

**CWP-172**  
**May 1995**



**Converting expressions to  
Thomsen notation**

Jack K. Cohen

Center for Wave Phenomena  
Colorado School of Mines  
Golden, Colorado 80401  
303/273-3557

# Converting Expressions to Thomsen Notation

Jack K. Cohen

## ABSTRACT

Leon Thomsen introduced a set of parameters that allow specialization to weakly transverse isotropic (TI) media without losing the capability of treating the general TI medium. For this reason, many studies of transverse isotropic media involve the conversion of expressions in the conventional notations to the corresponding expression in terms of Thomsen's parameters  $\gamma$ ,  $\delta$ , and  $\epsilon$ . These conversions are awkward because one of the transformation equations is *nonlinear* in  $\delta$ . For example, in the Voigt tensor notation,

$$C_{13} = -\rho c_S^2 + \rho \sqrt{(c_P^2 - c_S^2)^2 + 2c_P^2(c_P^2 - c_S^2)}\delta.$$

By introducing a modified parameter,  $\tilde{\delta}$ , this relation can be made linear. Indeed, the entire transformation from Voigt notation to Thomsen notation becomes linear. The same is true for the conversion from the other conventional notations. If an expression in the pure Thomsen parameters is desired, one can replace  $\tilde{\delta}$  by its definition in terms of  $\delta$  as the last step in the calculation. In the limit of *weak* transverse isotropy, the parameter  $\tilde{\delta}$  reduces to Thomsen's  $\delta$ . Thus, in this important special case, the conversion to pure Thomsen notation amounts to just replacing  $\tilde{\delta}$  by  $\delta$ . The *Mathematica* package, `Thomsen.m`, containing functions to automate conversions between the various TI notations accompanies this article.

## INTRODUCTION

Leon Thomsen (1986) introduced a set of parameters that allow specialization to weakly transverse isotropic (TI) media without losing the capability of treating the general TI medium. For this reason, many studies of transverse isotropic media involve the conversion of expressions in the Love, Voigt or Hookean tensor notations to the corresponding expression in terms of Thomsen's parameters  $\gamma$ ,  $\delta$ , and  $\epsilon$ . These conversions are awkward because one of the transformation equations is *nonlinear* in  $\delta$ . For example, in the Voigt notation,

$$C_{13} = -\rho c_S^2 + \rho \sqrt{(c_P^2 - c_S^2)^2 + 2c_P^2(c_P^2 - c_S^2)}\delta, \quad (1)$$

where  $\rho$  is the density and  $c_P, c_S$  are the compressional and shear speeds. By introducing a modified parameter,  $\tilde{\delta}$ , this relation can be made linear. Indeed, the entire transformation from Voigt notation to Thomsen notation becomes linear. It also turns out that many expressions are simpler in terms of a parameter  $f$  introduced by Ilya Tsvankin (1994). The five parameters  $c_P, f, \gamma, \epsilon$ , and  $\tilde{\delta}$  form a convenient *canonical* set of parameters. Once expressions have been cast in terms of these parameters, one can make subsequent substitutions that eliminate  $f$  in favor of  $c_S$  and/or  $\tilde{\delta}$  in favor of  $\delta$ .

In this report, I describe a *Mathematica* package to convert between the various notations for the TI parameters. In particular, the user has the choice of using  $c_S$  or  $f$  and the choice of using  $\tilde{\delta}$  or  $\delta$ .

