RESEARCH AVIATION FACILITY

Algorithm Memorandum:

Al Cooper
7 January 2011

Background

The virtual temperature is the temperature of a dry-air parcel that would have the same density as the actual moist-air parcel. The defining equation arises from the equation of state:

$$p = \rho_m R_m T = \rho_m R_d T_v \tag{1}$$

where p is the total pressure, ρ_m the density of moist air, R_m the gas constant for moist air (equal to R^*/M_m where R^* is the universal gas constant and M_m is the molecular weight of the moist air), $R_d = R^*/M_d$ is the gas constant for dry air of molecular weight M_d, T is the absolute temperature, and T_v is the absolute virtual temperature. Because the pressures are additive, (1) can also be written as

$$p = p_d + e = \rho_d \frac{R^*}{M_d} T + \rho_v \frac{R^*}{M_w} T \tag{2}$$

where R^* is the universal gas constant, M_d and M_w are the respective molecular weights of dry air and water, and ρ_d and ρ_v are the respective densities. Because $\rho_m = \rho_d + \rho_e$, equating (1) and (2) as expressions for the pressure gives

$$(\rho_d + \rho_v) \frac{R^*}{M_m} T = \rho_d \frac{R^*}{M_d} T + \rho_v \frac{R^*}{M_w} T \tag{3}$$

which leads to the gas constant for moist air:

$$R_m = \frac{\rho_d}{\rho_m} R_d + \frac{\rho_v}{\rho_m} R_w \tag{4}$$

with $R_w = R^*/M_w$ the gas constant for water vapor. The mixing ratio is $r = \rho_v/\rho_d$ so

$$R_m = \frac{1}{1+r} R_d + \frac{r}{1+r} R_w = R_d \frac{1 + r/\varepsilon}{1+r} \tag{5}$$

where $\varepsilon = R_d/R_w = M_w/M_d \simeq 0.622$. Then (1) gives the following equation for the absolute virtual temperature:

$$T_v = T \left(\frac{1 + r/\varepsilon}{1+r} \right) \simeq T \left(\frac{1 + 1.6078r}{1+r} \right) \tag{6}$$

Alternately, because $q = \rho_v/\rho_m$ is the dimensionless specific humidity and $\rho_d = \rho_m - \rho_v$, this can be written as

$$R_m = (1-q)R_d + qR_w = R_d(1-q(1-\frac{1}{\varepsilon})) = R_d(1+q\frac{1-\varepsilon}{\varepsilon}) \tag{7}$$

Then, using (1) gives the equation for the absolute virtual temperature:

$$T_v = \frac{R_m}{R_d}T = T\left(1+q\left(\frac{1-\varepsilon}{\varepsilon}\right)\right) \simeq T\left(1+0.6078q\right) \tag{8}$$

which also could have been obtained immediately from (6) using the relationships $q = r/(1+r)$ or $r = q/(1-q)$.

The virtual potential temperature Θ_v is defined analogously as the potential temperature of dry air that would have the same density as the moist air. For unsaturated air, the result is

$$\Theta_v = T_v(\frac{p_r}{p})^{R_d/c_{pd}} \tag{9}$$

where p_r is the reference pressure (conventionally 1000 hPa) and R_d/c_{pd} is normally taken to be 2/7, which would be the exact result if dry air were a perfect gas and which is very close to measured values. The subtlety in this equation is in the use of the total pressure p with values for the exponent that apply to dry air. The AMS Glossary says that weight of any liquid water should be included, in which case the appropriate virtual temperature would be modified to incorporate that weight. The actual formula in the Glossary[1] is $\Theta_v = \Theta(1+0.61r-r_w)$ where r_w is the mixing ratio of liquid water and Θ is specified to be "the actual potential temperature". Interpreted literally, this would require use of a gas constant and specific heat that are adjusted for the moisture in the air, not the dry-air values in (9). Because the uses for T_v and Θ_v are as surrogates for buoyancy, it is desirable to preserve the relationship to buoyancy as far as possible. If R_m/c_{pm} (applicable to moist air) were to be used, then variations in Θ_v would be introduced by humidity both via T_v and via R_m/c_{pm}, and the connection to buoyancy would be lost. To preserve this connection, one must used a fixed exponent in (9), even though this contradicts the specification in the AMS Glossary. The weakness in this approach is that vertical movements of parcels will obey potential temperature equations with the actual moist-air coefficients, so the benefit of retaining exact correspondence to buoyancy at the measurement level is lost when parcels move vertically. [2]

[1] small-value approximations have been used here; I think the formula without approximations would be $\Theta_v = \Theta(1+r/\varepsilon)/(1+r+r_W)$ where $r_w = \chi/\rho$ is the mixing ratio of liquid water, χ is the liquid water content [mass/volume], and ρ is the total air weight per volume including the weight of condensed water.

