

# SFN : UN LOGICIEL D'E.A.O. DE TRAITEMENT DU SIGNAL

G.JACQUET et J.THIEL,

Université d'Orléans,

U.F.R. Faculté des Sciences, B.P.6759, 45067 ORLEANS CEDEX 02.

## RESUME.

Ecrit en Turbo Pascal 5.0, SFN (Signaux et Filtres Numériques) est un logiciel **intégré** d'enseignement assisté par ordinateur du traitement numérique du signal. Structuré à partir de critères pédagogiques, il peut être utilisé pour une initiation concrète, mais complète, aux techniques d'analyse des signaux stationnaires et des filtres linéaires ainsi qu'à la synthèse de ces filtres. Incluant des algorithmes parmi les plus récents, il permet aussi à des utilisateurs avertis de tester simplement et rapidement des méthodes de filtrage.

## 1. INTRODUCTION.

A finalité pédagogique, le logiciel SFN (Signaux et Filtres Numériques) est conçu pour l'enseignement de certaines techniques du filtrage numérique sans avoir recours à une étude théorique préalable approfondie. Ces techniques concernent :

**L'analyse des signaux stationnaires.** Estimation des caractéristiques statistiques des signaux avant et après filtrage : valeurs moyennes, fonctions de répartition et densités de probabilité marginales et conjointe, fonctions d'auto et d'intercorrélation, auto et interspectres.

**La synthèse des filtres linéaires.** Trois méthodes sont disponibles pour créer des filtres numériques à la demande : Remez et série de Fourier pour les filtres non récursifs, fonctions modèles pour les filtres récursifs.

**L'analyse des filtres linéaires.** Réponses impulsionnelle et indicielle, représentations de Bode, Black et Nyquist, retards de phase et de propagation, constellation des pôles et des zéros, représentations tridimensionnelles de la fonction de transfert en  $z$ .

SFN est approprié pour un **apprentissage inductif** qui débute par la mise en oeuvre d'un filtre avec visualisation immédiate de ses caractéristiques et des résultats du filtrage, se poursuit par l'examen de l'influence des paramètres du filtre et s'achève par l'étude des algorithmes utilisés. Cette démarche, pratique et progressive, a pour but d'amener des néophytes à un niveau tel qu'ils puissent utiliser sans complexe des logiciels de filtrage numérique plus sophistiqués. De ce fait, SFN peut également être utile à des ingénieurs non spécialistes pour définir rapidement un filtre. D'autre part, **ses procédures peuvent être utilisées séparément** par des chercheurs et enseignants ayant de bonnes connaissances en traitement du signal. Il peut donc constituer une utile introduction à l'utilisation de bibliothèques de logiciels plus complets, existantes [1] ou en projet [2]. Contrairement à celles-ci, SFN est écrit en Pascal (avec Turbo Pascal 5.0)

## ABSTRACT.

Written in Turbo Pascal 5.0, SFN (Numerical Signals and Filters) is an integrated software package devoted to signal processing teaching. Structured from pedagogical criterions, it can be used for concrete and complete initiation for stationary signals and linear filters analysis as well as for filters design techniques. Enclosing algorithms among the most recent, SFN also allows experienced users to simply and quickly test methods of filtering.

afin de faciliter le suivi des algorithmes par des étudiants.

## 2. ORGANISATION DE SFN.

SFN fonctionne sur compatible P.C. et, afin de limiter l'encombrement du fichier exécutable en mémoire centrale, il est partitionné en unités compilées qui sont enregistrées dans un fichier de recouvrement. Ses entrées-sorties et sa structure interne ont été organisées pour faciliter son utilisation et sa compréhension, tant par le néophyte que par le spécialiste.

Pour obtenir une grande souplesse d'utilisation, les options suivantes ont été retenues :

- utilisation permanente de **menus déroulants** pour une interactivité maximum,
- représentation systématique des résultats sous **forme graphique** bi ou tridimensionnelle,
- programme d'**aide** résident en mémoire et option de suivi **pas à pas** des algorithmes,
- mise à disposition de **banques de signaux tests et de filtres types**,
- intégration possible de **filtres spécifiques** par saisie des coefficients ou par synthèse,
- possibilité d'analyser et/ou de filtrer des **signaux définis par l'utilisateur**.

