

Optimisation aérothermique d'un bâtiment

Sahi Gonsangbeu

12 Décembre 2022

Contents

Introduction	2
Modélisation	2
Aérodynamique	2
Champ de vitesse	2
Conditions aux limites:	2
Thermique	3
Champ de température	3
Conditions aux limites :	3
Resolution numérique	3
Optimisation	4
Algorithme de descente du gradient	4
Approche par différences finies	4
Méthode de l'équation de sensibilité	5
Méthode de l'équation adjointe	6
Résultats	6
Calculs des gradients	6
Résultats de l'algorithme du gradient	7
Conclusion	11

Introduction

Dans ce projet, on souhaite déterminer les flux de chaleur R_1 (ϕ_{R_1}) et R_2 (ϕ_{R_2}) et d'air v (ϕ_0) pour obtenir une température moyenne de 19°C dans la pièce.

Dans un premier temps, on modélise le problème par des équations aux dérivées partielles qu'on va résoudre à l'aide de la méthode des éléments finis.

Dans un second temps, on utilise l'algorithme de descente du gradient pour trouver les flux de chaleurs optimaux. On s'intéressera particulièrement à trois méthodes de calcul du gradient qui sont:

- L'approche par différences finies
- La méthode de l'équation de sensibilité
- La méthode de l'équation adjointe

Modélisation

Aérodynamique

Champ de vitesse

- On suppose que le champ de vitesse dérive d'un potentiel :

la vitesse est donc donnée par:

$$\vec{U}(x, y) = \overrightarrow{\text{grad}}\phi(x, y) = \nabla\phi(x, y) \quad (1)$$

- On suppose que l'écoulement est incompressible:

$$\text{div}(\vec{U}) = 0 \quad (2)$$

Les équations (1) et (2) donnent :

$$\Delta\phi = 0 \text{ sur } (\Omega) \quad (3)$$

Conditions aux limites:

- Mur + radiateurs glissants + fenêtre: $\vec{U} \cdot \vec{n} = 0$ sur $(\Gamma_R \cup \Gamma_W \cup \Gamma_F)$ où $\Gamma_F = \Gamma_{F_N} \cup \Gamma_{F_S}$

Donc

$$\overrightarrow{\nabla}\phi \cdot \vec{n} = 0$$

- Flux sortant contrôlé : $\vec{U} \cdot \vec{n} = \phi_0$

Donc

$$\overrightarrow{\nabla}\phi \cdot \vec{n} = \phi_0 \text{ sur } (\Gamma_0)$$

- Potentiel fixé entré : $\phi = \phi_i$ sur (Γ_V)

Thermique

Champ de température

On prend en compte les effets de convection et de diffusion:

$$\underbrace{\vec{U} \cdot \vec{\nabla} T}_{Convection} = \underbrace{\mu \Delta T}_{diffusion}$$

- $T(x, y)$: champ de température
- μ : coefficient de diffusion

Conditions aux limites :

- Murs isolants: $\vec{\nabla} T \cdot \vec{n} = 0$ sur (Γ_W)
- Température fixée aux fenêtres : $T = T_{F_N}, T = T_{F_S}$
- Pas de flux sur la ventilation : $\vec{\nabla} T \cdot \vec{n} = 0$ sur (Γ_V)
- flux controlé sur les radiateurs :

$$\vec{\nabla} T \cdot \vec{n} = \phi_{R_1} \text{ sur } (\Gamma_{R_1})$$

$$\vec{\nabla} T \cdot \vec{n} = \phi_{R_2} \text{ sur } (\Gamma_{R_2})$$

Resolution numérique

Dans cette partie, on se propose de résoudre les équations ci-dessus avec la méthode des éléments finis.

Pour cela on va trouver la formulation variationnelle des équations.

On cherche ϕ vérifiant: $\int_{(\Omega)} \Delta \phi v d\Omega = 0 \quad \forall v$

On réalise une intégration par parties (IPP):

$$- \int_{(\Omega)} \vec{\nabla} T \cdot \vec{\nabla} v d\Omega + \int_{(\Gamma)} \vec{\nabla} T \cdot \vec{n} v d\Gamma = 0$$

On cherche $\phi \in H_{\Gamma_i}^1$ tel que:

$$- \int_{(\Omega)} \vec{\nabla} T \cdot \vec{\nabla} v d\Omega + \int_{(\Gamma_i)} \phi_i v d\Gamma = 0 \quad \forall v \in H_{\Gamma_i}^1$$

