

Rapport de soutenance intermédiaire

ST7 - Simulation à haute performance pour la réduction d'empreinte

CEA-DAM : Localisation de source

Jonathan POLI
Tom LABIAUSSE
Mathis PASTORELLI
Pierre LABOURE
Thibault ROUSSET

Février-Mars 2022

Contents

1	Introduction	1
2	Etude d'un cas simplifié	1
3	Ecart entre deux signaux - fonction de coût	3
4	Approche par un algorithme de recherche locale	5

1 Introduction

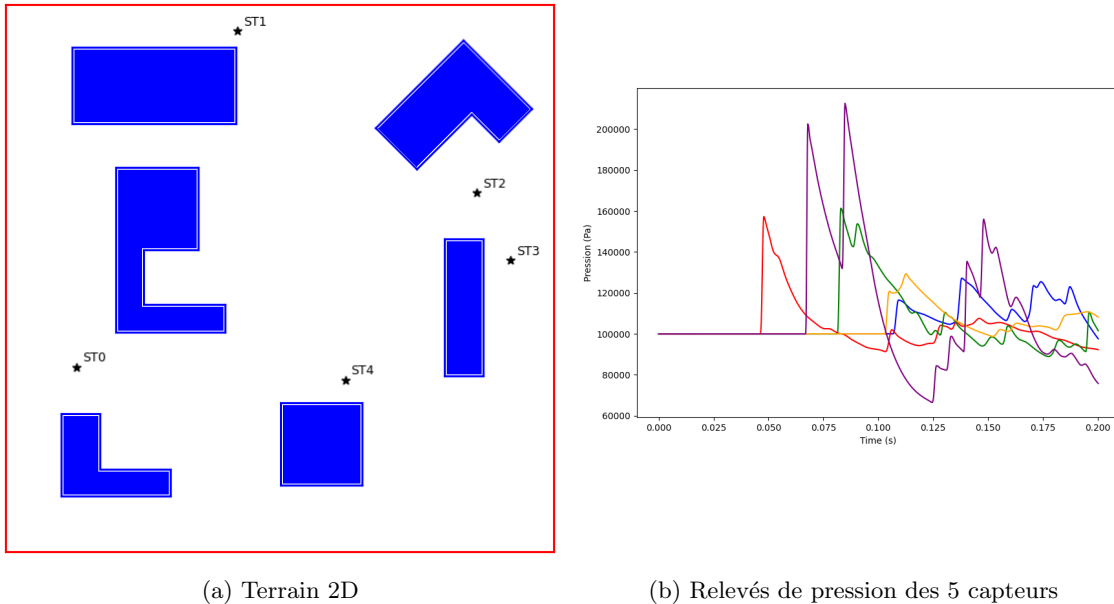


Figure 1: Problème de localisation de source

2 Etude d'un cas simplifié

Hypothèses simplificatrices

Afin de réaliser une première étude mathématique, nous allons faire plusieurs hypothèses très réductrices :

- L'onde propagée par l'explosion est isotrope et homogène, se déplaçant à une vitesse v connue, à savoir celle du son dans l'air : $v = 340m.s^{-1}$
- L'étude est réalisée en 2D et en négligeant la présence de bâtiments

Idée générale

Bien que le temps initial $t_{explosion}$ ne puisse pas être directement lue sur un capteur, il semble convenable de penser que 3 capteurs suffisent pour connaître l'emplacement exacte de la source, par triangulation.

