

Ross J. Angel*, Javier Gonzalez-Platas and Matteo Alvaro

EosFit7c and a Fortran module (library) for equation of state calculations

Abstract: The relationship between linear elasticity theory of solids and their equations of state (EoS) is reviewed, along with the commonly-used types of isothermal EoS, thermal expansion models, and P - V - T EoS. A new console program, EosFit7c, is presented. It performs EoS calculations and fitting for both volume and linear isothermal data, isobaric data and P - T data. Linear data is handled by cubing the quantities and treating them as volumes in all EoS formulations. Least-squares fitting of EoS to data incorporates the option to weight the fit with the measurement uncertainties in P , V and T simultaneously. The EosFit7c program is built with a new library of subroutines for EoS calculations and manipulation, written in Fortran. The library has been incorporated as a module, `cfml_eos`, in the publicly-available CrysFML library. The module handles Murnaghan, Tait, Birch-Murnaghan, Vinet, and Natural Strain EoS. For P - V - T calculations any of these isothermal EoS can be combined with a variety of published thermal expansion models, including a model of thermal pressure. The entire library has been revalidated against other software and against an *ab-initio* re-derivation of the EoS, which identified a number of small errors in published formulae for some EoS.

Keywords: equations of state, thermal expansion, EosFit

*Corresponding Author: **Ross J. Angel**, Dipartimento di Geoscienze, Università di Padova, Via G. Gradenigo 6, Padova, 35131, Italy, e-mail: rossjohnangel@gmail.com

Matteo Alvaro: Dipartimento di Geoscienze, Università di Padova, Via G. Gradenigo 6, Padova, 35131, Italy

Javier Gonzalez-Platas: Departamento de Física Fundamental II. Servicio de Difracción de Rayos X, Universidad de La Laguna, La Laguna, Tenerife 38206, Spain

Introduction

An equation of state (EoS) describes how the volume or density of a material varies with changes in pressure and temperature. It also defines how some of the elastic properties of the material change in response to compression

and expansion. Equations of state therefore provide not only fundamental thermodynamic data that is required, for example, for the calculation of equilibrium phase diagrams (e.g. Berman, 1988; Gottschalk, 1997; Holland & Powell, 1998; Stixrude & Lithgow-Bertelloni, 2005; Holland & Powell, 2011) but also give insights in to the details of interatomic interactions within the solid state, as it is these that resist the externally-applied compressive stresses (e.g. Brown, Klages & Skowron, 2003; Zhao, Ross & Angel, 2004; Fabbiani & Pulham, 2006) and control the dynamics that lead to thermal expansion (e.g. Willis & Pryor, 1975).

The majority of equation of state studies measure the volume or unit-cell parameter variations with pressure (and/or temperature), with the aim of deriving elastic parameters that are derivatives of the data. For example, the bulk modulus that defines the instantaneous volume variation with pressure is $K = -V(\partial P/\partial V)$. The bulk moduli of most inorganic solids range from ~40 to 400 GPa. Therefore the volume changes induced by compression over the relatively-easily accessible experimental pressure range (0–10 GPa) are only larger than the experimental uncertainties by 1 or 2 orders of magnitude. The need to obtain parameters that are derivatives of the original data, in combination with the small data range, clearly makes the reliable determination of the parameters difficult to achieve. Strong correlations between parameters, which include the pressure derivatives of K , exacerbate the problems. These issues were previously addressed by the development of the EosFit program (Angel, 2000a) which provided the capability of fitting P - V (and unit-cell parameter data) with various EoS, with the correct algebra and with options to fully weight the data with the measurement uncertainties. The Eosfit program includes utilities to do further EoS calculations beyond just fitting EoS parameters to P - V data.

The ready availability of a variety of spread sheets and algebraic software makes it relatively easy, in principle, to fit any EoS formulation to data. However, the complex nature of some equation of state functions has led to their algebraic forms being mis-stated in the literature. The additional possibility of errors in coding the complex equations, and the requirement to implement

specific (non-standard) weighting methods to overcome correlation problems, makes imperative the provision of a validated set of publicly-available self-consistent algorithms for EoS calculations. We have now translated the original code of EosFit to Fortran-95, revalidated it, and have built it in to a module that is part of the larger Crystallographic Fortran *Modules Library* CrysFML (Rodriguez-Carvajal & Gonzalez-Platas, 2003). This will ensure the long-term survival and availability to programmers of a validated set of easy-to-use procedures for EoS calculations.

In this paper we first briefly review the relationship between linear elasticity theory of solids and their equations of state, and then present the types of isothermal EoS, thermal expansion models, and thus P - V - T EoS that are implemented within the new EoS module of CrysFML. The handling of linear data (e.g. cell parameters) in a manner consistent with both volume EoS and linear elasticity theory is discussed, prior to a description of the implementation of these approaches in the CrysFML module. A new program, EosFit7c, that performs EoS calculations and fits P - V , V - T and P - V - T data (and linear equivalents), is presented as an example of the new features that are available in the module.

Isothermal equations of state

Basis in linear elasticity theory

The variation of the volume of a solid with hydrostatic pressure at fixed temperature is termed its ‘isothermal equation of state’. It is characterised by the bulk modulus of the material, $K = -V(\partial P/\partial V)_T$, which is a function of both temperature and pressure. For infinitesimal changes in pressure which give rise to infinitesimal changes in volume, the bulk modulus can also be defined in terms of the elastic tensor of the material by applying linear elasticity theory, or ‘Hooke’s law’. In linear elasticity it is assumed that the strains ε_i of a solid are linearly related to the magnitude of the applied stress field σ_j by the matrix equation $\varepsilon_i = s_{ij}\sigma_j$. The suffixes run from 1 to 6, with $i, j = 1, 2, 3$ referring to normal stresses or strains along orthogonal axes, and $i, j = 4, 5, 6$ referring to shear stresses or strains (e.g. Nye, 1957; Angel, Jackson, Reichmann & Speziale, 2009). The elastic properties of the material are represented by the values of the elements of the compliance matrix s_{ij} which is symmetric and can contain up to 21 independent elements for triclinic crystals, less for crystals and materials with higher symmetries. The compliance matrix is a convenient re-

presentation of the compliance tensor of the material (Nye, 1957).

Hydrostatic pressure is a special stress state in which the normal stresses are all equal ($\sigma_1 = \sigma_2 = \sigma_3 = P$) and there are no shear stresses ($\sigma_4 = \sigma_5 = \sigma_6 = 0$). Thus, at any pressure P , the strains caused by an infinitesimal increase in pressure δP can be calculated through linear elasticity theory by setting $\sigma_1 = \sigma_2 = \sigma_3 = -\delta P$. (Note that while pressure is considered to be a positive quantity, compressive stresses are by convention considered to be negative; Nye, 1957). Therefore each of the resulting strain elements is given by:

$$\varepsilon_i = (s_{i1}\sigma_1 + s_{i2}\sigma_2 + s_{i3}\sigma_3) = -(s_{i1} + s_{i2} + s_{i3})\delta P.$$

The sum of the three normal strains is, in the infinitesimal limit, equal to the fractional change in the volume $\partial V/V$, thus:

$$-\partial V/V = (\varepsilon_1 + \varepsilon_2 + \varepsilon_3) = [s_{11} + s_{22} + s_{33} + 2(s_{12} + s_{13} + s_{23})]\delta P.$$

Re-arrangement of this last equation shows that the bulk modulus for hydrostatic compression at any pressure is defined by six of the elements of the compressibility matrix at that same pressure:

$$K = -V\delta P/\delta V = [s_{11} + s_{22} + s_{33} + 2(s_{12} + s_{13} + s_{23})]^{-1}.$$

This bulk modulus for hydrostatic compression of a solid, whether a powder or single crystal, is therefore equal to the Reuss bound on the bulk modulus of a polycrystal made of the same material where it represents the volume response when every constituent grain is subject to the same stress.

