

- 0/ Probabilités, statistiques comme outils informatiques.
- 1/ Rappels de probabilités
- 2/ Génération aléatoire (principes et méthodes). **ON EST ICI.**
- 3/ Rappels et éléments de statistiques.
- 4/ Evaluation de performances.
- 5/ Modèles mathématiques et analyse.
- 6/ Simulation.
- 7/ Méthodologie de l'évaluation de performances

# Génération aléatoire

# Génération aléatoire

- ➊ Introduction & motivations.
- ➋ Les premiers exemples.
- ➌ Les générateurs à  $k$  pas.
- ➍ Générateur de loi uniforme.

## Utilisations de nbr aléatoires

- jeux, simulations
- crypto
- autres ...

## Utilisations de nbr aléatoires

- jeux, simulations
- crypto
- autres ...

## Deux types

- Physique (pièce de monnaie, courte-paille, aiguille de Buffon, quantique, ...)
- Algorithmique (fonctions RAND, RANDOM, ...). **ON EST ICI!**

## Utilisations de nbr aléatoires

- jeux, simulations
- crypto
- autres ...

## Deux types

- Physique (pièce de monnaie, courte-paille, aiguille de Buffon, quantique, ...)
- Algorithmique (fonctions RAND, RANDOM, ...). **ON EST ICI!.**

**Le but principal est de produire des résultats imprévisibles.**

## Von Neumann Middle-square-method

675219 ← 0

L'idée est (i) de prendre un chiffre sur  $n$  digits, (ii) de calculer son carré, de prendre (iii) les digits du milieu de ce carré et ainsi de suite ...

- (m.q. ) les chiffres obtenus sont périodiques
- Parfois, ces périodes sont très courtes



- (m.q. ) les chiffres obtenus sont périodiques
- Parfois, ces périodes sont très courtes

## Exemple!

Avec comme valeur initiale 3792, on trouve  $3792^2 = 14\mathbf{3792}64$ .

- (m.q. ) les chiffres obtenus sont périodiques
- Parfois, ces périodes sont très courtes

## Exemple!

Avec comme valeur initiale 3792, on trouve  $3792^2 = 14\mathbf{3792}64$ .

- les périodes n'excèdent pas  $10^n$ !

Cependant, cette idée a fonctionné pendant assez longtemps car les "erreurs" sont souvent spectaculaires et étaient faciles à détecter!!!!

## Algo BBS

$$x_{n+1} = x_n^2 \text{ MOD } (pq)$$

où  $p$  et  $q$  sont de très grand nombres premiers particuliers .

## Algo BBS

$$x_{n+1} = x_n^2 \text{ MOD } (pq)$$

où  $p$  et  $q$  sont de très grand nombres premiers particuliers .

## Inconvénient majeur

Cet algorithme est très lent et n'a pas d'applications pratiques à part en **cryptographie** .

### Définition

Ce sont les générateurs de la forme

$$x_n = f(x_{n-1}, x_{n-2}, \dots, x_{n-k}).$$

où  $x_0, x_1, x_{k-1}$  sont des conditions initiales.

## Définition

Ce sont les générateurs de la forme

$$x_n = f(x_{n-1}, x_{n-2}, \dots, x_{n-k}).$$

où  $x_0, x_1, x_{k-1}$  sont des conditions initiales.

## Deux exemples fameux

Le générateur linéaire à  $k$  pas:

$$x_0, x_1, x_{k-1}; x_{n+k} = \sum_{j=0}^{k-1} c_j x_{n+j}$$

## Définition

Ce sont les générateurs de la forme

$$x_n = f(x_{n-1}, x_{n-2}, \dots, x_{n-k}).$$

où  $x_0, x_1, x_{k-1}$  sont des conditions initiales.

## Deux exemples fameux

Le générateur linéaire à  $k$  pas:

$$x_0, x_1, x_{k-1}; x_{n+k} = \sum_{j=0}^{k-1} c_j x_{n+j}$$

Le congruentiel linéaire

$$x_0; x_{n+1} = ax_n + b \text{ MOD } m$$

## Le générateur à un pas comme cas d'étude

Dans l'exemple précédent, on s'aperçoit qu'on va travailler sur  $\mathbb{Z}_m$ . On se pose alors la question : comment choisir les paramètres pour optimiser le générateur?



## Le générateur à un pas comme cas d'étude

Dans l'exemple précédent, on s'aperçoit qu'on va travailler sur  $\mathbb{Z}_m$ . On se pose alors la question : comment choisir les paramètres pour optimiser le générateur?

### Définition

Soit  $F$  un ens. fini non vide. Soit  $x_0 \in F$  et  $f : F \rightarrow F$  une application. Un générateur à un pas est donné par le triplet  $(F, f, x_0)$ .

