

Chapitre 8

Exemples

Pour montrer l'usage de l'environnement *Motor-2*, l'application du découpage hiérarchique et la description de système aux différents niveaux d'abstraction, nous présentons trois exemples. Le *mur bicouche* est un exemple très simple, mais il permet de présenter l'application des concepts de base sans compliquer la démonstration par une physique complexe. Le deuxième exemple, celui de la *cellule de Hambourg* nous permet de valider le programme de simulation, puisque cet exemple a fait l'objet de comparaisons dans plusieurs environnements de simulation. Un dernier exemple – un local avec convection libre, échange radiatif et transfert conductif – montre l'application de *Motor-2* à un problème non-trivial.

8.1 Mur bicouche

Tout au long de ce mémoire, nous utilisons l'exemple du mur bicouche pour présenter les différents aspects de l'environnement *Motor-2*. Ici, cet exemple est traité complètement et en détail.

8.1.1 Description du problème

Nous décrivons notre système à l'aide de la grille constituée des différents niveaux d'abstraction, tels qu'ils ont été présentés au chapitre 4.1. Il peut être utile de se reporter à la figure 4.1 (page 48) pour mieux suivre les étapes qui va d'une description technique au niveau algorithmique et à l'exécution de la simulation même.

Nous commençons au niveau technique. Le problème se décrit comme un mur en béton sur lequel est clouée une couche isolante en laine de roche, protégée par une couche d'aggloméré. C'est le cas typique d'un mur extérieur de bâtiment.

Au niveau physique, nous associons la conduction thermique à travers ce mur. La distribution des températures sur la surface ne nous intéresse pas;

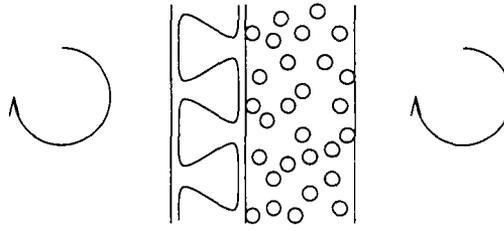


FIG. 8.1 - Le schéma physique du mur bicouche.

il est suffisant de considérer que ce problème a une seule dimension (dans la direction de l'épaisseur du mur). Le mur même a des échanges convectifs avec l'air intérieur et extérieur, et vérifie une loi avec coefficients d'échange. Pour simplifier cet exemple, l'aggloméré est négligé. La figure 8.1 montre la vue du niveau physique du mur. La géométrie de l'exemple est fixée à une surface de $10m^2$, l'épaisseur du béton est de $20cm$ et celle de l'isolation est de $8cm$.

On peut donc facilement distinguer quatre modules pour lesquels seulement deux *modèles* sont nécessaires. Les deux couches du mur (isolation et béton) avec capacité thermique sont des instances d'un modèle de conduction monodimensionnelle et les deux coefficients d'échange utilisent un modèle de résistance thermique. Entre les quatre composants nous supposons des interfaces qui ont tous le même type de couplage, un contact parfait. Les frontières des composants sont les surfaces où nous couplons les températures et les flux. On retrouve le schéma dans la figure 8.2

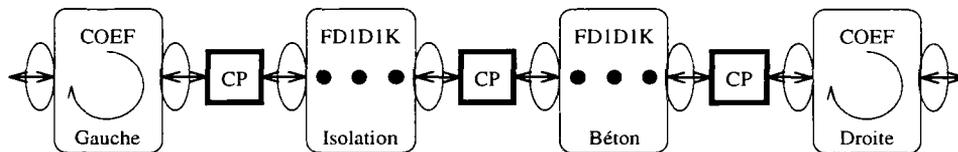


FIG. 8.2 - Le schéma de découpage. Les quatre modules avec leurs modèles associés et les trois interfaces pour le raccordement.

L'expression mathématique du modèle de conduction est la loi de FOURIER à une seule dimension :

$$\frac{dT}{dt} = \frac{\lambda}{\rho c} \frac{d^2T}{dx^2} \quad (8.1)$$

où λ est la conductivité, ρ la masse volumique, et c la capacité calorifique massique. La résistance thermique est une simple relation :

$$\phi = Ah\Delta T \quad (8.2)$$

avec h comme coefficient d'échange (l'inverse de la résistance thermique), S la surface d'échange, ΔT la différence de températures et $\phi = \frac{dQ}{dt}$ le flux de chaleur. Dans notre exemple concret, nous utilisons les valeurs caractéristiques thermo-physiques du tableau 8.3.

Au niveau algorithmique, nous devons résoudre les équations de l'étape mathématique. D'une part, il faut orienter les équations et d'autre part il faut les

conduction			convection		
	ISOLATION	BÉTON		GAUCHE	DROITE
λ [$\frac{W}{mK}$]	0.04	1.75			
ρ [$\frac{kg}{m^3}$]	30	2500	h [$\frac{W}{m^2K}$]	11	9
c [$\frac{J}{kg}$]	850	820			
A [m^2]	10	10	A [m^2]	10	10
l [m]	0.08	0.20			

FIG. 8.3 - Les valeurs thermo-physiques des matériaux du mur, des coefficients d'échange et des caractéristiques de la géométrie.

discrétiser. Nous choisissons de discrétiser en espace le modèle de conduction. L'équation de la chaleur s'exprime maintenant sous forme matricielle :

$$\frac{\partial \Theta}{\partial t} = \mathbf{A}\Theta + \mathbf{E}\mathbf{T} \quad (8.3)$$

où nous désignons par $\Theta = [\Theta_1(t), \dots, \Theta_N(t)]^t$ le vecteur d'état des températures dans les nœuds et $\mathbf{T} = [T_1, T_2]^t$ le vecteur qui contient les températures des bords. Les flux entrant et sortant aux bords sont exprimés maintenant par

$$\phi = \mathbf{J}\Theta + \mathbf{G}\mathbf{T} \quad (8.4)$$

Nous créons dix nœuds dans l'épaisseur du composant. Après discrétisation du temps, nous pouvons appliquer un algorithme d'intégration de l'équation différentielle. Ensuite nous pouvons exprimer les flux aux bords en fonction des températures imposées. La méthode simple d'EULER donne

$$\Theta_{i+1} = \Theta_i + \Delta t(\mathbf{A}\Theta_i + \mathbf{E}\mathbf{T}_i) \quad (8.5)$$

et le flux est donc

$$\phi_{i+1} = \mathbf{J}\Theta_{i+1} + \mathbf{G}\mathbf{T}_i \quad (8.6)$$

Nous avons donc les formules nécessaires pour recalculer l'état interne (le champ de températures) et les réponses des frontières (les flux) en fonction des valeurs entrantes des frontières (les températures aux bords). Dans la modélothèque de Motor-2 ces fonctions sont implémentées dans un modèle nommé FD1D1K. Ce nom est la version courte du nom complet FINITE DIFFERENCES, ONE DIMENSION, FIRST KIND. Ce nom évoque les modèles ancêtres de la bibliothèque qui sont

- 1° **ONE DIMENSION.** Ceci veut dire que toute instance de ce modèle a exactement deux frontières. Par défaut elles sont appelées LEFT et RIGHT.
- 2° **FIRST KIND.** Le type des frontières et l'orientation de leurs variables sont fixés aux frontières avec une température comme entrée et un flux comme sortie.

Les autres paramètres libres de ce modèle sont le nombre de nœuds internes et les valeurs thermo-physiques comme spécifiées dans la figure 8.3.

Les calculs des résistances thermiques ne nécessitent pas d'opérations internes. Il n'y a pas d'état interne. L'équation 8.2 s'applique directement. Le modèle dans la bibliothèque Motor-2 est COEF. Il est également dérivé de ONE DIMENSION et de FIRST KIND avec les mêmes conséquences que pour le modèle FD1D1K ; deux frontières avec températures comme entrées et flux comme sorties.

Les interfaces sont du type CONTACT PARFAIT. Elles contiennent et conservent une température qui est envoyée aux modules connectés. Les modules à leur tour répondent avec le flux. L'interface calcule la somme des flux et fait varier sa température pour annuler la somme de flux. Nous appliquons l'algorithme optimisé du chapitre 6.2.2 pour les interfaces du CONTACT PARFAIT.

8.1.2 Simulation et résultats

La première ligne de la figure 8.4 montre un module principal MUR_OUEST qui est un module composé de quatre sous-modules. Ceci est le choix le plus simple pour une hiérarchie de calcul ; tous les sous-modules se trouvent au même niveau. Trois interfaces internes imposent les conditions de raccordement. Nous appelons cette configuration le découpage **L**.

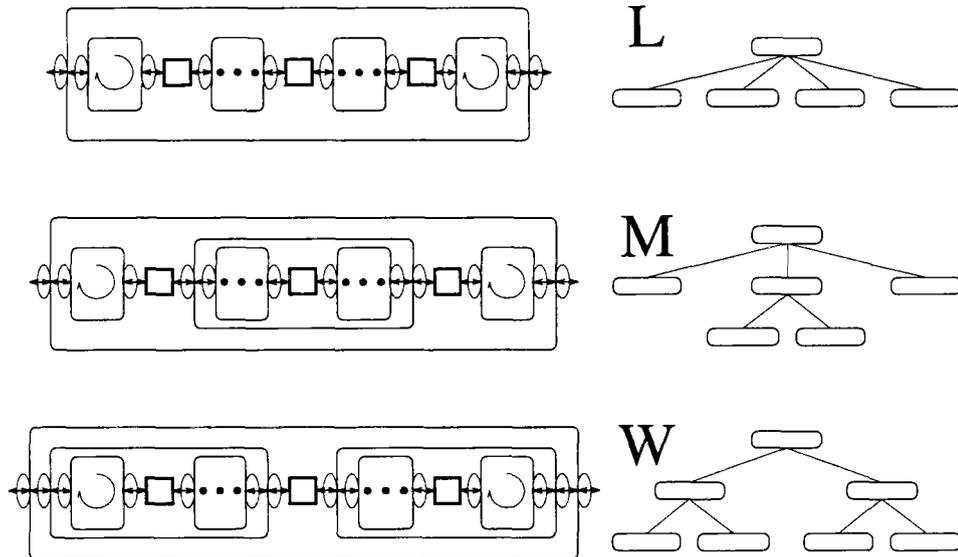


FIG. 8.4 - Différents graphes de découpage et couplage possible avec les noms de référence.

