

# Inversion de la transformée de Laplace numérique

## 1. Introduction

### 1. Généralités

Le précédent chapitre faisait référence à des signaux composés de plusieurs temps caractéristiques de relaxation. Ce chapitre a pour but de présenter les méthodes d'inversion permettant, à partir du signal mesuré, de remonter à la fonction de distribution de ces temps caractéristiques (inversion de la transformée de Laplace). Nous avons surtout cherché à évaluer l'efficacité des algorithmes en terme de traitement analytique ainsi que leurs limites. Les optimisations n'ont quasiment jamais porté sur le gain du temps de calcul car ce problème ne se pose pas :

- le traitement analytique des données se fait après la mesure, donc aucune contrainte temporelle concernant ce point,
- le temps de calcul maximum pour traiter ce problème reste (très) inférieur à la journée.

### 2. État de l'art

Plusieurs axes de recherches ont été abordés précédemment dans le problème de l'inversion de la transformée de Laplace (ou résolution d'une équation de Fredholm de première espèce [Baker 77]) qui reste toujours un problème d'actualité de grande envergure [Istratov 99]. Chacune des méthodes d'inversion présentées a des sensibilités au bruit, des pouvoirs de résolution et des fondements mathématiques différents.

Nous n'avons abordé que deux types d'approche dans la résolution de ce problème. En premier lieu, la recherche de fonctions propres dans la résolution de l'équation de Fredholm, particularisée aux exponentielles décroissantes [Provencher 76], puis généralisée pour un traitement global de minimisation [Provencher 82b]. Cet axe de recherche est encore d'actualité [Lamanna 05, Barone 01] mais, comme nous le verrons, l'utilisation des travaux de S. PROVENCHER nécessite certaines précautions quant à l'interprétation des solutions obtenues avec CONTIN, le logiciel qu'il a développé.

En deuxième lieu, nous avons étudié l'utilisation d'approximants numériques appliqués à la transformée en Z du signal numérique [Barone 01, Nielsen 00a]. La plupart du temps, ces approximants sont de type-Padé et nous montrerons certaines équivalences à ce sujet dans notre développement. Par ailleurs, la minimisation par les moindres-carrés type Levenberg-Marquardt [Nielsen 00b, Nielsen 99] peut être utilisée pour déterminer les amplitudes correspondant aux différents temps de relaxation obtenus. Bien que cela soit difficile à trois exponentielles. Nous montrerons que l'utilisation des polynômes de Padé permet d'aboutir à la distribution en amplitude sans avoir recours à ce type de minimisation.

Nous revisiterons l'historique de l'utilisation des polynômes de Padé notamment avec l'algorithme de Prony que nous confronterons à l'algorithme Z-Padé/SVD que nous avons développé.

### 3. Présentation du problème

Les signaux obtenus par relaxométrie RMN sont composés d'exponentielles décroissantes telle que

$$M(t) = M_0 \exp(-t/T) \quad (11.1)$$

Les informations qu'il est nécessaire de tirer de ces mesures sont les temps de relaxation  $T$  et les amplitudes correspondantes  $M_0$ . En ce qui concerne le traitement analytique nous ne considérerons que des exponentielles décroissantes. On peut toujours se ramener à ce cas mathématique même pour des mesures de temps de relaxation  $T_1$  par inversion-récupération et par saturation-récupération. En effet, les équations d'évolution sont de la forme suivante

$$M(t) = M_0 [1 - 2 \exp(-t/T)] \text{ et } M(t) = M_0 [1 - \exp(-t/T)] \quad (11.2)$$

respectivement pour l'inversion-récupération et pour la saturation-récupération. Ceci à condition de déterminer correctement l'aimantation à l'équilibre, et déterminer ainsi les constantes du problèmes permettant de se ramener à un signal du type 11.1. Comme nous l'avons vu dans le chapitre précédent, les signaux peuvent être composés de plusieurs exponentielles décroissantes. Le signal a donc pour expression :

$$M(t) = \sum_{k=1}^{N_e} M_{0,k} \exp(-t/T_k) \quad (11.3)$$

Ce modèle étant discret, il est plus juste de l'améliorer en considérant une sommation continue telle que

$$M(t) = M_0 \int_0^{+\infty} A_R(R) \exp(-Rt) dR \text{ avec } R = \frac{1}{T} \quad (11.4)$$

Nous verrons par la suite l'équivalence des deux descriptions sous certaines hypothèses concernant la distribution. Cette dernière équation constitue la transformée de Laplace de  $A_R$  (la fonction de distribution normalisée des taux de relaxation). L'analyse des signaux de relaxométrie RMN consiste en la détermination de la fonction de distribution  $A_R$  par inversion de la transformée de Laplace du signal. Ce problème mathématique est de grande envergure car il s'agit d'un problème *mal-posé*, la détermination des solutions est divergente et très sensible au bruit.

Dans ce chapitre, nous présenterons plusieurs méthodes pour remonter à la fonction de distribution  $A_R$  (ou  $A_T$ ). Dans un premier temps, nous étudierons l'utilisation et les limites du programme CONTIN de S. Provencher, qui reste une référence. Puis nous détaillerons l'algorithme mis au point par nos soins, basé sur la méthode dite Z-Padé/SVD que nous confronterons à son prédécesseur : l'algorithme de Prony.

### 4. Hypothèses et notations

#### a. Notations

On adoptera les notations suivantes pour ce chapitre :

- $N_p \in \mathbb{N}$  : le nombre de points expérimentaux,
- $\Delta_t \in \mathbb{R}$  : le temps d'échantillonnage homogène<sup>1</sup>,
- $t_k \in \mathbb{R}$  : l'instant correspondant au  $k^{\text{ème}}$  point expérimental, avec donc  $t_k = k \cdot \Delta_k$
- $y_k$  : les données expérimentales (signal) correspondant aux instants  $t_k$ .

---

<sup>1</sup>On rappelle que cette condition d'échantillonnage régulier peut ne pas être remplie pour des mesures de temps de relaxation  $T_1$  mais qu'elle est assurée pour les mesures de  $T_2$  par la séquence CPMG. On peut le cas échéant interpoler les points pour avoir un échantillonnage homogène.

## b. Hypothèses sur le signal

On fait un certain nombre d'hypothèses sur les données expérimentales qui seront nécessaires par la suite :

- *Fenêtre temporelle.* La fenêtre d'acquisition  $T_W = \Delta_t \cdot N_p$  doit être plus grande que le plus grand des temps de relaxation contenu dans le signal. Un rapport de cinq est nécessaire pour assurer une bonne détermination du plus grand des temps de relaxation recherchés.
- *Échantillonnage.* L'échantillonnage doit être suffisant pour décrire le plus petit des temps de relaxation, soit deux ou trois points de mesure pour décrire la décroissance en  $1/e$ . Le paramètre d'échantillonnage doit donc vérifier la relation  $\Delta_t \leq 0.6/R_{\max}$  [Provencher 76].
- *Bornes de recherches.* De ces deux considérations, on déduit l'intervalle de recherche des temps de relaxation  $[2\Delta_t; 5T_W]$ .

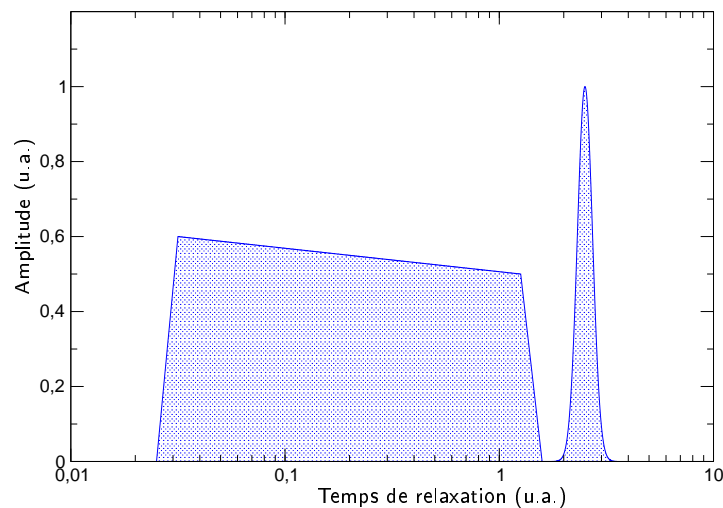
## c. Hypothèses sur la distribution

Dans l'hypothèse où la distribution n'est pas discrète et étant donné les applications de ces algorithmes, on peut faire les hypothèses suivantes sur la distribution  $A(R)$  :

- *Continuité.* Ces variables ayant des significations physiques, l'hypothèse de continuité est évidente,
- *Non-nulle sur un intervalle fermé et nulle ailleurs.* Le caractère fini des temps de relaxation et l'impossibilité de résoudre des temps trop courts imposent la nullité en dehors d'un compact proche de celui défini dans les hypothèses sur les mesures (à savoir  $[2\Delta_t; 5T_W]$ ),
- *Positivité.* Tous les processus de relaxation présentés ici concernent des processus de relaxation collectifs : on supposera alors la positivité de la fonction de distribution<sup>1</sup>.

## d. Exemple de distribution

Des hypothèses que nous venons de faire, on peut d'ores et déjà tracer une distribution vérifiant toutes ces hypothèses mais qui reste toutefois un peu « exotique » (figure II.1).



**Figure II.1** : Exemple de distribution de temps de relaxation simulée avec 10000 points, composée d'une rampe et d'une gaussienne.

<sup>1</sup>Il est possible de traiter le cas où le signal comporte des signaux périodiques : les solutions imaginaires pures peuvent alors être interprétées comme des solutions possibles [Yeramian 87]. Nous n'en avons pas fait l'étude car ceci sort du cadre de la thèse.

## II. CONTIN

Le programme CONTIN a été élaboré par S. PROVENCHER, il est disponible à l'adresse suivante (<http://s-provencher.com/pages/contin.shtml>) en code source FORTRAN. Les publications et les manuels se rapportant à CONTIN s'y trouvent également. Bien que nous allions nous montrer critiques dans cette section, il reste performant à condition de l'utiliser dans ses limites, de fournir des signaux « propres » et de ne pas tirer de conclusions trop hâtives sur les distributions obtenues. Cette dernière remarque reste générale pour tout problème de la transformée de Laplace, et nous reviendrons sur ce point ultérieurement.

