

Filtrage adaptatif: Etude d'un système anti-bruit

Ph. MOUYON

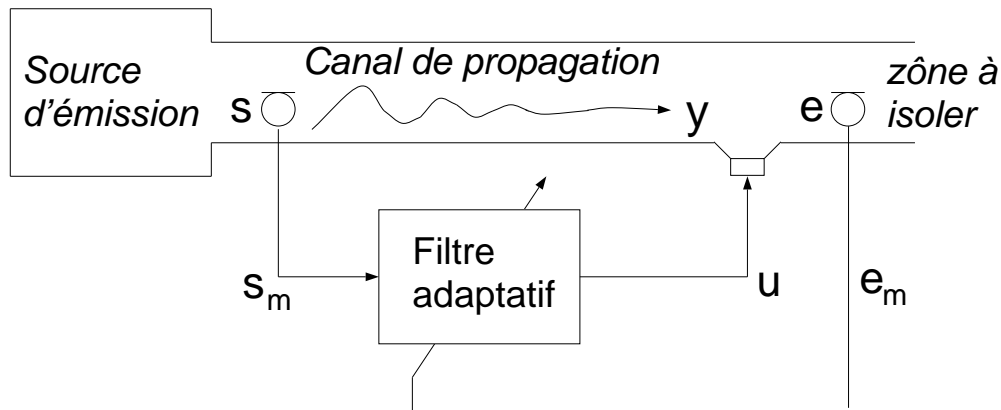
ONERA/DTIS, B.P. 4025, F31055 Toulouse

tél: 05.62.25.27.85, mél: philippe.mouyon@onera.fr

1 Introduction

1.1 Le dispositif physique

Le schéma de la figure (1) ci-après représente l'architecture générale d'un système de réduction adaptative du bruit acoustique en sortie d'un conduit.



Réjection adaptative du bruit acoustique dans les conduits

FIGURE 1 – Architecture générale

Le bruit incident s est émis par une source. Il se propage ensuite le long du conduit. On cherche à réduire le bruit transmis y en appliquant un contre-bruit u . Pour ce faire le conduit dans lequel le bruit se propage est instrumenté avec :

1. un microphone qui mesure la source s en amont de la zone à isoler acoustiquement.
2. un haut parleur chargé de créer le contre-bruit u .
3. un microphone de contrôle, qui mesure le bruit résiduel e entrant dans la zone à isoler, et permet ainsi d'adapter la commande.

1.2 Modélisation

La figure (2) montre le schéma bloc de l'ensemble du dispositif.

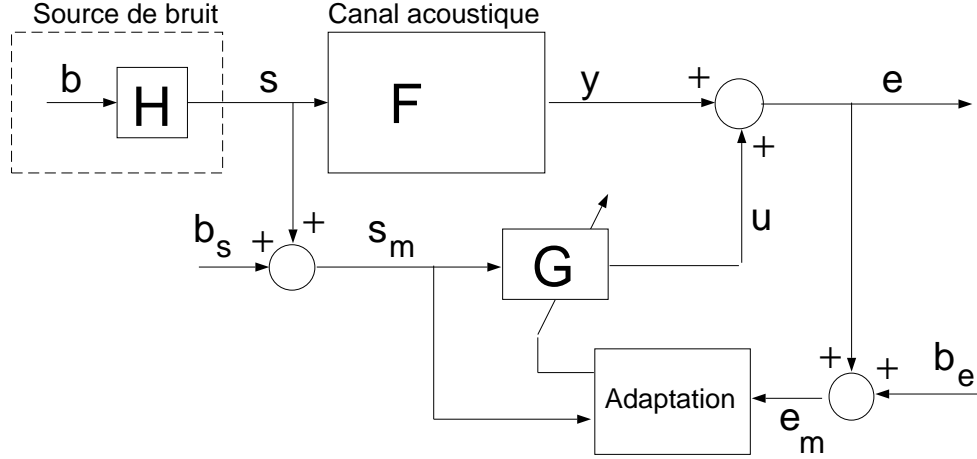


FIGURE 2 – Schéma bloc

Les différents éléments de ce schéma sont :

- Le bruit incident s est un signal à bande relativement étroite, dont la fréquence centrale peut varier sur une certaine plage. On le modélise comme un processus aléatoire obtenu en filtrant un bruit blanc b par un filtre passe bande : $s = H b$.
- Le conduit acoustique dont le comportement est modélisé par le filtre F . Sa sortie y est le bruit transmis qu'il faut essayer d'atténuer.
- Pour simplifier nous négligeons la dynamique des capteurs et de l'actionneur. De plus on suppose que la commande (i.e. l'anti-bruit u) s'applique directement au point de mesure du bruit résiduel, de sorte que :

$$e = y + u$$

- Les signaux mesurés à la cadence Δt sont :

$$\begin{aligned} s_m &= s + b_s \\ e_m &= e + b_e \end{aligned}$$

où b_s et b_e sont des incertitudes de mesure modélisées par des bruits blancs numériques centrés de variances respectives : $\sigma_{b_s}^2$ et $\sigma_{b_e}^2$.

