Mesures de complexité pour les fonctions : circuits, Kolmogorov

Adrien Nolhier

École Polytechnique, Master Foundation of Computer Science

26 août 2025

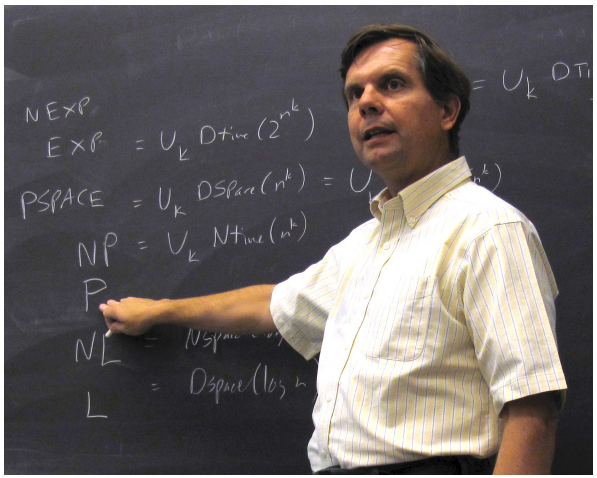
Stage de recherche encadré par Olivier Bournez au LIX.

Introduction

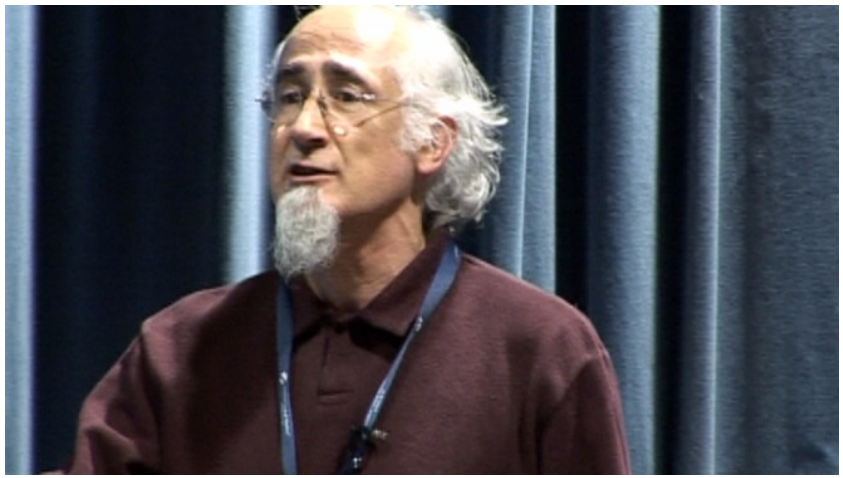
- Motivations : découvrir et mieux connaître les classes de complexité.

Introduction

- Motivations : découvrir et mieux connaître les classes de complexité.



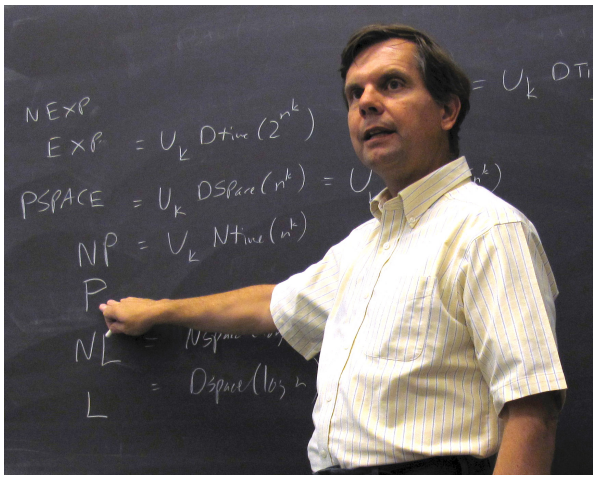
Eric Allender - *Power from random strings*, 2005