While linear elasticity thus defines the bulk modulus of a material under hydrostatic compression at any pressure, it cannot define an equation of state which describes the large (finite) changes in volume due to large (finite) changes in pressure. In this sense an equation of state is an extension of linear elasticity; although normally defined in terms of the volume variation with pressure, it can also be seen as a definition of the variation of bulk modulus with pressure. Because there is no absolute thermodynamic basis for specifying how the bulk modulus K varies with pressure, all EoS that have been developed and are in widespread use are based upon a number of assumptions (e.g. Anderson, 1995; Duffy & Wang, 1998; Holzapfel, 2001). The validity of such assumptions can only be judged in terms of whether the derived EoS reproduces experimental data for volume or elasticity. For materials that do not exhibit phase transi-

tions, isothermal equations of state are usually parameterized in terms of the values of the bulk modulus and its pressure derivatives, $K' = \partial K / \partial P$ and $K'' = \partial^2 K / \partial P^2$, evaluated at a reference pressure, normally taken as zero pressure. If the material undergoes a structural phase transition, additional parameters are required (e.g. Tröster, Schranz & Miletich, 2002; Schranz, Tröster, Koppens-Steiner & Miletich, 2007). In order to allow for the description of the variation of the volume with temperature and pressure, we also define a reference temperature for the EoS, T_{ref} . We then denote the values of the parameters describing an equation of state at the reference temperature and pressure with two subscripts ‘‘0’’, thus:

$$K_{00} = -V_{00} (\partial P / \partial V)_{P=0, T=T_{\text{ref}}}, \quad K'_{00} = (\partial K / \partial P)_{P=0, T=T_{\text{ref}}},$$

and

$$K''_{00} = (\partial^2 K / \partial P^2)_{P=0, T=T_{\text{ref}}}.$$

Using this notation leads to the value of the room-pressure bulk modulus at some temperature T being denoted $K_{0T} = -V_{0T} (\partial P / \partial V)_{P=0, T}$, and its isothermal pressure derivatives K'_{0T} and K''_{0T} . The subscript ‘‘ T ’’ is therefore *not* to be read as indicating isothermal as opposed to adiabatic moduli, although all of the moduli discussed here are isothermal. Note that the values of adiabatic bulk moduli are typically a few % larger than isothermal moduli, by a factor $(1 + \alpha_V \gamma T)$ in which α_V is the thermal expansion coefficient, γ the Anderson-Grüneisen parameter, and T the temperature.

There are many reviews of equations of state available in the literature (e.g. Anderson, 1995; Duffy & Wang, 1998; Holzapfel, 2001). In the following sections we restrict ourselves to briefly introducing the EoS that are coded in to the CrysFML EoS module and are available to a user in the programs built upon it, including our new console program, EosFit7c. The EoS have been chosen for the library because of their widespread use and/or suitability for fitting P - V data in the absence of phase transitions.

Murnaghan

The Murnaghan EoS (Murnaghan, 1937) can be derived from the assumption that the bulk modulus varies linearly with pressure, $K_{PT} = K_{0T} + K'_{0T}P$. This results in a relationship between P and V of:

$$V_{PT} = V_{0T} \left(1 + \frac{K'_{0T}P}{K_{0T}} \right)^{-1/K'_{0T}}$$

or as:

$$P_{VT} = \frac{K_{0T}}{K'_{0T}} \left[\left(\frac{V_{0T}}{V} \right)^{K'_{0T}} - 1 \right].$$

This illustrates the attraction of the Murnaghan EoS: it is ‘invertible’ (Freund & Ingalls, 1989), which means that it can be written as either a simple function of V , or as a simple function of P . This makes it very attractive for use in thermodynamic databases and calculations in which both functions and their integrals are frequently required. However, the Murnaghan EoS only reproduces P - V data and the correct values of the room pressure bulk modulus for small compressions up to about 10% (i.e. $V/V_0 \sim 0.9$). It fails at greater compression because it has $K''_{PT} = 0$, in contrast to experimental data which indicate that most solids exhibit a small negative value for K''_{0T} of the order of $-1/K_{0T}$ (e.g. Jackson & Niesler, 1982). The obvious simple extension of the Murnaghan EoS to include an additional term in the expression for the bulk modulus, $K_{PT} = K_{0T} + K'_{0T}P + K''_{0T}P^2/2$ results in an extremely complex and impractical function for volume as a function of pressure (Freund & Ingalls, 1989).

Tait

Freund & Ingalls (1989) showed that the ‘modified Tait equation’ of Huang & Chow (1974) is a generalised form of the Murnaghan EoS which remains easily invertible:

$$P_{VT} = \frac{1}{b} \left(\left[\frac{(V_{PT}/V_{0T}) + a - 1}{a} \right]^{-1/c} - 1 \right)$$

and

$$V_{PT} = V_{0T} (1 - a(1 - (1 + bP)^{-c})).$$

The three parameters a , b , c are defined in terms of the bulk modulus and its derivatives at room pressure:

$$a = \frac{1 + K'_{0T}}{1 + K'_{0T} + K_{0T}K''_{0T}} \quad K_{0T} = \frac{1}{abc}$$

$$b = \frac{K'_{0T}}{K_{0T}} - \frac{K''_{0T}}{1 + K'_{0T}} \quad \text{and the inverse relationships:} \quad K'_{0T} = \frac{c + 1}{ac} - 1$$

$$c = \frac{1 + K'_{0T} + K_{0T}K''_{0T}}{(K'_{0T})^2 + K'_{0T} - K_{0T}K''_{0T}} \quad K''_{0T} = \frac{b}{a} (1 - a)(c + 1).$$

If $K''_{0T} = 0$, then $a = 1$, $b = \frac{K'_{0T}}{K_{0T}}$, $c = \frac{1}{K'_{0T}}$, and the equation is reduced to the Murnaghan form (Freund & Ingalls, 1989). While precise values of K''_{0T} are difficult

to measure, it is clear that for most solids they are not zero. Holland & Powell (2011) therefore introduced a ‘truncation’ of the Tait equation by setting $K''_{OT} = \frac{-K'_{OT}}{K_{OT}}$.

In the EoS module of CrysFML this estimate of K''_{OT} is named the ‘3rd-order’ form, with the 4th-order form including a refineable K''_{OT} . There is no rationale for a 2nd-order formulation, but for completeness we can define a 2nd-order form as having $K'_{OT} = 4$, and thus $K''_{OT} = \frac{-4}{K_{OT}}$.

Thus all orders of the Tait equation as implemented in the EoS module of CrysFML have $K''_{OT} \neq 0$ and therefore fit P - V data of solids substantially better than the Murnaghan EoS.

Because the Tait EoS is invertible, the expressions for the bulk modulus and its pressure derivatives as a function of pressure can be obtained directly (since the parameters a , b , and c are constants) by differentiation with respect to pressure of the expression for the volume:

$$K_{PT} = K_{OT} \left(\frac{V_{PT}}{V_{OT}} \right) (1 + bP)^{(c+1)},$$

or:

$$K_{PT} = K_{OT} (1 - a(1 - (1 + bP)^{-c})) (1 + bP)^{(c+1)}.$$

Further differentiation with respect to pressure leads to:

$$K'_{PT} = (K'_{OT} + 1)[(1 + bP)^c(1 - a) + a] - 1$$

Setting $a = 1$ (and thus $K''_{OT} = 0$) in these equations leads to $K_{PT} = K_{OT} + K'_{OT}P$ and $K'_{PT} = K'_{OT}$ as required for the Murnaghan EoS.

Birch-Murnaghan

This ‘finite strain EoS’ is derived (Birch, 1947) from the assumption that the strain energy of a solid undergoing compression can be expressed as a Taylor series in the finite Eulerian strain, $f_E = [(V_{OT}/V_{PT})^{2/3} - 1]/2$. Expansion to 4th-order in the strain yields an EoS:

$$P = 3K_{OT}f_E(1 + 2f_E)^{5/2} \left(1 + \frac{3}{2}(K'_{OT} - 4)f_E + \frac{3}{2} \left(K_{OT}K''_{OT} + (K'_{OT} - 4)(K'_{OT} - 3) + \frac{35}{9} \right) f_E^2 \right).$$

The normalised pressure for the Birch-Murnaghan EoS is defined as (Stacey, Brennan & Irvine, 1981):

$$F = \frac{P}{3f_E(1 + 2f_E)^{5/2}}.$$

This allows the Birch-Murnaghan EoS to be expressed as a simple polynomial:

$$F = K_{OT} \left(1 + \frac{3}{2}(K'_{OT} - 4)f_E + \frac{3}{2} \left(K_{OT}K''_{OT} + (K'_{OT} - 4)(K'_{OT} - 3) + \frac{35}{9} \right) f_E^2 \right).$$

If this EoS is truncated at 2nd-order in the energy, then the coefficient of f_E must be identical to zero, which requires that K'_{OT} has the fixed value of 4 (higher-order terms are ignored). The 3rd-order truncation, in which the coefficient of f_E^2 is set to zero yields a three-parameter EoS (with V_{OT} , K_{OT} and K'_{OT}) with an implied value of K''_{OT} given by (Anderson, 1995):

$$K''_{OT} = \frac{-1}{K_{OT}} \left((3 - K'_{OT})(4 - K'_{OT}) + \frac{35}{9} \right).$$

