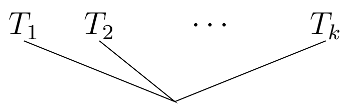
ARBRES DE GALTON-WATSON CRITIQUES - CORRIGÉ

On a rédigé les programmes demandés en pseudo-code, les variables des programmes apparaissant en gras.

1.1 Dans tous les programmes, on prendra μ égale à une loi de Poisson de paramètre 1 : $\mu(k) = \mathbb{P}[\xi = k] = \frac{e^{-1}}{k!}$. C'est effectivement une répartition raisonnable pour la taille d'une portée, si l'on impose la taille moyenne égale à 1. On suppose donnée une fonction `Poisson()` qui simule une telle variable. Alors, on produit les n premiers niveaux d'un arbre de Galton-Watson avec une telle variable de reproduction de la façon récursive suivante :

```
CalculGaltonWatson(n):
    si n == 0:
        retourner la liste vide [] (encodant l'arbre trivial racine)
    sinon:
        xi = Poisson()
        retourner une liste constituee de xi arbres
        aleatoires CalculGaltonWatson(n-1)
```

Pour dessiner un arbre à partir d'une liste de listes de..., on peut procéder comme suit. On définit un dessin d'un arbre T comme une suite de segments à coordonnées entières, une certaine largeur prévue pour le dessin, et un emplacement pour la racine (plus petit que la largeur). Supposons donnés k arbres $T_1, T_2, \ldots, T_k$, dont les dessins ont des largeurs $l_1, l_2, \ldots, l_k$. Alors, en prévoyant un espace de taille 1 entre chaque arbre, il faut une largeur $l_1 + l_2 + \cdots + l_k + k - 1$ pour dessiner l'arbre



De plus, le dessin de l'arbre global est obtenu en décalant les dessins de chacun des arbres $T_1, T_2, T_3, \ldots, T_k$ respectivement par les vecteurs $(0,1)$, $(l_1+1, 1)$, $(l_1+l_2+2, 1)$, *etc.* jusqu'à $(l_1 + l_2 + \cdots + l_{k-1} + k - 1, 1)$. Le programme suivant met en application ces observations, en calculant récursivement le dessin d'un arbre :

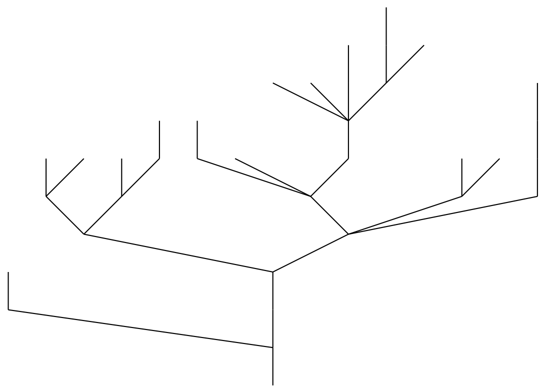
```
Dessin(arbre):
    si arbre == []:
        retourner la liste vide [], la largeur 0 et la racine 0
    sinon:
        k = longueur de la liste arbre (nombre de sous-arbres)
        listedessins = []
        l = [] (la liste des largeurs)
        fils = [] (la liste des racines)
        pour tout sousarbre dans arbre:
            sousdessin,souslargeur,sousracine = Dessin(sousarbre)
            ajouter sousdessin a listedessins, souslargeur a l,
            et sousracine a fils
        dessin = []
```

```
        racine = ((somme(l)+k-1)/2,0)
        pour i entre 1 et k:
            decalage = (somme(l[j] pour j entre 1 et i-1) + i-1,1)
            prendre la i-ieme liste dans sousdessin
            decaler tous ses vecteurs ((v,w),(x,y)) de decalage
            ajouter ces vecteurs decales a dessin
            ajouter aussi le vecteur (racine,fils[i]+decalage)
        retourner dessin, (somme(l)+k-1), (somme(l)+k-1)/2
```

On obtient enfin le dessin des n premières générations d'un arbre de Galton-Watson avec :

```
GaltonWatson(n):
    dessin,largeur,racine = Dessin(CalculGaltonWatson(n))
    tracer tous les segments de la liste dessin
```

On a tracé ci-après un exemple en taille $n = 10$, en gardant l'un des rares arbres de Galton-Watson critiques qui n'est pas encore éteint à cette génération :



1.2 La structure récursive des arbres permet de calculer aisément la taille des n premières générations d'un arbre vu comme une liste : c'est par définition la longueur de la liste, suivie des sommes des tailles des $n-1$ premières générations des éléments de la liste. Ces tailles sont donc données par la fonction :

```
taille_generations(n,L):
    si L == []:
        retourner une liste de n valeurs 0
    sinon
        mat = matrice dont les lignes sont les longueurs des n-1
        premieres generations des sous-arbres de la racine
        retourner la liste dont le premier element est la longueur
        de L, suivi des sommes des colonnes de mat
```

On en déduit immédiatement le programme :

```
Generations(n):
    retourner taille_generations(n,CalculGaltonWatson(n))
```

Sur l'exemple précéndent, on obtient la liste des tailles $[1, 2, 2, 2, 5, 10, 4, 5, 3, 1]$.

