

Modélisation hygrothermique et simulation

Soutenance du stage

M1 CSMI

Université de Strasbourg

Thomas Saigre

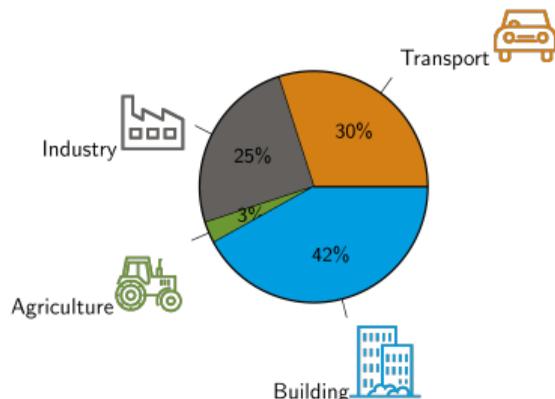
dirigé par

Vincent Chabannes, Zohra Djatouti,
Romain Hild & Christophe Prud'homme

26 août 2020

Introduction

- ▶ Selon le GIEC¹, la planète se réchauffe, et cela est en partie dû à l'activité humaine
- ▶ Le secteur qui consomme le plus est le bâtiment²



- ▶ Des campagnes ont été mises en place pour la transformation des bâtiments (Green Deal...)

¹Intergovernmental Panel on Climate Change. 2019.

²EDF. *Le bâtiment, premier poste de consommation.*

Stage

- ▶ Stage de 2 mois à Cemosis, dans le projet *4fast-sim ibat*, qui développe des outils rapides et efficaces pour identifier les potentielles sources d'énergies à sauver dans des bâtiments existants.
- ▶ Collaboration entre Cemosis et l'entreprise Synapse Concept
- ▶ *Ibat* est une plateforme qui récolte des données de températures dans des bâtiments existants
- ▶ But du stage : étudier et modéliser les transferts de chaleurs et d'humidité dans les bâtiments
- ▶ L'objectif principal est de faire rejoindre les données récoltées avec les modèles
- ▶ C'est la suite du projet effectué pendant le second semestre

Table des matières

Introduction

Résultats du projets

Développement de l'application

Benchmark : drying-out of a layer

Benchmark BESTEST

Hybridized Discontinuous Galerkin

Confort thermique

Conclusion

Résultats du projets

Petits rappels

- ▶ Étude des modes de transfert d'humidité dans une paroi (suction capillaire, diffusion de vapeur)
- ▶ Transferts thermiques (conduction...)

Petits rappels

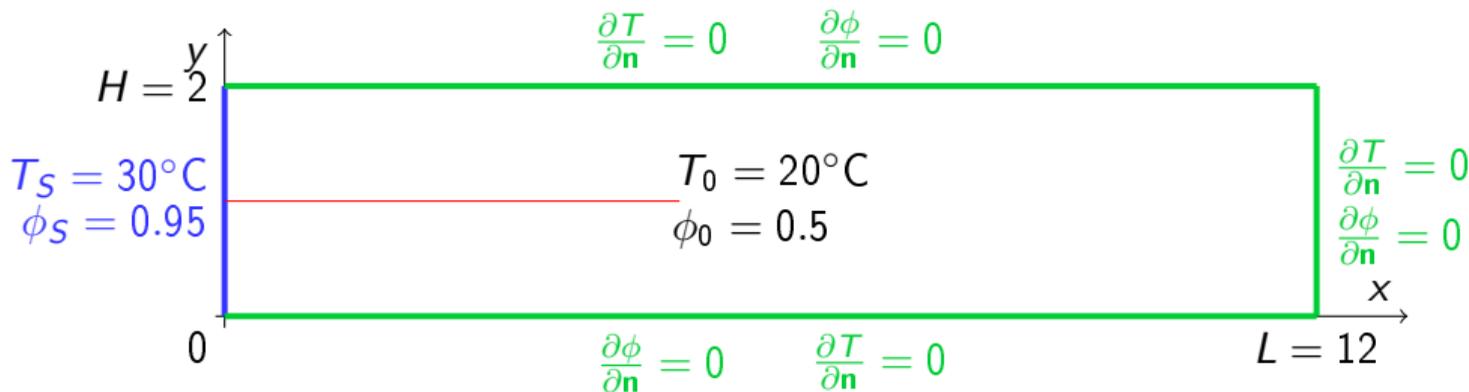
- ▶ Système d'équations à résoudre :

$$\underbrace{(\rho C_p)_{\text{eff}} \frac{\partial T}{\partial t}}_{\text{variation d'énergie}} + \nabla \cdot \left(\underbrace{-k_{\text{eff}} \nabla T}_{\text{conduction}} - \underbrace{L_v \delta_p \nabla(\phi p_{\text{sat}})}_{\text{évaporation}} \right) = Q \quad (E_T)$$

$$\underbrace{\xi \frac{\partial \phi}{\partial t}}_{\text{variation d'énergie}} + \nabla \cdot \left(\underbrace{-\xi D_w \nabla \phi}_{\text{capillarité}} - \underbrace{\delta_p \nabla(\phi p_{\text{sat}})}_{\text{Diffusion de vapeur}} \right) = G \quad (E_\phi)$$

- ▶ ϕ [-] : humidité relative
- ▶ T [K] : température
- ▶ Résolution d'un modèle (benchmark moisture uptake) avec Feel++.