Ces options sont intégrées dans la structure modulaire schématisée sur la figure 1. Aussi bien pour les signaux que pour les filtres, une unité de lecture et de création permet de constituer un fichier utilisateur par lecture dans une banque, saisie directe de valeurs ou encore par synthèse. Le filtre et le signal sélectionnés dans ces fichiers utilisateur, ainsi que le signal filtré, peuvent être analysés dans deux unités distinctes, une troisième unité étant dédiée à l'analyse conjointe des deux signaux avant et après filtrage.

Cette structure de base est complétée par des passerelles entre unités pour activer à tout moment un module d'analyse ou de sélection afin de faciliter les comparaisons entre plusieurs signaux ou filtres. D'autres

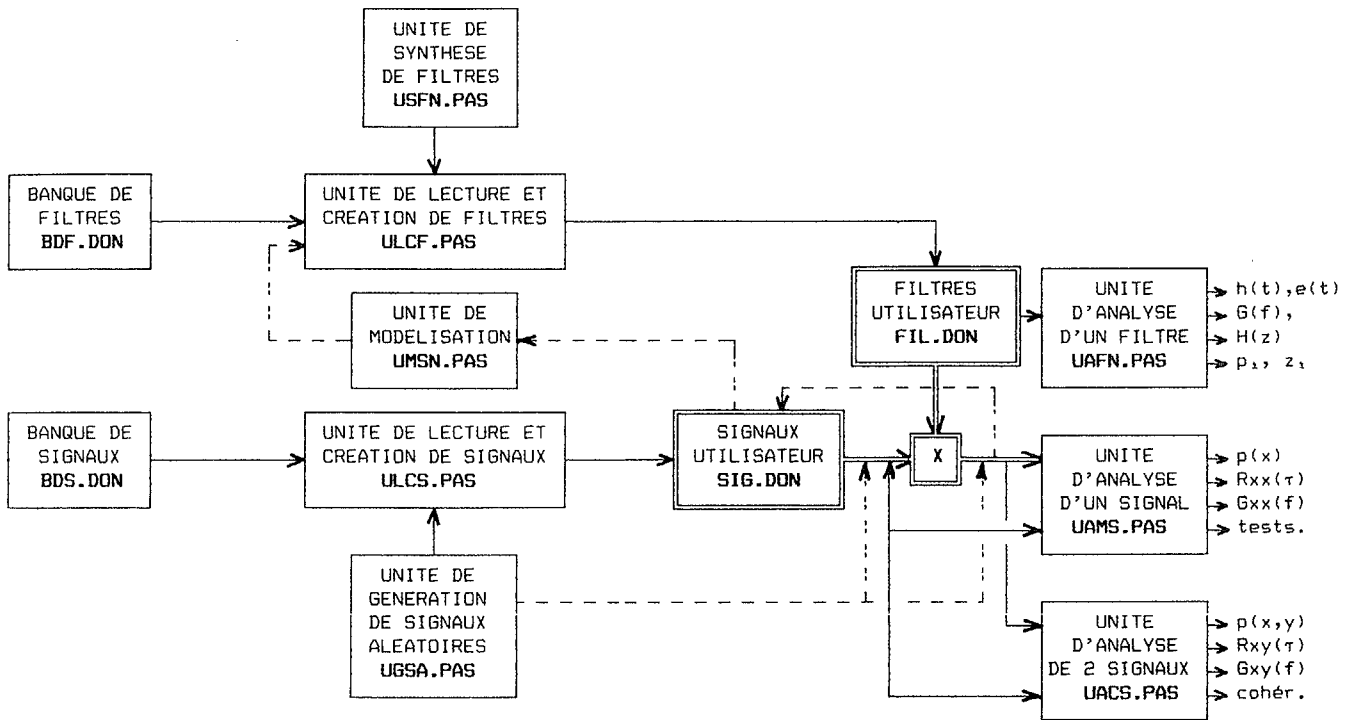


Figure 1. Structure de SFN.

passerelles, représentées en traits pointillés sur la figure 1, permettent de calculer les coefficients d'un filtre par modélisation d'un signal, d'ajouter du bruit à l'entrée ou à la sortie du filtre, ou encore de réaliser plusieurs filtrages successifs d'un même signal.

