# Supporting Information for "Modelling lithospheric deformation using a compressible visco-elasto-viscoplastic rheology and the effective viscosity approach"

Thibault Duretz[1,2], René de Borst[3], Philippe Yamato[1,4]

**Contents of this file**

13. Code S5

**Additional Supporting Information (Files uploaded separately)**

1. Caption for Figure S1. Reproduction of previously published results. Extension of a crust with plastic strain softening as in Huismans and Beaumont (2007). The 'symmetric plus mode' and 'asymmetric plug mode' have been reproduced with either V-E-P or V-E-VP rheological models. Strictly symmetric initial and boundary conditions lead to symmetric deformation (A, C). The introduction of non-symmetric initial condition (here, random noise on initial Lagrangian marker locations) was necessary a condition to model asymmetric extension (B, D). The reader is referred to the original publication for model parameters (Huismans and Beaumont, 2007).

2. Caption for Figure S2. Effect of the viscoplastic regularisation viscosity on the style of extension. A too large viscosity leads to large overstress and prevents from the occurrence of asymmetric extension (A). The 'Asymmetric plug mode' extension is recovered for sufficiently small regularisation viscosity. Models were all run with initial random noise. The reader is referred to the original publication for model parameters (Huismans and Beaumont, 2007).

3. Caption for Figure S3. Magnitudes of viscoplastic overstress during extension. Panel a) shows a model with a linear viscoplastic rheology ($n^{\mathrm{vp}} = 1.0$) and reference overstress of $2 \times 10^5$ Pa. The local magnitude of overstress is larger than 30 MPa and asymmetric extension does not take place (panel A). For a similar reference overstress but a non-linear viscoplastic model ($n^{\mathrm{vp}} = 1.5$), the local magnitude of overstress is lower than 20 MPa and asymmetric extension is recovered (panel B). Alternatively, asymmetric extension can

also be recovered with a linear viscoplastic model if the reference overstress is sufficiently decreased (panel C), leading to a reduction of the local value of overstress ($< 30$ MPa).

4. Caption for Figure S4. This flowchart describes the implementation of a non-linear Visco-Elasto-Viscoplastic code. Implementation details of local iterations routines are documented in the main text and made available in the appended Code

5. Caption for Figure S5. A comparison of lithospheric models in extension and compression. Panels A) and D) correspond to model that are fully converged and use a V-E-VP rheology. Panels B) and E) correspond to V-E-VP models in which only one global iteration was used. Panels B) and E) correspond to V-E-P model in which only one global iteration was used. The black lines indicate the location of the Moho. The white line is the 1.8 accumulated strain contour.

6. Caption for Figure S6. An illustration of the checkerboard-style shear banding patterns obtained for low reference overstress values. These models were run with M2Di and are based on those presented in Fig. 3 of the main article file. Model results are depicted after 100 steps of $\Delta t = 4 \times 10^{11}$ s.

7. Caption for Figure S7. Example of results of 0D V-E-VP calculations using the appended Julia language script. The upper panels correspond to the evolution of the second deviatoric stress invariant, the middle panel corresponds the pressure and the lower panels shows the norm of the difference between the consistent tangent operators derived using the effective viscosity approach (Appendix A) and the finite-step consistent tangent linearisation (Appendix B). The two cases (A ,B) can reproduced by changing the value of the TrialModel and PlastModel variables

8. Effect of different linearisation schemes on non-linear convergence using the test case Test 1 of Duretz et al. (2018). Panel A shows the convergence behaviour of momentum equation for Picard, Continuum tangent, Consistent tangent (Cons. Tangent) and Newton linearisaton of the effective viscosity approach (EVA Newton) that are derived in the Appendix. Picard linearisation necessitates several hundreds of iterations to converge this step. EVA Newton and Cons. Tangent have a similar behaviour. Panel B shows the total number of iterations required for each linearisation scheme. Panel C shows the evolution of the second deviatoric stress invariant - the different approaches yield similar results. For all cases, a line search procedure was used to optimize the convergence of global iterations (e.g., Duretz et al, 2018).

9. Effect of global momentum iterations on the results of numerical models of shear banding using Test 1 of Duretz et al. (2018). Left panels shows shear banding pattern after 34 steps for converged (upper) and non-converged (bottom) cases. Similarly, the right panels correspond to Test 1 including viscoplastic regularisation ($\eta^{\mathrm{vp}} = 4 \times 10^{18}$ Pa.s). The colormaps correspond to the pressure field in MPa and the black lines are isocontour of accumulated strain ($\epsilon_{\mathrm{acc}} = 10^{-2.6}$). Models that reached machine precision force equilibrium (lower panels) are characterised depict more intense strain localisation thus a longer extent of shear bands at a given time. Model results were depicted at time steps number 34.

10. Caption for Figure S10. Effect of bulk elasticity and plastic dilatancy on shear banding based on Test 1 of Duretz et al. (2018). The upper left panel correspond to the reference model (V-E-P). The simulation presented in the upper right panel includes viscoplastic regularisation (V-E-VP, $\eta^{\mathrm{vp}} = 4 \times 10^{18}$ Pa.s). The lower left panel accounts for

zero dilatancy (plastic incompressibility, $\psi = 0$). The lower right is strictly incompressible ($\psi = 0$ and $K \rightarrow$ inf), hence both effects of bulk elasticity and plastic dilatancy are neglected. The colormaps correspond to the pressure field in MPa and the black lines are isocontour of accumulated strain ($\epsilon_{\text{acc}} = 10^{-2.6}$). It is clear that both elastic and plastic incompressibility promote strain localisation. Model results were depicted at time steps number 34.

11. Caption for Figure S11. Effect of bulk elasticity and plastic dilatancy on the global convergence behaviour. Simulations were based on Test 1 of Duretz et al. (2018). The upper panel (A) shows the convergence behaviour of the V-E-P (reference), V-E-VP, V-E-VP with zero dilatancy and V-E-VP incompressible at time steps number 34. The middle panel (B) shows the number of iteration (Newton EVA) needed to fully converge each step. The incompressible case can lead to large iteration counts (step 26 and 34) has also seen in panel A. The lower panel (C) shows the evolution of the mean second deviatoric stress invariant for the 4 models. Models that include bulk elasticity and dilatancy exhibit a slight apparent hardening (V-E-P and V-E-VP) during shear banding. The incompressible model (V-E-VP incomp.) exhibit no hardening and the model with bulk elasticity and no dilatancy (V-E-VP - $\psi = 0$) exhibit a slight apparent softening

12. Caption for Code S1. This script allows to compute a Visco-Elasto-Viscoplastic stress loading. The two local iterative procedures described in the main article file are implemented. Both linear and non-linear rheological models are available for both the predictor (visco-elastic) and corrector (viscoplastic). The tangent operators based either on the Newton linearisation of the effective viscosity approach (Appendix A) or the finite-

step consistent tangent linearisation (Appendix B) are also evaluated. This code is written in Julia language.

**Introduction** This files contains the following informations: (1) the results of benchmark test against previously published results (Figure S1, S2, S3), (2) details regarding the implementation of the rheological models described in the main text (Figure S4 and Code S5), (3) a comparison of V-E-VP and V-E-P using a lithospheric deformation model, (4) an illustration of the checkerboard pattern that can be obtained at very low value of reference overstress.