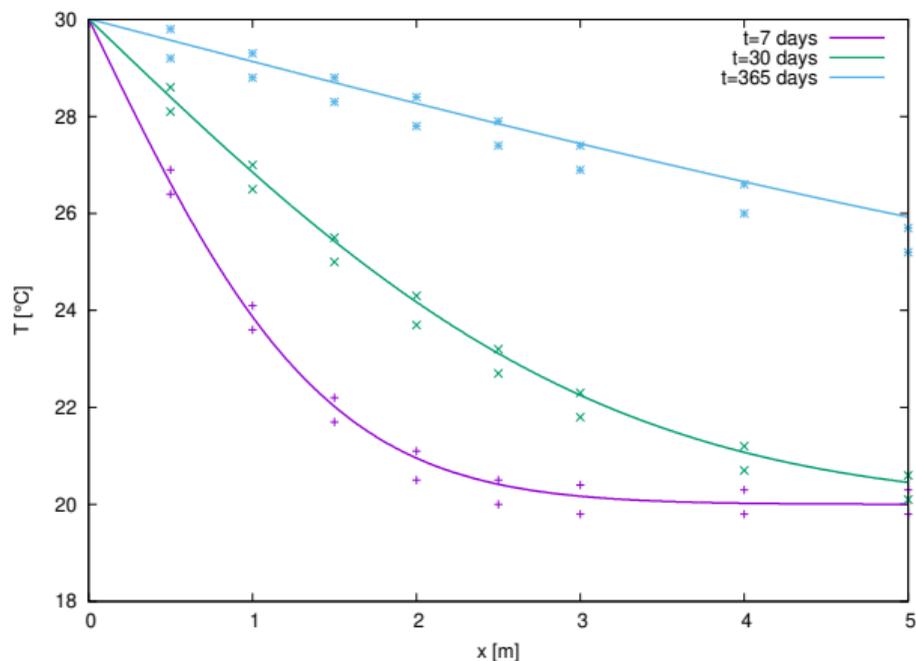
Un petit problème



Simulation faite sur 1 an, avec un pas de maillage valant 0.1

Un petit problème : Température

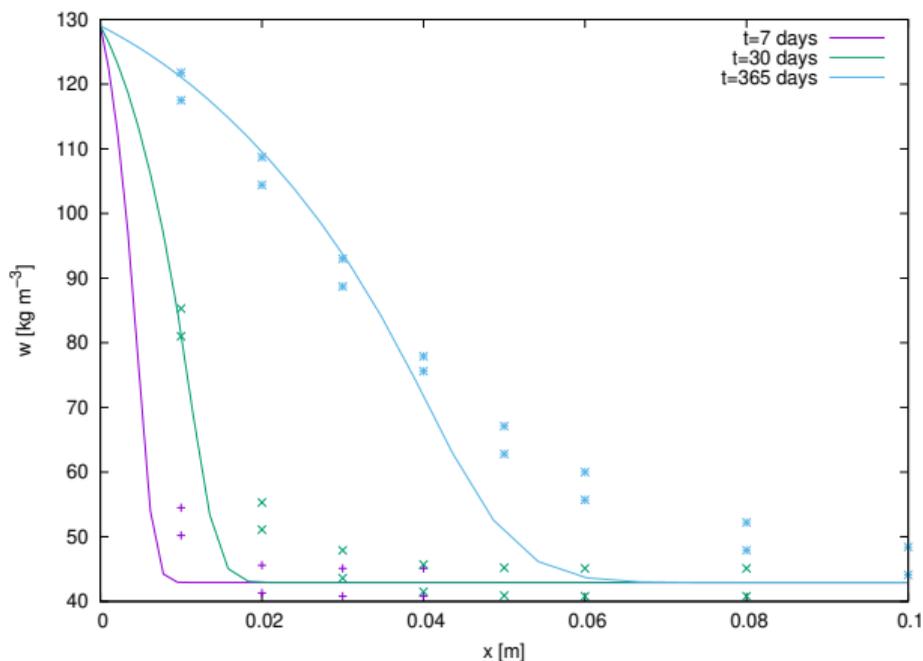
On avait obtenu ces résultats³ :



³CEN. *Hygrothermal performance of building components and building elements*. 2007.

Un petit problème : Quantité d'eau

On avait obtenu ces résultats³ :



³CEN. *Hygrothermal performance of building components and building elements*. 2007.

La solution

Il y avait une erreur dans une formule du modèle⁴ !

$$\delta_p = \frac{M_w}{RT_{ref}} \frac{26.1 \times 10^{-6}}{200} \frac{1 - \frac{w}{146}}{0.503 \left(1 - \frac{w}{146}\right)^2 + 0.497}$$

⁴Škerget, L. and Tadeu, Antonio. “BEM numerical simulation of coupled heat and moisture flow through a porous solid”. In: *Engineering Analysis with Boundary Elements* 40 (Mar. 2014), pp. 154–161. DOI: [10.1016/j.enganabound.2013.12.006](https://doi.org/10.1016/j.enganabound.2013.12.006).

La solution

Il y avait une erreur dans une formule du modèle⁴ !

$$\delta_p = \frac{M_w}{R_{H_2O} T_{ref}} \frac{26.1 \times 10^{-6}}{200} \frac{1 - \frac{w}{146}}{0.503 \left(1 - \frac{w}{146}\right)^2 + 0.497}$$

⁴Škerget, L. and Tadeu, Antonio. “BEM numerical simulation of coupled heat and moisture flow through a porous solid”. In: *Engineering Analysis with Boundary Elements* 40 (Mar. 2014), pp. 154–161. DOI: [10.1016/j.enganabound.2013.12.006](https://doi.org/10.1016/j.enganabound.2013.12.006).

La solution

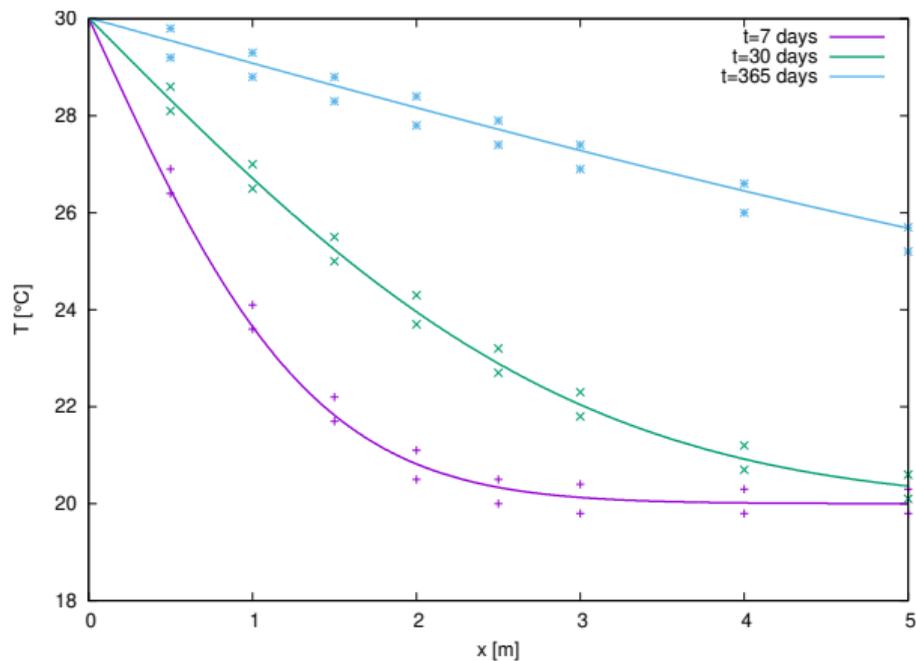
Il y avait une erreur dans une formule du modèle⁴ !

$$\delta_p = \frac{M_w}{R_{H_2O} T_{ref}} \frac{26.1 \times 10^{-6}}{200} \frac{1 - \frac{w}{146}}{0.503 \left(1 - \frac{w}{146}\right)^2 + 0.497}$$

