

Matlab Code

```

1  clc
2  clear all
3  close all
4
5  %% Debut courbes de sat
6
7
8  % Hypotheses :
9  Ptot=101325; %Pa
10
11
12  % Constantes
13  Cp_as = 1.005 % kJ/kg/K (the engineering toolbox)
14  Cp_eau = 4.180 %kJ/kg/K (the engineering toolbox)
15  Lv_eau = 2257.92 % kJ/kg
16  Cpair_humide=1040; %J/kg/K %
17  Cpair_humide=Cair Cpair + %eau
18  Cpeau (Cpeau 4180 J/kg/K Cpair=1005 J/kg/K)
19  R=8,3144621; %J/mol/K
20  Mas= 28.96546*10^-3 ; %kg/mol
21  Mv= 18.01528*10^-3 ;
22
23  hlg = 2501000; %J/kg
24  cw = 1860;
25
26  % Variables
27  Tmin=-15;
28  Tmax=50;
29  Tmaxaffiche=Tmax+4; %valeur de
30  pas pour le calcul
31  Tminaffiche=Tmin+4;
32  omega_maxaffich=0.06;
33
34  T=[Tmin:0.01:Tmax]; %T augmente de 1
35  degre par pas de 100
36
37  %% Trace de omega en fct de T
38
39  Pvs_min=pression_vapPa(Tmin);
40  Pvs_max=pression_vapPa(Tmax);
41
42  omega_max=0.621945*(Pvs_max./(Ptot-Pvs_max));
43
44  vs_min= ((Mv/Mas)) * (R*(Tmin+273.15)/(Ptot*Mv));
45  vs_max=((Mv/Mas)+omega_max) * (R*(Tmax+273.15)/(Ptot*Mv));
46
47  Pvs=pression_vapPa(T);
48  omega=0.621945*(Pvs./(Ptot-Pvs)); %
49  vient de psychrometric news
50
51  pressions_partiellesPa=
52  pressionpartielle_fomega(omega,Ptot);
53
54  pressions_partielleskgcm2=
55  pressions_partiellesPa.*1E-5;
56
57  %% Tracer du diagramme psychrometrique
58
59  figure(1) %figure diagramme
60  hold
61
62  %% Parametres de l'image
63
64  % taille de la feuille o?? on imprime
65  width = 16.5; % Width in inches
66  height = 23.4; % Height in inches
67  alw = 0.75; % AxesLineWidth
68  fsz = 10; % Fontsize
69  lw = 1.5; % LineWidth
70  msz = 8; % MarkerSize
71
72  % Choix du format d'affichage
73  ici
74
75  pbaspect([21 29.7 1])
76  %% Mise en forme
77  pos = get(gcf, 'Position');
78
79  % Vecteurs pour les ticks en abscisse et
80  ordonnee
81  abscisses=T(1:100:end);
82  ordonnees=[0:0.001:omega_maxaffich];
83
84  set(gca, 'FontSize', fsz, 'LineWidth',
85  alw, 'Box','off','XTick',abscisses,
86  'YTick',ordonnees); %<- Choix
87  propertes, choix notamment des
88  graduations abscisses/ordonnees
89
90  title('Abaque psychrometrique simplifie')
91
92  xlabel('T en {\circ}C')
93  ylabel('omega en kg/kg')
94  xlim([Tmin Tmaxaffiche])
95  ylim([0 omega_maxaffich])
96
97  % Gestion des axes
98  axes=gca;
99  axes.YAxisLocation='right'
100 axes.FontSize=8;
101
102 %% Trace de la courbe de saturation
103
104 Pvs=pression_vapPa(T);
105 courbe_sat=0.621945.*(Pvs./(Ptot-Pvs));
106
107 for h=1:10
108 plot(T, 0.1*h*courbe_sat, '-k','
109 LineWidth',lw,'MarkerSize',msz)
110 end
111
112 Tlegende=2*Tmax/3;
113 Pvs_legende=pression_vapPa(Tlegende);
114 courbe_sat_legende=0.621945.*(Pvs_legende
115 ./ (Ptot-Pvs_legende));
116
117 %droites et legendes selon les
118 pourcentages d'humidite
119 pas=0.2;
120 for i=0:pas:1
121 plot(T, i*courbe_sat, '-k')
122 n=i*100;
123 str=['\epsilon= ' num2str(n) '%'];
124 xt=Tlegende;
125 yt=(i*courbe_sat_legende)+(0.0007)*
126 exp(i/2);
127 text(xt,yt+0.001,str,'FontSize',10,'
128 Color','black','rotation',65*i)
129 end
130
131 title('Diagramme psychrometrique');
132
133 %% courbes isenthalpes
134
135 %pente pour tracer ensuite l'echelle
136 transversale
137 pente_ech_H=Cp_as/Lv_eau; %oppose du
138 coeff direct des isenthalpes, voir
139 formule L.147
140
141 %calcul de l'intersection entre le bord
142 de l'image et l'axe
143 k=1;
144 while courbe_sat(k)<=omega_maxaffich
145 k=k+1;
146 end
147 b=omega_maxaffich-pente_ech_H*T(k); %
148 coeff d'ordonnee a l'origine
149
150 Ech.h=pente_ech_H*T+b; %droite d'echelle
151
152 %Axe enthalpie
153
154 for h=-4:0.5:60 %nombre de droites,
155 intervalle d'enthalpies
156
157 h=4.184*h; %conversion du compteur
158 dans la bonne unite
159 iso_h=(h-Cp_as*T)./(Lv_eau+Cp_eau*T);
160 % fle de psychrometric news
161 %iso_h=1/2500*(h-1.826*T);