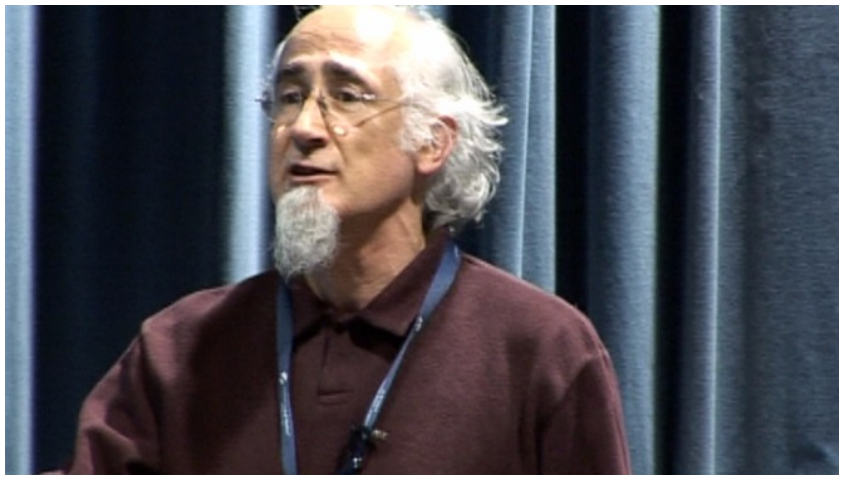
Jose L. Balcázar - *Computational power of neural networks*, 1998

Introduction

- Motivations : découvrir et mieux connaître les classes de complexité.



Eric Allender - *Power from random strings*, 2005



Jose L. Balcázar - *Computational power of neural networks*, 1998

- en comptant en quantité d'information plutôt qu'en ressource temporelle ou spatiale
- parallèle entre les deux mondes : Kolmogorov et les circuits, reliés par Allender avec $C = \text{Size}^H$.

Table des matières

1. MT avec oracle, MT = circuits, complexité en circuit
2. UTM efficace, circuit avec oracle, généralisation de MT = circuits
3. Complexités de Kolmogorov
   - Définition classique
   - Du type Allender
   - Du type Balcázar
4. Réseau de neurones
   - Définition
   - Non-uniformité et NN
   - Résultats

Définition (Machine de Turing avec oracle $A \subseteq \mathbb{N}$)

Machine de Turing à laquelle on ajoute la possibilité, ***au cours de son exécution, de "questionner l'oracle"****, en posant des questions du type*

"x appartient-il à A ?".

- gain en **temps de calcul**
- gain en **puissance calculatoire**

Deux grands types : oracle fini et oracle infini.

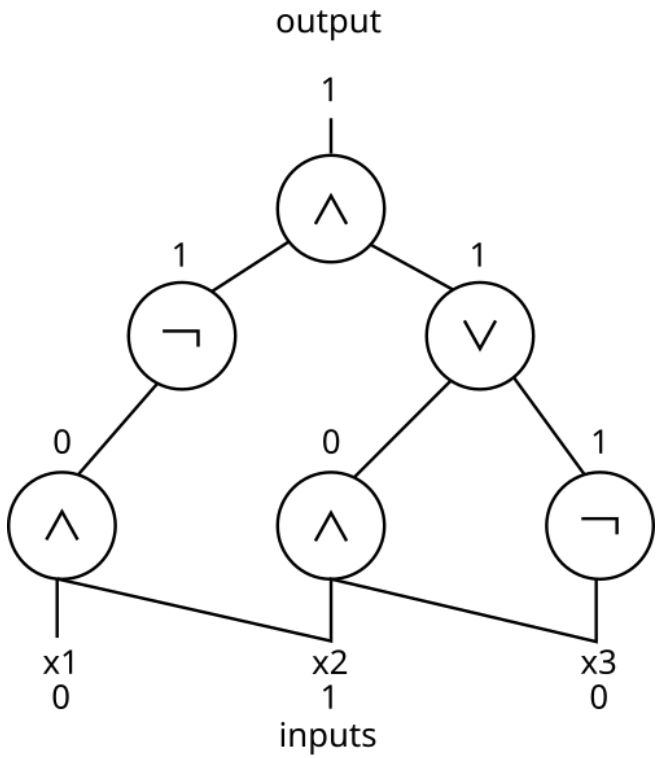
Définition (Circuit booléen D)

n bits d'entrée $x1, \ldots, xn$, des portes logiques ET, OU, NON et un bit de sortie.