The expressions for the bulk modulus and its first derivative for the 3rd-order Birch-Murnaghan EoS are therefore (Angel, 2000b):

$$K_{PT} = K_{OT}(1 + 2f_E)^{5/2} \left(1 + (3K'_{OT} - 5)f_E + \frac{27}{2}(K'_{OT} - 4)f_E^2 \right)$$

$$K'_{PT} = \frac{K_{OT}}{K_{PT}}(1 + 2f_E)^{5/2}$$

$$\times \left(K'_{OT} + \left(16K'_{OT} - \frac{143}{3} \right) f_E + \frac{81}{2}(K'_{OT} - 4)f_E^2 \right).$$

These are equivalent to the expressions given by Birch (1986) in his appendix 1, and by Anderson (1995) in his equations (6.52) to (6.55), except for a typographical error of K' for K'' in his equation (6.53). The expressions given by Stacey et al. (1981) are correct except that for K' , which is truncated at f_E rather than after the f_E^2 which is required for the expression to be exact. Expressions for the 2nd-order Birch-Murnaghan EoS can be obtained by setting $K'_{OT} = 4$ in all of the above.

We find that fits of the Birch-Murnaghan and Tait EoS to P - V data normally yield parameters that are indistinguishable within the uncertainties, but with the Birch-Murnaghan EoS typically having marginally better formal measures of statistical fit. The 4th-order fits of the two equations are normally statistically and numerically indistinguishable.

Natural strain

Poirier & Tarantola (1998) developed an EoS based upon the ‘natural’ or ‘Hencky’ measure of linear strain f_N

which, for hydrostatic compression, may be written as $f_N = 1/3 \ln(V_0/V)$. Note that this definition has been inverted with respect to that used in Angel (2000a) in order to obtain positive values of f_N on compression. This yields a pressure-volume relationship expanded to 4th-order in strain of:

$$P = 3K_{OT} \left(\frac{V_{OT}}{V_{PT}} \right) f_N \left[1 + \frac{3}{2} (K'_{OT} - 2) f_N + \frac{3}{2} \left(K_{OT} K''_{OT} + 1 + (K'_{OT} - 2) + (K'_{OT} - 2)^2 \right) f_N^2 \right]$$

which can also be written as in terms of a normalized pressure $F = \frac{P}{3 \left(\frac{V_{OT}}{V_{PT}} \right) f_N}$ as:

$$P = 3K_{OT} \left(\frac{V_{OT}}{V_{PT}} \right) f_N [1 + af_N + bf_N^2]$$

with $a = \frac{3}{2} (K'_{OT} - 2)$ and

$$b = \frac{3}{2} \left[1 + K_{OT} K''_{OT} + (K'_{OT} - 2) + (K'_{OT} - 2)^2 \right].$$

Truncation of this ‘‘Natural strain’’ EoS at 2nd-order in the strain is obtained by setting $a = b = 0$ and it implies a value of $K'_{OT} = 2$, different from that of the 2nd-order Birch-Murnaghan EoS. For truncation at 3rd-order in the strain, $a \neq 0, b = 0$, and the implied value of K''_{OT} is given by:

$$K''_{OT} = \frac{-1}{K_{OT}} \left[1 + (K'_{OT} - 2) + (K'_{OT} - 2)^2 \right].$$

This value for K''_{OT} is normally substantially larger than that implied by the truncation of the 3rd-order Birch-Murnaghan EoS, and this often results in a significantly poorer fit of P - V data. The bulk modulus and its derivatives for all orders of this EoS are:

$$K_{PT} = 3K_{OT} \left(\frac{V_{OT}}{V_{PT}} \right) \left(\frac{1}{3} + (K'_{OT} - 1) f_N + (a + b) f_N^2 + b f_N^3 \right)$$

$$K'_{PT} = \left(\frac{K_{OT}}{K_{PT}} \right) \left(\frac{V_{OT}}{V_{PT}} \right) [K'_{OT} + (4a + 2b + 3) f_N + 3(a + 2b) f_N^2 + 3b f_N^3]$$

$$K''_{PT} = \frac{K_{OT} V_{OT}}{K_{PT}^2 V_{PT}} \left[\frac{1}{3} (4a + 2b + 3) + 2(a + 2b) f_N + 3b f_N^2 - K'_{OT} [(K'_{OT} - 1) + 2(a + b) f_N + 3b f_N^2] \right].$$

Vinet

The finite-strain EoS do not accurately represent the volume variation of most solids under very high compression ($V_{PT}/V_{OT} < 0.6$), so Vinet, Ferrante, Rose & Smith (1986) and Vinet, Ferrante, Smith & Rose (1987) derived an EoS from a generalised inter-atomic potential. Following Schlosser & Ferrante (1988), the expression for pressure in the Vinet EoS is:

$$P = K_{OT} \frac{3f_V}{(1 - f_V)^2} \exp(\eta f_V)$$

with

$$f_V = 1 - \left(\frac{V_{PT}}{V_{OT}} \right)^{1/3} \quad \text{and} \quad \eta = \frac{3}{2} (K'_{OT} - 1).$$

This definition of f_V means that $f_V = 0$ at $P = 0$, and that f_V increases as a positive quantity with increasing pressure and compression. It therefore follows the conventions in the definition of magnitude and sign of Eulerian finite strain. This is a change from the implementation in previous versions of EosFit (Angel, 2000a). The presence of K'_{OT} as a refineable parameter also leads, by comparison with other finite strain EoS, to naming this a ‘3rd-order’ EoS.

In order to obtain a form of the EoS with which to construct plots of ‘normalised pressure’ against the strain f_V , a normalised pressure has previously been defined as $F_V = \ln \left(\frac{P(1 - f_V)^2}{3f_V} \right)$ (Vinet et al., 1986; Vinet et al., 1987; Schlosser & Ferrante, 1988). Then

$F_V = \ln(K_{OT}) + \frac{3}{2} (K'_{OT} - 1) f_V$, which should be linear in f_V with slope of $\frac{3}{2} (K'_{OT} - 1)$. But the intercept at $f_V = 0$ is $\ln(K_{OT})$. Therefore in the EoS module of CrysFML we implement

$F_V = \frac{P(1 - f_V)^2}{3f_V}$, for which $F_V = K_{OT} \exp(\eta f_V)$.

The y -axis intercept of a plot of F_V against f_V is thus K_{OT} , and it will give a horizontal line for a 2nd-order EoS with $K'_{OT} = 1$. A 3rd-order EoS with $K'_{OT} > 1$ will have a curved line, with increasing gradient with increasing f_V . For reasonable values of K'_{OT} the curvature is very slight. The slope at any point is $\frac{\partial F_V}{\partial f_V} = K_{OT} \eta \exp(\eta f_V)$, so the initial slope is $\frac{3K_{OT}}{2} (K'_{OT} - 1)$, a form entirely analogous to f - F plots of the Birch-Murnaghan EoS.

There is no theoretical basis for truncation of the Vinet EoS to lower order, although it yields an implied val-

ue for K'_{0T} of 1. The implied value of K''_{0T} for the Vinet EoS is given by Jeanloz (1988) as:

$$K''_{0T} = \frac{-1}{K_{0T}} \left[\left(\frac{K'_{0T}}{2} \right)^2 + \left(\frac{K'_{0T}}{2} \right) - \left(\frac{19}{36} \right) \right].$$

Expansions of the Vinet EoS to include a refineable K''_{0T} have been proposed but are not required to fit most experimental data of simple solids in the absence of phase transitions.

Thermal expansion

The volume thermal expansion of a material is defined as $\alpha(T) = V^{-1}(\partial V/\partial T)_p$. Integration of this expression yields the volume variation with temperature at constant pressure:

$$V_{0T} = V_{00} \exp \int_{T_{\text{ref}}}^T \alpha(T) dT.$$

Where T_{ref} is a reference temperature at which the volume is V_{00} .