### THE PARAMETERS $\tilde{\delta}$ AND $f$

First, simplify equation (1) by introducing Tsvankin's  $f$ :

$$f = \frac{c_P^2 - c_S^2}{c_P^2}, \quad (2)$$

yielding

$$C_{13} = -\rho c_S^2 + \rho c_P^2 \sqrt{f^2 + 2f\delta}. \quad (3)$$

Next, eliminate the square root by introducing  $\tilde{\delta}$  as

$$f^2 + 2f\delta = (f + \tilde{\delta})^2, \quad (4)$$

obtaining the *linear* expression

$$C_{13} = -\rho c_S^2 + \rho c_P^2 (f + \tilde{\delta}). \quad (5)$$

Equation (4) allows us to express  $\tilde{\delta}$  in terms of  $\delta$  or *vice versa*:

$$\tilde{\delta} = f \left( \sqrt{1 + \frac{2\delta}{f}} - 1 \right), \quad (6)$$

and

$$\delta = \tilde{\delta} \left( 1 + \frac{\tilde{\delta}}{2f} \right). \quad (7)$$

An application of the binomial theorem to equation (6) shows that in the limit of *weak* transverse isotropy, the parameter  $\tilde{\delta}$  reduces to Thomsen's  $\delta$ . Thus, in this important special case, the conversion to the Thomsen parameter amounts to just replacing  $\tilde{\delta}$  by  $\delta$ .

The quantity  $\tilde{\delta}$  introduced in this section also plays a role in the basic TI wave equations. In a companion report (*Thomsen Operators and Thomsen Matrices*: this volume), I show that the TI wave equation operator has the *exact* form,

$$\mathcal{L} = \mathcal{L}^{(0)} + \gamma \mathcal{L}^{(\gamma)} + \tilde{\delta} \mathcal{L}^{(\delta)} + \epsilon \mathcal{L}^{(\epsilon)}. \quad (8)$$

Here, the operator  $\mathcal{L}^{(0)}$  is the isotropic wave equation operator, while the “Thomsen operators,”  $\mathcal{L}^{(\gamma)}$ ,  $\mathcal{L}^{(\delta)}$ , and  $\mathcal{L}^{(\epsilon)}$ , characterize the anisotropic contributions. The isotropic operator and the three Thomsen operators are *independent* of the Thomsen parameters  $\gamma$ ,  $\delta$ , and  $\epsilon$ , so that the TI operator is *linear* in  $\gamma$ ,  $\epsilon$ , and the modified Thomsen parameter  $\tilde{\delta}$ .

### NOTATIONS FOR THE TI ELASTIC PARAMETERS

Using  $f$  and  $\tilde{\delta}$  as introduced above, the Voight parameters are expressed in terms of the Thomsen parameters,  $\gamma$ ,  $\delta$ ,  $\epsilon$ , and the material parameters by the equations (Thomsen, 1986):

$$\begin{aligned}
 C_{11} &= \rho c_P^2(1 + 2\epsilon) \\
 C_{33} &= \rho c_P^2 \\
 C_{13} &= -\rho c_S^2 + \rho c_P^2(f + \tilde{\delta}) \\
 C_{44} &= \rho c_S^2 \\
 C_{66} &= \rho c_S^2(1 + 2\gamma)
 \end{aligned} \tag{9}$$

Note that here, “3” is used as the index corresponding to the axis of symmetry—many authors use “1” as the distinguished direction.

The Love parameters are simply aliases for the Voight parameters (Musgrave, 1970):

$$\begin{aligned}
 A &= C_{11} \\
 C &= C_{33} \\
 F &= C_{13} \\
 L &= C_{44} \\
 N &= C_{66}
 \end{aligned} \tag{10}$$

In terms of the Voight notation, the fundamental Hookean parameters are given by:

$$\begin{aligned}
 C_{1111} &= C_{33} \\
 C_{2222} &= C_{3333} = C_{11} \\
 C_{2233} &= C_{3322} = C_{11} - 2C_{66} \\
 C_{1133} &= C_{3311} = C_{1122} = C_{2211} = C_{13} \\
 C_{1212} &= C_{2121} = C_{2112} = C_{1221} = C_{44} \\
 C_{1313} &= C_{3131} = C_{3113} = C_{1331} = C_{44} \\
 C_{2323} &= C_{3232} = C_{3223} = C_{2332} = C_{66}
 \end{aligned} \tag{11}$$

Finally, mention a lesser used notation appearing in Kuprazde (1976):

$$C_1 = C_{11}$$

$$\begin{aligned}
C_2 &= C_{11} - C_{66} \\
C_3 &= C_{13} \\
C_4 &= C_{33} \\
C_5 &= C_{44}
\end{aligned}
\tag{12}$$

