Générateurs de nombres aléatoires

Un générateur de nombre aléatoire est une procédure permettant de reproduire le hasard. Plus précisément, on souhaite construire un programme qui génère des nombres entre 0 et 1 selon la loi uniforme.

Le modèle que nous allons étudier ici est celui des générateurs congruentiels linéaires. Ce sont des suites définies pour des entiers a, c et n par

$$u_0 \leq n, \quad \text{et} \quad \forall k \geq 0, \quad u_{k+1} = au_k + c \mod n.$$

La suite des $\frac{u_k}{n}$ est alors une suite de nombres dans $[0,1]$. L'objectif est que cette suite ressemble à une suite aléatoire de nombres tirés indépendamment selon la loi uniforme sur $[0,1]$. Nous allons tester statistiquement si cela est satisfait en pratique.

Nous étudierons trois générateurs de nombres aléatoires :
— Le générateur Randu développé par IBM dans les années 60 défini par la suite récurrente $u_{k+1} = 65539u_k \mod 2^{31}$ avec u_0 impair.
— Le générateur de Turbo Pascal : $u_{k+1} = 129u_k + 907633385 \mod 2^{32}$.
— Le générateur d'Octave donné par $rand()$. Il est défini à partir du twister de Mersenne qui est un générateurs congruentiel non linéaire.

Donner deux raisons pour lesquelles ces suites ne peuvent pas définir parfaitement des tirages aléatoires selon la loi uniforme sur $[0,1]$.

Nous allons néanmoins étudier si les suites ainsi générées ont l'allure de suites tirées selon la loi uniforme.

Pour chaque générateur, effecteur des tests de moyenne, de variance, et d'adéquation à la loi uniforme. En faire un certain nombre en jouant sur la taille des échantillons testés.

Les nombres doivent de plus correspondre à des tirages deux à deux indépendants. Faire un test d'indépendance portant sur les tirages successifs. Adapter le test d'indépendance étudié en cours pour tester l'indépendance de trois tirages successifs.