**Figure S1.**

**Figure S2.**

**Figure S3.**

**Figure S4.**

**Figure S5.**

**Figure S6.**

**Figure S7.**

**Figure S8.**

**Figure S9.**

**Figure S10.**

**Figure S11.**

## Code S1.

```
# This is written in Julia language
iimport Plots
using LinearAlgebra
global const R = 8.3145 # Gas constant

######### Local rheological procedure: non-linear visco-elastic predictor #########
function LocalIterationsTrial(Eiid,etae,A,n,Q,T)
    # Setup initial guess
    eta_ve = etae
    eta0   = A^(-1/n)*exp(Q/n/R/T) # Pre-exponential term, here I've included Aarhenius term. In practice, all other guys should appear there (grain size, water fugacity, experimental correction term, etc.
    Cv     = (2*eta0)^(-n)
    # Newton iterations
    for iter=1:100
        Tii     = 2*eta_ve*Eiid
        Eiidv   = Cv*Tii^n
        f       = Eiid - Tii/2/etae - Eiidv
```

```
        dfdeta  =-Eiid/etae - n*Eiidv/eta_ve
        eta_ve -= f/dfdeta
        println("Visco-elastic trial: ", iter, " --- f = ", abs(f)/Eiid)
        if abs(f)/Eiid<1e-13; break; end
    end
    return eta_ve
end

######### Local rheological procedure: non-linear viscoplastic corrector #########
function LocalIterationsPlast(Eiidd,eta_ve,eta_vp0,n_vp,phi,psi,K,C,F_trial,P,Tii,dt)
    # Setup initial guess
    Tiic   = Tii
    Pc     = P
    F      = F_trial
    lamdot = F_trial/(eta_ve + K*dt*sin(psi)*sin(phi) + eta_vp0*Eiidd^(1/n_vp-1))
    eta_vp = eta_vp0*abs(lamdot)^(1/n_vp-1)
    # Newton iterations
    for iter=1:100
        eta_vp  = eta_vp0*abs(lamdot)^(1/n_vp-1)
        Pc      = P   + K*dt*lamdot*sin(psi)
        Tiic    = Tii - eta_ve*lamdot
        F       = Tiic - Pc*sin(phi) - C*cos(phi) - eta_vp*lamdot
        dFdlam  =-K*dt*sin(psi)*sin(phi) - eta_vp/n_vp - eta_ve
        lamdot -= F/dFdlam
        println("Viscoplastic  corr.: ", iter, " --- F = ", F/F_trial)
        if abs(F)/F_trial<1e-13; break; end
    end
    return lamdot, Tiic, Pc, F, eta_vp
end

function PartialDerivativeTrialViscosity(Txx,Tyy,Tzz,Txy,Tii,Exxd,Eyyd,Ezzd,Exyd,Eiid,eta_ve,A,Q,n,R,T)
    # 1.0: Visco-elastic trial
    Jii      = Tii^2;
    eta0     = A^(-1/n)*exp(Q/n/R/T) # Pre-exponential term, here I've included Aarhenius term. In practice, all other guys should appear there (grain size, water fugacity, experimental correction term, et
    Cv       = (2*eta0)^(-n)
    detadTxx     = -1.0*Cv*Jii.^(n/2 - 3/2)*Txx.*eta_ve^2*(n - 1)
    detadTyy     = -1.0*Cv*Jii.^(n/2 - 3/2)*Tyy.*eta_ve^2*(n - 1)
    detadTzz     = -1.0*Cv*Jii.^(n/2 - 3/2)*Tzz.*eta_ve^2*(n - 1)
    detadTxy     = -2.0*Cv*Jii.^(n/2 - 3/2)*Txy.*eta_ve^2*(n - 1)
    deta_ve_dExx = detadTxx * 2*eta_ve / (1 - 2*(detadTxx.*Exxd + detadTyy.*Eyyd + detadTxy.*Exyd + detadTzz.*Ezzd));
    deta_ve_dEyy = detadTyy * 2*eta_ve / (1 - 2*(detadTxx.*Exxd + detadTyy.*Eyyd + detadTxy.*Exyd + detadTzz.*Ezzd));
    deta_ve_dExy = detadTxy * 2*eta_ve / (1 - 2*(detadTxx.*Exxd + detadTyy.*Eyyd + detadTxy.*Exyd + detadTzz.*Ezzd));
    deta_ve_dEzz = detadTzz * 2*eta_ve / (1 - 2*(detadTxx.*Exxd + detadTyy.*Eyyd + detadTxy.*Exyd + detadTzz.*Ezzd));
    deta_ve_dP   = zeros(size(Txx));
    dDvedE    = [deta_ve_dExx; deta_ve_dEyy; deta_ve_dEzz; deta_ve_dExy; 0];
    return dDvedE
end

######### Consistent tangent: effective viscosity formulation after appendix A #########
function ConsistentTangentEVA(Txx,Tyy,Tzz,Txy,Tii,Exxd,Eyyd,Ezzd,Exyd,Eiid,eta_ve,dDvedE,eta_vp,n_vp,phi,psi,K,dt,lamdot,plastic)
    eta      = Tii/2/Eiid
    Jii      = Tii^2;
    deta_ve_dExx = dDvedE[1]; deta_ve_dEyy = dDvedE[2]; deta_ve_dEzz = dDvedE[3]; deta_ve_dExy = dDvedE[4];
    # 2.0: Viscoplastic corrector
    if plastic == 1
        # dFdE
        dFdExx        =   Exxd.*eta_ve./Eiid + 2*Eiid.*deta_ve_dExx
        dFdEyy        =   Eyyd.*eta_ve./Eiid + 2*Eiid.*deta_ve_dEyy
        dFdExy        = 2*Exyd.*eta_ve./Eiid + 2*Eiid.*deta_ve_dExy
        dFdEzz        =   Ezzd.*eta_ve./Eiid + 2*Eiid.*deta_ve_dEzz
        dFdP          = -sin(phi);
        #  dlamdotdE
        g             =  1.0 / (eta_ve + eta_vp./n_vp + K.*dt.*sin(psi).*sin(phi) );
        dlamdExx      = g .* (dFdExx - lamdot .*deta_ve_dExx);
        dlamdEyy      = g .* (dFdEyy - lamdot .*deta_ve_dEyy);
        dlamdExy      = g .* (dFdExy - lamdot .*deta_ve_dExy);
        dlamdEzz      = g .* (dFdEzz - lamdot .*deta_ve_dEzz);
        dlamdP        = g .* (dFdP                          );
        # deta_vp_dE
        deta_vp_dExx  = eta_vp./lamdot.*dlamdExx.*(1.0/n_vp-1);
        deta_vp_dEyy  = eta_vp./lamdot.*dlamdEyy.*(1.0/n_vp-1);
        deta_vp_dExy  = eta_vp./lamdot.*dlamdExy.*(1.0/n_vp-1);
        deta_vp_dEzz  = eta_vp./lamdot.*dlamdEzz.*(1.0/n_vp-1);
        deta_vp_dP    = eta_vp./lamdot.*dlamdP  .*(1.0/n_vp-1);
        # deta_vep_dE
        a             =  eta_vp + K.*dt.*sin(psi).*sin(phi);
        Tyield        =  Tii
        deta_vep_dExx = -Exxd.*Tyield./(4*Eiid.^3) + (a.*dlamdExx + lamdot.*deta_vp_dExx)./(2*Eiid);
        deta_vep_dEyy = -Eyyd.*Tyield./(4*Eiid.^3) + (a.*dlamdEyy + lamdot.*deta_vp_dEyy)./(2*Eiid);
        deta_vep_dExy = -Exyd.*Tyield./(2*Eiid.^3) + (a.*dlamdExy + lamdot.*deta_vp_dExy)./(2*Eiid);
        deta_vep_dEzz = -Ezzd.*Tyield./(4*Eiid.^3) + (a.*dlamdEzz + lamdot.*deta_vp_dEzz)./(2*Eiid);
        deta_vep_dP   =                  (sin(phi) +  a.*dlamdP   + lamdot.*deta_vp_dP  )./(2*Eiid);
        # ddivp
        dQdP       = -sin(psi)
        ddivp_dExx = -dQdP.*dlamdExx
        ddivp_dEyy = -dQdP.*dlamdEyy
        ddivp_dExy = -dQdP.*dlamdExy
        ddivp_dEzz = -dQdP.*dlamdEzz
        ddivp_dP   = -dQdP.*dlamdP
    else
        # Assume default values for VE case
        deta_vep_dExx, deta_vep_dEyy, deta_vep_dEzz, deta_vep_dExy, deta_vep_dP = deta_ve_dExx, deta_ve_dEyy, deta_ve_dEzz, deta_ve_dExy, 0.0
        ddivp_dExx, ddivp_dEyy, ddivp_dEzz, ddivp_dExy, ddivp_dP                = 0.0, 0.0, 0.0, 0.0, 0.0
    end

    dP_ddiv       = -K*dt
    deta_vep_ddiv = deta_vep_dP * dP_ddiv
    ddivp_ddiv    = ddivp_dP    * dP_ddiv

    #  ACHTUNG: here we use engineering strain rate:
    Gxyd = 2*Exyd
    #  Therefore the 4th line is divided by 2
    D_eva  = [2*eta+2*deta_vep_dExx*Exxd        2*deta_vep_dEyy*Exxd        2*deta_vep_dEzz*Exxd        2*deta_vep_dExy*Exxd        2*deta_vep_ddiv*Exxd;
                  2*deta_vep_dExx*Eyyd  2*eta+2*deta_vep_dEyy*Eyyd        2*deta_vep_dEzz*Eyyd        2*deta_vep_dExy*Eyyd        2*deta_vep_ddiv*Eyyd;
                  2*deta_vep_dExx*Ezzd        2*deta_vep_dEyy*Ezzd  2*eta+2*deta_vep_dEzz*Ezzd        2*deta_vep_dExy*Ezzd        2*deta_vep_ddiv*Ezzd;
                  1*deta_vep_dExx*Gxyd        1*deta_vep_dEyy*Gxyd        1*deta_vep_dEzz*Gxyd  1*eta+1*deta_vep_dExy*Gxyd        1*deta_vep_ddiv*Gxyd;
```

```
                K*dt*ddivp_dExx            K*dt*ddivp_dEyy            K*dt*ddivp_dEzz            K*dt*ddivp_dExy           -K*dt*(1.0-ddivp_ddiv);]

    return D_eva
end

######### Consistent tangent: finite-step formulation after appendix B #########
function ConsistentTangentCTL(Txx,Tyy,Tzz,Txy,Tii,Exxd,Eyyd,Ezzd,Exyd,Eiid,eta_ve,dDvedE,eta_vp,n_vp,phi,psi,K,dt,lamdot,plastic)
    # First derivatives of plastic flow potential
    dFdt = [0.5*Txx./Tii; 0.5*Tyy./Tii; 0.5*Tzz./Tii; Txy./Tii; -sin(phi);]
        # First derivatives of yield function
    dQdt = [0.5*Txx./Tii; 0.5*Tyy./Tii; 0.5*Tzz./Tii; Txy./Tii;  sin(psi);]
    # Second derivatives of plastic flow potential