### 3. GENERATION DE SIGNAUX.

Pour créer un signal par synthèse, l'utilisateur peut ajouter, multiplier ou convoluer les signaux "opérandes" suivants :

- signaux élémentaires : sinusoïde, rampe, impulsion, exponentielle, peigne de Dirac,
- séquence pseudo-aléatoire stationnaire de densité de probabilité fixée,
- signal créé par saisie directe des valeurs à l'écran ou par échantillonnage d'une fonction analytique,
- signal de la banque de signaux BDS.DON ou du fichier utilisateur SIG.DON.

Ces opérations sont réalisées interactivement en séquence, avec visualisation graphique des résultats intermédiaires, et le signal résultant de ces multiples combinaisons est enregistré dans SIG.DON.

A l'unité ULCS de lecture et création d'un signal est attachée une unité UGSA de génération de signaux aléatoires pour simuler du bruit. Pour créer une séquence à répartition uniforme, six générateurs de nombres pseudo-aléatoires sont disponibles, et notamment GFSR de Lewis et Payne [3]. Les sorties de ces générateurs peuvent ensuite être combinées pour obtenir un signal à densité de probabilité voulue, gaussienne par exemple. Enfin, UGSA permet de modifier ce signal pseudo-aléatoire pour obtenir une densité spectrale de puissance de forme donnée. On peut ainsi bruyter à la demande tout signal déterministe, stationnaire ou non.

La figure 2 reproduit le menu de création d'un signal par synthèse, étant entendu que SFN ne permettra d'analyser que les seuls signaux stationnaires.

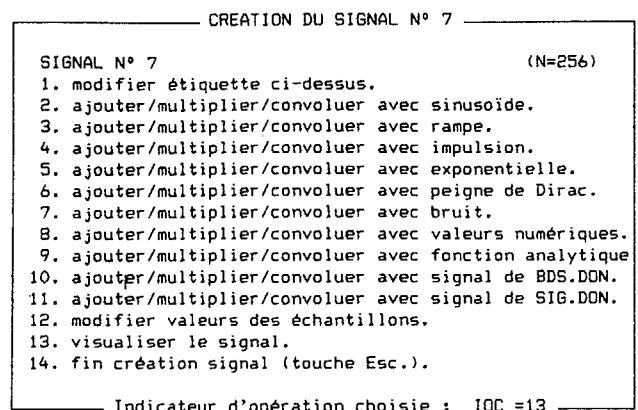


Figure 2. Menu de création d'un signal par synthèse.

### 4. ANALYSE DE SIGNAUX STATIONNAIRES.

Les signaux d'entrée et de sortie du filtre sont analysables séparément ou conjointement, par visualisation de leurs caractéristiques temporelles, fréquentielles et statistiques. La sortie d'une unité d'analyse marginale (UAMS) ou conjointe (UACS) est en effet une page graphique sélectionnée par un menu (reproduit sur la figure 3) qui permet aussi

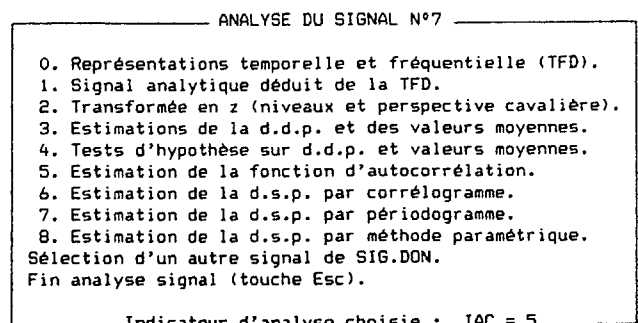


Figure 3. Menu d'analyse marginale d'un signal.

de changer de signal. Chaque page comprend deux ou trois courbes ou surfaces pour mieux comparer des résultats complémentaires tout en gardant une lisibilité suffisante.