162
163 k=1; %compteur pour trouver l'
164 intersection entre Pv_sat et

```

```

153         iso_h
154         while iso_h(k)>courbe_sat(k)
155             k=k+1;
156         end
157         %on repasse en valeurs entieres
158         h=h/4.184;
159         plot(T(k:end), iso_h(k:end), '-b')
160
161
162         %gestion de la legende
163         %on met la legende au point d'
164         intersection entre courbes Pvs
165         et isenthalpiques
166
167         if h<=47
168             %--> 47 a revoir
169             if mod(h,2)==0 && k>=2%seulement
170                 1/2
171                 str={h};
172                 text(T(k-2), iso_h(k-2), str,
173                     'FontSize',10,'Color','
174                     blue','rotation',(h)
175                     /38*50)
176
177             end
178
179             %creation de l'echelle en
180             diagonale
181
182             %if
183             %intersection isenthalpes et ax
184             diagonal
185             d=1;
186             while iso_h(k)>=Ech_h(k)
187                 k=k+1;
188             end
189
190             %on prolonge les courbes en
191             pointilles jusqu'a l'
192             echelle
193             markerstep=10; %pour espacer
194             les points des droites
195             au dessus de la courbe de
196             saturation
197             plot(T(k:markerstep:end),
198                 iso_h(k:markerstep:end),
199                 '.b','MarkerSize',0.5)
200
201             text(T(k), Ech_h(k), '-',
202                 'FontSize',10,'Color','
203                 blue','rotation',-45)
204             if h>0 && h<49 && mod(h,1)=
205                 str={h};
206                 text(T(k)-0.5, Ech_h(k)
207                     +0.0003, str,
208                     'FontSize',10,'Color'
209                     , 'blue', 'rotation'
210                     ,55.5)
211
212             if h==26 %pour etre
213                 milieu de l'axe
214                 text(T(k)+0.5, Ech_h
215                     )+0.002, 'h en
216                     kcal/kg',
217                     'FontSize',12,
218                     'Color','blue',
219                     'rotation',atan
220                     pente_ech.H)
221
222             end
223         end
224
225         %axe diagonale des enthalpies
226         plot(T, Ech_h, '-b')
227
228         %Trace des isovolumes specifiques
229
230         %CHECK --> Valeur de Vs a calculer en
231         fonction du reste
232
233         for vs=0.70:0.01:1
234             omega_vs=(vs.*Ptot.*Mv)/(R.*(T
235                 +273.15))-(Mv/Mas); %formul
236             exacte (file:///users/phelma/
237                 phelma2015/boudetal/Documents/
238                 Projet%20abaque%20air%20humide%
239                 docs%20ext%C3%A9rieurs/Calcul%
240                 des%20param%C3%A8tres%20de%20l'
241                 air%20humide%20-%20Projet%20
242                 AntiSecos.htm)
243
244             k=1; %compteur
245
246             %on cherche l'intersection entre la
247             courbe de saturation et les
248             isovolumes
249             while omega_vs(k)>courbe_sat(k)
250                 k=k+1;
251             end
252
253             plot(T(k:end), omega_vs(k:end), '-r')
254         end
255
256         %% Grid
257
258         %trace de la grille et des
259         autres echelles
260
261         %on trace la grille verticale (selon les
262         temperatures) sous la courbe de sat
263         stem(T(1:100:end), courbe_sat(1:100:end),
264             'k','LineWidth',lw/4, 'Marker','
265             ')
266         %et la grille selon les ordonnees
267         L_ordo=length(ordonnees);
268
269         %on voit que la fonction fsolve
270         fonctionne sur les T entre 50 et 11?