    I3     = Diagonal( [1/2; 1/2; 1/2; 1; 0] )
    I4     = Diagonal( [1; 1; 1; 1; 0] )
    d2Qdt2 = 1/Tii * ( I3 .- I4*(dQdt*dQdt')*I4)

    I = [ 1.0 0.0 0.0 0.0 0.0;
          0.0 1.0 0.0 0.0 0.0;
          0.0 0.0 1.0 0.0 0.0;
          0.0 0.0 0.0 1.0 0.0;
          0.0 0.0 0.0 0.0 1.0]

    #  ACHTUNG: here we use engineering strain rate: Gxy = 2*Exy
    #  Therefore the 4th line is divided by 2
    Dve     = [2*eta_ve 0        0        0          0;
               0        2*eta_ve 0        0          0;
               0        0        2*eta_ve 0          0;
               0        0        0        1*eta_ve   0;
               0        0        0        0        -K*dt]

    # E matrix and its inverse
    if plastic == 1
        E = I + lamdot*Dve*d2Qdt2
    else
        lamdot = 0.0
        E = I
    end
    Einv      = inv(E)
    # B Matrix, #  ACHTUNG: here we use engineering strain rate: Gxy = 2*Exy
    Ev        = [Exxd; Eyyd; Ezzd; 2*Exyd; 0];
    Ep        = lamdot.*dQdt
    B         = 2*dDvedE*([1;1;1;1;0].*(Ev.-Ep))'
    # Denominator of the consistent tangent
    dFdlamdot = -eta_vp/n_vp  # Here other terms will appear if softening is included
    den       = -dFdlamdot + dFdt'*Einv*Dve*dQdt
    # Consistent tangent
    D_ctl     = Einv*B .+ Einv*Dve*( I - plastic*dQdt*dFdt'*Einv*(B .+ Dve)/den )
    return D_ctl
end

######### MAIN #########
@views function main()
    # Choose model type
    TrialModel = "LinearViscoElasticity"
    PlastModel = "LinearViscoPlasticity"
    # TrialModel = "NonLinearViscoElasticity"
    # PlastModel = "NonLinearViscoPlasticity"
    # Material parameters
    # a. Elasticity
    G       = 3e10       # Shear modulus [Pa]
    K       = 2e10       # Bulk modulus [Pa]
    # b. Viscosity
    etav    = 1e23       # Viscosity used for "LinearViscoElasticity" [Pa.s]
    A       = 3.9811e-16 # Material parameter used for "NonLinearViscoElasticity" --- eta0 = A^(-1/n)
    Q       = 356e3      # Material parameter used for "NonLinearViscoElasticity"
    n       = 3.0
    # c. Viscoplasticity
    C       = 40e6       # Cohesion [Pa]
    phi     = 30*pi/180  # Friction angle
    psi     = 5*pi/180   # Dilation angle
    dS      = 1e6        # Reference overstress [Pa]
    n_vp    = 2.0        # Power-law viscoplasticity exponent
    # Loading conditions
    # a. Total strain rate tensor components
    Ebg     = 1e-15           # Reference strain rate
    exx     = Ebg;            # Normal strain rate [1/s]
    eyy     =-1.1*Ebg         # Normal strain rate [1/s] - introduce some divergence
    ezz     = 0.0             # Plane strain constraint
    exy     = 0.0             # Shear strain rate [1/s]
    div     = exx + eyy + ezz # Divergence  [1/s]
    T       = 773             # Temperature [K]
    # Security
    if TrialModel=="LinearViscoElasticity"
        n = 1.0
        A = 1/etav
    end
    if PlastModel=="LinearViscoPlasticity"
        n_vp = 1.0
    end
    # b. Deviatoric strain rate tensor components
    exxd    = exx - 1/3*div
    eyyd    = eyy - 1/3*div
    ezzd    = ezz - 1/3*div
    exyd    = exy
    # Numerical parameters
    nt      = 20
    dt      = 4*1e11            # Time step [s]
    # Storage for visualisation
    Tii_tab = zeros(nt)
    P_tab   = zeros(nt)
    diff    = zeros(nt)
    # Constant (only if dt is constant!)
    etae    = G*dt
    # Compute viscoplastic viscosity for a given reference overstress and strain rate
    eta_vp0 = dS/(abs(Ebg)^(1/n_vp))
    eta_vp  = eta_vp0
```

```
    # Initial condition: Deviatoric stress tensor components and pressure
    Txx, Tyy, Tzz, Txy, P  = 0.0, 0.0, 0.0, 0.0, 200e6
    # Time Loop
    for it=1:nt
        println("##### Time step: ", it, " #####")
        # Previous step
        Txx0, Tyy0, Tzz0, Txy0, P0  = Txx, Tyy, Tzz, Txy, P
        plastic, lamdot             = 0.0, 0.0
        # Effective strain rate
        Exxd   = exxd + Txx0/2/etae
        Eyyd   = eyyd + Tyy0/2/etae
        Ezzd   = ezzd + Tzz0/2/etae
        Exyd   = exyd + Txy0/2/etae
        Gxyd   = 2*exyd + Txy0/1/etae
        Eiid   = sqrt(1/2*(Exxd^2+Eyyd^2+Ezzd^2) + Exyd)
        # TRIAL STATE
        # a. Compute effective viscosity
        if TrialModel=="LinearViscoElasticity"   ; eta_ve = 1/(1/etae + 1/etav); end
        if TrialModel=="NonLinearViscoElasticity"; eta_ve = LocalIterationsTrial(Eiid,etae,A,n,Q,T); end
        # b. Deviatoric stress
        Txx     = 2*eta_ve*Exxd
        Tyy     = 2*eta_ve*Eyyd
        Tzz     = 2*eta_ve*Ezzd
        Txy     = 2*eta_ve*Exyd
        Tii     = sqrt(1/2*(Txx^2 + Tyy^2 + Tzz^2) + Txy^2)
        # c. Pressure
        P       = P0 - K*dt*div
        # PLASTIC CORRECTION
        # a. Check yield
        Pt      = P
        F_trial = Tii - P*sin(phi) - C*cos(phi)
        # b. Correct
        if F_trial >= 0
            Tiic, Pc, lamdot, plastic      = Tii, P, 0.0, 1.0
            dQdTxx, dQdTyy, dQdTzz, dQdTxy = Txx/2/Tii, Tyy/2/Tii, Tzz/2/Tii, Txy/Tii
            if PlastModel=="LinearViscoPlasticity"
                lamdot = F_trial/(eta_ve + K*dt*sin(psi)*sin(phi) + eta_vp0)
                Pc     = P    + K*dt*lamdot*sin(psi)
                Tiic   = Tii  - eta_ve*lamdot
                eta_vp = eta_vp0
                F      = Tiic - Pc*sin(phi) - C*cos(phi) - eta_vp*lamdot
            elseif PlastModel=="NonLinearViscoPlasticity"
                lamdot, Tiic, Pc, F, eta_vp = LocalIterationsPlast(Eiid,eta_ve,eta_vp0,n_vp,phi,psi,K,C,F_trial,P,Tii,dt)
            end
            println(">>> Plastic: Trial F = ", F_trial, " --> Corrected F: ", F)
            # Update stresses
            Txx    = 2*eta_ve*(Exxd - lamdot*dQdTxx  )
            Tyy    = 2*eta_ve*(Eyyd - lamdot*dQdTyy  )
            Tzz    = 2*eta_ve*(Ezzd - lamdot*dQdTzz  )
            Txy    = 2*eta_ve*(Exyd - lamdot*dQdTxy/2)
            Tii    = sqrt(1/2*(Txx^2 + Tyy^2 + Tzz^2) + Txy^2)
            P      = Pc
        end

        # CONSISTENT TANGENT OPERATORS

        ## Partial derivatives of VE trial viscosity (necessary for both linearizations)
        dDvedE = PartialDerivativeTrialViscosity(Txx,Tyy,Tzz,Txy,Tii,Exxd,Eyyd,Ezzd,Exyd,Eiid,eta_ve,A,Q,n,R,T)

        ## Assembly of the tangent operator following Appendix A
        D_eva = ConsistentTangentEVA(Txx,Tyy,Tzz,Txy,Tii,Exxd,Eyyd,Ezzd,Exyd,Eiid,eta_ve,dDvedE,eta_vp,n_vp,phi,psi,K,dt,lamdot,plastic)

        ## Assembly of the tangent operator following Appendix B
        D_ctl = ConsistentTangentCTL(Txx,Tyy,Tzz,Txy,Tii,Exxd,Eyyd,Ezzd,Exyd,Eiid,eta_ve,dDvedE,eta_vp,n_vp,phi,psi,K,dt,lamdot,plastic)

        println("D_eva:")
        [@show  D_eva[i,:] for i=1:5]
        println("D_ctl:")
        [@show  D_ctl[i,:] for i=1:5]
        println("D diff:")
        D_diff = (D_ctl .- D_eva)./norm(D_eva)
        [@show  D_diff[i,:] for i=1:5]

        # Store
        Tii_tab[it] = Tii
        P_tab[it]   = P
        diff[it]    = norm(D_diff)
    end
    # Visualise
    p1 = Plots.plot((1:nt)*dt/(1e3*365.25*24*3600), Tii_tab/1e6, legend=:none, title="Dev. stress invariant II", ylabel="Tau [MPa]")
    p2 = Plots.plot((1:nt)*dt/(1e3*365.25*24*3600),   P_tab/1e6, legend=:none, title="Pressure", ylabel="P [MPa]")
    p3 = Plots.plot((1:nt)*dt/(1e3*365.25*24*3600),        diff, legend=:none, title="Difference between tangent operators", xlabel="t [ky]")
    display(Plots.plot(p1,p2,p3, layout=(3,1) ))
end

main()
```

