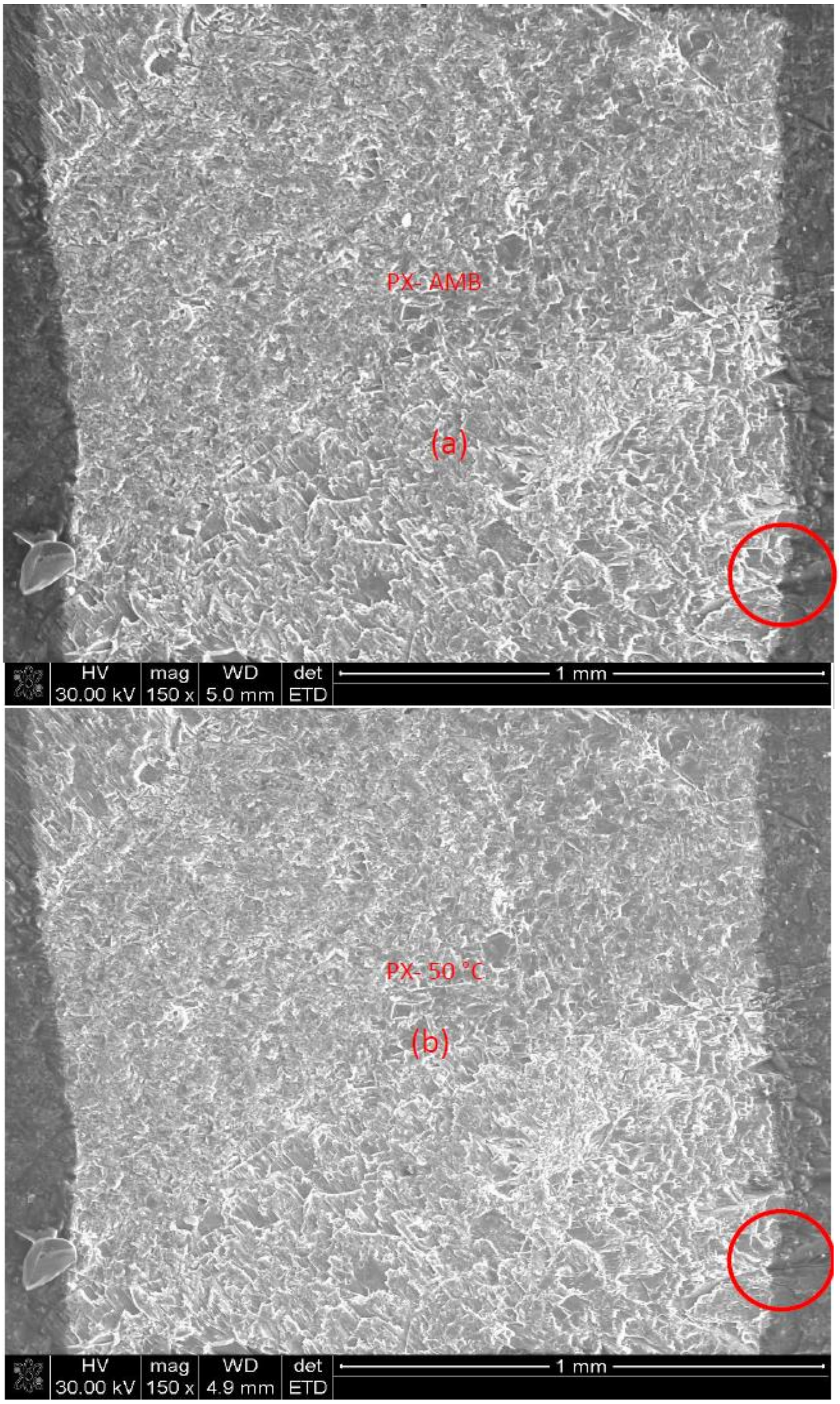

Figure A7 22: Micrograph of PX (a) before and (b) after heat treatment to 50ºC

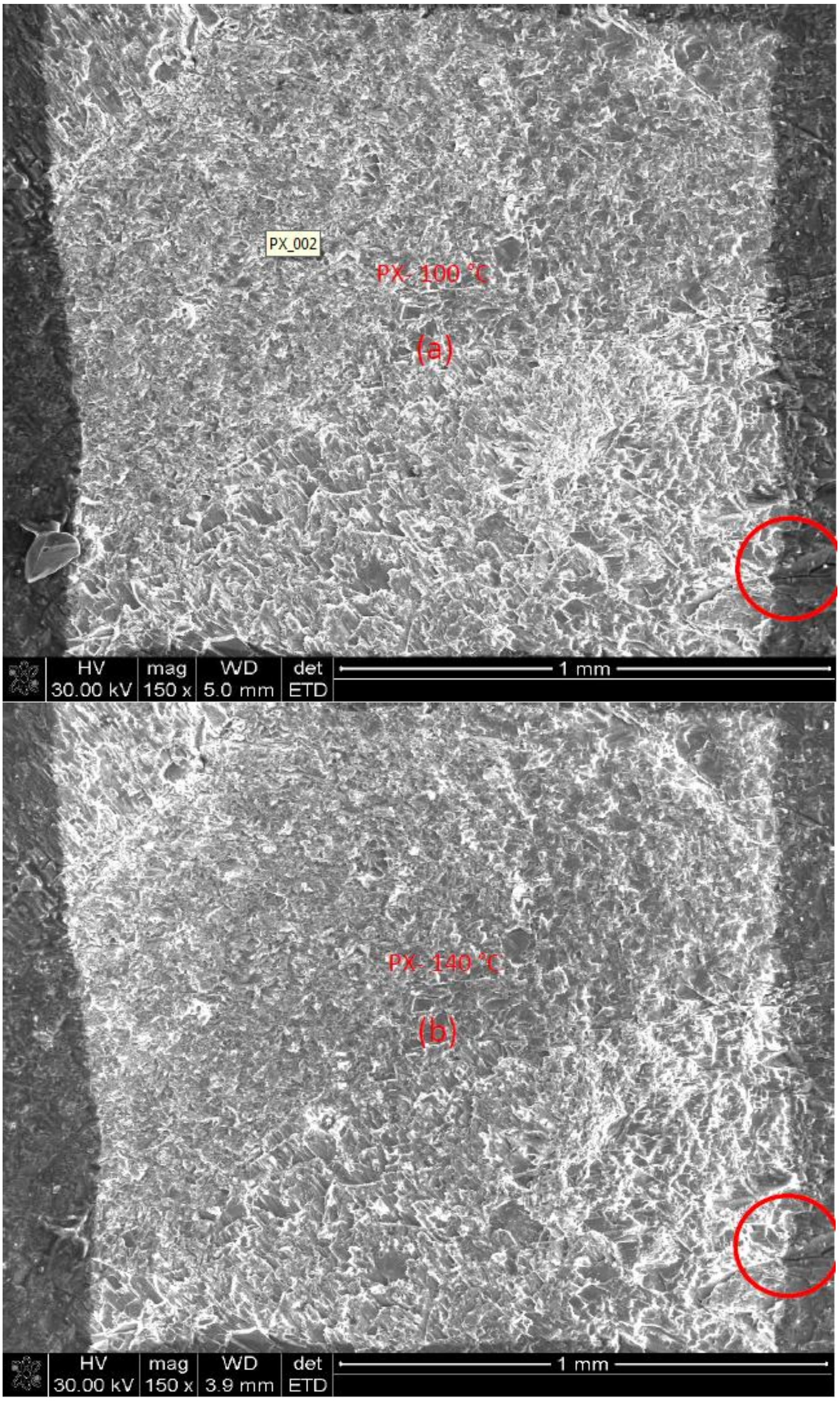

Figure A7 23: Micrograph of PX after heat treatment to 100ºC and 140ºC

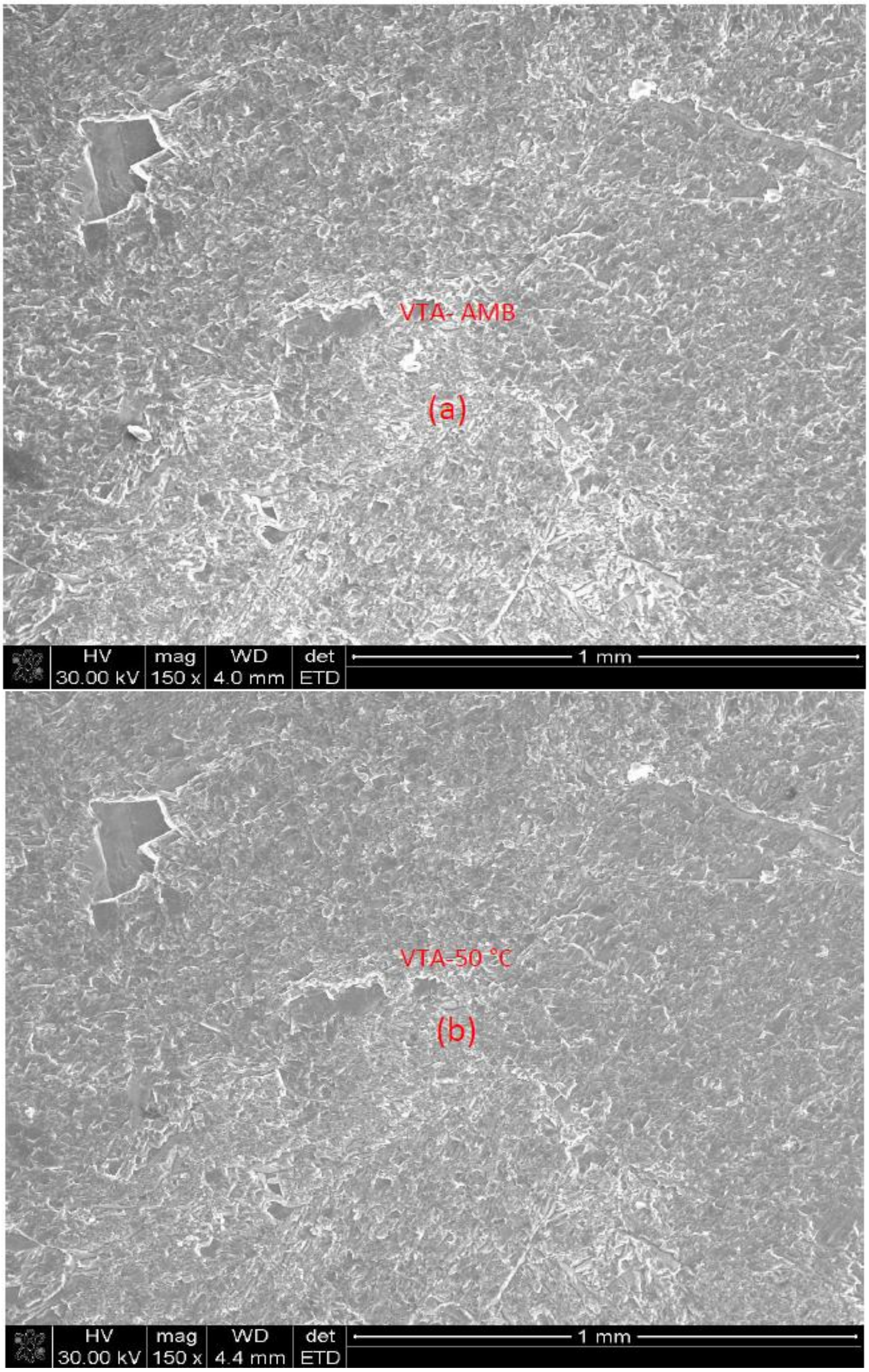

Figure A7 24: Micrograph of VTA (a) before and (b) after heat treatment to 50ºC

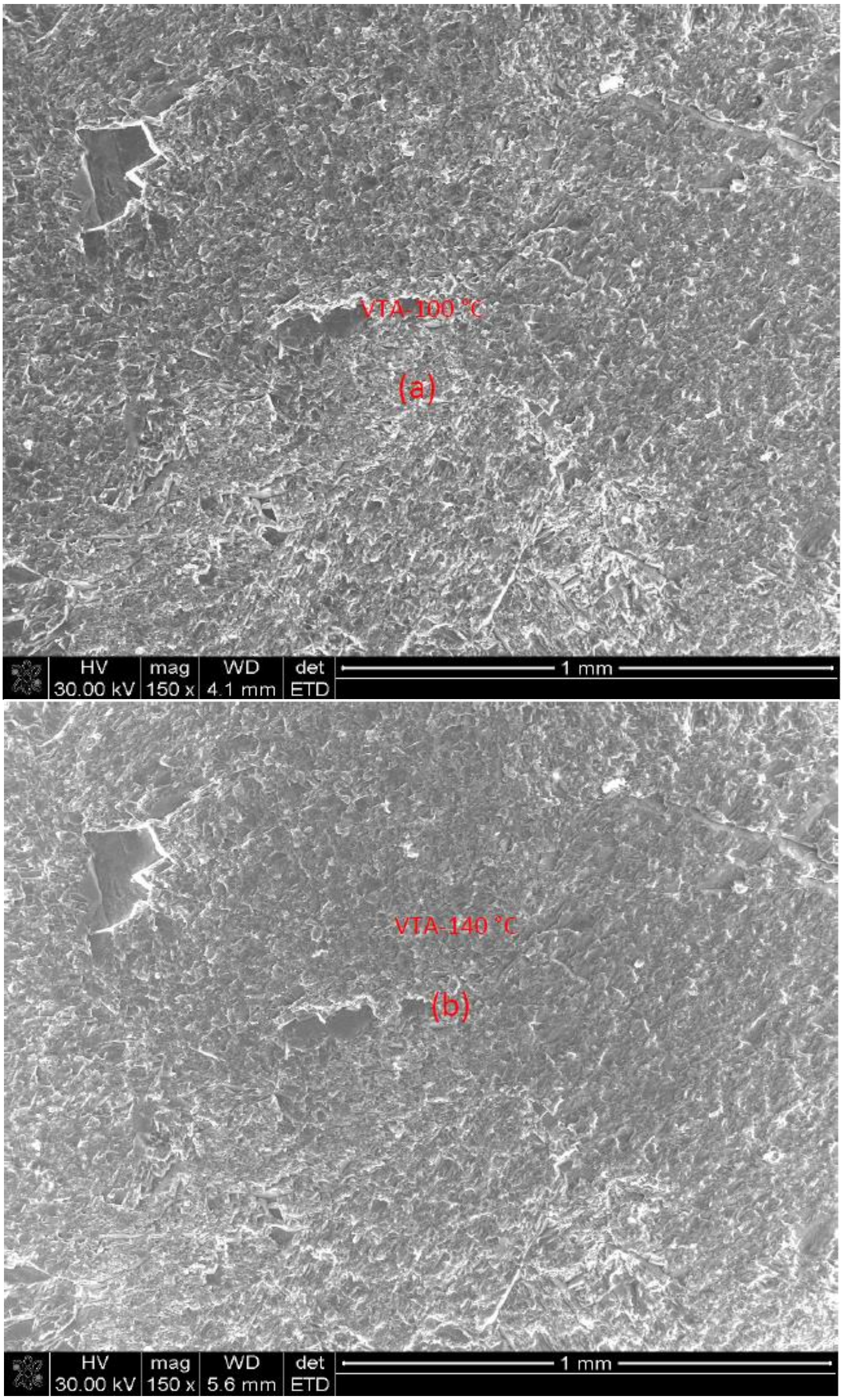

Figure A7 25: Micrograph of VTA after heat treatment to 100ºC and 140ºC

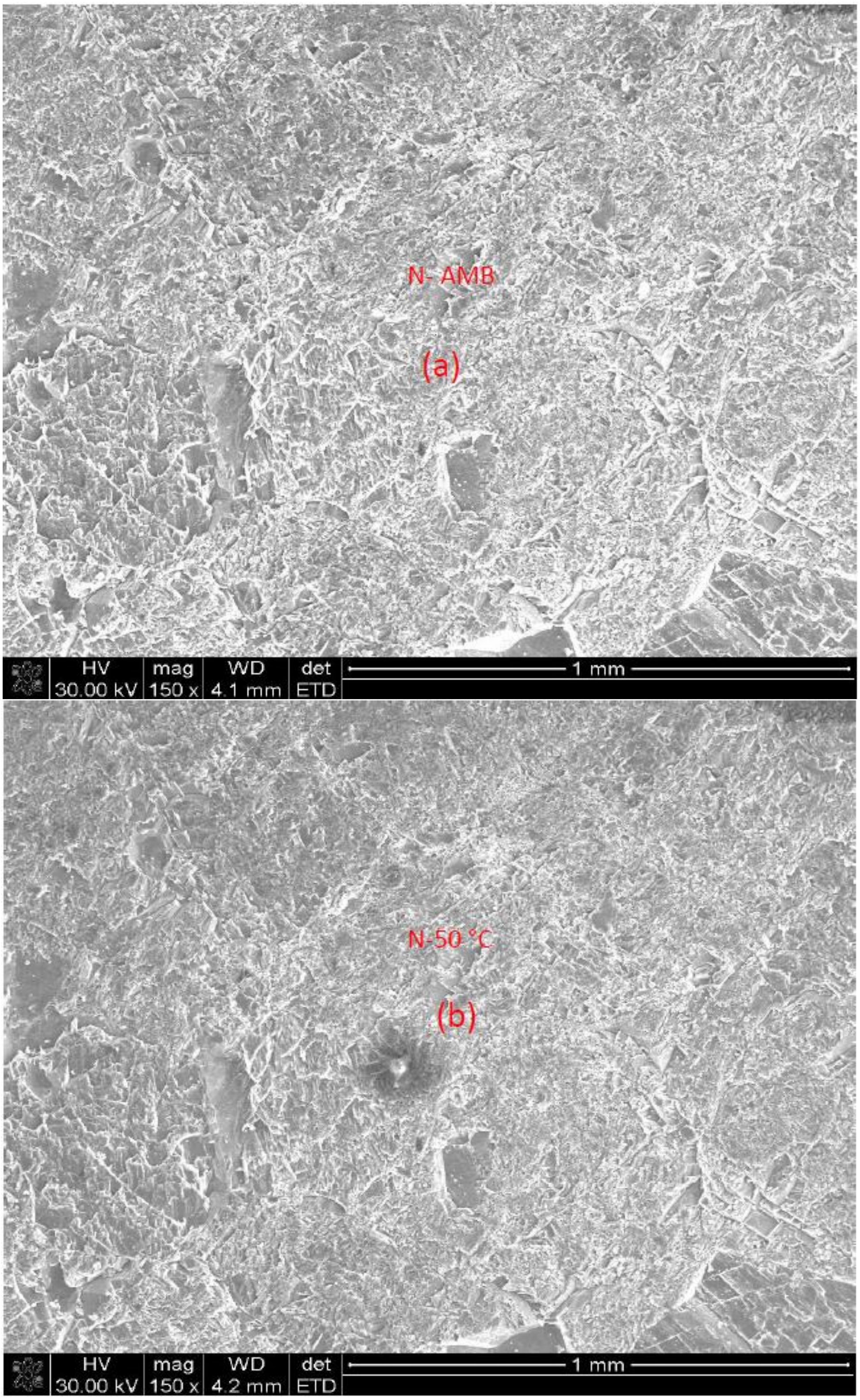

Figure A7 26: Micrograph of N (a) before and (b) after heat treatment to 50ºC

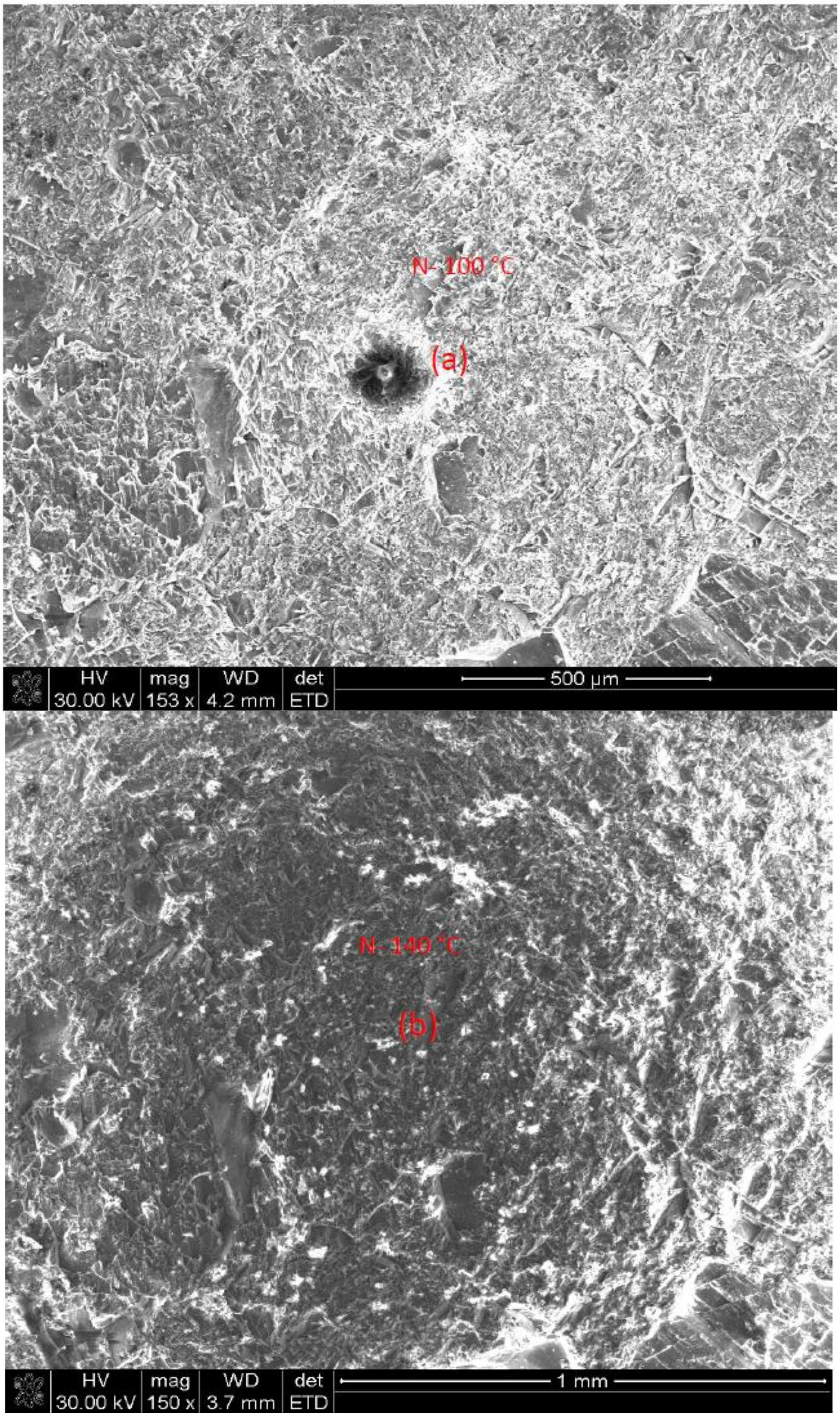

Figure A7 27: Micrograph of N after heat treatment to 100ºC and 140ºC

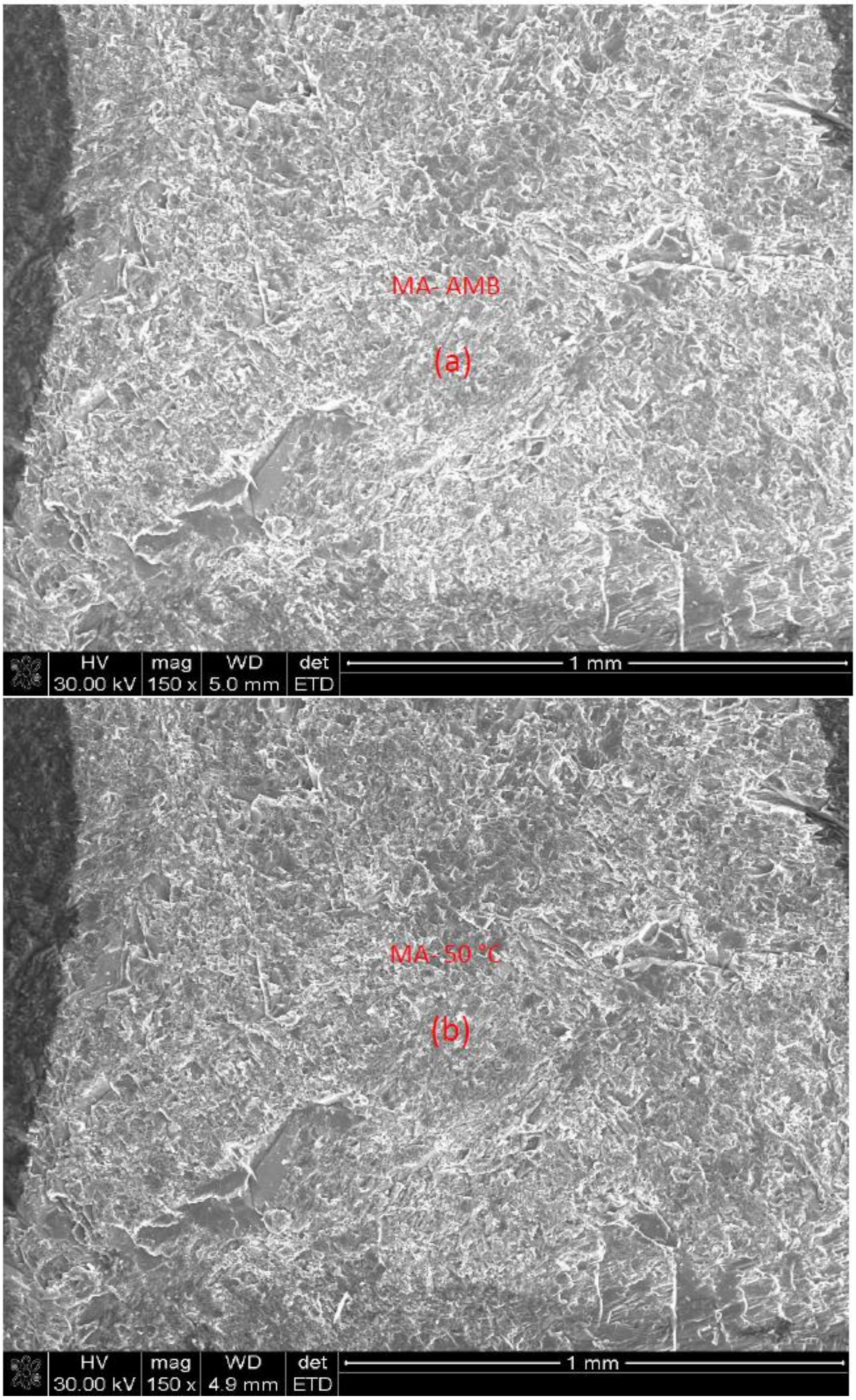

Figure A7 28: Micrograph of MA (a) before and (b) after heat treatment to 50ºC