- Un circuit **calcule une fonction**

$$f : 2^n \to 2.$$

- On appelle sa **taille** son nombre de portes binaires, noté $|D|$.
- Et sa **profondeur**, la taille maximal d'un chemin d'une entrée à la sortie.



Les circuits calculent pour une entrée de taille constante.

Question : comment calculer des fonctions $\mathbb{N} \to \mathbb{N}$ avec ?

Les circuits calculent pour une entrée de taille constante.

Question : comment calculer des fonctions $\mathbb{N} \to \mathbb{N}$ avec ?

Définition (Famille de circuits : "un circuit par taille d'entrée")

Suite $(D_i)_{i \in \mathbb{N}}$ telle que le circuit D_i calcule pour des entrées de taille i.

Comparons ce modèle avec la MT.

Comparaison MT et circuit(s)

Machine de Turing	**Circuit booléen**
temps	taille
espace	profondeur

Les ressources sont liées, au sens suivant :

Comparaison MT et circuit(s)

Machine de Turing	**Circuit booléen**
temps	taille
espace	profondeur

Les ressources sont liées, au sens suivant :

- On peut simuler un circuit à l'aide d'une machine avec perte de temps polynomiale.
- On peut simuler une machine avec une famille de circuits et perte de temps polynomiale.

Comparaison MT et circuit(s)

Machine de Turing	**Circuit booléen**
temps	taille
espace	profondeur

Les ressources sont liées, au sens suivant :

- On peut simuler un circuit à l'aide d'une machine avec perte de temps polynomiale.
- On peut simuler une machine avec une famille de circuits et perte de temps polynomiale.

⚠ Les machines sont dénombrables, contrairement aux familles de circuits de taille finie.

Définition (Complexité en circuit)

$$\text{Size}(f) = \min\{\text{size}(D) : D \textit{ calcule } f\}.$$

Avec un petit travail technique, on peut relier :

$$\text{chaînes binaires } x_f \leftrightarrow \text{fonctions } f : 2^n \to 2$$

On définit $\text{Size}(x) =$ "complexité en circuit de la fonction f_x".

1. MT avec oracle, MT = circuits, complexité en circuit
2. UTM efficace, circuit avec oracle, généralisation de MT = circuits
3. Complexités de Kolmogorov
   - Définition classique
   - Du type Allender
   - Du type Balcázar
4. Réseau de neurones
   - Définition
   - Non-uniformité et NN
   - Résultats

Définitions préliminaires

Définition (UTM d'Allender)

Il existe une machine universelle U simulant toute machine avec perte de temps au plus logarithmique.

Définitions préliminaires

Définition (UTM d'Allender)

Il existe une machine universelle U simulant toute machine avec perte de temps au plus logarithmique.

Définition (Circuit avec porte d'oracle pour $A \subseteq \mathbb{N}$, Allender)

Circuit classique auquel on ajoute une ***porte spéciale définie pour toute arité*** $r \in \mathbb{N}$*, telle que la porte décide si l'entier représenté par* $b_1 \ldots, b_r$ *appartient à l'oracle* A*.*

Définitions préliminaires

Définition (UTM d'Allender)

Il existe une machine universelle U simulant toute machine avec perte de temps au plus logarithmique.

Définition (Circuit avec porte d'oracle pour $A \subseteq \mathbb{N}$, Allender)

*Circuit classique auquel on ajoute une **porte spéciale définie pour toute arité** $r \in \mathbb{N}$, telle que la porte décide si l'entier représenté par $b_1 \ldots, b_r$ appartient à l'oracle A.*

Nous ne sommes plus restreints aux portes binaires.

Définition (Taille d'un circuit avec porte d'oracle, Allender)

*Désormais, nous définissons la taille d'un circuit comme la **somme des arités des portes utilisées**.*

généralisation de "MT = circuits"

But : montrer "$C = \text{Size}^H$".