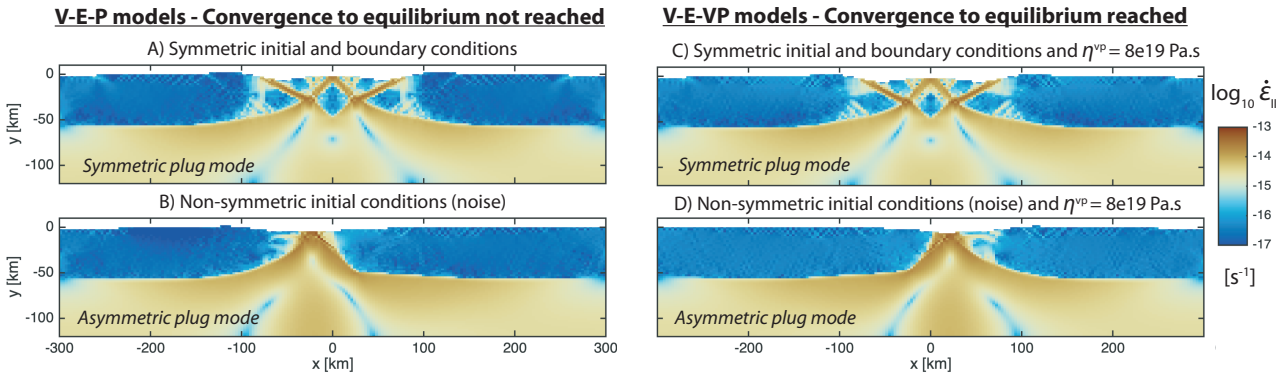


**Figure S1.** Reproduction of previously published results. Extension of a crust with plastic strain softening as in Huismans and Beaumont (2007). The 'symmetric plus mode' and 'asymmetric plug mode' have been reproduced with either V-E-P or V-E-VP rheological models. Strictly symmetric initial and boundary conditions lead to symmetric deformation (A, C). The introduction of non-symmetric initial condition (here, random noise on initial Lagrangian marker locations) was necessary a condition to model asymmetric extension (B, D). The reader is referred to the original publication for model parameters (Huismans and Beaumont, 2007). The numerical resolution was set to $200 \times 60$ cells and the model was run during 300 steps with a constant $\Delta t = 5 \times 10^{11}$ s.