1.3 La suite $(Y_n)_{n\in\mathbb{N}}$ est une martingale positive, donc elle converge presque sûrement, et sa limite est une variable aléatoire intégrable (lemme de Fatou). D'autre part, elle est à valeurs entières, donc cette convergence est stationnaire : pour presque tout $\omega \in \Omega$, il existe un entier $k(\omega)$ tel que $Y_n(\omega) = k(\omega)$ pour n assez grand. Supposons par l'absurde $\mathbb{P}[k(\omega) = k] > 0$ pour un entier $k \geq 1$, que l'on fixe dans la suite du raisonnement. Soit $\tau_1 < \tau_2 < \cdots$ les temps de visite de l'état k par la chaîne de Markov $(Y_n)_{n\in\mathbb{N}}$; ce sont des temps d'arrêt, et par la propriéte de Markov forte, les quantités $\tau_2 - \tau_1, \tau_3 - \tau_2, \ldots$ sont des variables aléatoires positives indépendantes et de même loi. Notons que k n'est pas un état absorbant, car la probabilité de transition

$$P(k, 0) = \mathbb{P}[\xi = 0]^k > 0$$

est non nulle (les hypothèses $\mathbb{E}[\xi] = 1$ et $\xi \neq 1$ impliquent $\mathbb{P}[\xi = 0] > 0$). Par conséquent, $\mathbb{P}[\tau_{j+1} - \tau_j = 1] < 1$, et par le lemme de Borel-Cantelli,

$$\mathbb{P}\left[\limsup_j \{\tau_{j+1} - \tau_j \geq 2\}\right] = 1.$$

Ce résultat est incompatible avec une probabilité non nulle d'être stationnaire à k (dans cet événement, $\tau_{j+1} - \tau_j = 1$ pour j assez grand). Donc, $k(\omega) = 0$ presque sûrement, ce qui achève la preuve de la Proposition 1.

Pour calculer $\mathbb{E}[s^{Y_n}]$, on raisonne par récurrence sur n, en conditionnant par rapport à la valeur de Y_{n-1} :

$$\mathbb{E}[s^{Y_n}] = \mathbb{E}[\mathbb{E}[s^{Y_n}|Y_{n-1}]] = \mathbb{E}[(\mathbb{E}[s^{\xi}])^{Y_{n-1}}] = \mathbb{E}[(G(s))^{Y_{n-1}}].$$

Comme $\mathbb{E}[s^{Y_0}] = s^M$, la formule $\mathbb{E}[s^{Y_n}] = (G^{\circ n}(s))^M$ s'en déduit. En particulier,

$$\mathbb{P}[Y_n = 0] = \sum_{k=0}^{\infty} \mathbb{P}[Y_n = k]\, 0^k = \mathbb{E}[0^{Y_n}] = (G^{\circ n}(0))^M.$$

La fonction $G : [0, 1] \to [0, 1]$ est convexe, avec $G(0) = \mathbb{P}[\xi = 0] > 0$, $G(1) = 1$ et $G'(1) = \mathbb{E}[\xi] = 1$. Elle admet donc pour unique point fixe $s = 1$, et $G(s) > s$ si $s \in [0, 1)$. Donc, si $x_n = G^{\circ n}(0)$, alors $(x_n)_{n\in\mathbb{N}}$ est une suite strictement croissante qui tend vers 1. Au voisinage de 1, $G(s) = s + \frac{\sigma^2(1-s)^2}{2} + o((1-s)^2)$. Soit $y_n = \frac{1}{1-x_{n+1}} - \frac{1}{1-x_n}$. On peut faire le développement limité

$$y_n = \frac{1}{1 - G(x_n)} - \frac{1}{1 - x_n} = \frac{1}{1 - x_n}\left(\frac{1}{1 - \frac{\sigma^2(1-x_n)}{2} + o(1 - x_n)} - 1\right) = \frac{\sigma^2}{2} + o(1).$$

On en déduit que $\frac{1}{1-x_n} \sim \frac{n\sigma^2}{2}$, et donc que

$$\begin{aligned}\mathbb{P}[Y_n > 0] = 1 - \mathbb{P}[Y_n = 0] = 1 - (x_n)^M &= 1 - \left(1 - \frac{2}{n\sigma^2} + o\left(\frac{1}{n}\right)\right)^M \\ &= 1 - \left(1 - \frac{2M}{n\sigma^2} + o\left(\frac{1}{n}\right)\right) = \frac{2M}{n\sigma^2} + o\left(\frac{1}{n}\right),\end{aligned}$$

ce qui démontre le Théorème 2.

