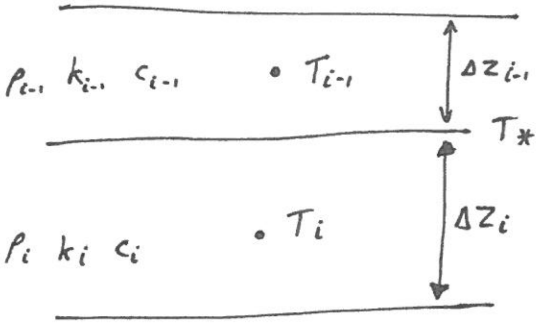


Find flux from layer $(i-1) \rightarrow (i)$

Find boundary temperature first.
Flux is continuous across the boundary.

$$\text{Flux}_{(i-1)\rightarrow(i)} = \frac{k_{i-1}\left(T_{i-1} - T_*\right)}{\frac{1}{2}\Delta z_{i-1}} = \frac{k_i\left(T_* - T_i\right)}{\frac{1}{2}\Delta z_i}$$

so: $$\boxed{T_* = \frac{k_i\,\Delta z_{i-1}\,T_i + k_{i-1}\,\Delta z_i\,T_{i-1}}{k_i\,\Delta z_{i-1} + k_{i-1}\,\Delta z_i}}$$

$$\boxed{\text{Flux } (i-1)\rightarrow(i) = \frac{2\,k_i\,k_{i-1}}{k_i\,\Delta z_{i-1} + k_{i-1}\,\Delta z_i}\left[T_{i-1} - T_i\right]}$$

(sub T_* back into flux expression)

similarly:

$$\boxed{\text{Flux } (i+1)\rightarrow(i) = \frac{2\,k_i\,k_{i+1}}{k_i\,\Delta z_{i+1} + k_{i+1}\,\Delta z_i}\left[T_{i+1} - T_i\right]}$$

$$\rho_i\,c_i\,\Delta z_i\left(T_i^{k+1} - T_i^{k}\right) = \Delta t\left[\text{Flux}_{(i-1)\rightarrow i} + \text{Flux}_{(i+1)\rightarrow i}\right]$$

Define:

$$\alpha_{ui} = \left[\frac{2\,k_i\,k_{i-1}}{k_i\,\Delta z_{i-1} + k_{i-1}\,\Delta z_i}\right]\frac{\Delta t}{\rho_i\,c_i\,\Delta z_i}$$

$$\alpha_{di} = \left[\frac{2\,k_i\,k_{i+1}}{k_i\,\Delta z_{i+1} + k_{i+1}\,\Delta z_i}\right]\frac{\Delta t}{\rho_i\,c_i\,\Delta z_i}$$

$$T_i^{k+1} - T_i^{k} = \begin{cases} \alpha_{ui}\left[T_{i-1}^{k} - T_i^{k}\right] + \alpha_{di}\left[T_{i+1}^{k} - T_i^{k}\right] & \underline{\text{Explicit}} \\ \alpha_{di}\left[T_{i-1}^{k+1} - T_i^{k+1}\right] + \alpha_{di}\left[T_{i+1}^{k+1} - T_i^{k+1}\right] & \underline{\text{Implicit}} \end{cases}$$

There's no need to evaluate T_*; however, it's sometimes useful e.g. Temperature at the ice table when layer i is ice and layer $(i-1)$ is regolith.

explicit

$$T_i^{k+1} - T_i^k = \alpha_{ui}\left[T_{i-1}^k - T_i^k\right] + \alpha_{di}\left[T_{i+1}^k - T_i^k\right]$$

$$T_i^{k+1} = \begin{array}{ccc} k3 & dia & k_1 \end{array}$$
$$\left[(\alpha_{ui}) \quad (1-\alpha_{ui}-\alpha_{di}) \quad (\alpha_{di})\right]\begin{bmatrix} T_{i-1}^k \\ T_i^k \\ T_{i+1}^k \end{bmatrix}$$

Implicit

$$T_i^{k+1} - T_i^k = \alpha_{ui}\left[T_{i-1}^{k+1} - T_i^{k+1}\right] + \alpha_{di}\left[T_{i+1}^{k+1} - T_i^{k+1}\right]$$

now have α_{up}'s + α_{down}'s off diagnol not same anymore

$$\left[(-\alpha_{ui}) \quad (1+\alpha_{ui}+\alpha_{di}) \quad (-\alpha_{di})\right]\begin{bmatrix} T_{i-1}^{k+1} \\ T_i^{k+1} \\ T_{i+1}^{k+1} \end{bmatrix} = T_i^k$$

Semi-Implicit (f=0 means explicit)

$$T_i^{k+1} - T_i^k = \overbrace{(1-f)\alpha_{ui}\left[T_{i-1}^k - T_i^k\right] + (1-f)\alpha_{di}\left[T_{i+1}^k - T_i^k\right]}^{(1-f) * \text{explicit}} + \overbrace{f\alpha_{ui}\left[T_{i-1}^{k+1} - T_i^{k+1}\right] + f\alpha_{di}\left[T_{i+1}^{k+1} - T_i^{k+1}\right]}^{f * \text{implicit}}$$

implicit

$$\left[(-f\alpha_{ui}) \quad (1+f\alpha_{ui}+f\alpha_{di}) \quad (-f\alpha_{di})\right]\begin{bmatrix} T_{i-1}^{k+1} \\ T_i^{k+1} \\ T_{i+1}^{k+1} \end{bmatrix} = \begin{array}{ccc} k3 & \text{explicit dia} & k1 \end{array}$$
$$\left[((1-f)\alpha_{ui}) \quad (1-(1-f)\alpha_{ui}-(1-f)\alpha_{di}) \quad ((1-f)\alpha_{di})\right]\begin{bmatrix} T_{i-1}^k \\ T_i^k \\ T_{i+1}^k \end{bmatrix}$$

<u>As before</u>: ① Evaluate r.h.s.