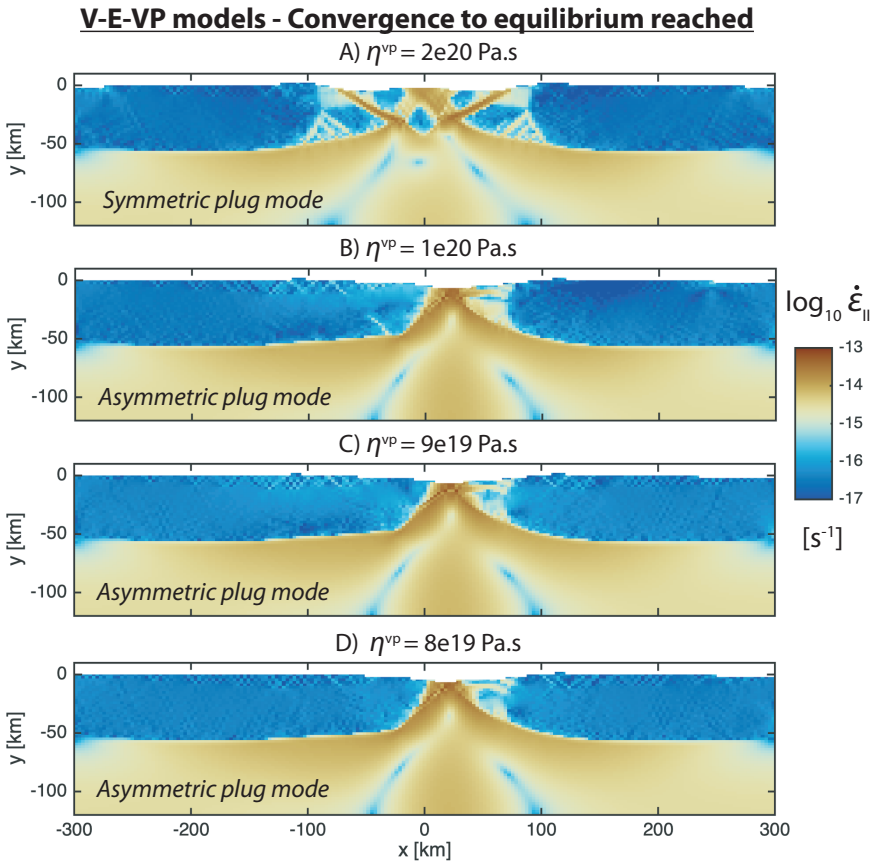


**Figure S2.** Effect of the viscoplastic regularisation viscosity on the style of extension. A too large viscosity leads to large overstress and prevents from the occurrence of asymmetric extension (A). The 'Asymmetric plug mode' extension is recovered for sufficiently small regularisation viscosity. Models were all run with initial random noise. The reader is referred to the original publication for model parameters (Huismans and Beaumont, 2007).

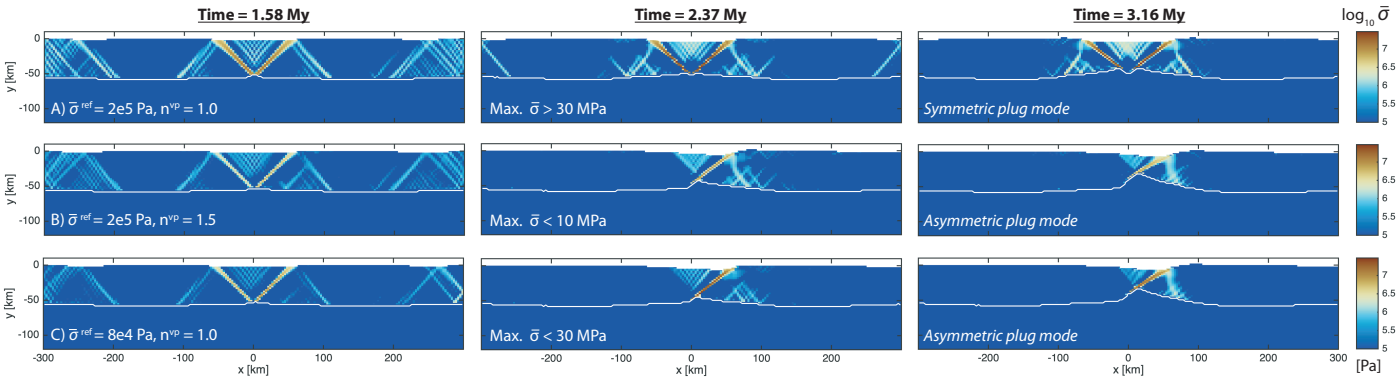


**Figure S3.** Magnitudes of viscoplastic overstress during extension. Panel a) shows a model with a linear viscoplastic rheology ($n^{\mathrm{VP}} = 1.0$) and reference overstress of $2 \times 10^5$ Pa. The local magnitude of overstress is larger than 30 MPa and asymmetric extension does not take place (panel A). For a similar reference overstress but a non-linear viscoplastic model ($n^{\mathrm{VP}} = 1.5$), the local magnitude of overstress is lower than 20 MPa and asymmetric extension is recovered (panel B). Alternatively, asymmetric extension can also be recovered with a linear viscoplastic model if the reference overstress is sufficiently decreased (panel C), leading to a reduction of the local value of overstress ($< 30$ MPa).