Les habituelles représentations temporelle et fréquentielles (par T.F.D.) d'un signal sont complétées par une représentation de l'amplitude de sa transformée en z par perspective cavalière et courbes de niveau. Dans l'analyse statistique marginale à l'ordre 1, les calculs d'estimation des valeurs moyennes sont prolongés par quelques tests d'hypothèse élémentaires. Les estimations de la fonction de répartition et de la densité de probabilité donnent lieu à des histogrammes cumulés et simples, avec test de normalité. L'analyse conjointe de deux signaux est illustrée par une représentation tridimensionnelle de la densité de probabilité conjointe estimée, comme illustré sur la figure 4.

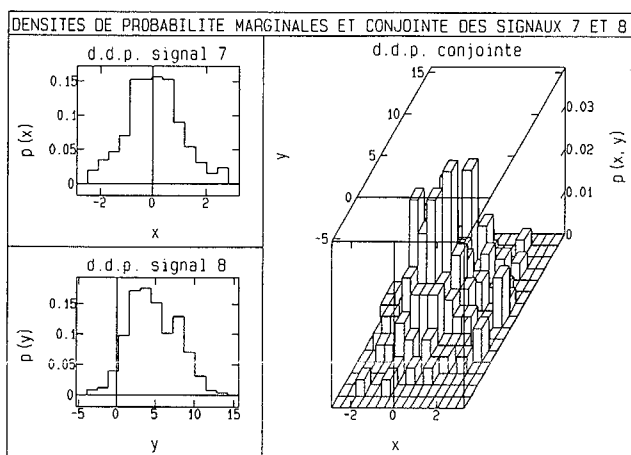


Figure 4. Densités de probabilité marginales et conjointe de 2 signaux.

Pour estimer la corrélation et la densité spectrale de puissance par méthode directe, l'utilisateur peut choisir entre un calcul global ou par sections (avec ou sans recouvrements) et dispose d'une bibliothèque complète de fenêtres. Plusieurs méthodes paramétriques d'analyse spectrale sont mises en oeuvre, l'objectif étant de comparer ces méthodes comme l'ont fait Kay et Marple [4]. Pour atteindre ce but, plusieurs algorithmes rapides proposés par Marple [1] ont été traduits en Pascal. Finalement, ces estimations d'auto et d'interspectres peuvent être complétées par celle de la fonction de cohérence entre sortie et entrée d'un système. Pour simuler celui-ci, il est possible d'injecter du bruit à l'entrée et/ou à la sortie du filtre, directement à partir de l'unité UGSA déjà présentée (voir Fig. 1).

### 5. CREATION DE FILTRES LINEAIRES.

L'unité de lecture et création de filtres, ULCF, a même structure de base que ULCS pour les signaux ; mais il n'est pas possible de combiner plusieurs filtres alors que, d'un autre côté, les opérations plus complexes de création de filtres par modélisation d'un signal ou par synthèse sont réalisées dans des unités séparées, UMSN et USFN, respectivement. La méthode de création d'un filtre est choisie par menu, puis les coefficients calculés sont représentés graphiquement avant enregistrement dans le fichier FIL.DON.

Pour créer un filtre linéaire, causal ou non, l'utilisateur peut tout d'abord directement en saisissant les coefficients à l'écran, ou

générer ces coefficients par échantillonnage d'une fonction analytique. Il peut également lire un enregistrement de la banque BDF.DON qui comprend notamment les filtres les plus classiques de lissage glissant et de différentiation [5]. Il peut enfin créer un filtre par modélisation d'un signal ou par synthèse.

Représenter un signal par la sortie d'un filtre AR, MA ou ARMA excité par du bruit blanc ramène évidemment à l'analyse spectrale paramétrique. De fait, le calcul des coefficients du filtre de rang donné est effectué dans un module commun aux trois unités UMSN, UAMS et UACS.

