4.25 Usikkerhedsbudget for beregnet størrelse: Varmeflow

Fouriers varmelov (1D) for varmetransporten gennem et materiale med arealet A og tykkelsen Δx :

$$Q = \frac{k \cdot A \cdot (T_2 - T_1)}{\Delta x}$$

T_1 og T_2 er temperaturen på hhv. kold og varm side

Der måles følgende:

- $A = 100{,}0\ cm^2 \pm 2{,}0\ cm^2$
- $k = 70\ {}^{W}/_{m \cdot K}$
- $T_2 = 150{,}00\ °C\ \pm 0{,}50\ °C$
- $T_1 = 80{,}00\ °C\ \pm 0{,}40\ °C$
- $\Delta x = 5{,}00\ cm\ \pm 0{,}10\ cm$

Den termiske ledningsevne, k, er et tabelopslag, uden angivelse af usikkerhed

Hint: det betyder ikke at der ikke er en usikkerhed forbundet med tallet).

A. Argumenter for fordelingstype for hvert enkelt bidrag

Hint: Der er ikke nogen rigtige eller forkerte svar, men antagelse og argument skal passe sammen – forsøg at benytte alle fordelingstyper (hvis du digter lidt)?

A kan anses som at være 2 terninger dat det er l * h så det kan argumenteres for at man får en trækant fordeling.

k er en opslags værdig så dens fordeling er højst sandsynlig en normal fordeling

De to temperature er nok firkant fordelinger da jeg kunne forstille mig at chancen er lige stor for de forskellige målinger

Δx kan måske forstilles at være en u fordeling vis man bruger en tommelstock til at måle f.eks. hvor man siger "close enough"

B. Beregn standardusikkerheden under ovenstående antagelser?

Bemærk: Bemærk at resultatet afhænger af dine valg i A, så forskellige resultater kan være rigtige

```
u = symunit;
syms us a

firkant_fordeling = us == a/sqrt(3) ;  %opskrive forskellige formler for de forskellige fordelinger
u_fordeling = us == a/sqrt(2) ;
trekant_fordeling = us == a/sqrt(6) ;
format longG
A_alpha = 2/100/100
```

```
A_alpha =
                    0.0002
```

```
k_alpha = 0.5 ; %Antager ussikerhed
T2_alpha = 0.5;
T1_alpha = 0.4;
delta_X_alpha = 0.1/100
```

```
delta_X_alpha =
                     0.001
```

```
u_A = vpa(solve(subs(trekant_fordeling,a,A_alpha), us), 4) %substituere værdiger ind i fordelings udregningerne
```

u_A = 8.165e-5

```
u_T2 = vpa(solve(subs(firkant_fordeling,a,T2_alpha), us), 4)
```

u_T2 = 0.2887

```
u_T1 = vpa(solve(subs(firkant_fordeling,a,T1_alpha), us), 4)
```

u_T1 = 0.2309

```
u_delta_x = vpa(solve(subs(u_fordeling,a,delta_X_alpha), us), 4)
```

u_delta_x = 0.0007071

```
u_k = vpa(solve(subs(trekant_fordeling,a,k_alpha), us), 4) %nok en normal fordeling men vi approksimere med en trekants fordeling
```

u_k = 0.2041

```
%% der laves ikke over k da ussikerheden er antaget.
```

C. Lav usikkerhedsberegningen så du bestemmer u(Q). ~~i excel. Sørg for at lave en skalerbar opstilling (hvor antallet af usikkerhedskomponenter nemt kan øges/reduceres) - tag evt. et kig på video 00009~~

```
syms A T2 T1 delta_x k
Q = (k * A * (T2 - T1)) / delta_x;
vars = symvar(Q)
```

vars = $\left(A \quad T_1 \quad T_2 \quad \delta_x \quad k\right)$

```
var_vals =[100/100/100 80 150 5/100 70];
var_ussikerheder = [u_A u_T1 u_T2 u_delta_x u_k];
STAT.Ophobningsloven(Q, vars, var_vals, var_ussikerheder) %% Svaret skulle gerne være i watt
```

ans = 17.060480649741019575455093762421

Kode til ophobnings funktionen

```
function U_tot = Ophobningsloven(EQ, vars, varValues, varUssikerheder)
%Udregner ophobningsloven for en given equation
% Input:
% EQ = Equation som skal laves ophobning om
% vars = array med var navne (symvar(EQ))
% varValues = Values af variablerne i samme rællefælge som vars
% varUssikerheder = ussikerhederne i samme rækkefølge som vars og
% varValues.
% Output:
% U_tot = den samlede ophobning (der er taget kvadarat af den)
ds = jacobian(EQ);
ds_num = subs(ds, vars, varValues);
for j = 1:length(ds)
    final(j) = sqrt((ds_num(j) * varUssikerheder(j))^2);
end
U_tot = sum(final);
end
```