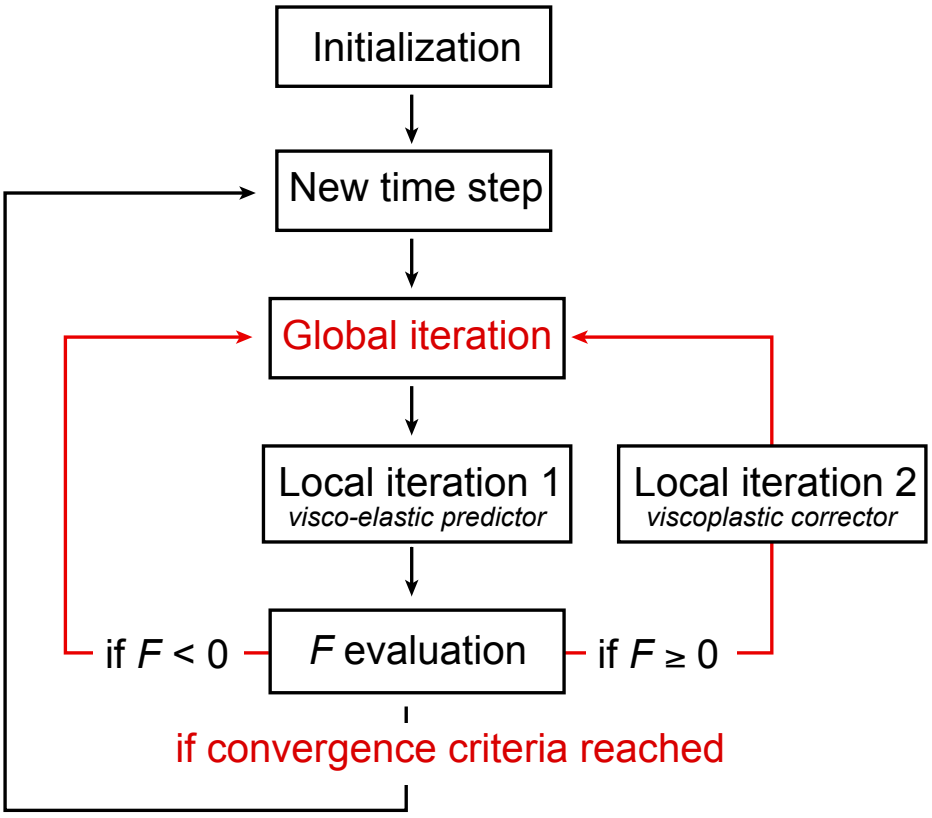


**Figure S4.** This flowchart describes the implementation of a non-linear Visco-Elasto-Viscoplastic code. Implementation details of local iterations routines are documented in the main text and made available in the appended Code.

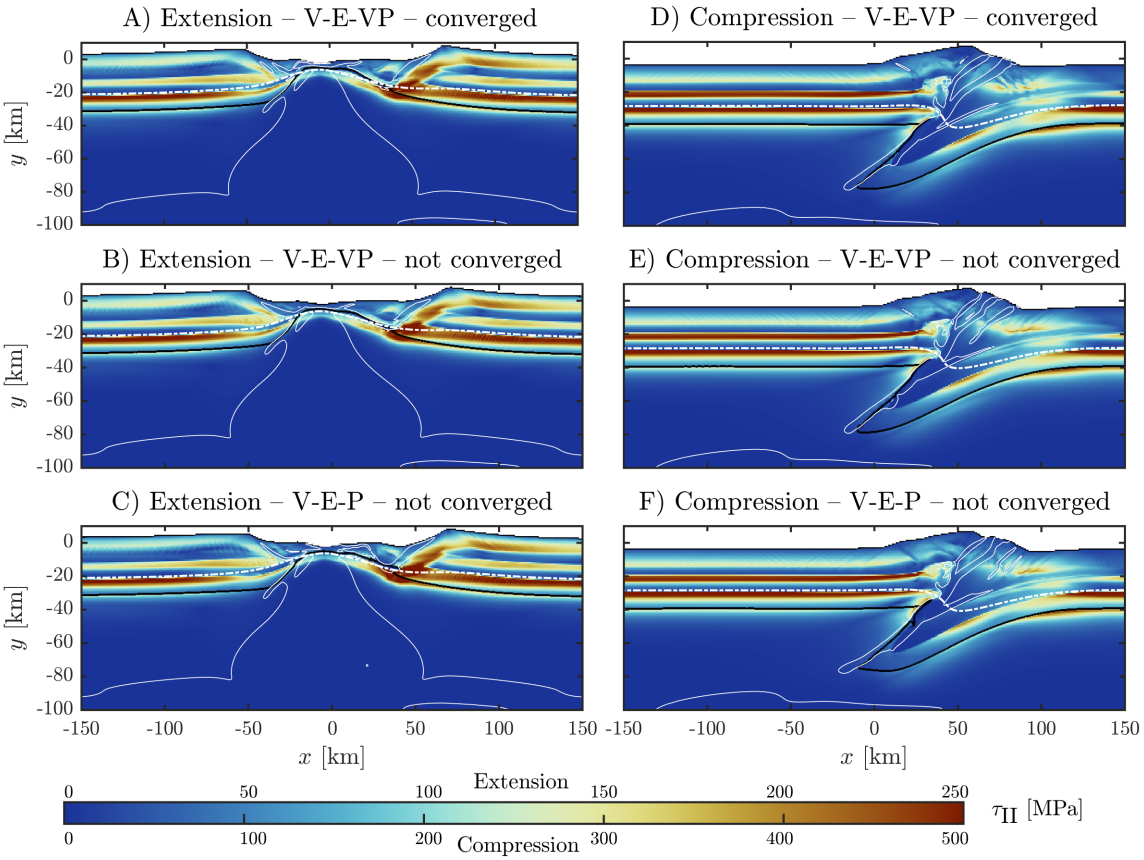


**Figure S5.** A comparison of lithospheric models in extension and compression. Panels A) and D) correspond to model that are fully converged and use a V-E-VP rheology. Panels B) and E) correspond to V-E-VP models in which only one global iteration was used. Panels B) and E) correspond to V-E-P model in which only one global iteration was used. The black lines indicate the location of the Moho. The white line is the 1.8 accumulated strain contour.

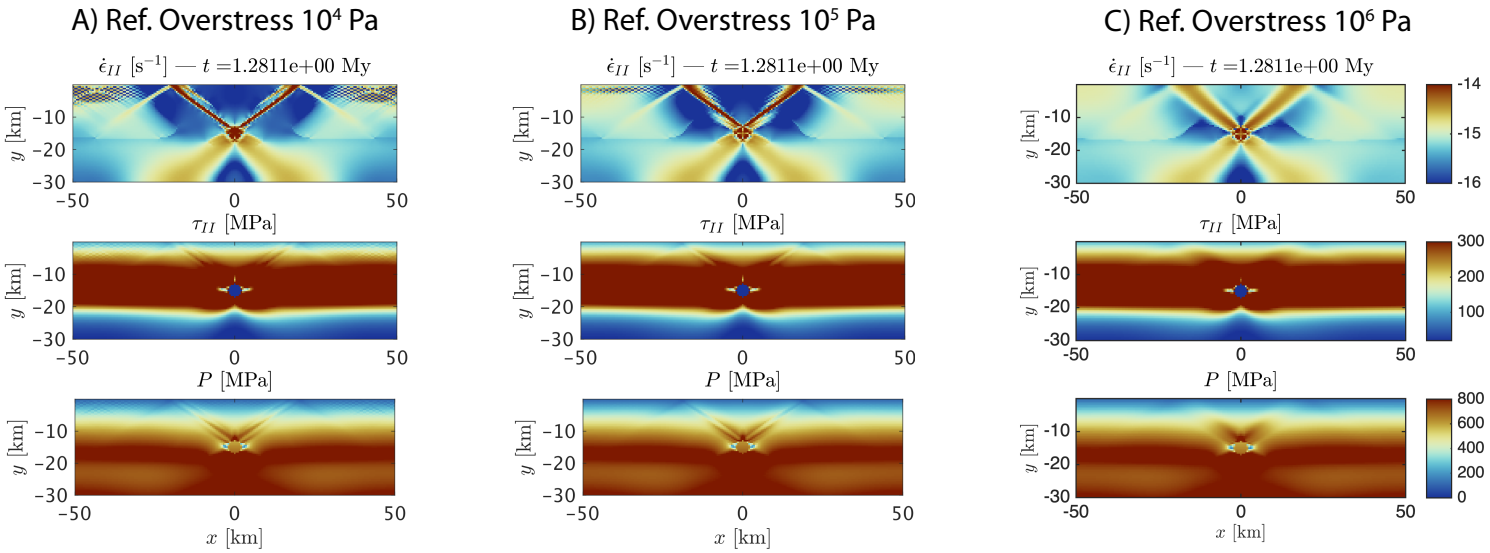


**Figure S6.** An illustration of the checkerboard-style shear banding patterns obtained for low reference overstress values. These models were run with M2Di and are based on those presented in Fig. 3 of the main article file. Model results are depicted after 100 steps of $\Delta t = 4 \times 10^{11}$ s.