② Use tridiagonal solver to find T^{k+1}'s.

for generic layer in middle

Special cases $i = N$

$$T_N^{k+1} - T_N^{k} = \underbrace{\frac{\Delta t}{\rho_N C_N \Delta Z_N} \cdot Q}_{\hookrightarrow \text{ upward flux}} + \underbrace{\alpha_{uN} \left[T_{N-1} - T_N \right]}_{\hookrightarrow \text{ downward flux}}$$

use $k/k{+}1$ for explicit/~~explicit~~ implicit

semi-implicit $i=N$

$$T_N^{k+1} - T_N^{k} = \frac{\Delta t . Q}{\rho_N C_N \Delta Z_N} \left[(1-f) + f \right] + (1-f)\alpha_{uN} \left[T_{N-1}^{k} - T_N^{k} \right] + f\alpha_{uN} \left[T_{N-1}^{k+1} - T_N^{k+1} \right]$$

$$\begin{bmatrix} (-f\alpha_{uN}) & (1+f\alpha_{uN}) \end{bmatrix} \begin{bmatrix} T_{N-1}^{k+1} \\ T_N^{k+1} \end{bmatrix} = \begin{bmatrix} ((1-f)\alpha_{uN}) & (1-(1-f)\alpha_{uN}) \end{bmatrix} \begin{bmatrix} T_{N-1}^{k} \\ T_N^{k} \end{bmatrix} + \underbrace{\begin{bmatrix} \frac{\Delta t \; Q}{\rho_N C_N \Delta Z_N} \end{bmatrix}}_{\text{Boundary } (N)}$$

last element of explicit diagnol

i.e. $\alpha_{dN} = 0$

Special case $i=1$

$\alpha_{d1} \neq \alpha_{u2}$

energy is same but converting to T change not

$$T_1^{k+1} - T_1^{k} = \underbrace{\frac{k_1 \, \Delta t}{\rho_1 c_1 \Delta z_1^2} \, 2\left(T_s - T_1\right)}_{\text{downward flux}} + \underbrace{\alpha_{d1} \left(T_2 - T_1\right)}_{\text{upward flux}}$$

(use $k/k{+}1$ for explicit/implicit)

$$T_1^{k+1} - T_1^{k} = \underbrace{\left[\frac{k_1 \, \Delta t}{\rho_1 c_1 \Delta z_1^2}\right]}_{\equiv \beta} 2a - (2-2b)\left[\frac{k_1 \Delta t}{\rho_1 c_1 \Delta z_1^2}\right] T_1 + \alpha_{d1} \left(T_2 - T_1\right)$$

$$T_1^{k+1} - T_1^{k} = 2a\beta - \left[\alpha_{d1} + (2-2b)\beta\right] T_1 + \alpha_{d1} T_2$$

semi-implicit $i=1$

β like α in 1st boundary condition for semi-implicit case w/ same properties

$$T_1^{k+1} - T_1^{k} = (1-f)2a^k\beta + f2a^{k+1}\beta - (1-f)T_1^{k} \left[\alpha_{d1} + (2-2b)\beta\right] - f T_1^{k+1} + \alpha_{d1} T_2^{k}(1-f) + \alpha_{d1} T_2^{k+1} f$$

Factor goes here too

implicit matrix 1st element

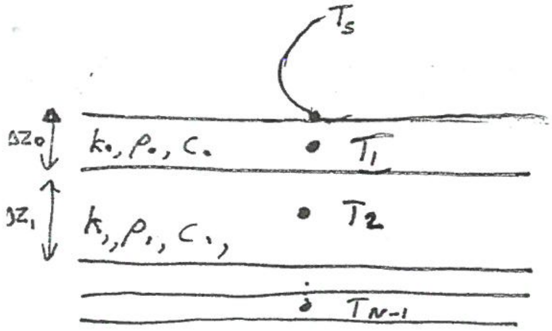
explicit diag 1st element

$$\begin{bmatrix} \left(1 + f\left[\alpha_{d1} + (2-2b)\beta\right]\right) & (-f\alpha_{d1}) \end{bmatrix} \begin{bmatrix} T_1^{k+1} \\ T_2^{k+1} \end{bmatrix} = \begin{bmatrix} \left(1 - (1-f)(\alpha_{d1} + (2-2b)\beta)\right) & (\alpha_{d1}(1-f)) \end{bmatrix} \begin{bmatrix} T_1^{k} \\ T_2^{k} \end{bmatrix} + \underbrace{\begin{bmatrix} 2\beta\left((1-f)a^k + fa^{k+1}\right) \end{bmatrix}}_{\text{Boundary } (1)}$$

$(3-2b)\alpha$ when properties the same

i.e. $\alpha_{u1} = 0$

β has essentially replaced this



Different material properties

Replace radiative terms with conduction.