## THE MATHEMATICA IMPLEMENTATION

The *Mathematica* implementation reserves the following variables for their meaning as notations for the elastic parameters:

rho, cp, cs, gamma, delta, deltaTilde, epsilon, eta, f,  
C11, C33, C13, C44, C66,  
A, C, F, L, N,  
C1, C2, C3, C4, C5, C6,  
C1111, C2222, C3333, C2233, C3322, C1133, C3311,  
C1122, C2211, C1212, C2121, C2112, C1221,  
C1313, C3131, C3113, C1331, C2323, C3232, C3223, C2332

Note well that users of this package cannot use these variables for any other purpose!

The conversion rules are implemented by the user functions:

**ConvertCij** converts an expression in the Voight parameters to an expression in the canonical variables  $c_P$ ,  $f$ ,  $\gamma$ ,  $\epsilon$ , and  $\tilde{\delta}$ .

**ConvertWeak** assumes the weak TI limit and converts an expression in the Voight and/or Thomsen parameters to an expression in the variables  $c_P$ ,  $f$ ,  $\gamma$ ,  $\epsilon$ , and  $\delta$ .

**EliminateF** replaces  $f$  by its definition in terms of  $c_P$  and  $c_S$ .

**EliminateCs** is a partial inverse of **EliminateF**; it replaces  $c_S$  in favor of  $c_P$  and  $f$ .

**EliminateDeltaTilde** replaces  $\tilde{\delta}$  by its definition in terms of  $\delta$  and  $f$ .

**EliminateDelta** is a partial inverse of **EliminateDeltaTilde**; it replaces  $\delta$  in favor of  $\tilde{\delta}$  and  $f$ .

**EliminateEta** replaces  $\eta$  by its definition in terms of  $\delta$  and  $\epsilon$ .

**EliminateEpsilon** is a partial inverse of **EliminateEta**; it replaces  $\eta$  in favor of  $\delta$  and  $\epsilon$ .

**EliminateLove** replaces the Love parameters by the Voight parameters.

**EliminateKuprazde** replaces the Kuprazde parameters by the Voight parameters.

**EliminateHooke** replaces the Hooke parameters by the Voight parameters.

**ConvertThomsen** converts an expression from Thomsen notation to Voigt notation. Any of the variants,  $\delta$ ,  $\tilde{\delta}$ ,  $\eta$ ,  $c_S$ , and  $f$  may appear in the expression.

**SineForm** writes a trigonometric expression using powers of sine.

**CosineForm** writes a trigonometric expression using powers of cosine.

The user functions are implemented with the aid of “private” or “hidden” rules such as:

```
CsToFRule =
  cs^n_Integer?Positive ->
    (cp^2(1 - f))^Quotient[n,2] cs^Mod[n,2]
```

...

```
DeltaRule = deltaTilde -> f(Sqrt[1+2delta/f] - 1)
```

...

```
CijRules =
{
  C11 -> rho cp^2 (1 + 2 epsilon),
  C33 -> rho cp^2,
  C13 -> -rho cs^2 + rho cp^2 (f + deltaTilde),
  C44 -> rho cs^2,
  C66 -> rho cs^2 (1 + 2 gamma)
}
```

...

These rules cannot be directly accessed by the user—the “public” functions described above provide the user interface. For example, the private `DeltaRule` rule is accessed by the public function `EliminateDeltaTilde`:

```
EliminateDeltaTilde[expression_] := expression /. DeltaRule//Simplify
```

The main conversion engines are the functions `ConvertCij` and `ConvertWeak`. The first merely provides public access to the appropriate private rules:

```
ConvertCij[expression_] := expression /. CijRules /. CsToFRule//Simplify
```