$\rightarrow$ On étudie "MT vs circuits" relativisés.

généralisation de "MT = circuits"

But : montrer "$C = \text{Size}^H$".
$\rightarrow$ On étudie "MT vs circuits" relativisés.
Preuves : généralisation de celles du livre de Sylvain Perifel.

Proposition

- *Soit C^A un circuit avec porte pour l'oracle A. Alors, U^A simule C^A avec perte de temps poly.*

généralisation de "MT = circuits"

But : montrer "$C = \text{Size}^H$".
$\rightarrow$ On étudie "MT vs circuits" relativisés.
Preuves : généralisation de celles du livre de Sylvain Perifel.

Proposition

- *Soit C^A un circuit avec porte pour l'oracle A. Alors, U^A simule C^A avec perte de temps poly.*
- *Soit M^A une machine de Turing avec l'oracle A. Alors, il existe une **famille de circuits simulant** M^A avec perte de taille poly.*

généralisation de "MT = circuits"

But : montrer "$C = \text{Size}^H$".
$\rightarrow$ On étudie "MT vs circuits" relativisés.
Preuves : généralisation de celles du livre de Sylvain Perifel.

Proposition

- *Soit C^A un circuit avec porte pour l'oracle A. Alors, U^A simule C^A avec perte de temps poly.*
- *Soit M^A une machine de Turing avec l'oracle A. Alors, il existe une* ***famille de circuits simulant*** *M^A avec perte de taille poly.*

Remarque : "simuler" = "les modèles définissent des fonctions égales en tout points".

1. MT avec oracle, MT = circuits, complexité en circuit
2. UTM efficace, circuit avec oracle, généralisation de MT = circuits
3. Complexités de Kolmogorov
   - Définition classique
   - Du type Allender
   - Du type Balcázar
4. Réseau de neurones
   - Définition
   - Non-uniformité et NN
   - Résultats

1. MT avec oracle, MT = circuits, complexité en circuit
2. UTM efficace, circuit avec oracle, généralisation de MT = circuits
3. Complexités de Kolmogorov
   - Définition classique
   - Du type Allender
   - Du type Balcázar
4. Réseau de neurones
   - Définition
   - Non-uniformité et NN
   - Résultats

Définition (Complexité de K. sur une machine quelconque)

Pour une machine de Turing M et une chaîne binaire finie x, on définit :

$$C_M(x) = \min\{|d| : M(d) = x\},$$

la taille de la plus petite chaîne d permettant de restituer x à l'aide de M.

"compresser au maximum x en tant que d, tel que M effectue la décompression"

La meilleure machine

On s'abstrait de la machine en raisonnant avec $=^+$.

Théorème

Il existe une machine U dite universelle telle que pour toute machine M,

$$C_U(x) \leq^* C_M(x)$$

Remarque : cette nouvelle définition de "machine universelle" et celle déjà étudiée coïncident.

Première complexité de Kolmogorov

Définition (Complexité de K.)

Soit U une machine universelle fixée. On définit

$$C(x) = C_U(x) = \min\{|d| : U(d) = x\}.$$

Première complexité de Kolmogorov

Définition (Complexité de K.)

Soit U une machine universelle fixée. On définit

$$C(x) = C_U(x) = \min\{|d| : U(d) = x\}.$$

Définition (Rappel de la définition de Size)

$$\text{Size}(x) = \min\{\text{size}(D) : D \text{ calcule } f_x\}.$$

Similitude dans la forme mais différence cruciale : Size est calculable, C ne l'est pas.

1. MT avec oracle, MT = circuits, complexité en circuit
2. UTM efficace, circuit avec oracle, généralisation de MT = circuits
3. Complexités de Kolmogorov
   - Définition classique
   - Du type Allender
   - Du type Balcázar
4. Réseau de neurones
   - Définition
   - Non-uniformité et NN
   - Résultats

- ajout d'une contrainte de temps
- transforme en problème de décision

Définition (KT, Allender)

$$\mathrm{KT}(x) = \min_{(d,T)\in 2^{<\mathbb{N}}\times\mathbb{N}} \{|d| + \mathrm{T} : U(d,\ i) \text{ calcule } x_i \text{ en temps } \mathrm{T}\}$$

- ajout d'une contrainte de temps
- transforme en problème de décision

Définition (KT, Allender)

$$\mathrm{KT}(x) = \min_{(d,T)\in 2^{<\mathbb{N}}\times\mathbb{N}} \{|d| + \mathrm{T} : U(d,\, i) \text{ calcule } x_i \text{ en temps } \mathrm{T}\}$$

- compromis entre $|d|$ et T.
- KT est **calculable**.