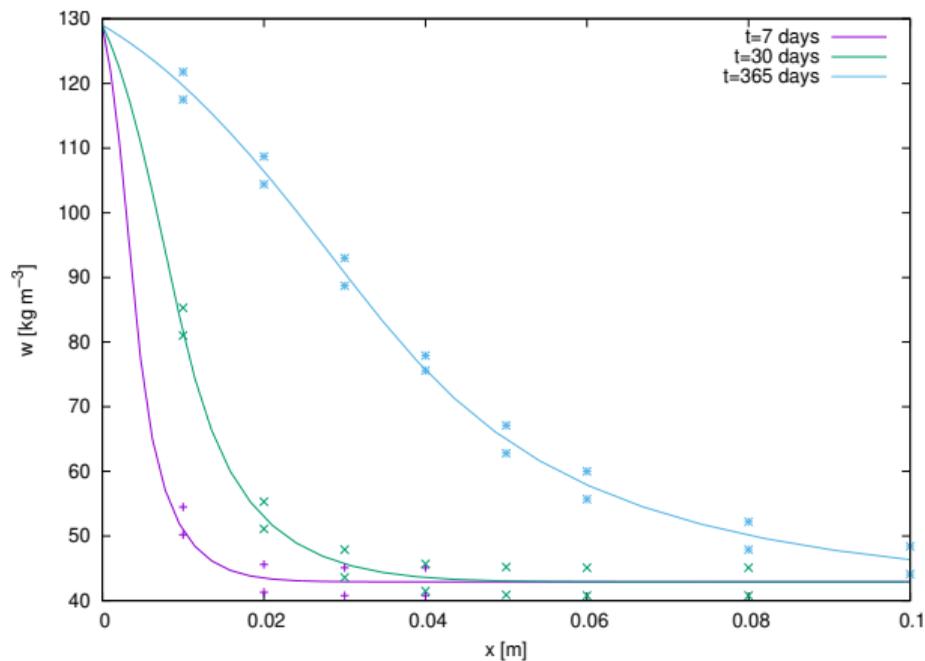
En fait $R_{H_2O} = \frac{R}{M_w}$, où R est la constante des gaz parfaits

⁴Škerget, L. and Tadeu, Antonio. "BEM numerical simulation of coupled heat and moisture flow through a porous solid". In: *Engineering Analysis with Boundary Elements* 40 (Mar. 2014), pp. 154–161. DOI: [10.1016/j.enganabound.2013.12.006](https://doi.org/10.1016/j.enganabound.2013.12.006).

La solution : Temperature



La solution : Quantité d'eau



Développement de l'application

Problème variationnel complet

Sur Ω :

$$(\rho C_p)_{\text{eff}} \frac{\partial T}{\partial t} + \nabla \cdot (-k_{\text{eff}} \nabla T - L_v \delta_p \nabla (\phi p_{\text{sat}})) = Q \quad (E_T)$$

$$\xi \frac{\partial \phi}{\partial t} + \nabla \cdot (-\xi D_w \nabla \phi - \delta_p \nabla (\phi p_{\text{sat}})) = G \quad (E_\phi)$$

Avec ces conditions aux bords, en écrivant $\partial\Omega = \Gamma_{T,D} \sqcup \Gamma_{T,N}$ et $\partial\Omega = \Gamma_{\phi,D} \sqcup \Gamma_{\phi,N}$:

$$\begin{array}{ll} T = d_T & \text{sur } \Gamma_{T,D} \\ \phi = d_\phi & \text{sur } \Gamma_{\phi,D} \\ \nabla T \cdot n = N_T & \text{sur } \Gamma_{T,N} \\ \nabla \phi \cdot n = N_\phi & \text{sur } \Gamma_{\phi,N} \end{array}$$

Un problème non linéaire

Les coefficients dépendent de T et ϕ , par exemple⁴ :

$$\xi = \frac{\partial}{\partial \phi} \frac{146}{\left(1 + (-8 \times 10^{-8} \cdot R_{\text{H}_2\text{O}} T_{\text{ref}} \rho_w \ln(\phi))^{1.6}\right)^{0.375}}$$
$$\delta_p = \frac{M_w}{RT_{\text{ref}}} \frac{26.1 \times 10^{-6}}{200} \frac{1 - \frac{w}{146}}{0.503 \left(1 - \frac{w}{146}\right)^2 + 0.497}$$

⁴CEN. *Hygrothermal performance of building components and building elements*. 2007.

Stratégies de résolution

On fait les approximations à l'ordre 1:

$$\frac{\partial X}{\partial t} \approx \frac{X^{n+1} - X^n}{\Delta t}$$

avec $X^n = X(t = n\Delta t)$.

Extrapolation du même ordre pour les coefficients :

$$c^{n+1} \approx c^n$$

Problème variationnel

Soient v une fonction test associée à T et q une fonction test associée à ϕ . Le problème variationnel au temps $n + 1$ est alors (après calculs) :

Trouver $(T^{n+1}, \phi^{n+1}) \in X \times M$ tel que $\forall (v, q) \in X \times M$:

$$a(T^{n+1}, v) + b(\phi^{n+1}, v) = f(v)$$

$$d(\phi^{n+1}, q) = g(q)$$

Avec :

$$\blacktriangleright a(T^{n+1}, v) = \int_{\Omega} (\rho C_p)_{\text{eff}}^n \frac{T^{n+1}}{\Delta t} v + \int_{\Omega} (k_{\text{eff}}^n \nabla T^{n+1}) \nabla v$$

$$\blacktriangleright b(\phi^{n+1}, v) = \int_{\Omega} L_v^n \delta_p^n p_{\text{sat}}^n \nabla \phi^{n+1} \nabla v + \int_{\Omega} L_v^n \delta_p^n \nabla p_{\text{sat}}^n \phi^{n+1} \nabla v - \int_{\Gamma_{\phi, N}} (L_v^n \delta_p^n \nabla p_{\text{sat}}^n \phi^{n+1} \cdot \mathbf{n}) v$$

$$\blacktriangleright f(v) = \int_{\Omega} \left((\rho C_p)_{\text{eff}}^n \frac{T^n}{\Delta t} + Q^{n+1} \right) v + \int_{\Gamma_{T, N}} k_{\text{eff}}^n N_T^{n+1} v + \int_{\Gamma_{\phi, N}} L_v^n \delta_p^n \nabla p_{\text{sat}}^n N_{\phi}^{n+1} v$$

Problème variationnel

Soient v une fonction test associée à T et q une fonction test associée à ϕ . Le problème variationnel au temps $n + 1$ est alors (après calculs) :

Trouver $(T^{n+1}, \phi^{n+1}) \in X \times M$ tel que $\forall (v, q) \in X \times M$:

$$a(T^{n+1}, v) + b(\phi^{n+1}, v) = f(v)$$

$$d(\phi^{n+1}, q) = g(q)$$

Avec :

$$\begin{aligned} \blacktriangleright d(\phi^{n+1}, q) &= \int_{\Omega} \xi^n \frac{\phi^{n+1}}{\Delta t} q + \int_{\Omega} \xi^n D_w^n \nabla \phi^{n+1} \nabla q + \int_{\Omega} \delta_p^n \nabla \phi^{n+1} p_{\text{sat}}^n \nabla q + \\ &\int_{\Omega} \delta_p^n \phi^{n+1} \nabla p_{\text{sat}}^n \nabla q - \int_{\Gamma_{N,\phi}} (\delta_p^n \phi^{n+1} \nabla p_{\text{sat}}^n \cdot \mathbf{n}) q \end{aligned}$$

$$\blacktriangleright g(q) = \int_{\Omega} \left(\xi^n \frac{\phi^n}{\Delta t} + G^{n+1} \right) q + \int_{\Gamma_{N,\phi}} (\xi^n D_w^n + \delta_p^n p_{\text{sat}}^n) N_{\phi}^{n+1} q$$

Problème variationnel

Soient v une fonction test associée à T et q une fonction test associée à ϕ . Le problème variationnel au temps $n + 1$ est alors (après calculs) :

Trouver $(T^{n+1}, \phi^{n+1}) \in X \times M$ tel que $\forall (v, q) \in X \times M$:

$$\begin{aligned} a(T^{n+1}, v) + b(\phi^{n+1}, v) &= f(v) \\ d(\phi^{n+1}, q) &= g(q) \end{aligned}$$

C'est un système linéaire triangulaire :

1. On commence par résoudre $d(\phi^{n+1}, q) = g(q)$, ce qui donne ϕ^{n+1}
2. On résoud $a(T^{n+1}, v) = f(v) - b(\phi^{n+1}, v)$, qui donne T^{n+1}

Développement de l'application

- ▶ Utiliser les `ModelProperties`⁵ de Feel++ pour permettre l'utilisation des fichiers JSON.

