

Travaux Pratiques N°5

Simulation et analyse de signaux aléatoires

On veut analyser les propriétés statistiques de mesures lidar Raman. Un lidar est composé d'un laser émettant des impulsions d'énergie E dans l'atmosphère. A chaque altitude z , une fraction de l'énergie est diffusée par les gaz et particules. Une fraction revient vers le lidar où un télescope de diamètre D collecte le flux optique. Ce flux optique est transmis vers des photo-détecteurs où il est converti en tension électrique. Chaque photon produit une petite impulsion en tension. On considère ici un signal Raman diffusé par les molécules de vapeur d'eau. Le flux de photons est très faible et le signal est analysé en "mode comptage de photons", c'est-à-dire qu'on compte les impulsions électriques arrivant dans des portes temporelles de $\Delta t = 50$ ns sur une durée $T_e = N \Delta t$. Chaque instant d'arrivée t du signal correspond à une couche différente de l'atmosphère sondée : $z = c t$, où c est la vitesse de la lumière. On compare les propriétés de ces signaux à celle d'un bruit blanc.

1) Simulation et analyse d'un bruit blanc

- a. Générer une réalisation d'un bruit blanc discret, gaussien, de variance $\sigma_b^2=1$, sur $N = 512$ points. Tracer ce signal ainsi que sa fonction d'auto-corrélation et sa densité spectrale de puissance.
- b. Vérifier qu'il est stationnaire et ergodique (on analysera la moyenne et l'écart-type sur une ou plusieurs réalisations).
- c. Tracer le signal moyenné sur $M = 20$ réalisations du bruit blanc et discuter ses propriétés (moyenne, écart-type).
- d. Estimer la fonction de corrélation et la DSP du processus en faisant la moyenne des corrélations et des modules carré de la FFT sur $M = 20$ réalisations du bruit blanc. Vérifier que l'on retrouve bien les propriétés attendues.

2) Lecture et analyse des données lidar

Le fichier 'eau.dat' contient des mesures d'intensité lidar organisées en matrice de taille $NZ \times NT$, où $NZ = 300$ est le nombre de portes temporelles d'un « profil lidar » et $NT = 357$ le nombre de profils observés avec un intervalle de temps régulier $\Delta\tau = 20$ s. Il s'agit donc d'un signal à deux dimensions, fonction de l'altitude $z = nz c \Delta t$, où $nz = 0..NZ-1$, et du temps $\tau = nt \Delta\tau$, où $nt = 0..NT-1$.

- a. Ecrivez un programme Scilab qui permet de lire le fichier et de tracer les graphes suivants : le signal en fonction de z , pour τ donné et le signal en fonction de τ , pour z donné. Tracez des exemples de profils.
- b. Évaluez la moyenne et l'écart-type du signal pour chaque altitude (statistiques sur l'ensemble des NT profils) et représentez-les en fonction de z . Représentez l'histogramme du signal pour plusieurs altitudes.
- c. Analysez pour une altitude z donnée si la série temporelle est stationnaire et ergodique. Pour cela, on découpera le vecteur de taille NT en P vecteurs de taille NT/P et on évaluera les moyennes et écart-types de ces vecteurs. On prendra par exemple $P = 4$.
- d. Représentez les fonctions d'autocorrélation temporelle du signal pour différentes altitudes et vérifiez que vous retrouvez la variance lorsque le décalage est nul.
- e. Comparez les allures des fonctions d'autocorrélation pour différentes altitudes (pour cela, il faut centrer et normaliser les signaux) et déterminez graphiquement le temps caractéristique de corrélation.

3) Filtrage

- a. Générer un bruit blanc que l'on filtrera par moyenne glissante, en appliquant une convolution par un filtre numérique moyenné de durée $C=5$, défini par la transmittance $\{h_n\}$, avec $h_n = 1/\sqrt{C}$ pour $1 \leq n \leq C$ et 0 sinon (NB : le 1^{er} indice des tableaux est 1). Tracer ce signal ainsi que sa fonction de corrélation et sa densité spectrale de puissance.
- b. Calculer la moyenne et l'écart-type du signal.

- c. Estimer la fonction de corrélation et la DSP du processus en faisant des moyennes sur $M=20$ réalisations. Vérifier que l'on retrouve bien les propriétés attendues (cf. TD 6).

NB : utilisez les fonctions intégrées de Scilab : `mean()`, `stdev()`, `corr()`, `fft()` et `convol()`.