

Comparing different versions of WINDOW to quantify impact of angle dependence code

Jacob C. Jonsson 3/4/2022

Goal:

Test to see how large impact the angle dependent code updates have on different systems. Going through from 7.4.14, 7.7.10, 7.8.28 to 7.8.39 (version dir-dif build Stephen made for me) gives some ability to see how much impact is due to the dir-dif code and how much is unrelated.

Result:

The difference in Tsol and Rsol between 7.8.28 and 7.8.39 was less than 0.004 for realistic cases which is reasonable based on the changes that were made to the angle-dependence code.

Optical calc settings:

Condensed (default 5vis, 10nir points), W6 quarter basis, do not do matrix if not needed. Venetian blind calculation is Directional-diffuse. The exception to these settings were case 13 which had full spectral range instead of condensed.

Test cases:

Databases stored in

gdrive/Shared Drive/IGDB/CGDB/InterimProcess/AoITesting

[specular single pane]

1. Clear3.dat
2. solarban 70_6.vta (NFRC ID 5439, default coating on S2)
This product was selected as it is a coated product with T=0 for a lot of the wavelengths in the NIR.

[specular double pane]

3. Clear3.dat + Clear3.dat
4. solarban 70_6.vta (coating on S2) + Clear3.dat
5. solarban 70_6.vta (S2) + solarban 70_6.vta (coating on S3)

[inbetween blind]

The Venetian A45 is the ISO test case with only visible and solar values (2-band) and the White satin case has spectral data.

6. Clear3.dat + VenetianA45 + Clear3.dat
7. solarban 70_6.vta (coating on S2) + VenetianA45 + Clear3.dat
8. solarban 70_6.vta (coating on S2) + White Satin Open 24003 + Clear3.dat

9. solarban 70_6.vta (coating on S2) + White Satin Closed 24004 + Clear3.dat
10. solarban 70_6.vta (coating on S2) + White Satin +45 24005 + Clear3.dat
11. solarban 70_6.vta (coating on S2) + White Satin -45 23998 + Clear3.dat
12. clear3.dat + White Satin -45 23998 + Clear3.dat
13. solarban 70_6.vta (coating on S2) + White Satin +45 24005 + Clear3.dat (*Full spectral*)

Great results for case 1-5, i.e. specular only with real products

Looked at Angular data results and not being able to copy the whole matrix decided to look at 60 degrees Tsol and Rfsol values. Identical to results for both versions of WINDOW. Cases 2,4,5 should be impacted by the code change ($T(\lambda)=0$ for several wavelengths of ID 5439). That the impact is so small that it does not show up in the usual precision for a realistic case is a good sign that we can include this.

Small difference in cases with inbetween Venetian blind

The angular report available for specular is not available for layers with shade. So I simply looked at Tsol and Rfsol in the Optical Data result tab.

Perfect agreement seen between 7.4 and 7.7 for all cases. No difference in case 6.

Figure 3 below focuses on difference between 7.8.28 and 7.4 as well as 7.8.39 to 7.8.28, the magenta triangles is the best measurement on the impact of the dir-dif code.

Case 13 uses full spectral data, no significant difference on the 7.8.39 changes.

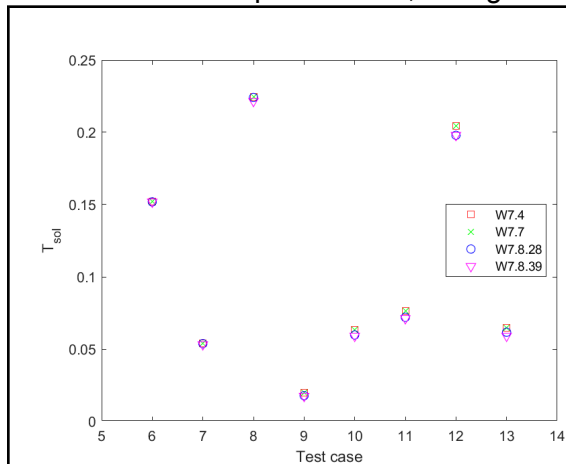


Fig. 1: Absolute Tsol value for cases 6-12.