The only thermodynamic constraints on the form of the function for $\alpha(T)$ are that $\alpha(T) = \partial\alpha/\partial T = 0$ at absolute zero. Consequently many different forms have been proposed in the literature, and several have become widely-used, for example in thermodynamic databases (e.g. Berman, 1988; Gottschalk, 1997; Holland & Powell, 1998; Holland & Powell, 2011). Some simple formulations that describe thermal expansion at high temperatures very well do not include the low- T saturation, but are widely used and are perfectly adequate for thermodynamic databases. On the other hand, some equations explicitly handle the saturation in thermal expansion as the temperature drops towards absolute zero, but these often result in unphysical values of thermal expansion at high temperatures in excess of 1000 K, where experiments indicate that $\alpha(T)$ increases approximately linearly with temperature. Following the reviews of various thermal expansion expressions based on analysis and fitting of data (Tribaudino, Angel, Camara, Nestola, Pasqual & Margiolaki, 2010; Tribaudino, Bruno, Nestola, Pasqual & Angel, 2011; Kroll, Kirfel, Heinemann & Barbier, 2012), we have implemented the following thermal expansion equations in the EoS module of CrysFML.

Berman equation

At the lowest level of approximation $\alpha(T)$ can be considered a constant, in which case integration yields

$\ln(V_{0T}/V_{00}) = \alpha_0(T - T_{\text{ref}})$, or $V_{0T} = V_{00} \exp[\alpha_0(T - T_{\text{ref}})]$. Truncation to first order in the expansion of the logarithmic terms results in the expression $V_{0T} = V_{00}(1 + \alpha_0(T - T_{\text{ref}}))$. Strictly, this truncation leads to a slightly varying thermal expansion coefficient because re-differentiation of the truncated equation leads to $\frac{\partial V_{0T}}{\partial T} = V_{00}\alpha_0$, which means that $\alpha = \frac{V_{00}}{V_{0T}}\alpha_0$. Berman (1988) proposed a simple extension to accommodate non-linear thermal expansion:

$$V_{0T} = V_{00}(1 + \alpha_0(T - T_{\text{ref}}) + \frac{1}{2}\alpha_1(T - T_{\text{ref}})^2).$$

Differentiation yields $\frac{1}{V_{0T}} \frac{\partial V_{0T}}{\partial T} = \frac{V_{00}}{V_{0T}}[\alpha_0 + \alpha_1(T - T_{\text{ref}})]$.

Given the small changes in volume with temperature, this is approximately $\alpha \approx [\alpha_0 + \alpha_1(T - T_{\text{ref}})]$. The parameter α_0 is the thermal expansion coefficient at T_{ref} . However, this equation is not valid for low temperatures because it predicts a finite value for $\alpha(T)$ at absolute zero except for the special case of $\alpha_1 = \alpha_0/T_{\text{ref}}$.

Fei equation

Earlier versions of EoSFit (Angel, 2000a) implemented a linear variation of thermal expansion as $\alpha = \alpha_0 + \alpha_1 T$ but without the truncation of the exponential term implied by the Berman equation. Fei (1995) proposed an expansion of this expression to $\alpha = \alpha_0 + \alpha_1 T + \alpha_2 T^{-2}$ (with T in Kelvin). This leads to the high-temperature volume at zero pressure given as:

$$V_{0T} = V_{00} \exp \left(\alpha_0(T - T_{\text{ref}}) + \frac{1}{2}\alpha_1(T^2 - T_{\text{ref}}^2) - \alpha_2 \left(\frac{1}{T} - \frac{1}{T_{\text{ref}}} \right) \right).$$

With this formulation, the actual values of α_0 , α_1 and α_2 that describe a V - T curve are those at 0 K, and *not* those at T_{ref} , so their values are independent of T_{ref} . It also has the advantage that the derivative $\frac{1}{V_{0T}} \frac{\partial V_{0T}}{\partial T}$ is exactly $\alpha = \alpha_0 + \alpha_1 T + \alpha_2 T^{-2}$ at all temperatures. The disadvantage is that the full expression predicts non-physical behaviour at low temperatures because the term in T^{-2} causes the value of α to diverge towards infinity as T approaches 0 K. If $\alpha_2 = 0$ the simplified form $\alpha = \alpha_0 + \alpha_1 T$ remains mathematically valid at all temperatures although it does not yield $\alpha = 0$ at $T = 0$ K.

Modified Holland-Powell (1998) equation

Pawley, Redfern & Holland (1996) proposed a model that ensures that the thermal expansion becomes approximately constant at high temperatures:

$$V_{0T} = V_{00}(1 + \alpha_0(T - T_{\text{ref}}) - 2\alpha_1(\sqrt{T} - \sqrt{T_{\text{ref}}})) .$$

Pawley et al. (1996) used 298 K as a fixed reference temperature, but there is no need to do so. They also proposed a simplification with $\alpha_1 = 10\alpha_0$ in which case the equation becomes:

$$V_{0T} = V_{00}\left(1 + \alpha_0(T - T_{\text{ref}}) - 2(10\alpha_0(\sqrt{T} - \sqrt{T_{\text{ref}}}))\right) .$$

This is sufficient to model low-resolution datasets, but maintains the saturation in thermal expansion at high temperatures. In order to accommodate this simplification in a simple manner which also allows the use of the more general equation, we modified the Pawley et al. (1996) equation to:

$$V_{0T} = V_{00}(1 + \alpha_0(T - T_{\text{ref}}) - 2(10\alpha_0 + \alpha_1)(\sqrt{T} - \sqrt{T_{\text{ref}}})) .$$

Note that the entire term $(10\alpha_0 + \alpha_1)$ of this equation is equal to the α_1 coefficient used by Pawley et al. (1996). Thus, when α_1 is fixed at zero the whole term $(10\alpha_0 + \alpha_1)$ becomes equal to $10\alpha_0$ and the simplified equation proposed by Pawley et al. (1996) and used in Holland & Powell (1998) is obtained. In the EoS module of CrysFML,

the correct value of $\alpha = \frac{1}{V_{PT}} \frac{\partial V_{PT}}{\partial T}$ is always calculated, so

the returned values from the program will differ from those obtained by inserting the parameters in to the approximate equations for thermal expansion given by the previous authors. This equation cannot be used at low

temperatures because below $T = \left(\frac{10\alpha_0 + \alpha_1}{\alpha_0}\right)^2$ the thermal expansion becomes negative and the volume is predicted to increase with decreasing temperature. If $\alpha_1 = 0$ this limiting temperature is 100 K. Note also that, even

when $\alpha_1 = 0$, $\alpha = \frac{1}{V_{0T}} \frac{\partial V_{0T}}{\partial T}$ at T_{ref} is *not* equal to α_0 .

Salje equation

The saturation of thermal expansion at low temperature can be addressed using an equation of the form suggested by Salje, Wruck and Thomas (1991):

$$V_{0T} = [p_0 + p_1\theta_{\text{sat}} \coth(\theta_{\text{sat}}/T)]^3$$

in which p_0 and p_1 are fitting parameters and θ_{sat} is termed the “saturation temperature”. In fact the thermal expansion coefficient only becomes zero below $T \approx \theta_{\text{sat}}/10$. The equation should only be used to describe volume variation at low temperatures, because it leads to an expression (Kroll et al., 2012) for the thermal expansion coefficient of:

$$\alpha(T) = \frac{3p_1\theta_{\text{sat}}^2}{V_{0T}^{1/3}} \left[\frac{\coth^2(\theta_{\text{sat}}/T) - 1}{T^2} \right] .$$

At moderate temperatures, above about $3\theta_{\text{sat}}$ (θ_{sat} is typically 200–500 K), the thermal expansion becomes almost independent of temperature, which is not observed for most materials. The value of p_1 is thus approximately 3 times the high-temperature value of α . At $T = 0$ K the coth function has the value of unity, so the volume at $T = 0$ is $[p_0 + p_1\theta_{\text{sat}}]^3$. Therefore, in order for the parameter V_{00} to have the value of the volume at the temperature $T_{\text{ref}} = 0$ K, we re-write the equation as:

$$V_{0T} = [V_{00}^{1/3} + p_1\theta_{\text{sat}} [\coth(\theta_{\text{sat}}/T) - 1]]^3 .$$

The Salje equation could also be re-written in terms of a reference temperature, but since it is only applicable to low-temperature thermal expansion, this is not implemented in the EoS module of CrysFML, which therefore sets $T_{\text{ref}} = 0$.

