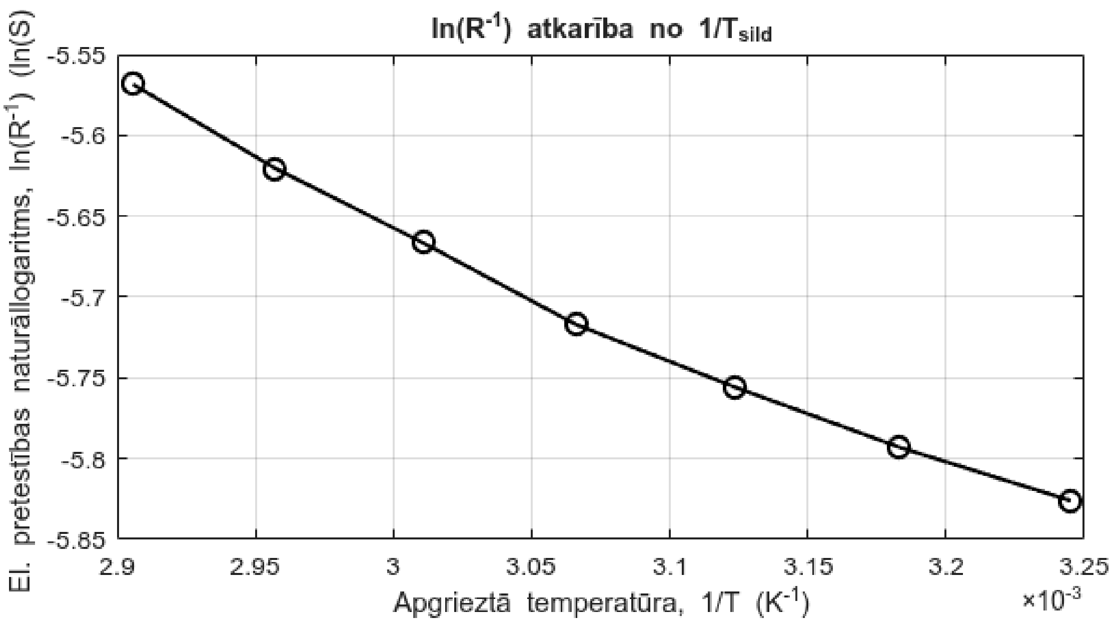
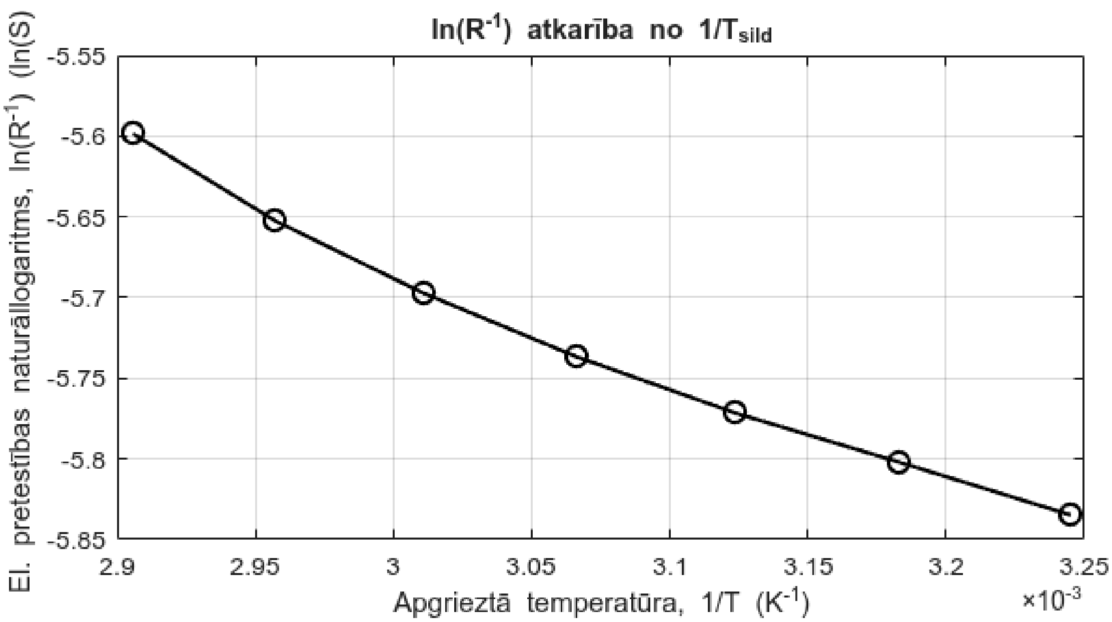
Table of Contents