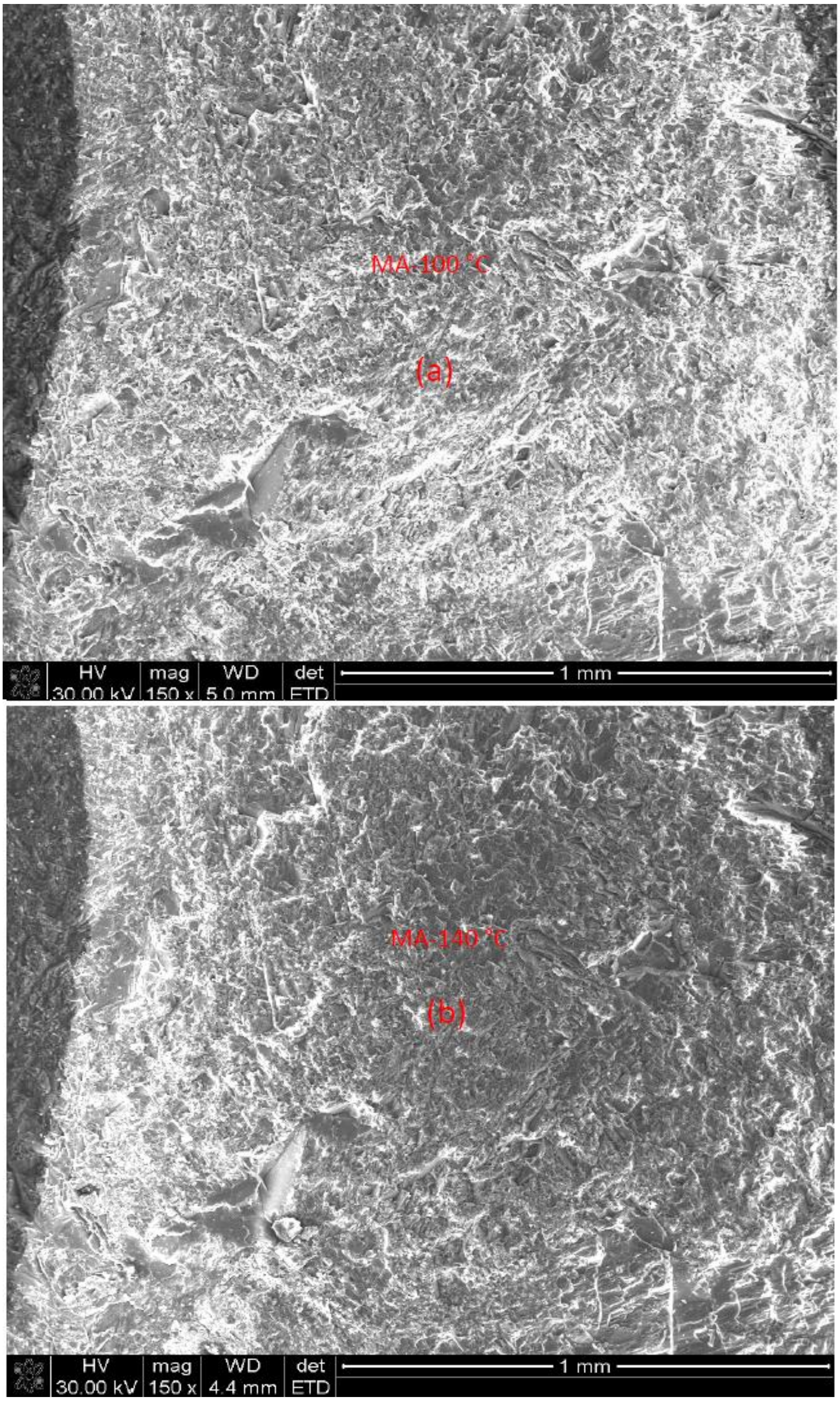

Figure A7 29: Micrograph of MA after heat treatment to 100ºC and 140ºC

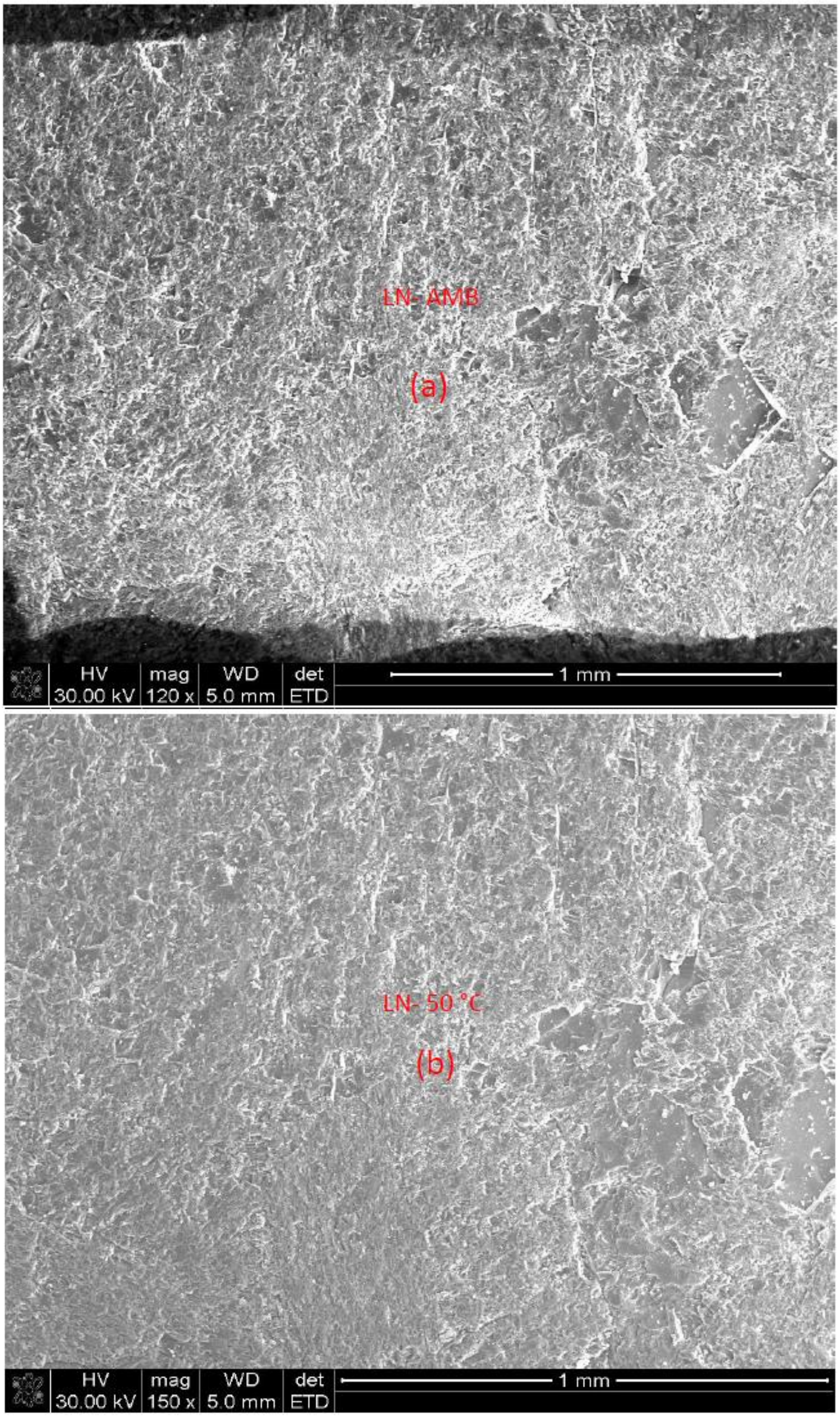

Figure A7 30: Micrograph of LN (a) before and (b) after heat treatment to 50ºC

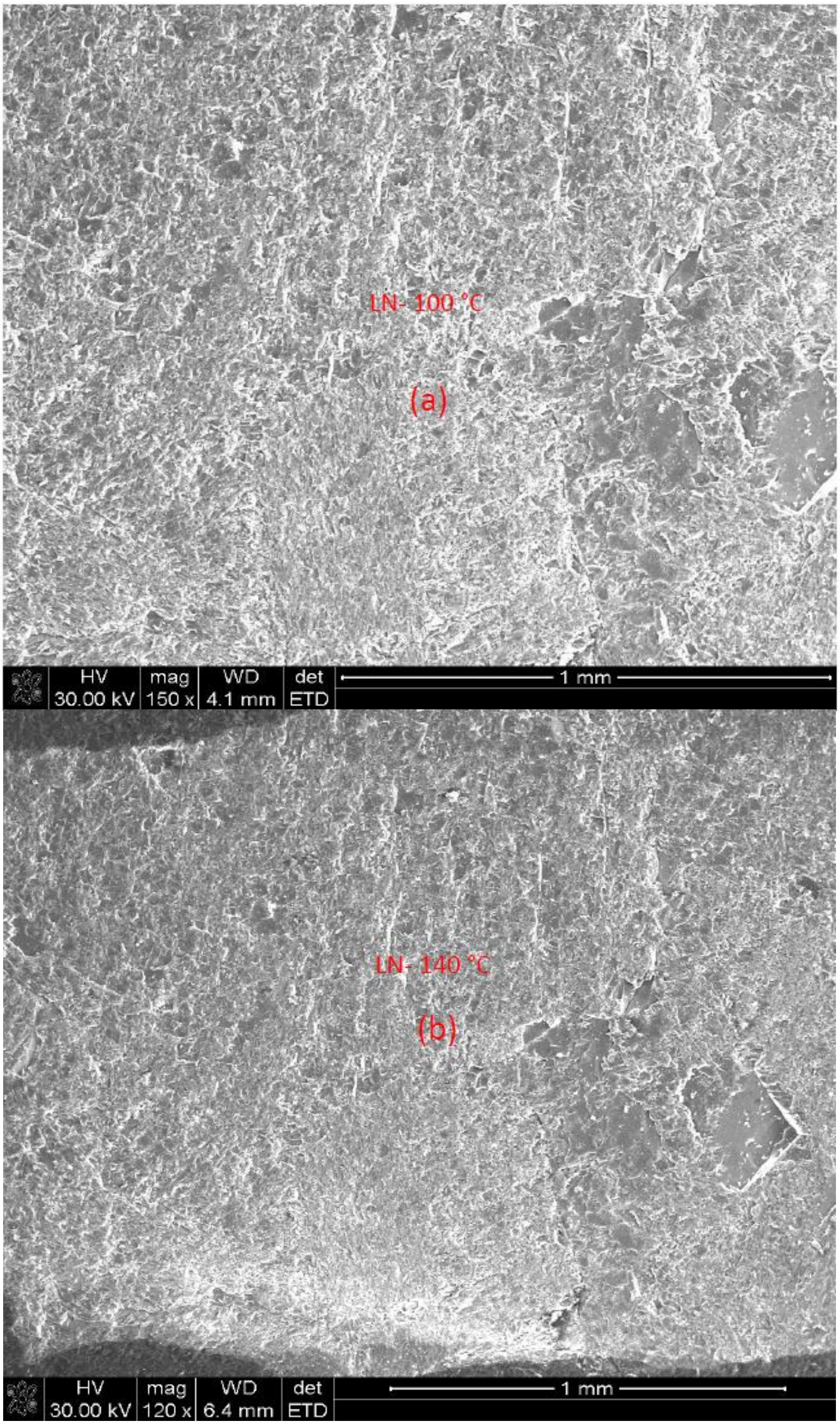

Figure A7 31: Micrograph of LN after heat treatment to 100ºC and 140ºC

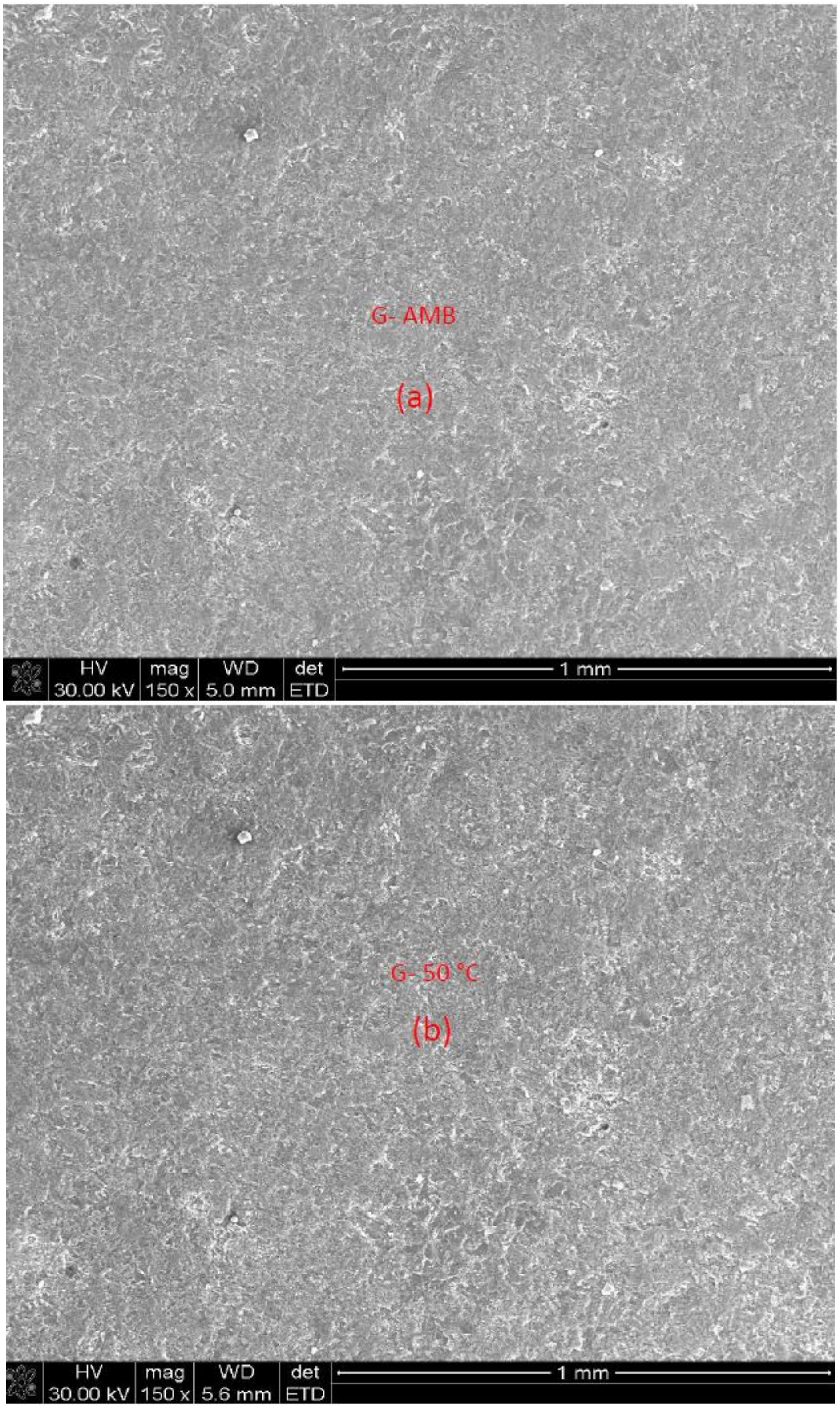

Figure A7 32: Micrograph of G (a) before and (b) after heat treatment to 50ºC

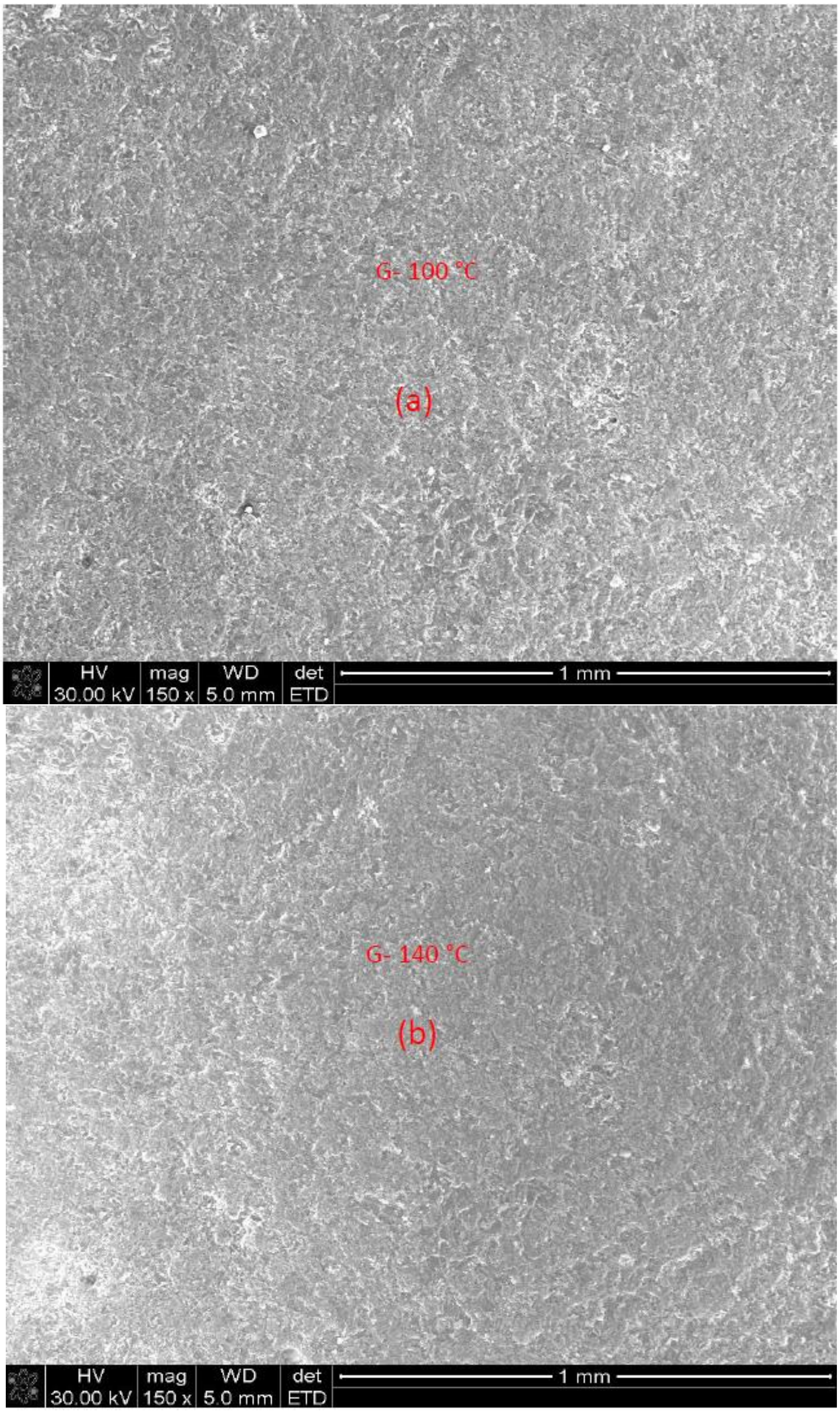

Figure A7 33: Micrograph of G after heat treatment to 100ºC and 140ºC

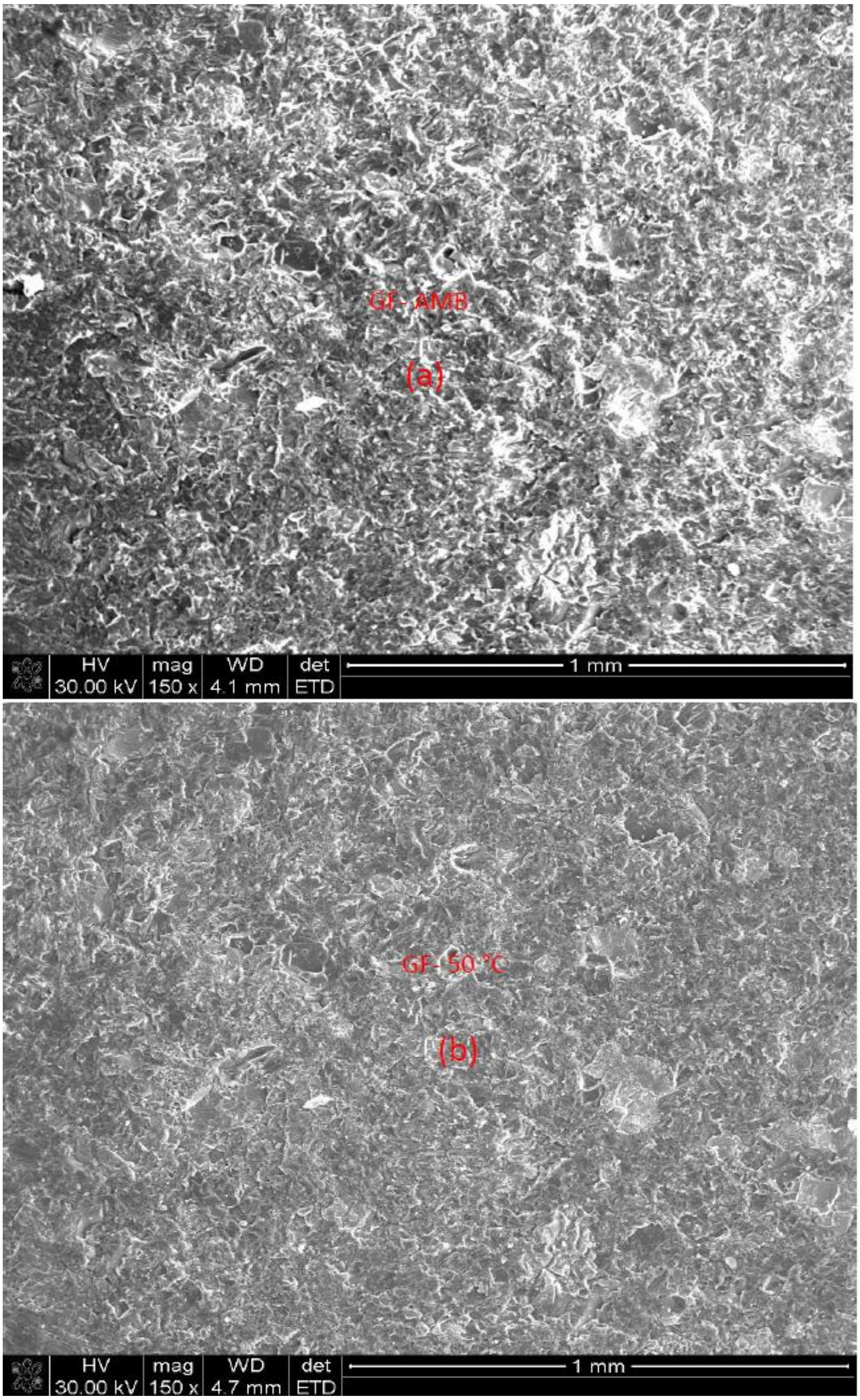

Figure A7 34: Micrograph of GF (a) before and (b) after heat treatment to 50ºC

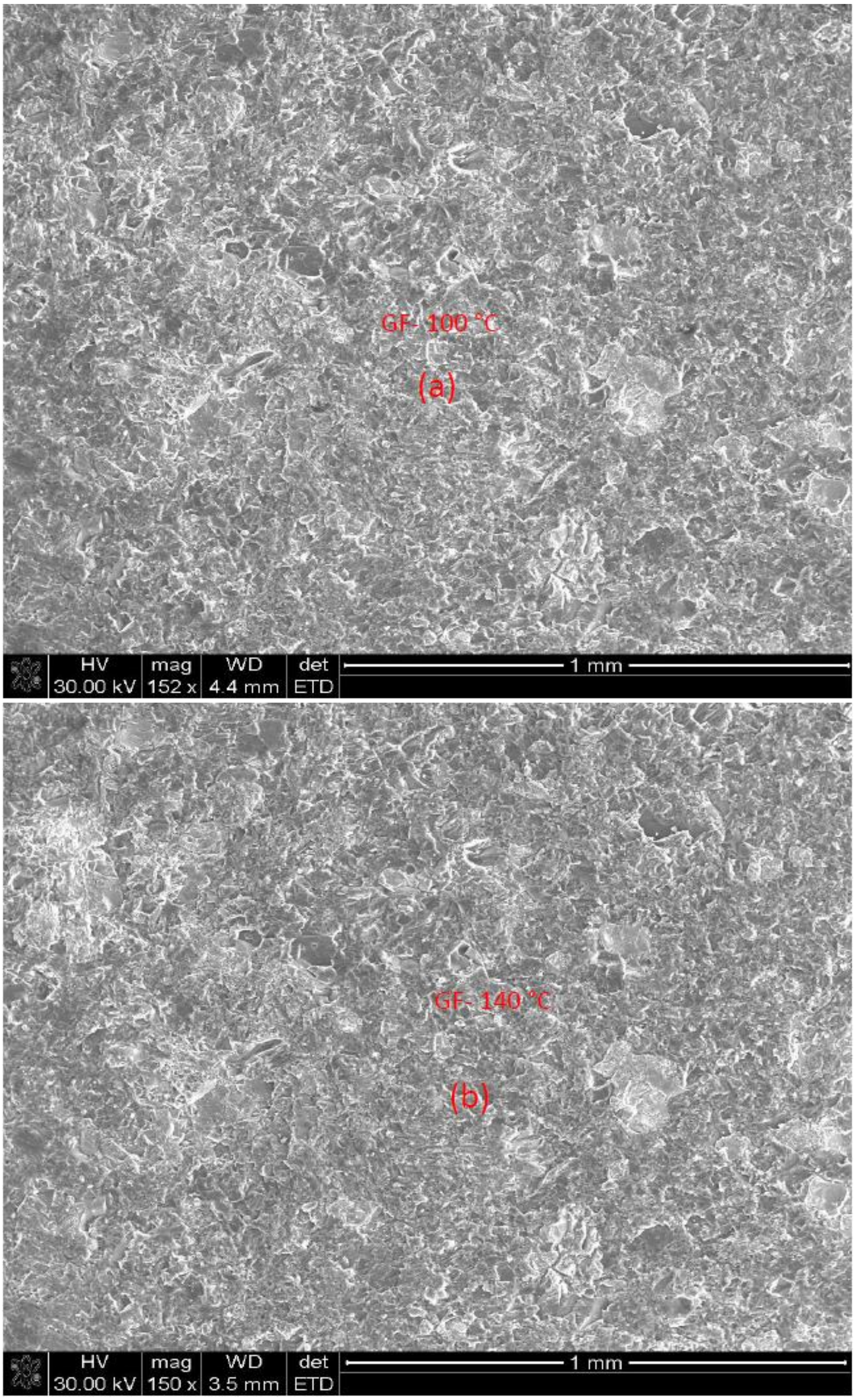

Figure A7 35: Micrograph of GF after heat treatment to 100ºC and 140ºC

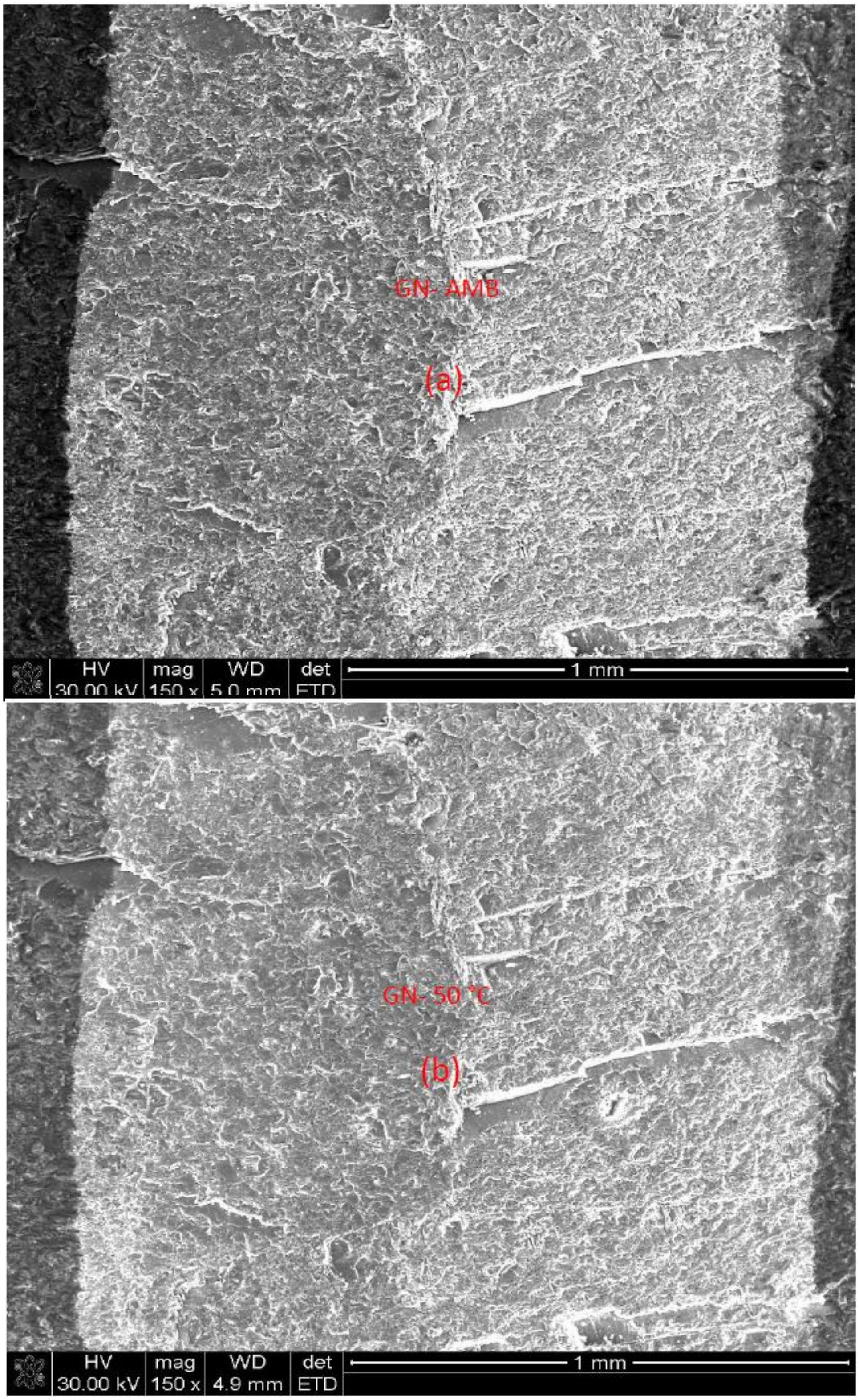