### 1. Présentation

#### a. Modélisation

Pour l'élaboration de son logiciel CONTIN, S. Provencher a utilisé une méthode de régularisation linéaire générale sous contraintes pour contourner le caractère *mal-posé* du problème [Provencher 82b, Provencher 82a]. On considère les données expérimentales  $y_k$ , modélisées par une fonction déterminée par le produit matriciel du vecteur de variable  $\vec{x}$  par l'opérateur d'observation  $\mathbf{O}_k$ , soit littéralement

$$y_k = \mathbf{O}_k \vec{x} + \varepsilon_k \text{ avec } k \in \llbracket 1; N_y \rrbracket \quad (11.5)$$

où  $\varepsilon_k$  est une variable représentant le bruit. On peut approximer le problème par une linéarisation intégrale de l'opérateur  $\mathbf{O}_k$  telle que

$$y_k = \int_a^b F_k(\lambda) s(\lambda) d\lambda + \sum_{i=1}^{N_L} L_{ki} \beta_i + \varepsilon_k \quad (11.6)$$

où  $\mathbf{O}_k$  et  $\vec{x}$  ont été remplacés respectivement par les fonctions  $F_k(\lambda)$  et  $s(\lambda)$ . Cette hypothèse mathématique est responsable d'une perte d'information dans le signal [Provencher 80]. La somme rajoutée peut être utilisée pour tenir compte d'une éventuelle composante continue ( $N_L = 1$  et  $\forall k, L_{k1} = 1$ ). La discrétisation de l'équation 11.6 est un passage obligé pour le traitement numérique

$$y_k = \sum_{m=1}^{N_g} c_m F_k(\lambda_m) s(\lambda_m) + \sum_{i=1}^{N_L} L_{ki} \beta_i + \varepsilon_k \quad (11.7)$$

où  $c_m$  constitue les poids de  $F_k$ . En utilisant un opérateur matriciel  $\mathbf{A} = [A_{i,j}]$  avec  $i, j \in \llbracket 1; N_y \rrbracket \times \llbracket 1; N_x \rrbracket$  pour la réécriture de cette dernière équation, on aboutit à

$$y_k = \sum_{j=1}^{N_x} A_{kj} x_j + \varepsilon_k \text{ où } N_x = N_g + N_L \quad (11.8)$$

Dans cette dernière équation apparaissent les paramètres importants de l'algorithme de résolution de CONTIN. Tout d'abord  $N_g$  le nombre de point sur la grille de recherche : le nombre d'exponentielles supposé dans la distribution même si certaines correspondent à une amplitude nulle. L'accroissement de  $N_g$  améliore la qualité mais augmente dans le même temps le temps de calcul (fonction cubique de  $N_g$ ). Enfin  $\lambda_{m_{\max}}$  et  $\lambda_{m_{\min}}$  les bornes de recherches, qui dans notre cas d'étude sont  $R_{\min}$  et  $R_{\max}$ . Il est évident que le problème n'est soluble que dans le cas où  $N_x \leq E(N_y/2)$  : on ne peut tirer une distribution ayant plus de points que ceux présents dans le signal, ceci en regard de la linéarisation du problème.

#### b. Méthode de résolution

Le problème est formulé comme une méthode par les moindres-carrés à laquelle est ajoutée un régulariseur quadratique. La solution optimale du problème doit **minimiser**  $V(\alpha)$  satisfaisant la rela-

tion II.9. Donc en plus d'assurer une minimisation des moments d'ordre 2 (distance modèle-expérience), il est possible de contraindre la recherche de solutions par d'autres critères (positivité, monotonie, convergence...).  $V(\alpha)$  a lors pour expression :

$$V(\alpha) = \left\| \frac{(\vec{y} - \mathbf{A}\vec{x})}{\mathbf{M}_\varepsilon} \right\|^2 + \alpha^2 \|\vec{r} - \mathbf{R}\vec{x}\|^2 \quad (\text{II.9})$$

où  $\vec{r}$  et  $\mathbf{R}$  peuvent être choisis par l'utilisateur,  $\mathbf{M}_\varepsilon$  est la matrice de covariance des  $\varepsilon_k$  et enfin  $\alpha$  est appelé paramètre de régularisation : il permet de faire varier l'importance accordée à  $\|\vec{r} - \mathbf{R}\vec{x}\|$ . C'est ici qu'il faut utiliser CONTIN avec précaution car le changement de valeur de ce paramètre influence les solutions. S. Provencher a décidé d'appliquer un principe de parcimonie, en appliquant des contraintes assurant la positivité de la distribution, sans pour autant que cela soit nécessaire [Bertero 85] et définissant son régulariseur par

$$\|\vec{r} - \mathbf{R}\vec{x}\|^2 = \int_a^b \left[ \frac{\partial^2 s(\lambda)}{\partial \lambda^2} \right]^2 d\lambda \quad (\text{II.10})$$

**définissant ainsi la meilleure solution comme celle subissant le moins de variations.** Ce type de régularisation est parfois très bien adaptée, notamment pour la détermination de la distribution en poids des polymères. Pour l'application de ce traitement analytique aux signaux de RMN, il est nécessaire de bien connaître le comportement du logiciel face à des signaux de RMN et à des simulations afin d'en déterminer les limites et les biais. Nous ne pouvons que conseiller la lecture des références qui se rapportent au programme de S. Provencher. Nous présentons ici une étude de l'influence de chacun des paramètres importants de l'algorithme utilisé par CONTIN.

Pour l'utilisation de CONTIN, un certain nombre de paramètres doivent être définis par l'utilisateur. Par la suite, nous ne retiendrons que ceux qui ont une influence sur les résultats obtenus à jeu de données égal et à problème mathématique équivalent :

- la définition de la grille de recherche par  $f_g(x)$ ,  $f_g^{-1}(x)$  et  $\frac{\partial f_g}{\partial x}$ , fonction bijective sur  $\mathbb{R}^+$
- le nombre de points constituant la grille  $N_g$ ,
- les bornes de la grille de recherche  $[\lambda_{\min}; \lambda_{\max}]$ ,
- les valeurs du paramètre de régularisation  $\alpha$ ,
- les valeurs de l'ordre du régulariseur.

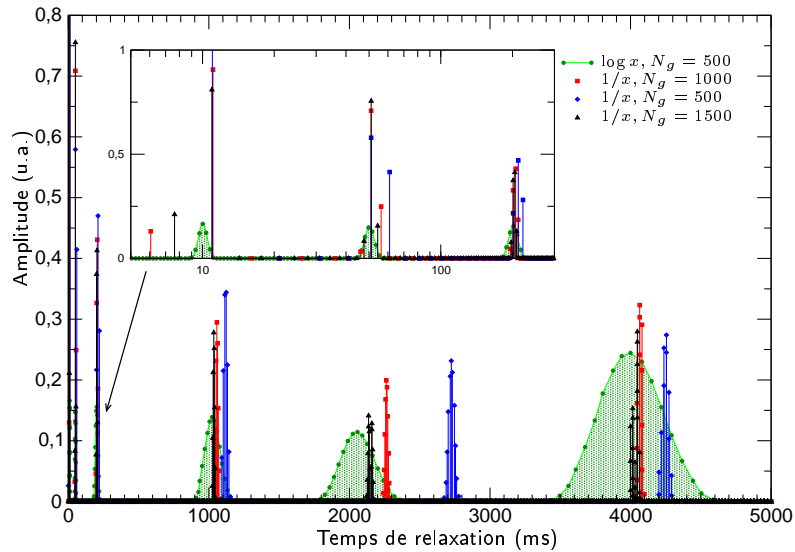
Les autres sont définis par défaut pour l'utilisation de CONTIN en vue de l'inversion de la transformée de Laplace.

## 2. Effet de grille

Dans son programme, S. Provencher laisse la possibilité de définir sa propre grille par une fonction de distribution qui doit être bijective. Par défaut, cette dernière est définie par la fonction  $\log(x)$ . Nous verrons par la suite que ce choix est le mieux adapté à la recherche de temps de relaxation dans le cas de la transformée de Laplace. Néanmoins nous nous sommes intéressés au cas d'une autre distribution, la plus intuitive après la définition logarithmique : la définition par  $1/x$ . Pour étudier ce cas, nous avons simulé un signal composé de plusieurs exponentielles suffisamment espacées pour que les problèmes de résolution ne soient pas prépondérants

$$S(t) = \sum_{k=1}^6 e^{-t/T_i} \text{ avec } T_i = \{10, 50, 200, 1000, 2000, 4000\} \text{ ms} \quad (\text{II.11})$$

Nous avons représenté sur la figure II.2 cette étude par les distributions obtenues pour différentes valeurs du nombre points sur la grille. On rappelle que l'on considère que le programme CONTIN a trouvé la solution stable lorsque l'augmentation du nombre de points constituant la grille n'influe plus sur la forme de la solution rendue par CONTIN.



**Figure II.2 :** Influence de la définition de la grille sur la convergence de CONTIN.

On remarque alors qu'il est nécessaire d'accroître le nombre de points sur la grille afin de trouver les bonnes valeurs des temps de relaxation et que les valeurs obtenues ne sont pas très correctes en regard de ce que l'on peut obtenir avec l'échelle logarithmique, bien que l'élargissement des pics soit plus important. Nous n'avons pas pu trouver les raisons mathématiques rigoureuses pour lesquelles l'échelle logarithmique semble être la plus adaptée, bien que cela se comprenne intuitivement. Nous reviendrons sur ce point plus tard.

### 3. Influence du bruit - Rôle de la qualité de la grille

#### a. Influence de $N_g$

Pour tester l'influence du bruit sur les résultats rendus par CONTIN, nous avons simulé un signal composé de deux exponentielles relativement proches que nous avons bruité par une variable aléatoire gaussienne  $\varepsilon = \mathcal{N}(0, 1)$  affectée d'un préfacteur d'amplitude  $n_\varepsilon$  :

$$S(t) = e^{-t/1000} + e^{-t/2000} + n_\varepsilon \varepsilon \quad (\text{II.12})$$

Sur la figure II.3 sont représentées les solutions fournies par CONTIN pour un nombre de points de grille constant, le bruit variant en amplitude. On remarque que plus le rapport signal sur bruit augmente plus la résolution est efficace. Ceci se comprend très bien qualitativement. Il est possible de remédier à cette lacune en augmentant le nombre de points sur la grille; comme on peut le voir figure II.4 où à amplitude de bruit fixée l'accroissement de  $N_g$  permet de résoudre les deux temps, au prix d'une augmentation du temps de calcul.