2. uzdevums

```
t = [35 41 47 53 59 65 71]; % Celsija grādi
T = t(:) + 273.15;          % Kelvini
delta_t = 0.5;

R_sild = [339 328 316 304 289 276 262];
R_dzes = [342 331 321 310 298 285 270];
delta_R = 0.01;

X = 1 ./ T;
Y_sild = log(1 ./ R_sild);
Y_dzes = log(1 ./ R_dzes);

figure;
plot(X, Y_sild, 'ko-', ...
    'LineWidth', 1.5, ...
    'MarkerSize', 8);

title('ln(R^{-1}) atkarība no 1/T_{sild}');
xlabel('Apgrieztā temperatūra, 1/T (K^{-1})');
ylabel('El. pretestības naturāllogaritms, ln(R^{-1}) (ln(S))');
grid on;

figure;

plot(X, Y_dzes, 'ko-', ...
    'LineWidth', 1.5, ...
    'MarkerSize', 8);

title('ln(R^{-1}) atkarība no 1/T_{sild}');
xlabel('Apgrieztā temperatūra, 1/T (K^{-1})');
ylabel('El. pretestības naturāllogaritms, ln(R^{-1}) (ln(S))');
grid on;

lnGamma_p_dzes = [0.003 -5.7138];
lnGamma_p_sild = [0.0031 -5.7311];
```





3. uzdevums

Vidējā aktivācijas enerģija dzesēšanas posmam

```
BoltsmanConst = 1.38*10^(-23);
activationEnergies = zeros(6, 1);
for i = 1:1:length(R_dzes)-1
    activationEnergies(i) = 2 * BoltsmanConst * ...
        ( log(1/R_dzes(i+1)) - log(1/R_dzes(i)) ) / ...
        ( 1/T(i) - 1/T(i+1) );
end
disp('Aktivācijas enerģijas dzesēšanas posmam =')
disp(activationEnergies)
```

```
avgActivationEnergy = mean(activationEnergies);
disp('Vidējā aktivācijas enerģija dzesēšanas posmam =');
disp(avgActivationEnergy);

clear activationEnergies
```

```
Aktivācijas enerģijas dzesēšanas posmam =
   1.0e-19 *

    0.1456
    0.1419
    0.1675
    0.1967
    0.2305
    0.2894

Vidējā aktivācijas enerģija dzesēšanas posmam =
   1.9527e-20
```

$\overline{\Delta E}_{\searrow} = 1,9527 \cdot 10^{-20}$ J/K

4. uzdevums, kļūdu aprēķini

Kvadrātsakne no kvadrātu summas - hipotenūzas funkcijas paplašinājums

```
function res = hyp(vect)
    res = 0;
    for i = 1:1:length(vect)
        res = res + vect(i)^2;
    end
    res = sqrt(res);
end
```

```
Kopējā kļūda temperatūras mērījumiem = 12.0069
Relatīvā kļūda temperatūras mērījumiem = 22.6546%
```

Temperatūras kļūdas aprēķins (Tiešo mērījumu kļūda)

```
avg_t = mean(t);
fprintf("Vidējā temperatūra (Celsija grādi) = %.4f\n", avg_t);

S_t = std(t)/sqrt(length(t));
fprintf("Vidējā kvadrātiskā kļūda = %.4f\n", S_t);

Delta_t_s = delta_t/3 * 1.96;
fprintf("Sistemātiskā kļūda temperatūras mērījumiem = %.4f\n", Delta_t_s);

Delta_t_g = S_t * 2.45;
fprintf("Gadījuma kļūda temperatūras mērījumiem = %.4f\n", Delta_t_g);

Delta_t = hyp([Delta_t_s, Delta_t_g]);
fprintf("Kopējā kļūda temperatūras mērījumiem = %.4f\n", Delta_t);
```

```
epsilon_t = Delta_t / avg_t;
fprintf("Relatīvā kļūda temperatūras mērījumiem = %.4f%%\n\n", epsilon_t *
100);

clear S_t Delta_t_s Delta_t_g
```

Vidējā temperatūra (Celsija grādi) = 53.0000
Vidējā kvadrātiskā kļūda = 4.8990
Sistemātiskā kļūda temperatūras mērījumiem = 0.3267
Gadījuma kļūda temperatūras mērījumiem = 12.0025

