

# Manuel de référence du code SPPS

Institut français des sciences et technologies des transports, de l'aménagement et des réseaux

*Auteurs :*

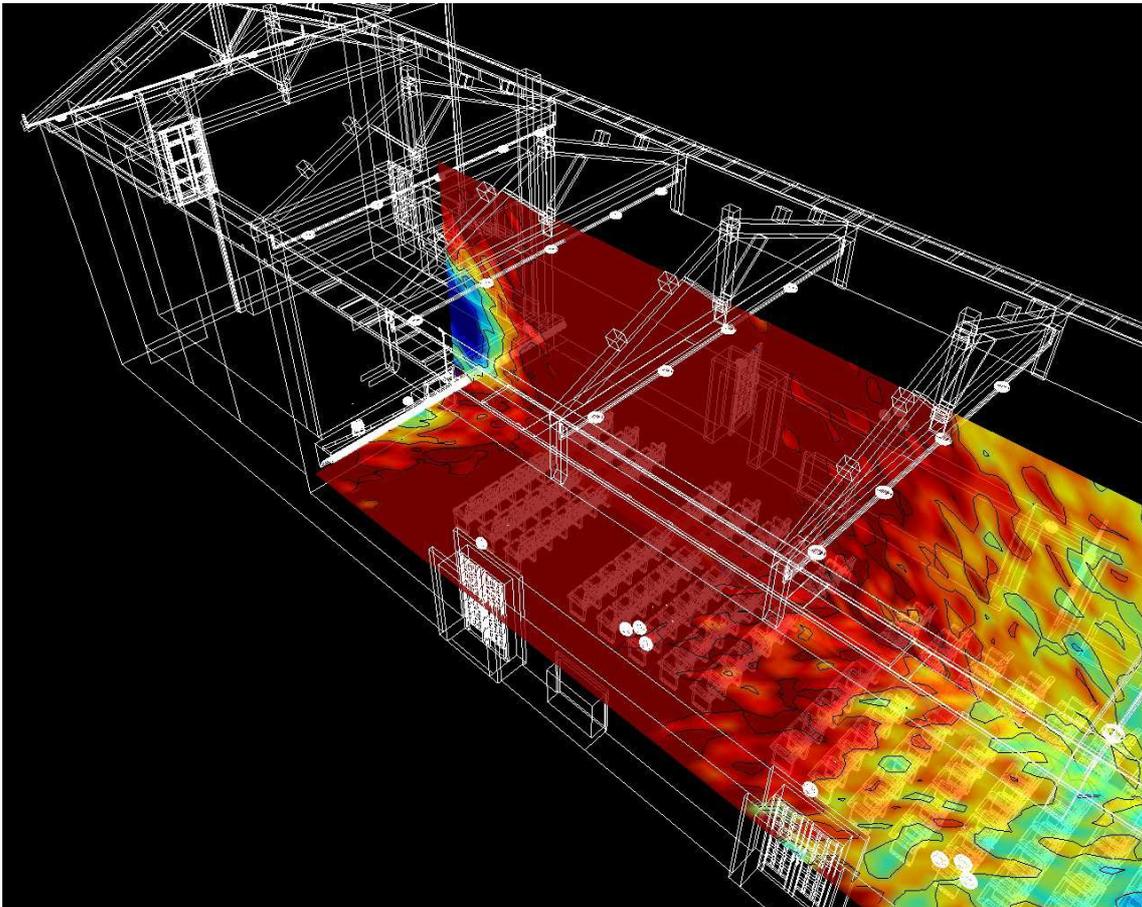
Judicaël Picaut et Nicolas Fortin

*Version du code :*

V2.1.1 du 7 juin 2011

*Date du document :*

10 juin 2011



# Table des matières

<b>Table des matières</b>	<b>2</b>
<b>Table des matières</b>	<b>4</b>
<b>Table des figures</b>	<b>8</b>
<b>Liste des tableaux</b>	<b>10</b>
<b>1 Historique du code</b>	<b>13</b>
<b>2 Démarrage rapide</b>	<b>15</b>
2.1 Présentation . . . . .	15
2.2 Propriétés et diffusion . . . . .	15
2.3 Installation . . . . .	15
2.4 Application . . . . .	15
2.5 Documentation . . . . .	16
2.6 Contact et information . . . . .	16
<b>3 Introduction</b>	<b>17</b>
3.1 Présentation . . . . .	17
3.2 Installation . . . . .	18
3.3 Principe général du code . . . . .	20
<b>4 Modélisation des phénomènes physiques</b>	<b>23</b>
4.1 Modélisation des sources sonores . . . . .	23
4.2 Modélisation de la propagation acoustique . . . . .	25
4.3 Modélisation des parois . . . . .	34
4.4 Calcul des niveaux sonores aux points d'observation . . . . .	41
<b>5 Description et fonctionnement du code SPPS</b>	<b>47</b>
5.1 Introduction . . . . .	47
5.2 Principe général du calcul . . . . .	47
5.3 Déroulement du calcul . . . . .	49
<b>6 Exécution du code</b>	<b>57</b>
6.1 Exécution avec l'interface I-Simpa . . . . .	57
6.2 Exécution en ligne de commande . . . . .	58
6.3 Exécution en langage Python™ . . . . .	59
6.4 Exécution en C++ . . . . .	59
<b>7 Format des fichiers d'entrée-sortie</b>	<b>61</b>
7.1 Liste et format des fichiers d'entrée/sortie . . . . .	61
7.2 Contenu et format du fichier de configuration . . . . .	61
7.3 Contenu et format du fichier de modèle géométrique .cbin . . . . .	72

7.4	Contenu et format du fichier du maillage du modèle .mbin . . . . .	72
7.5	Contenu et format des fichiers de résultat . . . . .	72
<b>8</b>	<b>Validation du code</b>	<b>81</b>
8.1	Principe . . . . .	81
8.2	Validation en acoustique des salles . . . . .	81
8.3	Validation en acoustique industrielle . . . . .	98
<b>9</b>	<b>Exemples d'application</b>	<b>107</b>
9.1	Utilisation de SPPS avec I-Simpa . . . . .	107
9.2	Utilisation de SPPS en ligne . . . . .	107
	<b>Bibliographie</b>	<b>117</b>
	<b>Acronymes</b>	<b>119</b>

# Table des matières

<b>Table des matières</b>	<b>2</b>
<b>Table des matières</b>	<b>4</b>
<b>Table des figures</b>	<b>8</b>
<b>Liste des tableaux</b>	<b>10</b>
<b>1 Historique du code</b>	<b>13</b>
<b>2 Démarrage rapide</b>	<b>15</b>
2.1 Présentation . . . . .	15
2.2 Propriétés et diffusion . . . . .	15
2.3 Installation . . . . .	15
2.4 Application . . . . .	15
2.5 Documentation . . . . .	16
2.6 Contact et information . . . . .	16
<b>3 Introduction</b>	<b>17</b>
3.1 Présentation . . . . .	17
3.1.1 Données d'entrée et de sortie du code SPPS . . . . .	17
3.1.1.1 Données d'entrée . . . . .	17
3.1.1.2 Données de sortie . . . . .	17
3.1.2 Phénomènes pris en compte . . . . .	17
3.1.3 Phénomènes non pris en compte . . . . .	18
3.1.4 Autres fonctionnalités de SPPS . . . . .	18
3.2 Installation . . . . .	18
3.2.1 Matériel requis . . . . .	18
3.2.2 Procédure d'installation de SPPS sous I-Simpa . . . . .	19
3.2.3 Procédure d'installation SPPS seul . . . . .	19
3.2.4 Librairies statiques utilisées par SPPS . . . . .	19
3.3 Principe général du code . . . . .	20
3.3.1 Concept des particules sonores . . . . .	20
3.3.2 Particules sonores et rayons sonores . . . . .	20
3.3.3 Principe du code . . . . .	21
3.3.4 Maillage spatial . . . . .	21
3.3.5 Maillage temporel . . . . .	22
<b>4 Modélisation des phénomènes physiques</b>	<b>23</b>
4.1 Modélisation des sources sonores . . . . .	23
4.1.1 Formalisme général . . . . .	23
4.1.2 Source omnidirectionnelle . . . . .	24
4.1.3 Source unidirectionnelle . . . . .	24
4.1.4 Sources « plan » (XY, YZ et XZ) . . . . .	25

4.2	Modélisation de la propagation acoustique . . . . .	25
4.2.1	Propagation acoustique . . . . .	25
4.2.2	Absorption atmosphérique . . . . .	26
4.2.3	Profil de célérité acoustique . . . . .	28
4.2.4	Diffusion par un encombrement . . . . .	30
4.3	Modélisation des parois . . . . .	34
4.3.1	Description physique . . . . .	34
4.3.1.1	Absorption, dissipation et transmission acoustique . . . . .	34
4.3.1.2	Diffusion acoustique . . . . .	35
4.3.2	Modélisation de la réflexion acoustique . . . . .	38
4.3.3	Modélisation des lois de réflexion . . . . .	38
4.3.3.1	Formalisme . . . . .	38
4.3.3.2	Réflexion spéculaire . . . . .	39
4.3.3.3	Réflexion uniforme . . . . .	39
4.3.3.4	Réflexion Lambert . . . . .	39
4.3.3.5	Réflexion normale en $w^n$ . . . . .	40
4.3.4	Modélisation de la transmission acoustique . . . . .	41
4.3.5	Vérification de la modélisation des parois . . . . .	41
4.4	Calcul des niveaux sonores aux points d'observation . . . . .	41
4.4.1	Niveau de pression acoustique au niveau d'un récepteur « volumique » . . . . .	42
4.4.2	Vecteur intensité au niveau d'un récepteur « volumique » . . . . .	43
4.4.3	Niveau de pression acoustique « latéral » au niveau d'un récepteur « volumique » . . . . .	43
4.4.4	Niveau d'intensité sonore sur un récepteur « surfacique » . . . . .	44
4.4.5	Niveau de pression acoustique sur un récepteur « surfacique » . . . . .	44
4.4.6	Niveau de pression acoustique global dans le modèle . . . . .	44
<b>5</b>	<b>Description et fonctionnement du code SPPS</b>	<b>47</b>
5.1	Introduction . . . . .	47
5.2	Principe général du calcul . . . . .	47
5.2.1	Pré-traitement du calcul grâce au maillage volumique du domaine . . . . .	47
5.2.2	Principe du code de calcul . . . . .	47
5.2.3	Structures . . . . .	48
5.2.3.1	Structure d'une maille . . . . .	48
5.2.3.2	Structure d'une face de maille . . . . .	48
5.2.3.3	Structure d'une particule sonore . . . . .	48
5.3	Déroulement du calcul . . . . .	49
5.3.1	Calcul parallèle sur les bandes de fréquence . . . . .	49
5.3.2	Boucle sur les sources sonores . . . . .	49
5.3.3	Boucle sur les particules sonores . . . . .	49
5.3.4	Suivi d'une particule sonore . . . . .	49
5.3.4.1	Trajectoire d'une particule . . . . .	49
5.3.4.2	Traitement des collisions . . . . .	50
5.3.5	Calcul des indicateurs acoustiques . . . . .	53
5.3.5.1	Récepteurs ponctuels . . . . .	53
5.3.5.2	Récepteurs surfaciques . . . . .	53
5.3.5.3	Énergie globale dans le modèle . . . . .	53
<b>6</b>	<b>Exécution du code</b>	<b>57</b>
6.1	Exécution avec l'interface I-Simpa . . . . .	57
6.2	Exécution en ligne de commande . . . . .	58
6.3	Exécution en langage Python™ . . . . .	59
6.4	Exécution en C++ . . . . .	59
<b>7</b>	<b>Format des fichiers d'entrée-sortie</b>	<b>61</b>
7.1	Liste et format des fichiers d'entrée/sortie . . . . .	61

7.2	Contenu et format du fichier de configuration . . . . .	61
7.2.1	Élément <configuration/> . . . . .	61
7.2.2	Élément <simulation/> . . . . .	61
7.2.3	Élément <freq_enum/> . . . . .	64
7.2.4	Élément <condition_atmosphérique/> . . . . .	64
7.2.5	Élément <surface_absorption_enum/> . . . . .	65
7.2.6	Élément <type_surface/> . . . . .	65
7.2.7	Élément <sources/> . . . . .	65
7.2.8	Élément <source/> . . . . .	65
7.2.9	Élément <recepteurss/> . . . . .	66
7.2.10	Élément <recepteur_surfacique/> . . . . .	66
7.2.11	Élément <recepteursp/> . . . . .	66
7.2.12	Élément <recepteur_ponctuel/> . . . . .	66
7.2.13	Élément <encombrement_enum/> . . . . .	66
7.2.14	Élément <encombrement/> . . . . .	66
7.3	Contenu et format du fichier de modèle géométrique .cbin . . . . .	72
7.4	Contenu et format du fichier du maillage du modèle .mbin . . . . .	72
7.5	Contenu et format des fichiers de résultat . . . . .	72
7.5.1	Organisation des fichiers de résultat . . . . .	72
7.5.1.1	Arborescence . . . . .	72
7.5.1.2	Détail de l'arborescence . . . . .	73
7.5.2	Fichiers pour un récepteur ponctuel . . . . .	75
7.5.2.1	Fichier pour le calcul du niveau de pression acoustique (.recp) . . . . .	75
7.5.2.2	Fichier pour le calcul des paramètres avancés (.gap) . . . . .	75
7.5.2.3	Fichier pour le calcul du vecteur « intensité » (.gabe) . . . . .	77
7.5.3	Fichiers de récepteur surfacique . . . . .	77
7.5.3.1	Fichier de récepteur surfacique (.csbin) . . . . .	77
7.5.4	Fichiers généraux . . . . .	77
7.5.4.1	Fichier de cumul dans le modèle (.recp) . . . . .	77
7.5.4.2	Fichier de suivi des particules (.pbin) . . . . .	77
7.5.4.3	Fichier de représentation des vecteurs « intensité » (.rpi) . . . . .	77
7.5.4.4	Fichier des statistiques du calcul (.gabe) . . . . .	79
<b>8</b>	<b>Validation du code</b> . . . . .	<b>81</b>
8.1	Principe . . . . .	81
8.2	Validation en acoustique des salles . . . . .	81
8.2.1	Salles de dimensions homogènes . . . . .	81
8.2.1.1	Comparaison avec les données de Yang (2000) . . . . .	81
8.2.1.2	Comparaison avec les données de Kuttruff (1997) . . . . .	82
8.2.1.3	Comparaison avec les données de Kang (2002) . . . . .	85
8.2.2	Couloirs . . . . .	89
8.2.2.1	Comparaison avec les données de Yang (2000) . . . . .	89
8.2.3	Locaux plats . . . . .	91
8.2.3.1	Comparaison avec les données de Korany (2001) . . . . .	91
8.2.3.2	Comparaison avec les données de Kuttruff (1997) . . . . .	93
8.2.3.3	Comparaison avec les données de Ondet (1989) . . . . .	93
8.2.4	Locaux couplés . . . . .	95
8.2.4.1	Salles de même dimension couplées par une cloison . . . . .	95
8.2.4.2	Salles de même dimension couplées par une porte . . . . .	96
8.2.4.3	Salles de dimension différentes couplées par une porte . . . . .	97
8.3	Validation en acoustique industrielle . . . . .	98
8.3.1	Comparaison avec les données de Ondet (1984,1988) . . . . .	98
8.3.2	Comparaison avec les données de Ondet (1989) . . . . .	102
<b>9</b>	<b>Exemples d'application</b> . . . . .	<b>107</b>

9.1	Utilisation de SPPS avec I-Simpa . . . . .	107
9.2	Utilisation de SPPS en ligne . . . . .	107
9.2.1	Principe . . . . .	107
9.2.2	Procédure . . . . .	107
9.2.3	Détail des scripts . . . . .	108
9.2.3.1	Script d'exécution . . . . .	108
9.2.3.2	Scripts de création des fichiers d'entrée . . . . .	108
9.2.3.3	Conversion des fichiers de résultat . . . . .	109
	<b>Bibliographie</b>	<b>117</b>
	<b>Acronymes</b>	<b>119</b>

# Table des figures

3.1	Arborescence de la distribution du code SPPS . . . . .	19
4.1	Géométrie de la source . . . . .	23
4.2	Distribution des angles d'émission pour une source omnidirectionnelle . . . . .	25
4.3	Simulation de la dispersion géométrique avec SPPS . . . . .	27
4.5	Organigramme de la prise en compte de l'absorption atmosphérique . . . . .	27
4.4	Modélisation de l'absorption atmosphérique . . . . .	28
4.6	Illustration de la prise en compte de l'absorption atmosphérique . . . . .	29
4.7	Schéma de principe de la réfraction acoustique . . . . .	31
4.8	Organigramme de la prise en compte de la réfraction acoustique . . . . .	31
4.9	Illustration de la modélisation de la réfraction acoustique avec SPPS . . . . .	32
4.10	Illustration de la diffusion par un encombrement . . . . .	33
4.11	Vérification de la prise en compte d'un encombrement . . . . .	35
4.13	Absorption et transmission acoustique dans une paroi . . . . .	35
4.12	Organigramme de la prise en compte de la diffusion par un encombrement . . . . .	36
4.14	Organigramme de la prise en compte de la réflexion/absorption par une paroi . . . . .	37
4.15	Géométrie élémentaire d'une réflexion par une paroi . . . . .	38
4.16	Lois de réflexion . . . . .	40
4.17	Vérification de la procédure de prise en compte des parois . . . . .	42
4.18	Niveau de pression acoustique pour un récepteur « volumique » . . . . .	43
4.19	Niveau d'intensité acoustique au niveau d'un récepteur « surfacique » . . . . .	44
5.1	Organigramme du code : boucle sur les fréquences . . . . .	50
5.2	Organigramme du code : boucle sur les sources sonores . . . . .	51
5.3	Organigramme du code : boucle sur les particules sonores . . . . .	52
5.4	Organigramme du code : suivi d'une particule sonore au cours du temps . . . . .	53
5.5	Suivi d'une particule sonore sur un pas de temps . . . . .	54
5.6	Organigramme du code : trajectoire d'une particule sur un pas de temps . . . . .	55
5.7	Organigramme du code : traitement des collisions . . . . .	56
7.1	Contenu du fichier de configuration. . . . .	61
7.2	Arborescence type des répertoires et fichiers de résultats issus du code SPPS . . . . .	72
8.1	Géométrie utilisée par Yang (2000) . . . . .	82
8.2	Comparaison des simulations SPPS avec les résultats de Yang (2000) . . . . .	83
8.3	Géométrie utilisée par Kuttruff (1997) pour des salles cubiques . . . . .	84
8.4	Comparaison des simulations SPPS avec les résultats de Kuttruff (1997) . . . . .	85
8.5	Géométrie utilisée par Kang (2002) . . . . .	85
8.6	Comparaison des simulations SPPS avec les résultats de Kang (2002) (EDT) . . . . .	87
8.7	Comparaison des simulations SPPS avec les résultats de Kang (2002) (SPL et EDT) . . . . .	88
8.8	Décroissances d'énergie dans un cube . . . . .	88
8.9	Géométrie utilisée par Yang (2000) . . . . .	89
8.10	Comparaison des simulations SPPS avec les résultats de Yang (2000) . . . . .	90

8.11 Géométrie utilisée par Korany (2001) pour des locaux plats . . . . .	91
8.12 Comparaison des simulations SPPS avec les résultats de Korany (2001) . . . . .	92
8.13 Géométrie utilisée par Kuttruff (1997) pour des locaux plats . . . . .	93
8.14 Géométrie utilisée par Ondet (1989) pour un local plat vide . . . . .	93
8.15 Comparaison de SPPS avec les mesures de Ondet [29] . . . . .	95
8.16 Deux salles couplées de dimension homogènes . . . . .	95
8.17 Deux salles couplées de dimension homogènes . . . . .	96
8.18 Deux salles couplées de dimension homogènes . . . . .	98
8.19 Hall d'usine ayant servi aux mesures réalisées par Ondet (1988) . . . . .	99
8.20 Géométrie utilisée par Ondet (1988) pour un local encombré . . . . .	100
8.23 Configurations étudiées par Ondet <i>et al</i> [29] dans un local plat avec encombrement . . . . .	102
8.21 Comparaison avec les résultats expérimentaux de Ondet (1984, 1988) (suite) . . . . .	103
8.22 Géométries des configurations étudiées par Ondet (1989) simulées avec le code SPPS . . . . .	104
8.24 Comparaison des données de Ondet (1989) pour un local plat encombré avec SPPS . . . . .	105

# Liste des tableaux

7.1	Extension et nature des fichiers d'entrée/sortie du code de calcul . . . . .	62
7.2	Structure du fichier de modèle géométrique . . . . .	73
7.3	Structure du fichier de maillage . . . . .	73
7.4	Structure du fichier .recp . . . . .	75
7.7	Série des paramètres de temps . . . . .	76
7.8	Série contenant la puissance acoustique cumulée des sources . . . . .	76
7.9	Série contenant la liste des bandes de fréquence . . . . .	76
7.10	Série contenant le bruit de fond . . . . .	76
7.11	Série contenant l'énergie sonore calculée . . . . .	77
7.14	Structure du fichier de cumul .recp . . . . .	77
7.15	Structure du fichier des trajectoires d'une quantité énergétique . . . . .	78
7.17	Série d'index du fichier .rpi. . . . .	78
7.18	Série de paramètres du fichier .rpi. . . . .	78
7.19	Série $x$ d'un récepteur ponctuel dans le fichier .rpi. . . . .	78
7.20	Série $y$ d'un récepteur ponctuel dans le fichier .rpi. . . . .	79
7.21	Série $z$ d'un récepteur ponctuel dans le fichier .rpi. . . . .	79
7.22	Structure du fichier des statistiques du calcul .gabe . . . . .	79
8.1	Configurations de Yang (2000) . . . . .	82
8.2	Configuration de Kuttruff (1997) pour des salles cubiques . . . . .	84
8.3	Comparaison des résultats de SPPS avec Kuttruff (1997) . . . . .	85
8.4	Configuration de Kang (2002) . . . . .	86
8.5	Résultats de Kang (2002) . . . . .	87
8.6	Configurations de Yang (2000) . . . . .	91
8.7	Configuration de Korany (2001) en champ mixte . . . . .	92
8.8	Configuration de Korany (2001) en champ diffus . . . . .	92
8.9	Comparaison des résultats de SPPS avec Korany (2001) . . . . .	92
8.10	Configuration de Kuttruff (1997) pour des locaux plats . . . . .	94
8.11	Configuration de Ondet (1989) pour un local plat vide . . . . .	94
8.12	Résultats de Ondet (1989) . . . . .	95
8.13	Configurations de locaux couplés par une ouverture . . . . .	96
8.14	Comparaison de SPPS avec la théorie classique des locaux couplés . . . . .	97
8.15	Configurations de locaux couplés par une ouverture . . . . .	97
8.16	Comparaison de SPPS avec la théorie classique des locaux couplés . . . . .	97
8.17	Comparaison de SPPS avec la théorie classique des locaux couplés . . . . .	98
8.18	Configuration de Ondet (1988) pour un local encombré . . . . .	100
8.19	Détail de la configuration de Ondet (1988) pour un local encombré . . . . .	101
8.20	Détail des configurations étudiées par Ondet (1989) . . . . .	104
8.21	Résultats de Ondet (1989) pour un local plat encombré . . . . .	104

# Listings

7.1	Descriptif général du fichier de configuration . . . . .	62
7.2	Descriptif de l'élément <configuration/> . . . . .	62
7.3	Descriptif de l'élément <simulation/> . . . . .	67
7.4	Descriptif de l'élément <freq_enum/> . . . . .	67
7.5	Descriptif de l'élément <condition_atmospherique/>. . . . .	68
7.6	Descriptif de l'élément <surface_absorption_enum/>. . . . .	68
7.7	Descriptif de l'élément <type_surface/> . . . . .	68
7.8	Descriptif de l'élément <sources/> . . . . .	68
7.9	Descriptif de l'élément <source/> . . . . .	69
7.10	Descriptif de l'élément <recepteurss/> . . . . .	69
7.11	Descriptif de l'élément <recepteur_surfacique/> . . . . .	69
7.12	Descriptif de l'élément <recepteursp/> . . . . .	69
7.13	Descriptif de l'élément <recepteur_ponctuel/> . . . . .	70
7.14	Descriptif de l'élément <encombrement_enum/> . . . . .	70
7.15	Descriptif de l'élément <encombrement/>. . . . .	71
9.1	Exemple de script Python™ d'exécution du code SPPS . . . . .	111
9.2	Exemple de script Python™ de génération de fichier de modèle . . . . .	112
9.3	Exemple de script Python™ de génération de fichier de maillage . . . . .	113
9.4	Suite du listing 9.3 . . . . .	114
9.5	Exemple de script Python™ de conversion d'un fichier GABE en CSV . . . . .	115
9.6	Suite du listing 9.5 . . . . .	116



# Historique du code

Le fichier `change.log` à la racine du répertoire d'installation de SPSS contient les informations sur les versions du logiciel :



# Démarrage rapide

## 2.1 Présentation

Le code SPPS est un code de lancer de particules pour des applications en :

- acoustique des salles ;
- acoustique du bâtiment ;
- acoustique industrielle ;
- acoustique extérieure.

Ce code permet :

- de modéliser un certain nombre de phénomènes propagatifs, tels que l'absorption, la transmission, la réflexion et la diffusion acoustique par des parois, la présence d'encombrements statistiques dans le domaine de propagation, la présence d'un gradient vertical de célérité du son, et l'atténuation atmosphérique ;
- de calculer des paramètres acoustiques (niveau sonore, vecteur intensité, intensité sonore) par bande de fréquence, pour des points d'observation dans le domaine de propagation ou sur des surfaces du domaine (cartographies) ;
- de suivre<sup>1</sup> le déplacement de particules sonores dans le domaine de propagation ;
- de choisir parmi deux algorithmes de calcul : un calcul « aléatoire » (calcul rapide mais moins précis, basé sur une méthode de Monte Carlo) et un calcul « énergétique » (calcul plus précis, mais plus long).

<sup>1</sup>Seul le fichier de données est créé, l'animation graphique en tant que telle doit être réalisée avec un autre logiciel.

## 2.2 Propriétés et diffusion

Le code SPPS est propriété de Ifsttar. Le développement du code a été réalisé par Nicolas Fortin (depuis 2007) et Judicaël Picaut (depuis 2001). La diffusion du code nécessite l'autorisation préalable de l'Ifsttar. Le fichier `licence.txt` à la racine du répertoire d'installation de SPPS décrit les termes de la licence du code SPPS.

## 2.3 Installation

La procédure d'installation est détaillée au paragraphe 3.2.

## 2.4 Application

Le code SPPS peut être utilisé :

- de manière autonome (en Python™, ou en C++) à l'aide de bibliothèques fournies (solution non conseillée). Un exemple d'utilisation est donné avec des scripts Python™ au chapitre 9 (voir également la *documentation de la bibliothèque LibSimpa*) ;
- à partir de l'interface I-Simpa (solution conseillée) développée par Ifsttar, dans le cadre d'un projet de recherche en collaboration avec les Universités de Poitiers et de La Rochelle. Le lecteur pourra se reporter au *manuel d'utilisation de l'interface I-Simpa*.

## 2.5 Documentation

La présente documentation<sup>2</sup> est destinée :

- à détailler le fonctionnement (modélisation des phénomènes physiques, principe et algorithmes de calcul) ;
- à apporter des éléments de validation du code ;
- à décrire l'utilisation du code (format des fichiers d'entrée et de sortie, exécution du code) ;
- à présenter un exemple d'utilisation du code ;

## 2.6 Contact et information

Pour toute information concernant le code SPPS, contacter :

Ifsttar Nantes

**Judicaël Picaut**

✉ Route de Bouaye, CS 4, F-44344 Bouguenais

☎ +33 (0)2 40 84 57 89

📠 +33 (0)2 40 84 59 92

✉ Judicael.Picaut@ifsttar.fr

<http://www.ifsttar.fr>

---

<sup>2</sup>Ce document n'est pas une documentation « développeur » (disponible par ailleurs).

# Introduction

## 3.1 Présentation

Le code de calcul SPPS est destiné à la prévision acoustique dans des domaines de propagation de formes complexes (salles, bâtiments, halls industriels, milieu urbain) en trois dimensions. Ce code étant basé sur le concept des particules sonores, il repose sur les hypothèses de l'acoustique géométrique (mêmes hypothèses que les modèles de lancer de rayons sonores), à savoir une approche « haute fréquence » (*i.e.* la longueur d'onde est petite devant les obstacles et les distances de propagation) et énergétique (*i.e.* on suppose que le champ sonore est incohérent, donc sans considérer les phénomènes d'interférence).

### 3.1.1 Données d'entrée et de sortie du code SPPS

#### 3.1.1.1 Données d'entrée

- la géométrie du modèle ;
- les propriétés acoustiques du modèle (conditions aux limites) ;
- les propriétés du domaine de propagation ;
- la position et les propriétés des sources sonores ;
- la position des points d'observation (récepteurs ponctuels) ;
- les plans de réception pour la représentation des cartographies (récepteurs surfaciques) ;
- les paramètres de calcul.

#### 3.1.1.2 Données de sortie

- le niveau de pression acoustique et d'intensité pour chaque point d'observation, pour chaque bande de fréquence, au cours du temps ;
- le niveau d'intensité et de pression acoustique pour chaque plan de réception, pour chaque bande de fréquence, au cours du temps ;
- le niveau sonore global dans l'ensemble du modèle, pour chaque bande de fréquence, au cours du temps.

### 3.1.2 Phénomènes pris en compte

Le code SPPS permet de modéliser les phénomènes propagatifs suivant :

- **Émission acoustique** : Seules des sources ponctuelles sont modélisées dans le code SPPS. Ces sources peuvent être caractérisées par une directivité propre (*cf.* paragraphe 4.1) ;
- **Propagation acoustique** : La modélisation de la propagation acoustique est inhérente au concept de particule sonore. La dispersion géométrique est automatiquement modélisée (*cf.* paragraphe 4.2.1) ;
- **Absorption atmosphérique** : La procédure de prise en compte de l'absorption atmosphérique intègre le calcul du coefficient d'absorption atmosphérique suivant la norme ISO 9613-1 (*cf.* paragraphe 4.2.2) ;
- **Gradient vertical de célérité** : Le code SPPS autorise la prise en compte d'un gradient vertical de célérité suivant un profil de type « log-lin » ;

- **Absorption acoustique par une paroi** : L'absorption acoustique par une paroi est considérée à travers le coefficient d'absorption  $\alpha$  de la paroi (cf. paragraphe 4.3.2) ;
- **Réflexion acoustique par une paroi** : Le code SPPS permet de modéliser le phénomène de réflexion acoustique par une paroi en considérant des réflexions mixtes, la paroi étant alors caractérisée par un coefficient de diffraction (cf. paragraphe 4.3.2). En plus de la réflexion spéculaire et de la réflexion diffuse de type Lambert, d'autres lois de réflexion sont également permises par le code SPPS ;
- **Transmission acoustique à travers une paroi** : La transmission acoustique à travers une paroi est modélisée par le biais du coefficient d'affaiblissement de la paroi (cf. paragraphe 4.3.4) ;
- **Diffusion acoustique par un encombrement** : Lorsque le domaine de propagation est caractérisé par une zone constituée de nombreux obstacles, cette zone est le siège de réflexions multiples (i.e. diffusion acoustique) et d'absorption. Dans le code SPPS, la diffusion par un encombrement est modélisée par un processus aléatoire (cf. paragraphe 4.2.4).

### 3.1.3 Phénomènes non pris en compte

Le code SPPS ne modélise pas toutes les conditions d'émission acoustique, de propagation acoustique, et retourne un nombre limité d'indicateurs acoustique. En particulier, il est important de préciser que le code SPPS ne prend pas en compte :

- **les phénomènes interférentiels** : par construction, compte-tenu de l'hypothèse « haute fréquence », les phénomènes interférentiels ne sont pas modélisés ;
- **la diffraction acoustique** : aucun phénomène de diffraction (diffraction par un panneau, par une arête...) n'est modélisé ;
- **l'effet d'un écoulement complexe** : l'effet d'un écoulement (comme le vent) n'est pas modélisé directement, hormis par le biais d'un profil de célérité vertical de type « log-lin » ;
- **la turbulence atmosphérique** ;
- **la directivité de sources acoustiques réelles** : la version actuelle modélise uniquement des sources avec une directivité « académique », à savoir omnidirectionnelle, unidirectionnelle, ou 2D dans un plan  $XY$ ,  $YZ$ , et  $XZ$  (dans le repère du modèle). En particulier, le format CLF n'est pas supporté<sup>1</sup>.) ;
- **les sources acoustiques en mouvement** : le code SPPS permet de modéliser un retard au déclenchement d'une source, mais pas une source en mouvement ;
- **les sources acoustiques permanentes** : le code SPPS ne modélise qu'une source de type impulsionnel. Pour obtenir, les résultats correspondants à une source permanente, il est nécessaire d'intégrer la réponse impulsionnelle au point d'observation considéré.

### 3.1.4 Autres fonctionnalités de SPPS

- calcul partagé : chaque calcul par bande de fréquence fait l'objet d'un processus unique. Le nombre simultané de processus est fonction du nombre de microprocesseurs disponibles sur la station de calcul ;

## 3.2 Installation

### 3.2.1 Matériel requis

Le code SPPS fonctionne sur un ordinateur ayant la configuration matérielle minimale suivante :

- Processeur Multi-cœur 32 bits : le code SPPS est parallélisé ; chaque cœur permet d'effectuer un calcul pas bande de fréquence en parallèle ;
- Fréquence d'horloge : Pas de minimum requis. La durée d'une simulation est toutefois dépendante de la fréquence d'horloge ;
- Mémoire vive : 128 Mo ;
- Espace disque nécessaire : 500 Mo recommandé. L'espace disque nécessaire est fonction de la configuration géométrique étudiée, des paramètres du calcul et des données de sorties demandées par l'utilisateur.

Le code SPPS a été testé sur les plateformes suivantes :

- PC avec Windows XP Pro 32 bits SP3 ;

<sup>1</sup><http://www.clfgroup.org/>

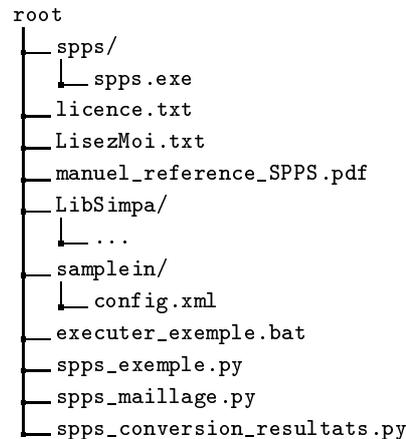
### 3.2.2 Procédure d'installation de SPPS sous I-Simpa

Le code SPPS est intégré directement à l'interface I-Simpa (cf. paragraphe 6.1). Aucune procédure d'installation est requise.

### 3.2.3 Procédure d'installation SPPS seul

Le code de calcul SPPS est diffusé sous forme de plusieurs fichiers (figure 3.1) :

- l'exécutable du code `spps.exe` dans le répertoire `spps/` ;
- un fichier de licence `licence.txt` à la racine ;
- un fichier d'information `LisezMoi.txt` à la racine ;
- le présent manuel de référence `manuel_reference_SPPS.pdf` à la racine ;
- un ensemble de fichiers dans le répertoire `LibSimpa` : librairies `Boost::Python-vc90-mt2` et `LibSimpa` nécessaires à l'utilisation de l'exemple d'application du code SPPS en Python™ ;
- un exemple d'application (voir le chapitre 9 pour son utilisation) :
  - un fichier de configuration `config.xml` dans le répertoire `samplein/` ;
  - plusieurs scripts en langage Python™ dans le répertoire racine :
    - \* un fichier de commande pour l'exécution automatique de l'exemple : `executer_exemple.bat`
    - \* un script principal `spps_exemple.py` pour l'exécution manuelle de l'application ;
    - \* des fonctions associées au script principal :
      - un script `spps_modele.py` pour la création d'un modèle géométrique ;
      - un script `spps_maillage.py` pour la création d'un maillage associé au modèle géométrique ;
      - un script `spps_conversion_resultats.py` pour la conversion des fichiers de résultat ;



**Figure 3.1** — Arborescence de la distribution du code SPPS. Le contenu du répertoire `LibSimpa` n'est pas détaillé. Le répertoire `root` désigne le répertoire d'installation choisi par l'utilisateur.

### 3.2.4 Bibliothèques statiques utilisées par SPPS

Le code SPPS est compilé avec les bibliothèques suivantes :

`lib_interface.lib` (Version C++, © Ifsttar)  
 Cette bibliothèque contient les formats des fichiers d'échange avec le logiciel d'interface utilisateur I-Simpa ;

`Boost` La bibliothèque `Boost`<sup>3</sup> regroupe un ensemble de bibliothèques C++. L'ensemble de la bibliothèque `Boost` est distribuée conformément aux termes de la licence *Boost Software License*<sup>4</sup> :

`Boost::Thread` Cette bibliothèque<sup>5</sup> permet d'exécuter simultanément plusieurs tâches (processus) en garantissant la synchronisation inter-processus lors de l'accès à des données partagées. Dans SPPS, cette bibliothèque est utilisée de manière à réaliser des calculs sur plusieurs bandes de fréquence simultanément ;

`Boost::Random::lagged_fibonacci` Cette bibliothèque<sup>6</sup> permet de générer des nombres pseudo-aléatoires suivant l'algorithme de Fibonacci. Dans SPPS, cette bibliothèque est utilisée de manière à générer des nombres aléatoires suivant une distribution uniforme, de manière à modéliser différents phénomènes propagatifs (cf. chapitre 4) ;

<sup>2</sup><http://www.boost.org/>

<sup>3</sup><http://www.boost.org/>

<sup>4</sup><http://www.boost.org/users/license.html>

<sup>5</sup>[http://www.boost.org/doc/libs/1\\_35\\_0/doc/html/thread.html](http://www.boost.org/doc/libs/1_35_0/doc/html/thread.html)

<sup>6</sup>[http://www.boost.org/doc/libs/1\\_35\\_0/libs/random/index.html](http://www.boost.org/doc/libs/1_35_0/libs/random/index.html)

### 3.3 Principe général du code

#### 3.3.1 Concept des particules sonores

À l'inverse de la théorie acoustique classique où l'étude du champ sonore s'appuie sur la propagation d'une onde dans un milieu matériel continu, l'approche utilisée dans le code SPPS est géométrique, et plus particulièrement particulière [18]. Dans un domaine de propagation complexe, le champ sonore est alors décomposé en une multitude de particules élémentaires, appelées particules sonores ou phonons, sans interaction mutuelle et porteuses d'une énergie infinitésimale et constante au cours du temps. Ces particules se propagent à la vitesse du son, soit en ligne droite (atmosphère homogène), soit avec des trajectoires courbes (en présence d'un profil de célérité, de turbulence atmosphérique...), entre deux chocs successifs avec les obstacles et limites du milieu de propagation. Lors d'une collision avec un obstacle ou une limite du domaine de propagation, les particules peuvent être absorbées ou réfléchies dans une nouvelle direction de propagation. L'acoustique géométrique devient ainsi un cas particulier de la dynamique des particules, de telle manière qu'un champ sonore peut être assimilé à un gaz de particules sonores. Dans ces conditions, la répartition de l'énergie du champ sonore est assimilée à la répartition des particules sonores. La densité locale d'énergie sonore étant proportionnelle à la densité locale de phonons, la seule difficulté réside dans la détermination de la répartition de ces particules sonores au cours du temps. Bien que des approches analytiques puissent être envisagées dans certaines conditions [27], nous avons choisi ici une approche entièrement numérique. Avec l'évolution des puissances de calcul, la simulation numérique d'un système composé de plusieurs centaines de millier de particules n'est plus une difficulté majeure.

#### 3.3.2 Particules sonores et rayons sonores

Ce concept est relativement similaire aux méthodes classiques de tracé de rayons sonores, implantées dans la plupart des logiciels actuels de prévision de bruit en milieu fermé ou ouvert. Néanmoins, même si dans la forme ces deux méthodes sont comparables, les différences majeures se situent au niveau de la gestion de l'énergie sonore portée par les particules sonores et les rayons sonores. En ce qui concerne cette dernière méthode, chaque rayon sonore est porteur d'une intensité dont l'amplitude décroît proportionnellement avec le carré de la distance de propagation, simulant ainsi le rayonnement acoustique d'une source sphérique (dispersion géométrique). À

l'inverse, dans le concept des particules sonores, chaque particule est porteuse d'une énergie élémentaire  $\epsilon$ , dont l'amplitude ne varie pas en fonction de la distance de propagation.

D'autre part, l'ensemble des phénomènes physiques intervenant pendant la propagation (réflexion, absorption, diffusion, absorption atmosphérique) ne sont pas considérés de la même manière dans les deux approches. Dans l'approche « rayons sonores », ces phénomènes interviennent sous la forme d'une pondération appliquée sur l'intensité sonore portée par un rayon. Par exemple, au contact avec une paroi de coefficient d'absorption  $\alpha$ , l'intensité du rayon après réflexion sera pondérée par le coefficient  $(1 - \alpha)$ . Dans l'approche « particules sonores », certains de ces phénomènes physiques peuvent être considérés de manière probabiliste. Au contact avec une paroi de coefficient d'absorption  $\alpha$ , la particule pourra par exemple avoir une probabilité  $(1 - \alpha)$  d'être réfléchie, et une probabilité  $\alpha$  d'être absorbée. Dans ce dernier cas, elle disparaît du domaine de propagation.

Un des atouts majeurs du concept de particules sonores réside surtout dans la prise en compte des réflexions diffuses sur les parois et de la diffusion par des encombrements. Dans les méthodes de rayons sonores, les réflexions par les parois sont déterminées par des méthodes de sources-images n'autorisant que des réflexions de type « spéculaire », parfaitement déterministes, et interdisant la prise en compte des phénomènes de « réflexion diffuse ». Dans l'approche particulière, et comme nous le verrons plus loin, ces réflexions diffuses sont considérées de manière probabiliste, l'angle de réflexion pouvant être choisi suivant une quelconque loi de réflexion arbitraire. Il en est également de même dans la prise en compte de la diffusion par des objets répartis dans le milieu de propagation (machines dans des locaux industriels par exemple). Dans l'approche « rayons sonores », il est nécessaire de connaître exactement la forme et la position de l'objet pour calculer de manière déterministe l'angle de réflexion d'un rayon sonore sur celui-ci. Dans l'approche particulière, les objets diffusants peuvent être considérés de manière statistique, sans aucune autre connaissance que leur distribution spatiale et leur forme moyenne, ce qui réduit de manière significative les coûts en termes de calcul et de définition du domaine d'étude. Cette description statistique (ou probabiliste), généralisable à la plupart des phénomènes physiques entrant en jeu lors la propagation du son, fait donc la grande force de l'approche « particules sonores » face à l'approche « rayons sonores ».

### 3.3.3 Principe du code

Le principe des simulations repose donc sur le suivi de particules sonores, porteuses d'une énergie initiale  $\epsilon$ , émises depuis une ou plusieurs sources sonores, dans une enceinte de volume  $V$  totalement ou partiellement fermée. Chaque particule se propage suivant des trajectoires rectilignes ou courbes, jusqu'à entrer en collision avec une paroi ou un objet diffusant. À chaque collision, la particule sonore (ou une partie de son énergie) peut être absorbée, réfléchi ou transmise, en fonction du coefficient d'absorption et de transmission de la paroi ou de l'objet. Le traitement de certains phénomènes physiques pouvant être réalisé par des tirages de nombres aléatoires, cette procédure de simulation peut donc être assimilée à une méthode de Monte Carlo [4]. Les phénomènes physiques simulés par ces tirages de nombres aléatoires seront d'autant mieux respectés que le nombre de tirages aléatoires sera grand, c'est-à-dire que le nombre initial  $N$  de particules sonores sera lui aussi très grand. Néanmoins, plus  $N$  sera grand, plus longue sera la durée des simulations. Le choix de  $N$  est donc un compromis entre le temps de calcul et la précision des résultats, mais est également fonction de la géométrie et des caractéristiques acoustiques même du domaine de propagation. À titre d'exemple, plus le domaine de propagation sera absorbant (au niveau des parois, des objets diffusants, de l'absorption atmosphérique), plus il faudra considérer de particules sonores pour avoir une description satisfaisante des phénomènes physiques.

Dans la version actuelle du code SPPS, deux modes de calcul sont proposés :

- **Modélisation « aléatoire »** : dans ce mode, l'énergie de la particule est constante. Les phénomènes d'absorption et d'absorption atmosphérique sont considérés de manière statistique : en fonction des valeurs de l'absorption atmosphérique et des coefficients d'absorption des matériaux, les particules peuvent être amenées à disparaître complètement du domaine de propagation, ou rester dans le domaine avec la même énergie. Les autres phénomènes physiques (diffusion par un encombrement, réflexion diffuse) sont traités également de manière statistique. Le nombre de particules sonores diminuant au cours du temps, les temps de calcul diminuent progressivement. Par ailleurs, la densité d'énergie sonore en un point du domaine est alors proportionnelle au nombre de particules sonores en ce même point.
- **Modélisation « énergétique »** : dans ce mode, l'énergie de la particule est pondérée en fonction des valeurs de l'absorption atmosphérique et des coefficients d'absorption des matériaux. Les autres phénomènes physiques (diffusion par un encombrement, réflexion diffuse) sont traités également de manière aléatoire. Puisque dans ce mode, le nombre de particules sonores est constant, la durée des simulations numériques est plus longue que pour le premier mode. Par ailleurs, la densité d'énergie sonore en un point du domaine est alors proportionnelle à la somme de l'énergie des de particules sonores en ce même point.

### 3.3.4 Maillage spatial

À la différence des codes de calcul classique basé sur les lancers de rayons et de faisceaux sonores, le code SPPS repose sur un maillage volumique du domaine de propagation.

#### Avantages :

- le maillage du domaine de propagation permet d'optimiser la propagation des particules sonores dans le domaine. En effet, un pré-traitement permet de déterminer pour chaque maille du domaine sa position par rapport aux limites du domaine, sa position à l'intérieur ou à l'extérieur d'un encombrement, ainsi que des propriétés générales, telles que la célérité du son dans la maille et les conditions météorologiques (*via* la hauteur par rapport au sol par exemple). Le suivi d'une particule sonore dans le domaine est donc de ce fait accéléré par la connaissance des propriétés des mailles adjacentes à la maille contenant la particule au pas de temps en cours ;
- le maillage volumique permet de réaliser un maillage surfacique des limites du domaine, qui peut être utilisé pour construire des récepteurs surfaciques, sur n'importe quelle limite du domaine ;
- la réalisation d'un maillage volumique nécessitant que le modèle géométrique soit parfaitement fermé, on s'assure ainsi qu'aucune particule sonore ne puisse disparaître du domaine de propagation en raison d'une « ouverture » (faces du modèles non-parfaitement jointes par exemple).

#### Inconvénients :

- la réalisation du maillage nécessitant que le modèle géométrique soit parfaitement fermé,

cela suppose que le modèle soit parfaitement construit. Les codes de calculs basés sur les rayons sonores n'ont pas cette contrainte, puisqu'ils « permettent » qu'un rayon puisse « sortir » du domaine et se propager à l'infini.

### 3.3.5 Maillage temporel

En parallèle, le suivi des particules sonores s'effectue par pas de temps constant, et non en temps continu. Il faut donc choisir un pas de temps suffisamment faible pour que les phénomènes physiques présents dans les processus d'émission sonore et de propagation acoustique, puissent être « ramenés » à un pas de temps donné.