Figure A7 36: Micrograph of GN (a) before and (b) after heat treatment to 50ºC

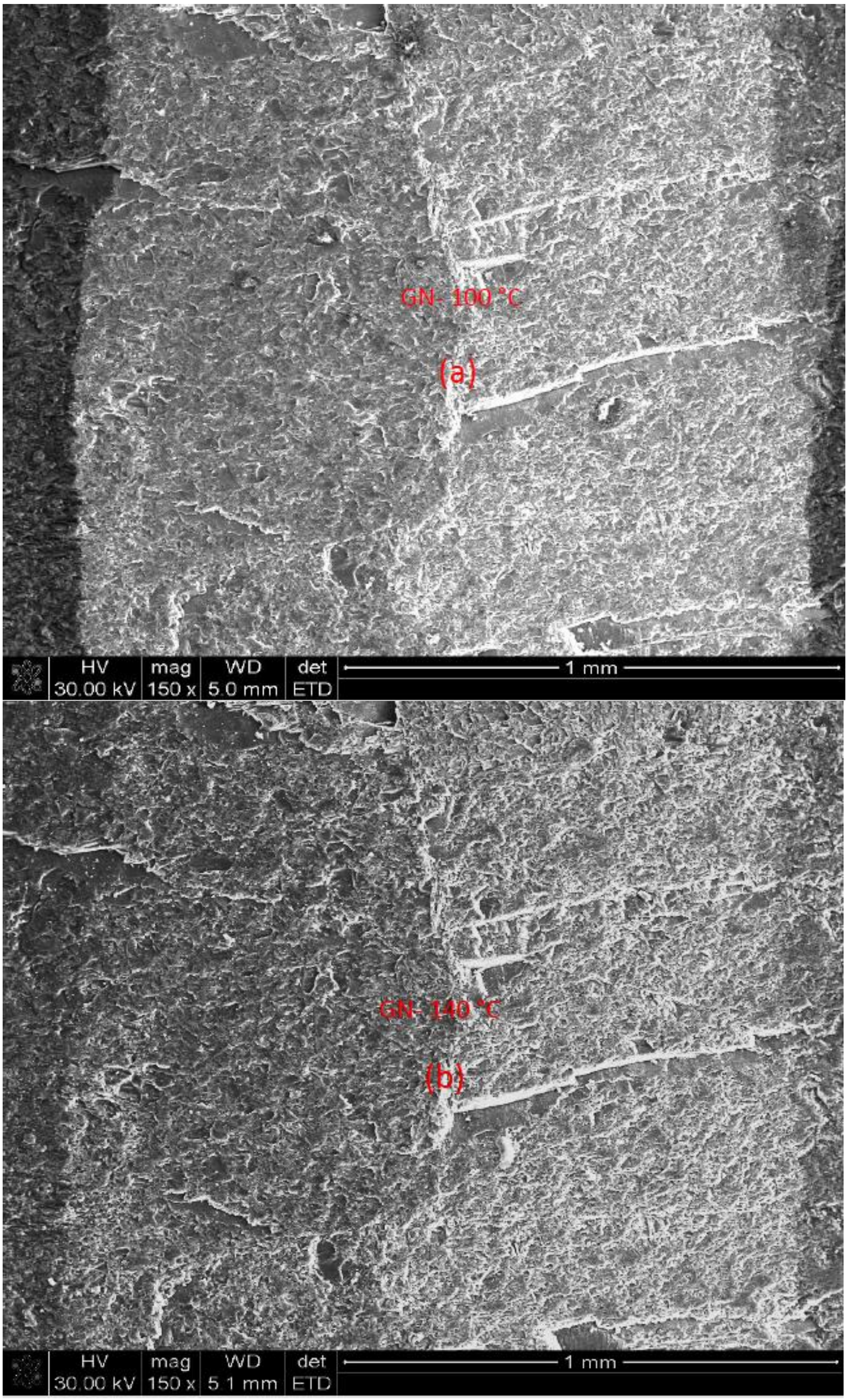

Figure A7 37: Micrograph of GN after heat treatment to 100ºC and 140ºC

Digital Imaging and Spectra Collecting with INCA

Procedure for digital imaging and measurement of EDS

1(a) The desired Accelerating Voltage was selected: for EDS, the recommended accelerating voltage is 1.5 to 2 times the highest energy line in the spectrum. 20kV was chosen since it will excite most X-ray lines from most elements for analysis.

(b) The Objective Aperture 3 was selected for EDS.

(c) The working distance (WD) was set to 10 mm.

(d) A high quality sample image was attained by fine alignments and adjustments.

(e) **The IR camera is turned off** by minimizing or closing the IR camera window before collecting any spectra.