⁵*Feel++ Tutorial Dev.*

<http://docs.feelpp.org/tutorial-dev/latest/12-CreateModels.html>. 2020.

Développement de l'application

- ▶ Utiliser les ModelProperties⁵ de Feel++ pour permettre l'utilisation des fichiers JSON.
- ▶ Entrer la valeur des coefficients directement dans le fichier
`"xi": "854.271 / (phi * pow(1-8.47458*log(phi) , 1.869)):phi"`
- ▶ Choisir quelles équations simuler sur quels domaines (heat, moisture ou heat-moisture)
- ▶ Appliquer les condition initiales et aux bords (Dirichlet et Neumann)
- ▶ Choisir quelles données exporter

⁵*Feel++ Tutorial Dev.*

<http://docs.feelpp.org/tutorial-dev/latest/12-CreateModels.html>. 2020.

Exemple de fichier JSON

```
1 {
2   "Name": "moisture-uptake",
3   "Parameters":
4   {
5     "G": 0,
6     ...
7   },
8   "Materials":      physique(s) utilisee(s) et valeurs des parametres
9   {
10    "Omega":
11    {
12      "physics": ["moisture"],      physique utilisee
13      "markers": ["Omega"],
14      "w": "116. / pow(1 - log(phi)/0.118,.869):phi",
15      "xi": "854.271 / (phi * pow( 1-8.47458*log(phi) , 1.869) ):phi",
16    }
17  },
```

```
18   "BoundaryConditions":
19   {
20       "phi":                champs
21       {
22           "Dirichlet":      type de condition
23           {
24               "Gamma_Di":   marqueurs
25               {
26                   "expr":0.65  valeur sur les marqueurs
27               },
28           },
29       }
30   },
31   "InitialConditions": {...}, idem
32   "PostProcess":
33   {
34       "Exports":            Export des champs vers paraview
35       {
36           "fields":["phi","w"]
37       }
38   }
39 }
```

Exemple de code

```
1 for (auto const& [bcid, bc] : this->props_.boundaryConditions2().flatten()) {
2     if (bcid.type() == "Neumann") {
3         if (bcid.field() == "T") // for heat
4             this->l_ += integrate(
5                 _range = markedfaces( this->mesh(), bc.markers() ),
6                 _expr = idv(this->coeff( "k_eff" )) * expr(bc.expression()) *
7                     id( T_ ) );
8         if ( bcid.field() == "phi" && (mat.hasPhysics( "moisture" ) || mat.
9             hasPhysics( "heat-moisture" )) ) { // for moisture
10            this->l_ += integrate(
11                _range = markedfaces( this->mesh(), bc.
12                    markers() ),
13                _expr = ... );
14        }
15    }
```

Exemple de code

```
17 for (auto const& [bcid, bc] : this->props_.boundaryConditions2().flatten())
18 {
19     if (bcid.type() == "Dirichlet" && bcid.field() == "T")
20     {
21         Feel::cout << "Apply Dirichlet on T : " << bc.markers() << " " << bc.
                expression() << std::endl;
22         auto Ts = expr( bc.expression() ) ;
23         Ts.setParameterValues({"t",t}) ;
24         this->a_ += on( _range=markedfaces(this->mesh(), bc.markers()), _rhs=
                this->l_, _element=T_, _expr=Ts) ;
25     }
26
27 }
```

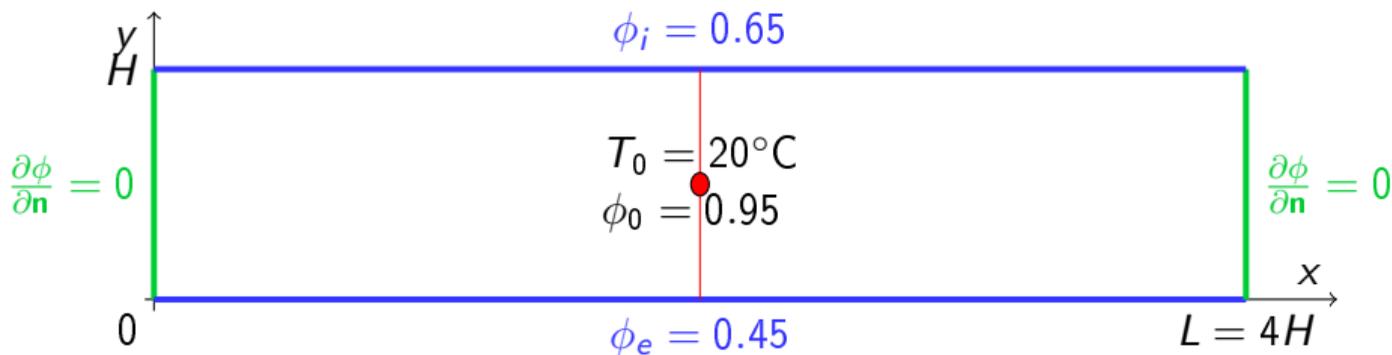
Benchmark : drying-out of a layer⁶⁷

⁶Škerget, L. and Tadeu, Antonio. “BEM numerical simulation of coupled heat and moisture flow through a porous solid”. In: *Engineering Analysis with Boundary Elements* 40 (Mar. 2014), pp. 154–161. DOI: [10.1016/j.enganabound.2013.12.006](https://doi.org/10.1016/j.enganabound.2013.12.006).

⁷Hagentoft, C-E. *HAMSTAD -Final report: Methodology of HAM-modeling, Report R-02:8*. Tech. rep. Gothenburg, Department of Building Physics, Chalmers University of Technology, 2002.

