



AKADEMIA GÓRNICZO-HUTNICZA
IM. STANISŁAWA STASZICA W KRAKOWIE

Automatyzacja obliczeń cieplnych przegród budowlanych

Zastosowanie Języka Python w Nauce i Inżynierii

Autor:
Szymon Kuczek

Spis treści

| | | |
|----------|---|----------|
| 1 | Cel i zakres projektu | 2 |
| 2 | Technologie i Biblioteki | 2 |
| 3 | Model Matematyczny | 3 |
| 4 | Studium Przypadku - Analiza Scenariuszy | 4 |
| 4.1 | Scenariusz 1: Stara Kamienica (Ściana nieocieplona) | 4 |
| 4.2 | Scenariusz 2: Standard Deweloperski | 5 |
| 4.3 | Scenariusz 3: Budynek Pasywny | 6 |
| 5 | Implementacja - Fragmenty Kodu | 7 |
| 6 | Podsumowanie | 7 |

1 Cel i zakres projektu

Celem projektu było stworzenie narzędzia inżynierskiego w języku **Python**, umożliwiającego automatyczną analizę parametrów cieplnych przegród wielowarstwowych. Program realizuje obliczenia współczynnika przenikania ciepła (U) oraz generuje wizualizację rozkładu temperatur w przekroju ściany ("wykres temperatury").

Kluczowym założeniem była **automatyzacja procesu**: od wczytania danych z arkuszy kalkulacyjnych (Excel), przez symboliczne wyprowadzenie równań fizycznych, aż po generację raportów graficznych. Narzędzie pozwala na szybką weryfikację zgodności przegrody z Warunkami Technicznymi (WT 2021).

2 Technologie i Biblioteki

W projekcie wykorzystano nowoczesny stos technologiczny języka Python, łączący inżynierię danych z obliczeniami symbolicznymi:

- **SymPy (Symbolic Python)**: Zamiast "sztywnego" kodowania wzorów, biblioteka ta została użyta do dynamicznego tworzenia modelu matematycznego przegrody. Pozwala to na obliczanie ścian o dowolnej liczbie warstw (n -warstw), generując symboliczne równanie oporu cieplnego R_{tot} w czasie rzeczywistym.
- **OpenPyXL**: Biblioteka służąca do integracji z powszechnie używanym w inżynierii formatem '.xlsx'. Program automatycznie pobiera parametry materiałowe (λ) z zewnętrznej bazy danych oraz wczytuje geometrię przegrody z plików projektowych.
- **Matplotlib**: Wykorzystana do wizualizacji wyników. Generuje profesjonalne wykresy rozkładu temperatury w przegrodzie, uwzględniając rzeczywiste grubości warstw oraz gradienty temperatur.

3 Model Matematyczny

Program opiera swoje obliczenia na jednowymiarowym przepływie ciepła w stanie ustalonym.

Opór całkowity przegrody R_{tot} wyznaczany jest jako suma oporów przejmowania ciepła oraz oporów przewodzenia poszczególnych warstw:

$$R_{tot} = R_{si} + \sum_{i=1}^n \frac{d_i}{\lambda_i} + R_{se} \quad \left[\frac{m^2 K}{W} \right] \quad (1)$$

Gdzie:

- R_{si}, R_{se} – opory przejmowania ciepła (wew./zew.),
- d_i – grubość i -tej warstwy [m],
- λ_i – współczynnik przewodzenia ciepła i -tej warstwy [W/mK].

Współczynnik przenikania ciepła U obliczany jest z zależności:

$$U = \frac{1}{R_{tot}} \quad \left[\frac{W}{m^2 K} \right] \quad (2)$$

Dzięki zastosowaniu biblioteki SymPy, wzór ten nie jest wpisany na stałe, lecz generowany dynamicznie dla listy zmiennych symbolicznych $d_1 \dots d_n, \lambda_1 \dots \lambda_n$.