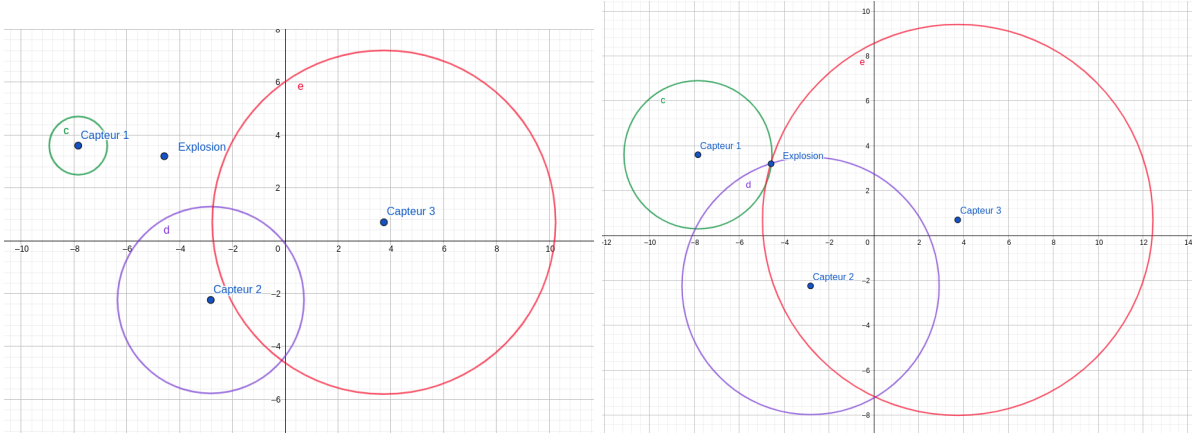
Démarche et preuve

Lors de l'explosion, l'onde va se propager dans toutes les directions à la vitesse v et ainsi être détectée par chacun des capteurs lorsque l'onde aura parcourue la distance la séparant du capteur.

En prenant le problème à l'envers, pour un capteur, c'est similaire à considérer que l'explosion a lieu au niveau du capteur, et attendre qu'elle parcoure la distance à la source d'origine.

Ainsi, en notant (x_i, y_i) les coordonnées de chaque capteur, (x, y) les coordonnées de la source de l'explosion, t_i la durée entre l'explosion et l'instant où l'explosion est détectée par chaque capteur, t_i^{mes} l'instant où l'explosion est détectée par chaque capteur et t_{exp} l'instant de l'explosion, on obtient :

$$\forall i, \begin{cases} (x_i - x)^2 + (y_i - y)^2 = (vt_i)^2 \\ t_i = t_i^{mes} - t_{exp} \end{cases}$$



(a) Cercles obtenus pour un certain t_{exp} non solution (b) Obtention de la solution par intersection des cercles

Figure 2: Recherche de la solution par inversion du problème avec 3 capteurs

Ensuite, par méthode graphique, on se rend compte qu'il existe une infinité de solutions pour 2 capteurs et un nombre fini non nul de solutions pour 3 capteurs (sauf s'ils sont alignés). Nous ne considérons pas le cas de 4 capteurs qui risque de nous diminuer le nombre de solutions à zéro à cause des hypothèses effectuées. Nous choisissons donc d'utiliser à chaque fois des trio de capteurs.

On obtient donc un système d'équations à trois inconnues : x , y et t_{exp} .

$$\begin{cases} (x_1 - x)^2 + (y_1 - y)^2 = v^2(t_1^{mes} - t_{exp})^2 \\ (x_2 - x)^2 + (y_2 - y)^2 = v^2(t_2^{mes} - t_{exp})^2 \\ (x_3 - x)^2 + (y_3 - y)^2 = v^2(t_3^{mes} - t_{exp})^2 \end{cases}$$

Résultats

Algorithme Nous allons sélectionner plusieurs trio de capteurs successivement. Pour chaque groupe de trois capteurs, nous allons résoudre le système d'équations obtenues par méthode numérique, en imposant $t_{exp} \leq t_i^{mes}$ afin de limiter les solutions physiquement invalides.

Les différents points (x, y) obtenus sont des possibles solutions approchées au problème initial.

Simulations Nous avons réaliser 10 simulations, 5 sans bâtiments et 5 avec des bâtiments afin d'observer la qualité des résultats.

