Modélisation dans Thermoptim d'une unité d'osmose inverse

Nous avons développé une classe externe dans laquelle la **solution saline** est modélisée par un **retard à l'ébullition** proportionnel à la concentration, les autres propriétés thermodynamiques étant celles de l'eau (classe EauSalee).

Lors d'une opération de dessalement, un des procédés consiste à effectuer une osmose inverse.
Une unité d'osmose inverse inverse se comporte comme un diviseur recevant en entrée l'eau salée sous pression, et d'où sortent deux fluides, le perméat correspondant à l'eau purifiée et la solution concentrée.

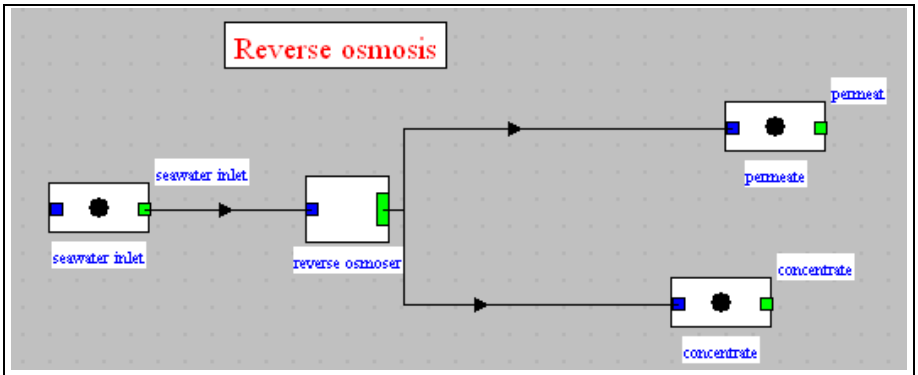

Figure 1 : Schéma du composant

La structure du modèle de l'unité est donnée figure 1.