2. There are three analysis options for the program:

(a) **[Analyzer]** – allows for EDS acquisition and quantitative analysis but no imaging acquisition.

(b) **[Point & ID]** – allows image acquisition and site specific EDS analysis.

(c) **[Mapping]** – allows for EDS mappings and linescans using a data mining approach (i.e. a full spectrum is stored from each pixel). The third analysis option, that is, mapping was used.

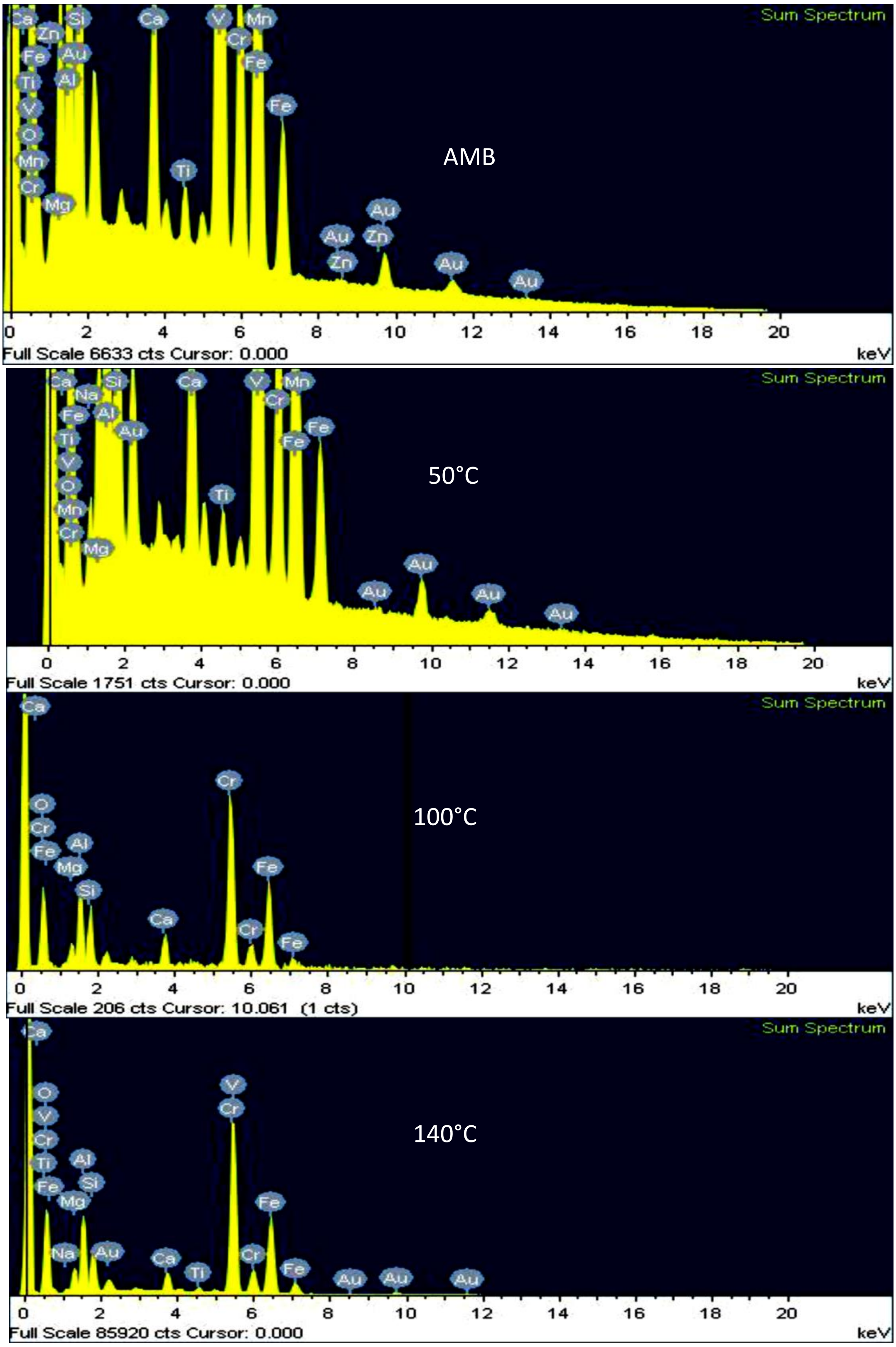

Figure A7 38: EDS spectra of chromitite for heat treatments at different temperatures

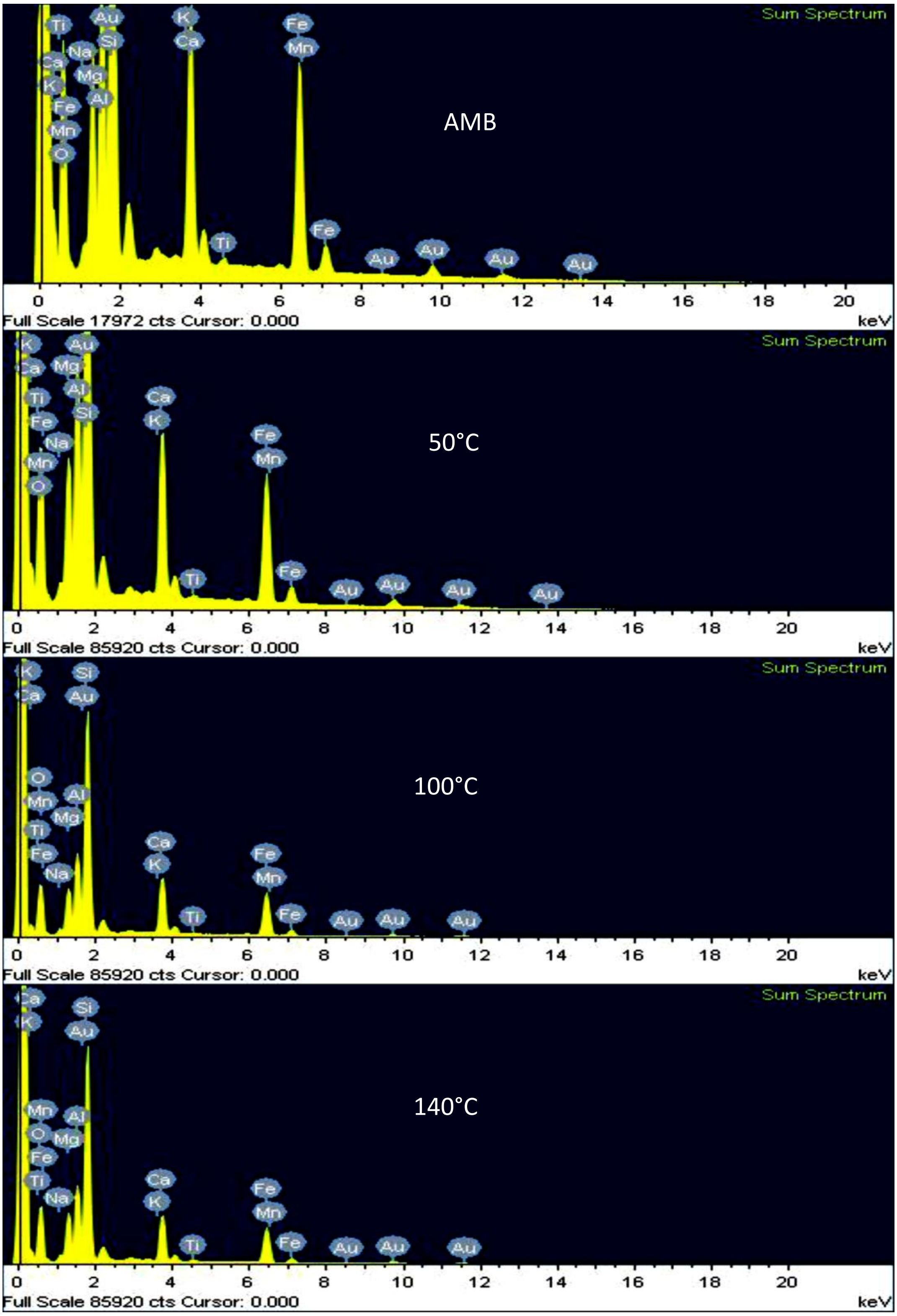

Figure A7 39: EDS spectra of gabbronorite for heat treatments at different temperatures

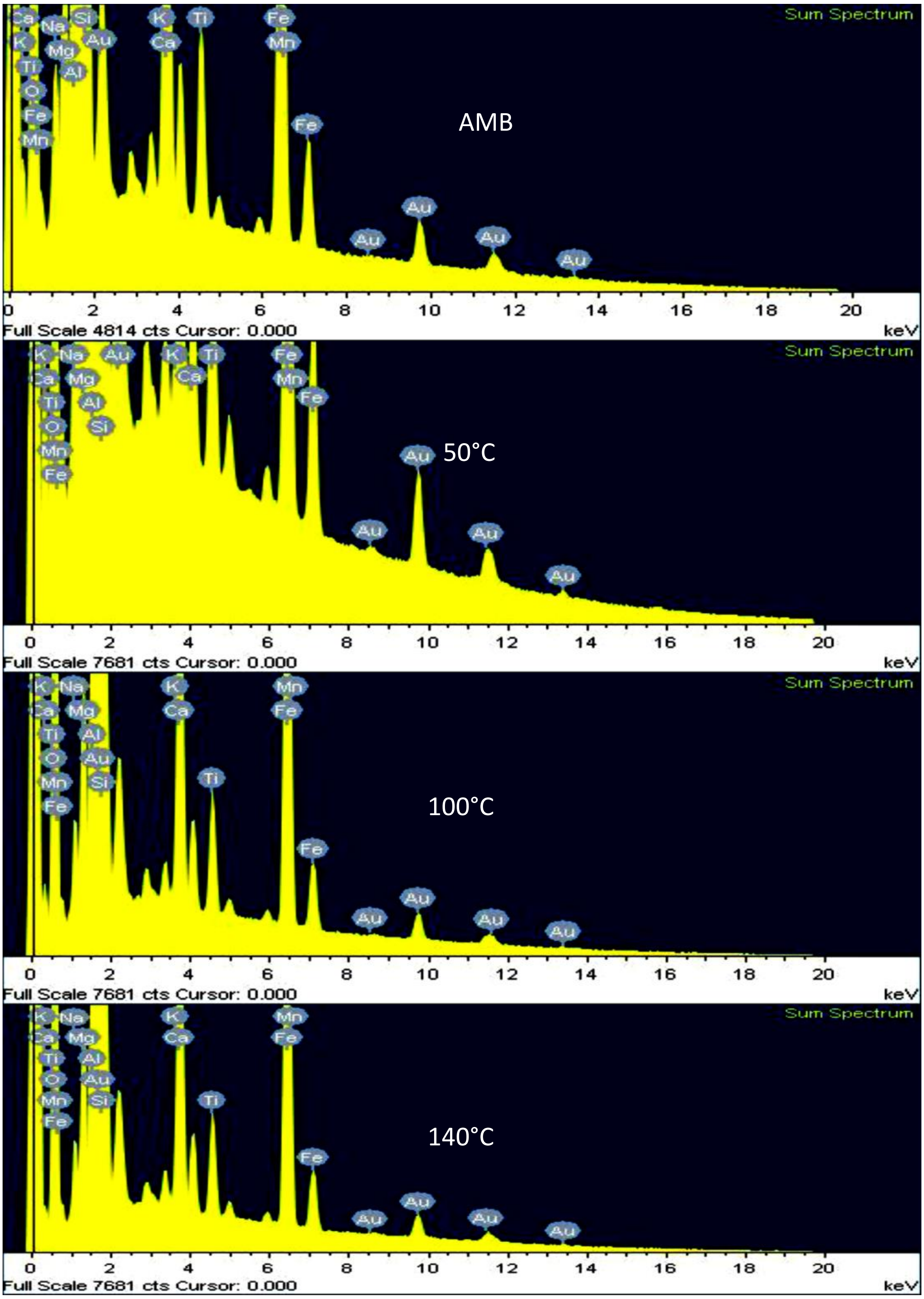

Figure A7 40: EDS spectra of granofels for heat treatments at different temperatures