$$IR_{\downarrow} + F_0 \cos(i)(1-A) - \underbrace{\varepsilon \sigma T_s^4} = \frac{2k_0}{\Delta z_0}(T_s - T_1)$$

Pick T_R as some reference temperature that's close to T_s.

i.e. $(T_s - T_R) \ll T_R$

Using T_s from the last timestep works well.

want less than ~1%

when CO_2 frost runs out and sun beating down or east facing slope, can get sudden jump in T that can be more than this

$$\varepsilon\sigma\left(T_R + (T_s - T_R)\right)^4$$

$$\varepsilon\sigma T_R^4\left(1 + \frac{T_s - T_R}{T_R}\right)^4$$

linearize:

$$\varepsilon\sigma T_R^4\left(1 + 4\frac{T_s - T_R}{T_R}\right)$$

$$\varepsilon\sigma T_R^4 + 4\varepsilon\sigma T_R^3 (T_s - T_R)$$

$$(4\varepsilon\sigma T_R^3) T_s - 3\varepsilon\sigma T_R^4$$

linearization of surface temp

if Tref changes a lot between timesteps might get wrong T's ~~for last~~ for several timesteps could drive explicit to instability but implicit will settle out

$$IR_{\downarrow} + F_0 \cos(i)(1-A) + 3\varepsilon\sigma T_R^4 = (4\varepsilon\sigma T_R^3) T_s + \left(\frac{2k_0}{\Delta z_0}\right) T_s - \frac{2k_0}{\Delta z_0} T_1$$

$$\left[\frac{\left(IR_{\downarrow} + F_0 \cos(i)(1-A) + 3\varepsilon\sigma T_R^4\right)}{\left(4\varepsilon\sigma T_R^3 + \frac{2k_0}{\Delta z_0}\right)}\right] + \left[\frac{\left(\frac{2k_0}{\Delta z_0}\right)}{\left(4\varepsilon\sigma T_R^3 + \frac{2k_0}{\Delta z_0}\right)}\right] T_1 = T_s$$

When frost free: $T_s = a + b \cdot T_1$

When frosted: $T_s = T_{frost}$

$$a + b\,T_1 = T_s$$

diff each timestep

$$a = \frac{\frac{\Delta z}{2k_0}\left(IR_{\downarrow} + F_0 \cos i\,(1-A) + 3\varepsilon\sigma T_R^4\right)}{1 + \frac{4\varepsilon\sigma T_R^3 \Delta z}{2k}} \qquad b = \frac{1}{1 + \frac{4\varepsilon\sigma T_R^3 \Delta z}{2k}}$$

contains solar flux info

$T_s = a + b\,T_1$

Explicit

top layer

$$\Delta T_\emptyset = \underbrace{\frac{k\,\Delta t}{\rho c\,\Delta z^2}}_{\equiv\,\alpha}\left[2\,(T_s - T_1) + (T_2 - T_1)\right]$$

1/2 layers away

$T_1^{k+1} - T_1^{k}$

$$\Delta T_\emptyset = \alpha\left[(2b - 3)\,T_1 + T_2 + 2a\right]$$

$$\Delta T_\emptyset = \left[(2b-3)\,\alpha\right] T_1 + \alpha T_2 + 2\alpha a$$

$$T_\emptyset^{k+1} = \left[1 - (3 - 2b^k)\,\alpha\right] T_1^k + \alpha T_2^k + (2\alpha a^k)$$

Q positive

$$T_B = \left(\frac{Q\,\Delta z}{2k}\right) + T_N$$

bottom layer

(slightly hotter than bottom layer)

$$\Delta T_N = \frac{k\,\Delta t}{\rho c\,\Delta z^2}\left[2\,(T_B^k - T_N^k) + (T_{N-1}^k - T_N^k)\right]$$

$$\Delta T_N = \alpha\left[\left(\frac{Q\,\Delta z}{k}\right) + (2-3)\,T_N^k + T_{N-1}^k\right]$$

$T_N^{k+1} - T_N^{k}$

$$\Delta T_N = -\alpha T_N^k + \alpha T_{N-1}^k + \alpha\frac{Q\,\Delta z}{k}$$

$$T_N^{k+1} = \alpha T_{N-1}^k + (1-\alpha)\,T_N^k + \left(\alpha\frac{Q\,\Delta z}{k}\right)$$

$j: 2 \rightarrow N-1$

timestep

$$T_j^{k+1} = \alpha\,T_{j-1}^k + (1-2\alpha)\,T_j^k + \alpha\,T_{j+1}^k + \emptyset$$

layer

In general:

k2

diffuse between layers - energy only moves one layer above/below for a given timestep

but just going in/out of 2 boundaries - middle wouldn't be 0 if there was radiogenic heating/source terms

k3 k1

$$T_j^{k+1} = \begin{bmatrix} [1-(3-2b^k)\alpha] & \alpha & 0 & 0 & 0 \\ \alpha & (1-2\alpha) & \alpha & 0 & 0 \\ 0 & \alpha & (1-2\alpha) & \alpha & 0 \\ 0 & 0 & \alpha & (1-2\alpha) & \alpha \\ 0 & 0 & 0 & \alpha & (1-\alpha) \end{bmatrix} \begin{bmatrix} T_1^k \\ T_2^k \\ T_3^k \\ T_4^k \\ T_5^k \end{bmatrix} + \begin{bmatrix} 2\alpha a^k \\ 0 \\ 0 \\ 0 \\ \frac{\alpha Q\,\Delta z}{k} \end{bmatrix}$$

update these each timestep

a, b change each timestep (w/ Tref)

Simple matrix multiplication + addition.