## Le générateur à un pas comme cas d'étude

Dans l'exemple précédent, on s'aperçoit qu'on va travailler sur  $\mathbb{Z}_m$ . On se pose alors la question : comment choisir les paramètres pour optimiser le générateur?

### Définition

Soit  $F$  un ens. fini non vide. Soit  $x_0 \in F$  et  $f : F \rightarrow F$  une application. Un générateur à un pas est donné par le triplet  $(F, f, x_0)$ .

### Caractéristiques d'un générateur à un pas

- (i) La suite  $(x_n)_{n \geq 0}$  est périodique.
- (ii) Il existe  $\mu \in \mathbb{N}$  et  $\lambda$  tels que  $\forall n \geq \mu$  et  $k \geq 0$  alors

$$x_n = x_{n+k\lambda} :$$

## Le générateur à un pas comme cas d'étude

Dans l'exemple précédent, on s'aperçoit qu'on va travailler sur  $\mathbb{Z}_m$ . On se pose alors la question : comment choisir les paramètres pour optimiser le générateur?

### Définition

Soit  $F$  un ens. fini non vide. Soit  $x_0 \in F$  et  $f : F \rightarrow F$  une application. Un générateur à un pas est donné par le triplet  $(F, f, x_0)$ .

### Caractéristiques d'un générateur à un pas

- (i) La suite  $(x_n)_{n \geq 0}$  est périodique.
- (ii) Il existe  $\mu \in \mathbb{N}$  et  $\lambda$  tels que  $\forall n \geq \mu$  et  $k \geq 0$  alors

$$x_n = x_{n+k\lambda} :$$

Autrement dit, les valeurs de la suite  $(x_n)$  sont dans un ensemble

$$\{x_j; 0 \leq j \leq \lambda - 1\} :$$

qu'on appelle **ORBITE** du générateur.

## Exemples

On peut calculer (exo de TDs) les valeurs suivantes :

Grain	fonction	orbite
0	$f(x) = x*x + x + 1 \text{ MOD } 41$	{0,1,3,13,19,12,34,2,7,16,27}
5	f	{5,31,9}
8	f	{8,32}
14	f	{14,6,2,7,16,27,19,12,34}

On peut calculer (exo de TDs) les valeurs suivantes :

Grain	fonction	orbite
0	$f(x) = x*x + x + 1 \text{ MOD } 41$	{0,1,3,13,19,12,34,2,7,16,27}
5	f	{5,31,9}
8	f	{8,32}
14	f	{14,6,2,7,16,27,19,12,34}

### Définition

Une suite  $(x_n)$  est **ultimement périodique** ssi il existe un rang  $N$  à partir duquel elle est périodique.

$$\exists N, \exists t > 0, \forall n \geq N \ x_{n+t} = x_n.$$

On peut calculer (exo de TDs) les valeurs suivantes :

Grain	fonction	orbite
0	$f(x) = x^*x + x + 1 \text{ MOD } 41$	$\{0,1,3,13,19,12,34,2,7,16,27\}$
5	f	$\{5,31,9\}$
8	f	$\{8,32\}$
14	f	$\{14,6,2,7,16,27,19,12,34\}$

### Définition

Une suite  $(x_n)$  est **ultimement périodique** ssi il existe un rang  $N$  à partir duquel elle est périodique.

$$\exists N, \exists t > 0, \forall n \geq N \ x_{n+t} = x_n.$$

Autrement dit, la suite peut s'écrire comme

$$(\cdots (x_N, x_{N+1} \cdots x_{N+t-1})^t)$$

On procède aux comparaisons suivantes

$x_0$	$x_1$	$\dots$	$x_{2^k-1}$	$\dots$
$x_1$	$x_2, x_3$	$\dots$	$x_{2^k}, \dots, x_{2^{k+1}-1}$	$\dots$

jusqu'à trouver une coïncidence  $x_{2^\ell-1} = x_m$  avec  $m \in [2^\ell, \dots, 2^{\ell+1}-1]$ . Celle-ci se produit dès que  $2^\ell - 1 \geq e$  (indice d'entrée dans le cycle qui est inconnu) et que  $2^\ell \geq \lambda$ . On connaît alors  $\lambda = m - (2^\ell - 1)$ .

On procède aux comparaisons suivantes

$x_0$	$x_1$	$\cdots$	$x_{2^k-1}$	$\cdots$
$x_1$	$x_2, x_3$	$\cdots$	$x_{2^k}, \cdots, x_{2^{k+1}-1}$	$\cdots$

jusqu'à trouver une coïncidence  $x_{2^\ell-1} = x_m$  avec  $m \in [2^\ell, \cdots, 2^{\ell+1}-1]$ . Celle-ci se produit dès que  $2^\ell - 1 \geq e$  (indice d'entrée dans le cycle qui est inconnu) et que  $2^\ell \geq \lambda$ . On connaît alors  $\lambda = m - (2^\ell - 1)$ .