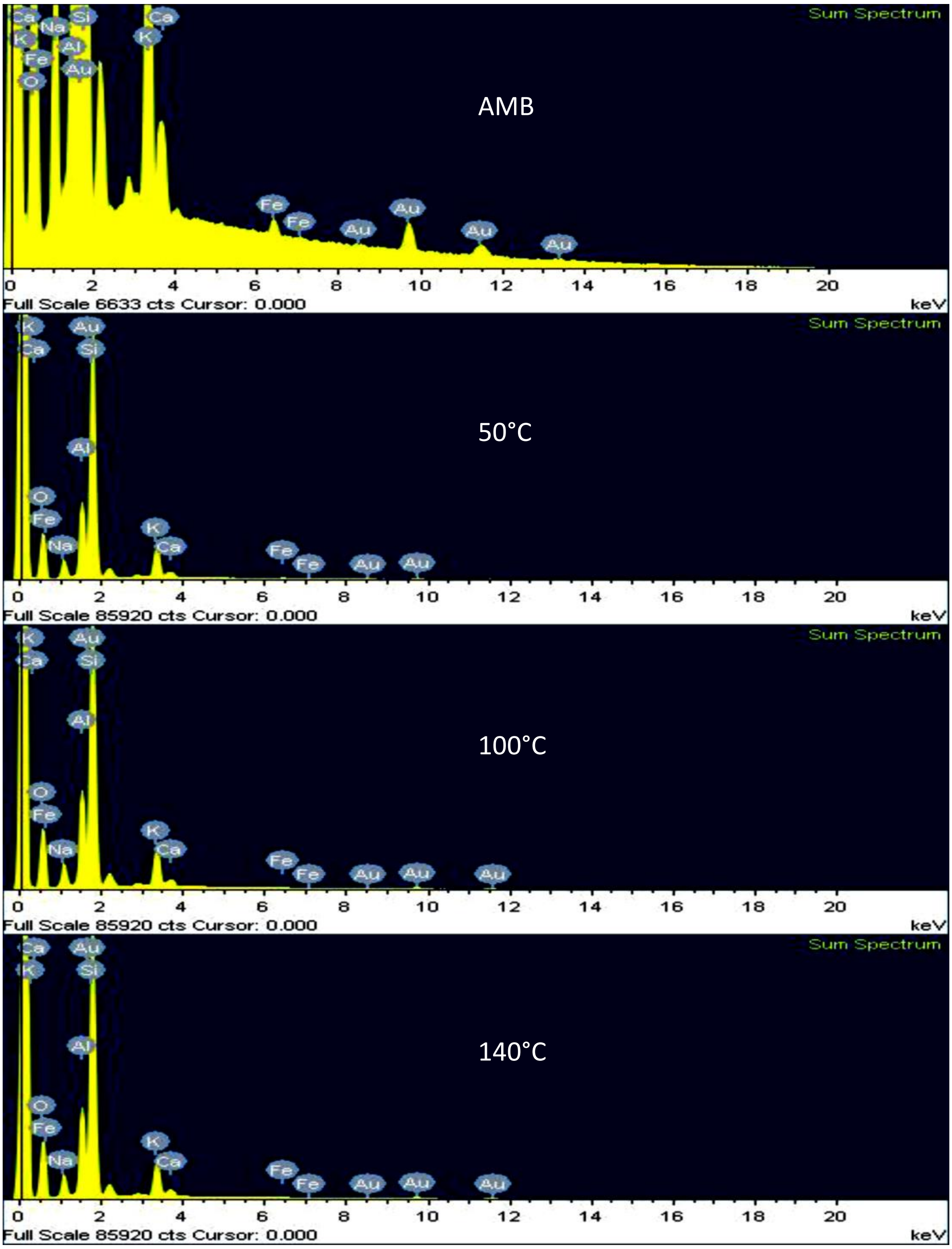

Figure A7 41: EDS spectra of granite for heat treatments at different temperatures

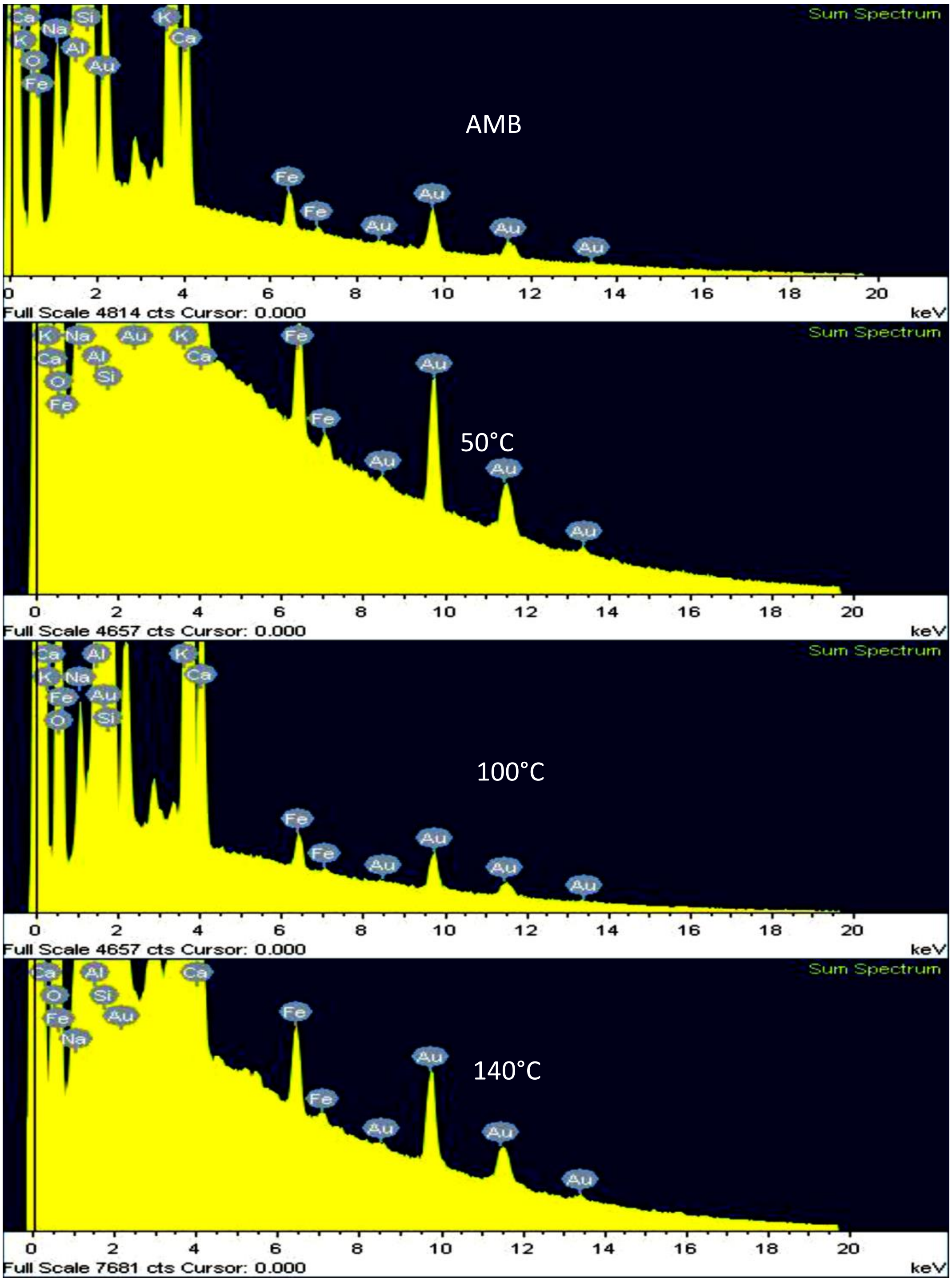

Figure A7 42: EDS spectra of leuconorite for heat treatments at different temperatures

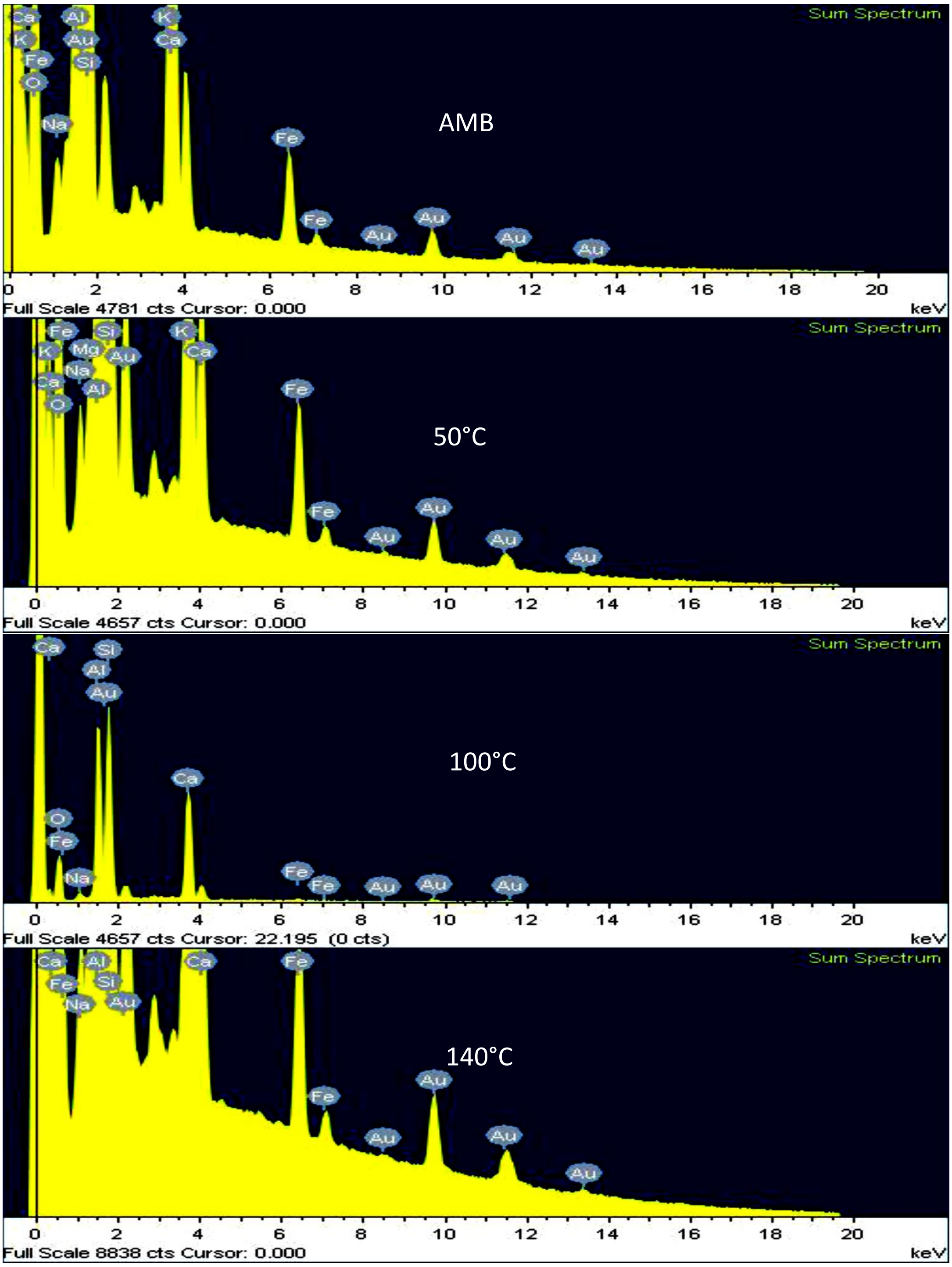

Figure A7 43: EDS spectra of mottled anorthosite for heat treatments at different temperatures

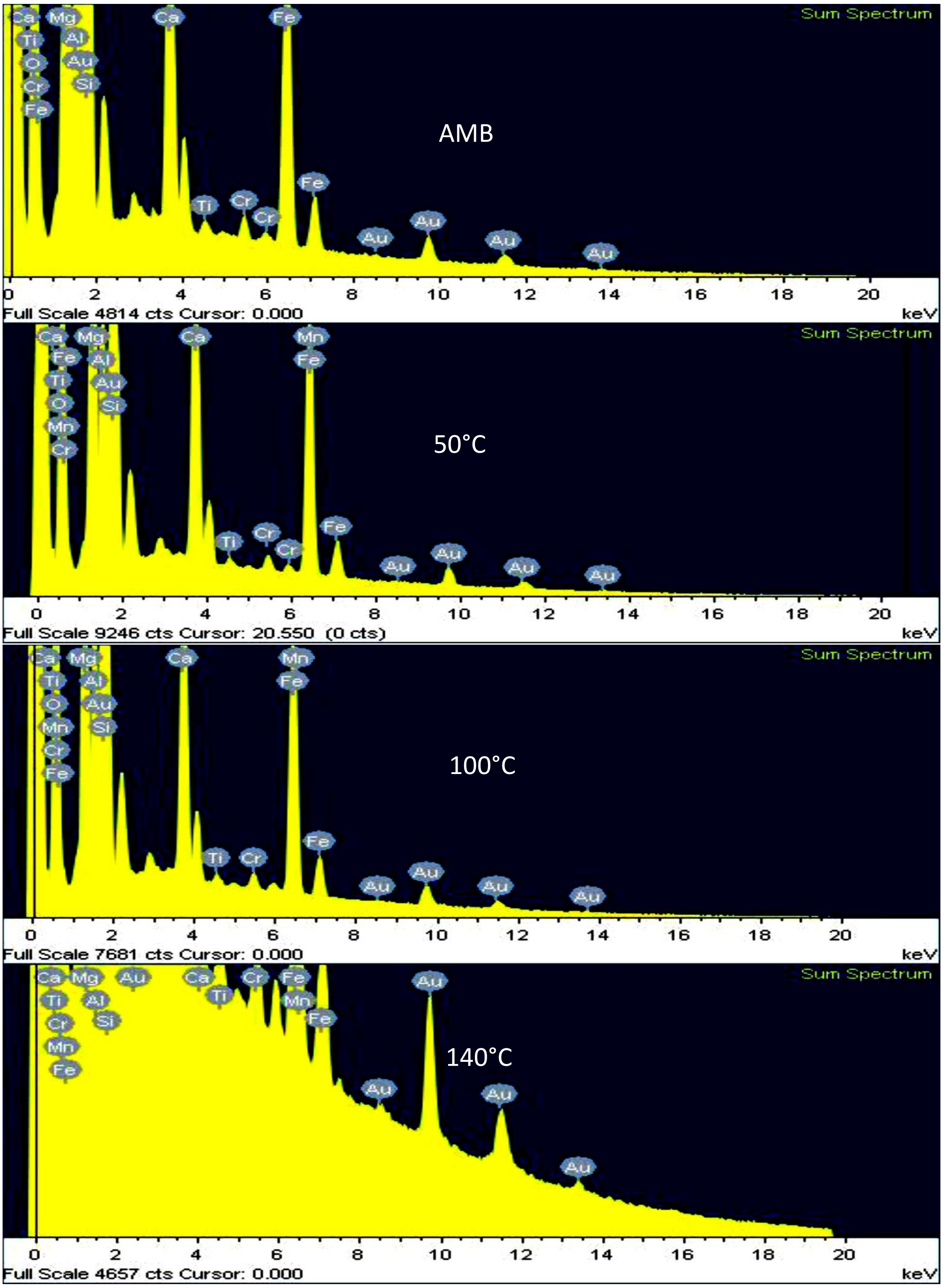

Figure A7 44: EDS spectra of norite for heat treatments at different temperatures

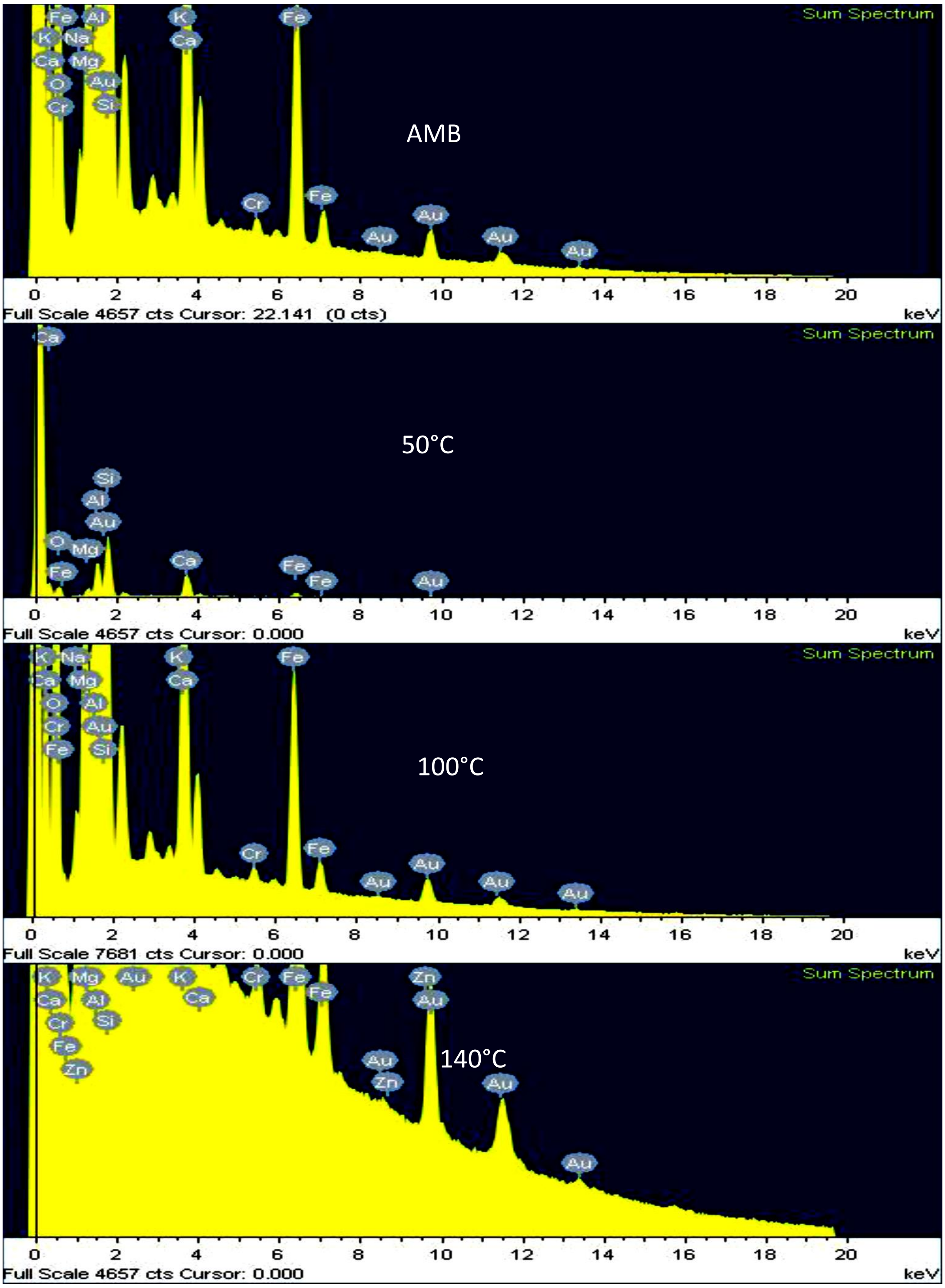

Figure A7 45: EDS spectra of pyroxenite for heat treatments at different temperatures

