Codage et compression

Comment communiquer en dépensant moins ?

Jean-Marc.Vincent@imag.fr

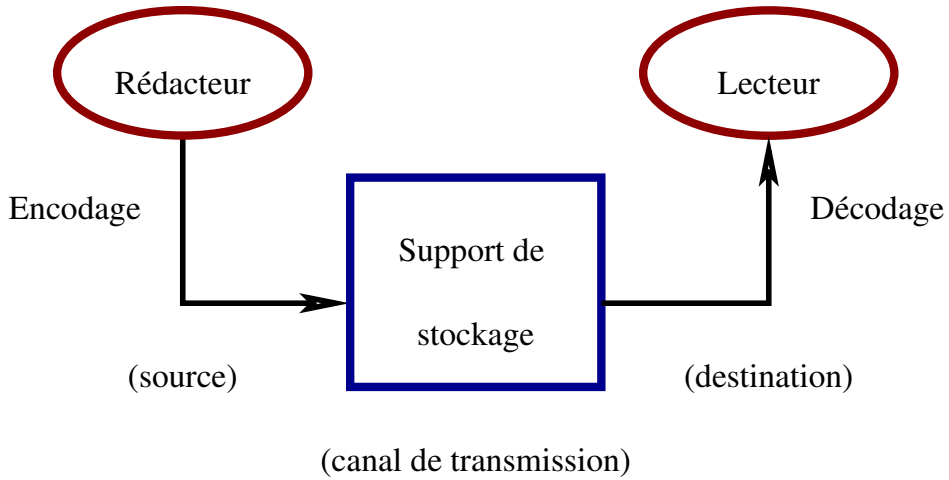
DU Informatique et Sciences du Numérique : Information

Novembre 2015

CODAGE ET COMPRESSION

UNIVERSITÉ Grenoble Alpes

TRANSMISSION DE L'INFORMATION



Critères de qualité d'un code :

- **Intégrité de l'information :** tolérance aux fautes (détection/correction des erreurs)
- **Sécurité de l'information :** authentification (cryptage)
- **Efficacité de la transmission :** compression des données

Donnée (message) : séquence finie de bits, éventuellement structurée

TRANSMISSION DE L'INFORMATION

À l'aide de jetons transmettre le message suivant

ABABACADABEBAAD

TRANSMISSION DE L'INFORMATION

À l'aide de jetons transmettre le message suivant

ABABACADABEBAAD

Codage

Lettre	A	B	C	D	E		
Nombre	7	4	1	2	1	15	
Code1	000	001	010	011	100	code de longueur fixe	3.0

TRANSMISSION DE L'INFORMATION

À l'aide de jetons transmettre le message suivant

ABABACADABEBAAD

Codage

Lettre	A	B	C	D	E		
Nombre	7	4	1	2	1	15	
Code1	000	001	010	011	100	code de longueur fixe	3.0
Code2	0	1	10	11	100	code ambigü	1.3

TRANSMISSION DE L'INFORMATION

À l'aide de jetons transmettre le message suivant

ABABACADABEBAAD

Codage

Lettre	A	B	C	D	E		
Nombre	7	4	1	2	1	15	
Code1	000	001	010	011	100	code de longueur fixe	3.0
Code2	0	1	10	11	100	code ambigü	1.3
Code3	0	10	1110	110	1111	code préfixe	1.9

TRANSMISSION DE L'INFORMATION

À l'aide de jetons transmettre le message suivant

ABABACADABEBAAD

Codage

Lettre	A	B	C	D	E		
Nombre	7	4	1	2	1	15	
Code1	000	001	010	011	100	code de longueur fixe	3.0
Code2	0	1	10	11	100	code ambigü	1.3
Code3	0	10	1110	110	1111	code préfixe	1.9
Code4	01	11	110	00	010		2.1

UNIVERSITÉ Grenoble Alpes

TRANSMISSION DE L'INFORMATION

À l'aide de jetons transmettre le message suivant

ABABACADABEBAAD

Codage

Lettre	A	B	C	D	E		
Nombre	7	4	1	2	1	15	
Code1	000	001	010	011	100	code de longueur fixe	3.0
Code2	0	1	10	11	100	code ambigü	1.3
Code3	0	10	1110	110	1111	code préfixe	1.9
Code4	01	11	110	00	010		2.1

Trouver un code uniquement déchiffrable de longueur minimale.

CODAGE

- Symboles $\mathcal{S} = \{s_1, \cdots, s_k\}$
- Codage

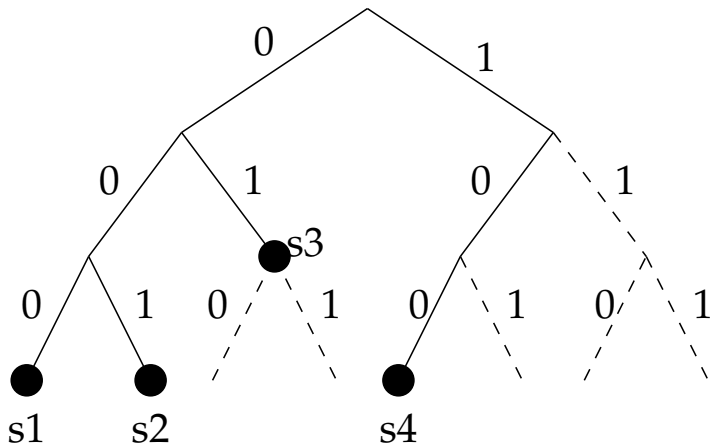
$$\begin{array}{rcl} C : \mathcal{S} & \longrightarrow & \{0,1\}^* \\ s_i & \longmapsto & c(s_i) \text{ longueur } l_i \end{array}$$

- Code vérifiant la propriété du préfixe (uniquement déchiffrable) ;
- **Inégalité de Kraft** Pour un codage ayant la propriété du préfixe

$$\sum_{i=1}^{k} 2^{-l_i} \leqslant 1, \tag{1}$$

Réciproquement, si (1) il existe un codage avec la propriété du préfixe.

INÉGALITÉ DE KRAFT (1)



On considère les 4 mots codes associés aux symboles $\{S_1, \cdots, S_4\}$

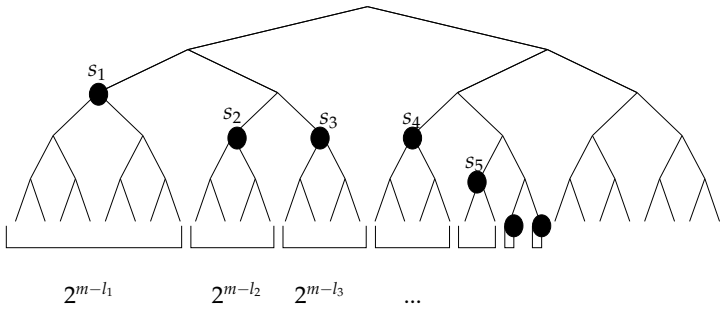
Symbole	Code	longueur
S_1	000	3
S_2	001	3
S_3	01	2
S_4	100	3

Dans ce cas,

$$\sum_{i=1}^{4} 2^{-l_i} = \frac{1}{2^3} + \frac{1}{2^3} + \frac{1}{2^2} + \frac{1}{2^3} = \frac{5}{8} \leqslant 1.$$

L'inégalité de Kraft est bien vérifiée.

INÉGALITÉ DE KRAFT (2)



On considère les 7 longueurs de mots codes, $(2, 3, 3, 3, 4, 5, 5)$. On vérifie aisément l'inégalité de Kraft.

Longueur	Code	symbole
2	00	s_1
3	010	s_2
3	011	s_3
3	100	s_4
4	1010	s_5
5	10110	s_6
5	10111	s_7

CODAGE ET COMPRESSION

UNIVERSITÉ Grenoble Alpes

COMPLEXITÉ D'UN CODE

Sources aléatoires : $p = (p_1, \cdots, p_k)$ fréquence de transmission ;
longueur moyenne du codage

$$L(c) = \sum_{i=1}^{k} p_i l_i;$$

$$L_{inf} = \inf_{c} L(c);\ c \text{ ayant la propriété du préfixe}$$

COMPLEXITÉ D'UN CODE

Sources aléatoires : $p = (p_1, \cdots, p_k)$ fréquence de transmission ; longueur moyenne du codage

$$L(c) = \sum_{i=1}^{k} p_i l_i;$$

$$L_{inf} = \inf_{c} L(c);\ c \text{ ayant la propriété du préfixe}$$

Théorème (Shannon 1948)

$$\mathcal{H}(p) \leqslant L_{inf} \leqslant \mathcal{H}(p) + 1;$$

avec

$$\mathcal{H}(p) = \sum_{i=1}^{k} p_i \log_2(\frac{1}{p_i});$$

quantité d'information contenue dans p (appelée également entropie).