Kroll form of Holland & Powell (2011)

From the preceding discussion, it is clear that simple expressions in temperature for the thermal expansion coefficient do not simultaneously meet the thermodynamic requirement $\alpha(T) = \partial\alpha/\partial T = 0$ at $T = 0$ and match the experimental observation that $\alpha(T)$ becomes linear with temperature at high temperatures. The solution is to use an equation for thermal expansion that explicitly relates the volume to lattice energy of the material. Kroll et al. (2012) show that the Kumar (2003, and references therein) version of this approach is superior to that of Suzuki (1975) and Wallace (1972). The disadvantage of the Kumar thermal expansion is that it is referenced to absolute zero. Holland & Powell (2011) developed a similar function that is expressed in terms of parameters at a reference temperature. Although this formulation is not quite as robust in extrapolation as the Kumar expression when the underlying data are sparse, when the data are sufficient it produces fits and parameters that are indistinguishable from those of the Kumar equation (Kroll et al.,

2012). Tribaudino et al. (2011) and Kroll et al. (2012) give different but equivalent expressions, of which the latter is perhaps clearer:

$$V_{0T} = V_{00} \left[-K'_{00} + (1 + K'_{00}) \left(1 - \frac{K'_{00}(K'_{00} + 2)}{(K'_{00} + 1)} A \right)^B \right].$$

The two expressions for A and B are:

$$A = \alpha_{0,T_{\text{ref}}} \left(\frac{\theta_E}{\xi_0} \right) \left(\frac{1}{\exp(\theta_E/T) - 1} - \frac{1}{\exp(\theta_E/T_{\text{ref}}) - 1} \right)$$

$$B = -1/K'_{00}(K'_{00} + 2).$$

In the expression for A , the factor

$$\xi_0 = \frac{(\theta_E/T_{\text{ref}})^2 \exp(\theta_E/T_{\text{ref}})}{(\exp(\theta_E/T_{\text{ref}}) - 1)^2}.$$

The Einstein temperature, θ_E , in the coth functions provides the saturation in α at low temperatures, below $T \approx \theta_E/10$. The value of α_0 is the thermal expansion coefficient at T_{ref} . The value of θ_E can be approximated from the molar standard state entropy (e.g. Holland & Powell, 2011), but tests indicate that its precise value is not critical for the correct description of the volume variation with temperature and, consequently, it normally cannot be reliably determined by refinement to data.

***P-V-T* equations of state**

Equations to describe the variation of volume with both pressure and temperature can be developed by combining any thermal expansion model with any isothermal equation of state, and a model of the variation of bulk modulus with temperature at room pressure, $\partial K_{0T}/\partial T$.

The simplest approach of assuming a linear variation of K_0 with temperature, so $\partial K_{0T}/\partial T$ is constant, has been justified at high temperatures by direct measurements of the bulk moduli of many materials by elasticity measurements (e.g. as summarised in Anderson, 1995). In combination with an expression for thermal expansion that allows $\partial\alpha/\partial T \neq 0$, and an isothermal EoS with $K'_{00} \neq 0$, this approach includes all second derivatives of the volume with respect to the intensive variables P and T , and is thus algebraically internally consistent. This is because $\partial K_{0T}/\partial T$ provides the cross-derivative $\partial^2 V/\partial T \partial P$; thus if $\partial K_{0T}/\partial T = 0$ then α does not change with pressure.

However, as Hellfrich & Connolly (2009) pointed out, this formulation with a constant $\partial K_{0T}/\partial T$ often leads to the prediction of non-physical negative thermal expansion

coefficients at reasonably modest pressures for a large number of materials. Hellfrich & Connolly (2009) used the definition of the Anderson-Gruneisen parameter δ , to propose an alternative description of the variation of K_{0T} with temperature:

$$K_{0T} = K_{00} \exp \left[-\delta \int_{T_{\text{ref}}}^T \alpha(T) dT \right].$$

This can be re-arranged as: $K_{0T} = K_{00} \left[\exp \left(\int_{T_{\text{ref}}}^T \alpha(T) dT \right) \right]^{-\delta}$

$$\text{and thus } K_{0T} = K_{00} \left[\frac{V_{00}}{V_{0T}} \right]^{\delta}.$$

Note that δ is approximately equal to K'_{00} , so as a first approximation P - V - T data can be handled without any additional parameters. An alternative approach that yields indistinguishable P - V - T relationships and also avoids negative values of thermal expansion, is the concept of thermal pressure.

Thermal pressure

The idea of thermal pressure (e.g. Anderson, 1995) is that the total pressure at a given V and T can be expressed as the sum of two terms:

$$P(V, T) = P(V, T_{\text{ref}}) + Pth(T).$$

The function $P(V, T_{\text{ref}})$ is the isothermal equation of state for the material at the reference temperature, but using the ‘observed’ volume from P and T . The thermal-pressure function $Pth(T)$ is the pressure that would be created by increasing the temperature from T_{ref} to T at constant volume at room pressure. The thermal pressure at T_{ref} is thus zero, so at T_{ref} the thermal-pressure EoS reduces to the isothermal EoS. The thermal pressure at other temperatures clearly depends on the bulk modulus. The inverse problem of determining V at a given P and T consists of calculating the $Pth(T)$, and then solving the isothermal EoS at T_{ref} to find V for an ‘effective pressure’ equal to $P(V, T_{\text{ref}}) = P(V, T) - Pth(T)$.

The Holland & Powell (2011) expression for the thermal pressure is incorporated in to the EoS module of CrysFML. It employs an Einstein function:

$$Pth = \alpha_0 K_{00} \left(\frac{\theta_E}{\xi_0} \right) \left(\frac{1}{\exp(\theta_E/T) - 1} - \frac{1}{\exp(\theta_E/T_{\text{ref}}) - 1} \right)$$

with ξ_0 being the same as in the thermal expansion equation of Kroll et al. (2012). This thermal-pressure model has the properties that the product αK becomes

constant at high temperatures while it decreases to zero at low temperatures. This means that both the bulk modulus K_{0T} and the thermal expansion become constant at low temperatures, and both have an approximately linear variation with temperature above θ_E . The exact expressions for thermal expansion and bulk modulus as a function of temperature depend on the choice of isothermal equation of state, but at T_{ref} and zero pressure $\alpha = \alpha_0$ (Holland & Powell, 2011).

Linear EoS

The variation of the linear dimensions of a material with temperature and pressure can be expressed in terms of the linear thermal expansion and compressibility matrices:

$$\alpha_i = \frac{1}{x_i} \left(\frac{\partial x_i}{\partial T} \right)_P \quad \text{and} \quad \beta_i = \frac{-1}{x_i} \left(\frac{\partial x_i}{\partial P} \right)_T.$$

Both of these can be converted to tensor forms (Nye, 1957; Knight, 2010) with the same factors for the shear terms ($i = 4, 5, 6$) as used for the conversion of strain between matrix elements and tensor components. The use of a negative sign in the definition of the compressibility is purely a convention to obtain positive values for the first three ($i = 1, 2, 3$) matrix elements. We have already shown that the linear strains due to a small increment in pressure are given by linear elasticity theory as:

$$\varepsilon_i = (s_{i1}\sigma_1 + s_{i2}\sigma_2 + s_{i3}\sigma_3) = -(s_{i1} + s_{i2} + s_{i3}) \delta P.$$

Given the definition of the infinitesimal normal strains as $\varepsilon_i = \partial x_i / x_i$, it follows that the linear compressibilities are also given by the sum of elements of the compliance matrix: $\beta_i = -\varepsilon_i / \delta P = (s_{i1} + s_{i2} + s_{i3})$. As for the volume variation with pressure and temperature, there is however no absolute thermodynamic basis for specifying the variation in individual distances or cell parameters of a crystal with P and T because this involves finite strains. The only constraint that we can apply is that the sum of the normal strains must be equal to the volume strain in the infinitesimal limit. As a consequence:

$$\alpha_1 + \alpha_2 + \alpha_3 = \frac{1}{V} \left(\frac{\partial V}{\partial T} \right)_P = \alpha_V \quad \text{and}$$

$$\beta_1 + \beta_2 + \beta_3 = \frac{-1}{V} \left(\frac{\partial V}{\partial P} \right)_T = \frac{1}{K}.$$

The only way to ensure such consistency between descriptions of volume variation and variation in individual

cell parameters or other distances in the same material is to use the same equations as used for the volume variation (Angel, 2000a). For linear quantities a , the quantity is cubed and then treated as a volume in the same equations described above for the volume variation with P and T . It is clear that for cubic or isotropic materials a description of the volume variation, $V = a^3$, will lead to values of $\alpha_V = 3\alpha_a$ and $\beta_V = 3\beta_a$. The same is true for all other symmetries, except that the values of cubed lengths are not the true volume. The EoS module of CrysFML handles these inter-conversions of parameter values internally and always returns the thermal expansion

in the linear case as $\alpha_i = \frac{1}{x_i} \left(\frac{\partial x_i}{\partial T} \right)_P$. For linear compression we can define a linear modulus as

$$M_i = -x_i \left(\frac{\partial P}{\partial x_i} \right)_T = \frac{1}{\beta_i} = (s_{i1} + s_{i2} + s_{i3})^{-1}.$$

