Briefing Notes — Thermodynamic Surfaces

Supporting Material for MECH 222 Computer Lab 4

Motivation. Engineering thermodynamics is a powerful real-world example of what can be accomplished when practical insights and advanced mathematics come together. A great way to build a gut-level understanding of basic thermodynamics and to make the general tools of multivariable calculus come alive is to explore thermodynamic property surfaces. In this computer lab, you will use Matlab and teamwork to help build your own thermodynamic property surface. This solid object, hacked together by cutting shapes from foamboard and taping them together, could be a great aid to the imagination as you work on other surfaces or think more deeply about these specific ones. The ability to imagine curves, surfaces, and solids in 3D is a core skill for a working engineer, and this lab will help you develop it.

IDEA 1: *Two Properties Determine the State*

For a simple compressible pure substance, everything about the thermodynamic state can determined by setting just two properties: one extensive property (normalized by mass to give, for example, v, s, or u), and one other property (either intensive or extensive). You get to choose which two properties to treat as the independent variables, and then all the others will be functions depending on them. For example, you might choose specific volume v and pressure P as the independent properties. Then temperature $T = T(v, P)$, specific internal energy $u = u(v, P)$, entropy $s = s(v, P)$, and all sorts of other properties become functions of the variables v and P. The graph of any such function is a surface in 3D. By now you are familiar with "the PvT surface": you can get this as the collection of points (P, v, T) in which the third component, $T = T(v, P)$, is determined by the first two. [This surface gives a nice visualization for computing the reversible boundary work by integrating $\delta w = P\,dv$.] Alternatively, you could choose v and T as the independent properties that determine all the others, figure out the function $P = P(v, T)$, and generate the *same* PvT surface by plotting all the points $(P(v, T), v, T)$ generated by different choices of v and T.

A huge number of alternatives is possible—e.g., one could plot $u = u(v, P)$ as a function of v and P and call the outcome "the Pvu surface"—but some choices are more practical than others. Another engineering favourite is "the PsT surface", in which we select entropy s and temperature T as the independent properties and illustrate the functional dependence of the pressure $P = P(s, T)$ on these inputs. The PsT surface is "natural" because engineers often work with the T–s diagram, which is simply a decorated contour map of the function $P = P(s, T)$ in the (s, T)-plane. The T–s diagram is useful in understanding thermodynamic processes, partly because these independent variables appear on the right side in

$$\delta q = T\,ds,$$

a key characteristic of a reversible process.

IDEA 2: *A Tour de Force of Careful Experimentation*

There is a big difference between a function and a formula. When we speak of pressure as "a function of" temperature and entropy, we mean that for particular values of T and s, there is exactly one corresponding value for P. Nature makes that true, but nature does not make it easy to calculate P given (s, T). Indeed, there is no simple formula for the properties of water and steam; theoretical derivations based on fundamental concepts in Physics and Chemistry give only approximations, whose scope is limited to certain extreme regions of the parameter space. So we turn to experiments. The International Association for the Properties of Water and Steam (IAPWS) negotiates, validates, and publishes approximate functions suitable for computer calculations. Of course the Association has a website, `http://www.iapws.org/`, and its main page offers a link to "Releases and Guidelines", where visitors can find the full text of the formulas we will use. In detail, our work is based on the *Revised Release on the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam,* dated 2007. This is a simplified version of the IAPWS general theory intended for rapid computation. Insiders call the document by its friendly name, IAWPS-IF97, or its nickname, IF97. In one of the simpler regions of the (P, T) plane, where the pressure is high and the temperature is low, so the water is a liquid well away from the critical point, the state equation is expressed in terms of the specific Gibbs Free Energy g, as follows:

$$g(P, T) = RT \sum_{i=1}^{34} n_i \left(7.1 - \frac{P}{P^*}\right)^{I_i} \left(\frac{T^*}{T} - 1.222\right)^{J_i}, \tag{1}$$

where $R = 0.461526$ kJ/(kg K), $P^* = 16.53 \times 10^6$ Pa, $T^* = 1386$ K, and the coefficients n_i, I_i, and J_i are given to 14-digit accuracy in a large table. Knowing g makes it possible to calculate related quantities like

