

```

In[ ] := (* Initialisation *)
(* Evaluate before starting writing "real code" *)
(* Usage e.g.: "ld [Spacekey]" becomes "␣",
so writing "a ld 5" turns into "a␣5" *)
SetOptions[EvaluationNotebook [],
  InputAutoReplacements → {(* special AceGen assignment operators: *)
    "ld" → "␣", "ls" → "␣", "rd" → "␣", "rs" → "␣",
    (* brackets and symbols: *) "dbl" → "⌈",
    "dbr" → "⌋", "lcb" → "{", "rcb" → "}", "lsb" → "[", "rsb" → "]", "->" → "→",
    (* shortcuts for
starting/ending a comment block: *) "co" → "(*", "cc" → "*)"
  }
]
(* Output the current time,
so we know when AceGen has been executed the last time *)
Now

```

Out[] :=

Thu 20 Jun 2024 09:46:05 GMT+2

```

In[ ]:= (* Clear all old variables initially to have a fresh start *)
ClearAll["Global`*"]
(* Start AceGen *)
<< AceGen`;
(* Name of the to be created subroutine/function
in the below specified programming language *)
NAME = "HyperElasticity_PF ";
(* Name of the AceGen session "NAME",
specify the programming language "Language"={C++,Matlab,Fortran,...},
and the execution mode "Mode"={Optimal,Prototype,Debug,Plain} *)
(* @note Changing the output programming language can be very simple here,
so feel free to take advantage of all the available languages. For instance,
first export to a Matlab-code,
because you can quickly and easily debug and check the code and its output. *)
SMSInitialize[NAME, "Language" → "Matlab", "Mode" → "Optimal"];
(* Start a module, which represents the to be created function,
with name "NAME" and the specified input and output arguments *)
SMSModule[NAME, Real[F$$[3, 3], listOfMaterialParameters$$ [2],
CauchyStress$$ [3, 3], Tangent$$ [3, 3, 3, 3]],
"Input" → {F$$, listOfMaterialParameters$$ },
"Output" → {CauchyStress$$, Tangent$$ }
];
(* Input declaration by copying AceGen variables to Mathematica variables *)
F = SMSReal[Table[F$$[i, j], {i, 3}, {j, 3}]];
listOfMaterialParameters = SMSReal[Table[listOfMaterialParameters$$ [i], {i, 2}]];

In[ ]:= (* Extract the bulk modulus kappa and the shear
modulus mu from the list of material parameters *)
(* @note You cannot use the default underscore "_" in variable names,
instead the special character "[Esc][Esc]" is used. *)
(* @note You can also use greek/etc. symbols like "α" for variables as
in classical Mathematica notebooks, e.g. "[Esc]kappa[Esc]" for κ. *)
κ = bulkMod_kappa = listOfMaterialParameters [[1]];
μ = shearMod_mu = listOfMaterialParameters [[2]];
λ = Lamé_lambda = bulkMod_kappa -  $\frac{2}{3}$  shearMod_mu;

(* Compute the right Cauchy-
Green (RCG) strain tensor "RCG_C" from the deformation gradient. *)
(* @note We just introduce RCG_C as auxiliary variable,
but take all our derivatives with respect to "F",
which already has a "unique signature" by using SMSReal on its definition. *)
RCG_C = FT.F;

```

```

In[ ] := (* Compute the strain energy density, here for example a neo-Hookean energy *)
(* @note As we do not use a SMSFreeze on RCG_C its dependence on F is still
visible to AceGen. Therefore, you can compute J by using F or RCG_C.' *)
Psi_NH[RCG_C_] := (
  J = SMSSqrt[ SMSDet[RCG_C] ];
  Psi_NH =  $\frac{\text{shearMod\_mu}}{2} (\text{Tr}[\text{RCG\_C}] - 3) - \text{shearMod\_mu} \text{Log}[J] + \frac{\text{Lame\_lambda}}{2} (\text{Log}[J])^2$ ;
  Return[Psi_NH];
);

(* Compute the first Piola-
Kirchhoff stress tensor "PK1_P" derived from the energy *)
PK1_P = SMSD[Psi_NH[RCG_C], F];
(* Push-forward "PK1_P" to Cauchy stress "Cauchy_sigma" *)
Cauchy_sigma =  $\frac{1}{\text{SMSDet}[F]} \text{PK1\_P} \cdot F^T$ ;

In[ ] := (* Export the output variables by copying
the Mathematica variables to AceGen variables *)
(* Output/Export the stress vector "Cauchy_sigma" as variable CauchyStress$$ *)
SMSExport[Cauchy_sigma , Table[CauchyStress$$[i, j], {i, 3}, {j, 3}]];

(* Compute the dP_dF elasticity tangent *)
DPK1_DF = SMSD[PK1_P, F];
(* Output/Export the derivative as a fourth-order tensor *)
SMSExport[DPK1_DF , Table[Tangent$$[i, j, k, l], {i, 3}, {j, 3}, {k, 3}, {l, 3}]];