THÉORÈME DE SHANNON $\mathcal{H}(p) \leqslant L_{inf} \leqslant \mathcal{H}(p) + 1$

Borne inférieure

$$\text{Minimiser } f(x_1, \cdots, x_k) = \sum_{i=1}^{k} p_i x_i, \text{ sous la contrainte } \sum_{i=1}^{k} 2^{-x_i} \leqslant 1.$$

Le point optimal (Lagrange) : $\boxed{l_i^* = -\log_2 p_i}$.

$$\text{En } l^* \quad \sum_{i=1}^{k} 2^{-l_i^*} = \sum_{i=1}^{k} 2^{-(-\log_2 p_i)} = \sum_{i=1}^{k} p_i = 1,$$

et

$$\sum_{i=1}^{k} p_i l_i^* = \sum_{i=1}^{k} p_i(-\log_2 p_i) = \mathcal{H}(p).$$

Donc pour tout code ayant la propriété du préfixe de longueurs $l_1, \cdots, l_k$

$$\mathcal{H}(p) = f(l_1^*, \cdots, l_k^*) \leqslant f(l_1, \cdots, l_k) = L(h).$$

UNIVERSITÉ Grenoble Alpes

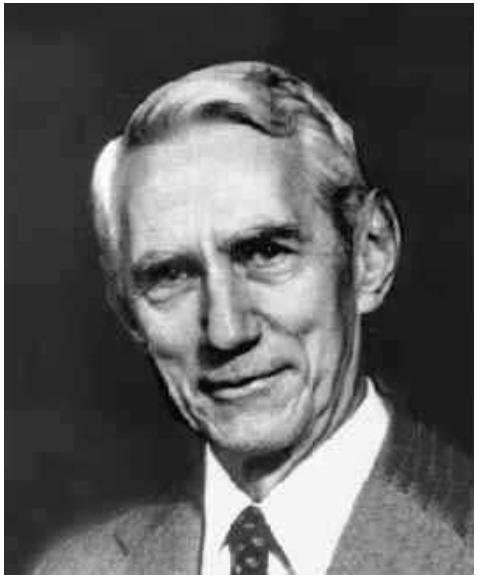
THÉORÈME DE SHANNON $\mathcal{H}(p) \leqslant L_{inf} \leqslant \mathcal{H}(p) + 1$

Borne supérieure

Soit $l^{sup} = (\lceil l_1^* \rceil, \cdots \lceil l_k^* \rceil$ (approximation entière de l'optimum)
l^{sup} vérifie l'inégalité de Kraft, il existe donc un codage ayant la propriété du codage de longueur l^{sup}.
La longueur moyenne de ce codage est

$$\sum_{i=1}^{k} p_i l_i^{sup} \leqslant \sum_{i=1}^{k} p_i (l_i^* + 1) = \mathcal{H}(p) + 1.$$

CLAUDE SHANNON (1916-2001)



Claude Elwood Shannon (30 avril 1916 à Gaylord, Michigan - 24 février 2001), ingénieur électrique, est l'un des pères, si ce n'est le père fondateur, de la théorie de l'information. Son nom est attaché à un célèbre "schéma de Shannon" très utilisé en sciences humaines, qu'il a constamment désavoué.

Il étudia le génie électrique et les mathématiques à l'Université du Michigan en 1932. Il utilisa notamment l'algèbre booléenne pour sa maîtrise soutenue en 1938 au MIT. Il y expliqua comment construire des machines à relais en utilisant l'algèbre de Boole pour décrire l'état des relais (1 : fermé, 0 : ouvert).
Shannon travailla 20 ans au MIT, de 1958 à 1978. Parallèlement à ses activités académiques, il travailla aussi aux laboratoires Bell de 1941 à 1972.
Claude Shannon était connu non seulement pour ses travaux dans les télécommunications, mais aussi pour l'étendue et l'originalité de ses hobbies, comme la jonglerie, la pratique du monocycle et l'invention de machines farfelues : une souris mécanique sachant trouver son chemin dans un labyrinthe, un robot jongleur, un joueur d'échecs (roi tour contre roi)...
Souffrant de la maladie d'Alzheimer dans les dernières années de sa vie, Claude Shannon est mort à 84 ans le 24 février 2001.

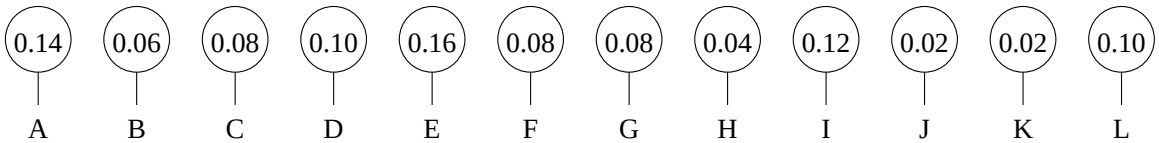
Wikipedia

UNIVERSITÉ Grenoble Alpes

Codage et compression

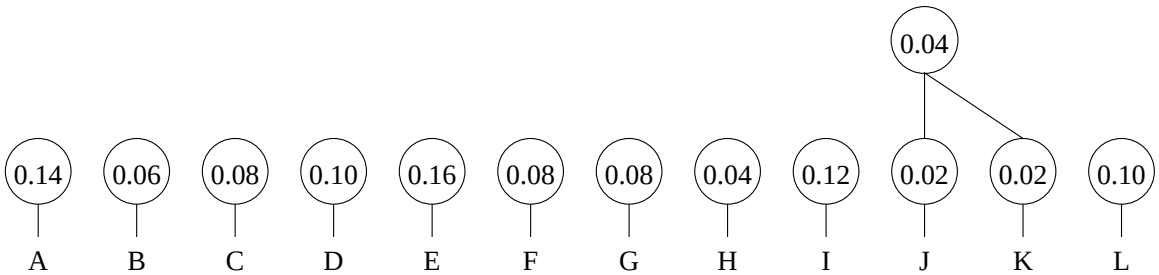
ALGORITHME DE HUFFMAN (1951)

A	0.14
B	0.06
C	0.08
D	0.10
E	0.16
F	0.08
G	0.08
H	0.04
I	0.12
J	0.02
K	0.02
L	0.10



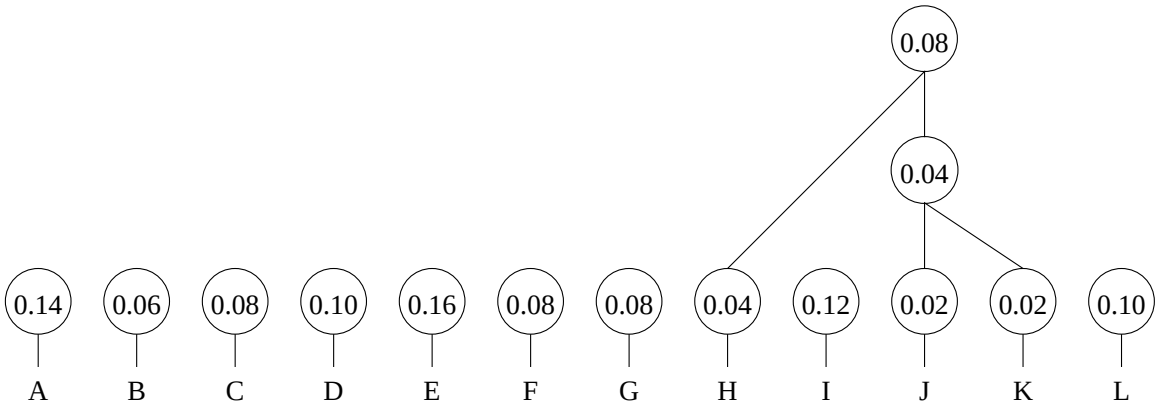
ALGORITHME DE HUFFMAN (1951)

A	0.14
B	0.06
C	0.08
D	0.10
E	0.16
F	0.08
G	0.08
H	0.04
I	0.12
J	0.02
K	0.02
L	0.10



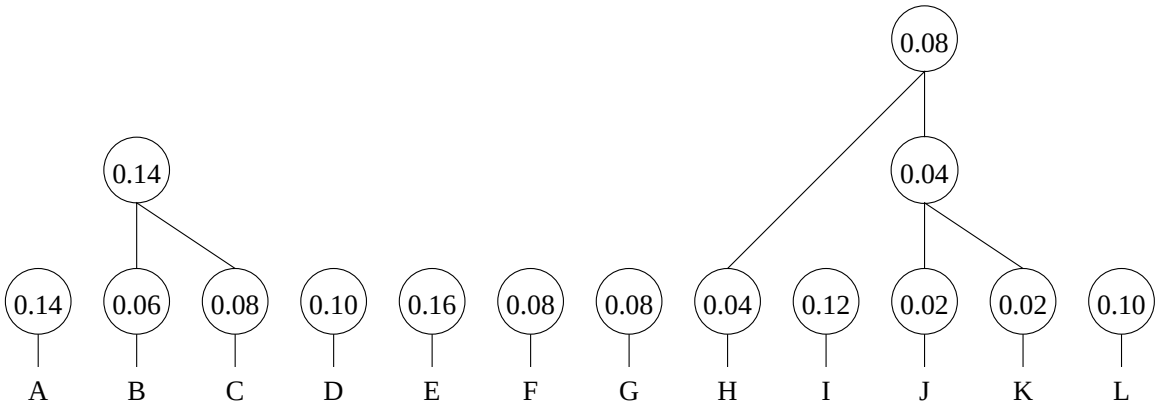
ALGORITHME DE HUFFMAN (1951)

A	0.14
B	0.06
C	0.08
D	0.10
E	0.16
F	0.08
G	0.08
H	0.04
I	0.12
J	0.02
K	0.02
L	0.10



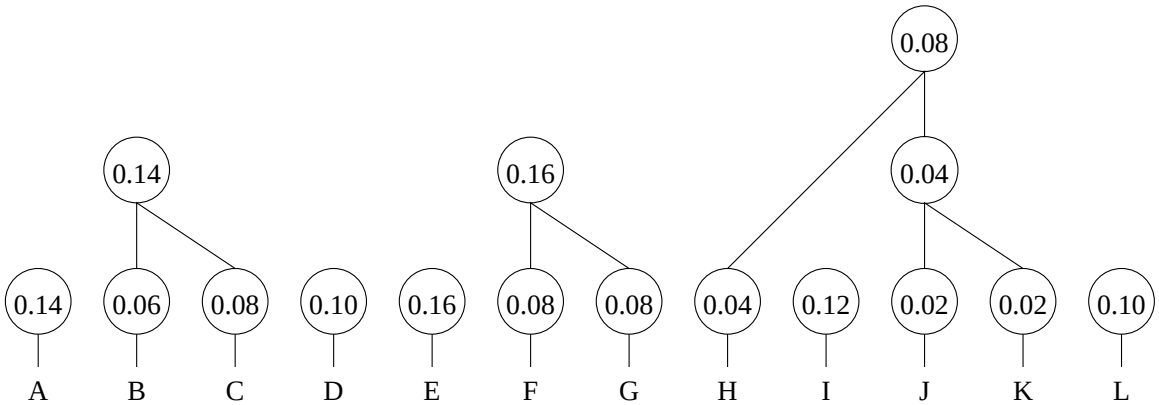
ALGORITHME DE HUFFMAN (1951)