Mais nous avons testé également d'autres configurations comme schématisées dans la figure 8.4. La configuration **M** est un découpage plus « physique ». Les deux couches du mur constituent ensemble un module composé MUR avec une interface entre les modules terminaux. Les trois modules EXT, MUR, et INT sont connectés par deux interfaces pour construire le module principale

OUEST_MUR_M. La configuration **W** est issue de considérations numériques. Comme l'évolution la plus rapide se trouve dans l'ISOLATION et le coefficient de EXT, ces deux modules sont mis ensemble pour éviter des activations non-nécessaires des modules de droite.

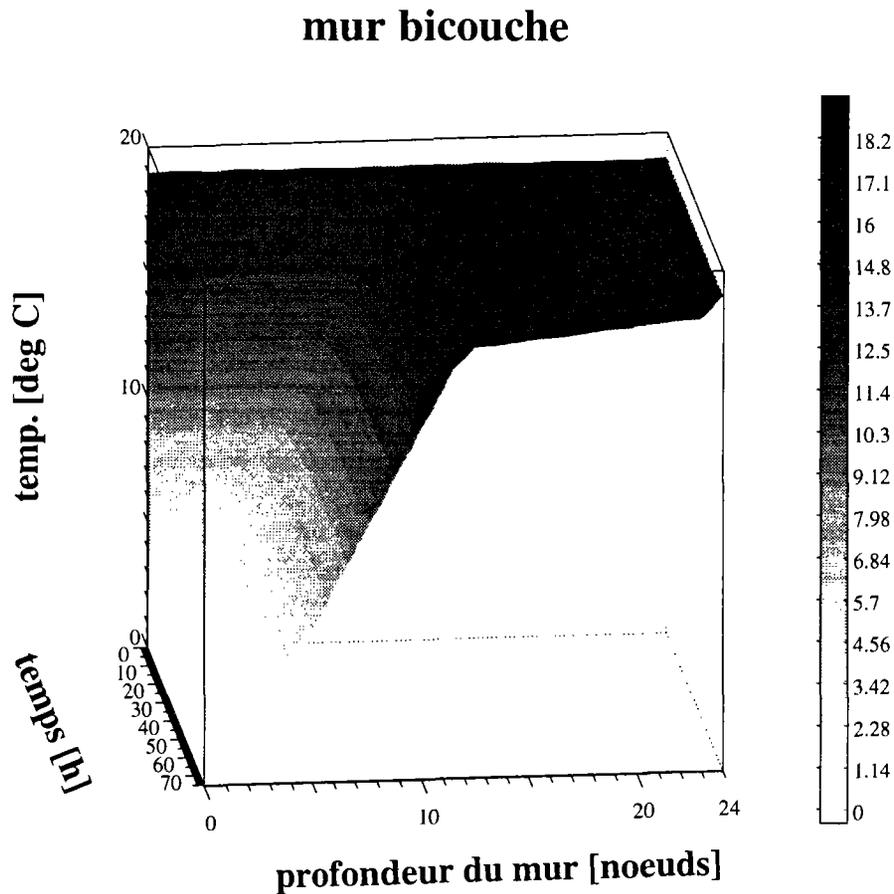


FIG. 8.5 - L'évolution temporelle jusqu'au régime permanent du champ de températures du mur bicouche.

La simulation effectuée à l'aide des configurations **L**, **M** et **W** est initiée avec une température uniforme dans tous les modules de 19°C . À l'instant $t = 0$, la température extérieure (à gauche) tombe à 0°C . La simulation est effectuée avec un pas de temps de 1800 secondes. La *Jacobienne* interne n'est réévaluée qu'après trois pas d'itération. Le seuil d'arrêt des itérations est fixé à une précision de 0.01 %. La simulation est poursuivie jusqu'à l'état stationnaire. Il est atteint après 75 heures. Non seulement les différents découpages, mais aussi les différentes méthodes de résolution, ont été testés à l'aide de ces exemples.

L'évolution temporelle des températures internes du mur est montrée dans la figure 8.5. La hauteur² indique la température. Au fond de la figure se trouve le

2. et les niveaux de gris qui sont mal visibles dans cette reproduction.

pas zéro où toutes les températures sont à 19°C . Au premier plan, la distribution de température a atteint le régime permanent. La profondeur du mur est discrétisée en 25 nœuds. Le nœud zéro est la température imposée de l'extérieur, le nœud 24 la température imposée de l'intérieur. Les nœuds 1, 12 et 23 contiennent les températures des interfaces. La couche de l'isolant est représentée par les dix nœuds 2 à 11, et la couche de béton par les nœuds 13 à 22.

On remarque que la température qui tombe à zéro à l'extérieur au début de la simulation ne devient sensible pour la couche de béton qu'après environ 10 heures. Il faut attendre encore plus que 60 heures pour stabiliser le système, ce qui est dû à la grande inertie du béton. Mais même après ce temps la température de la surface extérieure de béton n'est descendue qu'à 17.1°C . La partie la plus importante de la résistance thermique se trouve – comme voulu – dans la couche isolant.

Numériquement, toutes les configurations étudiées donnent la même solution, à une grandeur inférieure à la précision prêt. C'est le signe que le choix de la méthode des découpage n'affecte pas le résultat physique (!). Les différences entre les méthodes de résolution et entre les différentes configurations se manifestent dans le temps nécessaire et donc à la quantité de calculs nécessaire pour aboutir à la solution. La figure 8.6 montre une matrice contenant les temps de calcul³ où dans les lignes nous comparons les différentes méthodes et dans les colonnes se trouvent les différentes configurations de découpage.

	L	M	W
GLOBAL	7.4	13.8	14.0
LOCAL	4.8	5.5	6.7
ASYNC	6.8	7.5	8.1

FIG. 8.6 - Les temps de calculs pour les différentes configurations et méthodes de résolution.

La comparaison entre les différentes configurations montre que le choix **L** est toujours le plus rapide. Dans cette configuration, nous ne faisons pas de calcul hiérarchisé. Tous les modules se trouvent au même niveau. Si un découpage existe, comme pour les cas **M** et **W**, c'est la configuration **M** qui est la plus rapide. Ceci nous conforte dans ce sens que le choix de découpage issu des considérations physique donne également des bons résultats d'efficacité numérique. Le choix de l'endroit du découpage n'est pas toujours évident, surtout quand il n'y a pas de distinction apparente entre les composants d'un système. Ce problème est approfondi dans le chapitre 9.

Entre les différentes méthodes employées dans cette expérimentation, on remarque également de grandes différences. La méthode LOCALE est bel et bien la plus rapide pour toutes les configurations. Ceci est le résultat d'une évaluation incomplète de la matrice globale de dépendances **C**. Pour déterminer la

3. C'est le temps *user* d'UNIX en secondes entre le pas zéro et la fin de la simulation (75h temps simulé) sur une machine monoprocesseur.

sensibilité sur la variation de la température de l'interface entre le coefficient de EXT et l'ISOLANT, nous n'évaluons pas les modules de BÉTON et INT.

La méthode ASYNC est essentiellement une méthode LOCALE avec en plus des tâches indépendantes pour les interfaces. Cela donne plus de liberté sur l'ordre de calcul, ce qui cause des itérations supplémentaires au niveau algorithmique. Comme nous exécutons le programme sur une machine monoprocesseur, il n'y a pas de vraie distribution de calcul. Le système d'exploitation doit donc changer encore plus souvent pour attribuer le processeur aux tâches actives suite au nombre accru de tâches.



8.2 Cellule de Hambourg

Le nom de « *cellule de Hambourg* » désigne un problème bien défini de confort thermique, présenté pour la première fois à une conférence à Hambourg (d'où le nom) par le groupe RAMSES pour une démonstration de leur programme Zoom [10]. Les résultats issus de simulations de ce problème sont vérifiés à maintes reprises par d'autres logiciels comme Neptunix et Spark (voir §§ 2.3.3 et 2.3.4, [37] et [38]). C'est la raison pour laquelle nous reprenons aussi le même exemple. Il sert également à vérifier la simulation par Motor-2. En même temps, nous pouvons comparer la facilité de manipulation des différents environnements.

8.2.1 Description du problème

La cellule de Hambourg consiste en une pièce d'habitation comme le montre la figure 8.7. La pièce est excitée par des sollicitations climatiques sur sa façade extérieure sud qui comprend une fenêtre et par la température de la façade nord qui se trouve à l'intérieur de l'immeuble. Un radiateur électrique peut chauffer le local. Il est piloté par un régulateur dont la sonde se trouve à la surface intérieure du mur nord.

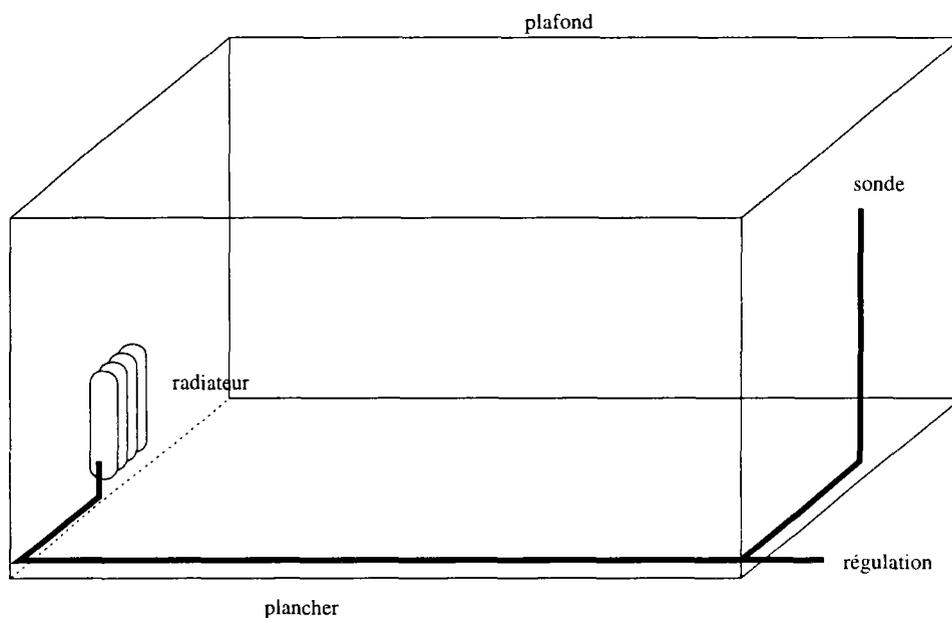


FIG. 8.7 - La cellule de Hambourg vue au niveau technique.