1.4 Puisque `Generations(n)` calcule la taille des générations jusqu'au temps n, on a déjà un programme vérifiant si un arbre de Galton-Watson survit ou non jusqu'au temps n. Une probabilité empirique est obtenue en répétant N expériences indépendantes, avec N grand :

```
ProbaSurvie(n,N):
    res = 0
    pour k entre 1 et N:
        calculer Y = Generations(n)
        ajouter 1 a res si Y[n] > 0
    retourner res/N
```

En taille $n = 20$ et avec $N = 1000$ essais (de loi poissonienne), on obtient la probabilité empirique de survie $\mathbb{P}[Y_{20} > 0] = 0.104$, très proche de la valeur asymptotique théorique $\frac{2}{n\sigma^2} = 0.1$.

2.1 Soit $\Omega_\infty \subset \Omega$ l'ensemble des arbres planaires infinis ; ceci veut dire que pour tout $n \in \mathbb{N}$, il existe $k(n) \geq 1$ noeuds de génération n qui ont une descendance. Donc, pour tout $n \geq 1$, l'observable $\xi_{n,1}$ (le nombre de fils de l'individu de n-ième génération le plus à gauche dans l'arbre) est strictement positive, à valeurs dans $\mathbb{N}^*$. Ainsi, on a une application bien définie

$$(\xi_{n,1})_{n\in\mathbb{N}} : \Omega_\infty \to (\mathbb{N}^*)^{\mathbb{N}}.$$

Cette application est surjective, car pour toute suite $(k_n)_{n\in\mathbb{N}}$ on peut par exemple considérer l'arbre avec $\xi_{n,1} = k_n$, et $\xi_{n,k\geq 2} = 0$ (à chaque génération, tout individu qui n'est pas le plus à gauche n'a pas de descendance). Or, l'ensemble des suites $(\mathbb{N}^*)^{\mathbb{N}}$ n'est pas dénombrable (il contient $\{1,2\}^{\mathbb{N}}$, qui est en bijection avec $[0,1]$ en utilisant le développement dyadique des réels). Donc, Ω_∞ n'est pas dénombrable, et a fortiori Ω n'est pas dénombrable.

En revanche, chaque ensemble Ω_n est dénombrable. En effet, l'ensemble $\mathbb{N}^{(\mathbb{N}^*)}$ des suites d'entiers presque nulles est contenu dans $\bigcup_{k=1}^{\infty}(\mathbb{N})^k$, qui est une union dénombrable d'ensembles dénombrables, donc dénombrable. Or, il y a une injection

$$\begin{aligned} \Omega_n &\to \left(\mathbb{N}^{(\mathbb{N})}\right)^n \\ \omega &\mapsto (\xi_{m,k}(\omega))_{m<n,\, k\geq 1} \end{aligned}$$

donc Ω_n est bien dénombrable.

2.2 La variable aléatoire Y_n est $\mathcal{F}_n$-mesurable, car elle peut être factorisée par l'application de troncation π_n (la valeur de $Y_n(\omega)$ ne dépend que des n premiers niveaux de l'arbre ω). On en déduit que τ est un temps d'arrêt, car pour tout n,

$$\{\tau = n\} = \left(\bigcap_{k=0}^{n-1}\{Y_k > 0\}\right) \cap \{Y_n = 0\}$$

est l'intersection d'événements qui sont dans des $\mathcal{F}_{k\leq n}$, donc est dans $\mathcal{F}_n$ puisque $(\mathcal{F}_n)_{n\in\mathbb{N}}$ est une filtration. Notons que τ est fini p.s., puisqu'on a démontré l'extinction presque sûre des arbres de Galton-Watson critiques ($\mathbb{E}[\xi] = 1$).

On a $\mathbb{E}[Y_0] = 1$ et $\mathbb{E}[Y_\tau] = 0$, donc le théorème d'arrêt ne s'applique pas. En effet, $(Y_n)_{n\in\mathbb{N}}$ est une martingale, mais aucune des hypothèses supplémentaires du théorème de Doob n'est vérifiée. Par exemple, τ n'est pas borné, ni même intégrable, car

$$\mathbb{E}[\tau] = \sum_{k=1}^{\infty} k\,\mathbb{P}[\tau = k] = \sum_{k=1}^{\infty} \mathbb{P}[\tau \geq k]$$

et cette série est divergente par le Théorème 2.