Toutefois on ne peut pas augmenter indéfiniment le nombre de points sur la grille afin de pouvoir résoudre les temps de relaxation à volonté et ceci pour plusieurs raisons :

- le temps de calcul augmente au cube du nombre de points sur la grille, mais les ordinateurs actuels sont suffisamment performants pour que ce temps de calcul soit un facteur de décision négligeable,
- $N_g$  est limité à 1500 dans le cadre de notre étude car l'une des particularités, et non des moindres, de FORTRAN est qu'il faut déclarer les variables statiquement au début de chaque routine, si bien que l'échange de données par matrice entre routines (puisque'il se fait par copie de variable) engendre une saturation rapide la mémoire vive (pourtant de 2 Go!).

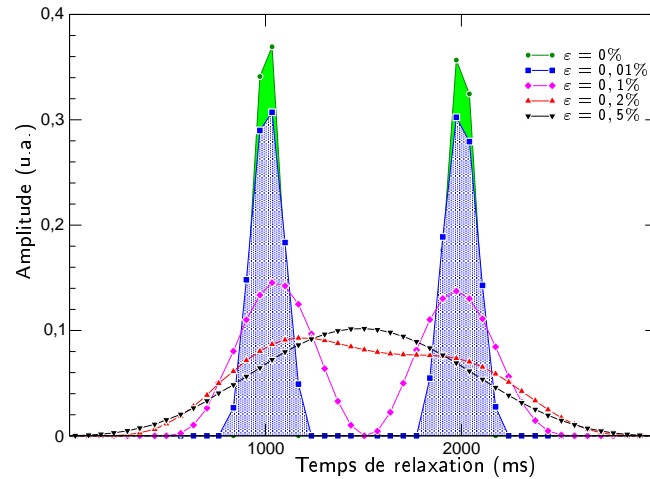


Figure II.3 : Influence du bruit sur la résolution de CONTIN.  $N_g$  est fixé à 100.

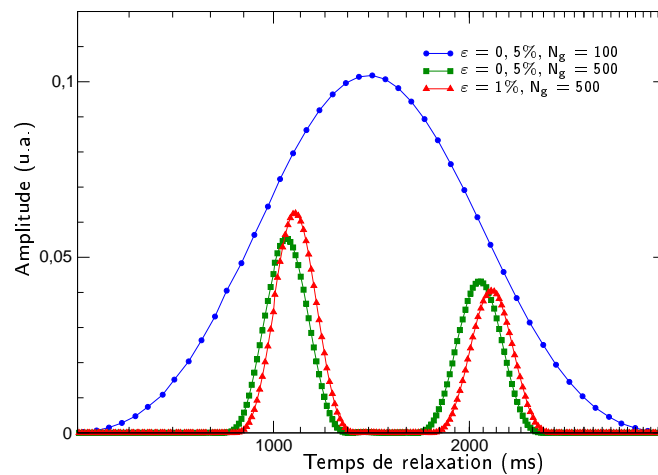


Figure II.4 : Amélioration de la résolution de CONTIN par augmentation du nombre points sur la grille

C'est l'une des raisons qui nous a poussé à explorer d'autres algorithmes de calcul et dans d'autres langages de programmation (notamment avec des allocations dynamiques de mémoire, des destructeurs de variables telque le C++ par exemple). Il est tout à fait possible de transposer le code FORTRAN en C++ avec allocation dynamique de mémoire (l'utilisation de la librairie *f2c*<sup>1</sup> ne suffit apparemment pas car elle conserverait l'allocation statique de mémoire), mais nous avons pu remarquer que dans tous les cas étudiés jusqu'à présent, nous n'observons pas de changement d'allure de la distribution rendue par CONTIN au delà de 1000 points.

- de toute façon  $N_g$  est limité à 40000 ! [Provencher 84],
- pour finir, nous avons choisi de ne pas persévérer dans cette voie, les améliorations à apporter à ce type de programmation soit purement mathématiques et non algorithmiques.

<sup>1</sup>La librairie *f2c* traduit littéralement le Fortran en langage C.

## b. Influence des bornes de recherche

### i. Généralités

Il est nécessaire de fournir à CONTIN les bornes de recherche  $\lambda_{m_{\min}}$  et  $\lambda_{m_{\max}}$ . Il paraît intuitif de fixer arbitrairement ces bornes de telle sorte à ne pas avoir de zone où la solution n'a pas de sens :

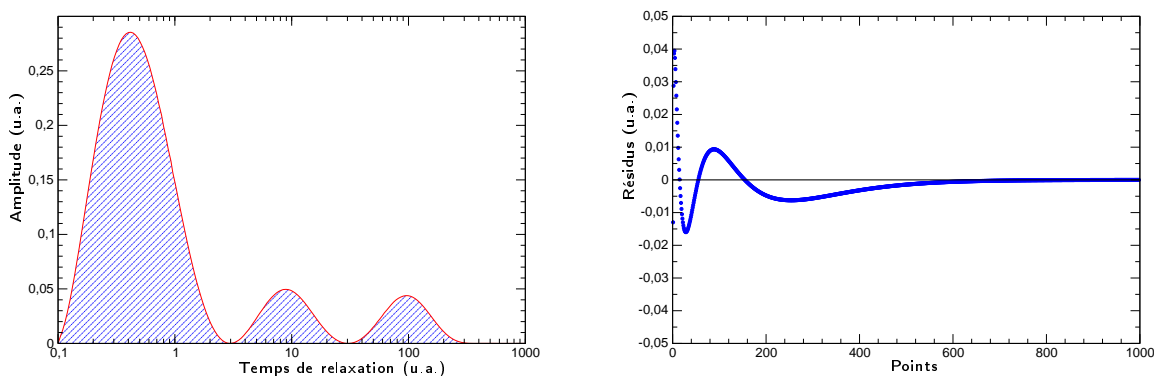
- $(\lambda_{m_{\max}})^{-1}$  ne peut être trop petit par rapport au temps d'échantillonnage. Il est difficile de définir une exponentielle si seulement deux ou trois points la décrivent. Ceci peut être le cas en relaxation RMN, où des temps de relaxation sont trop courts pour être correctement échantillonnés. Dans la littérature, on peut trouver des critères de recherche du type  $\lambda_{m_{\max}} \leq 0,693/\Delta t_k$  où  $\Delta t_k$  est l'échantillonnage [Provencher 76],
- $(\lambda_{m_{\min}})^{-1}$  ne peut être trop grand par rapport à la fenêtre de mesure. Là encore, le sous-échantillonnage laissera trop de doute sur les temps longs. D'un point de vue pratique concernant la RMN, il ne tient qu'à l'utilisateur de s'assurer que la fenêtre de mesure soit suffisamment grande pour décrire correctement la relaxation.

### ii. Cas de $\lambda_{m_{\max}} = T_{\min}$

Nous détaillons les cas de figures présentés ci-dessus sur des exemples : considérons un signal trop peu échantillonné avec  $\Delta t_k = 0.35$  et 1000 points tel que

$$y(t) = e^{(-t/0.1)} + e^{(-t/5)} + e^{(-t/50)} + 10^{-7} \varepsilon \quad (\text{II.13})$$

et possédant donc un temps de relaxation trop court pour que l'échantillonnage le décrive correctement.



**Figure II.5** : Distribution et résidus obtenus par CONTIN sur le cas :  $y(t) = e^{(-t/0.1)} + e^{(-t/5)} + e^{(-t/50)} + 10^{-7} \varepsilon$ .

Le résultat rendu par CONTIN sur ce type de cas se trouve sur la figure II.5 et n'est pas du tout en accord avec la simulation. En augmentant la fenêtre de recherche pour s'assurer de la nullité à ses extrémités le résultat est pire encore... Le problème vient du fait que CONTIN ne s'assure pas de l'existence de telle solution. En effet le bruit en début de courbe fait « croire » à la présence de temps de relaxation courts en regard du temps d'échantillonnage qui, dans ce cas, est trop grand (pas assez de points pour caractériser la première exponentielle), ce qui n'est pourtant pas le cas. Il faut restreindre CONTIN à ne chercher que dans la zone justifiée.

Toutefois, le fit de cette distribution reste relativement acceptable (figure II.5), bien que l'incertitude sur les temps courts, visible sur la distribution calculée avec CONTIN, reste un problème qui ne peut pas toujours être résolu, car l'échantillonnage a une certaine limite imposée par les conditions expérimentales. Dans le cas de la RMN, c'est le temps mort de la bobine qui est responsable en premier lieu de la limite inférieure du temps d'échantillonnage des mesures.



iii. Cas de  $\lambda_{m_{\min}} = T_{\max}$

Considérons le signal de mesure présenté sur la figure II.6. Sur la figure II.7 sont représentées les distributions obtenues par CONTIN avec différentes valeurs de  $\lambda_{m_{\min}}$ . L'observation du signal de mesure (figure II.6) montre clairement que le signal n'a pas totalement relaxé, une borne de recherche trop grande laisse le doute sur la présence de temps de relaxation de grandes valeurs (*i.e.*  $\lambda_{m_{\min}}$  de faible valeur) au détriment de solutions probables dans le champ de recherche. Toutefois les résidus montrent que le fit issu de la distribution (figure II.7) reste tout à fait acceptable.

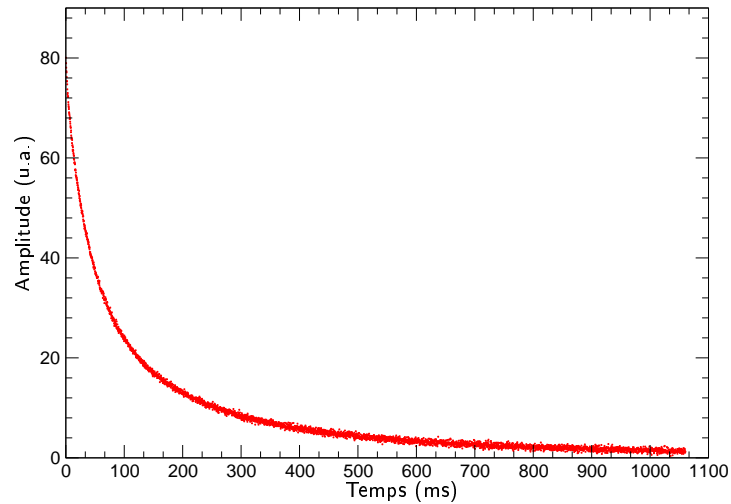


Figure II.6 : Signal de mesure expérimental (pétrole brut lourd) avec une fenêtre de mesure trop petite

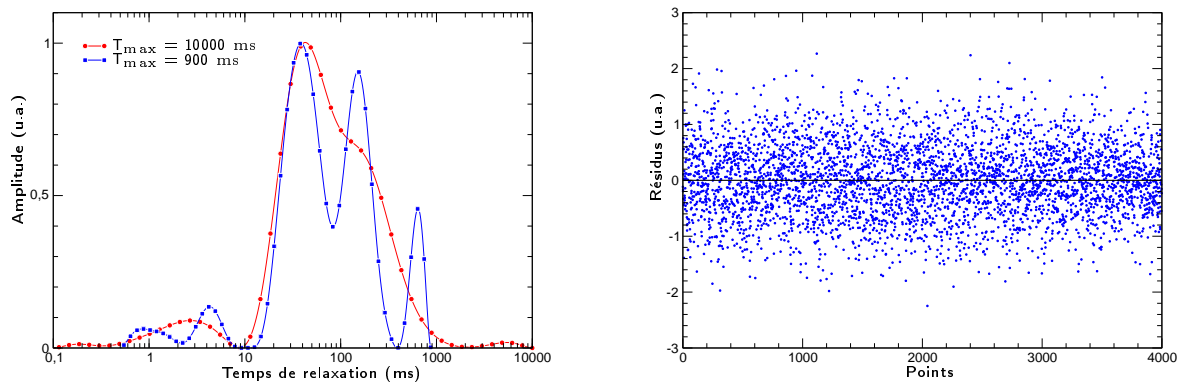


Figure II.7 : Distribution des temps de relaxation du signal figure II.6 obtenues avec CONTIN pour différentes valeurs de la borne inférieure, ainsi que les résidus correspondant.