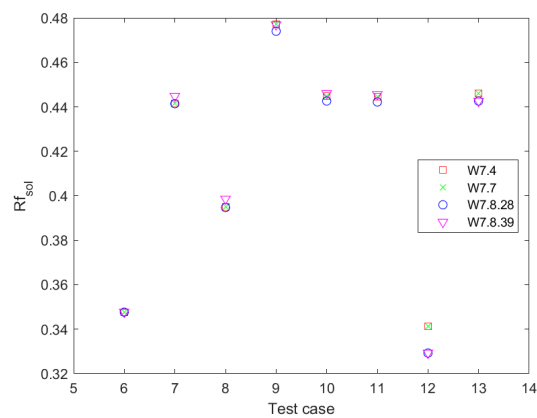


Fig. 2: Absolute Rfsol for cases 6-12

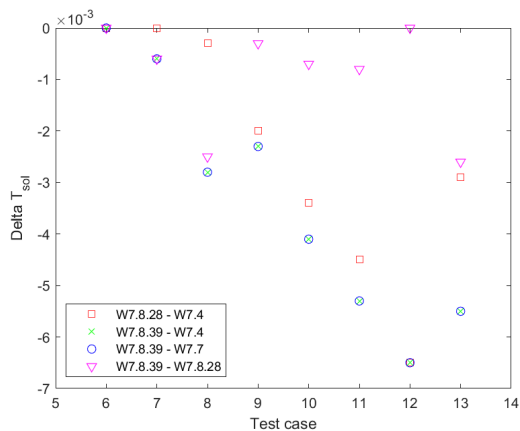


Fig. 3: Exact agreement for case 6. Largest difference in T_{sol} is seen between 7.8.28 and 7.4 is close to 0.007. 7.839 and 7.8.28 are within 0.003 for all cases.

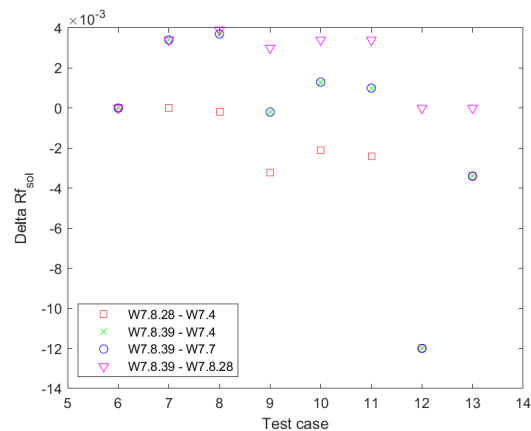


Fig. 4: Exact agreement for cases 6. less than 0.014 difference for cases 8-13. Largest difference in R_{sol} is seen between 7.8.28 and 7.4 is close to 0.012. 7.839 and 7.8.28 are within 0.004 for all cases.

From Stephen:

This is a list of the four changes I have found (so far) in the code that could impact results for non-dir-dif systems:

that seem to have the potential to affect results from things other than dir-dif.

1. The ANGHOMO change where R_theta is set to r_normal instead of 1 when t_normal = 0. i.e. changing

```
if (Ttot0. le. 0 .and. th .ne. 0) then
  Ttot_th = 0
  Rtot_th = 1
  RETURN
end if
```

to

```
if (Ttot0. le. 0 .and. th .ne. 0) then
  Ttot_th = 0
  Rtot_th = Rtot0
  RETURN
end if
```

2. In the angular_dependence function for coated glass added a check to see if the angle was normal incidence. If it is normal incidence then do not do the angular calculation and just return the input values. i.e.

```
// If normal angle return normal values instead of doing calculations.
if(FEQUAL_TOL(angle, 0.0, 1e-8))
{
    *T = tNorm;
    *Rf = rfNorm;
    *Rb = rbNorm;
    return ret;
}
```

3. Also in the angular_dependence function for coated glass if any of the normal values are 0 (note in this case zero is within e-4 not e-8 like above) then return 0 instead of the calculated value. i.e.

```
// if any of the norms are 0 return 0 for that instead of the calculated result
    if(FEQUAL_TOL(tNorm, 0.0, 1e-4))
    {
        *T = 0;
    }

    if(FEQUAL_TOL(rfNorm, 0.0, 1e-4))
    {
        *Rf = 0;
    }

    if(FEQUAL_TOL(rbNorm, 0.0, 1e-4))
    {
        *Rb = 0;
    }
```

4. There is a function that converts values outside of the range "zero" to "one" to either "zero" or "one". I use quotes there because we changed "zero" from 0.0000001 to 0.0 and "one" from 0.9999999 to 1.0.

This function is only used in the coated glass section of angular_dependence and is applied to the calculated t, rf, and rb values. So previously calculated angular dependence results could only be in the range [0.0000001, 0.9999999] but can now be in range [0, 1]