A	0.14
B	0.06
C	0.08
D	0.10
E	0.16
F	0.08
G	0.08
H	0.04
I	0.12
J	0.02
K	0.02
L	0.10



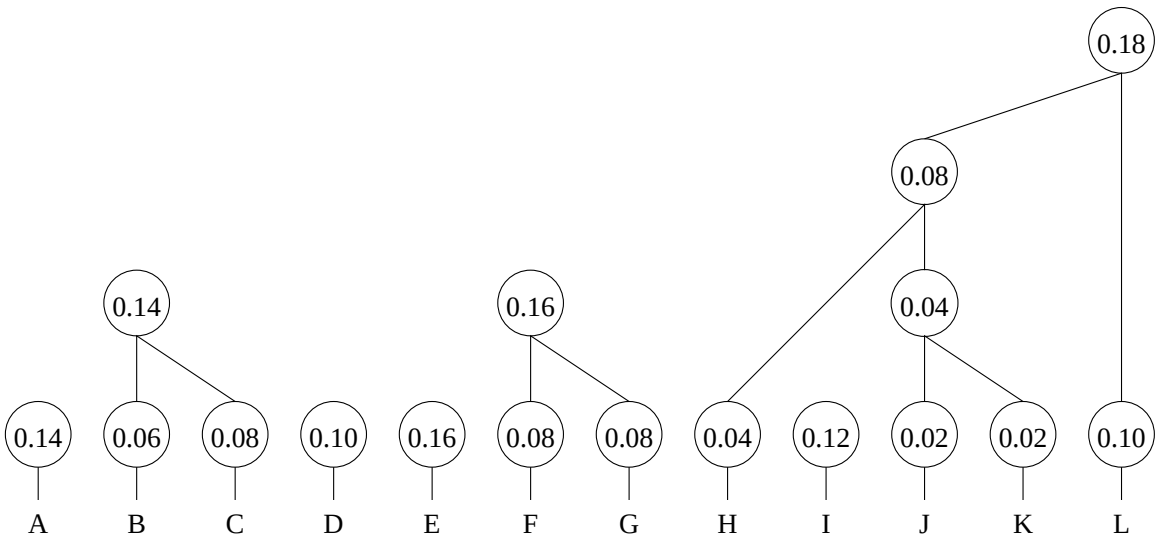
ALGORITHME DE HUFFMAN (1951)

A	0.14
B	0.06
C	0.08
D	0.10
E	0.16
F	0.08
G	0.08
H	0.04
I	0.12
J	0.02
K	0.02
L	0.10



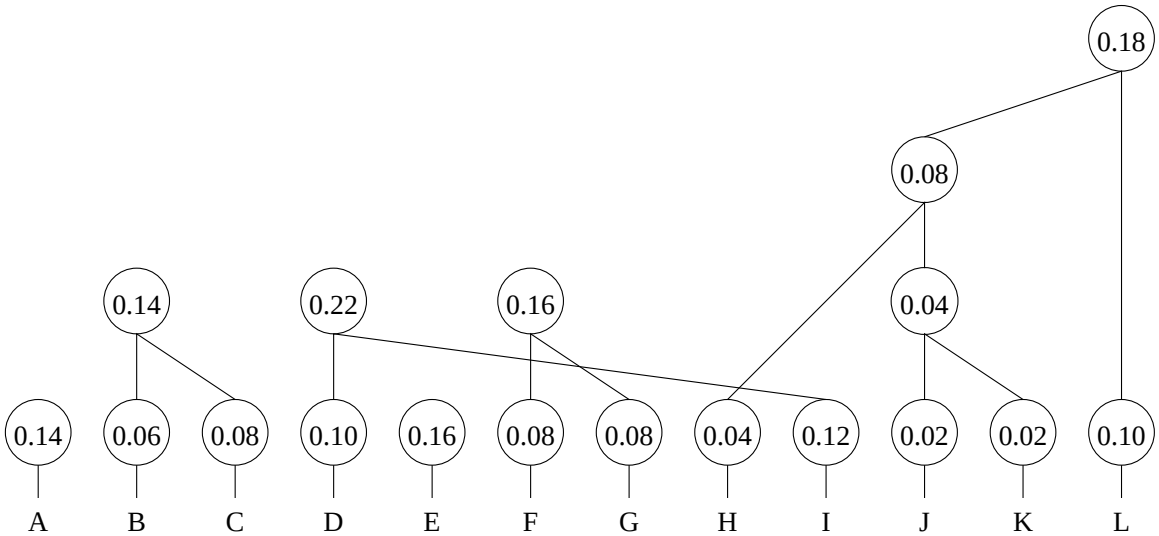
ALGORITHME DE HUFFMAN (1951)

A	0.14
B	0.06
C	0.08
D	0.10
E	0.16
F	0.08
G	0.08
H	0.04
I	0.12
J	0.02
K	0.02
L	0.10



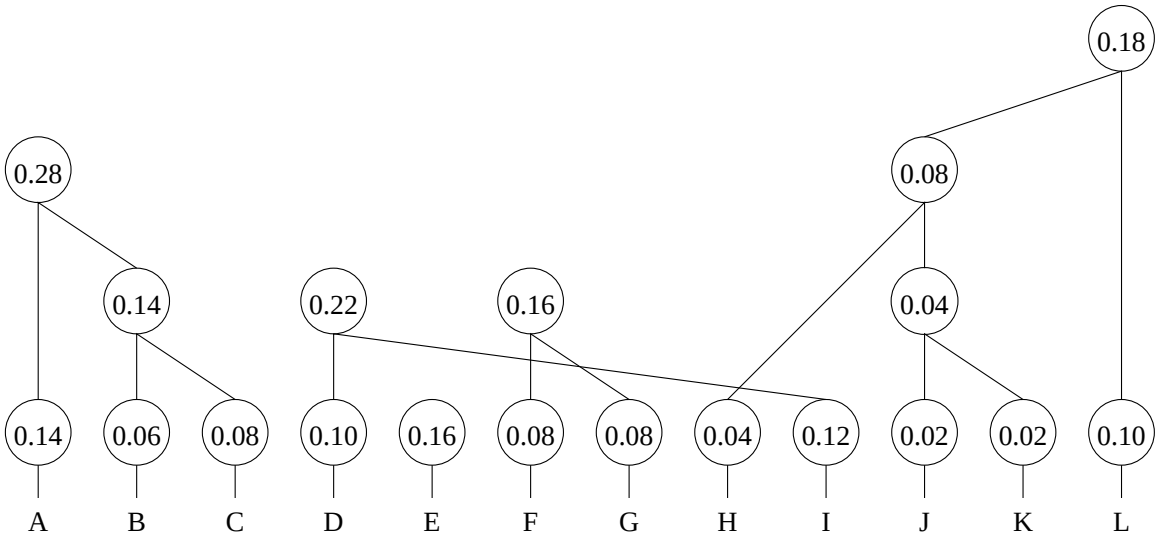
ALGORITHME DE HUFFMAN (1951)

A	0.14
B	0.06
C	0.08
D	0.10
E	0.16
F	0.08
G	0.08
H	0.04
I	0.12
J	0.02
K	0.02
L	0.10



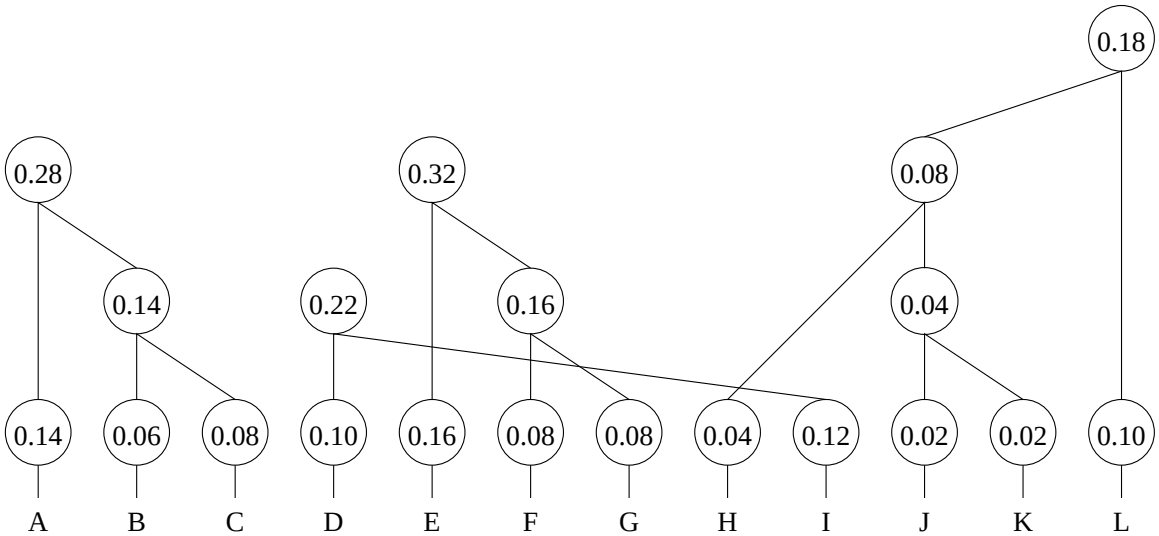
ALGORITHME DE HUFFMAN (1951)

A	0.14
B	0.06
C	0.08
D	0.10
E	0.16
F	0.08
G	0.08
H	0.04
I	0.12
J	0.02
K	0.02
L	0.10

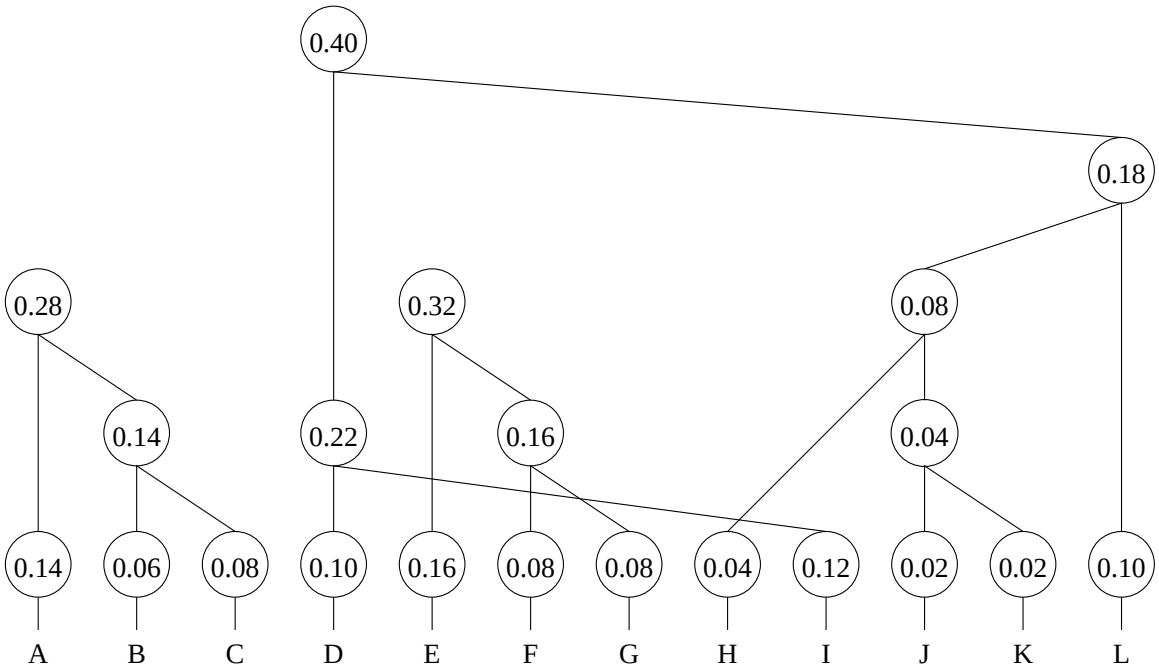


ALGORITHME DE HUFFMAN (1951)