**Maillage supporté par le code SPPS :**

- maillage tétraédrique ;

# Modélisation des phénomènes physiques dans SPSS

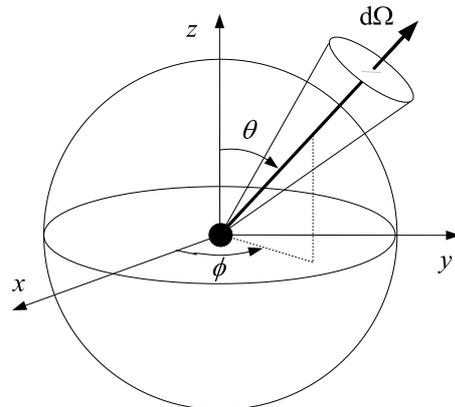
## 4.1 Modélisation des sources sonores

### 4.1.1 Formalisme général

**Directivité acoustique et densité de probabilité d'émission des particules** D'un point de vue de la modélisation numérique, le problème de l'émission sonore depuis une source ponctuelle revient à attribuer aux particules sonores, à un instant donné (qui peut être l'origine des temps), à la position exacte de la source, des directions de propagation en accord avec la directivité de la source. Par exemple, dans le cas d'une source sonore omnidirectionnelle, il s'agit de vérifier que les particules sonores, une fois émises, se répartissent uniformément sur une sphère centrée sur la source (figure 4.1). La première solution venant à l'esprit consiste à procéder à une discrétisation déterministe<sup>1</sup> du rayonnement de la source en coordonnées sphériques. Néanmoins, une telle discrétisation des angles d'émission provoque une concentration artificielle des particules sonores au niveau des pôles de la sphère symbolisant la source.

Il est en fait nécessaire de vérifier le respect du nombre de particules émises par angle solide élémentaire  $d\Omega = d\phi \sin\theta d\theta$ , vérifie la directivité de la source, soit sous une forme déterministe, soit de manière aléatoire (figure 4.1). Dans notre cas, nous avons choisi un tirage aléatoire des directions initiales des particules. Il est alors nécessaire de définir les densités de probabilité  $g(\phi)$  et  $p(\theta)$ , respectivement pour les angles  $\phi \in [0, 2\pi]$  et  $\theta \in [0, \pi]$ ,

<sup>1</sup>La méthode consiste à considérer des angles d'émission  $\theta$  et  $\phi$  par pas constant.



**Figure 4.1** — Géométrie élémentaire pour une source sonore. La direction d'émission d'une particule est définie en coordonnées sphériques par les angles  $\phi$  (azimut) compris entre 0 et  $2\pi$ , et  $\theta$  compris entre 0 et  $\pi$  (angle polaire), ou par l'angle solide élémentaire  $d\Omega$ .

telles que la densité de probabilité  $Q(\theta, \phi)$  d'émission dans la direction  $(\theta, \phi)$  (ou la densité de probabilité  $F(\Omega)$  d'émission dans l'angle solide élémentaire  $\Omega$ ) soit en accord avec la directivité de la source. En pratique,  $Q(\theta, \phi)$  n'est rien d'autres que la directivité de la source. Par définition, ces densités de probabilité

vérifient la relation suivante :

$$\begin{aligned} \int F(\Omega) d\Omega &= \iint Q(\theta, \phi) \sin \theta d\phi d\theta \quad (4.1) \\ &= \int_0^{2\pi} g(\phi) d\phi \int_0^\pi p(\theta) \sin \theta d\theta \\ &= 1. \end{aligned} \quad (4.3)$$

D'un point de vue de la puissance acoustique, cela revient bien à considérer que la puissance  $W$  de la source se répartie bien sur la surface d'une sphère centrée sur la source :

$$\iint W Q(\theta, \phi) \sin \theta d\phi d\theta = W. \quad (4.4)$$

Si les densités  $g(\phi)$  et  $p(\theta)$  sont indépendantes, elles vérifient :

$$\int_0^{2\pi} g(\phi) d\phi = 1, \quad (4.5)$$

et

$$\int_0^\pi p(\theta) \sin \theta d\theta = 1. \quad (4.6)$$

**Puissance de la source et énergie élémentaire des particules** D'un point de vue acoustique, pendant un intervalle de temps  $\Delta t$ , une source de puissance  $W$  émet une quantité d'énergie  $E = W \times \Delta t$ . D'un point de vue du concept particulaire, chaque particule est porteuse d'une énergie élémentaire  $\varepsilon_0$ . Si la source émet  $N$  particules sonores, la conservation de l'énergie entre les deux approches impose donc :

$$N \times \varepsilon_0 = W \times \Delta t, \quad (4.7)$$

soit

$$\varepsilon_0 = \frac{W}{N} \times \Delta t. \quad (4.8)$$

#### 4.1.2 Source omnidirectionnelle

**Description physique** Une source sonore omnidirectionnelle est une source qui rayonne dans toutes directions avec la même amplitude. La directivité de la source est assimilable à une sphère. Le coefficient de directivité  $Q$  associé à la directivité de la source est égal à 1.

**Modélisation** Dans le cas d'une source omnidirectionnelle, les densités  $g(\phi)$  et  $p(\theta)$  étant uniformes, les relations (4.5) et (4.6) vérifient :

$$\int_0^{2\pi} g(\phi) d\phi = \int_0^{2\pi} A d\phi = 1 \quad (4.9)$$

et

$$\int_0^\pi p(\theta) \sin \theta d\theta = \int_0^\pi B \sin \theta d\theta = 1, \quad (4.10)$$

où  $A = 1/2\pi$  et  $B = 1/2$  sont deux constantes de normalisation. En pratique, la méthode consiste dans un premier temps, à tirer un angle  $\theta$  entre 0 et  $2\pi$  de la manière suivante :

$$\phi = 2\pi \times u \in [0, 2\pi] \quad (4.11)$$

où  $u$  est un nombre aléatoire compris entre 0 et 1, et défini par une distribution uniforme. Si l'angle  $\theta$  est choisi de la même manière (i.e.  $\theta = \pi \times v$ ,  $v$  étant un nombre aléatoire suivant une distribution uniforme entre 0 et 1), la distribution des directions d'émission ne respecte pas l'uniformité par angle solide élémentaire, puisque la condition (4.6) n'est pas vérifiée. Dans ce cas, les directions autour des pôles seraient privilégiées. Il faut en fait déterminer les angles  $\theta$  qui vérifient une distribution proportionnelle à  $\sin \theta$ . Dans ce cas simple, la procédure consiste à appliquer la méthode de la fonction de distribution cumulative inverse. D'après la relation (4.6), la probabilité  $f(\hat{\theta})$  de tirer un angle  $\theta < \hat{\theta}$  est donnée par :

$$f(\hat{\theta}) = \frac{1}{2} \int_0^{\hat{\theta}} \sin \theta d\theta = \frac{1}{2} [\cos \hat{\theta} - 1]. \quad (4.12)$$

Sachant que cette distribution est comprise entre 0 et 1, le choix de l'angle  $\hat{\theta}$  est réalisé en tirant aléatoirement un nombre  $v \in [0, 1]$ , suivant une distribution uniforme, tel que :

$$\hat{\theta} = \cos^{-1}(2v - 1) \in [0, \pi]. \quad (4.13)$$

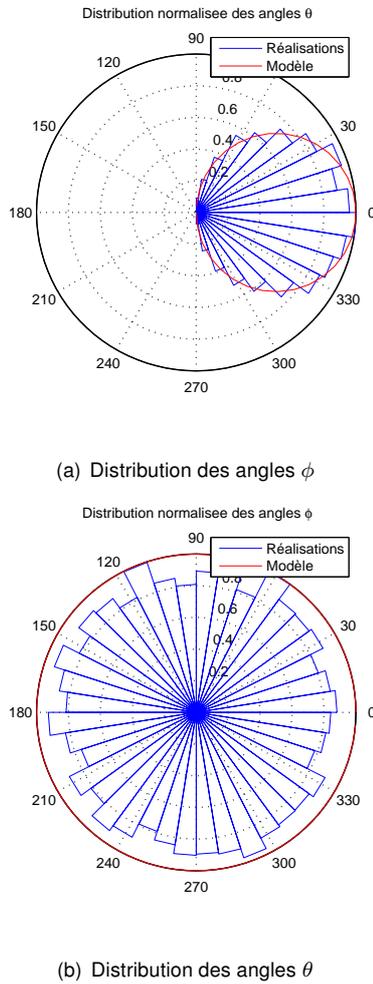
Les vecteurs de propagation  $\mathbf{v}(v_x, v_y, v_z)$ , de norme  $c$  (célérité du son), sont ensuite définis par les relations :

$$\mathbf{v} = \begin{cases} v_x = c \cos \theta \sin \phi \\ v_y = c \sin \theta \sin \phi \\ v_z = c \cos \phi \end{cases} \quad (4.14)$$

**Vérification** La figure 4.2 présente un exemple de répartition des angles  $\theta$  et  $\phi$  obtenus suivant cette méthode de tirage avec 10000 réalisations. On observe aisément que la distribution des angles vérifie la distribution théorique. Il est bien entendu que la qualité du tirage aléatoire est fonction de la méthode de génération des nombres aléatoires, et que le respect des distributions théoriques augmente avec le nombre de tirages.

#### 4.1.3 Source unidirectionnelle

**Description physique** Une source sonore unidirectionnelle est une source qui rayonne dans une seule direction de l'espace, définie par les angles  $\theta$  et  $\phi$ , au point d'émission et dans le repère absolu. Le



**Figure 4.2** — Distribution des angles d'émission pour une source omnidirectionnelle, suivant la procédure présentée au paragraphe 4.1.2. Simulation réalisée sous MATLAB<sup>®</sup> avec 10000 réalisations.

coefficient de directivité associé à la directivité de la source est alors défini par :

$$Q(\theta, \phi) = \delta(\theta, \phi) = \delta(\theta) \times \delta(\phi), \quad (4.15)$$

où  $\delta$  désigne la distribution de Dirac.

**Modélisation** La modélisation de ce type de source est entièrement déterministe et ne pose donc pas de problème. Il suffit de définir un vecteur  $\mathbf{s}(s_x, s_y, s_z)$  dans le repère absolu de la scène, tel que :

$$\mathbf{s} = \begin{cases} s_x = s \cos \theta \sin \phi \\ s_y = s \sin \theta \sin \phi \\ s_z = s \cos \phi \end{cases} \quad (4.16)$$

#### 4.1.4 Sources « plan » (XY, YZ et XZ)

**Description physique** Une source sonore de type « plan » est une source qui rayonne dans un plan. Par défaut, on considère trois sources XY, YZ et XZ définis par les trois plans de référence  $(xOy)$ ,  $(yOz)$  et  $(xOz)$  respectivement.

**Modélisation** La procédure consiste à déterminer aléatoirement la direction de propagation d'une particule dans un plan. Reprenant la convention d'angle présentée à la figure 4.1, on aura :

$$\begin{cases} \text{plan XY : } \theta = 2\pi \times u & \text{et} & \phi = \pi/2 \\ \text{plan YZ : } \theta = 0 & \text{et} & \phi = 2\pi \times u \\ \text{plan XZ : } \theta = \pi/2 & \text{et} & \phi = 2\pi \times u \end{cases} \quad (4.17)$$

où  $u$  désigne un nombre aléatoire compris entre 0 et 1. Par la suite, le vecteur de propagation  $\mathbf{v}(v_x, v_y, v_z)$ , de norme  $c$  (célérité du son), est défini par les relations :

$$\mathbf{v} = \begin{cases} v_x = c \cos \theta \sin \phi \\ v_y = c \sin \theta \sin \phi \\ v_z = c \cos \phi \end{cases} \quad (4.18)$$

## 4.2 Modélisation de la propagation acoustique

### 4.2.1 Propagation acoustique

**Description physique** En l'absence d'absorption et de réflexion sur les parois du domaine ou sur des objets, la décroissance de l'intensité sonore issue d'une source omnidirectionnelle s'écrit :

$$I = \frac{Q}{4\pi r^2}, \quad (4.19)$$

où  $r$  est la distance à la source, et  $Q$  la directivité de la source dans la direction d'observation ( $Q = 1$  pour une source omnidirectionnelle). Cette décroissance traduit le phénomène de dispersion géométrique, qui décrit l'« étalement » d'une onde sphérique au fur et à mesure de la propagation.

**Modélisation** Compte tenu de la méthode présentée au paragraphe 4.1.2 pour une source omnidirectionnelle, la dispersion géométrique est automatiquement respectée. En effet, la méthode numérique proposée permet d'obtenir une répartition uniforme des particules sur un angle solide élémentaire. Sur une sphère, la distribution des particules  $n(r)$  (en  $m^2$ ) est donc égale à :

$$n(r) = \frac{N}{4\pi r^2}, \quad (4.20)$$

où  $N$  est le nombre de particules. La distribution des particules vérifie donc bien la même décroissance que l'intensité. Il faut toutefois noter, que plus le point d'observation sera loin de la source, plus il faudra de particules sonores.

**Vérification** La figure 4.3 montre les résultats numériques de la propagation en champ libre<sup>2</sup>, pour une source sonore omnidirectionnelle, sans absorption atmosphérique (nombre de particules  $N = 20 \times 10^6$ ). L'accord est excellent.

#### 4.2.2 Absorption atmosphérique

**Description physique** Durant sa propagation dans l'air, une onde sonore est en partie atténuée du fait de mécanismes physiques particuliers (processus de transmission « classique », absorption moléculaire due à la relaxation rotationnelle, absorption moléculaire due à la relaxation vibratoire de l'oxygène et de l'azote) [3]. Ainsi, après une distance de propagation  $r$ , l'amplitude  $p_t$  de la pression acoustique décroît suivant la relation [8] :

$$p_t = p_i \exp\left(-\frac{\ln(10)}{20} \alpha_{\text{air}} \times r\right) \quad (4.21)$$

où  $p_i$  est la pression initiale. Considérant que l'intensité sonore est proportionnelle au carré de la pression acoustique,

$$I \propto p_t^2 \propto \exp\left(-2\frac{\ln(10)}{20} \alpha_{\text{air}} \times r\right) \quad (4.22)$$

et écrivant que l'intensité  $I$  de l'onde sonore décroît suivant la relation :

$$I = I_0 \exp -m r, \quad (4.23)$$

où  $I_0$  est l'intensité initiale de l'onde sonore, alors le coefficient d'absorption atmosphérique  $m$  (en Np/m) peut-être exprimé à partir du coefficient d'absorption atmosphérique  $\alpha_{\text{air}}$  (en dB/m), par la relation :

$$m = \frac{\ln 10}{10} \alpha_{\text{air}}. \quad (4.24)$$

Dans le code SPPS, le coefficient d'absorption atmosphérique  $\alpha_{\text{air}}$  est calculé suivant la norme ISO 9613-1 :1993 *Acoustique. Atténuation du son lors de sa propagation à l'air libre. Partie 1 : calcul de l'absorption atmosphérique* [8], en considérant la fréquence centrale de chaque bande de fréquence<sup>3</sup> de

<sup>2</sup>Le champ libre est simulé en considérant un long couloir avec des limites parfaitement absorbantes.

<sup>3</sup>Pour des raisons de simplicité, nous considérons la fréquence normale des bandes de fréquence [12], et non la fréquence centrale exacte.

calcul conformément à la norme de référence (cf. [8, §8.2.1]). Cette approximation est considérée comme valide si le produit de la distance source-récepteur (en km) par le carré de la fréquence centrale (en kHz) ne dépasse pas 6 km/kHz<sup>2</sup> pour les bandes de tiers d'octave et 3 km/kHz<sup>2</sup> pour les bandes d'octave. La distance de propagation ne doit néanmoins pas dépasser 6 km pour les bandes de tiers d'octave et 3 km pour les bandes d'octave, quelle que soit la fréquence centrale considérée.

**Modélisation « aléatoire »** En choisissant le mode de calcul « aléatoire », l'absorption atmosphérique est prise en compte comme une probabilité de disparition de la particule sonore durant son déplacement. La densité de probabilité correspondante peut-être définie à partir de la relation (4.23) :

$$f(r) = \exp -m r. \quad (4.25)$$

Cette quantité exprime la probabilité que la particule ne pas soit absorbée durant son déplacement  $r$ . La densité de probabilité  $f(r)$  est bien comprise entre 1 et 0 :  $f(0) = 1$ , la probabilité est maximale, la particule ne pouvant pas être absorbée si elle ne déplace pas ;  $f(\infty) = 0$ , la probabilité est nulle puisque la particule ne peut pas se propager à l'infini (figure 4.4). On vérifie également facilement que la probabilité de propagation est indépendante de la probabilité de propagation précédente :

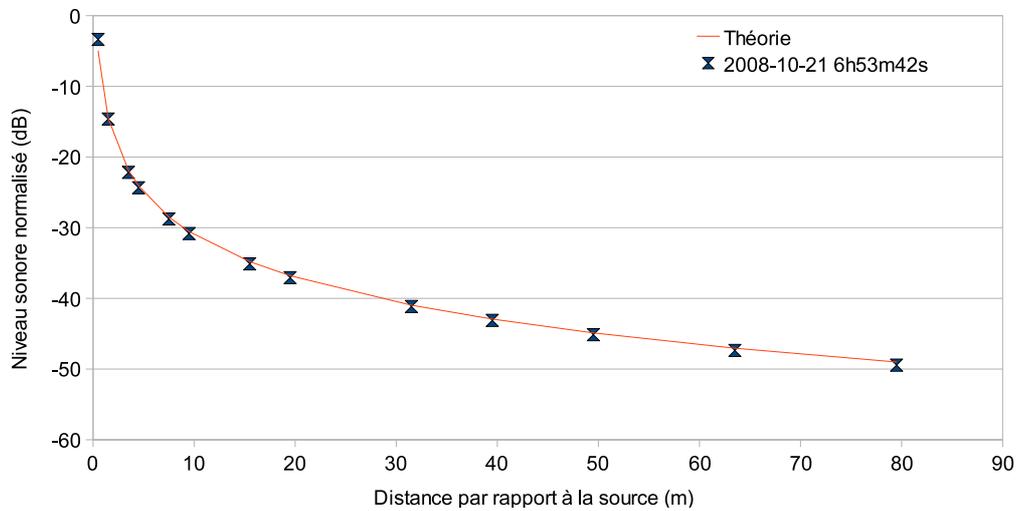
$$f\left(\sum_{n=1}^N r_i\right) = \prod_{i=1}^N f(r_n). \quad (4.26)$$

La prise en compte de l'absorption atmosphérique est relativement simple. Il suffit en effet de tirer un nombre aléatoire uniforme  $\zeta$  compris entre 0 et 1, à chaque pas de temps, et de comparer ce nombre à la densité de probabilité  $f(d_0)$  correspondant à un déplacement élémentaire  $d_0 = c\Delta t$  sur un pas de temps  $\Delta t$ . Si ce nombre  $\zeta$  est inférieur à  $f(d_0)$ , il y aura propagation. Dans le cas contraire, il y aura absorption atmosphérique, donc disparition de la particule. Même sur un faible nombre de particules, cette méthode permet de prendre en compte correctement l'absorption atmosphérique.

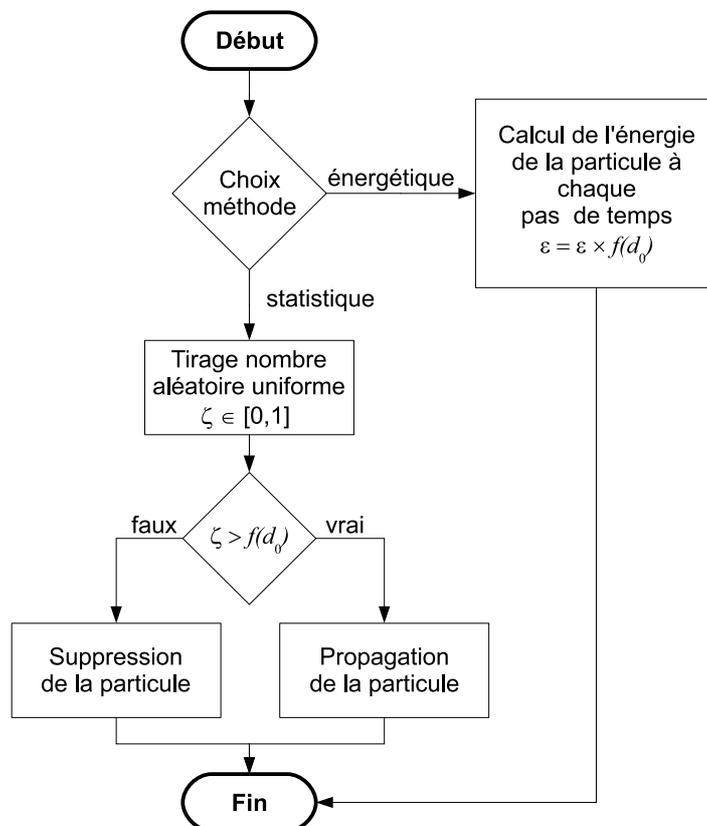
**Modélisation « énergétique »** En choisissant l'option de calcul « énergétique », l'énergie de la particule est pondérée tout au long de son déplacement, en utilisant la relation (4.23).

**Vérification** A titre d'illustration, la figure 4.6 présente la décroissance sonore calculée<sup>4</sup> par le code

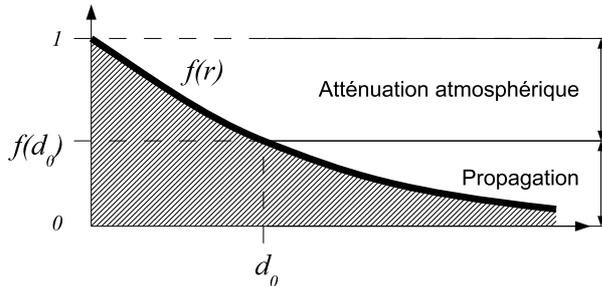
<sup>4</sup>La simulation est identique à celle présentée au paragraphe 4.2.1 pour la vérification du respect de la dispersion géométrique.



**Figure 4.3** — Vérification du respect de la dispersion géométrique avec le code *SPPS*. Les simulations numériques sont comparées avec la décroissance théorique ( $N = 20 \times 10^6$  particules). Le marqueur présente le résultat de la simulation (le libellé donne la date des simulations). La ligne continue présente la dispersion géométrique théorique.



**Figure 4.5** — Organigramme de la prise en compte de l'absorption atmosphérique dans le code *SPPS* par les méthodes « aléatoire » et « énergétique » ( $d_0$  désigne la distance de propagation sur un pas de temps :  $d_0 = c \times \Delta t$ ).



**Figure 4.4** — Prise en compte de l'absorption atmosphérique par un processus aléatoire. La courbe  $f(r)$  sépare le domaine de propagation du domaine d'absorption atmosphérique.

SPPS à 10 kHz pour des conditions atmosphériques classiques ( $T = 20^\circ\text{C}$ ,  $H = 50\%$ ,  $P = 101325 \text{ Pa}$ , soit  $m = 0.036 \text{ m}^{-1}$ ), pour les deux types de modélisation, comparée à la décroissance théorique présentée à la relation (4.23). Comme attendu, la modélisation « énergétique » donne un meilleur résultat que la modélisation « aléatoire », l'écart moyen par rapport à la courbe théorique étant de 0.17 dB et 0.41 dB respectivement, les temps de calcul étant par ailleurs similaires.

### 4.2.3 Profil de célérité acoustique

**Description physique** En milieu extérieur et pour des hauteurs de propagation limitées à 100 m, la propagation acoustique peut être influencée par les conditions micro-météorologiques régies par des lois thermiques (transfert de chaleur) et aérodynamiques (profils de vent). Les phénomènes qui apparaissent à ces altitudes présentent une interaction très forte avec le sol (topographie, température de surface et de sous-sol, hygrométrie, cultures, forêts, obstacles, bâti...). De plus, ils évoluent rapidement dans le temps et dans l'espace, ce qui rend leur description analytique et leur modélisation numérique complexe. Les facteurs thermiques et aérodynamiques qui influent sur la propagation sont les suivants :

- **Facteurs thermiques** : les échanges thermiques entre le sol et la couche basse de l'atmosphère conduisent à une variation de la température de l'air en fonction de la hauteur au dessus du sol, et donc à une variation de la vitesse du son.
- **Facteurs aérodynamiques** : compte tenu de l'état de rugosité de surface du sol, la vitesse du vent est toujours plus élevée en hauteur qu'au

niveau du sol. Dans une situation donnée, la vitesse du son en présence de vent correspond à la somme algébrique de la vitesse du son en l'absence de vent et de la projection du vecteur vent sur la direction de propagation considérée. Cette vitesse varie donc en fonction de la hauteur au-dessus du sol.

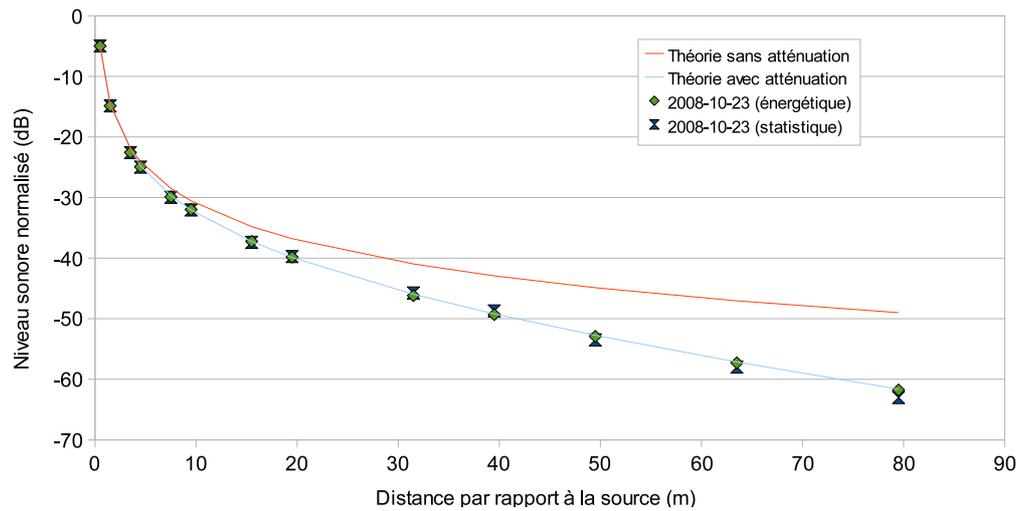
Par analogie avec les lois de l'optique, l'effet des conditions atmosphériques sur la propagation acoustique peut être décrit à travers l'expression de l'indice acoustique  $n$  du milieu de propagation. Si l'on se place dans une coupe verticale, cet indice est supposé varier avec l'altitude  $z$  et avec la distance source-récepteur  $r$ , tel que :

$$n(r, z) = \frac{c(r, z)}{c_0} = \langle n(r, z) \rangle + \mu(r, z), \quad (4.27)$$

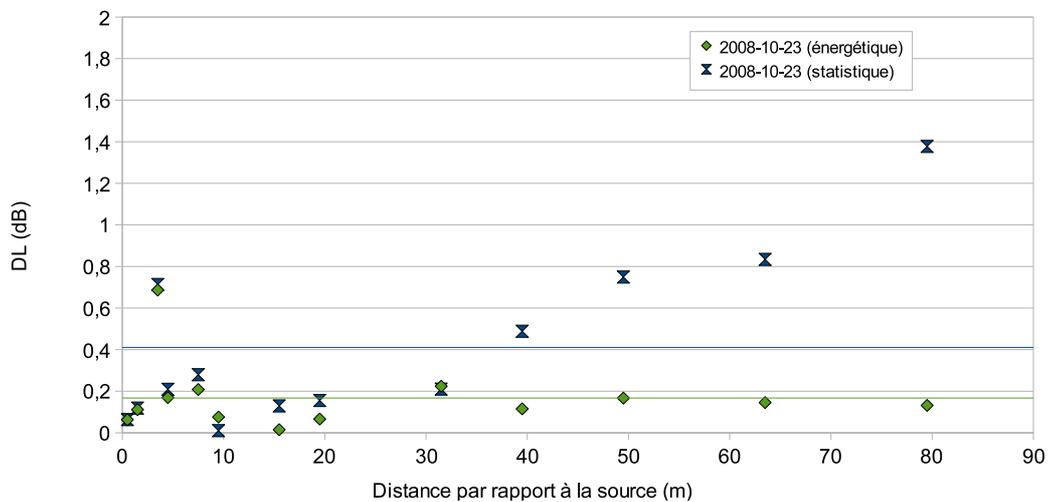
où  $c$  est la célérité effective de l'onde sonore dans le milieu traversé et  $c_0$  celle de référence. Ainsi, on peut distinguer deux phénomènes distincts influant sur la propagation acoustique, la réfraction et la turbulence atmosphérique. Ces phénomènes sont respectivement liés aux parties déterministe  $\langle n \rangle$  et stochastique  $\mu$  de l'indice du milieu de propagation. En pratique, ces phénomènes de réfraction et de turbulence coexistent et interagissent, ce qui conduit à des conditions de propagation complexes, ainsi qu'à une très large dispersion des niveaux sonores rencontrés *in situ*, tous paramètres identiques par ailleurs (topographie, nature de sol, géométrie source-récepteur...).

**Modèle de profil de célérité** Le profil moyen de célérité du son dépend ainsi des profils moyens de vent et de température. Ce profil de célérité peut être décrit de manière analytique, suivant qu'il suit une loi linéaire (« lin »), logarithmique (« log »), hybrides (« log-lin ») ou autre. Les profils « log » présentent ainsi l'avantage de traduire le très fort gradient vertical de célérité du son au voisinage immédiat du sol, mais ne rendent pas bien compte de l'évolution plus modérée avec l'altitude au-delà d'une certaine hauteur. A l'inverse, les profils « lin » minimisent les effets au voisinage du sol et ne sont donc pas représentatifs de la réalité lorsque l'on se place à très basse altitude. Un bon compromis consiste donc à utiliser des profils hybrides du type « log-lin » (valables surtout pour une atmosphère dite « stable »), exprimés à travers les paramètres  $a_{\log}$  et  $b_{\text{lin}}$  qui apparaissent dans l'expression analytique du profil vertical de la célérité effective du son :

$$\langle c(z) \rangle = c_0 + a_{\log} \times \ln \left( 1 + \frac{z}{z_0} \right) + b_{\text{lin}} \times (z - z_0), \quad (4.28)$$



(a) Atténuation sonore en présence d'absorption atmosphérique



(b) Écart moyen par rapport à la décroissance théorique

**Figure 4.6** — Illustration de la prise en compte de l'absorption atmosphérique dans le code *SPPS*, pour les deux modélisations « aléatoire » et « énergétique ». Comparaison avec la décroissance théorique et représentation des écarts *DL* (en dB). Simulations réalisées avec  $N = 20 \times 10^6$  particules, pour des conditions atmosphériques classiques :  $T = 20^\circ\text{C}$ ,  $H = 50\%$ ,  $P = 101325 \text{ Pa}$ ,  $F = 10 \text{ kHz}$  (soit  $m = 0.036 \text{ m}^{-1}$ ).

où  $z_0$  est le paramètre de rugosité, dont les valeurs typiques s'étendent de  $10^{-2}$  m environ pour l'herbe rase à plusieurs mètres en milieu urbain. Le gradient vertical s'exprime alors en dérivant suivant la variable  $z$  :

$$\frac{\partial(c(z))}{\partial z} = \frac{a_{\log}}{z_0} + b_{\text{lin}}. \quad (4.29)$$

La propagation dans un milieu à célérité variable a pour principal effet d'incurver les rayons sonores vers le bas ou vers le haut suivant que le gradient vertical de célérité du son soit respectivement positif (*conditions (très) favorables* à la propagation) ou négatif (*conditions (très) défavorables* à la propagation). L'état transitoire entre ces 2 états représente des *conditions homogènes* de propagation.

- **Homogène** : la célérité  $c$  est identique en tout point du domaine et égale à la célérité de référence  $c_0$ , cette dernière étant calculée en fonction des conditions de température et d'humidité, par la formule :

$$c_0 = 343.2 \sqrt{\frac{T}{T_{\text{ref}}}}, \quad (4.30)$$

où  $T$  est la température (en K), et  $T_{\text{ref}} = 293.15$  K la température de référence [8].

- **Très défavorable** :  $a_{\log} = -1$  et  $b_{\text{lin}} = -0.12$  ;
- **Défavorable** :  $a_{\log} = -0.4$  et  $b_{\text{lin}} = -0.04$  ;
- **Favorable** :  $a_{\log} = +0.4$  et  $b_{\text{lin}} = +0.04$  ;
- **Très favorable** :  $a_{\log} = +1$  et  $b_{\text{lin}} = +0.12$ .

La courbure du rayon, à la frontière des zones (I) et (II), est obtenue en appliquant la construction de Huygens-Fresnel, aboutissant à la loi de Snell-Descartes suivante [32] (figure 4.7) :

$$\frac{\cos \gamma_1}{c_1} = \frac{\cos \gamma_2}{c_2}, \quad (4.31)$$

où  $c_1$  et  $c_2$  sont respectivement les normes des vecteurs de propagation  $\mathbf{c}_1$  et  $\mathbf{c}_2$ , et où les angles  $\gamma_1$  et  $\gamma_2$  sont définis par rapport à l'axe horizontal au plan ( $xOy$ ). Par construction, la projection de la direction de propagation dans le plan ( $xOy$ ), définie par l'angle  $\phi$  en coordonnées sphériques, est conservée (figure 4.7).

**Modélisation** Quelque soit le mode de calcul choisi, la prise en compte de la célérité est identique. A chaque pas de temps, la norme de célérité est calculée en fonction du profil de célérité retenu, à partir de la relation (4.28). Pour déterminer la

nouvelle direction de propagation, dû au changement de célérité, il faut ensuite appliquer la relation (4.31). Connaissant l'angle  $\theta_1$  de la direction de propagation initiale, la nouvelle direction de propagation est définie par :

$$\cos \gamma_2 = \frac{c_2}{c_1} \cos \gamma_1 = \frac{c_2}{c_1} \frac{\sqrt{c_{1x}^2 + c_{1y}^2}}{c_1}. \quad (4.32)$$

Les coordonnées du vecteur propagation sont ensuite obtenues par :

$$\mathbf{c}_2 = \begin{cases} c_{2x} = c_2 \cos \gamma_2 \cos \phi \\ c_{2y} = c_2 \cos \gamma_2 \sin \phi \\ c_{2z} = c_2 \sin \gamma_2 \end{cases} \quad (4.33)$$

avec

$$\sin \phi = \frac{c_{1y}}{\sqrt{c_{1x}^2 + c_{1y}^2}}, \quad (4.34)$$

et

$$\cos \phi = \frac{c_{1x}}{\sqrt{c_{1x}^2 + c_{1y}^2}}. \quad (4.35)$$

D'un point de vue de la simulation numérique, le calcul du  $\cos \gamma_2$  par la relation (4.32) peut donner des valeurs supérieures à 1 ce qui n'est évidemment pas physique. Ce cas se produit lors d'une inversion de la courbure (*turning point*) d'un rayon. Pour éviter ce problème et imposer un changement de courbure, la procédure consiste à imposer la valeur de  $\gamma_2$  à  $1 - \epsilon$  ( $\epsilon$  étant une valeur très petite) et à changer l'orientation de la composante  $c_{2z}$  (*i.e.*  $c_{2z} \rightarrow -c_{2z}$ ).

A titre d'exemple, la figure 4.9 présente deux illustrations de la prise en compte de la réfraction acoustique en utilisant cette méthode. Cette figure peut être comparée directement avec les exemples donnés dans la référence [32, figures 4.5 et 4.6].

#### 4.2.4 Diffusion par un encombrement

La présence d'un grand nombre d'objets sur le parcours d'une onde sonore peut engendrer un processus de diffusion. Ce processus peut être simulé de manière déterministe, en modélisant chaque objet individuellement. Lorsque le nombre d'objets devient important, et que ces objets sont de tailles similaires (exemple d'un hall industriel avec de nombreuses machines (sans émission acoustique) ou boîte similaires), il peut être plus intéressant de modéliser statistiquement cet encombrement.

**Description physique** Afin de prendre en compte la diffraction et l'absorption des objets diffusants répartis dans le milieu de propagation, nous avons considéré une approche similaire à celle de Ondet et

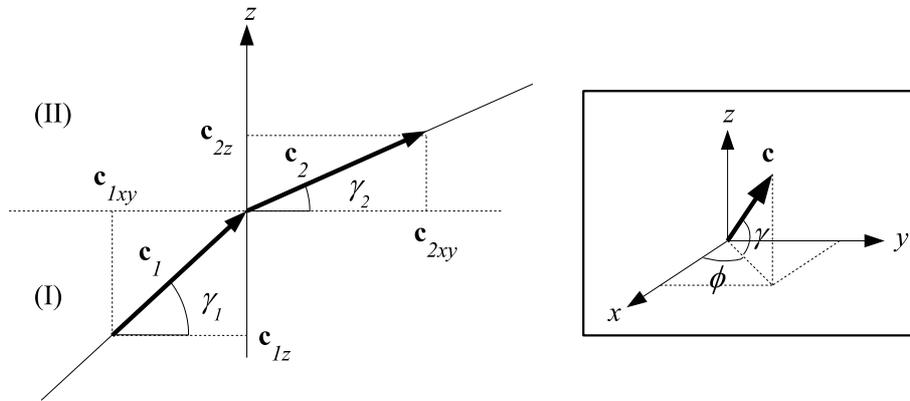


Figure 4.7 — Schéma de principe de la réfraction acoustique dans l'atmosphère en présence d'un profil de célérité vertical.

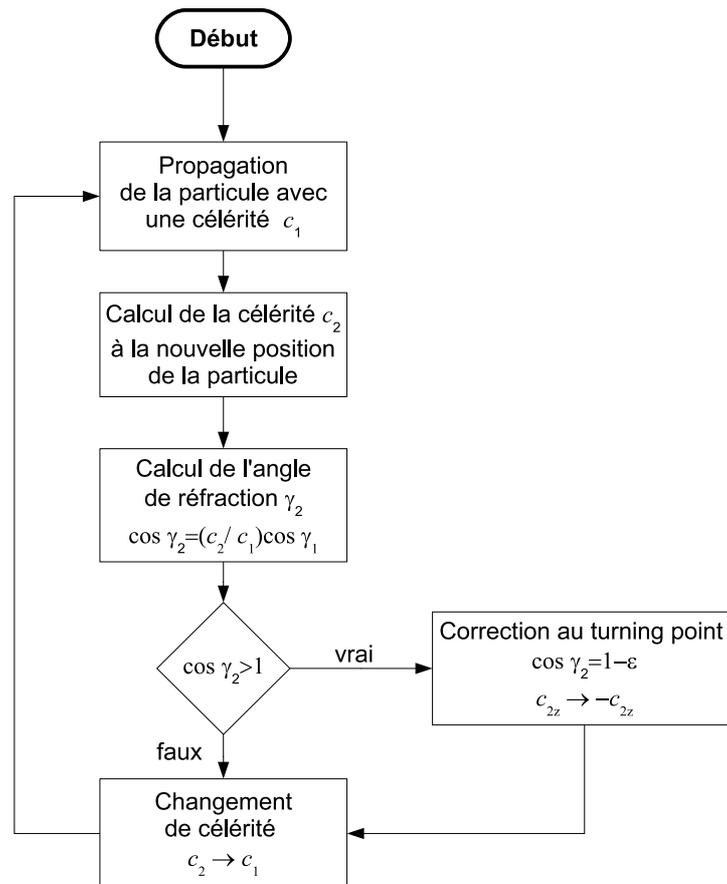
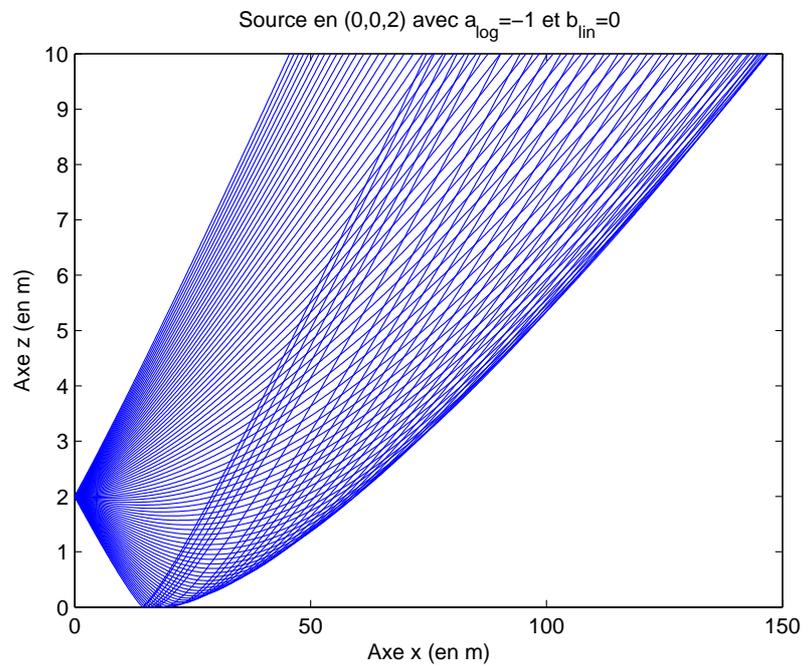
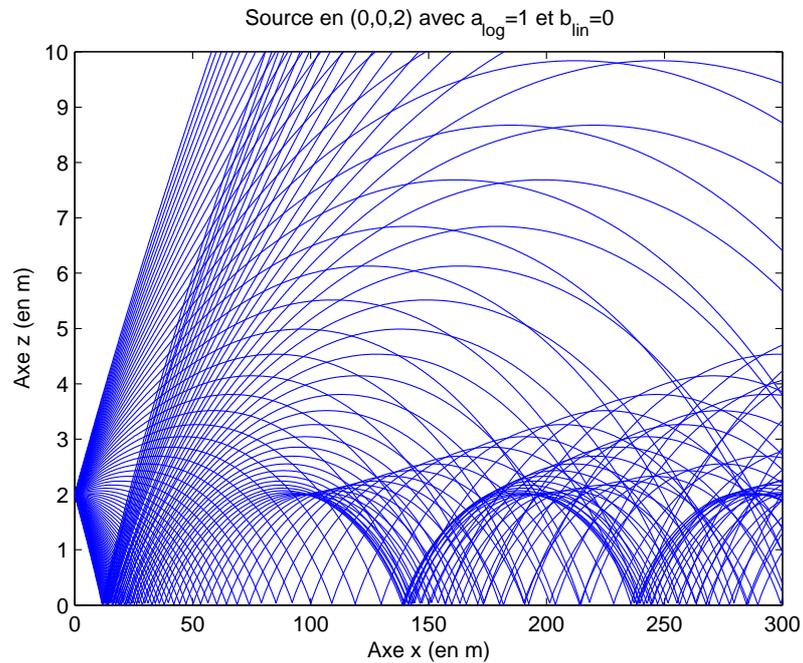


Figure 4.8 — Organigramme de la prise en compte de la réfraction acoustique d'une particule sonore (due à un profil vertical de célérité, suivant l'axe z).

(a) *Upward* réfraction(b) *Downward* réfraction

**Figure 4.9** — Illustration de la modélisation de la réfraction acoustique avec *SPPS*, en condition upward avec  $a_{\log} = -1$  ms et  $b_{\text{lin}} = 0$ , et en condition downward avec  $a_{\log} = +1$  ms et  $b_{\text{lin}} = 0$ . Dans cet exemple, le sol est supposé parfaitement réfléchissant.

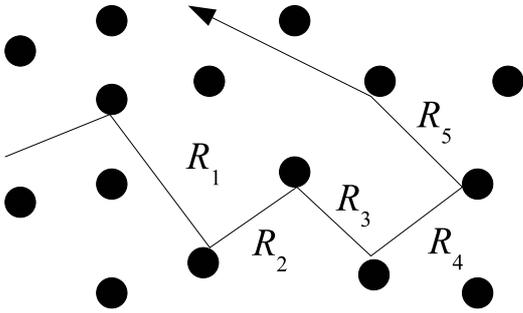


Figure 4.10 — Illustration de la diffusion par un encombrement.

Barbry présentée dans la référence [29], elle même reprise des travaux de Kuttruff [23], de Auletta [1, 2], et de Lindquist [28], entre autres. Dans cette approche,

- les objets diffusants sont considérés comme ponctuels. Les particules sont renvoyées dans toutes les directions de l'espace à chaque collision avec un objet diffusant (sauf en cas d'absorption). Cette hypothèse est valide en général quand la longueur d'onde est de l'ordre de grandeur de la dimension caractéristique de l'obstacle ;
- le phénomène de diffusion suit un processus de Poisson, ce qui signifie que la probabilité de collision d'une particule sonore avec un objet diffusant, suit une loi de Poisson. Les probabilités de collision sont indépendantes les unes des autres (la probabilité de collision pendant le temps  $t$  et  $t + dt$  est indépendante des collisions avant le temps  $t$ ) ;
- les objets de l'encombrement ne produisent pas de particules (*i.e.* ces objets ne sont pas des sources sonores).

Une onde sonore se propageant dans le milieu peut être amenée à rentrer en contact avec des objets diffusants, entraînant simultanément la diffraction de l'onde, et, en partie, son absorption. Par analogie, dans l'approche particulaire, une particule entrant en contact avec un objet diffusant peut être, soit absorbée, soit réfléchiée dans une autre direction de propagation (figure 4.10). A l'échelle macroscopique, c'est-à-dire considérant simultanément l'ensemble des particules sonores, il se produit un processus de diffusion, caractérisé par :

- le coefficient d'absorption  $\alpha_c$  des objets diffusants ;

- la densité d'encombrement  $n_c$  du milieu de propagation, défini par le nombre  $N_c$  d'obstacles présent dans le volume  $V_c$  :

$$n_c = \frac{N_c}{V_c}. \quad (4.36)$$

- la section moyenne de diffusion  $q_c$ , c'est-à-dire la surface moyenne de l'objet diffusant, vue par une particule dans une direction incidente donnée. En pratique, cette donnée est très difficile à obtenir, voir impossible, dès lors que les objets diffusants sont de formes complexes et différentes. Dans cette condition, il est commun d'assimiler l'objet diffusant à une sphère, ayant la même surface externe  $s_c$  que l'objet. Quelque soit l'angle d'incidence, la section visible de la sphère (section moyenne de diffusion) sera égale à un quart de la surface totale de la sphère, soit :

$$q_c = \frac{s_c}{4}. \quad (4.37)$$

- la section moyenne de diffraction par unité de volume  $\nu_c$ , encore appelée fréquence de diffusion, par

$$\nu_c = n_c q_c, \quad (4.38)$$

si tous les objets diffusants sont identiques, ou

$$\nu_c = \frac{1}{V} \sum_{p=1}^{N_c} \frac{s_{c_p}}{4} \quad (4.39)$$

si chaque objet diffusant  $p$  est défini par sa surface  $s_{c_p}$ . En pratique, et dans la suite du document, les objets diffusants seront considérés uniformes dans un même volume de diffusion. Néanmoins dans un même volume de propagation, plusieurs volumes de diffusion disjoints peuvent être considérés.

Puisque le phénomène de diffusion suit une loi de Poisson, la probabilité qu'une particule sonore subisse  $k$  collisions avec des objets diffusants, au bout d'un temps  $t_k$  est égale à :

$$W_k(ct_k) = \frac{(\nu_c ct_k)^k}{k!} \exp(-\nu_c ct_k), \quad (4.40)$$

où  $ct_k$  est la distance parcourue pendant un temps  $t_k$ , pouvant être exprimée à partir de la distance  $R_p$  séparant deux collisions (figure 4.10) :

$$ct_k = \sum_{p=1}^k R_p. \quad (4.41)$$

Les probabilités de collision étant indépendantes les unes des autres [29], on montre aisément que les

variables aléatoires  $R_i$  (notée  $R$  par la suite) suivent la densité de probabilité  $f(R)$  suivante :

$$f(R) = \nu_c \exp(-\nu_c R). \quad (4.42)$$

Le libre parcours moyen  $\lambda_c$  (distance moyenne entre deux collisions) est simplement obtenu en exprimant le premier moment de la densité de probabilité ci-dessus, à savoir :

$$\lambda_c = \int_0^{\infty} R f(R) dR = \frac{1}{\nu_c}. \quad (4.43)$$

**Modélisation** Par définition, la fonction de distribution cumulée, associée à cette densité de probabilité, est définie par la relation suivante :

$$p(\hat{R}) = \int_0^{\hat{R}} f(R) dR = 1 - \exp(-\nu_c \hat{R}). \quad (4.44)$$

Cette fonction de distribution cumulée exprime simplement la probabilité que la particule subisse une collision avec un objet diffusant durant un trajet de longueur  $\hat{R}$ . Cette fonction est donc nulle pour  $\hat{R} = 0$  et égale à 1 pour  $\hat{R} = \infty$ . La simulation numérique du processus de diffusion est réalisée par la méthode de la fonction de distribution cumulée inverse, obtenue en inversant la relation (4.44), soit :

$$\hat{R} = -\frac{1}{\nu_c} \ln[1 - p(\hat{R})]. \quad (4.45)$$

La fonction de distribution cumulée étant comprise entre 0 et 1, elle peut être assimilée à une variable aléatoire  $\xi$  comprise entre 0 et 1. En tirant une succession de variables aléatoires  $\xi_i$ , on peut ainsi déterminer une succession de trajets de longueur  $\hat{R}_i$  qui satisfait bien à la fonction de distribution (4.42) de notre problème :

$$\hat{R}_n = -\frac{1}{\nu_c} \ln[1 - \xi_n]. \quad (4.46)$$

Un exemple de tirage aléatoire utilisant la méthode de la distribution cumulative inverse, est proposé à la figure 4.11. La comparaison avec la distribution théorique (4.42) montre un excellent comportement de la méthode.