3.1 Si μ est la loi de Poisson de paramètre 1, alors pour $k \geq 1$

$$\mu^+(k) = k\,\mu(k) = \mathrm{e}^{-1}\,\frac{1}{(k-1)!} = \mu(k-1),$$

donc μ^+ est la loi de $1 + X$, où X suit une loi de Poisson de paramètre 1.

Pour encoder un arbre marqué, on peut rajouter dans les listes de listes de... un symbole spécial $*$ en première entrée d'un arbre avec racine marquée. Par exemple, $[*, [], [], [*, [*]]]$ est l'arbre marqué . Le programme `CalculGaltonWatsonPlus(n)` s'écrit alors :

```
CalculGaltonWatsonPlus(n):
    si n == 0:
        retourner la liste [*]
    sinon:
        xiplus = 1 + Poisson()
        res = [*]
        k = un entier aleatoire choisi dans [1,xiplus]
        pour i entre 1 et k-1:
            adjoindre CalculGaltonWatson(n-1) a res
        adjoindre CalculGaltonWatsonPlus(n-1) a res
        pour i entre k+1 et xiplus:
            adjoindre CalculGaltonWatson(n-1) a res
        retourner res
```

Le dessin d'un arbre marqué s'obtient presque de la même façon qu'avant : on le définit comme le quadruplet constitué d'une suite de segments, une suite de coordonnées de points marqués, une certaine largeur et une position pour la racine. En utilisant la structure récursive des arbres marqués, on écrit donc :

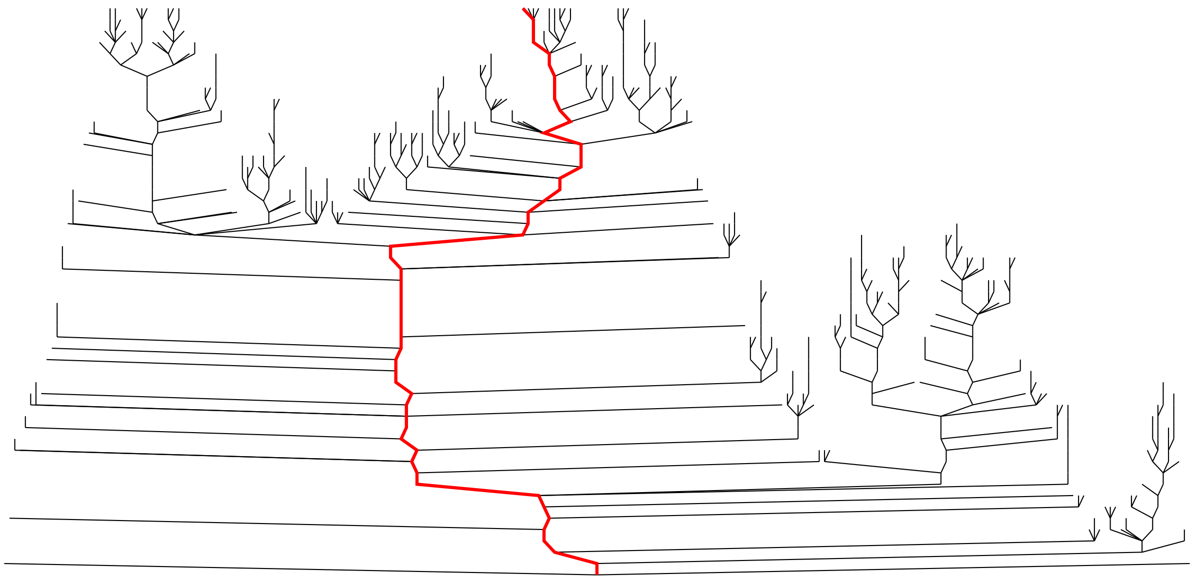
```
DessinMarque(arbre):
    si arbre == []:
        retourner [],[], 0, 0
    sinon:
        si la premiere coordonnee de arbre est *:
            k = (longueur de la liste arbre)-1 (nombre de sous-arbres)
            listedessins = []
            listepoints = []
            l = [] (la liste des largeurs)
            fils = [] (la liste des racines)
            pour tout sousarbre dans arbre, en ne comptant pas *:
                sousdessin,souspoint,souslargeur,sousracine
                = DessinMarque(sousarbre)
                ajouter sousdessin a listedessins,
                        souspoint a listepoints,
                        souslargeur a l,
                        sousracine a fils
            dessin = []
```

```
        marques = []
        racine = ((somme(l)+k-1)/2,0)
        ajouter racine a marques
        pour i entre 1 et k:
            decalage = (somme(l[j] pour j entre 1 et i-1) + i-1,1)
            prendre la i-ieme liste dans sousdessin
            decaler tous ses vecteurs ((v,w),(x,y)) de decalage
            ajouter ces vecteurs decales a dessin
            ajouter aussi le vecteur (racine,fils[i]+decalage)
            si listepoints[i] n'est pas vide:
                adjoindre (point+decalage) pour tout point
                dans listepoints[i]
        retourner dessin, marques, (somme(l)+k-1), (somme(l)+k-1)/2
    si la premiere coordonnee de arbre n'est pas *:
        retourner Dessin(arbre), et [] pour les points marques
```