A	0.14
B	0.06
C	0.08
D	0.10
E	0.16
F	0.08
G	0.08
H	0.04
I	0.12
J	0.02
K	0.02
L	0.10

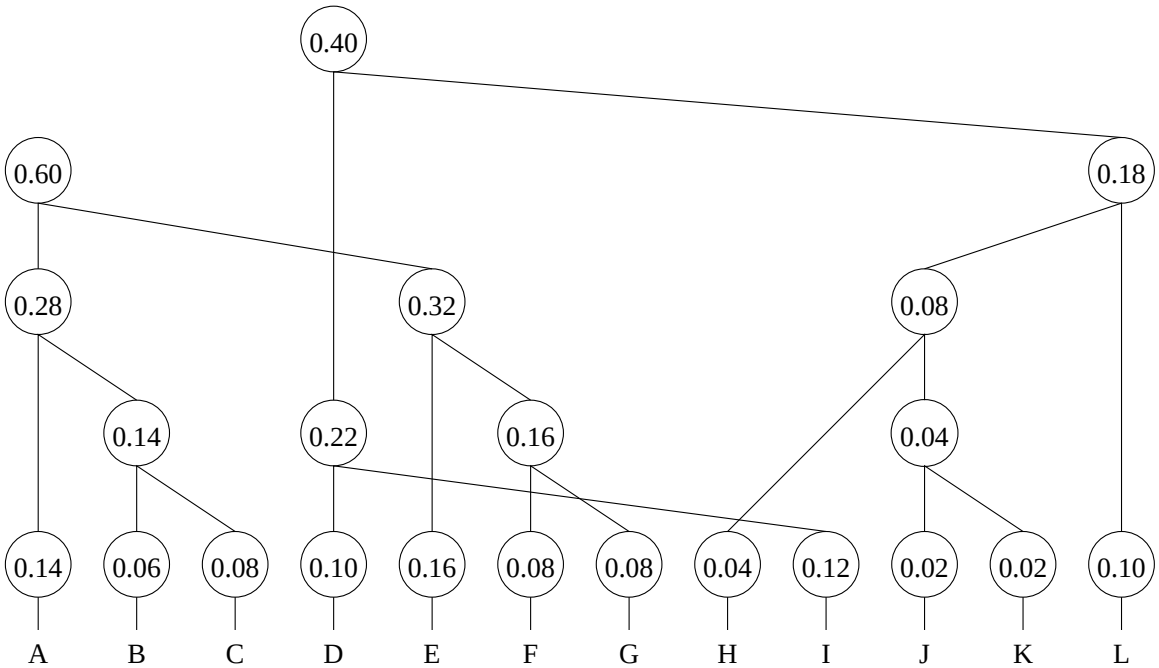


ALGORITHME DE HUFFMAN (1951)



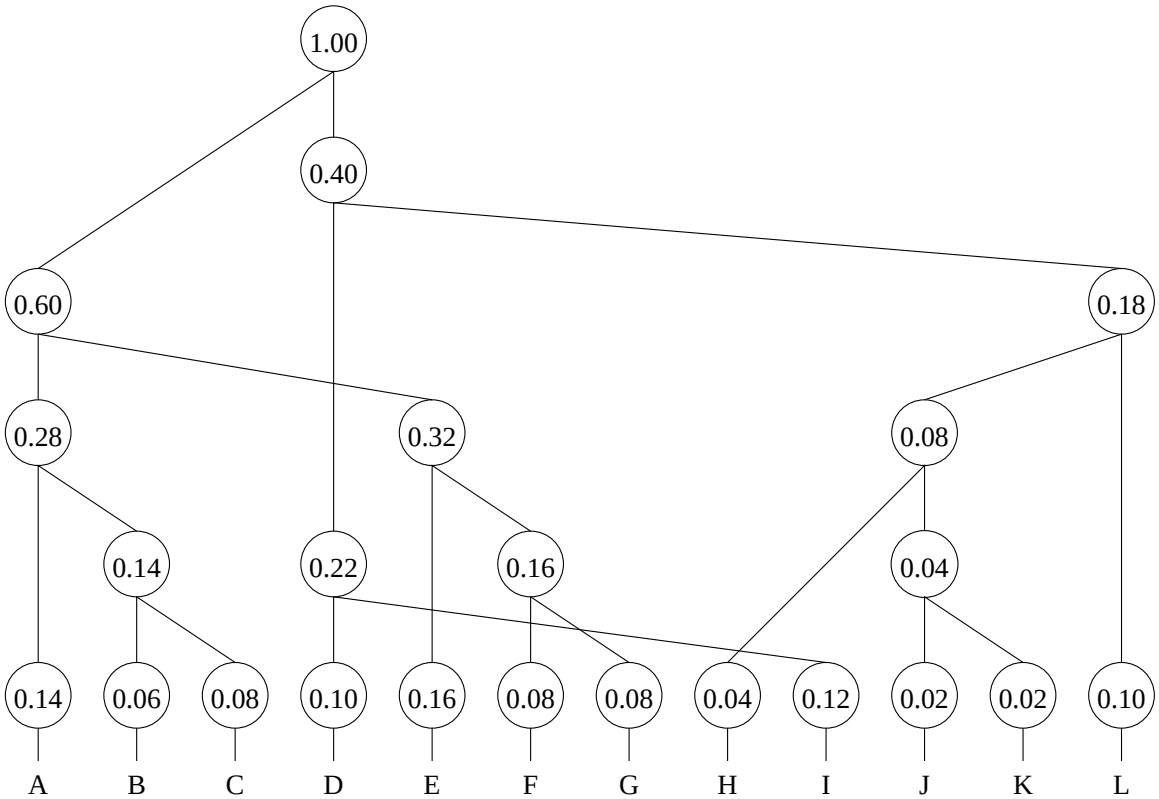
A	0.14
B	0.06
C	0.08
D	0.10
E	0.16
F	0.08
G	0.08
H	0.04
I	0.12
J	0.02
K	0.02
L	0.10

ALGORITHME DE HUFFMAN (1951)



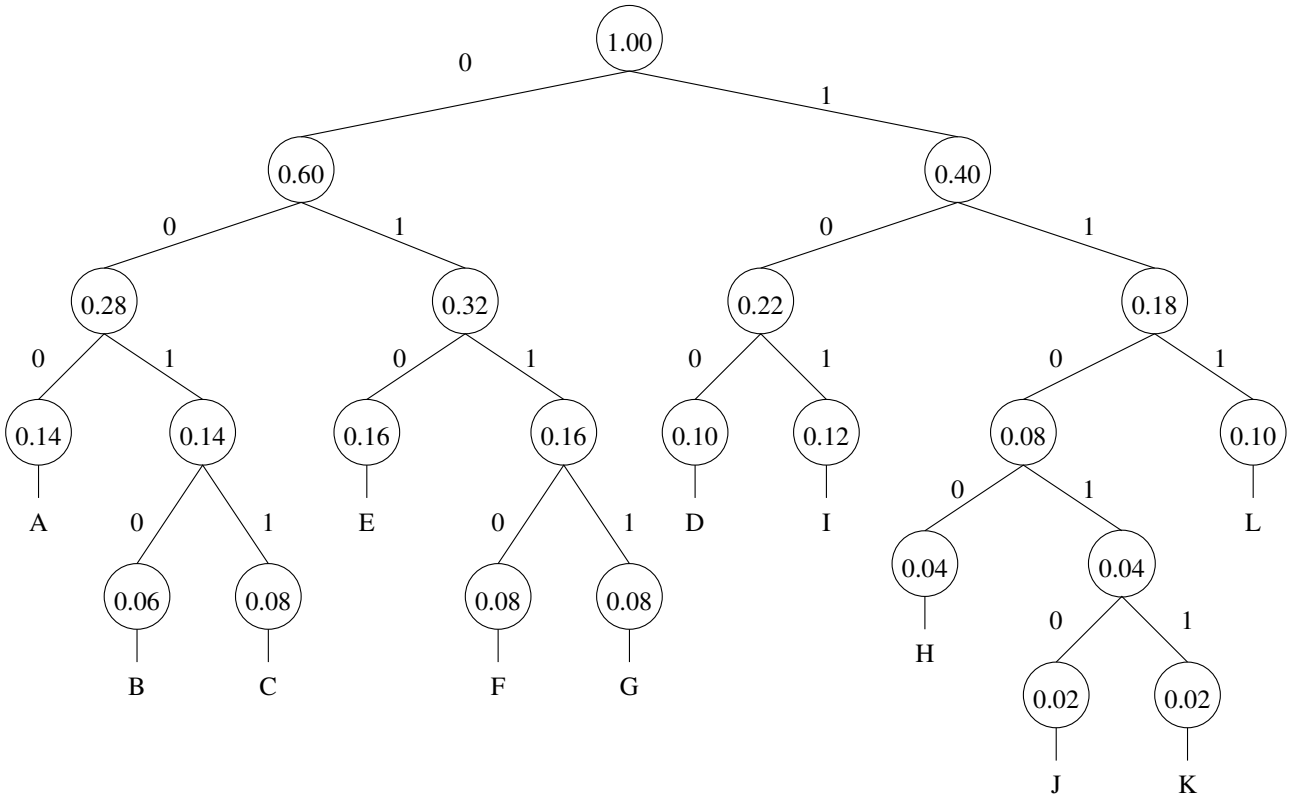
A	0.14
B	0.06
C	0.08
D	0.10
E	0.16
F	0.08
G	0.08
H	0.04
I	0.12
J	0.02
K	0.02
L	0.10