- specific volume $v = \left(\dfrac{\partial g}{\partial P}\right)_T$,
- specific entropy $s = -\left(\dfrac{\partial g}{\partial T}\right)_P$,
- specific enthalpy $h = g + Ts = g - \left(\dfrac{\partial g}{\partial T}\right)_P$.

Formula (1) is shown here mostly to impress you: our world is a complicated and beautiful place. The IAWPS-IF97 report divides the (P, T) plane into five regions, and gives a different impressive formula in each one.

IDEA 3: *A Public Good*

The technical knowledge captured in IAPWS-IF97 and the other scientific documents on the IAPWS web site is truly impressive. Figuring this stuff out took a lot of work from a large number of very smart people. It's wonderful that the results are freely available to anyone who wants to use them.

Continuing in the stream of generosity, a number of people and organizations have translated the formulas in IAPWS-IF97 into computer programs and online systems that are free to the public. (Others have built systems to make money for doing the same thing; some folks do both.) On the Web, the IAPWS-IF97 formulation is one of the options at

`http://www.steamtablesonline.com/`

Try it! Load that page and select the big red button labelled "Run Calculator". Enter some reasonable values for pressure (bar) and temperature (Celsius) and the page will rapidly calculate and report all the main thermodynamic properties.

A free Matlab implementation of IAPWS-IF97 is also available, from

`http://www.x-eng.com/`

The package is called "X Steam", by Magnus Holmgren. It's good, but not perfect, so you will be invited to use a recent local modification we call `XSteamUBC.m`. This is available for download from Vista.

IDEA 4: *Basic Operations in* `XSteam`

The single Matlab function `XSteamUBC` bundles together 82 thermodynamic functions, giving them a unified format. For example, suppose you are interested in water at temperature 473.15 K and pressure 22 Mpa. You can find the specific volume v and entropy s like this:

```
T = 473.15 - 273.15;       % XSteam wants temperature in Celsius
p = 22.E6 / 1.E5;          % XSteam wants pressure in bar
v = XSteamUBC('v_pT',p,T)  % XSteam selection v_pT will return v=v(p,T)
s = XSteamUBC('s_pT',p,T)  % XSteam selection s_pT will return s=s(p,T)
```

The wrapper function you call is always `XSteamUBC`. The first input argument is a character string describing the desired output (here v or s) and the desired inputs (here p and T). [All the IAPWS documents use lower-case "p" for pressure.] The comments at the beginning of the file `XSteamUBC.m` list all the functions provided and the symbols involved. Most of the variable names are standard.

Single-input functions work in the corresponding way. For example, to confirm that the point where $T = 200$ C and $P = 15.547$ bar is on the saturation curve, you have two options:

```
% OPTION 1:  Look up the pressure using a given temperature
Thot = 200;                       % XSteam wants temperature in Celsius
Phot = XSteamUBC('psat_T',Thot)   % Returned value is in bar
%
% OPTION 2:  Look up the temperature using a given pressure
MyP = 15.547;                     % XSteam wants pressure in bar
MyT = XSteamUBC('Tsat_p',MyP)     % Returned value is in Celsius
```

NOTE: The functions in `XSteam` are not vectorized. Each one does a single calculation and returns a single scalar value. To calculate property values for a long list of input points, you will need to write some kind of a loop to treat one point at a time.