#### 4. Influence du régulariseur

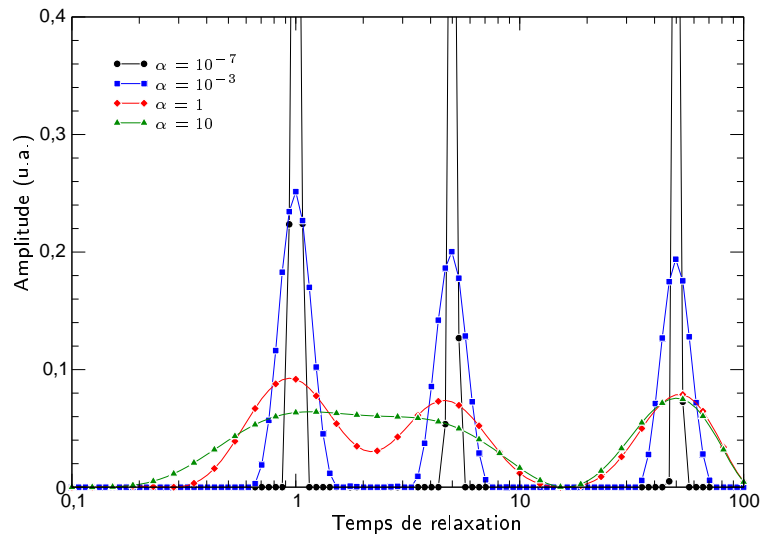
On rappelle que le paramètre de régularisation est défini par la relation suivante :

$$V(\alpha) = \left\| \frac{(\vec{y} - \mathbf{A}\vec{x})}{M_\varepsilon} \right\|^2 + \alpha^2 \|\vec{r} - \mathbf{R}\vec{x}\|^2 = \left\| \frac{(\vec{y} - \mathbf{A}\vec{x})}{M_\varepsilon} \right\|^2 + \alpha^2 \int_{\lambda_{m_{\min}}}^{\lambda_{m_{\max}}} \left[ \frac{\partial^2 s(\lambda)}{\partial \lambda^2} \right]^2 d\lambda \quad (\text{II.14})$$

Plus la valeur de  $\alpha$  est élevée, plus CONTIN attache de l'importance aux variations de la solution obtenue et plus il limite celles-ci. On a effectué plusieurs tests sur un signal **non-bruité** tel que :

$$S(t) = e^{-t/1.0} + e^{-t/5.0} + e^{-t/50.0} \quad (\text{II.15})$$

et constitué de 1000 points et dont la fenêtre temporelle est de 7 fois le temps de relaxation le plus grand (soit  $\Delta t_k = 0,35$ ). Nous avons effectué des calculs de distributions pour plusieurs valeurs de  $\alpha$  avec une grille de 100 points (figure II.8).



**Figure II.8 :** Influence du paramètre de régularisation de CONTIN

On remarque alors que l'influence de  $\alpha$  est bien celle que l'on attend : plus  $\alpha$  augmente moins CONTIN autorise des variations brusques dans la distribution. Cela peut même empêcher la résolution de deux temps de relaxation de signaux pourtant non-bruités. Mais cela pose un autre problème : celui de la confiance que l'on peut accorder de manière qualitative à la forme particulière de la distribution et surtout de manière quantitative à l'aire sous la courbe ! En effet, celle-ci dépend fortement de la valeur de  $\alpha$ . Ce dernier est, par défaut, optimisé pour minimiser la relation II.9. Dans cet exemple, la solution retenue à certes la meilleure valeur de  $\alpha$  mais ne correspond pas à la meilleure solution.

#### 5. Conclusion sur l'utilisation de CONTIN

L'utilisation de CONTIN nécessite un grand soin et beaucoup de précautions. Son comportement peut amener à de fausses conclusions. L'étude que nous avons faite avait pour but de l'éprouver et d'en connaître les limites. Même si la critique est parfois rude, il ne faut pas perdre de vue que les tests qui lui ont été appliqués traitent d'un problème en particulier : la transformée de Laplace. Les conclusions auxquelles nous sommes arrivés ne portent que sur ce problème là et en aucun cas sur d'autres problèmes que CONTIN peut résoudre.

Par ailleurs, nous n'avons présenté ici que des tests avec des distributions discrètes (peigne de Dirac). Nous n'avons pas encore éprouvé CONTIN sur des tests portant sur des fonctions de distribution plus complexes (figure II.1), ce cas sera traité dans la suite.

### III. Algorithme Z-Padé/SVD

#### 1. Introduction

##### a. Approche du problème

Lorsque le signal de mesure se résume à quelques exponentielles (typiquement jusqu'à 3 ou 4 selon le type de signal et si les exponentielles sont suffisamment différentiables), il est toujours possible d'utiliser la méthode des moindres-carrés afin de modéliser la courbe de mesure, à la condition que le vecteur solution de départ soit suffisamment proche de la solution réelle et qu'il y ait convergence. Ce type de test est fiable et sera systématique lorsque la distribution se résume à une distribution quasi-discrète d'exponentielles, surtout si on peut l'utiliser conjointement avec d'autres algorithmes permettant d'avoir une valeur approchée de la solution, ce que nous allons voir dans les sections suivantes.

Toutefois la modélisation par une distribution discrète n'est pas toujours suffisante. La distribution des temps de relaxation peut être plus *exotique*. Nous verrons que la détermination d'une fonction continue n'est pas toujours possible ou en tout cas peu fidèle. Dans un premier temps nous allons présenter comment déterminer les temps de relaxation d'un signal de mesure par des algorithmes basés sur l'utilisation des polynômes de Padé, appliqués à la transformée en Z du signal. Nous allons utiliser les propriétés de la transformée de Laplace [Schwartz 65]. En effet, reprenons l'expression de  $M(t)$

$$M(t) = M_0 \int_0^{+\infty} A_R(R) \exp(-Rt) dR \text{ avec } R = \frac{1}{T} \quad (II.16)$$

La transformée de Laplace de cette fonction<sup>1</sup> a pour expression

$$\begin{aligned} \mathcal{L}[M](x) &= \int_0^{+\infty} M(t) \exp(-xt) dt \\ &= \int_0^{+\infty} \int_0^{+\infty} [A_R e^{-(R+x)t}] dR dt \\ &= \int_0^{+\infty} \frac{A_R}{R+x} dR \end{aligned} \quad (II.17)$$

La détermination des pôles de la transformée de Laplace du signal (qui est une fonction de Stieltjes) et de leurs résidus permet de remonter à la distribution voulue [Claverie 88]. Toutefois, l'évaluation de la transformée de Laplace n'est pas chose facile d'autant que le signal est une donnée numérique. Nous allons utiliser pour cela la transformée en Z et ce pour plusieurs raisons :

- la transformée en Z et la transformée de Laplace sont liées sous certaines conditions comme nous le verrons dans la section suivante,
- la transformée en Z est une série formelle constituée des points de mesures qui sont les seules entités à notre disposition,
- puisque la transformée en Z est une série formelle, on pourra utiliser des outils mathématiques adaptés pour l'évaluer (approximants de Padé).

Dans cette section nous allons rappeler les principales définitions et propriétés de la transformée en Z, puis faire le lien avec notre cas d'étude, pour faire apparaître le problème mathématique à résoudre et enfin déterminer les temps de relaxation contenus dans un signal numérique ainsi que les amplitudes correspondantes.

---

<sup>1</sup>Ici pas de problème de convergence : la fonction  $e^{-x}$  converge normalement sur  $R_+$  (au sens de la norme qui dérive du produit scalaire usuel).

## b. Transformée en Z

Nous rappelons quelques points de la transformée en Z, un outil qui sera utile dans ce chapitre et dont quelques propriétés seront utilisées, notamment par sa relation avec la transformée de Laplace [Kunt 99].

### i. Introduction

La transformée en Z,  $X(z)$  d'un signal  $x(k)$  est définie par

$$X(z) = \sum_{k=-\infty}^{+\infty} x(k) z^{-k} \quad (11.18)$$

où  $z \in \mathbb{C}$  et où  $X(z) : \mathbb{C} \rightarrow \mathbb{C}$ . On utilisera la notation suivante :

$$X(z) = Z[x(k)] \quad (11.19)$$

Cette dernière définition est appelée transformée en Z bilatérale car les indices  $k$  de sommation couvrent  $\mathbb{Z}$ . Le système que l'on étudie étant causal, on se ramènera à une transformation en Z unilatérale :

$$X(z) = \sum_{k=0}^{+\infty} x(k) z^{-k} \quad (11.20)$$

L'existence de la transformée en Z est régie par la convergence de la série  $x(k)$ , on supposera que ce critère est toujours vérifié. Dans le cadre de notre étude, nous verrons que la série  $x(k)$  est nulle à partir d'un certain rang, assurant ainsi la convergence de la somme de cette série.