ALGORITHME DE HUFFMAN (1951)



A	0.14
B	0.06
C	0.08
D	0.10
E	0.16
F	0.08
G	0.08
H	0.04
I	0.12
J	0.02
K	0.02
L	0.10

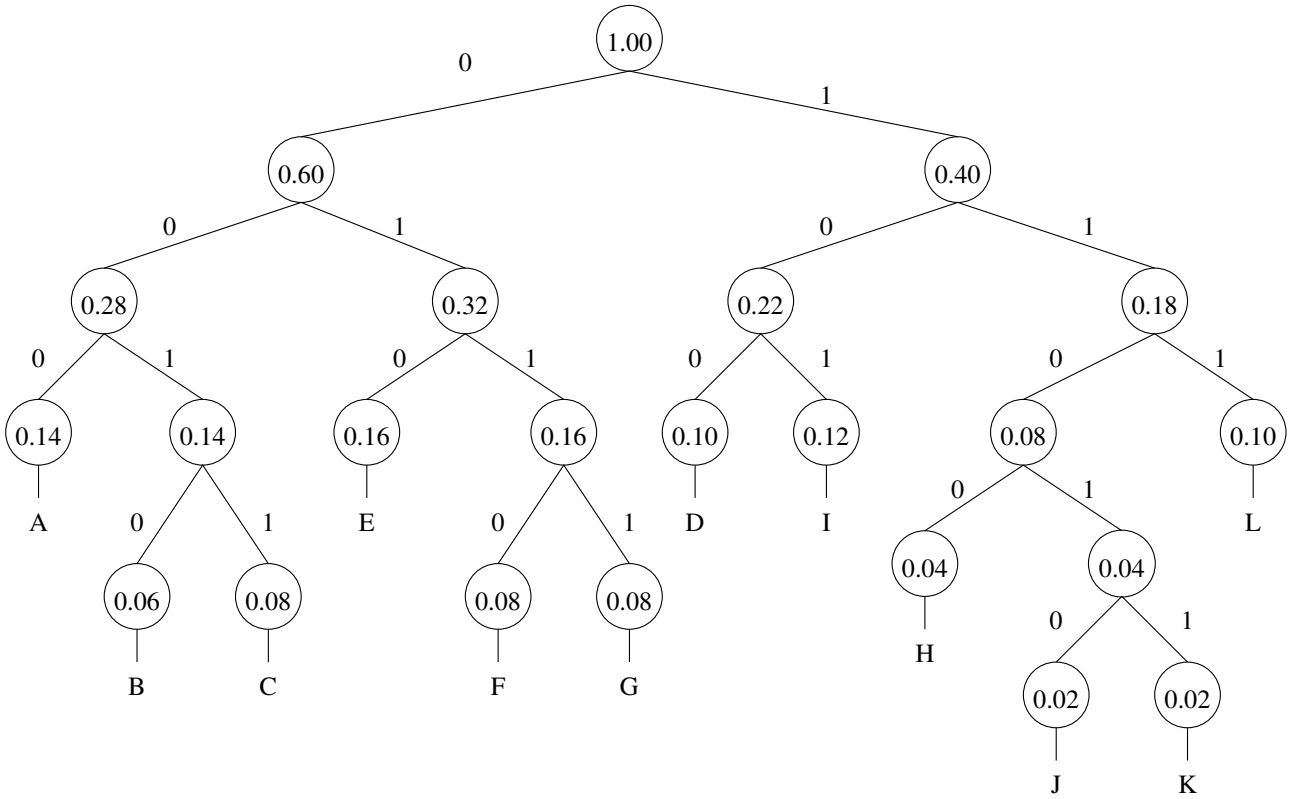
ALGORITHME DE HUFFMAN (1951)



A	0.14	000
B	0.06	0010
C	0.08	0011
D	0.10	100
E	0.16	010
F	0.08	0110
G	0.08	0111
H	0.04	1100
I	0.12	101
J	0.02	11010
K	0.02	11011
L	0.10	111

Codage optimal : L-moy = 3.42, Entropie = 3.38
Profondeur = $-\log_2$(probabilité)
Généralisation Lempel-Ziv,...

ALGORITHME DE HUFFMAN (1951)



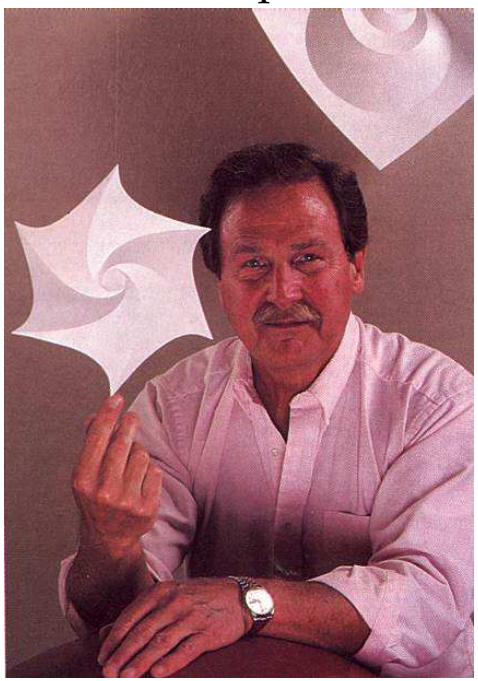
A	0.14	000
B	0.06	0010
C	0.08	0011
D	0.10	100
E	0.16	010
F	0.08	0110
G	0.08	0111
H	0.04	1100
I	0.12	101
J	0.02	11010
K	0.02	11011
L	0.10	111

Code optimal : L-inf = 3.42, Entropie = 3.38
Profondeur = $-\log_2$(probabilité)
Généralisation Lempel-Ziv(1978) et Lempel-Ziv-Welsh(1984)

UNIVERSITÉ Grenoble Alpes

DAVID A. HUFFMAN (1925-1999)

Avec ses sculptures



Huffman joined the faculty at MIT in 1953. In 1967, he went to University of California, Santa Cruz as the founding faculty member of the Computer Science Department. He played a major role in the development of the department's academic programs and the hiring of its faculty, and served as chair from 1970 to 1973. He retired in 1994, but remained active as an emeritus professor, teaching information theory and signal analysis courses.
Huffman made important contributions in many other areas, including information theory and coding, signal designs for radar and communications applications, and design procedures for asynchronous logical circuits. As an outgrowth of his work on the mathematical properties of "zero curvature Gaussian" surfaces, Huffman developed his own techniques for folding paper into unusual sculptured shapes (which gave rise to the field of computational origami).

http://www.huffmancoding.com/my-family/my-uncle/scientific-american

HUFFMAN'S ALGORITHM (1951) : IMPLANTATION

```
HUFFMAN_ALGORITHM (p)
    Données: Ensemble de k symboles S et poids p
    Résultat: Optimal prefix code
    Node x,y,z
    File_à_Priorité F
    for s ∈ S
    |   z=new_node(p(s),/,/)
    |   Insère (F,z)
    for i = 1 to K − 1
    |   x=Extrait (F)
    |   y=Extrait (F)
    |   z=new_node(x.p+y.p,x,y)
    |   Insère (F,z)
    Return Extrait (F)
```

HUFFMAN'S ALGORITHM (1951) : PREUVE

Théorème

Optimalité L'algorithme de Huffman construit un code optimal ayant la propriété du préfixe.

Invariant : La file a priorité contient une sous-forêt d'un arbre optimal de codage
La complexité de l'algorithme est $\mathcal{O}(k \log k)$

Lemme : fréquences faibles

Soit C un alphabet de k lettres. Soit x et y deux caractères de plus petite fréquence. Alors il existe un codage préfixé optimal pour C tel que les mots codes de x et de y ne diffèrent que par le dernier bit.

HUFFMAN'S ALGORITHM (1951) : PREUVE

Théorème

Optimalité L'algorithme de Huffman construit un code optimal ayant la propriété du préfixe.

Invariant : La file a priorité contient une sous-forêt d'un arbre optimal de codage
La complexité de l'algorithme est $\mathcal{O}(k \log k)$

Lemme : fréquences faibles

Soit C un alphabet de k lettres. Soit x et y deux caractères de plus petite fréquence. Alors il existe un codage préfixé optimal pour C tel que les mots codes de x et de y ne diffèrent que par le dernier bit.

Idée : prendre un arbre optimal et le transformer de manière à vérifier la propriété.

Lemme : propagation de l'optimalité

Soit T un arbre de codage optimal (complet) de C. Alors la fusion z de 2 feuilles sœurs x et y affectée de la somme des fréquences des feuilles $f(z) = f(x) + f(y)$ produit un arbre optimal pour l'alphabet C' dans lequel tous les caractères x et y ont été remplacés par z.

UNIVERSITÉ Grenoble Alpes

HUFFMAN'S ALGORITHM (1951) : PREUVE

Théorème

Optimalité L'algorithme de Huffman construit un code optimal ayant la propriété du préfixe.

Invariant : La file a priorité contient une sous-forêt d'un arbre optimal de codage
La complexité de l'algorithme est $\mathcal{O}(k \log k)$

Lemme : fréquences faibles

Soit C un alphabet de k lettres. Soit x et y deux caractères de plus petite fréquence. Alors il existe un codage préfixé optimal pour C tel que les mots codes de x et de y ne diffèrent que par le dernier bit.

Idée : prendre un arbre optimal et le transformer de manière à vérifier la propriété.

Lemme : propagation de l'optimalité

Soit T un arbre de codage optimal (complet) de C. Alors la fusion z de 2 feuilles sœurs x et y affectée de la somme des fréquences des feuilles $f(z) = f(x) + f(y)$ produit un arbre optimal pour l'alphabet C' dans lequel tous les caractères x et y ont été remplacés par z.