```

```

In[ ]:= (* Debugging/Verification *)

(* Output the stress tensor to the screen *)
SMSPrintMessage [NAME<>"<< Cauchy stress AceGen=", Cauchy_sigma];

(* Compute analytical Cauchy stress tensor *)
Cauchy_sigma_ay =

$$\frac{\mu}{\text{SMSDet}[F]} (F.F^T - \text{IdentityMatrix}[3]) + \frac{\lambda \text{Log}[\text{SMSDet}[F]]}{\text{SMSDet}[F]} \text{IdentityMatrix}[3];$$

(* SMSPrintMessage [NAME<>"<< Cauchy stress analytical=", Cauchy_sigma_ay]; *)
SMSPrintMessage [NAME<>"<< Cauchy stress error= ",

$$\frac{1}{\text{Cauchy\_sigma}[[1, 1]]} \text{Sum}[(\text{Cauchy\_sigma} - \text{Cauchy\_sigma\_ay})[[i, j]]^2, \{i, 3\}, \{j, 3\}]];$$

(* Compute the analytical tangent and compare it to the AceGen-Output *)
RCGi = SMSInverse [RCG_C];
dRCGi_dRCG =

$$-\frac{1}{2} \text{Table}[\text{RCGi}[[i, k]] \times \text{RCGi}[[j, l]] + \text{RCGi}[[i, l]] \times \text{RCGi}[[j, k]], \{i, 3\}, \{j, 3\}, \{k, 3\}, \{l, 3\}];$$

DPK2_DE_ay = (-2  $\mu$  + 2  $\lambda$  Log [SMSDet[F]]) dRCGi_dRCG +  $\lambda$  RCGi  $\otimes$  RCGi;
PK2_S = J SMSInverse [F] . Cauchy_sigma_ay . SMSInverse [F]^T;
DPK1_DF_ay = Table [PK2_S[[L, J]] KroneckerDelta [i, k] + Sum [
F[[i, I]]  $\times$  DPK2_DE_ay [[I, J, K, L]]  $\times$  F[[k, K]], {I, 3}, {K, 3}], {i, 3}, {J, 3}, {k, 3}, {L, 3}];

(* SMSPrintMessage [NAME<>"<< Tangent AceGen=", DPK1_DF];
SMSPrintMessage [NAME<>"<< Tangent ay      =", DPK1_DF_ay]; *)
SMSPrintMessage [NAME<>"<< relative error in tangent=",  $\frac{1}{\text{DPK1\_DF}[[1, 1, 1, 1]]}$ 
Sum [ (DPK1_DF - DPK1_DF_ay) [[i, j, k, l]]^2, {i, 3}, {j, 3}, {k, 3}, {l, 3}]];

```

In[]:= (* Output the time at the end of the execution *)

Now

```
(* Write output file containing all the
above defined functions introduced by SMSModule *)
(* Create output file named "NAME", '"LocalAuxiliaryVariables " →
True' is a command to exclude the AceGen internal array "v" from
the list of input and output arguments of the created subroutine *)
SMSWrite[NAME, "LocalAuxiliaryVariables " → True];
(* Print the content of the just created
file on screen (sensible only for small file sizes) *)
FilePrint[StringJoin[NAME, Which[SMSLanguage == "Fortran", ".f",
                                SMSLanguage == "Matlab", ".m",
                                SMSLanguage == "C++", ".cpp",
                                SMSLanguage == "C", ".c"]
]
```

]

Out[]:= Thu 20 Jun 2024 09:46:12 GMT+2

File: HyperElasticity_PF .m **Size:** 20200 **Time:** 12

Method	HyperElasticity_PF
No. Formulae	369
No. Leafs	8243

```
% *****
%* AceGen      7.505 Linux (16 Aug 22) *
%*            Co. J. Korelc 2020      20 Jun 24 09:46:19 *
% *****
% User       : Full professional version
% Notebook   : AceGen-HyperElasticity-unsym
% Evaluation time      : 12 s      Mode : Optimal
% Number of formulae   : 369      Method: Automatic
% Subroutine          : HyperElasticity_PF size: 8243
% Total size of Mathematica code : 8243 subexpressions
% Total size of Matlab code      : 19279 bytes
```

```
% *****      F U N C T I O N      *****
function[CauchyStress,Tangent]=HyperElasticity_PF(F,listOfMaterialParameters);
persistent v;
if size(v)<615
    v=zeros(615,'double');
end;
v(1)=F(1,1);
v(254)=(v(1)*v(1));
v(50)=2e0*v(1);
v(2)=F(1,2);
v(440)=v(1)*v(2);
v(255)=(v(2)*v(2));
```

```

v(53)=2e0*v(2);
v(438)=v(1)*v(53);
v(3)=F(1,3);
v(256)=(v(3)*v(3));
v(56)=2e0*v(3);
v(4)=F(2,1);
v(428)=v(1)*v(4);
v(259)=(v(4)*v(4));
v(51)=2e0*v(4);
v(5)=F(2,2);
v(488)=v(4)*v(5);
v(429)=v(2)*v(5);
v(343)=-(v(2)*v(4))+v(1)*v(5);
v(260)=(v(5)*v(5));
v(54)=2e0*v(5);
v(486)=v(4)*v(54);
v(6)=F(2,3);
v(430)=v(3)*v(6);
v(338)=v(3)*v(4)-v(1)*v(6);
v(261)=(v(6)*v(6));
v(57)=2e0*v(6);
v(7)=F(3,1);
v(264)=(v(7)*v(7));
v(52)=2e0*v(7);
v(8)=F(3,2);
v(415)=v(7)*v(8);
v(341)=v(2)*v(7)-v(1)*v(8);
v(437)=2e0*v(341);
v(335)=-(v(5)*v(7))+v(4)*v(8);
v(265)=(v(8)*v(8));
v(55)=2e0*v(8);
v(505)=v(55)*v(7);
v(9)=F(3,3);
v(337)=-(v(3)*v(7))+v(1)*v(9);
v(330)=-(v(6)*v(7))+v(4)*v(9);
v(326)=-(v(6)*v(8))+v(5)*v(9);
v(266)=(v(9)*v(9));
v(257)=v(1)*v(326)-v(2)*v(330)+v(3)*v(335);
v(58)=2e0*v(9);
v(11)=listOfMaterialParameters(2);
v(416)=v(11)/v(257);
v(27)=v(11)/2e0;
v(12)=listOfMaterialParameters(1)+(-2e0/3e0)*v(11);
v(402)=v(12)*log(v(257));
v(303)=-2e0*v(11)+2e0*v(402);
v(419)=v(303)/2e0;
v(417)=v(12)-v(303);
v(262)=v(402)/v(257);
v(13)=v(254)+v(259)+v(264);
v(14)=v(415)+v(440)+v(488);
v(403)=2e0*v(14);
v(141)=v(14)*v(58);
v(135)=v(14)*v(57);
v(128)=v(14)*v(56);
v(15)=v(1)*v(3)+v(4)*v(6)+v(7)*v(9);