Implicit

middle layers

$\boxed{j: 2 \to N-1}$

$$T_j^{k+1} - T_j^k = \alpha\left[T_{j+1}^{k+1} + T_{j-1}^{k+1} - 2T_j^{k+1}\right] \quad \text{Implicit}$$

ΔT_j

$$T_j^k = -\alpha T_{j-1}^{k+1} + (1+2\alpha)T_j^{k+1} - \alpha T_{j+1}^{k+1}$$

$\boxed{j=1}$

$$T_1^{k+1} - T_1^k = \alpha\left[2\left(T_s^{k+1} - T_1^{k+1}\right) + \left(T_2^{k+1} - T_1^{k+1}\right)\right]$$

$$T_1^{(k+1)} - T_1^k = 2\alpha a^{k+1} + (2b^{k+1}-3)\alpha T_1^{k+1} + \alpha T_2^{k+1}$$

$$T_1^k = -2\alpha a^{k+1} \quad [1-(2b^{k+1}-3)\alpha]T_1^{k+1} - \alpha T_2^{k+1}$$

$$T_1^k + (2\alpha a^{k+1}) = [1-(2b^{k+1}-3)\alpha]T_1^{k+1} - \alpha T_2^{k+1}$$

$\boxed{j=N}$

$$T_N^{k+1} - T_N^k = \alpha\left(2\left(T_B^{k+1} - T_N^{k+1}\right) + \left(T_{N-1}^{k+1} - T_N^{k+1}\right)\right)$$

$$T_N^{k+1} - T_N^k = \alpha\left(\frac{Q\Delta z}{k}\right) - \alpha T_N^{k+1} + \alpha T_{N-1}^{k+1}$$

$$T_N^k + \left(\frac{Q\Delta z}{k}\right)\alpha = (1+\alpha)T_N^{k+1} - \alpha T_{N-1}^{k+1}$$

In general:

$$T_j^k + \underbrace{\begin{pmatrix} (2\alpha a^{k+1}) \\ 0 \\ 0 \\ 0 \\ \left(\frac{Q\Delta z}{k}\alpha\right) \end{pmatrix}}_{\text{simple matrix addition}} = \underbrace{\begin{pmatrix} 1-(2b^{k+1}-3)\alpha & -\alpha & 0 & 0 & 0 \\ -\alpha & (1+2\alpha) & -\alpha & 0 & 0 \\ 0 & -\alpha & (1+2\alpha) & -\alpha & 0 \\ 0 & 0 & -\alpha & (1+2\alpha) & -\alpha \\ 0 & 0 & 0 & -\alpha & (1+\alpha) \end{pmatrix} T_j^{k+1}}$$

step 1 simple matrix addition

step 2 Solve this tridiagonal system with Gaussian elimination

Semi - implicit

J: 2 → N-1

$$T_j^{k+1} - T_j^k = \underbrace{f}_{} \overbrace{\alpha\left(T_{j+1}^{k+1} - 2T_j^{k+1} + T_{j-1}^{k+1}\right)}^{\text{Implicit}} + \underbrace{(1-f)}_{} \overbrace{\alpha\left(T_{j+1}^{k} - 2T_j^{k} + T_{j-1}^{k}\right)}^{\text{explicit}}$$

weighting factors

$f=0$: explicit

$f=1$: Implicit

$f=0.5$: Crank-Nicolson

$$(-\alpha f)\, T_{j-1}^{k+1} + (1 + 2f\alpha)\, T_j^{k+1} + (-\alpha f)\, T_{j+1}^{k+1} = (1-f)\alpha\, T_{j-1}^{k} + (1-2(1-f)\alpha)\, T_j^{k} + (1-f)\alpha\, T_{j+1}^{k}$$

J=1

$$T_1^{k+1} - T_1^k = \overbrace{(1-f)\alpha\left[(2b^k-3)\, T_1^k + T_2^k + 2a^k\right]}^{\text{explicit}} + \overbrace{f\alpha\left[(2b^{k+1}-3)\, T_1^{k+1} + T_2^{k+1} + 2a^{k+1}\right]}^{\text{Implicit}}$$

$$\left[1 - f\alpha(2b^{k+1}-3)\right] T_1^{k+1} + (-f\alpha)\, T_2^{k+1} = \left[1 + (1-f)\alpha(2b^k-3)\right] T_1^k + (1-f)\alpha\, T_2^k + 2\alpha\left[(1-f)a^k + (f)\, a^{k+1}\right.$$

J=N

$$T_N^{k+1} - T_N^k = (1-f)\alpha\left[T_{N-1}^k - T_N^k + \frac{Q\Delta z}{k}\right] + f\alpha\left[T_{N-1}^{k+1} - T_N^{k+1} + \frac{Q\Delta z}{k}\right]$$

$$(-f\alpha)\, T_{N-1}^{k+1} + (1+f\alpha)\, T_N^{k+1} = (1-f)\alpha\, T_{N-1}^k + (1-(1-f)\alpha)\, T_N^k + \left[\alpha \frac{Q\Delta z}{k}\right]$$

much simpler than top. layer as Q is a constant and $Q^k = Q^{k+1}$

Semi-Implicit : The saga continues !