Passons maintenant au niveau physique de la description. Les phénomènes physiques et d'inévitables simplifications de la modélisation interviennent dans cette phase. L'intérieur de la pièce est divisé en trois parties entre lesquelles s'effectue un échange convectif forcé. Le radiateur chauffe l'air entrant et le pousse vers la partie haute. De là, l'air retourne vers la partie basse et entre de

nouveau dans le radiateur. Les deux zones d'air sont considérées comme respectivement parfaitement mélangées. Cette division en parties haute et basse a été projetée sur les murs nord et sud. Ces deux murs et la fenêtre sont modélisés par une conduction monodimensionnelle. Le plafond et le plancher de la pièce ainsi que les murs des cotés sont considérés comme adiabatiques, c'est à dire, ils ne subissent pas d'échange thermique. Le radiateur a également une capacité thermique, mais il insert une puissance de chauffage qui est déterminé par un système de régulation.

Nous spécifions encore quelques détails du problème aux niveaux mathématique et algorithmique. La conduction dans les murs et la fenêtre est traitée par une méthode aux différences finies. Pour faciliter la description de la partie haute du mur sud, les propriétés thermo-physiques de la fenêtre sont incluses par pondération à celles du mur qui la supporte. Un seul nœud capacitif est utilisé pour chacun des volumes de l'air et pour le radiateur. La régulation du chauffage fonctionne comme un thermostat linéarisé : la fonction analytique « *tangente hyperbolique* » est utilisée pour représenter le changement abrupt entre allumé – éteint.

$$P = \frac{P_{\max}}{2} \left(1 - \tanh \frac{T_s - T_c + T_b}{T_b} \right)$$

P_{\max} est la puissance maximale du chauffage, T_s la température de la sonde, T_c la température de consigne, et T_b la largeur de bande de changement. On suppose une température constante à l'extérieur du mur nord créée par des thermostats parfaits. L'excitation climatique sur le mur sud est également supposée être une température constante pour des raisons de simplicité.

Dans la simulation avec Motor-2, nous reprenons également la même structure hiérarchique qui est déjà proposé dans la première publication originale [10]. Le problème global de la pièce est divisé en trois, le mur nord, l'intérieur de la pièce avec le système de régulation, et le mur sud avec la fenêtre. Chacune des parties géométriques est encore divisée en une partie haute et basse. Dans la figure 8.8 nous voyons le problème dans le schéma de la simulation numérique par Motor-2.

Nous retrouvons dans cette figure les modules qui sont utilisés dans la description du problème. Le module principal du problème est le module composé HH. Le schéma ne commence qu'avec ses trois sous-modules NORTH, ROOM et SOUTH. Le module SOUTH contient la partie basse du mur extérieur S2 et la fenêtre S1. Les deux modules sont du type FD1D1K pour la conduction monodimensionnelle. Des coefficients d'échange SOL_S1 et SOL_S2 les connectent avec la température constante de l'extérieur.

Le module composé de milieu contient l'essentiel du problème. Nous avons trois nœuds capacitifs, le volume d'air supérieure HO, le volume d'air inférieure BA, et la capacité calorifique du chauffage CE. Ces trois modules sont du type CN. Ils sont inter-connectés par le module central HTRN qui représente l'échange convectif entre les trois parties. À un débit constant, il y a transport circulaire du radiateur vers le volume en haut, puis vers le bas et rentrant de nouveau

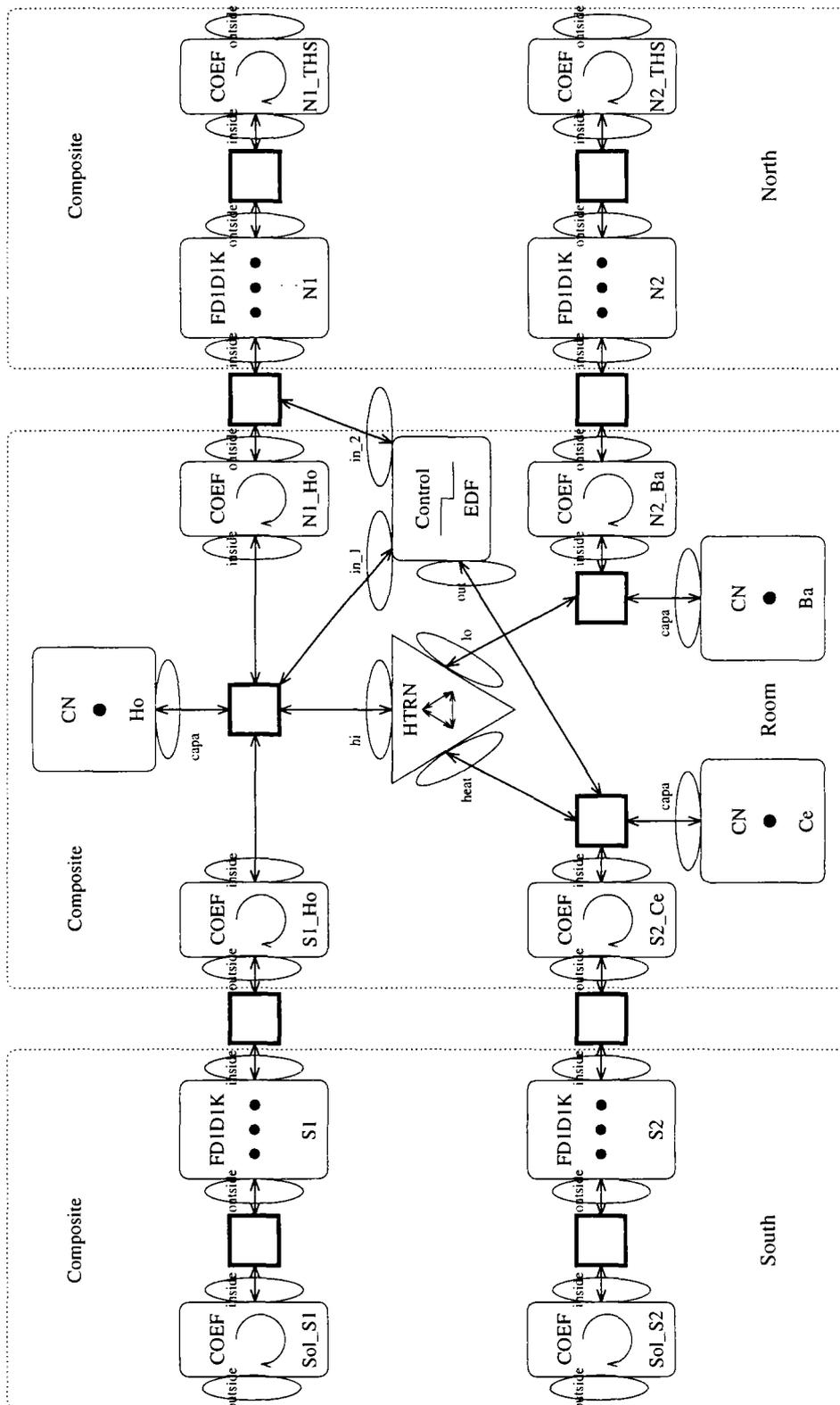


FIG. 8.8 - La cellule de Hambourg vue au niveau algorithmique.

dans le radiateur. Les trois modules sont également en échange convectif avec

les surfaces des murs sud et nord. Les modules S1_HO, N1_HO, S2_CE, et N2_BA représentent des résistances thermiques. La puissance de chauffage émise par le radiateur est déterminée par un dispositif de régulation. Le module EDF prend comme entrées les température du nœud HO et de la surface du mur N1. La moyenne des entrées est comparée avec la température consigne pour calculer le flux injecté dans le nœud CE.

Le troisième module composé du mur nord est symétrique de celui de la façade sud. Les modules N1 et N2 sont du types FD1D1K. Leur surface extérieure est connectée par des résistances thermiques N1_THS et N2_THS aux thermostats à températures constantes. Les températures constantes à l'extérieur comme à l'intérieure ne nécessitent pas de module dans *Motor-2*. Les valeurs initiales sont imposées dans un module, s'il n'y a pas de raccordement aux interfaces.

8.2.2 Résultats de la simulation et comparaison avec Spark

Nous avons utilisé des valeurs thermo-physiques qui ont déjà été utilisées dans une simulation avec *Spark* [38]. Pour simplifier les entrées, les modules de conduction monodimensionnelle ont tous les même valeurs. Elles se trouvent dans les tableaux 8.9, 8.10 et 8.11.

module		N1, N2, S1, S2	HO	BA	CE
capacité calorifique	[J/K]	$3 \times 1,2 \cdot 10^5$	24.000	12.000	10.000
conductance	[W/K]	300			

FIG. 8.9 - Les valeurs thermo-physiques des murs et des volumes d'air.

module	SOL_S1	SOL_S2	S1_HO	S2_CE	
conductance	[W/K]	36	18	24	6
module	N1_HO	N2_BA	N1_THS	N2_THS	
conductance	[W/K]	24	12	24	12

FIG. 8.10 - Les coefficients d'échange.

module	EDF	
température consigne	[K]	293
largeur de bande	[K]	3
puissance maximale	[W]	1000

FIG. 8.11 - Les paramètres de la régulation.