- ajout d'une contrainte de temps
- transforme en problème de décision

Définition (KT, Allender)

$$\mathrm{KT}(x) = \min_{(d,T)\in 2^{<\mathbb{N}}\times\mathbb{N}} \{|d| + \mathrm{T} : U(d,\, i) \text{ calcule } x_i \text{ en temps } \mathrm{T}\}$$

- compromis entre $|d|$ et T.
- KT est **calculable**.

Théorème (Allender)

$\mathrm{KT}^A(x) =_p \mathrm{Size}^A(x)$, *pour tout oracle A et chaîne x.*

Montrer que $C = \text{Size}^H$

Nous pouvons aussi montrer :

Théorème (Allender)

Pour toute chaîne finie x, $C(x) =_p \text{KT}^H(x)$.

En prenant $A = H$, on déduit : $C =_p \text{KT}^H =_p \text{Size}^H$.

1. MT avec oracle, MT = circuits, complexité en circuit
2. UTM efficace, circuit avec oracle, généralisation de MT = circuits
3. Complexités de Kolmogorov
   - Définition classique
   - Du type Allender
   - Du type Balcázar
4. Réseau de neurones
   - Définition
   - Non-uniformité et NN
   - Résultats

Définition (Balcázar)

*Soit α une chaîne **infinie**. On note $\alpha \in \mathsf{K}[f, g]$ s'il existe une chaîne infinie β telle que tout préfixe de taille au moins $f(n)$ est le code d'une machine qui sur l'entrée n affiche les n premiers bits de α en temps $g(n)$.*

La fonction f mesure la quantité d'information nécessaire au calcul, et g le temps d'exécution autorisé.

Avec l'idée du "**bit-à-bit**" et des **circuit avec oracles** d'Allender :

Définition (K', Balcázar "corrigé")

Soient $\alpha \in 2^{\mathbb{N}}$ *et* $f, g : \mathbb{N} \to \mathbb{N}$.

$$\begin{aligned} \alpha \in K'[f,g] \Longleftrightarrow & \exists \beta,\ \forall n,\ \exists C,\ |C| \leq g(n) \\ & \wedge \forall (i,b) \in [n+1] \times \{0,1,*\}, \\ & C^{\beta_{|f(n)}}(i,b) = 1 \text{ ssi } \alpha_i = b \end{aligned}$$