```

```

v(405)=v(15)*v(403);
v(142)=-(v(15)*v(55));
v(136)=-(v(15)*v(54));
v(129)=-(v(15)*v(53));
v(16)=v(255)+v(260)+v(265);
v(404)=-(v(15)*v(16));
v(118)=-(v(14)*v(14))+v(13)*v(16);
v(17)=v(2)*v(3)+v(5)*v(6)+v(8)*v(9);
v(407)=2e0*v(17);
v(160)=v(17)*v(403)+2e0*v(404);
v(130)=v(405)-v(13)*v(407);
v(77)=v(17)*v(7);
v(187)=2e0*v(77);
v(73)=v(17)*v(4);
v(182)=2e0*v(73);
v(69)=v(1)*v(17);
v(177)=2e0*v(69);
v(65)=v(14)*v(17)+v(404);
v(406)=v(404)+v(65);
v(64)=-(v(17)*v(55))+v(16)*v(58);
v(62)=-(v(17)*v(54))+v(16)*v(57);
v(60)=-(v(17)*v(53))+v(16)*v(56);
v(18)=v(256)+v(261)+v(266);
v(408)=v(14)*v(18);
v(173)=-(v(18)*v(403))+v(15)*v(407);
v(147)=-(v(15)*v(15))+v(13)*v(18);
v(66)=-(v(15)*v(17))+v(408);
v(409)=v(408)+v(66);
v(63)=v(18)*v(55)-v(17)*v(58);
v(61)=v(18)*v(54)-v(17)*v(57);
v(59)=v(18)*v(53)-v(17)*v(56);
v(31)=-(v(17)*v(17))+v(16)*v(18);
v(414)=v(31)*v(50);
v(23)=v(13)*v(31)+v(15)*v(65)-v(14)*v(66);
v(79)=sqrt(v(23));
v(435)=v(79)/v(257);
v(411)=(-v(11)+v(12)*log(v(79)))/2e0;
v(410)=-(v(411)/Power(v(23), 2))+v(12)/(4e0*v(23)*Power(v(79), 2));
v(117)=v(410)*v(13)*v(64)+v(406)*v(7)+v(14)*(-v(141)+v(77))+v(405)*v(8));
v(127)=v(117)*v(118);
v(116)=v(410)*v(13)*v(63)-v(409)*v(7)+v(15)*(v(142)+v(77))+v(405)*v(9));
v(157)=v(116)*v(147);
v(115)=v(410)*v(31)*v(52)+v(173)*v(8)+v(160)*v(9));
v(199)=v(115)*v(31);
v(114)=v(410)*(v(4)*v(406)+v(405)*v(5)+v(13)*v(62)+v(14)*(-v(135)+v(73)));
v(124)=v(114)*v(118);
v(113)=v(410)*(-v(4)*v(409))+v(405)*v(6)+v(13)*v(61)+v(15)*(v(136)+v(73)));
v(153)=v(113)*v(147);
v(112)=v(410)*(v(173)*v(5)+v(31)*v(51)+v(160)*v(6));
v(194)=v(112)*v(31);
v(111)=v(410)*(v(2)*v(405)+v(1)*v(406)+v(13)*v(60)+v(14)*(-v(128)+v(69)));
v(110)=v(410)*(v(3)*v(405)-v(1)*v(409)+v(13)*v(59)+v(15)*(v(129)+v(69)));
v(108)=v(410)*(v(173)*v(2)+v(160)*v(3)+v(414));
v(25)=v(411)/v(23);
v(413)=v(25)*v(53);