Les températures initiales sont de 287 K dans tous les modules. Nous déclenchons et suivons l'évolution des températures et des flux pendant 5000 secondes. Le régime permanent n'est pas encore atteint à ce moment, mais les données

disponibles de la simulation Spark s'y arrêtent. Comme pas de temps, nous avons choisi 5 secondes, car c'est le pas minimal que Neptunix détermine automatiquement. Les autres simulations faites par Spark et Zoom ont également utilisé cette valeur.

Le radiateur est donc mis en marche par la régulation afin d'atteindre la température de consigne. Il chauffe lui-même et l'air et introduit de cette façon une perturbation dynamique dans le système. La figure 8.12 montre l'évolution de la température du volume supérieur HO. Les deux autres figures 8.13 et 8.14 nous montrent l'erreur relative entre la simulation Motor-2 et Spark.

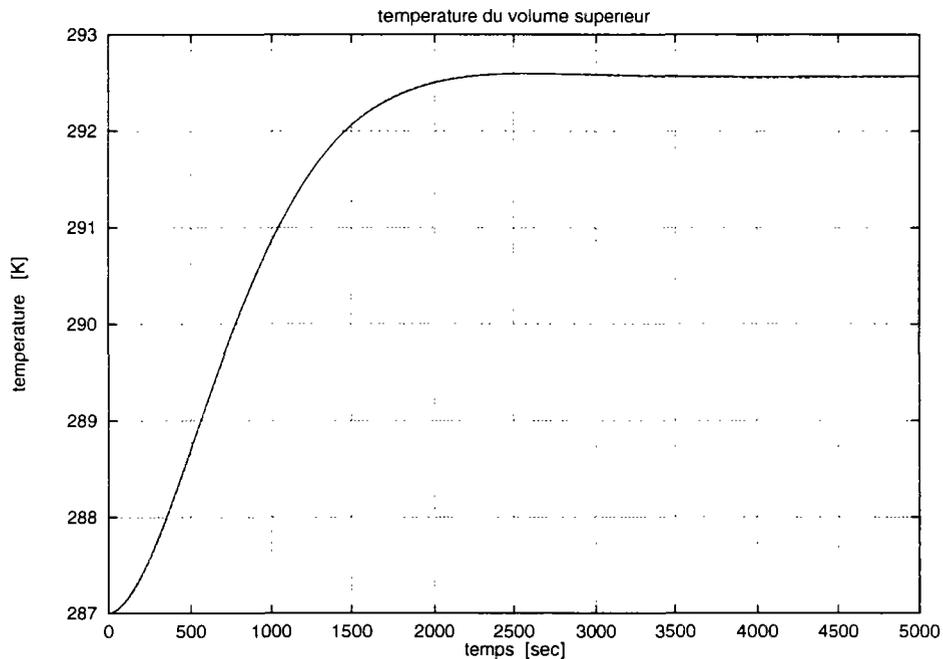


FIG. 8.12 - L'évolution de la température HO. La ligne continue provient de Motor-2, la ligne pointillés de Spark.

La température HO monte rapidement – pendant une demi-heure – et s'approche à quelques dixièmes de degrés près de la température de consigne. Environ trois quarts d'heure après le lancement, elle descend un peu et se stabilise ensuite à 292,6 K.

La superposition des résultats provenant de Motor-2 et Spark ne montre aucune différence visible (fig. 8.12). Les deux simulations donnent donc quasiment le même résultat. L'erreur relative, c'est à dire la différence entre Motor-2 et Spark normalisée est visible dans la figure 8.13. Elle est d'un ordre de $10^{-3} \%$ ce qui est la précision demandée dans Motor-2. Le graphe de la figure 8.14 montre les résultats de Motor-2 par rapport aux résultats Spark. Cela donne une droite dans la diagonale. Ce type de graphe aide à trouver des incohérences entre deux simulation en cas de problème.

L'évolution du flux de chaleur qui est introduit dans la pièce par le système de chauffage est montrée dans la figure 8.15. Il part d'une puissance presque

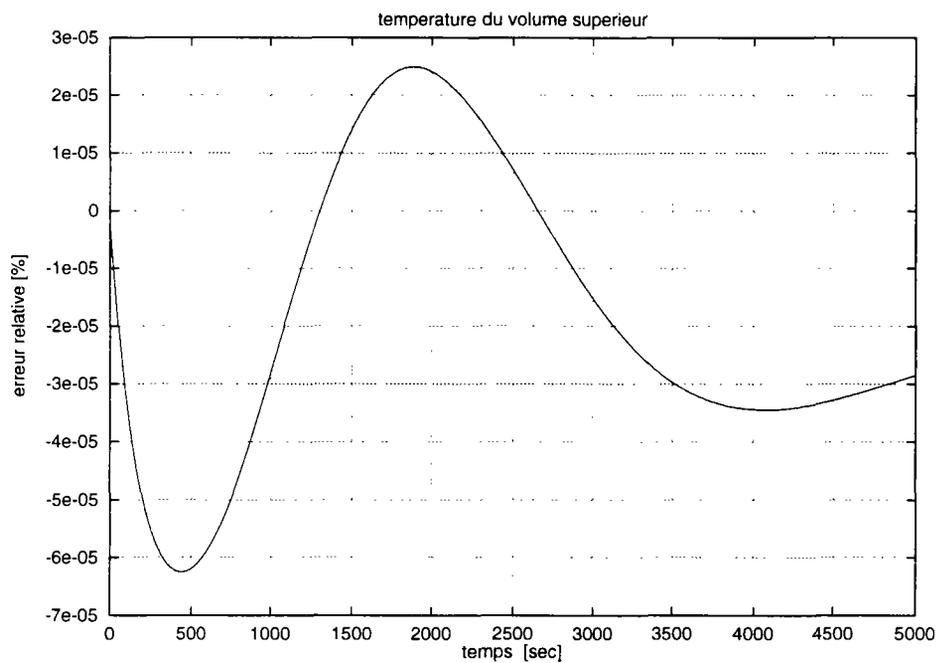


FIG. 8.13 - L'erreur relative entre Motor-2 et Spark.

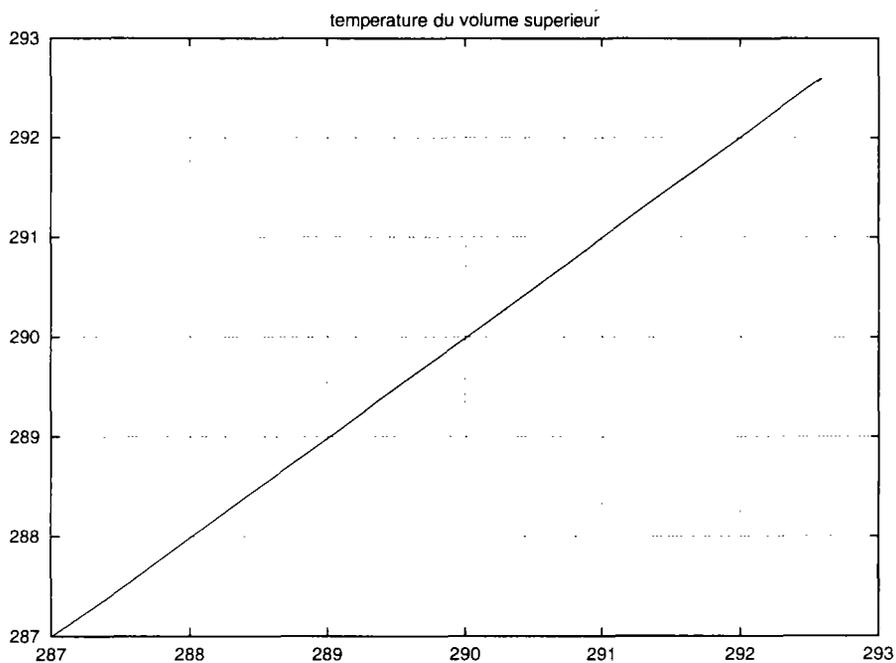


FIG. 8.14 - Le rapport Spark sur Motor-2.

maximale, et descend rapidement pendant la première demie heure et ensuite encore plus lentement. Au régime permanent (qui n'est pas montré dans cette figure), il arrive autour de 280 W.