Un autre benchmark

- ▶ Cette fois-ci on ne s'intéresse qu'au transfert d'humidité dans une paroi



- ▶ Solution analytique :

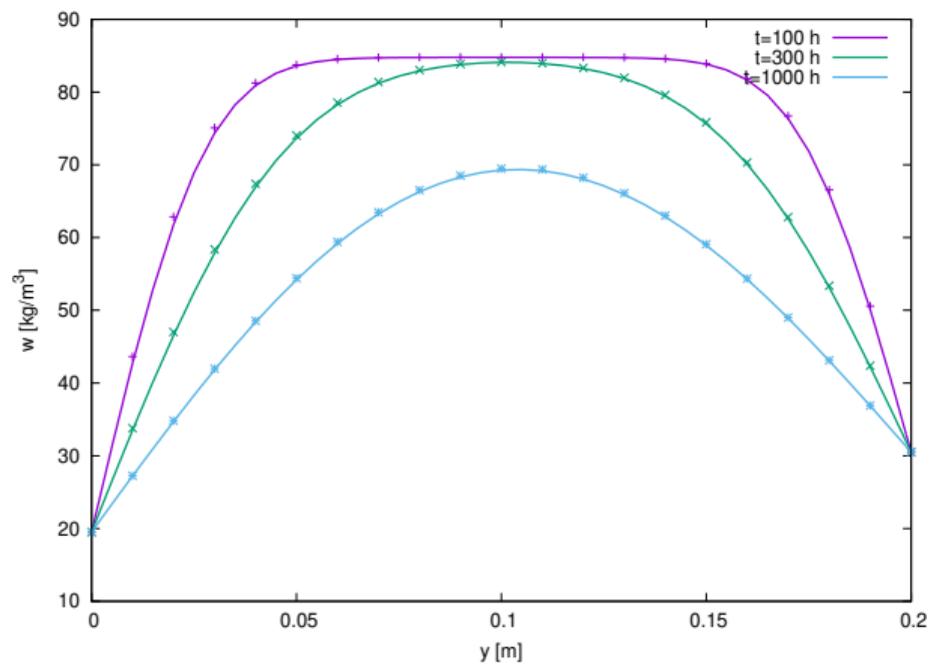
$$w(y, t) = w_i + \frac{w_e - w_i}{H} y + \sum_{n=1}^{\infty} B_n \sin\left(\frac{n\pi y}{H}\right) \exp\left(-D_w \left(\frac{n\pi}{H}\right)^2 t\right)$$

- ▶ Avec $B_n = 2 \left[\frac{w_0 - w_i}{n\pi} (1 - \cos(n\pi)) + \frac{w_e - w_i}{n\pi} \cos(n\pi) \right]$

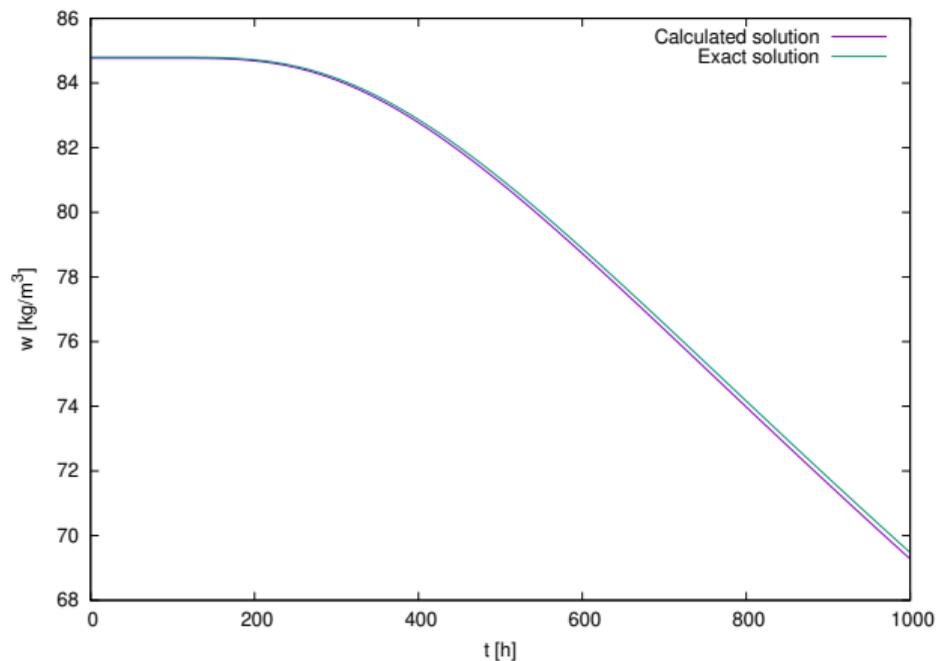
Résolution avec l'application `feelpplib_hm_heat_moisture`

- ▶ Au début : réécriture du fichier dans une autre branche sur GitHub
- ▶ Ensuite : Résolution avec l'application en utilisant les fichiers JSON

Résultats : À différents instants



Résultats : Au cours du temps



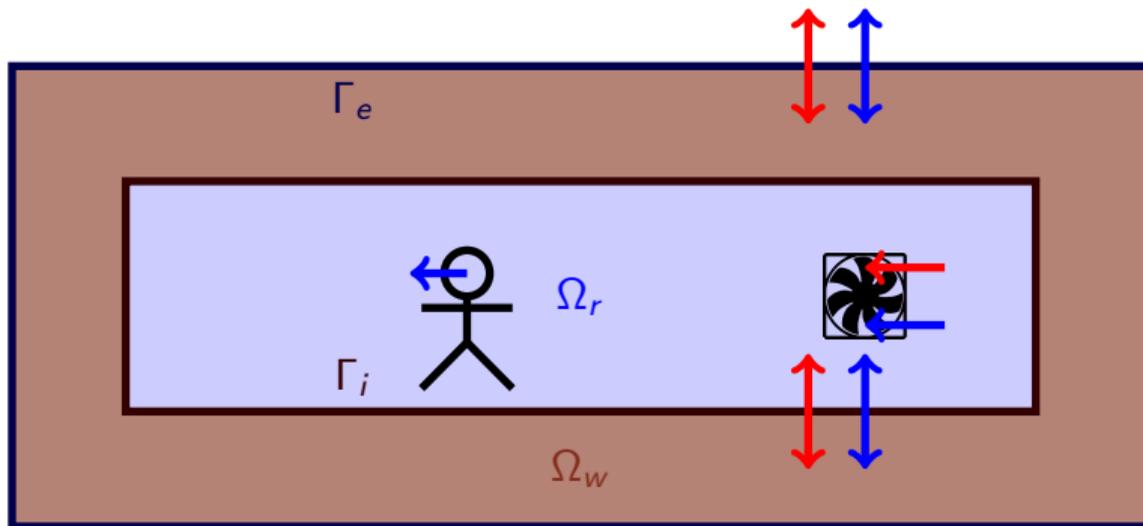
Benchmark BESTEST⁸⁹

⁸Mendes, N. and Chhay, M. and Berger, J. and Dutykh, D. “Numerical methods for diffusion phenomena in building physics”. In: *Springer Nature Switzerland AG (2019)*.

⁹J. Neymark and R Judkoff. “International energy agency building simulation test and diagnostic method for heating, ventilation, and air-conditioning equipment models (HVAC BESTEST)”. In: *Technical Report NREL/TP-550-30152. 1 : Cases E100-E200 (2002)*.