271         C mais bug ensuite
272         %dans une seconde boucle, on calcule donc
273         la solution a la main pour aider le
274         programme
275         for i=2:L_ordo
276             plot([Tmax, Tmax.affiche], [
277                 ordonnees(i), ordonnees(i)], 'k'
278                 );
279
280             omeg=ordonnees(i);
281             fun=@(x) 0.621945.*(pression_vapPa(
282                 x)./(Ptot-pression_vapPa(x))-
283                 omeg;
284             if i>10
285                 T.vapsat(i)=fsolve(fun, i-10);
286                 plot([T.vapsat(i), Tmax], [
287                     ordonnees(i), ordonnees(i)],
288                     'color',[0.2 0.2 0.2],
289                     'LineWidth',lw/4);
290             elseif i==2
291                 plot([Tmin, Tmax], [ordonnees(i),
292                     ordonnees(i)], 'color',[0.2
293                     0.2 0.2], 'LineWidth',lw/4);
294             else
295                 T.vapsat(i)=fsolve(fun, i);
296                 plot([T.vapsat(i), Tmax.affiche],
297                     [ordonnees(i), ordonnees(i)],
298                     'color',[1 0.2 0.2],
299                     'LineWidth',lw/4);
300             end
301
302             %Et on prolonge en pointilles l'axe
303             des humidites relatives qui est
304             le plus important
305         end
306
307         %% Axes secondaires
308
309         %ordonnees Volume specifique
310         %valeurs
311         line([Tmax Tmax],[0 100], 'color','r')
312
313         %valeurs a calculer !!
314         Vs_ord_min=0.88;
315         Vs_ord_max=0.965;
316         Vs_abs_min=0.7;
317
318         valeurs=[Vs_ord_min:0.005:Vs_ord_max];
319         for k=1:length(valeurs)
320             om=(valeurs(k).*Ptot.*Mv)/(R.*(Tmax
321                 +273.15))-(Mv/Mas);
322             str={valeurs(k)};
323             text(Tmax+0.5,om, str, 'Color','red')
324         end
325         text(Tmax,0.0601,{ 'volume specifique','en
326             m^3/kg'}, 'FontSize',10, 'rotation'
327             ,90, 'Color','red')
328
329         %graduations
330         valeurs_traits=[Vs_ord_min:0.001:
331             Vs_ord_max];
332         for k=1:length(valeurs_traits)
333             om=(valeurs_traits(k).*Ptot.*Mv)/(R
334                 .*(Tmax+273.15))-(Mv/Mas);
335             line([Tmax Tmax+0.4],[om om], 'color',
336                 'r')
337         end
338
339         %abscisses %Volume specifique
340
341         line([Tmin 56],[-0.001 -0.001], 'color','r')
342
343         %valeurs affichees
344         valeurs=[Vs_abs_min:0.01:Vs_ord_min];

```

```

280                                     316     a=valeursPa_traits(k);
281     for k=1:length(valeurs)          317     om=omega_fpression(a);
282     temp=(valeurs(k)*Ptot*Mv-Mv*R*273.15/8 318     line([52 52.4],[om om], 'Color',[0 0.5
        Mas)*Mas/(Mv*R);              0.5])
283     str={valeurs(k)};               319 end
284     text(temp,-0.001,str, 'Color','r') 320
285 end                                  321
286 text(Tmax,-0.001,{ 'volume specifique', 322
        'm^3/kg'}, 'Color','red','FontSize323
        ,10)                          324
287                                     325 logo_Phelma=imread('logo.png');
288 % graduations                       326
289 valeurs_traits=[0.730:0.001:0.915]; 327 imagesc([Tmin+5 Tmin+25], [0.06 0.052],
290 for k=1:length(valeurs_traits)      logo_Phelma)
291     temp=(valeurs_traits(k)*Ptot*Mv-Mv*328
        *273.15/Mas)*Mas/(Mv*R);      329
292     line([temp temp],[-0.001 -0.0015], 330
        'color','r')                  331
293 end                                  332
294                                     333 %A enlever du mode commentaire pour
295                                     334 sauvergarder l'image
296 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 335
297 %valeurs sur l'axe                  336 %
298 valeurs=[0:0.005:0.085];           337 Here we preserve the size of
299 valeursPa=valeurs./1E-5; %passage en ba338 the image when we save it.
300 for k=1:length(valeurs)             339     set(gcf,'InvertHardcopy','on'
301     a=valeursPa(k);                 340     );
302     om=omega_fpression(a);          341     set(gcf,'PaperUnits','inches'
303     str={valeurs(k)};               342     );
304     text(Tmax+2.5,om,str, 'Color',[0 0.5 343     papersize = get(gcf, '
305     0.5])                          344     PaperSize');
306 end                                  345     left = (papersize(1)- width)
307 text(Tmax+2,0.0601,{ 'pression partielle346
        'en kg/cm^2'}, 'FontSize',10, '  /2;
        rotation',90, 'Color',[0 0.5 0.5]) 347     bottom = (papersize(2)-
308                                     348     height)/2;
309 %axe                                349     myfiguresize = [left, bottom,
310 line([Tmax+2 Tmax+2], ylim, 'Color',[0 0.5 350     width, height];
311     0.5]);                          351     set(gcf, 'PaperPosition',
312                                     352     myfiguresize);
313 % graduations                       353
314 valeurs_traits=[0:0.001:0.085];     354 Save the file as PNG
315 valeursPa_traits=valeurs_traits./1E-5; 355 r300 correspond a la definition
316 for k=1:length(valeurs_traits)      356 print('Diagramme','-dpng','-r100'

```