On cherche T vérifiant:

$$\int_{(\Omega)} [\vec{U} \cdot \vec{\nabla} T - \mu \Delta T] v d\Omega = 0$$

On réalise une IPP comme pour l'équation précédente et on obtient:

$$\int_{(\Omega)} [\vec{U} \cdot \vec{\nabla} T + \mu \vec{\nabla} T \cdot \vec{\nabla} v] d\Omega - \int_{\Gamma} \mu \vec{\nabla} T \cdot \vec{n} v d\Gamma = 0 \quad \forall v$$

On cherche $T \in H_{\Gamma_i}^1$ tel que:

$$\int_{(\Omega)} [\vec{U} \cdot \vec{\nabla} \vec{T} + \mu \vec{\Delta} \vec{T} \cdot \vec{\Delta} v] d\Omega - \int_{\Gamma_{R_1} \cup \Gamma_{R_2}} \mu \phi_R v d\Gamma = 0 \quad \forall v \in H_{\Gamma_F}^1$$

On utilise **FreeFem++** pour résoudre ces équations. Puis on va utiliser les résultats pour résoudre notre problème d'optimisation qui va consister à minimiser la fonction coût suivante:

$$J = \frac{1}{2} \int_{\Omega} (T(x, y) - T^*)^2 d\Omega$$

$T^* = 19^\circ\text{C}$: Température cible

Optimisation

On va utiliser l'algorithme de descente du gradient. nous nous intéressons donc à trois méthodes de calcul du gradients. Avant d'arriver à ces méthodes, expliquons un peu le principe générale de cet algorithme.

Algorithme de descente du gradient

Soit une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $P \mapsto f(P)$ avec $P = (a_1, \dots, a_n)$ dont on sait calculer le gradient $gradf(P)$.

L'algorithme du gradient est donnée par:

Données - Un point initial $P_0 \in \mathbb{R}$ - un niveau d'erreur $\epsilon > 0$

Itération

On calcule une suite de points $P_1, P_2, \dots \in \mathbb{R}^n$ par récurrence de la façon suivante. Supposons que l'on ait déjà obtenu le point P_k :

- On calcul $gradf(P_k)$,
- On choisit un pas δ (appelé learning rate dans le monde de l'apprentissage automatique) et on calcule

$$P_{k+1} = P_k - \delta gradf(P_k)$$

Arrêt

On s'arrête lorsque $\|(gradf(P_k))\| \leq \epsilon$.

On peut aussi choisir un nombre d'itérations (epochs dans l'apprentissage automatique).

Cette méthode dépend du calcul du gradient de la fonction qu'on cherche à minimiser. Intéressons nous donc à trois méthodes de calcul de ce gradient.

Approche par différences finies

Soit α le paramètre à optimiser: $\alpha \in \mathbb{R}$

Dans notre cas α peut-être ϕ_0 , ϕ_{R_1} et ϕ_{R_2}

$T(\alpha)$: solution de l'EDP (trouvé par la méthode des éléments finis)

On note $f(\alpha) = J(T(\alpha))$: fonctionnelle coût

1er ordre

$$\frac{df}{d\alpha} = \frac{f(\alpha + \delta\alpha) - f(\alpha)}{\delta\alpha} + O(\delta\alpha) = \frac{J(T(\alpha + \delta\alpha)) - J(T(\alpha))}{\delta\alpha} + O(\delta\alpha)$$

2e ordre

$$\frac{f(\alpha + \delta\alpha) - f(\alpha - \delta\alpha)}{2\delta\alpha} + O(\delta\alpha^2) = \frac{j(T(\alpha + \delta\alpha)) - j(T(\alpha - \delta\alpha))}{2\delta\alpha} + O(\delta\alpha^2)$$

Cette méthode est simple à mettre en oeuvre mais elle présente quelques inconvénients:

- Précision : erreur de troncature pour $\delta\alpha$ grand et erreur d'arrondi pour $\delta\alpha$ petit
- Coût CPU croît linéairement avec le nombre de paramètres n
- Il faut faire un choix à priori de $\delta\alpha$

Vu tous les inconvénients de cette méthode, en général on l'évite.

Méthode de l'équation de sensibilité

On va s'intéresser dans ce rapport seulement à l'approche continue de cette méthode.

L'idée est de propager une perturbation du paramètre α en différentiant les EDP les unes après les autres.