In general

$$
\overset{\text{implicit}}{\begin{pmatrix}
(1-f\alpha(2b^{k+1}-3)) & (-f\alpha) & 0 & 0 & 0 \\
(-\alpha f) & (1+2\alpha f) & (-\alpha f) & 0 & 0 \\
0 & (-\alpha f) & (1+2\alpha f) & (-\alpha f) & 0 \\
0 & 0 & (-\alpha f) & (1+2\alpha f) & (-\alpha f) \\
0 & 0 & 0 & (-\alpha f) & (1+\alpha f)
\end{pmatrix}} T_j^{k+1} = \overset{\text{explicit}}{\begin{pmatrix}
(1+(1-f)\alpha(2b^{k}-3)) & ((1-f)\alpha) & 0 & 0 & 0 \\
((1-f)\alpha) & (1-2(1-f)\alpha) & ((1-f)\alpha) & 0 & 0 \\
0 & ((1-f)\alpha) & (1-2(1-f)\alpha) & ((1-f)\alpha) & 0 \\
0 & 0 & ((1-f)\alpha) & (1-2(1-f)\alpha) & ((1-f)\alpha) \\
0 & 0 & 0 & ((1-f)\alpha) & (1-(1-f)\alpha)
\end{pmatrix}} T_j^{k}
$$

$$
+ \begin{pmatrix}
2\alpha[(1-f)a^{k}+fa^{k+1}] \\
0 \\
0 \\
0 \\
\left(\dfrac{\alpha Q \Delta z}{K}\right)
\end{pmatrix}
$$

Step 2: Solve this tridiagonal system with Gaussian elimination.

Step 1: Do simple matrix multiplication & addition

Note: if $f=0$ then this matrix is:
$$\begin{pmatrix} 1&0&0&0&0 \\ 0&1&0&0&0 \\ 0&0&1&0&0 \\ 0&0&0&1&0 \\ 0&0&0&0&1 \end{pmatrix}$$
i.e. back to fully explicit.

Adding CO_2 frost to Mars thermal models.

We'll track frost mass with M_{CO_2}. Latent heat is L_{CO_2}.

Start Time Loop

if (M_{CO_2} EQ 0)

- Thermal model runs as before, calculate T_s.
- Did T_s fall below T_{frost}?

  No → do nothing

  Yes → Set T_s to T_{frost}

  Add M_{CO_2} of $(T_{frost} - T_s) \cdot \frac{\Delta z_1 \rho_1 c_1}{L_{CO_2}} \cdot \frac{1}{4}$

ENDIF

ELSE IF ($M_{CO_2} > 0$)

- Update temperatures with $T_s = T_{frost}$

  i.e. $T_s = a + bT_1$ formerly

  so $b = 0$, $a = T_{frost}$ in this case

- Update frost mass : $\Delta M_{CO_2} = -\frac{\Delta t}{L_{CO_2}}\left[\frac{2k_1 \cancel{\Delta t}}{\Delta z_1}(T_1 - T_{frost}) + (1-A_{frost})S - \varepsilon_{frost}\sigma T_{frost}^4\right]$

$$\Delta M_{CO_2} = \gamma T_1^* + \theta$$

where: $\gamma = -\frac{1}{L_{CO_2}}\frac{2k_1\Delta t}{\Delta z_1}$ scalar

$$\theta = \frac{\Delta t}{L_{CO_2}}\left[\frac{2k_1 \cancel{\Delta t}}{\Delta z_1}T_{frost} - (1-A_{frost})S + \varepsilon_{frost}\sigma T_{frost}^4\right]$$

array for each solar flux

do outside loop - for semi implicit

$\theta = f\theta^{k+1} + (1-f)\theta^k$

since θ's can all be calculated ahead of time - efficient will be tiny but can do outside of loop anyway

- Did M_{CO_2} fall below zero?
  - No → do nothing
  - Yes → Set $M_{CO_2} = 0$

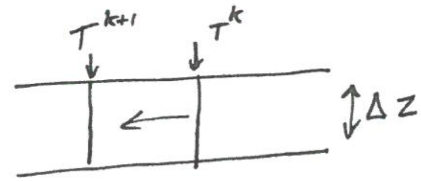
    Add to T_s : $\left(-M_{CO_2}\frac{4L_{CO_2}}{\rho_1 c_1 \Delta z_1}\right)$

END ELSE

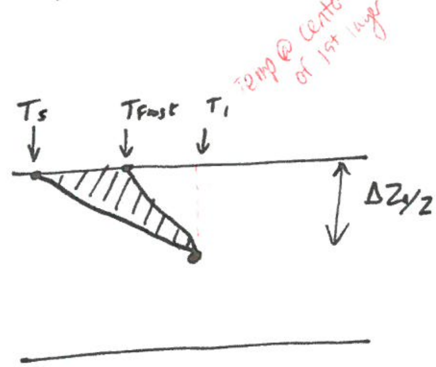
END TIME LOOP

WTF?! where did those factors of 4 & $\frac{1}{4}$ come from?