L'erreur relative de cette variable dans la comparaison Spark - Motor-2 est

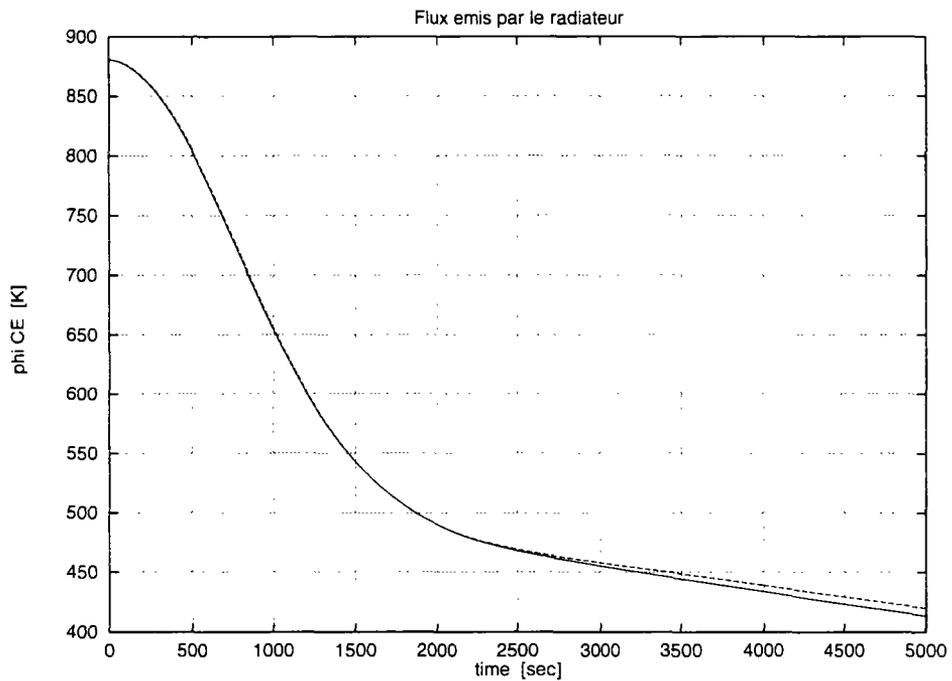


FIG. 8.15 - *Le flux introduit par le chauffage EDF.*

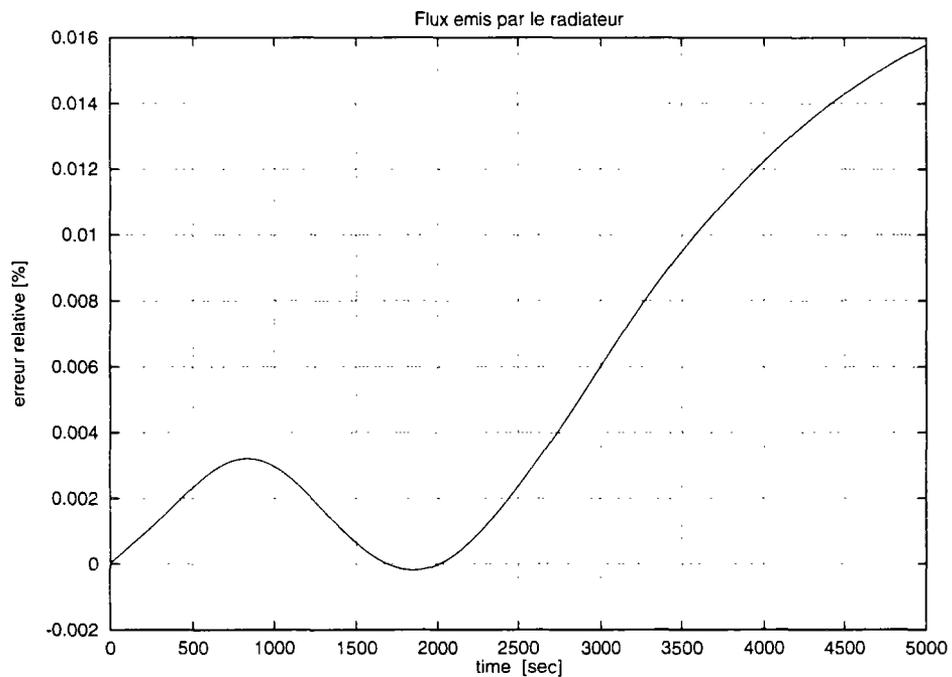


FIG. 8.16 - *L'erreur relative entre Spark et Motor-2 pour le flux de chauffage.*

un peu plus grande que pour la température du HO. Néanmoins elle reste largement dans les limites permises. D'autres variables ont été également suivies et comparées. Par exemple, les évolutions de la température du volume d'air inférieur BA et du flux de chaleur de cette partie dans le mur nord N2 se trouvent

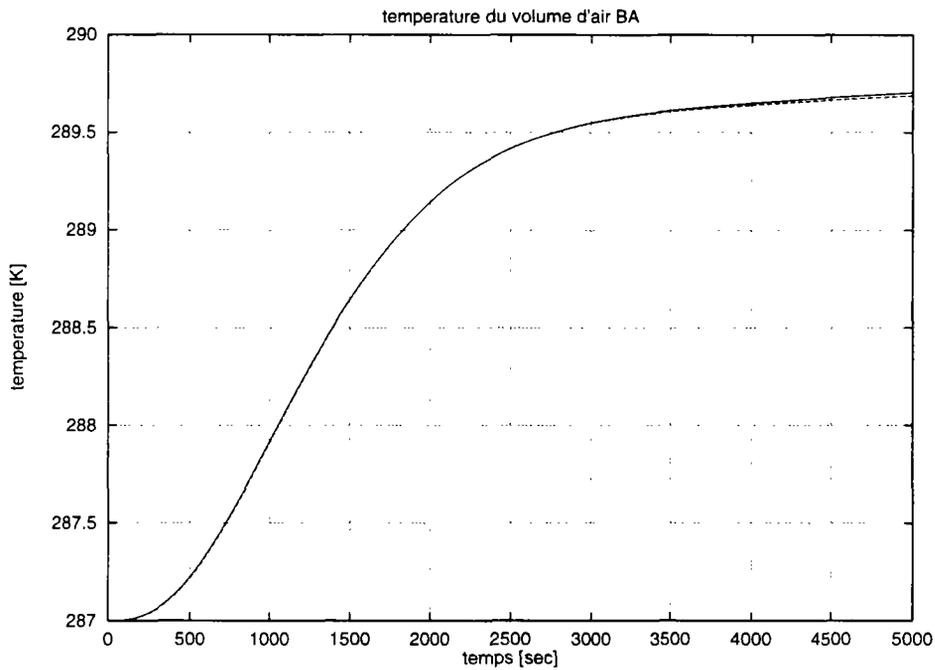


FIG. 8.17 - L'évolution de la température du module BA.

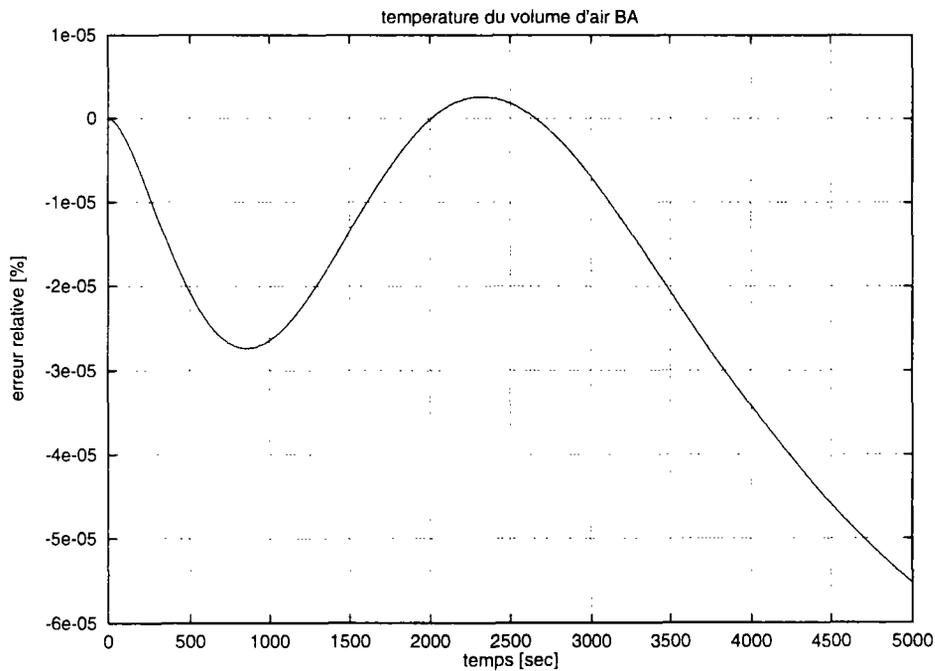


FIG. 8.18 - L'erreur relative entre Spark et Motor-2 pour la température BA.

dans les figures 8.17 et 8.19. Les figures 8.18 et 8.20 montrent les comparaisons avec les résultats correspondants de Spark.

Quelques tests simples ont été effectués pour comparer les méthodes internes de Motor-2. Pour cet exemple de la cellule de Hambourg, la méthode LOCALE

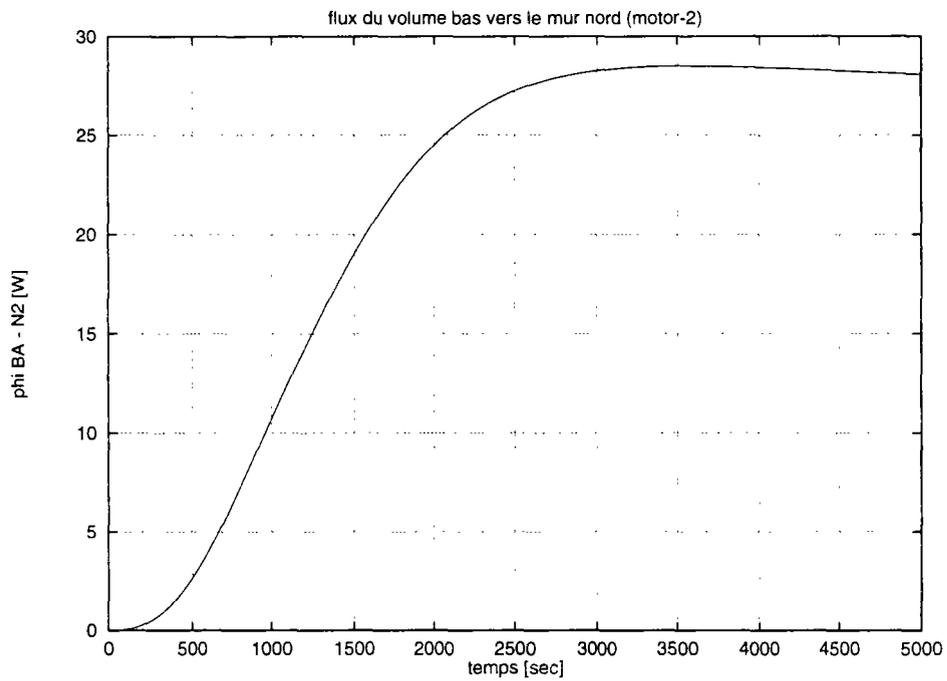


FIG. 8.19 - L'évolution du flux de chaleur entre les module BA et N2.

