Modélisation d'un éjecteur dans Thermoptim

Un éjecteur (figure 1) reçoit en entrée deux fluides généralement gazeux, mais qui peuvent aussi être liquides ou diphasiques :

- le fluide à haute pression, appelé fluide moteur ou primaire ;
- le fluide à basse pression, appelé fluide secondaire ou entraîné.

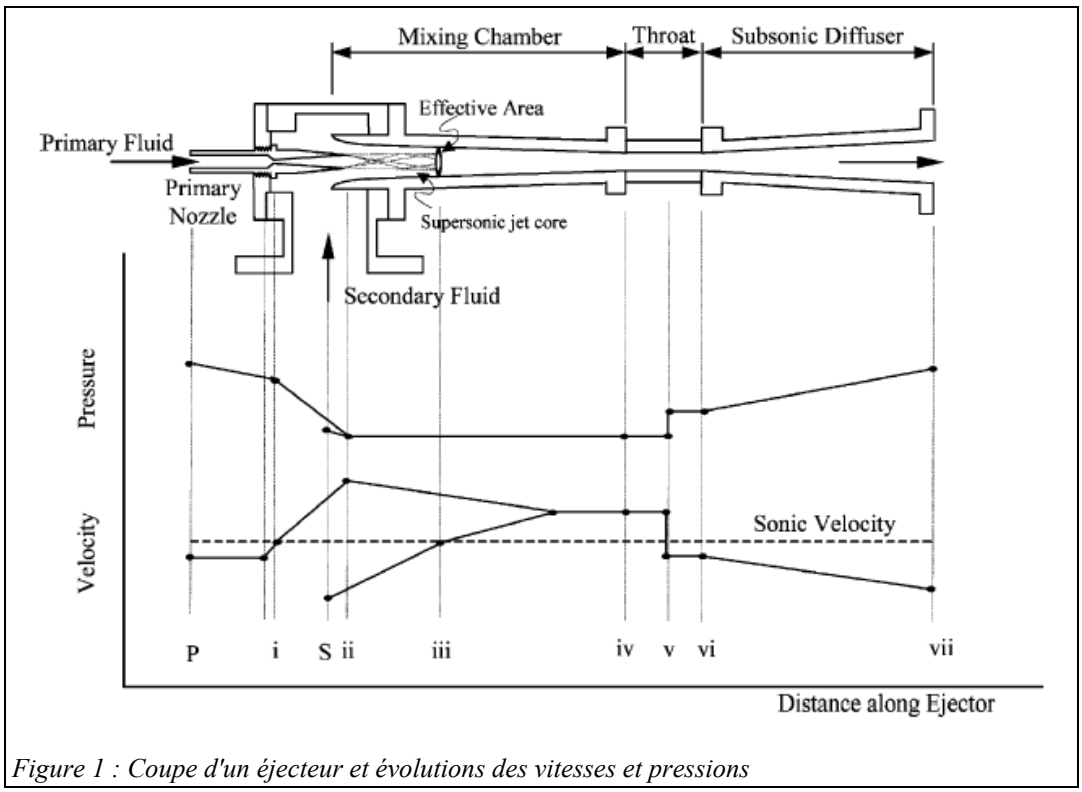

Figure 1 : Coupe d'un éjecteur et évolutions des vitesses et pressions

Le fluide moteur est accéléré dans un convergent-divergent, créant une baisse de pression dans la zone de mélange, ce qui a pour effet d'aspirer le fluide secondaire. Les deux fluides sont alors mélangés et une onde de choc peut prendre place dans la zone suivante (throat sur la figure 1). Il en résulte une augmentation de la pression du mélange et une baisse de sa vitesse, qui devient subsonique. Le diffuseur permet de convertir la vitesse résiduelle en augmentation de pression.

L'éjecteur réalise ainsi une compression du fluide secondaire au prix d'une baisse d'enthalpie du fluide primaire.

Les trois paramètres les plus importants pour caractériser le fonctionnement global d'un éjecteur sont :

- le **rapport d'entraînement w**, rapport du débit-masse secondaire au débit-masse primaire
- le **rapport de compression**, égal au rapport de la pression statique en sortie de diffuseur à la pression statique du fluide secondaire
- un **rapport de sections** (minimale sur maximale, ou du flux moteur au flux entraîné…) de l'éjecteur, qui détermine sa géométrie.