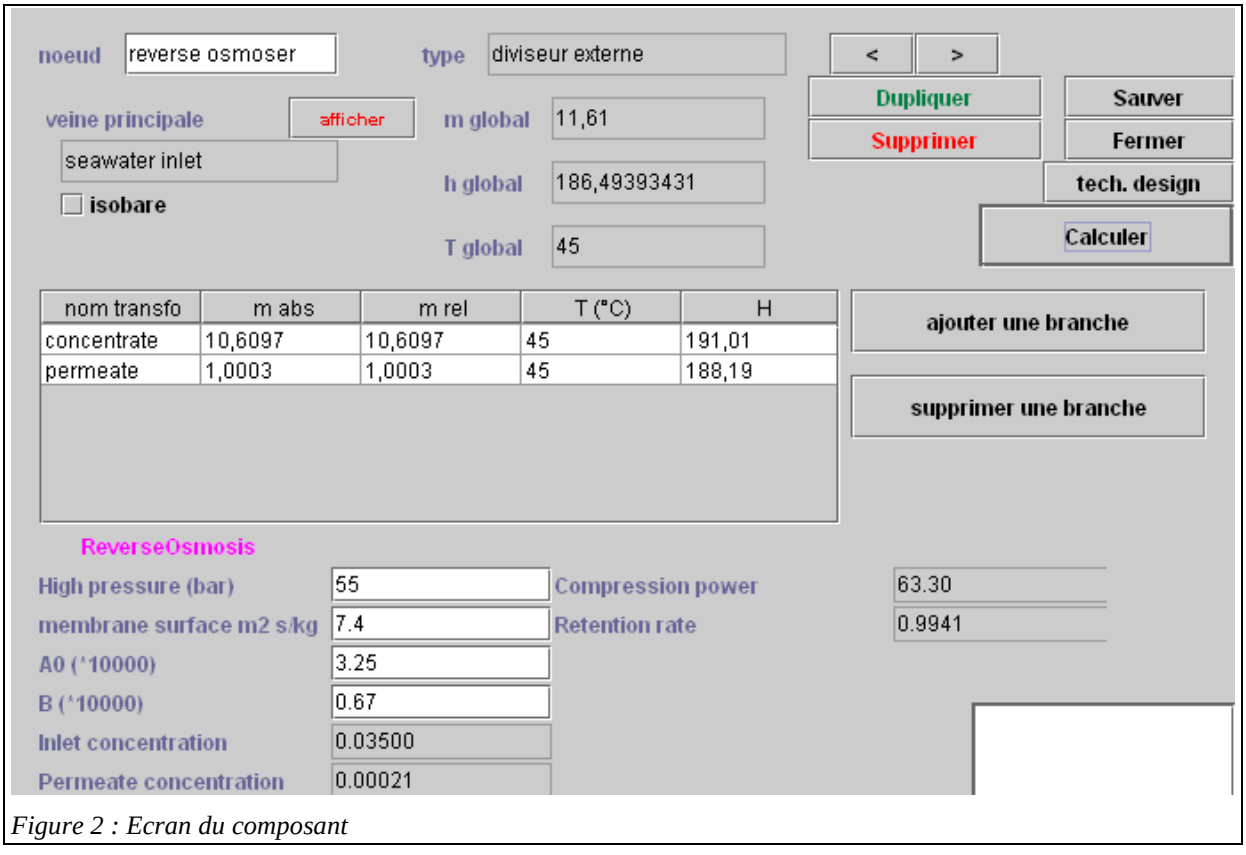

Figure 2 : Ecran du composant

Modèle physique

L'énergie consommée correspond uniquement au travail de compression de la solution initiale, et est donc beaucoup plus faible que celle mise en jeu (sous forme thermique) dans la plupart des autres dispositifs de dessalement.

La loi de van't Hoff (1) stipule que la pression osmotique exercée par le soluté est égale à la pression qu'il aurait exercé dans l'état gazeux parfait dans le même volume et à la même température. Si le soluté est dissocié sous forme d'ions, la pression osmotique est multipliée par le nombre n_i d'ions présents.

$$\pi = n_i \, X \, R \, T \qquad (1)$$

Pour l'eau de mer, l'ordre de grandeur de π est 25 à 30 bars.

ΔP étant la différence de pression à travers la membrane, on peut montrer que le débit de solvant J_e à travers la membrane est donné par (2).

$$J_e = A \, (\Delta P - \Delta \pi) \qquad (2)$$

A est appelée perméabilité à l'eau de la membrane. C'est un paramètre caractéristique de la membrane, qui dépend de la température selon une loi de type Arrhénius (3).

$$A = A_0 \exp\left[\frac{E}{R} \left(\frac{1}{298} - \frac{1}{T} \right) \right] \qquad (3)$$

Même s'il existe un transfert préférentiel du solvant, on ne peut empêcher qu'une faible fraction du soluté traverse aussi la membrane. Le débit J_s de soluté est donné par (4). Il est proportionnel à la différence de concentration.

$$J_s = B \, \Delta X \qquad (4)$$

B est la perméabilité au sel de la membrane. Elle dépend de la membrane mais pas de la température.

Le taux de conversion est le rapport du débit qui traverse la membrane au débit d'alimentation., et le taux de rétention est le rapport de la différence de concentration entre la solution initiale et le perméat à la concentration de la solution initiale.

Notons que le système d'équations ci-dessus correspond à un jeu d'équations du second degré en Xperm et Xconc que nous avons choisi de le résoudre par itération.

Les figures 2 et 3 montrent l'écran du composant et le synoptique Thermoptim d'une unité permettant de modéliser une membrane FilmTech SW30-4040 de Dow Chemical utilisée pour dessaler de l'eau de mer.

Les valeurs de A et B ont été estimées à partir de la documentation du constructeur.