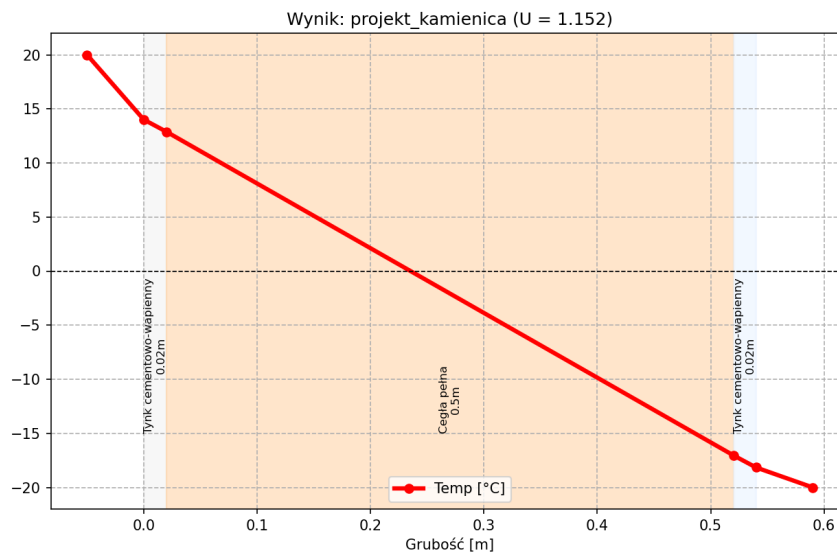
4 Studium Przypadku - Analiza Scenariuszy

W celu weryfikacji działania programu przeprowadzono analizę trzech typowych scenariuszy budowlanych. Warunki brzegowe przyjęto na poziomie: $T_{wew} = 20^{\circ}C$, $T_{zew} = -20^{\circ}C$.

4.1 Scenariusz 1: Stara Kamienica (Ściana nieocieplona)

Analizie poddano ścianę z cegły pełnej o grubości 50 cm.

- **Konstrukcja:** Tynk + Cegła pełna (50cm) + Tynk.
- **Wynik obliczeń:** $U \approx 1.25 \text{ W}/(\text{m}^2\text{K})$.
- **Diagnoza:** Przegroda nie spełnia współczesnych norm. Widoczny gwałtowny spadek temperatury w całej objętości muru, co grozi przemarzaniem.

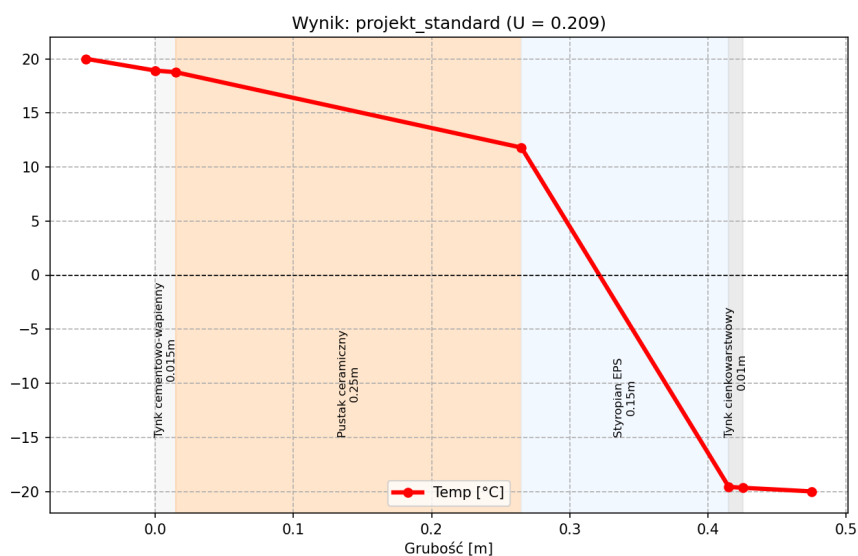


Rysunek 1: Rozkład temperatury w ścianie nieocieplonej.

4.2 Scenariusz 2: Standard Deweloperski

Ściana dwuwarstwowa typowa dla współczesnego budownictwa wielorodzinnego.

- **Konstrukcja:** Pustak ceramiczny (25cm) + Styropian EPS (15cm).
- **Wynik obliczeń:** $U \approx 0.21 \text{ W}/(\text{m}^2\text{K})$.
- **Diagnoza:** Wynik na granicy normy WT 2021 ($U_{max} = 0.20$). Wykres pokazuje, że główny spadek temperatury (izolacja) odbywa się w warstwie styropianu, chroniąc warstwę nośną przed ujemnymi temperaturami.