Concrètement, lorsque qu'une particule sonore  $n$  entre dans une zone d'encombrement, on lui associe une distance de collision  $R_n$  avec un objet de l'encombrement en appliquant la relation (4.46). Au fur et à mesure de la propagation de la particule sonore dans la zone d'encombrement, un test est réalisé pour déterminer si la distance cumulée  $d_n$  de la particule dans la zone d'encombrement est inférieure ou supérieure à  $R_n$ . Si la distance parcourue

$d_n$  est supérieure à la distance de collision  $R_n$  la particule entre en collision avec un objet. En mode « énergétique », l'énergie de la particule sonore est pondérée par le coefficient d'absorption moyen  $\alpha_c$  de l'encombrement et poursuit sa propagation dans une direction aléatoire (distribution uniforme<sup>5</sup>). En mode « aléatoire », un nouveau tirage aléatoire sur une variable uniforme  $u$  permet de déterminer si la particule est absorbée par l'objet diffusant (procédure identique à l'absorption par une paroi, cf. paragraphe 4.3.2), ou réfléchi dans une direction uniforme. Après chaque collision, la distance de propagation cumulée est remise à zéro, et un nouveau tirage est effectué pour déterminer la prochaine distance de collision. Quelque soit le mode de calcul choisi, tant que la distance parcourue par la particule est inférieure à la distance de collision, la particule poursuit son trajet sans changer de direction.

## 4.3 Modélisation des parois

### 4.3.1 Description physique

#### 4.3.1.1 Absorption, dissipation et transmission acoustique

Au contact d'une paroi, une onde sonore sera en partie réfléchi vers l'intérieur du domaine pour une part  $R$ , en partie dissipée par transformation de l'énergie acoustique en chaleur dans le matériau pour une part  $\beta$ , le reste étant transmis à travers le matériau dans le domaine adjacent pour une part  $\tau$ . Ce dernier coefficient est défini comme le facteur de transmission. Si  $W_i$  est la puissance incidente sur une paroi, alors une partie  $W_\tau = R W_i$  sera réfléchi, une partie  $W_d = \beta W_i$  sera dissipée dans le matériau, et une partie  $W_t = \tau W_i$  sera transmise à travers la cloison. Par construction<sup>6</sup>, les coefficients  $R$ ,  $\beta$  et  $\tau$  sont compris entre 0 et 1. Au final, le bilan énergétique de la paroi s'écrit alors :

$$R + \beta + \tau = 1. \quad (4.47)$$

Il est usuel de définir le coefficient d'absorption  $\alpha$  de la paroi comme la somme de la partie transmise  $\tau$  et de la partie dissipée  $\beta$ , sous la forme  $\alpha = \beta + \tau$ , de telle manière que le bilan énergétique ci-dessus s'écrit :

$$R = 1 - \alpha. \quad (4.48)$$

Le coefficient d'absorption d'un matériau peut être mesuré en utilisant les procédures standardisées ISO

<sup>5</sup>Comme pour une source omnidirectionnelle, cf. paragraphe 4.1.2.

<sup>6</sup>Il ne peut pas y avoir de création d'énergie, ni plus d'absorption que d'énergie incidente.

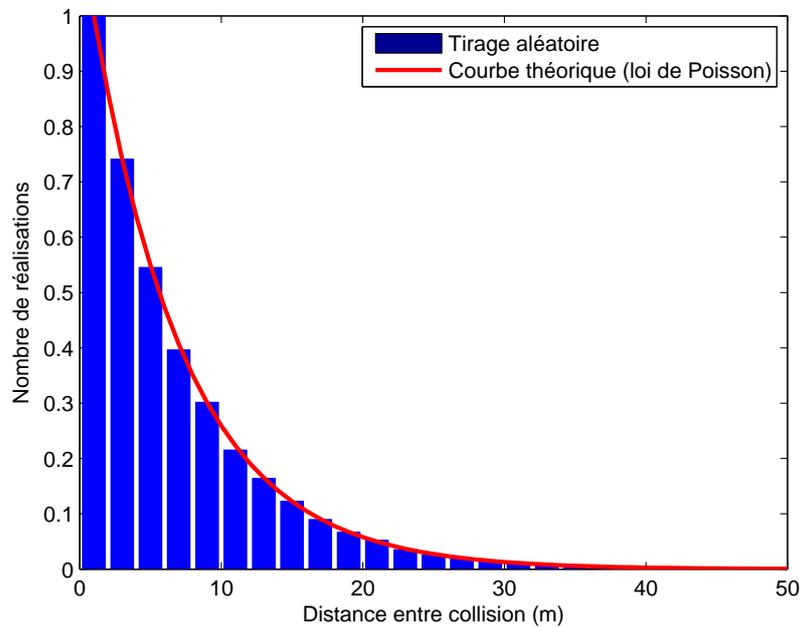


Figure 4.11 — Distribution des trajets  $\hat{R}_i$  obtenus par la méthode de la fonction de distribution cumulée inverse et comparaison avec la distribution théorique (4.42). Simulation réalisée sous MATLAB<sup>®</sup>.

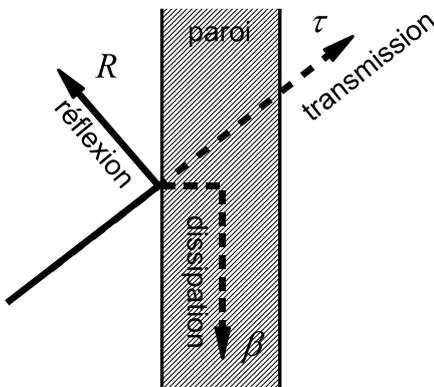


Figure 4.13 — Illustration des mécanismes de réflexion, d'absorption, de dissipation et de transmission acoustique par une paroi.

354 pour la méthode en chambre réverbérante [16] et ISO 10534 (parties 1 et 2) pour la méthode utilisant un tube d'impédance [10, 13]. En ce qui concerne le coefficient de transmission acoustique, le lecteur pourra se reporter aux différentes parties de la norme relative la transmission aérienne (parties 1, 3, 4 et 5, [11, 9, 14, 15]).

#### 4.3.1.2 Diffusion acoustique

D'autre part, suivant la forme, la taille et la distribution des irrégularités de la paroi, l'onde sonore pourra être réfléchi simultanément dans la direction spéculaire et dans d'autres directions. En acoustique des salles, il est commun de considérer qu'une fraction  $1 - s$  de l'énergie sonore sera réfléchi dans la direction de réflexion spéculaire, tandis qu'une fraction  $s$  de l'énergie sera réfléchi dans les autres directions de l'espace, suivant une loi de réflexion caractérisée par les irrégularités de la paroi [6]. Dans ce dernier cas, on parle alors de *réflexion diffuse*,  $s$  étant appelé coefficient de diffusion (*scattering coefficient* en anglais). En acoustique des salles, de nombreuses études théoriques et expérimentales sont actuellement en cours pour caractériser ou mesurer ces lois de réflexion [33, 5]. Toutefois, la pratique courante consiste à utiliser une loi de Lambert pour décrire une réflexion diffuse. La valeur du coefficient de diffusion  $s$  peut quant à elle être obtenue par une procédure de mesure standardisée (norme ISO 17497-1 de 2004 [17]).<sup>7</sup>

<sup>7</sup>Il est important de signaler que ce coefficient de diffusion est différent du coefficient de diffusion  $\delta$  (*diffusion coefficient* en anglais) proposé par l'AES, et qui définit quant à lui la « qualité de diffusion » d'une surface en comparant la réponse polaire (loi de réflexion) de la surface en question à une distribution uniforme. La mesure de ce coefficient fait actuellement l'objet de la rédac-

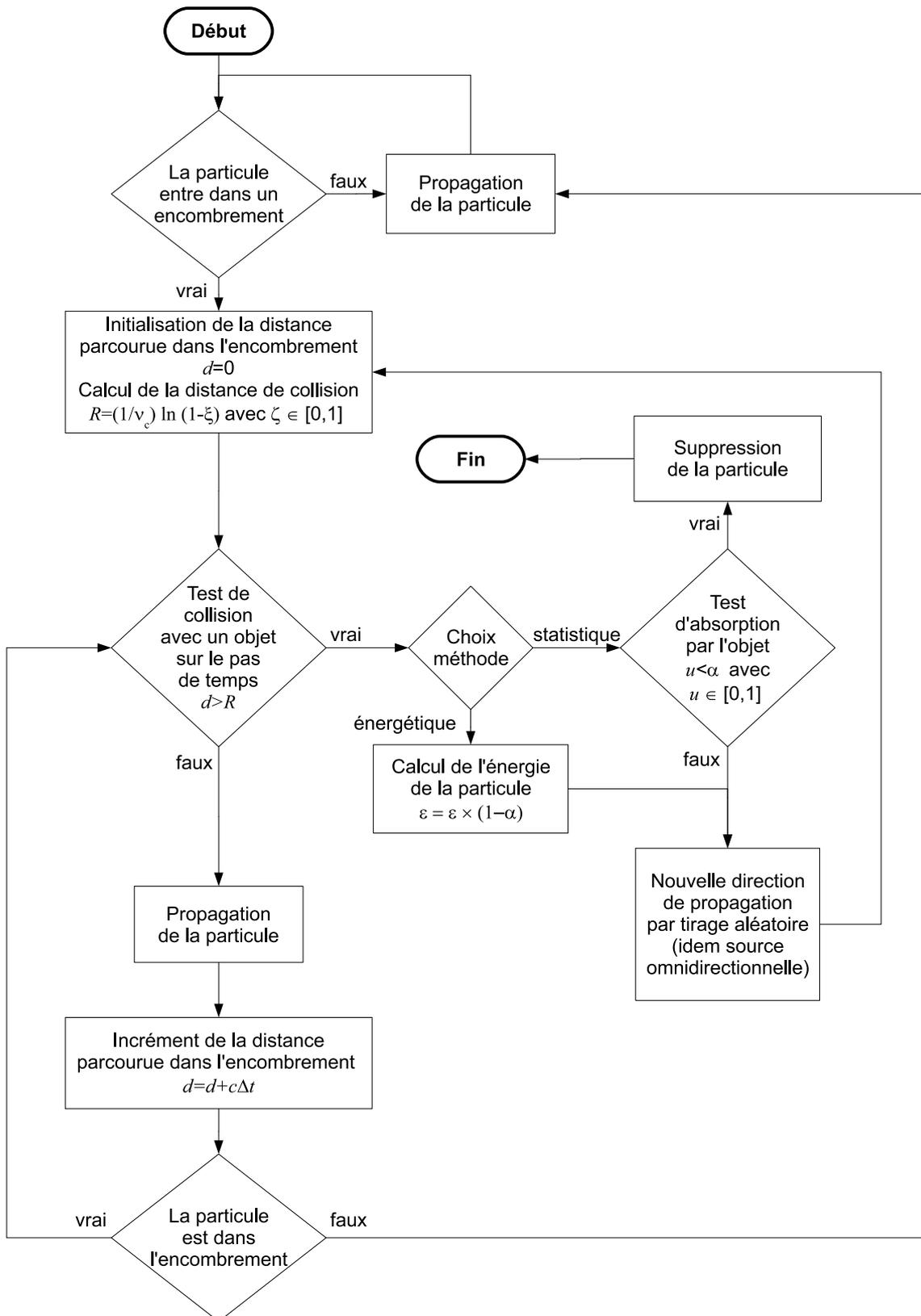
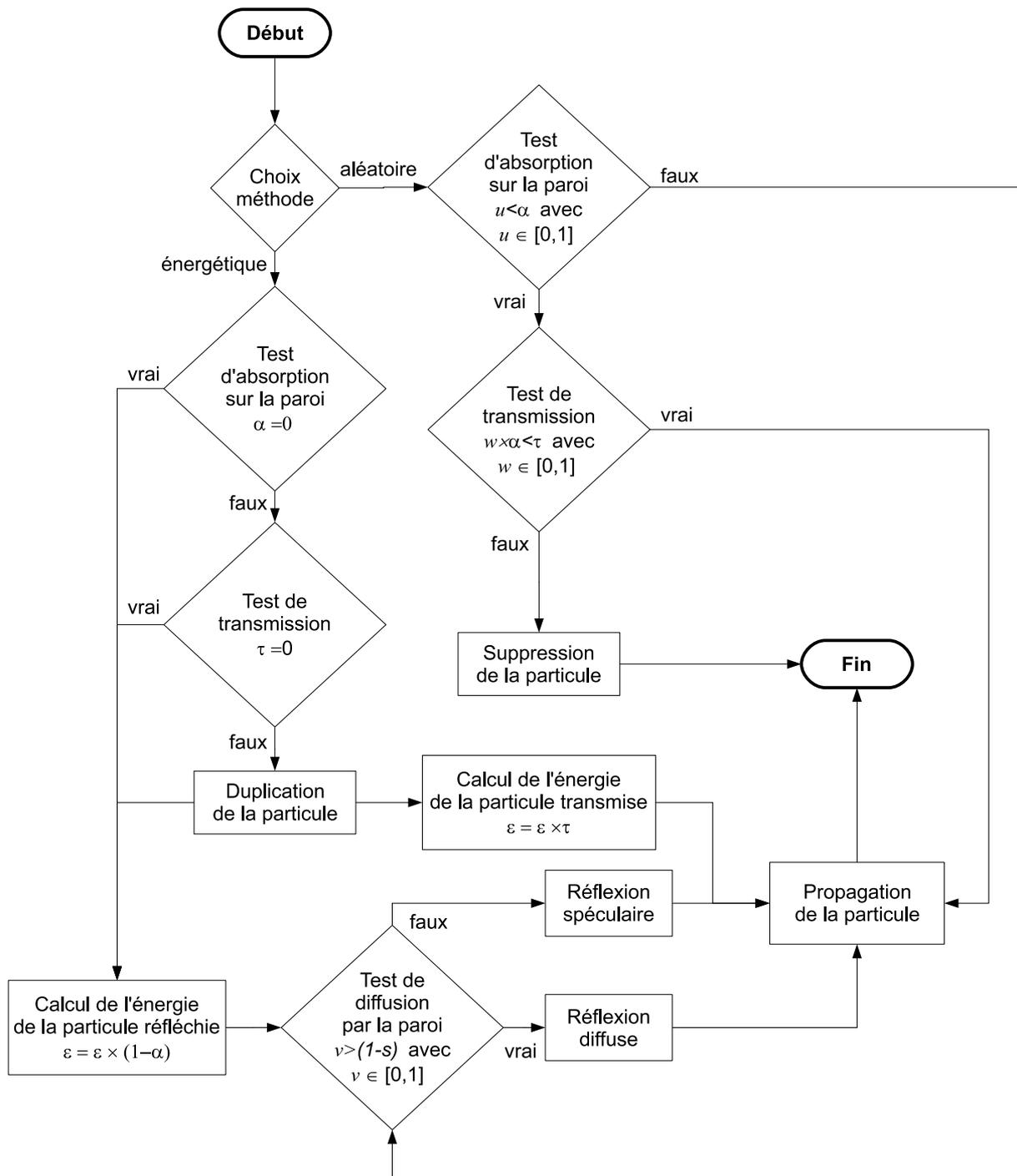


Figure 4.12 — Organigramme de la prise en compte de la diffusion d'une particule sonore par un objet de l'encombrement.



**Figure 4.14** — Organigramme de la prise en compte de la réflexion, de l'absorption et de la transmission acoustique par une paroi. Le coefficient  $\alpha$  est défini comme le coefficient d'absorption de la paroi tel que  $\alpha = \beta + \tau$ ,  $\beta$  étant le coefficient de dissipation dans la paroi, et  $\tau$  le coefficient de transmission. La réflexion spéculaire/diffuse est modélisée par le coefficient de diffraction  $s$ .

### 4.3.2 Modélisation de la réflexion acoustique

**Modélisation « aléatoire »** Dans un premier temps, une particule sonore entrant en collision avec une paroi pourra être soit absorbée par la paroi (avec une probabilité  $\alpha$ ), soit réfléchi dans une nouvelle direction de propagation (avec une probabilité  $R = 1 - \alpha$ ). En pratique, le choix absorption/réflexion est réalisé en tirant aléatoirement un nombre  $u$  entre 0 et 1, suivant une distribution uniforme. Si ce nombre est inférieur à  $\alpha = 1 - R$  (au point considéré), la particule est absorbée et disparaît du milieu de propagation. Si ce nombre est supérieur à  $\alpha = 1 - R$ , la particule est réfléchi et poursuit sa propagation dans une nouvelle direction de l'espace.

Dans un second temps, pour déterminer le type de réflexion (spéculaire ou diffuse), un nouveau tirage aléatoire  $v$  est réalisé entre 0 et 1. Si ce nombre est inférieur à la valeur de  $1 - s$  (i.e.  $v < (1 - s)$ ) au point considéré, la particule est réfléchi spéculairement, conformément aux lois de Snell-Descartes bien connues. Dans le cas contraire (i.e.  $v > (1 - s)$ ), la réflexion est diffuse. Dans ce dernier cas, il est faut déterminer la direction de réflexion diffuse (cf. paragraphe 4.3.3).

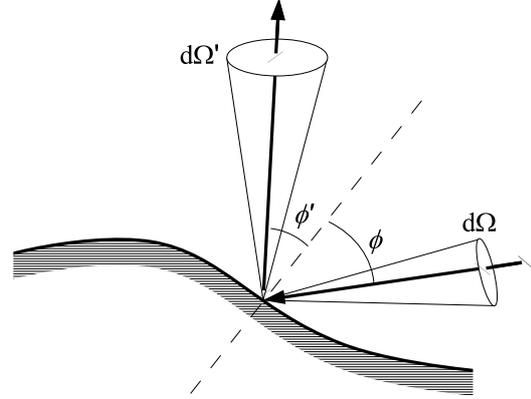
**Modélisation « énergétique »** Lorsqu'une particule entre en collision avec une paroi, son énergie  $\epsilon$  est pondérée par le coefficient de réflexion  $R = 1 - \alpha$ . Le choix d'une réflexion spéculaire ou diffuse, en fonction du coefficient de diffraction, est identique à la méthode « aléatoire » : un tirage aléatoire  $v$  est réalisé entre 0 et 1. Si ce nombre est inférieur à la valeur de  $1 - s$ , la particule est réfléchi spéculairement. Dans le cas contraire, la réflexion est diffuse, dans une direction à déterminer (cf. paragraphe 4.3.3). Un traitement entièrement « énergétique » serait possible en dupliquant la particule en deux particules, la première étant réfléchi dans la direction spéculaire et la seconde dans la direction diffuse (cf. remarque à la page 41).

### 4.3.3 Modélisation des lois de réflexion

#### 4.3.3.1 Formalisme

Considérons une particule sonore incidente sur une paroi, dont la direction incidente est définie par les coordonnées sphériques  $(\theta, \phi)$  (figure 4.15). Cette particule a une probabilité  $P(\theta, \phi; \theta', \phi') \equiv P(\Omega, \Omega')$  d'être réfléchi dans l'angle solide élémentaire  $d\Omega'$  =

tion de la partie 2 de la norme ISO 17497). Dans certains logiciels de simulation acoustique, le coefficient de diffraction  $s$  est parfois nommé à tort, coefficient de diffusion.



**Figure 4.15** — Géométrie élémentaire d'une réflexion par une paroi, en 3D :  $\phi$  et  $\phi'$  désignent respectivement les angles d'incidences et de réflexion par rapport à la normale à la paroi. Les angles  $\theta$  et  $\theta'$  dans le plan tangent, des directions d'incidence et de réflexion, ne sont pas représentés ( $\phi, \phi' \in [0, \pi/2]$  et  $\theta, \theta' \in [0, 2\pi]$ ).

$\sin \phi' d\phi' d\theta'$  [18, 19, 20, 27].  $P(\Omega, \Omega') d\Omega'$  représente en fait la fraction de l'intensité sonore incidente qui est réfléchi dans l'angle solide  $d\Omega'$ . Soit  $j(\theta, \phi)$  le flux incident de particules. Le flux de particules réfléchies  $j'(\theta', \phi')$  a pour expression :

$$j'(\theta', \phi') \cos \phi' = \int P(\theta, \phi; \theta', \phi') j(\theta, \phi) \cos \phi d\Omega. \quad (4.49)$$

En définissant la loi de réflexion  $R(\theta, \phi; \theta', \phi') \equiv R(\Omega, \Omega')$  de la manière suivante

$$R(\theta, \phi; \theta', \phi') = \frac{P(\theta, \phi; \theta', \phi')}{\cos(\phi')}, \quad (4.50)$$

la relation (4.51) s'écrit encore

$$j'(\theta', \phi') = \int R(\theta, \phi; \theta', \phi') j(\theta, \phi) \cos \phi d\Omega. \quad (4.51)$$

Il est important de souligner la différence entre la loi de réflexion  $R$  et la probabilité  $P$ .  $P$  désigne une probabilité de réflexion d'une particule sonore par angle solide élémentaire, tandis que  $R$  désigne le flux d'énergie réfléchi dans une direction  $\phi'$ , pour un flux incident dans la direction  $\phi$ .

La première loi de la thermodynamique imposent la conservation du flux sur la surface (en l'absence d'absorption). La loi de réflexion  $R(\Omega, \Omega')$  doit donc vérifier la condition suivante :

$$\int P(\Omega, \Omega') d\Omega' = \int R(\Omega, \Omega') \cos \phi' d\Omega' = 1, \quad (4.52)$$

ou encore en coordonnées sphériques,

$$\iint R(\theta, \phi; \theta', \phi') \cos \phi' \sin \phi' d\phi' d\theta' = 1. \quad (4.53)$$

La seconde loi de la thermodynamique impose cette fois :

$$\int P(\Omega, \Omega') d\Omega = \int R(\Omega, \Omega') \cos \phi d\Omega = 1, \quad (4.54)$$

ou encore en coordonnées sphériques

$$\iint R(\theta, \phi; \theta', \phi') \cos \phi \sin \phi d\phi d\theta = 1. \quad (4.55)$$

D'autre part, le principe de réciprocité impose que

$$R(\Omega, \Omega') = R(\Omega', \Omega). \quad (4.56)$$

Malgré quelques études récentes, et hormis la loi de Lambert largement utilisée pour modéliser des réflexions diffuses [25], il n'existe pas à notre connaissance d'autres lois de réflexion pour des parois réelles. Dans le code SPPS, nous proposons toutefois des modes de réflexion arbitraires, plus ou moins réels, présentés à la figure 4.16 et détaillées ci-dessous.

#### 4.3.3.2 Réflexion spéculaire

**Description physique** Le mode de réflexion le plus simple est défini par la réflexion spéculaire (figure 4.16(a)), et peut s'écrire en trois dimensions :

$$R(\theta, \phi; \theta', \phi') = 2\delta(\theta - \theta' \pm \pi) \delta(\sin^2 \phi - \sin^2 \phi'), \quad (4.57)$$

où les couples  $(\theta, \phi)$  et  $(\theta', \phi')$  désignent respectivement les directions incidentes et réfléchies des particules sonores sur la paroi. Bien que la forme de cette expression ne soit pas conventionnelle, cette relation vérifie les lois de Snell-Descartes, ainsi que les conditions (4.54) à (4.56).

**Modélisation** D'un point de vue numérique, la simulation de cette loi de réflexion ne pose pas de problème majeur, puisque les angles d'incidence de chaque particule sur une paroi sont connus. La modélisation est identique pour les approches « aléatoire » et « énergétique ».

#### 4.3.3.3 Réflexion uniforme

**Description physique** Une loi de réflexion uniforme produit une distribution des angles de réflexion  $P(\Omega')$  est équiprobable. Il ne faut pas confondre avec la loi de réflexion de Lambert, pour lequel l'uniformité est vérifiée par la loi de réflexion. Dans le cas d'une loi uniforme, les particules sont réfléchies uniformément dans tout le demi-espace, indépendamment de l'angle d'incidence. Dans ces conditions, la densité de réflexion après normalisation s'écrit :

$$P(\Omega') d\Omega' = P(\theta', \phi') d\Omega' = \left[ \frac{1}{2\pi} d\theta' \right] [\sin \phi' d\phi'], \quad (4.58)$$

ou encore

$$R(\theta', \phi') = \frac{1}{2\pi \cos \phi'}, \quad (4.59)$$

pour  $\theta' \in [0, 2\pi]$  et  $\phi' \in [0, \pi/2]$ .

**Modélisation** D'un point de vue numérique, la détermination de l'angle de réflexion est obtenue, là encore, par la méthode de la fonction de distribution cumulative inverse,

$$f(\hat{\phi}) = \int_0^{\hat{\phi}} \sin \phi' d\phi' = 1 - \cos \hat{\phi}, \quad (4.60)$$

ce qui donne

$$\hat{\phi} = \cos^{-1}(1 - u), \quad (4.61)$$

où  $u$  est un nombre aléatoire compris entre 0 et 1.

#### 4.3.3.4 Réflexion Lambert

**Description physique** Au sens de la lumière, une surface parfaitement diffuse est une surface qui paraît aussi lumineuse quelque soit l'angle d'observation, et quelque soit l'angle d'incidence. Au sens de l'acoustique, une surface parfaitement diffuse réfléchira la même énergie dans toutes les directions quelque soit l'angle d'incidence. D'un point de vue mathématique, cette condition impose que la loi de réflexion  $R$  soit indépendante de la direction de réflexion, donc de la direction d'incidence (par réciprocité). Après normalisation la loi de réflexion s'écrit :

$$R(\theta, \phi; \theta', \phi') = \frac{1}{2\pi} \times 2. \quad (4.62)$$

Le coefficient  $1/2\pi$  est lié à la normalisation suivant l'angle  $\theta'$  (distribution uniforme entre 0 et  $2\pi$ ). Le second coefficient (facteur 2) est lié à la normalisation suivant l'angle  $\phi'$ . La probabilité  $P(\Omega') d\Omega'$  se réduit donc à

$$\begin{aligned} P(\Omega') d\Omega' &= R \cos \phi' \sin \phi' d\theta' d\phi' \quad (4.63) \\ &= \left[ \frac{1}{2\pi} d\theta' \right] [2 \cos \phi' \sin \phi' d\phi'] \quad (4.64) \end{aligned}$$

où l'expression  $\cos \phi'$  est relative à la loi de Lambert. Il est aisé de vérifier que  $R$  vérifie les conditions (4.54) à (4.56). Il est important de souligner la différence entre une surface aléatoire et une surface « uniforme ». La première est conditionnée par une loi de réflexion  $R$  aléatoire uniforme, tandis que la seconde (cf. le paragraphe suivant) est définie par une probabilité  $P$  uniforme.

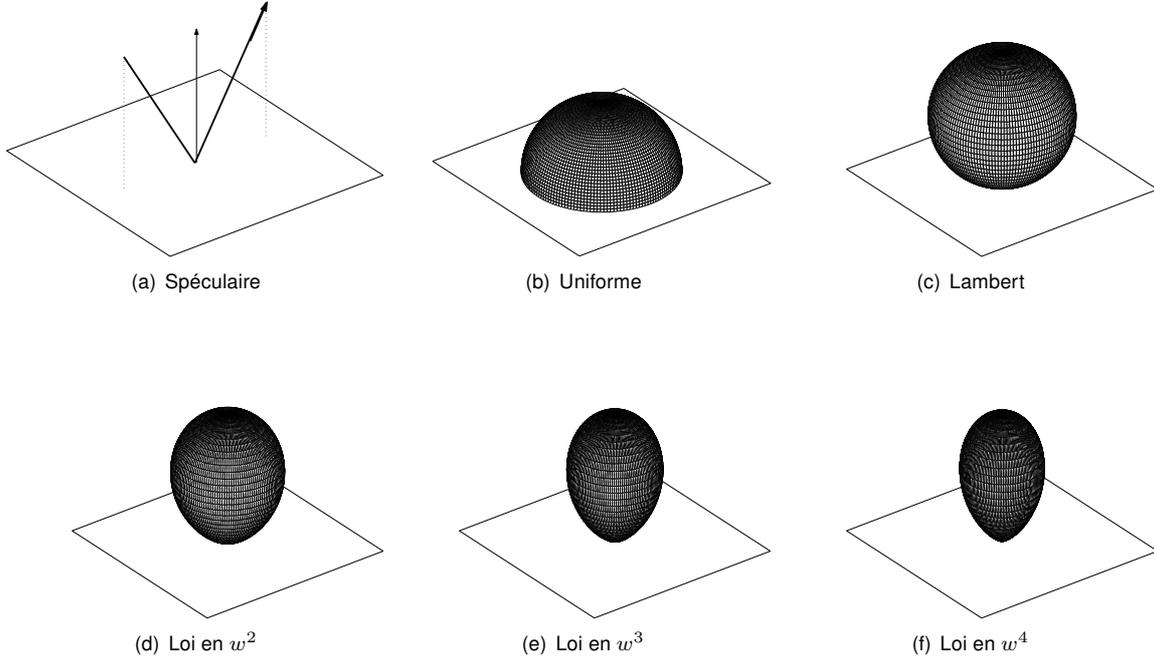


Figure 4.16 — Illustration des lois de réflexion proposées dans le code SPPS.

**Modélisation** D'un point de vue numérique, le tirage aléatoire des angles de réflexion (en trois dimensions) doit être réalisé en accord avec la distribution  $P(\Omega')$ , à savoir  $P(\phi')$  dans notre cas. Appliquant la méthode de la fonction de distribution cumulative inverse, la probabilité  $f(\hat{\phi})$  qu'une particule sonore soit réfléchié dans un angle  $\phi'$  compris entre 0 et  $\hat{\phi}$  est donnée par la relation suivante<sup>8</sup> :

$$f(\hat{\phi}) = 2 \int_0^{\hat{\phi}} \cos \phi' \sin \phi' d\phi' = \sin^2 \hat{\phi}. \quad (4.65)$$

Cette probabilité étant comprise entre 0 et 1, le choix de l'angle  $\hat{\phi}$  est réalisé en tirant aléatoirement un nombre  $u \in [0, 1]$ , tel que :

$$\hat{\phi} = \sin^{-1} \sqrt{u} = \cos^{-1} (1 - u)^{\frac{1}{2}}. \quad (4.66)$$

#### 4.3.3.5 Réflexion normale en $w^n$

**Description physique** Considérons une loi de réflexion indépendante de la direction incidente et définie uniquement par l'angle de réflexion  $\phi'$  autour de la normale à la paroi ( $\theta'$  étant uniforme entre 0 et  $2\pi$ ),

<sup>8</sup>Il est important de noter que d'autres auteurs [4, 7, 26, 34] mentionnent d'autres relations. La bonne formulation est fonction de la convention des angles. Il est également possible que certaines relations qui sont proposées ne soient pas exactes.

de la forme :

$$R(\Omega') = R(\theta', \phi') \quad (4.67)$$

$$= \frac{1}{2\pi} \times (n+1) \cos^{n-1} \phi' \quad (4.68)$$

$$= \frac{n+1}{2\pi} w^{n-1}. \quad (4.69)$$

où  $n$  est un entier positif. La quantité notée  $w = \cos \phi'$  n'est autre que la projection de la direction de réflexion sur la normale à la paroi. Conformément aux notations de ce document, la probabilité  $P$  aura donc pour expression :

$$P(\Omega') d\Omega' = P(\phi') \sin \phi' d\theta' d\phi' \quad (4.70)$$

$$= \left[ \frac{1}{2\pi} d\theta' \right] \quad (4.71)$$

$$\times [(n+1) \cos^n \phi' d\phi'] \quad (4.72)$$

$$= \frac{n+1}{2\pi} w^n d\theta' d\phi' \quad (4.73)$$

ce qui justifie l'appellation en *loi en  $w^n$* . Cette forme de réflexion est identique à celle introduite dans [27]. On peut remarquer que ce type de loi est une forme généralisée de la loi de Lambert ( $n = 1$ ) et de la loi uniforme ( $n = 0$ ).

**Modélisation** D'un point de vue numérique, le tirage aléatoire des angles de réflexion est réalisé en

appliquant la méthode de la fonction de distribution cumulative inverse. La probabilité  $f(\hat{\Omega})$  (c'est-à-dire la probabilité  $f(\hat{\phi})$  puisque la direction de réflexion ne dépend que l'angle par rapport à la normale) qu'une particule sonore soit réfléchié dans un angle solide élémentaire compris entre 0 et  $\hat{\Omega}$  est alors donnée par la relation

$$f(\hat{\phi}) = (n + 1) \int_0^{\hat{\phi}} \cos^n \phi' \sin \phi' d\phi' = 1 - \cos^{n+1} \hat{\phi}. \quad (4.74)$$

Cette probabilité étant comprise entre 0 et 1, le choix de l'angle  $\hat{\phi}$  est réalisé en tirant aléatoirement un nombre  $u \in [0, 1]$ , tel que :

$$\hat{\phi} = \cos^{-1} (1 - u)^{\frac{1}{n+1}}. \quad (4.75)$$

#### 4.3.4 Modélisation de la transmission acoustique

**Description physique** Comme nous l'avons indiqué plus haut, la partie de la puissance qui n'est pas réfléchié par la paroi est soit dissipée dans la paroi, soit transmise. La transmission acoustique est alors définie par le facteur de transmission  $\tau$ , défini comme le rapport de la puissance transmise  $W_t$  par la paroi sur la puissance incidente  $W_i$ . S'il n'y a pas de dissipation dans la paroi (i.e. tout est transmis à travers la paroi), alors  $\tau = \alpha$ . Dans le cas contraire,  $\tau < \alpha$ . En pratique, la transmission acoustique est plutôt définie par l'indice d'affaiblissement de la paroi, qui est fonction du facteur de transmission à travers la relation suivante :

$$R = 10 \log \left( \frac{W_i}{W_t} \right) = -10 \log \tau. \quad (4.76)$$

**Modélisation « aléatoire »** La modélisation est similaire à celle utilisée pour la réflexion acoustique. Pour déterminer si la particule sonore est dissipée ou transmise par la cloison, il est nécessaire de tirer un nombre  $w$  entre 0 et  $\alpha$ . Si ce nombre est inférieur à  $\tau$ , la particule est transmise et conserve sa direction de propagation. Dans le cas contraire la particule est dissipée et disparaît du milieu de propagation.

**Modélisation « énergétique »** La modélisation énergétique est réalisée pondérant l'énergie de la particule une fois transmise par la cloison, par le coefficient  $\tau$ .



*En mode « énergétique », en présence simultanée de transmission et de réflexion acoustique, il y a dans notre cas duplication de la particule sonore.*

*La particule dupliquée est transmise, avec pondération de l'énergie et en conservant sa direction initiale de propagation. La particule originale est réfléchié, avec pondération de l'énergie, la direction de réflexion étant choisies conformément à la méthode présentée au paragraphe 4.3.2 en mode « énergétique ». Une solution envisageable pour avoir un traitement entièrement « énergétique » de la réflexion sur une paroi serait de dupliquer la particule sonore en trois particules, la première étant transmise, les deux autres étant réfléchiés dans les directions spéculaire et diffuse respectivement. Cela a toutefois l'inconvénient d'augmenter de manière considérable la durée des calculs.*

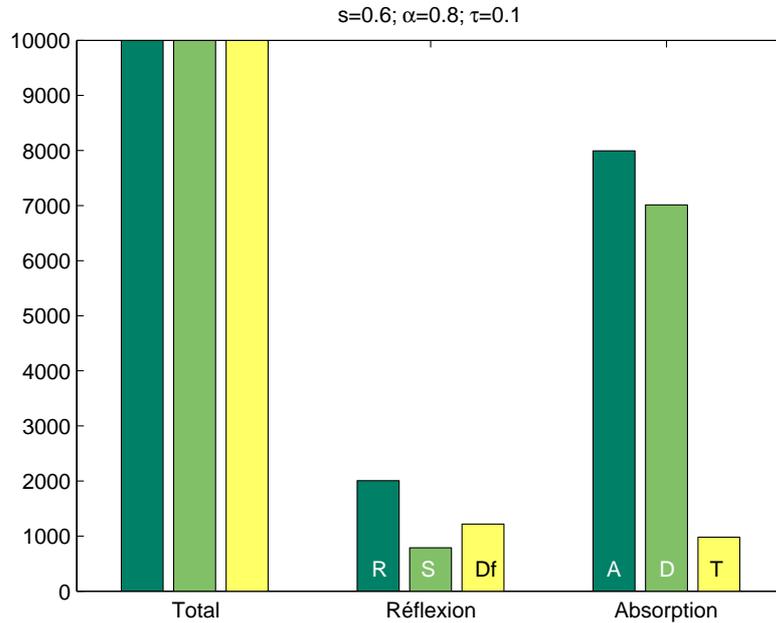
#### 4.3.5 Vérification de la modélisation des parois

La figure 4.17 illustre le résultat de la procédure de réflexion (réflexion=spéculaire/diffus, absorption=dissipation/transmission) en mode « aléatoire », avec les paramètres acoustiques suivants : coefficient de diffraction  $s = 0.6$ , coefficient d'absorption  $\alpha = 0.8$ , coefficient de transmission  $\tau = 10^{-R/10} = 0.1$  (indice d'affaiblissement  $R = 10$  dB). Avec 10000 réalisations, les différents phénomènes sont retrouvés (en termes de nombre de réalisations) avec les mêmes proportions que les paramètres acoustiques imposés.

### 4.4 Calcul des niveaux sonores aux points d'observation

Dans le code SPPS deux types de récepteurs sont considérés :

- les récepteurs « volumiques » modélisent les récepteurs « classiques » dits ponctuels. La notion de « point récepteur » n'étant pas applicable dans le code SPPS, puisqu'en théorie la probabilité qu'une particule sonore traverse un récepteur ponctuel est nul, il est nécessaire de donner un volume au point récepteur, pour comptabiliser le nombre de particules l'ayant traversé, et en déduire ainsi la densité d'énergie au point d'observation. Pour un récepteur ponctuel, le code SPPS retourne le niveau de pression acoustique, à chaque pas de temps du calcul et pour chaque bande de fréquence considérée ;
- les récepteurs « surfaciques » sont des éléments de surface (en principe des faces de la scène) sur lesquels sont calculés les intensités sonores incidentes, ce qui permet ensuite de construire



**Figure 4.17** — Vérification de la procédure de modélisation des phénomènes d'absorption (dissipation et transmission) et réflexion (spéculaire et diffus). Seul le processus de tirage aléatoire est modélisé. Répartition des réalisations en fonction des phénomènes. Le premier groupe de résultats « Total » correspond à la somme du nombre de réalisations (nombre initial, réflexion+absorption, spéculaire+diffus+dissipation+transmission) —ces trois quantités ont donc la même valeur—. Le second groupe correspond au processus de « Réflexion », réparti entre le nombre total de réflexion (R), les réflexions spéculaires (S) et la diffusion (Df). Le troisième groupe correspond au processus de « Absorption », réparti entre le nombre total d'absorption (A), la dissipation (D) et la transmission (T). Simulations réalisées avec 10000 réalisations et les paramètres acoustiques suivant :  $s = 0.6$ ,  $\alpha = 0.8$ ,  $\tau = 10^{-R/10} = 0.1$  ( $R = 10$  dB). Simulation réalisée sous MATLAB®.

des cartographies acoustiques. Pour un récepteur surfacique, le code SPSS retourne le niveau d'intensité sonore, à chaque pas de temps du calcul et pour chaque bande de fréquence considérée.

En parallèle, le code de calcul détermine également le niveau de pression acoustique global dans le modèle, en sommant les contributions de chaque particule, à chaque pas de temps et pour chaque bande de fréquence.

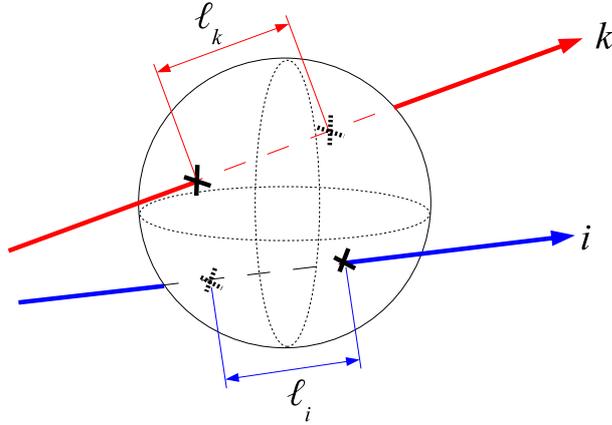
#### 4.4.1 Calcul du niveau de pression acoustique au niveau d'un récepteur « volumique »

La quantité d'énergie  $E_{\text{rec}}^j(n)$  (en J ou W.s) dans la bande de fréquence  $j$  reçue au niveau d'un récepteur volumique, au pas de temps  $n$  (i.e. au temps  $n\Delta t$ ) est égale à la somme des énergies  $\varepsilon_i^j$  apportées par chaque particule  $i$  dans la bande de fréquence  $j$ , traversant le volume récepteur au cours du pas de

temps  $n$  (figure 4.18) :

$$E_{\text{rec}}^j(n) = \sum_i^{N_0} \varepsilon_i^j = \sum_i^{N_0} \frac{W}{N} \varepsilon_i^j \times \Delta t_i, \quad (4.77)$$

où  $N_0$  est le nombre total de particules traversant le volume récepteur et  $\Delta t_i$  le temps de présence de la particule  $i$  dans le volume récepteur ( $\Delta t_i < \Delta t$ ), et  $\varepsilon_i^j$  le coefficient de pondération (compris entre 0 et 1) associé à la particule  $i$  dans la bande de fréquence  $j$ . Si le mode de calcul est « aléatoire », le coefficient  $\varepsilon_i^j$  est constant et égale à l'unité (1). Si le mode de calcul est « énergétique », le coefficient  $\varepsilon_i^j$  traduit la perte cumulée d'énergie dans la bande de fréquence  $j$  de la particule  $i$  tout au long de son trajet du fait des phénomènes physiques rencontrés (absorption par les parois et par les encombrements, absorption atmosphérique, transmission acoustique...). Le temps de présence  $\Delta t_i$  peut également être exprimé en fonction de la longueur du trajet de la particule  $i$  dans le volume récepteur, soit  $\ell_i$ , tel que  $\Delta t_i = \ell_i/c$ ,  $c$  étant la vitesse de la particule au niveau du point d'observa-



**Figure 4.18** — Principe du calcul du niveau de pression acoustique pour un récepteur « volumique ». La densité d'énergie dans le volume récepteur est calculée en sommant les contributions énergétiques de chaque particule traversant le récepteur. La contribution de la particule  $i$  est calculée à partir du trajet  $\ell_i$  de la particule dans le volume récepteur (relation (4.78)).

tion. En considérant un récepteur « volumique »<sup>9</sup> est défini par un volume sphérique de rayon  $r_{\text{rec}}$  (et de volume  $V_{\text{rec}}$ ), la **densité d'énergie**  $w_{\text{rec}}^j(n)$  (en  $\text{J}/\text{m}^3$ ) dans le volume récepteur, pour la bande de fréquence  $j$ , est donné par :

$$w_{\text{rec}}^j(n) = \frac{E_{\text{rec}}(n)}{V_{\text{rec}}} = \frac{W}{N} \frac{1}{V_{\text{rec}}} \sum_i^{N_0} \epsilon_i^j \ell_i. \quad (4.78)$$

L'**intensité sonore**  $I_{\text{rec}}^j(n)$  (en  $\text{W}/\text{m}^2$ ) au point récepteur est quant à elle donnée par :

$$I_{\text{rec}}^j(n) = c \times w_{\text{rec}}^j(n) = \frac{W}{N} \frac{1}{V_{\text{rec}}} \sum_i^{N_0} \epsilon_i^j \ell_i. \quad (4.79)$$

⚠ La donnée de sortie, notée  $P2_{\text{rec}}^j(n)$ , du code de calcul SPPS pour un récepteur ponctuel, n'est pas exactement l'intensité sonore  $I_{\text{rec}}^j(n)$ , mais l'intensité sonore pondérée par l'impédance caractéristique de l'air, produit de la masse volumique  $\rho_0$  par la célérité  $c$ . Cette quantité est donc homogène au carré de la pression acoustique (i.e. en  $\text{Pa}^2$ ) :

$$P2_{\text{rec}}^j(n) = \rho_0 c \times I_{\text{rec}}^j(n) \quad (4.80)$$

$$= \rho_0 c^2 \times w_{\text{rec}}^j(n) \quad (4.81)$$

$$= \rho_0 c \frac{W}{N} \frac{1}{V_{\text{rec}}} \sum_i^{N_0} \epsilon_i^j \ell_i. \quad (4.82)$$

<sup>9</sup>La définition du volume récepteur est un élément crucial pour la qualité et la représentativité des résultats. Il doit être suffisamment grand pour comptabiliser des particules sonores lors de leur propagation, mais pas trop pour que la densité d'énergie calculée au point d'observation soit représentative.

Le **niveau d'intensité sonore**  $L_{1,\text{rec}}^j(n)$  et le **niveau de pression acoustique**  $SPL_{\text{rec}}^j(n)$  (en dB) peuvent ensuite être déduits de l'intensité sonore par les relations suivantes :

$$L_{1,\text{rec}}^j(n) = 10 \log \left( \frac{P2_{\text{rec}}^j(n)}{\rho_0 c I_0} \right), \quad (4.83)$$

et

$$SPL_{\text{rec}}^j(n) = 10 \log \left( \frac{P2_{\text{rec}}^j(n)}{p_0^2} \right), \quad (4.84)$$

où  $I_0 = 10^{-12} \text{ W}/\text{m}^2$  et  $p_0 = 20 \times 10^{-6} \text{ Pa}$  désignent l'intensité sonore et la pression acoustique de référence. Chacune des quantités précédentes (énergie, densité d'énergie, intensité et niveaux sonores) sont calculées pour chaque bande de fréquence.

#### 4.4.2 Vecteur intensité au niveau d'un récepteur « volumique »

Le vecteur **intensité sonore**  $\mathbf{I}_{\text{rec}}^j(n)$  (en  $\text{W}/\text{m}^2$  ou  $\text{J}/\text{m}^2 \cdot \text{s}$ ) au point récepteur, pour la bande de fréquence  $j$ , est défini comme la somme des densités d'énergie portées par le vecteur vitesse  $\mathbf{v}_i$  des particules (de norme  $c_i$ ) traversant le volume récepteur :

$$\mathbf{I}_{\text{rec}}^j(n) = \frac{W}{N} \frac{1}{V_{\text{rec}}} \sum_i^{N_0} \epsilon_i^j \ell_i \frac{\mathbf{v}_i}{c_i}. \quad (4.85)$$

⚠ Il est important de noter que la norme du vecteur flux d'énergie  $\mathbf{I}_{\text{rec}}^j(n)$  ne donne pas l'intensité sonore  $I_{\text{rec}}^j(n)$ .