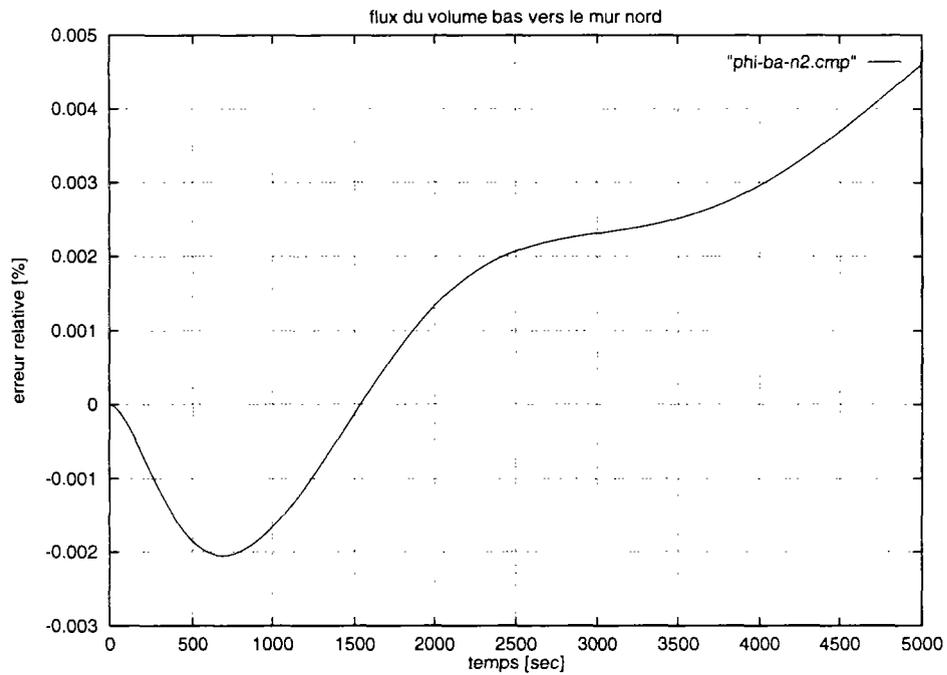


FIG. 8.20 - L'erreur relative entre Spark et Motor-2 pour le flux entre BA et N2.

de résolution indépendante par interface diminue de beaucoup le temps nécessaire de calcul. Dans la configuration décrite ici, la méthode GLOBALE met 192 secondes, la méthode LOCALE ne met que 55 secondes. C'est un facteur d'accélération de 3,5 (!).

8.2.3 Conclusion

La bonne correspondance entre les résultats obtenus par **Motor-2** et ceux obtenus par **Spark**, dans cet exemple, montre la fiabilité du programme **Motor-2**. L'erreur relative entre les deux systèmes se mesure toujours en fractions de pourcentage. La proximité des résultats est quand même surprenant compte tenu des approches très différentes de modélisation entre **Spark** et **Motor-2**. Il n'existe pas de calcul hiérarchisé dans **Spark**. toutes les équations qui constituent le problème sont écrites « à plat ». C'est ensuite l'algorithme interne qui connecte les équations et qui réduit le nombre de variables d'itération. Les modèles de **Motor-2** contiennent directement les équations orientées. Les variables de sortie et d'entrée sont définies. L'utilisateur doit spécifier des raccordements entre les modules par des interfaces.

Le même exemple a été également traité avec **Zoom** et **Neptunix** [37]. Les paramètres exactes de cette comparaison ne pouvaient pas être retrouvés, mais les différences entre les résultats obtenus par **Neptunix** et **Zoom** se situent également dans le même ordre de grandeur.

8.3 Local incluant convection libre

8.3.1 Description du problème

Il s'agit de développer un modèle de simulation des mouvements d'air dans un local. Le principe de calcul repose sur un découpage du volume d'air de la pièce en plusieurs zones pour lesquelles on écrit les bilans de masse et d'énergie. L'objectif est de proposer un modèle 3D d'échange convectif destiné à la prédiction du confort thermique dans un bâtiment. Pour la modélisation de ce problème, nous avons repris l'approche de la publication [13].

Ce découpage consiste à subdiviser un local en zones parallélépipédiques à génératrices verticales ou horizontales. Chaque zone est supposée à température homogène et à profil de pression hydrostatique. Les surfaces de contact entre ces zones sont supposées se comporter comme des parois fictives et totalement perméables. Les débits massiques entre deux zones obéissent à une loi de comportement en puissance découlant de la relation de BERNOULLI. Ils sont fonctions des pressions, des températures et de la taille des zones.

Le moteur de la convection provient de la température imposée sur les parois intérieures, le coefficient d'échange est supposé constant entre paroi et enceinte.

Dans les travaux de WURTZ, cette approche pour la convection est couplée avec d'autres phénomènes comme le rayonnement entre les surfaces de l'enceinte et la conduction dans les parois [34].

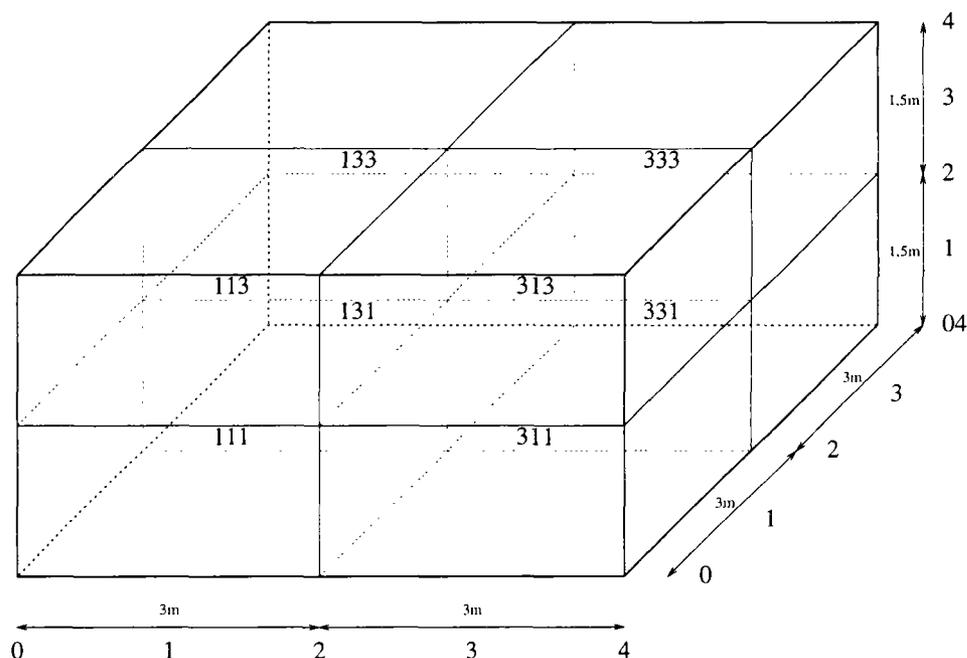


FIG. 8.21 - Le schéma de la pièce.

Nous considérons une pièce de $6 \times 6 \times 3m$ telle que présentée dans la figure 8.21. Un système de coordonnées compte de 0 à 4 dans les trois dimensions

qui servent d'indice de nœuds et de facettes. Les nœuds se trouvent toujours avec un indice impair, les facettes sur des indices pairs. Un nœud est donc localisé par ses trois coordonnées x , y et z . La cellule 131, par exemple, se trouve à l'indice 1 de x , 3 d' y , et 1 de z ; c'est le sous-volume gauche arrière bas.

8.3.2 Modélisation

Étant donné que les variations de pressions sont très faibles, on néglige la variable vitesse dans l'équation de BERNOULLI à l'intérieur des sous-volumes. Mais on écrit une équation de débit entre les deux sous-volumes en supposant une ouverture qui est la cause de la perte de charge Δp . Le type de représentation de mouvement d'air modélisé est donc celui de la figure 8.22.

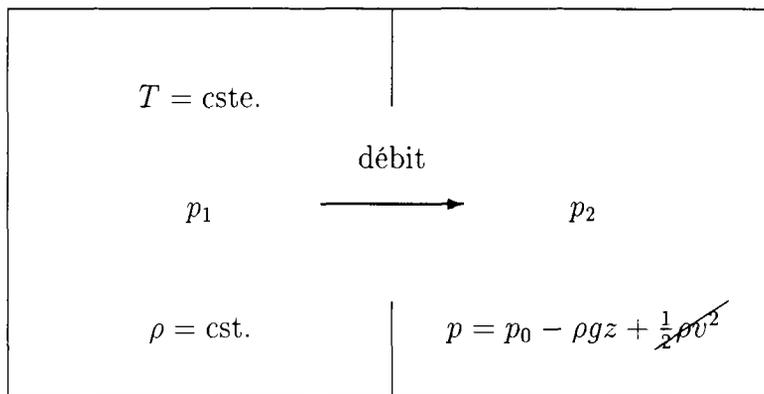


FIG. 8.22 - Modélisation de la circulation de l'air.

Au niveau des conditions aux limites, il s'agit de décrire à la fois les échanges d'air avec les parois et les « fuites » vers l'air extérieur dues aux bouches d'aération ou encore aux perméabilités de parois. Nous utilisons donc trois type de modèle :

- 1° les coefficients d'échange entre les parois et les nœuds de sous-volumes,
- 2° les équations décrivant les échanges et débits horizontaux,
- 3° les équations décrivant les échanges et débits verticaux.

Nous plaçons un module *Motor-2* entre deux nœuds de sous-volume et entre les nœuds et les températures des parois (fig. 8.23). Comme les équation de circulation de l'air nécessitent toujours des valeurs des deux nœuds connectés, l'essentiel des calculs se trouve entre les nœuds.