- Le filtre réjecteur G qui génère l'anti-bruit u à partir de la mesure s_m , et son algorithme d'adaptation qui cherche à minimiser la mesure de l'erreur e_m .

Les performances de la commande anti-bruit seront mesurées par le rapport entre la puissance du bruit résiduel e et celle du bruit transmis y :

$$\text{Mesure de performance : } \rho = \sigma_e^2 / \sigma_y^2$$

1.3 Filtre réjecteur optimal

Il y a plusieurs manières de définir le filtre réjecteur optimal selon le contexte et les contraintes imposées.

- **Le filtre optimal, G^{opt} :**

C'est le filtre qui optimise la réjection de bruit connaissant toutes les caractéristiques du montage. Il dépend du modèle et des niveaux de bruit. Par exemple quand $\sigma_{b_s} = 0$, le filtre optimal est évidemment égal à $-F$. Quand $\sigma_{b_s} \neq 0$, son expression est un peu plus complexe.

Comme la physique est décrite à temps continu, le filtre optimal est un filtre défini à temps continu. Il n'est donc pas réalisable par un code numérique.

- **Le filtre FIR optimal, G^* :**

Pour pouvoir implémenter le filtre réjecteur dans un ordinateur il faut restreindre l'espace de recherche du filtre réjecteur optimal à l'ensemble des filtres numériques (IIR ou FIR).

Dans la suite on cherche l'optimum sous forme FIR car ces filtres sont faciles à adapter. Cette propriété est importante parce que les caractéristiques du conduit (et du bruit acoustique) sont mal connues et variables.

La valeur optimale de G , noté G^* , sera donc une approximation sous forme FIR du filtre optimal (le domaine admissible dans lequel on cherche l'optimum est l'ensemble des filtres FIR).

- **Le filtre FIR optimal, G_0^* :**

Quand $\sigma_{b_s} = 0$, la valeur optimale du filtre FIR réjecteur est donc simplement la meilleure approximation FIR de $-F$. La qualité de cette approximation sera d'autant meilleure que l'ordre du filtre sera grand.

1.4 Plan de travail

Le BE comporte différentes parties. Dans la première partie du BE vous étudiez le filtre G_0^* :

- son calcul à partir de F ,
- l'ordre nécessaire à une approximation de qualité,
- l'influence du pas d'échantillonnage Δt ,
- et les performances qu'il permet d'atteindre en simulation.

Il vous suffira pour cela d'examiner le code Matlab qui est fourni, et de commenter les résultats qu'il donne pour différents niveaux de bruit, et différents pas d'échantillonnage.

Dans la seconde partie vous étudiez le processus d'adaptation du filtre par l'algorithme du gradient stochastique (LMS et NLMS). Il vous faudra écrire les quelques lignes de programmation correspondantes. Et vous évalueriez les performances obtenues dans diverses configurations (niveaux de bruit, modèle de canal stationnaire ou non).

La troisième partie est relative à l'utilisation de l'algorithme des moindres carrés récurrents (RLS). Vous devez coder cet algorithme et le tester dans les mêmes configurations que précédemment.

Le code fourni comprend un programme principal **job** qui permet de :

- définir le système à simuler (structures *source*, *canal* et *filtreRIF*), la manière de le simuler (structure *simulation*). Ceci est fait à l'aide des macros *init_modele* et *init_filtre*.
- générer le filtre FIR G_0^* optimal pour $\sigma_{b_s} = 0$ (macro *syn_fir_adapte* appelée par *init_filtre*).
- spécifier les conditions d'un ensemble de simulations (structure *spec_sim*) à l'aide de la macro *specif_simu*.
- évaluer les performances par simulation. Ceci est fait par la macro *eval_perf* qui utilise le schéma de simulation *Simu_FIR.xls*.

2 Etude de G_0^*

1. Synthèse du filtre G_0^* :

En examinant la macro **syn_fir_adapte** déterminer quel est le lien entre G_0^* et le filtre $-F$.

Quel est le nombre N de points de la réponse impulsionnelle utilisée, comment N est calculé et pourquoi ?

Examiner l'influence du pas d'échantillonnage *filtreRIF.dt* sur la qualité de l'approximation de F .

2. Performances de G_0^* quand $\sigma_{b_s} = 0$:

En l'absence de bruit de mesure, vérifier sur une simulation le bon fonctionnement du filtre G_0^* . Que valent les puissances du bruit acoustique incident, transmis et résiduel ?

Pourquoi n'obtient-on pas un bruit résiduel parfaitement nul ?