Finalement, `GaltonWatsonPlus(n)` s'obtient simplement par :

```
GaltonWatsonPlus(n):
    dessin, marques, largeur, racine
    = DessinMarque(CalculGaltonWatsonPlus(n))
    tracer tous les segments de la liste dessin
    marquer tous les points de la liste marques
```

On a dessiné ci-après un exemple en taille $n = 50$:



Pour le programme `GenerationsPlus(n)`, il est préférable d'écrire un programme *ad hoc*, en remarquant que si $P_k \in [\![1, Y_k]\!]$ est la position du sommet marqué de k-ième génération, alors la loi conditionnelle de P_{k+1} est celle de

$$\xi_1 + \xi_2 + \cdots + \xi_{P_k - 1} + U(\xi^+),$$

où les ξ_i sont des variables indépendantes de loi μ, ξ^+ est de loi μ^+, et $U(\xi^+)$ est obtenue en choisissant uniformément un entier dans $[\![1, \xi^+]\!]$. Ainsi, le programme suivant donne les tailles des n premières générations sous $\mathbb{GW}^+$, ainsi que les positions des sommets marqués :

```
GenerationsPlus(n):
    generations = []
    positions = []
    pos = 1
    gen = 1
    pour k entre 1 et n:
        descendance = liste constituee de gen variables
                      independantes Poisson()
        ajouter 1 au pos-ieme element de descendance
        gen = somme(descendance)
        adjoindre gen a generations
        pos = somme(descendance[i] pour i entre 1 et pos-1)
              + Uniform(1,descendance[pos])
        adjoindre pos a positions
    retourner generations et positions
```

On obtient par exemple les deux listes

$$[3, 6, 6, 11, 7, 14, 18, 15, 19, 17] \quad \text{et} \quad [1, 2, 2, 3, 1, 3, 4, 2, 2, 3]$$

en taille $n = 10$.

3.2 Soit (T_n, p_n) un arbre avec feuille marquée de taille n. Soit $k \geq 1$ le nombre de fils de la racine, (T_{n-1}, q_{n-1}) la descendance issue de l'unique fils marqué $i \in [\![1, k]\!]$, et $U_{n-1}^{(2)}, \ldots, U_{n-1}^{(k)}$ les descendances des autres fils. Par définition de $\mathbb{GW}^+$, pour obtenir comme n premiers niveaux (T_n, p_n), il faut :

(a) que la racine ait k fils, ce qui arrive avec probabilité $\mu^+(k) = k\,\mu(k)$.

(b) que le fils de la racine qui est marqué soit i, ce qui arrive avec probabilité $\frac{1}{k}$ conditionnellement au précédent choix.

(c) que les descendances respectives soient (T_{n-1}, q_{n-1}) et $U_{n-1}^{(2)}, \ldots, U_{n-1}^{(k)}$, ce qui se produit avec probabilité

$$\mathbb{GW}^+([T_{n-1}, q_{n-1}]) \prod_{j=2}^{k} \mathbb{GW}([U_{n-1}^{(j)}])$$

puisque la descendance du fils marqué suit la loi $\mathbb{GW}^+$, et les autres descendances suivent des lois $\mathbb{GW}$ indépendantes.

On en déduit :

$$\mathbb{GW}^+([T_n, p_n]) = \mu(k)\,\mathbb{GW}^+([T_{n-1}, q_{n-1}]) \prod_{j=2}^{k} \mathbb{GW}([U_{n-1}^{(j)}])$$

puisque $\frac{\mu^+(k)}{k} = \mu(k)$. Supposons démontré au rang $n-1$ que $\mathbb{GW}^+([T_{n-1}, q_{n-1}]) = \mathbb{GW}([T_{n-1}])$. Alors,

$$\mathbb{GW}^+([T_n, p_n]) = \mu(k)\,\mathbb{GW}([T_{n-1}]) \prod_{j=2}^{k} \mathbb{GW}([U_{n-1}^{(j)}]) = \mathbb{GW}([T_n]),$$

donc le résultat est aussi vrai au rang n.