Signalons à ce propos que l'un des problèmes pratiques rencontrés lors de l'utilisation d'un éjecteur dans un cycle est que ses performances dépendent beaucoup de ses conditions de fonctionnement : le rapport de compression obtenu est bien évidemment fonction du rapport d'entraînement, mais une variation de ce dernier induit une modification de la géométrie optimale de l'éjecteur, qu'il est bien évidemment impossible de réaliser.

Il s'ensuit qu'un éjecteur s'adapte assez mal à un fonctionnement hors des conditions de design.

Modèle de l'éjecteur

La modélisation des éjecteurs repose le plus souvent sur l'hypothèse que les fluides primaire et secondaire peuvent être assimilés à des gaz idéaux dans la chambre de mélange, ce qui est à peu près justifié compte tenu de la basse pression qui y règne. Toutefois, il existe aussi des cas où l'un de ces fluides est liquide ou diphasique, de telle sorte que le mélange peut être diphasique, et plusieurs hypothèses peuvent être retenues : soit mener les calculs avec les propriétés du fluide réel tout en utilisant l'hypothèse d'un écoulement monophasique, soit négliger la phase liquide dans le calcul des vitesses, soit considèrer un fluide équivalent.

Le calcul de l'éjecteur (modèle dit unidimensionnel) repose sur les hypothèses suivantes :

- la détente des fluides primaire et secondaire dans la tuyère d'entrée est supposée adiabatique, avec prise en compte des irréversibilités grâce à un rendement isentropique
- la pression reste constante dans la chambre de mélange (il existe des éjecteurs à section de mélange constante, mais ils sont moins performants que les autres et nous ne les considérerons pas ici)
- lorsque le flux mélangé est supersonique, un choc normal peut prendre place dans la chambre de mélange, ce qui ralentit le fluide et crée une surpression importante
- la compression dans le diffuseur est supposée adiabatique, avec prise en compte des irréversibilités grâce à un rendement isentropique
- les propriétés du fluide sont homogènes dans toute section.

Le modèle que nous avons retenu est celui proposé par Li et Groll, qui présente l'avantage d'être formulé d'une manière indépendante des propriétés du fluide, et nous l'avons légèrement reformulé, et complété pour tenir compte d'un choc éventuel, ces deux auteurs se limitant implicitement au cas où le flux mélangé est subsonique.

Dans ce modèle, on fait pour simplifier les calculs une hypothèse a priori sur la **perte de pression** entre l'aspiration du flux secondaire Pe et l'entrée dans la zone de mélange Pb. Nous considèrerons, ce qui est équivalent, un facteur multiplicatif de Pe. Il est clair qu'en pratique, on ne connaît pas la valeur de ce facteur, qui n'est pas directement mesurable. Si l'on veut s'imposer une valeur de la pression en sortie d'éjecteur Pd ou du rapport de compression Pd/Pe, il faut alors itérer sur la valeur de ce facteur, ce qui est facile à faire une fois le modèle paramétré et validé.

Le modèle décompose l'éjecteur en 4 principales zones :

- la zone de détente du flux moteur
- la zone de détente du flux entraîné
- la zone de mélange, avec choc éventuel
- le diffuseur.

Mise en équations

m_{mi} et m_{si} étant les débits-masses des flux moteur et entraînés, le rapport d'entraînement vaut :

$$w = \frac{m_{si}}{m_{mi} + m_{si}}$$

Zone de détente du flux moteur

Nous utiliserons l'indice mb pour caractériser l'état de sortie du fluide, qui se détend de manière adiabatique :

$s_{mb,is} = s_{mi}$

$h_{mb,is} = f(s_{mi}, P_b)$

$h_{mb} = h_{mi} - \eta_s (h_{mi} - h_{mb,is})$

La vitesse du fluide à l'entrée étant négligeable, celle en sortie vaut :

$C_{mb} = \sqrt{2\,(h_{mi} - h_{mb})}$

Le volume spécifique v_{mb} permet alors de connaître la section de passage par unité de débit-masse total :

$$a_{mb} = \frac{v_{mb}}{C_{mb}\,(1 + w)}$$

Zone de détente du flux entraîné

Nous utiliserons l'indice sb pour caractériser l'état de sortie du fluide, qui se détend de manière adiabatique :

$s_{sb,is} = s_{si}$

$h_{sb,is} = f(s_{si}, P_b)$

$h_{sb} = h_{si} - \eta_s\,(h_{si} - h_{sb,is})$

La vitesse du fluide à l'entrée étant négligeable, celle en sortie vaut :

$C_{sb} = \sqrt{2\,(h_{si} - h_{sb})}$

Le volume spécifique v_{sb} permet alors de connaître la section de passage par unité de débit-masse total :

$$a_{sb} = \frac{v_{sb}}{C_{sb}}\,\frac{w}{(1 + w)}$$

Zone de mélange

Nous utiliserons l'indice mix pour caractériser l'état de sortie du fluide, qui ne peut être déterminé que de manière itérative, en résolvant simultanément en pression les équations de conservation du débit-masse, de la quantité de mouvement et de l'enthalpie.