*Propriété* (voir annexe page 155). La transformée de Laplace d'un signal échantillonné est la transformée en Z du signal numérique correspondant évaluée pour  $z = \exp(s \Delta t)$

$$Z[x_a(k \Delta t)] \Big|_{z=e^{-s \Delta t}} = \mathcal{L}[x_e(t)] \quad (11.21)$$

Cette propriété importante de la transformation en Z nous permet d'exprimer  $y(t)$  comme une série de fonction. En effet, supposons que la distribution  $A_i$  soit un peigne de Dirac régulièrement espacés tel que

$$A(R) = \sum_{k=0}^{+\infty} A_k \delta(R - k \Delta_R) \quad (11.22)$$

Cela ne pose pas de problème, la discrétisation de  $A_R$  est libre de choix en général<sup>1</sup>. De la relation 11.21 on déduit l'expression de  $y(t)$  transformée en Z de  $A_R$  en  $z = \Delta_R t$

$$y(t) = \int_0^{+\infty} A(R) \exp(-R t) dR \quad (11.23)$$

$$= Z[A(k)] \Big|_{z=\exp(\Delta_R t)} \quad (11.24)$$

$$y(t) = \sum_{m=0}^{+\infty} A_m \exp(-m \Delta_R t) \quad (11.25)$$

Ce qui justifie l'équivalence des sommes d'exponentielles discrètes et continue sous la condition que la distribution soit de la forme 11.22.

<sup>1</sup> Comme on discrétise le problème, le choix de la grille n'appartient qu'à nous.

## ii. Utilisation des propriétés de la transformée en Z

Considérons un signal échantillonné de manière homogène dans le temps (condition nécessaire). On note par  $y(t_k) = y_k$  les  $N_p$  points de mesures aux instants  $t_k$  et on a l'expression suivante

$$y_k = y(k \Delta_t) = \sum_{m=0}^{+\infty} A_m \exp(-m \Delta_R k \Delta_t) = \sum_{m=0}^{+\infty} A_m \xi_m^{-k} \quad (11.26)$$

où  $\xi_m$  est défini par

$$\xi_m = e^{-m\Delta_R} \quad \text{avec} \quad m\Delta_R = R_m \quad (11.27)$$

ce qui nous permet de déterminer l'ensemble des temps de relaxation. Maintenant considérons la transformée en Z de  $y_k$  telle que

$$Z[y_k] = \sum_{k=0}^{+\infty} y_k z^{-k} = \sum_{k=0}^{+\infty} \sum_{m=0}^{+\infty} A_m (\xi_m z)^{-k} \quad (11.28)$$

Pour assurer la convergence de cette dernière série, on suppose que  $|\xi_m z| < 1$ . Néanmoins,  $\xi_m = 1$  est toujours solution, correspondant à un temps de relaxation infini (ou à un taux de relaxation nul). Ceci implique un domaine de convergence défini par le cercle unité plus un ensemble discret de solution  $\{\xi_m\}$ . D'un coté la série finie  $y_k$  nous permet d'inverser les signes somme dans la relation 11.28 ; de l'autre  $\sum_{k=0}^{+\infty} z^k = 1/(1-z)$  dans son domaine de convergence nous mène à l'expression de  $Z[y_k]$  telle que

$$Z[y_k] = \sum_{m=0}^{+\infty} \frac{A_m}{1 - (\xi_m z)^{-1}} \quad (11.29)$$

Le problème est alors reporté sur la détermination des pôles et des résidus de la transformée  $Z[y_k]$ . Il est possible d'écrire la relation 11.29 comme une somme continue telle que :

$$Z[y_k] = \int_0^{+\infty} \frac{A(R)}{1 - [\xi(R) z]^{-1}} dR \quad (11.30)$$

où  $\xi(R) = \exp(-R \Delta_t)$ . La relation 11.30 présente une fonction de Stieltjes [Gilewicz 78]. La propriété d'échantillonnage régulier apparaît comme absolument nécessaire puisque qu'elle pose le problème comme celui de la détermination des pôles et résidus d'une fraction rationnelle.

La principale raison pour laquelle nous avons utilisé la transformée en Z est qu'elle peut être approchée par des fonctions. Nous utiliserons pour cela les approximants de Padé qui permettent de déterminer les singularités des fonctions qu'ils approchent [Karlsson 80]. La deuxième raison est que les pôles de la fraction polynomiale 11.29 nous permettent d'accéder à la distribution (relation 11.27).

## 2. Approximants de Padé

Cette section introduit les notions et définitions des approximants de Padé [Baker 81, Brezinski 80, Gilewicz 78, Press 92] qui serviront à approcher la transformée en Z (relation 11.29), cette dernière apparaissant comme une série formelle. Les approximants de Padé d'une série  $c_k$  sont définis par deux polynômes  $P$  et  $Q$  de degré  $p$  et  $q$  respectivement, tels qu'ils approchent la somme de cette série de la sorte

$$\sum_{k=0}^{+\infty} c_k z^k = \frac{P(z)}{Q(z)} + O(z^{p+q+1}) \quad (11.31)$$

Les approximants de Padé sont les meilleures approximations locales pour la norme de la convergence uniforme. On les utilise précisément pour exhiber les propriétés non-locales des fonctions, par

exemple pour réaliser le prolongement analytique d'une fonction donnée par sa série. Dans le cadre de notre étude, nous ne les utiliserons que sur des cas locaux : pour approcher la fonction sur son domaine de définition.

### a. Propriétés

Nous ne rappelons ici que la définition et les propriétés utiles des polynômes de Padé, on se reportera à l'annexe page 156 pour plus de détails sur les approximants de Padé.

Soient deux polynômes  $P$  (de degré  $p$ ) et  $Q$  (de degré  $q$ ) tels que

$$f(t) - \frac{P(t)}{Q(t)} = O(t^{p+q+1}) \text{ pour } t \rightarrow 0. \quad (11.32)$$

où les polynômes  $P$  et  $Q$  ont pour expressions

$$\begin{cases} P(t) = a_0 + a_1 t + \dots + a_p t^p = \sum_{k=0}^p a_k t^k \\ Q(t) = 1 + b_1 t + \dots + b_q t^q = 1 + \sum_{k=1}^q b_k t^k \end{cases} \quad (11.33)$$

On peut avoir l'expression de l'approximant de Padé sous forme d'un rapport de deux déterminants [Gilewicz 78,p.188] [Guziński 80,p223] :

$$[p/q]_f(t) = \frac{\begin{vmatrix} \sum_{i=0}^{p-q} c_i t^{q+i} & \sum_{i=0}^{p-q+1} c_i t^{q-1+i} & \dots & \sum_{i=0}^p c_i t^i \\ c_{p-q+1} & c_{p-q+2} & \dots & c_{p+1} \\ \vdots & \vdots & & \vdots \\ c_p & c_{p+1} & \dots & c_{p+q} \end{vmatrix}}{\begin{vmatrix} t^q & t^{q-1} & \dots & 1 \\ c_{p-q+1} & c_{p-q+2} & \dots & c_{p+1} \\ \vdots & \vdots & & \vdots \\ c_p & c_{p+1} & \dots & c_{p+q} \end{vmatrix}} \quad (11.34)$$

De nombreux travaux ont été effectués afin d'optimiser le calcul des approximants de Padé en utilisant les propriétés des tables de Padé : [Gilewicz 78,ch.7] pour une étude classique et [Della Dora 80] pour une étude peu orthodoxe. Aucune des propriétés des tables de Padé n'a été utilisée, leurs implémentations sont longues et non rentables en regard du temps de calcul des déterminants 11.34 par les ordinateurs actuels. Nous avons utilisé la méthode du pivot de Gauss pour calculer ces déterminants (*Gaussian Elimination with partial pivoting for LU decomposition* [Golub 96]).

### b. Application des approximants de Padé

Soit  $f$  la transformée en  $Z$  des données expérimentales, soit  $\tilde{z}$  dans  $\mathbb{C}$  tel que  $\tilde{z} = z^{-1}$ . Alors la fonction  $f$  est telle que

$$f(\tilde{z}) = Z[y_k] = \sum_{k=0}^{+\infty} y_k z^{-k} = \sum_{k=0}^{+\infty} y_k \tilde{z}^k \quad (11.35)$$

Le domaine de convergence de  $f(\tilde{z})$  est  $\{\mathbb{C} \setminus \{|\tilde{z}| \geq 1\}\} \cup ]0; 1[$ . On note par  $\zeta_m$  les racines des approximants de Padé  $Q(z)$ . D'après les relation 11.27 et 11.29 le taux de relaxation  $R_m$  est tel que

$$R_m = -\frac{1}{\Delta_t \ln \zeta_m} \quad (11.36)$$

Seules les racines réelles de  $Q$  dans  $]0;1[$  sont physiquement acceptables. En les calculant et en utilisant la relation II.36, nous sommes en mesure de déterminer l'ensemble  $\{R_m\}$ .

Les amplitudes correspondantes  $A_m$  de chaque taux de relaxation calculé  $R_m$  peut être obtenu par réduction du degré du polynôme  $Q$  sur ses pôles. Soit  $\{\zeta_m\}$  l'ensemble des racines du polynômes  $Q$  et  $\{A_m\}$  les amplitudes correspondantes. Connaissant toutes les racines de  $Q$ , l'évaluation en  $\zeta_\mu$  mène à la relation

$$A_\mu = P(\zeta_\mu) \prod_{i=1, i \neq \mu}^q b_n (\zeta_\mu - \zeta_i) \quad (II.37)$$

D'autres approximations sont possibles, notamment celles qui font appels aux prédicteurs linéaires [Nielsen 00a], on montre que cette méthode est un cas particulier des approximants de Padé (voir annexe page 158).

### 3. Algorithme de Prony

#### a. Présentation

Dans cette section nous présentons l'algorithme découlant de l'utilisation des approximants de Padé, utilisé à l'ordre  $q = N_e$  [Prony 95]. Sa simplicité d'implémentation, en regard de ce qui a été présenté et du caractère historique, en fait un algorithme incontournable. Toutefois ces avantages sont un peu ternis par sa grande sensibilité au bruit [Istratov 99] et par l'incapacité de traiter des distributions continues.

Considérons que le signal est composé d'une somme discrète finie de  $N_e$  exponentielles. Dans ce cas la transformée en Z s'écrit

$$Z[y_k] = \sum_{i=0}^{N_e} \frac{A_i}{1 - (\xi_i z)^{-1}} \quad (II.38)$$

Le degré du polynôme de Padé qui approche  $Z[y_k]$  ne peut excéder  $N_e$ , le système étant surdéterminé au-delà. L'algorithme qui consiste à déterminer les pôles de l'approximant de Padé  $[N_e - 1, N_e]_{Z[y_k]}$  de la transformée en Z du signal  $y_k$  est connu sous le nom d'*algorithme de Prony*.