3.3 Si $\xi \sim \mathcal{P}(1)$, alors sa fonction génératrice est $\mathbb{E}[s^\xi] = \sum_{k=0}^\infty \mathrm{e}^{-1}\frac{s^k}{k!} = \mathrm{e}^{s-1}$. De plus, on peut calculer $\mathbb{E}_{\mathbb{GW}^+}[s^{Y_n}]$ par conditionnement, en utilisant le fait que μ^+ est la loi de $\xi+1$ avec $\xi \sim \mathcal{P}(1)$ (dans ce qui suit, toutes les espérances sont relativement à $\mathbb{GW}^+$) :

$$\mathbb{E}[s^{Y_n}] = \mathbb{E}\big[\mathbb{E}[s^{Y_n}|Y_{n-1}]\big] = \mathbb{E}\Big[\mathbb{E}[s^{1+\sum_{i=1}^{Y_{n-1}}\xi_i}|Y_{n-1}]\Big] = s\,\mathbb{E}[(G(s))^{Y_{n-1}}].$$

On obtient alors :

$$\begin{aligned}\mathbb{E}[s^{Y_0}] &= s;\\ \mathbb{E}[s^{Y_1}] &= s\,\mathbb{E}[(G(s))^{Y_0}] = s\,G(s);\\ \mathbb{E}[s^{Y_2}] &= s\,\mathbb{E}[(G(s))^{Y_1}] = s\,G(s)\,G^{\circ 2}(s),\end{aligned}$$

et par récurrence, $\mathbb{E}[s^{Y_n}] = s\,G(s)\,G^{\circ 2}(s)\,\cdots\,G^{\circ n}(s)$.

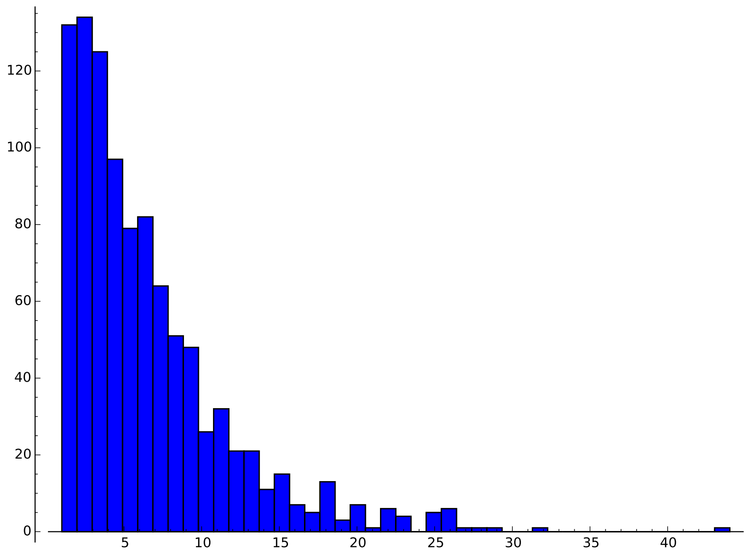
3.4 On décompose les probabilités suivant la valeur de k :

$$\begin{aligned}\mathbb{GW}^+(B_n) &= \sum_{k=1}^\infty \mathbb{GW}^+(B_n \text{ et } Y_n = k) = \sum_{k=1}^\infty \frac{\mathbb{GW}^+(Y_n = k)}{k}\\ &= \sum_{k=1}^\infty \mathbb{GW}(Y_n = k) = \mathbb{GW}(Y_n > 0) = \mathbb{GW}(A_n).\end{aligned}$$

4.1 Le programme `GenerationsPlus(n)` permet de vérifier facilement la réalisation de B_n : c'est le cas si et seulement si la position P_n vaut $P_n = 1$. Par rejet, on peut donc simuler Y_n sous la loi conditionnelle $\mathbb{GW}^+(\,\cdot\,|B_n)$, qui est la même que sous $\mathbb{GW}(\,\cdot\,|A_n)$:

```
Yconditionnel(n):
    tirer des variables GenerationsPlus(n), jusqu'a obtenir
    un resultat avec la n-ieme position egale a 1
    retourner la taille de la n-ieme generation
```

On vérifie alors une convergence en loi en traçant un histogramme des valeurs de $\frac{Y_n}{n}$, avec un grand nombre N d'essais. On a dressé ci-dessous cet histogramme lorsque $n = 10$ et $N = 1000$:



On observe clairement une distribution exponentielle, et la moyenne empirique 6.168 est proche de la valeur théorique $\frac{n\sigma^2}{2} = 5$.