Idée : raisonner par l'absurde

UNIVERSITÉ Grenoble Alpes

ALGORITHME DE HUFFMAN (1951) : PREUVE

Invariant : La file a priorité contient une sous-forêt d'un arbre optimal de codage

ALGORITHME DE HUFFMAN (1951) : PREUVE

Invariant : La file a priorité contient une sous-forêt d'un arbre optimal de codage

Initialisation

Cette précondition est vraie en début de l'algorithme.

ALGORITHME DE HUFFMAN (1951) : PREUVE

Invariant : La file a priorité contient une sous-forêt d'un arbre optimal de codage

Initialisation

Cette précondition est vraie en début de l'algorithme.

Preuve partielle

Si la précondition est vraie à l'entrée de l'itération, il existe un arbre optimal contenant la forêt incluse dans la file à priorité. Soit x et y les nœuds extraits de la FAP, d'après le lemme 1 il existe un arbre optimal tel que x et y soient 2 feuilles sœurs. Par le lemme 2 (arbre optimal), lorsque l'on réalise la fusion de x et y reste optimal.

ALGORITHME DE HUFFMAN (1951) : PREUVE

Invariant : La file a priorité contient une sous-forêt d'un arbre optimal de codage

Initialisation

Cette précondition est vraie en début de l'algorithme.

Preuve partielle

Si la précondition est vraie à l'entrée de l'itération, il existe un arbre optimal contenant la forêt incluse dans la file à priorité. Soit x et y les nœuds extraits de la FAP, d'après le lemme 1 il existe un arbre optimal tel que x et y soient 2 feuilles sœurs. Par le lemme 2 (arbre optimal), lorsque l'on réalise la fusion de x et y reste optimal.

Terminaison

L'algorithme, faisant un nombre fini d'itérations, se termine. On peut montrer que chaque itération diminue de 1 le nombre de nœuds dans la FAP. A la fin des itérations il ne reste qu'un nœud racine de l'arbre optimal.

UNIVERSITÉ Grenoble Alpes

ALGORITHME DE HUFFMAN (1951) : PREUVE

Optimalité

L'algorithme de Huffman produit un code ayant la propriété du préfixe de longueur moyenne optimale.

Algorithme glouton : tout choix est définitif
à tout moment de l'algorithme la forêt construite est une sous-forêt d'un arbre optimal.
Remarque : algorithme en $\mathcal{O}(k \log k)$

Codage et compression

INFORMATION

Information. n.f. (1274 lat *informatio* action de façonner, conception, formation d'une idée dans l'esprit, notion, idée, étymologie).

1. (juridique)
2. Renseignement sur quelqu'un ou quelque chose
3. Élément ou système pouvant être transmis par un signal ou une combinaison de messages

Une machine à calculer "peut communiquer à des utilisateurs les résultats de ses calculs, c'est à dire de l'information" (De Broglie)

Dictionnaire Robert
http://www.cnrtl.fr/lexicographie/information

RÉVOLUTION NUMÉRIQUE

Toute information se représente sous forme d'une séquence finie de **nombres** (réels) (de 0 et de 1)

Langage Universel $\Rightarrow$ Machine Universelle

RÉVOLUTION NUMÉRIQUE

Toute information se représente sous forme d'une séquence finie de **nombres** (réels) (de 0 et de 1)

Langage Universel ⇒ Machine Universelle

1. Signes et langages naturels
2. Signaux : son, image, video
3. forces, champs, flux
4. . . .

CODAGE DE SUITES DE BITS

La suite 01010101010101010101 est codée par

```
Répéter n fois écrire 0 puis 1
```

Le codage binaire d'une telle chaîne de longueur n est

$$\log_2(n) + C$$

taille du codage de n + taille du code de l'algorithme

CODAGE DE SUITES DE BITS

La suite 01010101010101010101 est codée par
`Répéter n fois écrire 0 puis 1`
Le codage binaire d'une telle chaîne de longueur n est

$$\log_2(n) + C$$

taille du codage de n + taille du code de l'algorithme
La suite des "décimales" de π est codée par la formule (Bailey-Borwein-Plouffe1995)

$$\pi = \sum_{k=0}^{+\infty} \frac{1}{16k} \left(\frac{4}{8k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6} \right).$$

Taille du codage

$$\log_2(n) + C$$

COMPLEXITÉ DE KOLMOGOROV (1974)

X^* ensemble des chaînes de bits de longueur finie.
Φ fonction calculable

$$\begin{array}{rcl} \Phi : X^* & \longrightarrow & X^* \\ y & \longmapsto & x \end{array}$$

COMPLEXITÉ DE KOLMOGOROV (1974)

X^* ensemble des chaînes de bits de longueur finie.
Φ fonction calculable

$$\begin{array}{rcl} \Phi : X^* & \longrightarrow & X^* \\ y & \longmapsto & x \end{array}$$

Définition (Complexité de x relativement à Φ)

$$\mathcal{K}_\Phi(x) = \begin{cases} \inf\{longueur(y) \;\; \Phi(y) = x\} \\ +\infty \;\; si\, x \notin \Phi(X^*) \end{cases}$$

COMPLEXITÉ DE KOLMOGOROV (1974)

X^* ensemble des chaînes de bits de longueur finie.
Φ fonction calculable

$$\begin{array}{rcl} \Phi : X^* & \longrightarrow & X^* \\ y & \longmapsto & x \end{array}$$

Définition (Complexité de x relativement à Φ)

$$\mathcal{K}_\Phi(x) = \begin{cases} \inf\{longueur(y) \;\; \Phi(y) = x\} \\ +\infty \;\; si\, x \notin \Phi(X^*) \end{cases}$$

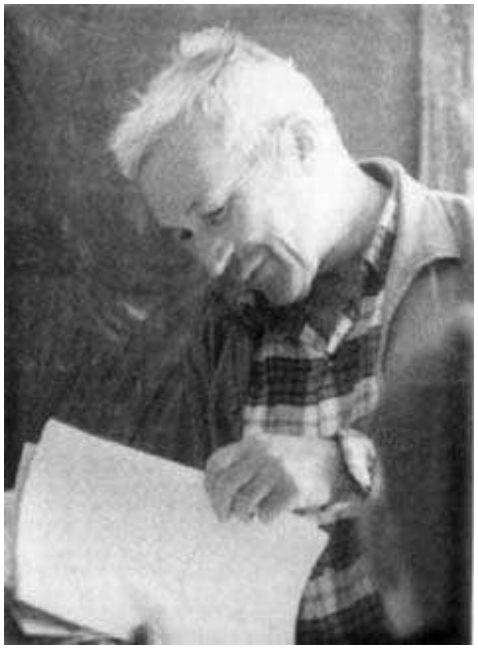
Définition (Complexité de Kolmogorov de la chaîne finie x)

$$\mathcal{K}(x) = \inf_\Phi\{\mathcal{K}_\Phi(x) + taille(\Phi)\}.$$

Remarque : $\mathcal{K}(x) \leqslant l(x)$.

UNIVERSITÉ Grenoble Alpes

Andreï Kolmogorov (1903-1987)



Mathématicien russe dont le nom est principalement attaché à l'axiomatisation du calcul des probabilités dans le cadre de la théorie des ensembles. Fils d'un agronome, Andreï Nikolaïevitch Kolmogorov est né à Tambov. Il entra à dix-sept ans à l'université de Moscou, où il suivit des cours de N. Lusin et de P. Urysohn ; chercheur associé à cette université depuis 1925, il y devint professeur en 1931 et directeur du département de mathématiques deux ans plus tard. En 1939, il fut élu à l'Académie des sciences de l'U.R.S.S.

Les premiers travaux de Kolmogorov portent sur les fonctions de variable réelle (séries trigonométriques, opérations sur les ensembles) ; en 1922, il a construit un exemple célèbre de fonction intégrable dont la série de Fourier est divergente en tout point, ce qui relançait le problème de la convergence des séries de Fourier. Quelques années plus tard, il étendit la sphère de ses recherches à la logique mathématique et aux problèmes de fondements. À partir de 1925, en collaboration avec A. Khintchine, Kolmogorov a étudié les problèmes de convergence de séries d'éléments aléatoires, sur lesquels il a publié de nombreux articles devenus classiques. Son mémoire fondamental, Théorie générale de la mesure et théorie des probabilités (1929), donne la première construction axiomatique du calcul des probabilités fondée sur la théorie de la mesure ; il développa ses idées dans l'ouvrage Grundbegriffe der Wahrscheinlichkeitsrechnung (trad. angl. Foundations of the Theory of Probability, 1950), publié en 1933. Avec son ouvrage Méthodes analytiques de théorie des probabilités, Kolmogorov est un des fondateurs de la théorie des processus stochastiques. Il a étudié plus spécialement ceux qui sont connus de nos jours sous le nom de processus de Markov où deux systèmes d'équations aux dérivées partielles portent son nom ; cette théorie a d'importantes applications en physique (mouvement brownien, diffusion). Mentionnons aussi des recherches très importantes sur les processus aléatoires stationnaires, dont Wiener a souligné le rôle dans la théorie statistique de l'information sur laquelle s'appuie, pour une part, la cybernétique.
Kolmogorov a également fait des recherches en topologie, géométrie, analyse fonctionnelle et approximation optimale des fonctions. Depuis 1950, il a publié des travaux sur la théorie de l'information, la mécanique classique et la théorie des automates finis. Il a consacré ses dernières années à des problèmes d'enseignement des mathématiques et a publié plusieurs ouvrages de pédagogie à l'usage des parents et des enseignants. Il termina sa vie à Moscou.

