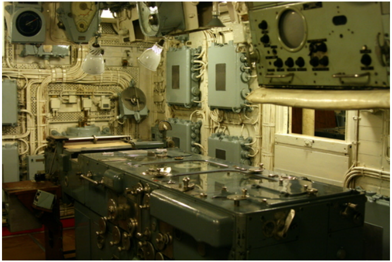
Olivier Bournez, Amaury Pouly

21 mars 2017

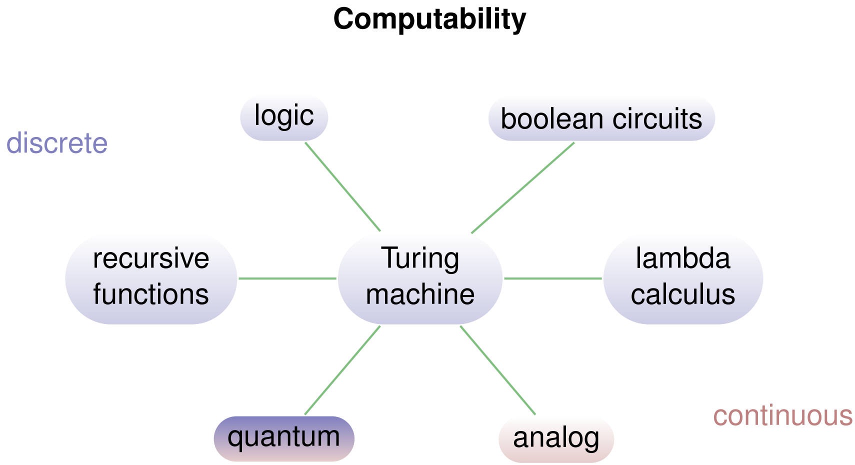
Digital vs analog computers

Digital vs analog computers

VS



Church Thesis

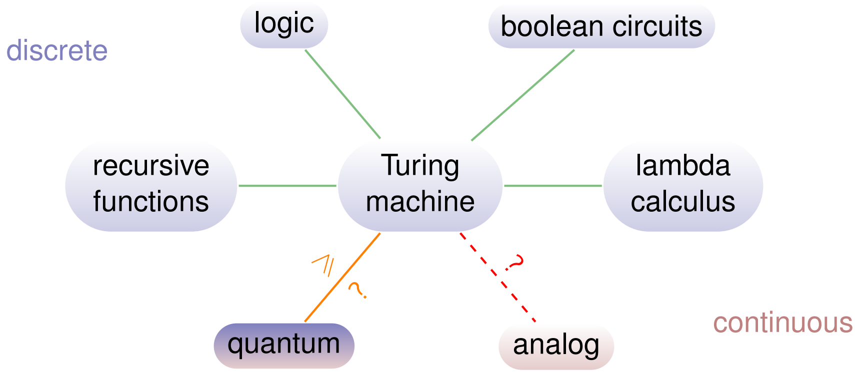


Church Thesis

All reasonable models of computation are equivalent.

Church Thesis

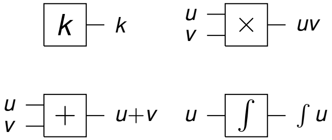
Complexity



Effective Church Thesis

All reasonable models of computation are equivalent for complexity.

Polynomial Differential Equations



General Purpose
Analog Computer

Differential Analyzer

Newton mechanics

Reaction networks :

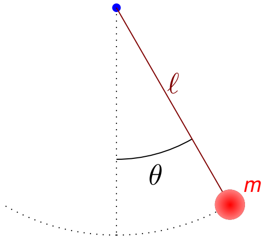
- chemical
- enzymatic

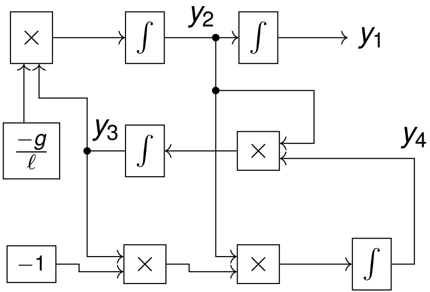
polynomial differential equations :

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

- Rich class
- Stable ($+, \times, \circ, /$, ED)
- No closed-form solution

Example of dynamical system





$$\ddot{\theta} + \frac{g}{\ell}\sin(\theta) = 0$$

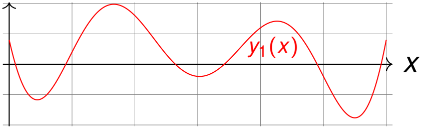
$$\begin{cases} y_1' = y_2 \\ y_2' = -\frac{g}{l} y_3 \\ y_3' = y_2 y_4 \\ y_4' = -y_2 y_3 \end{cases} \Leftrightarrow \begin{cases} y_1 = \theta \\ y_2 = \dot{\theta} \\ y_3 = \sin(\theta) \\ y_4 = \cos(\theta) \end{cases}$$