Think about area on thermal profiles representing thermal energy. e.g. →



when this block's temperature changes the thermal energy required is: $(T^{k+1} - T^{k}) \cdot \Delta z \cdot \rho \cdot c$



so, when frosting, T_s falls below T_{frost}. But we reset T_s to T_{frost} and book keep that energy as CO_2 frost. That energy is the shaded region times $\rho_1 c_1$

i.e. $$\rho_1 c_1 \left[\frac{\Delta z_1}{2} \cdot \frac{1}{2}(T_1 - T_s) - \frac{\Delta z}{2} \cdot \frac{1}{2}(T_1 - T_{frost}) \right]$$

$$\frac{1}{4} \rho_1 c_1 \Delta z_1 (T_{frost} - T_s)$$

so: $$\Delta M_{CO_2} = \frac{1}{L_{CO_2}} \frac{1}{4} \rho_1 c_1 \Delta z_1 (T_{frost} - T_s)$$

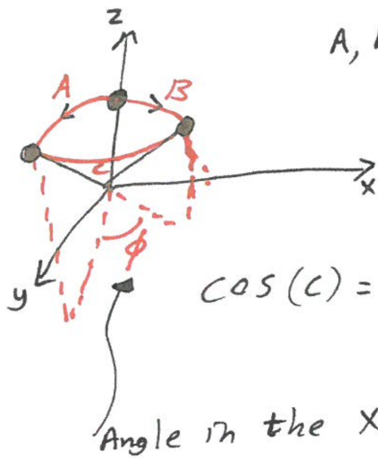
When defrosting, we do the reverse and convert the ΔMCO_2 (a negative number) to a ΔT so the the surface is no longer at T_{frost}.

$$T_s = T_{frost} - \frac{\Delta MCO_2 \cdot L_{CO_2}}{\rho_1 C_1 \Delta z_1} 4$$

Slopes - use the cosine rule for spherical triangles

Planetary surfaces are big slopes. i.e. flat at the equator (really the subsolar point) and vertical 90° away from that.

In general:



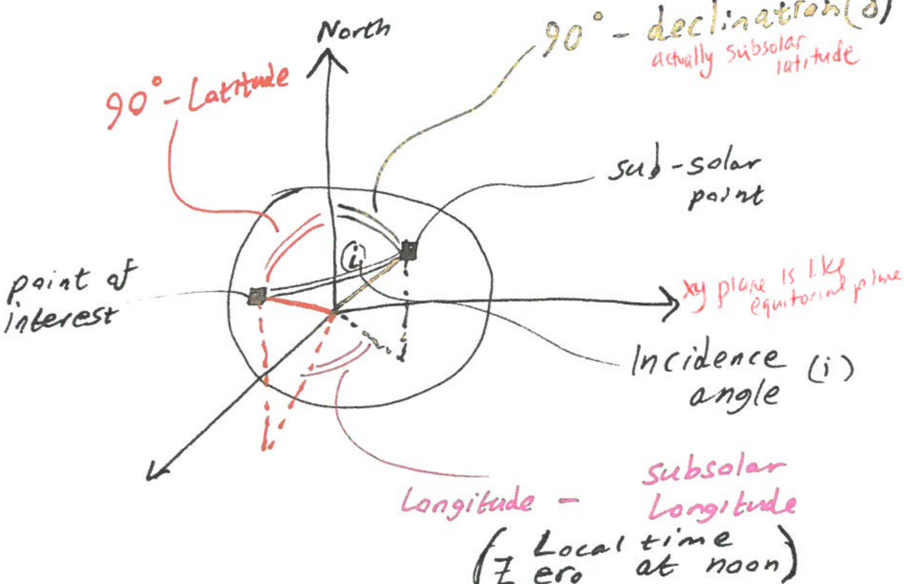
A, B, C : sides of a spherical triangle

ϕ: Inner angle between A & B

$$\cos(C) = \cos(A)\cos(B) + \sin(A)\sin(B)\cos(\phi)$$

For a planet

A = 90 - Latitude
B = 90 - solar declination
C = incidence angle
ϕ = Localtime (noon = 0)



$$\cos(i) = \cos(90-Lat)\cos(90-\delta) + \sin(90-lat)\sin(90-\delta)\cos(\text{local time})$$

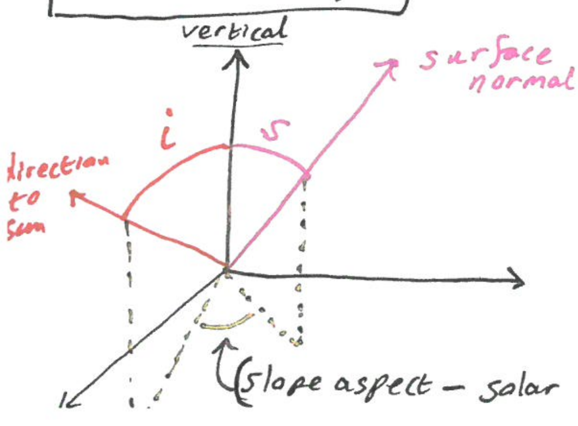
$$\cos(i) = \sin(lat)\sin(\delta) + \cos(lat)\cos(\delta)\cos(\text{local time})$$

Shadowed if $\cos(i) < 0$

This is the simplest horizon test possible

already done

For a slope



$$\cos(i_{slope}) = \cos(i).\cos(s) + \sin(i)\sin(s).\cos(\Delta\phi)$$

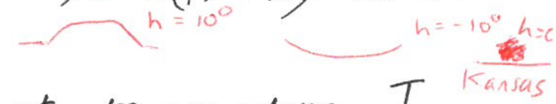
Slope is shadowed if $\cos(i_{slope}) < 0$

or $\cos(i) < 0$

Horizons

Useful for masking out sunlight and figuring out how much open sky is visible.

Let's store horizon height as $h(\phi)$ where ϕ is azimuth (north is zero). $h(\phi)$ may be <0 if the observer is on a hill.