The implementation of `ConvertWeak` is a bit harder. While *Mathematica* has a facility for expanding functions in multiple power series, there is no built-in way to impose that terms like  $\epsilon^2$  and  $\epsilon\delta$  are equally to be neglected. A simple way to impose such conditions is to introduce a scaling parameter, here  $x$ , and expand in this single

parameter (this is analogous to the methodology used in the Calculus of Variations and other applications where functional derivatives are needed). Despite the programming maxim of having modules do a single job, for user convenience I decided to embed a call to `ConvertCij` within this code:

```
ConvertWeak[expression_] :=
Module[{x, tmp},
  tmp = ConvertCij[expression];
  tmp = tmp /. EpsilonRule /. DeltaTildeRule /.
  {
    epsilon -> epsilon x,
    deltaTilde -> delta x,
    gamma -> gamma x
  };
Normal@Series[tmp, {x,0,1}] /. x->1 /. CsToFRule//
Simplify//PowerExpand
]
```

## USAGE EXAMPLES

### Voight to Thomsen Examples

The `cracks` expression defined below is used as an example to illustrate use of the functions in the `Thomsen.m` package. The corresponding equation,

$$C_{11}C_{33} - C_{13}^2 = 2C_{66}(C_{13} + C_{33}), \quad (13)$$

represents the relation between  $C_{ij}$ 's for TI media formed by a system of thin parallel horizontal cracks in a purely isotropic matrix. It comes from the fact that in this case there are only four independent parameters instead of five for general TI media. See the more detailed discussion in the paper by Schoenberg and Sayers (1995) (the only difference is that their cracks are vertical). First, convert the expression to the canonical parameters and store the result in the variable `generalCracks`:

```
cracks = C11 C33 - C13^2 - 2 C66 (C13 + C33);
generalCracks = ConvertCij[cracks]
      4          2
cp (-deltaTilde + 2 epsilon - 2 deltaTilde f -
      4 deltaTilde gamma - 8 f gamma +
      2          2
      4 deltaTilde f gamma + 8 f gamma) rho
```

Next, eliminate  $f$  in favor of the speed  $c_5$ :

EliminateF[%]

$$\begin{aligned}
 & \text{cp} \left( -2 \text{deltaTilde} + \frac{2 \text{cs} \text{deltaTilde}^2}{\text{cp}} - \text{deltaTilde}^2 + \right. \\
 & \left. 2 \text{epsilon} - \frac{8 \text{cs} \text{gamma}^2}{\text{cp}} + \frac{8 \text{cs} \text{gamma}^4}{\text{cp}} - \right. \\
 & \left. \frac{4 \text{cs} \text{deltaTilde} \text{gamma}^2}{\text{cp}} \right) \text{rho}
 \end{aligned}$$

Just as an illustration, switch back to the canonical representation by eliminating  $c_S$  in favor of  $f$ :

EliminateCs[%]

$$\begin{aligned}
 & \text{cp} \left( -\text{deltaTilde}^2 + 2 \text{epsilon} - 2 \text{deltaTilde} f - \right. \\
 & \left. 4 \text{deltaTilde} \text{gamma} - 8 f \text{gamma} + \right. \\
 & \left. 4 \text{deltaTilde} f \text{gamma} + 8 f^2 \text{gamma} \right) \text{rho}
 \end{aligned}$$

### Voigt to Weak TI Thomsen Examples

Do the canonical conversion in the limit of weak TI. Recall that in this limit,  $\delta$  and  $\tilde{\delta}$  are equal.

weakCracks = ConvertWeak[cracks]

$$\begin{aligned}
 & 2 \text{cp} \left( \text{epsilon} - \text{delta} f - 4 f \text{gamma} + 4 f^2 \text{gamma} \right) \\
 & \text{rho}
 \end{aligned}$$

Use *Mathematica* to solve for  $\delta$ :



delta /. Flatten@Solve[weakCracks == 0, delta]//Simplify

$$\frac{\text{epsilon}}{f} - 4 \text{ gamma} + 4 f \text{ gamma}$$

Use a typical value  $c_P/c_S = 2$  which is equivalent to the value  $f = 3/4$  to get a feel for this result:

% /. f -> 3/4

$$\frac{4 \text{ epsilon}}{3} - \text{gamma}$$

Introduce  $c_S$  in favor of  $f$ :

EliminateF[weakCracks]

$$2 \left( -c_P \delta^4 + c_P c_S^2 \delta^2 + c_P \epsilon^4 - 4 c_P c_S^2 \text{gamma} + 4 c_S^4 \text{gamma}^2 \right) \rho$$

Solve for  $\delta$  in the new variables:

delta /. Flatten@Solve[% == 0, delta] //Simplify

$$-\left( \frac{c_P \epsilon^4 - 4 c_P c_S^2 \text{gamma} + 4 c_S^4 \text{gamma}^2}{-c_P + c_P c_S^2} \right)$$

Use a standard Mathematica function to again get the result in our typical case:

Limit[%, cp -> 2 cs]//Simplify

$$\frac{4 \text{ epsilon}}{3} - \text{gamma}$$

ConvertWeak can also handle expressions involving the standard Thomsen parameters, as well as  $f$  and  $\tilde{\delta}$ :

```

ConvertWeak[C33(1 + 2 epsilon)^3 - C44(1 + deltaTilde + delta)^3]
      2
cp (-6 delta + 6 epsilon + f + 6 delta f) rho

Collect[%, {cp, rho, delta}]
      2
cp (6 epsilon + f + delta (-6 + 6 f)) rho

```

### Eliminating $\tilde{\delta}$

When  $\tilde{\delta}$  is eliminated, the underlying square root becomes explicit (unless we make the weak TI assumption):

```

EliminateDeltaTilde[generalCracks]
      4                      2
2 cp (epsilon - delta f - 2 f gamma + 2 f gamma -

      2 delta + f
2 f Sqrt[-----] gamma +
      f

      2      2 delta + f      2
2 f Sqrt[-----] gamma) rho
      f

```

And if we want to use the pure Thomsen parameters, continue by eliminating  $f$ :

```

EliminateF[%]
      4      2 2      4
2 (-(cp delta) + cp cs delta + cp epsilon -

      2 2      4
2 cp cs gamma + 2 cs gamma -

      2 2      2 2      2
2 cp cs Sqrt[-----] gamma +
      2 2
      cp - cs

      2 2      2 2      2
2 cs Sqrt[-----] gamma) rho
      2 2
      cp - cs

```

Now go the weak TI limit and recover our typical case for the third time:

```

ConvertWeak[%]
      4                2
      2 cp (epsilon - delta f - 4 f gamma + 4 f gamma)

      2
      rho

% /. f -> 3/4
      4 -3 delta      3 gamma      2
      2 cp (----- + epsilon - -----) rho
              4                4

Solve[% == 0, delta]//Flatten
      4 epsilon - 3 gamma
      {delta -> -----}
              3

```

### Using the $\eta$ Parameter

Alkhalifah and Tsvankin (1994) introduced the parameter

$$\eta = \frac{\epsilon - \delta}{1 + 2\delta}$$

to facilitate time processing in transversely isotropic media. In the package, we offer routines to trade  $\epsilon$  for  $\eta$  and *vice versa*:

```

EliminateEpsilon[epsilon]
      delta + eta + 2 delta eta

EliminateEta[%]
      epsilon

EliminateEpsilon[(epsilon-delta)/(1 + 2 delta)]
      eta

```

### Love Notation Example

You can convert Love notation expressions to Voight notation and then proceed as above:

```

EliminateLove[F + L]
      C13 + C44

```

ConvertCij[%]

$$cp^2 (\text{deltaTilde} + f) \text{ rho}$$

EliminateDeltaTilde[%]

$$cp^2 f \text{ Sqrt}\left[\frac{2 \text{ delta} + f}{f}\right] \text{ rho}$$

EliminateF[%]

$$(cp^2 - cs^2) \text{ Sqrt}\left[\frac{cp^2 - cs^2 + 2 cp^2 \text{ delta}}{cp^2 - cs^2}\right] \text{ rho}$$