La résolution de ces équations montre qu'il peut exister une ou deux pressions de mélange. L'explication physique est la suivante : si le flux mélangé est supersonique, une recompression avec onde de choc dit normal prend place du fait que la pression aval est supérieure à celle de l'écoulement. Les deux solutions mathématiques correspondent aux deux pressions avant et après le choc.

$$P_b\,(a_{mb} + a_{sb}) + \frac{C_{mb}}{(1 + w)} + \frac{w\,C_{sb}}{(1 + w)} = P_{mix}\,(a_{mb} + a_{sb}) + C_{mix}$$

L'équation enthalpique s'écrit :

$$h_{mi} + w\,h_{si} = (1 + w)\,\left(h_{mix} + \frac{C_{mix}^{\,2}}{2}\right)$$

La conservation du débit-masse s'écrit :

$v_{mix} = (a_{mb} + a_{sb})\,C_{mix}$

Nous retiendrons comme solution, dans le cas où il y en a deux, la pression la plus élevée, ce qui équivaut à prendre la plus basse et à résoudre les équations de l'onde de choc ci-dessous. Cette manière de faire est plus précise, les équations ci-dessous n'étant stricto sensu valables que pour les gaz parfaits.

Onde de choc

S'il y a onde de choc, les équations sont les suivantes, x représentant l'amont (supersonique), et y l'aval (subsonique).

$$M_y^2 = \frac{M_x^2 + \frac{2}{\gamma - 1}}{\frac{2\gamma}{\gamma - 1} M_x^2 - 1}$$

$$\frac{P_y}{P_x} = \frac{1 + \gamma M_x^2}{1 + \gamma M_y^2}$$

$$\frac{T_y}{T_x} = \left(\frac{P_y}{P_x}\right)^2 \left(\frac{M_y}{M_x}\right)^2$$

Références

Li, D., Groll A., Transcritical CO2 refrigeration cycle with ejector-expansion device, International Journal of Refrigeration 28 (2005) 766–773

Modélisation d'un éjecteur dans Thermoptim

Un éjecteur se comporte comme un mélangeur recevant deux fluides : le flux primaire et le flux secondaire, et dont sort le flux total recomprimé.

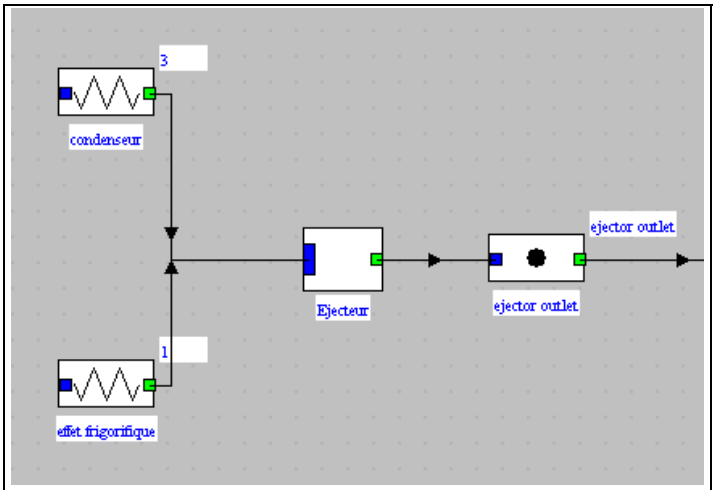

Figure 2 : Schéma du composant éjecteur

Etude de la classe externe FluidEjector

Des tests de cohérence du nœud sont effectués par la méthode checkConsistency() pour vérifier que les fluides connectés sont les bons.

On se reportera au tome 3 du manuel de référence pour les explications sur ce point, valables pour tous les nœuds externes.

L'écran de l'éjecteur est donné figure 3. Il possède quatre paramètres :

- le facteur Pe/Pb de pertes de charges à l'entrée du fluide secondaire dans l'éjecteur, qui détermine la pression minimale dans l'éjecteur
- le rendement isentropique des deux tuyères (fluide moteur et fluide entraîné)
- le rendement isentropique du diffuseur de sortie
- la facteur de frottement pour prendre éventuellement en compte une perte de charge dans la zone de mélange.