```

```

v(412)=-v(25)*v(50));
v(202)=-v(25)*v(64));
v(203)=-v(202)+v(117)*v(31);
v(200)=-v(25)*v(63));
v(201)=-v(200)+v(116)*v(31);
v(197)=-v(25)*v(62));
v(198)=-v(197)+v(114)*v(31);
v(195)=-v(25)*v(61));
v(196)=-v(195)+v(113)*v(31);
v(192)=-v(25)*v(60));
v(190)=-v(25)*v(59));
v(188)=v(117)*v(173)+(-2e0*v(141)-v(142)+v(187))*v(25);
v(185)=v(25)*(v(18)*v(52)-v(15)*v(58));
v(186)=v(116)*v(173)-v(185);
v(184)=v(115)*v(173)+v(200);
v(183)=v(114)*v(173)+(-2e0*v(135)-v(136)+v(182))*v(25);
v(180)=v(25)*(v(18)*v(51)-v(15)*v(57));
v(181)=v(113)*v(173)-v(180);
v(179)=v(112)*v(173)+v(195);
v(178)=v(111)*v(173)+(-2e0*v(128)-v(129)+v(177))*v(25);
v(176)=v(110)*v(173)+v(18)*v(412)+v(15)*v(25)*v(56);
v(171)=v(25)*(v(16)*v(52)-v(14)*v(55));
v(172)=v(117)*v(160)-v(171);
v(234)=v(188)*v(5)+v(203)*v(51)+v(172)*v(6);
v(212)=v(188)*v(2)+v(172)*v(3)+v(203)*v(50);
v(170)=v(116)*v(160)+(v(141)+2e0*v(142)+v(187))*v(25);
v(233)=v(186)*v(5)+v(201)*v(51)+v(170)*v(6);
v(211)=v(186)*v(2)+v(170)*v(3)+v(201)*v(50);
v(169)=v(115)*v(160)+v(202);
v(232)=v(184)*v(5)+v(199)*v(51)+v(169)*v(6);
v(210)=v(184)*v(2)+v(169)*v(3)+v(199)*v(50);
v(167)=v(25)*(v(16)*v(51)-v(14)*v(54));
v(168)=v(114)*v(160)-v(167);
v(209)=v(183)*v(2)+v(168)*v(3)+v(198)*v(50);
v(166)=v(113)*v(160)+(v(135)+2e0*v(136)+v(182))*v(25);
v(208)=v(181)*v(2)+v(166)*v(3)+v(196)*v(50);
v(165)=v(112)*v(160)+v(197);
v(207)=v(179)*v(2)+v(165)*v(3)+v(194)*v(50);
v(164)=v(111)*v(160)+v(16)*v(412)+v(14)*v(413);
v(158)=v(25)*(v(15)*v(52)-v(13)*v(58));
v(159)=v(117)*v(147)-v(158);
v(156)=v(115)*v(147)+v(185);
v(154)=v(25)*(v(15)*v(51)-v(13)*v(57));
v(155)=v(114)*v(147)-v(154);
v(150)=v(25)*(v(15)*v(50)-v(13)*v(56));
v(145)=v(25)*(-(v(14)*v(52))+v(13)*v(55));
v(146)=v(117)*v(130)-v(145);
v(245)=v(172)*v(4)+v(146)*v(5)+v(127)*v(57);
v(240)=v(188)*v(4)+v(159)*v(54)+v(146)*v(6);
v(227)=v(1)*v(172)+v(146)*v(2)+v(127)*v(56);
v(220)=v(1)*v(188)+v(146)*v(3)+v(159)*v(53);
v(144)=v(116)*v(130)+v(158);
v(239)=v(186)*v(4)+v(157)*v(54)+v(144)*v(6);
v(219)=v(1)*v(186)+v(144)*v(3)+v(157)*v(53);
v(143)=v(115)*v(130)+v(25)*(v(141)-v(142)-v(407)*v(52));

```



```

v(238)=v(184)*v(4)+v(156)*v(54)+v(143)*v(6);
v(218)=v(1)*v(184)+v(143)*v(3)+v(156)*v(53);
v(139)=v(25)*(-(v(14)*v(51))+v(13)*v(54));
v(140)=v(114)*v(130)-v(139);
v(224)=v(1)*v(168)+v(140)*v(2)+v(124)*v(56);
v(217)=v(1)*v(183)+v(140)*v(3)+v(155)*v(53);
v(138)=v(113)*v(130)+v(154);
v(216)=v(1)*v(181)+v(138)*v(3)+v(153)*v(53);
v(137)=v(112)*v(130)+v(25)*(v(135)-v(136)-v(407)*v(51));
v(215)=v(1)*v(179)+v(137)*v(3)+(v(112)*v(147)+v(180))*v(53);
v(134)=v(111)*v(130)-v(14)*v(412)-v(13)*v(413);
v(126)=v(116)*v(118)+v(145);
v(244)=v(170)*v(4)+v(144)*v(5)+v(126)*v(57);
v(226)=v(1)*v(170)+v(144)*v(2)+v(126)*v(56);
v(125)=v(115)*v(118)+v(171);
v(243)=v(169)*v(4)+v(143)*v(5)+v(125)*v(57);
v(225)=v(1)*v(169)+v(143)*v(2)+v(125)*v(56);
v(223)=v(1)*v(166)+v(138)*v(2)+(v(113)*v(118)+v(139))*v(56);
v(222)=v(1)*v(165)+v(137)*v(2)+(v(112)*v(118)+v(167))*v(56);
v(24)=v(118)*v(25)+v(27);
v(241)=2e0*v(24);
v(251)=v(241)+v(127)*v(58)+v(172)*v(7)+v(146)*v(8);
v(242)=v(241)+v(168)*v(4)+v(140)*v(5)+v(124)*v(57);
v(221)=v(1)*v(164)+v(134)*v(2)+v(241)+v(111)*v(118)*v(56);
v(26)=v(130)*v(25);
v(250)=v(26)+v(159)*v(55)+v(188)*v(7)+v(146)*v(9);
v(237)=v(26)+v(183)*v(4)+v(155)*v(54)+v(140)*v(6);
v(214)=v(1)*v(178)+v(26)+v(134)*v(3)+(v(111)*v(147)-v(150))*v(53);
v(28)=v(147)*v(25)+v(27);
v(235)=2e0*v(28);
v(249)=v(235)+v(157)*v(55)+v(186)*v(7)+v(144)*v(9);
v(236)=v(235)+v(181)*v(4)+v(153)*v(54)+v(138)*v(6);
v(213)=v(1)*v(176)+v(235)+(v(110)*v(130)+v(150))*v(3)+v(110)*v(147)*v(53);
v(29)=v(160)*v(25);
v(248)=v(29)+v(203)*v(52)+v(188)*v(8)+v(172)*v(9);
v(231)=v(29)+v(183)*v(5)+v(198)*v(51)+v(168)*v(6);
v(206)=v(178)*v(2)+v(29)+v(164)*v(3)+(-v(192)+v(111)*v(31))*v(50);
v(30)=v(173)*v(25);
v(247)=v(30)+v(201)*v(52)+v(186)*v(8)+v(170)*v(9);
v(230)=v(30)+v(181)*v(5)+v(196)*v(51)+v(166)*v(6);
v(205)=v(176)*v(2)+(v(110)*v(160)+(v(128)+2e0*v(129)+v(177))*v(25))*v(3)+v(30)+(-v(190)+v(110)*v(31
    ))*v(50);
v(32)=v(27)+v(25)*v(31);
v(228)=2e0*v(32);
v(246)=v(228)+v(199)*v(52)+v(184)*v(8)+v(169)*v(9);
v(229)=v(228)+v(179)*v(5)+v(194)*v(51)+v(165)*v(6);
v(204)=(v(108)*v(173)+v(190))*v(2)+v(228)+(v(108)*v(160)+v(192))*v(3)+v(108)*v(414);
v(33)=v(1)*v(228)+v(29)*v(3)+v(2)*v(30);
v(34)=v(2)*v(235)+v(26)*v(3)+v(1)*v(30);
v(35)=v(2)*v(26)+v(1)*v(29)+v(241)*v(3);
v(36)=v(228)*v(4)+v(30)*v(5)+v(29)*v(6);
v(37)=v(30)*v(4)+v(235)*v(5)+v(26)*v(6);
v(38)=v(29)*v(4)+v(26)*v(5)+v(241)*v(6);
v(42)=(v(1)*v(33)+v(2)*v(34)+v(3)*v(35))/v(257);
v(44)=(v(33)*v(4)+v(34)*v(5)+v(35)*v(6))/v(257);