### Kuprazde Notation Example

Again, you can convert Kuprazde notation expressions to Cij notation and then proceed as above:

EliminateKuprazde[C2]

$$C11 - C66$$

ConvertCij[%]

$$cp^2 (2 \text{ epsilon} + f - 2 \text{ gamma} + 2 f \text{ gamma}) \text{ rho}$$

EliminateF[%]

$$(cp^2 - cs^2 + 2 cp^2 \text{ epsilon} - 2 cs^2 \text{ gamma}) \text{ rho}$$

Collect[%, {cp, cs, rho}]

$$cp^2 (1 + 2 \text{ epsilon}) \text{ rho} + cs^2 (-1 - 2 \text{ gamma}) \text{ rho}$$

### Hooke Notation Example

EliminateHooke[C3322 + 2 C3223]

$$C11$$

## Converting from Thomsen to Love or Voight Notation

Problem: Verify that the expression

$$(1 + F)(-1 + F + 2 L)/(2 - 2 L)$$

is equal to Thomsen's  $\delta$  when the Love parameter  $C = 1$ :

`ConvertThomsen[delta] /. C -> 1`

$$\frac{(1 + F)(-1 + F + 2 L)}{2(1 - L)}$$

If we want the result in terms of the  $C_{ij}$ :

`EliminateLove[%]`

$$\frac{(1 + C13)(-1 + C13 + 2 C44)}{2 - 2 C44}$$

## A Final Realistic Example

Despite the general utility of *Mathematica*, and the specific utility of the package introduced here, real examples often require the skillful intervention of the scientist. Consider the expression in equation (6) of Tsvankin (1994):

$$\text{expr} = ((C11+C44) \text{Sin}[\text{theta}]^2 + (C33 + C44) \text{Cos}[\text{theta}]^2 + \text{Sqrt}[\left((C11 - C44) \text{Sin}[\text{theta}]^2 - (C33 - C44) \text{Cos}[\text{theta}]^2\right)^2 + 4(C13 + C44)^2 \text{Sin}[\text{theta}]^2 \text{Cos}[\text{theta}]^2]) / (2 C33);$$

Converting to Thomsen parameters gives:

$$\begin{aligned} \text{exprThom} = & \text{EliminateDeltaTilde@ConvertCij}[\text{expr}] \\ & \frac{cp}{2} (2 - f) \rho \text{Cos}[\text{theta}]^2 + \\ & \frac{cp}{2} (2 + 2 \text{epsilon} - f) \rho \text{Sin}[\text{theta}]^2 + \\ & \frac{\text{Sqrt}[cp \rho (\text{epsilon} - \text{epsilon} \text{Cos}[2 \text{theta}] - \end{aligned}$$

$$\frac{f \cos^2(2\theta) + 4 c_p^4 f (2\delta + f) \rho^2 \cos^2(\theta) \sin^2(\theta)}{(2 c_p^2 \rho)^2}$$

We can see the common factor of  $\rho c_p^2$  even if *Mathematica* cannot. Rather than wasting time trying to get *Mathematica* to recognize this factor, it is better to force it out directly. Here is one way:

```
ratiosq = Limit[exprThom, rho -> 1/cp^2]
((2 - f) Cos[theta]^2 + (2 + 2 epsilon - f) Sin[theta]^2 +
 Sqrt[(epsilon - epsilon Cos[2 theta] -
 f Cos[2 theta])^2 +
 4 f (2 delta + f) Cos[theta]^2 Sin[theta]^2]) / 2
```

The routines in the Thomsen package call on *Mathematica*'s *Simplify* routine; this routine has a canonical form for the trigonometric functions that does not always give what you may want. Thus, the package contains the functions *SineForm* and *CosineForm* to respectively emphasize powers of sine and powers of cosine. Sometimes none of the three forms will give exactly what you want, but often you will be able to "optimize" yourself after viewing the three possibilities.

```
ratiosq1 = SineForm[ratiosq]
1 - f/2 + epsilon Sin[theta]^2 +
 Sqrt[f^2 + 8 delta f Sin[theta]^2 -
 4 epsilon f Sin[theta]^2 + 4 epsilon^2 Sin[theta]^4 -
 8 delta f Sin[theta]^4 + 8 epsilon f Sin[theta]^4] / 2
```

