Mathematics for Skiers: Modeling Temperature Changes in Snowpack

Isaac Clark

April 22, 2025

Abstract

Snowpack dynamics play an important role in various environmental and human systems, influencing water resources, overall ecosystem health, and avalanche forecasting. Modeling the thermal behavior of snowpack layers is essential for understanding how energy from the atmosphere and solar radiation interacts with snow over time. This paper presents a mathematical model that simulates the temperature distribution within a layered snowpack over a 24-hour period. The model incorporates Fourier's law of heat conduction to model energy transfer through the layers of snow and accounts for heat exchange at the snowpack surface through both solar radiation and convective air temperature effects. By discretizing the continuous system into a finite number of layers, the model tracks how thermal energy is transferred vertically through the snow. To incorporate the changes that occure as part of the day-night cycle, sinusoidal functions are used to simulate the changing intensity of solar radiation and ambient air temperature. The resulting system of partial differential equations is solved numerically, providing insights into how upper snow layers warm and influence lower ones. This framework, while built on assumptions which simplify the model, offers a foundation for understanding the much more complex snow-climate interactions and could inform future models used in environmental science.

Introduction

The Earth's climate system is inherently complex and often exhibits chaotic behavior due to the interactions of countless dynamic variables. Among the many components of this system, snowpack is no exception and is affected by many seen and unseen variables. Understanding how snow responds to environmental factors such as sunlight and air temperature is crucial for predicting melt patterns, managing water resources, snowpack composition, and assessing avalanche risks. Despite the complexity of snowpack processes, modeling these systems mathematical can provide valuable insights into the fundamental physics governing thermal energy movement through snow.

This paper presents a numerical model aimed at simulating the temperature dynamics within a layered snowpack over a typical 24-hour cycle. The model is grounded in Fourier's law of thermal conduction to model heat transfer between layers and incorporates external boundary conditions that simulate the effects of incoming solar radiation and air temperature fluctuations. To represent the continuous snowpack as a manageable system that can be solved numerically, the domain is discretized into a finite number of horizontal layers. Each layer exchanges thermal energy with its immediate neighbors, allowing the model to simulate vertical heat diffusion using the one dimentional heat equation.

To reflect natural day-night cycles, sinusoidal functions are used to approximate solar radiation intensity and ambient air temperature changes throughout a 24-hour period. These time-dependent inputs enable the model to simulate how warming of the surface snow layers propagates downward over the course of a day. While the model intentionally adopts a number of simplifying assumptions—such as treating the snow as homogeneous and isotropic, ignoring wind and humidity variations, and neglecting melting processes—it remains useful for illustrating the core mechanisms of thermal transport in snow.

The primary goal of this research is to build a simple yet informative simulation that illustrates how external energy inputs of the environment affect snowpack temperature variability. By solving the resulting system of partial differential equations numerically, we visualize the evolving temperature distribution across snow layers. These results not only demonstrate the physical principles in action but also lay the groundwork for more advanced models that incorporate additional complexities such as melting, percolation, and heterogeneous snow properties.

Methods

Before setting up the system of PDEs that make up the core of the model, we first need to define some preliminary functions to simulate the day-night cycle and changing intensity of solar radiation and air temperature over a 24-hour period. To simulate the solar radiation intensity, we use a sinusoidal function defined as:

$$R(t) = R_{\max}\left(\sin^{2.5}\left(\pi\frac{t-t_{\text{rise}}}{t_{\text{set}}-t_{\text{rise}}}\right)\right) \tag{1}$$

Here, $R(t)$ is the solar radiation at time t, $R_{\max}$ is the maximum solar radiation, and t is time in seconds. Sunrise is at $t_{\text{rise}} = 27000$ (7:30 AM), and sunset is at $t_{\text{set}} = 63000$ (5:30 PM). To simulate air temperature over a 24-hour period, we use a similar sinusoidal function:

$$T_{\text{air}}(t) = T_{\text{air}}(0) + \left(10\sin^{2.5}\left(\pi\frac{t-t_{\text{rise}}}{t_{\text{set}}-t_{\text{rise}}}\right)\right) \tag{2}$$

Here, $T_{\text{air}}(t)$ is the air temperature at time t, and $T_{\text{air}}(0)$ is the minimum air temperature at $t = 0$. The amplitude of 10 ensures a maximum temperature 10°C above the minimum. Turning our attention to the system of PDEs, the model is based on the one dimentional heat equation [2] commonly defined as:

$$\frac{\partial T}{\partial t} = \frac{k}{C\rho}\frac{\partial^2 T}{\partial z^2} \tag{3}$$

where $T = T(z,t)$, k is the thermal conductivity, C is specific heat capacity, and ρ is density. Because thermal conductivity k depends on the snow density ρ, we define k with the equation [6]:

$$k = 0.023 + (((7.75 \times 10^{-5}\rho) + (1.105 \times 10^{-6}\rho^2))(2.22 - 0.023)) \tag{4}$$