4.2 Pour commencer, notons que pour tout n et tout $j \le n-1$,

$$\mathbb{E}_{\mathbb{GW}^+}[Y_{n,j}|B_n] = \mathbb{E}_{\mathbb{GW}^+}[Y_{n,j}|B_{n,j}].$$

En effet, la variable $Y_{n,j}$ et l'événement $B_{n,j}$ sont mesurables vis-à-vis de la tribu engendrée par les descendances des enfants de p_j qui sont différents de p_{j+1}. Par conséquent, les événements $B_{n,j}$ sont indépendants, et $Y_{n,j}$ est indépendante de $B_{n,j'}$, sauf si $j = j'$. On peut alors écrire :

$$\begin{aligned}\mathbb{P}[Y_{n,j} = k|B_n] &= \frac{\mathbb{P}[Y_{n,j} = k \text{ et } B_n]}{\mathbb{P}[B_n]} = \frac{\mathbb{P}[Y_{n,j} = k \text{ et } B_{n,j}]\,\mathbb{P}[\cap_{j'\neq j} B_{n,j'}]}{\mathbb{P}[B_{n,j}]\,\mathbb{P}[\cap_{j'\neq j} B_{n,j'}]}\\ &= \frac{\mathbb{P}[Y_{n,j} = k \text{ et } B_{n,j}]}{\mathbb{P}[B_{n,j}]} = \mathbb{P}[Y_{n,j} = k|B_{n,j}],\end{aligned}$$

d'où par sommation l'identité des espérances conditionnelles. Alors,

$$\mathbb{E}_{\mathbb{GW}^+}[Y_n|B_n] = \mathbb{E}_{\mathbb{GW}^+}\left[\left(1 + \sum_{j=0}^{n-1} Y_{n,j}\right)\Big|B_n\right] = 1 + \sum_{j=0}^{n-1} \mathbb{E}_{\mathbb{GW}^+}[Y_{n,j}|B_{n,j}].$$

Ensuite, $\mathbb{E}_{\mathbb{GW}^+}[Y_{n,j}|B_{n,j}] = \frac{\mathbb{E}_{\mathbb{GW}^+}[Y_{n,j}\,1_{B_{n,j}}]}{\mathbb{GW}^+(B_{n,j})} = \frac{\mathbb{E}_{\mathbb{GW}^+}[D_{n,j}\,1_{B_{n,j}}]}{\mathbb{GW}^+(B_{n,j})}$. On calcule $\mathbb{E}_{\mathbb{GW}^+}[D_{n,j}\,1_{B_{n,j}}]$ en conditionnant par la valeur de ξ_j^+, et par la position i de p_{j+1} parmi les ξ_j^+ enfants de p_j :

$$\begin{aligned}\mathbb{E}_{\mathbb{GW}^+}[D_{n,j}\,1_{B_{n,j}}] &= \sum_{k=1}^{\infty} \frac{\mu^+(k)}{k} \sum_{i=1}^{k} \begin{array}{l}\mathbb{E}_{\mathbb{GW}^+}[D_{n,j}|\text{il y a } k-i \text{ enfants de } p_j \text{ à droite de } p_{j+1}]\\ \times\mathbb{GW}^+(1_{B_{n,j}}|\text{il y a } i-1 \text{ enfants de } p_j \text{ à gauche de } p_{j+1})\end{array}\\ &= \sum_{k=1}^{\infty} \mu(k) \sum_{i=1}^{k} (k-i)\,\mathbb{GW}(Y_{n-j-1} = 0)^{i-1}\\ &= \sum_{k=1}^{\infty} \mu(k)\, \frac{k\,(1 - \mathbb{GW}(Y_{n-j-1} = 0)) - (1 - \mathbb{GW}(Y_{n-j-1} = 0)^k)}{(1 - \mathbb{GW}(Y_{n-j-1} = 0))^2}\\ &= \frac{G(\mathbb{GW}(Y_{n-j-1} = 0)) - \mathbb{GW}(Y_{n-j-1} = 0)}{(1 - \mathbb{GW}(Y_{n-j-1} = 0))^2}\\ &= \frac{\mathbb{GW}(Y_{n-j} = 0) - \mathbb{GW}(Y_{n-j-1} = 0)}{(1 - \mathbb{GW}(Y_{n-j-1} = 0))^2}.\end{aligned}$$

On calcule $\mathbb{P}[1_{B_{n,j}}]$ avec le même conditionnement :

$$\begin{aligned}\mathbb{GW}_+(1_{B_{n,j}}) &= \sum_{k=1}^{\infty} \frac{\mu^+(k)}{k} \sum_{i=1}^{k} \mathbb{GW}^+(1_{B_{n,j}}|\text{il y a } i-1 \text{ enfants de } p_j \text{ à gauche de } p_{j+1})\\ &= \sum_{k=1}^{\infty} \mu(k) \sum_{i=1}^{k} \mathbb{GW}(Y_{n-j-1} = 0)^{i-1}\\ &= \sum_{k=1}^{\infty} \mu(k) \frac{1 - \mathbb{GW}(Y_{n-j-1} = 0)^k}{1 - \mathbb{GW}(Y_{n-j-1} = 0)}\\ &= \frac{1 - G(\mathbb{GW}(Y_{n-j-1} = 0))}{1 - \mathbb{GW}(Y_{n-j-1} = 0)} = \frac{1 - \mathbb{GW}(Y_{n-j} = 0)}{1 - \mathbb{GW}(Y_{n-j-1} = 0)}.\end{aligned}$$