Thus the relationship between the linear moduli and bulk moduli is

$$K = \left(\frac{1}{M_1} + \frac{1}{M_2} + \frac{1}{M_3} \right)^{-1}.$$

For isotropic and cubic materials this reduces to $K = \frac{M_a}{3}$ and thus $\beta_V = 3\beta_a$, as re-

quired. The linear moduli and their pressure derivatives defined in this way (which correspond to the elastic compliances) have numerical values three times those of the corresponding volume bulk modulus. Again, the EoS module of CrysFML handles the inter-conversions internally and always returns the correct values of the moduli M , and its derivatives for the linear case. Note that this is a change from earlier versions of EosFit (Angel, 2000a) which returned volume-like values for the linear moduli. The consequence of using these conventions is that the corresponding values of finite strain and normalised pressure are those for volume, not for linear quantities. Therefore f - F plots of linear data yield values of intercepts and slopes of lines that correspond to the values of volume bulk moduli and their derivatives, not those of the linear moduli. When an f - F plot is used to display linear data, the moduli derived from its intercept and slope must therefore be multiplied by 3 to obtain the linear moduli and its pressure derivatives.

Implementation in CrysFML

CrysFML is a set of Fortran modules that can be ‘used’ (in the Fortran sense) by external programs for performing crystallographic and related calculations (Rodriguez-Carvajal & Gonzalez-Platas, 2003). Each module contains subroutines, functions and data structures (as user-defined types) for certain kinds of crystallographic operations, for example symmetry generation or structure-fac-

tor calculations. All of the modules are written in a subset of Fortran-95 that ensures that the library can be compiled and linked with a wide range of commercial and non-commercial compilers. We have written and validated a new EoS module for CrysFML called `cfml_eos` that allows the external programmer to easily write programs that can read, manipulate and fit EoS data, and perform related calculations for all of the EoS that we have described above. In order to use the library, the CrysFML code should be downloaded and compiled to a library, which must then be linked to the external program. Access to the procedures and data types defined in the EoS module is obtained through the inclusion of just a single Fortran statement:

```
USE CFML_EOS
```

The entire CrysFML has to be compiled and installed because the EoS module also makes use of other modules within CrysFML to manipulate unit-cell data and text information in files.

Manipulation of EoS and their parameters is greatly simplified for the programmer by the definition in the EoS module of a data type called `eos_type`. This data structure holds all of the parameters required to define a P - V - T equation of state, including the type of isothermal model, the type of thermal expansion model, whether it is a linear or volume EoS, and the names, values and descriptions of the parameters. Once a variable of this type has been declared in the external program, it can be initialized by the subroutine `init_eos` in the module, after which the values of any of the components can be set by the external program (see Example). If a linear EoS is specified, then all of the values in the `eos_type` variable are the linear quantities; the length, linear moduli and linear thermal expansion coefficients at the reference conditions. The `eos_type` variable can then be used in calls to functions in the `cfml_eos` module that return individual values of P , V or T at the values of the other two variables. The properties at any P , V or T of the EoS, the bulk modulus and its derivatives or the thermal expansion coefficient, at any given volume and temperature are obtained by calls to individual functions (Table 1). The subroutine `eos_cal` returns all of these values at any P and T as an array. If the EoS parameters have been obtained through fitting to data, the subroutine `eos_cal_esd` provides estimates of the uncertainties of the properties at pressure and temperature by the manipulation of the variance-covariance matrix from the parameter refinement. Subroutines are also provided to calculate the finite strain and normalized stress from the

EoS parameters (see Example) and the data. For P - V - T EoS we use the convention that normalized pressures F and finite strains f are defined isothermally in terms of V_{PT}/V_{0T} . Programming using the module is also made easier by the provision of a number of ‘utility’ routines, for example to set the implied values for an EoS, and to output the parameters (Table 1). The subroutine `deriv_partial_p` provides partial derivatives of EoS parameters with respect to pressure, which can then be used in external least-squares routines to fit EoS. For the reasons of numerical stability (see Angel, 2000a), the derivatives of pressure with respect to the parameters V_{00} , K_{00} , K'_{00} , K''_{00} and $\partial K_{00}/\partial T$ are all coded analytically. Numerical derivatives suffice for the parameters describing thermal expansion.

The use of sensible default values for the parameters of an EoS, which can be set by a call to `init_eos`, together with extensive exception trapping, make program crashes within the module exceedingly rare. If an error state occurs, for example because physically or mathematically-invalid results are calculated within the `cfml_eos` module, the `err_eos` flag is set true and an explanation is written to the `err_eos_message` character variable (Table 2), both of which can be interrogated by the external program (see Example). At the same time, the `cfml_eos` module always returns ‘reasonable’ finite values of variables when an error condition is met, and should never return undefined or infinite quantities (such as `nan`).

The handling of data is also simplified by use of two defined data types (Table 2). `Eos_data_type` can hold all of the unit-cell parameters and volume, the pressure and temperature, and all of their uncertainties for a single measurement. The `eos_data_list_type` is essentially an array of the data points, allowing the use of object-oriented style programming structures to manipulate the data. Utility routines are provided to dynamically allocate and deallocate memory to these structures, if that is required. Full details of all available data structures and subroutines are provided in the CrysFML help file provided as part of the distribution.

Data file format

Programmers can define their own data file formats, and write their own routines to populate the `eos_data_type` and `eos_data_list_type` variables in order to use the data manipulation and fitting routines within the `cfml_eos` module. But we have implemented a flexible, keyword-directed file format, and we provide the rou-

tines in the `cfml_eos` module to read the data from files. The over-riding design considerations were to allow all of the data associated with an experiment (perhaps just P and V , but possibly all of the unit-cell parameters) to be stored and read from a single file with the maximum flexibility and minimum input for the user. So there are only four keywords to provide information, as in this example:

```
TITLE KALSILITE HIGH P DATA GATTA ET AL (2011) AM MIN 96:1363
#
# Note that no volume data is given
#
SYSTEM hexagonal
TSCALE C
FORMAT 1 T P SIGP A SIGA C SIGC
25,0.000,0.004,5.16026,0.00021,8.71661,0.00021
25,0.328,0.004,5.14963,0.00020,8.70565,0.00019
25,0.565,0.004,5.14195,0.00012,8.69821,0.00013
```

TITLE allows a title to be put on the file. If the crystal system is specified with the keyword **SYSTEM**, then only the symmetry-independent unit-cell parameters have to be given. If the file is read with the `cfml_eos` module subroutine `read_eos_datafile`, that will set the remaining unit-cell parameters to correspond to the conventional setting of the crystal system. Cell parameters are not required; data consisting of only volumes can also be read, and in this case the **SYSTEM** keyword does not have to be used. The data itself can appear on one or two lines for each data ‘point’. The order of the individual data items is specified by the entries after the **FORMAT** keyword. Uncertainties in values can either be given by separate columns of values, or in the standard format with parentheses, e.g. 5.16026(21). The **TSCALE** keyword allows temperature data to be input as degrees C or F, with a default when no keyword is present of K. Data in °C and °F are immediately converted on reading to K, and the `cfml_eos` module works entirely in K.

Software validation

It has recently been noted that much published science relies on unvalidated software or algorithms (e.g. Barnes, 2010; Alden & Read, 2013; Hayden, 2013). Given that we have also identified errors in EoS in the published literature (as noted above), we record here the processes of validation of the `cfml_eos` module within CrysFML. The EosFit v5.2 program (Angel, 2000a) was previously

validated by the first author and many users independently against a variety of calculations in external programs including spread-sheets, algebraic manipulation software, and data analysis software. This led to a number of corrections and modifications to the code in the first years following its initial release. For the last decade the code has not been changed and produces results in good agreement with other programs and calculations. For the `cfml_eos` module, the entire algebra of all of the EoS was first derived again *ab-initio*, and the code confirmed against these new derivations. The results from individual sub-programs were checked against those from EosFit v5.2 where possible. For new features, including the Tait EoS and the new thermal expansion models, validation of individual routines was performed against spread-sheet calculations. The difference in coding methods and structures between Fortran and spread-sheets makes the exact duplication of errors between the two methods unlikely. Comparisons of the values of K_{PT} , K'_{PT} and K''_{PT} calculated with the analytical expressions in the code with the derivatives calculated numerically from values of V at closely-spaced values of P confirm that the code is internally consistent. The least-squares routines incorporated in to the EosFit7c program were then validated by comparing fitting results against those from version 5.2 for a wide variety of datasets. All of these tests can be repeated by users and programmers.