Les résultats du calcul s'affichent au-dessus des champs de saisie des paramètres. La première ligne indique :
- Pout, pression de sortie (bars)
- Tout, température de sortie (°C)
- Pmi/Pout, rapport de la pression du fluide moteur à la pression de sortie

- Pout/Psi, rapport de la pression de sortie à la pression du fluide entraîné
- xout, titre en sortie
- DP, perte de charge initiale du flux entraîné
- Pmel, pression de mélange (bars)

La seconde ligne fournit les sections (en mm2) à l'entrée de la zone de mélange pour les fluides moteur (Amb) et entraîné (Asb).

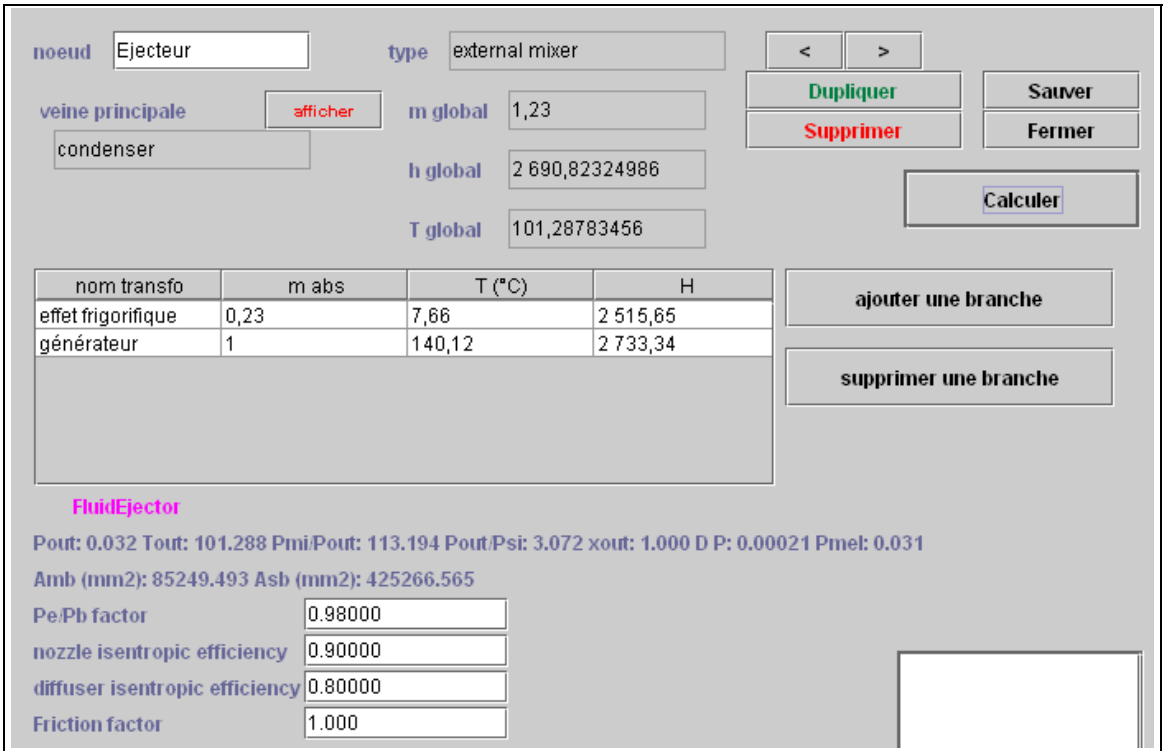

Figure 2 : Ecran du composant éjecteur

L'étude de la classe externe FluidEjector permet de voir comment le modèle a été implémenté. Comme on peut le voir, six étapes suffisent pour effectuer les calculs :

1) on commence par lire les paramétrages entrés à l'écran (on a ajouté ici un facteur de friction pour tenir compte de pertes par frottements dans la zone de mélange) :

```
DeltaP_factor=Util.lit_d(DeltaP_factor_value.getText());
eta_m=Util.lit_d(etaNozzle_value.getText());
eta_s=eta_m;
eta_d=Util.lit_d(etaDiff_value.getText());
fricFactor=Util.lit_d(fricFactor_value.getText());
```