Développement des équations

Un sous-volume a six sous-volumes voisins, désignés par *north*, *south*, *west*, *east*, *top*, et *bottom*. L'écart de pression entre deux sous-volumes entraîne un écoulement de vitesse v_m . Nous développons un ensemble d'équations simplifiées

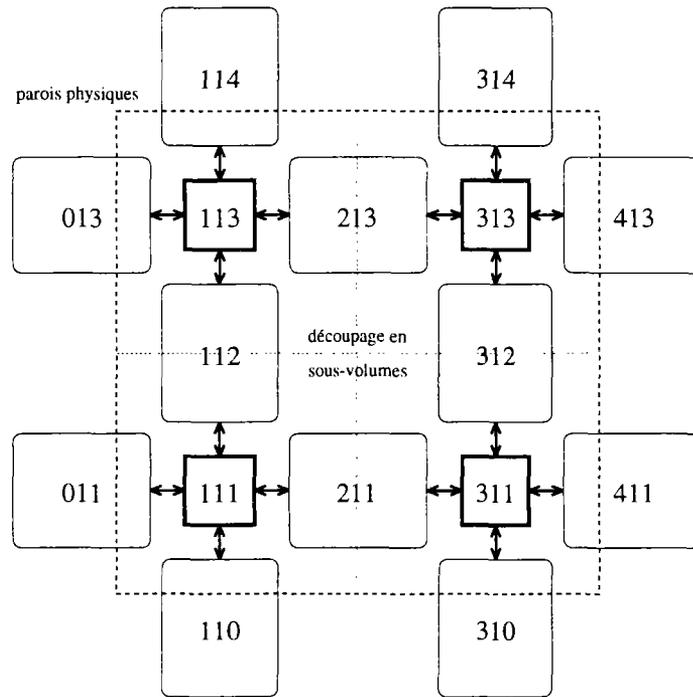


FIG. 8.23 - Les modules de Motor-2 se trouvent entre deux nœuds. Ici sont montrés les modules et les interfaces de la tranche $y = 1$.

basées sur l'équation de BERNOULLI. Elle permet d'écrire

$$p_1(z) = p_2(z) + \frac{1}{2} \rho_1 v_m^2 \quad (8.7)$$

d'où

$$\Delta p = \frac{1}{2} \rho_1 v_m^2 \quad (8.8)$$

ou encore

$$v_m = \sqrt{\frac{2}{\rho} \Delta p} \quad (8.9)$$

Or le débit massique théorique élémentaire vaut

$$d\dot{m} = \rho v l dz \quad (8.10)$$

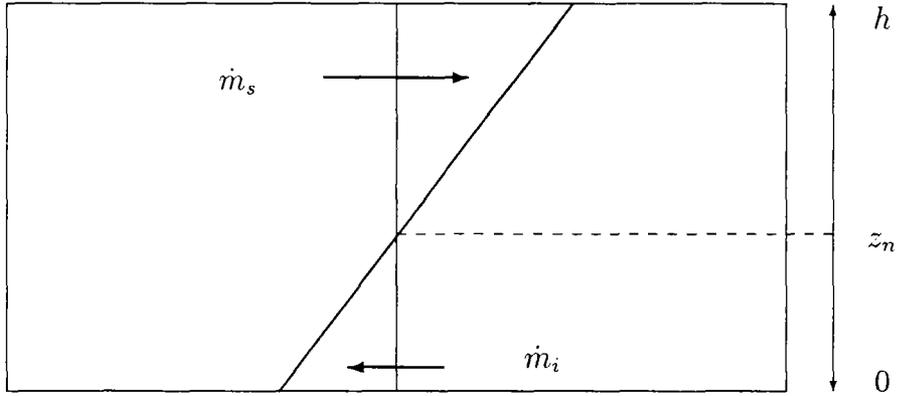
nous obtenons

$$d\dot{m} = \rho \sqrt{\frac{2}{\rho} \Delta p} l dz \quad (8.11)$$

L'équation du débit est donc de la forme

$$d\dot{m} = \rho k \Delta p^n l dz \quad (8.12)$$

où k représente la perméabilité et n est choisi en fonction du type d'écoulement. Ce sont des coefficients empiriques recherchés expérimentalement.

FIG. 8.24 - Le profil des différences de pressions Δp .

L'échange horizontal est décrit par deux flux. Ils proviennent d'un profil linéaire de différence de pression comme le montre la figure 8.24. On en déduit la cote z_n définie par $\Delta p(z_n) = 0$ et deux débits de sens opposés du part et d'autre de cet axe.

On a :

$$\Delta p(z_n) = 0 \quad (8.13)$$

d'où

$$z_n = \frac{p_1 - p_2}{(\rho_1 - \rho_2)g} \quad (8.14)$$

En intégrant \dot{m} de z_n à h (la hauteur d'un sous-volume) on obtient :

$$\dot{m}_s = kl\rho(\Delta\rho g)^n \frac{(h - z_n)^{n+1}}{n + 1} \quad (8.15)$$

puis en intégrant de 0 à z_n :

$$\dot{m}_i = kl\rho(\Delta\rho g)^n \frac{(z_n)^{n+1}}{n + 1} \quad (8.16)$$

Aux débits de masse sont couplés des flux d'enthalpie. La puissance transportée entre deux sous-volumes s'écrit donc :

$$\phi = \dot{m}_s c_{p1} T_1 - \dot{m}_i c_{p2} T_2 \quad (8.17)$$

Le débit vertical entre deux sous-volumes s'écrit de la même manière comme :

$$\dot{m} = k\rho S(p - p_t)^n \quad (8.18)$$

Aux contacts des parois, le flux est transmis par échange convectif :

$$\phi = hS\Delta T \quad (8.19)$$

Équations des cellules

L'équation des gaz parfaits nous donne une relation entre p , ρ et T dans chaque sous-volume :

$$p = \rho RT \quad (8.20)$$

Il faut également vérifier les bilans des débits

$$\sum \dot{m} = 0 \quad (8.21)$$

et les bilans des flux

$$\sum \phi = 0 \quad (8.22)$$

dans tous les nœuds.

Modèles Motor-2

Nous avons donc construit deux nouveaux modèles dans la bibliothèque Motor-2, un pour la circulation d'air horizontale CONV_HORIZ et un pour la circulation d'air verticale CONV_VERT. Les équations des bilans se prêtent bien pour les vérifications des interfaces dans Motor-2. Elles sont implémentées dans le nouveau modèle d'interface CONVECT_POINT. La figure 8.25 montre le schéma des équations dans Motor-2. Les interfaces sont placées directement sur les nœuds, les modules se trouvent toujours entre deux nœuds. Les interfaces envoient donc la température et la pression d'un sous-volume à tous les module d'échange. Ces derniers calculent les débits et les flux de chaleur avec lesquels ils répondent aux interfaces.

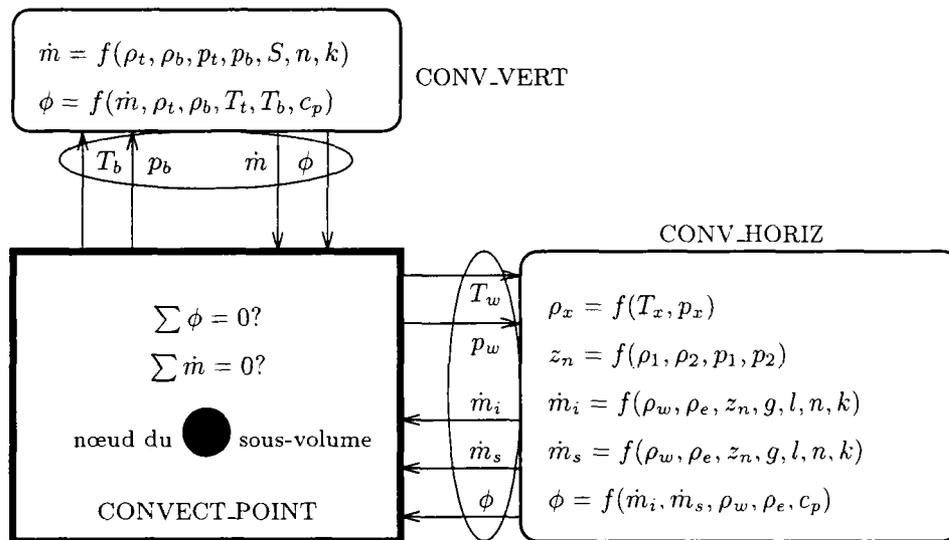


FIG. 8.25 - Organisation des équations.

Les modules prennent comme paramètres la géométrie pour déterminer la surface S et la hauteur h . Les coefficients qui spécifient le type d'écoulement et

la perméabilité sont n et k respectivement. Les autres grandeurs physiques sont les constantes R et g .

Le découpage de la pièce en huit sous-volumes crée huit nœuds et donc huit interfaces internes. Chacun des nœuds est inter-connecté à ses voisins par six modules décrivant la circulation d'air ce qui nécessite 36 modules. À ce nombre s'ajoute les 24 modules d'échange convectif entre les surfaces des parois et les nœuds de sous-volumes. Le problème entier comprend donc 60 modules, ainsi que le module principal qui décrit les connexions.

8.3.3 Simulation et résultats

Étant donné le nombre important de variables et le fait que le système est fortement non-linéaire, obtenir la convergence est loin d'être trivial. Seule la méthode GLOBALE est capable de trouver une solution. L'algorithme interne utilisé est celui de Minpack, un paquetage dédié à la solution des systèmes d'équations non-linéaires. Un autre moyen de trouver la solution est le facteur d'amortissement qui ralentit la convergence, mais qui permet d'amortir les variations brusques dans le cas d'équations non-continues (voir D.1). C'est utile ici, car lors d'un changement du sens des débits, il faut également changer les variables température et pression. En effet, pour décrire un débit ou un flux, on prend uniquement en compte les variables correspondant à l'origine du mouvement de l'air.

Dans le cadre de cet exemple, nous avons testé une configuration simple dans laquelle on connaît *a priori* l'allure de la circulation de l'air. L'objectif principal consistait à montrer la faisabilité d'un tel type de modélisation et la capacité de Motor-2 à trouver une solution. C'est la raison pour laquelle nous ne faisons qu'une simulation statique. C'est à dire que les températures des parois restent constantes, et qu'un seul « pas de temps » est donc nécessaire.

Pour notre simulation, la paroi gauche est froide (8°C) ($x = 0$), toutes les autres étant adiabatiques à la température 18°C . Les coefficients k de perméabilité sont mis à zéro pour les modules d'échange à travers les parois. Ils valent 1 à l'intérieur de la pièce. Le problème est symétrique au plan $y = 2$. Les quatre sous-volumes de devant devraient donner les mêmes résultats que les quatre sous-volumes de derrière. On ne devrait pas avoir d'échange entre les sous-volumes de devant et ceux de derrière.

Les figures 8.26 à 8.29 montrent les valeurs que nous avons obtenues dans une simulation avec Motor-2. Dans chacun des sous-volumes sont indiquées la température et la pression du nœud. L'épaisseur des flèches correspond à la « vitesse débitante » de convection⁴ Nous voyons bien une circulation d'air en sens inverse des aiguilles d'une montre dans les figures 8.26 et 8.27.