Implementation in EosFit7c

A new console program, named EosFit7c, has been written to perform EoS calculations and fitting with the `cfml_eos` module within CrysFML. This program provides the same utilities as the previous EosFit v5.2 program (Angel, 2000a), including the calculation of properties of an EoS, and EoS parameters can be determined with the program by fitting to P - V or P -*linear* data by the method of least-squares. Some changes have been made in the organisation of the command-line interface to reduce the number of available commands and thus make learning the program easier. There is now a single ‘input’ command with which the user can load parameters for both the compressional and thermal models. A single command returns all of the properties of the EoS at any T and P , in place of the individual commands to return individual parameter values in the previous version. The calculation of P from V and T is provided by a separate command. Output of all parameters for ranges of pressures and/or temperatures to text files allows any parameter of the EoS to be plotted with external programs.

Example program

```

use cfml_eos
implicit none
type (eos_type) :: eos           !declare eos variable
real,dimension(4) :: vals
!> set eos and parameters
call init_eos(eos)              !initialise eos variable
eos%imodel=2                    !Birch-Murnaghan
eos%iorder=3                    ! 3rd order
eos%params(1)=112.98            ! vo
eos%params(2)=37.12             ! ko
eos%params(3)=5.99              ! kprime
call set_kp_kpp_cond(eos)       ! set other default values

call write_info_eos(eos)        ! report the values

t=0.0                            ! temperature not used
do i=1,10
  call init_err_eos             ! initialises error system
  p=real(i)
  call eos_cal(p,t,eos,vals)     ! calculate all values
  call ffcals_eos(p,t,eos,fp,fs) ! calculate finite stress/strain

  if(err_eos) then              ! print warning and error messages
    write(6,'(f10.4,':':',a)')p,trim(err_eos_mess)
  else                            ! print eos values
    write(6,'(7f10.4)')p,vals(1:4),fs,fp
  endif
enddo
end

```

When uncertainties in parameters are available (e.g. from a least-squares fit to data), all calculated output values are accompanied by estimated uncertainties calculated from transformation of the variance-covariance matrix. We emphasise that these features allow the EosFit7c program to be used for EoS calculations, and it is not just limited to fitting EoS to data.

The development of the `cfml_eos` module has allowed a number of new features to be added to the EosFit7c program. The Tait isothermal equation of state has been added, along with all of the thermal expansion and thermal pressure models described above. The program reads both the new format data files, and the data files in the format for EosFit v5.2. If there is more than one set of fit-able data (e.g. if the file contains cell parameters as well as unit-cell volumes) then the user can select each in turn for fitting. The remainder of the program works only with one selected dataset at a time. In addition, if

sufficient cell parameter data is provided, the lattice repeat in any direction [UVW] or any plane spacing $d(hkl)$ can be selected and fit with a linear EoS. This allows the principal axes of strain in monoclinic and triclinic crystals to be fit directly with an EoS. For fitting, individual data can now be omitted from the least-squares fitting, which obviates the previous need to edit and re-read data files. The data can be converted to normalised pressure and finite strain for output to external plotting programs. Sets of EoS parameters can be saved and subsequently read back into the program from a specific 'eos' file format, which also stores the variance-covariance matrix if it is available. This allows uncertainties to be given for calculated pressure and volumes, without the need to refit the original data. The other important change from EosFit v5.2 is that, as described above in more detail, the definition of thermal expansion coefficients and moduli for linear EoS have been changed to

Tab. 1: Public subroutines and functions in the `cfml_eos` module.

| Data structure manipulation | Purpose |
|--|--|
| subroutine <code>allocate_eos_data_list(n, e)</code> | Allocates memory to a data list |
| subroutine <code>allocate_eos_list(n, e)</code> | Allocates memory to a eos list |
| subroutine <code>deallocate_eos_data_list(e)</code> | De-allocates memory for a data list |
| subroutine <code>deallocate_eos_list(e)</code> | De-allocates memory for a eos list |
| subroutine <code>init_eos_thermal(eospar)</code> | Initialises the thermal part of an eos |
| subroutine <code>init_eos_type(eospar,clin,ithermal)</code> | Initialises an eos |
| subroutine <code>init_eos_data_type(eospar)</code> | Initialises an eos data type |
| subroutine <code>init_err_eos()</code> | Initialises/resets the error system |
| Data manipulation | |
| subroutine <code>read_eos_datafile(fname,dat)</code> | Reads in data from a datafile |
| subroutine <code>ffcal_dat(v,v0,p,eospar,f,s)</code> | Calculates finite stress and strain from the data |
| subroutine <code>ffcal_dat_esd(v,sigv,v0,sigv0,p,sigp,eospar,f,sigf,s,sigs)</code> | Calculates finite stress and strain and their esd's from the data and its esd's |
| Eos initialisation | |
| subroutine <code>set_eos_names(eospar)</code> | Sets the names of variables |
| subroutine <code>set_eos_use(eospar)</code> | Sets flags to show which variables are used in the chosen types of eos and thermal expansion models |
| subroutine <code>set_kp_kpp_cond(eospar)</code> | Sets implied values of eos parameters |
| Eos properties | Returned values |
| subroutine <code>deriv_partial_p(v,t,eospar,td)</code> | Partial derivatives of P with respect to eos parameters |
| subroutine <code>eos_cal(p,t,eospar,parvals)</code> | $V_{PT}, K_{PT}, K'_{PT}, K''_{PT}, \partial K_{PT}/\partial T, \alpha_{PT}$ at the input P and T |
| subroutine <code>eos_cal_esd(p,t,eospar,esd)</code> | Esd's of the values of $V_{PT}, K_{PT}, K'_{PT}, K''_{PT}, \partial K_{PT}/\partial T, \alpha_{PT}$ at the input P and T |
| function <code>get_volume(p,t,eospar)</code> | V_{PT} at input P, T |
| function <code>k_cal(v,t,eospar)</code> | K_{PT} at input P, T |
| function <code>kp_cal(v,t,eospar)</code> | K'_{PT} at input P, T |
| function <code>kpp_cal(v,t,eospar)</code> | K''_{PT} at input P, T |
| function <code>dkdt_cal(p, t, eospar)</code> | $\partial K_{PT}/\partial T$ at input P, T |
| function <code>alpha_cal(p,t,eospar)</code> | α_{PT} at input P, T |
| subroutine <code>ffcal_eos(p,t,eospar,f,s)</code> | F, f at the input P, T |
| subroutine <code>write_info_eos(eospar, iout)</code> | All eos parameters to unit <code>iout</code> |
| function <code>get_pressure(v,t,eospar)</code> | P at input V, T |
| function <code>get_pressure_esd(v,t,eospar)</code> | Esd(P) at input V,T from esd's in eos parameters |
| function <code>get_volume_s(f,t,eospar)</code> | V at input finite strain and T |
| function <code>pthermal(t,eospar)</code> | Thermal pressure at T |
| function <code>pressure_f(f,s,eospar)</code> | Pressure from input f, F |
| function <code>strain(vv0,eospar)</code> | Finite strain from input V_{PT}/V_{0T} |

Tab. 2: Data structures and public variables in the `cfml_eos` module.

| Parameters | |
|---------------------------------|--|
| <code>n_eospar</code> | Max number of eos parameters |
| <code>ncol_data_max</code> | Max number of data items per line in a datafile |
| <code>err_eos</code> | .true. when error in the module |
| <code>err_eos_mess</code> | error message or information |
| Data structure types | |
| <code>eos_type</code> | All parameters for one eos |
| <code>eos_list_type</code> | A list of <code>eos_type</code> |
| <code>eos_data_type</code> | All the data for one data point (P,T,V , cell parameters and esd's) |
| <code>eos_data_list_type</code> | A list of <code>eos_data_type</code> to contain a whole dataset |

make them consistent with the definitions and conventions of linear elasticity theory.