$t = (53 \pm 12,0069)$°C pie $\beta = 0,95$ un $\varepsilon_t = 22,6546\%$

Pretestības kļūdas aprēķins (Tiešo mērījumu kļūda)

```
avg_R_sild = mean(R_sild);
avg_R_dzes = mean(R_dzes);
fprintf("Vidējā pretestība sildīšanas posmam = %.4f\n", avg_R_sild);
fprintf("Vidējā pretestība dzesēšanas posmam = %.4f\n", avg_R_dzes);

S_R_sild = std(R_sild)/sqrt(length(R_sild));
S_R_dzes = std(R_dzes) / sqrt(length(R_dzes));
fprintf("Vidējā kvadrātiskā kļūda sildīšanas posmam = %.4f\n", S_R_sild);
fprintf("Vidējā kvadrātiskā kļūda dzesēšanas posmam = %.4f\n", S_R_dzes);

Delta_R_sild_g = S_R_sild * 2.45;
Delta_R_dzes_g = S_R_dzes * 2.45;
fprintf("Gadījuma kļūda sildīšanas posmam = %.4f\n", Delta_R_sild_g);
fprintf("Gadījuma kļūda dzesēšanas posmam = %.4f\n", Delta_R_dzes_g);

Delta_R_s = delta_R / 3 * 1.96;
fprintf("Sistemātiskā kļūda pretestības mērījumiem = %.4f\n", Delta_R_s)

Delta_R_sild = hyp([Delta_R_s, Delta_R_sild_g]);
Delta_R_dzes = hyp([Delta_R_s, Delta_R_dzes_g]);
fprintf("Kopējā kļūda sildīšanas posmam = %.4f\n", Delta_R_sild);
fprintf("Kopējā kļūda dzesēšanas posmam = %.4f\n", Delta_R_dzes);

fprintf("Relatīvā kļūda sildīšanas posmam = %.4f%%\n", (Delta_R_sild /
avg_R_sild) * 100);
fprintf("Relatīvā kļūda dzesēšanas posmam = %.4f%%\n", (Delta_R_dzes /
avg_R_dzes) * 100);

clear S_R_sild S_R_dzes Delta_R_sild_g Delta_R_dzes_g
```

Vidējā pretestība sildīšanas posmam = 302.0000
Vidējā pretestība dzesēšanas posmam = 308.1429
Vidējā kvadrātiskā kļūda sildīšanas posmam = 10.5672
Vidējā kvadrātiskā kļūda dzesēšanas posmam = 9.6743
Gadījuma kļūda sildīšanas posmam = 25.8898
Gadījuma kļūda dzesēšanas posmam = 23.7020
Sistemātiskā kļūda pretestības mērījumiem = 0.0065
Kopējā kļūda sildīšanas posmam = 25.8898
Kopējā kļūda dzesēšanas posmam = 23.7020

```
Relatīvā kļūda sildīšanas posmam = 8.5728%
Relatīvā kļūda dzesēšanas posmam = 7.6919%
```

$R_{\text{sild.}} = (302 \pm 25,8898)_{\Omega}$ pie $\beta = 0,95$ un $\varepsilon_{R_{\text{sild.}}} = 8,5728\%$

$R_{\text{dzes.}} = (308,1429 \pm 23,7020)_{\Omega}$ pie $\beta = 0,95$ un $\varepsilon_{R_{\text{dzes.}}} = 7,6919\%$

Īpaši atrastā R_p kļūda (Tiešo mērījumu kļūda)

```
Delta_R_p = Delta_R_s;
fprintf("R_p sildīšanas posmam = %.4f Ω\n", 1/exp(lnGamma_p_sild(2)))
fprintf("R_p dzesēšanas posmam = %.4f Ω\n", 1/exp(lnGamma_p_dzes(2)))
fprintf("Kopējā R_p kļūda = %.4f\n", Delta_R_p);
fprintf("R_p Relatīvā kļūda sildīšanas posmam = %.4f%%\n", (Delta_R_p *
exp(lnGamma_p_sild(2))) * 100);
fprintf("R_p Relatīvā kļūda dzesēšanas posmam = %.4f%%\n", (Delta_R_p *
exp(lnGamma_p_dzes(2))) * 100);

clear Delta_R_s
```

```
R_p sildīšanas posmam = 308.3082 Ω
R_p dzesēšanas posmam = 303.0204 Ω
Kopējā R_p kļūda = 0.0065
R_p Relatīvā kļūda sildīšanas posmam = 0.0021%
R_p Relatīvā kļūda dzesēšanas posmam = 0.0022%
```