#### 4.4.3 Calcul du niveau de pression acoustique « latéral » au niveau d'un récepteur « volumique »

Pour le calcul de certains paramètres acoustiques, tels que ceux basés sur l'énergie latérale (LF avec une pondération en  $|\cos \theta|$  et LFC avec une pondération en  $\cos^2 \theta$ ), il est nécessaire de considérer une pondération de l'intensité sonore en fonction de l'angle  $\theta$  entre la direction d'observation du récepteur ponctuel (en principe orienté vers la source sonore) et de la direction incidente des particules au niveau du récepteur. De ce fait, le code de calcul SPPS retourne également les deux quantités suivantes, homogène au carré de la pression acoustique (i.e. en  $\text{Pa}^2$ ) :

$$P2_{\text{rec},\cos \theta}^j(n) = \rho_0 c^2 \times w_{\text{rec},\cos \theta}^j(n) \quad (4.86)$$

$$= \rho_0 c \frac{W}{N} \frac{1}{V_{\text{rec}}} \quad (4.87)$$

$$\times \sum_i^{N_0} \epsilon_i^j \ell_i |\cos \theta_i|, \quad (4.88)$$

et

$$P2_{\text{rec}, \cos^2 \theta}^j(n) = \rho_0 c^2 \times w_{\text{rec}, \cos^2 \theta}^j \quad (4.89)$$

$$= \rho_0 c \frac{W}{N} \frac{1}{V_{\text{rec}}} \quad (4.90)$$

$$\times \sum_i^{N_0} \epsilon_i^j \ell_i \cos^2 \theta_i. \quad (4.91)$$

où  $\theta_i$  est l'angle correspondant pour la particule  $i$ .

#### 4.4.4 Calcul du niveau d'intensité sonore sur un récepteur « surfacique »

La **puissance**  $W_{\text{surf}}^j$  (en W) reçue par un élément de surface de taille  $\Delta S$  de normale  $\mathbf{n}$  est égale à la somme de l'énergie apportée par chaque particule  $i$  dans la bande de fréquence  $j$  par unité de temps  $\Delta t$ , au pas de temps  $n$  (i.e. au temps  $n\Delta t$ ), soit :

$$W_{\text{surf}}^j(n) = \sum_i^{N_0} \frac{\epsilon_i^j \mathbf{c}_i}{\Delta t} \cdot \mathbf{n} = \frac{W}{N} \sum_i^{N_0} \epsilon_i \cos \theta_i, \quad (4.92)$$

où  $\mathbf{c}_i$  désigne le vecteur vitesse (de norme  $c$ ) de la particule  $i$ ,  $\theta_i$  l'angle entre la normale  $\mathbf{n}$  de la surface et la direction de la particule, et où  $N_0$  est le nombre total de particules sonores entrant en collision avec l'élément de surface  $\Delta S$ .

L'**intensité sonore**  $I_{\text{surf}}^j(n)$  (en  $\text{W}/\text{m}^2$ ) reçue par l'élément de surface  $\Delta S$  au pas de temps  $n$  est égale à la puissance reçue divisée par la surface, soit :

$$I_{\text{surf}}^j(n) = \frac{W}{N} \frac{1}{\Delta S} \sum_i^{N_0} \epsilon_i^j \cos \theta_i. \quad (4.93)$$



La donnée de sortie du code SPPS pour un récepteur surfacique est égale à l'intensité sonore  $I_{\text{surf}}^j(n)$ .

Le **niveau d'intensité sonore**  $L_{\text{surf}}^j(n)$  (en dB) peut ensuite être calculé par la relation suivante :

$$L_{\text{surf}}^j(n) = 10 \log \frac{I_{\text{surf}}^j(n)}{I_0}, \quad (4.94)$$

où  $I_0 = 10^{-12} \text{ W}/\text{m}^2$  désigne l'intensité de référence.

#### 4.4.5 Calcul du niveau de pression acoustique sur un récepteur « surfacique »

De façon équivalente au calcul du niveau d'intensité sonore sur un récepteur « surfacique », la pression acoustique quadratique  $P2_{\text{surf}}^j(n)$  (en Pa) reçue par l'élément de surface  $\Delta S$  au pas de temps  $n$  est égale au produit de l'intensité sonore (sans pondération de

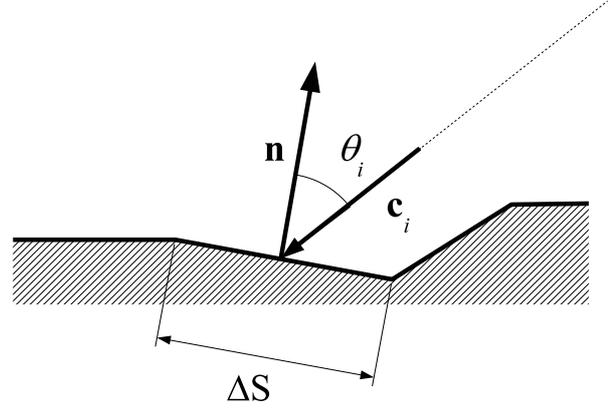


Figure 4.19 — Principe du calcul du niveau d'intensité acoustique au niveau d'un récepteur « surfacique ».

l'angle d'incidence sur la surface) par l'impédance caractéristique de l'air  $\rho_0 c$ , soit :

$$P2_{\text{surf}}^j(n) = \rho_0 c \frac{W}{N} \frac{1}{\Delta S} \sum_i^{N_0} \epsilon_i^j. \quad (4.95)$$



La donnée de sortie du code SPPS pour un récepteur surfacique est égale à la pression acoustique quadratique  $P2_{\text{surf}}^j(n)$ .

Le **niveau de pression acoustique**  $L_{\text{SPL}, \text{surf}}^j(n)$  (en dB) peut ensuite être calculé par la relation suivante :

$$L_{\text{SPL}, \text{surf}}^j(n) = 10 \log \frac{P2_{\text{surf}}^j(n)}{p_0^2}, \quad (4.96)$$

où  $p_0 = 20 \times 10^{-6} \text{ Pa}$  désigne la pression acoustique de référence.

#### 4.4.6 Calcul du niveau de pression acoustique global dans le modèle

L'intensité acoustique globale  $I_{\text{global}}^j(n)$  dans le modèle, pour la bande de fréquence  $j$ , est calculée en sommant les intensités portées par toutes les particules sonores présentes  $N_n$  dans le modèle au pas de temps  $n$  (i.e. au temps  $n\Delta t$ ) :

$$I_{\text{global}}^j(n) = \frac{W}{N} \frac{1}{V_{\text{rec}}} \sum_i^{N_n} \epsilon_i^j \ell_i. \quad (4.97)$$



La donnée de sortie, notée  $P2_{\text{global}}^j(n)$ , du code de calcul SPPS pour la valeur globale, n'est pas exactement l'intensité sonore  $I_{\text{global}}^j(n)$ , mais l'intensité

sonore pondérée par l'impédance caractéristique de l'air, produit de la masse volumique  $\rho_0$  par la célérité  $c$ . Cette quantité est donc homogène au carré de la pression acoustique :

$$P2_{\text{global}}^j(n) = \rho_0 c \frac{W}{N} \frac{1}{V_{\text{rec}}} \sum_i^{N_n} \epsilon_i^j \ell_i. \quad (4.98)$$

Le **niveau de pression acoustique**  $SPL_{\text{global}}^j(n)$  (en dB) peut ensuite être calculé par la relation suivante :

$$SPL_{\text{global}}^j(n) = 10 \log \left( \frac{P2_{\text{global}}^j(n)}{p_0^2} \right), \quad (4.99)$$

où  $p_0 = 20 \times 10^{-6}$  Pa désigne la pression acoustique de référence.



# Description et fonctionnement du code

## SPPS

### 5.1 Introduction

Ce chapitre ne donne qu'une description simplifiée des procédures utilisées dans le code de calcul. Ces informations sont destinées à éclairer l'utilisateur sur le fonctionnement général du code, notamment sur les algorithmes de modélisation des phénomènes physiques. Ce chapitre n'est pas une documentation « développeur ».

### 5.2 Principe général du calcul

#### 5.2.1 Pré-traitement du calcul grâce au maillage volumique du domaine

Le maillage du domaine par des mailles volumiques tétraédriques permet d'optimiser les calculs de propagation des particules sonores :

- une analyse de chaque face d'une maille, permet d'identifier les faces qui sont coplanaires avec les parois du domaine ou avec un encombrement du domaine ;
- en parcourant les mailles une à une, il est possible d'identifier les mailles adjacentes. Ainsi, lorsqu'une particule est présente dans une maille, il est possible de déterminer très rapidement les mailles qui vont être traversées au pas de temps suivant, et ainsi définir la possibilité d'une collision avec une face du domaine ou d'un encombrement. En parallèle, connaissant la maille courante, les tests de collision sont uniquement réalisés sur les faces de la maille et non

les faces du domaine, ce qui limite la durée des tests ;

- par ailleurs, en considérant la position du centre de la maille et le profil de célérité retenu, il est possible d'affecter une célérité moyenne dans la maille correspondante, laquelle sera affectée à la particule tout le temps qu'elle sera présente dans la maille en question.

 *Il est indispensable que le maillage soit réalisé pour tout le domaine de propagation. Si ce n'est pas le cas, une maille courante pourra ne pas avoir de mailles voisines bien que cette maille courante ne soit pas en bordure du domaine ou d'un encombrement. Une particule sonore située dans la maille courante ne pourra donc pas se propager aux mailles voisines. Si ce cas se présente, les particules sonores en question seront supprimées du domaine de propagation. Le fichier d'information du calcul, d'extension `.gabe`, précise le nombre de particules qui ont été supprimées en raison d'un « maillage incorrect ».*

#### 5.2.2 Principe du code de calcul

Le principe du code de calcul consiste à suivre chaque particule, l'une après l'autre, sur toute sa durée de vie (limitée par la durée de calcul fixée par l'utilisateur, ou si la particule sonore disparaît du domaine de propagation (par absorption, ou suite à un problème numérique)), pas de temps après pas de temps, maille après maille. Au contact avec une paroi du domaine, la particule subit une procédure de collision qui la dévie de sa trajectoire et modifie éventuellement son énergie. En entrant dans une zone

d'encombrement, la particule change d'état : elle peut également entrer en collision avec des objets de l'encombrement (existence statistique uniquement), avec dans ce cas une modification de la trajectoire et éventuellement de son énergie sonore. Enfin, les propriétés mêmes du domaine de propagation peuvent évoluer en fonction de la position dans l'espace et au cours du temps (présence d'un gradient de célérité par exemple). Cette modification peut alors entraîner une modification des propriétés de la particule (norme de la célérité du son ou déviation de trajectoire, par exemple).

### 5.2.3 Structures

Pour pouvoir suivre le déplacement des particules sonores, il est indispensable d'avoir accès à chaque instant de la procédure aux informations (données acoustiques et géométriques) des mailles, des faces et des particules sonores. Dans le code SPPS, ces informations sont organisées sous forme de structures, présentées ci-après (pour les plus importantes) :

#### 5.2.3.1 Structure d'une maille

Chaque maille est associée à une série d'informations :

- un indice ;
- l'indice des mailles voisines si elles existent ;
- les données de l'encombrement (si la maille fait parti d'un encombrement) ;
- un indice<sup>1</sup> de volume (0 si s'agit du volume principal) ;
- hauteur du centre de la maille par rapport à l'élément de surface le plus proche verticalement ;
- position du centre de la maille.

#### 5.2.3.2 Structure d'une face de maille

A chaque face du modèle (face réelle de la scène ou face d'un encombrement), sont également affectées plusieurs propriétés, utilisées tout au long de la procédure de calcul :

- indice des sommets des faces ;
- vecteur normal à la face ;
- informations sur le matériau associé (s'il existe) :

- indice associé au matériau dans le fichier de configuration XML ;
- loi de réflexion associé au matériau ;
- coefficient d'affaiblissement du matériau ;
- coefficient d'absorption du matériau ;
- coefficient de diffusion du matériau.

- informations sur l'encombrement associé (NULL si aucun) :

- indice de l'encombrement ;
- coefficient d'absorption des objets diffusants ;
- libre parcours moyen ;
- loi de diffusion des objets de l'encombrement.

- information sur le récepteur surfacique associé (s'il existe) :

- indice du récepteur surfacique associé (s'il existe)
- énergie actuelle (au pas de temps en cours) du récepteur surfacique.

#### 5.2.3.3 Structure d'une particule sonore

A chaque particule sonore, sont affectées plusieurs propriétés qui peuvent être amenées à évoluer au cours du temps, et/ou en fonction des domaines traversés. En particulier :

- position  $(x, y, z)$  de la particule dans le domaine ;
- direction de propagation  $(u, v, w)$  de la particule, de norme égale à la célérité du son ;
- indice de la maille courante de la particule ;
- pas de temps courant du calcul ;
- coordonnées du point de collision le plus proche sur la maille courante et indice de la face associée ;
- énergie (en cours) de la particule ;
- distance entre la particule et le point de prochaine collision avec un objet de l'encombrement ;
- état de la particule : information utilisée pour les statistiques du calcul (cf. paragraphe 7.5.4.4) ;

<sup>1</sup>Pour des applications futures.

- valeur limite de l'énergie portée par une particule. Si l'énergie de la particule devient inférieure à cette valeur limite (absorption, transmission acoustique, atténuation atmosphérique), alors la particule disparaît du milieu de propagation (*cf.* paragraphe 7.2.2, attribut `trans_epsilon` de l'élément `<simulation/>`);
- temps écoulé sur le pas de temps en cours.

## 5.3 Déroulement du calcul

### 5.3.1 Calcul parallèle sur les bandes de fréquence

Après lecture du fichier de configuration XML, et une fois les données d'entrée réparties sur les différentes variables, le calcul de la propagation acoustique des particules sonores est dans un premier temps distribué sur plusieurs tâches en parallèle, chaque tâche correspondant à une bande de fréquence (*cf.* organigramme de la figure 5.1). Le nombre de tâches en parallèle est fonction du nombre de microprocesseurs disponibles sur la station de calcul utilisée. Une fois toutes les tâches terminées, les données sont regroupées, puis sauvegardées dans les différents fichiers de sortie (*cf.* section 7.5).

### 5.3.2 Boucle sur les sources sonores

Pour chaque bande de fréquence, les calculs sont réalisés en bouclant sur les sources sonores (*cf.* organigramme de la figure 5.2).

### 5.3.3 Boucle sur les particules sonores

Pour chaque source sonore, le calcul est réalisé sur toute la durée de vie d'une particule (*i.e.* sur la durée de la propagation, ou tant que la particule possède une énergie suffisante, ou tant que la particule ne disparaît pas du domaine de propagation) avant de passer à une nouvelle particule (*cf.* organigramme de la figure 5.3).

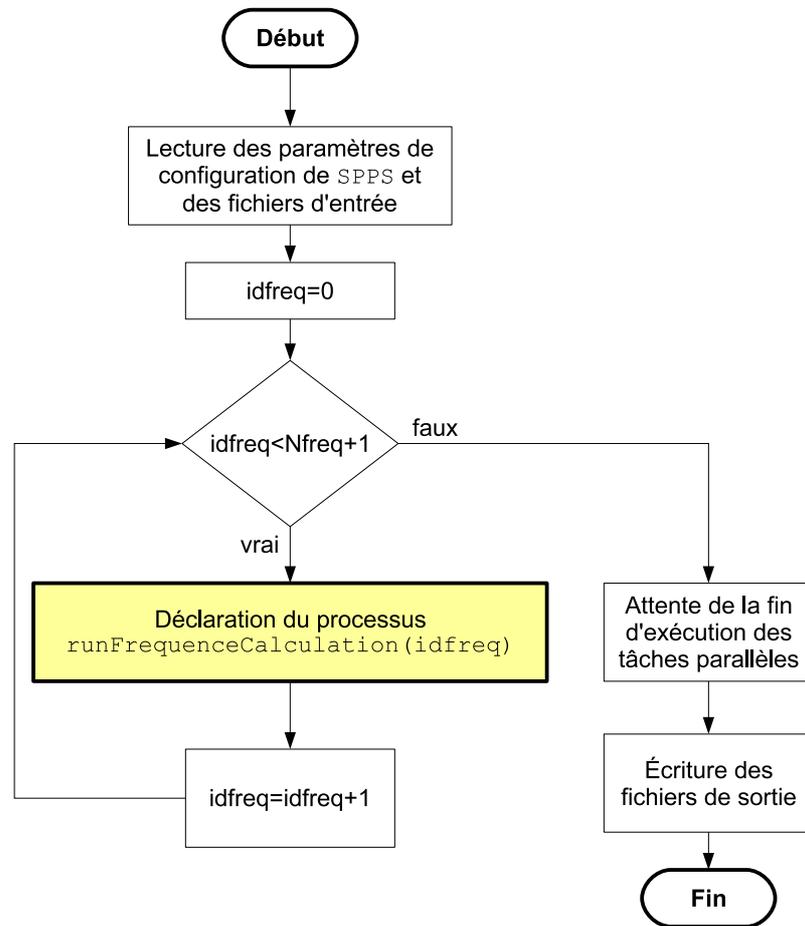
### 5.3.4 Suivi d'une particule sonore

#### 5.3.4.1 Trajectoire d'une particule

Chaque particule sonore est donc suivie individuellement jusqu'à la fin de vie. La propagation acoustique se fait pas de temps après pas de temps (*cf.* organigrammes des figures 5.4 et 5.6). A chaque nouveau pas de temps (ou après collision sur un le pas de temps en cours), la méthode consiste à déterminer les distances de collision jusqu'à la prochaine paroi

du modèle ou jusqu'à la prochaine face d'un encombrement (test de collision). La propagation acoustique se fait ensuite maille après maille en respectant la procédure « type » suivante (figure 5.5) :

1. La particule sonore étant dans la maille courante, un test de collision est effectué pour détecter la face de la maille sur laquelle la particule sonore va entrer en collision, compte tenu de sa direction de propagation (figure 5.5(a)) ;
2. La particule est ensuite translaturée librement sur ce segment jusqu'au point de collision ainsi calculé. Une fois le segment parcouru l'indice de la maille courante passe à celui de l'indice de la maille voisine liée par la face commune aux deux mailles, chaque face d'une maille étant associée à l'indice de la maille voisine ou à l'indice d'une face de la scène (figure 5.5(b)) ;
3. Si nécessaire, un nouveau test de collision par rapport aux quatre faces de la nouvelle maille courante est effectué. Un nouveau point de collision est calculé et un nouveau segment de rayon est parcouru. L'étape est la même que pour l'étape 2 (figures 5.5(c) et 5.5(d)) ;
4. Si la face de la maille de collision correspond à une face du modèle, le traitement de la réflexion est réalisé conformément aux propriétés acoustiques (absorption, loi de réflexion, diffusion, transmission) de la paroi en question (figure 5.5(e)). Dans ce cas, la maille courante de la particule reste inchangée, et un nouveau test de collision sur les faces de la maille est réalisé (figure 5.5(f)) ;
5. Si la face de la maille correspond à la face d'un encombrement, l'état de la particule change. La propagation acoustique de la particule intègre alors la procédure de collision et d'absorption correspondant à l'encombrement en question (calcul de la distance de collision avec le prochain objet de l'encombrement, réalisation de la collision/absorption). L'état de la particule reste inchangé tant que la particule reste dans l'encombrement en question. Si une des faces de l'encombrement correspond également à une face de modèle, le traitement de la collision sur la paroi est effectué, sans changement de l'état de la particule (*i.e.* la particule reste dans l'encombrement) ;
6. Cette procédure est répétée tant que le temps de propagation (somme des temps élémentaires de propagation sur chaque segment) est inférieure à la durée d'un pas de temps. Suivant les cas et



**Figure 5.1** — Organigramme du code : boucle sur les  $N_{freq}$  bandes de fréquence. Chaque calcul pour une bande de fréquence donne lieu à une tâche indépendante. En fonction du nombre de processeurs disponibles sur la station de calcul utilisée, il peut y avoir plusieurs tâches en parallèle. Le fonctionnement du processus `runFrequencyCalculation(idfreq)` est présenté à l'organigramme 5.2.

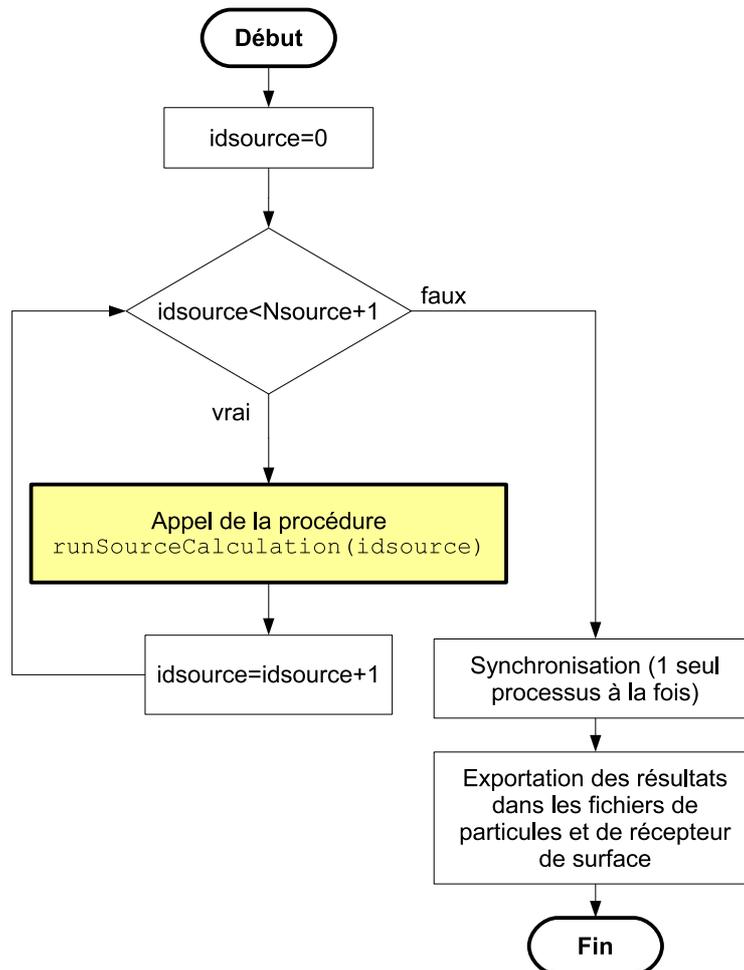
les options de calcul, cette procédure peut également être stoppée si l'énergie de la particule est devenue insuffisante ou si la particule a disparu du domaine de propagation (voir note suivante). Lorsque le dernier segment de collision détecté est supérieur à la distance pouvant être parcourue sur le temps restant (toujours dans la pas de temps courant), la particule est translaturée uniquement de la distance pouvant être parcourue.

 La disparition de la particule du domaine de propagation peut être provoquée par plusieurs causes. En mode de calcul « aléatoire », la particule peut être totalement absorbée par une paroi, par un objet de l'encombrement ou en raison de l'absorption atmosphérique. Une autre cause résulte de problèmes liés à la précision numérique. En particulier, si la position de la particule correspond au sommet d'une maille

(ou d'une face du modèle) –à la précision du calcul près–, la boucle correspondant au test de collision sur les faces de la maille courante n'a pas de sorties puisque plusieurs faces sont concernées par la même collision. Dans ce cas, il est nécessaire de supprimer la particule pour éviter un bouclage infini. Cette procédure est toutefois très rare. Le fichier d'information du calcul, d'extension `.gabe`, précise le nombre de particules qui ont supprimées en raison de « boucles infinies ».

### 5.3.4.2 Traitement des collisions

Le traitement des collisions sur les parois du domaine est décrit par l'organigramme de la figure 5.7. Il reprend le principe de la modélisation des parois présenté au paragraphe 4.3.



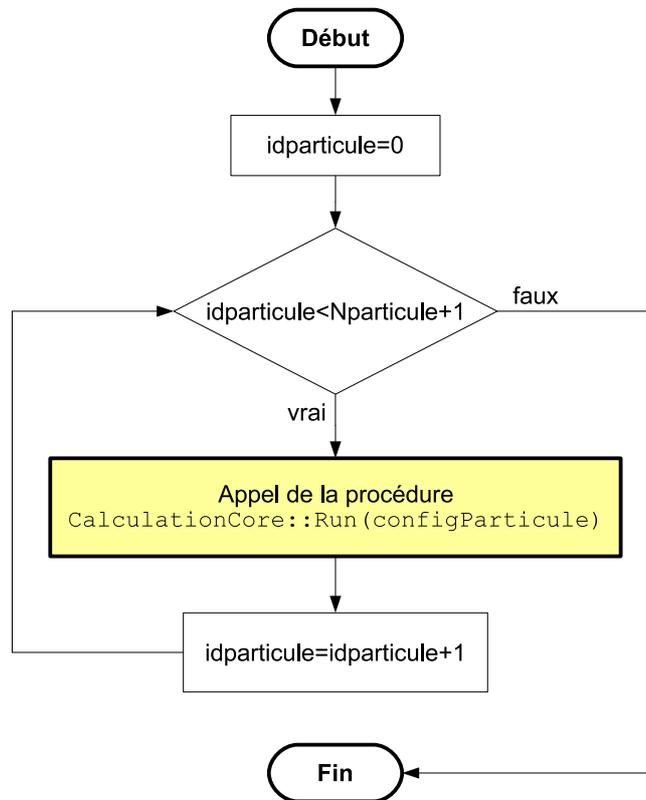
**Figure 5.2** — Organigramme du code : boucle sur les  $N_{source}$  sources sonores actives. Le fonctionnement du processus `runSourceCalculation(idsource)` est présenté à l'organigramme 5.3.

En mode de calcul « aléatoire », les phénomènes aux parois sont modélisés par des tirages aléatoires, pouvant se traduire par la disparition de la particules sonores. Si la particule ne disparaît pas du domaine, elle continue à se propager avec la même énergie (rappel : en mode « aléatoire » l'énergie d'une particule reste constante tout au long de sa durée de vie).

En mode de calcul « énergétique », les phénomènes physiques d'absorption et de transmission acoustique donnent lieu à une pondération sur l'énergie de la particule. La prise en compte de la transmission peut amener à dupliquer la particule sonore : la particule originale est d'abord réfléchiée par la paroi avec une pondération de l'énergie, la direction de réflexion étant fonction des autres propriétés de la paroi (diffusion, loi de réflexion), tandis qu'une seconde particule est créée puis transmise

avec une pondération de l'énergie par le coefficient de transmission. Dans l'organigramme 5.7, cela se traduit par un nouvel appel de la procédure `CalculationCore::Run(configurationPtransmise)` avec la particule transmise<sup>2</sup> en argument. De ce fait, le nombre de particules sonores dans le domaine peut augmenter de manière considérable, ce qui augmente également la durée du calcul. Pour limiter le nombre de particules se propageant dans le domaine (à chaque modification de l'énergie portée par une particule, pas uniquement à la transmission), un test est réalisé sur l'énergie portée la particule. Si cette énergie est inférieure à une limite fixée par l'utilisateur (paramètre `trans_epsilon` du fichier de configuration XML, correspondant (sur le principe, mais pas exactement) à la variable

<sup>2</sup>La particule transmise hérite des mêmes propriétés que la particule originale `configurationP`, mais avec une mise à jour de la maille courante et de l'énergie.



**Figure 5.3** — Organigramme du code : boucle sur les  $N_{particule}$  particules sonores. Le fonctionnement du processus `CalculationCore::Run(configParticule)` est présenté à l'organigramme 5.4.

ConfigurationP.energie\_epsilon de l'algorithme 5.7), la particule est supprimée du domaine de propagation.



- la prise en compte de la transmission acoustique peut être désactivée (suivant l'attribut `trans_calc` du fichier de configuration) ;
- suivant la valeur du paramètre `direct_calc` du fichier de configuration (calcul du champ direct uniquement), la particule sonore peut être automatiquement détruite lors de la première collision. Cela permet de ne considérer que le champ direct au niveau des récepteurs ponctuels et surfaciques, puisque les réflexions sont supprimées.



La prise en compte des propriétés d'un matériau sur un élément de surface peut dépendre d'incidence de la particule sur une surface, par rapport à la direction de la normale de la surface. Cette option dépend de la valeur du paramètre `side_material` de l'élément `<type_surface/>` du

fichier de configuration (cf. paragraphe 7.2.6). Par défaut (`side_material="1"`, i.e. mode « bilateral »), les mêmes propriétés sont considérées de part et d'autre de la surface (i.e. indépendamment du sens de la normale). Dans ce cas, il est possible de créer des cloisons composées uniquement d'une surface, avec des propriétés identiques de part et d'autre. Si `side_material="0"`, (i.e. mode « unilateral ») les propriétés du matériau sont considérées uniquement lorsque la particule se propage dans la direction opposée à la normale de la surface. Cette dernière option permet de créer des cloisons avec épaisseur (i.e. avec deux éléments de surface en vis-à-vis, et d'associer à chaque surface ses propres matériaux, chaque surface étant transparente pour une direction de propagation dans le sens de la normale à la surface). Ce paramètre `<type_surface/>` est normalement utilisé pour modéliser la transmission acoustique.

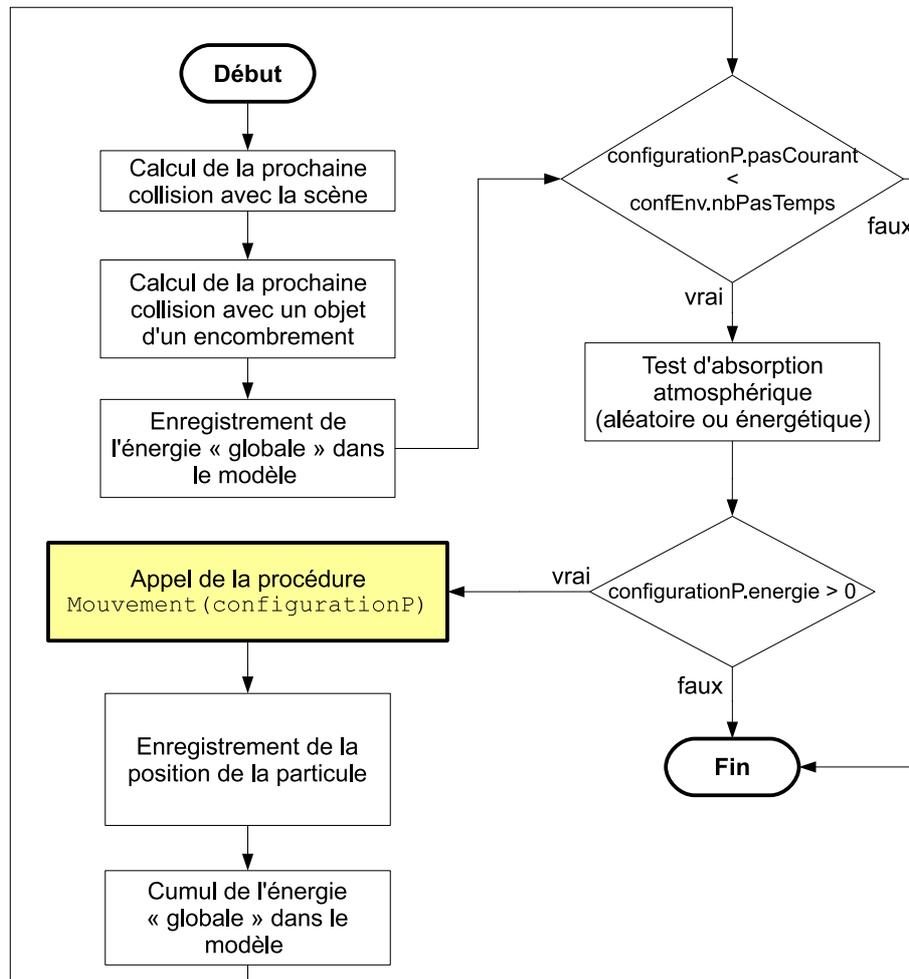


Figure 5.4 — Organigramme du code : suivi d'une particule sonore au cours du temps (*confEnv.nbPasTemps* pas de temps). Le fonctionnement du processus *Mouvement(configurationP)* est présenté à l'organigramme 5.6.

### 5.3.5 Calcul des indicateurs acoustiques

#### 5.3.5.1 Récepteurs ponctuels

A chaque déplacement d'une particule sonore sur un segment de trajectoire, un test est réalisé pour déterminer si cette particule va traverser le volume sphérique d'un récepteur ponctuel. Si c'est effectivement le cas, l'énergie<sup>3</sup> de particule (au moment du déplacement), est pondérée par la distance parcourue dans le volume récepteur (cf. paragraphe 4.4.1), et éventuellement par l'angle entre la direction incidente de la particule et l'orientation du récepteur (cf. paragraphe 4.4.3), puis cumulée à l'énergie (latérale) totale du récepteur, sur le pas de temps considéré.

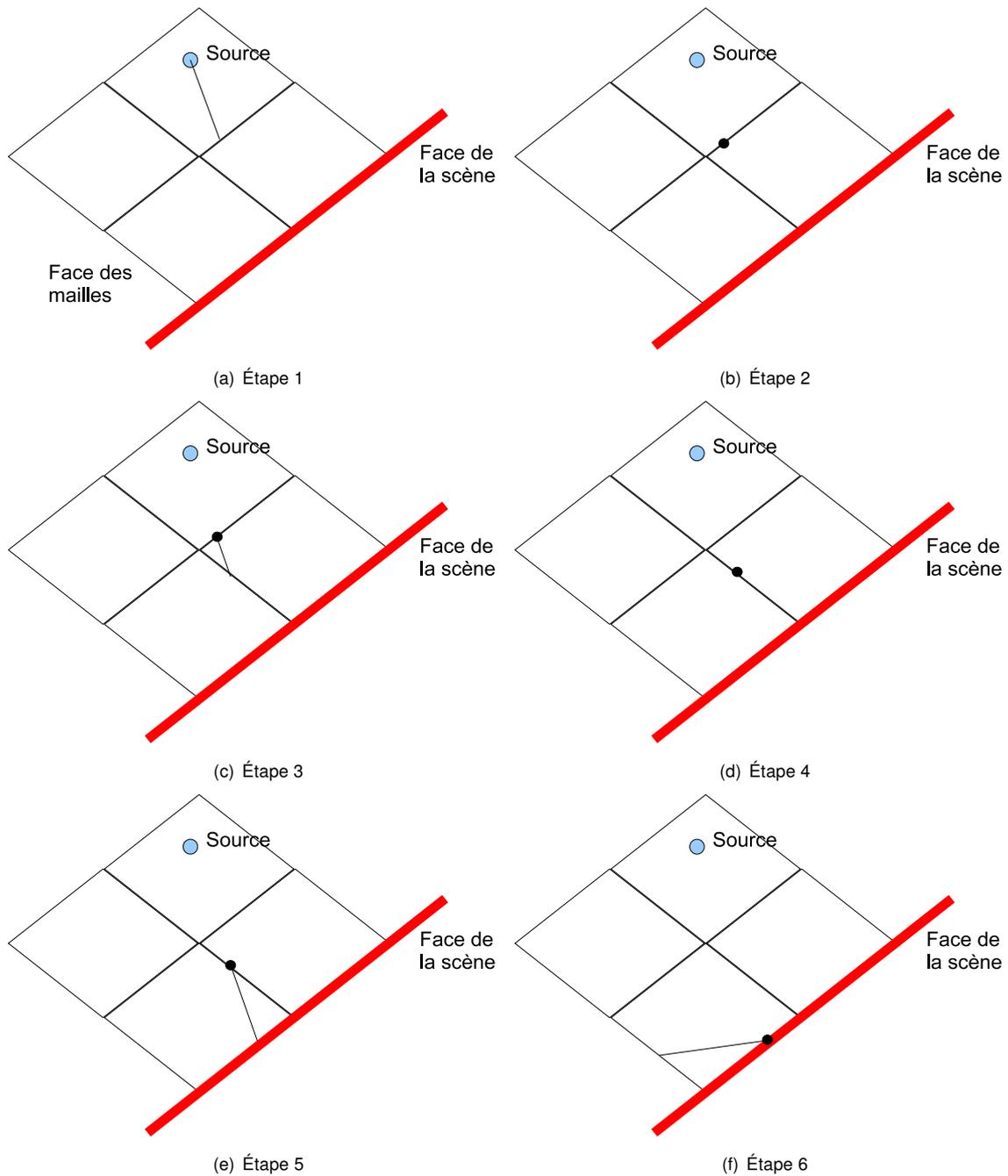
<sup>3</sup>il ne s'agit pas exactement de l'énergie de la particule à ce stade du calcul.

#### 5.3.5.2 Récepteurs surfaciques

A collision d'une particule sur une face du domaine, un test est réalisé pour déterminer si la face correspondante correspond également à une face d'un récepteur surfacique. Si c'est effectivement le cas, l'énergie de la particule est pondérée par le cosinus de l'angle d'incidence de la particule par rapport à la normale (cf. paragraphe 4.4.4), puis cumulée à l'énergie totale du récepteur surfacique, sur le pas de temps considéré.

#### 5.3.5.3 Énergie globale dans le modèle

A la fin de chaque pas de temps du calcul, l'énergie de chaque particule sonore vient se cumuler à la valeur de l'énergie globale dans le modèle (cf. organigramme de la figure 5.4).



**Figure 5.5** — Principe du suivi d'une particule sonore dans le domaine, sur un même pas de temps. Exemple d'une trajectoire avec collision sur une paroi.

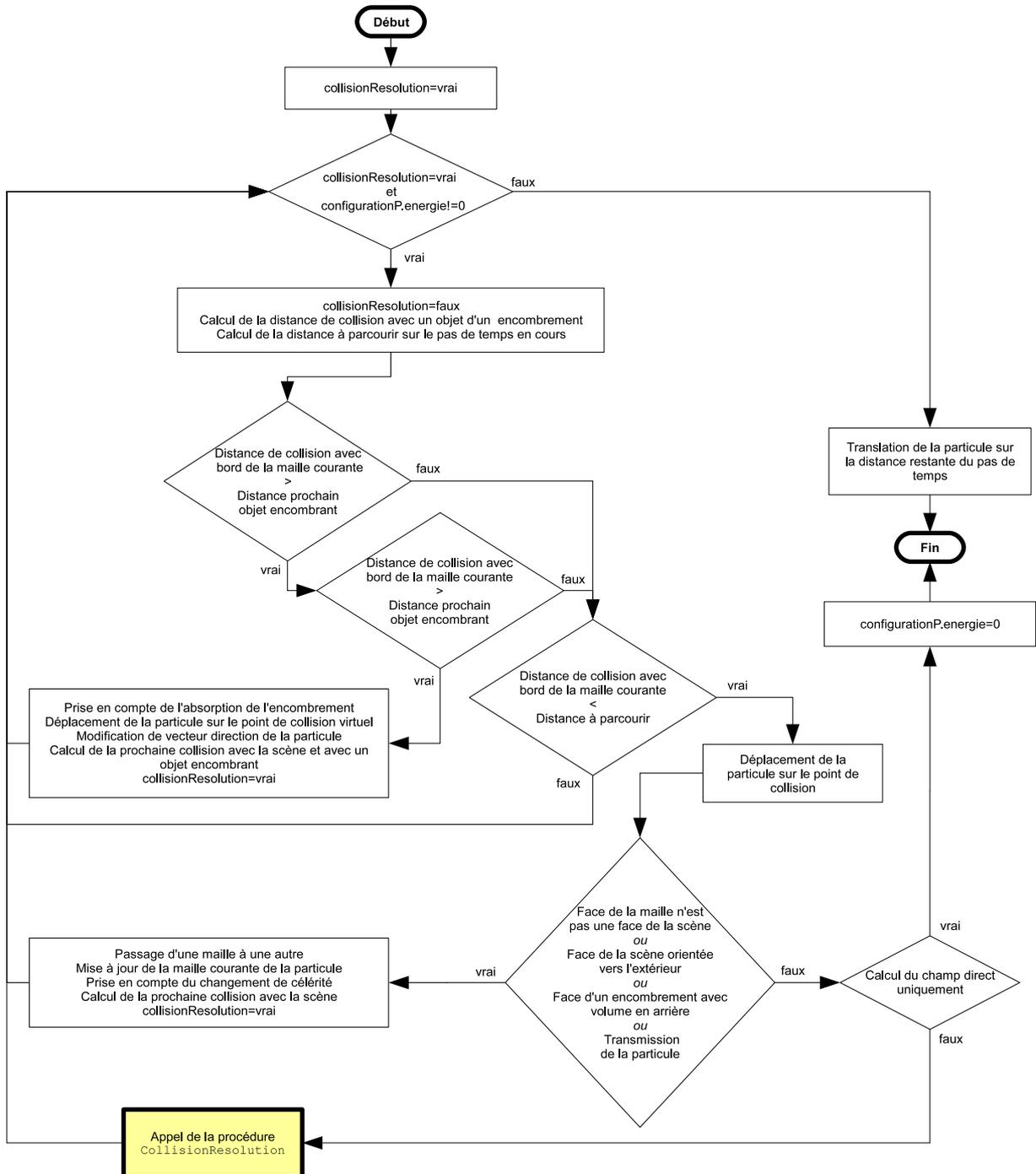


Figure 5.6 — Organigramme du code : trajectoire d'une particule sur un pas de temps. Le fonctionnement du processus *CollisionResolution* est présenté à l'organigramme 5.7.



## Exécution du code

### 6.1 Exécution avec l'interface I-Simpa

Il est recommandé d'utiliser le code SPPS *via* l'interface I-Simpa (l'interface I-Simpa est développée par l'Ifsttar et diffusée sous licence propriétaire). Dans ces conditions, l'interface gère directement les entrées (fichiers de configuration, de modèle géométrique et de maillage), de l'exécution et des sorties (post-traitement et affichage des résultats) du code SPPS, sans avoir à exécuter de lignes de commande (*cf.* Manuel d'utilisation de l'interface I-Simpa).

Les paramètres associés au code de calcul SPPS dans l'interface I-Simpa sont les suivants :

- **Choix des bandes tiers d'octave** : liste des bandes de tiers d'octave à cocher/décocher. Un menu contextuel sur cet élément (**Sélection automatique**) permet de faire des (dé)sélections automatiques de plusieurs bandes simultanément :
  - **Tout désélectionner** : désélectionne toutes les bandes de fréquence ;
  - **Tiers d'octave centrées sur les octaves** :
    - \* **Toutes les bandes [63-16000] Hz** ;
    - \* **Bâtiment/Routier [125-4000] Hz** : bandes d'octave standards pour la prévision acoustique en acoustique routière et en acoustique du bâtiment ;
  - **Tiers d'octave** :
    - \* **Toutes les bandes [50-20000] Hz** ;
    - \* **Bâtiment/Routier [100-5000] Hz** : bandes de tiers d'octave standards

pour la prévision acoustique en acoustique routière et en acoustique du bâtiment.

- **Maillage** : ces paramètres sont associés au mailleur tétraédrique TetGen installé dans l'interface I-Simpa. Se reporter au manuel de TetGen pour plus d'informations concernant les paramètres de maillage (*cf.* paragraphe ??) :
  - **Contrainte d'aire des récepteurs surfaciques** : case à cocher/décocher pour imposer une contrainte sur la valeur de l'aire des récepteurs surfaciques ;
  - **Contrainte d'aire des récepteurs surfaciques : valeur en m<sup>2</sup>** : valeur de la contrainte (nombre réel) sur l'aire des récepteurs surfaciques. Les surfaces sont maillées de telle manière que l'aire d'un élément de surface soit inférieure à cette valeur ;
  - **Contrainte de volume** : case à cocher/décocher pour imposer une contrainte sur la valeur du volume des mailles ;
  - **Contrainte de volume : valeur en m<sup>3</sup>** : valeur de la contrainte (nombre réel) sur le volume des mailles. Les volumes sont maillés de telle manière que le volume d'une maille soit si possible inférieur à cette valeur ;
  - **Contrainte de dimension : rapport rayon/arête** : contrainte sur le rapport entre le rayon englobant une maille et la longueur d'une arête. Cette option permet de créer des mailles de taille homogène, si nécessaire, et d'éviter les mailles allongées ;
  - **Correction de la scène 3D avant maillage** : voir le paragraphe ?? ;

- **Paramètres supplémentaires** : paramètres supplémentaires de maillage (en complément des paramètres déjà renseignés) ;
- **Paramètres utilisateurs** : paramètres de maillage (en remplacement des paramètres déjà renseignés).
- **Propriétés** : définit les paramètres de calcul du code :
  - **Calcul de l'absorption atmosphérique** : case à cocher/décocher pour activer/désactiver la prise en compte de l'absorption atmosphérique dans le calcul (définit le paramètre `disable_absatmo_computation` du fichier de configuration du code SPPS) ;
  - **Calcul de la diffusion par l'encombrement** : case à cocher/décocher pour activer/désactiver la prise en compte des encombrements dans le calcul (définit le paramètre `enc_calc` du fichier de configuration du code SPPS) ;
  - **Calcul de la transmission acoustique** : case à cocher/décocher pour activer/désactiver la prise en compte de la transmission acoustique dans le calcul (définit le paramètre `trans_calc` du fichier de configuration du code SPPS) ;
  - **Calcul du champ direct uniquement** : si la case est cochée, seul le champ direct est calculé (définit le paramètre `direct_calc` du fichier de configuration du code SPPS) ;
  - **Durée de la simulation (s)** : durée en seconde (valeur réelle) de la simulation (définit le paramètre `duree_simulation` du fichier de configuration du code SPPS) ;
  - **Limite d'extinction des particules (en 10 r)** : valeur réelle  $r$  définissant la limite de l'énergie des particules sonores. Les particules dont l'énergie a décru d'un facteur supérieur à  $10^r$  ne sont plus considérées dans les calculs (définit le paramètre `trans_epsilon` du fichier de configuration du code SPPS) ;
  - **Méthode de calcul** : liste déroulante pour le choix de la méthode de calcul (définit le paramètre `computation_method` du fichier de configuration du code SPPS) :
    - \* **Aléatoire** : méthode de calcul rapide mais moins précise ;
    - \* **Énergétique** : méthode de calcul précise mais lente ;
- **Nombre de particules par source** : nombre entier définissant le nombre de particules sonores émises par source sonore (définit le paramètre `nbparticules` du fichier de configuration du code SPPS) ;
- **Nombre de particules par source (animation)** : nombre entier définissant le nombre de particules sonores émises par source sonore, dont l'historique des trajets est stocké pour les besoins de représentations animées (définit le paramètre `nbparticules_rendu` du fichier de configuration du code SPPS) ;
- **Pas de temps (s)** : pas de temps en seconde (valeur réelle) de la simulation (définit le paramètre `Pasdetemps` du fichier de configuration du code SPPS) ;
- **Rayon des récepteurs ponctuels (m)** : rayon des récepteurs ponctuels (valeur réelle) (définit le paramètre `rayon_recepteurp` du fichier de configuration du code SPPS) ;
- **Récepteur surfacique : export par bande de fréquence** : case à cocher/décocher pour conserver (ou non) l'ensemble des valeurs calculées au niveau des récepteurs surfaciques, par bande de fréquence (définit le paramètre `output_recs_byfreq` du fichier de configuration du code SPPS). Si décochée, seules les valeurs globales sont conservées ;
- **Récepteur surfacique : type de calcul** : liste déroulante permettant de définir quelles quantités doivent être conservées pour les récepteurs surfaciques (définit le paramètre `surf_receiv_method` du fichier de configuration du code SPPS) :
  - \* **Cartographie intensité** : cartographie en niveau d'intensité sonore (adaptée pour une cartographie sur une surface de type scène) ;
  - \* **Cartographie SPL** : cartographie en niveau de pression acoustique (adaptée pour une cartographie sur une surface de type plan de coupe dans le volume de propagation) ;

## 6.2 Exécution en ligne de commande

De manière beaucoup moins convivial, il est également possible d'utiliser le code SPPS directement

en ligne de commande, en exécutant la commande suivante dans une fenêtre de commandes, le fichier `config.xml` étant le fichier de configuration au format XML :

```
spps.exe config.xml
```

En plus du fichier `config.xml` contenant toutes les informations du calcul (*cf.* section 7.2), l'utilisation du code SPPS en ligne nécessite la présence de plusieurs fichiers d'entrée :

- un fichier de modèle géométrique au format binaire CBIN : `model_geo.cbin` ;
- un fichier de maillage associé au modèle géométrique, également au format binaire CBIN : `model_mesh.cbin`.