#### b. Efficacité de l'algorithme de Prony

Nous avons éprouvé cet algorithme qui a le mérite d'être très rapide en terme d'exécution et d'une redoutable efficacité sur des signaux peu bruités. Son implémentation est tout à fait envisageable sur des systèmes embarqués avec peu de ressources CPU. On préférera quand même dans ce dernier cas, si l'on traite des monoexponentielles, utiliser un algorithme plus rapide [Halmer 04]. Pour tester cet algorithme d'inversion, nous avons simulé un signal composé de 3 exponentielles tel que

$$y(t) = e^{(-t)} + e^{(-t/10)} + e^{(-t/100)} + 10^{-7} \varepsilon \quad (II.39)$$

avec un échantillonnage de  $\Delta_k = 1.4$ , qui est un cas de sous-échantillonnage pour la mesure du temps de relaxation le plus court. L'algorithme de Prony fournit alors les résultats suivants

$$y(t) = 1.002 e^{(-t/0.99998)} + 0.995 e^{(-t/9.991)} + 1.002 e^{(-t/99.49)} + 4.14 \cdot 10^{-4} \quad (II.40)$$

détaillés dans le tableau B.1 page 161 où  $R_2 = 1/T_2$  et où les amplitudes correspondantes sont déterminées avec l'algorithme de Levenberg-Marquardt [Nielsen 99, Nielsen 00a]. On remarque bien l'efficacité de cet algorithme sur cet exemple, notamment en ce qui concerne les résidus de ce fit reportés sur la figure II.9.

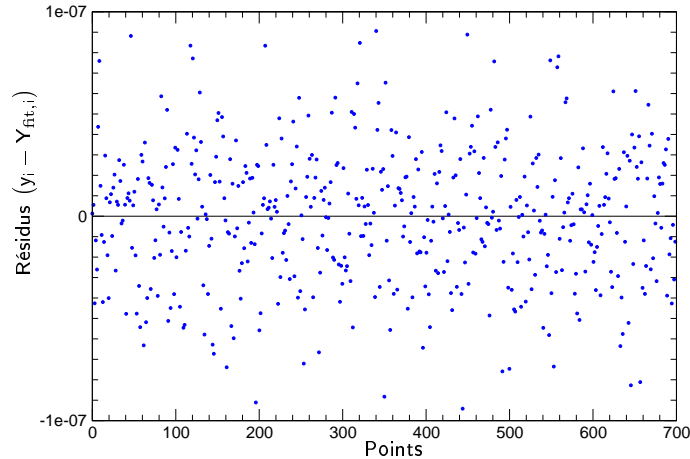


Figure II.9 : Comparaison entre modélisation par l'algorithme de Prony et la simulation

Il peut être intéressant de voir le résultat de CONTIN avec cet exemple. Sur la figure II.10 est représentée la distribution rendue par CONTIN et l'on remarque bien encore des problèmes de grande incertitude sur la détermination des temps courts alors que ceux-ci ne sont pas sous échantillonnés « de trop ». Sur la figure II.10, nous avons aussi représenté les résidus et la distribution obtenus avec CONTIN sur cet exemple. Ceci montre bien les limites de CONTIN sur ce type de cas.

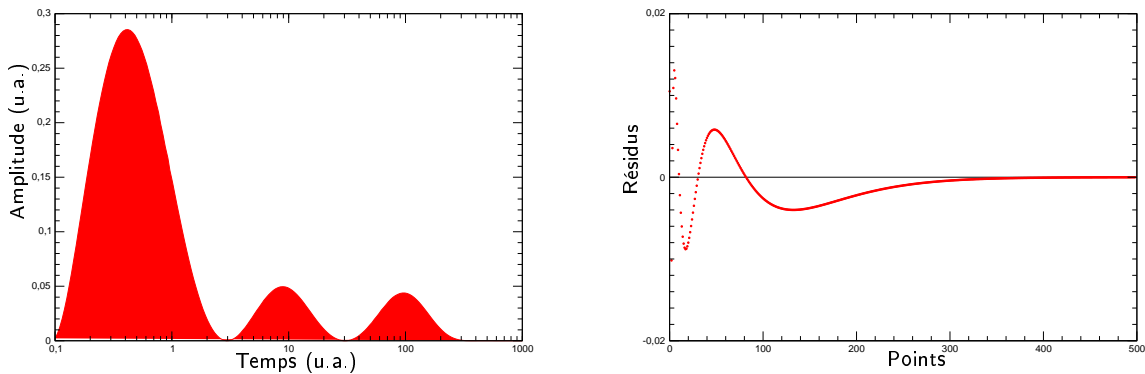


Figure II.10 : Résultat de CONTIN sur le test de l'algorithme de Prony

### c. Influence du bruit

Il est intuitif que le bruit « gêne » l'approximation faite avec les polynômes de Padé, ceux-ci n'étant pas assez précis. L'algorithme de Prony préconise de s'arrêter dans l'augmentation du degré des approximants de Padé  $[N - 1, N]_{Z[y_k]}$  lorsque les solutions apparaissent stables, dans ce cas  $N$  est le nombre d'exponentielles du signal. Si l'on augmente l'importance du bruit à  $10^{-3}$  dans l'exemple précédent (relation II.39), la stabilité apparaît entre  $q = 2$  et  $q = 3$  (Résultats détaillés dans le tableau B.2 page 161). Ainsi si l'on suit la logique de l'algorithme le nombre d'exponentielles se réduit donc à 2 pour ce cas. On peut essayer de pousser un peu le raisonnement en augmentant la qualité des approximants de Padé utilisés. On porte alors le degré des approximants de Padé à 30 et l'on obtient finalement les 3 temps de relaxation attendus.



Dans un premier temps, on pourrait se demander pourquoi ne pas modifier l'algorithme pour monter ainsi le degré jusqu'à trouver une solution acceptable. Tout d'abord, l'algorithme de Prony date un peu et les déterminants et racines de polynômes de degré 30 se déterminaient difficilement sans ordinateur. Ceci n'est plus vrai de nos jours avec la puissance des nouveaux ordinateurs et les nouveaux algorithmes de calcul adaptés. De plus, l'idéal serait d'automatiser l'algorithme : l'intervention d'un utilisateur ne devrait pas avoir lieu. L'appréciation du nombre de composantes exponentielles dans un signal ne devrait dépendre que du bruit. Ceci nous amène donc à la section suivante qui traite de la décomposition en valeurs singulières permettant de résoudre le système sur-déterminé.

#### 4. Singular Value Decomposition

La précédente relation II.34 donne accès aux coefficients des polynômes de Padé. Nous allons démontrer que le système linéaire basé sur la relation II.32 est, sous certaines conditions, sur-déterminé sauf cas particulier. On reprend la relation II.32 qui conduit à la relation suivante

$$(1 + b_1 z + \dots + b_q z^q) f(z) - (a_0 + a_1 z + \dots + a_p z^p) = O(z^{p+q+1}) \quad (\text{II.41})$$

en identifiant chaque coefficient, cette relation fournit un système linéaire qui définit les polynômes  $P$  et  $Q$  [Draux 80,p.186]

$$\left\{ \begin{array}{l} y_0 = a_0 \\ y_1 + b_1 y_0 = a_1 \\ \vdots \\ y_p + b_1 y_{p-1} + \dots + b_p y_0 = a_p \\ y_{p+1} + b_1 y_p + \dots + b_{p+1} y_0 = 0 \\ y_{p+2} + b_1 y_{p+1} + \dots + b_{p+2} y_0 = 0 \\ \vdots \\ y_{p+q} + b_1 y_{p+q-1} + \dots + b_q y_p = 0 \end{array} \right. \quad (\text{II.42})$$

avec la convention  $y_i = 0$  si  $i < 0$  et la condition d'existence

$$\left| \begin{array}{ccc} y_p & \dots & y_{p-q+1} \\ \vdots & & \vdots \\ y_{p+q-1} & \dots & y_p \end{array} \right| \neq 0 \quad (\text{II.43})$$

avec l'hypothèse que  $p < q - 1$  la relation II.42 peut être écrite comme un produit de matrice

$$\left[ \begin{array}{cccc} y_0 & & & \\ \vdots & \ddots & & \\ y_p & \dots & y_0 & \\ \vdots & & \vdots & \ddots \\ y_q & \dots & y_{q-p-1} & \dots & y_0 \\ \vdots & & \vdots & & \vdots \\ y_{p+q} & \dots & y_{q-1} & \dots & y_p \end{array} \right] \left[ \begin{array}{c} 1 \\ b_1 \\ \vdots \\ b_p \\ b_{p+1} \\ \vdots \\ b_q \end{array} \right] = \left[ \begin{array}{c} a_1 \\ a_2 \\ \vdots \\ a_p \\ 0 \\ \vdots \\ 0 \end{array} \right] \quad (\text{II.44})$$

Pour être compatible avec un nombre fini de données expérimentales, les deux paramètres  $p$  et  $q$  doivent vérifier la condition  $p + q \leq N_p$ . Pour utiliser la totalité des données expérimentales les paramètres  $p$  et  $q$  sont liés par la relation suivante

$$p + q = N_p \quad (\text{II.45})$$

Le système attendu est alors sur-déterminé. On peut écrire la relation II.44 où seulement  $b_i$  est impliqué :

$$\begin{bmatrix} y_p & \cdots & y_0 & & 0 \\ \vdots & & \vdots & \ddots & \\ y_{q-1} & \cdots & y_{q-p-1} & \cdots & y_0 \\ \vdots & & \vdots & & \vdots \\ y_{p+q-1} & \cdots & y_{q-1} & \cdots & y_p \end{bmatrix} \begin{bmatrix} b_1 \\ \vdots \\ b_q \end{bmatrix} = \begin{bmatrix} -y_{p+1} \\ -y_{p+2} \\ \vdots \\ -y_{p+q} \end{bmatrix} \quad (II.46)$$

Cependant la relation II.45 impose  $q \leq p$ , la dernière relation II.46 devient alors

$$\underbrace{\begin{bmatrix} y_{q-1} & \cdots & y_0 \\ \vdots & & \vdots \\ y_{N_p-1} & \cdots & y_{N_p-q} \end{bmatrix}}_{\mathbf{Y}} \underbrace{\begin{bmatrix} b_1 \\ \vdots \\ b_q \end{bmatrix}}_{\mathbf{b}} = \underbrace{\begin{bmatrix} -y_{N_p-q} \\ \vdots \\ -y_{N_p} \end{bmatrix}}_{\tilde{\mathbf{y}}} \quad (II.47)$$

Ce dernier système est alors sur-déterminé dans tous les cas, mis à part la cas limite où  $p = q = N_p/2$ . Cette technique a l'avantage d'utiliser la totalité des données expérimentales lors de la résolution du système sur-déterminé par SVD (Singular Value Decomposition) tandis que l'algorithme de Prony n'en utilise qu'une partie.