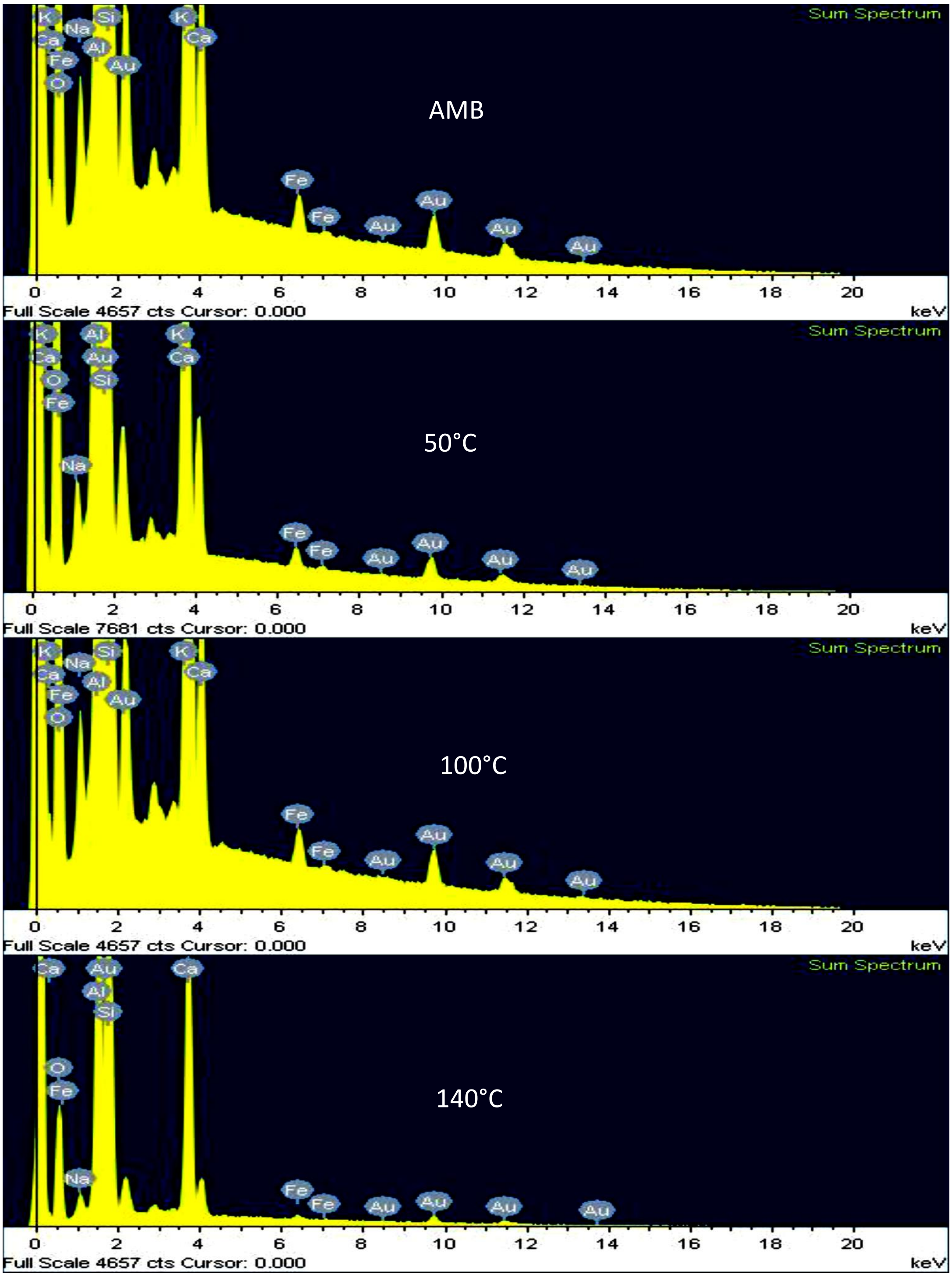

Figure A7 46: EDS spectra of varitextured anorthosite for heat treatments at different temperatures

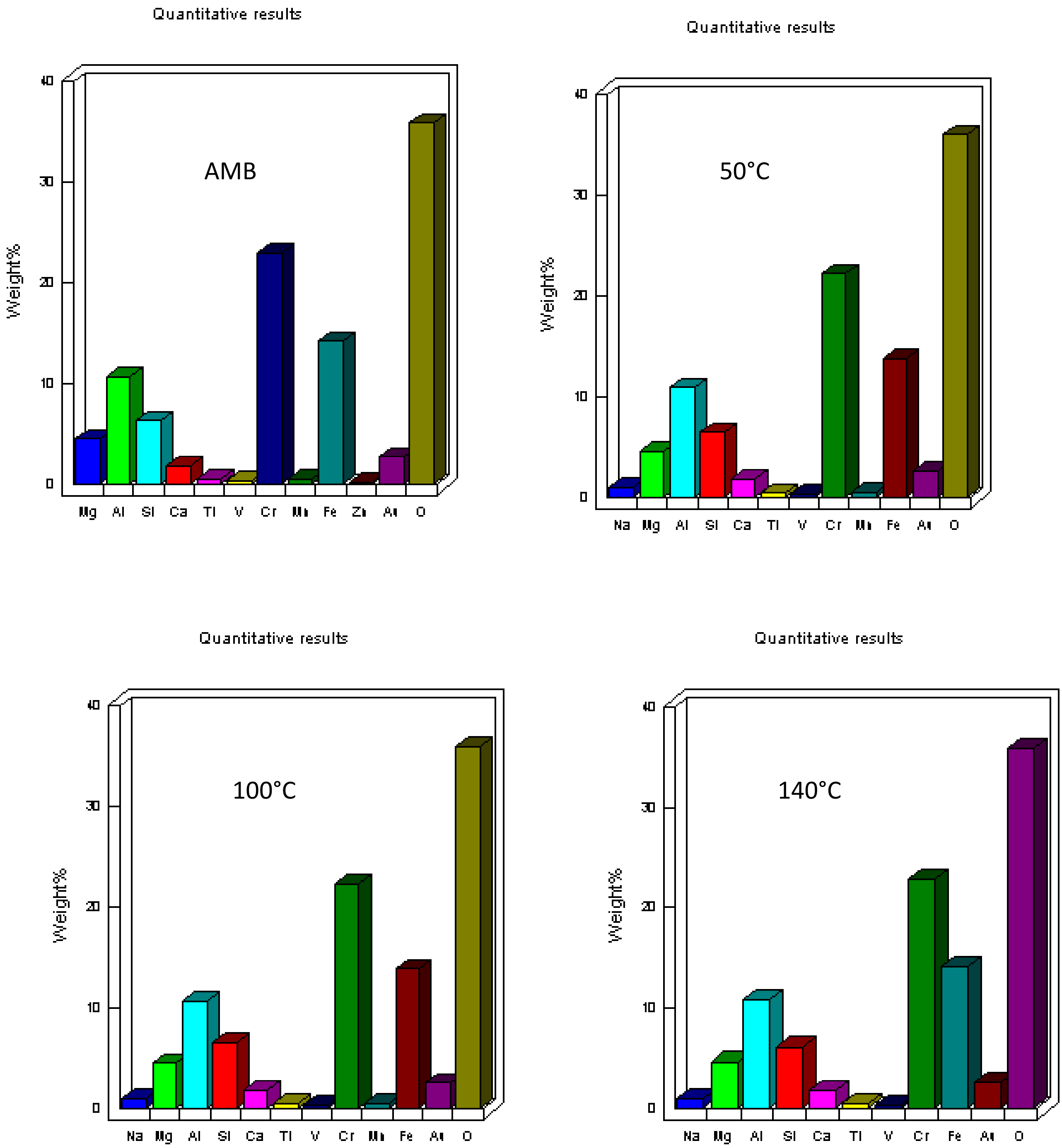

Figure A7 47: Elemental composition of chromitite for heat treatments at different temperatures

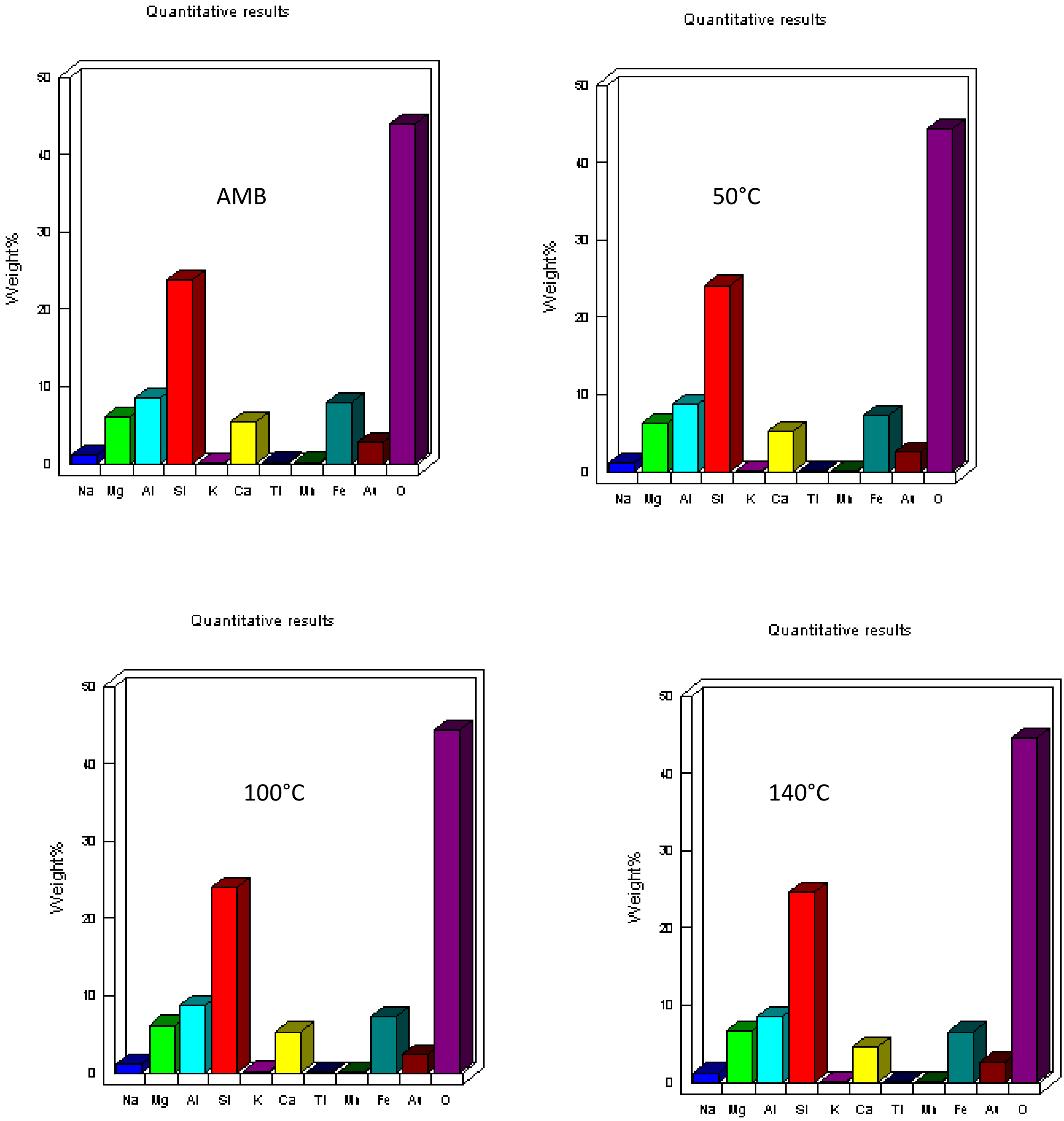

Figure A7 48: Elemental composition of gabbronorite for heat treatments at different temperatures

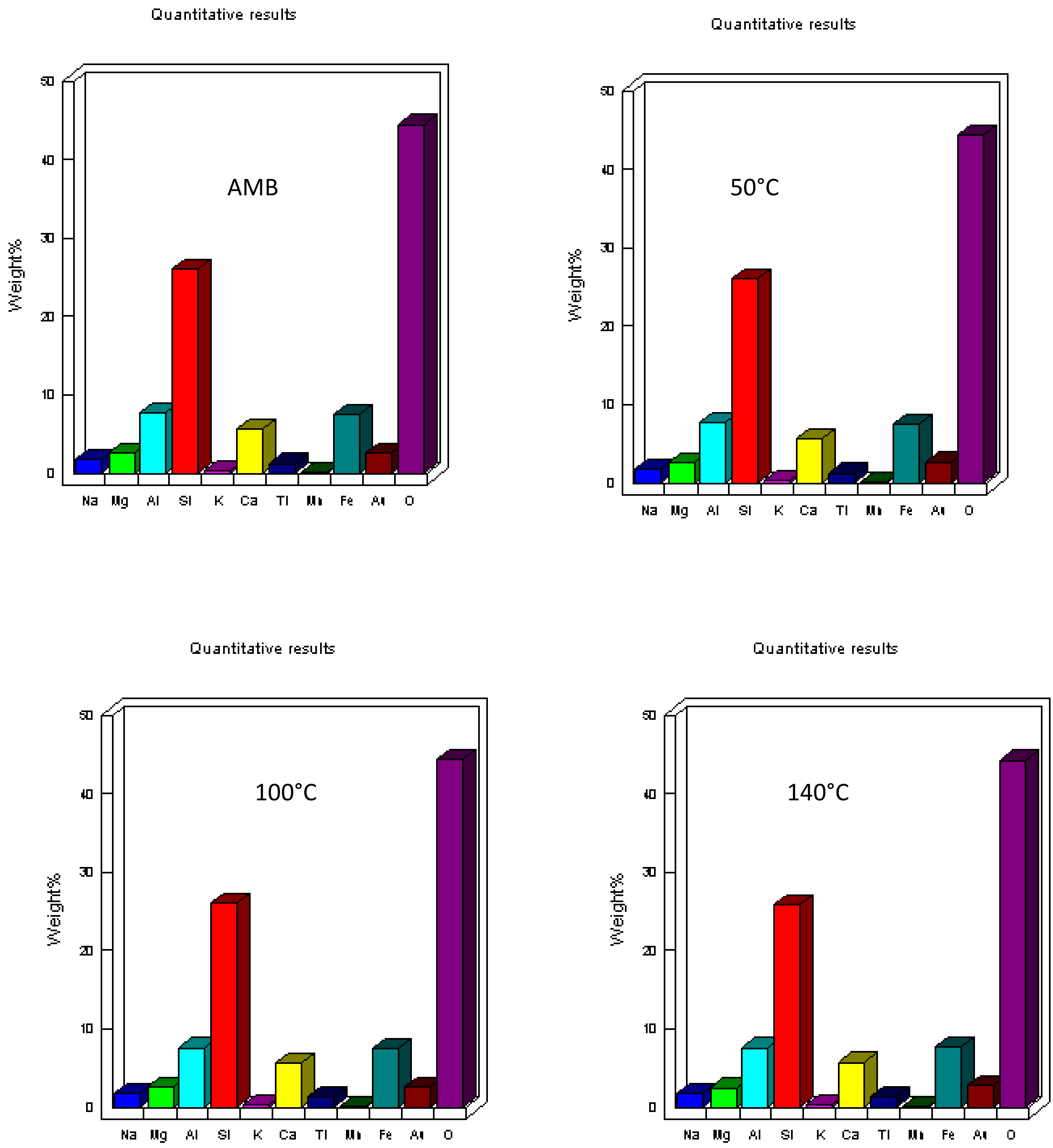

Figure A7 49: Elemental composition of granofels for heat treatments at different temperatures