For thermal emission of a surface at temperature T_s

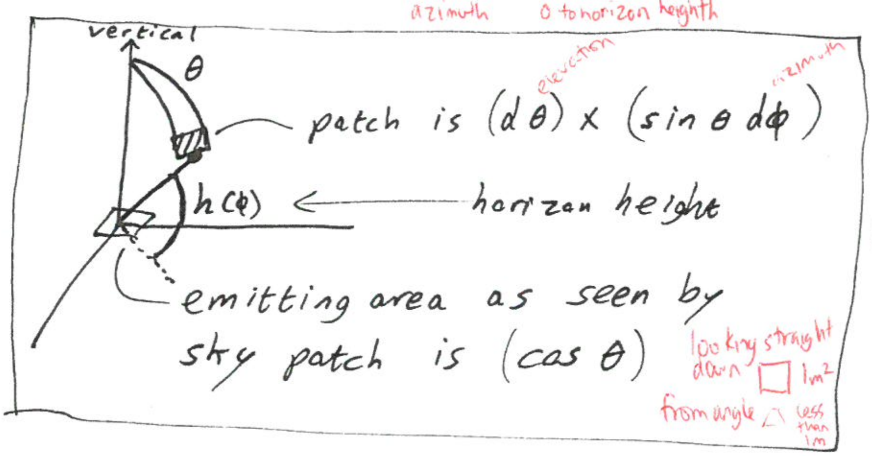
$$\text{Energy emitted per ster-radian} = \int_0^\infty \varepsilon B(\lambda, T_s)\, d\lambda = \varepsilon \frac{\sigma}{\pi} T_s^4$$

↑ Planck function.

To get energy emitted to space we need to integrate over the sky's solid angle.

$$\text{Flux} = \int_{\phi=0}^{2\pi} \int_{\theta=0}^{\pi/2-h} \varepsilon \frac{\sigma}{\pi} T_s^4 \cos(\theta) \sin\theta\, d\phi\, d\theta$$

$\phi=0$: azimuth; $\theta=0$: 0 to horizon heighth; $\cos(\theta)$: area reduction; $\sin\theta\, d\phi$: size of patch on sky in one direction; $d\theta$: size of patch on sky in other direction



$$\text{Flux} = \varepsilon \frac{\sigma}{\pi} T_s^4 \frac{1}{2} \int_{\phi=0}^{2\pi} \int_{\theta=0}^{\pi/2-h} \sin(2\theta)\, d\theta\, d\phi$$

⋮ integration then trig functions

$$\text{Flux} = \varepsilon \frac{\sigma}{\pi} T_s^4 \frac{1}{2} \int_{\phi=0}^{2\pi} \cos^2(h)\, d\phi$$

Special case: Flat ground; $h(\phi) = 0$ (flat horizon too)

$\cos^2(0) = 1$

$$\text{Flux} = \varepsilon \frac{\sigma}{\pi} T_s^4 \frac{1}{2} \int_{\phi=0}^{2\pi} d\phi$$

$$\text{Flux} = \varepsilon \sigma T_s^4$$

i.e. what we all know & love

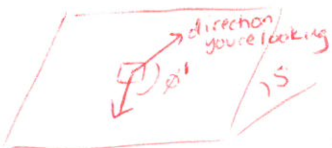
Special case II : Slope surrounded by infinite flat plane

Horizon as seen by a sloping surface, varies with azimuth

In the downhill direction it's zero; in the uphill direction it's s (the slope).

Think of a vector $\vec{d}$ in the horizontal plane that makes an angle ϕ with the downhill direction. $\phi = \pi$ directly uphill

distance traveled in uphill direction : $|\vec{d}| . \cos(\pi - \phi)$

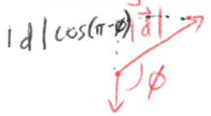


ϕ = angle projected onto the horizontal

height gained : $|\vec{d}| . \cos(\pi - \phi) . \tan(s)$

$$\tan(\text{Horizon height}) = \frac{\text{height gained}}{|\vec{d}|}$$

looking down normal (projected onto flat plane)



$$h(\phi) = \tan^{-1}\left(\tan(s) . \underbrace{\cos(\pi - \phi)}_{-\cos(\phi)}\right)$$

$\Delta z = |d| \cos(\pi - \phi) \tan(s)$ "elevation gain depending on how much uphill looking)

$\tan(h) = \frac{\Delta z}{|d|}$

$$\text{So } h(\phi) = \begin{cases} 0 & : \phi = (-\pi/2) \to (+\pi/2) \\ \tan^{-1}(-\tan(s)\cos(\phi)) & : \phi \ (\tfrac{\pi}{2}) \to (\tfrac{3\pi}{2}) \end{cases}$$