Dans ces conditions, il est impératif que ces deux fichiers soient conformes aux spécifications du code SPPS (*cf.* sections 7.3 et 7.4). Par ailleurs, les fichiers de sortie du code devront être traités directement par l'utilisateur.

## 6.3 Exécution en langage Python™

La création des fichiers d'entrée et le traitement des fichiers de sortie peuvent toutefois être facilités par l'utilisation de la librairie `LibSimpa` (© Ifsttar) dans le langage Python™ (de la version 2.5 à la version 3 exclue<sup>1</sup>). Le chapitre 9 présente un exemple d'application complet (création des fichiers d'entrée, exécution du code, lecteur et conversion des fichiers de sortie).

## 6.4 Exécution en C++

La création des fichiers d'entrée et le traitement des fichiers de sortie peuvent aussi être réalisé en C++ grâce la librairie `LibInterface` (© Ifsttar).

---

<sup>1</sup>[www.python.org](http://www.python.org).



# Format des fichiers d'entrée-sortie

## 7.1 Liste et format des fichiers d'entrée/sortie

Les extensions et la nature des fichiers d'entrée/sortie utilisés par le code de calcul SPPS sont présentées dans le tableau 7.1 ci-dessous et détaillées dans les sections suivantes.

## 7.2 Contenu et format du fichier de configuration

Le fichier XML de configuration du calcul (fichier `config.xml`) contient les informations nécessaires à l'exécution d'un calcul. Toutes les données sont de type `string`. Le fichier doit être créé conformément au standard *Extensible Markup Language (XML) 1.0*. L'organisation du fichier de configuration est au présenté au listing 7.1).

```

configuration
├── paramètres de simulation
│   └── liste des fréquences de simulation
├── définition des conditions atmosphériques
├── définition des matériaux utilisés
├── définition des sources sonores
├── définition des récepteurs ponctuels
├── définition des récepteurs surfaciques
└── définition des encombrements
  
```

Figure 7.1 — Contenu du fichier de configuration.

### 7.2.1 Élément <configuration/>

L'élément <configuration/> est la racine du document XML (listing 7.2) :

- `workingdirectory` : chemin d'accès au dossier existant comportant les fichiers de modèle et de maillage ;

### 7.2.2 Élément <simulation/>

L'élément <simulation/> est propre au code de calcul utilisé, en l'occurrence SPPS dans notre cas, et contient les paramètres du calcul (listing 7.3), à travers les attributs suivants :

- `recepteurss_directory` : nom du dossier à utiliser pour stocker les fichiers de résultat des récepteurs de surface ;
- `recepteurss_filename` : nom du fichier (avec extension `.csbin`) à utiliser pour stocker les résultats d'un récepteur de surface ;
- `recepteurss_filename` : nom du fichier (avec extension `.csbin`) à utiliser pour stocker les résultats d'un plan de coupe ;
- `particules_directory` : nom du dossier à utiliser pour stocker les fichiers de résultats des animations des particules ;
- `particules_filename` : nom du fichier (avec extension `.pbin`) à utiliser pour stocker les résultats des animations des particules ;
- `receiversp_directory` : nom du dossier à utiliser pour stocker les résultats pour les récepteurs ponctuels ;
- `receiversp_filename` : nom du fichier (avec extension `.recp`) à utiliser pour stocker les résultats d'un récepteur ponctuel (pour le calcul du niveau de pression acoustique) ;

**Table 7.1** — Extension et nature des fichiers d'entrée/sortie du code de calcul.

Extension	Type	Entrée/Sortie	Description	Détail
.xml	XML	Entrée	Fichier de configuration du calcul	Paragraphe 7.2
.cbin	Binaire	Entrée	Modèle géométrique (faces et sommets)	Paragraphe 7.3
.mbin	Binaire	Entrée	Maillage associé au modèle (mailles tétraédriques)	Paragraphe 7.4
.recp	Binaire	Sortie	Données associées au calcul de quantités énergétiques au niveau d'un récepteur ponctuel	Paragraphe 7.5.2.1
.gap	Binaire	Sortie	Données associées au calcul de paramètres « avancés » (niveau sonore latéral) au niveau d'un récepteur ponctuel	Paragraphe 7.5.2.2
.gabe	Binaire	Sortie	Données associées au calcul du vecteur « intensité » au niveau d'un récepteur ponctuel	Paragraphe 7.5.2.3
.csbin	Binaire	Sortie	Données associées au calcul de l'intensité sonore au niveau d'un récepteur surfacique	Paragraphe 7.5.3.1
.recp	Binaire	Sortie	Données associées au calcul de la quantité énergétique cumulée pour l'ensemble du modèle	Paragraphe 7.5.4.1
.pbin	Binaire	Sortie	Trajectoire des particules sonores au cours du temps	Paragraphe 7.5.4.2
.rpi	Binaire	Sortie	Données associées au calcul du vecteur « intensité » au niveau de tous les récepteurs ponctuels	Paragraphe 7.5.4.3
.gabe	Binaire	Sortie	Fichier des statistiques du calcul	Paragraphe 7.5.4.4

**Listing 7.1** — Descriptif général du fichier de configuration

```
<?xml version="1.0" encoding="iso-8859-1"?>
<configuration>
  <simulation/>
  <condition_atmosphérique/>
  <surface_absorption_enum/>
  <sources/>
  <recepteurss/>
  <recepteursp/>
  <encombrement_enum/>
</configuration>
```

**Listing 7.2** — Descriptif de l'élément <configuration/>

```
<configuration
  workingdirectory="folder_data\">
```

- `receiversp_filename_adv` : nom du fichier (avec extension `.gap`) à utiliser pour stocker les résultats d'un récepteur ponctuel (pour le calcul du niveau de pression acoustique latéral);
- `cumul_filename` : nom du fichier (avec extension `.recp`) à utiliser pour stocker les résultats du cumul de la pression acoustique dans le domaine;
- `tetrameshFileName` : nom du fichier (avec extension `.mbin`) contenant le maillage de la scène;
- `modelName` : nom du fichier (avec extension `.cbin`) contenant le modèle de la scène;
- `intensity_folder` : nom du dossier contenant (pour chaque bande de fréquence, un dossier par bande) les fichiers `intensity_filename` des vecteurs « intensité » pour tous les récepteurs volumiques simultanément. Ce dossier peut être nécessaire pour construire une animation graphique des vecteurs « intensité »;
- `intensity_filename` : nom du fichier (avec extension `.rpi`) contenant les vecteurs « intensité » pour tous les récepteurs volumiques simultanément;
- `intensity_rp_filename` : nom du fichier (avec extension `.gabe`) contenant le vecteur « intensité » pour chaque récepteur ponctuel;
- `stats_filename` : nom du fichier (avec extension `.gabe`) contenant les informations sur le calcul;
- `abs_atmo_calc` : calcul de l'absorption atmosphérique (entier égal à 0 ou 1) :
  - si `abs_atmo_calc="1"` le calcul prend en compte l'absorption atmosphérique;
  - si `abs_atmo_calc="0"` le calcul ne prend pas en compte l'absorption atmosphérique;
- `enc_calc` : calcul de la diffusion par des encombrements (entier égal à 0 ou 1) :
  - si `enc_calc="1"` le calcul prend en compte la diffusion par les encombrements (si définis);
  - si `enc_calc="0"` le calcul ne prend pas en compte la diffusion par les encombrements;
- `direct_cal` : calcul du champ direct ou du champ total (entier égal à 0 ou 1) :
  - si `direct_cal="1"` seul le champ direct est calculé<sup>1</sup> au niveau des récepteurs ponctuels et surfaciques;
  - si `direct_cal="0"` le champ total (champ direct et autres contributions) est calculé;
- `duree_simulation` : durée de la simulation en seconde (valeur réelle positive en s);
- `output_recs_byfreq` : choix de l'exportation ou non des cartographies (au niveau des récepteurs surfaciques) par bande de fréquence (valeur égale à 0 ou 1) :
  - si `output_recs_byfreq="1"` les cartographies sont sauvegardées pour chaque bande de fréquence, en plus de la cartographie en niveau global;
  - si `output_recs_byfreq="0"` seule la cartographie en niveau global est sauvegardée;
- `surf_receiv_method` : type de calcul pour un récepteur surfaciques :
  - si `surf_receiv_method="0"` : calcul de l'intensité sonore;
  - si `surf_receiv_method="1"` : calcul de la pression acoustique.
- `computation_method` : choix de la méthode de calcul (entier égal à 0 ou 1)
  - si `computation_method="1"` le calcul est réalisé en mode aléatoire;
  - si `computation_method="0"` le calcul est réalisé en mode énergétique;
- `Nbparticules` : Nombre de particules émises par source sonore (nombre entier positif);
- `nbparticules_rendu` : Nombre de particules (nombre entier positif ou nul), par source sonore, considéré pour la représentation de l'animation des particules (`nbparticules_rendu ≤ Nbparticules`);
- `Pasdetemps` : Pas de temps de propagation en seconde (valeur réelle en s);
- `trans_epsilon` : paramètre pour le calcul de la valeur limite de l'énergie d'une particule. Si l'énergie de la particule est inférieure à  $10^{-r}$  de son énergie initiale, avec `trans_epsilon=r`,

<sup>1</sup> Les particules sont détruites à la première collision sur une paroi du domaine.

alors particule disparaît du domaine de propagation (*i.e.* l'énergie de la particule est négligeable). Ce paramètre est actif uniquement pour si le calcul est réalisé en mode énergétique (*i.e.* `computation_method="0"`);

- `rayon_recepteurp` : Rayon des sphères des récepteurs ponctuels en mètre (valeur réelle positive en m).



- si `nbparticules_rendu="0"`, aucune particules ne sera conservée pour la représentation de l'animation;
- le nombre de particules `nbparticules_rendu` pour le rendu ne peut pas être supérieur au nombre total de particules `Nbparticules`;
- suivant les valeurs pour `abs_atmo_calc`, `enc_calc`, `direct_cal`, certaines informations données dans le fichier de configuration ne seront pas prises en comptes dans le calcul.

### 7.2.3 Élément `<freq_enum/>`

L'élément `<freq_enum/>` liste les bandes de fréquence utilisées (ou non) pour le calcul. Pour les récepteurs surfaciques, le niveau sonore global est obtenu en considérant uniquement les bandes de fréquence retenues. La liste des fréquences est définie de la manière suivante (listing 7.4) :

- `freq` : Centre de la bande de tiers d'octave ; fréquence en Hertz (entier en Hz) ;
- `docalc` : Indique au code de calcul si ce tiers d'octave doit être calculé :
  - si `docalc="1"` le calcul est réalisé pour la bande de fréquence considérée ;
  - si `docalc="0"` le calcul n'est pas réalisé pour cette bande de fréquence ;



*Remarques :*

- l'exemple donné ici n'est pas exhaustif. Le nombre et la fréquence médiane des bandes sont définies par l'utilisateur.
- il faut s'assurer que la définition des bandes est homogène pour l'ensemble de la simulation (paramètres du calcul, émission acoustique, propriétés des parois) ;

- il est également possible de ne lister que les bandes utilisées pour le calcul, plutôt que de lister des bandes dont certaines ne seront pas utilisées (`docalc="0"`).

### 7.2.4 Élément

`<condition_atmosphérique/>`

L'élément `<condition_atmosphérique/>` précise les conditions atmosphériques du calcul (listing 7.5) à travers les attributs suivants :

- `absatmo` : Valeur imposée du coefficient d'absorption atmosphérique (valeur réelle positive en Np/m). Cette valeur est utilisée uniquement si le paramètre `disable_absatmo_computation` est fixé à 1 ;
- `disable_absatmo_computation` : choix du mode de la prise en compte de l'absorption atmosphérique :
  - si `disable_absatmo_computation="1"` la valeur définie par le paramètre `absatmo` est imposée pour toutes les bandes de fréquence ;
  - si `disable_absatmo_computation="0"` le coefficient d'absorption atmosphérique est calculé pour chaque bande de fréquence, en fonction de la valeur du centre de la bande, et des conditions atmosphériques ;
- `alog` : Paramètre  $a_{\log}$  du profil de célérité de type « log-lin » (valeur réelle en m/s) ;
- `blin` : Paramètre  $b_{\ln}$  du profil de célérité de type « log-lin » (valeur réelle en m/s) ;
- `humidite` : Humidité relative de l'air en % (valeur réelle comprise entre 0 et 100) ;
- `pression` : Pression atmosphérique de l'air en Pascal (valeur réelle positive en Pa) ;
- `z0` : Paramètre de hauteur de rugosité (en mètre) du profil de célérité de type « log-lin » (valeur réelle positive en m) ;
- `temperature` : Température de l'air ambiant en °C (valeur réelle en °C).

### 7.2.5 Élément

`<surface_absorption_enum/>`

L'élément `<surface_absorption_enum/>` permet de lister les matériaux utilisés dans le modèle géométrique et de détailler leur propriétés acoustiques (absorption, transmission, diffusion et loi de réflexion diffuse) par bande de fréquence (listing 7.6).

- `loi="0"` : loi de réflexion spéculaire ;
- `loi="1"` : loi de réflexion uniforme ( $w^0$ ) ;
- `loi="2"` : loi de réflexion de Lambert ( $w^1$ ) ;
- `loi="3"` : loi de réflexion  $w^2$  ;
- `loi="4"` : loi de réflexion  $w^3$  ;
- `loi="5"` : loi de réflexion  $w^4$  ;

### 7.2.6 Élément `<type_surface/>`

L'élément `<type_surface/>` désigne un matériau (listing 7.7). Les attributs sont les suivants :

- `id` : identifiant du matériau. L'identifiant 0 correspond au matériau par défaut. On retrouve ces identifiants dans le fichier de maillage du modèle (`model_mesh.cbin` dans l'exemple) pour chaque triangle composant le modèle ;
- `masse_volumique` : valeur de la masse volumique du matériau ( $\text{kg/m}^3$ ) ; ce paramètre n'est pas nécessaire puisque non utilisée dans SPPS ;
- `resistivite` : valeur de la résistivité du matériau ( $\text{kgNs/m}^4$ ) ; ce paramètre n'est pas nécessaire puisque non utilisée dans SPPS ;
- `side_material` : modalité de la prise en compte du matériau par rapport à la normale de la surface ;
  - si `side_material="1"` (i.e. mode « bilatéral »), les propriétés du matériaux sont considérées simultanément dans les deux sens ;
  - si `side_material="0"` (i.e. mode « unilatéral »), les propriétés du matériaux sont considérées uniquement lorsque la particule se propage dans la direction opposée à la normale de la surface.
- `freq` : centre de la bande de tiers d'octave ; fréquence en Hertz (entier en Hz) ;
- `absorb` : coefficient d'absorption du matériau (valeur réelle comprise entre 0 et 1) ;
- `affaiblissement` : affaiblissement acoustique du matériau en décibel (valeur réelle positive en dB). Ce champ ne doit pas être défini pour les matériaux (ou les bandes) sans transmission ;
- `diffusion` : coefficient de diffusion du matériaux (valeur réelle comprise entre 0 et 1) ;
- `loi` : loi de réflexion diffuse du matériau (entier positif ou null) :

#### Remarques :

- les valeurs d'absorption (*absorb*, notée  $\alpha$ ) et d'affaiblissement (*affaiblissement*, noté  $\gamma$ ) doivent être cohérentes. Si  $\tau$  est le coefficient de transmission associé à  $\gamma$  (i.e.  $\tau = 10^{\gamma/10}$ ), alors il faut s'assurer que  $\tau < \alpha$ . Il ne peut pas en effet y avoir plus d'énergie transmise par la paroi que d'énergie absorbée par la paroi puisque  $\alpha = \beta + \tau$ ,  $\beta$  étant l'énergie dissipée dans la paroi ;
- si l'absorption est totale (i.e. `absorb="1"`), la diffusion acoustique est nulle ;
- si la diffusion est nulle (i.e. `diffusion="0"`), le code de calcul impose automatiquement une loi de réflexion spéculaire sur la paroi en question, quelle que soit la valeur de `loi`.

### 7.2.7 Élément `<sources/>`

L'élément `<sources/>` permet de lister les sources sonores (listing 7.8).

### 7.2.8 Élément `<source/>`

L'élément `<source/>` permet de définir les propriétés géométriques et acoustiques (puissance par bande de fréquence) d'une source sonore (listing 7.9) :

- `id` : indice de la source (nombre entier positif) ;
- `x y z` : coordonnées de la source dans le repère du modèle géométrique (valeurs réelles en m) ;
- `u v w` : orientation de la source dans le repère du modèle géométrique (valeurs réelles en m). Ce vecteur sera normalisé par le code de calcul ;
- `directivite` : directivité de la source (nombre entier positif) :
  - 0 : omnidirectionnel ;
  - 1 : unidirectionnel ;
  - 2 : plan XY ;
  - 3 : plan YZ ;

– 4 : plan XZ ;

- `delay` : retard au déclenchement de la source, en seconde (valeur réelle en s) ;
- `db` : puissance de la source sonore, en décibel (valeur réelle en dB).

### 7.2.9 Élément `<recepteurss/>`

L'élément `<recepteurss/>` permet de lister les récepteurs de surface (listing 7.10) :

### 7.2.10 Élément `<recepteur_surfacique/>`

L'élément `<recepteur_surfacique/>` permet de définir les récepteurs de surface (listing 7.11) :

- `id` : numéro d'identification du récepteur. Ce numéro se retrouve dans le fichier de modèle core-Bin ;
- `name` : libellé du récepteur de surface (optionnel) ;

### 7.2.11 Élément `<recepteursp/>`

L'élément `<recepteursp/>` permet de lister les récepteurs ponctuels (listing 7.12) :

### 7.2.12 Élément `<recepteur_ponctuel/>`

L'élément `<recepteur_ponctuel/>` permet de définir les propriétés des récepteurs ponctuels (listing 7.13) :

- `id` : identifiant du récepteur ponctuel (entier positif) ;
- `lbl` : libellé du récepteur ponctuel (optionnel) ;
- `x y z` coordonnées de la source dans le repère du modèle géométrique (valeurs réelles en m) ;
- `u v w` : orientation de la source dans le repère du modèle géométrique (valeurs réelles en m). Ce vecteur sera normalisé par le code de calcul ;
- `db` : niveau de bruit de fond au niveau du récepteur en décibel (valeur réelle dB).

### 7.2.13 Élément `<encombrement_enum/>`

L'élément `<encombrement_enum/>` liste les zones d'encombrement (listing 7.14).

### 7.2.14 Élément `<encombrement/>`

L'élément `<encombrement/>` définit les propriétés des zones d'encombrement (listing 7.15) :

- `id` : identifiant de l'encombrement ;
- `alpha` : coefficient d'absorption des objets de l'encombrement (valeur réelle comprise entre 0 et 1) ;
- `lambda` : libre parcours moyen dans l'encombrement en mètres (valeur réelle positive en m) ;
- `loi_diff` : loi de diffusion par les objets de l'encombrement :
  - 0 : Omnidirectionnel ;

Listing 7.3 — Descriptif de l'élément &lt;simulation/&gt;

```

<simulation
  recepteurss_directory="folder_surface_receivers\"
  recepteurss_filename="file_surface_receiver.csbin"
  recepteurss_cut_filename="file_surface_receiver_cut.csbin"
  particules_directory="folder_particles\"
  particules_filename="filename_particle.pbin"
  receiversp_directory="folder_receiver\"
  receiversp_filename="file_receiver_SPL.recip"
  receiversp_filename_adv="file_receiver_adv_SPL.gap"
  cumul_filename="file_cumul.recip"
  tetrameshFileName="tetramesh.mbin"
  modelName="mesh.csbin"
  intensity_filename="file_vector_intensity.rpi"
  intensity_folder="folder_intensity_animation"
  intensity_rp_filename="file_receiver_vect_intensity.gabe"
  stats_filename="file_statsSPPS.gabe"
  abs_atmo_calc="1"
  enc_calc="0"
  trans_calc="0"
  direct_calc="0"
  duree_simulation="2,000000"
  output_recs_byfreq="0"
  surf_receiv_method="0"
  trans_epsilon="5,000000"
  computation_method="1"
  nbparticules="50000"
  nbparticules_rendu="0"
  pasdetemps="0,010000"
  rayon_recepteurp="0,310000">
</simulation>

```

Listing 7.4 — Descriptif de l'élément &lt;freq\_enum/&gt;

```

<freq_enum>
  <bfreq freq="20000" docalc="0"/>
  <bfreq freq="16000" docalc="0"/>
  <bfreq freq="12500" docalc="0"/>
  <bfreq freq="10000" docalc="0"/>
  <bfreq freq="8000" docalc="0"/>
  <bfreq freq="6300" docalc="0"/>
  <bfreq freq="5000" docalc="1"/>
  <bfreq freq="4000" docalc="1"/>
  <bfreq freq="3150" docalc="1"/>
  <bfreq freq="2500" docalc="1"/>
  <bfreq freq="2000" docalc="1"/>
  <bfreq freq="1600" docalc="1"/>
  <bfreq freq="1250" docalc="1"/>
  <bfreq freq="1000" docalc="1"/>
  <bfreq freq="800" docalc="1"/>
  <bfreq freq="630" docalc="1"/>
  <bfreq freq="500" docalc="1"/>
  <bfreq freq="400" docalc="1"/>
  <bfreq freq="315" docalc="1"/>
  <bfreq freq="250" docalc="1"/>
  <bfreq freq="200" docalc="1"/>
  <bfreq freq="160" docalc="1"/>
  <bfreq freq="125" docalc="1"/>
  <bfreq freq="100" docalc="1"/>
  <bfreq freq="80" docalc="0"/>
  <bfreq freq="63" docalc="0"/>
  <bfreq freq="50" docalc="0"/>
</freq_enum>

```

Listing 7.5 — Descriptif de l'élément `<condition_atmospherique/>`

```
<condition_atmospherique
  absatmo="0,000000"
  disable_absatmo_computation="0"
  alog="0,000000"
  blin="0,000000"
  humidite="50,000000"
  pression="101325,000000"
  z0="0,020000"
  temperature="20,000000"/>
```

Listing 7.6 — Descriptif de l'élément `<surface_absorption_enum/>`

```
<surface_absorption_enum>
  <type_surface ... />
  <type_surface ... />
  <type_surface ... />
</surface_absorption_enum>
```

Listing 7.7 — Descriptif de l'élément `<type_surface/>`

```
<type_surface id="101" masse_volumique="0,000000" resistivite="100000,000000"
  side_material="1">
  <bfreq freq="20000" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="16000" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="12500" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="10000" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="8000" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="6300" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="5000" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="4000" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="3150" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="2500" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="2000" absorb="0,100000" diffusion="0,000000" affaiblissement="
    23,000000" loi="0" />
  <bfreq freq="1600" absorb="0,200000" diffusion="0,000000" affaiblissement="
    15,000000" loi="0" />
  <bfreq freq="1250" absorb="0,500000" diffusion="0,000000" affaiblissement="
    40,000000" loi="0" />
  <bfreq freq="1000" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="800" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="630" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="500" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="400" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="315" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="250" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="200" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="160" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="125" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="100" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="80" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="63" absorb="0,000000" diffusion="0,000000" loi="0"/>
  <bfreq freq="50" absorb="0,000000" diffusion="0,000000" loi="0"/>
</type_surface>
```

Listing 7.8 — Descriptif de l'élément `<sources/>`

```
<sources>
  <source ... />
  <source ... />
  <source ... />
</sources>
```

Listing 7.9 — Descriptif de l'élément `<source/>`

```

<source id="170"
  x="2,000000" y="2,000000" z="2,000000"
  u="1,000000" v="1,000000" w="1,000000"
  directivite="0"
  delay="0,000000">
  <bfreq freq="20000" db="75,686363"/>
  <bfreq freq="16000" db="75,686363"/>
  <bfreq freq="12500" db="75,686363"/>
  <bfreq freq="10000" db="75,686363"/>
  <bfreq freq="8000" db="75,686363"/>
  <bfreq freq="6300" db="75,686363"/>
  <bfreq freq="5000" db="75,686363"/>
  <bfreq freq="4000" db="75,686363"/>
  <bfreq freq="3150" db="75,686363"/>
  <bfreq freq="2500" db="75,686363"/>
  <bfreq freq="2000" db="75,686363"/>
  <bfreq freq="1600" db="75,686363"/>
  <bfreq freq="1250" db="75,686363"/>
  <bfreq freq="1000" db="75,686363"/>
  <bfreq freq="800" db="75,686363"/>
  <bfreq freq="630" db="75,686363"/>
  <bfreq freq="500" db="75,686363"/>
  <bfreq freq="400" db="75,686363"/>
  <bfreq freq="315" db="75,686363"/>
  <bfreq freq="250" db="75,686363"/>
  <bfreq freq="200" db="75,686363"/>
  <bfreq freq="160" db="75,686363"/>
  <bfreq freq="125" db="75,686363"/>
  <bfreq freq="100" db="75,686363"/>
  <bfreq freq="80" db="75,686363"/>
  <bfreq freq="63" db="75,686363"/>
  <bfreq freq="50" db="75,686363"/>
</source>

```

Listing 7.10 — Descriptif de l'élément `<recepteurss/>`

```

<recepteurss>
  <recepteur_surfacique ... />
  <recepteur_surfacique ... />
  <recepteur_surfacique ... />
</recepteurss>

```

Listing 7.11 — Descriptif de l'élément `<recepteur_surfacique/>`

```

<recepteur_surfacique id="164" name="" />

```

Listing 7.12 — Descriptif de l'élément `<recepteurssp/>`

```

<recepteurssp>
  <recepteur_ponctuel ... />
  <recepteur_ponctuel ... />
  <recepteur_ponctuel ... />
</recepteurssp>

```

Listing 7.13 — Descriptif de l'élément &lt;recepteur\_ponctuel/&gt;

```

<recepteur_ponctuel id="5882"
  name="" lbl="Récepteur 1"
  x="9,025040" y="10,033779" z="1,800000"
  u="0,056478" v="-0,994498" w="-0,088227">
  <bfreq freq="20000" db="-4,428017"/>
  <bfreq freq="16000" db="-5,428017"/>
  <bfreq freq="12500" db="-6,428016"/>
  <bfreq freq="10000" db="-7,428016"/>
  <bfreq freq="8000" db="-8,428017"/>
  <bfreq freq="6300" db="-9,428017"/>
  <bfreq freq="5000" db="-10,428017"/>
  <bfreq freq="4000" db="-11,428017"/>
  <bfreq freq="3150" db="-12,428017"/>
  <bfreq freq="2500" db="-13,428017"/>
  <bfreq freq="2000" db="-14,428017"/>
  <bfreq freq="1600" db="-15,428017"/>
  <bfreq freq="1250" db="-16,428017"/>
  <bfreq freq="1000" db="-17,428017"/>
  <bfreq freq="800" db="-18,428017"/>
  <bfreq freq="630" db="-19,428017"/>
  <bfreq freq="500" db="-20,428017"/>
  <bfreq freq="400" db="-21,428017"/>
  <bfreq freq="315" db="-22,428017"/>
  <bfreq freq="250" db="-23,428017"/>
  <bfreq freq="200" db="-24,428017"/>
  <bfreq freq="160" db="-25,428017"/>
  <bfreq freq="125" db="-26,428017"/>
  <bfreq freq="100" db="-27,428017"/>
  <bfreq freq="80" db="-28,428017"/>
  <bfreq freq="63" db="-29,428017"/>
  <bfreq freq="50" db="-30,428017"/>
</recepteur_ponctuel>

```

Listing 7.14 — Descriptif de l'élément &lt;encombrement\_enum/&gt;

```

<encombrement_enum>
  <encombrement ... />
  <encombrement ... />
  <encombrement ... />
</encombrement_enum>

```

Listing 7.15 — Descriptif de l'élément `<encombrement/>`

```
<encombrement id="11431">
  <bfreq freq="20000" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="16000" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="12500" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="10000" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="8000" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="6300" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="5000" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="4000" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="3150" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="2500" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="2000" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="1600" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="1250" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="1000" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="800" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="630" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="500" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="400" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="315" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="250" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="200" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="160" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="125" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="100" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="80" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="63" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
  <bfreq freq="50" alpha="0,000000" lambda="1,000000" loi_diff="0"/>
</encombrement>
```

### 7.3 Contenu et format du fichier de modèle géométrique

.cbin

Le fichier de modèle géométrique d'extension .cbin est un fichier binaire, permettant d'obtenir toutes les informations sur le modèle 3D. La structure de ce fichier est donnée au tableau 7.2.

La librairie `lib_interface` (fichier d'entête `input_output/bin.h`) dans le langage C++ permet de manipuler ce type de fichier, en utilisant la classe `formatCoreBIN::CformatBIN`. La manipulation de ce fichier peut également se faire par l'intermédiaire de la librairie `LibSimpa` dans le langage Python™ en utilisant la classe `LibSimpa.CformatBIN` (voir documentation de la librairie `LibSimpa`).

### 7.4 Contenu et format du fichier du maillage du modèle .mbin

Le fichier de modèle géométrique d'extension .mbin est un fichier binaire, contenant le maillage volumique du modèle. La structure de ce fichier est donnée au tableau 7.3.

La librairie `lib_interface` (fichier d'entête `input_output/bin.h`) dans le langage C++ permet de manipuler ce type de fichier, en utilisant la classe `formatMBIN::CMBIN`. Toutefois, comme pour le fichier de modèle, il est plus aisée de construire le fichier de maillage en utilisant la librairie `LibSimpa`, grâce à la classe `LibSimpa.CMBIN`.

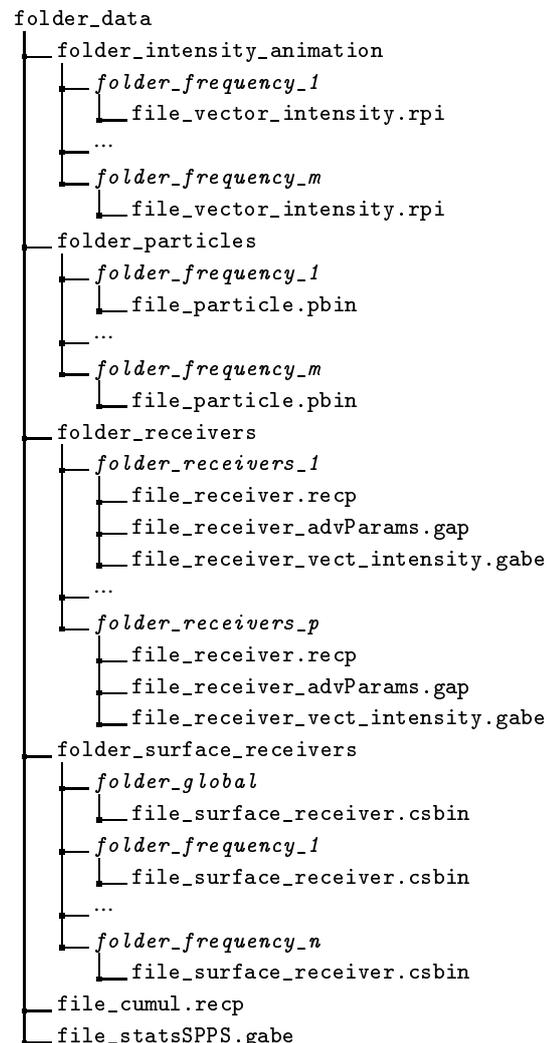
### 7.5 Contenu et format des fichiers de résultat

#### 7.5.1 Organisation des fichiers de résultat

Le code SPPS copie les résultats dans plusieurs fichiers et répertoires en fonction de leur nature.

Le format de base de la plupart des fichiers de résultat est le format GABE. Ce format de fichier binaire contient des séries de données où chaque série peut être de format différent (nombre à décimal, chaîne de caractères, nombre entier...).

La librairie `lib_interface` (fichier d'entête `input_output/gabe/gabe.h`) dans le langage C++ permet de manipuler ce type de fichier, en utilisant la classe `formatGABE::GABE`. La manipulation



**Figure 7.2** — Arborescence type des répertoires et fichiers de résultats issus du code SPPS. Les noms des dossiers et fichiers affichés en italique sont créés par le code de calcul. Les autres noms des dossiers et fichiers sont attribués par l'utilisateur dans le fichier de configuration.

de ce fichier peut également se faire par l'intermédiaire de la librairie `LibSimpa` dans le langage Python™ en utilisant la classe `LibSimpa.Gabe_rw` (voir documentation de la librairie `LibSimpa`).

Un exemple d'utilisation en Python™ est proposé dans le chapitre 9.

#### 7.5.1.1 Arborescence

De manière générale, l'arborescence des répertoires et des fichiers de résultat est donnée à la figure 7.2<sup>2</sup>.

<sup>2</sup>L'arborescence réelle dépend du nombre de récepteurs ponctuels, du nombre de récepteurs de surface, du nombre de bandes de fréquence considérées et des paramètres de calcul.

**Table 7.2** — Structure du fichier binaire de modèle géométrique *model\_geo.cbin*.

Taille	Occurrence	Description
int	1	Version du format majeur
int		Version du format mineur
unsigned short unsigned long unsigned long	1	<i>Nœud sommets</i> : Type de nœud (ici 0) Position du premier fils (0) Position du prochain frère (position du nœud groupe)
unsigned long		Nombre de sommets
float float float		Nombre de sommets
unsigned short unsigned long unsigned long	1	<i>Nœud groupe</i> : Type de nœud (ici 1) Position du premier fils (0) Position du prochain frère (fin de fichier)
Char * 255 unsigned int		1
unsigned int unsigned int unsigned int unsigned int int int	Nombre de faces	Indice du sommet A Indice du sommet B Indice du sommet C Indice du matériau (voir structure XML) Indice* du récepteur de surface (voir structure XML) Indice* de l'encombrement (voir structure XML) * l'indice est égal à -1 si la face n'est associée à aucun encombrement ou récepteur surfacique

**Table 7.3** — Structure du fichier binaire du fichier de maillage *mesh\_geo.cbin*.

Taille	Occurrence	Description
unsigned long unsigned long	1	Nombre de tétraèdres Nombre de nœuds
float float float		Nombre de nœuds
int int int int int	Nombre de tétraèdres	Indice des sommets A B C D Indice du volume
int int int int int		4

 En utilisant l'interface *I-Simpa*, les fichiers de résultat sont automatiquement disponibles dans l'onglet « Résultats » de la fenêtre « Projet », suivant cette arborescence.

### 7.5.1.2 Détail de l'arborescence

- `folder_data` :

- nom du répertoire (existant) contenant les fichiers de modèle `.cbin` et de maillage `.mbin`, dans lequel sera sauvegardé l'ensemble des résultats de calcul ;
- le nom du répertoire est défini par l'attribut `workingdirectory` de l'élément `<configuration/>` du fichier XML de configuration ;

- `folder_intensity_animation` :

- nom du répertoire contenant les fichiers d'animation des vecteurs intensités sonores ;
- le nom du répertoire est défini par l'attribut `intensity_folder` de l'élément `<simulation/>` du fichier XML de configuration ;
- `folder_frequency_n` :
  - \* nom du sous-répertoire désignant la bande de fréquence *n* ;
  - \* le nom du répertoire est créé automatiquement à partir de l'attribut `freq` de l'élément `<freq_enum/>` de la bande de fréquence *n* retenue pour le calcul.
- `file_vector_intensity_pri` :
  - \* nom du fichier contenant les coordonnées des vecteurs « intensité » au

- cours du temps, simultanément pour tous les récepteurs ponctuels, et pour une bande de fréquence (cf. section 4.4.2);
- \* le nom du fichier est défini par l'utilisateur à travers l'attribut `intensity_filename` de l'élément `<simulation/>`.
- `folder_particles` :
    - nom du répertoire contenant les fichiers d'animation des particules ;
    - ce répertoire est créé uniquement si le calcul est demandé, à savoir si l'attribut `nbparticules_rendu` de l'élément `<simulation/>` est non nul ;
    - le nom du répertoire est défini par l'utilisateur à travers l'attribut `particules_directory` de l'élément `<configuration/>`.
  - `folder_frequency_m` :
    - nom du sous-répertoire désignant la bande de fréquence  $m$  ;
    - le nom du répertoire est créé automatiquement à partir de l'attribut `freq` de l'élément `<freq_enum/>` de la bande de fréquence  $n$  retenue pour le calcul.
  - `file_particle.pbin` :
    - nom du fichier contenant l'animation des particules (position au cours de temps pour chaque particule sonore) dans la bande de fréquence considérée ;
    - le nom du fichier est défini par l'utilisateur à travers l'attribut `particules_filename` de l'élément `<configuration/>`.
  - `folder_receivers` :
    - nom du répertoire contenant les résultats des récepteurs ponctuels ;
    - le nom du répertoire est défini par l'utilisateur à travers l'attribut `receiversp_directory` de l'élément `<configuration/>`.
  - `folder_receivers_p` :
    - nom du répertoire contenant les résultats pour le récepteur  $p$  ;
  - le nom de répertoire est créé automatiquement en fonction du libellé du récepteur (attribut `lbl` de l'élément `<recepteur_ponctuel/>`).
  - `file_receiver.recip` :
    - nom du fichier contenant la donnée  $P2_{\text{rec}}^j(n)$  pour le récepteur en question (cf. section 4.4.1) ;
    - les résultats pour toutes les bandes de fréquence considérées sont donnés dans le même fichier ;
    - le nom du répertoire est défini par l'utilisateur à travers l'attribut `receiversp_filename` de l'élément `<configuration/>`.
  - `file_receiver_advParams.gap` :
    - nom du fichier contenant les données complémentaires  $P2_{\text{rec},\cos\theta}^j(n)$  et  $P2_{\text{rec},\cos^2\theta}^j(n)$  pour le récepteur en question (cf. section 4.4.3) ;
    - les résultats pour toutes les bandes de fréquence considérées sont donnés dans le même fichier ;
    - le nom du répertoire est défini par l'utilisateur à travers l'attribut `receiversp_filename_adv` de l'élément `<configuration/>`.
  - `file_receiver_vect_intensity.gabe` :
    - nom du fichier contenant les coordonnées du vecteur « intensité »  $\mathbf{I}_{\text{rec}}^j(n)$  pour le récepteur en question, pour toutes les bandes de fréquence (cf. section 4.4.2) ;
    - les résultats pour toutes les bandes de fréquence considérées sont donnés dans le même fichier ;
    - le nom du fichier est défini par l'utilisateur à travers l'attribut `intensity_rp_filename` de l'élément `<simulation/>`.
  - `folder_surface_receivers` :
    - nom du répertoire contenant les résultats des récepteurs surfaciques ;
    - le nom du répertoire est défini par l'utilisateur à travers l'attribut `recepteurss_directory` de l'élément `<configuration/>`.
  - `folder_global` :

- nom du répertoire contenant les résultats des récepteurs surfaciques en global ;
  - le nom de ce répertoire est fixé par le code SPPS.
- `file_surface_receiver.csbin` :
    - nom du fichier contenant les résultats pour la bande de fréquence  $n$  ;
    - le nom du fichier est défini par l'utilisateur à travers l'attribut `recepteurss_filename` `recepteurss_directory` de l'élément `<configuration/>`.
  - `folder_frequency_n` :
    - nom du répertoire contenant les résultats pour la bande de fréquence  $n$  ;
    - le répertoire est créé uniquement si l'attribut `output_recs_byfreq` de l'élément `<simulation/>` est fixé à "1" ;
    - le nom du répertoire est créé automatiquement à partir de la valeur médiane de la bande de fréquence  $n$ .
  - `file_cumul.recip` :
    - nom du fichier contenant le résultat cumulé pour tout le modèle ;
    - le nom du répertoire est défini par l'utilisateur à travers l'attribut `cumul_filename` de l'élément `<configuration/>`.
  - `file_statsSPPS.gabe` :
    - nom du fichier contenant les informations de calcul ;
    - le nom du fichier est défini par l'utilisateur à travers l'attribut `stats_filename` de l'élément `<simulation/>`.

## 7.5.2 Fichiers pour un récepteur ponctuel

### 7.5.2.1 Fichier de récepteur volumique (.recp) pour le calcul du niveau de pression acoustique

Ce type de fichier utilisé pour le calcul du niveau de pression acoustique (cf. paragraphe 4.4.1) pour un récepteur volumique.

Les résultats au niveau des récepteurs ponctuels sont enregistrés dans un fichier binaire au format GABE avec une extension `.recp`. Ce fichier définit un tableau (axe des temps en ligne, axe des bandes de fréquence en colonne), contenant les quantités

$P_{rec}^j(n)$  (homogène au carré de la pression acoustique,  $j$  étant l'indice des fréquences, et  $n$  l'indice du pas de temps). Ces données sont organisées de la manière suivante (tableau 7.4) :

- la première colonne, de type chaîne de caractères, contient les libellés des pas de temps, à partir de la deuxième ligne<sup>3</sup> ;
- la première ligne, de type chaîne de caractères, contient les libellés des bandes de fréquence, à partir de la deuxième colonne ;
- à partir de la seconde colonne, chaque colonne contient les résultats du calcul  $P_{rec}^j(n)$  (de type décimal) pour chaque bande de fréquence (indice  $j$ ), en fonction du pas de temps (indice  $n$ ).

**Table 7.4** — Structure du fichier `.recp` contenant les résultats du calcul pour un récepteur ponctuel.

	125Hz	250 Hz	...
10 ms	$P_{rec}^1(1)$	$P_{rec}^2(1)$	...
20 ms	$P_{rec}^1(2)$	$P_{rec}^2(2)$	...
30 ms	$P_{rec}^1(3)$	$P_{rec}^2(3)$	...
⋮	...	...	...

 Pour bénéficier d'une traduction automatique sous *I-Simpa*, le fichier en question doit être libellé `soundpressure.recip`. Dans le cas contraire, l'élément correspondant dans l'arbre de données portera le nom du fichier, quelque soit la langue.

### 7.5.2.2 Fichier de récepteur volumique (.gap) pour le calcul de paramètres avancés (pression acoustique « latérale »)

Ce type de fichier utilisé pour le calcul des paramètres avancés pour un récepteur volumique.

**Format général** Les résultats des paramètres avancés au niveau des récepteurs ponctuels sont enregistrés dans un fichier binaire au format GABE avec une extension `.gap`. Ce fichier contient un ensemble de données organisées sous forme de séries, chaque série étant distribuées sur une ou plusieurs colonnes. Le tableau ?? illustre le format du fichier `.gap`. La première colonne est la série d'index, les colonnes suivantes contiennent les données du calcul, à savoir les données  $P_{rec}^j(n)$ ,  $P_{rec,cos\theta}^j(n)$  et  $P_{rec,cos^2\theta}^j(n)$  ( $j$  désignant l'indice de la bande de

<sup>3</sup>La première cellule (première ligne, première colonne) est laissée vide.

fréquence, et  $n$  l'indice du pas de temps). A noter que la première données est redondante  $P_{rec}^j(n)$  avec celle présente dans le fichier de récepteur surfacique pour les le calcul des niveau de pression acoustique (paragraphe 7.5.2.1).

 Pour bénéficier d'une traduction automatique sous *I-Simpa*, le fichier en question doit être libellé *advParams.gap*. Dans le cas contraire, l'élément correspondant dans l'arbre de données portera le nom du fichier, quelque soit la langue.

**Série d'index** La première colonne (tableau ??), de type `entier`, contient la série d'index, *i.e.* les numéros de colonne pour chaque type de série de données. Le dernier élément de la colonne (ligne 7) contient la nombre de pas de temps. L'ensemble de la série contient 8 lignes, numérotés de 0 à 7.

#### Série des paramètres de temps [H]

Cette série de paramètre retourne le pas de temps de calcul  $\Delta t$ , ainsi que la durée totale de la simulation  $t_0$  (tableau 7.7). Ces paramètres sont de type `decimal`, et exprimés en seconde (s). Ces données sont redondantes avec les paramètres du fichier de configuration du calcul.

**Table 7.7** — Série des paramètres de temps (`decimal`, en s).

1	Pas de temps $\Delta t$ (s)
2	Durée totale $t_0$ (s)

**Série contenant la puissance acoustique cumulée des sources sonores** Cette série contient la puissance cumulée (pondérée par le terme  $\rho c$ ) de toutes les sources sonores actives dans le modèle, par bande de fréquence (tableau 7.8). Les données sont de type `decimal` et exprimées en Watt (W). Ce cumul des puissances acoustiques peut être nécessaire pour le calcul de certains paramètres acoustiques, comme la force (*strength*, G). Si  $W_{ij}$  est la puissance active de la source  $i$  dans la bande de fréquence  $j$ , alors, la valeur  $P_j$  retournée dans cette liste pour chaque bande de fréquence  $j$ , est :

$$P_j = \sum_i^N \rho c W_{ij}, \quad (7.1)$$

$N$  désignant le nombre de sources actives.

**Table 7.8** — Série contenant la puissance cumulée de toutes les sources sonores actives dans le modèle (`decimal`, en W).

1	Puissance acoustique $P_0$ dans la bande de fréquence 1
⋮	⋮
$N_f$	Puissance acoustique $P_n$ dans la bande de fréquence $N_f$

#### Série contenant la liste des bandes de fréquence

Cette série retourne la liste des bandes de fréquence utilisée dans les calculs (tableau 7.9). Cette donnée est de type `entier`<sup>4</sup>, exprimée en Hertz (Hz).

**Table 7.9** — Série contenant la liste des bandes de fréquence (`entier`, en Hz).

1	Valeur de la fréquence 1
⋮	⋮
$N_f$	Valeur de la fréquence $N_f$

**Série contenant le bruit de fond** Cette série de données retourne le spectre de bruit de fond *BdF* pour le récepteur récepteur considéré. La série contient autant de lignes que de bandes de fréquence (tableau 7.10). Les données sont de type `decimal`, et exprimées en décibel (dB).

**Table 7.10** — Série contenant le spectre (`decimal`, en dB).

1	Niveau de bruit de fond sur la première bande de fréquence 1
⋮	⋮
$N_f$	Niveau de bruit de fond sur la bande de fréquence $N_f$

#### Série contenant l'énergie sonore calculée

Chaque bande de fréquence est divisée en 3 séries successives où chaque ligne correspond à un pas de temps (tableau 7.11). Les données sont de type `decimal`, homogène au carré de la pression ( $\text{Pa}^2$ ).

<sup>4</sup>Il s'agit du centre de la bande de fréquence, *i.e.* la fréquence médiane normalisée.

**Table 7.11** — Série contenant l'énergie sonore calculée pour chaque fréquence et chaque pas de temps.

1	$P2_{rec}^j(1)$	$P2_{rec,cos\theta}^j(1)$	$P2_{rec,cos^2\theta}^j(1)$
⋮	⋮	⋮	⋮
$N_t$	$P2_{rec}^j(N_t)$	$P2_{rec,cos\theta}^j(N_t)$	$P2_{rec,cos^2\theta}^j(N_t)$

### 7.5.2.3 Fichier de récepteur volumique (.gabe) pour le calcul du vecteur « intensité »

Ce fichier de données contient le vecteur intensité sonore  $I_{rec}^j(n)$  au point de réception considéré, pour simultanément pour toutes les bandes de fréquence  $j$ ). Le fichier est au format binaire GABE, avec une extension .gabe, et contient plusieurs séries de données (tableau ??). La première colonne contient l'échelle des temps. Les colonnes suivantes contiennent les résultats par bande de fréquence par séries successives de 3 colonnes, respectivement pour les coordonnées  $x$ ,  $y$  et  $z$  (autant de séries de 3 colonnes que de bandes de fréquence). La première ligne et la première colonne sont de type chaîne de caractères, les autres données sont de type decimal.

## 7.5.3 Fichiers de récepteur surfacique

### 7.5.3.1 Fichier de récepteur surfacique (.csbin) pour la représentation de l'intensité acoustique (cartographie)

Ce format de fichier binaire permet de stocker les intensités acoustiques des récepteurs surfaciques au cours du temps. La structure du fichier est organisée de la manière suivante (tableau ??) :

La librairie `lib_interface` (fichier `input_output/exportRecepteurSurf/std_rsbin.hpp`) dans le langage C++ permet de manipuler ce type de fichier, en utilisant la classe `formatRSBIN::rsurf_io`. La manipulation de ce fichier peut également se faire par l'intermédiaire de la librairie `LibSimpa` dans le langage Python™ en utilisant la classe `LibSimpa.rsurf_io` (voir documentation de la librairie `LibSimpa`).