Useful `XSteam` functions. There are 82 functions built into `XSteam`. The short list of 25 shown below mentions more than enough of these to get your work done in today's lab.

```
% Tsat_p  Saturation temperature
% T_ph    Temperature as a function of pressure and enthalpy
% T_ps    Temperature as a function of pressure and entropy
% T_hs    Temperature as a function of enthalpy and entropy


% psat_T  Saturation pressure
% p_hs    Pressure as a function of h and s



% hV_p    Saturated vapour enthalpy
% hL_p    Saturated liquid enthalpy
% hV_T    Saturated vapour enthalpy
% hL_T    Saturated liquid enthalpy
% h_pT    Enthalpy as a function of pressure and temperature
% h_ps    Enthalpy as a function of pressure and entropy


% vV_p    Saturated vapour volume
% vL_p    Saturated liquid volume
% vV_T    Saturated vapour volume
% vL_T    Saturated liquid volume
% v_pT    Specific volume as a function of pressure and temperature
% v_ph    Specific volume as a function of pressure and enthalpy
% v_ps    Specific volume as a function of pressure and entropy
```

```
% sV_p    Saturated vapour entropy
% sL_p    Saturated liquid entropy
% sV_T    Saturated vapour entropy
% sL_T    Saturated liquid entropy
% s_pT    Specific entropy as a function of pressure and temperature
% s_ph    Specific entropy as a function of pressure and enthalpy
```

IDEA 5: *Thermodynamic Contours and Surfaces*

The mathematical theory of parametric curves offers a way to draw contours on thermodynamic diagrams. Consider the famous T–s diagram on your formula sheet, where the horizontal axis shows the interval of s values from 0 to 10 kJ/(kg K) and the vertical axis shows the temperature interval from 0 to 800 C. You can draw any curve you like on these axes by choosing some parameter "t", specifying two functions that express the coordinates $s = s(t)$ and $T = T(t)$ in terms of t, and then joining the points $(s(t), T(t))$ generated by varying t.

Isobars on a T–s diagram. An *isobar* is a curve of constant pressure. To draw an isobar on the T–s diagram, we look for parametric definitions that link T and s with P. One option in `XSteam` is `T_ps`. In mathematical notation, it gives us a function f such that $T = f(P, s)$. A parametric description of the curve where the pressure has some specified value P_0 is then

$$s(t) = t,\ T(t) = f(P_0, t), \qquad s_{\min} \le t \le s_{\max}.$$

The exceptionally simple form for s above suggests dropping the redundant letter t and using s itself as the parameter. The isobar is generated simply by joining the points

$$(s, T(P_0, s)), \qquad s_{\min} \le s \le s_{\max}. \tag{2}$$

You could draw this in Matlab by defining a list of s-values with the `linspace` command, computing a corresponding list of T-values using repeated calls to `XSteam('T_ps',...)`, and then saying something like `plot(s,T)`.

Isobars on the PsT surface. To see the isobars just mentioned in 3D, at a height related to the pressure involved, you would need to extend the parametric equation (2) above by adding a third component to track the pressure. This choice will work:

$$(s, T(P_0, s), P_0), \qquad s_{\min} \le s \le s_{\max}. \tag{3}$$

The space curve could be drawn with Matlab's `plot3` command, something like

```
plot3(s,T,p0*ones(size(s)))
```

Isenthalpic curves on a T–v diagram. An *isenthalpic curve* is a path along which the specific enthalpy h is constant. To draw one on a T–v diagram (i.e., in the (v, T)-coordinate plane), we need a parametric link between T, v, and h. Scanning the list of options available from `XSteam` suggests `v_ph` and `T_ph`: if we select the reference value h_0, we can use pressure as a parameter and trace the curve defined by

$$v = v(p, h_0),\ T = T(p, h_0), \qquad p_{\min} \le p \le p_{\max}. \tag{4}$$

You could draw this in Matlab by defining a list of p-values with the `logspace` command, computing corresponding lists of v and T-values using repeated calls to `XSteam('v_ph',...)`, and `XSteam('T_ph',...)`, and then saying something like `plot(v,T)`.