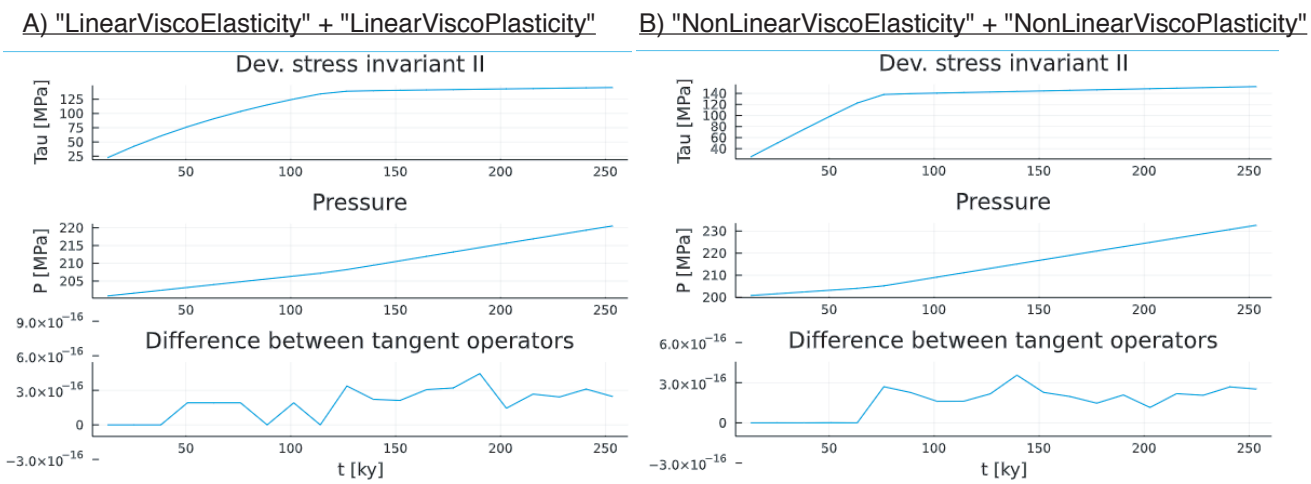


**Figure S7.** Example of results of 0D V-E-VP calculations using the appended Julia language script. The upper panels correspond to the evolution of the second deviatoric stress invariant, the middle panel corresponds the pressure and the lower panels shows the norm of the difference between the consistent tangent operators derived using the effective viscosity approach (Appendix A) and the finite-step consistent tangent linearisation (Appendix B). The two cases (A ,B) can be reproduced by changing the value of the TrialModel and PlastModel variables.

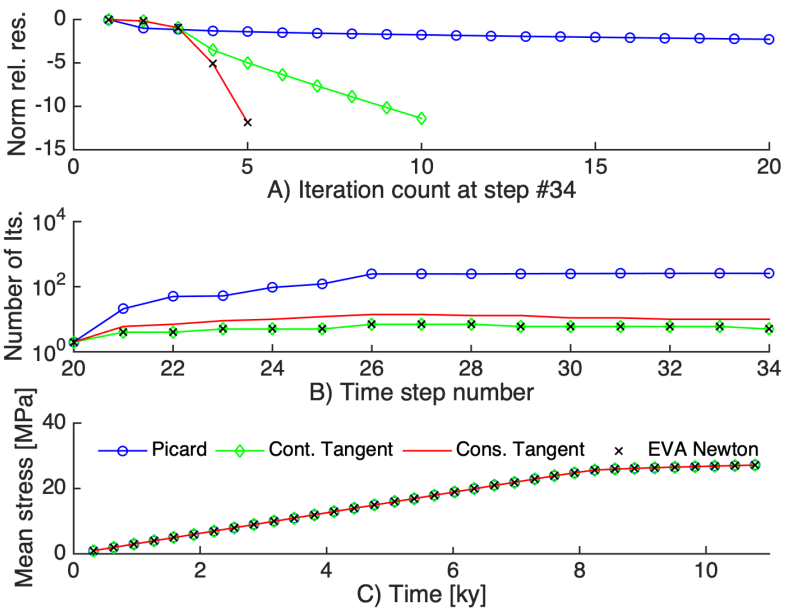


**Figure S8.** Effect of different linearisation schemes on non-linear convergence using the test case Test 1 of Duretz et al. (2018). Panel A shows the convergence behaviour of momentum equation for Picard, Continuum tangent, Consistent tangent (Cons. Tangent) and Newton linearisaton of the effective viscosity approach (EVA Newton) that are derived in the Appendix. Picard linearisation necessitates several hundreds of iterations to converge this step. EVA Newton and Cons. Tangent have a similar behaviour. Panel B shows the total number of iterations required for each linearisation scheme. Panel C shows the evolution of the second deviatoric stress invariant - the different approaches yield similar results. For all cases, a line search procedure was used to optimize the convergence of global iterations (e.g., Duretz et al, 2018).