#### a. Décomposition en valeurs singulières du système

On rappelle que le système est défini avec  $\mathbf{Y} \in \mathbb{R}^{(N_p-q+1) \times q}$ . On suppose que l'on peut écrire ce dernier comme une décomposition en valeur singulière : soient  $\mathbf{U}$  et  $\mathbf{V}$  deux matrices orthogonales respectivement dans  $\mathbb{R}^{(N_p-q) \times (N_p-q)}$  et dans  $\mathbb{R}^{q \times q}$  ; et soit  $\mathbf{\Sigma}$  définie telle que

$$\begin{cases} \mathbf{Y} = \mathbf{U} \mathbf{\Sigma}^T \mathbf{V} \\ \mathbf{\Sigma} = \begin{cases} \text{diag } \sigma_i & \text{si } q = N_p/2 \\ \begin{bmatrix} \text{diag } \sigma_i \\ \mathbf{0}_{(N_p+1) \times (N_p-q+1)} \end{bmatrix} & \text{sinon} \end{cases} \end{cases} \quad (II.48)$$

Cette relation mène alors à l'équation matricielle suivante

$$\mathbf{Y} \mathbf{b} = \mathbf{U} \mathbf{\Sigma}^T \mathbf{V} \mathbf{b} = \tilde{\mathbf{y}} \quad (II.49)$$

Seule les valeurs singulières non-nulles sont retenues pour calculer les solutions du problème. L'ensemble des solutions correspondants aux valeurs singulières égales à zéro sont ignorées. Dans le cas d'un système sur-déterminé où  $\mathbf{\Sigma}$  possède plus de lignes que de colonnes, le système est résolu dans le sens des moindres-carrés, retournant la solution  $x$  qui minimise  $\|\mathbf{\Sigma} \mathbf{b} - \tilde{\mathbf{y}}\|_2$ .

L'algorithme que nous avons utilisé pour faire cette transformation est l'algorithme de Golub-Reinsch [Golub 96]. Comme nous l'avons vu, le bruit engendre une déformation des polynômes d'approximation et l'utilisation de la décomposition en valeurs singulières permet de restaurer suffisamment le signal afin d'affiner les polynômes d'approximation. Nous allons voir son action sur un exemple dans la suite.

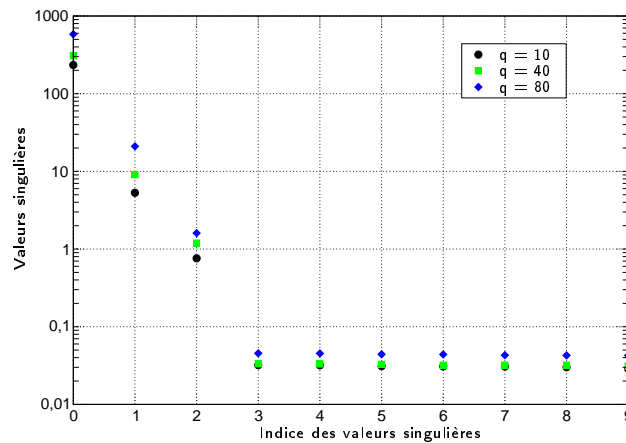
### b. Application à un exemple

Considérons le signal simulé suivant

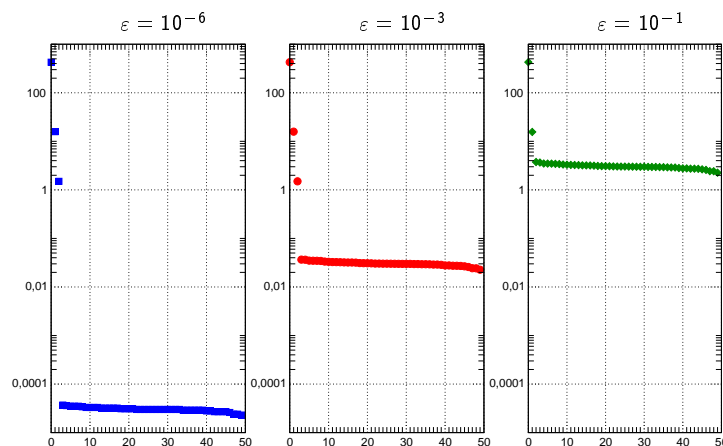
$$S(t) = 2e^{-t} + 4e^{-t/10} + 8e^{-t/100} + \varepsilon\mathcal{N}(0, 1) \quad (II.50)$$

Dans un premier temps, on fixe le bruit à  $\varepsilon = 10^{-3}$  et on augmente progressivement le degré de la SVD. On remarque alors que les valeurs singulières se détachent de plus en plus du bruit de fond (figure II.11). En effet, l'augmentation du degré des approximants tend à améliorer leur qualité et permet de mieux différencier les valeurs singulières du bruit de fond.

Dans un deuxième temps, on fixe le degré de la SVD à  $q = 50$  et on fait varier le bruit. Les valeurs singulières obtenues par la SVD se trouvent sur la figure II.12. On remarque alors que l'augmentation du bruit a tendance à atténuer la différence entre le bruit de fond et les valeurs singulières. À degré d'approximation constant, le bruit croissant rend leur qualité insuffisante au discernement des valeurs singulières.



**Figure II.11 :** Valeurs singulières pour l'amplitude du bruit simulé  $\varepsilon = 10^{-3}$  pour différentes valeurs de  $q$ . Les valeurs singulières sont classées par ordre décroissant

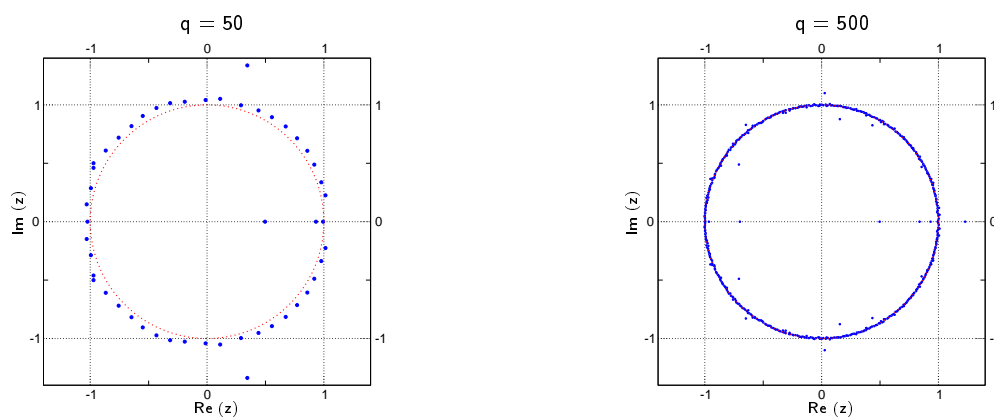


**Figure II.12 :** Valeurs singulières pour  $q = 50$  pour différents valeurs de bruit ( $\varepsilon$ )

### c. Racines des polynômes d'interpolation

Comme nous l'avons vu précédemment, les racines des polynômes d'approximation sont les solutions du problème, mais comme nous l'avons précisé dans le calcul de la transformée en  $Z$  du signal, le rayon de convergence de la série entière que caractérise la transformée en  $Z$  est de 1. Ainsi l'ensemble  $\mathcal{D}$  des pôles de cette transformée constitue, dans le plan complexe, le cercle unité ainsi que les pôles discrets réels représentant les temps de relaxation. Les solutions des polynômes d'approximation sont donc incluses dans cet ensemble.

Comme nous l'avons montré, nous évaluons par les approximants de Padé la transformée en  $Z$  du signal échantillonné de manière homogène, et donc sa transformée de Laplace qui est une fonction de Stieltjes (relation II.17). Gilewicz a conjecturé que les racines des approximants de Padé des séries de fonctions de Stieltjes convergeaient vers le domaine maximal de convergence de la série [Gilewicz 78,p.255]. Les résultats de la figure II.13 vont dans ce sens. Cette conjecture sera prouvée par la suite [Varga 94].



**Figure II.13** : Racines du polynôme  $P_f(z)$  pour  $q = 50$  et  $q = 500$ , les solutions recherchées se trouvent sur l'axe réel positif.

On remarque alors que l'ensemble des solutions possède toujours des éléments sur l'axe réel (tant que le degré des approximants est suffisant). Lorsque le degré  $q$  augmente les autres racines convergent vers le cercle unité ; par l'extérieur étant donné la condition convergence  $|\tilde{z}| > 1$ .

En augmentant le bruit à  $10^{-3}$ , il est nécessaire d'augmenter  $q$  à une valeur de  $q = 280$  (tableau B.3 page 162) pour obtenir des résultats satisfaisants, qui peuvent être ajustés par un fit non-linéaire qui doit être placé sous contraintes si le point de départ est trop éloigné de la solution.

### d. Retour sur le test de CONTIN

Retournons sur l'exemple du paragraphe iii. de la page 53. Les résultats obtenus par l'algorithme de SVD se trouvent dans le tableau B.4 page 162, nous avons utilisé la valeur maximale du degré  $q$ . Les résultats montrent les limites actuelles de l'algorithme. En effet, le temps court n'est pas détecté comme on peut le voir sur les résidus (figure II.14). Ceci prouve à nouveau qu'il faut attacher une grande importance à obtenir un signal correctement échantillonné et avec un rapport signal/bruit suffisant.