Computing with the GPAC

Generable functions

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

$f(x) = y_1(x)$



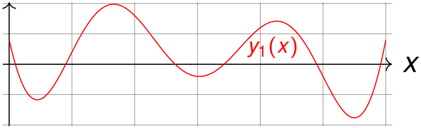
Shannon's notion

Computing with the GPAC

Generable functions

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

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



Shannon's notion

sin, cos, exp, log, ...

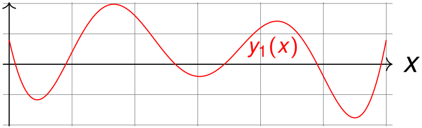
Strictly weaker than Turing machines [Shannon, 1941]

Computing with the GPAC

Generable functions

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

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



Shannon's notion

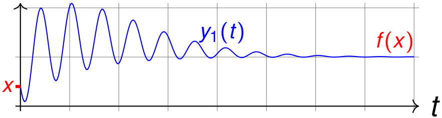
sin, cos, exp, log, ...

Strictly weaker than Turing machines [Shannon, 1941]

Computable

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

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



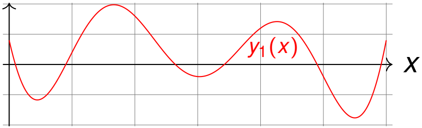
Modern notion

Computing with the GPAC

Generable functions

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

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



Shannon's notion

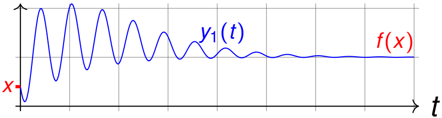
sin, cos, exp, log, ...

Strictly weaker than Turing machines [Shannon, 1941]

Computable

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

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



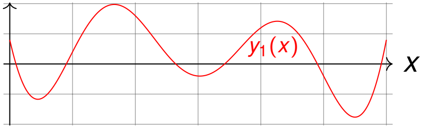
Modern notion

sin, cos, exp, log, Γ, ζ, ...

Turing powerful [Bournez et al., 2007]

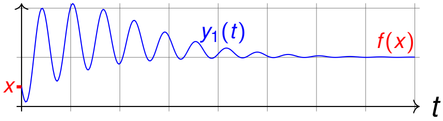
Universal differential equations

Generable functions



subclass of analytic functions

Computable functions



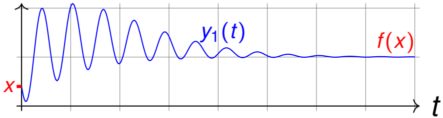
any computable function

Universal differential equations

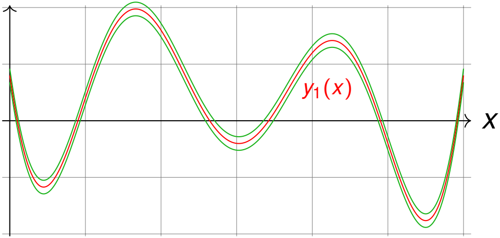
Generable functions

subclass of analytic functions

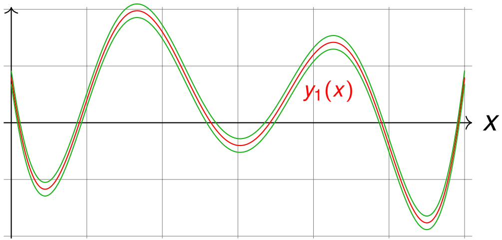
Computable functions



any computable function



Universal differential equation (Rubel)



Theorem (Rubel)

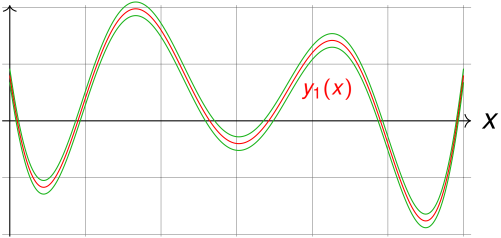
There exists a **fixed** polynomial p and $k \in \mathbb{N}$ such that for any continuous functions f and ε, there exists a solution y to

$$p(y, y', \ldots, y^{(k)}) = 0$$

such that

$$|y(t) - f(t)| \leqslant \varepsilon(t).$$

Universal differential equation (Rubel)



Theorem (Rubel)