Illustrons cette méthode dans le cas $\alpha = \phi_0$

- Equation du potentiel:

$$\int_{\Omega} \nabla \vec{\phi} \cdot \nabla \vec{v} d\Omega + \int_{\Gamma_0} \phi_0 v d\Gamma = 0 \quad \forall v \in H_{\Gamma_i}^1$$

En dérivant formellement par rapport à α et en notant $\phi_{\alpha} = \frac{\partial \phi}{\partial \alpha}$, on obtient:

$$\int_{\Omega} \nabla \vec{\phi}_{\alpha} \cdot \nabla \vec{v} d\Omega + \int_{\Gamma_0} v d\Gamma = 0 \quad \forall v \in H_{\Gamma_i}^1$$

- Equation vitesse: $\vec{U} = \nabla \vec{\phi}$

On dérive par rapport à α et on obtient en notant $\overrightarrow{U_{alpha}} = \frac{\partial \vec{U}}{\partial \alpha}$:

$$\vec{U}_{alpha} = \nabla \vec{\phi}_{\alpha}$$

- Equation de transport de la chaleur:

$$\int_{(\Omega)} [\vec{U} \cdot \nabla \vec{T} + \mu \Delta \vec{T} \cdot \Delta \vec{v}] d\Omega - \int_{\Gamma_R} \mu \phi_R v d\Gamma = 0 \quad \forall v \in H_{\Gamma_F}^1$$

On dérive par rapport à α et on obtient en notant $T_{\alpha} = \frac{\partial T}{\partial \alpha}$:

$$\int_{(\Omega)} [\vec{U} \cdot \nabla T_{\alpha} + \mu \Delta T_{\alpha} \cdot \Delta \vec{v}] d\Omega = 0 \quad \forall v \in H_{\Gamma_F}^1$$

- Fonction coût: $J = \frac{1}{2} \int_{\Omega} (T - T^*)^2 d\Omega$

On dérive par rapport à α et on obtient:

$$\frac{\partial J}{\partial \alpha} = \int_{\Omega} (T - T^*) T_{\alpha} d\Omega$$

Méthode de l'équation adjointe

Contrairement à l'approche précédente, on part de la fonction coût:

$$J = \frac{1}{2} \int_{\Omega} (T - T^*)^2 d\Omega$$

On lui ajoute les formulations variationnelles des équations, puis on dérive par rapport à α en notant:

- $T_{\alpha} = \frac{\partial T}{\partial \alpha}$
- $\phi_{\alpha} = \frac{\partial \phi}{\partial \alpha}$
- $\phi'_0 = \frac{\partial \phi_0}{\partial \alpha}$
- $\phi'_R = \frac{\partial \phi_R}{\partial \alpha}$
- $J = \frac{\partial J}{\partial \alpha}$

on obtient:

$$\begin{aligned} J' &= \int_{\Omega} (T - T^*) T_{\alpha} d\Omega \\ &+ \int_{\Omega} \nabla \vec{\phi}_{\alpha} \cdot \nabla \vec{v}_1 d\Omega - \int_{\Gamma_0} \phi'_0 v_1 d\Gamma \\ &+ \int_{\Omega} [\nabla \vec{\phi}_{\alpha} \cdot \nabla \vec{T} v_2 + \nabla \vec{\phi} \cdot \nabla \vec{T}_{\alpha} v_2 + \mu \nabla \vec{T}_{\alpha} \cdot \nabla \vec{v}_2] d\Omega - \int_{\Gamma_R} \mu \phi'_R v_2 d\Gamma \quad \forall v_1, v_2 \in H^1_{\Gamma_i} \end{aligned}$$

On choisit v_1 et v_2 pour annuler les termes de sensibilités T_{α} et ϕ_{α} et on obtient:

soit $\lambda_2 \in H^1_{\Gamma_F}$, tel que:

$$\int_{\Omega} (T - T^*) w_2 d\Omega + \int_{\Omega} \nabla \vec{\phi} \cdot \nabla \vec{w}_2 \lambda_2 + \mu \nabla \vec{w}_2 \cdot \nabla \vec{\lambda}_2 d\Omega = 0 \quad \forall w_2 \in H^1_{\Gamma_F}$$

soit $\lambda_1 \in H^1_{\Gamma_i}$, tel que:

$$\int_{\Omega} \nabla \vec{w}_1 \cdot \nabla \vec{\lambda}_1 d\Omega + \int_{\Omega} \nabla \vec{w}_1 \cdot \nabla \vec{T} \lambda_2 d\Omega = 0 \quad \forall w_2 \in H^1_{\Gamma_i}$$

λ_1 et λ_2 sont appelées variables adjointes.