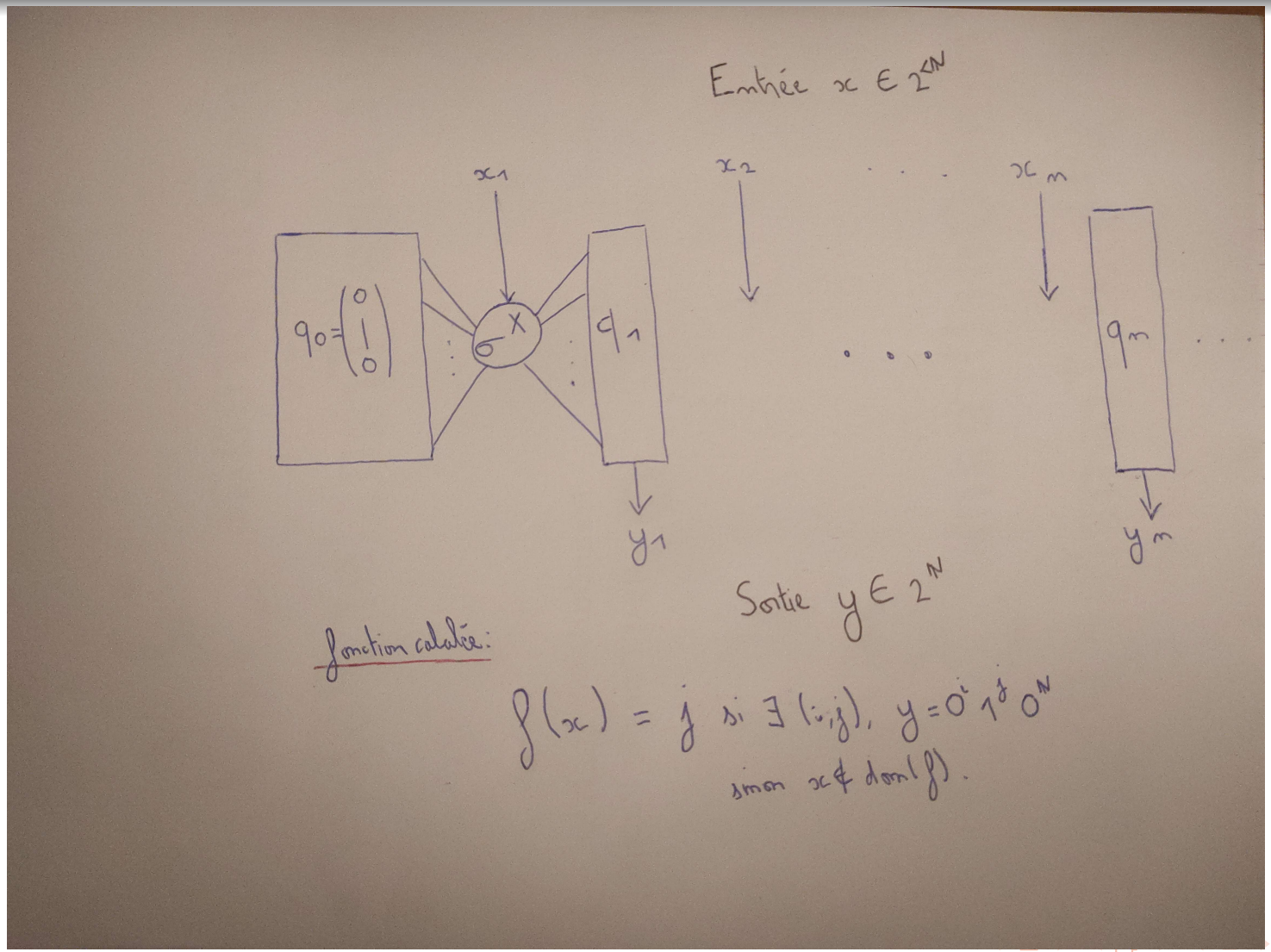
Notation par classe :

$\alpha \in K'[\mathcal{F}, \mathcal{G}]$ s'il existe $f, g \in \mathcal{F} \times \mathcal{G}$ tels que $\alpha \in K'[f,g]$.

1. MT avec oracle, MT = circuits, complexité en circuit
2. UTM efficace, circuit avec oracle, généralisation de MT = circuits
3. Complexités de Kolmogorov
   - Définition classique
   - Du type Allender
   - Du type Balcázar
4. Réseau de neurones
   - Définition
   - Non-uniformité et NN
   - Résultats

1. MT avec oracle, MT = circuits, complexité en circuit
2. UTM efficace, circuit avec oracle, généralisation de MT = circuits
3. Complexités de Kolmogorov
   - Définition classique
   - Du type Allender
   - Du type Balcázar
4. Réseau de neurones
   - Définition
   - Non-uniformité et NN
   - Résultats

Fonctionnement du réseau de neurones



Fonctionnement du réseau de neurones

$\rightarrow$ la puissance du NN dépend de l'ensemble $X \subseteq \mathbb{R}$ de poids autorisés pour effectuer le calcul dans la fonction de saturation σ.

Fonctionnement du réseau de neurones

$\rightarrow$ la puissance du NN dépend de l'ensemble $X \subseteq \mathbb{R}$ de poids autorisés pour effectuer le calcul dans la fonction de saturation σ.