To simplify further, one has to isolate the square root term:

$$\begin{aligned}
 a &= 1 - f/2 + \text{Sin}[\text{theta}]^2 \text{epsilon}; \\
 b &= (\text{ratiosq1} - a)^2 // \text{Simplify} \\
 &= (f^2 + 8 \text{delta} f \text{Sin}[\text{theta}]^2 - 4 \text{epsilon} f \text{Sin}[\text{theta}]^2 + \\
 &\quad 4 \text{epsilon}^2 \text{Sin}[\text{theta}]^4 - 8 \text{delta} f \text{Sin}[\text{theta}]^4 + \\
 &\quad 8 \text{epsilon} f \text{Sin}[\text{theta}]^4) / 4
 \end{aligned}$$

$$\begin{aligned}
 b1 &= \text{Collect}[b, \{\text{epsilon}, \text{delta}, f\}] \\
 &= \frac{f^2}{4} + \text{epsilon} \text{Sin}[\text{theta}]^2 + \\
 &\quad \text{delta} f (2 \text{Sin}[\text{theta}]^2 - 2 \text{Sin}[\text{theta}]^4) + \\
 &\quad \text{epsilon} f (-\text{Sin}[\text{theta}]^2 + 2 \text{Sin}[\text{theta}]^4)
 \end{aligned}$$

Finally, look at the weak limit:

$$\begin{aligned}
 \text{weakratiosq} &= \text{ConvertWeak}[\text{ratiosq1}] // \text{SineForm} \\
 &= 1 + 2 \text{delta} \text{Sin}[\text{theta}]^2 - 2 \text{delta} \text{Sin}[\text{theta}]^4 + \\
 &\quad 2 \text{epsilon} \text{Sin}[\text{theta}]^4
 \end{aligned}$$

$$\begin{aligned}
 &\text{Collect}[\text{weakratiosq}, \{\text{epsilon}, \text{delta}\}] \\
 &= 1 + 2 \text{epsilon} \text{Sin}[\text{theta}]^4 + \\
 &\quad \text{delta} (2 \text{Sin}[\text{theta}]^2 - 2 \text{Sin}[\text{theta}]^4) \\
 & // \text{Simplify} \\
 &= 1 + 2 \text{epsilon} \text{Sin}[\text{theta}]^4 + \frac{\text{delta} \text{Sin}[2 \text{theta}]^2}{2}
 \end{aligned}$$

## REFERENCES

- Alkhalifah, Tariq, & Tsvankin, Ilya. 1994 (May). Velocity analysis for transversely isotropic media. *Pages 7–40 of: Bleistein, Norman, et al. (eds), Project review.* Center for Wave Phenomena, Golden, Colorado.
- Kuprazde, V. D., Gegelia, T. G., Basheleishvili, M. O., & Burchuladze, T. V. 1976. *Three-dimensional problems of the mathematical theory of elasticity and thermoelasticity.* North-Holland Series in Applied Mathematics and Mechanics, vol. 25. Amsterdam: North-Holland Pub. Co.
- Musgrave, M. J. P. 1970. *Crystal acoustics.* San Francisco: Holden Day.
- Schoenberg, Michael, & Sayers, Colin M. 1995. Seismic anisotropy of fractured rock. *Geophysics*, **60**(1), 204–211.
- Thomsen, Leon. 1986. Weak elastic anisotropy. *Geophysics*, **51**(10), 1954–1966.
- Tsvankin, Ilya. 1994 (May). P-wave signatures and parametrization of transversely isotropic media: An overview. *Pages 349–384 of: Bleistein, Norman, et al. (eds), Project review.* Center for Wave Phenomena, Golden, Colorado.