Après avoir résolu les équations adjointes et fixé λ_1 et λ_2 , on a:

$$J' = - \int_{\Gamma_0} \phi'_0 \lambda_1 d\Gamma - \int_{\Gamma_R} \mu \phi'_R \lambda_2 d\Gamma$$

Résultats

Calculs des gradients

Comme la méthode par différences finies a un coût CPU élevé, nous allons montrer les résultats de calculs de gradient seulement par les méthodes de sensibilité et adjointes.

On implémente ces méthodes à l'aide **FreeFem++** et on obtient:

Pour la méthode de sensibilité, voir figure 1

Pour la méthode de l'équation adjointe, voir figure 2

```
Gradient de J par rapport a Alpha (sensibilite phi0) : 393.436717888234
Gradient de J par rapport a Alpha (sensibilite R1) : -10.4550002511169
Gradient de J par rapport a Alpha (sensibilite R2) : -10.3692466922505
```

Figure 1: Calcul de gradient de la fonction coût par la méthode de sensibilité

```
Gradient de J par rapport a Alpha (adjointe phi0) : 410.522135985179
Gradient de J par rapport a Alpha (adjointe R1) : -10.1912655496526
Gradient de J par rapport a Alpha (adjointe R2) : -10.1923093633336
```

Figure 2: Calcul du gradient de la fonction coût par la méthode de l'équation adjointe

On obtient de valeurs de gradients sensiblement égales.

Intéressons nous maintenant à l'algorithme de descente du gradient.

Résultats de l'algorithme du gradient

Dans cette partie on va s'intéresser aux résultats obtenus en appliquant l'algorithme du gradient pour l'optimisation (ici minimisation) de notre fonction coût.

On pourra voir l'effet des paramètres suivants sur l'optimisation:

- Le pas (ou encore taux d'apprentissage)
- Le nombre d'itération (ou nombre d'epochs)
- Le calcul du gradient (adjoint ou sensibilité)
- Le point initial

On commence avec l'algorithme du gradient avec les paramètres suivant:

- le pas $\delta = 1$
- epochs = 80
- point initial ($\phi_0 = 0.5$, $\phi_{R_1} = 12$, $\phi_{R_2} = 8$)
- gradient calculé avec la méthode de l'équation adjointe

On obtient:

- fonction coût optimale: 74.7319465038974

flux de chaleur optimaux :

- $\phi_0^* = -0.699749866676842$
- $\phi_{R_1}^* = 11.4876338357527$
- $\phi_{R_2}^* = 11.1506055433064$

La figure 3 nous donne la configuration pour avoir une température moyenne à 19°C dans la pièce.

En modifiant le pas $\delta = 0.1$ au lieu de $\delta = 1$, on obtient à peu près les mêmes résultats avec une fonction coût optimale à 75.4629008889292

Augmentons le nombre d'itération (nombre d'epochs) epochs = 100 au lieu de epochs = 80 (on garde $\delta = 1$ et les valeurs initiales restent les mêmes), on obtient une fonction coût optimale à 74.6591909692114 avec à peu près les mêmes résultats pour les flux de chaleurs optimaux.

Changeons maintenant les valeurs initiales, on va partir de $\phi_0 = 0$, $\phi_{R_1} = 0$, $\phi_{R_2} = 0$ et on garde epochs = 100, $\delta = 1$, on obtient:

- fonction coût optimale: 75.892136499522

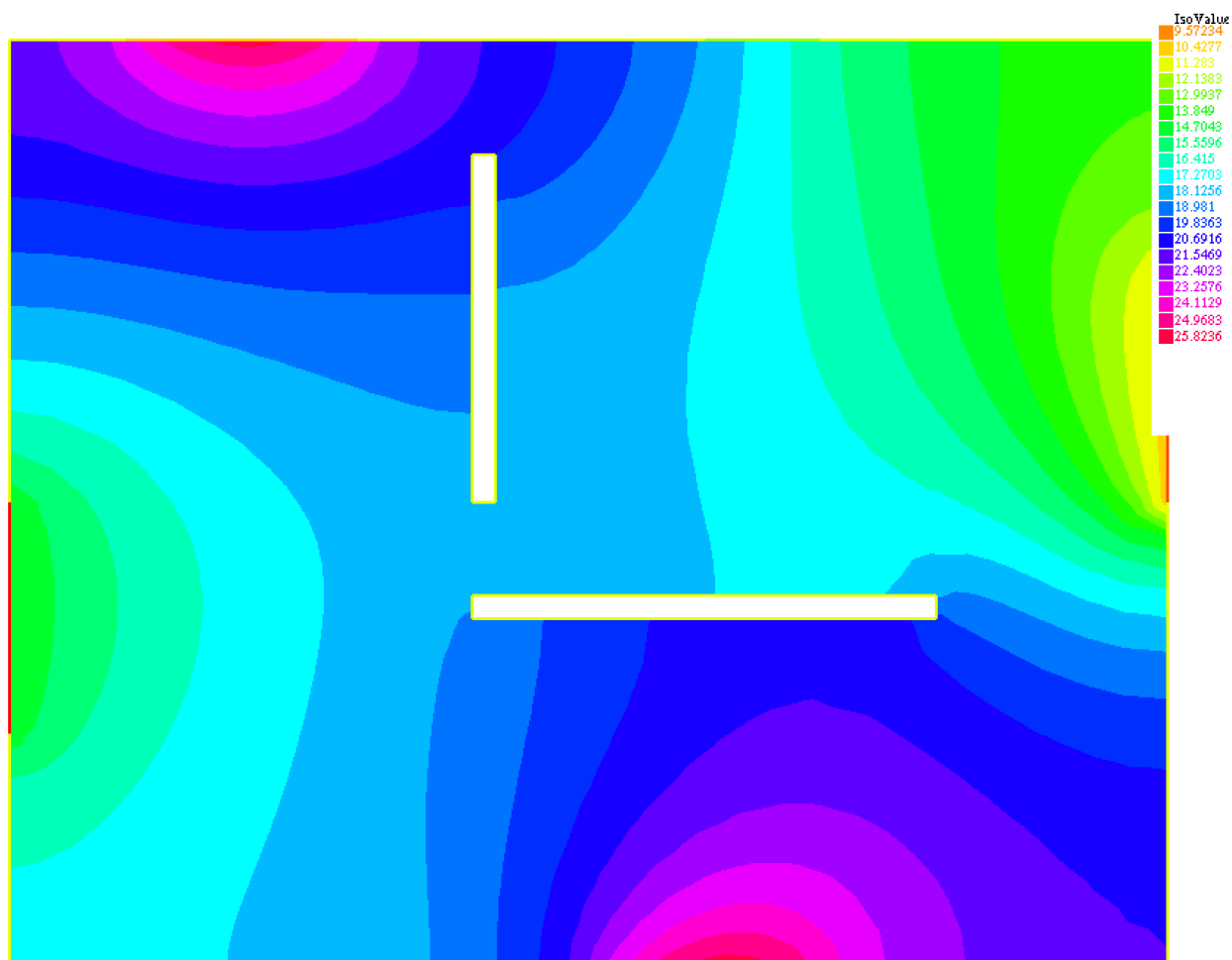


Figure 3: configuration des flux de chaleur dans la pièce pour obtenir une température moyenne proche de 19°C

flux de chaleur optimaux :

- $\phi_0^* = -0.665789519286456$
- $\phi_{R_1}^* = 8.73011116098717$
- $\phi_{R_2}^* = 11.8940297830694$

Voyons maintenant l'impact de la méthode de calcul du gradient sur les résultats

On garde les paramètres suivants:

- le pas $\delta = 1$
- epochs = 80
- point initial ($\phi_0 = 0.5$, $\phi_{R_1} = 12$, $\phi_{R_2} = 8$)

et on utilise la méthode de l'équation de sensibilité pour calculer le gradient de la fonction coût, on obtient les résultats suivants:

- fonction coût optimale: 222.764749308778

flux de chaleur optimaux :

- $\phi_0^* = -3.89692438401261$
- $\phi_{R_1}^* = 12.1158834230565$
- $\phi_{R_2}^* = 8.1168417772155$

La fonction coût optimale est moins bonne qu'avec l'autre méthode de calcul du gradient.

La figure 4 nous donne la configuration pour avoir une température moyenne à 19°C dans la pièce lorsqu'on a utilisé la méthode de l'équation de sensibilité pour le calcul du gradient.

Le calcul du gradient avec la méthode de l'équation de sensibilité semble donner de moins bons résultats par rapport au calcul avec la méthode de l'équation adjointe.