## 7.5.4 Fichiers généraux

### 7.5.4.1 Fichier de cumul (.recp) pour le calcul du niveau de pression acoustique global dans le modèle

Les résultats cumulés dans tous le domaine (le domaine étant alors défini comme un unique récepteur volumique) sont enregistrés dans un fichier binaire au format GABE avec une extension .recp. Ce fichier

est identique à celui défini pour un récepteur volumique (cf. paragraphe 7.5.2.1) et contient les quantités  $P2_{global}^j(n)$  (homogène au carré de la pression acoustique,  $j$  étant l'indice des fréquences, et  $n$  l'indice du pas de temps) utilisé pour la calcul du niveau de pression acoustique global (cf. paragraphe 4.4.6). Ces données sont organisées de la manière suivante (tableau 7.4) :

- la première colonne, de type chaîne de caractères, contient les libellés des pas de temps, à partir de la deuxième ligne<sup>5</sup> ;
- la première ligne, de type chaîne de caractères, contient les libellés des bandes de fréquence, à partir de la deuxième colonne ;
- à partir de la seconde colonne, chaque colonne contient les résultats du calcul  $P2_{global}^j(n)$  (de type decimal) pour chaque bande de fréquence (indice  $j$ ), en fonction du pas de temps (indice  $n$ ).

**Table 7.14** — Structure du fichier .recp contenant les résultats du calcul cumulé sur l'ensemble du domaine.

	125Hz	250 Hz	...
10 ms	$P2_{global}^1(1)$	$P2_{global}^2(1)$	...
20 ms	$P2_{global}^1(2)$	$P2_{global}^2(2)$	...
30 ms	$P2_{global}^1(3)$	$P2_{global}^2(3)$	...
⋮	...	...	...

### 7.5.4.2 Fichier de suivi des particules (.pbin)

Ce fichier binaire permet de stocker la position des particules (*quantité énergétique*) au cours du temps. Ce fichier n'est créé que si l'attribut `nbparticules_rendu` du fichier de configuration XML est différent de "0". La structure du fichier (format spécifique `libinterface:ParticleIO`; voir la documentation de la bibliothèque `libinterface`) est donnée dans le tableau 7.15. Ce fichier peut être utilisé pour représenter graphique le déplacement d'une quantité énergétique au cours du temps.

### 7.5.4.3 Fichier de représentation des vecteurs « intensité » (.pri)

**Format général** Ce fichier de données contient le vecteur « intensité sonore »  $I_{rec}^j(n)$  pour tous les récepteurs ponctuels considérés (pour chaque bande

<sup>5</sup>La première cellule (première ligne, première colonne) est laissée vide.

**Table 7.15** — Structure du fichier binaire contenant la position quantité énergétique au cours du temps (fichier `.rpi`).

Taille	Occurrence	Description
unsigned long unsigned long unsigned long unsigned long unsigned long unsigned long unsigned long	1	Entête du fichier : Nombre de particules dans le fichier Version du format Taille en octet de l'information du fichier d'entête Taille en octet d'une information de pas de temps Taille en octet d'une information de quantité énergétique Non utilisé Pas de temps de la simulation
unsigned long unsigned short	1	Entête d'une particule : Nombre de pas de temps Premier pas de temps
float float float float	nombre de pas de temps	Position $x y z$ Énergie
unsigned long unsigned short	1	Entête de la quantité énergétique : Nombre de pas de temps Premier pas de temps
float float float float	nombre de pas de temps	Position $x y z$ Énergie
:	:	:

**Table 7.17** — Signification de la première colonne (série d'index) du fichier au format GABE d'extension `.rpi`.

1	Nombre de récepteurs ponctuels
2	Nombre de pas de temps
3	Nombre de colonnes définissant un récepteur ponctuel
4	Numéro de la colonne du premier élément du récepteur ponctuel

de fréquence  $j$ ). Le format de fichier de base est le format binaire GABE, avec une extension `.rpi`, et contient plusieurs séries de données (tableau ??). Ce fichier peut être utilisé pour représenter simultanément les vecteurs « intensité » pour tous les points d'observation (graphique animé par exemple). Les fichiers `.rpi` sont une réorganisation en bande de fréquence, des fichiers de vecteurs « intensité » pour les récepteurs ponctuels `.gabe` (cf. paragraphe 7.5.2.3).

**Série d'index** Cette série, située dans la première colonne, de type `entier`, contient les informations pour chaque type de séries de données.

**Série de paramètres** La deuxième colonne contient la série de paramètres, de type `decimal`, limitée dans cette version à la seule valeur du pas de temps  $\Delta t$ .

**Série  $x$  d'un récepteur ponctuel** Cette colonne, de type `float`, contient la valeur de la coordonnée

**Table 7.18** — Signification de la seconde colonne (série de paramètres) du fichier au format GABE d'extension `.rpi`.

1	Pas de temps (s)
---	------------------

**Table 7.19** — Série  $x$  d'un récepteur ponctuel dans le fichier au format GABE d'extension `.rpi`.

1	Position $x$ du récepteur
2	Direction $x$ du vecteur au pas de temps 1 (premier pas de temps)
3	Direction $x$ du vecteur au pas de temps 2
4	Direction $x$ du vecteur au pas de temps 3

$x$  (première ligne) de la position du récepteur et les coordonnées  $x$  du vecteur intensité à chaque pas de temps pour le récepteur ponctuel considéré.

**Série  $y$  d'un récepteur ponctuel** Cette colonne, de type `float`, contient la valeur de la coordonnée  $y$  (première ligne) de la position du récepteur et les coordonnées  $y$  du vecteur intensité à chaque pas de temps pour le récepteur ponctuel considéré.

**Série  $z$  d'un récepteur ponctuel** Cette colonne, de type `float`, contient la valeur de la coordonnée  $z$  (première ligne) de la position du récepteur et les coordonnées  $z$  du vecteur intensité à chaque pas de temps pour le récepteur ponctuel considéré.

**Table 7.20** — Série  $y$  d'un récepteur ponctuel dans le fichier au format GABE d'extension `.rpi`.

1	Position $y$ du récepteur
2	Direction $y$ du vecteur au pas de temps 1 (premier pas de temps)
3	Direction $y$ du vecteur au pas de temps 2
4	Direction $y$ du vecteur au pas de temps 3

**Table 7.21** — Série  $z$  d'un récepteur ponctuel dans le fichier au format GABE d'extension `.rpi`.

1	Position $z$ du récepteur
2	Direction $z$ du vecteur au pas de temps 1 (premier pas de temps)
3	Direction $z$ du vecteur au pas de temps 2
4	Direction $z$ du vecteur au pas de temps 3

**7.5.4.4 Fichier des statistiques du calcul (.gabe)**

Le fichier des statistiques du calcul SPPS est un fichier binaire au format GABE et d'extension `.gabe`. Il contient un certain nombre d'information sur le déroulement du calcul (tableau 7.22). La première ligne et la première colonne (excepté la première cellule qui est vide) sont des chaînes de caractères (les fréquences en colonnes, les informations en lignes). Les autres cellules du tableau de données sont des entiers :

- Particules absorbées par l'atmosphère : nombre de particules disparaissant du domaine de propagation en raison de l'absorption atmosphérique (différent de 0, uniquement en mode de calcul « aléatoire ») ;
- Particules absorbées par les matériaux : nombre de particules disparaissant du domaine de propagation en raison de l'absorption par des faces du domaine (différent de 0, uniquement en mode de calcul « aléatoire ») ;
- Particules absorbées par un encombrement : nombre de particules disparaissant du domaine de propagation en raison de l'absorption par des objets de l'encombrements (différent de 0, uniquement en mode de calcul « aléatoire ») ;
- Particules perdues dû aux boucles infinies : problème numérique causant une disparition artificielle de particules sonores (cf. Notes de la page 50) ;
- Particules perdues dû au maillage incorrecte : si le maillage est incorrecte (mailles manquantes

par exemple), il est possible qu'une particule sonore dans une maille courante doivent se propager dans des mailles voisines ((cf. Notes de la page 47) ;

- Particules restantes : particules restantes dans le domaine de propagation à la fin du calcul ;
- Total : total des colonnes précédentes. Le nombre total de particules doit en principe correspondre au nombre de particules par source sonore, multiplié par le nombre de sources sonores actives, sauf si la transmission acoustique est active et que le mode de calcul est « énergétique » (en raison de la duplication des particules sonores).

**Table 7.22** — Structure du fichier des statistiques du calcul `.gabe`.

	125Hz	...
Part. absorbées / atmosphère	...	...
Part. absorbées / matériaux	...	...
Part. absorbées / encombrement	...	...
Part. perdues / boucles infinies	...	...
Part. perdues / maillage incorrecte	...	...
Part. restantes	...	...
Total	...	...



# Validation du code

## 8.1 Principe

Ce chapitre présente des éléments de validation du code SPPS, dans le cadre de son utilisation pour la modélisation de champs sonores en milieu fermé (acoustique des salles et locaux industriels). La validation en milieu extérieur (milieu ouvert, péri-urbain ou urbain) n'a pas été encore été réalisé à ce stade. Pour valider le code, nous aurions pu réaliser des mesures de niveaux sonores et de temps de réverbération dans diverses situations (salles homogènes, couloirs, halls industriels...), afin de les comparer aux résultats des simulations SPPS. Néanmoins, afin d'être rigoureux et exhaustif, ce travail aurait nécessité un très grand nombre d'expérimentations et de caractéristiques acoustiques différentes (réflexions spéculaires, diffuses, mixtes...), ce qui en pratique n'aurait pas été réalisable. A l'inverse, on trouve dans la littérature un certain nombre de résultats expérimentaux, numériques et analytiques concernant les champs sonores dans de tels espaces. Il nous a donc semblé intéressant de comparer ces données aux résultats du code SPPS.

Afin d'être le plus exhaustif possible, nous avons donc recherché dans la littérature des données utilisables dans des configurations très diverses. Pour étudier le comportement du code SPPS en fonction des configurations étudiées, nous avons entrepris de classer ces données bibliographiques en plusieurs catégories de salles et plusieurs types de condition de réflexion au niveau des parois. Parmi toutes les données bibliographiques disponibles, nous n'avons retenu que celles qui étaient suffisamment documentées pour réaliser des simulations numériques avec le code SPPS dans des configurations équivalentes.

## 8.2 Validation en acoustique des salles

### 8.2.1 Salles de dimensions homogènes

#### 8.2.1.1 Comparaison avec les données de Yang (2000) [35] en champ spéculaire

**Données** Dans un article récent [35], Yang et Shield ont proposé un programme informatique (NORMAL) pour le calcul de champs sonores dans des espaces allongés, basé sur le tracé de rayons sonores. Afin de valider ce programme, des simulations ont été réalisées pour une salle quasi-cubique et une longue salle, puis comparées à la théorie classique de l'acoustique des salles (Théorie de Sabine). Les valeurs SPL obtenues par le programme NORMAL et les valeurs calculées par la théorie classique sont proposées pour chacun des points d'observation défini dans [35, tableau 2, page 138] (cf. figure 8.9). De même pour les temps de réverbération EDT et RT30 (cf. [35, tableau 3, page 139]).

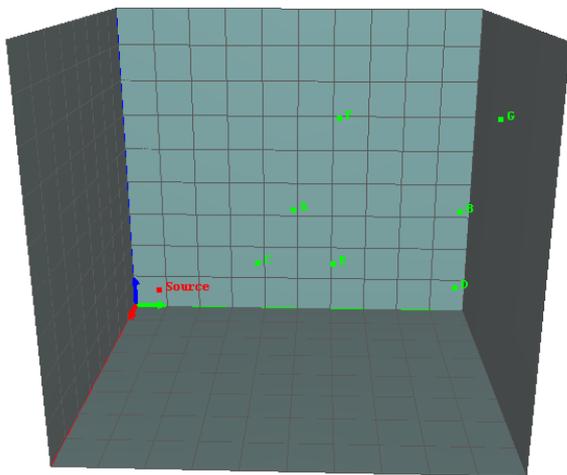
**Paramètres de SPPS** L'absorption atmosphérique peut être prise en compte par le programme NORMAL. Néanmoins, dans les simulations présentées, cette prise en compte n'est pas précisée. Nous supposons donc qu'elle n'est pas considérée. Par ailleurs, la valeur de la célérité du son n'est pas précisée non plus ; nous nous sommes placés dans des conditions atmosphériques normales (température 20°C, 50% d'humidité relative). Les paramètres de simulation SPPS sont les suivants : 2 millions de particules sonores, pas de temps de 2 ms pour une durée de simulation de 3 s. Les simulations sont réalisées sur une seule bande de fréquence. De manière à obtenir une puissance sonore de 90 dB pour la source dans

**Table 8.1** — Caractéristiques de la configuration étudiée dans la référence [35].

Paroi	Sol	Toit	Mur G.	Mur D.	Ext. 1	Ext. 2
Absorption	0.1					
Diffraction $\delta$	$\delta = 0.00$					
Réflexion	Spéculaire					

Position	X	Y	Z
Salle	8 m	10 m	9 m
Source (90 dB)	1 m	1 m	1 m

Récepteurs	A	B	C	D	E	F	G
X	4.0 m	4.0 m	2.5 m	2.5 m	2.5 m	7.5 m	7.5 m
Y	5.0 m	9.0 m	4.0 m	9.0 m	6.0 m	6.0 m	9.0 m
Z	4.5 m	4.5 m	2.5 m	2.5 m	2.5 m	7.5 m	7.5 m

**Figure 8.1** — Représentation sous I-Simpa de la géométrie utilisée dans les simulations de Yang, simulée avec le code SPPS.

la bande de calcul considérée, la puissance globale est fixée à 104.31 dB<sup>1</sup>.

**Résultats** Les résultats sont présentés à la figure 8.2. En ce qui concerne les niveaux sonores calculés pour des récepteurs de référence localisés dans la salle (figure 8.2(a)), les résultats obtenus par SPPS sont très proches de ceux du programme NORMAL, ainsi que ceux de la théorie classique de la réverbération. Les écarts maximum observés avec le programme NORMAL sont de l'ordre de 0.12 dB en moyenne. Concernant les temps de réverbération,

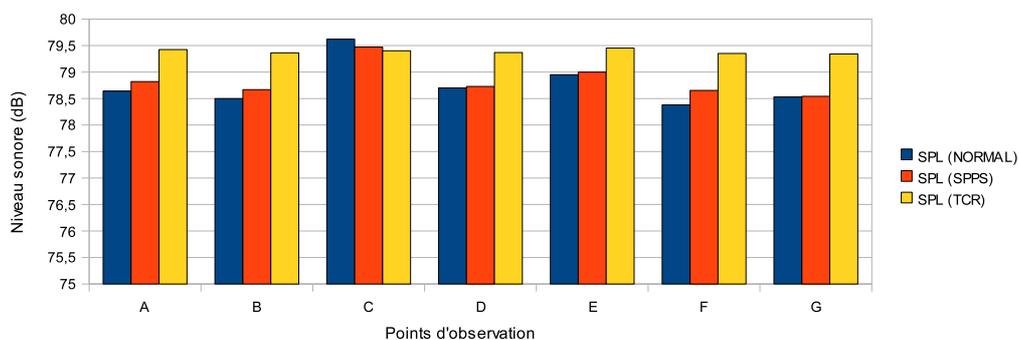
<sup>1</sup> Avec une source de bruit rose, la puissance globale se répartit uniformément sur les 27 bandes de calcul utilisées par le code SPPS, soit 90 dB par bande.

tion, les figures 8.2(b) et 8.2(c) montre également un bon accord entre les temps de réverbération EDT et RT30 calculés par le code SPPS et par le programme NORMAL, avec un écart maximum de l'ordre de 0.17 s pour l'EDT (0.09 s d'écart moyen) et 0.10 s pour le RT30 (0.06 s d'écart moyen). Toutefois, les valeurs du TR semblent légèrement supérieures à celle de Yang. Enfin, le temps de réverbération Sabine calculé par nos soins est de 2.42 s, soit très proche de la valeur annoncée par Yang, à savoir 2.39 s, mais également très proche de la valeur obtenue par le code SPPS en prenant en compte la décroissance globale de l'énergie dans la salle, soit 2.45 s pour le TR et 2.35 s pour l'EDT. **En conclusion, le code SPPS est bien validé pour cette configuration.**

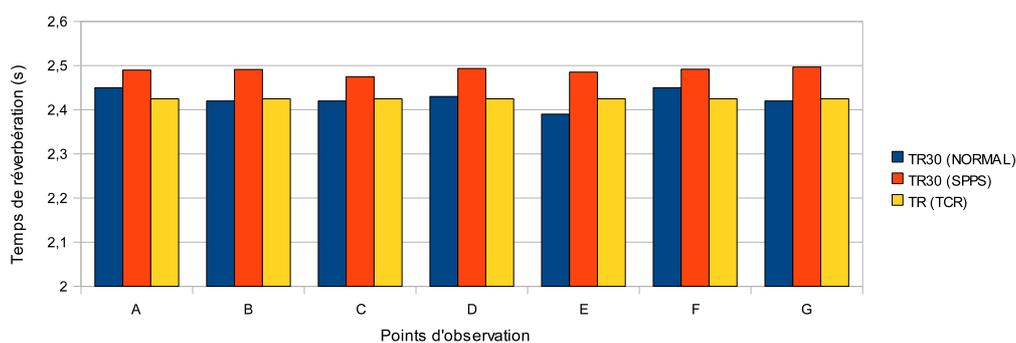
### 8.2.1.2 Comparaison avec les données de Kuttruff (1997) [24] en champ diffus

**Données** Dans le cadre de ses travaux en acoustique des salles, Kuttruff a développé une méthode intégrale pour le calcul du champ sonore dans une salle diffuse [25]. Cette méthode a été appliquée dans de nombreuses configurations et a montré un comportement intéressant, même s'il existe à notre connaissance peu de validation expérimentale. Dans un article récent, Kuttruff a notamment appliqué cette méthode pour des salles cubiques et rectangulaires, caractérisées par des répartitions non-uniformes de l'absorption acoustique [24]. Les résultats présentés dans ce paragraphe concerne uniquement les applications aux salles cubiques (configurations K6 et K5, figure 8.3 et tableau 8.2). L'application à des locaux plats (configurations K4 à K2) sera discutée au paragraphe 8.2.3.2.

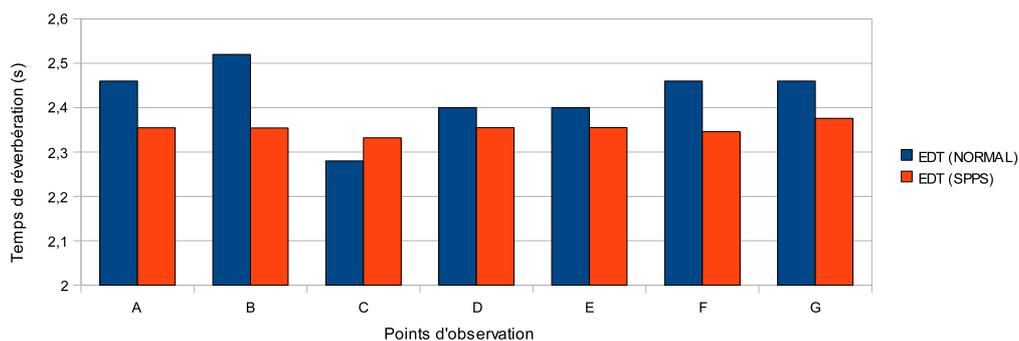
La position de la source n'est pas clairement indiquée. Par extension avec la configuration sphérique présentée dans ce même article, nous supposons



(a) Niveaux sonores



(b) Temps de réverbération (TR)



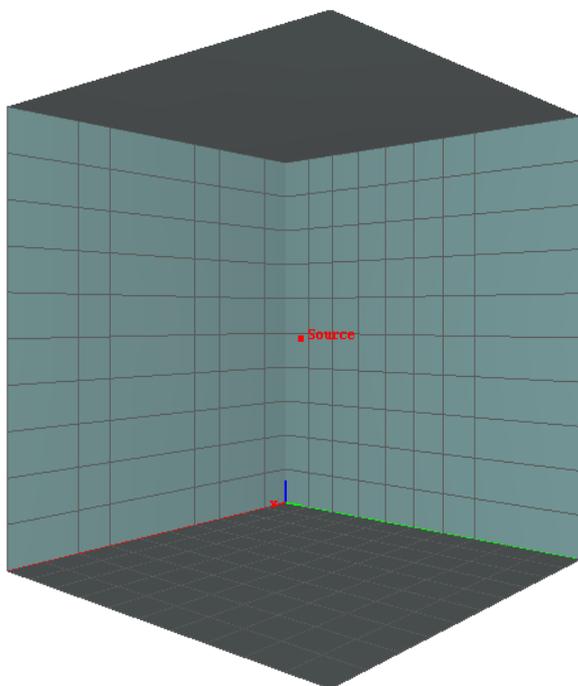
(c) Temps de réverbération (EDT)

**Figure 8.2** — Comparaison des simulations numériques *SPPS* avec les résultats de Yang (2000), pour les (a) niveaux sonores, (b) TR et (c) EDT. Les valeurs obtenues par le code *TCR* modélisant la théorie classique de la réverbération sont également données pour le niveau sonore et le temps de réverbération<sup>2</sup>.

**Table 8.2** — Caractéristiques des deux salles cubiques étudiées par [24].

Paroi	Sol	Plafond	Mur G.	Mur D.	Ext. 1	Ext. 2
Absorption (Config. K6)	$\alpha = \bar{\alpha} = 1/6$					
Absorption (Config. K5)	1.0	0.0	0.0	0.0	0.0	0.0
Diffraction $\delta$	$\delta = 1.00$					
Réflexion	Lambert					

Position	X	Y	Z
Salle	10 m	10 m	10 m
Source (0 dB)	5 m	5 m	5 m
Points d'observation	—	—	—

**Figure 8.3** — Représentation sous *I-Simpa* de la géométrie utilisée dans les simulations de Kuttruff (1997), simulée avec le code *SPPS*, pour des salles cubiques.

que la source est située au milieu de la salle. La méthode de calcul des temps de réverbération n'est également pas précisée. Par construction, la méthode intégrale de Kuttruff permet de calculer la décroissance d'énergie en sommant les contributions énergétiques de chaque élément de paroi, à chaque pas de temps. Cette procédure permet ensuite de remonter au temps de réverbération *global*. Ce principe a également été retenu dans nos simulations : les temps de réverbération ont été calculés à partir de la décroissance globale d'énergie dans la salle.

**Paramètres de SPPS** L'absorption atmosphérique ne semble pas avoir été considérée par Kuttruff. Par ailleurs, la valeur de la célérité du son n'est pas précisée non plus ; nous nous sommes placés dans des conditions atmosphériques normales (température 20°C, 50% d'humidité relative). Les paramètres de simulation SPPS sont les suivants : 500 000 particules sonores, pas de temps de 2 ms pour une durée de simulation de 2 s. Les simulations sont réalisées sur deux bandes de fréquence, la première pour simuler la configuration K6, la seconde pour simuler la configuration K5. De manière à obtenir une puissance sonore de 0 dB pour la source dans la bande de calcul considérée, la puissance globale est fixée à 14.31 dB<sup>3</sup>.

**Résultats** Le tableau 8.3 et la figure 8.4 illustrent la comparaison entre la méthode de Kuttruff et le code SPPS, ainsi qu'avec les formules de Sabine et d'Eyring<sup>4</sup>. Il semble que Kuttruff ait *a priori* calculé les TR sur la partie linéaire des courbes obtenues par sa méthode ; il convient donc de comparer ses résultats avec la valeur du RT30 SPPS.

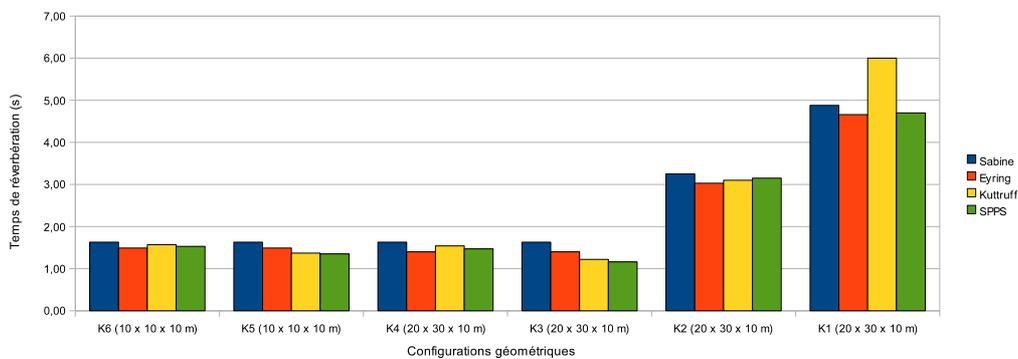
En ce qui concerne la configuration cubique caractérisée par une absorption uniforme (configuration K6), les valeurs des TR données par SPPS et Kuttruff sont similaires, et très proches de celle de Sabine et d'Eyring. Ce résultat est tout à fait attendu dans la mesure où la formule de Sabine est bien adaptée à des

<sup>3</sup>Avec une source de bruit rose, la puissance globale se répartit uniformément sur les 27 bandes de calcul utilisées par le code SPPS, soit 0 dB par bande.

<sup>4</sup>L'application de la formule d'Eyring, faisant intervenir le terme  $\ln(1-\alpha)$  pose normalement problème quand  $\alpha = 1$ , donc en particulier pour les configurations K5, K3, K2 et K1. Il est probable que Kuttruff ait considéré l'aire d'absorption équivalente comme étant nulle pour les parois définies par  $\alpha = 1$  (application de la formule de Sabine).

**Table 8.3** — Comparaison des temps de réverbération obtenus par les formules de Sabine et d'Eyring avec le modèle de Kuttruff (1997) et le code SPPS, pour des salles cubiques et non-homogènes, caractérisées par des distributions d'absorption différentes. En présence d'une absorption importante (configurations K5, K3, K2 et K1 avec une absorption  $\alpha = 1.00$  sur une des parois), la formule d'Eyring est plus adaptée. A l'inverse, en présence d'une absorption faible et homogène, la formule de Sabine est plus adaptée (configurations K6 et K4). Les valeurs les plus adaptées à chaque configuration sont indiquées par le symbole \*.

Configuration	Sabine	Eyring	Kuttruff	SPPS (RT30)	SPPS (EDT)
(K6) 10 × 10 × 10 m	1.63*	1.49	1.57	1.53	1.52
(K5) 10 × 10 × 10 m	1.63	1.49*	1.37	1.35	1.34
(K4) 10 × 20 × 30 m	1.63*	1.39	1.54	1.47	1.47
(K3) 10 × 20 × 30 m	1.63	1.39*	1.22	1.16	1.10
(K2) 10 × 20 × 30 m	3.25	3.02*	3.10	3.15	3.09
(K1) 10 × 20 × 30 m	4.88	4.65*	6.00	4.70	4.71



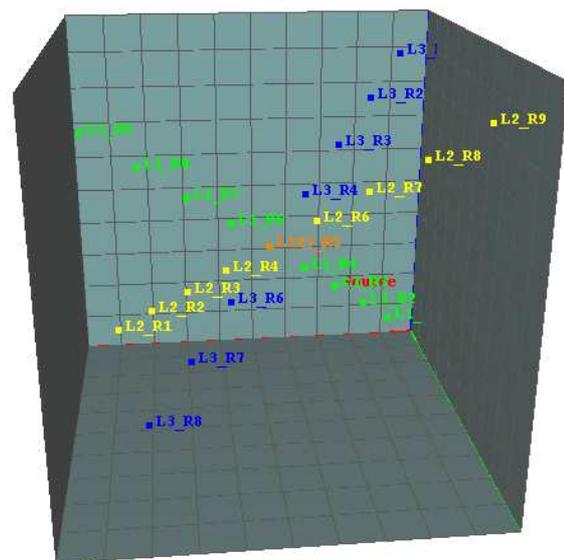
**Figure 8.4** — Comparaison des simulations numériques SPPS avec les résultats de Kuttruff (1997).

géométries homogènes, d'absorption faible<sup>5</sup> et homogènes. Pour la configuration cubique d'absorption non-uniforme (configuration K5), le code SPPS et le modèle de Kuttruff sont là encore similaires, et assez proche de la formule d'Eyring (plus appropriée que la formule de Sabine pour cette configuration). Les écarts entre le modèle de Kuttruff et le code SPPS sont respectivement de 2.55% et 1.35% pour les configurations K6 et K1. **En conclusion, le code SPPS est bien validé pour ces configurations.**

### 8.2.1.3 Comparaison avec les données de Kang (2002) [21] en champ diffus

**Données** Dans un récent ouvrage [21], Kang a proposé une méthode pour la modélisation des champs sonores diffus dans une salle, basée sur le principe de radiosité, qu'il a comparé avec la théorie classique de Sabine et d'Eyring. Il propose également une application de sa méthode à une salle cubique parfaitement diffuse et d'absorption uniforme ( $\alpha = 0.2$ ).

<sup>5</sup>L'application de la formule d'Eyring dans cette configuration n'est en principe pas très appropriée, puisque l'absorption est forte sur une des parois, pour la configuration K6.



**Figure 8.5** — Représentation sous *I-Simpa* de la géométrie utilisée dans les simulations de Kang, simulée avec le code SPPS.

**Table 8.4** — Caractéristiques de la configuration étudiée par [21, figure 2.20, page 57].

Paroi	Sol	Toit	Mur G.	Mur D.	Ext. 1	Ext. 2
Absorption	0.2					
Diffraction $\delta$	$\delta = 1.0$					
Réflexion	Diffus					

Position	X	Y	Z
Salle	10 m	10 m	10 m
Source	3 m	3 m	3 m

Récepteurs (ligne 1)	1	2	3	4	5	6	7	8	9
X	1 m	2 m	3 m	4 m	5 m	6 m	7 m	8 m	9 m
Y	1 m	2 m	3 m	4 m	5 m	6 m	7 m	8 m	9 m
Z	1 m	2 m	3 m	4 m	5 m	6 m	7 m	8 m	9 m
Récepteurs (ligne 2)	1	2	3	4	5	6	7	8	9
X	9 m	8 m	7 m	6 m	5 m	4 m	3 m	2 m	1 m
Y	1 m	2 m	3 m	4 m	5 m	6 m	7 m	8 m	9 m
Z	1 m	2 m	3 m	4 m	5 m	6 m	7 m	8 m	9 m
Récepteurs (ligne 3)	1	2	3	4	5	6	7	8	9
X	1 m	2 m	3 m	4 m	5 m	6 m	7 m	8 m	9 m
Y	1 m	2 m	3 m	4 m	5 m	6 m	7 m	8 m	9 m
Z	9 m	8 m	7 m	6 m	5 m	4 m	3 m	2 m	1 m

Les temps de réverbération EDT calculés par Kang, ont été extraits de la figure 2.21, page 57 de [21] et sont donnés le tableau 8.5. La valeur du RT30 en chacun des points est constante et égale à 1.25 s (d'après l'auteur). Le point 5 étant commun aux 3 lignes de calcul, les valeurs de l'EDT sont identiques en ce point. Le résultat du point 3 de la ligne 1 (le récepteur et la source sonore sont confondus) n'est pas donné par Kang, pour des raisons numériques probablement.

**Paramètres de SPPS** L'absorption atmosphérique n'a pas été considérée. La valeur de la célérité du son n'étant pas précisée, nous nous sommes placés dans des conditions atmosphériques normales (température 20°C, 50% d'humidité relative). Les paramètres de simulation SPPS sont les suivants : 1 million de particules sonores, pas de temps de 2 ms pour une durée de simulation de 2 s. Les simulations sont réalisées sur une seule bande de fréquence. De manière à obtenir une puissance sonore de 0 dB pour la source dans la bande de calcul considérée, la puissance globale est fixée à 14.31 dB<sup>6</sup>.

**Résultats** Excepté pour les points proches de la source (points 1 à 4 de la ligne 1), nous obser-

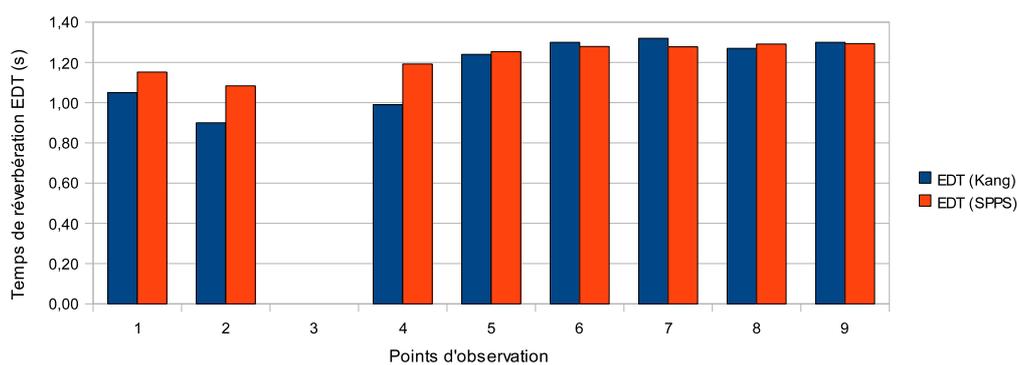
vons que les valeurs des EDT (figure 8.6), RT30 (figure 8.7(a)) et des niveaux sonores (figure 8.7(b)) sont très proches les unes des autres, et globalement uniformes dans la salle. A l'inverse, et comme l'a remarqué Kang dans son ouvrage, proche de la source les valeurs de l'EDT et du RT30 peuvent différer de manière importante. Ceci est le résultat des fluctuations d'amplitude sur les premiers instants des courbes de décroissance (réflexions diffuses précoces), pour des récepteurs très proches de la source. La figure 8.8 illustre ce point et est tout à fait équivalente à celle présentée par Kang ([21, figure 2.22, page 58]). Pour l'EDT, les écarts moyens entre les simulations de Kang et SPPS sont de l'ordre de 0.20 s, 0.0 s et 0.03 s pour les lignes 1, 2 et 3 respectivement.

En ce qui concerne les niveaux sonores, Kang ne précise pas les valeurs obtenues par sa méthode. Il précise simplement que les niveaux sonores dans l'ensemble de la salle cubique sont très proches, avec des variations inférieures à 2 dB, excepté pour les récepteurs proches de la source. La figure 8.7(b) donne les niveaux sonores calculés par le code SPPS. Comme attendu, au niveau de la source (point 1 de la ligne 1), le niveau sonore est plus élevé que dans le reste de la salle. Pour les points les plus éloignés la remarque énoncée par Kang (i.e. une variation maximale de 2 dB) est également vérifiée dans notre cas, puisque les écarts-types sont par exemple inférieurs

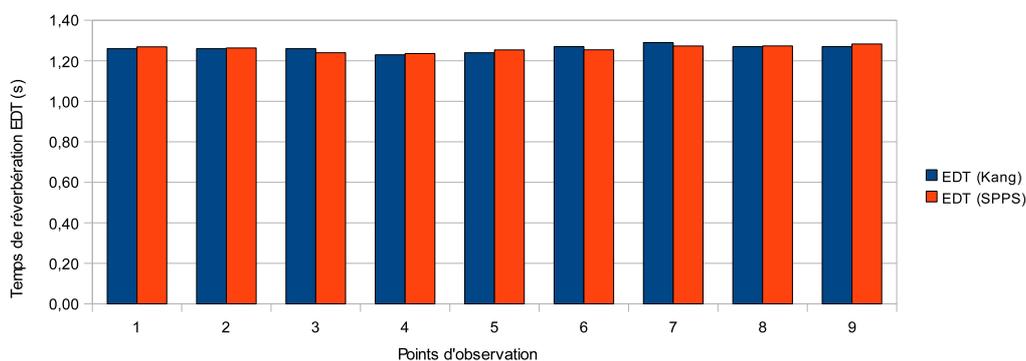
<sup>6</sup>Avec une source de bruit rose, la puissance globale se répartie uniformément sur les 27 bandes de calcul utilisées par le code SPPS, soit 0 dB par bande.

Table 8.5 — EDT calculés par Kang [21, d'après la figure 2.21, page 57].

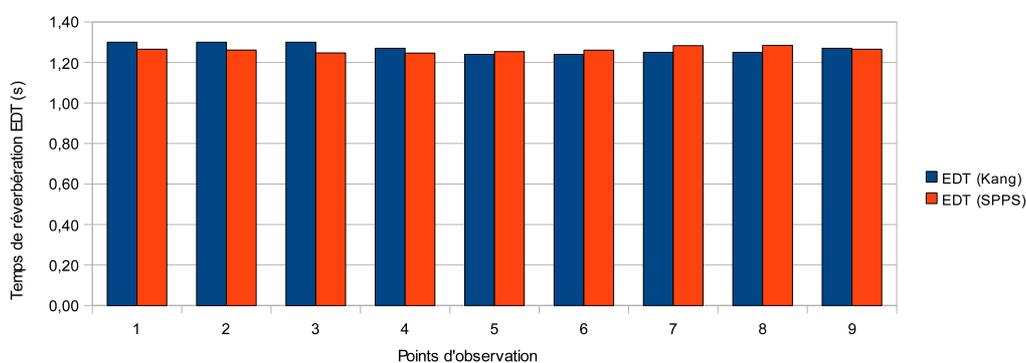
Récepteurs	1	2	3	4	5	6	7	8	9
EDT Ligne 1	1.05 s	0.90 s	—	0.99 s	1.24 s	1.30 s	1.32 s	1.30 s	1.30 s
EDT Ligne 2	1.26 s	1.26 s	1.26 s	1.23 s	1.24 s	1.27 s	1.29 s	1.27 s	1.27 s
EDT Ligne 3	1.30 s	1.30 s	1.30 s	1.27 s	1.24 s	1.24 s	1.25 s	1.25 s	1.27 s



(a) Ligne 1

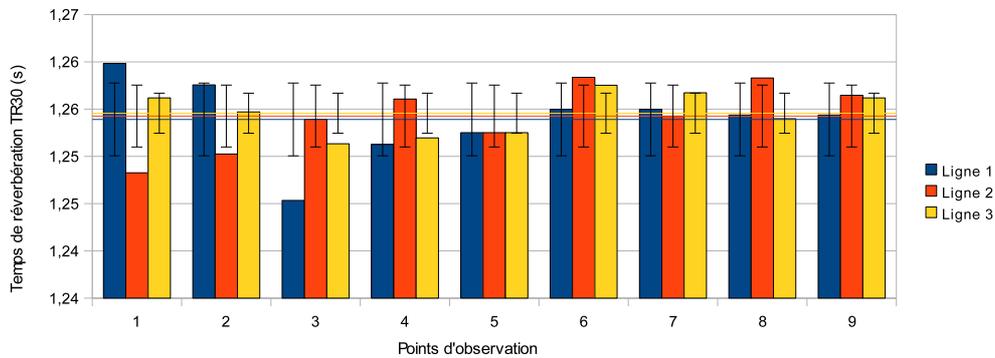


(b) Ligne 2

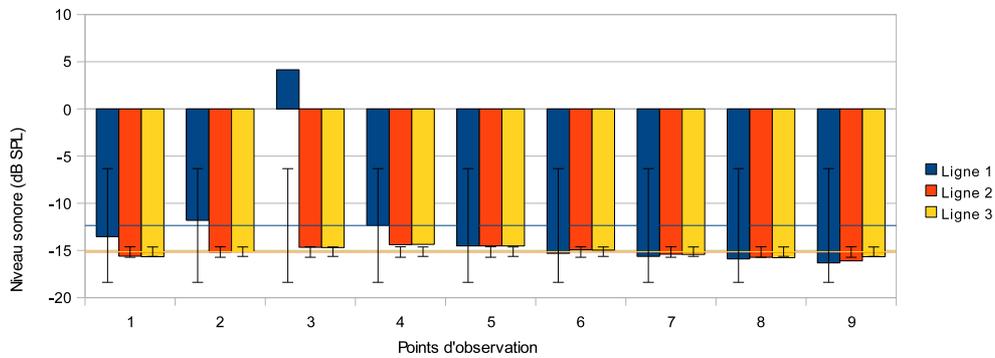


(c) Ligne 3

Figure 8.6 — Comparaison des EDT calculés par SPPS avec les résultats de Kang (2002), pour trois lignes d'observation (lignes 1 à 3). La source et le point 3 de la ligne 1 étant confondus, la valeur de l'EDT n'a pas réellement de sens et n'est donc pas précisée.

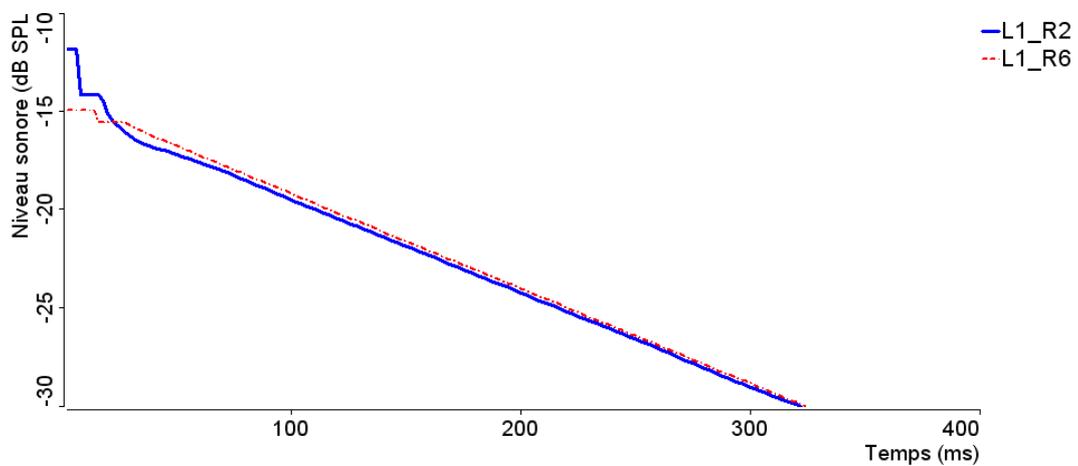


(a) Temps de réverbération RT30



(b) Niveau sonore SPL

**Figure 8.7** — Comparaison des RT30 et SPL calculés par *SPPS* avec les résultats de Kang (2002), pour trois lignes d'observation (lignes 1 à 3). Les écarts-types par rapport à la valeur moyenne de chaque ligne sont également indiqués. Ces écarts-types sont très faibles pour les lignes 2 et 3. Pour la ligne 1, correspondant à des points d'observation proches de la source, l'écart-type est plus fort, notamment pour le niveau sonore.



**Figure 8.8** — Courbes de décroissance aux points 2 (L1\_R2) et 6 (L1\_R6) de la ligne 1. Cette figure peut être comparée directement à la figure 2.22, page 58 de [21].

à 0.60 dB pour les lignes 2 et 3. **En conclusion, le code SPPS est bien validé pour cette configuration.**

## 8.2.2 Couloirs

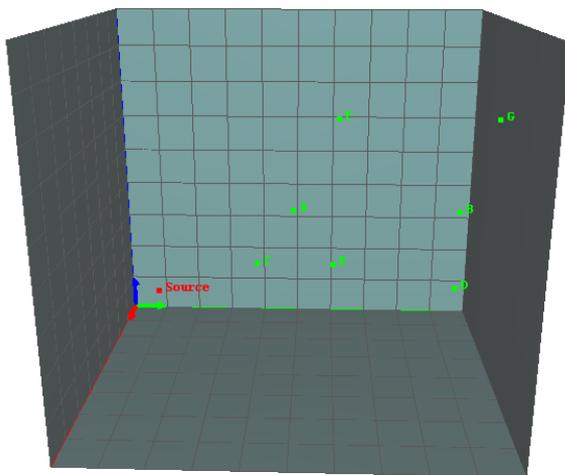
### 8.2.2.1 Comparaison avec les données de Yang (2000) [35] pour un couloir en champ spéculaire

**Données** Dans un article récent [35], Yang et Shield ont proposé un programme informatique (NORMAL) pour le calcul de champs sonores dans des espaces allongés, basé sur le tracé de rayons sonores. Afin de valider ce programme, des simulations ont été réalisées pour une salle quasi-cubique et une longue salle, puis comparées à la théorie classique de l'acoustique des salles (Théorie de Sabine). Les valeurs SPL obtenues par le programme NORMAL et les valeurs calculées par la théorie classique sont proposées pour chacun des points d'observation défini dans [35, tableau 2, page 138] (cf. figure 8.9). De même pour les temps de réverbération EDT et RT30 (cf. [35, tableau 3, page 139]).

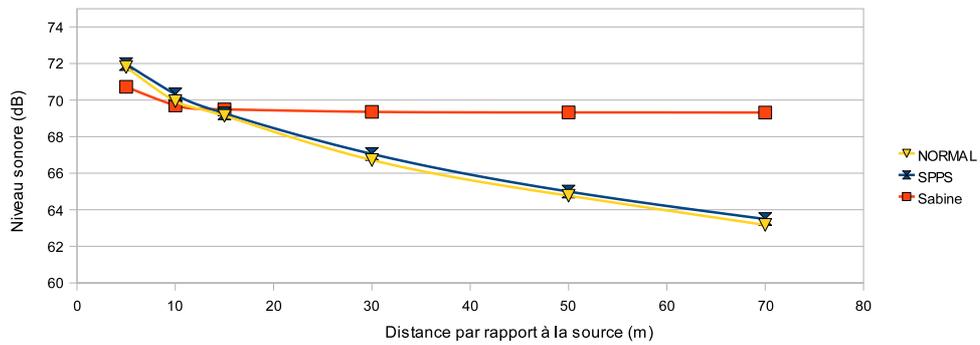
**Paramètres de SPPS** L'absorption atmosphérique n'a pas été considérée. La valeur de la célérité du son n'étant pas précisée, nous nous sommes placés dans des conditions atmosphériques normales (température 20°C, 50% d'humidité relative). Les paramètres de simulation SPPS sont les suivants : 5 million de particules sonores, pas de temps de 2 ms pour une durée de simulation de 1.5 s. Les simulations sont réalisées

sur une seule bande de fréquence. De manière à obtenir une puissance sonore de 90 dB pour la source dans la bande de calcul considérée, la puissance globale est fixée à 104.31 dB.

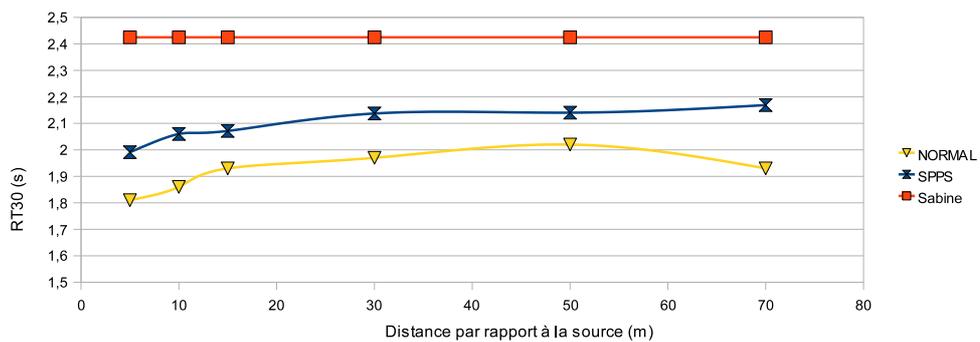
**Résultats** Les comparaisons sont données à la figure 8.10 pour les niveaux sonores SPL et les temps de réverbération RT30 et EDT. Cette figure montre un excellent accord pour les niveaux sonores, les différences étant inférieures à 0.4 dB pour les six points d'observation. Les temps de réverbération RT30 et EDT montrent des comportements similaires entre les deux simulations (NORMAL et SPPS), avec toutefois une surestimation des RT30 pour le code SPPS de l'ordre de 6 à 13%, et une sous-estimation des EDT de l'ordre de 5 à 17%. Il est cependant difficile de préciser laquelle des deux approches est la plus exacte. Là encore, les résultats obtenus par la théorie classique de la réverbération sont très différents des deux autres modèles, ce qui montre une nouvelle fois l'inapplicabilité de cette théorie pour des espaces allongés. **En conclusion, le code SPPS semble bien validé pour cette configuration.**



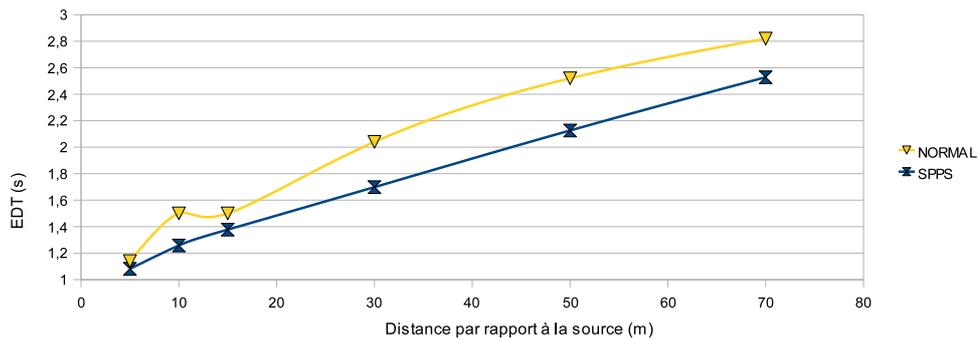
**Figure 8.9** — Représentation sous *I-Simpa* de la géométrie utilisée dans les simulations de Yang, simulée avec le code SPPS.