Isenthalpic curves on the PvT surface. To see the paths where h is constant stuck to the PvT surface, we just need to track all three of the named variables. Equation (4) suggests the parametric collection of points

$$(v(p,h_0), T(p,h_0), p), \qquad p_{\min} \le p \le p_{\max}. \tag{5}$$

The space curve defined here can be drawn with a command like `plot3(v,T,p)`.

Other Constant-Property Curves. On a T–s diagram, like Figure A-9 in your Thermodynamics textbook, we are interested in three families of curves:

(i) constant pressure P (isobars);

(ii) constant specific volume v;

(iii) constant specific enthalpy h.

On the PsT surface, the curves of constant P are quite easy to visualize (they are horizontal slices), but the curves in (ii) and (iii) remain interesting.

On a T–v diagram, we are interested in three families of curves:

(i) constant pressure P (isobars);

(ii) constant specific entropy s; and

(iii) constant specific enthalpy h.

On the PvT surface, the curves of constant P are quite easy to visualize (they are horizontal slices), but the curves in (ii) and (iii) remain interesting.

IDEA 6: *Logarithmic Scales*

For plotting quantities whose magnitudes span a huge range, logarithmic scales are a good choice. Matlab offers two styles of logarithm: `log` computes the natural, or base-e, logarithm, while `log10` computes the common, or base-10, logarithm.

The built-in command `logspace` generates vectors with logarithmically-spaced entries instead of linearly spaced ones, and the common logarithm is used in the definition. Saying

```
x = logspace(-2,3,51)
```

will define a vector with 51 elements from 10^{-2} to 10^3.

When working with P, v, T, s, and h, the pictures look best if we use common-log scales for P and v and simple linear scales for T, s, and h. This explains why these notes suggest `linspace` for building a list of s-values for use in lines (2)–(3), but `logspace` for building a list of p-values for use in lines (4)–(5).

IDEA 7: *Who Does What (and Who Did What)*

Each lab group will build a different surface, with an allocation matched with the name of the famous thinker in the group's title. After each group's surface is assembled it will tour with the group for their thermo tutorials. For example, Archimedes will have one of the two PvT surfaces, and a tutorial might cover the "isentropic compressibility" of a substance. The isentropic compressibility controls the speed of sound in a fluid, and it is directly related to the slope of the surface as you ski Mount Water along an isentrope.

Archimedes. Archimedes students will produce a PvT surface, on which several iso-enthalpy contours will be drawn. Archimedes is famous for explaining the buoyant force on an object immersed in a fluid. Buoyancy can be important for objects immersed in a gas (e.g., dirigibles), but Archimedes more famously worked with liquid water, so you may have a special task involving the liquid portion of the surface.

Bernoulli. The Bernoulli brothers were to Math and Physics what the Bach family was to music (`http://en.wikipedia.org/wiki/Bernoulli`). It was Daniel Bernoulli who gave us the famous equation describing frictionless flow of a fluid. This is a special case of isentropic flow, so Bernoulli students will examine the PsT surface with isenthalpic contours marked on it. This surface will be useful in your task of determining the change in velocity that must occur when a fluid expands isentropically from high to low pressure.

Carnot. Carnot's famous cycle is a rectangle on a T–s diagram. On the PsT suface, this cycle is a more interesting space curve, as you will demonstrate. The letter C can also stand for "Clausius", who taught us how to determine the heat transferred in a reversible process by integrating $T\,ds$.

da Vinci. Da Vinci was the Renaissance man, capable of doing it all. One of the things he did was design a steam cannon (`http://en.wikipedia.org/wiki/Steam_cannon`), based on the idea of rapidly heating a small amount of water injected into a preheated chamber (constant volume before the projectile starts to move). In order to understand this device, it is critical to know the behaviour of the function P(v,T), especially in the range from $1 < P < 400$ bar and $100 < T < 600°C$. Some MIT engineering students actually built a steam cannon. There are videos on YouTube.