To set up the system of PDEs, we must first define the boundary conditions for the system, we choose the upper boundary:

$$-k\frac{\partial T}{\partial z}\bigg|_{z=0} = (1-\alpha)R(t) + h(T_{\text{air}}(t) - T(0,t)) \tag{5}$$

where α is the albedo and h is the heat transfer coefficient [1], and we choose the lower boundary:

$$\frac{\partial T}{\partial z}\bigg|_{z=N-1} = 0 \tag{6}$$

With definite boundary conditions for the model, we apply them to the heat equation 3 and derive the following complete system of PDEs for a finite number of layers:

$$\frac{\partial T}{\partial t} = \frac{1}{\rho C}\frac{(1-\alpha)R(t) + h(T_{\text{air}}(t) - T(0,t))}{\Delta z} + \frac{k}{\rho C}\frac{(T(1,t) - T(0,t))}{\Delta z^2} \tag{7}$$

$$\frac{\partial T(i,t)}{\partial t} = \frac{k}{\rho C}\frac{T(i+1,t) - 2T(i,t) + T(i-1,t)}{\Delta z^2}, \quad \text{where} \quad i = \{1,2,\ldots,N-2\} \tag{8}$$

$$\frac{\partial T}{\partial t} = \frac{k}{\rho C}\frac{T(N-2,t) - T(N-1,t)}{\Delta z^2} \tag{9}$$

where H is the total depth of the snow, N is the number of layers, and $\Delta z = H/(N-1)$.

Results

Running the system in Python with defined variables, we plot the temperature of each layer over time. The resulting figure is displayed below:

Values used for the simulation:

- $T_{\text{air}} = 263\text{K}$ (-10.15C)
- $R_{\max} = 1000\text{W/m}^2$ [4]
- $H = 25\text{cm}$

- $\alpha = 0.9$ [3]
- $h = 10\text{W/m}^2\text{K}$ [5]
- $\rho = 300\text{kg/m}$ [3]
- $C = 2090\text{J/kgK}$ [3]
- $N = 25$

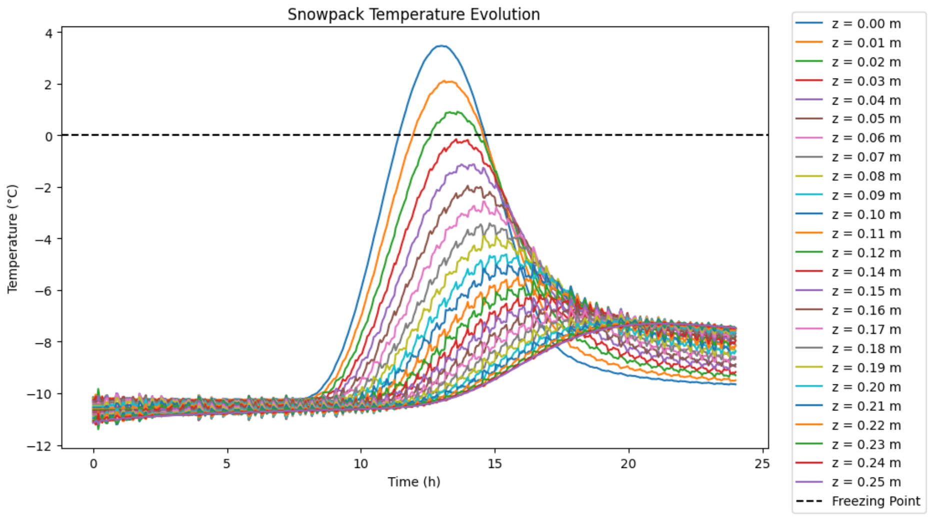

Figure 1: Simulated Temperatures

Conclusions

This study presents a simplified yet insightful model for simulating temperature dynamics within snowpack, providing a deeper and more intuitive understanding of the interactions between the warmer upper layer and cooler under layer of snow. While employing simplifying assumptions—such as neglecting melting, assuming the snow is homogeneous and isotropic, and ignoring the effects of wind or humidity—we are still left with a model that can serve as the basis for more advanced and complex systems.

The simulation reveals how the surface warming during the day gradually penetrates into the deeper layers, while the deeper layer simultaineously absorb this heat energy and cool the upper layers. The absence of solar energy input at night causes the whole system to cool, with the upper layers noteably cooling faster than the lower layers, which could be an interesting area for further analysis. These results provide a plausible explanation for the interactions of energy exchange at the snow-atmosphere level.