2) on effectue ensuite les calculs relatifs au flux moteur :

```
//calculs relatifs au flux moteur

//calcul de l'enthalpie en fin de détente isentropique
double T=refrig.getT_from_sP(Smi,Pb);//température isentropique
double x=refrig.getQuality()[0] ;
refrig.CalcPropCorps(T,Pb,x) ;
getSubstProperties(nomCorps);
double Hmb_is=Hsubst;//enthalpie isentropique
double Hmb=Hmi-eta_m*(Hmi-Hmb_is);//enthalpie réelle (kJ/kg)
Cmb=Math.pow(2000*(Hmi-Hmb), 0.5);//vitesse à l'entrée de la zone de mélange (m/s)

//calcul du volume spécifique à l'entrée de la zone de mélange
T=refrig.getT_from_hP(Hmb,Pb);//température à l'entrée de la zone de mélange
x=refrig.getQuality()[0] ;
refrig.CalcPropCorps(T,Pb,x) ;
getSubstProperties(nomCorps);
vmb=Vsubst;//volume spécifique (m3/kg)
Amb=vmb/Cmb/(1+w);//section à l'entrée de la zone de mélange par unité de débit total (m2/(kg/s))
Xmb=x;
```

3) puis ceux relatifs au flux entraîné :

```
//calculs relatifs au flux entraîné

//calcul de l'enthalpie en fin de détente isentropique
T=refrig.getT_from_sP(Ssi,Pb);//température isentropique
x=refrig.getQuality()[0] ;
refrig.CalcPropCorps(T,Pb,x) ;
getSubstProperties(nomCorps);
double Hsb_is=Hsubst;//enthalpie isentropique
eta_s=1;
double Hsb=Hsi-eta_s*(Hsi-Hsb_is);//enthalpie réelle (kJ/kg)
Csb=Math.pow(2000*(Hsi-Hsb), 0.5);//vitesse à l'entrée de la zone de mélange (m/s)

//calcul du volume spécifique à l'entrée de la zone de mélange
T=refrig.getT_from_hP(Hsb,Pb);//température à l'entrée de la zone de mélange
x=refrig.getQuality()[0] ;
refrig.CalcPropCorps(T,Pb,x) ;
getSubstProperties(nomCorps);
vsb=Vsubst;//volume spécifique (m3/kg)
Asb=vsb/Csb/(1+w)*w;//section à l'entrée de la zone de mélange par unité de débit total (m2/(kg/s))
Xsb=x;
```

4) on recherche alors la valeur de la pression de mélange, en utilisant pour cela la fonction de dichotomie Util.dichoT, qui cherche la pression P telle que le résidu défini par getPressure(P) soit nul :

```
//  recherche d'une pression de mélange sur base conservation quantité mouvement et enthalpie
//indice as pour after shock
double Pmax=Pb+(Cmb/(1+w)+Csb*w/(1+w))/(100000*(Amb+Asb)*fricFactor);
System.out.println("1 Xmb\t"+Xmb+"\tXsb\t"+Xsb+"\tPmax\t"+Pmax);
System.out.println("\tdelta");
double Pmix;
double Pm1=Util.dicho_T (this,0,1, "P_m",0.8*Pb, (Pmax+0.8*Pb)/2.,0.001);//recherche de la pression en sortie de la se
System.out.println("1 Pm1\t"+Pm1+"\tCmix\t"+Cmix+"\tSmix\t"+Smix+"\tTmix\t"+Tmix);
double Pm2=Util.dicho_T (this,0,1, "P_m", (Pmax+0.8*Pb)/2.,Pmax,0.001);//recherche de la pression en sortie de la sect
System.out.println("1 Pm2\t"+Pm2+"\tCmix\t"+Cmix+"\tSmix\t"+Smix+"\tTmix\t"+Tmix);
if(Pm1>Pm2) Pmix=Pm1;//s'il y a deux solutions, on retient la pression la plus haute (après onde de choc)
else Pmix=Pm2;
getPressure(Pmix);//pour effectuer une remise à jour variables du mélange

  double getPressure(double P){//recherche de la pression en sortie de la section de mélange (P est la pression)
      Cmix=fricFactor*((Pb-P)*100000*(Amb+Asb)+Cmb/(1+w)+Csb*w/(1+w));//vitesse en sortie de la zone de mélange (m/s)
      Hmix=(Hmi+w*Hsi)/(1+w)-Cmix*Cmix/2000.;//enthalpie en sortie de la zone de mélange (kJ/kg)

      //calcul du volume spécifique en sortie de la zone de mélange
      Tmix=refrig.getT_from_hP(Hmix,P);//température en sortie de la zone de mélange
      Xmix=refrig.getQuality()[0] ;
      refrig.CalcPropCorps(Tmix,P,1) ;
      getSubstProperties(nomCorps);
      vmix=Vsubst;
      Smix=Ssubst;
      double delta=vmix-(Amb+Asb)*Cmix;
      return delta;
}
```