$R_{p\,\text{dzes.}} = (308,3082 \pm 0,0065)_{\Omega}$ pie $\beta = 0,95$ un $\varepsilon_{R_{p\,\text{dzes.}}} = 0,0021\%$

$R_{p\,\text{dzes.}} = (303,0204 \pm 0,0065)_{\Omega}$ pie $\beta = 0,95$ un $\varepsilon_{R_{p\,\text{dzes.}}} = 0,0022\%$

Aktivācijas enerģijas kļūdas aprēķins (Netiešo mērījumu kļūda)

```
syms R_1 R_2 T_1 T_2;

activationEnergyFormula = 2 * BoltsmanConst * ...
    (log(1/R_2) - log(1/R_1)) / (1/T_1 - 1/T_2);

activationEnergyDerivativeByR_1 = diff(activationEnergyFormula, 1, R_1);
activationEnergyDerivativeByR_2 = diff(activationEnergyFormula, 1, R_2);
activationEnergyDerivativeByT_1 = diff(activationEnergyFormula, 1, T_1);
activationEnergyDerivativeByT_2 = diff(activationEnergyFormula, 1, T_2);

activationEnergySubFrom = [R_1, R_2, T_1, T_2];
activationEnergy1SubTo = [R_sild(1), 1/exp(lnGamma_p_sild(2)), T(1), 1/
lnGamma_p_sild(1)];
activationEnergy2SubTo = [1/exp(lnGamma_p_sild(2)), R_sild(7), 1/
lnGamma_p_sild(1), T(7)];
activationEnergy3SubTo = [R_dzes(1), 1/exp(lnGamma_p_dzes(2)), T(1), 1/
lnGamma_p_dzes(1)];
activationEnergy4SubTo = [1/exp(lnGamma_p_dzes(2)), R_dzes(7), 1/
lnGamma_p_dzes(1), T(7)];

activationEnergy1 = double(subs(activationEnergyFormula, ...
    activationEnergySubFrom, activationEnergy1SubTo));
```

```
Delta_activationEnergy1 = double(hyp([ ...
    subs(activationEnergyDerivativeByR_1, activationEnergySubFrom,
activationEnergy1SubTo) * Delta_R_sild, ...
    subs(activationEnergyDerivativeByR_2, activationEnergySubFrom,
activationEnergy1SubTo) * Delta_R_p, ...
    subs(activationEnergyDerivativeByT_1, activationEnergySubFrom,
activationEnergy1SubTo) * Delta_t, ...
    subs(activationEnergyDerivativeByT_2, activationEnergySubFrom,
activationEnergy1SubTo) * Delta_t ...
]));

fprintf("Aktivācijas enerģija sildīšanas posma sākumam = %E\n",
activationEnergy1);
fprintf("Aktivācijas enerģijas kopējā kļūda sildīšanas posma sākumam = %E\n",
Delta_activationEnergy1);
fprintf("Aktivācijas enerģijas relatīvā kļūda sildīšanas posma sākumam = %.4f%
%\n", abs(Delta_activationEnergy1 / activationEnergy1) * 100);

activationEnergy2 = double(subs(activationEnergyFormula, ...
    activationEnergySubFrom, activationEnergy2SubTo));

Delta_activationEnergy2 = double(hyp([ ...
    subs(activationEnergyDerivativeByR_1, activationEnergySubFrom,
activationEnergy2SubTo) * Delta_R_p, ...
    subs(activationEnergyDerivativeByR_2, activationEnergySubFrom,
activationEnergy2SubTo) * Delta_R_sild, ...
    subs(activationEnergyDerivativeByT_1, activationEnergySubFrom,
activationEnergy2SubTo) * Delta_t, ...
    subs(activationEnergyDerivativeByT_2, activationEnergySubFrom,
activationEnergy2SubTo) * Delta_t ...
]));

fprintf("Aktivācijas enerģija sildīšanas posma beigām = %E\n",
activationEnergy2);
fprintf("Aktivācijas enerģijas kopējā kļūda sildīšanas posma beigām = %E\n",
Delta_activationEnergy2);
fprintf("Aktivācijas enerģijas relatīvā kļūda sildīšanas posma beigām = %.4f%%
\n", abs(Delta_activationEnergy2 / activationEnergy2) * 100);

activationEnergy3 = double(subs(activationEnergyFormula, ...
    activationEnergySubFrom, activationEnergy3SubTo));

Delta_activationEnergy3 = double(hyp([ ...
    subs(activationEnergyDerivativeByR_1, activationEnergySubFrom,