Si l'on regarde les quatre sous-volumes de gauche ($x = 1$) de la figure 8.28, nous ne voyons que des débits verticaux qui descendent des sous-volumes supérieures. Par contre, dans les sous-volumes de droite ($x = 3$) de la figure 8.29,

4. C'est le débit divisé par la surface de la facette.

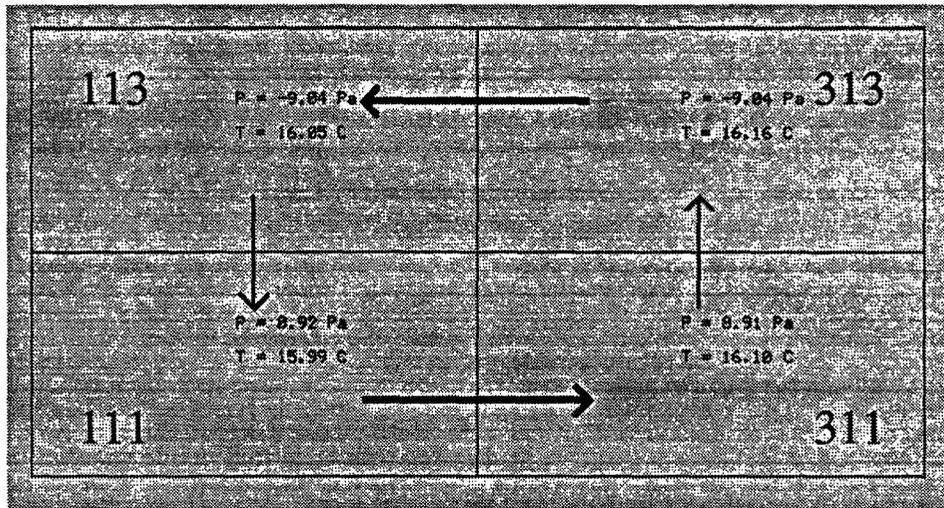


FIG. 8.26 - La convection entre les quatre sous-volumes du premier plan ($y = 1$).

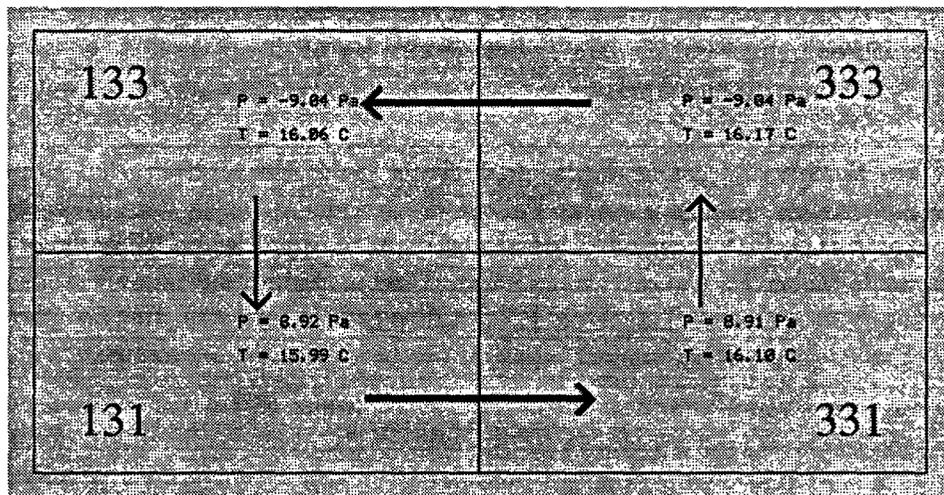


FIG. 8.27 - La convection entre les quatre sous-volumes de l'arrière plan ($y = 3$).

on remarque des échanges entre les sous-volumes de devant et celles de derrière. L'inspection des valeurs numériques montre que ces échanges sont très faibles. C'est certainement dû aux perturbations de l'algorithme de résolution. Au cours du développement de cet exemple, nous avons remarqué que le système est particulièrement sensible aux valeurs des paramètres et des limites. Un petit changement de valeurs d'entrée peut causer de graves problèmes numériques comme par exemple une divergence de calculs ou une multiplication des itérations nécessaires.

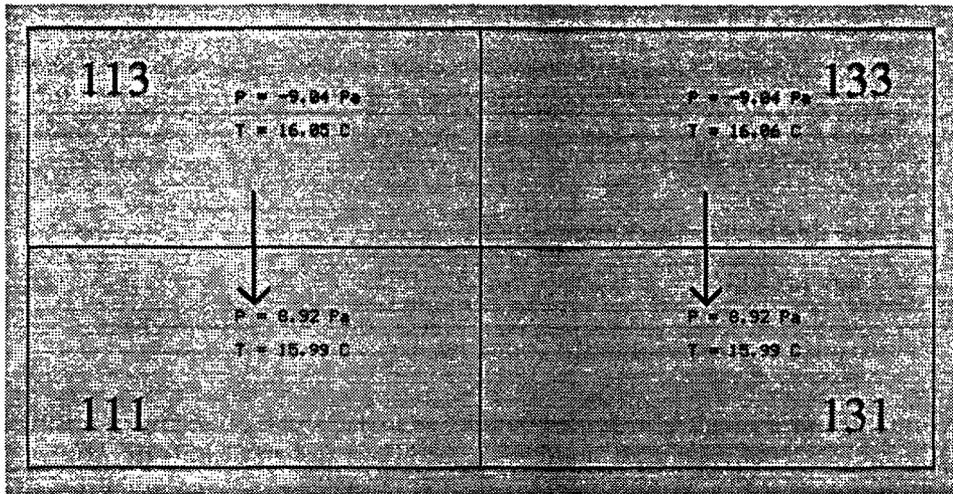


FIG. 8.28 - La convection entre les quatre sous-volumes de gauche ($x = 1$).

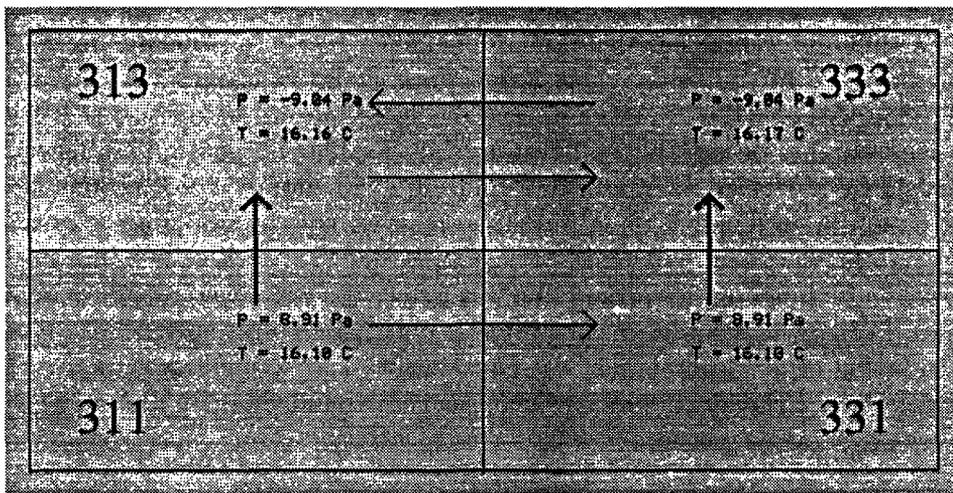


FIG. 8.29 - La convection entre les quatre sous-volumes de droite ($x = 3$).

8.3.4 Comparaison avec Spark

Le problème a été également simulé avec l'environnement Spark. Ici nous faisons aussi la distinction entre les calculs qui sont liés aux nœuds et calculs qui se trouvent entre deux nœuds. Dans des macros, on rassemble les équations correspondantes.

Pour chaque sous-volume, on écrit :

- le bilan des débits massiques,
- le bilan des flux de chaleur,

- une équation d'état,
- la variation hydrostatique

Pour chaque échange on calcule :

- le flux de chaleur entre deux zones,
- le débit massique,
- l'altitude de pression neutre (pour les échanges horizontaux)

On développe ensuite toutes les équations mises à plat, puis **Spark** réduit au maximum par substitution le nombre de variables pour faire les itérations sur un système réduit.

Les résultats obtenus avec **Spark** sont très proches des résultats de **Motor-2**. Les pressions et les débit sont les mêmes. Pour les températures, on trouve toujours un faible écart entre les deux systèmes.

La différence entre ces deux environnements réside surtout au niveau de la description du problème : dans **Spark**, on écrit les équations telle qu'on les a développées et c'est le noyau interne de **Spark** qui en déduit par réduction un petit nombre de variables d'itération. L'utilisateur ne peut qu'influencer le choix de ces variables retenues. La résolution et réduction des équations ne se font donc pas en fonction de la physique, mais en vue de minimiser le nombre de variables. Cela peut entraîner, par exemple, que l'algorithme travaille sur les masses volumiques qui ne varient que très peu ce qui pose des problèmes numériques. Dans la modélisation par **Motor-2**, l'utilisateur doit spécifier lui même l'orientation et le regroupement des équations et leur implémentation dans les modèles. Les variables d'itérations choisies correspondent donc davantage à la nature physique du phénomène.

8.3.5 Conclusion

L'ordre de grandeur des températures ainsi que de la circulation d'air correspond bien à la réalité attendue ce qui montre surtout la faisabilité de notre étude.

Dans cet exemple, nous couplons déjà deux phénomènes qui sont la circulation de l'air entre les sous-volumes et l'échange convectif entre les sous-volumes et les parois. Dans d'autres exemples, nous utilisons des modèle de rayonnement (cf. § 6.4.4) et des modèles des conduction en 1D et 2D (cf. exemples précédents et § 9.4). Une simulation qui prend en compte tous ces phénomènes semble maintenant possible. L'étude complète du comportement thermique d'un bâtiment par l'intermédiaire d'un modèle zonal couplé permettra d'une part d'homogénéiser la température dans un local ce qui réduira les consommations énergétiques et améliorera le confort, d'autre part de réduire les vitesses de l'air en particulier dans les locaux de grande hauteur.