5) on calcule le nombre de Mach du mélange pour savoir si l'onde de choc éventuelle a été prise en compte. Si nécessaire, on détermine la recompression qu'elle induit :

```
//calcul de Cp et gamma en sortie de la zone de mélange
r=8.31459/Msubst;
refrig.CalcPropCorps(Tmix,Pmix,1);
getSubstProperties(nomCorps);
double H=Hsubst;
refrig.CalcPropCorps(Tmix+1,Pmix,1);
getSubstProperties(nomCorps);
double Cp=Hsubst-H;
gamma=Cp/(Cp-r);

Machm=Cmix/Math.pow(gamma*Pmix*100000*vmix,0.5);


//Calculs relatifs au choc : Machm doit être supérieur à 1 pour que cela ait un sens
if(Machm>=1){//s'il n'y a qu'une solution et qu'elle correspond à un écoulement supersonique
//on prend des équations valables pour les gaz parfaits, mais on boucle le bilan massique avec les propriétés exactes

    double y=(2./(gamma-1)+Machm*Machm)/(2.*gamma/(gamma-1)*Machm*Machm-1);//équation 5.22 poly meca flu
    Machas=Math.pow(y,0.5);

    Pas=Pmix*(1+gamma*Machm*Machm)/(1+gamma*y);//équation 5.16 poly meca flu
    Tas=Tmix*y/Machm/Machm*Pas*Pas/Pmix/Pmix;//équation 5.12 poly meca flu
}
else{//écoulement subsonique ou solution après onde de choc
    Pas=Pmix;
    Tas=Tmix;
    Machas=Machm;
    System.out.println("cas subsonique Pas\t"+Pas+" Tas\t"+Tas);
}

refrig.CalcPropCorps(Tas,Pas,1);
getSubstProperties(nomCorps);
double Has=Hsubst;
double Sas=Ssubst;
refrig.CalcPropCorps(Tas+1,Pas,1);
getSubstProperties(nomCorps);
Cp=Hsubst-Has;
gamma=Cp/(Cp-r);

Cas=Machas*Math.pow(gamma*r*Tas*1000,0.5);//vitesse après le choc (m/s)
```

6) on effectue les calculs relatifs au diffuseur de sortie

```
//calcul du diffuseur
double Hd=(Hmi+w*Hsi)/(1+w);//bilan global
double Hd_is=eta_d*(Hd-Has)+Has;

double[]val=refrig.getPTx_from_sh(Sas,Hd_is);//inversion en s,h pour zone vapeur
//attention : ce n'est pas toujours implémenté dans les corps externes
//on pourrait prendre la solution gaz idéaux
```

7) le nœud est mis à jour en utilisant les méthodes génériques décrites dans le manuel de référence

```
//Mise à jour du noeud externe

vTransfo= new Vector[nBranches+1];
vPoints= new Vector[nBranches+1];
setupRichSolution(msr,Td,Pd,xd);
setupPoorSolution(msi,Tsi,Psi,Xsi);
setupRefrigerant(mi,Tmi,Pmi,Xmi);
updateMixer(vTransfo,vPoints,Td,Hd_is);

JLabelRes.setText("Pout: "+Util.aff_d(Pd,3)+" Tout: "+Util.aff_d(td,3)
+" Pmi/Pout: "+Util.aff_d(Pmi/Pd,3)+" Pout/Psi: "+Util.aff_d(Pd/Psi,3)+" xout: "+Util.aff_d(xd,3)
+" D P: "+Util.aff_d(Psi-Pb,5)+" Pmel: "+Util.aff_d(Pmix,3));
JLabelRes2.setText("Amb (mm2): "+Util.aff_d(Amb*msr*1.e6,3)+" Asb (mm2): "+Util.aff_d(Asb*msr*1.e6,3));


updateStraightlyConnectedProcess(outputProcess, outputProcess,
    false,//boolean downstream,
    true,//boolean inletPoint,
    true,//boolean outletPoint,
    false,//boolean updateT,
    0,//double T,
    false,//boolean updateP,
    0,//double P,
    true,//boolean updateX,
    xd);
```