On conclut que :

$$\mathbb{GW}^+[Y_n|B_n] = 1 + \sum_{j=0}^{n-1} \frac{\mathbb{GW}(Y_{n-j}=0) - \mathbb{GW}(Y_{n-j-1}=0)}{\mathbb{GW}(Y_{n-j-1}>0)\,\mathbb{GW}(Y_{n-j}>0)}$$

$$= 1 + \sum_{j=0}^{n-1} \frac{1}{\mathbb{GW}(Y_{n-j}>0)} - \frac{1}{\mathbb{GW}(Y_{n-j-1}>0)} = \frac{1}{\mathbb{GW}(Y_n>0)}.$$

4.3 Comme $(Y_n)_{n\in\mathbb{N}}$ est une martingale sous $\mathbb{GW}$, on peut écrire, toutes les espérances et probabilités s'entendant sous la mesure $\mathbb{GW}$:

$$1 = \mathbb{E}[Y_0] = \mathbb{E}[Y_n] = \mathbb{E}[Y_n\, 1_{Y_n>0}] = \mathbb{E}[Y_n|A_n]\,\mathbb{P}[A_n].$$

On en déduit immédiatement :

$$\mathbb{E}[Y_n|A_n] = \frac{1}{\mathbb{P}[A_n]} \sim \frac{n\sigma^2}{2}$$

par le Théorème 2.

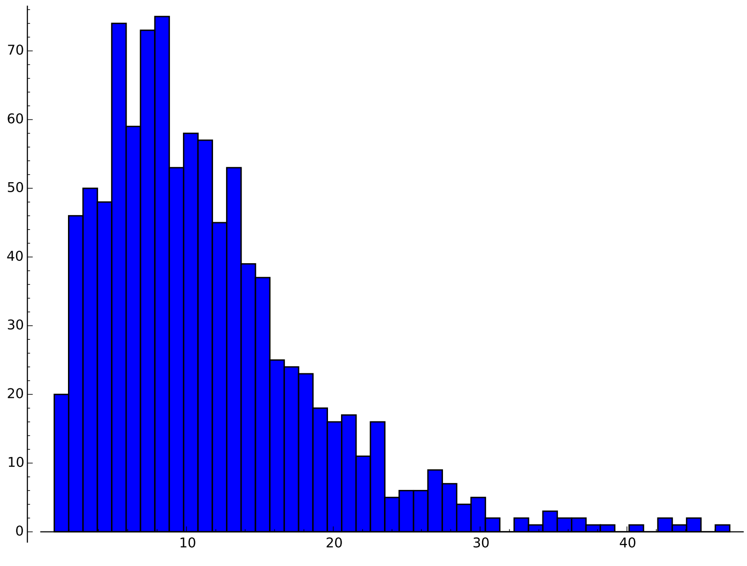
4.4 Conditionner à B_n revient à "oublier" les sommets G_n à gauche de p_n dans le décompte de Y_n. Cette observation est justifiée par les asymptotiques

$$\mathbb{E}_{\mathbb{GW}^+}[Y_n|B_n] \sim \mathbb{E}_{\mathbb{GW}^+}[D_n] \sim \frac{n\sigma^2}{2}.$$

Donc, on peut s'attendre à récupérer la distribution limite de Y_n sans conditionnement en prenant deux copies indépendantes de la distribution limite de Y_n sous $\mathbb{GW}^+(\,\cdot\,|B_n)$. Ainsi, on peut raisonnablement conjecturer sous $\mathbb{GW}^+$:

$$\lim_{n\to\infty} \frac{Y_n}{n} = \mathcal{E}\left(\frac{2}{\sigma^2}\right) + \mathcal{E}\left(\frac{2}{\sigma^2}\right) = 1_{(x>0)} \left(\frac{2}{\sigma^2}\right)^2 x\,\mathrm{e}^{-\frac{2x}{\sigma^2}}\,dx$$

qui est une loi Γ, d'espérance σ^2. Avec `GenerationsPlus(n)`, on simule aisément un grand nombre N de copies de Y_n sous $\mathbb{GW}^+$, ce qui donne pour $n = 10$ et $N = 1000$ l'histogramme



qui confirme bien notre conjecture.