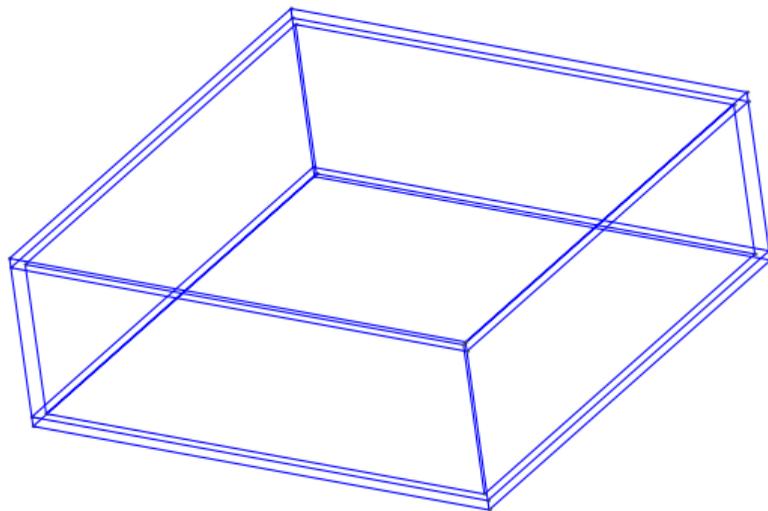
Benchmark BESTEST

- ▶ Modèle plus complet : ajouter la simulation de la température et de l'humidité dans une pièce et dans le mur



- ▶ Faire les simulations en 3D
- ▶ À terme : aller vers le confort des usagers, et mesurer la consommation énergétique

Géométrie



Modèles

$$(\rho_m c_m)_{\text{eff}} \frac{\partial T_w}{\partial t} + \nabla \cdot (-k \nabla T_w - L_v \delta_p \nabla (\phi_w p_{\text{sat}})) = 0 \quad (H_w)$$

$$\underbrace{\rho_a c_a \frac{\partial T_r}{\partial t}}_{\text{Variation d'énergie}} - \underbrace{h_i \cdot A_w \cdot (T_w - T_r)}_{\text{échanges thermiques}} - \underbrace{\eta \cdot \rho_a c_a \cdot V_a \cdot (T_e - T_r)}_{\text{ventilation}} = H \quad (H_r)$$

Conditions aux limites :

- ▶ Sur Γ_i entre le mur et la pièce :

$$-k \nabla T_w = h_i (T_r - T_{w_{s,i}}) + h_m \cdot L_v (w_r - w_w)$$

- ▶ Sur Γ_e entre le mur et l'extérieur

$$-k \nabla T_w = h_e (T_e - T_{w_{s,e}}) + h_m \cdot L_v (w_e - w_w)$$

Modèles

$$\xi \frac{\partial \phi_w}{\partial t} + \nabla \cdot (-\xi D_w \nabla \phi_w - \delta_p \nabla (\phi_w p_{\text{sat}})) = 0 \quad (M_w)$$

$$\underbrace{V_a \frac{\partial w_r}{\partial t}}_{\text{variation d'énergie}} = \underbrace{\eta V_a (w_e - w_r)}_{\text{ventilation}} + \underbrace{\dot{m}_{\text{gen}}}_{\text{génération}} + \underbrace{h_m A_w (w_w - w_r)}_{\text{échanges entre le mur et la pièce}} \quad (M_r)$$

Conditions aux limites :

► Sur Γ_i :

$$-\xi D_w \nabla \phi_w - \delta_p \nabla (\phi_w p_{\text{sat}}) = h_{m,i} (w_r - w_w)$$

► Sur Γ_e :

$$-\xi D_w \nabla \phi_w - \delta_p \nabla (\phi_w p_{\text{sat}}) = h_{m,e} (w_e - w_w)$$

Résolution

- ▶ On considère que la température est constante en espace dans la pièce
- ▶ Utilisation d'un espace produit dans Feel++ : X_h correspondant au mur et W_h pour la pièce
- ▶ Pour la pièce $W_h = Pch<0> \simeq \mathbb{R}$
- ▶ On fait comme d'habitude : formulation variationnelle, puis résolution

Résolution

- ▶ Problème à résoudre $A(T^{n+1}, V) = f(V)$
- ▶ $T^{n+1} = (T_w^{n+1}, T_r^{n+1})$ fonction trial et $V = (v, u)$ fonction test

Résolution

- Problème à résoudre $A(T^{n+1}, V) = f(V)$
- $T^{n+1} = (T_w^{n+1}, T_r^{n+1})$ fonction trial et $V = (v, u)$ fonction test

$$A(T^{n+1}, V) = \left[\begin{array}{l} \left(\int_{\Omega_w} (\rho_m c_m)^n \frac{T_w^{n+1}}{\Delta t} v + \int_{\Omega_w} (k^n \nabla T_w^{n+1}) \nabla v + \int_{\Omega_w} (L_v^n \delta_p^n \phi_w^{n+1} \nabla (p_{sat}^n)) \nabla v + \int_{\Omega_w} (L_v^n \delta_p^n p_{sat}^n \nabla (\phi_w^{n+1})) \nabla v \right. \\ \left. - \int_{\Gamma_i} (-h_i T_w^{n+1} - h_m L_v (w_w^{n+1})) v + \int_{\Gamma_e} (h_e T_w^{n+1} + h_m L_v w_w^{n+1}) v - \int_{\Gamma_i} ((L_v^n \delta_p^n \phi_w^{n+1} \nabla (p_{sat}^n))) v \right. \\ \left. - \int_{\Gamma_e} ((L_v^n \delta_p^n \phi_w^{n+1} \nabla (p_{sat}^n))) v - \int_{\Gamma_i} ((L_v^n \delta_p^n p_{sat}^n \nabla (\phi_w^{n+1}))) v - \int_{\Gamma_e} ((L_v^n \delta_p^n \phi_w^{n+1} \nabla (p_{sat}^n))) v \right. \\ \left. - \int_{\Omega_r} h_i A_w T_w^{n+1} u \right. \end{array} \right] - \int_{\Gamma_i} (h_i T_r^{n+1} + h_m L_v w_r^{n+1}) v$$

$$\left[\begin{array}{l} \int_{\Omega_r} \left(\left(\frac{\rho_a c_a}{\Delta t} + h_i A_w + \eta \rho_a c_a V_a \right) T_r^{n+1} \right) u \end{array} \right]$$

$$f(V) = \left[\begin{array}{l} \int_{\Omega_w} (\rho_m c_m)^n \frac{T_w^n}{\Delta t} v + \int_{\Gamma_e} (h_e T_e^{n+1} + h_m w_e^{n+1}) v \\ \int_{\Omega_r} \left(H^{n+1} + \frac{\rho_a c_a}{\Delta t} T_r^n + \eta \rho_a c_a V_a T_e^{n+1} \right) u \end{array} \right]$$