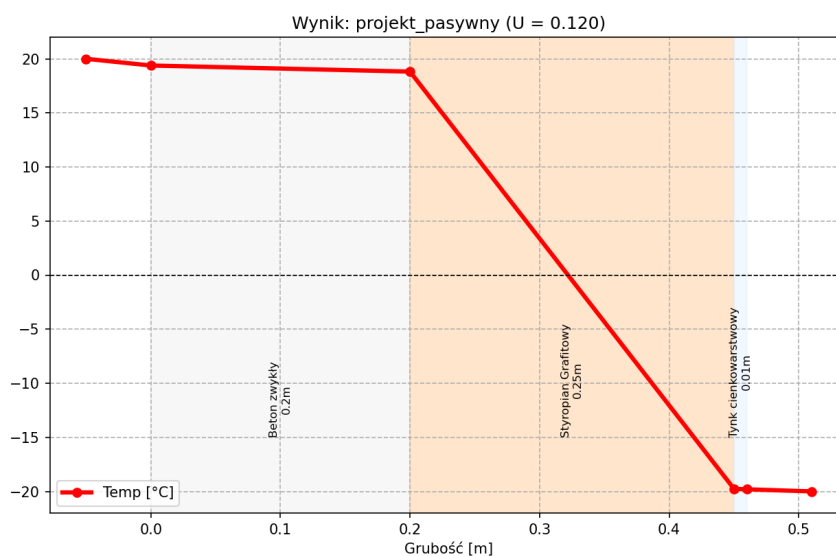


Rysunek 2: Rozkład temperatury - Standard Deweloperski.

4.3 Scenariusz 3: Budynek Pasywny

Przegroda o podwyższonych parametrach izolacyjnych.

- **Konstrukcja:** Żelbet (20cm) + Styropian Graftowy (25cm, $\lambda = 0.031$).
- **Wynik obliczeń:** $U \approx 0.12 \text{ W}/(\text{m}^2\text{K})$.
- **Diagnoza:** Przegroda wysoce energooszczędna. Warstwa konstrukcyjna znajduje się całkowicie w strefie dodatnich temperatur (ok. 19°C), co eliminuje naprężenia termiczne konstrukcji.



Rysunek 3: Rozkład temperatury - Standard Pasywny.

5 Implementacja - Fragmenty Kodu

Poniżej przedstawiono kluczowy fragment kodu odpowiadający za symboliczne wyprowadzenie równania oporu cieplnego przy użyciu biblioteki SymPy.

```
1 def oblicz_przegrode_symbolicznie(warstwy_dane):
2     # 1. Definicja symboli
3     R_tot_sym = sp.symbols('R_tot')
4     R_si, R_se = sp.symbols('R_si R_se')
5
6     # Symbole dynamiczne dla n-warstw
7     n = len(warstwy_dane)
8     d_syms = sp.symbols(f'd_1:{n+1}')
9     lam_syms = sp.symbols(f'lambda_1:{n+1}')
10
11    # 2. Wzór na opór całkowity (generowany automatycznie)
12    suma_oporow = sum(d/l for d, l in zip(d_syms, lam_syms))
13    rownanie_R = sp.Eq(R_tot_sym, R_si + suma_oporow + R_se)
14
15    # 3. Podstawienie wartości liczbowych (Subs)
16    wartosci = {R_si: R_SI, R_se: R_SE}
17    for i, (nazwa, d_val, lam_val) in enumerate(warstwy_dane):
18        wartosci[d_syms[i]] = d_val
19        wartosci[lam_syms[i]] = lam_val
20
21    R_wynik = rownanie_R.rhs.subs(wartosci)
22    return float(1/R_wynik) # Zwroc U-value
```

Listing 1: Symboliczne obliczanie oporu przegrody

6 Podsumowanie

Stworzone oprogramowanie stanowi w pełni funkcjonalny prototyp narzędzia klasy *Engineering Automation*. Główne zalety rozwiązania to:

1. **Skalowalność:** Możliwość analizy setek plików projektowych w kilka sekund (tryb wsadowy).
2. **Elastyczność:** Oparcie o zewnętrzną bazę Excel pozwala na łatwą modyfikację parametrów materiałowych bez ingerencji w kod źródłowy.
3. **Precyzja:** Zastosowanie obliczeń symbolicznych eliminuje błędy zaokrągleń na etapach pośrednich.

Projekt ten łączy wiedzę domenową (Fizyka Budowli) z umiejętnościami programistycznymi, co wpisuje się w nowoczesny trend cyfryzacji w budownictwie (BIM/Automated Design).