There exists a **fixed** polynomial p and $k \in \mathbb{N}$ such that for any continuous functions f and ε, there exists a solution y to

$$\begin{aligned} 3y'^4y''y''''^2 \quad & -4y'^4y'''^2y'''' + 6y'^3y''^2y'''y'''' + 24y'^2y''^4y'''' \\ & -12y'^3y''y'''^3 - 29y'^2y''^3y'''^2 + 12y''^7 \qquad = 0 \end{aligned}$$

such that

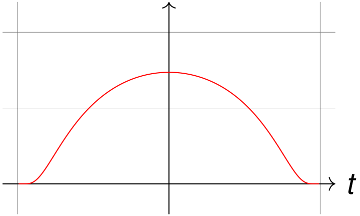
$$|y(t) - f(t)| \leqslant \varepsilon(t).$$

Problem : Rubel is «cheating».

Rubel's proof in one slide

- Take $f(t) = e^{\frac{-1}{1-t^2}}$ for $-1 < t < 1$ and $f(t) = 0$ otherwise

$$(1 - t^2)^2 f''(t) + 2tf'(t) = 0.$$

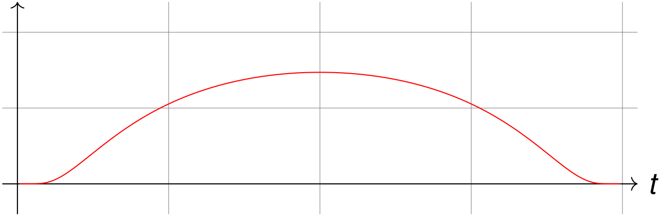


Rubel's proof in one slide

- Take $f(t) = e^{\frac{-1}{1-t^2}}$ for $-1 < t < 1$ and $f(t) = 0$ otherwise

$$(1 - t^2)^2 f''(t) + 2tf'(t) = 0.$$

- Can do the same with $cf(at + b)$ (translation+scaling)

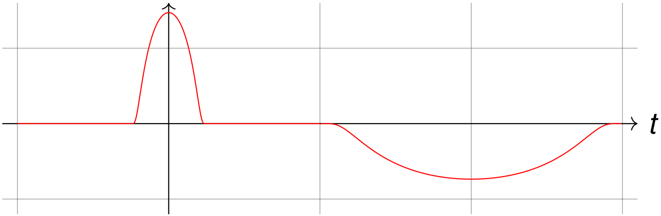


Rubel's proof in one slide

- Take $f(t) = e^{\frac{-1}{1-t^2}}$ for $-1 < t < 1$ and $f(t) = 0$ otherwise

$$(1-t^2)^2 f''(t) + 2tf'(t) = 0.$$

- Can do the same with $cf(at+b)$ (translation+scaling)
- Can glue together arbitrary many such pieces

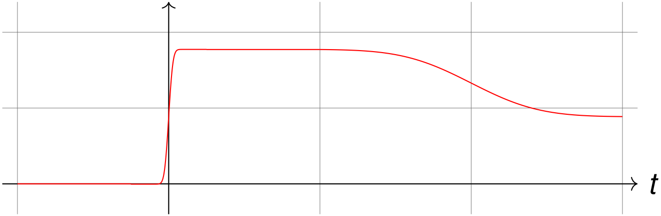


Rubel's proof in one slide

- Take $f(t) = e^{\frac{-1}{1-t^2}}$ for $-1 < t < 1$ and $f(t) = 0$ otherwise

$$(1-t^2)^2 f''(t) + 2tf'(t) = 0.$$

- Can do the same with $cf(at+b)$ (translation+scaling)
- Can glue together arbitrary many such pieces
- Can arrange so that $\int f$ is solution : piecewise pseudo-linear

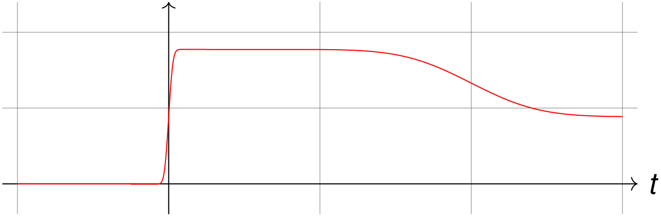


Rubel's proof in one slide

- Take $f(t) = e^{\frac{-1}{1-t^2}}$ for $-1 < t < 1$ and $f(t) = 0$ otherwise

$$(1-t^2)^2 f''(t) + 2tf'(t) = 0.$$

- Can do the same with $cf(at+b)$ (translation+scaling)
- Can glue together arbitrary many such pieces
- Can arrange so that $\int f$ is solution : piecewise pseudo-linear



Conclusion : Rubel's equation allows any piecewise pseudo-linear functions, and those are **dense in** C^0