Pour créer un filtre non récursif par synthèse, deux méthodes classiques sont implantées dans SFN. La synthèse par transformée de Fourier d'une réponse en fréquence délimitée par un gabarit est faite en utilisant la même banque de fenêtres que pour l'analyse spectrale par périodogramme. Quant à la méthode Min-Max de Remez, une traduction en Pascal du programme de McClellan et coll. [6] a été réalisée. Enfin, pour créer un filtre récursif par synthèse, le calcul direct des coefficients par les fonctions modèles SFN a été programmé en suivant les indications de [7].

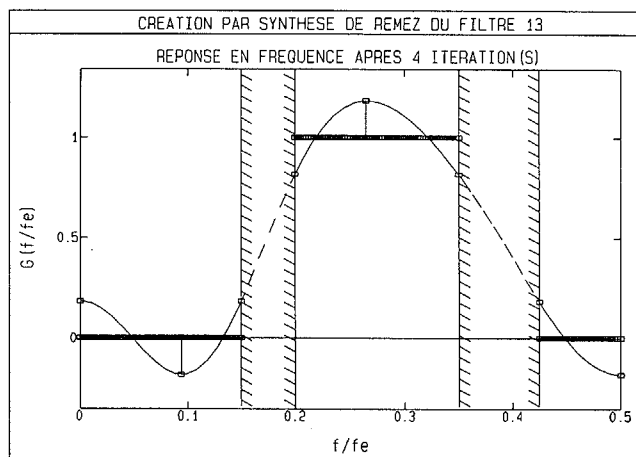
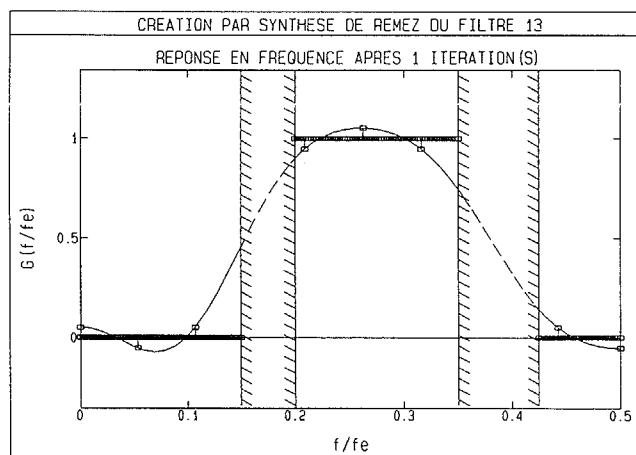


Figure 5. procédure itérative de Remez en mode pas à pas

Pour ces trois méthodes, la saisie des caractéristiques du filtre (gabarit, fréquences de coupure, ...) est conversationnelle avec utilisation de graphiques. Afin de faciliter la compréhension des algorithmes de synthèse, le déroulement des opérations peut être suivi en mode pas à pas, comme illustré sur la figure 5.



## 6. ANALYSE DE FILTRES.

Un filtre numérique du fichier FIL.DON peut être complètement caractérisé par diverses méthodes d'analyse avec représentation graphique adaptée bi et tridimensionnelle des résultats pour en faciliter l'interprétation.

Les réponses temporelles (impulsionnelle et indicielle), fréquentielles (échelles linéaires, diagrammes de Bode, Black et Nyquist), les retards de phase et de propagation sont représentés par des graphes simples. Les pôles et les zéros de la fonction de transfert sont calculés puis positionnés dans le plan complexe-z, ce qui permet de relier la position de ces singularités avec les autres caractéristiques du filtre (stabilité, zéros de transmission ...). Enfin la visualisation de l'amplitude de la fonction de transfert en z, par perspective cavalière et courbes de niveau, complète l'information fournie par la constellation des pôles et des zéros.

La figure ci-dessous montre la réponse en fréquence (en échelle linéaire) et la transformée en z d'un filtre récuratif.