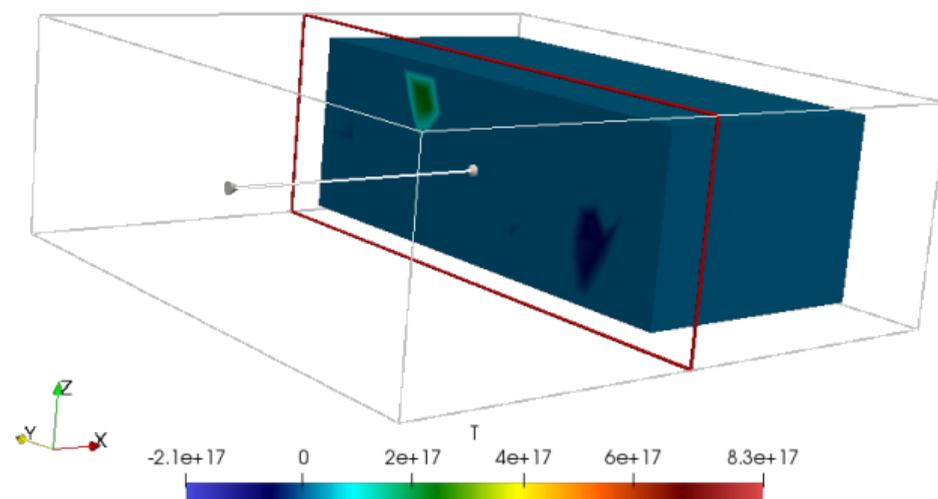
Avec Feel++

- ▶ On utilise les espaces produits de Feel++ : `ps_ = product (Xh_, Wh_) ;`
- ▶ `A` correspond à une `blockform2`, et `f` à une `blockform`
- ▶ Après la résolution, on récupère séparément les champs :

```
1 void solve()  
2 {  
3     auto T = this->ps_.element();  
4     this->bbf_.solve( _solution = T, _rhs = this->blf_ );  
5     T_ = T( 0_c );  
6     Ta_ = T( 1_c );  
7 }
```

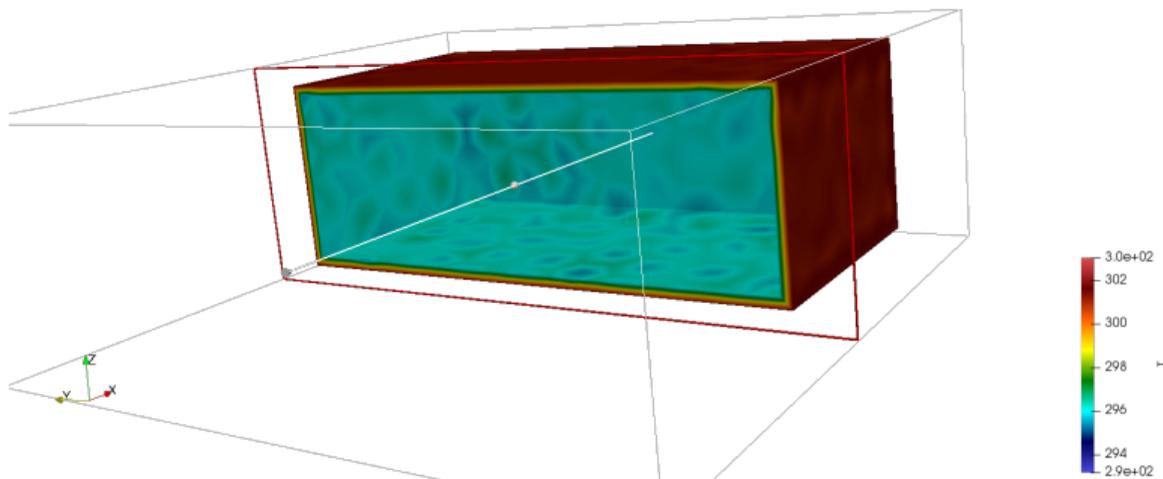
Résultats

- ▶ À la fin du stage, pas grand chose...
- ▶ Il restait quelques erreurs dans le code et la géométrie, mais on avait des résultats



Résultats

- ▶ À la fin du stage, pas grand chose...
- ▶ Il restait quelques erreurs dans le code et la géométrie, mais on avait des résultats
- ▶ Le 24/08 : Des résultats qui n'ont pas l'air aberrant (une vérification avec Dymola va être faite)



Hybridized Discontinuous Galerkin¹⁰¹¹

¹⁰Silvia Bertoluzza et al. "A HDG method for elliptic problems with integral boundary condition: Theory and Applications". In: *progress* (2020).

¹¹Feel++ Mathematics. *HDG methods for diffusion-reaction*.
<http://docs.feelpp.org/math/hdg/diffusion-reaction/>. 2020.

Système d'équations

$$\rho_m c_m \frac{\partial T_w}{\partial t} + \nabla \cdot (-k \nabla T_w) = 0 \quad (H_w)$$

$$\rho_a c_a \frac{\partial T_r}{\partial t} - h_i \cdot A_w \cdot (T_w - T_r) - \eta \cdot \rho_a c_a \cdot V_a \cdot (T_e - T_r) = H \quad (H_r)$$

- ▶ T_w température dans le mur
- ▶ T_r température dans la pièce (supposée constante en espace)

HDG

On introduit le flux thermique

$$\mathbf{q}_w = -k\nabla T_w$$

En remplaçant dans (H_w) , on obtient le problème mixte avec des équations du premier ordre :

$$\rho_m c_m \frac{\partial T_w}{\partial t} + \nabla \cdot (-k\nabla T_w) = 0 \quad (H_w)$$

HDG

On introduit le flux thermique

$$\mathbf{q}_w = -k \nabla T_w$$

En remplaçant dans (H_w) , on obtient le problème mixte avec des équations du premier ordre :

$$\rho_m c_m \frac{\partial T_w}{\partial t} + \nabla \cdot \mathbf{q}_w = 0 \quad (H_w)$$

HDG

On introduit le flux thermique

$$\mathbf{q}_w = -k\nabla T_w$$

En remplaçant dans (H_w) , on obtient le problème mixte avec des équations du premier ordre :

$$\rho_m c_m \frac{\partial T_w}{\partial t} + \nabla \cdot \mathbf{q}_w = 0 \quad (H_w)$$

- ▶ La condition intégrale $\int_{\Gamma} \mathbf{q}_w \cdot \mathbf{n} = I_{\text{target}} = \int_{\Gamma} h_i (T_w - T_r)$ permet de faire le couplage entre les deux équations (H_w) et (H_r)
- ▶ Cette méthode donne une approximation optimale pour le flux \mathbf{q}_w et la température T_w

HDG

- ▶ Les fonctions étant discontinues, la méthode requiert beaucoup plus de degrés de liberté que pour les méthodes usuelles
- ▶ Les conditions de continuité sur les solutions sont imposées faiblement, tout comme les conditions de Dirichlet (on est donc dans des espaces plus vastes)

HDG

- ▶ Les fonctions étant discontinues, la méthode requiert beaucoup plus de degrés de liberté que pour les méthodes usuelles
- ▶ Les conditions de continuité sur les solutions sont imposées faiblement, tout comme les conditions de Dirichlet (on est donc dans des espaces plus vastes)
- ▶ En calculant la formulation variationnelle, il apparaît que sur chaque élément, \mathbf{q}_w et T_w peuvent être rapidement calculés à partir de la trace \hat{T}_w
- ▶ Sur chaque élément on a un problème de petite taille pour calculer \hat{T}_w , qui peut être résolu avec un toolbox de Feel++.