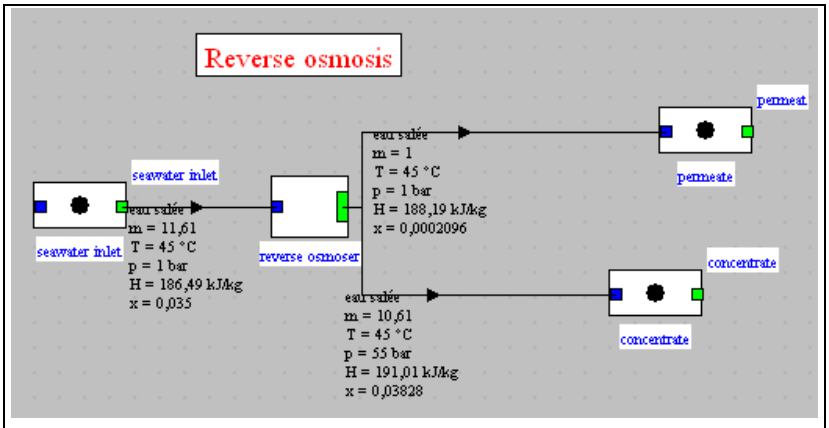

Figure 3 : Synoptique du composant

La puissance de compression pour produire 1 kg/s d'eau douce est égal à 63,3 kW, soit près du double de celui requis par la compression mécanique de vapeur. Toutefois, il est possible de récupérer une partie de cette puissance en détendant le concentrat dans une turbine couplée au compresseur.

Etude de la classe externe ReverseOsmosis

Des tests de cohérence sur la construction du diviseur externe sont effectués par la méthode checkConsistency() pour vérifier que les fluides connectés sont les bons : dans ce cas, un produit en entrée et le produit et de l'eau et en sortie. Les tests ici implémentés sont très sommaires et pourraient être améliorés. On se reportera au tome 3 du manuel de référence pour les explications sur ce point, valables pour tous les nœuds externes.

L'étude de la classe externe ReverseOsmosis permet de voir comment le modèle a été implémenté. Trois étapes suffisent pour effectuer les calculs :

1) on commence par effectuer les initialisations à partir des valeurs entrées à l'écran du composant et de celles de la transfo d'entrée, et on passe le relais à une méthode de recherche de la solution par dichotomie sur la concentration en sels du perméat Xperm

```
Pconc=Util.lit_d(HP_value.getText());
double tauCompr=Vinlet*(Pconc-Pinlet)*100*minlet;
tauCompr_value.setText(Util.aff_d(tauCompr,2));
A0=Util.lit_d(A0_value.getText())*1e-4;
B_=Util.lit_d(B_value.getText())*1e-4;
At=getmembraneWaterPermeability(Tinlet);

deltaPtm=(Pconc-Pperm);// (bar)
Amembr=Util.lit_d(Amembr_value.getText())*minlet;

Xperm=Util.dicho_T(this, 0, 0, "calcXperm", 0, Xinlet/10, 0.0001);
```

2) la méthode resXperm calcule le résidu de la valeur de Xperm, calculée de manière itérative entre 0 et Xinlet/10

```
public double f_dicho(double X, double T,String fonc){

    if (fonc.equals("calcXperm"))return resXperm(X);

    return 0;
}

double resXperm(double X){

        deltaPitm=((Xconc+Xinlet)/2-X)*2/M_NaCl*R_/1.01325*Tinlet;//MPa
        deltaPitm=deltaPitm*10;//bars

        mperm=Amembr*At*(deltaPtm-deltaPitm);//débit de perméat
        mconc=minlet-mperm;//débit de la solution concentrée

        Js=B_*Amembr*((Xconc+Xinlet)/2-X);//débit de sel dans le perméat

        Xperm=Js/mperm;
        Xconc=(minlet*Xinlet-Js)/mconc;//concentration en sortie

        double z=1000*(Xperm-X);
        return z;
}
```

3) le taux de rétention est alors déterminé, et le nœud est ensuite mis à jour en utilisant les méthodes génériques décrites dans le manuel de référence

```
    double R=(Xinlet-Xperm)/Xinlet;//retention rate
    R_value.setText(Util.aff_d(R,4));

    vTransfo= new Vector[nBranches+1];
    vPoints= new Vector[nBranches+1];
    setupPoorSolution(minlet,Tinlet,Pinlet,Xinlet);
    setupConcSolution(mconc,Tinlet,Pconc,Xconc);
    setupBuees(mperm,Tinlet,Pperm,Xperm);
    updateDivider(vTransfo,vPoints,Tinlet,Hinlet);
    de.updateProcess(setEnergyTypes(permProcess,tauCompr,0,0));


    Xinlet_value.setText(Util.aff_d(Xinlet,5));
    Xperm_value.setText(Util.aff_d(Xperm,5));
    double z=At/A0;
    System.out.println("deltaPtm: " + deltaPtm + "\tdeltaPitm "+ deltaPitm);
    System.out.println("At/A0 "+ z+ "\tJs " + Js+ "\tR " + R);
}
```