[2] This could be remedied by introducing a different formula for Θ_v in which T_v is replaced by T_v^* and the moist-air exponent is used, but T_v^* includes an additional adjustment to compensate for the variation in buoyancy introduced by using the actual moist-air exponent. This does not seem to be a good alternative for an RAF output variable because there is no precedent for its use.

Present Processing Code

TVIR is not in recent datasets, and the code in the program segment to calculate it (tvir.c) is obviously not functional, so TVIR should not be used without change.

Here is the current code for virtual potential temperature THETAV:

```
void sthetav(DERTBL *varp) {
  NR_TYPE thetav = 0.0, atx, tvir, psxc, edpc, sphum;
  atx = GetSample(varp, 0);
  psxc = GetSample(varp, 1);
  edpc = GetSample(varp, 2);
  sphum = psxc - 0.378 * edpc;
  if (sphum != 0.0)     sphum = 622.0 * edpc / sphum;
  tvir = 1.0 - 0.6e-3 * sphum;
  if (tvir != 0.0)
    tvir = (atx + Kelvin) / tvir - Kelvin;
  if (psxc != 0.0)
    thetav = (tvir + Kelvin)
           * pow( (double)(1000.0 / psxc), (double)0.28571);
  PutSample(varp, thetav);
}
```

This equation recalculates the specific humidity, also available in the subroutine sphum.c, and it duplicates the code from sphum.c. This is correct apart from an insignificant departure from the exact best-current-estimate value for the ratio of molecular weights: $\varepsilon = 0.62198$. SPHUM, however, has not been included in recent data files. It can always be calculated readily from the mixing ratio r, which is present: $q = r/(1+r)$. For THETAV, the code in use is equivalent to (8), except for the use of $(1-0.6q)^{-1}$ in place of $(1+0.6078q)$. The difference is insignificant for levels of humidity normally encountered in the atmosphere.

Reasons For Proposing Changes

1. Virtual temperature is a useful variable and it would be good to include it in processed data files. Its calculation is less ambiguous than that for virtual potential temperature.

2. The relevant code fragment tvir.c is obviously not functional and will produce serious errors. If it is desirable to calculate virtual temperature, that routine should be updated.

3. Although the calculation of Θ_v is suitably accurate, the equation could easily be replaced by a form that does not make unnecessary approximations. The inaccuracies arise primarily from the use of $(1-0.6q)^{-1}$ in place of the correct form $\left(1+q\left(\frac{1-\varepsilon}{\varepsilon}\right)\right) \simeq (1+0.6078q)$. Changing this would also save users and future reviewers from having to wonder why the form being used is not the conventional one.

4. Duplicating the code for calculation of specific humidity in this routine runs the risk that a future change to that code, either in this routine or in "sphum.c", would lead to inconsistency. Specific humidity would also be a good variable to include in data files, even though users can easily recalculate it. For virtual temperature and virtual potential temperature, however, it may be preferable to use (6) because it uses mixing ratio instead.

Recommendations

1. Calculate virtual temperature (TVIR) as follows:

```
// (in global data somewhere, MW = 18.01526 and MD = 28.9644)
// input is ATX [deg C] and MR [g/kg]
// output is TVIR [deg C]
// factors of 0.001 are conversion factors to obtain
//     dimensionless mixing ratios [kg/kg]
TVIR = (ATX + KELVIN) * (1. + 0.001 * MR * MD / MW)
     / (1. + 0.001 * MR) - KELVIN;
```

2. Calculate potential virtual temperature (THETAV) as follows:

```
// input is TVIR and PSXC; output is thetav [kelvin]
thetav = (TVIR + KELVIN) * pow((1000./PSXC), 0.28571);
```

—- END —-