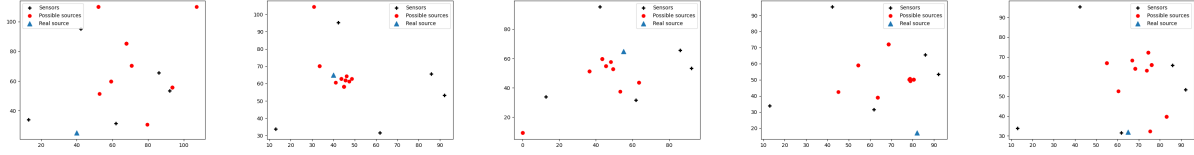


Figure 3: Simulations avec bâtiments

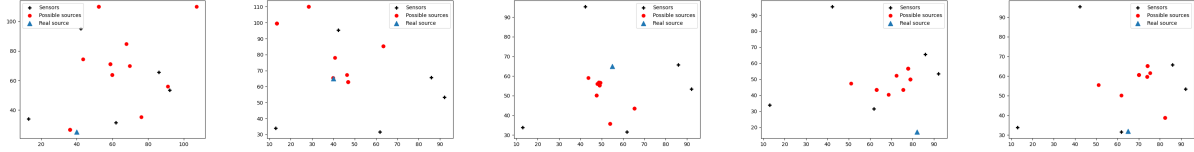


Figure 4: Simulations sans bâtiments

Analyse des résultats Les résultats sont parfois très bons, avec la grande majorité des points possibles proche de la véritable source, parfois un peu mauvais, avec au moins une source proche de la solution réelle et enfin mauvais, avec aucun point à proximité de la solution. Ensuite, de manière générale, les résultats sont plus précis lorsqu'il n'y pas de bâtiment, ce qui fait sens au vu des hypothèses.

Interprétation Plusieurs problèmes peuvent être à l'oeuvre :

- l'hypothèse de la vitesse constante $v = 340m.s^{-1}$ est possiblement très fausse.
- Le solveur numérique ne converge pas correctement vers la solution réelle
- Le solveur numérique converge possiblement vers une solution mathématiquement acceptable qui n'est pas la bonne physiquement.

3 Ecart entre deux signaux - fonction de coût

Afin d'évaluer la pertinence d'une estimation (X, Y) de la position de la source, nous n'avons d'autre choix que d'effectuer une simulation avec une source positionnée en (X, Y) . On compare ensuite les relevés de pression en simulation à ceux obtenus dans le cas réel. Il s'agit donc de pouvoir déterminer si deux points de l'espace 2D considéré sont proches en comparant uniquement deux séries de relevés de pression. Nous devons donc déterminer une fonction de coût permettant d'évaluer la "distance" entre deux relevés de pression.

Pour cela, il faut tout d'abord synchroniser les signaux. Nous nous basons donc sur une station qui sert de référence. De là, nous repérons les positions des premiers maxima des deux signaux sur chacune des station, ce qui permet, une fois que cela est fait, d'effectuer une comparaison sur les autres stations avec un même point de départ pour les signaux. Ainsi, deux signaux de formes identiques mais ayant été translatés temporellement n'auront pas une distance importante.

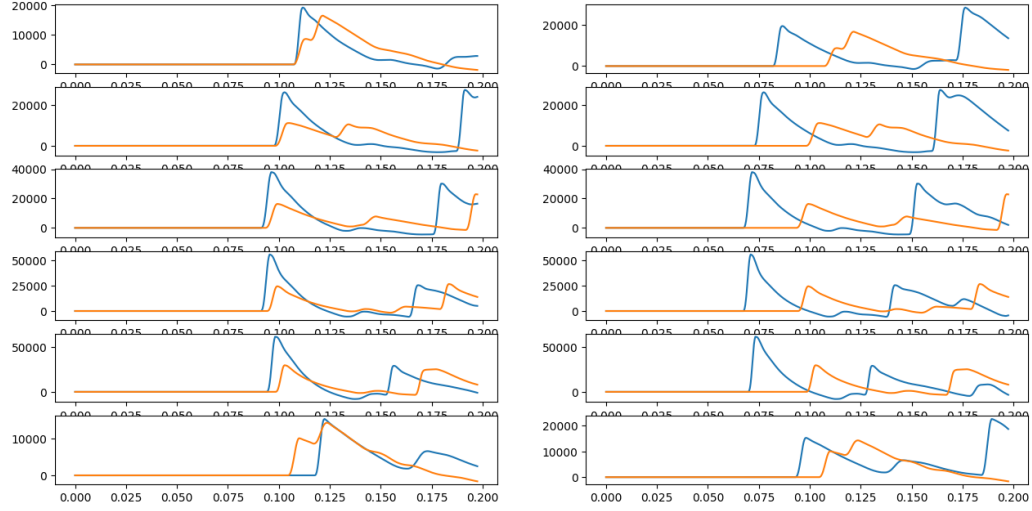


Figure 5: Synchronisation des signaux oranges sur les signaux bleus à partir du retard sur la première station