```

```

v(45)=(v(33)*v(7)+v(34)*v(8)+v(35)*v(9))/v(257);
v(46)=(v(36)*v(4)+v(37)*v(5)+v(38)*v(6))/v(257);
v(47)=(v(36)*v(7)+v(37)*v(8)+v(38)*v(9))/v(257);
v(48)=(2e0*(v(24)*v(266)+v(265)*v(28)+v(264)*v(32)+v(30)*v(415)+(v(29)*v(7)+v(26)*v(8))*v(9))/v(257);
CauchyStress(1,1)=v(42);
CauchyStress(1,2)=v(44);
CauchyStress(1,3)=v(45);
CauchyStress(2,1)=v(44);
CauchyStress(2,2)=v(46);
CauchyStress(2,3)=v(47);
CauchyStress(3,1)=v(45);
CauchyStress(3,2)=v(47);
CauchyStress(3,3)=v(48);
Tangent(1,1,1)=v(204);
Tangent(1,1,2)=v(205);
Tangent(1,1,3)=v(206);
Tangent(1,2,1)=v(207);
Tangent(1,2,2)=v(208);
Tangent(1,2,3)=v(209);
Tangent(1,3,1)=v(210);
Tangent(1,3,2)=v(211);
Tangent(1,3,3)=v(212);
Tangent(1,2,1)=v(205);
Tangent(1,2,2)=v(213);
Tangent(1,2,3)=v(214);
Tangent(1,2,2,1)=v(215);
Tangent(1,2,2,2)=v(216);
Tangent(1,2,2,3)=v(217);
Tangent(1,2,3,1)=v(218);
Tangent(1,2,3,2)=v(219);
Tangent(1,2,3,3)=v(220);
Tangent(1,3,1,1)=v(206);
Tangent(1,3,1,2)=v(214);
Tangent(1,3,1,3)=v(221);
Tangent(1,3,2,1)=v(222);
Tangent(1,3,2,2)=v(223);
Tangent(1,3,2,3)=v(224);
Tangent(1,3,3,1)=v(225);
Tangent(1,3,3,2)=v(226);
Tangent(1,3,3,3)=v(227);
Tangent(2,1,1,1)=v(207);
Tangent(2,1,1,2)=v(215);
Tangent(2,1,1,3)=v(222);
Tangent(2,1,2,1)=v(229);
Tangent(2,1,2,2)=v(230);
Tangent(2,1,2,3)=v(231);
Tangent(2,1,3,1)=v(232);
Tangent(2,1,3,2)=v(233);
Tangent(2,1,3,3)=v(234);
Tangent(2,2,1,1)=v(208);
Tangent(2,2,1,2)=v(216);
Tangent(2,2,1,3)=v(223);
Tangent(2,2,2,1)=v(230);
Tangent(2,2,2,2)=v(236);

```

```

Tangent(2,2,2,3)=v(237);
Tangent(2,2,3,1)=v(238);
Tangent(2,2,3,2)=v(239);
Tangent(2,2,3,3)=v(240);
Tangent(2,3,1,1)=v(209);
Tangent(2,3,1,2)=v(217);
Tangent(2,3,1,3)=v(224);
Tangent(2,3,2,1)=v(231);
Tangent(2,3,2,2)=v(237);
Tangent(2,3,2,3)=v(242);
Tangent(2,3,3,1)=v(243);
Tangent(2,3,3,2)=v(244);
Tangent(2,3,3,3)=v(245);
Tangent(3,1,1,1)=v(210);
Tangent(3,1,1,2)=v(218);
Tangent(3,1,1,3)=v(225);
Tangent(3,1,2,1)=v(232);
Tangent(3,1,2,2)=v(238);
Tangent(3,1,2,3)=v(243);
Tangent(3,1,3,1)=v(246);
Tangent(3,1,3,2)=v(247);
Tangent(3,1,3,3)=v(248);
Tangent(3,2,1,1)=v(211);
Tangent(3,2,1,2)=v(219);
Tangent(3,2,1,3)=v(226);
Tangent(3,2,2,1)=v(233);
Tangent(3,2,2,2)=v(239);
Tangent(3,2,2,3)=v(244);
Tangent(3,2,3,1)=v(247);
Tangent(3,2,3,2)=v(249);
Tangent(3,2,3,3)=v(250);
Tangent(3,3,1,1)=v(212);
Tangent(3,3,1,2)=v(220);
Tangent(3,3,1,3)=v(227);
Tangent(3,3,2,1)=v(234);
Tangent(3,3,2,2)=v(240);
Tangent(3,3,2,3)=v(245);
Tangent(3,3,3,1)=v(248);
Tangent(3,3,3,2)=v(250);
Tangent(3,3,3,3)=v(251);
disp(sprintf("\n%s  %f %f %f %f %f %f %f %f %f %f ", "HyperElasticity_PF<<  Cauchy stress AceGe
    ,v(44),v(45),v(44),v(46),v(47),v(45),v(47),v(48)));
v(258)=v(262)+(-1e0+v(254)+v(255)+v(256))*v(416);
v(340)=(-(v(258)*v(330))+v(337)*v(44)+v(338)*v(45))/v(257);
v(263)=v(262)+(-1e0+v(259)+v(260)+v(261))*v(416);
v(342)=(v(263)*v(337)-v(330)*v(44)+v(338)*v(47))/v(257);
v(267)=v(262)+(-1e0+v(264)+v(265)+v(266))*v(416);
v(344)=(v(267)*v(338)-v(330)*v(45)+v(337)*v(47))/v(257);
disp(sprintf("\n%s  %f ", "HyperElasticity_PF<<  Cauchy stress error= ", (Power(-(v(258)+v(42),2)+Powe
    -v(263)+v(46),2)+Power(-(v(267)+v(48),2))/v(42)));
v(270)=v(31)/v(23);
v(418)=v(12)*v(270);
v(271)=-(v(66)/v(23));
v(282)=(v(271)*v(271));
v(272)=v(65)/v(23);