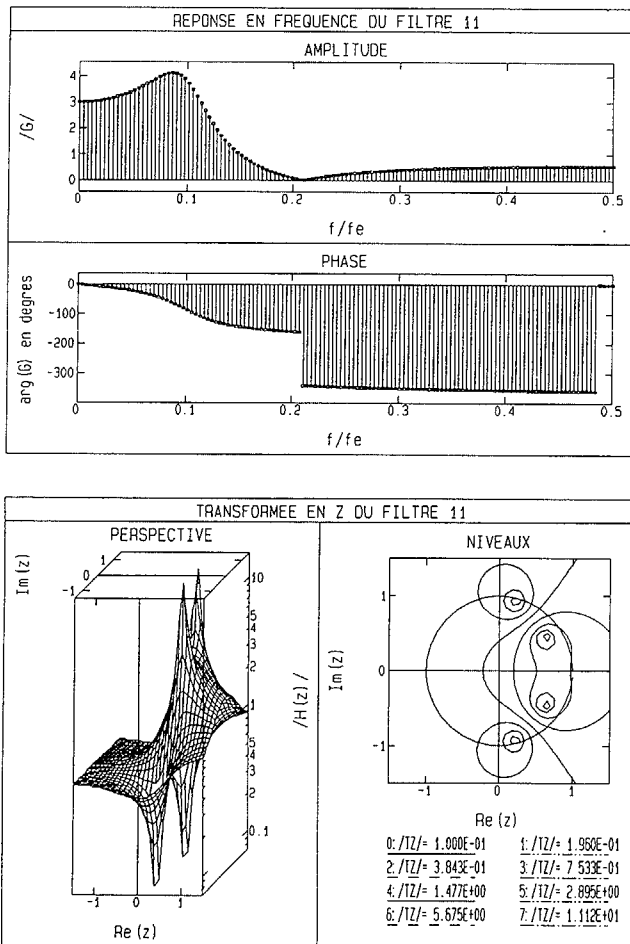


Figure 6. Réponse en fréquence et fonction de transfert d'un filtre récuratif.

## 7. CONCLUSION.

SFN est avant tout un logiciel intégré à but pédagogique mais ses procédures indépendantes peuvent être utilisées dans un programme spécifique de filtrage, ce qui constitue la dernière des six étapes de la méthode d'apprentissage proposée :

- visualisation de l'action de filtres types sur des signaux tests,

- analyse des caractéristiques des filtres,
- création de nouveaux filtres par saisie directe des coefficients au clavier,
- utilisation de méthodes de synthèse,
- étude des algorithmes en utilisant le mode pas à pas et la documentation des fichiers sources,
- intégration de modules de SFN dans des programmes réalisés par l'utilisateur.

L'évolution de cet outil est envisagée dans plusieurs directions : interfaçage avec une carte d'acquisition pour analyser des signaux réels, intégration d'autres méthodes de synthèse, identification de processus stationnaires et analyse spectrale par la méthode du maximum d'entropie au sens de Jaynes-Skilling [8]. Sans attendre, SFN est à la disposition de la communauté universitaire comme contribution à la mise au point d'un logiciel standard en diffusion libre pour illustrer les concepts élémentaires de traitement du signal.

## BIBLIOGRAPHIE.

- [1] S.L. MARPLE, Digital spectral analysis with applications, Prentice-Hall, Englewood Cliffs, New Jersey, 1987.
- [2] Assemblée Générale du GRECO Traitement du Signal et Image, Paris, 28 Février 1989.
- [3] T.G. LEWIS, W.H. PAYNE, Generalized Feedback Shift Register Pseudorandom Number Algorithm, Journal of the Association for Computing Machinery, 20, 3, 1973.
- [4] S.M. KAY, S.L. MARPLE, Spectrum analysis, a modern perspective, Proceedings of the IEEE, 69, 11, 1981.
- [5] R.W. HAMMING, Digital filters, Prentice-Hall, Englewood Cliffs, New Jersey, 1977.
- [6] J.H. McCLELLAN, T.W. PARKS, L.R. RABINER, FIR Linear phase filter design program, dans "Programs for digital signal processing" IEEE Press, John Wiley & Sons, 1979.
- [7] M. BELLANGER, Traitement numérique du signal, Masson, Paris, 1981.
- [8] D.S. STEPHENSON, Linear prediction and maximum entropy methods in NMR spectroscopy, Progress in NMR spectroscopy, 20, 1988.