**Local-Basis-Function Equation of State for Ice VII-X to 450 GPa at 300 K**

J. Michael Brown and Baptiste Journaux

Earth and Space Sciences

University of Washington

Seattle, WA

**Supplementary Materials – Descriptions of Five Functions Required for Numerical Analysis and the Example Scripts**

The accompanying Helmholtz toolbox contains five functions that are necessary for local basis function creation and evaluation. Accompanying scripts use these functions for analysis of data.In the provided MATLAB scripts, the analysis requires four steps: (1) load data, (2) set options, (3) fit data, (4) display results. This provides a straightforward workflow that encourages exploration of how modifications of assumptions provide differences in the resulting equation of state. It is important to try modifications of all adjustable elements in order to explore the parameter sensitivity and quality variations of fits.

The functions and scripts have been tested within both MATLAB and OCTAVE. All numerical work appears identical. Some graphing behavior differs and may produce less than optimal figures using OCTAVE that can be corrected with minor changes in the scripts.

**GETSTRAINS**: given a list of volumes, and the volume associated with zero strain, this function determines the strains based on a specified strain metric:

Strains=getStrains(V,Vo,strainflg)

Where strainflg is a character string defining the strain metric. Available strains are “Eulerian”, “log”, “Vinet”, and “comp” (V/V)

**COLLOCATE**: This function is the backbone of b-spline analysis. It is the implementation of the de Boor algorithm to determine basis function values and values of specified derivatives of basis functions at collocation sites. Given a set of *l* knots in vector *t*, a specified spline order *k*, a set of *n* independent variable points *x*, and the number of derivatives needed *drv*, the command

*B=collocate(t,k,x,drv)*

returns an array *B* of size *n* by *m* by *drv+1* with basis functions and derivatives of basis function determined at the collocation points. Based on the discussion of knots given above, the horizontal extent of the array, *m,* is determined by the relationship *m=l-k*. For *drv=0* (no derivatives requested), the array *B* is a matrix without a third dimension. For *drv=1*, the array has a depth of 2, where *B(i,j,1)* contains the representation basis functions and *B(i,j,2)* contains basis functions for the first derivative of the representation. Setting *drv* to larger integers returns, in addition, the higher order derivatives (restricted by the order(degree) of the representation).

**FN\_F\_VAL**: Given a local basis function representation (*LBF*), that is parameterized using knots specified by a strain metric, and independent variable values for density in a vector *rho*, the command

*EOS=fn\_F\_val(LBF,rho)*

Returns predictions of equation of state properties in the structure EOS where

*EOS.F* is the Helmholz energy in units determined by the

pressure-volume units used in construction of the

representation

*EOS.P* is the pressure in units determined by the input

*EOS.K* is the isothermal bulk modulus in the same units as

those for pressure

*EOS.Kp* is the isothermal first pressure derivative of the bulk

modulus (dimensionless)

**LIN\_F\_FIT**: Given data and options (both are structures) that describe the nature of the fitting

*LBF=lin\_F\_fit(data,options)*

returns a b-spline structure for the equation of state based on linear Tikhonov inversions.

The elements of input structures are:

*data.PV* contains a n x 2 matrix with column 1 containing pressures

and densities or volumes in column 2

*data.K* contains a m x 3 or m x 2 matrix where the first column is

density or volume, second column is bulk modulus and the

optional third column contains pressures for measurements

of the bulk modulus. Using a linear solution, K can only be

used if the strain (determined by either density or volume)

is known. In most cases, K is reported as a function of

pressure. If a trial linear solution is available, then the volume at

each measurement pressure can estimated and placed in

this matrix.

*data.Vo* contains the ambient pressure value of density or volume

*data.Vflg* contains a flag set to 0 if data are in density units or 1 if data are

in volumes

*options.strainflg* contains a character string that defines the strain metric

‘*comp*’ for (V-Vo)/V

‘*log*’ for log strains

‘*Eulerian*’ for Eulerian finite strains

‘*Vinet*’ for the Vinet strain metric

*options.k* specifies the spline order

*options.knt* contains a list knot values in the volume or density units

of the data. Initial and final knot multiplicity is added

within the fitting function

*Options.kntflg* is an optional input. If present, then the values

provided in options.knt are interpreted as being in the

specified strain unit rather than in the specified

volume/density unit

*options.K\_weight* is optional. If specified, then it provides a relative

weight for the bulk modulus data in simultaneous fitting

of P-V and K-V data.

*options.Reg* is optional. If specified, it contains a list of points in

volume/density where regularization is to be applied.

If scalar, then that number of regularization points are

distributed between the first and last knots.

*options.drv* specifies which derivative of the potential is to be

minimized for regularization

*options.lam* specifies the size of the regularization damping – larger

to enforce minimization of the specified potential

derivative, smaller to allow closer fitting of data.

Elements of the output structure are

*LBF.knots* specifies the knots for the spline

*LBF.coefficients*: the model parameters in energy, Units depend on

the units used for input data

*LBF.order* order of spline

*LBF.number* number of model coefficients

*LBF.dim*: dimension of spline

*LBF.time\_stamp*: record of date and time of creation

*LBF.rho0*: the density of the material at 1 bar

*LBF.strainflg* the strain metric used

*LBF.Data* the data and options structures associated with

creation of the spline

**NLIN\_F\_FIT**: Given an initial equation of state, *LBFo*, this function is called as:

*LBFn = nlin\_F\_fit(LBFo, data, options)*

where *LBFn* is additionally constrained by data for the first pressure derivative of the bulk modulus. This function implements solution of Equation (11). Since K’ is expected to monotonically approach an asymptotic value at infinite pressure that is greater than or equal to 5/2, the addition of explicit values for *K’* at very high pressure (small volumes) regularizes any fit in a regime where pressure-volume data are not available. The representation is accordingly forced to both fit data in the regime of data and to approach the expected behavior of *K*’ at higher pressures.

In addition to the previously defined input, the necessary additional input is:

*data.Kp* with first column containing density/volume points and

the second column containing Kp values

*options.Kp\_weight* contains the relative weight for the constraint on *K’*