activationEnergy3SubTo) * Delta_R_dzes, ...
    subs(activationEnergyDerivativeByR_2, activationEnergySubFrom,
activationEnergy3SubTo) * Delta_R_p, ...
    subs(activationEnergyDerivativeByT_1, activationEnergySubFrom,
activationEnergy3SubTo) * Delta_t, ...
    subs(activationEnergyDerivativeByT_2, activationEnergySubFrom,
activationEnergy3SubTo) * Delta_t ...
]));
```

```
fprintf("Aktivācijas enerģija dzesēšanas posma sākumam = %E\n",
activationEnergy3);
fprintf("Aktivācijas enerģijas kopējā kļūda dzesēšanas posma sākumam = %E\n",
Delta_activationEnergy3);
fprintf("Aktivācijas enerģijas relatīvā kļūda dzesēšanas posma sākumam = %.4f%
%\n", abs(Delta_activationEnergy3 / activationEnergy3) * 100);

activationEnergy4 = double(subs(activationEnergyFormula, ...
    activationEnergySubFrom, activationEnergy4SubTo));

Delta_activationEnergy4 = double(hyp([ ...
    subs(activationEnergyDerivativeByR_1, activationEnergySubFrom,
activationEnergy4SubTo) * Delta_R_p, ...
    subs(activationEnergyDerivativeByR_2, activationEnergySubFrom,
activationEnergy4SubTo) * Delta_R_dzes, ...
    subs(activationEnergyDerivativeByT_1, activationEnergySubFrom,
activationEnergy4SubTo) * Delta_t, ...
    subs(activationEnergyDerivativeByT_2, activationEnergySubFrom,
activationEnergy4SubTo) * Delta_t ...
]));

fprintf("Aktivācijas enerģija dzesēšanas posma beigām = %E\n",
activationEnergy4);
fprintf("Aktivācijas enerģijas kopējā kļūda dzesēšanas posma beigām = %E\n",
Delta_activationEnergy4);
fprintf("Aktivācijas enerģijas relatīvā kļūda dzesēšanas posma beigām = %.4f%%
\n", abs(Delta_activationEnergy4 / activationEnergy4) * 100);

clear activationEnergyDerivativeByR_1 activationEnergyDerivativeByR_2
activationEnergyDerivativeByT_1 activationEnergyDerivativeByT_2 ...
    activationEnergySubFrom activationEnergy1SubTo activationEnergy2SubTo
activationEnergy3SubTo activationEnergy4SubTo
```

```
Aktivācijas enerģija sildīšanas posma sākumam = 1.804224E-20
Aktivācijas enerģijas kopējā kļūda sildīšanas posma sākumam = 2.575700E-20
Aktivācijas enerģijas relatīvā kļūda sildīšanas posma sākumam = 142.7594%
Aktivācijas enerģija sildīšanas posma beigām = 2.312031E-20
Aktivācijas enerģijas kopējā kļūda sildīšanas posma beigām = 2.304596E-20
Aktivācijas enerģijas relatīvā kļūda sildīšanas posma beigām = 99.6784%
Aktivācijas enerģija dzesēšanas posma sākumam = 1.362262E-20
Aktivācijas enerģijas kopējā kļūda dzesēšanas posma sākumam = 1.209472E-20
Aktivācijas enerģijas relatīvā kļūda dzesēšanas posma sākumam = 88.7841%
Aktivācijas enerģija dzesēšanas posma beigām = 3.377266E-20
Aktivācijas enerģijas kopējā kļūda dzesēšanas posma beigām = 5.896499E-20
Aktivācijas enerģijas relatīvā kļūda dzesēšanas posma beigām = 174.5938%
```

$\Delta E_1 = (1,8042 \pm 2,5757) \cdot 10^{-20}$ J/K$_\Omega$ pie $\beta = 0,95$ un $\varepsilon_{\Delta E_1} = 142,7594\%$

$\Delta E_2 = (2,312 \pm 2,3046) \cdot 10^{-20}$ J/K$_\Omega$ pie $\beta = 0,95$ un $\varepsilon_{\Delta E_1} = 99,6784\%$

$\Delta E_3 = (1,3623 \pm 1,2095) \cdot 10^{-20}$ J/K$_\Omega$ pie $\beta = 0,95$ un $\varepsilon_{\Delta E_1} = 88,7841\%$

$\Delta E_4 = (3,3773 \pm 5,8965) \cdot 10^{-20}$ J/K$_\Omega$ pie $\beta = 0,95$ un $\varepsilon_{\Delta E_1} = 174,5938\%$

Published with MATLAB® R2025b