```

```

v(288)=(v(272)*v(272));
v(273)=v(147)/v(23);
v(420)=v(272)*v(273);
v(274)=(v(14)*v(15)-v(13)*v(17))/v(23);
v(296)=(v(274)*v(274));
v(290)=-(v(271)*v(274));
v(275)=v(118)/v(23);
v(431)=v(273)*v(275);
v(421)=v(271)*v(275);
v(280)=-(v(271)*v(272));
v(287)=-(v(272)*v(274));
v(302)=(v(270)*v(270))*v(417);
v(304)=v(270)*v(271)*v(417);
v(422)=v(304)*v(7);
v(305)=v(270)*v(272)*v(417);
v(446)=v(1)*v(302)+v(2)*v(304)+v(3)*v(305);
v(425)=v(305)*v(7);
v(306)=-(v(282)*v(303))+v(273)*v(418);
v(307)=v(280)*v(303)+v(274)*v(418);
v(491)=v(307)*v(4);
v(460)=v(2)*v(306)+v(3)*v(307);
v(456)=v(1)*v(307);
v(308)=-(v(288)*v(303))+v(275)*v(418);
v(474)=v(2)*v(307)+v(3)*v(308);
v(309)=v(12)*v(282)+(-(v(270)*v(273))-v(282))*v(419);
v(441)=v(306)+v(309);
v(310)=-(v(12)*v(280))+(-(v(270)*v(274))+v(280))*v(419);
v(497)=v(310)*v(4);
v(493)=v(310)*v(5);
v(489)=v(309)*v(5)+v(310)*v(6);
v(487)=v(305)*v(4)+v(493);
v(475)=v(1)*v(310);
v(453)=v(2)*v(310);
v(449)=v(2)*v(309)+v(3)*v(310);
v(447)=v(1)*v(304)+v(449);
v(442)=v(307)+v(310);
v(439)=v(1)*v(305)+v(453);
v(432)=v(310)*v(7);
v(311)=v(271)*v(273)*v(417);
v(423)=v(311)*v(8);
v(312)=-(v(12)*v(290))+v(419)*(v(290)-v(420));
v(500)=v(312)*v(4);
v(496)=v(309)*v(4)+v(312)*v(6);
v(492)=v(312)*v(5);
v(478)=v(2)*v(312);
v(470)=v(1)*v(312);
v(461)=v(1)*v(309)+v(3)*v(312);
v(463)=v(2)*v(311)+v(461);
v(450)=v(456)+v(478);
v(426)=v(312)*v(8);
v(313)=v(287)*v(303)+v(12)*v(421);
v(502)=v(313)*v(6);
v(479)=v(3)*v(313);
v(480)=v(478)+v(479);
v(476)=v(475)+v(479);