3. Performances de G_0^* quand $\sigma_{b_s} > 0$:

On suppose que $\sigma_{b_e}^2 = 0$. On étudie l'évolution de ρ quand σ_{b_s} varie de 0 à environ $2\sigma_s$ (adapter/compléter *specif_simu* pour cela). La macro *eval_perf* trace la courbe automatiquement. La commenter.

Montrer que, pour $\sigma_{b_s}^2 = \sigma_s^2$, le filtre G_0^* ne donne pas les meilleures performances atteignables. Si $\sigma_{b_s}^2 = +\infty$, quel est le filtre optimal G_∞^* ?

Etudier aussi l'influence du pas d'échantillonnage sur les performances du filtre.

Quel est l'impact d'un σ_{b_e} non nul sur les performances du filtre ?

3 Adaptation par NLMS

L'algorithme d'adaptation doit ajuster le vecteur des coefficients de la réponse impulsionnelle du filtre réjecteur G , pour minimiser le bruit résiduel e . Dans cette partie vous étudiez l'adaptation par l'algorithme du gradient.

1. Algorithme du NLMS

- (a) Dans le schéma simulink, ouvrir la macro *Adapt_rif* et programmer l'adaptation du filtre réjecteur par l'algorithme du NLMS à pas constant. La réponse impulsionnelle du filtre G sera initialisée à zéro (commenter les lignes de *job* qui la fixaient à G_0^*). Indiquer que le filtre doit être adapté avec le NLMS (macro *init_modele*).
- (b) Simuler l'adaptation en l'absence de bruit de mesure ($\sigma_{b_e}^2 = \sigma_{b_s}^2 = 0$), avec un pas μ à choisir (à spécifier dans *spec_sim*). L'adaptation améliore-t-elle les performances ?
Est-ce que le filtre converge vers un filtre stationnaire ?

2. Analyse paramétrique pour le NLMS

- (a) Simuler maintenant l'adaptation quand $\sigma_{b_s}^2$ varie de 0 à σ_s^2 (mais toujours avec $\sigma_{b_e}^2 = 0$). Commenter l'évolution de ρ . Comparer avec les résultats obtenus sans adaptation.
- (b) Analyser l'influence du choix du pas d'adaptation μ sur les performances (toujours à $\sigma_{b_e}^2 = 0$, et pour quelques valeurs de σ_{b_s}).
- (c) Peut-on augmenter le pas d'échantillonnage sans trop dégrader les performances ?
- (d) Etudier la dégradation du fonctionnement lorsque $\sigma_{b_e}^2$ varie de 0 à σ_s^2 (pour quelques valeurs de μ et de $\sigma_{b_s}^2$).

3. Comportement en instationnaire

- (a) Le spectre de la source varie lentement (de 1 à 5 rd/s). Spécifier cette variation (macro **init_modele**).
- (b) Régler μ pour suivre cette variation de modèle. Quel compromis faut-il faire ?

4 Adaptation par RLS

1. Algorithme du RLS

Programmer l'algorithme du RLS avec facteur d'oubli λ pour adapter la réponse impulsionnelle du filtre anti-bruit.

Vérifier son bon fonctionnement pour une source stationnaire en l'absence de bruit de mesure ($\sigma_{b_s} = 0$, $\sigma_{b_e} = 0$).

Attention, il vous faudra modifier le schéma simulink pour introduire l'évolution de la matrice P . Bien spécifier sa valeur initiale (de dimensions correctes). Vous pouvez analyser l'influence de λ , qui remplace μ dans cet algorithme.

2. Analyse paramétrique pour le RLS

Réaliser la même analyse paramétrique que celle faite avec le NLMS. Comparer les performances obtenues pour une source stationnaire, et pour une source instationnaire.

5 Annexe

Dans la simulation, le bruit incident s est un signal à bande étroite. Il est obtenu par filtrage d'un bruit blanc normalisé (i.e. tel que $\Phi_{bb} = 1$) à travers un filtre $H(p)$ résonant (i.e. passe-bande peu amorti). On a choisi :

$$\textbf{Filtre générateur à bande étroite : } H(p) = \frac{\omega_1^2}{p^2 + 2\xi_1\omega_1 p + \omega_1^2}$$

où $\xi_1 = 0.01$. La fréquence centrale ω_1 peut varier sur une certaine plage (de 1 à 5 rd/s).

Le comportement du conduit acoustique entre s et y est modélisé par le transfert $F(p)$ suivant :

$$\textbf{Conduit acoustique résonnant : } F(p) = \frac{k\omega_2^2}{p^2 + 2\xi_2\omega_2 p + \omega_2^2}$$

où $\xi_2 = 0.3$ et $\omega_2 = 3$ rd/s. Il s'agit donc d'un système qui présente une résonance pouvant coïncider avec la fréquence centrale du bruit incident. C'est pour cela que le bruit incident est gênant.