Il convient, une fois cela fait, de définir une distance entre deux signaux. Ce qui nous intéresse principalement ici est la position relative des formes (et maximas) des deux signaux. Pour effectuer une telle comparaison, nous calculons l'intercorrélation des deux signaux. Nous nous intéressons à ces caractéristiques car, a priori, deux signaux acoustiques captés par des sources espacées spatialement ne devraient (en l'absence de bâtiments) différer que par leurs abscisses. En présence de bâtiments, l'on peut supposer que si les deux points sont raisonnablement espacés, l'influence des bâtiments est minime sur la distorsion des signaux (sauf dans le cas d'un passage line-of-sight à non-line-of-sight, ce qui peut se produire sur certains capteurs mais, a priori, pas sur tous les capteurs en même temps)

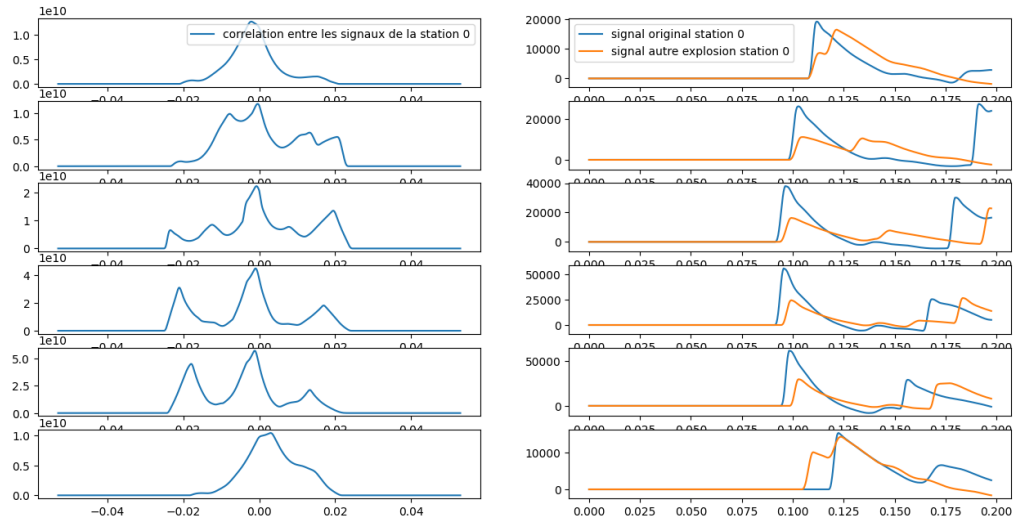


Figure 6: Calcul des corrélations entre les signaux synchronisés

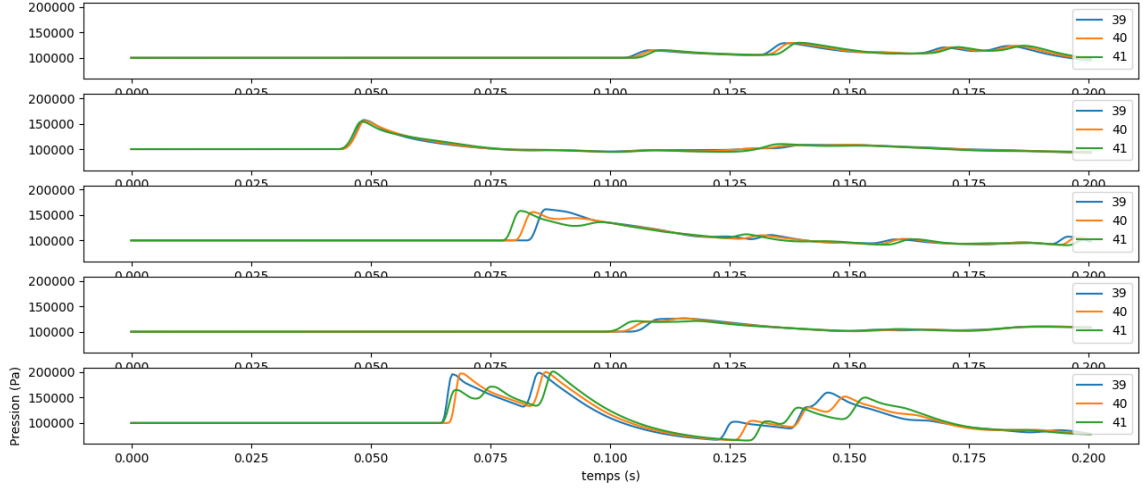


Figure 7: Explosions décalées d'un mètre les unes des autres (coordonnées : $y = 65m, x \in 39m, 40m, 41m$