On peut donc procéder aux comparaisons

$x_0$	$x_1$	$\cdots$	$x_k$	$\cdots$
$x_\lambda$	$x_{\lambda+1}$	$\cdots$	$x_{\lambda+k}$	$\cdots$

en sachant que la première a lieu pour  $k = e$ .



On procède aux comparaisons

$x_0$	$x_1$	$\dots$	$x_d$	$\dots$
$x_1$	$x_2$	$\dots$	$x_{2k}$	$\dots$

la première coïncidence se produit pour  $k$  multiple de  $\lambda$  qui dépasse  $e$ .  
L'indice  $k$  trouvé est forcément un multiple de  $\lambda$  et on procède alors  
aux comparaisons suivantes

$x_0$	$x_1$	$\dots$	$x_\ell$	$\dots$
$x_d$	$x_{d+1}$	$\dots$	$x_{d+\ell}$	$\dots$

On procède aux comparaisons

$x_0$	$x_1$	$\dots$	$x_d$	$\dots$
$x_1$	$x_2$	$\dots$	$x_{2k}$	$\dots$

la première coïncidence se produit pour  $k$  multiple de  $\lambda$  qui dépasse  $e$ .  
L'indice  $k$  trouvé est forcément un multiple de  $\lambda$  et on procède alors  
aux comparaisons suivantes

$x_0$	$x_1$	$\dots$	$x_\ell$	$\dots$
$x_d$	$x_{d+1}$	$\dots$	$x_{d+\ell}$	$\dots$

**La première coïncidence se produit pour  $\ell = \mu$ .** On finit  
en déterminant  $\lambda$  par

$x_0$	$x_1$	$\dots$	$x_d$	$\dots$
$x_1$	$x_2$	$\dots$	$x_{2k}$	$\dots$

## Algorithme de Floyd

On procède aux comparaisons

$x_0$	$x_1$	$\dots$	$x_d$	$\dots$
$x_1$	$x_2$	$\dots$	$x_{2k}$	$\dots$

la première coïncidence se produit pour  $k$  multiple de  $\lambda$  qui dépasse  $e$ .  
L'indice  $k$  trouvé est forcément un multiple de  $\lambda$  et on procède alors aux comparaisons suivantes

$x_0$	$x_1$	$\dots$	$x_\ell$	$\dots$
$x_d$	$x_{d+1}$	$\dots$	$x_{d+\ell}$	$\dots$

La première coïncidence se produit pour  $\ell = \mu$ . On finit en déterminant  $\lambda$  par

$x_0$	$x_1$	$\dots$	$x_d$	$\dots$
$x_1$	$x_2$	$\dots$	$x_{2k}$	$\dots$

Le premier  $k$  qui donne une égalité est  $k = \lambda$ .

Ils sont de la forme

$$x_0; x_{n+1} = ax_n + b \text{ MOD } m$$

Ils sont ceux qui sont le plus souvent utilisés dans divers systèmes et langages. En voici quelques exemples

$x_0 = 0$	$x_{n+1} = 4x_n + 1 \text{ MOD } 9$	$(0 \rightarrow 1 \rightarrow 5 \rightarrow 3 \rightarrow 4 \rightarrow 8 \rightarrow 6 \rightarrow 7 \rightarrow 2)^\infty$
$x_0 = 0$	$x_{n+1} = 2x_n + 1 \text{ MOD } 5$	$(0 \rightarrow 1 \rightarrow 3 \rightarrow 2)^\infty$
$x_0 = 4$	$x_{n+1} = 2x_n + 1 \text{ MOD } 5$	$(4 \rightarrow)^\infty$

Utiliser Brent ou Floyd pour trouver la période de votre générateur préféré!

### **Théorème. Hull, Dobell 1966**

Pour qu'un GCL  $x_{n+1} = ax_n + b \text{ MOD } m$  soit de période maximale, IL FAUT ET IL SUFFIT que:

- (i)  $b$  soit inversible MOD  $m$  (ils sont premier entre eux);
- (ii)  $a \equiv 1 [p]$  pour tout  $p$  premier diviseur de  $m$ ;
- (iii) Si 4 divise  $m$  alors  $a \equiv 1 [4]$ .