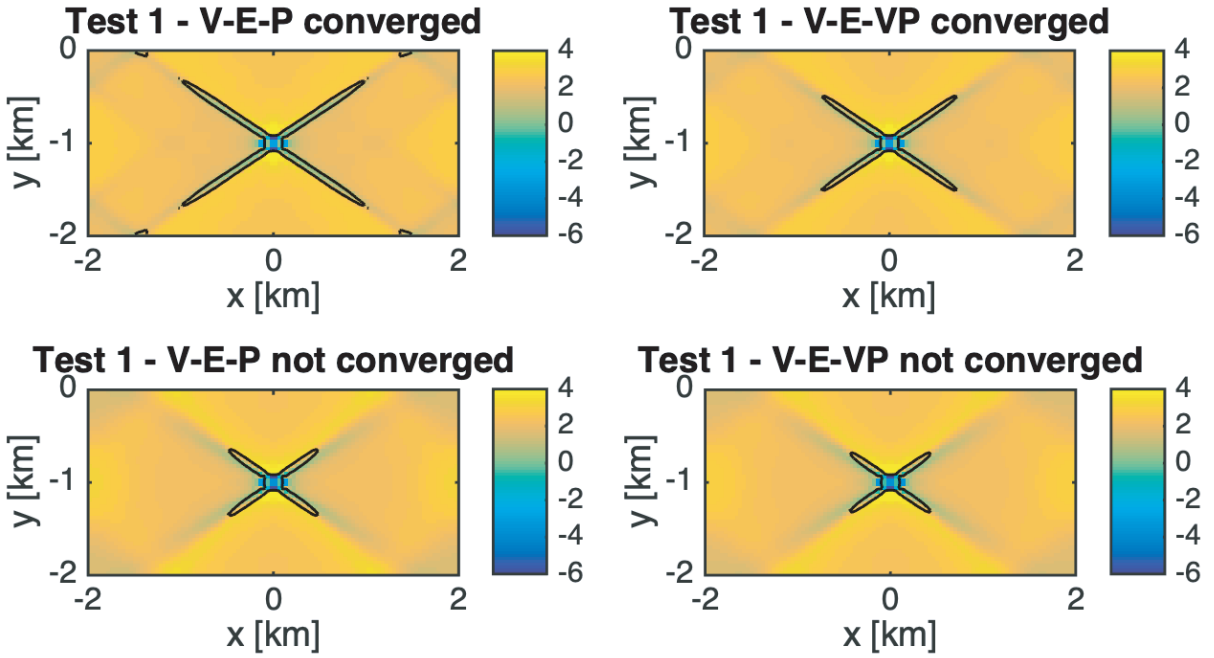


**Figure S9.** Effect of global momentum iterations on the results of numerical models of shear banding using Test 1 of Duretz et al. (2018). Left panels shows shear banding pattern after 34 steps for converged (upper) and non-converged (bottom) cases. Similarly, the right panels correspond to Test 1 including viscoplastic regularisation ($\eta^{\mathrm{vp}} = 4 \times 10^{18}$ Pa.s). The colormaps correspond to the pressure field in MPa and the black lines are isocontour of accumulated strain ($\epsilon_{\mathrm{acc}} = 10^{-2.6}$). Models that reached machine precision force equilibrium (lower panels) are characterised depict more intense strain localisation thus a longer extent of shear bands at a given time. Model results were depicted at time steps number 34.

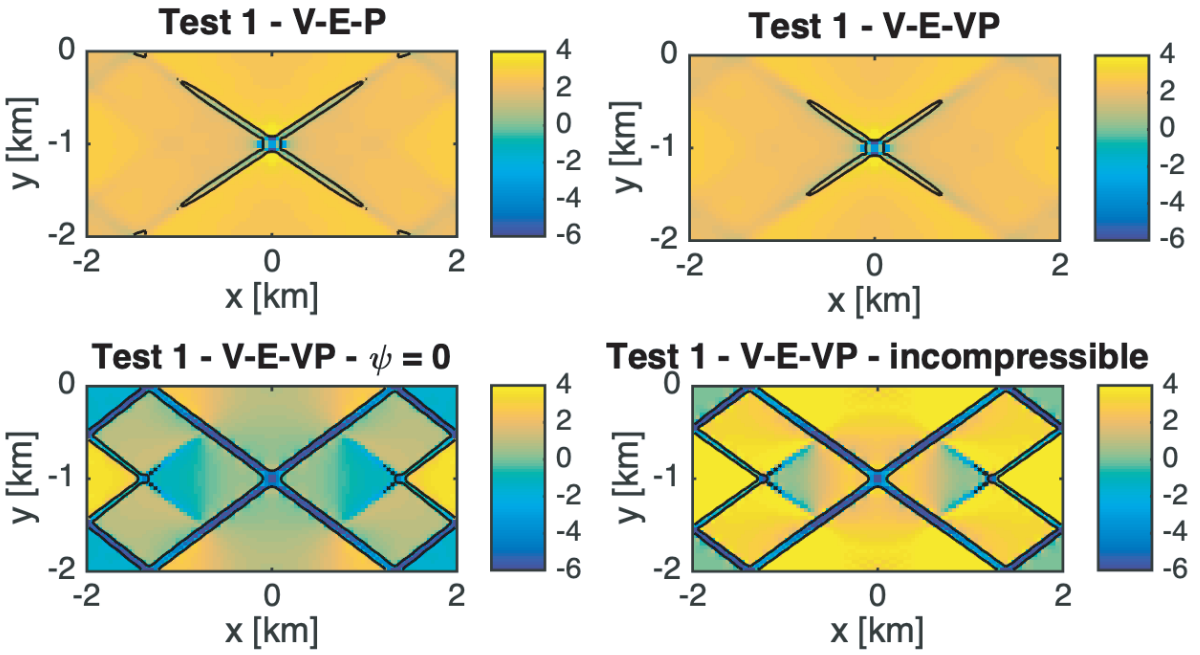


**Figure S10.** Effect of bulk elasticity and plastic dilatancy on shear banding based on Test 1 of Duretz et al. (2018). The upper left panel correspond to the reference model (V-E-P). The simulation presented in the upper right panel includes viscoplastic regularisation (V-E-VP, $\eta^{\mathrm{vp}} = 4 \times 10^{18}$ Pa.s). The lower left panel accounts for zero dilatancy (plastic incompressibility, $\psi = 0$). The lower right is strictly incompressible ($\psi = 0$ and $K \rightarrow$ inf), hence both effects of bulk elasticity and plastic dilatancy are neglected. The colormaps correspond to the pressure field in MPa and the black lines are isocontour of accumulated strain ($\epsilon_{\mathrm{acc}} = 10^{-2.6}$). It is clear that both elastic and plastic incompressibility promote strain localisation. Model results were depicted at time steps number 34.

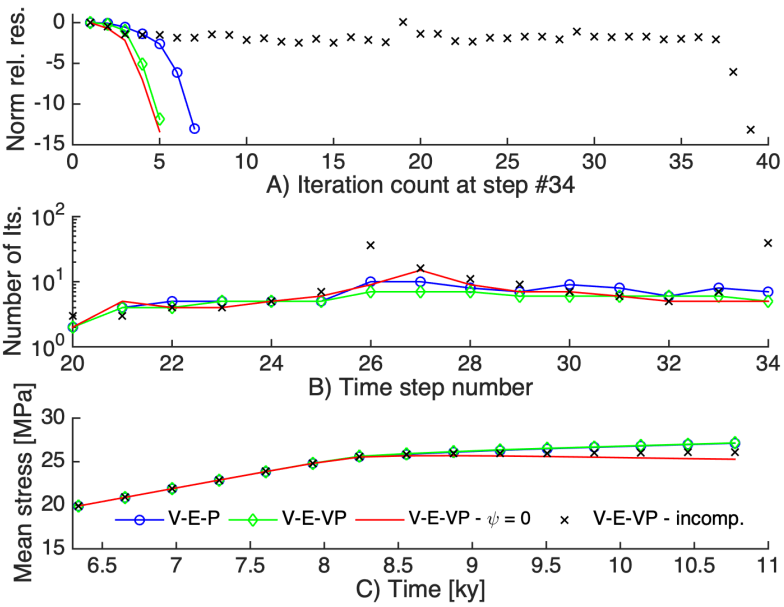


**Figure S11.** Effect of bulk elasticity and plastic dilatancy on the global convergence behaviour. Simulations were based on Test 1 of Duretz et al. (2018). The upper panel (A) shows the convergence behaviour of the V-E-P (reference), V-E-VP, V-E-VP with zero dilatancy and V-E-VP incompressible at time steps number 34. The middle panel (B) shows the number of iteration (Newton EVA) needed to fully converge each step. The incompressible case can lead to large iteration counts (step 26 and 34) has also seen in panel A. The lower panel (C) shows the evolution of the mean second deviatoric stress invariant for the 4 models. Models that include bulk elasticity and dilatancy exhibit a slight apparent hardening (V-E-P and V-E-VP) during shear banding. The incompressible model (V-E-VP incomp.) exhibit no hardening and the model with bulk elasticity and no dilatancy (V-E-VP - $\psi = 0$) exhibit a slight apparent softening.