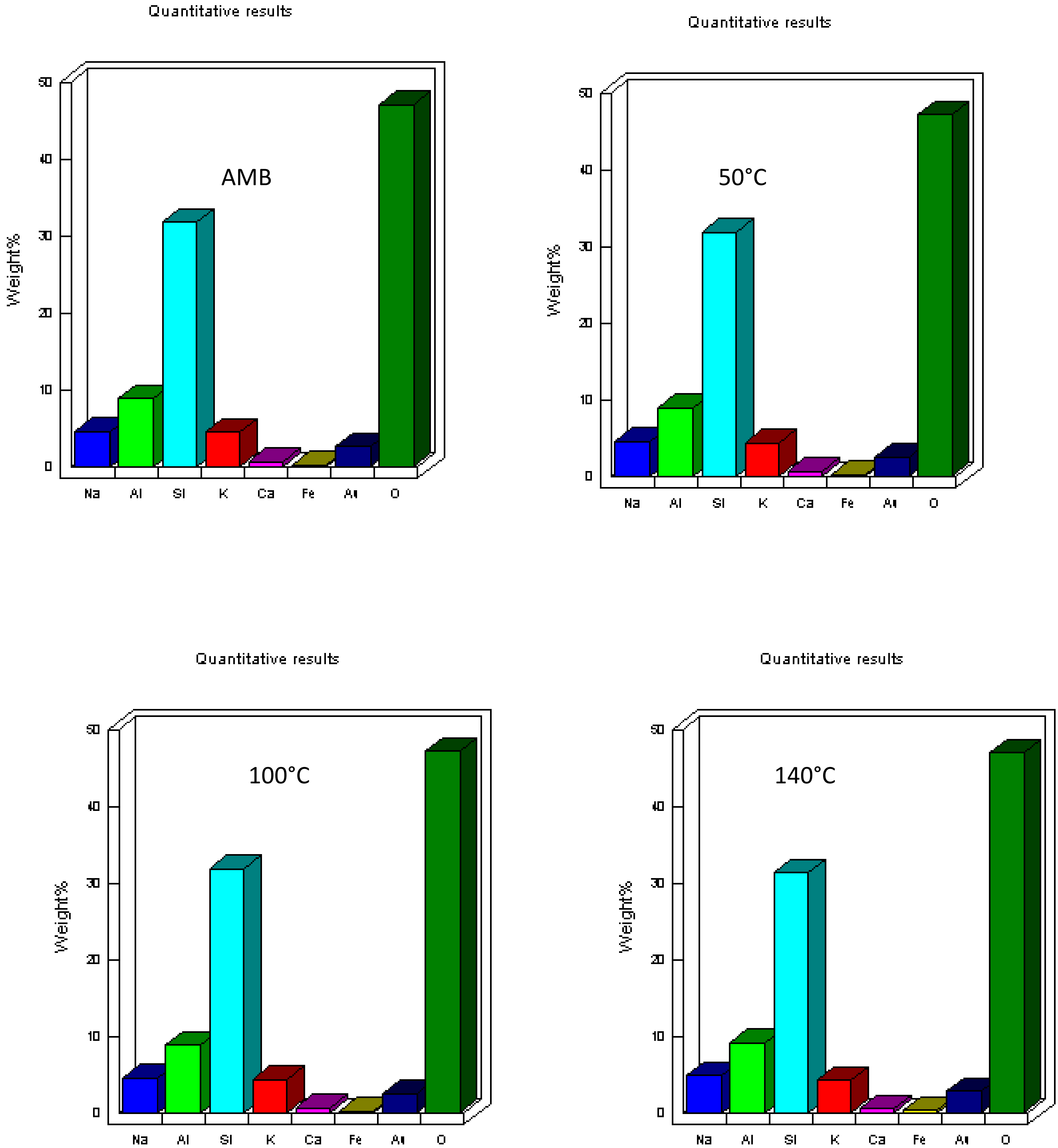

Figure A7 50: Elemental composition of granite for heat treatments at different temperatures

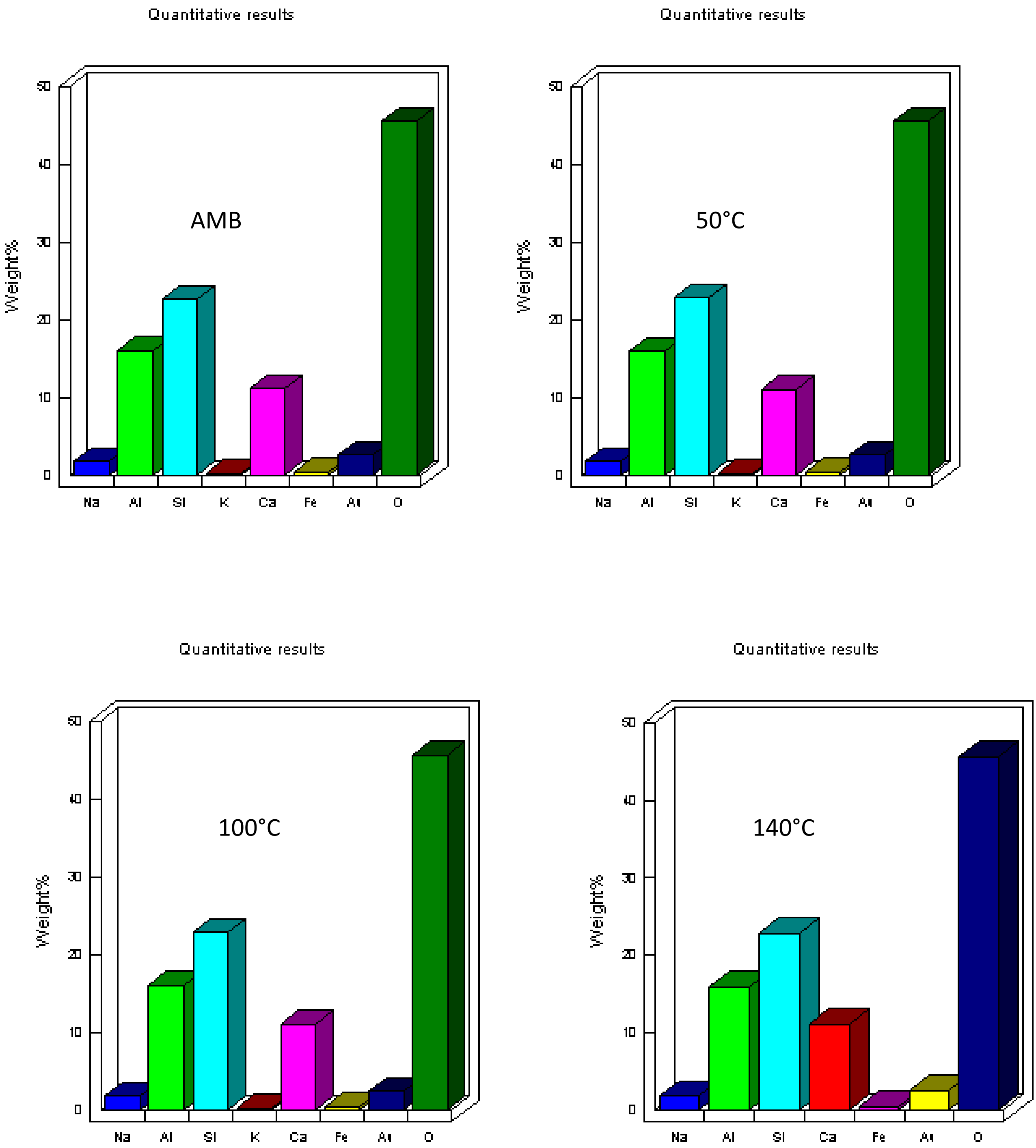

Figure A7 51: Elemental composition of leuconorite for heat treatments at different temperatures

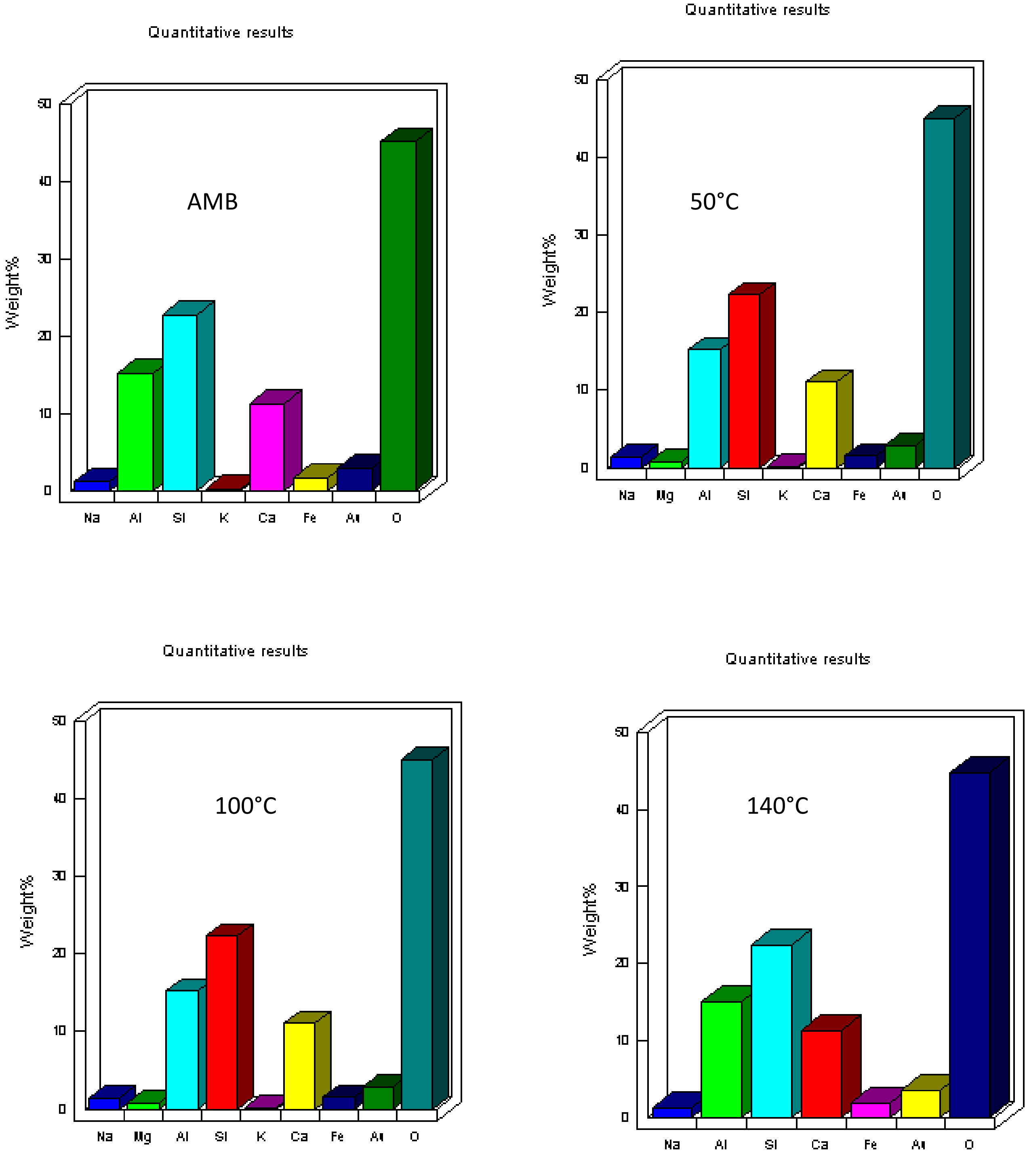

Figure A7 52: Elemental composition of mottled anorthosite for heat treatments at different temperatures

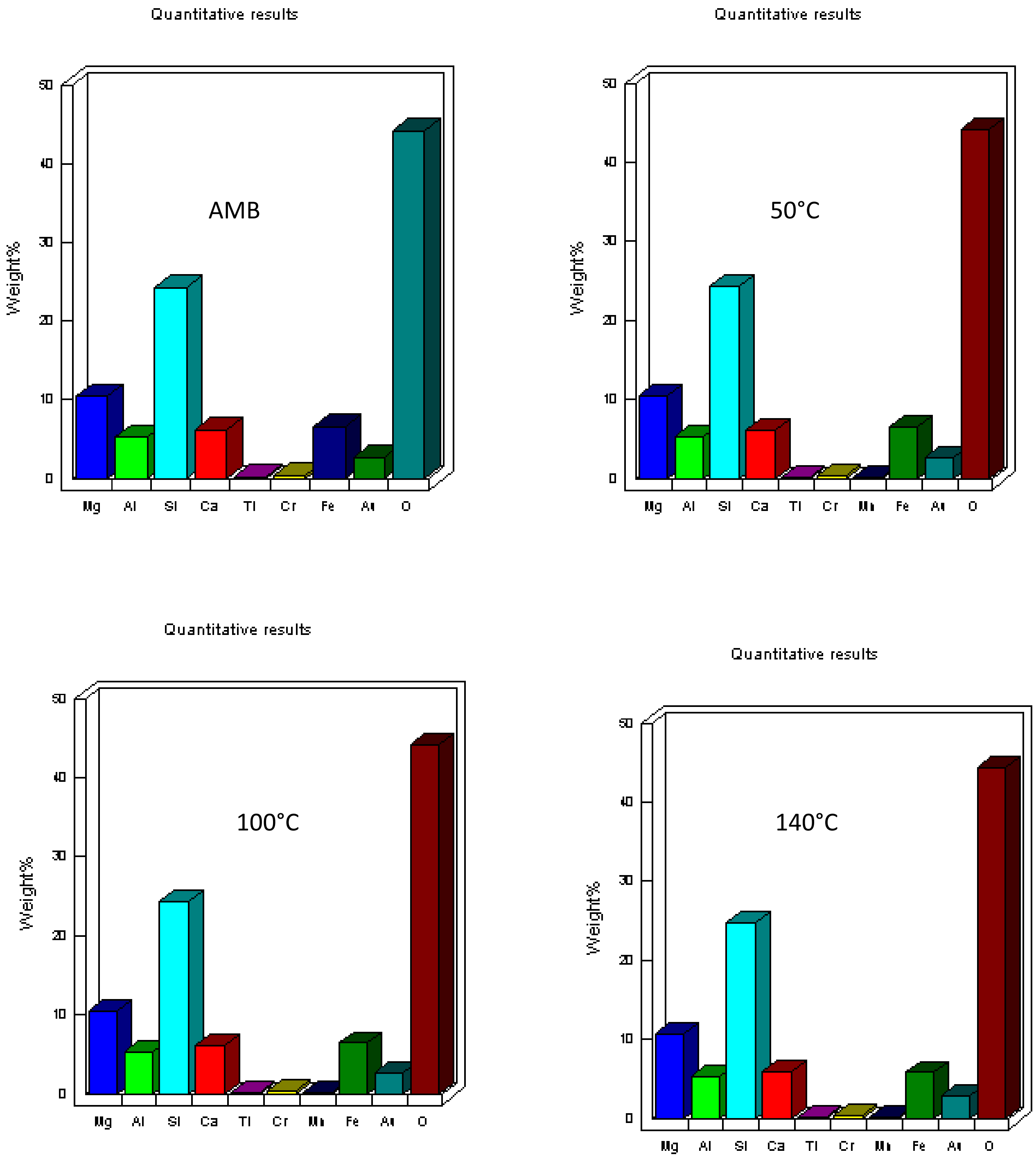

Figure A7 53: Elemental composition of norite for heat treatments at different temperatures

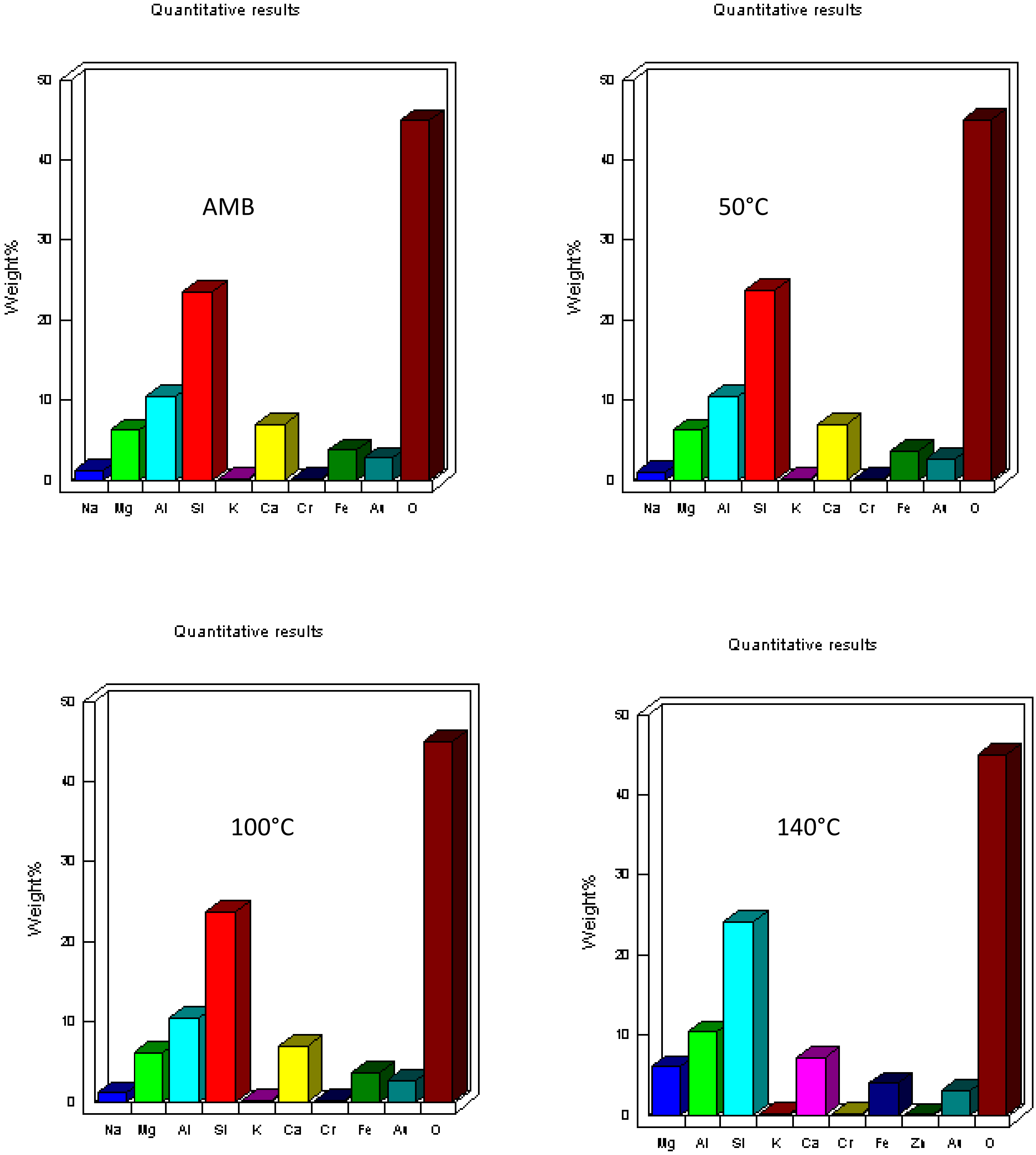

Figure A7 54: Elemental composition of pyroxenite for heat treatments at different temperatures