SUITE $\mathcal{K}$-ALÉATOIRE

Définition

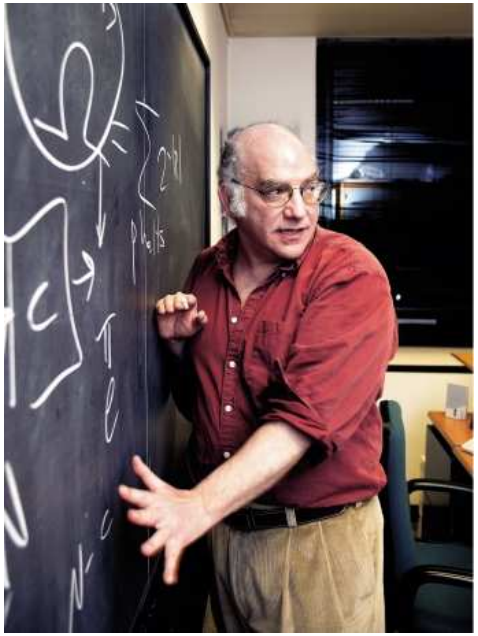
Une suite infinie $x = \{x_n\}$ est $\mathcal{K}$-aléatoire ssi il existe c tel que

$$\mathcal{K}(x_1 x_2 \cdots x_n) \geqslant n - c.$$

- suites que l'on ne peut pas coder (compresser)
- complexité linéaire
- découverte simultanée par Solomonov, Chaitin et Kolmogorov de 1964 à 1967.
- construction cohérente basée sur la machine de Turing universelle et le théorème d'invariance.
- Autres définitions basées sur le même principe Chaitin-Levin 1966.

UNIVERSITÉ Grenoble Alpes

GREGORY CHAITIN (1947-)



Gregory John Chaitin (born 1947) is an Argentine-American mathematician and computer scientist.
Beginning in the late 1960s, Chaitin made important contributions to algorithmic information theory and metamathematics, in particular a new incompleteness theorem similar in spirit to Gödel's incompleteness theorem.

Chaitin has defined Chaitin's constant Ω, a real number whose digits are equidistributed and which is sometimes informally described as an expression of the probability that a random program will halt. Ω has the mathematical property that it is definable but not computable. Chaitin's early work on algorithmic information theory paralleled the earlier work of Kolmogorov.
Chaitin also writes about philosophy, especially metaphysics and philosophy of mathematics (particularly about epistemological matters in mathematics). In metaphysics, Chaitin claims that algorithmic information theory is the key to solving problems in the field of biology (obtaining a formal definition of life, its origin and evolution) and neuroscience (the problem of consciousness and the study of the mind). Indeed, in recent writings, he defends a position known as digital philosophy. In the epistemology of mathematics, he claims that his findings in mathematical logic and algorithmic information theory show there are "mathematical facts that are true for no reason, they're true by accident. They are random mathematical facts". Chaitin proposes that mathematicians must abandon any hope of proving those mathematical facts and adopt a quasi-empirical methodology.
Chaitin's mathematical work is generally agreed to be correct, and has been cited, discussed and continued by many mathematicians. Some philosophers or logicians strongly disagree with his philosophical interpretation of it. Philosopher Panu Raatikainen argues that Chaitin misinterprets the implications of his own work and his conclusions about philosophical matters are not solid. The logician Torkel Franzén criticizes Chaitin's interpretation of Gödel's Incompleteness Theorem and the alleged explanation for it that Chaitin's work represents.
Chaitin is also the originator of using graph coloring to do register allocation in compiling, a process known as Chaitin's algorithm.

UNIVERSITÉ Grenoble Alpes

LEONID LEVIN (1948-)

Leonid Levin (born November 2, 1948, in Dnipropetrovsk USSR) is a computer scientist. He studied under Andrey Kolmogorov.
He obtained his first Ph.D. in 1972 at Moscow University. Later, he emigrated to the USA in 1978 and earned another Ph.D at the Massachusetts Institute of Technology in 1979.

He is well known for his work in randomness in computing, algorithmic complexity and intractability, foundations of mathematics and computer science, algorithmic probability, theory of computation, and information theory.
His life is described in a chapter in the book : Out of Their Minds : The Lives and Discoveries of 15 Great Computer Scientists.
Levin independently discovered a theorem that was also discovered and proven by Stephen Cook. This NP-completeness theorem, often called by inventors' names (see Cook-Levin Theorem) was a basis for one of the seven "Millennium Math. Problems" declared by Clay Mathematics Institute with a $ 1,000,000 prize offered. It was a breakthrough in computer science and is the foundation of computational complexity. Levin's journal article on this theorem was published in 1973 ; he had lectured on the ideas in it for some years before that time (see Trakhtenbrot's survey below), though complete formal writing of the results took place after Cook's publication.
He is currently a professor of computer science at Boston University, where he began teaching in 1980.

SUITE $\mathcal{K}$-ALÉATOIRE : PROPRIÉTÉS

Proposition (Presque toutes les suites sont $\mathcal{K}$-aléatoires)

Le nombre de suites x de longueur n et de complexité $\mathcal{K}(x) \geqslant n - c$ est minoré par

$$2^n(1 - 2^{-c})$$

SUITE $\mathcal{K}$-ALÉATOIRE : PROPRIÉTÉS

Proposition (Presque toutes les suites sont $\mathcal{K}$-aléatoires)

Le nombre de suites x de longueur n et de complexité $\mathcal{K}(x) \geqslant n - c$ est minoré par

$$2^n(1 - 2^{-c})$$

Preuve : le nombre de suites de longueur $< n - c$ est

$$2^0 + 2^1 + \cdots + 2^{n-c-1} = 2^{n-c} - 1.$$

SUITE $\mathcal{K}$-ALÉATOIRE : PROPRIÉTÉS

Proposition (Presque toutes les suites sont $\mathcal{K}$-aléatoires)

Le nombre de suites x de longueur n et de complexité $\mathcal{K}(x) \geqslant n - c$ est minoré par

$$2^n(1 - 2^{-c})$$

Preuve : le nombre de suites de longueur $< n - c$ est

$$2^0 + 2^1 + \cdots + 2^{n-c-1} = 2^{n-c} - 1.$$

Montrer qu'une suite est $\mathcal{K}$-aléatoire est, en général, indécidable

CODAGE ET COMPRESSION

UNIVERSITÉ Grenoble Alpes

COMPLEXITÉ DU JEU DE PILE OU FACE BIAISÉ

$$x = \{x_n\}_{n\in\mathbb{N}} \quad \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} x_i = p \neq \frac{1}{2}.$$

COMPLEXITÉ DU JEU DE PILE OU FACE BIAISÉ

$$x = \{x_n\}_{n\in\mathbb{N}} \quad \lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} x_i = p \neq \frac{1}{2}.$$

Proposition (Complexité et entropie)

$$\mathcal{K}(x_1 x_2 \cdots x_n) \leqslant n[-p\log_2 p - (1-p)\log_2(1-p)] = n\mathcal{H}_2(p).$$

*$\mathcal{H}_2(p)$ est appelée **entropie** de la distribution $(p, 1-p)$.*

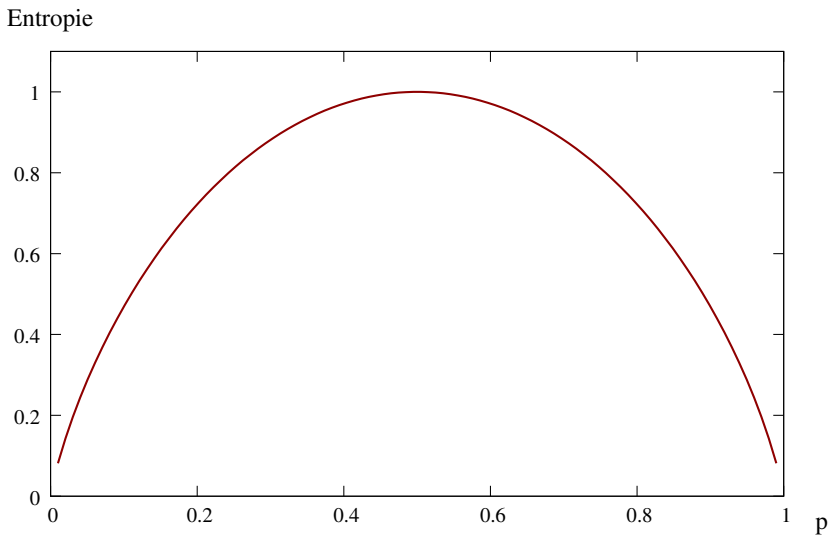
COMPLEXITÉ DU JEU DE PILE OU FACE BIAISÉ

$$x = \{x_n\}_{n\in\mathbb{N}} \quad \lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} x_i = p \neq \frac{1}{2}.$$

Proposition (Complexité et entropie)

$$\mathcal{K}(x_1 x_2 \cdots x_n) \leqslant n[-p\log_2 p - (1-p)\log_2(1-p)] = n\mathcal{H}_2(p).$$

*$\mathcal{H}_2(p)$ est appelée **entropie** de la distribution $(p, 1-p)$.*



JEU DE PILE OU FACE BIAISÉ : PREUVE $p < \frac{1}{2}$

Classer les suites de n bits par
1) ordre croissant sur le nombre de 1 ;
2) ordre alphabétique sur les suites ayant le même nb de 1.

JEU DE PILE OU FACE BIAISÉ : PREUVE $p < \frac{1}{2}$

Classer les suites de n bits par
1) ordre croissant sur le nombre de 1 ;
2) ordre alphabétique sur les suites ayant le même nb de 1.
Exemple : Pour $n = 4$
0000 0001 0010 0100 1000 0011 0101 0110 1001 1010 1100 0111 1011 1101 1110 1111.

UNIVERSITÉ Grenoble Alpes

JEU DE PILE OU FACE BIAISÉ : PREUVE $p < \frac{1}{2}$

Classer les suites de n bits par
1) ordre croissant sur le nombre de 1 ;
2) ordre alphabétique sur les suites ayant le même nb de 1.
Exemple : Pour $n = 4$
0000 0001 0010 0100 1000 0011 0101 0110 1001 1010 1100 0111 1011 1101 1110 1111.
Pour ce codage Φ, donner une suite x de taille n il suffit de donner son rang $r(x)$.

$$\mathcal{K}_\Phi(x) \sim \log_2 r(x); \quad \text{on néglige } l(\Phi) \text{ et } \log_2 l(x) \ .$$

Si k "1" dans x alors $r(x) \leqslant C_n^0 + C_n^1 + \cdots C_n^k$.