Future improvements could involve incorporating variable snow properties, water content, and multi-phase processes such as melting, refreezing, and percolation. Additionally, including this thermal model with existing weather data or integrating it into larger climate or avalanche forecasting systems could increase its practical relevance. Nevertheless, this study demonstrates how mathematics can explain complex natural patterns and highlights the value of modeling in environmental science and winter sports safety.

References

[1] Wikipedia contributors. *Convection (heat transfer)*. en. 2025. URL: https://en.wikipedia.org/wiki/Convection_(heat_transfer).

[2] Wikipedia contributors. *Heat equation*. en. 2025. URL: https://en.wikipedia.org/wiki/Heat_equation.

[3] Wikipedia contributors. *Snow*. en. 2025. URL: https://en.wikipedia.org/wiki/Snow.

[4] Wikipedia contributors. *Solar irradiance*. en. 2025. URL: https://en.wikipedia.org/wiki/Solar_irradiance.

[5] SolidWorks Corp. *Convection Heat Coefficient*. en. 2015. URL: https://help.solidworks.com/2015/english/SolidWorks/cworks/c_convection_heat_coefficient.htm?format=P&value=.

[6] V. R. Dutch et al. "Impact of measured and simulated tundra snowpack properties on heat transfer". In: *The Cryosphere* 16.10 (2022), pp. 4201–4222. DOI: 10.5194/tc-16-4201-2022. URL: https://tc.copernicus.org/articles/16/4201/2022/.

*ChatGPT was used as a tool to aid in the ideation and drafting of this paper. All content and ideas are the original creations of the author.

Appendix: Python Code

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import solve_ivp
import warnings

warnings.filterwarnings('ignore')

def solar_radiation(t):
    time_of_day = t % 86400
    if 27000 <= time_of_day <= 63000:  # Sun rises at 7:30 AM (27000 s
        ) and sets at 5:30 PM (63000 s)
        return 825 * (np.sin(np.pi * (time_of_day - 27000) / (63000 -
            27000)) ** 2.5)
    else:
        return 0

def air_temperature(t, T_air):
    time_of_day = t % 86400
    if 27000 <= time_of_day <= 63000:  # Sun rises at 7:30 AM (27000 s
        ) and sets at 5:30 PM (63000 s)
        return T_air + (10 * np.sin(np.pi * (time_of_day - 27000) /
            (63000 - 27000)) ** 2.5)
    else:
        return T_air

def snowpack(t, T, k, a, h, T_air, N):
    dTdt = np.zeros_like(T)
    R = solar_radiation(t)
    T_air = air_temperature(t, T_air)

    # Upper boundary (z=0)
    dTdt[0] = (1 / (rho * C)) * (((1 - a) * R + h * (T_air - T[0])) /
       (H / (N - 1)) + k * (T[1] - T[0]) / (H / (N - 1))**2)

    # Interior points (z=i)
    for i in range(1, N - 1):
        dTdt[i] = (k / (rho * C)) * (T[i+1] - 2*T[i] + T[i-1]) / (H /
           (N - 1))**2

    # Lower boundary (z=H)
    dTdt[N - 1] = (k / (rho * C)) * (T[N - 2] - T[N - 1]) / (H / (N -
       1))**2

    return dTdt

T_air = 263  # Night air temperature (K)
a = 0.9  # Albedo
H = .25  # Total snow depth (m)
h = 10  # Heat transfer coefficient (W/m^2 K)
rho = 300  # Snow density (kg/m^3)
R = 1000  # Solar radiation (W/m^2)
C = 2090  # Specific heat capacity (J/kg K)
N = 25  # Number of layers
k = 0.023 + (((7.75e-5 * rho) + (1.105e-6 * rho**2)) * (2.22 - 0.023))
      # Thermal conductivity (W/m K)

# Initial condition: linear or uniform
T0 = np.linspace(263, 262, N)

# Time domain (1 day)
t_span = (0, 86400) # 30 hours in seconds
t_eval = np.linspace(*t_span, 500)

# Solve the PDE
sol = solve_ivp(snowpack, t_span, T0, args=(k, a, h, T_air, N), t_eval
   =t_eval)

plt.figure(figsize=(10, 6))
for i in range(0, N, 1):
    plt.plot(sol.t / 3600, sol.y[i] - 273.15, label=f'z = {i*(H / (N -
       1)):.2f} m')
plt.axhline(y=0, color='k', linestyle='--', label='Freezing Point')
plt.xlabel('Time (h)')
plt.ylabel('Temperature (C)')
plt.title('Snowpack Temperature Evolution')
plt.legend(loc='upper left', bbox_to_anchor=(1.025, 1.05))
plt.show()
```