On constate ici que la seule variation majeure observable entre les signaux est une différence temporelle : leur forme est similaire, mais plus deux signaux sont éloignés, plus le retard est important.

Nous estimons ensuite la distance entre les deux signaux via la norme 2 de la différence de l'intercorrélation et de l'autocorrélation du signal considéré comme le signal source. Ceci permet de mesurer la "ressemblance" du signal simulé vis-à-vis du signal mesuré.

$$d(x_{sim}, x_{mes}) = \int (\gamma_{x_{sim}, x_{sim}}(t) - \gamma_{x_{sim}, x_{mes}}(t))^2 dt$$

$$\text{où } \gamma_{x,y}(\tau) = \int x(t)y(t-\tau)dt$$

4 Approche par un algorithme de recherche locale

La première approche que nous avons décidée de mettre en oeuvre est un algorithme de recherche locale. Cette méthode s'appuie sur la notion de voisinage définie dans l'ensemble des solutions, qui est dans notre cas 2D un sous-ensemble borné de \mathbb{R}^2 . Afin de définir un tel concept de voisinage qui conviendrait à notre problème, nous avons discrétisé l'espace à l'aide d'un quadrillage régulier dont la finesse peut varier en ajustant la taille des mailles. Chaque noeud représentant ainsi un point de l'espace. La *Figure 8* présente un exemple d'une telle discrétisation.

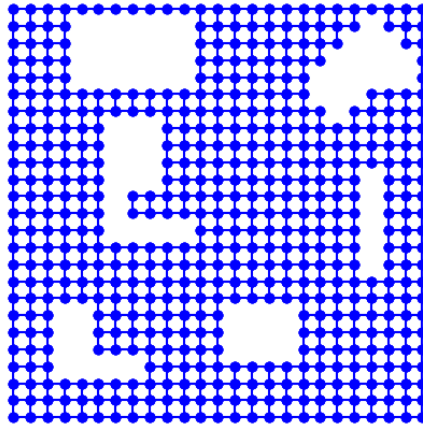


Figure 8: Discrétisation de l'espace 2D par une grille

Pour chaque point (X, Y) du maillage, on choisit ensuite de définir son voisinage comme étant l'ensemble des points avec lesquels il se trouve directement lié sur le quadrillage. La *Figure 9* donne un exemple où le point orange possède huit voisins verts. La présence de bâtiment supprimant des noeuds du maillage, un point peut posséder entre 2 et 8 voisins.

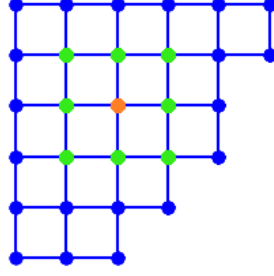


Figure 9: Exemple de voisinage

Le premier algorithme testé consiste à enchaîner des étapes de recherches exhaustives de la meilleure estimation de source parmi tous les voisins d'un point initial. Autrement dit on implémente une *greedy local search*. Cette procédure nécessite donc d'effectuer jusqu'à 8 simulations pour identifier le meilleur voisin d'un point. Une grande importance est donc accordée au temps d'exécution de chaque simulation. Nous avons donc tenté d'augmenter le pas spatial afin d'accélérer les simulations. Ainsi, à titre d'exemple, le temps d'exécution varie de 38s à 15s lorsqu'on augmente le pas de 0.4m à 0.8m. Cependant, comme on peut le voir en *Figure 10*, les signaux relevés sont affectés de manière non négligeable. Trop augmenter le pas risquerait donc de fausser les calculs de coût d'une solution empêchant ainsi la convergence de l'algorithme vers une solution optimale. On se limitera donc à un pas spatial de 60cm.