(a) Niveaux sonores



(b) Temps de réverbération (TR)



(c) Temps de réverbération (EDT)

**Figure 8.10** — Comparaison des simulations numériques SPPS avec les résultats de Yang (2000), pour les (a) niveaux sonores, (b) TR et (c) EDT. Les valeurs obtenues par le code TCR (formules de Sabine) modélisant la théorie classique de la réverbération sont également données pour le niveau sonore et le temps de réverbération.

**Table 8.6** — Caractéristiques de la configuration étudiée dans la référence [35].

Paroi	Sol	Toit	Mur G.	Mur D.	Ext. 1	Ext. 2
Absorption	0.1					
Diffraction $\delta$	$\delta = 0.00$					
Réflexion	Spéculaire					

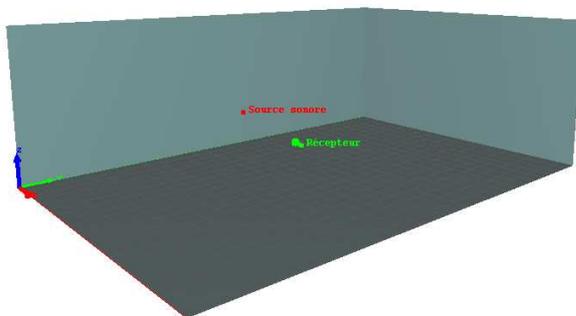
Position	X	Y	Z
Salle	8 m	10 m	9 m
Source (90 dB)	1 m	1 m	1 m

Récepteurs	A	B	C	D	E	F	G
X	4.0 m	4.0 m	2.5 m	2.5 m	2.5 m	7.5 m	7.5 m
Y	5.0 m	9.0 m	4.0 m	9.0 m	6.0 m	6.0 m	9.0 m
Z	4.5 m	4.5 m	2.5 m	2.5 m	2.5 m	7.5 m	7.5 m

## 8.2.3 Locaux plats

### 8.2.3.1 Comparaison avec les données de Korany (2001) [22]

**Données** Korany *et al.* [22] ont proposé un nouveau modèle numérique pour la prédiction des champs sonores dans des salles, permettant de prendre en compte les phénomènes de diffusion sur les parois du domaine. Cette méthode est basée sur les modèles classiques de tracé de rayons et de sources-images, mais intègre en parallèle un modèle de radiosité pour simuler les réflexions diffuses. A travers cette étude, Korany *et al.* ont réalisé un certain nombre de simulations pour une salle rectangulaire (figure 8.11), en faisant varier le coefficient de diffraction  $\delta$  des parois. Ces résultats ont ensuite été comparés aux approches classiques de Sabine et d'Eyring, ainsi qu'à celle de Kuttruff [24] (*cf.* §8.2.3.2). Dans l'article de référence, les auteurs ont étudiés une salle rectangu-



**Figure 8.11** — Représentation sous *I-Simpa* de la géométrie utilisée dans les simulations de Korany (2001) pour des locaux plats.

laire ( $30 \times 20 \times 10$  m) caractérisée par une distribution non-uniforme d'absorption et plusieurs conditions de diffusion uniformes (tableau 8.7), ainsi qu'une salle rectangulaire  $30 \times 20 \times 10$  m totalement diffuse, caractérisée par une absorption totale au niveau du sol et une absorption nulle sur le reste des parois. Bien que leur modèle permette probablement de calculer des niveaux sonores, Korany *et al.* ne présentent que des temps de réverbération, pour différentes valeurs de  $\delta$ . Les temps de réverbération calculés par la méthode de Korany sont donnés dans le tableau 2 de la page 886 de l'article de référence [22], pour chaque valeur du coefficient de diffraction, et sont également comparés aux valeurs obtenues par les formules de Sabine, d'Eyring et de Kuttruff (si applicable).

**Paramètres de SPPS** Deux séries de simulations ont été réalisées. La première correspond aux données du tableau 8.7 et la seconde aux données du tableau 8.8. L'absorption atmosphérique n'a pas été considérée. La valeur de la célérité du son n'étant pas précisée, nous nous sommes placés dans des conditions atmosphériques normales (température 20°C, 50% d'humidité relative). Les paramètres de simulation SPPS sont les suivants : 1 million de particules sonores, pas de temps de 2 ms pour une durée de simulation de 1.5 s. Pour la première configuration, les simulations sont réalisées sur plusieurs bandes de fréquence, chaque bande étant associée à un coefficient de diffraction. Pour la seconde configuration, les simulations sont réalisées sur une seule bande de fréquence.

**Résultats** La comparaison des résultats est présentée dans le tableau 8.9 pour les deux configurations de mesure, et à la figure 8.12 (configuration 1 uniquement). Les résultats du code SPPS suivent la même

**Table 8.7** — Caractéristiques de la configuration étudiée par [22] (local plat en champ mixte). Configuration géométrique 1.

Paroi	Sol	Plafond	Mur G.	Mur D.	Ext. 1	Ext. 2
Absorption $\alpha$	0.9	0.1	0.3	0.3	0.6	0.6
Diffraction $\delta$	$\delta = 0 : 0.2 : 1$					
Réflexion	Mixte (spéculaire à diffus)					

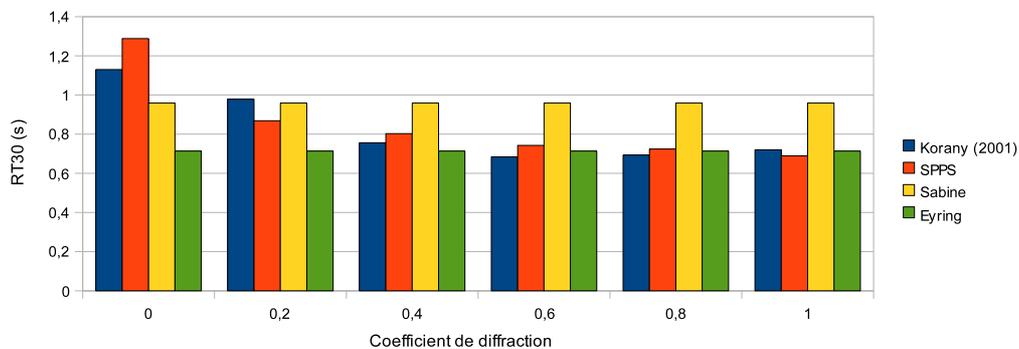
Position	X	Y	Z
Salle	20 m	30 m	10 m
Source (0 dB)	2.5 m	15 m	3 m
Points d'observation	15 m	10 m	4 m

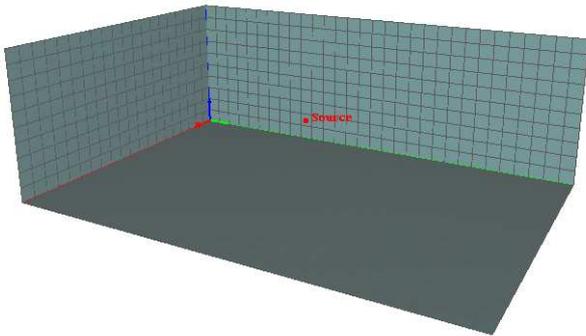
**Table 8.8** — Caractéristiques de la configuration étudiée par [22] (local plat en champ diffus). La dimension de la salle, la position de la source et des récepteurs sont identiques à celles présentées dans le tableau 8.7. Configuration géométrique 2.

Paroi	Sol	Plafond	Mur G.	Mur D.	Ext. 1	Ext. 2
Absorption $\alpha$	1.0	0	0	0	0	0
Diffraction $\delta$	$\delta = 1$					
Réflexion	Diffus (Lambert)					

**Table 8.9** — Comparaison des temps de réverbération ( $RT_{30}$ ) obtenus par le modèle de Korany<sup>7</sup>, les formules de Sabine, d'Eyring et Kuttruff (si applicable) avec le code SPPS, pour des locaux plats caractérisés une absorption non-uniforme, et pour différentes valeurs du coefficient de diffraction  $\delta$ . Les résultats pour les deux configurations de mesure sont présentés dans le même tableau.

RT30 (s)	Coefficient de diffraction $\delta$						
	Config. 1						Config. 2
	0.0	0.2	0.4	0.6	0.8	1.0	1.0
Korany (2001)	1.13	0.98	0.76	0.68	0.69	0.72	1.09
SPPS	1.29	0.87	0.80	0.74	0.72	0.69	1.11
Sabine			0.96				1.63
Eyring			0.71				1.40
Kuttruff (1997)	–	–	–	–	–	–	1.22

**Figure 8.12** — Comparaison des temps de réverbération obtenus par le code SPPS avec les résultats de Korany (2001), les formules de Sabine, d'Eyring et de Kuttruff (1997). Cf. tableau 8.9.



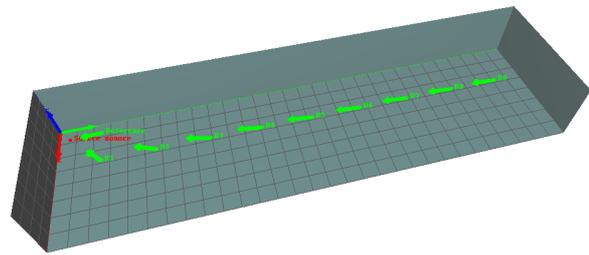
**Figure 8.13** — Représentation sous *I-Simpa* de la géométrie utilisée dans les simulations de Kuttruff (1997), pour des locaux plats.

tendance que le modèle de Korany. Pour la configuration 1, les écarts sont légèrement plus importants quand le coefficient de diffraction est plus petit ( $d = 0.0$  et  $0.2$ , *i.e.* les surfaces sont principalement spéculaires), mais sont globalement assez faibles (entre 14% et 4%). Pour la configuration 2, la valeur obtenue avec le code SPPS est très similaire à celle des modèles de Korany et Kuttruff. Comme attendu, les formules de Sabine et d'Eyring ne sont pas adaptées à ce type de local allongé. La relation d'Eyring donne toutefois des résultats satisfaisant quand le coefficient de diffraction est plus élevé (parois fortement diffuses). **En conclusion, le code SPPS est bien validé pour ces deux configurations.**

### 8.2.3.2 Comparaison avec les données de Kuttruff (1997) [24] en champ diffus

**Données** Ce paragraphe fait suite à la section 8.2.1.2, présentant une comparaison du code de calcul avec le modèle de radiosité de Kuttruff pour une salle cubique. Nous reprenons ici la même procédure pour des locaux plats (configurations K4 à K1, figure 8.13 et tableau 8.10). Les paramètres de simulation sont identiques au cas des salles cubiques.

**Résultats** Les temps de réverbération calculés par les deux méthodes ainsi que par les formules classiques de Sabine et d'Eyring sont donnés dans le tableau 8.3 (configurations K4 à K1). Les locaux plats étant caractérisés par des dimensions non-homogènes, l'utilisation des formules de Sabine et d'Eyring n'est en principe pas adaptée. Comme attendu, on observe donc que les résultats obtenus par ces formules sont différents de ceux obtenus par le modèle de Kuttruff et par le code SPPS. Pour les configurations K4 à K2, le modèle de Kuttruff et le



**Figure 8.14** — Représentation sous *I-Simpa* de la géométrie utilisée par Ondet (1989), simulée avec le code SPPS : local plat vide.

code SPPS donne des résultats similaires, avec des écarts de 1.47%, 1.16% et 3.15% respectivement. Par contre, la configuration K1 donne des résultats très différents (écart de 21.67%), ceux du code SPPS donnant des valeurs plus proches des formules classiques (mais en principe pas adaptée). Hormis ce cas particulier, pour lequel il est difficile de conclure, **le code SPPS est bien validé pour les trois autres configurations.**

### 8.2.3.3 Comparaison avec les données de Ondet (1989) [29] en champ spéculaire

**Données** Dans un article plus ancien, Ondet et Barbry ont réalisé une expérimentation dans une salle rectangulaire ( $8 \times 30 \times 3.85$  m) caractérisée par des parois réfléchissantes (réflexions spéculaires), afin de valider un modèle numérique basé sur les rayons sonores (RAYSCAT), et prenant en compte la présence d'objets diffractants (figure 8.14). Plusieurs configurations ont testées, mais nous n'avons retenu dans cette partie que la configuration de la salle « vide » (configuration (a) de [29]). Les caractéristiques de la salle sont données dans le tableau 8.11.

**Paramètres de SPPS** Nous avons réalisé les simulations pour la bande d'octave 2000 Hz, les résultats expérimentaux étant donnés à cette fréquence. L'absorption atmosphérique a été considérée ( $m = 0.0014$  Np/m). Les paramètres de simulation SPPS sont les suivants : 1 million de particules sonores, pas de temps de 5 ms pour une durée de simulation de 1.5 s.

**Résultats** Les résultats dans la bande d'octave 2000 Hz sont présentés sous la forme d'une courbe de niveau en fonction de la distance source-récepteurs, normalisée par rapport à la puissance de la source ( $SPL - L_W$ ,  $L_W = 102.9$  dB). Les valeurs

**Table 8.10** — Caractéristiques des locaux plats étudiés par [24].

Paroi	Sol ( $xOy$ )	Plafond ( $xOy$ )	Mur G. ( $yOz$ )	Mur D. ( $yOz$ )	Ext. 1 ( $xOz$ )	Ext. 2 ( $xOz$ )
Absorption (Conf. K4)	$\alpha = \bar{\alpha} = 3/11$					
Absorption (Conf. K3)	1.0	0.0	0.0	0.0	0.0	0.0
Absorption (Conf. K2)	0.0	0.0	0.0	1.0	0.0	0.0
Absorption (Conf. K1)	0.0	0.0	0.0	0.0	0.0	1.0
Diffraction $\delta$	$\delta = 1.00$					
Réflexion	Lambert					

Position	X	Y	Z
Salle	20 m	30 m	10 m
Source (0 dB)	10 m	15 m	5 m
Points d'observation	—	—	—

**Table 8.11** — Caractéristiques de la configuration étudiée par [29] pour un local plat vide.

Paroi	Sol	Plafond	Mur G.	Mur D.	Ext. 1	Ext. 2
Absorption $\alpha$	0.05	0.15	0.10	0.10	0.10	0.10
Diffusion $\delta$	$\delta = 0.00$					
Réflexion	Spéculaire					

Position	X	Y	Z
Salle	8 m	30 m	3.85 m
Source (102.9 dB)	1.5 m	1.0 m	0.85 m
Position de référence	2.0 m	3.0 m	1.5 m
Points d'observation	4.0 m	(3 : 3 : 27) m	1.5 m

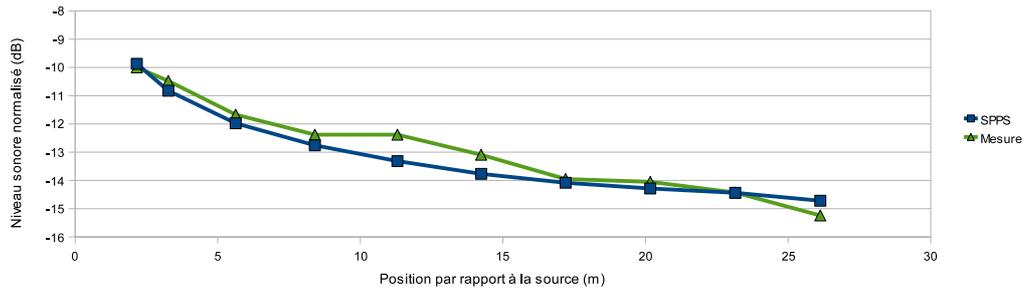
suivantes<sup>8</sup> ont été relevées sur la figure 11(a) page 794 de l'article de référence [29], complétées par les valeurs obtenues avec le code SPPS :

La figure 8.15 présente une comparaison des niveaux de pression mesurés et calculés par le code SPPS (données brutes, sans recombinaison et compensation), ainsi qu'avec la théorie classique de la réverbération. Les incertitudes des mesures, estimées par Ondet (de l'ordre de 1 dB), sont également représentées. Cette figure montre une nouvelle fois un très bon comportement de SPPS, les écarts avec les mesures variant entre 0.1 et 1.0 dB (écart moyen de 0.37 dB). **En conclusion, le code SPPS est bien validé pour cette configuration.**

<sup>8</sup>Nous n'avons repris que les données expérimentales. Les résultats des simulations numériques avec le programme RAYSCAT n'ont pas été repris.

**Table 8.12** — Résultats expérimentaux obtenus par Ondet et simulations numériques SPPS. (\*) Position de référence.

Position Y (m)	3.0*	3.0	6.0	9.0	12.0	15.0	18.0	21.0	24.0	27.0
Mesure (dB)	-10.0	-10.5	-11.7	-12.4	-12.4	-13.1	-14.0	-14.0	-14.4	-15.2
SPPS (dB)	-9.9	-10.8	-12.0	-12.8	-13.3	-13.8	-14.1	-14.3	-14.4	-14.7

**Figure 8.15** — Comparaison des résultats de SPPS avec les mesures réalisées par Ondet et Barbry [29].

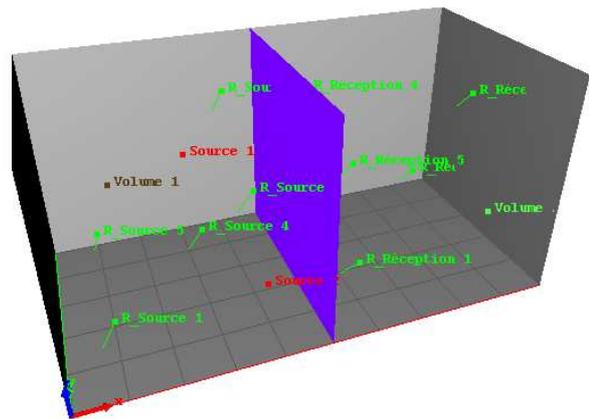
## 8.2.4 Locaux couplés

Dans cette partie nous présentons une validation du code SPPS pour des locaux couplés, par comparaison avec la théorie classique des locaux couplés [24]. Plusieurs configurations géométriques sont considérées (deux locaux homogènes et non-homogènes en volume) et plusieurs conditions de transmission à travers la cloison de couplage (cloison avec ou sans porte, les deux surfaces pouvant être définies par des coefficients d'affaiblissement et d'absorption différents).

### 8.2.4.1 Salles de même dimension couplées par une cloison : comparaison avec la théorie classique des locaux couplés

**Données** Nous nous intéressons ici au cas de deux salles de même dimension (5 m de large, 5 m de long et 5 m de hauteur) séparée par une cloison sans épaisseur (figure 8.16) et caractérisée par un coefficient d'affaiblissement  $R$  variant de 0 dB (transmission totale, *i.e.* sans cloison) à 30 dB. L'absorption acoustique de chacune des deux salles est uniforme et égale à 0.1. L'ensemble des parois est 100% diffus.

Deux sources omnidirectionnelles de puissance 100 W chacune, rayonnent simultanément et sont localisée en  $(3, 3.75, 2.75)$  et en  $(4, 1, 1)$ . Dans le cadre de cette étude, les niveaux sonores sont calculés pour 5 récepteurs dans la salle source (notés Récepteur\_ $S_x$ ,  $x$  allant de 1 à 5) et pour 5 récepteurs dans la salle de réception (notés Récepteur\_ $R_x$ ,  $x$  allant de 1 à 5) (tableau 8.13).

**Figure 8.16** — Représentation sous I-S'Impa de la géométrie utilisée, simulée avec le code SPPS, pour deux salles de même dimension couplées par une cloison sans épaisseur, avec transmission.

**Paramètres de SPPS** L'absorption atmosphérique n'a pas été considérée dans les simulations. Pour le calcul de la célérité du son, nous nous sommes placés dans des conditions atmosphériques normales (température 20°C, 50% d'humidité relative). Les paramètres de simulation SPPS sont les suivants :  $10^6$  particules sonores, pas de temps de 10 ms pour une durée de simulation de 1.5 s, mode de calcul « Aléatoire ».

**Résultats** Une moyenne linéaire permet de remonter au niveau sonore moyen par salle ( $L_S$  et  $L_R$  res-

**Table 8.13** — Configurations de deux locaux couplés : position des récepteurs.

Récepteurs	R1	R2	R3	R4	R5	S1	S2	S3	S4	S5
X	6 m	9 m	9 m	6 m	6.5 m	1 m	4 m	3.75 m	3 m	1 m
Y	1 m	4 m	2 m	4 m	2.5 m	1 m	4 m	1 m	2.25 m	4 m
Z	1 m	1 m	4 m	3.75 m	2.5 m	1 m	4 m	3.25 m	1.75 m	1 m

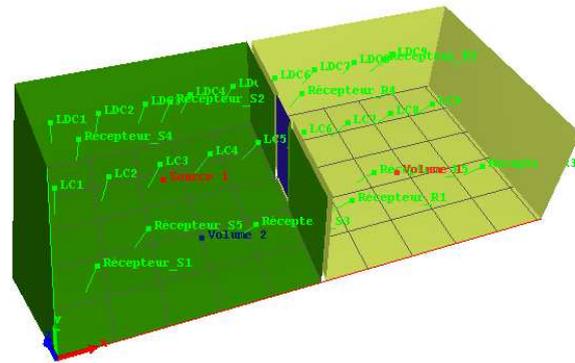
pectivement pour la salle source et la salle de réception) pour ensuite calculer la différence de niveau sonore entre salles :  $\Delta = L_S - L_R$ . Cette dernière valeur peut alors être comparée avec celle obtenue en appliquant la théorie classique des locaux couplés. Les résultats sont détaillés au tableau 8.14 et montre un très bon accord pour les configurations avec un affaiblissement de 10, 20 et 30 dB. Le cas  $R = 0$  dB est particulier puisqu'il correspond à une transmission totale (*i.e.* sans cloison), ce qui ne peut normalement pas être modélisé par la théorie classique des locaux couplés. Les résultats obtenus par SPPS sont donc différents dans ce cas, mais *a priori* plus proche de la réalité. **En résumé, le code SPPS semble bien validé pour cette configuration.**

#### 8.2.4.2 Salles de même dimension couplées par une porte : comparaison avec la théorie classique des locaux couplés

**Données** Nous nous intéressons ici au cas de deux salles de même dimension (5 m de large, 5 m de long et 2.5 m de hauteur) séparée par une cloison d'épaisseur 10 cm (figure 8.17), contenant une porte centrée (0.90 m de large, 2.5 m de haut, sans épaisseur au centre de la cloison). Seule la porte est caractérisée par un coefficient d'affaiblissement. L'absorption acoustique de chacune des deux salles (y compris la cloison de séparation) est uniforme, mais peut-être différente pour chaque salle. La porte est définie par un coefficient d'absorption différent des deux salles. L'ensemble des parois est 100% diffus. Au total, 9 configurations ont été testées.

Une source omnidirectionnelle de puissance 100 W chacune est localisée au centre de la première salle (dite salle « source »), en (2.5, 2.5, 1.25). Dans le cadre de cette étude, les niveaux sonores sont calculés pour 5 récepteurs dans la salle source (notés Récepteur\_Sx, x allant de 1 à 5) et pour 5 récepteurs dans la salle de réception (notés Récepteur\_Rx, x allant de 1 à 5) (tableau 8.15).

**Paramètres de SPPS** L'absorption atmosphérique n'a pas été considérée dans les simulations. Pour le calcul de la célérité du son, nous nous sommes placés dans des conditions atmosphériques normales



**Figure 8.17** — Représentation sous I-Simpa de la géométrie utilisée, simulée avec le code SPPS, pour deux salles de même dimension couplées par une cloison (avec épaisseur, sans transmission) contenant une porte sans épaisseur mais avec transmission.

(température 20°C, 50% d'humidité relative). Les paramètres de simulation SPPS sont les suivants :  $10^6$  particules sonores, pas de temps de 10 ms pour une durée de simulation de 1.5 s. Les modes de calcul « Aléatoire » (9 configurations) et « Énergétique » (1 seule configuration) ont été considérés<sup>9</sup>. Dans ce dernier cas, la limite d'extinction des particules a été fixée à  $r = 5$  (soit  $10 \log 10^{-5} = -50$  dB)

**Résultats** Comme au paragraphe 8.2.4.1, nous nous sommes intéressés à la différence de niveau sonore  $\Delta = L_S - L_R$  entre les deux salles, comparés avec les valeurs obtenues en appliquant la théorie classique des locaux couplés. Les résultats sont détaillés au tableau 8.17.

Les résultats montrent un très bon accord entre le code SPPS et la théorie classique tant que les absorptions moyennes dans les salles sont faibles et peut différentes. Lorsque une des absorptions devient grande et que la différence d'absorption entre deux salles s'accroît, la différence entre les deux modèles s'accroît très légèrement (mais reste négligeable), ce qui est toutefois attendu puisque la théo-

<sup>9</sup>Le mode de calcul « Énergétique » étant environ 10 fois plus long, il n'a été appliqué que sur une seule configuration

**Table 8.14** — Comparaison des différences de niveaux sonores  $\Delta$  calculés par le code SPPS et par la théorie classique des locaux couplés, pour deux salles couplées par une cloison, pour plusieurs valeurs de transmission acoustique (un affaiblissement  $R$  de 0 dB correspond à une transmission totale, i.e. une porte ouverte).

Coefficient d'absorption			Affaib. $R$ (dB)	SPPS (aléatoire)	Théorie classique
Salle source	Salle de réception	Cloison			
0.1	0.1	1.0	0	0.82	1.76
0.1	0.1	0.1	10	7.26	7.78
0.1	0.1	0.1	20	17.31	17.78
0.1	0.1	0.1	30	27.10	27.78

**Table 8.15** — Configurations de deux locaux couplés : position des récepteurs.

Récepteurs	R1	R2	R3	R4	R5	S1	S2	S3	S4	S5
X	6 m	8 m	9 m	6 m	7 m	1 m	3 m	4 m	1 m	2 m
Y	1 m	4 m	1 m	4 m	2 m	1 m	4 m	1 m	4 m	2 m
Z	1 m	2 m	1 m	1.5 m	0.5 m	1 m	2 m	1 m	1.5 m	0.5 m

**Table 8.16** — Comparaison des différences de niveaux sonores  $\Delta$  calculés par le code SPPS et par la théorie classique des locaux couplés, pour deux salles couplées par une ouverture, pour plusieurs configurations d'absorption et de transmission acoustique (un affaiblissement  $R$  de 0 dB correspond à une transmission totale, i.e. une porte ouverte).

Coefficient d'absorption			Affaib. $R$ (dB)	SPPS (énergétique)	SPPS (aléatoire)	Théorie classique
Salle source	Salle de réception	Porte				
0.1	0.1	1.0	0	–	7.58	7.29
0.1	0.3	1.0	0	–	11.81	11.49
0.3	0.3	1.0	0	11.57	11.65	11.49
0.3	0.1	1.0	0	–	7.28	7.29
0.05	0.5	1.0	0	–	14.09	13.58
0.5	0.05	1.0	0	–	4.59	5.02
0.1	0.3	0.1	20	–	31.32	31.20
0.1	0.3	0.1	10	–	21.20	21.20
0.1	0.3	0.2	10	–	21.28	21.23

rie classique est fondée sur les hypothèses de champ diffus, qui, dans ce cas, ne sont plus vérifiées. **En résumé, le code SPPS est bien validé pour cette configuration.**

#### 8.2.4.3 Salles de dimension différentes couplées par une porte : comparaison avec la théorie classique des locaux couplés

**Données** Nous nous intéressons ici au cas de deux salles de dimension surface au sol (5 m de large par 5 m de long) mais de hauteur différentes (2.5 m pour la salle source ; 5 m pour la salle de réception) séparée par une cloison d'épaisseur 10 cm, contenant une porte centrée (0.90 m de large, 2.5 m de haut, sans épaisseur, au centre de la cloison). L'ensemble des parois est 100% diffus. La position des sources et

des récepteurs est identique au cas présenté au paragraphe 8.2.4.2 (tableau 8.15). Au total, 4 configurations d'absorption et de transmission ont été testées.

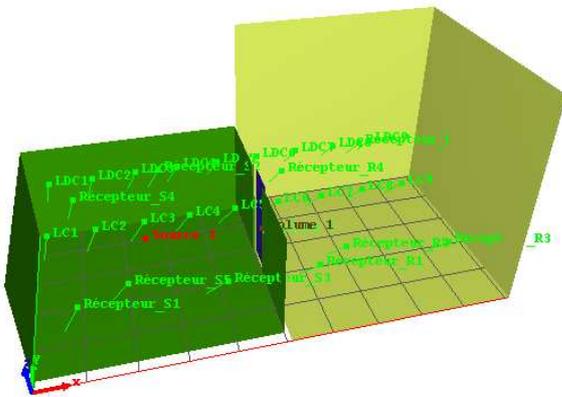
**Paramètres de SPPS** Les mêmes paramètres que ceux utilisés au paragraphe 8.2.4.1 ont été utilisés.

**Résultats** Comme au paragraphe 8.2.4.1, nous nous sommes intéressés à la différence de niveau sonore  $\Delta = L_S - L_R$  entre les deux salles, comparés avec les valeurs obtenues en appliquant la théorie classique des locaux couplés. Les résultats sont détaillés au tableau 8.17.

Là encore, les résultats montrent un très bon accord entre le code SPPS et la théorie classique. **À nouveau, le code SPPS est bien validé pour cette configuration.**

**Table 8.17** — Comparaison des différences de niveaux sonores  $\Delta$  calculés par le code SPPS et par la théorie classique des locaux couplés, pour deux salles de dimensions différentes, couplées par une ouverture, pour plusieurs configurations d'absorption et de transmission acoustique (un affaiblissement  $R$  de 0 dB correspond à une transmission totale, i.e. une porte ouverte).

Coefficient d'absorption			Affaib. $R$ (dB)	SPPS	Théorie
Salle source	Salle de réception	Porte	Porte	(aléatoire)	classique
0.1	0.1	1.0	0	8.82	8.80
0.1	0.3	0.1	0	12.76	13.17
0.1	0.3	0.1	20	32.00	32.98
0.1	0.3	0.2	10	22.29	23.00



**Figure 8.18** — Représentation sous *I-Simpa* de la géométrie utilisée, simulée avec le code SPPS, pour deux salles de dimensions différentes couplées par une cloison (avec épaisseur, sans transmission) contenant une porte sans épaisseur mais avec transmission.

## 8.3 Validation en acoustique industrielle

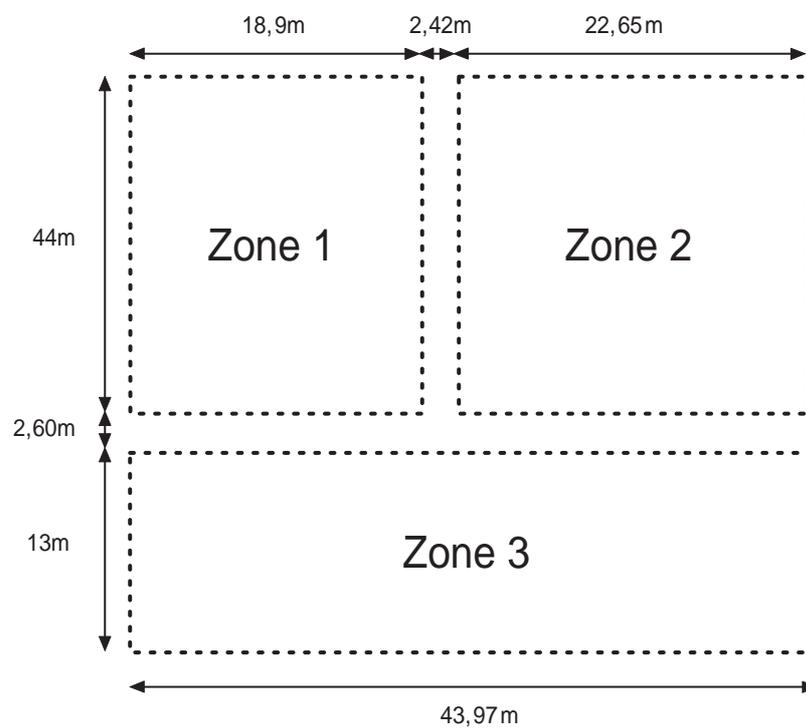
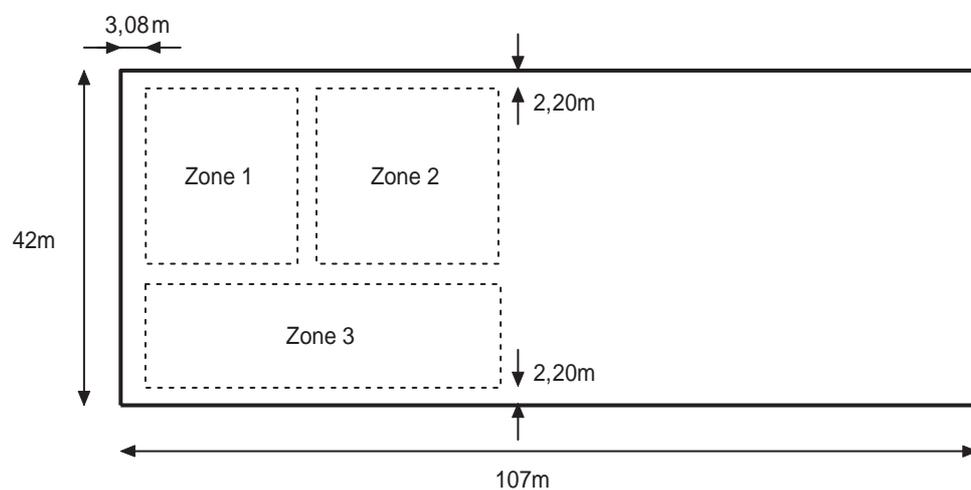
### 8.3.1 Comparaison avec les données de Ondet (1984,1988) [31] pour un local industriel

**Données** Dans un article de 1988 [31], Ondet *et al* ont présenté des mesures du champ sonore dans un hall d'usine présentant différentes zones d'encombrement. Le local plat de dimension  $107 \times 42 \times 5$  m est défini par trois zones d'encombrement de même caractéristique ( $\lambda_c = 5$  m,  $\alpha_c = 0.07$ ), localisées sur les 51 premiers mètres de la salle (figures 8.19 et 8.20, tableau 8.18). Les parois du local sont supposées produire des réflexions spéculaires. L'absorption acoustique est de 0.07 sur les murs latéraux, 0.05 au sol et 0.1 au plafond.

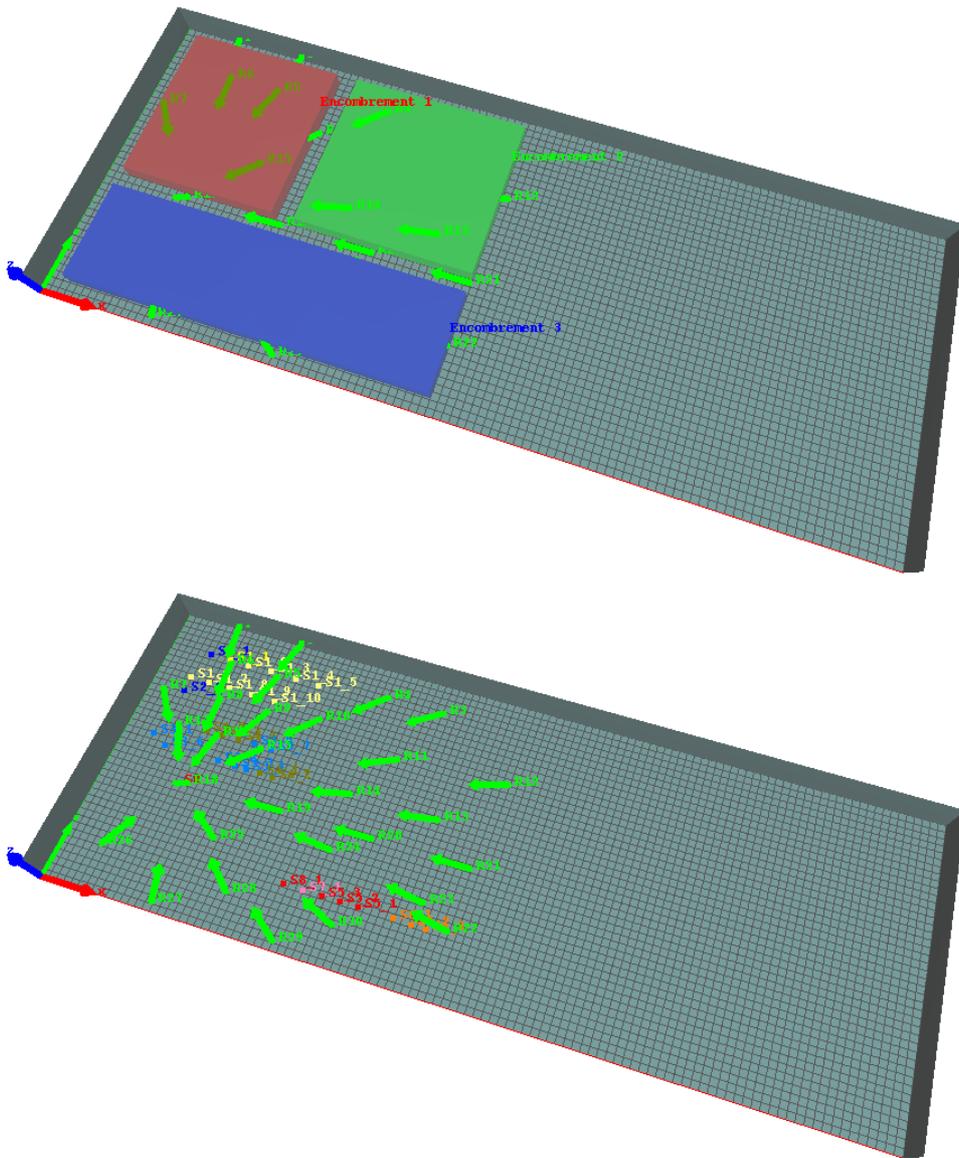
Trente points de mesure ont été considérés par les auteurs, répartis dans les trois zones d'encombrement et dans les allées. La position exacte des points de mesure est donnée dans le tableau 2 (page 60) du rapport [30]. Cinq configurations de mesure ont été considérées, la première en utilisant une source sonore maîtrisée, les 4 suivantes avec plusieurs configurations de sources sonores actives (machine en fonctionnement). Faute d'information suffisante, excepté pour la source maîtrisée  $S_0$ , la position des sources sonores (machines réelle) a été extraite de la figure 2 de la référence [31]. La position des sources et le mode de fonctionnement pour chaque configuration sont donnés dans le tableau 8.19. Le niveau de puissance des sources à 1000 Hz est donné dans le tableau 3 du rapport [30]. Les résultats des mesures dans la bande d'octave 1000 Hz sont donnés dans le tableau 4 (page 65) de [30] pour les 5 configurations de mesure.

**Paramètres de SPPS** L'absorption atmosphérique a été considérée. La valeur de la célérité du son n'étant pas précisée, nous nous sommes placés dans des conditions atmosphériques normales (température 20°C, 50% d'humidité relative). Les paramètres de simulation SPPS sont les suivants : 1 million de particules sonores, pas de temps de 3 ms pour une durée de simulation de 1.5 s. Les simulations sont réalisées sur la seule bande de fréquence 1000 Hz. Les simulations ont été réalisées pour les 5 configurations de sources actives.

**Résultats** Les figures 8.21(a) à 8.21(e) présentent la comparaison des résultats expérimentaux obtenus par Ondet et Barbry, avec les simulations numériques par le programme SPPS, pour les 5 configurations de mesure. Ces figures mettent en évidence une surestimation systématique d'amplitude par le code SPPS, de l'ordre de 2.7 à 4.3 dB en moyenne. Si on



**Figure 8.19** — Configuration du hall d'usine ayant servi aux mesures de Ondet et al [31]. Faute de précision suffisante dans les articles, les dimensions et la position de chacune de ces zones ont été mesurées sur la figure page 57 de [30] (cf. tableau 8.18).



**Figure 8.20** — Représentation des 3 zones d'encombement, des sources sonores et des récepteurs . Configuration étudiée par Ondet et al [31] : local plat de dimension  $107 \times 42 \times 5$  m, défini par trois zones d'encombement de même caractéristique ( $\lambda_c = 5$  m,  $\alpha_c = 0.07$ ), 30 récepteurs et 22 sources sonores.

Zone	$x_i$ (m)	$x_s$ (m)	$y_i$ (m)	$y_s$ (m)	$z_i$ (m)	$z_s$ (m)	$\lambda_c$ (m)	$\alpha_c$
1	3.08	21.98	17.08	39.8	0	2.5	5	0.07
2	24.4	47.05	17.08	39.8	0	2.5	5	0.07
3	3.08	47.05	2.20	15.2	0	2.5	5	0.07

**Table 8.18** — Localisation et caractéristiques des zones d'encombement du local étudié par Ondet [31]. Faute d'information suffisante, les positions  $x_i, x_s \dots$  ont été extraites de la figure de la page 57 de la référence [30].

**Table 8.19** — Position des sources sonores et configuration de mesure. D'après [30, tableau 3, page 63]. Le symbole ✖ désigne une source en fonctionnement.

Position des sources				Sources actives				
N°	<i>x</i>	<i>y</i>	<i>z</i>	Conf. 1	Conf. 2	Conf. 3	Conf. 4	Conf. 5
S <sub>0</sub>	10.5	17.0	0.1	✖				
S <sub>1,1</sub>	9.0	36.5	0.5		✖	✖		✖
S <sub>1,2</sub>	11.9	36.5	0.5		✖	✖		✖
S <sub>1,3</sub>	15.3	36.5	0.5		✖	✖		✖
S <sub>1,4</sub>	18.9	36.5	0.5		✖	✖		✖
S <sub>1,5</sub>	22.4	36.5	0.5		✖	✖		✖
S <sub>1,6</sub>	5.1	32.3	0.5		✖	✖		✖
S <sub>1,7</sub>	7.8	32.3	0.5		✖	✖		✖
S <sub>1,8</sub>	10.8	32.3	0.5		✖	✖		✖
S <sub>1,9</sub>	14.3	32.3	0.5		✖	✖		✖
S <sub>1,10</sub>	17.6	32.3	0.5		✖	✖		✖
S <sub>2,1</sub>	6.2	36.5	0.5		✖	✖		✖
S <sub>2,2</sub>	5.1	30.0	0.5		✖	✖		✖
S <sub>3,1</sub>	4.3	22.8	0.5			✖		✖
S <sub>3,2</sub>	14.3	21.5	0.5			✖		✖
S <sub>3,3</sub>	16.7	21.5	0.5			✖		✖
S <sub>3,4</sub>	18.4	21.5	0.5			✖		✖
S <sub>3,5</sub>	17.8	25.3	0.5			✖		✖
S <sub>3,6</sub>	6.5	21.5	0.5					✖
S <sub>3,7</sub>	20.3	25.3	0.5					✖
S <sub>4,1</sub>	20.3	21.5	0.5			✖		✖
S <sub>4,2</sub>	22.1	21.5	0.5			✖		✖
S <sub>4,3</sub>	13.2	25.3	0.5			✖		✖
S <sub>4,4</sub>	15.4	25.3	0.5			✖		✖
S <sub>4,5</sub>	10.8	25.3	0.5			✖		✖
S <sub>5,1</sub>	39.4	8.3	0.5				✖	✖
S <sub>5,2</sub>	37.0	8.3	0.5				✖	✖
S <sub>5,3</sub>	34.6	8.3	0.5				✖	✖
S <sub>6,1</sub>	44.0	8.3	0.5				✖	✖
S <sub>6,2</sub>	46.4	8.3	0.5				✖	✖
S <sub>6,3</sub>	48.3	8.3	0.5				✖	✖
S <sub>7,1</sub>	32.1	8.3	0.5				✖	✖
S <sub>8,1</sub>	29.4	8.3	0.5				✖	✖

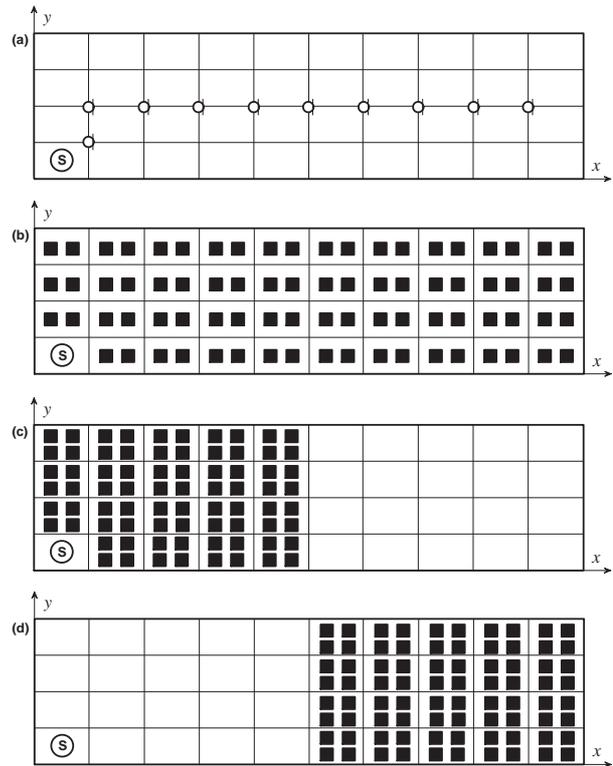
écarter une mauvaise<sup>10</sup> modélisation de l'encombrement dans le code SPPS, les raisons de ce décalage sont multiples : incertitudes sur la position et la taille des zones d'encombrement, incertitudes sur la position exacte des sources sonores (position estimée graphiquement avec une mauvaise précision), incertitudes sur les paramètres de l'encombrement (absorption, libre parcours moyen<sup>11</sup>), incertitudes sur le niveau de puissance de sources. . . A noter, que nous avons déjà observé cette surestimation avec la version MATLAB<sup>®</sup> de SPPS. Abstraction faite de ce décalage, l'évolution globale des niveaux sonores aux points de mesure est toutefois bien respectée entre les mesure et les simulations.

### 8.3.2 Comparaison avec les données de Ondet (1989) [29] pour un local industriel

**Données** Dans l'article de 1989 [29] déjà discuté au paragraphe 8.2.3.3, Ondet *et al* ont présenté d'autres mesures de champs sonores dans une salle d'essai, pour différentes configurations d'encombrement, afin de valider leur propre modèle RAYSCAT. Les 4 configurations d'encombrement (local vide, et local avec 3 encombrements différents) sont présentées aux figures 8.22 et 8.23, et sont détaillées dans le tableau 8.20. La source sonore a été disposée en  $x = 1.5$  m,  $y = 1$  m, et  $z = 0.85$  m dans la salle de dimension  $30 \times 8 \times 3.85$  m. Les données concernant la position des points des mesures et de la salle sont équivalentes à celles présentées au paragraphe 8.2.3.3.