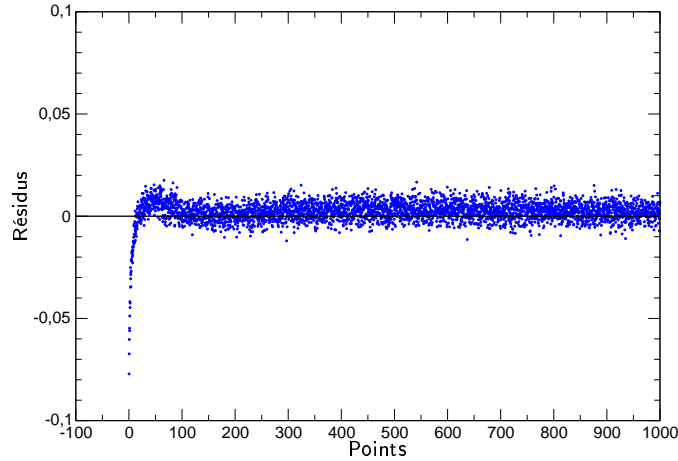


Figure II.14 : Résidus du fit obtenu par la SVD sur l'exemple de la page 53

## IV. Comparaison des algorithmes

### 1. Distribution discrète : comparaison entre Prony et Z-Padé/SVD

Pour confronter les deux méthodes, nous avons choisies de simuler une distribution discrète de 2 à 4 temps de relaxation suffisamment espacés pour être bien résolus (voir tableau II.1). Le paramètre d'échantillonnage est choisi suffisamment petit pour décrire les temps de relaxation les plus petits (voir page 46) :  $\tau_k = 0.02$ . Le nombre de points expérimentaux est lui aussi suffisant pour décrire les temps de relaxation les plus grands, il est choisi tel que :  $T_W = 5 \cdot \tau_{\max} = N_p \cdot \tau_k$ .

Indice	$\tau_k$	$A_k$
$k = 1$	0.1	0.1
$k = 2$	1.0	0.2
$k = 3$	10.0	0.3
$k = 4$	100.0	0.4

Tableau II.1 : Caractéristiques du signal de test. Les signaux simulés sont alors :  $s_i(t) = \sum_{k=1}^i A_k \exp(-t/\tau_k)$

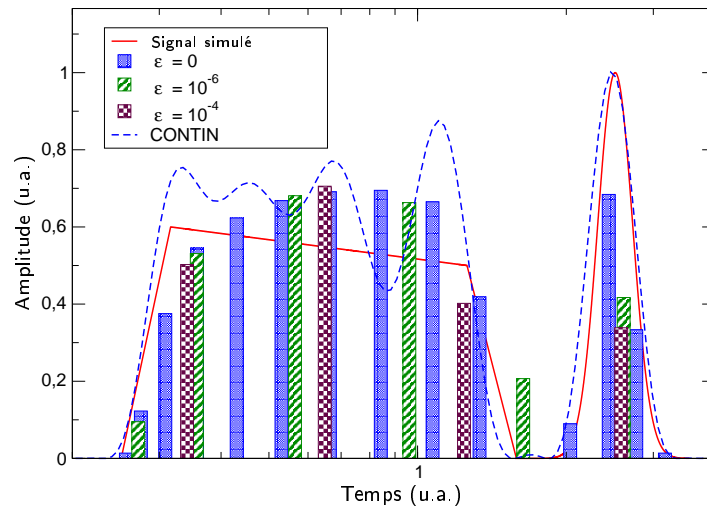
De plus, on ajoute un bruit gaussien d'amplitude  $\varepsilon$  rapporté à l'amplitude totale du signal (placé arbitrairement à 1). Les deux algorithmes ont été testés sur des signaux *rigoureusement* identiques. Les résultats se trouvent dans les tableaux B.5 et B.6 page 163. On remarque alors que l'efficacité de l'algorithme Z-Padé/SVD par rapport à celui de Prony, les solutions sont trouvés avec une meilleure précision et pour un bruit plus important.

### 2. Simulation d'une distribution continue

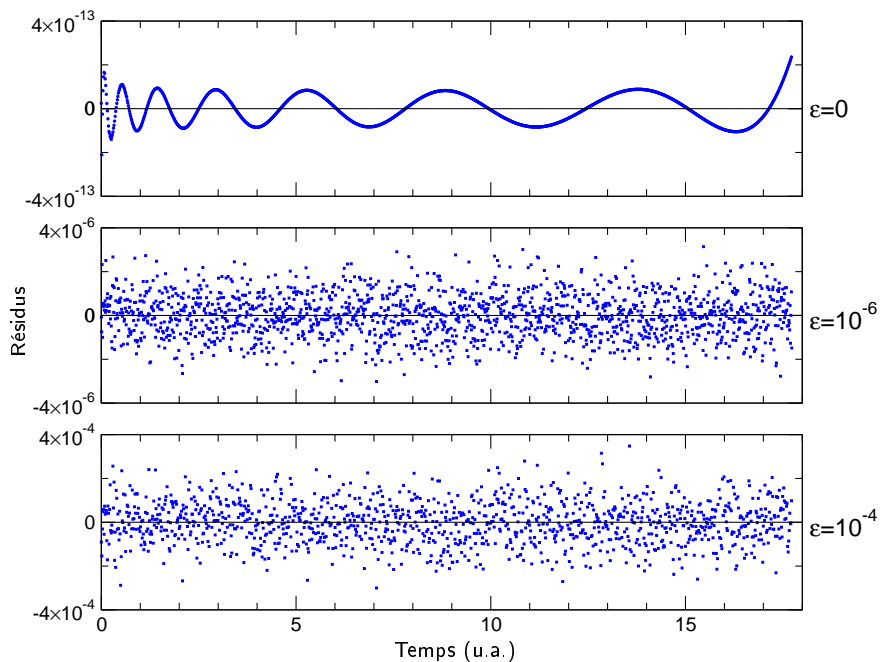
Jusqu'à présent nous n'avons testé les différents algorithmes que sur des distributions discrètes. Nous allons désormais les testés sur la distribution simulée de la figure II.1. Les résultats de l'algorithme Z-Padé/SVD se trouvent sur la figure II.15.

Comme on peut le remarquer : sans bruit ajouté le nombre de temps de relaxation est bien supérieur à ceux obtenu pour des distributions discrètes et les résidus de la figure II.16 montrent que les résultats sont en bon accord. Le fait que l'on n'obtienne pas une distribution de temps de relaxation bien plus fournie vient de l'impossibilité de résoudre deux temps de relaxation en rapport d'environ 1.6 et ceci sans bruit ajouté [Clayden 92]. Bien sûr, on peut remarquer que l'augmentation du bruit engendre

une diminution du nombre de temps de relaxation trouvés. Mais les résidus montrent encore une fois que la distribution discrète obtenue génère un signal en bon accord avec le signal simulé. Quant à la distribution obtenue avec CONTIN sur le signal non-bruité est sans appel, la présence d'oscillations n'a pas de sens.



**Figure II.15** : Distribution quasi-continue obtenue avec l'algorithme Z-Padé/SVD en comparaison avec la distribution simulée. En pointillé, le résultat obtenu avec CONTIN sur la même distribution simulée *sans* bruit ajouté.



**Figure II.16** : Résidus des résultats par Z-Padé/SVD.

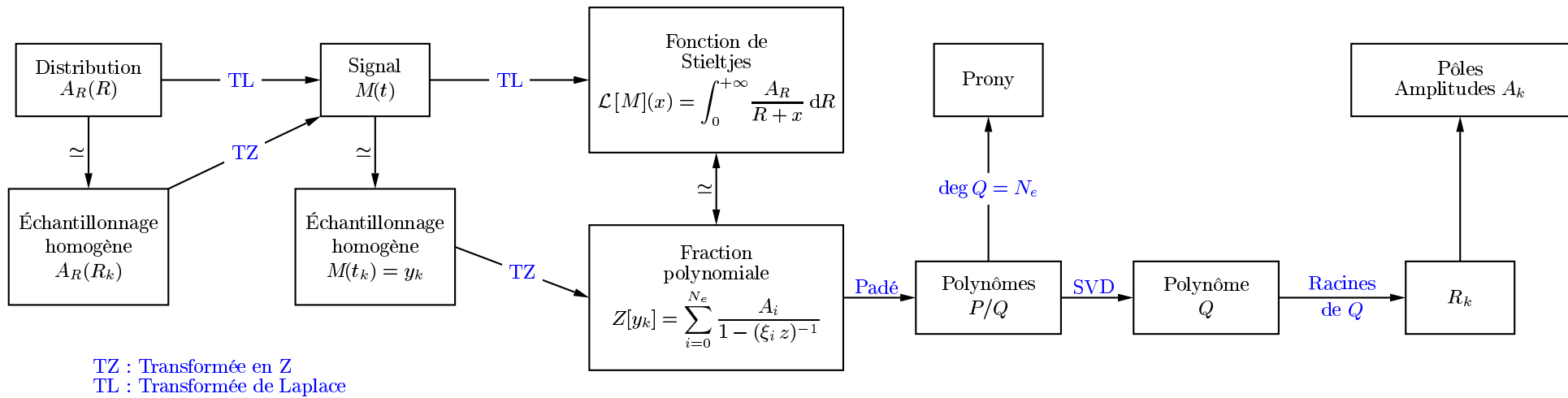
## V. Conclusion

L'inversion de la transformée de Laplace reste un défi mathématique. Il n'en demeure pas moins que le succès de notre démarche expérimentale repose sur la qualité de cette inversion. Nous avons détaillé les principaux algorithmes de recherche (figure II.17), d'autres existent.

Néanmoins, la modélisation de certains cas physiques réels par une distribution continue doit être faite avec la plus grande précaution. Il est préférable de considérer un signal composé de quelques exponentielles (distribution discrète) au lieu d'une distribution dont on sait qu'il n'est pas possible de la déterminer complètement.

D'autres travaux ont porté sur la détermination des distributions continues [Barone 01]. La plupart du temps la divergence des solutions est évitée par un régularisateur comme celui utilisé pour CONTIN. Bien que les simulations soient convaincantes, il est très difficile d'associer l'étendue de la distribution obtenue à des réalités physiques. Comme on a pu le voir sur l'exemple des distributions continues, la distribution discrète est souvent en très bon accord. Ce qui suppose que la modélisation par une distribution continue le sera aussi mais *quid* de l'interprétation de forme ? On s'en tiendra donc dans la suite à une description par des distributions discrètes. Et nous utiliserons essentiellement la méthode Z-Padé/SVD, celle-ci étant la plus efficace et la plus sûre. Bien que le fit non-linéaire par la méthode Levenberg-Marquardt soit suffisant dans la plupart des cas, ce dernier ne converge pas toujours.

D'un point de vue expérimental, les investigations que nous avons faites montrent qu'il faut connaître préalablement le système étudié. Et qu'il faut se garder de vouloir utiliser des distributions de temps de relaxation afin d'en tirer des conclusions sur les propriétés d'un système.



**Figure II.17** : Diagramme synthétique d'étude où sont représentées les correspondances entre les différentes fonctions utilisées. Le chemin d'étude part donc de la distribution  $A_R(R)$  (en haut à gauche) pour aboutir à la détermination des  $R_k$  et  $A_k$  (à droite).