Théorème (Équivalence MT et NN à poids rationnels, Balcázar)

Soit $f :\subseteq 2^{<\mathbb{N}} \rightarrow 2^{<\mathbb{N}}$ une fonction partielle sur des chaînes binaires. Alors,

$$\exists e \in \mathbb{N},\ \Phi_e(x) = f(x) \Longleftrightarrow \exists \mathcal{N},\ \Phi_{\mathcal{N}}(x) = f(x),\ \forall x \in \operatorname{dom} f.$$

De plus, si f est calculable en temps $T(x)$ avec une machine de Turing, alors elle est calculable en temps $O(T(x) + n)$ avec un réseau.

$\rightarrow$ restreindre selon $\mathrm{K}[f, g]$ pour f et g bien choisis.

1. MT avec oracle, MT = circuits, complexité en circuit
2. UTM efficace, circuit avec oracle, généralisation de MT = circuits
3. Complexités de Kolmogorov
   - Définition classique
   - Du type Allender
   - Du type Balcázar
4. Réseau de neurones
   - Définition
   - Non-uniformité et NN
   - Résultats

Montré pour K par Balcázar, nous avons conduit la preuve pour K'.

Théorème ($\mathrm{P}/\mathcal{F} = \mathcal{N}_{\mathrm{K}'[\mathcal{F},\mathrm{poly}]}[\mathrm{FP}]$, adapté de Balcázar, 6.2)

Soit $\mathcal{F}$ une classe de bornes de conseil raisonnables. Alors, la classe $\mathrm{P}/\mathcal{F}$ est exactement la classe des langages reconnus en temps polynomial par un réseau de neurones avec des poids dans $\mathrm{K}'[\mathcal{F}, \mathrm{poly}]$.

En particulier, $P/\mathrm{poly} = \mathcal{N}_{\mathbb{R}}$.

1. MT avec oracle, MT = circuits, complexité en circuit
2. UTM efficace, circuit avec oracle, généralisation de MT = circuits
3. Complexités de Kolmogorov
   - Définition classique
   - Du type Allender
   - Du type Balcázar
4. Réseau de neurones
   - Définition
   - Non-uniformité et NN
   - Résultats

Théorème de hiérarchie

Théorème (Théorème de Hiérarchie, adapté de Balcázar)

Soient $\mathcal{F}$ et $\mathcal{G}$ deux classes closes par $\mathrm{O}(.)$ et telles que $\mathcal{F} \prec \mathcal{G}$. Alors,

$$\mathcal{N}_{\mathrm{K}'[\mathcal{F},\mathrm{poly}]} \subsetneq \mathcal{N}_{\mathrm{K}'[\mathcal{G},\mathrm{poly}]}.$$

Corollaire

Par le théorème "MT $=$ NN($\mathbb{Q}$)" :

Théorème

Pour tout $k > 0$, $\mathrm{P}/\log^k = \mathcal{N}_{\mathrm{K}'[\log^k,\mathrm{poly}]}$ et, l'inclusion suivante est stricte :

$$\mathrm{P}/\log^k \subsetneq \mathrm{P}/\log^{k+1}.$$

Conclusion

- Caractérisation des classes non-uniformes de la forme $P/\log^k$ à l'aide de circuits (possible aussi pour P/poly mais pas forcément pour toutes les classes non-uniformes).

- Définitions d'Allender prometteuses pour établir de nouveaux résultats.

Références principales

- Eric Allender, Harry Buhrman, Michal Koucký, Dieter Melkebeek, and Detlef Ronneburger. Power from random strings. SIAM J. Comput., 35 :1467–1493, 01 2006.
- José Balcázar, Ricard Gavaldà, and Hava Siegelmann. Computational power of neural networks : A characterization in terms of kolmogorov complexity. Information Theory, IEEE Transactions on, 43 :1175 – 1183, 08 1997.
- Sylvain Perifel. Complexité algorithmique. Références sciences. Ellipses, 2014.
- Benoît Monin et Ludovic Patey. Calculabilité. Calvage et Mounet, 2022.