EoS fitting

The EosFit7c program implements all of the recommendations with respect to fitting EoS parameters to P - V (or P -linear) data of Angel (2000a), in exactly the same way as did EosFit v5.2. We note the key points here. Because the relative experimental uncertainties in P are usually significantly greater than those in V , all fits are performed by least-squares minimization of the difference in observed and calculated *pressures*, and not volumes. The

choice of weighting scheme is left to the user, but if weights are derived from volume and/or temperature uncertainties, then the “effective variance method” (Orear, 1982) is used to calculate the overall weight, w , applied to a data point as:

$$\frac{1}{w} = \sigma_p^2 + \sigma_v^2 \left(\frac{K}{V}\right)^2 + \sigma_T^2 (\alpha K)^2.$$

Because the weight of an individual data point clearly depends on the parameter values of the EoS, the weights are updated in every least-squares cycle. Once the least-squares process has converged, a full set of statistical fit criteria plus the variance/covariance matrix and the correlation coefficients between parameters are provided to the user to allow critical evaluation of the results.

Availability

The CrysFML subroutine library, which includes the EoS module described in this paper, is described at http://www.ill.eu/other_sites/fullprof/php/programs.html, and can be downloaded from <https://forge.epn-campus.eu/projects/crysfml>. The EosFit7c console program to perform EoS calculations and fit EoS parameters to data is available for Windows and Linux operating systems at www.rossangel.net, together with example datasets and complete documentation.

Conclusions and future prospects

The development of the `cfml_eos` module within CrysFML provides a validated set of subroutines that perform all conceivable EoS calculations, allowing the programmer to focus on writing software to solve scientific problems without having to worry about the technicalities of complex EoS formulations. The inclusion of the module within a library of much wider use should ensure its longer-term survival and availability. The `cfml_eos` module is structured so that it could be easily extended to include further types of isothermal EoS or thermal expansion models. The authors have also developed a GUI version of EosFit7 (Gonzalez-Platas et al., in prep.) that incorporates full facilities for plotting EoS as well as the features in EosFit7c, and will continue to develop other EoS utilities based on the `cfml_eos` module in CrysFML, and to maintain and upgrade the module itself.

Acknowledgements: This software development was supported by ERC starting grant 307322 to Fabrizio Nestola of the University of Padova, and by the Ministerio Español de Ciencia e Innovación, Project MAT2010-21270-C04-02, and MALTA Consolider CSD2007-0045 to Javier Gonzalez-Platas. We thank all previous users of EosFit for their continued feedback on the program that has led to many improvements in the software and helped in its validation. We especially acknowledge Tiziana Boffa-Ballaran and Fabrizio Nestola for valuable discussions on various aspects of EoS and the development of the module and the program.

References

- K. Alden, M. Read, *Nature* **2013**, 502, 448.
- O. L. Anderson, *Equations of State of Solids for Geophysics and Ceramic Science*, Oxford University Press, Oxford, UK **1995**.
- R. J. Angel, Equations of state. In: *High-pressure and high-temperature crystal chemistry* (Eds. R. M. Hazen, R. T. Downs), Reviews in Mineralogy and Geochemistry, Vol. 41, pp. 35–60. Mineralogical Society of America and Geochemical Society, (Washington, DC) **2000a**.
- R. J. Angel, High-pressure structural phase transitions. In: *Transformation Processes in Minerals* (Eds. S. A. T. Redfern, M. A. Carpenter), Reviews in Mineralogy & Geochemistry, Vol. 39, pp. 85–104. Mineralogical Society of America and Geochemical Society, (Washington, DC) **2000b**.
- R. J. Angel, J. M. Jackson, H. J. Reichmann, S. Speziale, *European Journal of Mineralogy* **2009**, 21, 525.
- N. Barnes, *Nature* **2010**, 467, 753.
- R. G. Berman, *Journal of Petrology* **1988**, 29, 445.
- F. Birch, *Physical Review* **1947**, 71, 809.
- F. Birch, *Journal of Geophysical Research* **1986**, 91, 4949.
- I. Brown, P. Klages, A. Skowron, *Acta Crystallographica B* **2003**, 59, 439.
- T. S. Duffy, Y. B. Wang, Pressure-volume-temperature equations of state. In: *Ultrahigh-Pressure Mineralogy: Physics and Chemistry of the Earth's Deep Interior* (Ed. R. Hemley), Reviews in Mineralogy, Vol. 37, pp. 425–458. Mineralogical Society of America, (Washington, DC) **1998**.
- F. Fabbiani, C. Pulham, *Chemical Society Reviews* **2006**, 35, 932.
- Y. Fei, Thermal expansion. In: *Mineral Physics & Crystallography: A Handbook of Physical Constants* (Ed. T. Ahrens), Vol. 2, pp. 29–44. American Geophysical Union, (Washington, DC) **1995**.
- J. Freund, R. Ingalls, *Journal of Physics & Chemistry of Solids* **1989**, 50, 263.
- G. D. Gatta, R. J. Angel, N. Rotiroti, J. Zhao, M. Alvaro, M. A. Carpenter, *American Mineralogist* **2011**, 96, 1363.
- M. Gottschalk, *European Journal of Mineralogy* **1997**, 9, 175.
- E. C. Hayden, *Nature* **2013**, 501, 472.
- G. Hellfrich, J. A. D. Connolly, *American Mineralogist* **2009**, 94, 1616.
- T. J. B. Holland, R. Powell, *Journal of Metamorphic Geology* **1998**, 16, 309.
- T. J. B. Holland, R. Powell, *Journal of Metamorphic Geology* **2011**, 29, 333.

- W. B. Holzapfel, *Zeitschrift für Kristallographie* **2001**, 216, 473.
- Y. K. Huang, C. Y. Chow, *Journal of Physics D: Applied Physics* **1974**, 7, 2021.
- I. Jackson, H. Niesler, The elasticity of periclase to 3 GPa and some geophysical implications. In: *High-Pressure Research in Geophysics* (Eds. S. Akimoto, M. H. Manghnani), pp. 93–113. Center for Academic Publications, (Tokyo) **1982**.
- R. Jeanloz, *Physical Review B* **1988**, 38, 805.
- K. Knight, *Physics and Chemistry of Minerals* **2010**, 37, 529.
- H. Kroll, A. Kirfel, R. Heinemann, B. Barbier, *European Journal of Mineralogy* **2012**, 24, 935.
- M. Kumar, *Physics and Chemistry of Minerals* **2003**, 30, 556.
- F. Murnaghan, *American Journal of Mathematics* **1937**, 49, 235.
- J. F. Nye, *Physical properties of crystals*, Oxford University Press, Oxford **1957**.
- J. Orear, *American Journal of Physics* **1982**, 50, 912.
- A. R. Pawley, S. A. T. Redfern, T. J. B. Holland, *American Mineralogist* **1996**, 81, 335.
- J.-P. Poirier, A. Tarantola, *Physics of Earth and Planetary Interiors* **1998**, 109, 1.
- J. Rodriguez-Carvajal, J. Gonzalez-Platas, *IUCr Computing Commission Newsletter* **2003**, 1, 50.
- E. K. H. Salje, B. Wruck, H. Thomas, *Zeitschrift für Physik B – Condensed Matter* **1991**, 82, 399.
- H. Schlosser, J. Ferrante, *Physical Review B* **1988**, 37, 4351.
- W. Schranz, A. Tröster, J. Koppensteiner, R. Miletich, *Journal of Physics Condensed Matter* **2007**, 19, 275202.
- F. D. Stacey, B. J. Brennan, R. D. Irvine, *Geophysical Surveys* **1981**, 4, 189.
- L. Stixrude, C. Lithgow-Bertelloni, *Geophysical Journal International* **2005**, 162, 610.
- I. Suzuki, *Journal of Physics of the Earth* **1975**, 23, 145.
- M. Tribaudino, R. J. Angel, F. Camara, F. Nestola, D. Pasqual, I. Margiolaki, *Contributions to Mineralogy and Petrology* **2010**, 160, 899.
- M. Tribaudino, M. Bruno, F. Nestola, D. Pasqual, R. J. Angel, *American Mineralogist* **2011**, 96, 992.
- A. Tröster, W. Schranz, R. Miletich, *Physical Review Letters* **2002**, 88, 055503.
- P. Vinet, J. Ferrante, J. Rose, J. Smith, *Journal of Geophysical Research* **1987**, 92, 9319.
- P. Vinet, J. Ferrante, J. Smith, J. Rose, *Journal of Physics C: Condensed Matter* **1986**, 19, L467.
- D. C. Wallace, *Thermodynamics of crystals*, John Wiley and Sons, New York **1972**.
- B. Willis, A. Pryor, *Thermal vibrations in crystals*, Cambridge University Press, Cambridge **1975**.
- J. Zhao, N. L. Ross, R. J. Angel, *Acta Crystallographica B* **2004**, 60, 263.

Received November 21, 2013; accepted January 19, 2014

Published online February 24, 2014