**Preuve.** Le sens “période maximale”  $\rightarrow$  critère.

On a  $x_n = a^n x_0 + [n]_a \cdot b$

### Théorème. Hull, Dobell 1966

Pour qu'un GCL  $x_{n+1} = ax_n + b \text{ MOD } m$  soit de période maximale, IL FAUT ET IL SUFFIT que:

- (i)  $b$  soit inversible MOD  $m$  (ils sont premier entre eux);
- (ii)  $a \equiv 1 [p]$  pour tout  $p$  premier diviseur de  $m$ ;
- (iii) Si 4 divise  $m$  alors  $a \equiv 1 [4]$ .

**Preuve.** Le sens “période maximale”  $\rightarrow$  critère.

On a  $x_n = a^n x_0 + [n]_a \cdot b$

S'il y a une période maximale, elle passe par zéro.

### Théorème. Hull, Dobell 1966

Pour qu'un GCL  $x_{n+1} = ax_n + b \text{ MOD } m$  soit de période maximale, IL FAUT ET IL SUFFIT que:

- (i)  $b$  soit inversible  $\text{MOD } m$  (ils sont premier entre eux);
- (ii)  $a \equiv 1 [p]$  pour tout  $p$  premier diviseur de  $m$ ;
- (iii) Si 4 divise  $m$  alors  $a \equiv 1 [4]$ .

**Preuve.** Le sens “période maximale”  $\rightarrow$  critère.

On a  $x_n = a^n x_0 + [n]_a \cdot b$

S'il y a une période maximale, elle passe par zéro. On pose  $y_0 = x_{n_0} = 0$ , on a  $y_n = [n]_a \cdot b$  et la période est max pour  $y_n$ . Il existe  $n_1$  tel que  $[n_1]_a \cdot b = 1$  d'où (i).



Soit  $p$  un diviseur de  $m$ . Si la période est maximale dans  $\mathbb{Z}_m$  alors elle l'est aussi dans  $\mathbb{Z}_p$ .

Si  $a \not\equiv 1 [p]$  on a dans  $\mathbb{Z}_p$  le point fixe  $\frac{b}{1-a}$  ce qui est incompatible avec la période maximale. D'où **(ii)**.

Si 4 divise  $m$ , comme  $a \equiv 1 [2]$  on a  $a \equiv 1, 3 [4]$ . Si  $a \equiv 3 \equiv -1 [4]$  on a  $x_{n+2} = -(-x_n + b) + b = x_n$ , il n'y a donc pas de période maximale d'où **(iii)**.

On écrit  $m = \prod_p p^{\alpha(p)}$ , il suffit de démontrer que la période est maximale dans tous les  $\mathbb{Z}_{p^\alpha}$ .

On regarde le cycle de 0 engendré par  $y_0 = 0$ ;

on a  $y_n \equiv [n]_a \cdot b [m]$ . Puisque  $b$  est inversible modulo  $m$ , il suffit de m. q le critère entraîne  $[n]_a$  est de période  $m$  dans  $\mathbb{Z}_m$ .

On montre que la période de  $[n]_a$  est maximale à l'aide de l'id.  
 $[sq]_a = [s]_a \cdot [q]_{a^s}$ .

$$15x + 3 \text{ MOD } 7 \text{ ???}$$

$$8x + 3 \text{ MOD } 7 \text{ ???}$$

$$7x + 4 \text{ MOD } 11 \text{ ???}$$

## Bons et mauvais exemples (2/2)

```
> restart:
  f[1]:=x->15*x + 3 mod 7:
> f[2]:=x->8*x + 3 mod 7:
> f[3]:=x->7*x+4 mod 11:
> for j from 1 to 3 do x[0,j]:=12: od;
```

$$x_{0,1} := 12$$

$$x_{0,2} := 12$$

$$x_{0,3} := 12$$

(1)

```
> for i from 1 to 40 do
  for j from 1 to 3 do
    x[i,j]:=f[j](x[i-1,j]);
  od;
od;
> for j from 1 to 3 do
  seq( x[i,j], i=0..40);
  printf("-----\n");
od;
```

12, 1, 4, 0, 3, 6, 2, 5, 1, 4, 0, 3, 6, 2, 5, 1, 4, 0, 3, 6, 2, 5, 1, 4, 0, 3, 6, 2, 5, 1, 4, 0,  
3, 6, 2, 5, 1, 4, 0, 3, 6

-----  
12, 1, 4, 0, 3, 6, 2, 5, 1, 4, 0, 3, 6, 2, 5, 1, 4, 0, 3, 6, 2, 5, 1, 4, 0, 3, 6, 2, 5, 1, 4, 0,  
3, 6, 2, 5, 1, 4, 0, 3, 6

-----  
12, 0, 4, 10, 8, 5, 6, 2, 7, 9, 1, 0, 4, 10, 8, 5, 6, 2, 7, 9, 1, 0, 4, 10, 8, 5, 6, 2, 7, 9,