JEU DE PILE OU FACE BIAISÉ : PREUVE $p < \frac{1}{2}$

Classer les suites de n bits par
1) ordre croissant sur le nombre de 1 ;
2) ordre alphabétique sur les suites ayant le même nb de 1.
Exemple : Pour $n = 4$
0000 0001 0010 0100 1000 0011 0101 0110 1001 1010 1100 0111 1011 1101 1110 1111.
Pour ce codage Φ, donner une suite x de taille n il suffit de donner son rang $r(x)$.

$$\mathcal{K}_\Phi(x) \sim \log_2 r(x); \quad \text{on néglige } l(\Phi) \text{ et } \log_2 l(x) \ .$$

Si k "1" dans x alors $r(x) \leqslant C_n^0 + C_n^1 + \cdots C_n^k$.
Pour $k < \frac{n}{2}$ et $\frac{k}{n} \overset{+\infty}{\sim} p$

$$\begin{aligned} \log_2 r(x) &\leqslant \log_2(C_n^0 + C_n^1 + \cdots C_n^k) \sim \log_2 C_n^k \\ &\sim \log_2(p^{-k}(1-p)^{n-k}) \sim n\mathcal{H}_2(p). \end{aligned}$$

CQFD

ENTROPIE

Définition (Entropie discrète)

Pour une distribution de probabilité $p_1, \cdots, p_n$

$$\mathcal{H}(p_1, \cdots, p_k) = -\sum_{i=1}^{k} p_i \log p_i.$$

Propriétés

1. $\mathcal{H}$ maximale pour $p = \frac{1}{k}$
2. $\mathcal{H}(x) = 0$ pour $p = 0$ ou $p = 1$ (suite déterministe).
3. ....

INTERPRÉTATION DE BOLTZMAN (1877)

Modèle :

- n particules «indistingables» (n énorme),
- k états possibles pour les particules (niveaux d'énergie),
- macro-état **observable**

$$\underline{n} = (n_1, \cdots, n_k);$$

Nombre de configurations représentées par $\underline{n}$

$$\mathcal{C}(n, n_1, \cdots, n_k) = \frac{n!}{n_1! \cdots n_k!}.$$

INTERPRÉTATION DE BOLTZMAN (1877) (SUITE)

Comparaison des probabilités d'apparition de 2 macro-états

$$R(\underline{n}, \underline{n}') = \frac{\mathcal{C}(n, n_1, \cdots, n_k)}{\mathcal{C}(n, n'_1, \cdots, n'_k)} = \frac{\prod n'_i!}{\prod n_i!}.$$

Pour n très grand, $p_i = \frac{n_i}{n}$ (formule de Stirling)

$$R(\underline{n}, \underline{n}') \sim e^{n(\mathcal{H}(p) - \mathcal{H}(p'))}.$$

- Un seul macro-état p^* sera observable (réalise le maximum d'entropie).
- Décroissance exponentielle autour de ce macro-état.
- Si aucune contraintes $p^* = \frac{1}{k}$ sera observé.
- Sous la contrainte d'energie moyenne par particule fixée E :

$$p_i^* = Cexp(-\lambda E_i).$$

UNIVERSITÉ Grenoble Alpes

LUDWIG BOLTZMANN (1844-1906)

Ludwig Eduard Boltzmann, physicien autrichien né le 20 février 1844 à Vienne (Autriche), mort le 5 septembre 1906 à Duino.
Il est considéré comme le père de la physique statistique, fervent défenseur de l'existence des atomes. Validant l'hypothèse de Démocrite selon laquelle la matière peut être considérée comme un ensemble d'entités indivisibles, Ludwig Boltzmann, à l'aide de son équation cinétique dite "de Boltzmann", théorise de nombreuses équations de mécanique des fluides.

Ludwig Boltzmann obtient son doctorat à l'université de Vienne en 1866, avec une thèse sur la théorie cinétique des gaz, dirigée par Josef Stefan, dont il devient ensuite assistant.
Il étudia successivement à Graz, Heidelberg et Berlin, où il suivit les cours de Bunsen, Kirchhoff et Helmholtz.
En 1869, il obtient une chaire de physique théorique à Graz, où il reste pendant 4 ans. En 1873, il accepte une chaire de mathématiques à Vienne, mais revient à Graz 3 ans plus tard, cette fois pour enseigner la physique expérimentale. Il devient membre étranger de la Royal Society en 1899.
Il entretint des échanges, parfois vifs, avec les physiciens à propos de ses travaux. Cela affecta particulièrement Boltzmann et entraina des crises de dépression qui l'ont conduit à une première tentative de suicide à Leipzig, puis à une seconde à Duino , près de Trieste, qui lui sera malheureusement fatale. Boltzmann meurt avant même d'avoir vu ses idées s'imposer.
Au Cimetière central de Vienne, la tombe de Boltzmann a une équation inscrite au-dessus de la statue du physicien. Cette épitaphe est l'équation $S = kln\omega$, laquelle exprime l'entropie S en fonction du nombre ω des états d'énergie équiprobables possibles, avec k la constante de Boltzmann.
Les conceptions atomistiques qui sont à la base des recherches de Boltzmann lui ont valu une vigoureuse hostilité de la part de ses confrères. Faute de développements nouveaux, ses résultats entraînèrent un certain discrédit sur ses travaux théoriques, jusqu'à ce que ceux-ci soient remis à l'honneur par les découvertes de Max Planck dans l'analyse du rayonnement du corps noir, puis celles d'Albert Einstein avec l'effet photoélectrique.

UNIVERSITÉ Grenoble Alpes

Codage et compression

SYNTHÈSE

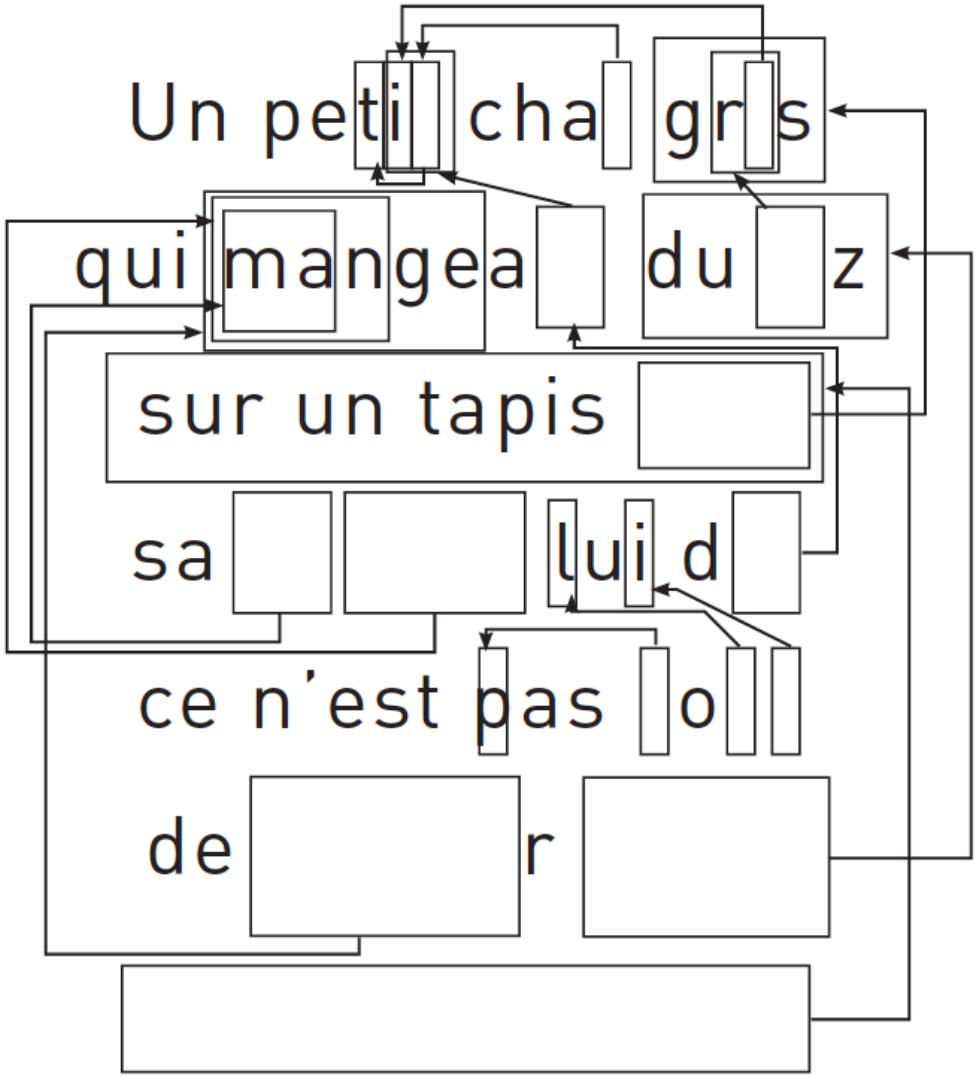
Concepts fondamentaux

- définition cohérente de la notion d'information : complexité de Kolmogorov
- extensions à d'autres indicateurs (le *temps* avec la profondeur de Bennett) ou d'autres modèles de machines
-

Opérationnels

- Quantification de l'information $\mathcal{H}(p)$ (bonne approximation)
- Algorithmes de codage basés sur la propriété du préfixe $\Rightarrow$ arbre (automate d'état fini)
- $\mathcal{H}(p)$ donne le taux de compression
- Généralisation à des codes de mots (Lempel-Ziv-Welsch)
- Algorithmique sur les textes basée sur des représentations statistiques
- Machine learning

CS UNPLUGGED



RÉFÉRENCES

- **Introduction aux sciences de l'information**, Jean-Yves Le Boudec, Patrick Thiran et Rüdiger Urbanke, Presses polytechniques et universitaires romandes, 2015
- **Théorie des codes**, J-G. Dumas, J-L. Roch, E. Tannier et S. Varrette, Dunod 2007, Site
- **L'information : L'histoire - La théorie - Le déluge** James Gleick, Cassini 2015
- **Introductions aux Probabilités** Pierre Brémaud. Springer-Verlag, Berlin, 1984.
- **Algorithmique** Thomas Cormen, Charles Leiserson, Ronald Rivest, and Clifford Stein Dunod, 2011.
- **Incertitude et Information** Silviu Guiasu and Radu Theodorecu. Les Presses de l'Université LAVAL, Québec, 1971.
- **A mathematical theory of communication** Claude Elwood Shannon, Bells Systems Technical Journal, 27, 1948.

Sources pour les biographies MacTutor History

UNIVERSITÉ Grenoble Alpes

Merci M Vidberg http://vidberg.blog.lemonde.fr/