```

```

v(452)=v(1)*v(308)+v(2)*v(313);
v(314)=v(12)*v(288)+(-(v(270)*v(275))-v(288))*v(419);
v(490)=v(314)*v(6);
v(454)=v(3)*v(314);
v(455)=v(453)+v(454);
v(448)=v(439)+v(454);
v(444)=v(308)+v(314);
v(315)=v(290)*v(303)+v(12)*v(420);
v(498)=v(315)*v(5);
v(466)=v(2)*v(315);
v(462)=v(466)+v(475);
v(451)=v(1)*v(306)+v(3)*v(315);
v(443)=v(312)+v(315);
v(316)=-(v(12)*v(287))+v(419)*(v(287)-v(421));
v(503)=v(314)*v(4)+v(316)*v(5);
v(494)=v(316)*v(6);
v(477)=v(1)*v(314)+v(2)*v(316);
v(467)=v(3)*v(316);
v(468)=v(466)+v(467);
v(457)=v(456)+v(467);
v(445)=v(313)+v(316);
v(424)=v(316)*v(9);
v(385)=v(4)*v(422)+v(423)*v(5)+v(424)*v(6);
v(362)=v(1)*v(422)+v(2)*v(423)+v(3)*v(424);
v(358)=v(304)*v(428)+v(311)*v(429)+v(316)*v(430);
v(317)=v(272)*v(275)*v(417);
v(483)=v(3)*v(317)+v(477);
v(427)=v(317)*v(9);
v(390)=v(4)*v(425)+v(426)*v(5)+v(427)*v(6);
v(372)=v(1)*v(425)+v(2)*v(426)+v(3)*v(427);
v(367)=v(305)*v(428)+v(312)*v(429)+v(317)*v(430);
v(319)=(v(273)*v(273))*v(417);
v(320)=v(273)*v(274)*v(417);
v(495)=v(320)*v(5)+v(500);
v(464)=v(1)*v(311)+v(2)*v(319)+v(3)*v(320);
v(458)=v(2)*v(320)+v(470);
v(433)=v(320)*v(8);
v(321)=-(v(296)*v(303))+v(12)*v(431);
v(481)=v(1)*v(315)+v(3)*v(321);
v(469)=v(1)*v(313)+v(2)*v(321);
v(322)=v(12)*v(296)+v(419)*(-(v(296)-v(431)));
v(504)=v(316)*v(4)+v(322)*v(5);
v(499)=v(322)*v(6);
v(482)=v(1)*v(316)+v(2)*v(322);
v(471)=v(3)*v(322);
v(472)=v(470)+v(471);
v(465)=v(458)+v(471);
v(459)=v(321)+v(322);
v(323)=v(274)*v(275)*v(417);
v(501)=v(317)*v(4)+v(323)*v(5);
v(484)=v(3)*v(323)+v(482);
v(473)=v(1)*v(317)+v(2)*v(323);
v(434)=v(323)*v(9);
v(392)=v(4)*v(432)+v(433)*v(5)+v(434)*v(6);
v(374)=v(1)*v(432)+v(2)*v(433)+v(3)*v(434);