**Paramètres de SPPS** Les simulations numériques ont été réalisées en prenant en compte l'absorption atmosphérique, suivant le coefficient proposé par les auteurs, soit  $m = 0.0014$  m<sup>-1</sup>. Les paramètres de simulation SPPS sont les suivants : 1 million de particules sonores, pas de temps de 5 ms pour une durée de simulation de 1.5 s. Les simulations sont réalisées sur la seule bande de fréquence 2000 Hz.

**Résultats** Les résultats des simulations SPPS, comparés aux résultats expérimentaux donnés par les auteurs sont illustrés sur les figures 8.24(a) à 8.24(c) et détaillés dans le tableau 8.21. La configuration

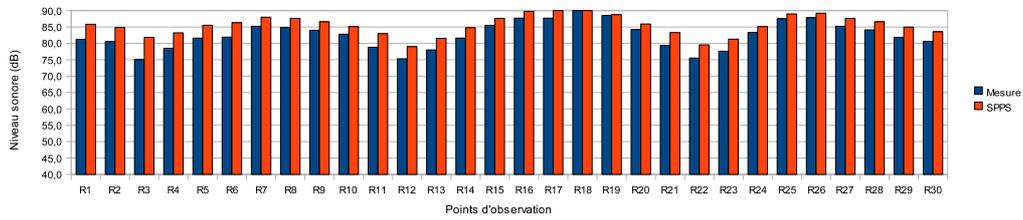


**Figure 8.23** — Configurations étudiées par Ondet *et al* [29]. L'encombrement est constitué d'obstacles de  $0.5 \times 0.5 \times 3$  m<sup>3</sup> : (a) la salle est vide (cf. paragraphe 8.2.3.3). Les microphones sont localisés à 1.5 m de hauteur, dans l'axe de la salle ( $x = 3$  m à 27 m par pas de 3 m). Un microphone de référence est situé en  $x = 3$  m et  $y = 2$  m. La source est localisée en  $x = 1.5$  m,  $y = 1$  m,  $z = 0.85$  m ; (b) la salle est entièrement encombrée (80 obstacles) ; (c) et (d) la salle est à demi-encombrée (80 obstacles), respectivement sur la première et la seconde moitié de la salle.

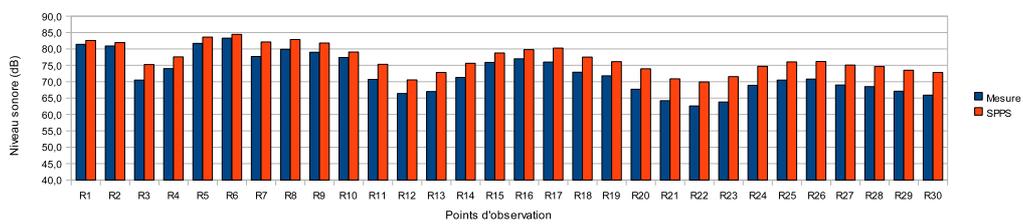
(a) correspondant au cas de la salle vide n'est pas présentée ici puisqu'elle a déjà été validée au paragraphe 8.2.3.3. Les résultats sont là encore très bons puisque les écarts-moyens sont de l'ordre de 0.7, 1.1 et 1.8 dB respectivement pour les configurations b, c et d, les écarts les plus importants étant localisés globalement au niveau des points d'observation les plus proches de la source. Ces écarts peuvent venir de la différence de comportement entre un encombrement déterministe (la mesure a été réalisée avec des encombrements régulièrement espacés), et d'un encombrement statistique dans le code SPPS (modélisation par l'intermédiaire d'un libre parcours moyen). **En conclusion, le code SPPS est bien validé pour cette configuration.**

<sup>10</sup>Les autres comparaisons réalisées dans le cadre de la validation du code SPPS montrent un très bon accord comportement du code.

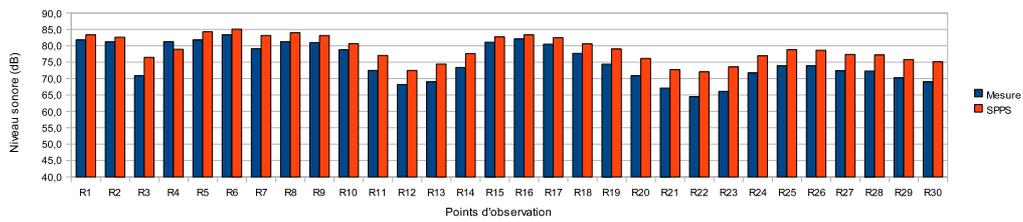
<sup>11</sup>L'article de référence [31] et le rapport de mesure correspondant [30], mentionne des fréquences de collisions différentes, respectivement de  $1/5$  m<sup>-1</sup>, et  $0.07/0.1$  m<sup>-1</sup>. Nous nous sommes placés toutefois dans la configuration qui semble la plus avantageuse, soit  $1/5$  m<sup>-1</sup> (i.e.  $\lambda = 1/5$  m<sup>-1</sup>) donnée dans [31].



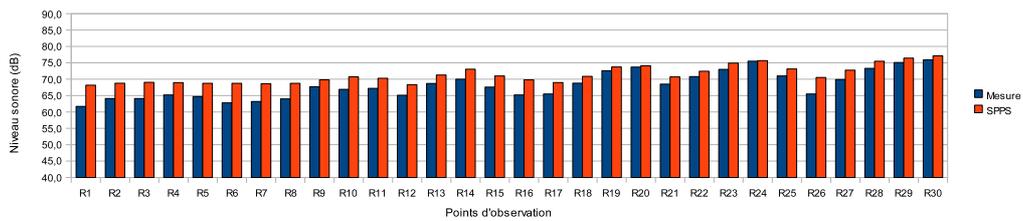
(a) Configuration 1



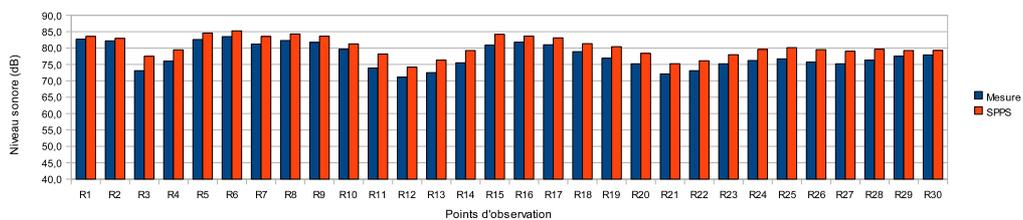
(b) Configuration 2



(c) Configuration 3

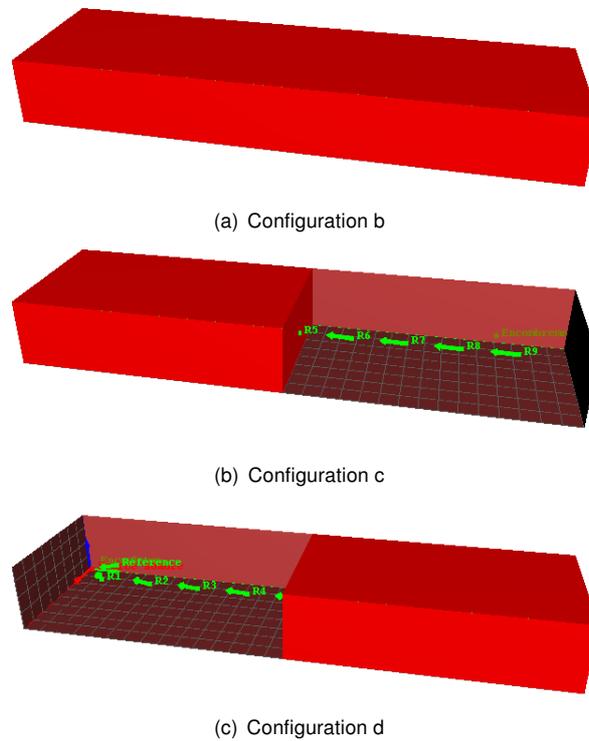


(d) Configuration 4



(e) Configuration 5

**Figure 8.21** — Comparaison de la simulation *SPFS* avec les résultats expérimentaux obtenus par Ondet et al [30, 31] dans les configurations 1 à 5, pour la bande de fréquence 1000 Hz.



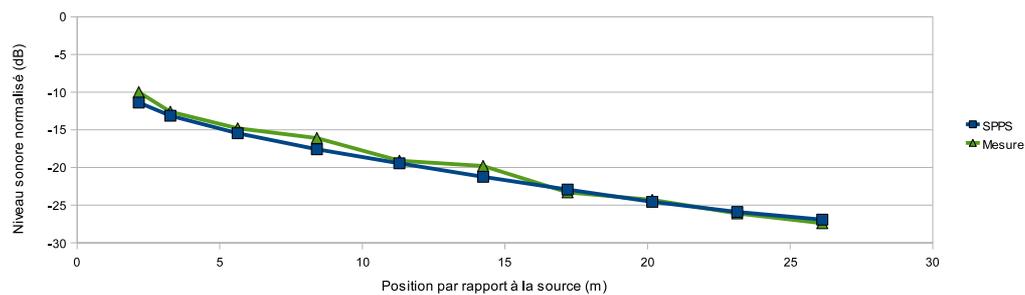
**Figure 8.22** — Configurations étudiées par Ondet et al [29] dans un local plat présentant un encombrement : configuration (b), la salle dans son intégralité est une zone d'encombrement ; Configuration (c), la première moitié de la salle est une zone d'encombrement ; configuration (d), la seconde moitié de la salle est une zone d'encombrement.

Configuration	Zone d'encombrement	$\lambda_c$ (m)	$\alpha_c$
(a)	Aucune (salle vide)	$\infty$	—
(b)	Toute la salle	7.6	0.3
(c)	$x < 15$ m	3.9	0.3
(d)	$x > 15$ m	3.9	0.3

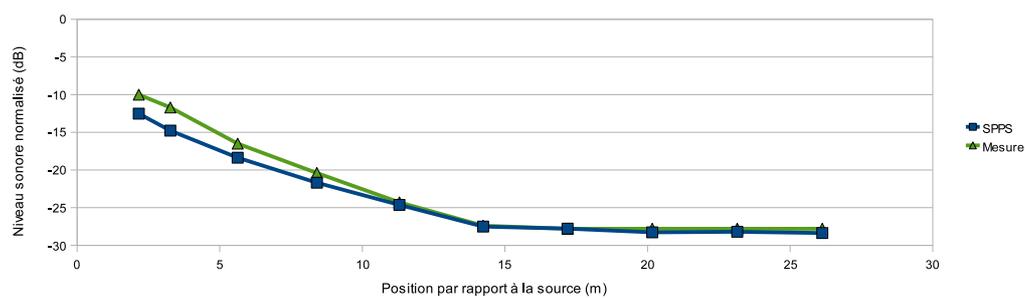
**Table 8.20** — Détail des configurations étudiées par Ondet et al [29] (cf. figure 8.23). Les encombrements sont des blocs de polystyrène de  $0.5 \times 0.5 \times 3$  m et d'absorption  $\alpha_c = 0.3$ . Les encombrements sont répartis sur (presque) toute la hauteur de la salle. La configuration (a), correspondant à un local vide, a déjà fait l'objet d'une validation au paragraphe 8.2.3.3.

**Table 8.21** — Résultats expérimentaux obtenus par Ondet et simulations numériques SPPS. (\*) Position de référence.

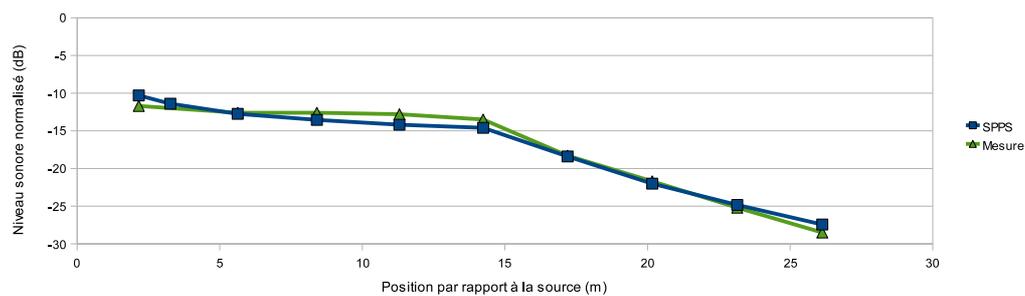
Position Y (m)	3.0*	3.0	6.0	9.0	12.0	15.0	18.0	21.0	24.0	27.0
Configuration b										
Mesure (dB)	-10.0	-12.6	-14.8	-16.1	-19.1	-19.8	-23.3	-24.3	-26.1	-27.4
SPPS (dB)	-11.4	-13.1	-15.5	-17.6	-19.4	-21.2	-22.9	-24.6	-25.9	-26.9
Configuration c										
Mesure (dB)	-10.0	-11.7	-16.5	-20.4	-24.3	-27.4	-27.8	-27.8	-27.8	-27.8
SPPS (dB)	-12.5	-14.8	-18.4	-21.7	-24.6	-27.5	-27.9	-28.3	-28.2	-28.4
Configuration d										
Mesure (dB)	-11.7	—	-12.6	-12.6	-12.8	-13.5	-18.3	-21.7	-25.2	-28.5
SPPS (dB)	-10.3	-11.4	-12.7	-13.6	-14.2	-14.6	-18.4	-22.0	-24.8	-27.4



(a) Configuration b



(b) Configuration c



(c) Configuration d

**Figure 8.24** — Comparaison des résultats expérimentaux obtenus par Ondet et al [29] dans un local présentant un encombrement (bande d'octave 2 kHz) aux résultats de la simulation avec le code SPPS.



## Exemples d'application

Le code SPPS peut être exécuté « en ligne » ou en utilisant l'interface I-Simpa.

Dans le premier cas, l'utilisateur doit créer les fichiers d'entrée et extraire lui-mêmes les informations des fichiers de résultats à la sortie de l'exécutable SPPS. Cette procédure est relativement fastidieuse. À titre d'exemple, une application est proposée à la section 9.2.

La tâche peut être largement simplifiée en utilisant l'interface I-Simpa développée par l'Ifsttar; l'ensemble des fichiers d'entrée et de sortie sont alors totalement gérés par l'interface. Le manuel d'utilisation de l'interface présente plusieurs exemples d'application avec le code SPPS; l'utilisateur est invité à s'y reporter.

### 9.1 Utilisation de SPPS avec I-Simpa

Se reporter au manuel d'utilisation de l'interface I-Simpa (chapitre « Tutoriaux »).

### 9.2 Utilisation de SPPS en ligne

#### 9.2.1 Principe

L'utilisation de ces scripts nécessite l'installation au préalable de Python™ version 2.5 à 3, cette dernière étant exclue ([www.python.org](http://www.python.org)).

Afin d'illustrer l'utilisation du code SPPS, nous présentons un exemple d'application pour une géométrie cubique de 5 m d'arrête, avec une source sonore impulsionnelle localisée en (1, 1, 2) et un récepteur ponctuel en (4, 4, 2). La salle est caractérisée uniquement par deux matériaux, et ne possède pas d'en-

combrement. Le détail de la configuration et des paramètres de calcul est donné dans le fichier de configuration `config.xml` (cf. paragraphe 9.2.3.2).

Pour des raisons pratiques, il est plus simple d'utiliser le langage Python™, avec la librairie associée LibSimpa, que le langage C++ avec la librairie LibInterface. Dans l'exemple proposé, nous nous limitons donc uniquement aux scripts Python™ pour la réalisation des fichiers d'entrée, l'exécution du code et la conversion des fichiers de sortie (ou fichier de résultats) au format CSV (éditable avec un tableur, la « virgule » étant choisie séparateur). L'utilisateur pourra se reporter à la documentation de la librairie LibSimpa pour plus de précisions.

#### 9.2.2 Procédure

La procédure est la suivante<sup>1</sup> :

1. ouvrir l'invite de commande Python™;
2. se placer dans le répertoire<sup>2</sup> contenant le script `spps_exemple.py`. À la racine de ce fichier, vous devez également trouver le répertoire `samplein/` contenant le fichier de configuration `config.xml`, et le répertoire SPPS contenant l'exécutable du code (par exemple) :

```
os.chdir("C:\tmp\application_spps\")
```

3. dans l'invite de commande, lancer la commande :

<sup>1</sup>La procédure présentée peut être réalisée automatiquement en exécutant le fichier `executer_exemple.bat` localisé à la racine.

<sup>2</sup>Les fichiers et répertoires nécessaires sont normalement créés à l'installation. Voir la procédure d'installation au paragraphe 3.2

```
exec("spps_exemple.py")
```

Pendant l'exécution, la progression du calcul (compteur) et certaines informations sont affichées dans la fenêtre de commande. A la fin de l'exécution, les fichiers suivants sont créés :

- modèle géométrique (salle cubique) : `sampleout/modele_geo.cbin`;
- maillage associé au modèle : `sampleout/modele_mesh.mbin`;
- fichiers de cumul énergétique dans le modèle : `cumul_energetique.rec` et `cumul_energetique.csv`;
- fichiers d'information du calcul : `statSPPS.gabe` et `statSPPS.csv`;
- fichiers de résultats au niveau du récepteur ponctuel dans le répertoire `sampleout/recepteurs_ponctuels/Récepteur 1/` :
  - fichier pour le calcul des paramètres avancés : `energie_laterale.gabe` et `energie_laterale.csv`;
  - fichier de niveau sonore : `energie.rec` et `energie.csv`;
  - fichier de vecteur intensité<sup>3</sup> : `ponct_vecteur_intensite.gabe` et `ponct_vecteur_intensite.csv`;
- fichiers de particules sonores dans le répertoire `sampleout/particules/` pour chaque bande de fréquence : `trajectoire_particules.pbin` et `trajectoire_particules.csv`.

### 9.2.3 Détail des scripts

#### 9.2.3.1 Script d'exécution

Le script général `spps_exemple.py` (cf. listing 9.1) permet la création des fichiers d'entrée (exécution des deux scripts présentés aux paragraphes 9.2.3.2 et 9.2.3.2 pour la génération du modèle et du maillage associé), l'exécution du code, et la conversion des fichiers de sortie binaires (cf. paragraphe 9.2.3.3) créés par le code SPPS dans un format CSV éditable dans un tableur :

<sup>3</sup>Un répertoire `intensity_animation` est également créé mais n'apporte pas d'informations supplémentaires. Les fichiers situés dans ce répertoire avec l'extension `.rpi` sont utilisés uniquement pour la représentation graphique des vecteurs intensités pour chaque bande de fréquence

- la première partie du script permet d'importer la librairie `LibSimpa` et de charger les trois fonctions supplémentaires (fonction de création du modèle et du maillage, et fonction de conversion des données) qui vont être utilisées par la suite. Les librairies `subprocess` et `os` (utilisation du système d'exploitation, voir documentation de Python™) sont également chargées ;
- la seconde partie permet de créer les fichiers d'entrée du modèle, à savoir le modèle géométrique `mesh.cbin` et le maillage `tetramesh.mbin`, dans le répertoire `sampleout/` (cf. paragraphes 9.2.3.2 et 9.2.3.2) ;
- la troisième partie permet d'appliquer l'exécutable SPPS `spps.exe`, sur le fichier de configuration associé `config.xml` ;
- enfin, la quatrième partie permet de convertir les données des fichiers binaires de sortie créés par le code SPPS en fichiers CSV, à l'aide de la fonction `ConversionDossierSpps('sampleout')` présentée au paragraphe 9.2.3.3. Les fichiers sont alors créés dans le répertoire `sampleout/`.

#### 9.2.3.2 Scripts de création des fichiers d'entrée

**Fichier de configuration (fichier `.xml`)** Le fichier de configuration du calcul a été réalisé conformément à la spécification présentée au paragraphe 7.2. Dans l'exemple proposé ici<sup>4</sup>, les calculs sont réalisés :

- pour toutes les bandes de tiers d'octave comprises entre 50 Hz et 5000 Hz ;
- en considérant trois matériaux différents d'indice 66, 100 et 0 (matériau par défaut) ;
- avec une source ponctuelle en (1, 1, 2) ;
- un récepteur ponctuel en (4, 4, 2) ;
- sans encombrement ;
- sans récepteur surfacique ;
- et un jeu de paramètres de calcul.

<sup>4</sup>Le fichier de configuration n'est pas présenté dans le document. Le lecteur pourra éditer ce fichier pour vérifier le détail du contenu.

**Modèle géométrique (fichier .cbin)** Le script `spps_modele.py` (listing 9.2) permet de générer un modèle géométrique simple, pour une géométrie cubique, avec conversion au format `.cbin` requis. L'utilisateur pourra s'inspirer de ce script pour construire d'autres fichiers de modèle avec ses propres données de sommets et de faces :

- la première ligne (hors commentaires, ligne N°10) du code permet de charger la librairie ;
- la première partie du code permet de définir les 8 sommets du cube (notés `sommets`), l'origine du repère étant donné par le second sommet. Chacune des 6 faces du modèle est alors divisée en deux triangles, ce qui donne un total de 12 faces élémentaires (notées `faces`). A chaque face sont ensuite attribuées plusieurs identifications : un indice d'encombrement, un indice de matériau, et un indice de récepteur surfacique. Les spécifications de ce fichier sont précisées au paragraphe 7.3. Dans cet exemple, aucune des faces n'est associée à un encombrement ou récepteur surfacique (indice -1), et uniquement deux matériaux sont considérés (66 et 100) ;
- La seconde partie du code permet de mettre les données précédentes au format requis, créant la variable `model`, grâce aux deux fonctions `ls.ioModel()` et `ls.ioFace()`, respectivement pour les sommets et les faces. Enfin, la dernière ligne de cette partie permet de sauvegarder le modèle géométrique avec le nom (et l'arborescence) défini dans le paramètre `filepath` ;
- l'appel à cette fonction est réalisé par la commande suivante, créant ainsi le fichier de modèle "modele\_geo.cbin" ;

**Maillage du modèle (fichier .mbin)** Le script `spps_maillage.py` (listings 9.3 et 9.4) permet de générer un maillage tétraédrique très simple, adapté pour le modèle géométrique présenté plus haut. L'utilisateur pourra s'inspirer de ce script, notamment pour la partie de conversion au format `.mbin`. Dans cet exemple, le maillage est réalisé manuellement, mais l'utilisateur pourra utiliser son propre maillage tétraédrique :

- comme précédemment le chargement de la librairie `LibSimpa` est réalisé à la première ligne ;
- les deux fonctions suivantes (cf. listing 9.3), notées `push_mesh_nodelist(maillage,vertices)` et `push_tetraedres(maillage,tetralst)`

permettent de convertir des tableaux Python™ au format requis par le code SPPS ;

- la réalisation du fichier maillage est réalisée dans la seconde partie du listing (listing 9.4). Dans un premier temps, les 10 sommets des tétraèdres (notés `m_sommets`) et les 14 tétraèdres (notés `tetras`) dans un cube sont créés manuellement. Chaque tétraèdre est défini par 4 sommets, un indice d'encombrement, et les caractéristiques de chaque face (indice de la face du modèle correspondante (si existante, -1 sinon), indice du tétraèdre voisin à la face correspondante (-2 si aucune), indice des 3 sommets de la face). Voir le paragraphe 7.4 pour plus de détails. Dans un second temps, la création du fichier est réalisé par la fonction `CreerMaillage(filepath)`, qui restructure les données (avec les deux fonctions précédentes) et écrit le fichier `filepath` (incluant l'arborescence) ;
- l'appel à cette fonction est réalisé par la commande suivante, créant ainsi le fichier de modèle "modele\_mesh.mbin" ;

### 9.2.3.3 Conversion des fichiers de résultat

**Script de conversion des fichiers** Le script `spps_conversion_resultats.py` (listings 9.5 et 9.6) permet de convertir en fichiers CSV (la « virgule » étant choisie séparateur), l'intégralité des fichiers binaires au format GABE créés en sortie du code SPPS, à savoir les fichiers de résultat au niveau d'un récepteur volumique (extensions `.recp`, `.gap` et `.pri`), et le fichier contenant les informations sur le calcul (extension `.gabe`) :

- la première partie du code permet de charger les libraires ;
- les deux fonctions suivantes `gettext(filename)` et `getname(filename)` sont créées pour extraire le nom et l'extension des fichiers de sorties ;
- le détail de la conversion des données est définie dans la fonction `GabeToCsv(filepath, csvpath)` permettant de convertir un fichier GABE de résultat, nommé `filepath` (incluant l'arborescence), en un fichier CSV nommé `csvpath` (incluant l'arborescence). La lecture des données dans un fichier GABE est réalisée par la fonction `ls.Gabe_rw()` ;
- la procédure complète de conversion est définie par la fonction

ConversionDossierSpps('sampleout'),  
 qui convertie simultanément tous les fichiers  
 GABE présents dans le dossier sampleout et  
 d'extension .gabe, .gap, .recp et .pri, en  
 fichiers CSV de même nom mais d'extension  
 .csv.

**Contenu des fichiers .csv** Le contenu de chaque fichier .csv est explicite. Ces fichiers peuvent être édités avec un tableur. Les quantités « énergétiques » données dans ces tableaux sont celles définies pour chacun des fichiers binaires correspondant. Par ailleurs, la structure des données des fichiers .csv reprend en général celle des fichiers binaires (*cf.* section 7.5).

**Fichier de récepteur ponctuel (paramètres avancés) (.gap)** Le fichier .csv correspondant reprend exactement la structure du fichier binaire correspondant, mais nécessite au préalable une transposition (inversion des lignes et des colonnes) dans le tableur.

**Fichier de récepteur ponctuel pour le vecteur intensité sonore (.pri)** Les premières lignes du tableau rappellent uniquement les bandes de fréquence considérées. Les données importantes commencent à la ligne correspondant à l'apparition des données sur la troisième colonne. A partir de cette ligne, la première colonne donne l'échelle des temps. Les colonnes suivantes donnent, par série de 3 colonnes (*i.e.* pour une bande de fréquence), les coordonnées  $(x, y, z)$  du vecteur d'intensité sonore.

**Fichier de récepteur ponctuel et global (.recp)** La première colonne donne l'axe des temps. Les colonnes suivantes donnent les quantités d'énergie pour chaque bande de fréquence de calcul.

**Fichier d'information du calcul (.gabe)** Les données sont organisées en colonne pour les bandes de fréquence. Chaque ligne donne une information sur le suivi du nombre de particules dans le domaine.

**Fichier de suivi des particules sonores (.pbin)** Un fichier est défini pour chaque bande de fréquence. Pour chaque fichier, la première colonne donne l'axe des temps ; les trois colonnes suivantes donnent respectivement les coordonnées  $(x, y, z)$  sont données en colonne ; la dernière colonne donne la quantité d'énergie portée par la particule. Les informations concernant les particules sont données successivement (1 ligne de séparation entre deux particules).

**Listing 9.1** — Exemple de script Python™ permettant d'exécuter le code SPPS. Ce script principal fait appel à plusieurs scripts pour la création du modèle et du maillage, présentés dans les sections suivantes.

```

1 # -*- coding: cp1252 -*-
2
3 # Titre: Exemple de script pour l'utilisation du code SPPS
4 # Description: Application à une salle cubique (5m)
5 # Date: Avril 2009
6 # Auteur: N. Fortin et J. Picaut
7 # Contact: Judicael.Picaut@lcpc.fr
8
9 #####
10 # Partie 1: Importation de la librairie et des fonctions
11 import libsimpa as ls
12 from spps_modele import CreerModele
13 from spps_maillage import CreerMaillage
14 from spps_conversion_resultats import ConversionDossierSpps
15
16 # Librairies standards
17 # Importation de la librairie Python permettant d'exécuter un logiciel externe
18 import subprocess as sp
19 import os
20
21 #####
22 ## Partie 2: Création des fichiers d'entrée
23 if not os.path.exists("sampleout"):
24     os.mkdir("sampleout")
25 if not os.path.exists("samplein/config.xml"):
26     print "Vous devez créer le fichier XML de configuration."
27     exit(-1)
28
29 CreerModele("sampleout/modele_geo.cbin")
30 CreerMaillage("sampleout/modele_mesh.mbin")
31
32 #####
33 ## Partie 3: Exécution du code SPPS
34 sp.check_call(["spps/spps.exe", "samplein/config.xml"])
35
36 #####
37 ## Partie 4: Conversion des données de sortie en fichier CSV
38 ConversionDossierSpps('sampleout')
```

**Listing 9.2** — Exemple de script Python™ générant un fichier de modèle au format `.cbin` à partir d'une liste de points et de faces. L'exécution de la commande `CreerModele("modele_geo.cbin")` dans une console Python™ (après importation de la librairie `LibSimpa`) permet de créer le fichier de modèle `modele_geo.cbin`.

```

1  # -*- coding: cp1252 -*-
2
3  # Titre: Script de génération d'un modèle géométrique utilisé par le code SPSS
4  # Description: Création d'une salle cubique (5m)
5  # Date: Avril 2009
6  # Auteur: N. Fortin et J. Picaut
7  # Contact: Judicael.Picaut@lpc.fr
8
9  # Chargement de la librairie
10 import libsimpa as ls
11
12 #####
13 # Partie 1:
14 # Création des tableaux de données géométriques (sommets et faces)
15
16 # Définition des 8 sommets du cube
17 sommets=[ [5.0, 0.0, 0.0],
18           [0.0, 0.0, 0.0],
19           [0.0, 5.0, 0.0],
20           [5.0, 5.0, 0.0],
21           [0.0, 5.0, 5.0],
22           [5.0, 5.0, 5.0],
23           [0.0, 0.0, 5.0],
24           [5.0, 0.0, 5.0]
25         ]
26 # Définition des 12 faces triangulaires du cube
27 # [ sommetA, sommetB, sommetC, idencembrement, idmateriau, idrecepteursurf ]
28 faces= [ [0, 1, 2, -1, 66, -1],
29          [0, 2, 3, -1, 66, -1],
30          [2, 4, 5, -1, 100, -1],
31          [2, 5, 3, -1, 100, -1],
32          [2, 6, 4, -1, 100, -1],
33          [2, 1, 6, -1, 100, -1],
34          [1, 0, 7, -1, 100, -1],
35          [6, 1, 7, -1, 100, -1],
36          [0, 3, 5, -1, 100, -1],
37          [7, 0, 5, -1, 100, -1],
38          [7, 5, 4, -1, 66, -1],
39          [6, 7, 4, -1, 66, -1]
40        ]
41 #####
42 # Partie 2:
43 # Création du modèle 3D et sauvegarde dans le format requis
44
45 def CreerModele(filepath):
46     """
47     Création du modèle 3D
48     """
49     model=ls.ioModel()
50     # Ajout des sommets
51     for sommet in sommets:
52         model.vertices.append(ls.t_pos(sommet[0],sommet[1],sommet[2]))
53     # Ajout des faces
54     for face in faces:
55         newface=ls.ioFace()
56         newface.a=face[0]
57         newface.b=face[1]
58         newface.c=face[2]
59         newface.idEn=face[3]
60         newface.idMat=face[4]
61         newface.idRs=face[5]
62         model.faces.append(newface)
63     # Sauvegarde du modèle
64     ls.CformatBIN().ExportBIN(filepath,model)

```

**Listing 9.3** — Exemple de script Python™ générant un fichier de maillage au format `.mbin` à partir d'une liste de points et de faces. L'exécution de la commande `CreerMaillage("model_mesh.mbin")` dans une console Python™ (après importation de la librairie `LibSimpa`) permet de créer le fichier de maillage `model_mesh.mbin`.

```

1  # -*- coding: cp1252 -*-
2
3  # Titre: Script de génération d'un maillage au format utilisé par le code SPPS
4  # Description: Création du maillage associé à une salle cubique (5m)
5  # Date: Avril 2009
6  # Auteur: N. Fortin et J. Picaut
7  # Contact: Judicael.Picaut@lpc.fr
8
9  # Chargement de la librairie
10 import libsimpa as ls
11
12 def push_mesh_nodelist(maillage,vertices):
13     """
14         Transforme un tableau Python en liste de positions de noeud dans un maillage
15         Le tableau python doit être de taille nbsommets*3
16     """
17     for vertex in vertices:
18         newnode=ls.t_binNode()
19         newnode.node[0]=vertex[0]
20         newnode.node[1]=vertex[1]
21         newnode.node[2]=vertex[2]
22         maillage.nodes.append(newnode)
23
24 def push_tetraedres(maillage,tetralst):
25     """
26         Transforme un tableau Python en structure de données de maillage
27     """
28     for tetra in tetralst:
29         newtetra=ls.bintetrahedre()
30         newtetra.sommets[0]=tetra[0][0]
31         newtetra.sommets[1]=tetra[0][1]
32         newtetra.sommets[2]=tetra[0][2]
33         newtetra.sommets[3]=tetra[0][3]
34         newtetra.idVolume=tetra[1]
35         for idface in range(0,4):
36             face=tetra[2][idface]
37             newface=newtetra.tetrafaces[idface]
38             newface.marker=face[0]
39             newface.neighbour=face[1]
40             newface.sommets[0]=face[2][0]
41             newface.sommets[1]=face[2][1]
42             newface.sommets[2]=face[2][2]
43         maillage.tetraedres.append(newtetra)

```

Listing 9.4 — Suite du listing 9.3

```

44
45 # Définitions des sommets des tétraèdres
46 m_sommets = [ [5.0, 0.0, 0.0],
47               [0.0, 0.0, 0.0],
48               [0.0, 5.0, 0.0],
49               [5.0, 5.0, 0.0],
50               [0.0, 5.0, 5.0],
51               [5.0, 5.0, 5.0],
52               [0.0, 0.0, 5.0],
53               [5.0, 0.0, 5.0],
54               [0.0, 2.5, 2.5],
55               [2.5, 0.0, 2.5]
56           ]
57 # Définition des tétraèdres
58 # -----Sommets du tétraèdre__  _Encombrement_  __4 faces du
   tétraèdre-----
59 # [ [ sommetA, sommetB, sommetC, sommetD ] , idVolume      , [ [idfacemodel ,
   idtetravoisin, [ SommetA, SommetB, SommetC]] , .. ] ]
60 tetras=[[ [3, 9, 2, 0], 0, [[-1, 11, [9, 0, 2]], [1, -2, [2, 0, 3]], [-1, 2, [3, 0, 9]],
   [-1, 12, [9, 2, 3]]]],
61        [[5, 9, 7, 8], 0, [[-1, 10, [9, 8, 7]], [-1, 9, [7, 8, 5]], [-1, 13, [5, 8, 9]], [-1,
   5, [9, 7, 5]]]],
62        [[3, 9, 0, 5], 0, [[-1, 5, [9, 5, 0]], [8, -2, [0, 5, 3]], [-1, 13, [3, 5, 9]], [-1, 0,
   [9, 0, 3]]]],
63        [[2, 8, 5, 4], 0, [[-1, 9, [8, 4, 5]], [2, -2, [5, 4, 2]], [4, -2, [2, 4, 8]], [-1, 4,
   [8, 5, 2]]]],
64        [[3, 8, 5, 2], 0, [[-1, 3, [8, 2, 5]], [3, -2, [5, 2, 3]], [-1, 12, [3, 2, 8]], [-1,
   13, [8, 5, 3]]]],
65        [[5, 9, 0, 7], 0, [[6, -2, [9, 7, 0]], [9, -2, [0, 7, 5]], [-1, 1, [5, 7, 9]], [-1, 2,
   [9, 0, 5]]]],
66        [[6, 1, 8, 9], 0, [[-1, 7, [1, 9, 8]], [-1, 10, [8, 9, 6]], [7, -2, [6, 9, 1]], [5, -2,
   [1, 8, 6]]]],
67        [[2, 9, 8, 1], 0, [[-1, 6, [9, 1, 8]], [5, -2, [8, 1, 2]], [-1, 11, [2, 1, 9]], [-1,
   12, [9, 8, 2]]]],
68        [[7, 8, 6, 4], 0, [[4, -2, [8, 4, 6]], [11, -2, [6, 4, 7]], [-1, 9, [7, 4, 8]], [-1,
   10, [8, 6, 7]]]],
69        [[5, 8, 7, 4], 0, [[-1, 8, [8, 4, 7]], [10, -2, [7, 4, 5]], [-1, 3, [5, 4, 8]], [-1, 1,
   [8, 7, 5]]]],
70        [[7, 9, 6, 8], 0, [[-1, 6, [9, 8, 6]], [-1, 8, [6, 8, 7]], [-1, 1, [7, 8, 9]], [7, -2,
   [9, 6, 7]]]],
71        [[2, 9, 1, 0], 0, [[6, -2, [9, 0, 1]], [0, -2, [1, 0, 2]], [-1, 0, [2, 0, 9]], [-1, 7,
   [9, 1, 2]]]],
72        [[9, 3, 2, 8], 0, [[-1, 4, [3, 8, 2]], [-1, 7, [2, 8, 9]], [-1, 13, [9, 8, 3]], [-1, 0,
   [3, 2, 9]]]],
73        [[3, 9, 5, 8], 0, [[-1, 1, [9, 8, 5]], [-1, 4, [5, 8, 3]], [-1, 12, [3, 8, 9]], [-1, 2,
   [9, 5, 3]]]]
74
75 def CreerMaillage(filepath):
76     """
77     Création du fichier de maillage
78     """
79     maillage=ls.trimeshmodel()
80     # Ajout des sommets
81     push_mesh_nodelist(maillage,m_sommets)
82
83     # Ajout des tétraèdres
84     push_tetraedres(maillage,tetras)
85
86     # Sauvegarde du maillage
87     ls.CMBIN().SaveMesh(filepath,maillage)

```

**Listing 9.5** — Exemple de script *Python*<sup>TM</sup> permettant de convertir un fichier de résultat au format *GABE* dans un format *CSV*.

```

1 # -*- coding: cp1252 -*-
2
3 # Titre: Script de conversion en fichiers CSV
4 # Description: Script de conversion des fichiers de sortie créés par SPPS en fichier CSV
5 # Date: Avril 2009
6 # Auteur: N. Fortin et J. Picaut
7 # Contact: Judicael.Picaut@lcpc.fr
8
9 # Chargement des librairies
10 import os
11 import libsimpa as ls
12
13 def getext(filename):
14     """
15     Retourne l'extension du nom d'un fichier
16     """
17     return filename[filename.rfind(".")+1:]
18
19 def getname(filename):
20     """
21     Retourne le nom du fichier sans l'extension
22     """
23     return filename[:filename.rfind(".")]
24 def PbinToCsv(filepath, csvpath):
25     """
26     Converti un fichier PBIN (Particle BINary) en format CSV (Comma Separated Values)
27     """
28     print("Conversion d'un fichier de particules \n%s" % (filepath))
29     preader=ls.ParticuleIO()
30     preader.OpenForRead(filepath)
31     timeStep,nbParticules,nbTimeStepMax=preader.GetHeaderData()
32     fichcsv=file(csvpath,'w')
33     for idpart in range(0,nbParticules):
34         firstTimeStep, NbTimeStep=preader.NextParticle()
35         curstep=firstTimeStep*timeStep
36         for idstep in range(0,NbTimeStep):
37             curstep+=timeStep
38             x,y,z,energy=preader.NextTimeStep()
39             fichcsv.write("%f,%f,%f,%f,%s\n" % (curstep,x,y,z,energy))
40         fichcsv.write("\n")
41     fichcsv.close()

```

Listing 9.6 — Suite du listing 9.5

```

42
43 def GabeToCsv(filepath, csvpath):
44     """
45         Converti un fichier GABE (Generic Array Binary Exchange) en format CSV (Comma
46         Separated Values)
47     """
48     # Instanciation du lecteur
49     reader=ls.Gabe_rw()
50     # Lecture du fichier gabe
51     if reader.Load(filepath):
52         # Conversion en liste
53         data=reader.ToList()
54         # Rotation des données (les colonnes deviennent des lignes) si même nombre de
55         # ligne sur chaque colonne
56         if map(len,data)==[len(data[0])*len(data):
57             data=zip(*data)
58         # Ecriture des données
59         fich=open(csvpath,'w')
60         for line in data:
61             firstcol=True
62             for col in line:
63                 if not firstcol:
64                     fich.write(",")
65                 else:
66                     firstcol=False
67                     fich.write(str(col)) # Ecriture de la cellule et virgule
68                     fich.write("\n") # Retour à la ligne
69         fich.close()
70
71 def ConversionDossierSpps(folderresults):
72     for root, dirs, files in os.walk(folderresults):
73         for filename in files:
74             ext=gettext(filename)
75             name=getname(filename)
76             if ext in ["gabe","gap","recp","pri"]:
77                 GabeToCsv(root+os.sep+filename,root+os.sep+name+".csv")
78             elif ext=="pbin":
79                 PbinToCsv(root+os.sep+filename,root+os.sep+name+".csv")

```

# Bibliographie

- [1] N. Auletta : Contribution à l'étude de la propagation du son dans les locaux industriels encombrés – 1ère partie (étude théorique). *Revue d'Acoustique*, 75:477–487, 1985. 33
- [2] N. Auletta : Contribution à l'étude de la propagation du son dans des locaux industriels encombrés – 2ème partie. *Revue d'Acoustique*, 76:5–18, 1986. 33
- [3] H. E. Bass, L.C. Sutherland, J. Piercy et Evans L. : Absorption of sound by the atmosphere. In *Physical Acoustics*, volume XVII, chapitre 3, pages 143–232. Academic Press Inc, 1984. 26
- [4] P. J. Burns, J. D. Matlby et M. A. Christon : Large-scale surface to surface transport for phonons and electrons via Monte Carlo. *Computing Systems in Engineering*, 1(1):75–99, 1990. 21, 40
- [5] T.J. Cox et P. D'Antonio : *Acoustic Diffusers and Absorbers ; Theory, Design and Application*. Spon Press, London, first édition, 2004. 35
- [6] J. J. Embrechts, D. Archambeau et G.B. Stan : Determination of the scattering coefficient of random rough diffusing surfaces for room acoustics applications. *Acustica/acta acustica*, 87:482–494, 2001. 35
- [7] M. Hodgson : Evidence of diffuse surface reflections in rooms. *J. Acoust. Soc. Am.*, 89(2):765–771, 1991. 40
- [8] International Organization for Standardization : ISO 9613-1 :1993 : Acoustics – Attenuation of sound during propagation outdoors. Part 1 : calculation of the absorption of sound by the atmosphere, 1993. ICS : 17.140.01. :1993
- [9] International Organization for Standardization : ISO 140-3 :1995, Acoustics – Measurement of sound insulation in buildings and of building elements. Part 3 : laboratory measurements of airborne sound insulation of building elements, 1995. ICS : 91.120.20. :1995
- [10] International Organization for Standardization : ISO 10534-1 :1996, Acoustics – Determination of sound absorption coefficient and impedance in impedance tubes. Part 1 : method using standing wave ratio, 1996. ICS : 17.140.01. :1996
- [11] International Organization for Standardization : ISO 140-1 :1997, Acoustics – Measurement of sound insulation in buildings and of building elements. Part 1 : requirements for laboratory test facilities with suppressed flanking transmission, 1997. ICS : 91.120.20. :1997
- [12] International Organization for Standardization : ISO 266 :1997, Acoustics – Preferred frequencies., 1997. ICS : 17.140.01. :1997
- [13] International Organization for Standardization : ISO 10534-2 :1998, Acoustics – Determination of sound absorption coefficient and impedance in impedance tubes. Part 2 : transfer-function method., 1998. ICS : 17.140.01. :1998
- [14] International Organization for Standardization : ISO 140-4 :1998, Acoustics – Measurement of sound insulation in buildings and of building elements. Part 4 : field measurements of airborne sound insulation between rooms, 1998. ICS : 91.120.20. :1998
- [15] International Organization for Standardization : ISO 140-5 :1998, Acoustics – Measurement of sound insulation in buildings and of building elements. Part 5 : field measurements of airborne sound insulation of facade elements and facades, 1998. ICS : 91.060.10 ; 91.120.20. :1998
- [16] International Organization for Standardization : ISO 354 :2003, Acoustics – Measurement of sound absorption in a reverberation room, 2003. ICS : 91.120.20. :2003
- [17] International Organization for Standardization : ISO 17497-1 :2004, Acoustics – Sound-scattering properties of surfaces - Part 1 :

- measurement of the random-incidence scattering coefficient in a reverberation room, 2004. ICS : 91.120.20. :2004
- [18] W. B. Joyce : Classical-particle description of photons and phonons. *Physical Review D*, 9(12): 3234–3256, june 1974. 20, 38
- [19] W. B. Joyce : Sabine's reverberation time and ergodic auditorium. *J. Acoust. Soc. Am.*, 58(3): 643–655, 1975. 38
- [20] W. B. Joyce : Exact effect of surface roughness on the reverberation time of a uniformly absorbing spherical enclosure. *J. Acoust. Soc. Am.*, 64(5):1429–1436, Nov 1978. 38
- [21] J. Kang : *Acoustics of Long Spaces*. Thomas Telford, London, 2002. 85, 86, 87, 88
- [22] N. Korany, J. Blauert et O. Abdel Alim : Acoustic simulation of rooms with boundaries of partially specular reflectivity. *Appl. Acoust.*, 62:875–887, 2001. 91, 92
- [23] H. Kuttruff : Sound decay in reverberation chambers with diffusing elements. *The Journal of the Acoustical Society of America*, 69(6):1716–1723, 1981. 33
- [24] H. Kuttruff : Energetic sound propagation in rooms. *Acustica/acta acustica*, 83:622–628, 1997. 82, 84, 91, 93, 94, 95
- [25] H. Kuttruff : *Room Acoustics*. Spon Press, London, fourth édition, 2000. 39, 82
- [26] Y. W. Lam : A comparison of three diffuse reflection modeling methods used in room acoustics computer models. *J. Acoust. Soc. Am.*, 100(4):2181–2192, 1996. 40
- [27] T. Le Pollès : *Modélisation des champs diffus en acoustique architecturale par la théorie des transports : application au milieu urbain*. Thèse de doctorat, Université du Maine, 2003. 20, 38, 40
- [28] E. A. Lindqvist : Sound attenuation in large factory spaces. *Acustica*, 50:313–328, 1982. 33
- [29] A. M. Ondet et J. L. Barbry : Modeling of sound propagation in fitted workshops using ray tracing. *J. Acoust. Soc. Am.*, 85(2):787–796, 1989. 9, 33, 93, 94, 95, 102, 104, 105
- [30] A.M. Ondet et J.L. Barbry : Prévion des niveaux sonores dans les locaux encombrés. Rapport d'étude 52, Institut National de Recherche et de Sécurité, 1984. ISSN 0397-4529. 98, 99, 100, 101, 102, 103
- [31] A.M. Ondet et J.L. Barbry : Sound propagation in fitted rooms – comparison of differents models. *Journal of Sound and Vibration*, 125(1): 137–149, 1988. 98, 99, 100, 102, 103
- [32] Eric M. Salomons : *Computaional atmospheric acoustics*. Kluwer Academic Publishers, 2001. 30
- [33] M. Vorländer et E. Mommertz : Definition and measurement of random-incidence scattering coefficients. *Appl. Acoust.*, 60:187–199, 2000. 35
- [34] Z. Xiangyang, C. Ke-an et S. Jincai : Development of a hybrid computer model for simulating the complicated virtual sound field in enclosures. *Appl. Acoust.*, 63:481–491, 2002. 40
- [35] L. N. Yang et B. M. Shield : Development of a ray tracing computer model for the prediction of the sound field in long enclosures. *J. Sound Vib.*, 229(1):133–146, 2000. 81, 82, 89, 91

# Acronymes

<b>AES</b>	Audio Engineering Society
<b>EDT</b>	<i>Early Decay Time</i>
<b>Ifsttar</b>	Institut Français des Sciences et Technologies des Transports, de l'Aménagement et des Réseaux
<b>RT30</b>	<i>Reverberation time</i> (calculé sur une décroissance de 30 dB)
<b>SPL</b>	<i>Sound pressure level</i>
<b>TR</b>	Durée de Réverbération