Split the previous integral :

$$\text{Flux} = \varepsilon \frac{\sigma}{\pi} T_s^4 \ \frac{1}{2} \left[\int_{-\pi/2}^{\pi/2} \cos^2(0)\, d\phi + \int_{\pi/2}^{3\pi/2} \underbrace{\cos^2(h)}_{1+\tan^2 h}\, d\phi \right]$$

$$\text{Flux} = \varepsilon \frac{\sigma}{\pi} T_s^4 \ \frac{1}{2} \left[\pi + \int_{\pi/2}^{3\pi/2} \frac{d\phi}{1 + \tan^2(h)} \right]$$

$$\text{Flux} = \varepsilon \frac{\sigma}{\pi} T_s^4 \ \frac{1}{2} \left[\pi + \underbrace{\int_{\pi/2}^{3\pi/2} \frac{d\phi}{1 + \tan^2 s \cos^2 \phi}} \right]$$

standard integral $= \pi \cos(s)$

$$\text{Flux} = \varepsilon \frac{\sigma}{\pi} T_s^4 \ \pi \ \frac{1}{2}(1 + \cos(s))$$

$$\text{Flux} = \varepsilon \sigma T_s^4 \ \underbrace{\cos^2(s/2)}$$

outgoing flux to space is modified by this Factor for a slope of s.

if $s = 0$, $\to 1$ and flux $= \varepsilon \sigma T_s^4$

if 90° slope, get $\frac{1}{2}$ $\to$ see $1/2$ the sky

Flux to Space: $\varepsilon_s \sigma T_s^4 \cos^2\left(\frac{s}{2}\right)$

Flux to surroundings: $\varepsilon_s \sigma T_s^4 \sin^2\left(\frac{s}{2}\right)$

Total flux is still $\varepsilon_s \sigma T_s^4$

Flux from surroundings: $\varepsilon_{Flat} \sigma \underbrace{T_{Flat}^4}$ $\sin^2\left(\frac{s}{2}\right)$ — if $s=0$, this is 0

→ you need to separately simulate T_{Flat}. Also $\varepsilon_s \neq \varepsilon_{Flat}$

$$\text{Net outgoing flux} = \underbrace{\varepsilon_s \sigma T_s^4 \cos^2\left(\tfrac{s}{2}\right)}_{\text{To space}} + \underbrace{\varepsilon_s \sigma T_s^4 \sin^2\left(\tfrac{s}{2}\right)}_{\text{To surroundings}} - \underbrace{\varepsilon_s}_{\text{fraction absorbed on slope.}} \underbrace{\varepsilon_{Flat} \sigma T_{Flat}^4 \sin^2\left(\tfrac{s}{2}\right)}_{\text{from surroundings}}$$

$$\text{Net outgoing flux} = \varepsilon_s \sigma T_s^4 \left[1 - \varepsilon_{Flat} \sin^2\left(\tfrac{s}{2}\right) \left(\frac{T_{Flat}}{T_s}\right)^4 \right]$$

Kirchoff's law two emissis could be different or same

Sometimes people assume $\varepsilon_{Flat} = 1$ & $T_{Flat} = T_s$ @ slope — bad assumption

so: net outgoing flux $= \varepsilon_s \sigma T_s^4 \cos^2\left(\frac{s}{2}\right)$

A bad assumption for high latitudes & pole-facing slopes.

need to do 2 thermal models here
- sloping terrain
- flat terrain (assume not affected by slope)

Atmospheric Radiation:

Can be approximated as 4% of the noontime flux. For IR wavelengths.

$$0.04\, S_{noon}\, \underbrace{\varepsilon_s}_{\text{fraction absorbed}}\, \underbrace{\cos^2\left(\tfrac{s}{2}\right)}_{\text{skyview factor}}$$

At Vis wavelengths it's approximated as 2% of flux. i.e. varies diurnally

$$0.02 \cdot S \cdot \underbrace{(1-A)}_{\text{fraction absorbed.}}\, \underbrace{\cos^2\left(\tfrac{s}{2}\right)}_{\text{skyview factor}}$$

$$F_{incoming} \leftarrow \begin{cases} F_{solar\ \text{direct}} = (1-A_s) \cdot S \cdot \cos(i_s) \\ F_{scat.\ VIS} = (1-A_s) \cdot S \cdot 0.02 \cdot \Sigma \\ F_{scat.\ IR} = \varepsilon_s \cdot S_{noon} \cdot 0.04 \cdot \Sigma \\ F_{Terrain\ IR\ \text{reflected}} = \varepsilon_s \cdot \left[\varepsilon_f \sigma T_f^4\right] (1-\Sigma) \\ F_{Terrain\ VIS\ \text{reflected}} = (1-A_s) \cdot \left[A_f\, S \cos(i_f)\right] \cdot (1-\Sigma) \end{cases}$$

assumes sun above horizon use the rest to figure out if actually above horizon

assume each m^2 isotropically radiating

Assumes Flat terrain for surroundings.

fine if surrounded by flat terrain if not would need to couple all the equations - hard, comp. expensive

"terrain view" only fraction makes it to slope

$$F_{outgoing} = -\varepsilon_s \sigma T_s^4$$

where: ☆ Σ is the sky view factor: $\frac{1}{2\pi}\int_0^{2\pi} \cos^2(h(\phi))\, d\phi$

i.e. $\Sigma = 1$ for flat terrain

$\Sigma = \cos^2(s/2)$ for a slope of s surrounded by flat terrain

i_s, i_f : incidence angles for the slope & flat terrain

A_s, A_f : Albedo for the slope & flat terrain

$\varepsilon_s, \varepsilon_f$: Emissivity for the slope & flat terrain

$$\boxed{T_s = a + b T_1 \quad \text{where} \quad a = \frac{\left(\frac{\Delta z_1}{2k_1}\right)\left[F_{incoming} + 3\varepsilon_s \sigma T_R^4\right]}{1 + \frac{4\varepsilon_s \sigma T_R^3 \Delta z_1}{2k_1}}}$$

Merge with previous model here