Why I don't like Rubel's result

- the solution y is not unique, **even with added initial conditions** :

$$p(y, y', \dots, y^{(k)}) = 0, \quad y(0) = \alpha_0, y'(0) = \alpha_1, \dots, y^{(k)}(0) = \alpha_k$$

Why I don't like Rubel's result

- the solution y is not unique, **even with added initial conditions** :

$$p(y, y', \dots, y^{(k)}) = 0, \quad y(0) = \alpha_0, y'(0) = \alpha_1, \dots, y^{(k)}(0) = \alpha_k$$

- ...even with a countable number of extra conditions :

$$p(y, y', \dots, y^{(k)}) = 0, y^{(d_i)}(a_i) = b_i, i \in \mathbb{N}$$

In fact, this is fundamental for the proof to work !

Why I don't like Rubel's result

- the solution y is not unique, **even with added initial conditions** :

$$p(y, y', \dots, y^{(k)}) = 0, \quad y(0) = \alpha_0, y'(0) = \alpha_1, \dots, y^{(k)}(0) = \alpha_k$$

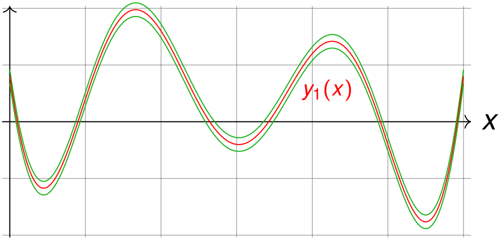
- ...even with a countable number of extra conditions :

$$p(y, y', \dots, y^{(k)}) = 0, y^{(d_i)}(a_i) = b_i, i \in \mathbb{N}$$

In fact, this is fundamental for the proof to work !

- Rubel's interpretation : this equation is universal
- My interpretation : this equation allows almost anything

Universal differential equation (PIVP)



Theorem

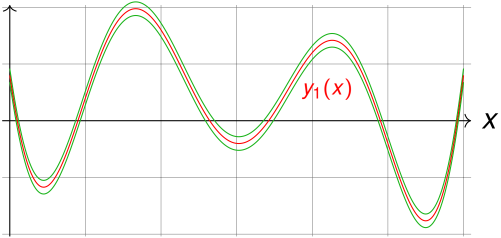
There exists a **fixed** polynomial p and $d \in \mathbb{N}$ such that for any continuous functions f and ε, there exists $\alpha \in \mathbb{R}^d$ such that

$$y(0) = \alpha, \qquad y'(t) = p(y(t))$$

has a **unique solution** and this solution satisfies such that

$$|y(t) - f(t)| \leqslant \varepsilon(t).$$

Universal differential equation (DAE)



Theorem

There exists a **fixed** polynomial p and $k \in \mathbb{N}$ such that for any continuous functions f and ε, there exists $\alpha_0, \ldots, \alpha_k \in \mathbb{R}$ such that

$$p(y, y', \ldots, y^{(k)}) = 0, \quad y(0) = \alpha_0, y'(0) = \alpha_1, \ldots, y^{(k)}(0) = \alpha_k$$

has a **unique analytic solution** and this solution satisfies such that

$$|y(t) - f(t)| \leqslant \varepsilon(t).$$

Vague proof idea

Key ingredients :

- fast-growing function
- (analog) bit generator

$\rightarrow$ On the white board.

A new notion of computability

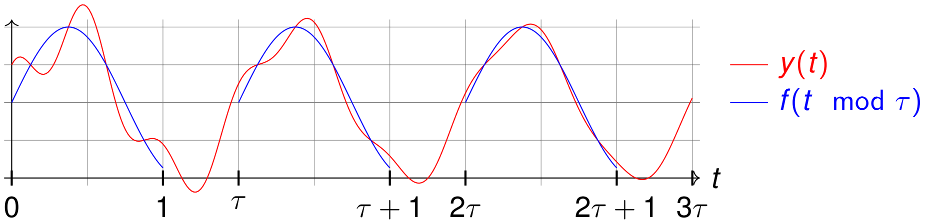
Almost-Theorem

$f : [0,1] \to \mathbb{R}$ is **computable** if and only if there exists $\tau > 1$, $y_0 \in \mathbb{R}^d$ and p polynomial such that

$$y'(0) = y_0, \qquad y'(t) = p(y(t))$$

satisfies

$$|f(x) - y(x + n\tau)| \leqslant 2^{-n}, \qquad \forall x \in [0,1], \forall n \in \mathbb{N}$$



Conclusion

- Rubel's universal differential is **very weak**
- We provide a stronger result
- Another notion of analog computability