Confort thermique

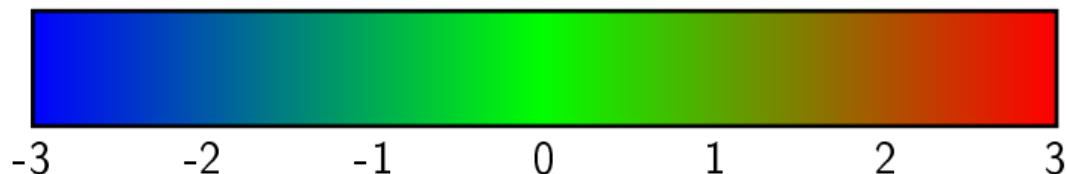
Predicted Mean Vote (PMV)

- ▶ Prédit le vote moyen d'un groupe de personnes dans une échelle thermique¹²

¹²NSI/ASHRAE. *Standard 55. Thermal Environment Conditions for Human Occupancy*. 2004.

Predicted Mean Vote (PMV)

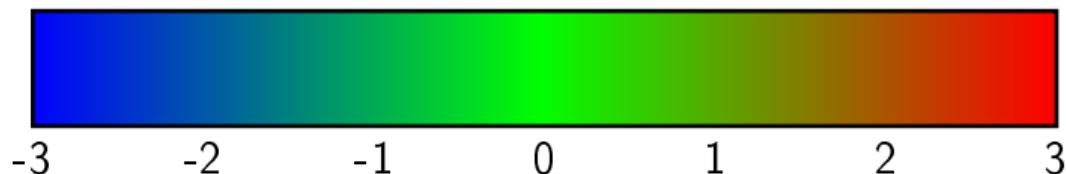
- ▶ Prédit le vote moyen d'un groupe de personnes dans une échelle thermique¹²
- ▶ -3 : trop froid
- ▶ +3 : trop chaud
- ▶ 0 : Équilibre thermique entre le corps et l'environnement



¹²NSI/ASHRAE. *Standard 55. Thermal Environment Conditions for Human Occupancy*. 2004.

Predicted Mean Vote (PMV)

- ▶ Prédit le vote moyen d'un groupe de personnes dans une échelle thermique¹²
- ▶ -3 : trop froid
- ▶ +3 : trop chaud
- ▶ 0 : Équilibre thermique entre le corps et l'environnement



- ▶ Peut être influencé par l'activité physique, l'habillement, la ventilation...

¹²NSI/ASHRAE. *Standard 55. Thermal Environment Conditions for Human Occupancy*. 2004.

PMV

Une formule pour l'estimer¹³ :

$$\begin{aligned} PMV = & [0.303 \cdot \exp(-0.036 M) + 0.028] \cdot \\ & \{ (M - W) - 3.05 \cdot 10^{-3} \cdot [5733 - 6.99 (M - W) - p_a] \\ & - 0.42 [(M - W) - 58.15] - 1.7 \cdot 10^{-5} M (5867 - p_a) - 0.0014 \cdot M (34 - T_a) \\ & - 3.96 \cdot 10^{-8} \cdot f_{cl} [(T_{cl} + 273)^4 - (T_{mr} + 273)^4] - f_{cl} \cdot h_c \cdot (T_{cl} - T_a) \} \end{aligned}$$

¹³ISO 7730. *Ergonomie des ambiances thermiques — Détermination analytique et interprétation du confort thermique par le calcul des indices PMV et PPD et par des critères de confort thermique local.* 2005.

PMV

Une formule pour l'estimer¹³ :

$$\begin{aligned} PMV = & [0.303 \cdot \exp(-0.036 M) + 0.028] \cdot \\ & \{ (M - W) - 3.05 \cdot 10^{-3} \cdot [5733 - 6.99 (M - W) - p_a] \\ & - 0.42 [(M - W) - 58.15] - 1.7 \cdot 10^{-5} M (5867 - p_a) - 0.0014 \cdot M (34 - T_a) \\ & - 3.96 \cdot 10^{-8} \cdot f_{cl} [(T_{cl} + 273)^4 - (T_{mr} + 273)^4] - f_{cl} \cdot h_c \cdot (T_{cl} - T_a) \} \end{aligned}$$

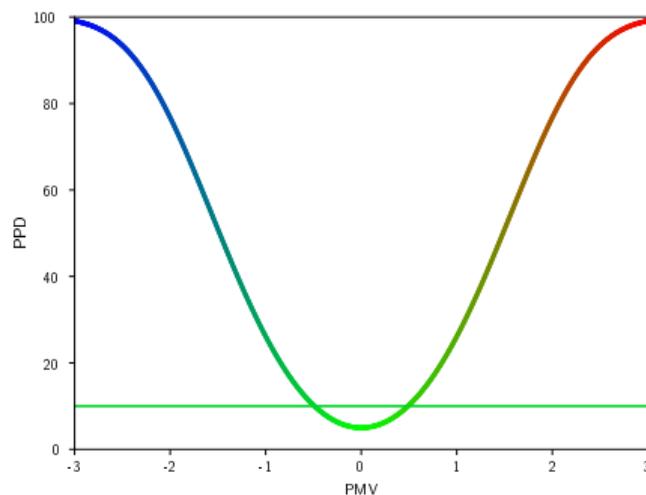
Certaines quantités dépendent de T et ϕ (par exemple T_a , T_{mr} , T_{cl} , p_a ...)

¹³ISO 7730. *Ergonomie des ambiances thermiques — Détermination analytique et interprétation du confort thermique par le calcul des indices PMV et PPD et par des critères de confort thermique local.* 2005.

Predicted Percentage of Dissatisfied (PPD)

Donne le pourcentage de personnes non satisfaites par la chaleur. Cette quantité dépend directement du *PMV* :

$$PPD = 100 - 95 \cdot \exp(-0.03353 \cdot PMV^4 - 0.2179 \cdot PMV^2)$$



Conclusion

- ▶ On a continué à étudier des modèles de transferts de chaleur et d'humidité dans les bâtiments complets
- ▶ On les a modélisés et simulés, ce qui nous a permis de comparer les résultats pour vérifier et valider notre modèle
- ▶ On peut maintenant utiliser l'application avec des configurations via les fichiers JSON (même si celle en 3D ne donne pas encore de bon résultats)

Conclusion

- ▶ On a continué à étudier des modèles de transferts de chaleur et d'humidité dans les bâtiments complets
- ▶ On les a modélisés et simulés, ce qui nous a permis de comparer les résultats pour vérifier et valider notre modèle
- ▶ On peut maintenant utiliser l'application avec des configurations via les fichiers JSON (même si celle en 3D ne donne pas encore de bon résultats)
- ▶ Pour la suite :
 - ▶ trouver ce qui ne va pas dans l'application
 - ▶ Ajouter les indices de confort à l'application
 - ▶ Implémenter la résolution avec HDG

Le rapport du stage, ainsi que les références sont disponibles sur le site de Cémosis dédié au projet *ibat* : <https://docs.cemosis.fr/ibat/latest>