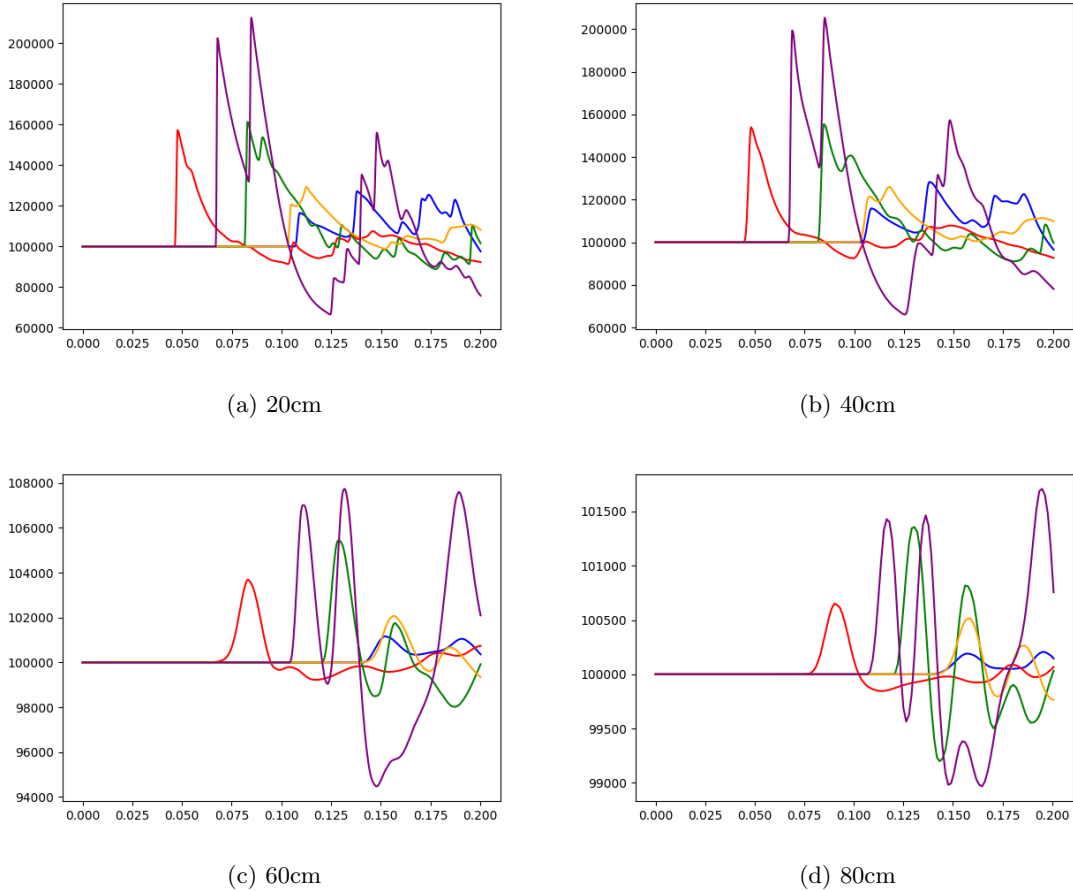
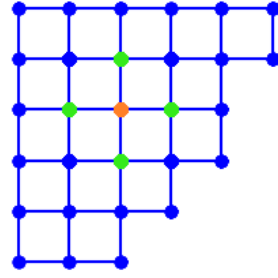
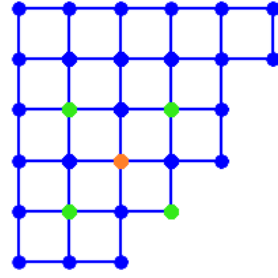


Figure 10: Exemples de relevés de pression avec différents pas spatiaux de simulation

Un autre moyen d'accélérer l'exécution de l'algorithme consiste à n'évaluer qu'un sous-ensemble des voisins d'un point donné. A titre d'exemple, au lieu d'évaluer les 8 voisins les plus proches, on peut se contenter d'en évaluer seulement 4 en choisissant (ou alternant) les deux styles d'évaluation suivants :



(a) Evaluation "droite"



(b) Evaluation "diagonale"