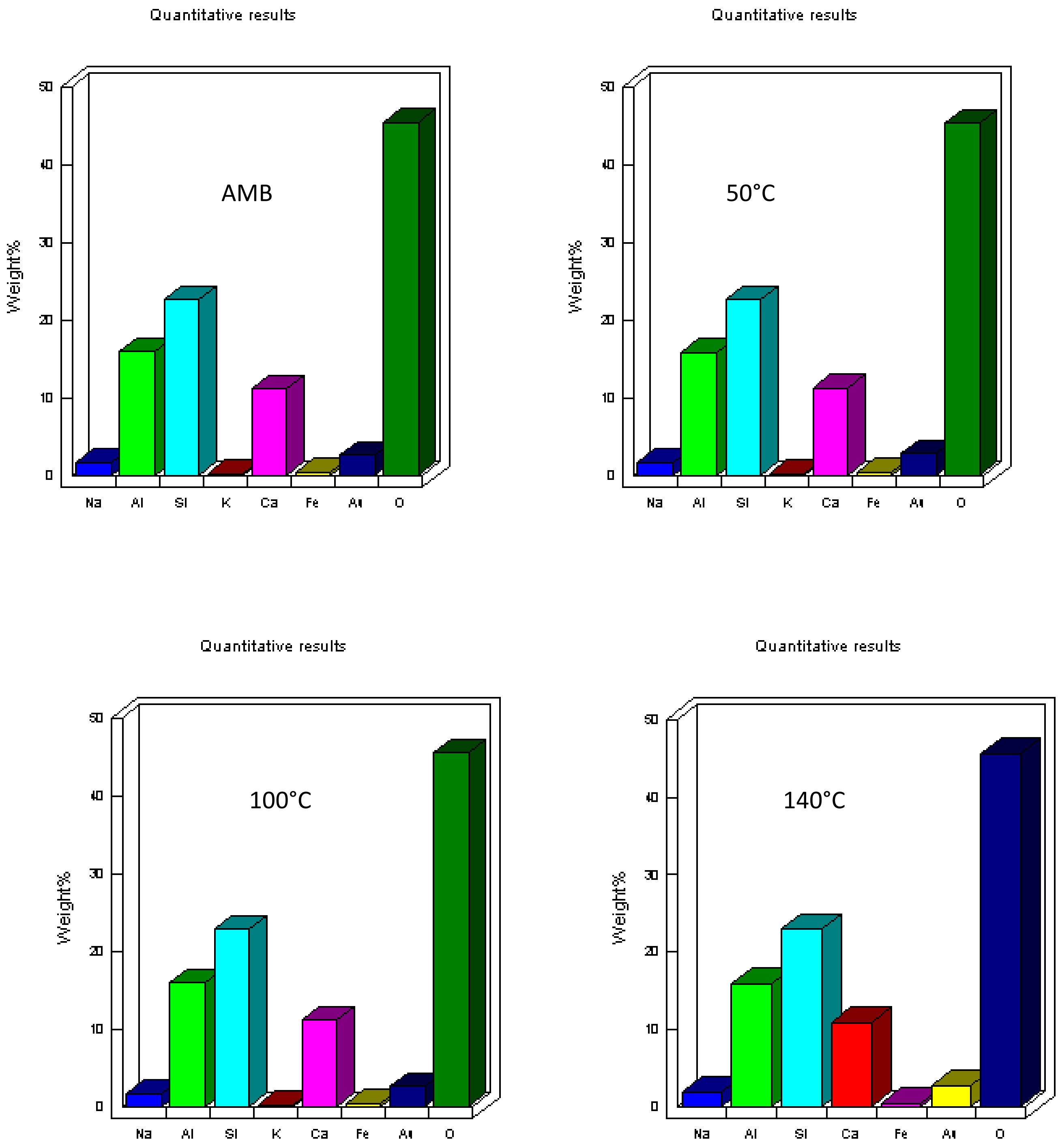

Figure A7 55: Elemental composition of varitextured anorthosite for heat treatments at different temperatures

APPENDIX A 8

The thermal conduction model presented below is taken from FLAC thermal analysis manual (Itasca Consulting Group, 2012).

FLAC Thermal Conduction Model

Temperature and the two components of the heat flux are the variables involved in heat conduction in FLAC. These variables are related through the energy-balance equation and transport laws derived from Fourier's law of heat conduction. Substitution of Fourier's law into the energy-balance equation yields the differential equation of heat conduction, which may be solved for particular geometries and properties, given specific boundary and initial conditions. Thermal volumetric strains are introduced into the incremental mechanical and fluid constitutive laws to account for thermal-stress and thermal-pore pressure couplings (Itasca Consulting Group, 2012).

Energy-Balance Equation

The differential expression of the energy balance has the form

$$-\nabla \cdot \mathbf{q}^T + {q_v}^T = \frac{\partial \zeta T}{\partial t} \qquad \text{(A.1)}$$

where $\mathbf{q}^T$ is the heat-flux vector in [W/m^2],
${q_v}^T$ is the volumetric heat-source intensity in [W/m^3],
and ζT is the heat stored per unit volume in [J/m^3].

Generally, temperature changes may be caused by changes in both energy storage and volumetric strain, ε, and the thermal constitutive law relating those parameters may be written as:

$$\frac{\partial T}{\partial t} = M_T \left(\frac{\partial \zeta T}{\partial t} - \beta_v \frac{\partial \epsilon}{\partial t} \right) \qquad \text{(A.2)}$$

Where M_T and βv are material constants and T is temperature.

In FLAC, a particular case of this law for which $\beta v = 0$ and $M_T = \frac{1}{\rho C_v}$ is considered, where ρ is the mass density of the medium in [kg/m^3], and Cv is the specific heat at constant volume in [J/kg°C]. The hypothesis here is that strain changes play a negligible role in influencing the temperature – a valid assumption for quasi-static mechanical problems involving solids and liquids (Itasca Consulting Group, 2012). Therefore, the equation may be given as:

$$\frac{\partial \zeta T}{\partial t} = \rho C_v \frac{\partial T}{\partial t} \qquad \text{(A.3)}$$

Substitution of Equation A3 in Equation (A.1) yields the energy-balance equation

$$-\nabla \cdot \mathbf{q}^T + q_v{}^T = \rho C_v \frac{\partial T}{\partial t} \qquad \text{(A.4)}$$

Note that for nearly all solids and liquids, the specific heats at constant pressure and at constant volume are essentially equal. Consequently, Cv and Cp can be used interchangeably.

Thermal Analysis

Transport Law

Fourier's law defines the relation between the heat-flux vector and the temperature gradient. For a stationary, homogeneous, isotropic solid, this constitutive law is given in equation A.1

$$\boldsymbol{q}^T = -\boldsymbol{k}^T \nabla T \qquad \text{(A.5)}$$

Where T is the temperature [°C], and k_{ij}^{T} is the thermal conductivity tensor in [W/m°C].

Mechanical Coupling: Thermal Strains

Solution of thermal-stress problems requires reformulation of the stress-strain rate relations, which is accomplished by subtracting from the total strain rate that portion due to temperature change. The shearing strain increments are unaffected because free thermal expansion results in no angular distortion in an isotropic material. The thermal-strain rate associated with the free expansion corresponding to temperature rate ∂T /∂t has the form given in equation A.6

$$\frac{\partial \epsilon_{ij}^{T}}{\partial t} = \alpha_t \frac{\partial T}{\partial t} \delta_{ij} \qquad \text{(A.6)}$$

Where α_t [1/°C] is the coefficient of linear thermal expansion, and δij is the Kronecker delta.

As an example, the mechanical constitutive equations for an elastic material are given in equation A.7

$$\frac{\partial \sigma_{ij}}{\partial t} + \alpha \frac{\partial P}{\partial t} \delta_{ij} = 2G\left(\frac{\partial \epsilon_{ij}}{\partial t} + \alpha_t \frac{\partial T}{\partial t} \delta_{ij}\right) + \left(K - \frac{2}{3}G\right) \cdot \left(\frac{\partial \epsilon_{kk}}{\partial t} + 3\alpha_t \frac{\partial T}{\partial t}\right) \delta_{ij}$$

(5.7)

σ_{ij} and ϵ_{ij} are total stresses and strains, α is Biot coefficient, K and G are bulk and shear

moduli, and δij is the Kronecker delta.

The differential equation of motion and the rate of strain-velocity relations are based upon mechanical considerations, and are unchanged for thermomechanics (Itasca Consulting Group, 2012).

APPENDIX A 9

DATA FILE FOR LABORATORY MODELLING – NORITE – 50°C

```
; Triaxial test of strain-softening material

; with controlled velocity

config  thermal

grid 36,95

gen 0.0,0.0 0.0,0.095 0.036,0.095 0.036,0.0 i=1,37 j=1,96

model ss i=1,36 j=1,95

model th_isotropic

apply temperature 50.0

prop conductivity=2.32 spec_heat=840 thexp=6.56E-6

call servo.fis

fix  y j=1

fix  y j=96

ini yvel -2.5e-6 j 96

ini yvel 2.5e-6 j 1

pro den 3045 bulk 73.2e9 she 29.4e9 co 29.6e6 fric 49.7 ten 11e6 dil 51.5

pro ftab 1 ctab 2 dtab 3

table 1 .001 49 .0015 52 .005 51.5 .03 49.7 .1 10 .1 10 1 10

table 2 .001 32.9e6 .0015 32.2e6 .005 30.5e6 .03 29.6e6 .1 28e6 .1 12e6 1 12e6

table 3 .001 51.5 .0015 40 .005 40 .03 40 .1 5 .1 5 1 5
```

```
app pres 10e6 i 1
app pres 10e6 i 37
ini sxx -10e6 syy -10e6 szz -10e6
def sigmav
  sum=0.0
  loop i (1,igp)
    sum=sum+yforce(i,jgp)
  end_loop
  sigmav=sum/(x(igp,jgp)-x(1,jgp))
end
def ev
  ev=(ydisp(18,1)-ydisp(18,96))/(y(18,96)-y(18,1))
end
hist sigmav
hist ev
hist yv i 1 j 1
hist unbal
history 5 temperature i=20, j=96
set high_unbal=5e6
set low_unbal=2e6
set high_vel=1e-8
step 32000
```

- SERV0.FIS

```
def servo
  while_stepping
  if unbal>high_unbal then
    loop i (1,igp)
      yvel(i,jgp)=yvel(i,jgp)*0.975
      if abs(yvel(i,jgp)) > high_vel then
        yvel(i,jgp) = sgn(yvel(i,jgp))*high_vel
      end_if
      yvel(i,1)=yvel(i,1)*0.975
      if abs(yvel(i,1)) > high_vel then
        yvel(i,1) = sgn(yvel(i,1))*high_vel
      end_if
    end_loop
  end_if
  if unbal<low_unbal then
    loop i (1,igp)
      yvel(i,jgp)=yvel(i,jgp)*1.025
      if abs(yvel(i,jgp)) > high_vel then
        yvel(i,jgp) = sgn(yvel(i,jgp))*high_vel
```

```
      end_if
      yvel(i,1)=yvel(i,1)*1.025
      if abs(yvel(i,1)) > high_vel then
        yvel(i,1) = sgn(yvel(i,1))*high_vel
      end_if
    end_loop
  end_if
end
```

DATA FILE FOR EXCAVATION AT 1073 m BELOW SURFACE

```
; Source: Block: 3x3
config extra=7 thermal
grid 200,61
model mohr
mark i 1 201 j 31
mark i 1 201 j 32
mark i 95 j 30,33
mark i 106 j 30,33
fix  y j 62
fix  y j 1
fix  x i 1
fix  x i 201
;CR-REEF N-fW, Px-HW,
```

```
prop density=3194.06 bulk=5.387E10 shear=3.356E10 cohesion=1.04E7
friction=40.4 dilation=10.1 tension=5.05E6 i= 1,200 j=32,61

prop density=4049.96 bulk=8.783E10 shear=1.387E10 cohesion=6.46E6
friction=35.83 dilation=8.96 tension=1.79E6 i =1,200 j=31

prop density=3045 bulk 5.147e10 she 2.065e10 co 9.8e6 fric 42.07 ten 4.25e6 dil
10.52 i= 1,200 j=1,30

model th_isotropic

prop conductivity=3.65 spec_heat=860.0 thexp=6.84E-6 i= 1,200 j=32,61;PX

model th_isotropic

prop conductivity=2.41 spec_heat=740.0 thexp=6.67E-7 i =1,200 j=31;CR

model th_isotropic

prop conductivity=2.32 spec_heat=840.0 thexp=6.56E-6 i= 1,200 j=1,30;N

set gravity=9.81

set=large

initial temperature=50

fix  temperature 27.0 i 92 111 j 32 34

fix  temperature 27.0 i 92 111 j 29 31

initial sxx = -40.9e6 var 0, 1e6

initial syy = -32.8e6 var 0, 1e6

initial szz = -21.2e6 var 0, 1e6

set mech=off

cycle 100

def convergence

whilestepping

v_converge = ydisp(100,31)-ydisp(100,32)

h_converge = xdisp(96,30.5)-xdisp(105,30.5)

end
```

```
convergence
def _failure
whilestepping
;point1_hor-'sidewall-1'
temp1=-0.5*(sxx(106,30.5)+syy(106,30.5))
temp2=sqrt(sxy(106,30.5)^2+0.25*(sxx(106,30.5)-syy(106,30.5))^2)
h1_sig1=max(temp1+temp2,-szz(106,30.5))
S1=(h1_sig1-h1_sig3)/75.21E6
;point2_hor-'sidewall-2'
temp3=-0.5*(sxx(107,30.5)+syy(107,30.5))
temp4=sqrt(sxy(107,30.5)^2+0.25*(sxx(107,30.5)-syy(107,30.5))^2)
h2_sig1=max(temp3+temp4,-szz(107,30.5))
h2_sig3=min(temp3-temp4,-szz(107,30.5))
S2 =(h2_sig1-h2_sig3)/75.21E6
;point3_hor-'sidewall-3'
temp5=-0.5*(sxx(108,30.5)+syy(108,30.5))
temp6=sqrt(sxy(108,30.5)^2+0.25*(sxx(108,30.5)-syy(108,30.5))^2)
h3_sig1=max(temp5+temp6,-szz(108,30.5))
h3_sig3=min(temp5-temp6,-szz(108,30.5))
S3 =(h3_sig1-h3_sig3)/75.21E6
;point1_ver-'hangingwall-1'100,22))
temp8=sqrt(sxy(100,32)^2+0.25*(sxx(100,32)-syy(100,32))^2)
v1_sig1=max(temp7+temp8,-szz(100,32))
v1_sig3=min(temp7-temp8,-szz(100,32))
H1=(v1_sig1-v1_sig3)/75.21E6
;point2_ver-'hangingwall-2'
```

```
temp9=-0.5*(sxx(100,33)+syy(100,33))

temp10=sqrt(sxy(100,33)^2+0.25*(sxx(100,33)-syy(100,33))^2)

v2_sig1=max(temp9+temp10,-szz(100,33))

v2_sig3=min(temp9-temp10,-szz(100,33))

H2=(v2_sig1-v2_sig3)/75.21E6

;point3_ver-'hangingwall-3'

temp11=-0.5*(sxx(100,34)+syy(100,34))

temp12=sqrt(sxy(100,34)^2+0.25*(sxx(100,34)-syy(100,34))^2)

v3_sig1=max(temp11+temp12,-szz(100,34))

v3_sig3=min(temp11-temp12,-szz(100,34))

H3=(v3_sig1-v3_sig3)/75.21E6

end

 _failure

;Histories

hist v_converge

hist h_converge

hist H1

hist H2

hist H3

hist S1

hist S2

hist S3

hist unbal

set echo off

def zonk

; Extract given region of zones and store existing forces on
```

```
; the boundary of this region in extra arrays 6 and 7.
; INPUT:  i1, i2  bounding i-numbers of region (i2 > i1)
;         j1, j2  bounding j-numbers of region (j2 > j1)
  j2p1 = j2 + 1
  i2p1 = i2 + 1
  loop i (i1,i2p1) ; (record existing fix conditions)
    loop j (j1,j2p1)
      ex_6(i,j) = 0.0
      ex_7(i,j) = 0.0
      if and(flags(i,j),2) # 0 then
        ex_6(i,j) = 1.0
      end_if
      if and(flags(i,j),4) # 0 then
        ex_7(i,j) = 1.0
      end_if
    end_loop
  end_loop
  command ; (allow forces to act on boundary grid-points)
    model null i=i1,i2 j=j1,j2
    fix x y i=i1,i2p1 j=j1,j2p1
    ini xv=0 yv=0 i=i1,i2p1 j=j1,j2p1
    step 1
    free x y i=i1,i2p1 j=j1,j2p1
  end_command
  loop i (i1,i2p1) ; (restore fix conditions)
    loop j (j1,j2p1)
```

```
    if ex_6(i,j) # 0.0 then
      flags(i,j) = or(flags(i,j),2)
    end_if
    if ex_7(i,j) # 0.0 then
      flags(i,j) = or(flags(i,j),4)
    end_if
  end_loop
end_loop
loop ii (i1,i2p1)
  loop jj (j1,j2p1)
    if and(flags(ii,jj),8) = 0 then
      ex_6(ii,jj) = xforce(ii,jj)
      ex_7(ii,jj) = yforce(ii,jj)
      xfa  = -xforce(ii,jj)
      yfa  = -yforce(ii,jj)
      command
        apply xforce=xfa yforce=yfa i=ii j=jj
      end_command
    end_if
  end_loop
end_loop
end
def relax
; Relax forces around newly-mined region gradually.
; INPUT:  n_big_steps   = number of reductions in applied forces
;         n_small_steps = number of FLAC steps within each of the above
```

```
;                              steps
  int n_small_steps n_big_steps
  loop n (1,n_big_steps)
    top    = 1.0 * (n_big_steps - n)
    bot    = 1.0 * n_big_steps
    factor = top / bot
    command
      set echo off
    end_command
    loop i (i1,i2p1)
      loop j (j1,j2p1)
        if and(flags(i,j),8) = 0 then
          xfa  = -ex_6(i,j) * factor
          yfa  = -ex_7(i,j) * factor
          command
            apply xforce=xfa yforce=yfa i=i j=j
          end_command
        end_if
      end_loop
    end_loop
    command
      set echo on
    end_command
    command
      print factor
      step n_small_steps
```

```
    end_command
  end_loop
end
set echo=on
set i1=96 i2=105 j1=31 j2=31
set n_big_steps = 10 n_small_steps = 100
zonk
relax
set mech=on
set thermal=off
solve
```