```

```

v(369)=v(310)*v(428)+v(320)*v(429)+v(323)*v(430);
v(325)=(v(275)*v(275))*v(417);
v(485)=v(3)*v(325)+v(473);
v(327)=v(3)*v(8)-v(2)*v(9);
v(328)=-(v(3)*v(5))+v(2)*v(6);
v(333)=(v(267)*v(328)+v(326)*v(45)+v(327)*v(47))/v(257);
v(332)=(v(263)*v(327)+v(326)*v(44)+v(328)*v(47))/v(257);
v(331)=(v(258)*v(326)+v(327)*v(44)+v(328)*v(45))/v(257);
v(329)=(v(326)*v(331)+v(327)*v(332)+v(328)*v(333))*v(435);
v(334)=(-(v(330)*v(331))+v(332)*v(337)+v(333)*v(338))*v(435);
v(336)=(v(331)*v(335)+v(332)*v(341)+v(333)*v(343))*v(435);
v(339)=(-(v(330)*v(340))+v(337)*v(342)+v(338)*v(344))*v(435);
v(345)=(v(335)*v(340)+v(341)*v(342)+v(343)*v(344))*v(435);
v(346)=((v(258)*(v(335)*v(335))+v(263)*(v(341)*v(341))+v(267)*(v(343)*v(343))+2e0*v(335)*v(343)*v
(45)+v(437)*(v(335)*v(44)+v(343)*v(47)))*v(79))/v(23);
disp(sprintf("\n%s  %f  ", "HyperElasticity_PF<<  relative error in tangent=", (2e0*Power(v(205)-v(25
)*v(304)-v(255)*v(311)-v(256)*v(316)-v(334)-v(440)*v(441)-v(3)*(v(1)*v(442)+v(2)*v(443)), 2)
...
+2e0*Power(v(206)-v(254)*v(305)-v(255)*v(312)-v(256)*v(317)-v(336)-v(440)*v(442)-v(3)*(v(1)*v(444)
...
+v(2)*v(445)), 2)+2e0*Power(v(214)-v(254)*v(310)-v(255)*v(320)-v(256)*v(323)-v(345)-v(440)*v(443)-v
...
(3)*(v(1)*v(445)+v(2)*v(459)), 2)+Power(v(204)-v(254)*v(302)-v(255)*v(309)-v(256)*v(314)-v(329)-v
...
(304)*v(438)-v(439)*v(56), 2)+Power(v(213)-v(254)*v(309)-v(255)*v(319)-v(256)*v(322)-v(339)-v(311)*v
...
(438)-v(458)*v(56), 2)+Power(v(221)-v(254)*v(314)-v(255)*v(322)-v(256)*v(325)-v(346)-v(316)*v(438)-v
...
(473)*v(56), 2)+Power(v(229)-v(259)*v(302)-v(260)*v(309)-v(261)*v(314)-v(329)-v(304)*v(486)-v(487)*v
...
(57), 2)+Power(v(236)-v(259)*v(309)-v(260)*v(319)-v(261)*v(322)-v(339)-v(311)*v(486)-v(495)*v(57), 2)
...
+Power(v(242)-v(259)*v(314)-v(260)*v(322)-v(261)*v(325)-v(346)-v(316)*v(486)-v(501)*v(57), 2)
...
+2e0*Power(v(207)-v(4)*v(446)-v(447)*v(5)-v(448)*v(6), 2)+2e0*Power(v(208)-v(358)-v(4)*v(449)-v(451)
...
)*v(5)-v(450)*v(6), 2)+2e0*Power(v(209)-v(367)-v(4)*v(455)-v(457)*v(5)-v(452)*v(6), 2)+2e0*Power(v
...
(215)-v(358)-v(4)*v(460)-v(461)*v(5)-v(462)*v(6), 2)+2e0*Power(v(216)-v(4)*v(463)-v(464)*v(5)-v(465)
...
)*v(6), 2)+2e0*Power(v(217)-v(369)-v(4)*v(468)-v(472)*v(5)-v(469)*v(6), 2)+2e0*Power(v(222)-v(367)-v
...
(4)*v(474)-v(476)*v(5)-v(477)*v(6), 2)+2e0*Power(v(223)-v(369)-v(4)*v(480)-v(481)*v(5)-v(482)*v(6), 2)
...
)+2e0*Power(v(224)-v(4)*v(483)-v(484)*v(5)-v(485)*v(6), 2)+2e0*Power(v(230)-v(259)*v(304)-v(260)*v
...
(311)-v(261)*v(316)-v(334)-v(441)*v(488)-(v(4)*v(442)+v(443)*v(5))*v(6), 2)+2e0*Power(v(231)-v(259)
...
)*v(305)-v(260)*v(312)-v(261)*v(317)-v(336)-v(442)*v(488)-(v(4)*v(444)+v(445)*v(5))*v(6), 2)
...
+2e0*Power(v(237)-v(259)*v(310)-v(260)*v(320)-v(261)*v(323)-v(345)-v(443)*v(488)-(v(4)*v(445)+v(459)
...
)*v(5))*v(6), 2)+Power(v(249)-v(264)*v(309)-v(265)*v(319)-v(266)*v(322)-v(339)-v(311)*v(505)-v(58)*
...
(v(433)+v(312)*v(7)), 2)+Power(v(246)-v(264)*v(302)-v(265)*v(309)-v(266)*v(314)-v(329)-v(422)*v(55)
...
-v(58)*(v(425)+v(310)*v(8)), 2)+Power(v(251)-v(264)*v(314)-v(265)*v(322)-v(266)*v(325)-v(346)-v(316)
...
)*v(505)-v(58)*(v(317)*v(7)+v(323)*v(8)), 2)+2e0*Power(v(210)-v(446)*v(7)-v(447)*v(8)-v(448)*v(9), 2)
...
+2e0*Power(v(211)-v(362)-v(449)*v(7)-v(451)*v(8)-v(450)*v(9), 2)+2e0*Power(v(212)-v(372)-v(455)*v(7)
...
-v(457)*v(8)-v(452)*v(9), 2)+2e0*Power(v(218)-v(362)-v(460)*v(7)-v(461)*v(8)-v(462)*v(9), 2)
...
+2e0*Power(v(219)-v(463)*v(7)-v(464)*v(8)-v(465)*v(9), 2)+2e0*Power(v(220)-v(374)-v(468)*v(7)-v(472)
...
)*v(8)-v(469)*v(9), 2)+2e0*Power(v(225)-v(372)-v(474)*v(7)-v(476)*v(8)-v(477)*v(9), 2)+2e0*Power(v
...
(226)-v(374)-v(480)*v(7)-v(481)*v(8)-v(482)*v(9), 2)+2e0*Power(v(227)-v(483)*v(7)-v(484)*v(8)-v(485)
...
)*v(9), 2)+2e0*Power(v(232)-(v(302)*v(4)+v(304)*v(5)+v(305)*v(6))*v(7)-(v(304)*v(4)+v(489))*v(8)-(v
...
(487)+v(490))*v(9), 2)+2e0*Power(v(233)-v(385)-v(489)*v(7)-(v(306)*v(4)+v(315)*v(6))*v(8)-(v(491)+v
...
(492))*v(9), 2)+2e0*Power(v(238)-(v(306)*v(5)+v(307)*v(6))*v(7)-v(496)*v(8)-(v(497)+v(498))*v
...
(9), 2)+2e0*Power(v(239)-(v(496)+v(311)*v(5))*v(7)-(v(311)*v(4)+v(319)*v(5)+v(320)*v(6))*v(8)-(v(495)
...
)+v(499))*v(9), 2)+2e0*Power(v(234)-v(390)-(v(490)+v(493))*v(7)-(v(491)+v(494))*v(8)-(v(308)*v(4)+v
...
(313)*v(5))*v(9), 2)+2e0*Power(v(240)-v(392)-(v(494)+v(498))*v(7)-(v(499)+v(500))*v(8)-(v(313)*v(4)
...
+v(321)*v(5))*v(9), 2)+2e0*Power(v(243)-v(390)-(v(307)*v(5)+v(308)*v(6))*v(7)-(v(497)+v(502))*v(8)-v
...
(503)*v(9), 2)+2e0*Power(v(244)-v(392)-(v(492)+v(502))*v(7)-(v(315)*v(4)+v(321)*v(6))*v(8)-v(504)*v
...
(9), 2)+2e0*Power(v(245)-(v(503)+v(317)*v(6))*v(7)-(v(504)+v(323)*v(6))*v(8)-(v(501)+v(325)*v(6))*v
...
(9), 2)+2e0*Power(v(247)-v(264)*v(304)-v(265)*v(311)-v(266)*v(316)-v(334)-v(415)*v(441)-(v(442)*v(7)
...
+v(443)*v(8))*v(9), 2)+2e0*Power(v(248)-v(264)*v(305)-v(265)*v(312)-v(266)*v(317)-v(336)-v(415)*v
...
(442)-(v(444)*v(7)+v(445)*v(8))*v(9), 2)+2e0*Power(v(250)-v(264)*v(310)-v(265)*v(320)-v(266)*v(323)
...

```

```
-v(345)-v(415)*v(443)-(v(445)*v(7)+v(459)*v(8))*v(9),2))/v(204)));
```

```
function [x]=SMSKDelta(i,j)
```

```
if (i==j) , x=1; else x=0; end;
```

```
end
```

```
function [x]=SMSDeltaPart(a,i,j,k)
```

```
l=round(i/j);
```

```
if (mod(i,j) ~= 0 | l>k) , x=0; else x=a(l); end;
```

```
end
```

```
function [x]=Power(a,b)
```

```
x=a^b;
```

```
end
```

```
function [x]=SMSTernaryOperator(a,b,c)
```

```
if (c) , x=a; else x=b; end;
```

```
end
```

```
end
```