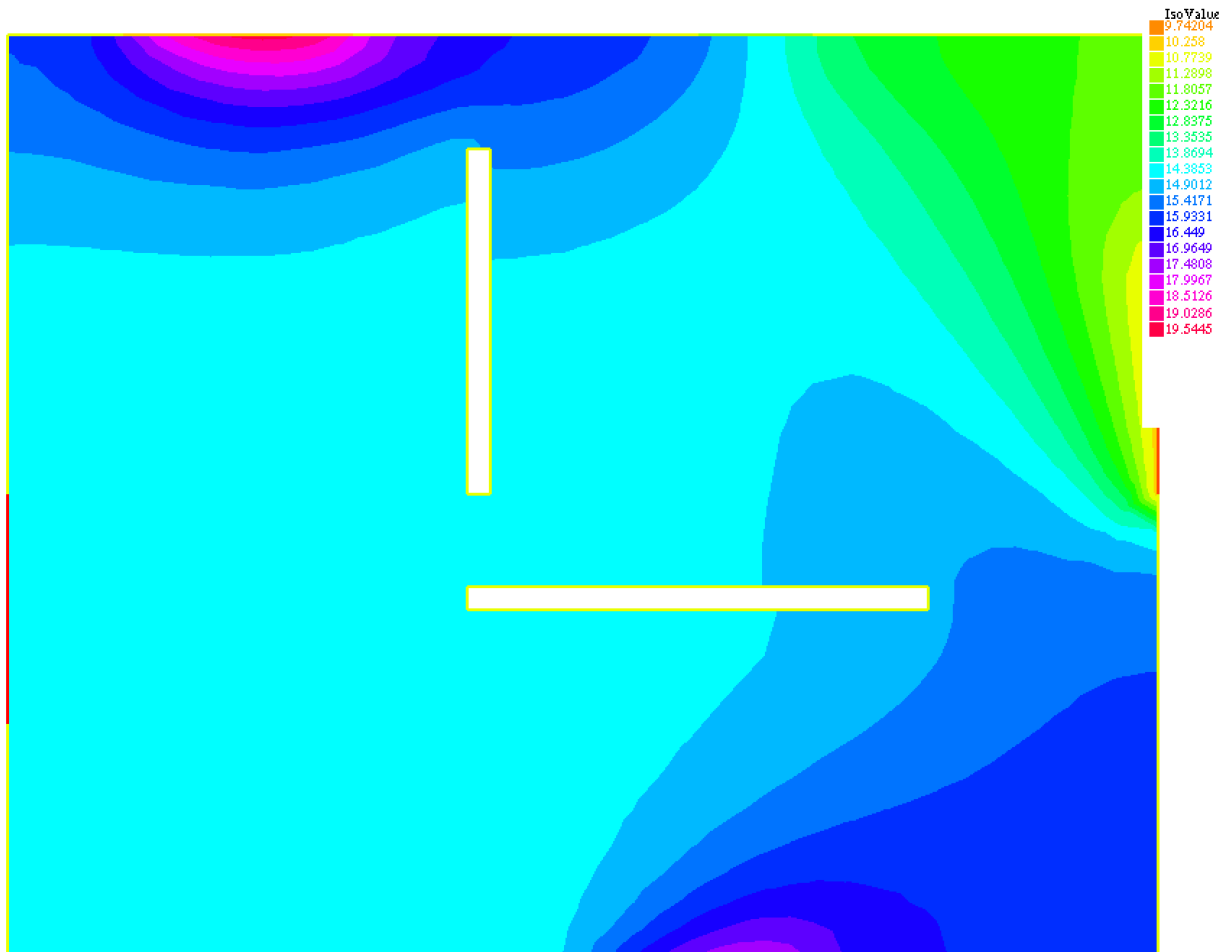


Figure 4: configuration pour avoir une température moyenne à 19°C dans la pièce (méthode de l'équation de sensibilité utilisée pour le calcul du gradient)

Conclusion

Dans ce rapport, nous avons résolu un problème simple d'optimisation couplée à une étude thermique et une étude dynamique. On a pu voir que l'algorithme de descente du gradient peut aider dans la phase d'optimisation mais il faut faire attention à la méthode de calcul du gradient de la fonction coût et à d'autres paramètres comme le pas δ (learning rate en Machine Learning) et le nombre d'itération qui peuvent avoir un impact sur la qualité des résultats de l'optimisation. On pourra par la suite utiliser d'autres algorithmes d'optimisation pour voir s'ils apportent des améliorations.